DEPARTAMENTO DE LENGUAJES Y SISTEMAS INFORMÁTICOS E INGENIERÍA DE SOFTWARE

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INFORMÁTICOS

UNIVERSIDAD POLITÉCNICA DE MADRID

TESIS DOCTORAL

INCORPORACIÓN DE TÉCNICAS DE HCI EN UN PROCESO ÁGIL MEDIANTE PATRONES

AUTOR

LEYDI OFELIA CABALLERO CHI

DIRECTOR

DRA. ANA MARÍA MORENO SÁNCHEZ-CAPUCHINO

MADRID 2017

A

J. Manuel B. R.

A

Plácido, Ara y Oscar

Agradecimientos

A Dios por permitirme coincidir en los momentos adecuados de esta vida con las personas indicadas, guiarme durante el camino e iluminarme para continuar hasta lograr esta importante meta.

En particular y de manera muy especial a mi tutora la Dra. Ana M. Moreno Sánchez Capuchino por su enorme paciencia, tiempo, apoyo, ánimo, consejos y correcciones que son invaluable para mi aprendizaje y crecimiento académico, profesional e incluso personal. Siempre agradecida contigo Ana.

A mi esposo Manuel por su amor, confianza, respeto, motivación, paciencia y apoyo incondicional durante todo el desarrollo de este proyecto.

A mis padres Plácido y Aracelly, a mi hermano Oscar. Por su cariño, ánimo y apoyo en este reto y toda la vida.

A mis amigos por su soporte durante estos años ayudándome a la distancia en varios sentidos para enfocarme en culminar este trabajo. En particular David L., agradecimiento eterno.

A la L.C.C. Carmen Zozaya y L.C.C. Verónica Castillo por su confianza para trabajar una parte fundamental de este estudio con sus respectivos equipos de desarrollo de software resultando valiosas contribuciones para esta investigación. Agradezco al Departamento de innovación e implementación de sistemas del Consejo de la Judicatura y TSJYUC de Yucatán, México.

Gracias a todas y cada una de aquellas personas tanto en México como en España que en alguna medida han contribuido con su ayuda para la realización y culminación de este trabajo.

¡Muchas Gracias!

Resumen

La usabilidad como atributo de calidad del software ha sido ampliamente reconocida y estudiada en las últimas décadas. Una vasta literatura muestra los beneficios de considerar este factor como parte del proceso de desarrollo de software asegurando como resultado un producto usable y útil para los usuarios en contextos específicos. En el área de la Ingeniería de Software la disciplina Interacción Humano-Computador (en inglés Human-Computer Interaction ó HCI) remarca la importancia de la usabilidad como factor que contribuye a un buen diseño de interacción para el usuario. Esta filosofía proporciona técnicas y herramientas que al aplicarlas durante el proceso de desarrollo permiten obtener un adecuado nivel de usabilidad en el software.

La comunidad HCI aboga desde el surgimiento del manifiesto ágil por el reconocimiento de la importancia e inclusión de un buen nivel de usabilidad en los desarrollos ágiles de software. Los esfuerzos enfocados a esta situación han sido importantes y los resultados sobre la incorporación de técnicas HCI durante el proceso de desarrollo ágil muestran la viabilidad de la integración de estas dos disciplinas. Uno de los principales factores que influyen en la decisión de incorporar técnicas HCI en el entorno de desarrollo ágil es su posible impacto en la duración de las iteraciones. Por otro lado, una de las razones más señaladas por la cual los equipos ágiles que no cuentan con especialistas HCI deciden no usar dichas técnicas durante el proceso del desarrollo es la falta de conocimiento y formación sobre el uso las mismas.

Este trabajo aborda la necesidad de información para los desarrolladores ágiles proporcionando guías que sirvan de apoyo y les permitan aplicar técnicas HCI en sus proyectos software aún sin ser o tener especialistas HCI en sus equipos. Las guías son descritas como patrones con el objetivo de que los equipos puedan utilizar las soluciones propuestas en diferentes contextos del proceso ágil. Se aportan dos patrones para aplicar dos de las técnicas HCI más utilizadas: persona y usability testing. Una vez definida la estructura y contenido de los patrones éstos fueron validados en dos contextos: académico e industrial. La validación en un contexto académico permitió verificar que el formato elegido para describir el patrón proporcionaba información clara y fácil de entender para los miembros del equipo que lo utilizaran. Por otro lado la validación en la industria con proyectos reales elaborados siguiendo la metodología SCRUM permitió verificar la eficiencia de los patrones como guías durante el proceso de desarrollo para la definición de perfiles de usuario descritos como personas y aplicar las pruebas de usabilidad.

Los resultados de esta investigación contribuyen a la integración de las disciplinas HCI y Ágil dentro del marco de la Ingeniería de Software incorporando de una manera sencilla las técnicas HCI en el proceso de desarrollo ágil con la ayuda de patrones.

Abstract

Usability as an attribute of software quality has been widely recognized and studied in recent decades. The vast literature shows the benefits of considering this factor as part of the software development process, ensuring a usable and useful product for users when applied it in specific contexts. In software engineering, the discipline Human-Computer Interaction (HCI) emphasizes the importance of usability as a factor contributing to good user interaction design. This philosophy provides techniques and tools that, applying it through the development process, allow reaching an adequate level of usability in the software.

Since agile manifesto starting to rise, HCI community advocates the importance recognition and inclusion of a considerable usability level on agile software development. The focused efforts related to this issue have been relevant and results show the feasibility of integration of these disciplines. One of the main factors that influences the decision to apply HCI techniques and tools to reach a good usability level in the agile context is the potential impact on the iterations duration. On the other hand, literature shows that agile teams without HCI specialists usually decide not to include HCI techniques in software development due to lack of knowledge and training.

This work addresses the need of information for agile developers by providing guidelines that will support and allow them to apply HCI techniques in software projects even without HCI training or HCI experts as team members. The guidelines are described as patterns in order for the teams to use the proposed solutions in different contexts of the agile process. We have developed patterns to apply two of the mostly used HCI techniques: persona and usability testing. Once the structure and content of the patterns were defined, these were validated in two contexts: academic and industrial. The validation in an academic context verified that the format used to describe the pattern provided the information to the development team in a clear and easy way. On the other hand, the validation in industry with real projects developed using SCRUM methodology verified the efficiency of the patterns as guidelines for developers to define and describe their users' profiles as personas and to apply the usability tests through the development process.

The results of this work contribute to the integration of HCI and agile disciplines within software engineering incorporating HCI techniques into agile development process with patterns as support guidelines.

Índice

Capítulo 1	Introducción	22
1.1	Áreas de investigación	24
1.2	Usabilidad de software	24
1.2.1	Usabilidad y Experiencia de Usuario	25
1.2.2	Interacción persona-ordenador	26
1.2.3	Diseño Centrado en el Usuario.....	27
1.3	Desarrollo Ágil.....	29
1.3.1	Filosofía.....	30
1.3.2	Metodologías ágiles.....	31
1.4	Integración de Usabilidad en un Desarrollo Ágil.....	34
1.5	Planteamiento del Trabajo de Investigación	35
1.5.1	Problema de Investigación	35
1.5.2	Objetivo de la Tesis.....	36
1.5.3	Estrategia de la investigación	36
1.6	Estructura de la Tesis	37
Capítulo 2	Estado del Arte	39
2.1	Introducción	41
2.2	Revisión de la literatura.....	41
2.2.1	Preguntas de investigación	41
2.2.2	Proceso de búsqueda	42
2.2.3	Selección de estudios y extracción de datos.....	42
2.2.4	Resultados	43
2.3	Prácticas HCI en Desarrollo ágil de Software (Qué).....	46
2.4	Momentos de Integración HCI en el proceso ágil (Cuándo).....	52
2.5	Organización del equipo ágil de desarrollo al aplicar técnicas HCI (Quién)	59
2.6	Resumen de la revisión de literatura	63
CAPÍTULO 3	ENFOQUE DE LA SOLUCIÓN	65
3.1	Introducción	67
3.2	Patrones	67
3.3	Estrategia de solución.....	75
3.3.1	Adaptaciones en el uso de las técnicas HCI dentro del entorno ágil.....	75
3.3.2	Técnicas HCI: Persona y Usability Testing	76

3.3.3	Patrones de producto	77
Capítulo 4	Formalización de la solución: Patrón “Agile Persona”	80
4.1	Introducción	82
4.2	Persona	82
4.2.1	La idea original: Personas como una herramienta de segmentación de mercado	82
4.2.2	Persona en HCI.....	83
4.2.3	Persona en el Desarrollo Ágil de Software	87
4.2.4	Adaptaciones al incorporar la técnica persona en el proceso ágil	91
4.3	Patrón “Agile persona”	92
4.3.1	Definiendo el patrón “Agile persona”	92
4.3.2	Adaptaciones consideradas en el patrón “Agile persona”	101
4.3.3	Estructura del Patrón “Agile persona”	104
Capítulo 5	Formalización de la solución: Patrón “Agile Usability Testing”	109
5.1	Introducción	111
5.1.1	Usability Testing in HCI	111
5.1.2	Usability Testing en el Desarrollo Ágil.....	113
5.1.3	Adaptaciones al incorporar la técnica usability testing en el proceso ágil	118
5.2	Patrón “Agile Usability Testing”	119
5.2.1	Definiendo el patrón “Agile Usability Testing”	119
5.2.2	Adaptaciones consideradas en el patrón “Agile Usability Testing”	133
5.2.3	Estructura del Patrón “Agile Usability Testing”	134
Capítulo 6	Evaluación de los patrones propuestos.....	140
6.1	Introducción	142
6.2	Estrategia de evaluación.....	142
6.3	Evaluación Académica	145
6.3.1	Contexto y participantes.....	145
6.3.2	Proceso de evaluación	146
6.3.3	Proyecto Piloto	146
6.3.4	Proyectos Docentes	155
6.3.5	Discusión de resultados en evaluación académica	162
6.4	Evaluación en proyectos reales de desarrollo.....	163
6.4.1	Diseño del Estudio	164
6.4.2	Preparación para la Recolección de Datos	167
6.4.3	Recolección de Datos	169
6.4.4	Análisis de datos.....	176
6.4.5	Discusión resultados de evaluación en proyectos reales	216

6.5	Conclusiones sobre la evaluación.....	223
Capítulo 7	Conclusiones y futuras líneas de investigación.....	226
7.1	Conclusiones	228
7.2	Líneas futuras de investigación.....	229
Referencias		231
Anexos		242
Anexo A.	Lista de estudios primarios de la Revisión de Literatura.....	244
Anexo B.	Proyecto Piloto PEAP	257
Anexo C.	Proyecto de Evaluación Académica PEAA	265
Anexo D.	Patrones evaluados.....	274
Anexo E.	Pruebas piloto a la estructura de la entrevista	289
Anexo F.	Instrumentos utilizados: Entrevista y Encuesta.....	293
Anexo G.	Proyecto real de Evaluación en Ambiente Universitario PRAU	297
Anexo H.	Proyecto real de Evaluación en Industria PREI.....	310

Índice de Figuras

Figura C1-1 Ciclo de vida del diseño centrado en el usuario de acuerdo a ISO 13407	29
Figura C1-2 Visión general del proceso de desarrollo ágil (Rubin, 2012).....	32
Figura C1-3 Fases del proyecto de investigación.....	37
Figura C2-1 Estudios analizados por año para el período 2001 – primera mitad 2013 (Caballero et al., 2016)	44
Figura C2-2 Artículos analizados por año para el período segunda mitad 2013 - primer trimestre de 2017	44
Figura C2-3 Metodologías ágiles utilizadas de acuerdo al análisis de publicaciones del período 2001 – primera mitad 2013 (Caballero et al. 2016).....	46
Figura C2-4 Metodologías ágiles utilizadas en los proyectos de los estudios analizados para el período 2001 – primer trim. 2017.....	46
Figura C2-5. Clasificación de las técnicas HCI identificadas de acuerdo a cuatro dimensiones (Brhel et al., 2015).....	47
Figura C2-6. Número de artículos relacionados con la dimensión "prácticas" (Brhel et al., 2015)	48
Figura C2-7 Técnicas o artefactos HCI usados en Agile. Período de análisis 2001 – primera mitad 2013 (Caballero et al., 2016)	49
Figura C2-8 Técnicas o artefactos HCI usados en Agile. Período de análisis 2001 – primer trimestre 2017	49
Figura C2-9 Integración de instrumentos HCI en XP (Zahid Hussain et al., 2008a).....	53
Figura C2-10 Proceso de desarrollo integrando UCD en desarrollo ágil (Sy, 2007)	54
Figura C2-11 Modelo de integración entre SCRUM y el Diseño UX (Najafi & Toyoshiba, 2008)	55
Figura C2-12 Enfoque de integración “Interaction Design First”	56
Figura C2-13 Enfoque de integración “Minimal Up-Front Design”	56
Figura C2-14 Enfoque “Implicit Interaction Design”	57
Figura C2-15 Enfoque “Iteration Ahead” – Iteración inicial	57
Figura C2-16 Enfoque “Iteration Ahead” – Iteración posterior	57
Figura C2-17 Enfoque de integración HCI utilizado en proyectos ASD. Período de análisis 2001 – primera mitad de 2013 (Caballero et al. 2016).....	58
Figura C2-18 Enfoque de integración HCI en un proyecto de desarrollo ágil. Período de análisis 2001 – primer trimestre 2017	58
Figura C2-19 Prácticas HCI utilizadas por enfoque de integración en proyectos ASD. Período de análisis 2001 – primera mitad 2013 (Caballero et al. 2016)	58
Figura C2-20 Prácticas HCI utilizadas por enfoque de integración en proyectos ASD. Período de análisis 2001 – primer trimestre 2017	59
Figura C2-21 Organización de los equipos ágiles trabajando HCI en el proceso de desarrollo. Período de análisis 2001 – primera mitad de 2013 (Caballero et al. 2016)	61
Figura C2-22 Organización de los equipos ágiles trabajando HCI en el proceso de desarrollo. Período de análisis 2001 – primer trimestre de 2017	61
Figura C3-1 Ejemplo patrón de Diseño (Gamma et al., 1994).....	69
Figura C3-2 Ejemplo de Patrón de Análisis (Fowler, 1996).....	69
Figura C3-3 Ejemplo de un patrón de arquitectura (Buschmann et al., 1996).....	70
Figura C3-4 Ejemplo patrón de diseño de interfaz (Tidwell, 2005).....	71
Figura C3-5 Ejemplo de un patrón de usabilidad ágil (Bertholdo et al., 2014)	73

Figura C3-6 Ejemplo de patrón para AUCDI (Salah et al., 2015)	73
Figura C3-7 Patrón de diseño para "Go Back" (L Carvajal et al., 2013)	74
Figura C3-8 Artefacto High-Level Design Component Responsibilities ejemplo "Warning" ...	74
Figura C3-9 Ejemplo de un Patrón de producto ("Promise - PPL," 2009)	78
Figura C3-10 Formato de la descripción del patrón de producto	79
Figura C4-1 Plantillas de documento base (Pruitt et al., 2010)	87
Figura C4-2 Descripción de persona con una ilustración y corta narrativa	87
Figura C4-3 Percepción de persona en Mercadotecnia, HCI y Agile	88
Figura C4-4 Etapas para la creación de persona durante el proceso Mobile-D (Haikara, 2007)	89
Figura C4-5 Proceso XP adaptado con la integración de técnicas HCI (Zahid Hussain et al., 2008a)	90
Figura C4-6 Modelo de integración entre SCRUM y el diseño UX (Najafi & Toyoshiba, 2008)	90
Figura C4-7 Estructura final del patrón "Agile Persona"	92
Figura C4-8 Cómo aplicar persona en un entorno ágil	94
Figura C4-9 Quién está involucrado en el proceso y cómo se involucra	94
Figura C4-10 Cuando se aplica persona en un proceso ágil	97
Figura C4-11 Cuando aplicar persona durante la planeación de la iteración	98
Figura C4-12 Cuando aplicar persona durante el desarrollo de la iteración	98
Figura C4-13 Descripción ejemplo de persona	100
Figura C4-14 Plantilla para la descripción de persona	101
Figura C5-1 Estructura final del patrón "Agile Usability Testing"	120
Figura C5-2 Cómo es aplicada la técnica de pruebas de usabilidad en un proceso ágil	125
Figura C5-3 Cuando es aplicado el patrón "Agile Usability Testing"	126
Figura C5-4 Momentos en los que puede aplicarse pruebas de usabilidad durante la iteración	127
Figura C5-5 Aplicando pruebas de usabilidad con el enfoque "Interaction Design First"	128
Figura C5-6 Aplicando pruebas de usabilidad con el enfoque "Minimal Up Front Design"	129
Figura C5-7 Aplicando pruebas de usabilidad con el enfoque "Implicit Interaction Design" ..	130
Figura C5-8 Aplicando pruebas de usabilidad con el enfoque "Parallel Track"	130
Figura C5-9 Plantilla de la lista de revisión para pruebas de usabilidad	132
Figura C6-1 Etapas de evaluación de los patrones definidos	144
Figura C6-2 Diagrama solución del patrón "Agile persona" versión PAPv0	149
Figura C6-3 Diagrama solución del patrón "Agile persona" versión PAPv1	150
Figura C6-4 Diagrama solución modificado de acuerdo a observaciones finales del equipo PEAP. Patrón "Agile Persona" versión PAPv2	154
Figura C6-5 Cómo aplicar la técnica persona en un entorno ágil (proceso "Apply persona") .	155
Figura C6-6 Diagrama solución del patrón "Agile persona" versión PAPv0	157
Figura C6-7 Contextos, casos y unidades de análisis	164
Figura C6-8 Encuesta Facilidad/Dificultad de utilizar los patrones	169
Figura C6-9 Distribución de la evaluación a la Facilidad/Dificultad de utilizar los patrones...	177
Figura C6-10 Distribución resultado de la evaluación a la Facilidad/Dificultad de utilizar los patrones. Grupo PRAU	177
Figura C6-11 Distribución resultado de la evaluación a la Facilidad/Dificultad de utilizar los patrones. Grupo PREI	177
Figura C6-12 Distribución global de los resultados a la evaluación de los desarrolladores a la pregunta P.A1.1	178
Figura C6-13 Distribuciones de evaluaciones a la pregunta P.A1.1 de cada uno de los grupos de evaluación	178

Figura C6-14 Distribución global de los resultados a la evaluación de los desarrolladores a la pregunta P.A1.2.....	179
Figura C6-15 Distribuciones de evaluación a la pregunta P.A1.2 de cada uno de los grupos de evaluación	180
Figura C6-16 Resultados globales de la evaluación sobre la efectividad del patrón “Agile Persona” como guía para implementar la técnica de pruebas de usabilidad durante el desarrollo del proyecto	187
Figura C6-17 Resultados globales de la evaluación sobre la efectividad del patrón “Agile Usability Testing” como guía para implementar la técnica de pruebas de usabilidad durante el desarrollo del proyecto.....	187
Figura C6-18 Distribución global de los porcentajes obtenidos como resultado a la evaluación de los desarrolladores a la pregunta P.A2.1 correspondiente al aspecto de Efectividad de los productos patrón	190
Figura C6-19 Resultados de evaluación a la pregunta P.A2.1 de cada uno de los grupos de evaluación	190
Figura C6-20 Distribución global de los porcentajes obtenidos como resultado a la evaluación de los desarrolladores a la pregunta P.A2.2	191
Figura C6-21 Resultados de evaluación a la pregunta P.A2.2 de cada uno de los grupos de evaluación	191
Figura C6-22 Distribución global de los porcentajes obtenidos como resultado a la evaluación de los desarrolladores a la pregunta P.A2.3	193
Figura C6-23 Resultados de evaluación a la pregunta P.A2.3 de cada uno de los grupos de evaluación	193
Figura C6-24 Distribución global de los porcentajes obtenidos como resultado a la evaluación de los desarrolladores a la pregunta P.A2.4	194
Figura C6-25 Resultados de evaluación a la pregunta P.A2.4 de cada uno de los grupos de evaluación	195
Figura C6-26 Distribución global de los porcentajes obtenidos como resultado a la evaluación de los desarrolladores a la pregunta P.A2.5	196
Figura C6-27 Resultados de evaluación a la pregunta P.A2.5 de cada uno de los grupos de evaluación	196
Figura C6-28 Distribución global de los porcentajes obtenidos como resultado de la evaluación a la pregunta P.A2.6.....	198
Figura C6-29 Resultados de evaluación a la pregunta P.A2.6 de cada uno de los grupos de evaluación	198
Figura C6-30 Evaluación global sobre los resultados percibidos por los desarrolladores al aplicar las técnicas de usabilidad	200
Figura C6-31 Evaluación global sobre los resultados percibidos por los desarrolladores al aplicar las técnicas de usabilidad	201
Figura C6-32 Distribución global de las valoraciones a la pregunta P.A3.1	203
Figura C6-33 Resultados de evaluación a la pregunta P.A3.1 por grupo de evaluación.....	203
Figura C6-34 Distribución global de las valoraciones a la pregunta P.A3.1	204
Figura C6-35 Distribución valoraciones a la pregunta P.A3.2 por grupo de evaluación	204
Figura C6-36 Distribución global de las valoraciones a la pregunta P.A3.3	206
Figura C6-37 Distribución valoraciones a la pregunta P.A3.3 por grupo de evaluación	206
Figura C6-38 Distribución global de las valoraciones a la pregunta P.A3.4	209
Figura C6-39 Distribución valoraciones a la pregunta P.A3.4 por grupo de evaluación	209

Figura C6-40 Distribución global de los porcentajes obtenidos como resultado de la evaluación a la pregunta P.A3.5	212
Figura C6-41 Distribución de valoraciones por grupo de evaluación a la pregunta P.A3.5	212
Figura C6-42 Distribución global de los porcentajes obtenidos como resultado de la evaluación a la pregunta P.A3.6	214
Figura C6-43 Distribución de valoraciones por grupo de evaluación a la pregunta P.A3.6	215

Índice de Tablas

Tabla C1–1 Resumen de técnicas HCI clasificadas de acuerdo a la etapa del desarrollo de software (www.usabilitynet.org)	27
Tabla C1–2 Resumen de Metodologías Ágiles más populares	32
Tabla C2–1 Extracto de la información recabada de los estudios analizados	45
Tabla C3–1 Resumen de la estructura de patrón utilizada por Alexander C. (C Alexander et al., 1977)	68
Tabla C3–2 Patrones de usabilidad ágil para las primeras etapas del ciclo UCD propuestos en (Bertholdo et al., 2014)	72
Tabla C3–3 Patrones de usabilidad ágil para las etapas finales del ciclo UCD propuestos en (Bertholdo et al., 2016)	72
Tabla C3–4 Resumen de características técnicas HCI más utilizadas en proyectos ágiles	77
Tabla C4–1 Características de la técnica persona en Marketing y HCI	83
Tabla C4–2 Pasos y Fases del ciclo de vida para construir persona	84
Tabla C4–3 Extracto del análisis sobre el uso de la técnica Persona en proyectos ágiles	89
Tabla C4–4 Descripción del patrón "Agile Persona" (Parte 1 de 3)	93
Tabla C4–5 Relación del patrón persona con los pasos de Cooper y las fases del ciclo de vida de Pruitt	102
Tabla C5–1 Características de las pruebas de usabilidad	112
Tabla C5–2 Extracto del análisis sobre el uso de Pruebas de usabilidad en proyectos ágiles... ..	115
Tabla C5–3 Patrón "Agile Usability Testing" (<i>Parte 1 de 5</i>)	120
Tabla C6–1 Grupos participantes en la revisión de los productos patrón	144
Tabla C6–2 Equipos participantes en la evaluación académica de los patrones	145
Tabla C6–3 Fuentes de retroalimentación durante la evaluación Académica	146
Tabla C6–4 Actividades realizadas en el PEAP como parte del proceso de evaluación del patrón "Agile persona"	147
Tabla C6–5 Tareas del sprint planning y el sprint backlog correspondiente al sprint 1 equipo PEAP	151
Tabla C6–6 Observaciones del equipo del PEAP durante el proceso del desarrollo de software	152
Tabla C6–7 Situaciones relevantes identificadas a partir del reporte final del equipo y la forma en la que abordaron la situación	153
Tabla C6–8 Descripción de los proyectos en el grupo de evaluación PEAA	156
Tabla C6–9 Observaciones relacionadas con el uso del patrón "Agile persona" por el grupo PEAA	158
Tabla C6–10 Extracto del calendario de actividades del proyecto PEAA3	160
Tabla C6–11 Formato utilizado por cada equipo para definir las personas identificadas	161
Tabla C6–12 Descripción de los proyectos en el grupo de evaluación PREI	166
Tabla C6–13 Juegos desarrollados por los equipos	170
Tabla C6–14 Participación de cada desarrollador en el grupo de evaluación PEAU	171
Tabla C6–15 Actividades durante la evaluación en el proyecto de software educativo	171
Tabla C6–16 Actividades realizadas durante los sprints	172
Tabla C6–17 Registro de las entrevistas realizadas a los equipos del proyecto PRAU	172
Tabla C6–18 Seguimiento a las respuestas de la encuesta en línea para el proyecto PRAU	173
Tabla C6–19 Actividades durante la evaluación de los productos patrón en la Industria	173
Tabla C6–20 Proyectos de Software desarrollados en la Industria (Unidades de Análisis)	173
Tabla C6–21 Proyectos en los que participó cada desarrollador	174

Tabla C6–22 Visión general de los sprints y elementos de cada proyecto	175
Tabla C6–23. Condensado de los totales de las validaciones aplicadas en la encuesta a P.A1.1 y P.A1.2.....	176
Tabla C6–24 Aspectos positivos patrón “Agile Persona”	180
Tabla C6–25 Aspectos negativos patrón “Agile Persona”	181
Tabla C6–26 Aspectos positivos patrón “Agile Usability Testing”	183
Tabla C6–27 Aspectos negativos producto patrón “Agile Usability Testing”	185
Tabla C6–28 Totales condensados preguntas aspecto A2.....	186
Tabla C6–29 Totales condensados preguntas aspecto A2 grupo PRAU.....	188
Tabla C6–30 Totales condensados preguntas aspecto A2 grupo PREI.....	189
Tabla C6–31 Totales condensados preguntas aspecto A3.....	199
Tabla C6–32 Condensado de las frecuencias totales de las preguntas del aspecto A3 grupo PREI	202
Tabla C6–33 Condensado de las frecuencias totales de las preguntas del aspecto A3 grupo PREI	202
Tabla C6–34 Porcentajes globales para las preguntas P.A1.1 y P.A1.2	217
Tabla C6–35 Porcentajes globales para las preguntas P.A1.1 y P.A1.2 por grupo de evaluación	217
Tabla C6–36 Distribución de porcentajes globales para las preguntas P.A2.1 y P.A2.4	218
Tabla C6–37 Distribución de porcentajes globales para las preguntas P.A2.2 y P.A2.5	218
Tabla C6–38 Distribución de porcentajes globales para las preguntas P.A2.3 y P.A2.6	219
Tabla C6–39 Porcentajes globales para las preguntas del aspecto A2 por grupo de evaluación	220
Tabla C6–40 Porcentajes globales para las preguntas del aspecto A3	221
Tabla C6–41 Distribución de valoraciones para los cuestionamientos del aspecto A3 por grupo de evaluación	222

Capítulo 1 Introducción

1.1 Áreas de investigación

Esta tesis se enmarca dentro del campo de la Ingeniería de Software (IS) y más concretamente, se centra en el proceso ágil de desarrollo de software. Dentro de esta área, la investigación realizada aborda la incorporación de técnicas de usabilidad durante las distintas etapas de dicho proceso ágil.

Desde el punto de vista del desarrollo de software la usabilidad es un atributo de calidad (ISO/IEC 25010, 2011) (Sommerville, 2011). Este atributo representa una ventaja competitiva proporcionando beneficios en diferentes aspectos. Entre ellos, desde la perspectiva del usuario se reduce la curva de aprendizaje y se le facilita un uso más eficiente del sistema software (Dumas et al., 1999). Desde el punto de vista empresarial, la usabilidad conduce a un incremento de la productividad en el desarrollo así como disminución de costos en formación y soporte (Dumas et al., 1999) (Bias et al., 2005) (Barnum, 2010) (J. Nielsen et al., 2012). De cara al desarrollo, la usabilidad proporciona ahorros en rediseño y mantenimiento, lo que repercute en una disminución en la necesidad de realizar cambios y liberar actualizaciones (Bias et al., 2005) (Barnum, 2010) (J. Nielsen et al., 2012).

Los diferentes modelos de proceso de software que describen las actividades que se realizan y los productos que se obtienen durante el desarrollo, se basan generalmente en los paradigmas de cascada, desarrollo iterativo e ingeniería de software basada en componentes. Una de las propuestas con mayor relevancia surgidas en los últimos años en relación al proceso de desarrollo del software son las metodologías ágiles (Sharp et al., 2015). El desarrollo ágil constituye un marco de trabajo que aborda una gestión del proyecto que permite mejor respuesta al cambio y control en los tiempos de entrega del software (Sommerville, 2011). Los métodos ágiles tienen características propias (Cockburn, 2007) pero alineados de acuerdo a una filosofía de valores y principios comunes (Beck et al., 2000). Desarrollar de manera iterativa e incremental con una participación activa del usuario y entregas de un producto valioso para él en períodos de 1 a 4 semanas son características que hacen atractiva la propuesta de construir software siguiendo la disciplina ágil (Sommerville, 2011).

Respecto a la incorporación de técnicas de usabilidad en un proceso ágil, aunque los primeros trabajos datan de hace una década, no se puede considerar hoy por hoy un área madura. Existen distintas aproximaciones en la literatura; en algunos casos se han definido marcos de trabajo p.e. (Lee J et al., 2007) (Singh, 2008) (Obendorf et al., 2008) (Da Silva et al., 2011); en otros se han propuesto adaptaciones de algunas técnicas de usabilidad para su uso en un contexto ágil (Sy, 2007) (Zahid Hussain et al., 2008a) (Jakob N, 2008). Sin embargo, tal como se discute más adelante, actualmente la comunidad adolece de pautas detalladas que los desarrolladores puedan utilizar para abordar este proceso de integración. Esta limitación es sobre la que se centra el presente trabajo de investigación, generando como resultado del mismo, guías concretas que permitan a los desarrolladores ágiles incorporar ciertas técnicas de usabilidad en un proceso ágil.

En las secciones siguientes de este capítulo se revisan los conceptos fundamentales sobre usabilidad (sección 1.2) y desarrollo ágil (sección 1.3) que definen el marco contextual de la investigación. Seguidamente se aborda la problemática de la integración de ambas áreas (sección 1.4) y se esboza el problema de investigación en el que se trabajará (sección 1.5). Finalmente se presenta la estructura de la tesis (sección 1.6).

1.2 Usabilidad de software

Durante las últimas décadas, la usabilidad de software ha sido un área de estudio relevante (Goodwin, 1987) (Jakob N, 1993) (ISO 9241-11, 1998) (ISO/IEC 25010, 2011) (J. Nielsen et al., 2012) (Wale-Kolade et al., 2016). Durante este tiempo han emergido una variedad de conceptos relacionados con el término “Usabilidad”.

1.2.1 Usabilidad y Experiencia de Usuario

Dentro de las definiciones más reconocidas de usabilidad está la establecida por Nielsen quien la describe como:

“Usabilidad es un atributo de calidad que evalúa qué tan fácil son las interfaces de usuario para utilizar” (Jakob Nielsen, 1993)

Otras definiciones muy conocidas y referenciadas son las establecidas por estándares internacionales como el ISO/IEC 9241 sobre ergonomía e interacción humano-computador, parte 11 (guía de usabilidad) en donde usabilidad se define como:

“Medida en la que un producto puede ser utilizado por usuarios específicos para alcanzar metas específicas con efectividad, eficiencia y satisfacción en un contexto especificado de uso” (ISO 9241-11, 1998)

Basándose en el concepto de usabilidad del estándar ISO 9241-11 mencionado anteriormente, el estándar ISO 9241 parte 210 que corresponde a diseño centrado en el humano para sistemas interactivos extiende el concepto de usabilidad más allá de los productos y describe este atributo como:

“Medida en la que un sistema, producto o servicio puede ser utilizado por usuarios específicos para alcanzar metas específicas con efectividad, eficiencia y satisfacción en un contexto específico de uso” (ISO 9241-210, 2010)

El estándar ISO/IEC 25010:2011 es parte del conjunto de estándares denominado SQuaRE (Systems and Software Quality Requirements and Evaluation) y sustituye al estándar ISO 9126-1:2001. Este estándar define un modelo de calidad en uso y un modelo de la calidad del producto presentando la definición de usabilidad, también derivada del estándar ISO 9241, como se menciona a continuación:

“Grado en el cual un producto o sistema puede ser utilizado por usuarios específicos para alcanzar metas específicas con efectividad, eficiencia y satisfacción en un contexto específico de uso” (ISO/IEC 25010, 2011)

Por su parte el estándar IEEE 1061 define usabilidad como:

“Usabilidad es la facilidad con la que un usuario puede aprender a operar, preparar entradas e interpretar salidas de un sistema o componente” (IEEE, 1998).

Independientemente del contexto en el que se defina, en esencia se reconoce la usabilidad como un factor o atributo de calidad que es deseable en un producto y que lo hace fácil de utilizar (Carvajal L., 2012). Al incorporar este atributo durante el proceso de desarrollo se asegura un software que permita ser utilizado de manera fácil e intuitiva con un desempeño eficiente y eficaz generando de esta forma una agradable experiencia para el usuario.

Sin embargo, desde hace más de una década ya se identificaron varios obstáculos al incorporar el atributo usabilidad en el desarrollo de software (Seffah A, et al., 2004). A pesar del tiempo transcurrido desde la publicación del trabajo de Seffah, aún en nuestros días se perciben dichos obstáculos. Entre ellos la falta de unicidad en la definición de usabilidad generando confusión a los desarrolladores y la falta de preparación por parte de los desarrolladores en el área de usabilidad de software. Con el fin de evitar que la usabilidad se considere un proceso aparte del ciclo de desarrollo de software Seffah et al. (2004) apuntan que es importante tener en cuenta que el diseño de la interacción no necesariamente se desarrolle por un experto HCI o diseñador. Señalan que proporcionar formación a los desarrolladores para que puedan desempeñarse como expertos de usabilidad dentro del equipo ayuda a cerrar la brecha entre las prácticas de IS y usabilidad. Indican también que recopilar y compartir las buenas prácticas permite a los desarrolladores adquirir más conocimiento a partir de experiencias previas del mismo equipo o equipos de desarrollo. En particular para proyectos de software pequeños sugieren subsanar la falta de un especialista

en usabilidad utilizando guías, patrones de usabilidad, heurísticas y diagramas de flujo de trabajo para el análisis de requerimientos funcionales. Por su parte Juristo et al. en (Juristo N. et al, 2007) indican que la falta de una definición estándar de usabilidad también es un obstáculo para que los desarrolladores construyan software usable. (Dubey et al., 2010) señalan los diferentes puntos de vista sobre los atributos de usabilidad y la falta de una definición auténtica de este término como las razones de un pobre nivel de usabilidad en los sistemas de software.

Otro factor que se identifica como un obstáculo para incorporar un nivel de usabilidad en los desarrollos de software es el limitado presupuesto asignado por las empresas para realizar las actividades que implica el aplicar las técnicas necesarias de usabilidad (Hao et al., 2011)

(Scheiber et al., 2012) señalan el riesgo de las pequeñas y medianas empresas de software de quedar relegadas frente a empresas de software más grandes o internacionales como consecuencia de la falta de conocimiento en usabilidad. También mencionan el impacto negativo que tiene un bajo nivel de usabilidad en el uso del software por parte de los clientes. (Bornoe et al., 2014) encontraron durante su investigación que los profesionales del área de usabilidad requerían adquirir más experiencia sobre cómo aplicar de manera efectiva diferentes métodos de usabilidad. Por su parte (Ogunyemi et al., 2016) identifican dentro de los factores que limitan la adopción de prácticas de usabilidad en el desarrollo de software la falta de experiencia por parte de los profesionales para aplicarlas y el conocimiento implícito de estos sobre el tema.

Las cuestiones de actitud por parte de los desarrolladores también se han identificado como obstáculos para la integración de la usabilidad en la IS (Wale-Kolade et al., 2016).

Uno de los enfoques muy relacionado con la usabilidad de software surgido más recientemente es el de Experiencia de Usuario (en inglés User Experience o UX). Aunque generalmente se utilizan indistintamente existe una diferencia entre éstos. Norman D. explica el término “experiencia de usuario” como la forma en la que experimentamos el mundo, la vida, los servicios en la vida cotidiana incluidos los sistemas de computación (Norman D., 2016). Una definición más específica indica que la UX abarca todos los aspectos de la interacción del usuario final con la empresa, sus servicios y sus productos (Norman & Nielsen, 2014). El estándar ISO 9241-11:1998 define Experiencia de usuario como *“las percepciones y respuestas de la persona como resultado del uso y/o uso anticipado de un sistema, producto o servicio”* (ISO 9241-11, 1998). En general la UX hace referencia a la percepción del usuario utilizando en nuestro caso el producto software mientras que la usabilidad es, como se ha indicado antes, un atributo del producto software que contribuye a proporcionar una mejor UX.

1.2.2 Interacción persona-ordenador

El campo de Interacción persona-ordenador (en inglés Human-Computer Interaction o HCI) aporta respaldo teórico y propone técnicas para incorporar/mejorar el atributo usabilidad en un sistema de software. HCI se enfoca en el estudio y profundización de las diferentes maneras en las que el humano (usuario) interactúa con el computador a través de la interfaz del software.

“HCI es la disciplina relacionada con el diseño, evaluación e implementación de sistemas de computación interactivos para uso humano y con el estudio de los principales fenómenos que los rodean” (SIGCHI, n.d.).

Una de las principales preocupaciones de la comunidad HCI es la escasa importancia que aún se le da a la usabilidad como atributo de calidad en el software. En esta línea para mejorar la percepción de la usabilidad como factor clave en la IS se han realizado propuestas y recomendaciones en el uso de técnicas HCI y su integración en el proceso de desarrollo de software. En (Juristo et al., 2007) los autores proponen categorizar las recomendaciones existentes en la literatura HCI en tres grupos: recomendaciones de usabilidad con impacto en la interfaz de usuario, recomendaciones de usabilidad con impacto en el proceso de desarrollo y recomendaciones de usabilidad con impacto en el diseño.

De la amplia variedad de técnicas HCI, en la tabla C1-1 se presenta un resumen de éstas clasificadas de acuerdo a la etapa del desarrollo de software en la que se utilizan. Para esta clasificación consideramos las etapas de requerimientos, diseño, implementación y evaluación.

Tabla C1-1 Resumen de técnicas HCI clasificadas de acuerdo a la etapa del desarrollo de software (www.usabilitynet.org)

Requerimientos	Diseño	Implementación	Evaluación
Contextual Inquiry (Beyer et al., 1998)	Paper prototype (Sharp et al., 2007)	Rapid prototyping (Medlock et al., n.d.)	Heuristic Evaluation (Jakob Nielsen, 1993)
Scenarios (Carroll, 2000)	Heuristic Evaluation (Jakob Nielsen, 1993)	Style Guides	Usability Testing (Dumas et al., 1999) (Jeffrey et al., 2008)
Persona (Cooper et al., 2007)	Parallel Design (Sy, 2007)		Diagnostic Evaluation
Focus Group (Lazar et al., 2010) (Sharp et al., 2007)	Storyboarding (Sharp et al., 2007)		
Ethnography (Lazar et al., 2010)	Wizard of Oz (Sharp et al., 2007)		
Surveys (Lazar et al., 2010) (Barnum, 2010)	Interface Design Patterns (Tidwell, 2005)		
Interviews (Lazar et al., 2010)s			
Observation (Lazar et al., 2010)			
Brain Storming (Lazar et al., 2010)			
Card Sorting (Lazar et al., 2010) (Barnum, 2010)			
Affinity Diagramming (Beyer et al., 1998)			

En ocasiones los equipos de desarrollo no cuentan con expertos en usabilidad o en HCI y este rol es desempeñado por desarrolladores. Sin embargo, la falta de conocimiento de éstos sobre usabilidad dificulta la aplicación de las técnicas HCI. Para minimizar el impacto de esta situación en los últimos años se ha dirigido también el esfuerzo a proporcionar artefactos de apoyo a los desarrolladores de software en el tema de HCI. Una de las técnicas que más se ha utilizado por la comunidad HCI para comunicar las recomendaciones y permitir la reutilización son los patrones. El concepto de patrón presentado por Christopher Alexander (Christopher Alexander, 1979) ha sido ampliamente usado en la comunidad de software para reutilizar conocimiento. De manera análoga se ha usado en HCI.

En (Seffah et al., 2009) los patrones se describen como una herramienta de diseño de la interacción para capturar buenas prácticas que permitan hacer frente a problemas que ocurren en diferentes contextos. Seffah et al (2009) refieren como ejemplos de patrones aquellos definidos para la visualización y presentación de información. También mencionan colecciones de patrones para el diseño de interacción. Dentro de los patrones mencionados en (Sanchez-Segura et al., 2010) como los más relevantes en la IS se encuentran los patrones de arquitectura, patrones de análisis, patrones de diseño, patrones de proceso, patrones de mejora y patrones organizacionales. Por otro lado, también se han propuesto líneas guía estructuradas a partir de patrones que sirvan de apoyo a los desarrolladores para incorporar usabilidad en diferentes niveles del proceso de desarrollo (Juristo et al., 2007) (Carvajal, 2009) (Carvajal L. et al., 2013). En el capítulo 3 de este trabajo se profundiza en el uso de patrones en el campo de HCI.

1.2.3 Diseño Centrado en el Usuario

Por su alcance, HCI es también un área de investigación interdisciplinaria en donde han surgido diferentes metodologías y filosofías cada una dependiendo del contexto de interacción entre el usuario y

el sistema. Una de las más relevantes es el Diseño Centrado en el Usuario (en inglés User Centered Design o UCD). Esta aproximación surge originalmente en 1985, y considera como elemento primordial al usuario final, por lo que promueve su participación en todo el proceso de desarrollo. Los seguidores de esta disciplina comulgan con el hecho de que para obtener un sistema realmente fácil de entender y utilizar el diseño debe estar enfocado por completo al usuario final. Por lo tanto, UCD busca proveer un enfoque general para el diseño de sistemas interactivos haciendo a los usuarios finales y sus experiencias el punto focal del proceso de diseño (Seffah et al., 2009).

En el estándar ISO 9241-210 se utiliza en la descripción del enfoque UCD el término “Diseño centrado en el humano” (en inglés Human Centered Design o HCD) en lugar de “Diseño Centrado en el usuario”. El estándar anota que aplica esa diferencia de nombre para enfatizar que la norma también toma en cuenta el impacto en todas las personas involucradas (stakeholders) y no solamente a aquellos típicamente considerados como usuarios. En general estos dos términos frecuentemente se utilizan como sinónimos.

El estándar define HCD como:

“Diseño centrado en el humano es un enfoque para el desarrollo de sistemas interactivos que tiene por objetivo hacer sistemas usables y útiles enfocándose en los usuarios, sus necesidades y requerimientos y aplicando factores humanos/ergonómicos y conocimiento de usabilidad y técnicas” (ISO 9241-210, 2010).

Los principios que en nuestros días han sido aceptados como la base para el enfoque centrado en el usuario fueron establecidos en 1985 por Gould & Lewis (Gould et al., 1985). Los tres principios definidos para ser aplicados en etapas tempranas durante el desarrollo de un sistema software se describen brevemente a continuación:

- Enfoque temprano en los usuarios y tareas. Antes de elaborar el diseño inicial del sistema los diseñadores deben entender a los potenciales usuarios desde la perspectiva del comportamiento de éstos y de la naturaleza del desarrollo que se espera llevar a cabo para ellos.
- Medición empírica. De manera temprana en el proceso de desarrollo deben observarse, registrarse y analizarse el desempeño y reacciones de los posibles usuarios al utilizar simulaciones y prototipos.
- Diseño iterativo. Conforme se van identificando problemas durante las pruebas con el usuario éstos deben repararse.

En resumen, los principios enfatizan un temprano y continuo enfoque en el usuario midiendo empíricamente los resultados observados sobre la interacción de éste con el sistema, rediseñando y probando de nuevo en base a dicha retroalimentación, todo ello de forma iterativa.

El ciclo de vida UCD definido originalmente en el estándar ISO 13407 (reemplazado por el estándar ISO 9241:210) considera dentro de las fases iterativas el entender y especificar el contexto de uso, especificar los requerimientos del usuario (y de la organización), producir soluciones de diseño y evaluar los diseños. A partir de estas etapas se obtiene un sistema que satisface los requerimientos del usuario y organizacionales. En la figura C1-1 se representan las fases del proceso de diseño centrado en el usuario propuesto en dicho estándar.

Figura C1-1 Ciclo de vida del diseño centrado en el usuario de acuerdo a ISO 13407

Gulliksen et al (Gulliksen et al., 2003) hacen notoria la falta de una definición única para el concepto de UCD. Ellos proponen, basados, entre otras referencias, en los principios de Gould et al., (1985), doce principios que aporten en la unicidad del concepto del diseño centrado en el usuario; término que ellos refieren como el proceso de Diseño de Sistemas Centrado en el usuario (en inglés User-Centred Systems Design o UCSD). Gulliksen et. al. (Gulliksen et al., 2003) atribuyen como obstáculo para aplicar UCD en la IS la falta de conocimiento sobre cómo aplicar los métodos de usabilidad y los potenciales beneficios que esto representa. Mencionan que hay una creciente preocupación de los desarrolladores de software sobre la usabilidad de sus productos, sin embargo frecuentemente no saben cómo mejorar dicho atributo de calidad.

(Eriksson et al., 2009) destacan el impacto que tiene en la práctica la participación de los desarrolladores en la aplicación de técnicas UCD, experiencia que les permite apreciar mejor los beneficios de la usabilidad en el desarrollo de software. Por ello (Eriksson et al., 2009) señalan la importancia de enfocar esfuerzos de investigación en obtener resultados que propongan métodos y herramientas que puedan ser utilizados por los profesionales en el área, incluidos los desarrolladores.

La administración del ciclo de diseño generado por una ambigua, poco clara y evolutiva experiencia de usuario es señalado en (Moreno et al., 2013) como un desafío importante para los desarrolladores en la integración UCD dentro de un proceso de desarrollo de software. (Moreno et al., 2013) apuntan que la comunicación entre las disciplinas UCD e IS requiere mejorarse para un mejor entendimiento entre estas dos comunidades sobre las limitaciones que cada una debe cuidar en la aplicación de las técnicas de usabilidad durante un proceso de desarrollo de software.

El estudio de (Ogunyemi et al., 2016) sobre la adopción de técnicas de usabilidad por empresas de desarrollo de software pequeñas y medianas en un país en desarrollo mostró dentro de las limitaciones para integrar el enfoque HCD en los desarrollos de software el tiempo y presupuesto estimado para los proyectos. También el desconocimiento sobre los estándares definidos y conceptualizaciones relacionadas con el tema de usabilidad genera una falta de valor e interés para las empresas de software sobre la importancia de utilizar técnicas HCD en sus desarrollos. En base a los resultados obtenidos de su investigación (Ogunyemi et al., 2016) proponen un camino como opción que permita incorporar HCD en los desarrollos de software.

1.3 Desarrollo Ágil

Una de las filosofías en desarrollo de software que ha crecido en popularidad durante los últimos años dentro de la IS son las nuevas metodologías de desarrollo de software, el llamado movimiento ágil cuyo

auge adquirido en nuestros días es bien conocido. Los resultados exitosos que se han observado en diferentes aspectos tanto administrativos como técnicos al incorporar los diferentes marcos de trabajo que promulgan el “agilísimo” han propiciado que los equipos de desarrollo adopten cada vez más esta disciplina (Ferreira et al., 2007b) (Barksdale et al., 2012) (Bertholdo et al., 2014) (Sharp et al., 2015). Remarcada por valores y principios que destacan la continua y constante entrega de software así como una mejor relación con el usuario y mayor participación de éste dentro del proceso, la filosofía ágil ha propiciado cambios en paradigmas de equipos de desarrollo y en compañías enteras.

1.3.1 Filosofía

El peso de una metodología o “methodology weight” es el producto del número de elementos de control que ésta considera (p.e. estándares, actividades, medidas de calidad) y el grado de precisión con el que deben seguirse dichos controles (Cockburn, 2007). Las metodologías ligeras o “Lighthweight methodologies” son aquellas en las que los elementos de control aplicados durante el desarrollo de software solamente son los necesarios. De esta forma, avanza la construcción del proyecto de software de acuerdo a las características que aportan valor al cliente evitando controles excesivos en el proceso de desarrollo. La filosofía ágil toma forma oficialmente durante la reunión realizada en 2001 donde participaron expertos investigadores y profesionales para compartir sus ideas, puntos de vista y experiencias. El objetivo era identificar coincidencias prácticas entre metodologías ligeras que permitieran a los equipos mejorar su proceso de desarrollo de software principalmente y, entre otras cosas, para trabajar rápidamente y responder al cambio. De esta reunión y en común acuerdo de los participantes surge el término “ágil” y el manifiesto que establece las bases de esta disciplina a través de cuatro valores y los doce principios derivados de éstos (www.agilemanifesto.org). Los autores hacen hincapié en el manifiesto sobre la importancia que tienen todos los elementos considerados en la descripción de los valores. Sin embargo enfatizan aquellos elementos que para el enfoque ágil son más importantes estableciendo los valores de la siguiente forma:

Individuos e interacciones sobre procesos y herramientas. Este valor resalta la importancia de las personas que conforman el equipo como el principal factor de éxito de un proyecto. Más allá de contar con un proceso efectivo y herramientas muy sofisticadas, sin miembros en el equipo con una buena interacción que se sientan identificados y comprometidos con dicho equipo resultará igual de complicado sacar adelante el desarrollo del software. Martín R. et al (Martin, 2003) aconseja que en lugar de construir primero un entorno y esperar a que el equipo automáticamente se adapte, mejor se trabaje en crear el equipo y luego dejar al equipo configurar el contexto de desarrollo según su necesidad. Por su parte, Cockburn (Cockburn, 2007) comenta que es preferible usar un proceso no documentado con buenas interacciones entre el equipo, más que un proceso documentado con una interacción hostil.

Software funcional sobre documentación exhaustiva. Generar solamente la documentación necesaria es lo que expresa este valor; el software funcional es el que mejor expresa lo que se ha alcanzado por parte del equipo. En relación a la transferencia de conocimiento a nuevos elementos del equipo, el código y el equipo son la mejor manera de transferir información (Martin, 2003).

Colaboración del cliente sobre negociación de contratos. La continua colaboración con el cliente genera la retroalimentación que da pie a la negociación correcta abarcando aquellos logros y entregas reales más que aquellos que se consideran en los contratos que se pactan antes de que un desarrollo inicie.

Respuesta al cambio sobre seguir un plan. La planificación que se realice debe ser preferiblemente a corto plazo sobre las tareas que están próximas a realizarse en el desarrollo. Planificar más allá de lo que está por realizarse en el proyecto puede impactar de manera negativa en la rapidez y efectividad de respuesta al presentarse alguna situación durante el proceso de desarrollo que implique realizar cambios no esperados.

Los valores descritos previamente dan pie a los doce principios alrededor de los que se centra la filosofía:

Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.

Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.

Entregamos software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.

Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.

Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.

El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.

El software funcionando es la medida principal de progreso.

Los procesos Ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.

La atención continua a la excelencia técnica y al buen diseño mejora la Agilidad.

La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.

Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.

A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para la continuación ajustar y perfeccionar su comportamiento en consecuencia.

A partir de estos principios se han definido métodos de desarrollo de software que satisfacen estas recomendaciones en distinta medida y que se les conoce con el nombre de metodologías, métodos o técnicas ágiles.

1.3.2 Metodologías ágiles

Una vez establecido y publicado el manifiesto ágil, los resultados de utilizar las metodologías que comulgan con esta filosofía han demostrado que, en conjunto o en sus enfoques individuales, representan una poderosa herramienta en el proceso del desarrollo de software. Se utilizan de acuerdo a las características del proyecto siempre tomando en cuenta que el objetivo es entregar aplicaciones que generen valor al cliente. Las metodologías “ligeras” descritas previamente, adheridas al manifiesto ágil, son actualmente parte de estos métodos ágiles p.e. SCRUM, XP, FDD, Crystal (Cockburn, 2007). El esquema de la figura C1-2 sobre el enfoque de desarrollo ágil que presenta Rubin (Rubin, 2012) resulta muy útil para tener un panorama genérico del proceso de desarrollo ágil e identificar los elementos comunes que están presentes en él.

En inicio se crea una pila de producto (product backlog), término que se utiliza para hacer referencia a la lista priorizada de características o funcionalidades (features) que se requieren en el producto y otras capacidades necesarias para desarrollarlo. Los equipos auto-organizados y multifuncionales van trabajando en iteraciones cortas delimitadas generalmente por un tiempo de entre una semana a un mes. Durante dichas iteraciones se van desarrollando los elementos de la lista de acuerdo a la priorización más alta verificada al inicio de cada iteración durante la planificación de ésta (iteration planning). Al final de cada iteración el equipo revisa (iteration review) junto con todas aquellas personas involucradas con el proyecto, incluyendo el dueño del producto, las funcionalidades finalizadas. La retroalimentación

obtenida de esta revisión proporciona información para definir el trabajo de la siguiente iteración y planear como realizarlo. En cuanto una iteración termina se inicia todo el proceso de nuevo con la planificación de la siguiente iteración.

Figura C1-2 Visión general del proceso de desarrollo ágil (Rubin, 2012)

Las metodologías que se consideran ágiles tienen características muy particulares, el objetivo de cada una puede ser diferente e incluso pueden tener sus propios valores y principios (Beck et al., 2004). Algunas han obtenido mayor presencia debido a la aplicabilidad que los equipos de desarrollo han identificado en ellas. En la tabla C1-2 se observan aquellas metodologías ágiles que en la literatura se identifican como las más utilizadas por los equipos de desarrollo (Cockburn, 2007)(T. Da Silva et al., 2011)(Salvador et al., 2014)(VersionOne, 2017). El nombre de la metodología se encuentra en la columna “Metodología”. En la columna “Naturaleza de la metodología” se identifican con la palabra “Agilismo” aquellas metodologías surgidas originalmente de la corriente ágil, y con “Adaptación” aquellas surgidas a partir de adaptaciones al agilismo de metodologías existentes. Por último se presenta una breve descripción de la metodología.

Tabla C1-2 Resumen de Metodologías Ágiles más populares

Metodología	Naturaleza de la metodología	Descripción
SCRUM (Schwaber et al., 2002)	Ágil	Marco de trabajo para un desarrollo iterativo e incremental de las funcionalidades requeridas por el usuario. Los equipos de desarrollo son capaces de auto organizarse y auto gestionar sus tareas a través del proceso de desarrollo del software
XP, eXtreme Programming (Beck et al., 2004)	Ágil	Metodología de desarrollo pone más énfasis en la adaptabilidad que en la previsibilidad. Los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. Adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos.

Crystal (Cockburn, 2007)	Ágil	Familia de metodologías centradas en la gente y la comunicación. La filosofía de Crystal es que el desarrollo de software es visto como un juego cooperativo de invención y comunicación, con la meta primaria de entregar software funcional y útil. La meta secundaria es establecer el siguiente partido.
FDD, Feature Driven Development (Coad et al., 1999)	Ágil	Proceso ágil diseñado y probado para entregar resultados funcionales de manera frecuente, tangible y constantemente. Es iterativo, enfatiza la calidad en cada etapa. Provee información exacta sobre el progreso del desarrollo
DSDM, Dynamic Systems Development Method	Adaptación	Provee un marco de trabajo de controles y mejores prácticas para el desarrollo rápido de aplicaciones (en inglés Rapid Application Development o RAD)
AUP, Agile Unified Process (S. W. Ambler, 2006)	Adaptación	Versión simplificada del proceso racional unificado (en inglés Rational Unified Process o RUP). Describe un enfoque simple y fácil de entender para desarrollar aplicaciones de software de negocios aplicando por un lado técnicas ágiles y por otra parte conceptos relacionados con RUP.
DAD, Disciplined Agile Delivery (Ambler, 2012)	Adaptación	Es un marco de trabajo de proceso. Un enfoque híbrido entre scrum con estrategias probadas de modelado ágil, xp y el proceso unificado. Extiende el ciclo de vida de Scrum enfocado a la construcción para conducir el ciclo de vida completo desde el inicio del proyecto hasta la entrega de la solución a los usuarios finales.
SCRUMBAN (Ladas, 2009)	Adaptación	Marco de trabajo en el que se hace hincapié en la aplicación de sistemas Kanban dentro de un contexto de scrum y estratificar el proceso kanban junto scrum como un vehículo de cambio evolutivo. Se trata de ayudar y amplificar las capacidades inherentes en Scrum.

Aunque originalmente los lineamientos de la disciplina ágil se establecieron pensando en la ingeniería del software y en particular en los equipos de desarrollo (Martin, 2003)(Cockburn, 2007), estos lineamientos han sido adoptados en diferentes ámbitos para el proceso de generación de productos o servicios. El crecimiento en importancia del enfoque ágil despertó el interés para aplicarlo en proyectos diferentes a software y ha sido utilizado en áreas de negocios, periodismo, leyes e ingeniería civil y construcción de proyectos tales como una casa o un aeropuerto (Cockburn, 2007). El documento llamado “Declaración de interdependencia” (en inglés Declaration of Interdependence o DOI) fue publicado en un esfuerzo por extender los beneficios de la filosofía a proyectos no relacionados con software (Anderson et al., 2005).

El DOI, que también considera la administración del producto, fue desarrollado como un seguimiento al manifiesto ágil por investigadores y profesionales incluyendo algunos autores de éste (Cockburn, 2007).

1.4 Integración de Usabilidad en un Desarrollo Ágil

Diversos profesionales e investigadores destacan las ventajas que implica el incorporar técnicas y/o métodos de usabilidad en un entorno ágil. Patton (Patton, 2002) afirma que las técnicas de diseño de interacción propiamente aplicadas no interfieren con los valores del desarrollo ágil de software, apunta que estas técnicas incrementan la comunicación entre el cliente y desarrollador. Jokela y Abrahamsson (Jokela et al., 2004) manifiestan que incluso una colaboración cercana entre los desarrolladores ágiles y los stakeholders no garantiza una buena usabilidad. Ambler (S. Ambler, 2008) alega que una buena usabilidad final puede asegurarse solamente por actividades sistemáticas de diseño centrado en el usuario durante el desarrollo de las iteraciones.

Dado el énfasis de considerar al usuario como el “centro” de todo el proceso del diseño de interacción del software, UCD se visualiza como una de las líneas de interacción persona-ordenador para la integración de la usabilidad en un entorno de desarrollo ágil. Más que las diferencias entre UCD y Ágil destacan las similitudes entre los dos procesos que permiten visualizar porqué dicha integración es posible.

Por una parte, los principios de UCD (Gulliksen et al., 2003) consideran:

- Al usuario final como punto medular en el proceso de diseño
- Involucramiento activo del usuario durante el proceso de desarrollo
- Un desarrollo iterativo e incremental del software

Por otro lado, en el manifiesto ágil (Beck et al., 2000) se indica:

- Los métodos ágiles promueven que su mayor prioridad es satisfacer a los clientes mediante la entrega temprana y continua de software con valor.
- La gente de negocios y los desarrolladores trabajen juntos de forma cotidiana durante todo el proyecto.
- Los métodos ágiles aceptan los cambios de requerimientos y entregan software funcional frecuentemente.

La literatura muestra resultados sobre los beneficios de la incorporación de técnicas HCI en el desarrollo de proyectos ágiles de software y el impacto que esto ha tenido (T. Da Silva et al., 2011)(Jurca et al., 2014)(Salah et al., 2014) (Salvador et al., 2014)(Brhel et al., 2015)(Magües et al., 2016a).

Da Silva et al (T. Da Silva et al., 2011) identifican aquellas técnicas HCI que son más utilizadas en desarrollos ágiles y la manera en la que los equipos las incorporan en su ambiente de desarrollo ágil. A partir de los descubrimientos encontrados en su estudio proponen un marco de trabajo para la integración de estos artefactos durante el proceso ágil. En (Salvador et al., 2014) los autores identifican el momento en el que los desarrolladores ágiles incorporan las técnicas a su proceso de desarrollo. Los autores establecen inicialmente tres posibles etapas para poder realizar su análisis: requerimientos, diseño e implementación. Los resultados de su investigación muestran que las técnicas de usabilidad son incorporadas por los equipos de desarrollo principalmente durante la fase de implementación con un 50% de los estudios analizados. Un 40% reportó incorporar técnicas HCI durante la fase de diseño y un 22% indicó que utilizan las técnicas durante la fase de requerimientos. El porcentaje de estudios que reportan incorporar las técnicas durante la fase de implementación refleja que se considera la usabilidad en las etapas tardías o finales del desarrollo de software. En un esfuerzo por identificar y establecer principios que sirvan para lograr los mejores resultados al combinar HCI y el desarrollo ágil, Brhel et al. (Brhel et al., 2015) realizaron un análisis de la literatura existente sobre el tema. En su estudio consideraron las dimensiones de integración propuestas para estas dos disciplinas en (Barksdale et al., 2012): integración

en el proceso, integración de prácticas, integración de personal, integración social e integración de tecnología. (Magües et al., 2016b) basan su estudio en la clasificación utilizada por (Brhel et al., 2015).

Para un mejor entendimiento de los factores que impactan en la integración de usabilidad dentro del proceso de desarrollo ágil y la forma en la que lo hacen, en la literatura se aportan diferentes tipos de estudio. Por una parte, se presentan trabajos que permiten tener una comprensiva visión general sobre limitantes y áreas de oportunidad en este campo (Fox D et al., 2008) (Salah D et al., 2009) (Sohaib et al., 2010). También, se proponen marcos de trabajo con métodos y prácticas para la integración de HCI y el desarrollo ágil de software (en inglés Agile Software Development o ASD) (Constantine, 2002) (Zahid Hussain et al., 2008a) (Sy, 2007) (J C Lee, et al., 2009) (T. Da Silva et al., 2011) (Barksdale J et al., 2012). Otros estudios se enfocan a la incorporación de usabilidad en alguna metodología ágil en particular (Jia Y et al., 2012) y también se proponen principios que permitan dar dirección a una integración efectiva (Brhel et al., 2015). En el capítulo siguiente se presenta un estudio detallado de estos trabajos.

A pesar de la amplia gama de resultados publicados sobre los beneficios al utilizar técnicas HCI durante todo el proceso de desarrollo de software todavía se percibe su falta de aplicación por los equipos ágiles debido entre otras razones al desconocimiento sobre su uso (Brown, 2013) (Salah et al., 2014). Diferentes factores influyen en la decisión de incorporar las técnicas y herramientas de usabilidad en el entorno de desarrollo ágil siendo los principales la inversión en tiempo y por ende el impacto económico que esto implica (Sohaib et al., 2010)(Sharp et al., 2015)(Magües et al., 2016)(Laubheimer et al., 2017).

1.5 Planteamiento del Trabajo de Investigación

Las metodologías ágiles representan una innovación exitosa en la IS. El usuario puede palpar constante y continuamente resultados valiosos durante el proceso de construcción del software. Sin embargo, resulta fundamental que el resultado final sea funcional y fácil de usar.

Con entregas continuas y un timeboxing definido el equipo requiere una alta dedicación al desarrollo de las iteraciones. Esto implica limitar al máximo actividades que puedan “interferir” o retrasar las tareas necesarias para el desarrollo del producto software correspondiente. La incorporación de técnicas HCI implica incrementar tiempo de desarrollo por lo que el uso de éstas debe realizarse de manera ordenada sin ocasionar sobrecarga en el entorno ágil. Esto incluye no sólo la posible adaptación de las técnicas HCI sino también un estudio detallado del momento en que dichas técnicas han de incorporarse dentro del proceso de ágil.

La organización del equipo ágil al incorporar las técnicas HCI y el perfil de sus miembros son también características que se han visto afectadas en pro de integrar un mayor nivel de usabilidad en los desarrollos ágiles. Originariamente, la disciplina HCI establecía que las técnicas y métodos derivados de esta filosofía únicamente debían ser aplicados por los expertos en la materia (Dumas et al., 1999)(Seffah et al., 2004)(Mammel et al., 2007) (Ritter et al., 2014). Conforme la IS ha ido evolucionando, esta postura se ha flexibilizado al identificar la necesidad de que personal no experto en HCI aplique determinadas técnicas durante el desarrollo.

1.5.1 Problema de Investigación

Más allá de la cuestión “tiempo” uno de los factores que se identifica en la literatura HCI como un posible obstáculo para la incorporación de un buen nivel de usabilidad en el software es la falta de conocimiento de los desarrolladores sobre el tema. Este factor se observa en proyectos de software desarrollados con metodologías tradicionales o ágiles. (Dayton et al., 2009) señala, dentro de sus recomendaciones para agilizar los métodos de usabilidad, que se debe educar al equipo de desarrollo sobre cómo incorporar procesos y técnicas HCI. En particular el autor menciona proporcionar formación al equipo para crear historias de usuario, persona y pruebas de usabilidad iterativas.

Actualmente se cuenta con suficiente evidencia que respalda el hecho de que la integración de filosofías HCI-Ágil es posible y, al realizarse minimiza con éxito la falta de un buen nivel de usabilidad en el

software desarrollado dentro del contexto ágil. Procesos como UCD comparten importantes similitudes con el entorno ágil. Por ejemplo, procesos de desarrollo iterativos y énfasis en la participación del usuario durante todo el proceso de desarrollo (Chamberlain et al., 2006) (Fox et al., 2008) (Salah et al., 2009). Esta similitud ha permitido identificar factores para alcanzar y potenciar los beneficios de una integración entre estas dos disciplinas.

Tal como se detallará en el capítulo del Estado del Arte, hasta el momento los esfuerzos han estado dirigidos principalmente a verificar y confirmar que es posible integrar herramientas HCI dentro del proceso de desarrollo ágil. Se han identificado, entre otras cuestiones relevantes, cuáles son las metodologías ágiles en las que más han sido utilizadas técnicas de usabilidad y cuáles son las técnicas HCI más utilizadas dentro de un entorno ágil. Además, se ha estudiado la forma en la que éstas son aplicadas en dicho contexto. También se han definido diferentes procesos que pueden ser utilizados como marcos de trabajo para lograr esta combinación de métodos ágiles y HCI. Aunque son aportaciones significativas no resuelven por completo el problema de falta de usabilidad en proyectos ágiles de software. Tampoco resulta evidente y trivial identificar la mejor manera de tratar la usabilidad al integrar las filosofías ágil y HCI durante el proceso de desarrollo de software ya que dicha integración involucra diversos aspectos de cada una.

Al momento de la revisión de la literatura que se expone en el capítulo 2, *no se identificaron propuestas orientadas a proporcionar una guía que sirva de apoyo específicamente para incorporar técnicas de usabilidad en un entorno ágil reutilizando buenas prácticas*. Debido a lo anterior el estudio y definición de guías que proporcionen a los equipos ágiles la información necesaria para incorporar técnicas que aporten usabilidad en sus desarrollos se presenta como un área abierta de investigación.

1.5.2 Objetivo de la Tesis

Aunado al obstáculo que representa la falta de conocimiento de los desarrolladores sobre usabilidad, se percibe la necesidad de contar con guías que permitan utilizar técnicas HCI en el software sin afectar el dinamismo del proceso ágil. El planteamiento de Kane (Kane, 2003), expresado hace más de una década, que destaca la necesidad de patrones de interfaz de usuario para proporcionar una manera útil a los desarrolladores ágiles para considerar factores de usabilidad fue pionero en este sentido. Sin embargo, más recientemente autores como (Sohaib et al., 2010) señalan que es complicado para las organizaciones software mantener prácticas HCI cuando adoptan el desarrollo ágil debido a la falta de tácticas y prácticas en las que apoyarse. Los autores indican que los métodos empleados en las metodologías ágiles no son suficientes para la perspectiva de usabilidad.

Como aportación a la solución del problema planteado previamente, este trabajo tiene por objetivo:

Proponer patrones que sirvan a los equipos ágiles como guías de apoyo para incorporar técnicas de usabilidad en diferentes momentos de su proceso de desarrollo.

Concretamente, este trabajo se centra en la incorporación en el entorno ágil de dos técnicas HCI: persona y pruebas de usabilidad.

1.5.3 Estrategia de la investigación

Nuestra investigación se desarrolló a través de las diferentes fases que se muestran en la figura C1-3 y que describimos a continuación. Inicialmente se desarrolla una **revisión de literatura** con el objetivo de identificar aquellas técnicas HCI más utilizadas en desarrollos de software realizados siguiendo una metodología ágil. También se identifican las adaptaciones que éstas han tenido para poder ser utilizadas así como las buenas prácticas al integrarlas en dicho ambiente de desarrollo. Como hemos mencionado previamente, las técnicas pueden utilizarse en diferentes momentos, algunas en particular durante la etapa exploratoria del proyecto buscando obtener un panorama general del proyecto, mientras que otras se utilizan principalmente en la etapa de refinamiento para mejorar la interacción. Posteriormente se procede a **identificar los patrones** de integración de estas técnicas en el entorno ágil, proceso que implica analizar

las buenas prácticas reportadas en base a quién, cómo y cuándo se aplican las técnicas HCI durante este desarrollo del software.

Figura C1-3 Fases del proyecto de investigación

Con el objetivo de proporcionar un medio para reutilizar buenas prácticas y obtener resultados tangibles dentro de un contexto en particular, una vez identificados los posibles patrones de las técnicas objeto de nuestro estudio, se procede con la **descripción** de éstos. Para ello se utilizó el formato de patrón propuesto como parte del modelo arquitectónico descrito en (Sanchez-Segura et al., 2010). Para verificar que la información documentada en el patrón es la necesaria y lo suficientemente clara para los elementos del equipo de desarrollo, se aplica la **evaluación académica** sobre uno de los patrones trabajando con un proyecto piloto y una evaluación por parte de un grupo académico. La validación académica se efectuó con grupos de estudiantes para obtener retroalimentación sobre la claridad del contenido de la información reunida en los productos patrón. Una vez aplicada la evaluación académica y actualizada la información del patrón de acuerdo a lo observado en esta etapa se procede al proceso de **evaluación de los patrones en proyectos reales**. Para esto se utilizó el método de casos de estudio que resultó adecuado ya que los patrones fueron validados en proyectos reales desarrollados en un contexto universitario y en la industria.. Esta validación se llevó a cabo en un contexto de desarrollo real de proyectos ágiles para verificar la efectividad de las soluciones propuestas como guías para implementar técnicas de usabilidad en el proceso ágil. La validación se complementa con el uso de entrevistas y encuestas como artefactos de investigación que nos permitan evaluar los resultados del uso de los patrones por parte de los equipos ágiles. Una vez obtenida la información de retroalimentación se **analizan los resultados** aplicando las mejoras pertinentes en los patrones de acuerdo a las observaciones obtenidas de la retroalimentación.

1.6 Estructura de la Tesis

Para describir el proceso de definición y validación de las propuestas de solución presentadas en este trabajo el documento se estructura de la siguiente manera.

En el capítulo 2 se analiza el estado de la cuestión a través de los resultados de la revisión de la literatura realizada sobre cuestiones inherentes a la integración HCI-Ágil. Se describen y evalúan los trabajos relacionados con la integración de herramientas HCI en el proceso de desarrollo ágil, las adaptaciones y problemas a los que se han enfrentado los equipos ágiles y cómo los han solventado. Se presentan los parámetros considerados en la revisión sistemática y cómo ésta se llevó a cabo.

El capítulo 3 aborda la estrategia de solución donde se profundiza sobre las técnicas de usabilidad persona y pruebas de usabilidad consideradas en las soluciones propuestas. Se analiza cómo han contribuido a mejorar la usabilidad dentro del proceso de desarrollo ágil.

En el capítulo 4 y capítulo 5 se describen a detalle los patrones generados como solución para apoyar a los equipos ágiles a incorporar técnicas de usabilidad en su entorno de desarrollo. Las propuestas presentadas en estos capítulos incluyen mejoras derivadas de la retroalimentación obtenida en la validación.

En el capítulo 6 se describen los procesos de validación académica y validación en proyectos reales que se realizaron sobre los patrones propuestos.

En el capítulo 7 se presentan las conclusiones sobre este estudio así como las áreas de oportunidad y futuras líneas de investigación derivadas de los resultados observados.

Por último se presentan las referencias bibliográficas del presente trabajo, seguidas por la sección correspondiente a los anexos con el contenido descrito a continuación:

Anexo A. Lista de estudios primarios de la revisión de literatura

Anexo B. Proyecto piloto PEAP

Anexo C. Proyecto de evaluación académica PEAA

Anexo D. Patrones evaluados en proyectos reales

Anexo E. Pruebas piloto a la estructura de entrevista

Anexo F. Instrumentos utilizados: Entrevista y Encuesta

Anexo G. Proyecto real de Evaluación en Ambiente Universitario PRAU

Anexo H. Proyecto real de Evaluación en Industria PREI

Capítulo 2 Estado del Arte

2.1 Introducción

En este capítulo se presenta el estado de la cuestión que atañe al tema de nuestro estudio. Es importante observar de cerca cada una de las disciplinas involucradas así como sus ramificaciones para tener el panorama que nos permita visualizar la importancia de la contribución de esta investigación. Las dos filosofías, HCI y metodologías ágiles, han obtenido mayor relevancia en los últimos años y se percibe aún mucho más inclusión de éstas para futuro dentro de la IS. Por lo tanto resulta primordial contar con información suficiente sobre ellas para poder definir adecuadamente nuevas propuestas que aporten un claro avance en dar solución al problema identificado y descrito en el capítulo 1.

Primeramente en la sección 2.2 se exponen los detalles sobre la revisión de literatura realizada con el objetivo de obtener información sobre el estado del tema de nuestro estudio al momento de su desarrollo. La revisión permitió identificar trabajos previos cuyos resultados fueron relevantes para evaluar e interpretar durante la investigación. Los datos obtenidos proporcionaron una visión sobre las posibles áreas de oportunidad en la integración HCI-Agile.

Posteriormente y derivado de lo encontrado en la literatura, se describen en detalle factores identificados como determinantes al incorporar técnicas HCI en un entorno ágil. La sección 2.3 presenta los resultados sobre las prácticas HCI que son mayormente utilizadas durante un proceso de desarrollo ágil. La sección 2.4 describe los diferentes momentos en los que se observa se utilizan las técnicas HCI en el ambiente ágil y cómo éstas son aplicadas. La sección 2.5 aborda la manera en la que se organiza el equipo ágil y cómo se asignan responsabilidades en él para cubrir las tareas correspondientes al diseño de la interacción utilizando herramientas HCI.

2.2 Revisión de la literatura

Con el propósito de obtener evidencia probada que nos permitiera formular una aportación significativa sobre la incorporación de técnicas HCI en un entorno de desarrollo ágil, se realizó una revisión de trabajos publicados sobre el tema. Parte del objetivo de dicha revisión fue identificar los desafíos a los que se enfrentan los equipos de desarrollo al realizar dicha integración y cómo los han sobrellevado. Los resultados de lo observado permiten proporcionar opciones válidas derivadas de estas experiencias para facilitar a los equipos ágiles la integración HCI-Ágil.

2.2.1 Preguntas de investigación

Buscando analizar y extraer información relevante para nuestro estudio sobre la incorporación de técnicas HCI dentro del proceso de desarrollo ágil, se definieron las siguientes preguntas de investigación:

- RQ1 – En cuáles métodos ágiles son utilizadas técnicas HCI?
- RQ2 – Cuáles son las técnicas HCI más utilizadas y en qué métodos ágiles han sido incorporadas?
- RQ3 – En qué momento del proceso de desarrollo ágil son integradas las técnicas HCI?
- RQ4 – Cuál es la estructura del equipo dentro de un proyecto ágil aplicando prácticas HCI

RQ1 está orientada a obtener información sobre los métodos ágiles que han sido utilizados en combinación con técnicas HCI. Los resultados de estudios previos reflejan a Scrum y XP como las metodologías ágiles en las que mayormente se han incorporado técnicas HCI durante el desarrollo del software. Para efectos de nuestra investigación, nos enfocamos a confirmar la relevancia actual al momento de este estudio de aquellos métodos identificados como los más utilizados en combinación con técnicas y artefactos HCI. RQ2 tiene por objetivo obtener detalles sobre las técnicas HCI que han sido aplicadas en proyectos de desarrollo ágil. La

información que abarcan las preguntas RQ1 y RQ2 ha sido ampliamente estudiada y discutida en la literatura por lo tanto nuestro objetivo es comparar y analizar con más detalle cómo las técnicas HCI han sido usadas en cada método ágil identificado en respuesta a RQ1. Esto es, entender si dichas técnicas requieren alguna adaptación para ser utilizadas en el entorno ágil o pueden ser aplicadas de acuerdo al enfoque HCI original. RQ3 se ocupa del impacto de las prácticas HCI sobre los procesos ágiles; en otras palabras, esta pregunta de investigación permite analizar el grado en el cual el proceso ágil es modificado cuando adopta prácticas HCI. Finalmente, RQ4 aborda cuestiones en relación con la organización del equipo. El análisis con esta pregunta se enfoca especialmente sobre los responsables de aplicar las técnicas HCI durante el desarrollo y la colaboración de éstos con los encargados de la implementación del diseño.

2.2.2 Proceso de búsqueda

(T. Da Silva, et al., 2011) describen y presentan los resultados de la revisión sistemática de literatura sobre la integración de los enfoques UCD y métodos ágiles observada para el periodo del año 2001 hasta mediados de 2010. Para efectos de nuestra investigación, el objetivo es ampliar el rango de dicha revisión de literatura hasta el primer trimestre de 2017. Para lo anterior realizamos la búsqueda en una primera etapa a partir de mediados del año 2001 hasta mediados del año 2013. Posteriormente se realizó una segunda etapa para actualizar la información de la segunda mitad del año 2013 hasta el primer trimestre de 2017. Para dar respuesta a las preguntas de investigación planteadas en la sección anterior, la cadena de búsqueda utilizada por (T. Da Silva, et al., 2011) es válida y proporciona información relevante para el estudio que presentamos en este trabajo. Por lo tanto optamos por utilizarla en ambas etapas de búsqueda mencionadas previamente y está conformada de la siguiente manera:

(Usability OR Human-computer interaction OR Computer-human interaction OR Human factor OR User experience OR User centered design OR User Interface) AND (Agile OR Agile method OR Agile Development OR Agile Practice OR Agile Project OR Agile lifecycle OR SCRUM OR Extreme programming OR Lean development OR Feature Driven Development OR Dynamic system development).

El utilizar dicha cadena de búsqueda permite realizar y complementar la revisión de literatura sobre la integración de usabilidad en los desarrollos de software ágiles del año 2001 hasta el primer trimestre de 2017. En resumen se actualizó la revisión de literatura en un rango de 7 años adicionales a los 10 años analizados en (T. Da Silva, et al., 2011).

La búsqueda de información se realizó en varias de las bibliotecas digitales más reconocidas actualmente incluyendo IEEE Explore Digital Library, Elsevier Science Direct, CiteSeer, ISI Web of Science, ACM digital library, Springer, Scopus and Google Scholar.

Es importante señalar que el objetivo de investigación de T. Da Silva et al. (2011) es diferente al de nuestro estudio por ende las preguntas de investigación y el análisis de los resultados de búsqueda en T. Da Silva et al. (2011) también tienen un enfoque diferente al que se considera en este trabajo. Aquellas publicaciones coincidentes con T. Da Silva et al. (2011) fueron analizados de nuevo para proporcionar respuestas concretas a nuestras preguntas de investigación.

2.2.3 Selección de estudios y extracción de datos

Una vez obtenido el resultado de la búsqueda se escogieron aquellos estudios que cumplían alguno de los siguientes criterios:

- Estudios en los que específicamente se reportaban resultados sobre proyectos de software contruidos siguiendo alguna de las metodologías ágiles y en los que se utilizaban técnicas HCI durante el proceso de desarrollo
- Estudios con aportaciones relevantes como lecciones aprendidas, guías y/o recomendaciones sobre la integración de las filosofías HCI y Ágil

Se descartaron aquellos estudios duplicados.

Los artículos fueron examinados exhaustivamente obteniendo y registrando los siguientes datos para cada uno de ellos:

- Title
- Authors
- Publication year
- Paper classification. Los trabajos fueron clasificados como Artículos de Investigación (En inglés Research Paper o RP) aquellos que reportaban resultados o experiencia a partir de un trabajo realizado por investigadores sobre proyectos específicos de software. Los artículos clasificados como Artículos de Industria (En inglés Industry Paper o IP) son los que reportan resultados obtenidos por profesionales aplicados en proyectos reales dentro de la industria del software.
- Agile methodology. En este campo se registra la metodología o metodologías utilizadas en el desarrollo del software. Para aquellos artículos que no especifican un método ágil en particular en este campo fueron clasificados como “No especificado” (en inglés “Not specified”).
- HCI techniques applied during the agile process. Para aquellos artículos que no especifican una técnica UCD en particular este campo fué clasificado como “No especificado” (en inglés “Not specified”).
- Time/stage of the agile process at which the HCI techniques were applied.
- Team description, indica cuáles son los roles del equipo y cómo interactúan cuando las prácticas HCI se incorporan dentro del proyecto de desarrollo ágil.

Debido al dinamismo con el que se van generando nuevos descubrimientos y propuestas sobre el tema de esta investigación resulta importante incluir y contrastar en este documento referencias actuales sobre la información obtenida a partir de la revisión de la literatura. Esto último enriquece la vigencia e importancia de los resultados de este trabajo.

2.2.4 Resultados

Como mencionamos previamente los estudios analizados en nuestra investigación fueron resultado de dos etapas de búsqueda utilizando la cadena definida por (T. Da Silva, et al., 2011).

La primera etapa de búsqueda abarcó dos periodos de tiempo: 2001 – mid 2010 el rango de tiempo considerado en (T. Da Silva, et al., 2011) y el período mid 2010 – mid 2013 complementando los resultados. En el primer período de búsqueda se eligieron 57 estudios a considerar en nuestra investigación. Para el segundo período de búsqueda de la primera etapa (abarcando mid 2010 – mid 2013) fue aplicada en cada una de las fuentes digitales mencionadas en la sección 2.2.2 obteniendo 42 resultados de los cuáles fueron elegidos 11 estudios para efectos de nuestra investigación. La segunda etapa de búsqueda se aplicó para el periodo de tiempo mid 2013 – primer trimestre 2017 y se realizó consultando en las bibliotecas digitales de IEEE, Springer y ACM obteniendo un total de 303 publicaciones de las cuáles se eligieron 19 para nuestro estudio. Cada resultado de información fue analizado considerando los criterios de inclusión e identificado

aquellos estudios que se encontraban en más de una de las bibliotecas para tomarlos en cuenta una sola vez. Fueron seleccionados en total 87 estudios primarios para nuestra investigación. La lista completa la tenemos en el Anexo A.

La distribución por año de las publicaciones analizadas durante la primera etapa de búsqueda considerando publicaciones del año 2001 hasta el primer semestre de 2013 se observa en la figura C2-1.

Figura C2-1 Estudios analizados por año para el período 2001 – primera mitad 2013 (Caballero et al., 2016)

Al actualizar los resultados de búsqueda para la segunda mitad del año 2013 hasta el primer trimestre de 2017 la revisión por año de las publicaciones es la que se observa en la figura C2-2.

Figura C2-2 Artículos analizados por año para el período segunda mitad 2013 - primer trimestre de 2017

De un total de 303 artículos relacionados con el tema de la integración HCI-ASD para fines de nuestro estudio y de acuerdo a los criterios de inclusión se mantiene en promedio el número de artículos considerados en el análisis con un leve repunte para aquellas publicaciones del año 2016. Hasta la fecha en la que se desarrolló la segunda etapa de búsqueda las publicaciones del año 2017 no se consideró alguna para la investigación.

En la tabla C2-1 observamos un extracto de la información extraída de los estudios analizados. Los datos reflejan información de acuerdo a los descritos en la sección 2.2.3 de este trabajo. La columna “First Author” identifica el apellido del primer autor del trabajo, entre paréntesis se indica el año de publicación. En la columna “Title” se presenta el título del estudio; la columna “Source” indica la biblioteca digital en donde se

encontró el artículo. “Classification” es la columna en la que se identifica el tipo de publicación de acuerdo a la clasificación definida para nuestra investigación. La columna “Agile Methodology” indica el método ágil utilizado en el desarrollo de los proyectos de software objetos de estudio en aquellas publicaciones que especificaban el dato. En “HCI technique(s) or artifacts” se presenta información sobre las técnicas HCI utilizadas en los proyectos de software del artículo. La columna “Application moment in Agile” indica en que momento fueron aplicadas las técnicas de acuerdo a la clasificación establecida para este trabajo y descrita en secciones posteriores. Finalmente la columna “Team Collaboration” proporciona información sobre el tipo de colaboración existente entre el equipo ágil durante el desarrollo del (los) proyecto(s) observado(s) en el trabajo que se reporta. Para aquellos datos que eran parte de nuestro interés de estudio y no se encontraban especificados en los trabajos analizados se indicaron en el registro de la información como No Especificado (En inglés “Not specified”).

Tabla C2–1 Extracto de la información recabada de los estudios analizados

First Author (Year)	Title	Source	Classification	Agile Methodology	HCI technique(s) or artifacts	Application moment in agile	Team Collaboration
Lester (2011)	Combining agile methods and user-centered design to create a unique user experience: An empirical inquiry.	Scopus	RP	XP	User stories Prototype	Implicit Interaction Design	Close Collaboration
Isomursu (2012)	User experience design goes agile in lean transformation – A case study	Scopus	IR	SCRUM	User stories Scenarios Prototype	Overlapped Track	Close Collaboration
da Silva (2013)	Ten lessons learned from integrating interaction design and Agile Development	IEEE	RP	SCRUM	Usability Testing User stories Prototype	Overlapped Track	Close Collaboration
Abdallah (2013)	Quantified Extreme Scenario based design approach	Scopus	RP	QXSBD	Scenarios Prototype Usability Testing	Overlapped Track	Close Collaboration
Anwar (2014)	User-centered design practices in scrum development process: A distinctive advantage?	IEEE	RP	SCRUM	Usability Testing Persona Prototype Heuristic Evaluation	Interaction Design First	Close Collaboration
Peres (2014)	AGILEUX Model: Towards a Reference Model on Integrating UX in Developing Software Using Agile Methodologies	IEEE	RP	SCRUM	Persona Contextual Inquiry Prototypes Cognitive Walkthrough	Overlapped track	Close Collaboration
Choma (2015)	Communication of Design Decisions and Usability Issues: A Protocol Based on Personas and Nielsen’s Heuristics	SPRINGER	RP	SCRUM	Usability Testing Persona Prototypes Heuristic Evaluation	Implicit Interaction Design	Close Collaboration
Silva (2015)	Usability Evaluation Practices within Agile Development	IEEE	RP	SCRUM	Prototypes	Minimal up-front design	Cross-functional
De Freitas (2016)	AGILUS: A Method for Integrating Usability Evaluations on Agile Software Development	SPRINGER	RP	SCRUM	Prototypes Heuristic Evaluation	Overlapped track	Cross-functional
Sftsos (2016)	Integrating user-centered design practices into agile Web development: A case study	IEEE	RP	Other	Usability Testing Scenarios Persona Prototype	Overlapped track	Close collaboration

2.3 Prácticas HCI en Desarrollo ágil de Software (Qué)

Al analizar las publicaciones obtenidas en la primera etapa de búsqueda aplicada en nuestro estudio (2001 – mid 2013) se obtuvo la distribución que se observa en la figura C2-3 publicada en Caballero et al. (2016). La gráfica muestra un resultado similar al presentado por (Salvador et al., 2014) donde XP es el método ágil más utilizado seguido con una ligera diferencia por SCRUM. Sin embargo, la distribución generada a partir del análisis de todas las publicaciones consideradas en nuestra revisión (2001 – prim. Trim. 2017) indica una preferencia por SCRUM como se observa en la figura C2-4. La distribución de ambas gráficas se genera a partir del número de proyectos considerados en cada estudio donde se incorporan técnicas HCI. También se identifican algunos proyectos que utilizan otros métodos ágiles como FDD, WebTDD, InterMod y XSBD clasificados en la distribución gráfica dentro de la categoría “Otros” (en inglés Other). Los estudios que no proporcionaron a detalle información sobre la metodología ágil utilizada en sus proyectos se clasificaron en la categoría “No Especificado” (en inglés Not specified).

Figura C2-3 Metodologías ágiles utilizadas de acuerdo al análisis de publicaciones del período 2001 – primera mitad 2013 (Caballero et al. 2016)

Figura C2-4 Metodologías ágiles utilizadas en los proyectos de los estudios analizados para el período 2001 – primer trim. 2017

La literatura refleja la diversidad de técnicas HCI que han sido utilizadas en los últimos años dentro de los procesos de desarrollo ágil. Uno de los trabajos de revisión a la literatura más relevantes elaborados en la

última década sobre la integración de prácticas HCI en procesos ágiles de desarrollo lo presentan (T. Da Silva et al., 2011). Como parte de los resultados de su investigación T. Da Silva et al. (2011) identifican técnicas y artefactos más utilizados en un entorno ágil. Dentro de éstos se mencionan los siguientes: diseño mínimo por adelantado (LDUF por sus siglas en inglés), prototipos, historias de usuario, pruebas de usuario (en inglés usability testing), inspección de usabilidad y trabajar el diseño un sprint por adelantado.

En (Jia et al., 2012) los autores analizan la información obtenida a partir de una encuesta para identificar las herramientas de usabilidad más utilizadas en el proceso SCRUM. El análisis de los datos reflejaron como las de uso más frecuente los workshops, prototipos de bajo nivel, entrevistas, reuniones con los usuarios y escenarios. (Barksdale et al., 2012) realizan un trabajo de revisión cuya principal contribución descansa en la organización de los estudios existentes relacionados con la integración de UCD y ASD a través de cinco dimensiones. Identifican dichas dimensiones de integración como: prácticas, proceso, tecnología, gente y aspectos sociales. Por su parte Salvador et al (Salvador et al., 2014) clasifican las técnicas mencionadas en los trabajos analizados de acuerdo al enfoque HCI para el que fueron utilizadas: inspección, testing, investigación y complementario. Para cada una de las clasificaciones mencionadas resultaron las más utilizadas la evaluación heurística, tests formales, investigación individual y prototipado rápido respectivamente.

Posteriormente, en (Brhel et al., 2015) para efectos de su investigación clasifican los estudios analizados de acuerdo a las dimensiones utilizadas y descritas en (Barksdale et al., 2012). En la figura C2-5 podemos observar el sistema de clasificación resultante para las técnicas y artefactos HCI. La figura C2-6 muestra los totales reflejados por el número de artículos analizados en Brhel (2015) que muestran relación con determinadas técnicas HCI clasificados en la dimensión “Prácticas”. Esta última a su vez se sub clasifica en técnicas enfocadas a la investigación del usuario (user research), conceptualización (conceptualization), diseño (design) y evaluación (evaluation).

Figura C2-5. Clasificación de las técnicas HCI identificadas de acuerdo a cuatro dimensiones (Brhel et al., 2015)

La distribución de la gráfica en la figura C2-6 muestra principalmente la técnica prototipos como la más utilizada durante el diseño seguido de usability testing para la evaluación de la interacción. Además, para la etapa de conceptualización entre las más mencionadas se encuentran las historias de usuario, mock-ups, escenarios y persona.

Figura C2-6. Número de artículos relacionados con la dimensión "prácticas" (Brhel et al., 2015)

Con el objetivo de conocer la evolución de la industria incorporando técnicas HCI en sus procesos de desarrollo ágil, (Ovad et al., 2015) dan un seguimiento a ocho empresas durante un rango de tiempo. Como parte de los resultados obtienen que a través del tiempo los prototipos y las pruebas de usabilidad continúan como la técnica más utilizada en las empresas. El uso de los workshop aumenta y la técnica persona en general obtiene en el mismo porcentaje de uso.

(Magües et al., 2016a) clasifican las técnicas HCI utilizadas en el proceso de desarrollo de proyectos ágiles de acuerdo a tres de las etapas de desarrollo en IS: ingeniería de requerimientos, diseño y evaluación. Magües et al. (2016) identifican persona como la técnica más utilizada para la obtención de requerimientos y los screen snapshots así como los prototipos durante el diseño. Las pruebas de usabilidad (en inglés usability testing) se usan comúnmente durante la etapa de evaluación.

La figura C2-7 muestra la distribución publicada en (Caballero et al., 2016) obtenida a partir de nuestra revisión hasta el período primera mitad de 2013 y en la figura C2-8 se observa la distribución de resultados obtenida también en nuestro estudio pero abarcando hasta el primer trimestre de 2017. Los totales representan el número de proyectos ágiles descritos en los estudios analizados con relación a técnicas HCI incorporadas en el proceso de desarrollo.

Figura C2-7 Técnicas o artefactos HCI usados en Agile. Período de análisis 2001 – primera mitad 2013 (Caballero et al., 2016)

Figura C2-8 Técnicas o artefactos HCI usados en Agile. Período de análisis 2001 – primer trimestre 2017

De manera general en ambas distribuciones se observan resultados similares a los mostrados por diferentes autores. En nuestro análisis los prototipos se identifican como la técnica más utilizada seguida de usability testing, escenarios y persona.

Patton recomienda el uso de un prototipo abstracto para identificar los componentes a incluir en el contexto de la interacción y prototipos en papel para crear una maqueta de la interfaz de usuario (Patton, 2002). Los proyectos descritos en Chamberlain et. al. (Chamberlain et al., 2006) desarrollan diferentes tipos de prototipos de acuerdo al objetivo de obtención de información que en el momento del proyecto requieren. Para los proyectos de (Ferreira et al., 2007a) los prototipos en papel o elaborados en power point fueron útiles para obtener retroalimentación de los clientes. Meszaros y Aston (Meszaros et al., 2006) describen su experiencia utilizando prototipos de software en las pruebas de usabilidad. Hussain et al. (2008) desarrollaron una aplicación para teléfonos móviles utilizando prototipos de baja y alta fidelidad. Como resultado encontraron

que para aplicaciones móviles el uso de prototipos en papel debe ser complementado con prototipos de alta fidelidad para descubrir la percepción que los usuarios tienen sobre la aplicación real (Zahid Hussain et al., 2008b). En (Gonçalves et al., 2011) se presenta POLVO como herramienta que propone la automatización de prototipos de baja fidelidad como apoyo durante el proceso de diseño. (Rivero et al., 2013) proponen MockRE un proceso de modelado adaptado a SCRUM. MockRE que incorpora el uso de mockups como prototipos de la interfaz de usuario que permite obtener requerimientos y mejorar la participación de los usuarios finales durante el diseño del software.

Como aportación para sobrellevar los retos que implica la integración de técnicas HCI y las metodologías ágiles en el desarrollo de software, (Anwar et al., 2014) describen un marco de trabajo que facilita la integración de SCRUM y técnicas UCD. En dicho marco de trabajo los prototipos son una técnica fundamental del proceso. En su estudio para conocer las diferentes formas en las que se realiza la evaluación de usabilidad en los desarrollos ágiles (d. Silva et al., 2015) descubrieron que se utilizan diferentes tipos de prototipos, de baja o alta fidelidad, durante todo el proceso de desarrollo. El tipo de prototipo a utilizar depende de las necesidades del proyecto y la preparación del diseñador para aplicar esta técnica. Por otra parte, con el objetivo de mejorar la calidad del software y la satisfacción del usuario (de Freitas et al., 2016) presentan el método AGILUS para integrar evaluaciones de usabilidad en el proceso de desarrollo ágil. Durante el primer paso de AGILUS, el diseño de la interacción, se indica aplicar una evaluación formativa o empírica utilizando prototipos.

Las pruebas de usabilidad es otra técnica HCI que refleja relevancia en los desarrollos con enfoque ágil y una de las que más adaptaciones ha tenido para poder implementarse en un ambiente ágil. Las pruebas de usabilidad originalmente requieren definirse a partir de ciertas pautas que para el dinamismo con el que trabajan los procesos ágiles resultan poco prácticas de aplicar. Sin embargo, adaptadas al ambiente ágil este tipo de pruebas han resultado muy benéficas para conocer mejor si las aplicaciones que se están considerando son realmente útiles y utilizables para el usuario. Hudson (Hudson, 2003) resalta que las pruebas de usabilidad no necesariamente dependen de un grande y caro laboratorio para realizarlas. Para respaldar los resultados obtenidos aplicando las pruebas de usabilidad éstas pueden utilizarse junto con otras técnicas HCI durante el desarrollo en un ambiente ágil. Meszaros y Aston (Meszaros et al., 2006) sugieren utilizar prototipos de papel para facilitar las pruebas de usabilidad. (Zahid Hussain et al., 2008b) indican que utilizaron maquetas de baja fidelidad y mockups de alta fidelidad en sus pruebas de usabilidad.

Otra de las características de las pruebas de usabilidad por las que pareciera difícil implementar en un desarrollo ágil es que éstas deben realizarse con usuarios reales. Aunque es cierto que las metodologías ágiles reconocen como importante la interacción con el usuario, no siempre es sencillo trabajar directamente con los usuarios reales de la aplicación. (T. S. Da Silva, 2013) presentan lecciones aprendidas al implementar técnicas UCD en desarrollos ágiles. Como parte de éstas corroboran como alternativa el aplicar las pruebas de usabilidad con usuarios internos en caso de no contar con la disposición completa de los usuarios finales. Para asegurar la participación de los usuarios clave en las pruebas de usabilidad, es importante generar la suficiente confianza en ellos que aumente su interés de participar. Con este objetivo (McGinn et al., 2013) proponen una combinación de RITE y el método propuesto por Krug S. (2015) para aplicar pruebas de usabilidad durante todo el proceso de desarrollo ágil aplicando la técnica de manera rápida, con la misma efectividad e involucrando a usuarios finales en el entorno ágil. Permitiendo aplicar las modificaciones necesarias al diseño de interfaz antes de iniciar la implementación del código.

Como parte de las herramientas “ligeras” incluidas en el proceso ágil híbrido definido por Sfetsos et al (2016) para mejorar la calidad del producto final eliminando el número de defectos los autores indican aplicar pruebas de usabilidad durante el sprint Zero y posteriores.

Los escenarios son otra de las herramientas HCI comúnmente utilizadas en el proceso de desarrollo ágil principalmente para describir las diferentes formas, de acuerdo a contexto y ambiente, en las que el usuario utiliza o puede utilizar el sistema. La definición de los escenarios muestra una importante correlación con las metas de usabilidad a considerar para el proyecto en desarrollo (Jia et al., 2012) lo cual resulta una valiosa propiedad para los equipos ágiles. (Hakim et al., 2003) presentan una herramienta que ayuda a organizar el conjunto de escenarios estando siempre disponibles y mejorando la comunicación entre los equipos de desarrollo y diseño. Lee (Jason Chong Lee, 2011) y Obendorf et al (Obendorf et al., 2008) utilizan escenarios para obtener una vista y capturar el alcance de las metas y requerimientos del usuario. En (J C Lee et al., 2009) y (Jason Chong Lee, 2011) se reportan los resultados de utilizar el enfoque llamado eXtreme Scenario-Based Design (XSBD) donde el registro central de diseño (CDR por sus siglas en inglés) es la representación clave de diseño y sirve como apoyo en las actividades de sincronización y evaluaciones de usabilidad. Como parte de las actividades del marco de trabajo que incorpora técnicas HCI con SCRUM (Anwar et al., 2014) indican hacer uso de escenarios para conocer los requerimientos de aquellos usuarios que no tienen suficiente información sobre las características del software a desarrollar. También indican utilizar esta técnica en las etapas de planning y durante el desarrollo de los sprints. (Choma J. et al., 2016) definen el contenido del formato de historia de usuario propuesto, llamado UXStories, a partir de la descripción previa de los escenarios identificados en los que se utilizará el sistema.

En cuanto a las técnicas utilizadas para la obtención de requisitos en los procesos ágiles, persona es una de las más recurrentes. Cumpliendo con su propósito original descrito por Cooper (Cooper A. et al., 2007) esta técnica permite al equipo conocer e identificar al usuario final del software. También ha sido aprovechado como medio de comunicación entre el equipo teniendo siempre en mente al usuario. Broschinsky y Baker (Broschinsky et al., 2008) utilizan persona para identificar las necesidades reales del usuario y sus características. Singh (Singh, 2008) utiliza persona dentro de su enfoque ágil U-SCRUM como herramienta para mejorar la comunicación del equipo. (Zahid Hussain et al., 2008a) y (Wolkerstorfer et al., 2008) utilizan una adaptación de la técnica generando al inicio del proceso de desarrollo una definición básica de sus personas en base a la información obtenida a través de estudios del usuario. Posteriormente durante las iteraciones las definiciones iniciales se refactorizan. Esto es, pueden actualizarse de acuerdo a retroalimentación de iteraciones previas, se eliminan las que no se adapten a las nuevas necesidades que se requieran cubrir o incluso se crean nuevas personas. A esta adaptación de la herramienta los autores le llaman "Extreme Persona". En su proceso adaptado de diseño contextual (en inglés contextual design) (Rahim, I et al., 2014) agrupan los pasos del proceso en tres etapas. En la primera etapa se incluyen la investigación contextual (en inglés contextual inquiry), work model (modelo de trabajo), la consolidación (consolidation) y se agrega la técnica persona como un paso más en el proceso. (Choma, J et al. 2015) proponen un protocolo de comunicación en el que persona es una de las técnicas utilizadas como parte del vocabulario común para un mayor entendimiento de las recomendaciones o soluciones de diseño entre los diseñadores y programadores del equipo ágil. La primera técnica aplicada en el proyecto de estudio de (da Silva et al., 2016) es persona, a partir de las definiciones obtenidas se inicia con la aplicación de las evaluaciones de usabilidad. Los autores concluyen que, además de reducir el re trabajo, aplicar evaluaciones de usabilidad antes de los ciclos de desarrollo ayudan para que el equipo de desarrollo y los stakeholders entiendan la importancia de integrar UCD y el desarrollo ágil desde las etapas iniciales.

Contextual Inquiry es otra de las herramientas utilizadas para conocer mejor al usuario. Miller describe esta herramienta como una técnica de investigación para descubrir cómo luce el ambiente de trabajo del usuario, cómo desarrolla su trabajo y cuáles son sus metas (Miller, 2005). (Felker et al., 2012) utilizan contextual inquiry en combinación con prototipos para obtener una visión holística del proyecto desarrollado utilizando SCRUM como metodología ágil. Como parte del marco de trabajo propuesto por (Anwar et al., 2014) los autores sugieren las entrevistas contextuales como una de las técnicas a aplicar antes de iniciar el desarrollo de las iteraciones para conocer al usuario. También se sugiere aplicar este artefacto durante la etapa de

planeación para identificar los requerimientos del product owner. En el modelo AgileUX para incorporar técnicas HCI dentro del proceso de mejora CMMI nivel dos descrito en (Peres et al., 2014) se sugiere aplicar contextual inquiry. (Kropp et al., 2016) reportan su experiencia aplicando técnicas UCD en proyectos ágiles de precio fijo. Identifican actividades a realizar por un especialista UCD on site y las presentan agrupadas en cuatro etapas: inicialización, conceptualización, implementación y seguimiento. Una de las actividades de la primera etapa indica aplicar contextual inquiry con los potenciales usuarios y stakeholders.

Una de las técnicas que han resultado muy importantes al momento de integrar usabilidad con el enfoque HCI son las historias de usuario. En términos de interacción persona-ordenador éstas se utilizan para capturar las experiencias del usuario. Patton (Patton, 2005) aplicó historias de usuario para organizar la información sobre las necesidades del usuario, menciona que las historias de usuario deben mantenerse pequeñas y simples incluso para describir procesos iterativos complejos aunque esto signifique crear más historias. Singh propone el proceso U-SCRUM (Singh, 2008) donde el product backlog definido en este enfoque incluye historias con mayor detalle en relación a los criterios de aceptación y considera dentro de las historias de mayor prioridad aquellas que tienen impacto en la usabilidad. (Coatta et al., 2010) afirman que las historias de usuario ayudan a entender la funcionalidad y experiencia que el producto necesita tomando en cuenta las expectativas y deseos del usuario. Isomursu (Isomursu et al., 2012) describe el proceso de adopción de una metodología ágil en una compañía e identifica las historias de usuario como una importante herramienta durante todo el proceso de desarrollo que ayudó al equipo a tener una perspectiva completa que le permitiera entregar productos que cubrieran la funcionalidad y necesidades reales del usuario.

2.4 Momentos de Integración HCI en el proceso ágil (Cuándo)

Uno de los factores clave al incorporar las técnicas HCI dentro del desarrollo ágil es identificar el momento adecuado para aplicarlas en el proceso y obtener un buen nivel de usabilidad. Para conocer mejor cuándo los equipos ágiles utilizan las técnicas HCI se aplicó un análisis exhaustivo sobre aquellos estudios, parte de nuestra investigación, que indicaban explícitamente el momento en el cual utilizaron las técnicas.

La comunidad HCI aboga por que el diseño de la interfaz se debe realizar por completo al inicio del proceso de software previo a la implementación del código o cualquier otra etapa del desarrollo. En inicio esta práctica resulta en cierta forma incompatible con los principios de la filosofía ágil sin embargo investigadores y profesionales del área reportan los resultados de sus experiencias con equipos ágiles incorporando la usabilidad de esta forma. (Patton, 2002) describe su experiencia con este enfoque, afirma que aunque requiere tiempo extra aplicar técnicas de usabilidad al principio del proceso ágil ayuda a identificar los diferentes componentes de interacción con anticipación. (Meszaros et al., 2006) argumentan que el tener una visión holística del sistema desde el inicio del desarrollo reduce el re-trabajo que pudiera generarse durante las iteraciones para corregir inconsistencias de diseño. (Felker et al., 2012) concluyen que el aplicar una investigación contextual al inicio del proyecto para obtener una visión holística del proyecto fue muy útil para que los equipos pudieran estar enfocados posteriormente durante los ciclos ágiles únicamente en el diseño e implementación.

En el proyecto objeto de su estudio (Rahim et al., 2014) antes de la implementación del código se elabora por completo el diseño de interfaz utilizando una adaptación del método contextual design. Las diferentes etapas del método son agrupadas en tres fases. Incorporando dentro de la primera de ellas el uso de la técnica persona e incluyendo en la etapa final la implementación del código del prototipo funcional utilizando extreme programming.

Otra manera en la que los equipos ágiles han incorporado el uso de las técnicas HCI es realizar un diseño de interacción inicial a mejorar posteriormente durante las iteraciones. Los equipos aplican actividades de diseño

centrado en el usuario utilizado una iteración inicial, la llamada iteración 0 o sprint 0 en el caso de SCRUM. El objetivo es dedicar este tiempo solamente para conocer mejor al usuario y tener un panorama general sobre sus necesidades y metas. De esta forma el diseño puede proyectarse con más enfoque al usuario a partir de la información obtenida desde el inicio del proyecto y antes de cualquier otra etapa que en el caso de los desarrollos ágiles se dan durante las iteraciones. En particular, al ser los dos procesos UCD y Agile iterativos e incrementales las técnicas de usabilidad que se integran al proceso ágil pueden a su vez utilizarse en esos sentidos contribuyendo al nivel de usabilidad gradualmente. Constantine et al. (Constantine, 2002) sostienen que desarrollar un mínimo diseño por adelantado es necesario para estar bien organizado y presentarles a los usuarios una interfaz consistente y entendible. En su trabajo los autores señalan una serie de puntos a considerar para obtener un diseño mínimo por adelantado de una forma sencilla. Chamberlain et al. (Chamberlain et al., 2006) describen su experiencia con SCRUM al dedicar un sprint al inicio del proyecto en lo que ellos llaman “fase de análisis del negocio” para obtener requerimientos. Para esto, incorporan la técnica persona y adoptan en las iteraciones posteriores un enfoque evolutivo. (Ferreira et al., 2007a) reportan el resultado de uno de sus equipos de estudio al obtener al inicio del desarrollo únicamente información suficiente de los usuarios para generar prototipos que les permitieran obtener retroalimentación del usuario sobre el diseño inicial e irlo refinando posteriormente durante las iteraciones. Los elementos del equipo expresaron que tener un cierto porcentaje del diseño de la interfaz de usuario antes de que inicie la implementación es esencial y suficiente para el proyecto. Lee et al. (J C Lee et al., 2009) proponen un proceso de desarrollo combinando XP y el diseño basado en escenarios (en inglés Scenario Based Design or SBD). En este proceso al inicio del proyecto en la fase de requerimientos se genera un esqueleto incluyendo toda la interacción con el usuario la cual es continuamente revisada y actualizada. Con el objetivo de obtener una idea general de la interfaz de usuario y evitar re-trabajo durante las iteraciones Kollman et al. (Kollman, 2009) sugieren utilizar una fase de investigación inicial. Independientemente del nombre que tenga ya sea fase de exploración, sprint 0, iteración 0, etc.

En (Zahid Hussain et al., 2008a) y (Wolkerstorfer et al., 2008) se describe la adaptación aplicada al proceso XP incorporando en diferentes momentos del desarrollo técnicas de usabilidad como persona, pruebas de usabilidad y evaluaciones de usabilidad aplicadas por expertos. Los autores incorporan una adaptación a la técnica persona resultando lo que ellos llaman “extreme persona”, al inicio del proceso se obtiene información sobre los usuarios a través de estudios de usuario para definir las personas relacionadas. Al final de cada iteración la visión sobre los usuarios es más amplia lo cual ayuda para actualizar las definiciones iniciales o crear nuevas personas si es necesario. En la figura C2-9 podemos ver el modelo propuesto por los investigadores.

Figura C2-9 Integración de instrumentos HCI en XP (Zahid Hussain et al., 2008a)

(Anwar et al., 2014) proponen un marco de trabajo que incorpora el uso de técnicas HCI en el proceso ágil de desarrollo de software enfocado principalmente a sitios web y aplicaciones móviles. Durante las primeras etapas del proceso, llamadas planeación y el sprint de diseño (o sprint 0), se usan técnicas HCI como persona y prototipos para obtener información y tener una visión del proyecto y necesidades de interacción del usuario antes de iniciar con las iteraciones de desarrollo. Durante los sprints posteriores llamados “Sprint de desarrollo” los diseñadores de interacción, en caso de ser necesario, rediseñan la interfaz apoyándose en otro grupo de herramientas HCI como los prototipos, heuristic evaluation y escenarios entre otras. Derivado del estudio de (d. Silva et al., 2015) sobre la forma en que se llevan a cabo las evaluaciones de usabilidad en ambientes ágiles, concluyen que se realizan de manera continua durante todo el proceso de desarrollo con un objetivo diferente. Desde la iteración inicial 0 para verificar ideas de diseño y metas del negocio. Durante las iteraciones se aplican también las evaluaciones de usabilidad para mejorar el diseño y priorizar las tareas. En la etapa final ayudan a los diseñadores de interfaz a organizar las tareas priorizando las características que pudiera ser necesario reparar para liberar lo previsto. (de Carvalho et al., 2016) enfocan su estudio en comprender el impacto de aplicar evaluaciones de usabilidad antes de los ciclos de desarrollo. Concluyen que además de minimizar el retrabajo evaluar la usabilidad antes del desarrollo contribuye al cambio de mentalidad en relación a la importancia de integrar en etapas tempranas los procesos de desarrollo ágil y UCD.

Una de las prácticas que ha resultado muy útil al momento de integrar las herramientas HCI en el proceso ágil es trabajar bloques sobrepuestos de diseño y desarrollo en paralelo. Los diseñadores trabajan los diseños para la interfaz del usuario al menos una iteración por adelantado a las implementaciones que realizan los desarrolladores. Esto es, en la iteración en curso se elabora el diseño que será implementado en la siguiente iteración durante el bloque de desarrollo; también se implementan los diseños elaborados en iteraciones previas. Sy (Sy, 2007) presenta sus resultados y propone un proceso que integra técnicas UCD en el proceso SCRUM considerando la iteración inicial como una pequeña fase para obtener requerimientos al inicio del proyecto. Posteriormente, el diseño y la implementación de este se llevan a cabo en bloques paralelos sobrepuestos que la autora denomina “Bloque del desarrollador” y “Bloque del diseñador de interacción”. En la figura C2-10 podemos observar el proceso utilizado por Sy.

Sy indica que durante los primeros ciclos los desarrolladores trabajaron en la codificación del software que solamente requería diseño mínimo. De esta manera los diseñadores tuvieron tiempo para realizar investigaciones de usabilidad.

Figura C2-10 Proceso de desarrollo integrando UCD en desarrollo ágil (Sy, 2007)

El diseño de interacción que se realiza en el bloque correspondiente se trabaja con al menos dos ciclos de adelanto respecto a las características de interacción que se implementan en el bloque de desarrollo.

Fox, Sillito y Maurer (Fox et al., 2008) describen su proceso de desarrollo donde consideran un estado inicial para descubrir y entender las necesidades del usuario antes de que se inicie con la codificación. Después de la fase inicial los miembros del equipo trabajan concurrentemente: mientras el equipo de desarrollo implementa las características de la actual iteración el equipo de diseño continúa con el diseño de la interfaz de usuario a ser utilizado en la siguiente iteración. (Beyer et al., 1998) reportan los resultados de aplicar Contextual Inquiry para obtener información de los usuarios y construir las historias de usuario en la primera iteración desarrollando la interfaz de usuario. Señalan la importancia de no mezclar el diseño de la interfaz de usuario dentro del trabajo de implementación y sugieren continuar las iteraciones en paralelo.

De manera similar Najafi et al. (Najafi et al., 2008) utilizan inicialmente un sprint exclusivamente para actividades HCI. El equipo UX (User eXperience) utilizó el sprint cero específicamente para entender a los usuarios explorando su contexto e identificando sus metas. Estas tareas exploratorias las realizan incorporando la técnica persona. En los sprints posteriores se desarrollan en paralelo actividades correspondientes al diseño de la interfaz en el bloque que denominan “User experience” y las actividades de implementación en el bloque “Engineering Development”. Los diseños se definen un sprint antes de ser implementados. La figura C2-11 muestra el modelo Scrum adaptado que utilizaron en su estudio donde se observan las tareas que realizaron en cada uno de los dos bloques durante los sprints .

Figura C2-11 Modelo de integración entre SCRUM y el Diseño UX (Najafi & Toyoshiba, 2008)

Por su parte Cho (2009) utiliza técnicas HCI como persona, maquetas y escenarios para obtener información de referencia en la definición de las historias de usuario y criterios de aceptación. En su trabajo ella describe la manera en la que el equipo trabaja en paralelo actividades de implementación y desarrollo, por un lado los programadores implementando código mientras que los diseñadores, analistas de negocio y líderes de proyecto preparan el próximo sprint.

El proceso RITE + Krug propuesto por (McGinn et al., 2013) para incorporar las pruebas de usabilidad en un desarrollo ágil considera realizar las actividades de diseño de interfaz que involucran técnicas HCI un sprint por adelantado a las de desarrollo. De esta manera el equipo de diseño tiene tiempo para realizar las modificaciones o mejoras necesarias en el diseño de interacción antes de que sea implementado el código. Por su parte los desarrolladores durante las iteraciones dedicadas al diseño de la interfaz se enfocan en estabilizar la funcionalidad actual del producto software.

(Peres et al., 2014) proponen un modelo que permita incluir técnicas que mejoren el nivel de UX de un desarrollo ágil de software en un proceso de mejora CMMI. En particular, para la administración de

requerimientos (área de proceso nivel dos CMMI) (Peres et al., 2014) recomiendan que los diseñadores UX trabajen una o más iteraciones por adelantado de los desarrolladores además de apoyarlos durante la implementación del código.

(Kropp et al., 2016) reportan su experiencia al implementar técnicas UCD en su proceso ágil de desarrollo trabajando en paralelo un bloque de diseño y otro de desarrollo. Las actividades UCD agrupadas en cuatro etapas: iniciación, conceptualización, implementación y seguimiento se llevan a cabo antes del inicio del proyecto y durante el bloque de diseño.

Los equipos ágiles también optan por elaborar el diseño de la interfaz durante la iteración en curso. Adaptando el proceso SCRUM, (Rivero et al., 2013) proponen un enfoque de desarrollo por modelos donde durante cada iteración los usuarios participan diseñando, junto con los elementos del equipo, mockups que reflejen sus requerimientos y que posteriormente serán implementados en prototipos. Con el objetivo de que los desarrolladores obtengan retroalimentación en cualquier momento del sprint, Krushe S, et al. (2014) combinan dos enfoques: una herramienta “ligera” basada en escenarios para producir prototipos en el momento que sea necesario y Rugby, un método ágil adaptado de SCRUM.

A partir de lo expuesto previamente, en general se identifican cuatro enfoques en los que se incorporan técnicas HCI al proceso de desarrollo ágil. A continuación se describen y los identificamos con su nombre en inglés para hacer referencia a ellos de manera más universal:

- *Interaction design first.* Se aplican las técnicas HCI antes de iniciar las iteraciones ágiles para obtener una visión holística del sistema a construir. Este enfoque también es referenciado en la literatura como “Big Design Up front” (T. Da Silva et al., 2011). El comportamiento de la incorporación HCI con este enfoque lo podemos observar en la figura C2-12.

Figura C2-12 Enfoque de integración “Interaction Design First”

- *Minimal up-front interaction design.* Un mínimo diseño de interacción es definido antes de iniciar con las iteraciones ágiles. Posteriormente el diseño de Interfaz de usuario definido al inicio es refinado durante las iteraciones. Esto permite a los desarrolladores dedicar tiempo a discutir sobre las líneas guías básicas que permiten consistencia y coherencia en el desarrollo de las siguientes iteraciones. La figura C2-13 muestra de manera gráfica la forma en la que se trabaja el diseño en este enfoque.

Figura C2-13 Enfoque de integración “Minimal Up-Front Design”

- *Implicit interaction design*. Las técnicas HCI son aplicadas para diseñar la interacción de las historias de usuario a ser desarrolladas durante la misma iteración. De manera gráfica en la figura C2-14 tenemos la representación de este enfoque.

Figura C2-14 Enfoque “Implicit Interaction Design”

- *Parallel track*. El diseño se trabaja al menos una iteración por delante a la implementación. Por lo tanto permite a los desarrolladores durante la iteración abordar la implementación del código de las historias de usuario cuya interacción ha sido diseñada previamente. Generalmente las actividades de diseño para iteraciones posteriores y las actividades de implementación de código se trabajan en bloques paralelos dentro de la misma iteración. En las figuras C2-15 y C2-16 se representa el proceso de integración UCD en agile desarrollando el diseño de la interfaz en iteraciones por delante a las actividades de implementación.

Figura C2-15 Enfoque “Iteration Ahead” – Iteración inicial

Figura C2-16 Enfoque “Iteration Ahead” – Iteración posterior

Para visualizar un panorama general de cuándo son aplicadas las técnicas HCI dentro del proceso ágil clasificamos los estudios analizados en la revisión de literatura tomando en cuenta los enfoques de integración descritos anteriormente. Para la primera etapa de búsqueda obtuvimos la distribución de la figura C2-17 publicada en (Caballero et al., 2016). Para el análisis incluyendo la segunda etapa de búsqueda observamos la distribución en la figura C2-18. Aquellos estudios que no proporcionan información suficiente que permitiera identificar el enfoque utilizado se reflejan en la categoría “Not specified” de ambas gráficas.

Los resultados de ambas distribuciones (figuras C2-17 y C2-18) muestran que la interacción de las técnicas se realiza mayormente con dos enfoques: minimal up-front design y Overlapped Track. La distribución obtenida de nuestra SLR hasta mediados de 2013 (figura C2-17) señala el enfoque minimal up-front design como el más utilizado. Mientras que la distribución obtenida considerando las publicaciones hasta el primer trimestre de 2017 muestra el enfoque Overlapped Track como el más utilizado. Esto refleja que el uso de este enfoque en la integración de técnicas HCI en el proceso de desarrollo ágil ha ganado terreno en los últimos años.

Figura C2-17 Enfoque de integración HCI utilizado en proyectos ASD. Período de análisis 2001 – primera mitad de 2013 (Caballero et al. 2016)

Figura C2-18 Enfoque de integración HCI en un proyecto de desarrollo ágil. Período de análisis 2001 – primer trimestre 2017

Las diferentes prácticas HCI que se aplican comúnmente en los enfoques de integración fueron clasificadas de acuerdo a éstos. Los resultados para la primera etapa de búsqueda de nuestra revisión de literatura se reflejan en la figura C2-19 y la distribución resultante de la revisión de literatura incluyendo las dos etapas de búsqueda se muestra en la figura C2-20.

Figura C2-19 Prácticas HCI utilizadas por enfoque de integración en proyectos ASD. Período de análisis 2001 – primera mitad 2013 (Caballero et al. 2016)

Figura C2-20 Prácticas HCI utilizadas por enfoque de integración en proyectos ASD. Período de análisis 2001 – primer trimestre 2017

Choma et al. (2015) aplican el protocolo propuesto durante las iteraciones de un proyecto de rediseño web como un lenguaje en común para mejorar la comunicación entre los diseñadores UX y los desarrolladores ágiles durante el proceso SCRUM. Choma et al. (2015) observaron que mejoró la comprensión por parte de los desarrolladores de las decisiones de diseño a implementar.

Considerando las etapas del proceso UCD (ISO 13407) en DaSilva et al (T. Da Silva et al., 2011) concluyen que la integración UCD y las metodologías ágiles debe enfocarse durante el diseño y en la evaluación de la usabilidad. Durante la etapa de diseño se utilizan persona y los prototipos de bajo nivel. Por otro lado en las evaluaciones del diseño generalmente se utilizan también prototipos de bajo nivel. Por su parte los resultados presentados por Salvador et al (Salvador et al., 2014) reflejan que en la mayoría de los trabajos considerados en su estudio los métodos de usabilidad se aplicaron durante la fase de implementación y pruebas del diseño. (Brhel et al., 2015) indican que los métodos de usabilidad deben utilizarse en las fases tempranas del desarrollo y en cada iteración con el objetivo de reducir los problemas de usabilidad en la versión final del software.

2.5 Organización del equipo ágil de desarrollo al aplicar técnicas HCI (Quién)

Originalmente las actividades relacionadas con la incorporación o mejora de la usabilidad del software se asignaban específicamente a especialistas en el área del diseño de la interfaz. Las actividades relacionadas con técnicas y prácticas HCI únicamente podían realizarlas expertos en usabilidad ya que teóricamente únicamente ellos cuentan con el conocimiento necesario para desarrollarlas con eficacia. En los últimos años dentro de la IS se ha estudiado y reconocido la importancia de involucrar a otros elementos del equipo para lograr un nivel de usabilidad adecuado del software que se está desarrollando. El hecho de que el concepto de usabilidad se atribuya únicamente a los expertos en la materia se considera un obstáculo para una mejor comprensión de la importancia de este factor como atributo fundamental de la calidad del software (Seffah A et al., 2004). Por falta de conocimiento de todos los elementos del equipo sobre usabilidad muchas veces un nivel adecuado para este atributo no se alcanza o definitivamente no se considera para el software.

Independientemente de la metodología utilizada en un proyecto de software los desarrolladores tienen la relación más cercana con el diseño o quien lo realiza ya que les impacta en primera instancia en la implementación. Por lo anterior se han generado varias recomendaciones HCI enfocadas a involucrar a estos elementos del equipo en el tema de usabilidad (Juristo et al., 2007) (Seffah et al., 2009). Uno de los factores que se señalan para mejorar la percepción de la importancia de la usabilidad en el desarrollo de software es que todos los elementos del equipo cuenten con la preparación o información necesaria sobre el tema. Por medio de talleres, preparación profesional o guías que les proporcionen suficiente información que les permita entender e incluso utilizar técnicas de usabilidad durante el proceso de desarrollo aunque no sean especialistas o expertos en HCI (Juristo et al., 2007) (Seffah et al., 2009) (Carvajal L. et al., 2013).

Tomando en cuenta lo puntualizado anteriormente la situación se observa también en los equipos ágiles principalmente por la característica de multifuncionalidad y auto organización. En caso de no contar con un experto en HCI y con el conocimiento necesario, todo el equipo de desarrollo ágil tiene la capacidad para desempeñar actividades de diseño relacionadas con usabilidad. Los resultados obtenidos hasta el momento a través del análisis de la literatura proporcionan información que permite identificar factores y definir recomendaciones. A continuación se describen las diferentes clasificaciones que identificamos en cuanto a quién aplica las técnicas HCI y cómo se organizan los equipos ágiles durante el proceso de desarrollo en el que se incorporaron éstas.

En relación con el responsable del diseño de la interacción y por ende aplicar las técnicas de usabilidad en el equipo ágil identificamos los siguientes roles:

- *Experto*, representa aquellos especialistas de diseño como los expertos UX, expertos en usabilidad, expertos HCI, expertos UCD y diseñadores gráficos expertos.
- *No experto*, se refiere a aquellos miembros del equipo sin formación como especialistas HCI que están enfocados a diseñar la interacción. Los desarrolladores pueden representar este rol en el proceso ágil.

En cuanto a la división de actividades de diseño e implementación se identifican las siguientes formas de organización en los equipos ágiles:

- *Equipos independientes*. Donde un equipo – grupo de especialistas en HCI, elementos que desempeñan el rol de diseñadores, un especialista HCI o un elemento que desempeña el rol de diseñador - desarrolla las actividades de diseño y otro equipo las tareas de implementación. Hay dos posibles situaciones en este caso. El primero es el que se da cuando existe una colaboración cercana entre estos dos equipos. Durante todo el proceso de desarrollo ambos equipos participan en el diseño de la interfaz y durante la etapa de implementación, un constante feedback entre ambos equipos es un punto medular. La otra situación que se observa es la que se presenta cuando la comunicación entre los dos equipos únicamente se da durante la entrega del diseño a implementar por el equipo de desarrollo y posteriormente no existe mayor comunicación durante el resto del proceso.
- *Equipo cross-functional*. El equipo está conformado por desarrolladores, diseñadores UI, expertos UCD, stakeholders y Project managers. Todos los elementos del equipo se involucran durante el diseño y durante la etapa de implementación.

Las figuras C2-21 y C2-22 muestran las distribuciones obtenidas de acuerdo al tipo de organización de los equipos ágiles al aplicar técnicas HCI. Los resultados se obtuvieron durante nuestra revisión de literatura a partir de la información recabada de aquellos proyectos en donde se tenía información suficiente para identificar este dato. Aquellos estudios donde no se mencionaba específicamente colaboración entre los equipos se clasificaron en la categoría “Not specified”. De manera similar a los apartados anteriores presentamos los resultados considerando las dos etapas de búsqueda. La figura C2-21 muestra la distribución

obtenida en la revisión para el período 2000 – mid 2013 publicada en (Caballero et al., 2016) y la figura C2-22 muestra la distribución actualizada incluyendo hasta el primer trim. 2017.

Figura C2-21 Organización de los equipos ágiles trabajando HCI en el proceso de desarrollo. Período de análisis 2001 – primera mitad de 2013 (Caballero et al. 2016)

Figura C2-22 Organización de los equipos ágiles trabajando HCI en el proceso de desarrollo. Período de análisis 2001 – primer trimestre de 2017

Se observa en ambas distribuciones que los equipos mantienen su preferencia por trabajar sus tareas de diseño e implementación de manera independiente pero colaborando cercanamente durante todo el proceso de construcción del software. Incluso incluyendo el análisis de las publicaciones para el período actualizado de búsqueda se muestra un ligero incremento en el número de proyectos donde se utilizó este tipo de organización para su desarrollo. Lievesley y Yee notaron que es esencial que el equipo de diseño trabaje independientemente del equipo de desarrollo en las etapas creativas tempranas con el objetivo de preservar las valiosas distinciones en los estilos de trabajo Lievesley (2006). Posteriormente se requiere estar en continua comunicación para asegurar una cadena paralela de trabajo. Ferreira et al. (2007) describen como las iteraciones contribuyen a mejorar la comunicación y relaciones de trabajo entre los diseñadores y los

desarrolladores de software. En su modelo de bloques paralelos Sy (2007) sugiere que durante los ciclos iniciales los desarrolladores deben iniciar trabajando sobre la arquitectura que tendrá la codificación del software o en características importantes que solamente necesiten mínimo diseño. De esta forma los diseñadores en esas etapas tempranas tienen tiempo para realizar las actividades necesarias de diseño y preparar las interfaces a ser implementadas en los ciclos posteriores. Una vez iniciados los ciclos de implementación a los diseños previos es importante que los diseñadores trabajen de manera cercana con los desarrolladores para responder preguntas sobre el diseño que está siendo desarrollado.

Siguiendo este mismo enfoque de equipos independientes con cercana colaboración Coatta et al. (2010) señalan que el equipo UCD y el equipo de desarrollo necesitan ayudarse mutuamente. Los equipos involucrados en el proceso de desarrollo participan tanto en la definición de historias de usuario como durante todo el proceso de desarrollo Coatta (2010).

Ferreira et al. (Ferreira et al., 2007a) reportan un caso de estudio donde el equipo de desarrollo trabajó separadamente del equipo de diseño al inicio del proceso. El diseño de la interfaz de usuario fue proporcionado al equipo de desarrollo una vez que estuvo completa. Posteriormente los diseñadores realizaban pruebas diarias desde la perspectiva de diseño y apoyaban a los desarrolladores con explicaciones más amplias a dudas que estos consultaban respecto al diseño propuesto. Ferreira et al. (2007) observaron que trabajando de esta forma puede tener ventajas como mejorar las pruebas de usabilidad y la relación entre los diferentes equipos. Los grupos de desarrollo de software de las dos compañías observadas por (d. Silva et al., 2015) están conformados por diseñadores UX y desarrolladores. Independientemente de la ubicación física en la que se encuentren (algunos se integran en uno solo espacio como equipo) tienen una cercana colaboración durante todo el proceso de desarrollo. En una de las compañías los elementos que se encargan del diseño UX trabajan en varios proyectos mientras que en la otra compañía el equipo UX apoya a un solo proyecto. Las actividades de usabilidad se desarrollan principalmente por los elementos expertos HCI. En (McGinn et al., 2013) el equipo UX trabaja de manera cercana con el equipo de desarrollo durante los sprints dedicados a la implementación de código de las funcionalidades respondiendo preguntas, proporcionando retroalimentación y ajustando el diseño en caso de ser necesario.

Los proyectos que reportan haber trabajado de manera independiente el diseño y la implementación con mínima o ninguna colaboración entre los equipos responsables de cada actividad generalmente observaron algunas complicaciones durante el proceso de desarrollo. Trabajar por separado puede representar problemas para los equipos, principalmente por falta de comunicación y coordinación entre las actividades que realizan. En (Chamberlain et al., 2006) los autores reportan la experiencia de trabajo con dos equipos independientes y además ubicados en diferentes lugares. La ubicación de los equipos resultó un factor que interfirió en la colaboración de los dos equipos generando problemas. El equipo de diseño decidió no utilizar la metodología ágil, en este caso SCRUM, para llevar la gestión de sus actividades.

En el caso de los equipos cross-functional uno de los principales beneficios observados es una mejor comunicación dentro del equipo sin embargo también pueden darse situaciones en donde ésta no se produzca de manera eficiente. Najafi et al. (Najafi et al., 2008) comparan los resultados de dos equipos cross-functional trabajando proyectos de software con metodologías ágiles. Uno de los equipos presentó problemas debido a que los roles del diseñador HCI y el desarrollador se distanciaron durante el proceso. Los diseñadores no estaban informados sobre las características que estaban siendo implementadas en los sprints lo que impactaba en actividades clave como las pruebas con el usuario, generando constante cambio en los requerimientos. En el segundo proyecto el equipo encargado de las actividades relacionadas con la experiencia de usuario fue considerado durante todo el proceso de desarrollo utilizando artefactos HCI en cada actividad.

Una de las estrategias utilizadas por Cho (2009) para identificar las metas de los usuarios y mejorar la comunicación del equipo fue usar el equipo cross-functional. Conformado de esta manera todos los elementos del equipo intercambian ideas sobre la manera en la que deben construirse las características a ser implementadas. En las reuniones los diseñadores de interfaz presentan especificaciones de diseño de baja fidelidad con el objetivo de construir un consenso de equipo. Se identifican dependencias externas, requerimientos funcionales adicionales así como necesidades del negocio. Los roles HCI son parte del equipo y apoyan a los desarrolladores durante el sprint.

Tzanidou y Ferreira (2010) apoyan la creación de equipos cross-functional para garantizar la continua participación del diseñador de interfaz durante todo el proceso de desarrollo. De esta manera se asegura que los desarrolladores mantienen la consistencia en la interfaz de usuario como fue especificada por la guía de estilo o maqueta definida. Parson et al. (2007) identifica la cercana colaboración entre un equipo cross-functional como uno de los factores clave para una integración exitosa de las técnicas HCI en el enfoque ágil. Los elementos del equipo además de los conocimientos propios de su rol cuentan con conocimientos suficientes que les permiten compartir ideas en conjunto sobre el diseño durante todo el proceso de construcción del software. En (de Freitas et al., 2016) parte del equipo ágil se encargaba de diseñar los prototipos cuya usabilidad sería evaluada utilizando el método AGILUS mientras otros elementos del equipo implementaban el código de aquellas funcionalidades aprobadas previamente.

El equipo de desarrollo del estudio presentado por (de Carvalho et al., 2016) no contaba con especialistas UX, sin embargo ejecutaron sin problema las evaluaciones de usabilidad. El equipo aplicó durante una etapa inicial antes de los ciclos de desarrollo e incluso durante éstos técnicas HCI como persona, prototipos y heurísticas.

2.6 Resumen de la revisión de literatura

Como hemos visto en las secciones anteriores durante los últimos años se han incrementado los esfuerzos por parte de las áreas de HCI e IS en proporcionar información que permita comprender mejor y corroborar la importancia de la usabilidad en el desarrollo ágil de software. Se observa que la principal dirección de dichos esfuerzos se orienta hacia incrementar la relevancia que los equipos ágiles atribuyen a la usabilidad como atributo de calidad en el desarrollo de sus proyectos de software. Para ello se han desarrollado propuestas que respaldan la posibilidad de realizar una efectiva integración entre HCI y el desarrollo ágil sin afectar los principios sobre los que descansa dicha filosofía. Los estudios muestran las ventajas que representa incorporar técnicas HCI en el proceso de desarrollo ágil para la calidad del software en construcción. La literatura ofrece información sobre diferentes formas en las que se puede sacar provecho a la integración de las técnicas HCI dentro del proceso ágil. Además de la mejora en la calidad del software se describen los beneficios en otros diferentes aspectos como la relación interna entre el equipo ágil y la relación de éste con el usuario. También es posible que el proceso de desarrollo adopte ciertos cambios al incorporar la usabilidad en la construcción del software. Se proponen marcos de trabajo para que esta incorporación proporcione resultados más efectivos aunque no todos exponen una validación empírica. Profesionales e investigadores comparten sus experiencias y lecciones aprendidas buscando contribuir con buenas prácticas para integrar técnicas HCI en el desarrollo ágil.

Hasta el momento, la información no se proporciona a los miembros del equipo ágil de una manera estandarizada que les sirva de apoyo en la incorporación de éstas técnicas durante todo el proceso de desarrollo, sean o no expertos HCI. Existen coincidencias entre los resultados obtenidos pero todavía no hay un consenso para el uso de técnicas HCI en proyectos ágiles. Es evidente la dificultad que implica el proporcionar una línea única a seguir al momento de incorporar técnicas de usabilidad en el desarrollo de proyectos de software. Cada uno tiene características muy particulares por lo que ninguno es igual. A través

de la revisión de literatura se observa la variedad de estudios que proponen estrategias para implementar técnicas de usabilidad en el desarrollo ágil. Sin embargo se percibe la necesidad de guías más prácticas que se adapten al dinamismo que exige el desarrollar software utilizando prácticas ágiles. Siguiendo los principios fundamentales de la filosofía los equipos ágiles se enfocan en la entrega continua de software valioso para el usuario. La implementación de procesos o actividades adicionales que pudieran implicar inversión de tiempo representan para ellos una influencia que eventualmente afecta la calendarización de entregas. El hecho de que todo el equipo ágil cuente con la información necesaria sobre usabilidad cobra mayor relevancia por el dinamismo y constante comunicación que por naturaleza implica el proceso de desarrollo. Por lo tanto, la provisión de una sencilla y clara visión de cómo integrar técnicas HCI en el proceso de desarrollo ágil puede ayudar a los equipos a mejorar en su software el nivel de usabilidad que este ofrece.

La estructura de este trabajo se define para contribuir a este tema que aún resulta un problema de investigación abierto. Nuestra investigación pretende contribuir proporcionando la información necesaria a los equipos ágiles para que puedan incorporar ciertas técnicas de HCI en sus procesos. Tal como se verá en el capítulo siguiente el objetivo es generar ciertos artefactos de documentación que permitan a los desarrolladores utilizar de manera eficaz técnicas HCI dentro de su proceso de desarrollo ágil.

Capítulo 3 ENFOQUE DE LA SOLUCIÓN

3.1 Introducción

Tal como se ha visto en el capítulo 2, distintos autores han presentado evidencia sobre las adaptaciones que se aplican a las técnicas HCI para incorporarlas en sus procesos ágiles. Y también viceversa, adaptaciones que el proceso ágil requiere al usar las técnicas necesarias para incorporar usabilidad en el software. Los equipos ágiles no siempre cuentan con expertos en el área de HCI que apliquen o adapten las técnicas. Por lo que, es habitual que los mismos desarrolladores asuman este rol. En esta línea una de las principales preocupaciones que subrayan las comunidades que abogan por mantener la usabilidad en los productos es la falta de guías concretas para apoyar a los equipos ágiles en la integración de este atributo dentro de su contexto de desarrollo.

Sohaib y Khan, (Sohaib et al., 2010) señala que debido a la falta de tácticas y prácticas que ayuden a los desarrolladores en el uso de prácticas UCD, es complicado para las organizaciones de software que utilizan métodos ágiles, tratar la usabilidad dentro de sus procesos. En definitiva, surge la necesidad del equipo, en particular los desarrolladores, de contar con información fácil de entender y concisa que les permita incorporar técnicas HCI de manera sencilla y eficiente en sus procesos ágiles.

Una de las técnicas que ha llamado la atención de diversas áreas incluidas la IS y la comunidad HCI para documentar y compartir buenas prácticas son los patrones. El uso de patrones permite que el conocimiento que contienen pueda ser reutilizado en situaciones similares o bien proporcionar un punto de partida para la solución en contextos similares.

El uso de patrones para mejorar la usabilidad en proyectos ágiles fue propuesto por Kane hace más de una década (Kane, 2003). Kane planteó la utilidad de patrones para la interfaz de usuario de forma que los desarrolladores ágiles puedan considerar factores de usabilidad. El concepto de patrón será también, el que se utilizará en esta tesis para materializar las guías propuestas de incorporación de técnicas de HCI en un proceso ágil.

En la sección 3.2 nos enfocamos a conocer un poco más sobre el concepto de patrón describiendo las características y algunos ejemplos de este importante artefacto, con el fin de tener un panorama general sobre las ventajas que representa. La sección 3.3 aborda la estrategia de solución describiendo cada uno de los puntos que se tomaron en cuenta para desarrollar nuestra propuesta: las adaptaciones de las técnicas HCI en un ambiente ágil, selección de las técnicas a describir por medio de patrones y el formato del patrón a utilizar.

3.2 Patrones

En nuestros días es bien conocida la utilidad que ha representado el concepto “patrón” en diferentes áreas, por no decir todas, donde la reutilización de buenas prácticas resulta una contribución importante como apoyo en la creación de productos con características similares. El término patrón fue acuñado por Christopher Alexander quien desde su perspectiva todos los procesos y tareas de construcción exitosas tienen una característica invariable y fundamental (Christopher Alexander, 1979). Un paso más allá de los patrones está el lenguaje de patrones conformado por un conjunto de patrones que se relacionan entre sí dando lugar a una estructura interconectada que proporcionan en palabras de Alexander C. “...*el poder de crear una variedad infinita de nuevas y únicas construcciones*”.

Aunque Alexander C. no establece una definición única de patrón, en el área de la IS, como en otras más, se han aprovechado los diferentes enfoques alrededor de la conceptualización básica de este método. En general Alexander C. describe un patrón como una manera para reutilizar el conocimiento de diseño a partir de un modelo problema-solución cuya solución propuesta al utilizarse permita enfrentar, resolviendo o minimizando, aquellos problemas observados. Haciendo referencia al trabajo de Alexander C. (1977), Seffah et al. (Seffah et al., 2012) señalan tres elementos esenciales de un patrón: contexto, problema y solución. El contexto describe un conjunto de situaciones en las que el patrón puede aplicarse; el problema considera los objetivos y

limitaciones que pueden darse dentro del contexto; la solución describe los elementos que constituyen el patrón, la relación entre estos elementos, sus responsabilidades y colaboración.

Al ser un patrón el punto de referencia y apoyo para la creación de algo, es importante describir sus elementos de tal forma que realmente la información que proporciona resulte de utilidad tanto para quienes lo usarán en la construcción del producto como para aquellos que están relacionados con el proceso para el que se utiliza. Stakeholders, clientes o incluso usuarios finales pueden participar en determinados momentos durante el desarrollo del software en etapas donde se utilicen patrones por lo que es fundamental una clara descripción de éste. Como parte de su lenguaje de patrones, Alexander utiliza un formato para cada patrón que considera los elementos que se presentan en la tabla C3-1.

Tabla C3-1 Resumen de la estructura de patrón utilizada por Alexander C. (C Alexander et al., 1977)

Campo	Significado
Nombre	Nombre del patrón
Figura	Representación para ejemplificar una instancia del patrón con un párrafo corto estableciendo el contexto de ésta
Contexto del patrón	Descripción breve del contexto del patrón
Descripción del problema	Descripción concisa del problema. Incluyendo motivos, respaldo empírico, evidencia (motivo del patrón) y las fuerzas involucradas en la resolución del problema.
Solución	Descripción de la solución
Diagrama	Diagrama solución considerando los patrones relacionados
Patrones relacionados	Patrones con los cuáles el patrón descrito guarda alguna relación de colaboración para obtener determinado resultado

El área de la ingeniería de software cuenta con varias colecciones de patrones. Dentro de las aportaciones más relevantes se reconocen los siguientes:

Patrones de diseño

Con el fin de contribuir al desarrollo orientado a objetos, Gamma et al. (1994) utilizan el concepto de patrones como una forma de registrar y compartir su experiencia en diseño de una manera que permita reutilizar la solución que dieron en determinadas situaciones. En Gamma et al. (Gamma, Helm, Johnson, Vlissides, & Booch, 1994) se refieren los patrones de diseño como “*descripciones de objetos de comunicación y clases que son personalizadas para resolver un problema general de diseño en un contexto particular*”. Presentan en su trabajo una lista en forma de catálogo de aquellos patrones de diseño que identifican como los más comúnmente utilizados en el desarrollo orientado a objetos. De acuerdo a los autores, un patrón de diseño sistemáticamente nombra, explica y evalúa un diseño general que conduce un problema recurrente de diseño en sistemas orientados a objetos. Gamma et al. (1994) identifican cuatro elementos esenciales en un patrón: nombre del patrón, el problema, la solución y las consecuencias.

La descripción de un patrón de diseño considera información sobre: nombre del patrón y clasificación, intención del patrón, otro nombre con el que es conocido el patrón (en inglés Also Known As), motivación, aplicabilidad, estructura, participantes. También incluye datos sobre colaboraciones, consecuencias, implementación, ejemplo de código, usos conocidos y patrones relacionados. La figura C3-1 muestra como ejemplo información resumida del patrón de diseño “Observer”.

Patrones de análisis

Los patrones de análisis sirven de ayuda para definir los requerimientos del software. Se derivan de la idea de que en la etapa de análisis se pueden emplear plantillas para problemas de modelado comunes. El principal propósito de los patrones de análisis es permitir el descubrimiento de los requisitos, más que sugerir formas de

satisfacerlos (Kodaganallur & Shim, 2006). Fowler et al. (Fowler, 1996) describe los patrones de análisis como modelos conceptuales de negocios. Modelos que representan la manera en la que las personas de este campo visualizan los negocios más que la forma en la que un sistema de computación es diseñado.

Aunque no se indica de manera específica un formato o plantilla utilizada para la descripción de los patrones, en general identificamos cuatro partes en la estructura que utiliza Fowler para presentarlos: problema, solución, ejemplo(s) y principio de modelado asociado. Diferentes tipos de diagramas se utilizan como apoyo principalmente en la descripción del problema y la solución. A continuación en la figura C3 – 2 tenemos como ejemplo una descripción resumida del patrón “Quantity” parte de los patrones de análisis del catálogo presentado en Fowler (1996).

Figura C3-1 Ejemplo patrón de Diseño (Gamma et al., 1994)

Cantidad (Quantity)

La forma más simple y común de registrar medidas en los sistemas de cómputo es registrar un número en un campo designado para una medida en particular. Un problema con este método es que utilizando un número para representar la altura de una persona no es muy apropiado. Qué significa decir que mi altura es 6 o que mi peso es 185? Para darle sentido al número, necesitamos unidades.

En este contexto un concepto muy útil es *cantidad*. Elemento que incluye apropiadas operaciones aritméticas y comparativas.

Ejemplo: Podemos representar un peso de 185 libras como una cantidad con monto de 185 y unidad de libras.

Ejemplo: \$80 puede ser representado como una cantidad con un monto de 80 y como unidades dólares americanos.

Principio de modelado. Cuando múltiples atributos interactúan con un comportamiento que puede ser utilizado en varios tipos, combinar los atributos dentro de un nuevo tipo fundamental.

Figura C3-2 Ejemplo de Patrón de Análisis (Fowler, 1996)

Patrones de arquitectura

Un patrón en arquitectura de software describe un problema de diseño recurrente que surge en contextos específicos de diseño, y presenta un probado esquema genérico para la solución. Este esquema solución se especifica describiendo los componentes que lo componen, sus responsabilidades y relaciones entre ellos y las formas en las cuales colaboran entre sí (Buschmann et al., 1996). Los patrones arquitectónicos expresan esquemas fundamentales de organización estructural para sistemas de software. Proporcionan un conjunto de subsistemas predefinidos, especifican sus responsabilidades e incluyen reglas y líneas guía para organizar las relaciones entre ellos (Buschmann et al., 1996).

Basados en la estructura contexto-problema-solución los autores consideran en la plantilla para la descripción de sus patrones información sobre: nombre, ejemplo, contexto, problema, solución, estructura, implementación y ejemplos resueltos. En la figura C3-3 tenemos algunos de los datos considerados en la descripción del patrón Model view-controller (MVC).

Figura C3-3 Ejemplo de un patrón de arquitectura (Buschmann et al., 1996)

Los patrones han sido ampliamente utilizados en el campo de la ingeniería de software donde se cuenta con diversas colecciones y catálogos.

Durante los últimos años en el área de HCI el concepto de patrón y lenguajes de patrones también ha despertado particular interés. Resulta atractiva la característica de reusabilidad que los patrones ofrecen para utilizarse sin mayor problema en contextos similares.

Patrones de HCI

(Seffah et al., 2012) apuntan una importante relación entre los intereses indicados por Alexander C. de sus patrones y la filosofía HCI: crear productos usables y agradables para los usuarios. Dicha relación permite visualizar la viabilidad de aprovechar las ventajas ofrecidas por los patrones para compartir buenas prácticas de diseño de la interacción. Seffah et al. (2012) hacen referencia al trabajo de Alexander C. con el que indica que las prácticas tradicionales de diseño arquitectónico fallan al crear productos que cubran las necesidades reales de los usuarios. Esto es, al no tomar en cuenta las opiniones de los usuarios y no tener el conocimiento suficiente sobre el usuario los productos desarrollados no son los esperados por éste. En los ejemplos presentados previamente observamos que se describen con el enfoque orientado a objetos, esto es una característica de la mayoría de los patrones propuestos en la literatura de IS (Seffah, 2015). Si bien la mayoría de los patrones se describen con este enfoque también pueden describirse de diferente manera manteniendo el nivel de abstracción necesario que debe considerar un buen patrón de diseño. Este es el caso de los patrones de HCI.

En HCI diferentes catálogos de patrones han sido presentados y están disponibles como los de Tidwell (Tidwell, 2005) que presenta patrones para el diseño de la interfaz. En la figura C3-4 se observa como ejemplo la descripción resumida del patrón “Clear entry points” presentado por Tidwell (2005).

Clear entry points

Figure 3-8. From <http://ingdirect.com>

What. Present only a few entry points into the interface; make them task-oriented and descriptive.

use when. You're designing a task-based application, or any other application used largely by first-time or infrequent users. It's also helpful for some web sites. But if the application's purpose is clear to basically everyone who starts it, and if most users might be irritated by one more navigation step than is necessary (like applications designed for intermediate- to-expert users), then don't use it.

Why. Some applications and web sites, when opened, present the user with what looks like a morass of information and structure: lots of tiled panels, unfamiliar terms and phrases, irrelevant ads, or toolbars that just sit there disabled. They don't give the hesitant user any clear guidance on what to do first. "Okay, here I am. Now what?"

For these users' sake, list a few options for getting started. If those options match a user's expectations, he can confidently take one of those options and begin work—this contributes to Instant Gratification (Chapter 1). If not, at least he knows now what the application actually does because you've defined the important tasks or categories right up front. You've made the application more self-explanatory.

How. When the site is visited, or the application started, present these entry points as “doors” into the main content of the site or application. From these starting points, guide the user gently and unambiguously into the application until he has enough context to continue by himself.

Examples

Small-screen devices benefit from this pattern, too. Most PDAs and cell phones have a Hub and Spoke design, in which applications are chosen from a starting screen.

Figure 3-9. Google is best known for doing one thing amazingly well. Its home page design focuses users' attention on that one thing: you can't miss that search box! The other stuff is ranked into secondary (such as Web or Images) and tertiary (such as Advanced Search, Patents, and Labs). Like ING Direct's site

Figura C3-4 Ejemplo patrón de diseño de interfaz (Tidwell, 2005)

Tidwell describe los patrones como características estructurales y de comportamiento que mejoran la “habitabilidad” de algo como una interfaz de usuario, un sitio web, programa orientado a objetos, o incluso un edificio. Tidwell considera que los patrones hacen las herramientas más útiles y usables. Apunta que los patrones no son simples reglas o heurísticas pero sí una descripción de mejores prácticas dentro de un dominio dado de diseño (Tidwell, 2005).

Otra importante colección de patrones en HCI es “Amsterdam” compuesta por patrones de diseño de interacción presentado por Welie M. (Welie, 2008) y disponible on-line (www.welie.com). Este grupo de patrones se describe especificando el problema, la solución, cuándo usarlo, cómo, porqué y ejemplos.

En el área de metodologías ágiles Da Silva et al. (Bertholdo et al., 2014) y Bertholdo et al. (Bertholdo, Kon, & Gerosa, 2016) describen y proponen un grupo de patrones de usabilidad ágil. Esto es, patrones que permiten realizar ciertas prácticas y actividades ágiles durante las etapas del ciclo UCD definido en el estándar ISO 13407.

En (Bertholdo et al., 2014) se proponen patrones a utilizar durante las primeras etapas del ciclo UCD mientras que (Bertholdo et al., 2016) proponen patrones para las etapas finales de dicho ciclo. La agrupación de los patrones propuestos se observan la tabla C3-2 y C3-3.

En ambos grupos de patrones tanto aquellos para las etapas iniciales como las finales del ciclo UCD la plantilla para su descripción considera los campos: contexto, problema (incluyendo las principales fuerzas involucradas), solución y ejemplo.

Tabla C3–2 Patrones de usabilidad ágil para las primeras etapas del ciclo UCD propuestos en (Bertholdo et al., 2014)

Etapa del proceso UCD	Patrones propuestos para las primeras etapas del ciclo UCD
Identificar necesidades para el diseño centrado en el humano	<ul style="list-style-type: none"> • Sprint cero (Sprint Zero) • Un sprint por adelante (One sprint ahead) • Especialistas UX como product owners (UX specialists as product owners) • El tiempo de los usuarios es valioso (Users Time is Valuable) • Bloques paralelos (Parallel tracks) • Especialistas UX como miembros a tiempo completo del equipo ágil (UX specialists as Full-Time Member of the Agile Team)
Especificar el contexto de uso	<ul style="list-style-type: none"> • Poco diseño por adelantado (Little Design Up Front) • Plan de contacto con los usuarios (Contact Plan of Users)
Especificar requerimientos	<ul style="list-style-type: none"> • Historias de usuario (User Stories) • Más colaboración, menos documentos (More Collaboration, Less documents) • Prototipos como especificación (Prototypes as specification)

Tabla C3–3 Patrones de usabilidad ágil para las etapas finales del ciclo UCD propuestos en (Bertholdo et al., 2016)

Etapa del proceso UCD	Patrones propuestos para las etapas finales del ciclo UCD
Crear soluciones de diseño	<ul style="list-style-type: none"> • Prototipos de baja fidelidad • Prototipos de alta fidelidad • Estudio de diseño • Diseño colaborativo y participativo
Evaluar el diseño	<ul style="list-style-type: none"> • Pruebas con usuarios • Evaluación por inspección • Método RITE • Pruebas de aceptación

En la figura C3-5 tenemos un ejemplo resumido del patrón llamado *Sprint Zero* que corresponde al grupo de patrones para la primera etapa UCD: Identificar necesidades para el diseño centrado en el Humano.

Pattern: Sprint Zero

Contexto: El sprint cero es una práctica que busca una mejor definición del panorama general del proyecto. Es un estado antes del inicio de la implementación del proyecto.

Problema: Falta del “big picture” o panorama general del sistema al principio del desarrollo con métodos ágiles. Los elementos del equipo podrían trabajar sobre funcionalidades equivocadas debido a que no entendieron cuáles fueron las prioridades reales.

Solución: Un sprint corto antes de la implementación del código para definir un panorama general del producto, metas generales, planeación aproximada de los siguientes sprints y definir los principios de diseño.

Ejemplo: Chamberlain et al. sugieren que un sprint cero debe estar listo antes de la planificación, de esta forma los aspectos de usabilidad pueden ser discutidos durante el reléase o sprint planning. Belchev y Baker reportan contextual inquiry siendo usado como entrevistas uno a uno conducidas en el área de trabajo de los usuarios para observar el trabajo en proceso.

Figura C3-5 Ejemplo de un patrón de usabilidad ágil (Bertholdo et al., 2014)

Por su parte, (Salah, Paige, & Cairns, 2015) identifican determinados retos que implica la integración del desarrollo de software ágil y el diseño centrado en el usuario (en inglés Agile and User Centered Design Integration o AUCDI). Tomando como base los factores y estrategias de éxito observadas en los resultados de su estudio Salah et al. (2015) proponen patrones que faciliten a los profesionales de la industria sobrellevar los retos del proceso AUCDI. El formato para describir los patrones se compone del nombre del patrón, identificador numérico del patrón, contexto, problema, fuerzas y solución. Los patrones propuestos son los siguientes:

- Menos es más! (Less is More!)
- Sesiones de prueba de usabilidad junto con las pruebas de desarrollo ágil (Usability Testing Sessions Alongside Agile Development Tests)
- Hazlo RITE (Do it RITE)
- Desarrolladores y profesionales UCD (Developers and UCD practitioners)
- Sitios web como documentación ligera (Websites as lightweight documentation)

Como ejemplo de la estructura de los patrones mencionados en la figura C3-6 se presenta un resumen del patrón “Desarrolladores y profesionales UCD” (en inglés Developers and UCD practitioners).

Pattern 4: Desarrolladores y profesionales UCD**Contexto:**

Grandes organizaciones pueden permitirse un profesional UCD dedicado por equipo o algunos equipos, lo cual facilita la ubicación de recurso y consistencia del diseño, sin embargo, en pequeñas organización los profesionales UCD son usualmente compartidos entre un número de equipos o trabajan a tiempo parcial.

Problema:

La naturaleza específica del proceso ágil puede dar lugar a una carga extra en los profesionales UCD quienes se ocupan de un número de equipos o trabajan a tiempo parcial. Los profesionales UCD en equipos ágiles son asignados a muchos equipos, acudiendo a largas reuniones y las cortas líneas de tiempo no les proporcionan suficiente tiempo para terminar su trabajo. Además, el constante cambio entre diferentes proyectos puede causar una severa pérdida de productividad para los miembros del equipo de diseño.

Fuerzas:

- El número de profesionales UCD en la mayoría de las compañías de software es limitado
- El tiempo de los profesionales UCD se divide entre varios proyectos y esto dificulta para ellos su disponibilidad para asistir en las frecuentes sesiones de evaluación de usabilidad.

Solución:

El incremento en la carga de los profesionales UCD en equipos ágiles puede ser abordado eligiendo un número de desarrolladores quienes estén interesados en UCD y asesorarlos hasta que sean capaces de realizar evaluaciones de usabilidad y revisiones de la interfaz de usuario. Esto puede generar un decremento en la carga de trabajo de los profesionales UCD y que los desarrolladores obtengan más experiencia en las evaluaciones de usabilidad y revisiones de interfaz lo cual da lugar a mejorar la calidad del trabajo.

Figura C3-6 Ejemplo de patrón para AUCDI (Salah et al., 2015)

Dentro de los patrones de usabilidad también han sido propuestos lenguajes de patrones para proporcionar a los diseñadores la información que les permitan definir diseños en contextos específicos y relacionados. En esta línea (Seffah & Taleb, 2012) presentan UPADE web language y Experience pattern language.

Patrones de diseño de usabilidad

Además de las soluciones propuestas enfocadas al diseño de la interacción también se tienen propuestas importantes en la IS que proponen mecanismos de apoyo a los equipos de desarrollo para incorporar la usabilidad. (L Carvajal et al., 2013) describen lo que nombran líneas guías de usabilidad (en inglés usability guidelines) para el desarrollo de software. El objetivo de estas directrices es proporcionar a los desarrolladores de software recomendaciones de diseño para ayudarles a incorporar de manera confiable y repetible ciertas funcionalidades de usabilidad (Undo, Abort, Warning etc) al diseñar su sistema. La figura C3-6 muestra la estructura de las líneas guía conformada por siete artefactos agrupados para visualizar mejor su organización como artefactos de análisis y artefactos de diseño. Aunque para fines de descripción los artefactos se agrupan como se mencionó previamente, los desarrolladores pueden utilizarlos como lo requieran, independientemente de la clasificación asignada en las categorías de análisis o diseño.

Figura C3-7 Patrón de diseño para "Go Back" (L Carvajal et al., 2013)

Uno de los artefactos de las líneas guía es el de las responsabilidades de usabilidad de los componentes de diseño de alto nivel (en inglés high-level design component responsibilities for usability). Este artefacto proporciona una descripción abstracta de las responsabilidades del sistema (componentes de software). En la figura C3-7 observamos un ejemplo de este artefacto.

System Resp.	High-Level Design Component Responsibilities
W_SR-1 Be aware of damaging actions	There must be a <i>wrapping component</i> for each <i>domain component</i> that includes at least one sensitive method. This component mimics the structure of the <i>domain component</i> , and it will 'sit' between it and any invoking class. All <i>wrapping components</i> must know whether it is safe to invoke actions in their <i>DomainComponent</i> . For example, if 'SalesItem' has a sensitive action, it must be renamed 'SalesItemDomain', for example, and this way its <i>wrapping component</i> can take the name 'SalesItem'. When called, actions in the <i>wrapping component</i> can respond that they <i>cannot yet</i> be executed, issuing a warning. This component is responsible for knowing which actions trigger which kind of warning and for triggering them when needed.
W_SR-2 Notify	If the issued <i>warning</i> is a <i>notification</i> , it must be sent to the <i>UI</i> . The call is then returned to the <i>wrapping component</i> for execution. As it is now safe, the <i>wrapping component</i> calls the action in the <i>DomainClass</i> .
W_SR-3 Request confirmation	If the issued <i>warning</i> is a <i>confirmation</i> , it must also be sent to the <i>UI component</i> . In this case, however, it must wait for the user to okay the invocation. After okaying, the invocation is returned to the <i>wrapping component</i> for execution. As it is now safe, the <i>wrapping component</i> invokes the action in the <i>DomainClass</i> .
W_SR-4 Request authentication	Finally, if the issued <i>warning</i> is an <i>authentication</i> , it must also be sent to the <i>UI component</i> . The <i>UI component</i> will then go through any necessary steps to cross-check credentials. After user authentication, the call is returned to the <i>wrapping component</i> for execution. As it is now safe, the <i>wrapping component</i> calls the action in the <i>DomainClass</i> .
W_SR-5 Display warning	The <i>UI Component</i> is responsible for displaying the <i>warning</i> information and for receiving and processing the necessary user input to satisfy the given <i>warning</i> (if applicable)

Figura C3-8 Artefacto High-Level Design Component Responsibilities ejemplo "Warning"

3.3 Estrategia de solución

Como hemos visto, los patrones relacionados con la usabilidad que existen en la literatura aunque útiles para mejorar este atributo de calidad, no concretan guías específicas para la incorporación de técnicas HCI dentro del proceso ágil o viceversa. Este trabajo propone dos patrones como una aportación a la solución del problema identificado en capítulos anteriores sobre la falta de información que sirva como apoyo a los equipos ágiles para incorporar en su proceso técnicas HCI de manera sencilla y eficiente.

Para determinar mejor la estructura de la solución propuesta consideramos tres puntos:

- Adaptaciones aplicadas a las técnicas de interacción persona-ordenador al incorporarlas en el entorno ágil
- Selección de las técnicas a documentar en los patrones
- Estructura del patrón a utilizar.

La definición de los patrones propuestos se desarrolla a partir de los resultados obtenidos en nuestra revisión de literatura descrita en el capítulo anterior.

3.3.1 Adaptaciones en el uso de las técnicas HCI dentro del entorno ágil

El estudio de la literatura ha permitido observar que las técnicas de usabilidad han sido incorporadas con ciertas adaptaciones en los proyectos ágiles, obteniendo resultados exitosos en cuanto al nivel de usabilidad y UX. Desde nuestro punto de vista enfocamos las adaptaciones más relevantes en quién, cómo y cuándo se aplican las técnicas HCI:

- En un entorno de desarrollo ágil las tareas y actividades HCI no necesariamente se realizan por expertos en la materia como originalmente establece la filosofía HCI. Los desarrolladores del equipo ágil desempeñan en ocasiones el rol de los expertos en HCI aún que no cuenten con la formación indispensable para ello.
- Técnicas HCI que originalmente requerían aplicarse de acuerdo a ciertos lineamientos específicos y muy particulares, al ser incorporadas en un entorno ágil se aplican con más flexibilidad adaptándose al dinamismo de tiempo y espacio que conllevan las aproximaciones ágiles.
- La incorporación de las técnicas HCI se realiza en diferentes momentos durante el desarrollo del software adaptándose al ritmo de un proceso de ágil. Técnicas y artefactos que solamente podían aplicarse y/o definirse en su totalidad en determinado momento de un desarrollo de software, con las nuevas metodologías no necesariamente se definen por completo y/o no se aplican únicamente en determinado momento.

El concepto original HCI de cuya evolución se hace más evidente en proyectos ágiles, es sobre los miembros del equipo responsable de realizar las tareas de diseño de la interacción. Originariamente, dichas tareas eran realizadas únicamente por los expertos en el área HCI. Como se ha indicado ya, actualmente también los desarrolladores pueden desempeñar el rol de diseñador de la interacción o realizar las tareas pertinentes del diseño de interfaz. Debido a esto cobra aún mayor importancia el contar con información necesaria y válida sobre el uso de técnicas HCI en proyectos ágiles. Información descrita de tal forma que permita a los desarrolladores aplicar las herramientas para el diseño de la interacción de una manera práctica y sencilla a pesar de no ser expertos en el área, consiguiendo un buen nivel de usabilidad en las aplicaciones. Por otro lado, las prácticas de HCI usadas en un proceso ágil de desarrollo se aplican con ciertos cambios, incluso aquellas cuyo proceso de aplicación es iterativo, obteniendo de esta manera los beneficios esperados de usabilidad. Uno de los factores clave para aprovechar las ventajas de las técnicas de usabilidad en un proceso de desarrollo es incorporarlas en el momento adecuado. En nuestro estudio identificamos que el principal enfoque con el que se incorporan las técnicas HCI en un proyecto de desarrollo ágil es el “Minimal up-front” esto es, al principio del proyecto se obtiene únicamente la información necesaria que permita iniciar con el diseño, conforme se va

avanzando con las iteraciones el diseño inicial se revisa y mejora. Con este objetivo en mente de proporcionar información de apoyo, consideramos definir patrones que ayuden a los equipos ágiles a incorporar en su proceso técnicas HCI.

3.3.2 Técnicas HCI: Persona y Usability Testing

Como hemos visto en el capítulo anterior existe evidencia sobre las prácticas HCI que son más utilizadas en los proyectos ágiles (T. Da Silva et al., 2011) (Salvador et al., 2014) (Brhel et al., 2015). Comunicación, simplicidad, rápida y fácil descripción son algunos de las principales consideraciones que se manejan para decidir cuál es la técnica HCI ideal a utilizar en el proceso ágil. Tomando en cuenta los resultados de nuestra revisión de literatura tenemos como las técnicas más utilizadas en desarrollos ágiles las siguientes: prototipos, usability testing, escenarios, persona, contextual inquiry y heuristic evaluation.

Para fines de nuestro estudio organizamos las técnicas anteriores de acuerdo a los siguientes criterios:

- aquellas enfocadas en conocer y entender al usuario para definir las necesidades reales de éste y
- las que se utilizan para obtener retroalimentación del usuario sobre los diseños de interfaz definidos.

En la primera clasificación podemos considerar a persona y contextual inquiry, de estas técnicas la de persona enfatiza en el hecho de aprender sobre el usuario conociendo sus actividades, sus metas y comportamiento. Persona abarca un poco más allá de conocer el comportamiento del usuario únicamente dentro del marco de sus actividades en su entorno de trabajo.

Por el lado de las técnicas que obtienen retroalimentación del usuario sobre el diseño de la interfaz, tenemos que los prototipos sirven de apoyo a su vez a otras técnicas para validar los diseños y funcionalidad de las aplicaciones a implementar, no tienen que ser evaluados por los usuarios finales del software aunque es muy recomendable. Usability testing es una técnica que busca obtener retroalimentación sobre el diseño de la interacción, para ello se apoya en otras técnicas como prototipos y “thinking aloud”¹. Esta técnica hace hincapié en la necesidad de realizar las pruebas de usabilidad con los usuarios finales del software y los observadores que aplican las pruebas no necesariamente requieren ser expertos profesionales en HCI. El proceso iterativo de estas pruebas permite una fácil integración en el ambiente de desarrollo ágil adaptándose al ritmo de este entorno. Las pruebas heurísticas se aplican por evaluadores con experiencia y conocimiento de las heurísticas que serán pauta para evaluar la interacción, por lo que no resulta sencillo ni recomendable aplicarlas sin las habilidades necesarias (Jakob Nielsen, 1993). El proceso también es iterativo. Los escenarios son técnicas narrativas que ayudan a entender el cuándo, dónde y cómo la situación a evaluar tiene lugar (Sharp et al., 2007).

Para la elección de las técnicas sobre las cuales definir los patrones se consideraron los criterios de organización mencionados en esta sección y las adaptaciones más relevantes aplicadas a las técnicas HCI cuando se aplican en un entorno ágil mencionadas en la sección 3.3. También se tomó en cuenta si al aplicar la técnica es necesaria la participación de usuarios finales o representantes de éstos. Resumiendo los criterios tenemos:

- Las técnicas HCI pueden aplicarse por los mismos desarrolladores y no necesariamente por un experto HCI
- El uso de las técnicas HCI durante el desarrollo del proceso ágil puede realizarse en diferentes momentos de acuerdo al objetivo por el que se aplican. En este criterio consideramos los dos tipos de fases descritas en el capítulo 2 sección 2.2.2, fase exploratoria y de refinamiento.
- Las técnicas pueden aplicarse con el objetivo de obtener requerimientos e identificar situaciones relacionadas con el diseño de la interacción a definir. También pueden aplicarse para validar el diseño de la interacción definida.

¹ Técnica por medio de la cual el usuario va describiendo en voz alta las acciones que va realizando mientras interactúa con la interfaz de la aplicación.

- La participación de los usuarios finales al aplicar las técnicas en un entorno ágil no necesariamente es obligatoria sin embargo siguiendo los valores de la filosofía sí se valora ampliamente su participación o la de usuarios representativos de éstos.

A manera de resumen en la tabla C3-4 presentamos información relacionada con los criterios mencionados previamente sobre las técnicas identificadas como las más utilizadas en proyectos ágiles. La columna “Técnica en HCI” corresponde al nombre de la técnica; “Experto HCI” indica si la técnica solamente puede aplicarse por un experto en el área de HCI; la columna “Momento de incorporación” indica la fase en la que se usa la técnica durante el proceso ágil; en la columna “Objetivo” identificamos las técnicas que se aplican para obtener los requerimientos y aquellas que se aplican para validar el diseño de interacción del usuario y mejorarla a partir de la retroalimentación si es necesario; “Usuarios finales” indica si la técnica necesariamente involucra usuarios finales o representantes de éstos al momento de aplicarse.

Tabla C3-4 Resumen de características técnicas HCI más utilizadas en proyectos ágiles

Técnica HCI	Experto HCI	Momento de incorporación	Objetivo	Usuarios finales
Persona	No necesariamente	Exploración / refinamiento	Requerimientos	Si
Contextual Inquiry	No necesariamente	Exploración	Requerimientos	Si
Prototipos	No necesariamente	Exploración / refinamiento	Validación	No necesariamente
Usability Testing	No necesariamente	Exploración / refinamiento	Validación	Si
Heuristic Evaluation	Si	Refinamiento	Validación	No
Scenarios	No necesariamente	Exploración / refinamiento	Validación	No necesariamente

En un entorno ágil, casi la totalidad de técnicas pueden ser aplicadas por un no experto en HCI. De las técnicas analizadas Contextual inquiry se utiliza únicamente en la etapa de exploración mientras que Heuristic Evaluation solamente en la de refinamiento. La mayoría se utiliza tanto en la etapa de exploración como de refinamiento. Considerando el objetivo al aplicar las técnicas, como se mencionó previamente persona y contextual inquiry se aplican para obtener requerimientos y las técnicas de prototipos, usability testing, heurístico evaluation y escenarios se aplican para validar el diseño de interacción ya definido. En cuanto a la participación de los usuarios finales persona, contextual inquiry y usability testing son técnicas que incluso en un entorno ágil destacan la participación de los usuarios finales al usarlas. Tomando en cuenta el objetivo de aplicación de las técnicas – requerimientos o validación – nos enfocamos en aquellas que además de no ser necesariamente aplicadas por expertos HCI pudieran usarse en ambas etapas tanto de exploración como de refinamiento y en las que la participación de usuarios finales o representantes de éstos es fundamental. De las dos técnicas que se utilizan con el objetivo de obtener requerimientos para la definición del diseño de la interacción se eligió persona y de las técnicas cuyo objetivo es validar el diseño de la interacción se eligió usability testing. Las dos técnicas elegidas se remarcan con fondo color gris en la tabla C3-2.

3.3.3 Patrones de producto

Como parte de su modelo arquitectónico para mejorar la eficiencia de uso de los procesos de software, Sánchez-Segura et al. (Sánchez-Segura et al., 2010) proponen un formato que denominan *patrón de producto* para describir buenas prácticas. Aunque la estructura propuesta surge de las conceptualizaciones de los patrones presentados por Alexander C. los autores señalan como principal diferencia el que los patrones de producto reúnen el conocimiento de expertos en ingeniería de software para obtener un producto específico de software. Como vimos en la sección 3.2 los diferentes patrones y catálogos de patrones más relevantes en IS y HCI proporcionan información para obtener a través de la replicación de las actividades descritas determinadas características del software ya sea en el análisis, diseño del sistema, diseño de la interfaz, o bien para aplicar buenas prácticas de desarrollo sobre ciertas situaciones. En el caso de un patrón de producto una vez aplicado se

obtiene como resultado un producto en sí, por ejemplo una historia de usuario, un escenario, etc. En nuestro trabajo nosotros obtenemos efectivamente un producto del desarrollo de software: la descripción de una persona o las pruebas de usabilidad por ello la elección de este formato para generar nuestra solución propuesta. La estructura del formato describe en detalle el proceso para obtener los productos mencionados y como incorporarlos en el proceso ágil por lo que el concepto de patrón de producto se adapta a nuestro objetivo de proporcionar una descripción lo suficientemente sencilla pero que a la vez contenga la información necesaria dirigida a los desarrolladores ágiles.

La descripción de un patrón de producto considera el esquema básico contexto-problema-solución respaldado por diferentes campos para proporcionar información lo suficientemente detallada para la reutilización de la experiencia documentada. Los campos que describen el patrón de producto son: Name, Related pattern, Initial Context, Result Context, Problem, Forces, Solution, Roles, Entries, Lessons Learned, Templates, Examples, Exit, Collaboration, Capability level, Information resources y basic knowledge. Hacemos referencia a ellos utilizando su nombre en inglés tal y como son descritos en Sánchez-Segura et al. (2010).

En la figura C3-9 se muestra como ejemplo un resumen de la descripción del patrón de producto “Diagrama Caso de Uso” definido de acuerdo a los datos mencionados previamente y publicado on-line.

Figura C3-9 Ejemplo de un Patrón de producto (“Promise - PPL,” 2009)

El patrón de producto demostró ser un apoyo para mejorar la eficiencia de uso de las buenas prácticas en IS, mejorar algunos parámetros de calidad de los productos de software y mejorar la comunicación para reducir el re trabajo a realizar durante el desarrollo de software (Sanchez-Segura et al., 2010). En nuestra solución propuesta inicialmente utilizamos los datos de un patrón de producto tal como se describen originalmente en (Sanchez-Segura et al., 2010). Buscando facilitar la ubicación y uso de la información a utilizar del patrón, los datos se organizaron en formato de tabla como se muestra en la figura C3-10 donde también se observa la descripción del contenido a registrar.

Name	<i>Nombre del patron (puede ser una palabra o frase corta)</i>
Initial context	<i>Descripción de las precondiciones sobre las cuales el patrón es aplicable</i>
Result Context	<i>Contexto resultado al aplicar el patrón</i>
Problem	<i>Descripción del problema en el que busca contribuir la solución del patrón</i>
Forces	<i>Restricciones a considerar para la aplicación del patrón</i>
Solution	<i>Solución propuesta por el patrón para el problema indicado. Se utilizan diagramas para representar el proceso de solución</i>
Roles	<i>Personas o participantes en la aplicación del patrón</i>
Basic Knowledge	<i>Conocimiento básico que se requiere para aplicar el patrón</i>
Entries	<i>Información de entrada o productos previos para aplicar el patrón</i>
Exit	<i>Producto obtenido como resultado de aplicar el patrón</i>
Collaboration	<i>Nombres de los patrones con los que tiene relación el patrón</i>
Capability label	<i>Nivel de madurez del proceso de mejora con el que esté relacionado el patrón</i>
Lessons learned	<i>Experiencias que se han obtenido al aplicar el patrón de producto en proyectos previos</i>
Examples	<i>Ejemplos de la aplicación del patrón</i>
Templates	<i>Plantillas a utilizar durante la aplicación del patrón para obtener el resultado</i>
Information Resources	<i>Recursos de información tales como referencias y documentación que pueden utilizarse para desarrollar este patrón</i>

Figura C3-10 Formato de la descripción del patrón de producto

Para fines de nuestro estudio haremos referencia a los patrones de producto simplemente con el término “patrón” para mantener el concepto generalizado.

Capítulo 4 Formalización de la solución: Patrón “Agile Persona”

4.1 Introducción

A través de las secciones siguientes iremos conociendo los orígenes, características y adaptaciones que se han aplicado a la técnica persona en proyectos ágiles. El objetivo es identificar las mejores prácticas a documentar y compartir con los equipos ágiles para facilitar la incorporación en el entorno ágil de esta técnica de manera sencilla y rápida. En la sección 4.2 se describen las diferentes formas en que la técnica persona se ha definido e incorporado en áreas como la mercadotecnia, HCI y en el desarrollo de proyectos utilizando metodologías ágiles. La sección 4.3 aborda la definición del patrón “Agile persona” describiendo cada uno de los datos considerados en la estructura del patrón. Finalmente en la sección 4.4 se presenta el patrón completo.

4.2 Persona

4.2.1 La idea original: Personas como una herramienta de segmentación de mercado

El concepto de Persona, nace originalmente en el área de la investigación de mercado. En este campo el uso básico de persona surge del concepto de *comunidad de individuos* introducido por Jenkinson (Jenkinson, 1994). El autor señala que un grupo connota una comunidad de individuos, esto es, individuos que tienen algo en común. Jenkinson remarca que los clientes deben ser el centro del pensamiento de la organización y la relación asociativa que conecta a los individuos y la compañía. Especifica la diferencia entre “segmento” y “agrupación”. Segmento implica dividir una población en grupos de acuerdo a factores tales como edad, género, intereses, etc., la meta es obtener soluciones específicas. Agrupar significa fusionar personas para formar una clase de consumidores que comparten características comunes con soluciones a medida. Agrupar requiere más conocimiento íntimo y afinidad entre las personas (Jenkinson, 1994).

Jenkinson presenta los dos siguientes beneficios de la segmentación basada en la comunidad (Jenkinson, 2009):

- Más conocimiento sobre los clientes. Una sofisticada investigación sobre el stakeholder se utiliza para entender y categorizar las necesidades, actitudes, requerimientos y valores emocionales que sustentan diferentes comunidades de clientes.
- Una organización más enfocada al cliente.

Jenkinson propone, entre otras cosas, implementar un enriquecido modelo de segmentación que la compañía debe comunicar al equipo y a la administración convirtiendo los segmentos de la comunidad en gente de la vida real utilizando las descripciones arquetipo como representación de los clientes o personas. El término “personas” se refiere a la descripción arquetipo desarrollada para perfiles de usuario en línea (Jenkinson, 1994).

Ring et. al. (Ring et al., 2008) definen persona como un proceso de perfilado y un modelo de diseño conductual utilizado por los profesionales de diseño digital. Hacen hincapié en la diferencia entre los retratos de matices que son perfiles del consumidor que existen por encima y más allá de cualquier asignación de creatividad específica. Personas, por el contrario, siguen un tipo de usuario a través de una acción muy específica o caso de uso, tales como ir de compras o ver la televisión.

En nuestros días muchas compañías de mercadotecnia han adoptado persona como una importante técnica para segmentación que les ayuda a definir arquetipos para representar a los diferentes tipos de usuario dentro de un conjunto objetivo demográfico, de actitudes o de comportamiento que puede utilizar un sitio, marca o producto de manera similar. Siguiendo el marco de trabajo propuesto por Assael (Assael, 1999), las compañías de mercado pueden ser capaces de capturar una imagen precisa de sus actuales y futuros clientes. La herramienta ha resultado una garantía para implementar exitosas estrategias de marketing por lo que se ha vuelto muy popular en este campo. Las experiencias compartidas por parte de las compañías al utilizar persona han enriquecido el concepto de persona como una metodología para ejemplificar y estudiar los comportamientos de los consumidores.

4.2.2 Persona en HCI

A partir de lo descrito en la sección previa, se infiere el hecho de que en relación al conocimiento de “el cliente” incluido su comportamiento, el campo del marketing y HCI tienen objetivos en común. Marketing se enfoca en conocer su comportamiento y satisfacer las necesidades del cliente. HCI se enfoca en satisfacer las necesidades y metas del usuario aplicando investigación de usuario para diseñar software usable y útil que cubra las necesidades de dicho usuario. Uno de los roles del comportamiento del consumidor presentados en (Loudon & Della Bitta, 1979) es el “usuario”. Para marketing el usuario es la persona más involucrada en el consumo o uso de la compra. Podemos hacer una similitud con el “usuario” en el diseño de software el cual se describe como la persona que está más involucrada directamente en el consumo o uso del software entregado.

“Persona” en HCI fue presentada por Alan Cooper (Cooper, 1999) como parte de las efectivas herramientas mentales consideradas en el método de diseño Goal-Directed Design. Cooper describe persona como una poderosa herramienta de comunicación y para el diseño de interacción. Aunque persona en particular no hace referencia a gente real, la descripción se realiza con un considerable rigor y precisión directamente de observaciones realizadas (Cooper et al., 2007). La técnica se utiliza para definir modelos de usuario que permiten especificar y precisar información sobre necesidades y metas; son arquetipos hipotéticos de usuarios actuales. Una definición de persona encapsula un conjunto distinto de patrones de comportamiento relacionados con el uso de un producto en particular. Dichos patrones se identifican a través del análisis de datos de entrevista respaldados por datos cuantitativos suplementarios según proceda. Para cada patrón de comportamiento significativo identificado es necesario sintetizar los detalles de sus datos (Cooper et al., 2007).

Sharp et al. (Sharp et al., 2007) describen el proceso de persona relacionado con la definición de Cooper como la colección de atributos para un “usuario típico” llamado perfil del usuario. Cualquier producto puede tener un número diferente de perfiles de usuario los cuales son *“descripciones de usuarios típicos del producto en desarrollo representados por seres humanos específicos en los que se enfocan los diseñadores y para los que diseñan el producto”* (Sharp et al., 2007). Para (Pruitt et al., 2010) personas son *representaciones ficticias, específicas y concretas de los usuarios objetivo* que ayudan al equipo y a la organización a estar más enfocados en el usuario.

En el campo del marketing el concepto de persona arroja luz sobre el proceso de ventas y en el área del diseño de interfaz el concepto arroja luz sobre la definición del producto y el proceso de desarrollo (Cooper et al., 2007). En ambos casos el objetivo es identificar características y comportamientos comunes entre los stakeholders del producto: clientes para marketing y usuarios finales para el desarrollo de software. Los segmentos definidos en marketing para aprender e identificar comportamientos de los clientes también son aplicados de manera similar en el diseño de software cuando las personas se van clasificando y priorizando. Las personas que se crean en el diseño de software tendrán comportamientos en común y esto permite segmentarlos y priorizar, evitando representaciones de usuario que eventualmente no serán usuarios representativos.

Cooper (Cooper, 1999) remarca el punto de vista de marketing y el del diseño de interacción relacionado con persona. En la tabla C4-1 podemos observar características de estos dos enfoques identificadas por Cooper (2009).

Tabla C4-1 Características de la técnica persona en Marketing y HCI

Marketing	Interaction design
<ul style="list-style-type: none"> - Diseñar un producto para una persona que cubre necesidades de otros clientes. - Limitando el foco podemos generar cliente fanático - Fidelidad en nuestro mercado objetivo - Después de obtener ruido sobre tu producto, puedes construir sobre ello y extender tu producto dentro de otros segmentos de mercado. 	<ul style="list-style-type: none"> - Nuestra meta es diseñar software que puede doblarse, estirarse y adaptarse a las necesidades del usuario. - Los segmentos de mercado juegan un rol en el desarrollo de persona debido a que estos pueden actuar como filtro inicial para limitar el alcance de entrevistas a personas dentro de los mercados objetivo (Cooper et al., 2007).

Identificar las personas correctas requiere un entendimiento adecuado de las metas de los usuarios incluyendo sus expectativas individuales o motivaciones (Cooper, 1999). Con el enfoque HCI podemos utilizar persona

junto con otras técnicas como escenarios, prototipos, estudios de grupo y encuestas con el objetivo de conocer mejor al usuario, sus necesidades y a partir de ese conocimiento obtener un buen diseño.

Para efectos de nuestro estudio consideramos los enfoques de Cooper et al. (1999 y 2014) y Pruitt et al. (2010) en relación a los pasos del proceso a seguir para construir personas. Cooper señala en (Cooper et al., 2007) que las definiciones de persona son derivadas de patrones identificados al entrevistar y observar a los usuarios y potenciales usuarios (algunas veces clientes). Por su parte Pruitt (Pruitt et al., 2010) afirma que sin crear los perfiles de clientes objetivo y estudiándolos a través del ciclo de vida del desarrollo del producto es imposible entender realmente las necesidades del usuario, contexto y puntos débiles. En un aspecto general de diseño ambos autores reconocen y destacan la importancia de definir el perfil de usuario para elaborar el diseño de interfaz. Cooper (Cooper et al., 2007) propone una serie de pasos para construir personas y Pruitt (Pruitt et al., 2010) propone un ciclo de vida para construir persona a través de diferentes fases. Ambos métodos buscan generar una descripción que identifique lo más exacto posible a los usuarios finales del software.

Para construir un modelo persona Cooper et al. (Cooper et al., 2007) propone una serie de pasos que podemos observar en la tabla C4-2 al igual que las fases del ciclo de vida de persona presentado por Pruitt para construir persona.

Tabla C4-2 Pasos y Fases del ciclo de vida para construir persona

Pasos de Cooper A.	Fases del ciclo de vida de Pruitt J.
<ol style="list-style-type: none"> 1. Identificar variables de comportamiento 2. Mapear temas de entrevista a variables de comportamiento 3. Identificar patrones de comportamiento significativos 4. Sintetizar características y metas relevantes 5. Comprobar si hay redundancia e integridad 6. Ampliar descripción de atributos y comportamientos 7. Designar tipos de personas 	<ol style="list-style-type: none"> Fase 1. Planificación de la familia Fase 2. Concepción de persona y gestación Fase 3. Nacimiento de persona y madurez Fase 4. Edad adulta de persona Fase 5. Tiempo de vida límite de persona, reutilización y jubilación

Pruitt et al. (Pruitt et al., 2010) propone un ciclo de vida de persona para establecer un proceso que permita a los profesionales crear persona siguiendo un procedimiento más específico para construir y evitar errores durante el proceso de desarrollo. El ciclo de vida describe herramientas para utilizar personas durante el proceso de desarrollo de un producto. Algunos de los objetivos del ciclo de vida son obtener una definición creíble de persona, mejorar el mecanismo de disponibilidad para compartir las definiciones de persona en cualquier momento para el equipo de desarrollo y permitir un fácil conocimiento al equipo acerca de cómo utilizar las definiciones obtenidas.

Una característica que puntualizan ambas propuestas es identificar modelos del comportamiento de los usuarios y metas a través de una definición precisa y específica de patrones descritos como personas. Es necesaria una rigurosa investigación de usuario y obtener información suficiente para identificar los usuarios finales del software a ser desarrollado. A continuación se repasan brevemente los procesos de creación para persona presentados por Cooper et al. (2007 y 2014) y Pruitt et al. (2010).

4.2.2.1 Modelando personas (Cooper et al., 2007)

Originalmente como parte del proceso de modelado de persona (Cooper et al., 2007) proponen los pasos presentados en la tabla C4-2. Posteriormente en Cooper et al. (2014) los autores describen los siguientes ocho pasos:

1. Agrupar por rol a los sujetos entrevistados
2. Identificar variables de comportamiento
3. Mapear temas de entrevista a variables de comportamiento
4. Identificar patrones de comportamiento significativos
5. Sintetizar características y metas relevantes

6. Comprobar si hay redundancia e integridad
7. Designar tipos de personas
8. Ampliar descripción de atributos y comportamientos

El proceso de Cooper (2014) para crear las definiciones de personas indica que una vez que se haya realizado la investigación inicial, que no necesariamente debe ser exhaustiva, y una organización de los datos obtenidos se inicia con el primer paso para construir las personas necesarias *agrupando los entrevistados de acuerdo a roles*. Posteriormente en el segundo paso se *identifican las variables de comportamiento* por cada uno de los roles considerados, estas variables generalmente pueden identificarse a partir de considerar actividades, actitudes, aptitudes, motivaciones y habilidades de los entrevistados. Una vez identificadas las variables de comportamiento se *“mapean” o se agrupan los entrevistados* de acuerdo a éstas para ir identificando las coincidencias que existen entre ellos en cuanto al comportamiento. A partir del agrupamiento podemos *identificar patrones de comportamiento*, los sujetos que aparecen agrupados en seis a ocho variables es probable que representen un patrón de comportamiento significativo. Cooper (20014) señala que un patrón es válido cuando existe una conexión lógica o causal entre los comportamientos agrupados.

Una vez identificados los patrones se procede a *sintetizar las características* de cada uno a partir de los datos obtenidos y observados considerando al menos: actividades y motivaciones que generan los comportamientos, ambientes de uso, frustraciones relacionadas con el comportamiento utilizando las soluciones actuales, demografía, habilidades, experiencia o habilidades relacionadas con el comportamiento, actitudes y emociones, interacciones relevantes y formas alternativas de hacer lo mismo. En este paso se asigna un nombre a la persona por medio del cual se hará referencia a ella para visualizarla como real, además se anexa información demográfica como edad, ubicación geográfica, puesto de trabajo, etc. Otro punto importante que se define en este paso son las metas, identificando las conexiones lógicas entre cada grupo de comportamiento de los entrevistados se pueden inferir las metas que dan lugar a estos comportamientos. En el siguiente paso se *verifica si hay información que nos falta complementar o información redundante*. Es probable que se tenga que realizar investigación adicional sobre el usuario para completar ciertos datos, agregar nuevas personas o eliminar otras cuyas características han sido descritas para otra persona resultando redundante definir varias con un mismo objetivo. Como siguiente paso se asigna una *clasificación a las personas* como primaria, secundaria, complementaria, cliente, atendido o negativo. Las personas clasificadas como primarias son el principal objetivo del diseño de la interacción. Profundizando un poco más en los tipos de personas consideradas por Cooper tenemos que:

Primaria. Un producto puede tener solamente una persona primaria por interfaz pero puede ser que se tengan múltiples interfaces distintas cada una enfocada a una distinta persona primaria. En algunos casos dos interfaces pueden ser dos aplicaciones separadas que actúan sobre los mismos datos o puede suceder también que dos interfaces puedan ser dos conjuntos de funcionalidades proporcionadas a los usuarios dependiendo de su rol o personalización. Si un producto tiene varias personas primarias puede significar que el enfoque es muy amplio.

Secundaria. Las necesidades de diseño para la persona secundaria son satisfechas mayormente con la interfaz de la persona primaria. Conforme se va trabajando sobre el diseño, el enfoque principal es diseñar para la persona primaria y después solamente ajustar el diseño para la persona secundaria.

Complementaria. Sus necesidades están completamente representadas por una combinación de las personas primaria y secundaria y están a su vez completamente satisfechas por la solución ideada para una de las primarias.

Cliente. Este tipo de personas conducen las necesidades de los clientes, no usuarios finales. En ocasiones es importante identificar esta clasificación ya que pueden presentarse situaciones en las que personas tipo cliente pueden ser personas primarias para su propia interfaz administrativa. Tienen un tratamiento de personas secundarias.

Atendida. No son usuarios del producto pero son directamente afectadas por el uso del producto. Las personas atendidas proporcionan una forma de dar seguimiento a ramificaciones de orden social y físico a los productos. Tienen un tratamiento de personas secundarias.

Negativa (anti-personas). No son usuarios del producto, su uso es puramente retórico para ayudar a comunicar a los miembros del equipo que la correspondiente representación de persona definitivamente no es el objetivo del diseño del producto.

En la última etapa del proceso de construcción de personas propuesto por Cooper se *amplía la descripción sobre los atributos y comportamientos*, las características identificadas al realizar la síntesis se pueden describir como una narrativa y complementar la descripción. Se asigna una imagen a la definición de persona ya sea una fotografía individual o imágenes agrupadas que representen toda o algunas de las actividades que la persona realiza.

4.2.2.2 El ciclo de vida de persona de Pruitt J. et al. (2010)

Por su parte, la fase inicial del ciclo de vida de persona propuesto por Pruitt et al., en (Pruitt et al., 2010) se denomina *planeación de la familia* y es una etapa de investigación y análisis. Se recaba información sobre los problemas y necesidades de la organización, se realiza investigación sobre el usuario, se identifican fuentes de información que proporcionarían datos para definir las personas. La definición completa de personas debe incluir información básica esencial: metas, roles, comportamientos, segmento, el ambiente y actividades típicas que hacen la persona sólida, rica, única y relevante para el diseño del producto.

En la fase posterior *concepción y gestación* se extrae información útil de las diferentes fuentes de datos para construir personas, se priorizan las ya identificadas, se documenta la definición de éstas y se validan las definiciones creadas. En esta etapa se priorizan las primeras descripciones considerando únicamente dos clasificaciones: persona primaria y persona no-primaria (o secundaria) donde la primaria es el objetivo del diseño, coincidiendo con el concepto de Cooper et al. (2007). En relación a la documentación de las personas se describen dos tipos de documentos base: personas ad hoc y personas data-drive. Los documentos ad hoc funcionan principalmente cuando los equipos no tienen tiempo o recursos para crear por completo personas basadas en datos. Los documentos base para personas basadas en datos (data-drive) requieren más tiempo para aplicar un profundo análisis de datos. Aunque hay versiones más cortas de personas basadas en datos mientras más detallada es la información en esta documento base mejor es la descripción obtenida de persona. Como podemos inferir, este tipo de documento base es recomendado cuando un equipo está extremadamente limitado en tiempo y recursos. Pruitt et al. (2010) señala tres puntos básicos a considerar en los documentos base: etiquetas para los grupos que surgieron del ejercicio de asimilación, temas relevantes para tu dominio del producto o negocio y encabezados comunes en los documentos de persona que pueden ayudar a crear una persona realista, útil y completa.

En la etapa de *nacimiento y maduración* de persona el equipo ya conoce y utiliza las definiciones de persona desarrolladas en la fase previa. Sin embargo, a medida que se van utilizando, las descripciones de las personas definidas pueden requerir actualizarse de acuerdo a nuevos datos que surgen durante el desarrollo del proyecto. Durante la *etapa adulta de persona* la importancia de las personas definidas es ya reconocida por el equipo y la organización, se relacionan ya las definiciones de persona con los procesos correspondientes identificando su papel en éstos, situación que permite trabajar considerando la perspectiva de los usuarios finales. En la última etapa *logros de la vida, reutilización y jubilación* del ciclo de vida de persona se evalúa la trayectoria de utilidad de las personas que se trabajaron durante el desarrollo del proyecto que está finalizando, se determina si alguna de las personas puede reutilizarse o es mejor descartarla en los siguientes proyectos.

Ambas propuestas consideran documentar la descripción de las personas a considerar como las representaciones de los usuarios finales. No hay un formato estándar para documentar la descripción de persona, se mencionan elementos que pueden considerarse en las descripciones representativas que se van generando pero aún no se ha establecido una estructura general para éstas. Pruitt et al. (Pruitt et al., 2010) menciona dos tipos de documentos base para saber y aprender sobre las actividades, metas, expectativas y necesidades de los usuarios: Ad hoc y Data-drive persona. El *documento Ad hoc* de Pruitt (2010) es una descripción menos pesada y sugerida principalmente para aquellos equipos de desarrollo que no cuentan con el tiempo y recursos necesarios para crear una definición más exhaustiva. Incluye información como: nombre de persona, una foto de la persona, nivel de prioridad, expresiones particulares de la persona, un relato general que describe quién es la persona, preguntas en términos de la persona y cuestiones relacionadas con el producto a desarrollar que en base a la información identificada del usuario se visualiza es interesante para la persona saber y generar en cierta forma mayor interés sobre el producto. El documento base Data-Driven requiere mayor descripción de los datos

contenidos en él por lo que su uso está enfocado para aquellos equipos que tienen la posibilidad de contar con el tiempo suficiente para generar una definición más rica en información. Los documentos Data-Driven deben contener abundantes ligas a, bastantes pie de páginas o comentarios sobre datos específicos, ligas a los reportes originales de investigación que respaldan y explican las características de las personas, indicaciones sobre cuáles características se respaldan de datos y cuáles son ficticias o basadas en suposiciones. La figura C4-1 muestra ejemplos de plantillas de documentos base tipo Data-Driven lo más resumidos posibles.

Persona Name: Job/Role Description:
Short Narrative (description of the person acting out his or her primary scenario(s)):	
Data Sources and/or Sources of Assumptions:

Persona Name: User Class or Segment (including market size, importance):
Job, Role, Activities:	
Goals:	
Abilities, Skills, Knowledge:	
Personal Details:	
Data Sources and/or Sources of Assumptions:	

Figura C4-1 Plantillas de documento base (Pruitt et al., 2010)

Cooper y Pruitt concuerdan en que la representación debe considerar elementos como fotos o ilustraciones y una descripción general que contenga factores importantes que permitan conocer y entender las necesidades del usuario final a través de los comportamientos identificados. Cooper (Cooper et al., 2007) señala que un collage de imágenes combinadas con narrativas cuidadosamente escritas son una forma efectiva para transmitir los aspectos emocionales y experimentales de una persona. Por otro lado Pruitt (2010) menciona que utilizar fotos o ilustraciones para representar a personas es crucial debido a que ayudan al equipo a creer en las personas y entender que cada definición describe realmente a una sola persona, un solo usuario. La figura C4-2 muestra una representación general de una persona.

Michael is 35 years old, he is a management accountant in charge of the company's store. Every day when Michael arrives to his office verify the reports of the movements registered on previous day by his assistants and the schedule of payments, purchases and deliveries programming for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in a shorter time. Michael and his assistants needs to know amount, price and total that at the end of the day were implied in the movements, he expects has a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality.
"the main thing is to save without compromising the quality of the products purchased"
"Delivery on time"

Figura C4-2 Descripción de persona con una ilustración y corta narrativa

4.2.3 Persona en el Desarrollo Ágil de Software

A partir del concepto de marketing, el concepto de persona adoptado en HCI ha mostrado buenos resultados al ser incorporado en el proceso del desarrollo de software en un entorno ágil. La figura C4-3 muestra una visión general de la manera en la que las concepciones de persona en cada una de las áreas contribuyen entre ellas.

Figura C4-3 Percepción de persona en Mercadotecnia, HCI y Agile

El proceso de persona ha sido adaptado en proyectos ágiles para obtener resultados en menor tiempo sin minimizar el objetivo original de la herramienta pero satisfaciendo los principios ágiles. ¿Cómo han sido adoptados o adaptados estos lineamientos para crear persona en el desarrollo ágil? ¿Las adaptaciones hechas han funcionado para obtener un exitoso desempeño ágil?

Relacionado con estas adaptaciones hemos de considerar el momento de aplicación de persona en el proceso de desarrollo ágil. En (Caballero et al., 2014) los autores hacen referencia a los momentos de aplicación como fase exploratoria y fase de refinamiento. Durante la fase exploratoria las personas son definidas al principio del proyecto antes de trabajar sobre cualquier ciclo ágil y antes de liberar cualquier código. De esta forma se obtiene una perspectiva completa o parcial de las metas y especulaciones del usuario para mejorar los resultados de los ciclos o sprints derivados y de las observaciones y retroalimentación recibidas. En la fase de refinamiento, esto es, a través de los ciclos ágiles, las personas definidas en la fase exploratoria son evaluadas utilizando la retroalimentación de cada ciclo previo. En esta fase nuevas personas pueden ser creadas. Esta distinción de momentos puede ayudar a determinar si las personas existentes deben mantener la definición previa, ser modificada, eliminada, fusionada con otras o es necesario crear nuevas descripciones de persona.

La tabla C4-2 muestra un extracto de los artículos analizados en la revisión de literatura descrita previamente en la sección 2.2 donde encontramos información relevante sobre el uso de persona en proyectos ágiles. En esta tabla se presenta información sobre el origen del concepto de la técnica persona utilizada por cada autor, concepto utilizado en el dominio ágil (original o adaptación) y el momento en el proceso ágil donde persona (o alguna variante de la técnica) fue aplicada. En relación a la etapa dentro del proceso de desarrollo ágil en donde se aplican las prácticas HCI consideramos la fase exploratoria y la fase de refinamiento:

- *Fase exploratoria*, las técnicas HCI son aplicadas antes del desarrollo de las iteraciones para identificar situaciones relacionadas al diseño y obtener retroalimentación necesaria antes que cualquier código se implemente y libere.
- *Fase de refinamiento*, las técnicas HCI son aplicadas durante las iteraciones ágiles para mejorar el diseño definido previamente.

Tabla C4-3 Extracto del análisis sobre el uso de la técnica Persona en proyectos ágiles

Autor/s	Ref.	Origen de persona	Concepto utilizado	Fase proceso ágil
S. Chamberlain et. al.	(Chamberlain et al., 2006)	HCI-Cooper	Persona	Exploratory phase
J.Haikara 2007	(Haikara, 2007)	HCI-Cooper	Persona	Exploratory phase Refinement phase
D.Sy 2007	(Sy, 2007)	HCI-Cooper	Persona	Exploratory phase
Z. Hussain et. al.	(Zahid Hussain et al., 2008a)	HCI-Cooper	Extreme Persona	Exploratory phase Refinement phase
P.Wolkerstorfer et. al.	(Wolkerstorfer et al., 2008)	HCI-Cooper	Extreme Persona	Exploratory phase Refinement phase
M.Najafi et. al.	(Najafi & Toyoshiba, 2008)	HCI-Cooper	Persona	Exploratory phase
M. Singh 2008	(Singh, 2008)	HCI-Cooper + Marketing	User persona	Exploratory phase
D.Broschinsky et.al.	(Broschinsky & Baker, 2008)	HCI-Cooper	Persona	Exploratory phase
L. Cho 2009	(Cho, 2009)	HCI-Cooper	Persona	Exploratory phase Refinement phase
J.Gonzalves et. al.	(Gonçalves & Santos, 2011)	HCI-Cooper	Persona	Exploratory phase

En (Chamberlain et al., 2006), Chamberlain y Sharp introducen métodos de diseño por adelantado donde personas son definidas en base a investigación significativa del usuario antes de que cualquier código esté hecho. Haikara (Haikara, 2007) amplió el proceso de diseño de interacción combinando con el marco de trabajo ágil Mobile-D fases para la creación de persona durante el proceso. La persona primaria es identificada en el primer día de planeación y posteriormente las expectativas de la persona son refinadas. La figura C4-4 muestra el proceso adaptado de desarrollo incluyendo la técnica persona.

Figura C4-4 Etapas para la creación de persona durante el proceso Mobile-D (Haikara, 2007)

Sy (Sy, 2007) utiliza la técnica persona al inicio del desarrollo ágil del proyecto en el ciclo cero. El autor afirma que de esta forma ayuda a desarrollar breves y vívidas descripciones de los usuarios objetivo y flujos de trabajo.

Hussain et al. (Zahid Hussain et al., 2008a) y Wolkestorfer et al. (Wolkestorfer et al., 2008) extienden el paradigma XP a la definición de persona con pequeños e iterativos pasos y refactorización resultando lo que ellos llaman una “persona extrema”. Al inicio del proceso los usuarios son investigados a través de estudios de usuario para desarrollar personas. Al final de cada iteración la visión sobre los usuarios es más amplia. Esto ayuda para el siguiente ciclo de refactorización para actualizar las descripciones de personas o desarrollar nuevas personas si la información obtenida durante los ciclos recientes muestra que las personas consideradas hasta el momento no cubren las nuevas perspectivas. En la figura C4-5 se puede observar el modelo.

Figura C4-5 Proceso XP adaptado con la integración de técnicas HCI (Zahid Hussain et al., 2008a)

De manera similar Najafi et al. (Najafi et al., 2008) consideran métodos HCI en el sprint cero de un proceso Scrum. En este caso el equipo encargado de la experiencia de usuario específicamente utiliza el sprint cero para entender, explorar el contexto e identificar las metas del usuario. La figura C4-6 muestra el modelo Scrum adaptado utilizando bloques paralelos uno para el diseño (User Experience) y otro para la implementación (Engineering Development).

Figura C4-6 Modelo de integración entre SCRUM y el diseño UX (Najafi & Toyoshiba, 2008)

Singh (Singh, 2008) utilizó la definición de persona original de Alan Cooper y describe un usuario persona como un usuario arquetipo resultando de la combinación de una investigación de mercado, estudios etnográficos y observaciones. El autor propone el modelo U-Scrum el cual incorpora un product owner de usabilidad desde el inicio del proyecto. Su rol es trabajar con el product owner scrum para obtener acuerdos sobre la experiencia del usuario a implementar. El producto owner de usabilidad trabaja con el equipo de marketing definiendo personas al inicio del proceso. En (Broschinsky & Baker, 2008) las personas se definían también al inicio del proyecto antes de iniciar con la implementación para ayudar a los desarrolladores a conocer a sus usuarios y familiarizarse con ellos. Las descripciones de las personas son validadas con los clientes y desarrolladores on-site. Durante las iteraciones posteriores las personas son refinadas en base a las reuniones de revisión pero

preservando lo esencial de las descripciones originales. En el proyecto de estudio de (Cho, 2009) durante el proceso ágil se utilizan “Feature teams” (Frank et al., 2009) equipos auto organizados encargados de completar el diseño, planeando y desarrollándolo en el mismo sprint. Cada sprint tiene fases de exploración y refinamiento. Conforme el software va evolucionando a través de los sprints nuevas funcionalidades son construidas dentro de los componentes existentes. El conjunto inicial de personas es entonces actualizado o nuevas personas son creadas.

(Gonçalves et al., 2011) presentan una herramienta de software que combina Scrum y la metodología Goal-Directed Design (GDD) de Alan Cooper (Cooper et al., 2007). El propósito de la herramienta es desarrollar un sistema que genera prototipos de baja fidelidad para ayudar a identificar y definir personas en el ciclo cero antes de los sprints ágiles. Uno de los artefactos propuestos por (da Silva et al., 2015) son las Lean Personas que representan a un conjunto de usuarios finales. Las descripciones, llamadas hipótesis de persona, se basan en el formato simplificado propuesto en (Miller et al., 2006) y se generan basadas en conocimiento previo sobre los usuarios potenciales del sistema y no necesariamente haciendo una investigación mínima o exhaustiva para identificar a los usuarios finales. (Choma J et al., 2015) también utilizan dicho formato en su estudio para las descripciones de persona.

4.2.4 Adaptaciones al incorporar la técnica persona en el proceso ágil

A partir de la revisión de literatura descrita en la sección anterior podemos identificar principalmente tres adaptaciones de la técnica persona al utilizarse en un entorno de desarrollo ágil:

Minima investigación sobre el usuario en lugar de una específica y rigurosa obtención de datos para definir persona. El concepto original de persona enfatiza realizar una investigación a fondo sobre el usuario obteniendo de esta manera suficiente información para definir e identificar las personas que servirán como apoyo durante todo el proceso de desarrollo. Los equipos ágiles únicamente realizan la investigación necesaria sobre el usuario que les permita identificar características clave para iniciar con una definición de las personas. Los métodos más sencillos que sirven de apoyo a los equipos ágiles para sus investigaciones son la observación y las entrevistas.

El momento de aplicar la técnica persona. En un desarrollo ágil la técnica persona se aplica en las fases de exploración y refinamiento, inicialmente y siguiendo la definición original de la técnica ésta se aplica al inicio del proceso de desarrollo durante la etapa de exploración antes de realizar cualquier otra tarea asegurando una definición precisa del arquetipo que representa las metas y comportamiento del usuario. Aunque elaborar el diseño por adelantado consumirá algo de tiempo al principio del proyecto de esta manera con enfoque en el usuario final del software se evitará durante el desarrollo de software tanto como sea posible la pérdida de tiempo por re trabajo. Los equipos ágiles aplican persona al inicio del proceso con una investigación mínima obteniendo únicamente la suficiente información relevante relacionada con los usuarios para definir una descripción inicial de persona. Una vez iniciado el desarrollo de los sprints las definiciones de persona se van refinando o incluso se pueden crear nuevas personas involucradas con las historias de usuario a desarrollar en el sprint de acuerdo a la retroalimentación de iteraciones previas y/o a las observaciones del product owner. Enfatizando al principio del desarrollo de software, la planeación de la iteración y durante las iteraciones donde nuevas personas pueden surgir o alcanzar la madurez a través del refinamiento de sus descripciones asegurando que el producto software satisfará las expectativas y necesidades del usuario.

Una tercera adaptación es *el formato de la plantilla* utilizada para describir la definición de persona. Aunque el concepto original propone un formato simple definiendo únicamente la información más importante relacionada con el usuario y una imagen para identificar a la persona y hacerla más real, de hecho la cuestión importante es el proceso para obtener información relevante. La plantilla propuesta permite a los equipos ágiles describir a la persona con una forma más estructurada siguiendo la regla de simplicidad y rapidez. En el entorno ágil los datos clave deben identificarse más rápido por lo que describirlos de una manera estructurada ayuda a conocer y mantener en mente el comportamiento y metas de los usuarios.

4.3 Patrón “Agile persona”

Con el objetivo de proporcionar una forma que permita reutilizar la información derivada de la experiencia de los equipos ágiles que han aplicado exitosamente la herramienta persona, proponemos un patrón considerando quién, cuándo y cómo la herramienta persona puede ser utilizada en un contexto ágil.

4.3.1 Definiendo el patrón “Agile persona”

Como hemos visto previamente, la técnica persona ha sido “agilizada” es decir, la técnica es adaptada al entorno ágil para incorporarla dentro del proceso de desarrollo. Queda claro que puede ser utilizada en diferentes momentos reflejando también mejora en la comunicación que se genera entre los stakeholders y desarrolladores.

Utilizamos como punto de partida la estructura del patrón de producto propuesto en (Sanchez-Segura et al., 2010) descrito previamente en la sección 3.3.3. Como resultado del proceso de evaluación aplicado a los patrones y que se detalla en el capítulo 6 de este trabajo, se identifican adaptaciones mínimas en formato y contenido a la estructura de (Sanchez-Segura et al., 2010) para describir el patrón “Agile Persona”.

En relación al formato, se modificó el nombre del campo que indica el(los) patrón(es) con el(los) que se relaciona el patrón descrito cambiando de “Collaboration” al de “Related Pattern”. También se reubicó su posición en la plantilla colocándolo al principio del formato después del nombre del patrón (campo “Name”). En cuanto al contenido, la adaptación se realizó sobre el campo “Capability label” el cual (Sanchez-Segura et al., 2010) indican se refiere al nivel de madurez con el que se relaciona el patrón dentro de un programa de mejora de procesos (en caso de tener alguno). El patrón de “Agile persona” no contiene dicho campo ya que la solución no considera relación específica de su uso dentro de algún proceso de mejora. Sin embargo se considera esta área de oportunidad para estudios futuros en esta misma línea de investigación.

En la figura C4-7 se muestra la estructura final del patrón “Agile Persona” después de aplicar las modificaciones al formato derivadas de las evaluaciones.

Name	Nombre del patrón, puede ser una palabra o frase corta	Related pattern	Nombre del (los) patrón(es) con el (los) que tiene relación
Initial context	Descripción de las precondiciones sobre las cuáles el patrón es aplicable		
Result Context	Contexto resultado al aplicar el patrón		
Problem	Descripción del problema en el que contribuye la solución del patrón		
Forces	Restricciones a considerar para la aplicación del patrón		
Solution	Solución propuesta por el patrón para el problema indicado. Se utilizan diagramas para representar el proceso de solución		
Roles	Personas o participantes en la aplicación del patrón		
Basic Knowledge	Conocimiento básico que se requiere para aplicar el patrón		
Entries	Información de entrada o productos previos para aplicar el patrón		
Exit	Producto obtenido como resultado de aplicar el patrón		
Lessons learned	Experiencias que se han obtenido al aplicar el patrón en proyectos previos		
Examples	Ejemplos de la aplicación del patrón		
Templates	Plantillas a utilizar durante la aplicación del patrón para obtener el resultado		
Information Resources	Recursos de información tales como referencias y documentación que pueden utilizarse para desarrollar este patrón		

Figura C4-7 Estructura final del patrón “Agile Persona”

En la tabla C4-4 se presenta la descripción del patrón “Agile persona” con el formato y contenido del patrón actualizado. Para una descripción más clara de la información contenida en el patrón dividimos la plantilla en tres partes, el formato completo podemos visualizarlo en la sección final de este capítulo.

Tabla C4-4 Descripción del patrón "Agile Persona" (Parte 1 de 3)

Name	Agile persona	Related pattern	None
Initial context	This pattern can be used in agile projects to help development team to learn and understand about the user's needs, goals and behavior in order to identify the final users and develop the right functionalities for them with useful and usable interfaces.		
Result Context	Specific models descriptions of the final users' features will be obtained. Such descriptions allow the development team to know about the users for whom the software will be developed. Those descriptions will be available to review at any moment through the agile development process.		
Problem	The agile development team must be able to identify and define as soon as possible the final users for whom the software will be developed in this way avoiding wasting time on reworking the development process.		
Forces	<p>Kind of the organization This pattern can be used in agile projects regardless of company size.</p> <p>Kind of system to be developed This pattern can be used in agile projects that use development methodologies following the agile philosophy.</p> <p>Kind of client The user must be available and always involved through the process of the persona's definition.</p> <p>Other</p> <ul style="list-style-type: none"> • Meetings for interviews, pattern's review or feedback between development team and user must be established in a friendly and trusted environment • Users, UI designers and developers must be involved in the presentation, reviews and feedbacks meetings • Pattern persona must be keep available for the team through the whole development process at any moment 		

Nuestro patrón tiene por *nombre* “Agile Persona” para representar el nombre de la herramienta HCI y el ambiente de desarrollo para el cual es propuesta esta estructura reusable. En el campo de *patrón relacionado* se indica “None” ya que por el momento el patrón Agile persona no tiene relación con otros patrones desarrollados; sin embargo una vez que forme parte de un lenguaje de patrones se irá actualizando este dato en su descripción.

El *contexto inicial* de uso del patrón indica proyectos ágiles en los que sirve como apoyo al equipo de desarrollo ágil para aprender y entender las necesidades, metas y comportamiento de los usuarios. Lo anterior con el objetivo de identificar a los usuarios finales y desarrollar las funcionalidades correctas para ellos con interfaces útiles y usables. El *contexto resultado* son las descripciones de las personas que identifican a los usuarios finales y permanecen disponibles para el equipo ágil durante todo el proceso de desarrollo para consulta y/o mejora.

El *problema* que “Agile Persona” tiene por objetivo atacar es la pérdida de tiempo y re trabajo para el equipo que se da como resultado de un estudio equivocado; el patrón apoya al equipo ágil para obtener conocimiento sobre el usuario identificando y definiendo tan pronto como sea posible a los usuarios finales. Como *restricciones (Forces)* para conseguir nuestro contexto de resultado se consideran, entre otras, que puede ser utilizado independientemente del tamaño de la compañía, puede utilizarse en proyectos ágiles que utilizan cualquier metodología de desarrollo apegada a la filosofía ágil, la participación del usuario y la disponibilidad de la descripción de persona a través de todo el proceso de desarrollo son cruciales.

La *solución* del patrón se presenta en la tabla C4-4 (parte 1, 2 y 3) y se describe a continuación. Los diagramas se presentan individualmente para una mejor visualización de los procesos considerados y la relación entre ellos. En la primera parte de nuestra solución se presenta el diagrama que observamos en la figura C4-8 en el que se muestran los procesos que implica aplicar la técnica persona en un entorno ágil. El patrón describe que son cinco procesos necesarios para incorporar la herramienta: Investigación mínima sobre el usuario por medio de entrevistas y observaciones (Minimal user research), describir y priorizar persona (Describe and prioritize persona), revisar la definición de persona (Review persona definition), actualizar la descripción y priorizar persona (Update description and prioritize persona). Una última actividad es detallar la definición de la historia de usuario esto definiendo por parte del product owner los criterios de aceptación y el equipo de desarrollo definiendo las tareas necesarias a desarrollar durante la iteración.

Figura C4-8 Cómo aplicar persona en un entorno ágil

En la figura C4-9 se representan los roles involucrados y cómo participan en el proceso de creación de las personas del proyecto de software a desarrollar. Todo el equipo ágil debe participar durante la identificación y descripción de las personas, el development team junto con el product owner tienen una participación esencial al momento de generar las descripciones precisas que se requieren para representar y mantener en mente a los usuarios finales durante todo el proceso de desarrollo.

Figura C4-9 Quién está involucrado en el proceso y cómo se involucra

Independientemente del momento en el que se incorpore la técnica en el proceso ágil, ya sea exploración o refinamiento, ésta implica que el equipo de desarrollo realice inicialmente una mínima investigación sobre el usuario. Obteniendo de esta manera información esencial que permita establecer una descripción inicial de aquellas personas que podrían, en primera instancia, representar a los usuarios finales. Esta investigación puede apoyarse con observaciones y entrevistas a todos aquellos involucrados con el sistema. Una vez obtenidos los datos iniciales necesarios el equipo de desarrollo describe y prioriza las personas, el product owner revisa las descripciones y en base a su retroalimentación se determina si es necesario actualizarlas y priorizar de nuevo a las personas. Esta actualización se deriva de la retroalimentación del sprint anterior, en caso de que se hubiera desarrollado alguno, y/o de las observaciones del product owner al revisar las descripciones de personas. Si no es necesario algún refinamiento a las personas entonces están ya listas para trabajarse definiendo los criterios de aceptación y/o las tareas a realizar durante las iteraciones.

Tabla C4-4 Descripción del patrón "Agile Persona" (Parte 2 de 3)

Solution

In agile environment, incorporating persona involves different activities: develop minimal user research, describe and prioritize persona descriptions, review persona definitions, update description/prioritize persona and obtain a detailed definition of the user story. Figure 1 depicts how persona is applied through these processes.

In order to construct the user models we applied persona technique by following the next steps:

1. Minimal User Research: development team develops minimal user research using artifacts such as interviews and observation in order to get just basic information about the user in order to identify relevant information to define user models. These models will be improve through subsequent iterations.
2. Describe and prioritize persona: using a template provided as part of the pattern the development team describe the user's features, needs, goals and behaviors. Then, development team prioritizes personas identified.
3. Review persona definition: product owner and development team review the persona's descriptions and prioritization assigned.
4. Update description/ prioritize persona: if a refinement is necessary, the personas' descriptions are updated and prioritized again according to the feedback from previous review or product owner's observations.
5. Detailed definition of the user story: once the persona's description is finished then, tasks and acceptance criteria related with this can be detailed. In this way the user stories remain relative to their corresponding persona.

Figure 1 - How to Apply persona at Agile environment

Figure 2 depicts an activity diagram which shows who develops the corresponding activities regarding the roles involved. In general the activities are performed as described below:

- The development team receives information previously gathered from the Minimal User Research (observation and interviews).
- The development team defines persona as user models with hypothetical archetypes describing the user's features identifying needs, goals and behavior variables.
- The development team prioritizes personas identified.
- The user review together with development team, the persona's description and prioritization.
- If it is necessary the development team refines persona's definitions by creating a new persona product pattern or by updating existing patterns after each iteration feedback or product owner's review.
- User stories can then be detailed according to the personas definition. This activity is developed by both product owner (establishing acceptance criteria) and development team (defining tasks).

Figure 2 - Who are involved defining persona pattern

Figure 3 depicts a general way of the agile development process highlighting the moment to apply persona with a gray background. In this diagram sprint planning and sprint development has an asterisk (*) and a double asterisk (**) respectively to indicate a reference to the

processes shown in figure 4 and 5 where persona technique is also applied. In the diagrams the process named “Apply persona” involves the activities described previously and depicted in figure 1.

Persona technique is applied at the beginning of the development process in an explorative way in order to get information about the users visualizing the general scope of the project considering this information to define initial product backlog and interface design. The agile team decides how deeply to conduct the user research to get necessary information.

Figure 3 - When to apply persona through agile development process

At this moment, *the beginning of the development process*, “Apply persona” implies doing the following tasks according to the processes described previously with reference to figure 1:

1. Do a minimum user research to get relevant information in order to define initial user’s descriptions.
2. Once possible target users are identified then describe them and prioritize these descriptions.
3. Review the persona definitions. This task is performed by the development team together with the product owner.
4. If a necessary refinement to persona is identified an update to the definition is applied according to the feedback gathered from previous iterations or product owner’s review. Verify the prioritization of persona again and update it if necessary.
5. Define the user story details by specifying tasks and acceptance criteria related to it.

Persona is applied in an explorative way at sprint planning and sprint development. At this time applying this technique aims to verify and improve previous persona’s descriptions. Figure 4 depicts when persona is applied in iteration planning and figure 5 when persona is applied in iteration development.

Figure 4 - When apply persona at iteration planning

Figure 5 - When apply persona at iteration development

During *iteration planning* the technique is applied by developing the following tasks:

1. Review feedback from previous iterations (if there have been previous iterations)
2. Relate the personas’ definition with the user stories to develop

3. Determine if refinement is needed or creation for a new persona definition
 - If it is necessary to create a new persona, then minimal user research is applied and persona's description defined, prioritized and reviewed.
 - If refinement to persona is necessary then update the definition according to the feedback gathered from the previous iteration or product owner's review. Verify the prioritization of persona and updated it if necessary. This verification is applied by product owner and development team.
4. Define the user story details by specifying tasks and acceptance criteria related to it.

At *iteration development* persona is applied by developing the following tasks. For each user story to develop in the iteration we are going to:

- Relate the personas' definition with the user stories to develop. Verifying definitions with the personas related with the users involved in the user story.
- Validate the refinement needed or creation of a new persona definition
 - If it is necessary create a new persona, then minimal user research is applied
 - If a necessary refinement to persona is identified we update the definition according to the feedback gathered from the previous iteration or product owner's review. Verify the prioritization of persona and updated it if is necessary.
- Define the user story details by specifying tasks and acceptance criteria related to it.
- Development team decides to implement the user story or continue to design next user story.

During iteration development parallel tracks of user stories implementation or design can be worked at the same time if the development team has enough elements and timeboxing.

When the last user story of the iteration backlog is done a shippable product is ready to be delivered. Feedback from persona incorporation through the iteration is available to be used at the iteration backlog for the next iteration(s).

La figura C4-10 muestra cuándo la técnica persona es aplicada durante el proceso ágil de desarrollo. Al inicio se observa explícitamente el proceso "Apply Persona" antes de la definición del product backlog inicial esto es, la técnica se aplica en la etapa de exploración. En el mismo diagrama los procesos "Sprint planning" y el de "Sprint development" se señalan con (*) y (**) respectivamente indicando que en estos procesos también se aplica la técnica persona y haciendo referencia a los diagramas correspondientes.

Figura C4-10 Cuándo se aplica persona en un proceso ágil

Considerando las actividades que implica aplicar la técnica persona (figura C4-8) en los diferentes momentos en un entorno de desarrollo ágil tenemos que, *al inicio del proceso de desarrollo ágil* el equipo obtiene información a través de una **investigación mínima** sobre el usuario utilizando observaciones y entrevistas. Como Cooper et al (Cooper et al., 2007) señala la meta al identificar un conjunto de personas es representar la diversidad de motivaciones, comportamientos, actitudes, aptitudes y modelos mentales. Para realizar esto incluso de una manera ágil es fundamental la información que obtenemos directamente de los usuarios aunque la investigación inicial sobre el usuario se desarrolla en menos tiempo y no representa un diseño up-front completo. En un proceso ágil los datos obtenidos deben permitir describir solamente una representativa pero importante parte de la definición de persona siendo completada y refinada a través de las subsecuentes

iteraciones. Entonces el equipo de desarrollo define una primera concepción de persona **describiendo y priorizando el tipo de persona**.

Para mantener en mente la definición de persona de los usuarios objetivo definimos una plantilla cuya estructura surge principalmente a partir del concepto de documentos base tipo Ad hoc de persona descritos por Pruitt et al. (Pruitt et al., 2010) y comentados en la sección anterior 4.2.2.1. Adaptamos este tipo de documento base con el objetivo de proporcionar al equipo un artefacto que le permita describir de una manera organizada y sencilla la información relevante en menos tiempo. Más adelante se presenta un ejemplo y describimos la información contenida en la plantilla.

Una vez descrito el perfil **la definición de persona se revisa** por el product owner y el equipo de desarrollo (expertos HCI son opcionales en el equipo). Si es necesario realizar un refinamiento entonces el equipo **actualiza la definición** de acuerdo a las observaciones derivadas de la revisión del product owner y se verifica la priorización actualizando también si fuera necesario. El product backlog es definido con las historias de usuario relacionadas con las personas identificadas. Las personas definidas ayudan al equipo a definir las historias de usuario en el **product backlog**. Una vez que el product backlog es definido las descripciones de las personas necesitan estar disponibles en cualquier momento para el equipo de desarrollo, usuarios y product owner ya que son una referencia durante el proceso de desarrollo. Además también es importante tener disponible las descripciones para poder aplicar adaptaciones cuando es necesario actualizar de acuerdo a los refinamientos que se van requiriendo. El **backlog de la iteración (sprint)** es definido identificando las historias de usuario a ser consideradas en el siguiente sprint. Si no es el último sprint entonces se verifica el product backlog y se determina el siguiente sprint backlog.

Las figuras C4-11 y C4-12 muestran la aplicación de la técnica durante la planeación de la iteración (Sprint planning) y durante el desarrollo de ésta (Sprint Development) respectivamente es decir, en la etapa de refinamiento.

Figura C4-11 Cuándo aplicar persona durante la planeación de la iteración

Figura C4-12 Cuándo aplicar persona durante el desarrollo de la iteración

Durante la *planeación de la iteración (sprint planning)* el equipo ágil revisa retroalimentaciones derivadas de sprints previos (en caso de que haya). Se relacionan las personas con las historias de usuario del product backlog. El equipo de desarrollo determina si es necesario crear nuevas personas o actualizar las definiciones de las ya existentes. Posteriormente se revisa el product backlog y en caso de ser necesario se actualiza creando o modificando las historias de usuario tomando en cuenta las descripciones de persona. Finalmente se determina el sprint backlog.

Durante el *desarrollo de la iteración (sprint)*, el equipo relaciona personas con las historias de usuario a ser desarrolladas en el sprint con el objetivo de confirmar si la descripción original de las personas esta correcta, si hay algún refinamiento a aplicar o si es necesario crear alguna nueva persona. Si el equipo necesita definir una nueva persona se realiza el proceso ya mencionado que implica desarrollar una mínima investigación del usuario, describir y priorizar a la(s) persona(s) utilizando la plantilla correspondiente. De esta manera las adaptaciones o creación de personas pueden darse a través de todo el proceso de desarrollo. El siguiente paso involucra la implementación de la historia de usuario desarrollando el código necesario. Como podemos observar en la figura C4-12 durante el desarrollo de la iteración el equipo puede trabajar bloques paralelos de diseño de interfaz y desarrollo. La técnica persona sirve de apoyo durante el bloque de diseño mientras otros elementos del equipo implementan los diseños definidos en iteraciones previas.

Después del campo solución descrito previamente tenemos los que se muestran en la tabla C4-4 (Parte 3 de 3).

Tabla C4-4 Descripción del patrón "Agile Persona" (Parte 3 de 3)

Roles	<ul style="list-style-type: none"> Development Team (Agile Developers, Scrum Master, Product owner, etc.) Users (stakeholders, end-users, representative users) UI Designer (optional)
Basic Knowledge	<p>Knowledge</p> <ul style="list-style-type: none"> Knowing about requirements gathering <p>Skills</p> <ul style="list-style-type: none"> Capability to apply user research to get data through direct observation, interviews and questionnaires. Capability of data analysis identifying needs and establishing requirements.
Entries	User information
Exit	Persona's descriptions as user Models
Lessons learned	<ul style="list-style-type: none"> The whole development team must be involved in the personas definition. After the UI designers and/or developers have prioritized the personas then the models have to be presented for review in a meeting of all the development team and users. Users, UI designers and developers must be involved in the presentation, reviews and feedbacks meetings. Meetings for interviews, product pattern's review or feedback between development team and user must be established in a friendly and trusting environment. The pattern persona must be kept available to the team through the whole development process.
Examples	<div> <div> <p>Persona</p> <p>Name: Michael Age: 35 Occupation: Accountant</p> </div> <div> <p>Activities</p> <p>Michael is a 35 years old. He is a management accountant in charge of a company's store. Every day when Michael arrives at his office he verifies the reports of movements registered on the previous day by his assistants and the schedule of payments, purchases and deliveries programmed for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in the least time. When the number of items requested exceeds the minimum the requesting area is informed for an approximate date on which they will receive the products requested. There are specific processes for the purchases depending on the kind of product required by the company and Michael applies, helped by his assistants, the corresponding process.</p> <p>Goals and Needs using the solution</p> <p>Michael expects to have information about the processes of his department at any moment. He needs it and wants to verify. Michael needs to know ahead of time the departments' requests to verify if the products required are available in the store or, otherwise, buy them for delivery on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants need to know the amount, price and total that at the end of the day were implied in the movements. He expects to have a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with his suppliers and the company's departments.</p> <p>"the main thing is to save without compromising the quality of the products purchased"</p> <p>"Delivery on time"</p> <p>Type of Persona Primary</p> </div> </div>
Templates	https://drive.google.com/file/d/0B1YqXzQvZlQuZ2ZJTzZJU0YwVzg/edit?usp=sharing
Information Resources	<p>Cooper, A. (1999), Inmates are running the asylum, The: Why high-tech products drive us crazy and How to Restore the Sanity. Ed Sams.</p> <p>Cooper, A., Reimann R., Cronin D. (2007), About face 3: The essentials of Interaction Design. Ed John Wiley & Sons.</p> <p>Sharp, H., Rogers, Y., Preece J. (2007), Interaction Design: beyond human-computer interaction. Ed John Wiley & Sons.</p> <p>Sanchez-Segura M., Medina-Dominguez F., Amescua A., Mora-Soto A. Improving the efficiency of use of software engineering practices using product patterns. Recovered May 2014 from http://www.sciencedirect.com</p> <p>Caballero, L., Moreno, A., & Seffah, A. (2014). Persona as a Tool to Involving Human in Agile Methods: Contributions from HCI and Marketing. In S. Sauer, C. Bogdan, P. Forbrig, R. Bernhaupt, & M. Winckler (Eds.), Human-Centered Software Engineering SE - 20 (Vol. 8742, pp. 283–290). Springer Berlin Heidelberg. doi:10.1007/978-3-662-44811-3_20</p>

Los *roles* involucrados en el proceso de definición de persona son el equipo ágil incluido el product owner y los usuarios, tanto stakeholders como usuarios finales o representativos de estos. Un diseñador de interfaz puede ser incluido opcionalmente. Los *conocimientos básicos* que los elementos del equipo requieren para aplicar este patrón únicamente se limita a que tengan conocimientos sobre técnicas para adquirir requerimientos. Relacionado con estos conocimientos las habilidades necesarias son la capacidad de aplicar investigación de usuario para obtener datos por medio de observación directa, entrevistas y cuestionarios. También se indica como *habilidad* necesaria la capacidad de análisis de datos identificando necesidades del usuario y estableciendo requerimientos. Los *datos de entrada* que recibe el patrón es la información de los usuarios y la *salida* es la descripción de las personas como modelos de los usuarios. Las *lecciones aprendidas* que se pretende aportar a los elementos del equipo de desarrollo son cuatro.

El *ejemplo* presenta una descripción completa de una persona con la información contenida en todos los campos de la plantilla. Como podemos observar en la figura C4-13 el formato está dividido en cuatro secciones. La primera contiene información sobre algunos detalles de la persona como nombre, edad y ocupación; en la segunda sección se describen de forma narrativa las *actividades* que desarrolla la persona; la tercera sección corresponde a la descripción también a manera de narrativa corta de *las metas y necesidades* de la persona que espera cubrir al utilizar el producto software; la última sección corresponde a la clasificación del *tipo de persona* en el que se prioriza esta persona para ello considerando dos posibles tipos: primaria o secundaria. De acuerdo a las clasificaciones descritas en el estado del arte la persona primaria identifica el objetivo primario para el diseño de una interface, la persona secundaria identifica a los usuarios que son mayormente satisfechos con la interfaz diseñada para la persona primaria pero que pueden tener necesidades adicionales que pueden cubrirse sin alterar la funcionalidad del producto para servir a la persona primaria.

Persona
Name: Michael Age: 35 Occupation: Accountant
Activities Michael is a 35 years old. He is a management accountant in charge of a company's store. Every day when Michael arrives at his office he verifies the reports of movements registered on the previous day by his assistants and the schedule of payments, purchases and deliveries programed for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in the least time. When the number of items requested exceeds the minimum the requesting area is informed for an approximate date on which they will receive the products requested. There are specific processes for the purchases depending on the kind of product required by the company and Michael applies, helped by his assistants, the corresponding process.	
Goals and Needs using the solution Michael expects to have information about the processes of his department at any moment. He needs it and wants to verify. Michael needs to know ahead of time the departments' requests to verify if the products required are available in the store or, otherwise, buy them for delivery on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants need to know the amount, price and total that at the end of the day were implied in the movements. He expects to have a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with his suppliers and the company's departments. "the main thing is to save without compromising the quality of the products purchased" "Delivery on time"	
Type of Persona	Primary

Figura C4-13 Descripción ejemplo de persona

Los equipos ágiles necesitan obtener información suficiente relacionada con persona pero de una forma que signifique invertir menos tiempo describiendo datos relevantes, por ello las secciones de actividades y la de metas y necesidades de la plantilla propuesta consideran la descripción utilizando una narración corta de tal forma que el equipo no se pierda buscando datos relevantes dentro de una amplia información.

Para poder utilizar esta *plantilla* se indica en el campo correspondiente la liga a través de la cual se puede acceder a la plantilla on-line para descargarla en caso de ser necesario y poder utilizarla. La plantilla a la que se accede y podemos observar en la figura C4-14 contiene información donde indica los datos que deben considerarse en el llenado de la misma.

Persona	
Photo. <i>A Realistic photo representing the user (Not the real user)</i>	Name. <i>Name to identify the user Model</i> Age: <i>Age of the user</i> Occupation: <i>Position held or profession</i>
Activities	
<i>What the user does; frequency and volume</i> <i>Why the user is engaged in the product domain</i>	
Goals in using the solution	
<i>How the user thinks about the product domain and technology</i> <i>Expectations from the product</i> <i>Desired results</i>	
Type of Persona	<i>Primary / Secondary</i> <i>Primary. Represent the primary target for the design of an interface</i> <i>Secondary. Has specific additional needs that can be accommodated without upsetting the product's ability to serve the primary persona</i>

Figura C4-14 Plantilla para la descripción de persona

Por último se proporcionan cinco *fuentes de información* que el equipo puede consultar para obtener mayor información sobre la técnica persona y datos que les puedan servir de ayuda en la aplicación de esta.

4.3.2 Adaptaciones consideradas en el patrón “Agile persona”

En la definición del patrón “agile persona” se tomaron en cuenta las adaptaciones que se aplican a la técnica al usarlas en un entorno ágil identificadas a través de la revisión de la literatura. Las principales adaptaciones son: investigación mínima sobre el usuario, momento de aplicación de la técnica y formato de la plantilla. Estas adaptaciones se describen en la sección 4.2.4.

Las dos primeras adaptaciones se reflejan en la definición del patrón a través de los procesos de la solución propuesta donde la técnica requiere al inicio una *investigación mínima sobre el usuario* y se aplica durante el proceso ágil en dos diferentes etapas: *exploración y refinamiento*. Comenzando en la etapa de exploración con una investigación mínima sobre el usuario en lugar de una específica y rigurosa para definir la(s) persona(s). Posteriormente las descripciones de persona pueden ser actualizadas, en caso de que sea necesario, durante una segunda etapa: la de refinamiento. En cuanto al *formato de la plantilla*, la proporcionada para usarse en el patrón se estructuró considerando la sencillez que los equipos ágiles requieren para describir a las personas de una manera simple y rápida.

Además de las adaptaciones anteriores, en su definición el patrón considera una más en relación a la *clasificación de las personas*. Cooper (Cooper et al., 2007) establece como un principio de diseño enfocarlo por cada interface solamente a una persona primaria. Cooper presenta los diferentes tipos de persona descritos en 4.2.2.1 para priorizar y determinar los usuarios objetivo esto es, la persona primaria. Para elegir la persona primaria un proceso de eliminación es aplicado probando cada persona comparando sus metas con las de otros. La persona secundaria tiene necesidades adicionales específicas que pueden ser cubiertas sin afectar los servicios proporcionados a la persona primaria. Tipos de necesidades adicionales son satisfechos por la solución para una persona primaria identificada, personas del tiempo “Cliente” y “Servida” tienen el tratamiento de personas secundarias. Personas negativas identifican tipos de usuarios para los que el producto no está siendo diseñado. Por su parte Pruitt et al. (Pruitt & Adlin, 2010) además de la persona primaria mencionan dos tipos más: Anti-persona y no primaria (o secundaria) anti-persona es referenciado como persona negativa por Cooper.

Las personas clasificadas como no-primaria son aquellas que usan el producto con menos frecuencia o utilizan en particular alguna interfaz como parte de su trabajo. Pruitt et al. (2010) argumentan que si se está claro el rol de las personas no-primarias y el hecho de que no serán descartadas a pesar de no ser elegidos, las decisiones de priorización pueden resultar en un limitado grupo de personas primarias. Cooper destaca el hecho de identificar a las personas primarias como crucial. Personas de tipo suplementarias, clientes y servidas son cubiertas como personas secundarias y persona negativa es un tipo asignado para comunicar que una persona de esta clasificación definitivamente no es el objetivo del diseño.

Decidimos considerar para el patrón solamente la clasificación de personas primaria y secundaria para facilitar la priorización de las personas definidas. Durante las iteraciones la clasificación asignada a las personas puede cambiar mejorando el proceso de identificación de los usuarios finales. Anti-persona o persona negativa es una clasificación no importante a considerar en el desarrollo ágil, los esfuerzos del equipo deben estar enfocados en usuarios directamente relacionados con el software a desarrollar. Clasificar una persona como no-primaria puede dar lugar a confusión sobre la importancia de personas que no son precisamente la primaria. Para los equipos ágiles un importante factor es definir las historias de usuario a considerar en el product backlog a desarrollar durante las iteraciones lo cual se realiza de manera más rápida si el equipo identifica los usuarios finales tan pronto como sea posible.

Aunque el proceso de construcción de personas ha sido adaptado al desarrollo ágil estas adaptaciones aún reflejan una fuerte alineación con la esencia del objetivo de la herramienta misma que refleja la relación con los pasos definidos por Cooper y las fases del ciclo de vida de persona presentado por Pruitt et al. La tabla C4-5 nos permite visualizar la relación de la información contenida en el patrón listada en la columna “Patrón Agile Persona” con los pasos de Cooper y las fases de Pruitt indicadas en la columna correspondiente. Como podemos ver en la tabla C4-5 los pasos de Cooper son completamente cubiertos aunque en diferente orden debido a la naturaleza ágil y las fases de Pruitt son aplicadas de diferentes formas.

Tabla C4-5 Relación del patrón persona con los pasos de Cooper y las fases del ciclo de vida de Pruitt

Patrón “Agile Persona”	Pasos de Cooper (Cooper et al. 2014)	Fases de Pruitt (Pruitt & Adlin, 2010)
El equipo de desarrollo obtiene información relevante realizando una investigación mínima sobre el usuario.	2. Identificar variables de comportamiento 3. Mapear las entrevistas de los sujetos a variables de comportamiento	Planificación de la familia (Family Planning)
El equipo de desarrollo define persona como modelos del usuario con arquetipos hipotéticos describiendo las características del usuario identificando las necesidades, metas y variables de comportamiento.	4. Identificar patrones significativos de comportamiento 5. Sintetizar características y metas relevantes	Concepción de persona (Persona Conception)
El dueño del producto (product owner) revisa junto con el equipo de desarrollo la descripción de persona.	6. Verificar redundancia y complementar información	Gestación de persona (Persona Gestation)
El equipo de desarrollo junto con el dueño del producto (product owner) prioriza personas identificadas (primaria o secundaria)	7. Designar tipos de persona	Gestación de persona (Persona Gestation)
El equipo de desarrollo refina las definiciones de persona, si es necesario, actualizando las descripciones de los patrones existentes utilizando la retroalimentación obtenida después de cada iteración o de acuerdo a las observaciones del product owner. Incluso puede ser necesario crear nuevos patrones de personas.	8. Ampliar la descripción de atributos y comportamientos	Gestación de persona (Persona Gestation)

A pesar de que los cambios reflejados, principalmente en la precisión de la investigación de usuario aplicada inicialmente y antes de la definición de personas, la esencia del diseño para asegurar un nivel adecuado de usabilidad en el software enfocado al usuario final se preserva. La etapa de planificación de la familia y los pasos iniciales de Cooper para identificar las variables comportamiento y mapear a los sujetos con éstas se cubren en un desarrollo ágil con la investigación mínima mencionada.

La fase Concepción y gestación de Pruitt es la que más se cubre en nuestro patrón y que a su vez cada una de ellas tiene sus propios pasos. La concepción es considerada en el momento cuando el equipo de desarrollo define persona describiendo características, necesidades, metas y variables de comportamiento. Esta es señalada por Cooper como la actividad de identificar patrones significativos de comportamiento así como sintetizar características y metas relevantes (pasos 4 y 5). La gestación se aborda desde el proceso de priorización, la revisión de la definición del product owner y el refinamiento de descripciones previas de persona. En términos de los pasos de Cooper la persona se prioriza y las descripciones se verifican buscando redundancia y se complementan para ampliar la descripción de los atributos y comportamientos (pasos 6, 7 y 8).

Aunque la fase de nacimiento y maduración no se sigue exactamente como Pruitt señala, podemos considerar que esta fase se hace presente en nuestro patrón Agile Persona por la disponibilidad de las definiciones de persona al equipo ágil y usuarios. De esta manera persona es utilizada como una potente herramienta de comunicación que ayuda al equipo a incorporar las personas dentro de su diseño y proceso de desarrollo. Los elementos del equipo mantienen en mente a los usuarios objetivos.

La etapa adulta de persona se refiere a la fase en la que se asegura que las personas identificadas y definidas son utilizadas; el equipo es provisto con procedimientos relacionados con persona, instrucciones, líneas guía, plantillas y herramientas que ellos pueden fácilmente entrelazar con sus demás tareas. La plantilla propuesta para describir personas será utilizada durante todo el proceso de desarrollo aseguran que las personas son usadas.

Nuestro patrón propuesto tiene por objetivo ayudar a los equipos ágiles a identificar a los usuarios finales, el uso de las definiciones de persona después del final del proyecto de software no ha sido considerado todavía en la descripción del patrón por lo que no se describe relación alguna con la fase de logros, reutilización y jubilación que se contempla en el ciclo de vida de persona descrito por Pruitt et al. (2010).

4.3.3 Estructura del Patrón “Agile persona”

Name	Agile persona	Related pattern	None
Initial context	This pattern can be used in agile projects to help development team to learn and understand about the user's needs, goals and behavior in order to identify the final users and develop the right functionalities for them with useful and usable interfaces.		
Result Context	Specific models descriptions of the final users' features will be obtained. Such descriptions allow the development team to know about the users for whom the software will be developed. Those descriptions will be available to review at any moment through the agile development process.		
Problem	The agile development team must be able to identify and define as soon as possible the final users for whom the software will be developed in this way avoiding wasting time on reworking the development process.		
Forces	Kind of the organization This pattern can be used in agile projects regardless of company size. Kind of system to be developed This pattern can be used in agile projects that use development methodologies following the agile philosophy. Kind of client The user must be available and always involved through the process of the persona's definition.		
Solution			

In agile environment, incorporating persona involves different activities: develop minimal user research, describe and prioritize persona descriptions, review persona definitions, update description/prioritize persona and obtain a detailed definition of the user story. Figure 1 depicts how persona is applied through these processes.

In order to construct the user models we applied persona technique by following the next steps:

1. Minimal User Research: development team develops minimal user research using artifacts such as interviews and observation in order to get just basic information about the user in order to identify relevant information to define user models. These models will be improve through subsequent iterations.
2. Describe and prioritize persona: using a template provided as part of the pattern the development team describe the user's features, needs, goals and behaviors. Then, development team prioritizes personas identified.
3. Review persona definition: product owner and development team review the persona's descriptions and prioritization assigned.
4. Update description/ prioritize persona: if a refinement is necessary, the personas' descriptions are updated and prioritized again according to the feedback from previous review or product owner's observations.
5. Detailed definition of the user story: once the persona's description is finished then, tasks and acceptance criteria related with this can be detailed. In this way the user stories remain relative to their corresponding persona.

Figure 1- How to Apply persona at Agile environment

Figure 2 depicts an activity diagram which shows who develops the corresponding activities regarding the roles involved. In general the activities are performed as described below:

- The development team receives information previously gathered from the Minimal User Research (observation and interviews).
- The development team defines persona as user models with hypothetical archetypes describing the user's features identifying needs, goals and behavior variables.
- The development team prioritizes personas identified.
- The user review together with development team, the persona's description and prioritization.

- If it is necessary the development team refines persona's definitions by creating a new persona product pattern or by updating existing patterns after each iteration feedback or product owner's review.
- User stories can then be detailed according to the personas definition. This activity is developed by both product owner (establishing acceptance criteria) and development team (defining tasks).

Figure 2 Who are involved defining persona pattern

Figure 3 depicts a general way of the agile development process highlighting the moment to apply persona with a gray background. In this diagram sprint planning and sprint development has an asterisk (*) and a double asterisk (**) respectively to indicate a reference to the processes shown in figure 4 and 5 where persona technique is also applied. In the diagrams the process named “Apply persona” involves the activities described previously and depicted in figure 1.

Persona technique is applied at the beginning of the development process in an explorative way in order to get information about the users visualizing the general scope of the project considering this information to define initial product backlog and interface design. The agile team decides how deeply to conduct the user research to get necessary information.

Figure 3 When to apply persona through agile development process

At this moment, *the beginning of the development process*, “Apply persona” implies doing the following tasks according to the processes described previously with reference to figure 1:

1. Do a minimum user research to get relevant information in order to define initial user's descriptions.
2. Once possible target users are identified then describe them and prioritize these descriptions.
3. Review the persona definitions. This task is performed by the development team together with the product owner.
4. If a necessary refinement to persona is identified an update to the definition is applied according to the feedback gathered from previous iterations or product owner's review. Verify the prioritization of persona

again and update it if necessary.

5. Define the user story details by specifying tasks and acceptance criteria related to it.

Persona is applied in an explorative way at sprint planning and sprint development. At this time applying this technique aims to verify and improve previous persona's descriptions. Figure 4 depicts when persona is applied in iteration planning and figure 5 when persona is applied in iteration development.

Figure 4 - When apply persona at iteration planning

Figure 5 - When apply persona at iteration development

During *iteration planning* the technique is applied by developing the following tasks:

1. Review feedback from previous iterations (if there have been previous iterations)
2. Relate the personas' definition with the user stories to develop
3. Determine if refinement is needed or creation for a new persona definition
 - a. If it is necessary to create a new persona, then minimal user research is applied and persona's description defined, prioritized and reviewed.
 - b. If refinement to persona is necessary then update the definition according to the feedback gathered from the previous iteration or product owner's review. Verify the prioritization of persona and updated it if necessary. This verification is applied by product owner and development team.
4. Define the user story details by specifying tasks and acceptance criteria related to it.

At *iteration development* persona is applied by developing the following tasks. For each user story to develop in the iteration we are going to:

1. Relate the personas' definition with the user stories to develop. Verifying definitions with the personas related with the users involved in the user story.
2. Validate the refinement needed or creation of a new persona definition
 - a. If it is necessary create a new persona, then minimal user research is applied
 - b. If a necessary refinement to persona is identified we update the definition according to the feedback gathered from the previous iteration or product owner's review. Verify the prioritization of persona and updated it if is necessary.
3. Define the user story details by specifying tasks and acceptance criteria related to it.
4. Development team decides to implement the user story or continue to design next user story.

During iteration development parallel tracks of user stories implementation or design can be worked at the same time if the development team has enough elements and timeboxing.

When the last user story of the iteration backlog is done a shippable product is ready to be delivered. Feedback from persona incorporation through the iteration is available to be used at the iteration backlog for the next iteration(s).

Roles	<ul style="list-style-type: none"> ▪ Development Team (Agile Developers, Scrum Master, Product owner, etc.) ▪ Users (stakeholders, end-users, representative users) ▪ UI Designer (optional)
Basic Knowledge	<p>Knowledge</p> <ul style="list-style-type: none"> ▪ Knowing about requirements gathering <p>Skills</p> <ul style="list-style-type: none"> ▪ Capability to apply user research to get data through direct observation, interviews and questionnaires.

	<ul style="list-style-type: none"> ▪ Capability of data analysis identifying needs and establishing requirements 										
Entries	User information										
Exit	Persona's descriptions as user Models										
Lessons learned	<ul style="list-style-type: none"> ▪ The whole development team must be involved in the personas definition. ▪ After the UI designers and/or developers have prioritized the personas then the models have to be presented for review in a meeting of all the development team and users. ▪ Users, UI designers and developers must be involved in the presentation, reviews and feedbacks meetings. ▪ Meetings for interviews, product pattern's review or feedback between development team and user must be established in a friendly and trusting environment. ▪ The product pattern persona must be kept available to the team through the whole development process. 										
Examples											
<table border="1"> <thead> <tr> <th colspan="2">Persona</th></tr> </thead> <tbody> <tr> <td> </td><td> Name: Michael Age: 35 Occupation: Accountant </td></tr> <tr> <td colspan="2"> Activities Michael is a 35 years old. He is a management accountant in charge of a company's store. Every day when Michael arrives at his office he verifies the reports of movements registered on the previous day by his assistants and the schedule of payments, purchases and deliveries programed for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in the least time. When the number of items requested exceeds the minimum the requesting area is informed for an approximate date on which they will receive the products requested. There are specific processes for the purchases depending on the kind of product required by the company and Michael applies, helped by his assistants, the corresponding process. </td></tr> <tr> <td colspan="2"> Goals and Needs using the solution Michael expects to have information about the processes of his department at any moment. He needs it and wants to verify. Michael needs to know ahead of time the departments' requests to verify if the products required are available in the store or, otherwise, buy them for delivery on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants need to know the amount, price and total that at the end of the day were implied in the movements. He expects to have a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with his suppliers and the company's departments. "the main thing is to save without compromising the quality of the products purchased" "Delivery on time" </td></tr> <tr> <td>Type of Persona</td><td>Primary</td></tr> </tbody> </table>		Persona			Name: Michael Age: 35 Occupation: Accountant	Activities Michael is a 35 years old. He is a management accountant in charge of a company's store. Every day when Michael arrives at his office he verifies the reports of movements registered on the previous day by his assistants and the schedule of payments, purchases and deliveries programed for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in the least time. When the number of items requested exceeds the minimum the requesting area is informed for an approximate date on which they will receive the products requested. There are specific processes for the purchases depending on the kind of product required by the company and Michael applies, helped by his assistants, the corresponding process.		Goals and Needs using the solution Michael expects to have information about the processes of his department at any moment. He needs it and wants to verify. Michael needs to know ahead of time the departments' requests to verify if the products required are available in the store or, otherwise, buy them for delivery on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants need to know the amount, price and total that at the end of the day were implied in the movements. He expects to have a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with his suppliers and the company's departments. "the main thing is to save without compromising the quality of the products purchased" "Delivery on time"		Type of Persona	Primary
Persona
Name: Michael Age: 35 Occupation: Accountant										
Activities Michael is a 35 years old. He is a management accountant in charge of a company's store. Every day when Michael arrives at his office he verifies the reports of movements registered on the previous day by his assistants and the schedule of payments, purchases and deliveries programed for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in the least time. When the number of items requested exceeds the minimum the requesting area is informed for an approximate date on which they will receive the products requested. There are specific processes for the purchases depending on the kind of product required by the company and Michael applies, helped by his assistants, the corresponding process.											
Goals and Needs using the solution Michael expects to have information about the processes of his department at any moment. He needs it and wants to verify. Michael needs to know ahead of time the departments' requests to verify if the products required are available in the store or, otherwise, buy them for delivery on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants need to know the amount, price and total that at the end of the day were implied in the movements. He expects to have a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with his suppliers and the company's departments. "the main thing is to save without compromising the quality of the products purchased" "Delivery on time"											
Type of Persona	Primary										
Templates	https://drive.google.com/file/d/0B1YqXzQvZlQuZ2ZJTzZJU0YwVzg/edit?usp=sharing										
Information Resources	<p>Cooper, A. (1999), Inmates are running the asylum, The: Why high-tech products drive us crazy and How to Restore the Sanity. Ed Sams.</p> <p>Cooper, A. ,Reimann R., Cronin D. (2007), About face 3: The essentials of Interaction Design. Ed John Wiley & Sons.</p> <p>Sharp, H., Rogers, Y., Preece J. (2007) , Interaction Design: beyond human-computer interaction. Ed John Wiley & Sons.</p> <p>Sanchez-Segura M., Medina-Dominguez F., Amescua A., Mora-Soto A. Improving the efficiency of use of software engineering practices using product patterns. Recovered May 2014 from http://www.sciencedirect.com</p> <p>Caballero, L., Moreno, A., & Seffah, A. (2014). Persona as a Tool to Involving Human in Agile Methods: Contributions from HCI and Marketing. In S. Sauer, C. Bogdan, P. Forbrig, R. Bernhaupt, & M. Winckler (Eds.), Human-Centered Software Engineering SE - 20 (Vol. 8742, pp. 283–290). Springer Berlin Heidelberg. doi:10.1007/978-3-662-44811-3_20</p>										

Pattern's fields description

Name: pattern name, it could be a word or short phrase

Related pattern: name (s) of the pattern(s) which is related with

Initial context: preconditions under which the pattern is applicable - a description of the initial state before the pattern is applied

Result Context: state or configuration of the system after the pattern has been applied.

Problem: description of the problem to be solved

Forces: restrictions around the execution of this pattern

Solution: describe how to achieve the objective.

Roles: people or participants involved

Entries: previously obtained products necessary to develop this pattern

Lessons learned: Experiences gathered while using this pattern in previous projects

Templates: templates that can be used to obtain the exit of this pattern

Examples: sample applications of the pattern

Exit: Product obtained when the pattern is used

Information resources: references and documentation (i.e. books, papers) used to develop this product.

Basic knowledge: Previous knowledge required to understand and use this pattern

Capítulo 5 Formalización de la solución: Patrón “Agile Usability Testing”

5.1 Introducción

El presente capítulo presenta información relativa a la formulación del patrón “Agile Usability Testing” incluyendo la revisión de la definición original de la técnica pruebas de usabilidad y las adaptaciones que se han aplicado al utilizarla en proyectos ágiles. La sección 5.2 describe las características propias de la técnica pruebas de usabilidad (en inglés Usability Testing) y la forma en la que los equipos ágiles la incorporan en su entorno de desarrollo. En la sección 5.3 se aborda la concepción del patrón detallando los datos contenidos en su estructura y describiendo las principales adaptaciones de uso en desarrollos ágiles consideradas en su definición. Finalmente se presenta patrón completo incluyendo las mejoras identificadas a partir de la retroalimentación de la validación aplicada a éste. Usability Testing

5.1.1 Usability Testing in HCI

Usabilidad generalmente se considera como el asegurar que los productos interactivos son fáciles de aprender, eficaz al utilizarlo y agradable desde la perspectiva del usuario (Sharp et al., 2007). Diferentes herramientas UCD están disponibles para ayudar a alcanzar un importante nivel de usabilidad en los productos. La usabilidad se puede mejorar conociendo cómo los usuarios utilizan y perciben el software, que tan fácil es para ellos desarrollar tareas con el objetivo de satisfacer sus necesidades de información cubriendo metas específicas. Los autores abogan por la importancia de la opinión del usuario a través de todo el proceso completo del desarrollo de software (Jakob Nielsen, 1993) (Mayhew, 1999).

(Jeffrey et al., 2008) señalan que la meta de las pruebas de usabilidad es identificar y rectificar deficiencias de usabilidad. No obstante su origen con enfoque experimental cuando las pruebas de usabilidad se realizan en el momento adecuado durante el desarrollo del producto son un indicador casi infalible de potenciales problemas y los medios para resolverlos. Por otro lado Sharp (Sharp et al., 2007) apuntan que esta técnica busca perfeccionar un producto lo más rápido posible. Barnum (Barnum, 2010) describe las pruebas de usabilidad como la actividad que se enfoca en observar a los usuarios trabajar con el producto, desarrollando tareas que tienen un significado real para ellos. Carbon et al. (2004) afirman que cuando las pruebas de usabilidad son parte del diseño y desarrollo el conocimiento obtenido sobre la experiencia del usuario respalda todos los aspectos de diseño y desarrollo (Carbon, Dörr, & Trapp, 2004). Nielsen destaca el involucramiento de usuarios reales en las pruebas de usuario como un método fundamental que proporciona información relacionada con los usuarios y sus problemas con el sistema (Jakob Nielsen, 1993). Dumas et al. en (Dumas & Redish, 1999) hacen hincapié en que la usabilidad requiere usuarios y define las pruebas de usabilidad como una forma sistemática para observar a los usuarios actuales probando un producto y recolectando información sobre las formas específicas en las que el producto es fácil o difícil de utilizar para ellos. Dumas y Redish enfatizan que las pruebas de usuario deben ser aplicadas con frecuencia en etapas tempranas del desarrollo y no solamente al final del proceso de desarrollo.

Con el objetivo de obtener información sobre los usuarios que nos permita identificar las deficiencias de usabilidad y definir actividades para mejorarla se han propuesto un conjunto de características y pasos para apoyar en la aplicación de pruebas de usuario efectivas. (Jakob Nielsen, 1993) describe tareas involucradas en el proceso de prueba de usabilidad, (Dumas & Redish, 1999) señala características de las pruebas de usabilidad y (Jeffrey et al., 2008) identifica elementos básicos. Independientemente de la perspectiva los autores proporcionan elementos para ayudar a definir eficientemente las pruebas de usabilidad. La tabla C5-1 muestra las características descritas por estos autores.

Como podemos ver el primer paso común es definir la meta o metas de la prueba en relación con el objetivo de evaluación, esto es, con qué fin se aplicará la prueba de usabilidad. (Jakob Nielsen, 1993) y (Barnum, 2010) describen dos tipos de evaluación para clarificar el propósito de la prueba: formativa y sumativa. La *formativa* para aprender cuales aspectos detallados de la interfaz son buenos o malos y ayudar a mejorar el diseño (aplica como parte de un proceso de diseño iterativo mientras el producto está en desarrollo) y *sumativa* la cual tiene como objetivo evaluar la calidad general de una interfaz (aplica cuando el producto está terminado).

Tabla C5-1 Características de las pruebas de usabilidad

Tareas (Tasks) J. Nielsen (Jakob Nielsen, 1993)	Características (Characteristics) J. Dumas (Dumas & Redish, 1999)	Elementos básicos (Basic elements) J. Rubin (Jeffrey et al., 2008)
<ul style="list-style-type: none"> Definir metas de las pruebas para clarificar el propósito y planear las pruebas Los usuarios de prueba deben ser tan representativos como sea posible de los potenciales usuarios del sistema Elegir experimentadores, alguien tiene que ser el experimentador y estar a cargo de la prueba. Definir las tareas de la prueba lo más representativo posible de los usos en los que eventualmente el sistema apoyará y proveer una cobertura razonable de las partes más importantes de la interfaz de usuario. Medir las propiedades de usabilidad de interés 	<ul style="list-style-type: none"> La meta principal es mejorar la usabilidad de un producto Los participantes representan usuarios reales Los participantes realizan tareas reales Observar grabar lo que los participantes hacen y dicen Analizar los datos, diagnosticar problemas reales y recomendar cambios para resolver estos problemas 	<ul style="list-style-type: none"> Desarrollar preguntas de investigación o los objetivos de la prueba en lugar de hipótesis Utilizar una muestra representativa de usuarios finales los cuales pueden o no ser elegidos aleatoriamente Representación del ambiente de trabajo actual Observación de usuarios finales quienes utilizan o revisan la representación del producto Controlada y, a veces extensa entrevista y el sondeo de los participantes por el monitor de prueba Recolección de datos sobre el desempeño cualitativo y cuantitativo y medidas de preferencia. Recomendación de mejoras al diseño del producto

Jeffrey et al. en (Jeffrey et al., 2008) exponen los siguientes tipos de pruebas:

- Formativa*, se realiza muy temprano en el ciclo de desarrollo cuando un producto está todavía en las etapas preliminares de ser definido y diseñado. El objetivo es examinar la efectividad de los diseños iniciales. En el caso de software estos diseños pueden incluir prototipos, maquetas, representaciones estáticas de pantallas o incluso borradores en papel de las pantallas.
- Evaluativa o sumativa*, se realizan al inicio o a la mitad dentro del ciclo de desarrollo del producto, generalmente después de un diseño fundamental o de alto nivel o cuando la organización del producto ha sido establecida. En lugar de solamente explorar el carácter intuitivo de un producto, el interés es identificar que tan bien un usuario puede realmente desarrollar tareas realistas en toda regla y detectar deficiencias específicas de usabilidad en el producto.
- Validación*, se realiza para evaluar cómo el producto se compara con algunos estándares o marcas predeterminados de usabilidad, ya sea un estándar de rendimiento relacionados con el proyecto, uno interno de la empresa o la norma histórica, o incluso estándar de rendimiento de un competidor. El enfoque central de pruebas de validación es la recolección de datos cuantitativos que representan una de las principales diferencias con las pruebas de evaluación, aunque ambas se llevan a cabo de manera similar.

De la tabla C5-1 podemos destacar las siguientes características sobre la prueba de usabilidad:

- Identificación de las metas de la prueba (evaluación formativa o sumativa)
- Trabajar con usuarios reales
- Planear las prueba
 - Elegir los usuarios que participaran en la prueba (reales o representativos)
 - Definir espacio en donde se realizará la prueba

- Definir tareas a desarrollar durante la prueba
- Recomendar mejoras
- Medir el desempeño

Además de los aspectos mencionados, la prueba de usabilidad es un proceso iterativo (Dumas & Redish, 1999) desde el pre diseño (probando un producto similar o versión anterior), posteriormente a través del diseño inicial (probando prototipos) y durante el desarrollo (probando diferentes aspectos, probando de nuevo los cambios aplicados). También puede utilizarse en conjunto con otros métodos de usabilidad (thinking aloud, observación, cuestionarios, entrevistas) para alcanzar el propósito de la evaluación (Jakob Nielsen, 1993). Jeffrey afirma que la prueba de usabilidad es más poderosa y más efectiva cuando se implementa como parte de un proceso iterativo del desarrollo de un producto (Jeffrey et al., 2008).

La prueba de usabilidad proporciona dos ventajas: primero, trabajar con usuarios reales de los cuales obtener información de primera mano para identificar deficiencias a mejorar en el diseño y proceso de desarrollo. Segundo, siendo un proceso iterativo permite conocer y aprender constantemente sobre las necesidades de los usuarios y corroborar si nuestro desarrollo las está cubriendo de la mejor forma. Aplicando mejoras gradualmente permite visualizar resultados en corto tiempo.

El concepto original de usabilidad busca aplicar un experimento controlado con requerimientos sofisticados para aplicar este tipo de pruebas que lo hace parecer un método caro y que requiere mucho tiempo. Sin embargo, en los últimos años como muchos otros conceptos y técnicas HCI, el método de pruebas de usabilidad ha sido adaptado de acuerdo a las actuales necesidades y enfoques del desarrollo de un producto software. En esta línea, consideraciones relacionadas con ventajas tecnológicas como pruebas remotas sin observadores in-situ son ahora aceptables, los elementos sofisticados como laboratorios portátiles pueden ser reemplazados con modernos y ligeros equipos. Requerimientos como laboratorios de prueba con una estricta distribución no son necesarios para aplicar las pruebas de usabilidad (Jakob Nielsen, 1993) (Dumas & Redish, 1999). En cuanto al número de participantes en las pruebas no es necesario aplicarlas con un amplio número de usuarios. Dumas et al. en (Dumas & Redish, 1999) señala que una prueba típica de usabilidad puede aplicarse con seis a doce participantes organizados en tres grupos de test. Además, Dumas et al. (1999) concuerdan con el señalamiento en (Jakob Nielsen, 1993) en cuanto a que se obtiene un rendimiento sobre el costo de las pruebas al aplicarlas con grupos de tres a cinco participantes. En (Jakob Nielsen, 2000) el autor señala que después de cierto número de pruebas aplicados en grupos de cinco participantes se observa el mismo resultado en cuanto a problemas de usabilidad observados. Por su parte en (Jakob Nielsen & Hinderer Sova, 2003) indican para estudios en los que se requiere obtener resultados estadísticos considerar de 10 a 12 participantes y para estudios de usabilidad menos formales evaluar con cuatro a cinco participantes por grupo de prueba. (Krug, 2005) indica como número ideal tres o cuatro participantes en cada ronda de pruebas. Por su parte (Barnum, 2010) considera suficiente aplicar las pruebas de usabilidad con cinco usuarios de un mismo grupo y, con tiempo y presupuesto suficiente, pueden ser diez participantes divididos en dos grupos o más.

5.1.2 Usability Testing en el Desarrollo Ágil

Como se ha indicado en el capítulo 3, las pruebas de usabilidad son una de las técnicas HCI más utilizadas dentro de los desarrollos ágiles. Su naturaleza iterativa y la importancia de involucrar usuarios reales en las pruebas hacen de éstas una técnica ideal para adoptar en metodologías ágiles. Algunos métodos como Discount Usability (Jakob Nielsen, 1993) y Rapid Iterative Test and Evaluation method (RITE) (Medlock et al., n.d.) que consideran en sus prácticas las pruebas de usabilidad sobre cualquier tipo de desarrollo de software han sido adoptados en los contextos ágiles con buenos resultados (Fisher & Bankston, 2009).

En (T. Da Silva et al., 2011) se identifica que los mecanismos más utilizado por los equipos ágiles como complementos para aplicar las pruebas de usuario son “thinking aloud” y prototipos de alto y bajo nivel. Los autores también corroboraron que los laboratorios de usabilidad considerados como primordiales en el concepto inicial de esta técnica no son utilizados generalmente en el desarrollo ágil para aplicar las pruebas. Otro resultado de estos autores fue que las pruebas de aceptación pueden integrarse con las pruebas de usabilidad.

A partir de los resultados de su estudio (Salah et al., 2014) identifican varios desafíos y factores que restringen la integración de las filosofías Agile y UCD a la que hacen referencia como Agile User Centered Design Integration (AUCDI). Salah et al. (2014) exploran las prácticas propuestas para hacer frente a estos desafíos. En particular identifican las pruebas de usabilidad como uno de estos desafíos con los siguientes sub-desafíos: métodos de prueba de usabilidad, programación de pruebas de usabilidad, acceso a los usuarios para pruebas de usabilidad y período de tiempo más corto para iterar el diseño.

Entre las experiencias compartidas en su trabajo Sy (Sy, 2007) indica que aplicó pruebas de usabilidad formativa en cada iteración a través de un proceso paralelo diseñando la interfaz un ciclo por delante a la implementación. Sy señala como principal adaptación a las técnicas HCI implementadas el cambio en la programación y frecuencia de las investigaciones y la forma en la que son reportados los resultados. (Detweiler, 2007) identificó retos e implementó tres fases para el diseño de la interacción (entendiendo usuarios, definiendo la interacción y diseño de interfaz) en el desarrollo ágil. Para la fase del diseño de la interfaz, donde los prototipos son iterativamente creados, Detweiler sugiere mostrar prototipos a los diseñadores y usuarios finales para obtener retroalimentación temprana y frecuente. Otra práctica sugerida es mantener sesiones de revisión obligatoria después de cada sesión de retroalimentación o prueba de usabilidad. En (Dayton et al., 2009) los autores concluyen a partir de su estudio que aplicar los métodos de usabilidad de forma más ágil representa un reto complicado e identifican la planeación anticipada como obstáculo al incorporar las pruebas de usabilidad en el proceso ágil. Para minimizar este obstáculo sugieren planear anticipadamente sprints de 2 a 4 semanas para visualizar puntos y definir un plan para aplicar pruebas de usabilidad dentro del ciclo de desarrollo; otra sugerencia es utilizar procesos rápidos de prueba iterativa como RITE y programar las pruebas de usabilidad regular y rutinariamente. En relación a la participación de un experto como parte del equipo ellos discuten que es válido para preparar al equipo de desarrollo sobre cómo incorporar el proceso centrado en el usuario utilizando técnicas como las pruebas de usabilidad de forma iterativa.

Hussain et al. (Zahid Hussain et al., 2009) señalan entre otras lecciones aprendidas, que aquellos desarrolladores que asisten en las pruebas de usabilidad obtienen más visión sobre el usuario que otros. Ellos recomiendan desarrollar pequeños test después de algunas iteraciones y pruebas más amplias solamente después de cada liberación (release) cuando no se esperan cambios mayores y el sistema sea bastante estable. Con el objetivo de preparar a los miembros del equipo para entender la importancia de la fase de las pruebas de usabilidad y el proceso que implica sugieren proporcionar un taller al inicio del proyecto o antes de la liberación en la que será aplicada la fase de prueba de usabilidad. Otra sugerencia es tener un pre-plan considerando cuándo desarrollar cierto tipo de prueba de usabilidad. En su estudio Nielsen et al. (L. Nielsen & Madsen, 2012) identificaron mejoras que los expertos en diseño de interfaz implementan para aplicar las pruebas de usabilidad en los proyectos ágiles lo cual representa retos para ellos debido principalmente a las restricciones de tiempo. Señalan que los especialistas en diseño tienen que adaptar sus habilidades en el ambiente ágil con la finalidad de integrar ingeniería de usabilidad para obtener un software usable. Decisiones rápidas, flexibilidad, reportes orales y pruebas realizadas repetidamente durante el proceso de desarrollo son soluciones que permiten la aplicación de pruebas de usabilidad ajustándose al proceso de desarrollo ágil. Nielsen et al. (2012) concluyen que las pruebas de usabilidad tienen que ser desarrolladas como parte de las iteraciones incluso si esto significa reclutar usuarios finales, desarrollar las pruebas, analizar y reportar los resultados de las pruebas en ciclos reducidos.

Con el objetivo de identificar y describir dichos patrones, a partir de la revisión de literatura nos enfocamos en analizar aquellos estudios donde las pruebas de usabilidad fueron utilizadas por los equipos ágiles durante el desarrollo de software. Para efectos de nuestro estudio consideramos los enfoques y etapas de incorporación descritas en secciones previas del estado del arte que nos ayuden a organizar mejor la información sobre cuándo, cómo y quién aplicó la técnica durante el proceso de desarrollo en el entorno ágil. En el caso del momento en el que se aplicaron las pruebas de usabilidad consideramos los enfoques y las etapas descritos en las secciones de este trabajo 2.4 y 4.2.3 respectivamente también presentadas en (Caballero et al., 2014). Los autores identifican cuatro enfoques –Interaction design first, Minimal up-front design, Implicit interaction design, parallel track- en los que se incorpora el diseño centrado en el usuario al proceso de desarrollo ágil; así mismo se manejan dos etapas –exploratoria y de refinamiento- en las que pueden usarse las técnicas durante los enfoques mencionados. En relación con el responsable del diseño de interfaz en el equipo ágil consideramos los roles descritos en la

sección 2.5 - Experto y no experto. Con el fin de considerar una clasificación para el tipo de pruebas de usabilidad aplicadas en el desarrollo ágil nos basamos en la clasificación presentada por (Jeffrey et al., 2008) y descritas en la sección 5.1.1.

La tabla C5-2 proporciona un extracto del análisis de los trabajos donde encontramos información relevante sobre el uso de las pruebas de usabilidad en proyectos ágiles. Esta tabla muestra información sobre el enfoque de integración de las actividades HCI durante las actividades de desarrollo (HCI and Agile); la fase del proceso en el que se aplicaron las pruebas de usabilidad dentro del proceso de desarrollo ágil de software (Testing agile moment); el tipo de prueba de usabilidad aplicada (Type of usability tests); cuál fue el rol responsable de aplicar la técnica dentro del ambiente de desarrollo ágil (HCI rol); y por último los mecanismos utilizados como apoyo para las pruebas de usabilidad (HCI mechanisms).

Tabla C5-2 Extracto del análisis sobre el uso de Pruebas de usabilidad en proyectos ágiles

Author/s	Ref.	HCI and Agile	Testing Agile Moment	Type of usability Test	HCI Rol	HCI Mechanisms
Miller Lynn	(Miller, 2005)	Parallel track	Refinement phase	Formative Assessment	Expert	Low-fi prototypes (paper prototypes) High-fi prototypes
McInerney et al.	(McInerney & Maurer, 2005)	Not specified	Refinement phase	Assessment	Expert	Not specified
Meszaros et. al.	(Meszaros & Aston, 2006)	Minimal Up-front design	Exploratory phase Refinement phase	Formative Assessment	No expert	Low-fi prototypes (paper prototypes) Wizard of Oz
Chamberlain et. al.	(Chamberlain et al., 2006)	Minimal Up-front design Parallel track	Refinement phase	Formative Assessment	Expert	Low-fi prototypes
J. Ferreira et. Al	(Ferreira et al., 2007a)	Parallel track	Exploratory phase Refinement phase	Formative Assessment Validation	Expert	HTML mock-up Low-fi prototypes
Lee et. al.	(Jason Chong Lee & McCrickard, 2007)	Implicit Interaction Design	Refinement phase	Assessment Validation	No expert	Expert walkthroughs Heuristic Evaluation High-fi prototypes
Williams et. al.	(H. Williams & Ferguson, 2007)	Parallel track	Refinement phase	Formative Assessment	Expert	Low-fi prototypes (paper prototype, wireframe) High-fi prototypes (Interactive prototypes)
Ungar et. al.	(Ungar & White, 2008)	Parallel track	Exploratory phase Refinement phase	Formative Validation	Expert	Prototypes
Hussain et. al.	(Z Hussain et al., 2008)	Parallel track	Refinement phase	Formative Assessment Validation	Expert	Low-fi Prototypes (paper mockup) High-fi prototypes
Fox et. al.	(Fox et al., 2008)	Parallel track	Exploratory phase Refinement phase	Formative Assessment Validation	Expert / No expert	Low-fi prototypes High-fi prototypes
Illmensee et. al.	(Illmensee & Muff, 2009)	Parallel track	Refinement phase	Assessment	Expert	Low-fi prototypes High-fi prototypes (Semi functional prototypes)
Belchev et al.	(Belchev & Baker, 2009)	Implicit Interaction Design	Refinement phase	Assessment	Expert	Thinking Aloud High-fi prototypes (functional software)
Federoff et. al.	(Federoff et al., 2009)	Parallel track	Exploratory phase Refinement phase	Formative Assessment	Expert	High-fi prototypes (Interactive prototypes)
B. Losada et. al.	(Losada et al., 2013)	Parallel track	Exploratory phase Refinement phase	Formative Assessment Validation	Expert	Low-fi prototype (paper prototype) Thinking Aloud

Como podemos observar los resultados mostrados en la tabla C5-2 indican que los equipos ágiles principalmente desarrollan actividades de implementación y diseño en bloques paralelos con el diseño de la interfaz definida en ciclos por adelantado. (Miller, 2005) describe cómo el equipo ágil incorporó usabilidad en un proyecto interconectando el diseño y la implementación paralelamente. Chamberlain et al. (Chamberlain et al., 2006) trabajaron con tres proyectos e identificaron dentro de las mejores prácticas pruebas de usabilidad sobre prototipos en papel y el trabajo en paralelo de diseñadores y desarrolladores. En (Z Hussain et al., 2008) con relación al proceso iterativo, los autores mencionan que iteraciones cortas permiten que los desarrolladores sean capaces de refactorizar el sistema continuamente en paralelo con la retroalimentación de los diseñadores de interfaz. (Illmensee & Muff, 2009) señalan que trabajar ciclos iterativos por adelantado permite a los diseñadores obtener retroalimentación y ser capaces de resolver rápidamente problemas imprevistos de diseño dentro de las iteraciones.

También observamos que en el entorno ágil principalmente aplican pruebas de usabilidad en la fase de refinamiento. Los especialistas entrevistados en el estudio de McNerney et al. (McNerney & Maurer, 2005) comentaron que las pruebas de usabilidad en sus proyectos ágiles se realizan generalmente en cada iteración o después de algunas iteraciones es decir, por liberación de un producto potencialmente útil (release). Uno de los equipos de estudio en (Chamberlain et al., 2006) programó pruebas de usabilidad iterativas con constante retroalimentación durante el desarrollo. (Jason Chong Lee & McCrickard, 2007) señalan que en los desarrollos estudiados las evaluaciones de usabilidad son aplicadas más en profundidad al final de cada liberación. (Z Hussain et al., 2008) expone su enfoque iterativo y cómo las interfaz de usuario son probadas durante el proceso de desarrollo incrementando y evolucionando de esta manera la fidelidad de los prototipos a ser desarrollados. (Illmensee & Muff, 2009) describen como el equipo sobrelleva los problemas adoptando un proceso de desarrollo ágil e incorporando técnicas HCI en ese contexto para ayudar a los equipos a mejorar la calidad de la funcionalidad. Con el fin de recoger y utilizar la retroalimentación de los usuarios reales se aplicaron pruebas de usabilidad por cada iteración.

A pesar de que el análisis indica que las pruebas se aplican mayormente como una fase de refinamiento, aplicar pruebas de usuario al inicio del desarrollo también ha dado buenos resultados. (Meszaros et al., 2006) señalan que aplicar pruebas de usuario al inicio del proceso de desarrollo ágil puede ayudar a entender la visión del proyecto e identificar características que originalmente podrían ser consideradas fuera del alcance del proyecto. Los autores anteriores describen el proceso que siguieron para implementar pruebas de usabilidad y destacan lo útil que fue utilizar prototipos de baja fidelidad al inicio del proyecto y durante el proceso de desarrollo de manera iterativa para refinar el software. (Ferreira et al., 2007a) reportan que las iteraciones ágiles facilitan aplicar pruebas de usabilidad permitiendo a los desarrolladores de software incorporar resultados de pruebas en iteraciones posteriores. En (Najafi & Toyoshiba, 2008) el equipo UX aplicó pruebas de usabilidad al inicio del proceso de desarrollo durante el sprint 0 y en iteraciones posteriores con el fin de refinar el diseño. (Ungar & White, 2008) describen su experiencia complementando el diseño en un proceso ágil con el método de diseño de estudio con el objetivo de mejorar la colaboración para explorar alternativas de diseño propuestas entre los mismos diseñadores, desarrolladores y los stakeholders. Al inicio trabajaron con el sprint 0 donde las pruebas de usabilidad fueron aplicadas, aunque no especifican el proceso de aplicación. Los autores mencionan que durante las iteraciones las pruebas de usabilidad también fueron aplicadas un ciclo adelantado con los usuarios finales desarrollando tareas y grabando la prueba.

Otro dato importante a recalcar es el hecho de que ya sea en la fase exploratoria o de refinamiento pueden ser aplicadas pruebas de usabilidad formativa, de evaluación o de validación. Barnum (Barnum, 2010) señala la relación de pruebas de usabilidad formativa con pequeños estudios de investigación sobre el usuario los cuales son mucho más apropiados para aplicar en desarrollos ágiles. Pruebas formativas son aplicadas al inicio del proceso de desarrollo, por ejemplo en el ciclo 0, o al inicio de cada iteración. Las pruebas de evaluación son las más utilizadas, los equipos ágiles las aplican principalmente al final de las iteraciones es decir, en la fase de refinamiento. Cuando hay tiempo o el equipo determina la necesidad de aplicar las pruebas de usabilidad de tipo evaluación se aplican después de las liberaciones (o releases). Las pruebas de validación son usadas generalmente al final de cada iteración. (Jason Chong Lee & McCrickard, 2007) describen la aplicación de un marco de trabajo que considera evaluaciones de usabilidad con los usuarios, aplican éstas por medio de

evaluaciones ligeras de usabilidad sobre el software funcionando al final de cada iteración y al final de cada liberación (release). (Najafi & Toyoshiba, 2008) describen cómo implementar mecanismos de usabilidad ayuda a desarrollar software exitoso que cubre las necesidades y metas de los usuarios; comentan que las pruebas de usuario fueron aplicadas de dos formas: desarrollando tareas específicas y procesos que los mismos usuarios elegían realizar en el software al momento de la prueba. Este proceso de prueba fue aplicado en cada sprint aunque no especifican si también aplicaron pruebas de usabilidad en el sprint 0.

Continuando con los datos presentados en la tabla C5-2 podemos observar que el rol de Experto tiene fuerte presencia en los proyectos ágiles analizados en nuestro estudio. Sin embargo, si no hay un elemento con experiencia en el diseño, el equipo ágil puede ser entrenado y aprender como incorporar técnicas HCI como las pruebas de usabilidad iterativas obteniendo resultados válidos (Dayton et al., 2009). (Meszaros et al., 2006) señalan que no pudieron contar con un diseñador de interfaz pero el Project manager y el agile coacher hicieron el trabajo de construir un prototipo inicial que representó un factor clave para el proyecto completo.

En relación a los mecanismos HCI utilizados para soportar las pruebas de usabilidad, la tabla C5-2 muestra que los prototipos (de baja fidelidad como prototipos en papel y de alta fidelidad como prototipos interactivos) y el método thinking aloud son utilizados principalmente como apoyo en dichas pruebas de usabilidad. (Miller, 2005) trabajó en versiones liberadas previamente mejorando las características utilizando prototipos de baja fidelidad en las pruebas de usabilidad. (Z Hussain et al., 2008) mencionan los prototipos en papel como mecanismos para obtener conceptos básicos correctos. (H. Williams & Ferguson, 2007) reportan su experiencia trabajando con un equipo ágil distribuido, ellos utilizan prototipos interactivos (prototipos de alta fidelidad) para aplicar las pruebas de usabilidad y obtener retroalimentación que aplica en el refinamiento del diseño en la siguiente iteración. Otro dato que destacan es el haber trabajado con un grupo de usuarios representativos y algunos expertos. En (Ferreira et al., 2007a) se analizan cuatro diferentes proyectos pero tres de ellos aplican en una manera formal las pruebas de usabilidad utilizando como mecanismos las maquetas, prototipos de baja fidelidad y pruebas de aceptación. (Jason Chong Lee & McCrickard, 2007) combinaron las evaluaciones con pruebas de aceptación, de esta forma los clientes verificaron la funcionalidad y proporcionaron retroalimentación sobre usabilidad.

(Fox et al., 2008) observan resultados de las pruebas de usabilidad al aplicarlas con investigación contextual y prototipos de baja fidelidad (prototipos en papel) para identificar y corregir problemas de usabilidad antes de que el diseño inicial de la interfaz se proporcione a los desarrolladores para su implementación. Posteriormente durante las iteraciones las pruebas de usabilidad se aplican sobre prototipos de alta fidelidad incluyendo software funcional.

(Federoff & Courage, 2009) comparten su experiencia trabajando pruebas de usabilidad con prototipos interactivos en un desarrollo de transición a una metodología ágil, obtuvieron resultados satisfactorios implementando este enfoque para pruebas de usuario teniendo disponible un grupo de usuarios y trabajando el desarrollo en bloques paralelos. (Illmensee & Muff, 2009) reportan la adaptación del espacio implementado para aplicar las pruebas de usabilidad en el proceso de desarrollo ágil como cuartos de conferencia en lugar de laboratorios de usabilidad. (Belchev & Baker, 2009) describen su experiencia implementando técnicas de usabilidad en un software ya liberado. En su proyecto registraron las sesiones de las pruebas de usabilidad en audio y video. Las pantallas de los usuarios y movimientos del ratón fueron capturados por un software y la técnica “thinking aloud” fue usada con la finalidad de grabar los comentarios de los usuarios. Además comentan que un video clip corto de usuarios desarrollando una tarea puede ayudar a tomar decisiones y obtener apoyo de los stakeholders. Losada et al. (2013) describen el desarrollo de una aplicación móvil utilizando InterMod, una metodología ágil que implementa técnicas de usabilidad a través del proceso iterativo. Las pruebas de usabilidad son aplicadas con potenciales usuarios finales a quienes se les pidió desarrollar tareas específicas utilizando prototipos en papel y thinking aloud; las mejoras fueron desarrolladas en bloques paralelos de diseño e implementación (Losada et al., 2013).

5.1.3 Adaptaciones al incorporar la técnica usability testing en el proceso ágil

Con el objetivo de incorporar con éxito las pruebas de usabilidad en un ambiente de desarrollo ágil se han sido realizado varias adaptaciones al usarlas, a partir de nuestro estudio identificamos las siguientes como las principales:

El enfoque de las pruebas de usabilidad aplicadas, para los equipos ágiles resulta más más útil y efectivo aplicar pequeños e iterativos estudios para obtener información cualitativa de la retroalimentación de los usuarios. Esto es, las pruebas de usabilidad son aplicadas como un enfoque formativo. Sin embargo, el enfoque sumativo no es negativo para el desarrollo de software ágil y puede ser aplicado si se cuenta con tiempo para ello. Barnum et al. (Barnum, 2010) señala algunas características de los estudios pequeños de usabilidad que pueden ser aprovechados en ágil:

- a) Se incorporan dentro del desarrollo del producto a bajo costo
- b) Se incorporan dentro del desarrollo del producto sin afectar adversamente la línea de tiempo
- c) Hechos en etapas tempranas y continuamente

Número de participantes, equipos ágiles aplican las pruebas de usabilidad con pocos usuarios. Los expertos en el área señalan que considerando el propósito de las pruebas puede no ser necesario un gran número de participantes. Nielsen en su proceso de Discount Usability (Jakob Nielsen, 1993) indica que con 5 usuarios los resultados de las pruebas comienzan a repetirse por lo que este número de participantes es ideal para evaluar aquellas características de la interfaz que interesan. (Dumas & Redish, 1999) sugieren tres participantes en pruebas de usabilidad como un mínimo absoluto y refieren estudios que sugieren menos de diez participantes para detectar información relevante. Aunque el número de participantes en la prueba puede ser más de lo posible, en equipos ágiles el factor tiempo es fundamental y aplicar las pruebas a muchos participantes requiere invertir más tiempo para probar y analizar los resultados. Hay otras técnicas como pruebas Beta que puede ser utilizadas como soporte de las pruebas de usabilidad para abarcar un número mayor de participantes y aprovechar los resultados de ello.

Participantes, las pruebas de usabilidad abogan realizarlas con el usuario final. Sin embargo en los desarrollos ágiles no siempre se tiene la posibilidad de contar con usuarios reales disponibles para trabajar con ellos. XP considerar un cliente on situ lo cual puede ser una ventaja pero incluso en este caso algunas veces es difícil tener usuarios disponibles para las pruebas. Por esto, con el enfoque ágil las pruebas de usabilidad se aplican con diferentes tipos de participantes que no son necesariamente los usuarios finales. Las pruebas se realizan con usuarios representativos o incluso un grupo de usuarios disponible con el que cuenta el equipo de desarrollo para probar las aplicaciones. Sin embargo, probar con usuarios finales es altamente recomendable incluso en los proyectos ágiles para tener observaciones de aquellos usuarios quienes tienen las necesidades reales.

Laboratorios de usabilidad, varios autores señalan que los laboratorios de usabilidad no son una condición necesaria para aplicar las pruebas de usabilidad. El desarrollo ágil de software toma esta idea y confirma el hecho. En cuanto a espacios especialmente estructurados para las pruebas son usados de manera mínima. Una práctica funcional reportada por algunos equipos fue grabar en audio o video las pruebas de usabilidad y presentar la grabación a todo el equipo obteniendo de esta forma un mejor involucramiento de este con las metas del usuario. Además, el análisis y registro de la retroalimentación del usuario se realizan de una manera más rápida.

Resultados exitosos han sido reportados aplicando de diferentes maneras las pruebas al usuario. La mayormente usada son las pruebas en-casa (in-home) donde el equipo de desarrollo y los participantes se reúnen en un lugar específico para aplicar las pruebas. La aplicación remota es otra manera que ha dado buenos resultados, esta opción toma ventajas de la tecnología para minimizar las distancias en caso de que esta represente un problema al aplicar las pruebas o la naturaleza del software en desarrollo dé lugar a que tengan que realizarse a distancia.

Análisis de las pruebas de usabilidad, las observaciones de los usuarios son clasificadas al momento del reporte de los testers una vez finalizada la prueba para determinar si los problemas deben ser reparados al momento

dentro de la iteración en curso o considerados para las siguientes iteraciones como una mejora a ser agregada dentro del backlog del proyecto. Esto es, las características que pueden ser mejoradas sin mayor problema técnico son implementadas de una manera rápida en la misma iteración aportando al avance del proyecto.

Bloques paralelos con ciclos de diseño por adelantado, aunque el desarrollo ágil defiende la entrega continua de software valioso para el usuario desde la primera iteración, para desarrollar software útil incluso con metodologías ágiles es necesario considerar un primer ciclo para obtener un enfoque general del proyecto. No es necesario obtener el panorama completo pero sí suficiente información para tener una planeación más fuerte. Actividades de diseño llevadas a cabo por adelantado para subsecuentes iteraciones trabajadas en paralelo con tareas de desarrollo definidas previamente permite a los equipos de desarrollo mantener las entregas continuas de software valioso y a la vez también mejora la usabilidad de este.

Cualitativo más que cuantitativo, las pruebas de usabilidad destacan el establecimiento de metas cuantitativas de usabilidad para asegurar un buen nivel de este atributo. Sin embargo, aplicando pruebas de usabilidad se pueden recolectar medidas de rendimiento como datos cuantitativos y medidas subjetivas como datos cuantitativos y cualitativos (Dumas & Redish, 1999). Los equipos ágiles consideran principalmente los comentarios de los usuarios y expertos como datos clave para analizar y determinar cómo proceder con futuras iteraciones. Un análisis completo de las pruebas de usabilidad puede consumir tiempo. En una ambiente ágil es suficiente reportar la situación, discutirla en relación a las siguientes historias de usuario y planear el curso de acción (Illmensee & Muff, 2009).

Además de los puntos anteriores una situación que se observa en la aplicación de las pruebas de usabilidad y en general al incorporar técnicas HCI en un entorno de desarrollo ágil, es el hecho de que pueden ser aplicadas por elementos del equipo que no necesariamente son expertos HCI. Otro punto que vale la pena mencionar es el reiterativo uso de los equipos ágiles de prototipos de papel o interactivos como apoyo en las pruebas de usabilidad combinando también con otras herramientas como “thinking aloud”.

5.2 Patrón “Agile Usability Testing”

Una vez analizada la manera en la que los equipos ágiles han utilizado las pruebas de usabilidad a partir de nuestra revisión de literatura, hemos obtenido la información suficiente que nos permita identificar un patrón de buenas prácticas para sobrellevar los problemas que conlleva la aplicación de esta técnica en un contexto de desarrollo ágil. A continuación describimos el patrón “Agile Usability Testing” propuesto como apoyo a los equipos ágiles en la aplicación de las pruebas de usabilidad como técnica HCI que permite evaluar a usabilidad de las aplicaciones desarrolladas.

5.2.1 Definiendo el patrón “Agile Usability Testing”

El contenido del patrón se definió considerando las adecuaciones aplicadas a la técnica, sugerencias, descubrimientos y lecciones aprendidas que se reportan en los diversos estudios analizados en la revisión de literatura realizada. Para plasmar la estructura e información de este patrón de nuevo utilizamos el formato del patrón de producto propuesto como parte del modelo arquitectónico propuesto en (Sanchez-Segura et al., 2010) y que utilizamos para el patrón “Agile persona” descrito en el capítulo anterior.

La estructura y contenido del patrón “Agile usability” que se presenta en esta sección es resultado del proceso de evaluación aplicado a los patrones que se detalla en el capítulo 6 de este trabajo. Una vez aplicadas las mejoras derivadas de las evaluaciones resulta la estructura final que se muestra en la figura C5-1 y es la misma para el patrón descrito en el capítulo anterior.

Name	Nombre del patrón, puede ser una palabra o frase corta	Related pattern	Nombre del (los) patrón(es) con el (los) que tiene relación
Initial context	Descripción de las precondiciones sobre las cuáles el patrón es aplicable		
Result Context	Contexto resultado al aplicar el patrón		
Problem	Descripción del problema en el que contribuye la solución del patrón		
Forces	Restricciones a considerar para la aplicación del patrón		
Solution	Solución propuesta por el patrón para el problema indicado. Se utilizan diagramas para representar el proceso de solución		
Roles	Personas o participantes en la aplicación del patrón		
Basic Knowledge	Conocimiento básico que se requiere para aplicar el patrón		
Entries	Información de entrada o productos previos para aplicar el patrón		
Exit	Producto obtenido como resultado de aplicar el patrón		
Lessons learned	Experiencias que se han obtenido al aplicar el patrón en proyectos previos		
Examples	Ejemplos de la aplicación del patrón		
Templates	Plantillas a utilizar durante la aplicación del patrón para obtener el resultado		
Information Resources	Recursos de información tales como referencias y documentación que pueden utilizarse para desarrollar este patrón		

Figura C5-1 Estructura final del patrón "Agile Usability Testing"

En la tabla C5-3 se presenta el formato del patrón “Agile Usability Testing”. Para una descripción más clara de la información contenida en el patrón dividimos la plantilla en cinco partes, el formato completo podemos visualizarlo en la sección final de este capítulo.

Tabla C5-3 Patrón "Agile Usability Testing" (Parte 1 de 5)

Name	Agile Usability Testing	Related pattern	None
Initial context	This pattern can be used in agile projects to validate how easy to use is the user interface design. This validation can be applied at the beginning of the process development and/or into the iterations. Low-fi prototypes, high-fi prototypes or even working software are tested with the users in order to identify interaction design features to improve. Tests are applied with real users or representative users from them.		
Result Context	Observations about user interface design and functionality are gathered from the user's feedback through tests applied directly to them. The team analyzes, prioritizes and decides which features can be fixed at the current iteration. The remaining features that require more time or analysis to be improved are moved to the backlog for future iterations.		
Problem	The agile development team must be able to identify and implement as soon as possible those features that assure delivering useful software.		
Forces	<p>Kind of the organization This pattern can be used in agile projects regardless of company size.</p> <p>Kind of system to be developed This pattern can be used in agile projects that use development methodologies following the agile philosophy.</p> <p>Kind of client The user must be available and always involved through the process of the persona's definition.</p> <p>Other</p> <ul style="list-style-type: none"> • Meetings for interviews, pattern's review or feedback between development team and user must be established in a friendly and trusting environment • Users, UI designers and developers must be involved in the presentation, reviews and feedbacks meetings • Pattern Agile Usability Testing must be kept available to the team through the whole development process at any moment 		

Nuestro patrón tiene por *nombre* “Agile Usability Testing” para representar el nombre de la técnica HCI a aplicar en un contexto ágil. En el campo de *patrón relacionado* se especifica “None” ya que por el momento el patrón Agile Usability Testing no tiene relación con otros patrones sin embargo una vez que forme parte de un lenguaje de patrones se irá actualizando este dato en su descripción. Definimos nuestro *contexto inicial* y *contexto resultado* enfocados en la aplicación de pruebas de usabilidad tipo formativas, de mejora y de validación como un artefacto para validar el diseño y las decisiones sobre los requisitos con los usuarios. El identificar de manera temprana características en el diseño de interfaz que requieran mejora y aplicar las adecuaciones necesarias al momento evitando re trabajos posteriores, representa un *problema* para el equipo ágil debido a las limitaciones del tiempo establecido en la iteración. El objetivo del patrón es contribuir a minimizar o evitar dicho problema. Como *restricciones (Forces)* para este patrón se considera que puede ser utilizado independientemente del tamaño de la compañía en proyectos que utilizan cualquier metodología de desarrollo apegada a la filosofía ágil. Por otra parte, la participación del usuario y la disponibilidad de la descripción del patrón “Agile Usability Testing” a través de todo el proceso de desarrollo son cruciales.

El siguiente campo que se presenta en el patrón es la *solución* propuesta. La Tabla C5-3 (Parte 2 de 5) muestra dicha solución conformada por diferentes elementos. Primero se presentan y describen las actividades que se realizan al aplicar las pruebas de usabilidad, esto con el objetivo de que al hacer referencia al proceso “Usability testing” en los diagramas posteriores de la solución se tenga en mente las actividades involucradas con él. Posteriormente, se presentan dos diagramas en donde a partir del primero de ellos se visualiza el panorama general de los momentos en los que se propone aplicar las pruebas de usabilidad esto es, al inicio del desarrollo y durante las iteraciones. En el segundo diagrama se observa el proceso para la aplicación de las pruebas dentro de una iteración. Por último se van describiendo cada uno de los posibles enfoques para incorporar la técnica de pruebas de usabilidad en el proceso de desarrollo ágil

Tabla C5-3 Patrón "Agile Usability Testing" (Parte 2 de 5)

Solution
<p>The process of usability testing from the pattern (figure 1) involves three activities: planning the testing, applying the tests, report and analyzing the users' observations.</p> <ol style="list-style-type: none"> 1. Planning the tests, important features must be defined before applying usability testing in order to get valuable information from the users interaction: <ol style="list-style-type: none"> i) Firstly the objective of the tests: knowing the goal of applying the tests assures focus on tasks definition on the user's behavior target. ii) Define the tasks to be applied on the usability testing: it must be focused on gathering information related to the testing's goal. iii) The place to apply the test: it must be defined according to the capacity, availability, advancement of development and tools. Usability testing could be applied in-home, remotely or on-site. iv) Define test participants: it is highly recommended to apply the tests with end users of the software; however it is well known that it is not always possible. Nevertheless usability testing must be applied with end users or representative users. Developers or team elements playing the user role are not recommended. Finally, the participants are recruited and the usability testing applied. 2. Applying usability tests, the usability testing must be recorded in some way, either video recording, sound recording or capture screen windows (by some software tool). Developers and those stakeholders involved in any way with the software development must be present on the usability testing. During the usability session the observations are registered in a list. 3. Analyzing usability feedback by the whole team, those features which do not represent major changes and can be corrected before the end of the iteration are entered into the tracking system and implemented. New features requested or those too large to be implemented are considered an enhancement required and classified in the review list as urgent, important or desirable to have. These features are considered to be entered into the project backlog for subsequent iterations. <p>Once the usability testing review has ended and all the features have been analyzed, then the shippable product of the iteration is delivered.</p> <div data-bbox="587 1697 1077 2016"> <pre> graph TD A[Planning usability Testing] --> B[Apply usability tests] B --> C[Usability Feedback] C --> D[Analyze usability Feedback collected] </pre> <p>Planning usability Testing</p> <ul style="list-style-type: none"> Define the objective of the usability test Define space to apply tests (in-home, remotely, on-site) Identify users to test (end users, representative users) Recruit participants (3-5) Define tasks or functionalities to test Define post-test questionnaire (if it is necessary and there is time) <p>Apply usability tests</p> <ul style="list-style-type: none"> Tests can be applied at low-fi prototype, high-fi prototype or working code Recording Usability test (audio/video/screen capture) Developers must be on tests Apply Thinking aloud Create usability feedback list <p>Usability Feedback</p> <p>Analyze usability Feedback collected</p> <p>The whole team must be present. Those features which are not too large to be fixed are identified as "Bugs" and are fixed at moment of the feedback analysis. New features or too large to be fixed are identified as "Enhancements" and considered to next iterations.</p> </div>

Figure 1 How “Usability Testing” is applied in the solution pattern

Figure 2 shows when usability testing is used through an agile software development process. The process named “development iteration” has a symbol “μ” to identify the relation to the iteration process depicted in figure 3 which shows when usability testing is applied. In the diagrams the process identified as “Usability Testing” involves the activities described previously and depicted in figure 1.

Figure 2 When pattern “Usability Testing” is applied in agile process

Figure 3 When product pattern “Usability Testing” is applied in agile iteration

At the beginning of the software process the development team gets a scope of the project gathering enough information from users. Depending on time and resources, this scope could be a whole interaction design or just a general scope defining enough interaction design with subsequent enhancements during iterations. Thus, we have a few options:

- If agile team gets information to make a whole interaction design before any other development tasks then formative usability testing is applied. Initial features can be designed by creating low-fi prototypes (like paper prototypes) to be evaluated with usability testing and then from the feedback of this review define the product backlog. Once the whole scope is designed and product backlog defined then the normal agile process continues defining iteration backlog with the user stories to develop. After the iteration development, the usability feedback gathered is applied to the product backlog update if there is another iteration to be developed. We call this approach “Interaction Design First”. Figure 4 shows the way applying usability testing marked with bold arrows through the development process.

Figure 4 Applying usability testing during iteration with approach “Interaction Design First”

- b) If at the beginning of the development process the agile team gets sufficient information to make a general interaction design and decides to apply formative usability, then initial features can be designed by creating low-fi prototypes (like paper prototypes) to be evaluated with usability testing and subsequently from the feedback of this review define the initial product backlog. Once the general scope is designed and initial product backlog defined the normal agile process continues defining iteration backlog with the user stories to develop. Through the iteration development interaction design is refined by gathering data, creating low-fi prototypes and applying usability testing to them. Usability feedback is used to define which features are fixed during the actual iteration or considered in next iterations. When iteration development is finished, the usability feedback gathered is used to update the product backlog. We call this approach as “Minimal Up-Front Design”. Figure 5 shows the way applying usability testing marked with bold arrows through the development process.

Figure 5 Applying usability testing during iteration with approach “Minimal Up-Front Design”

- c) Agile team gets just enough information at the beginning of the process and decides not to define interaction design. The product backlog is defined without previous formative usability testing. Once initial product backlog is defined then the normal agile process continues defining iteration backlog with the user stories to develop. Through the iteration development interaction design is defined for the actual user stories and enhanced in subsequent tasks or even iterations. We call this approach as “Implicit interaction design”. Figure 6 shows the way applying usability testing marked with bold arrows through the development process.

Figure 6 Applying usability testing during iteration with approach “Implicit Interaction Design”

- d) If the development team has enough elements to work user interface design and implementation activities at the same time on parallel tracks, then get data from the user by developing research activities previous product backlog definition working this activities in a first iteration focused just to get enough information about the user and the software project (it could be named “cycle 0”). Once this initial iteration is complete initial backlog can be defined and start with the iterations where interaction design is implemented, tested and enhanced. Through iterations development team work with design track and implement track; at the design track the interface design is defined ahead for subsequently iterations, at the implement track use the feedback from previous iterations to develop the interface. We call this approach “Parallel Track”. Figure 7 depicts both design track (bold arrows) and implement track (dotted lines).

Figure 7 Applying usability testing during iteration with approach “Parallel Track”

At the end of the iteration if there is a release programmed, then usability testing is shown as optional but this test is strongly recommended and again at the end of entire project.

A continuación se describe con más detalle la solución propuesta. La figura C5-2 muestra los procesos que, de acuerdo a nuestra solución, involucra aplicar las pruebas de usabilidad en un ambiente ágil esto es, cómo se aplica la técnica en dicho entorno.

Figura C5-2 Cómo es aplicada la técnica de pruebas de usabilidad en un proceso ágil

La planeación de las pruebas de usabilidad involucra la parte más fuerte de actividades debido a la importancia de éstas para preparar las pruebas con el objetivo de obtener la información esperada. Las actividades involucradas son:

- Definir el objetivo de las pruebas lo cual representa un factor clave para mantener en mente la meta perseguida al aplicar la prueba.
- Definir tareas o funcionalidades a probar durante la sesión de prueba
- Decidir dónde aplicar la prueba (no necesariamente en un laboratorio de usabilidad)
- Identificar a los usuarios participantes (preferiblemente usuarios finales o representativos de estos)
- Reclutar participantes
- Definir cuestionario post-prueba (Si fuera necesario y se contara con tiempo suficiente para ello)

Una vez que el plan de las pruebas es definido entonces se procede a **aplicar las pruebas de usabilidad**. De acuerdo al avance del proyecto y objetivo de la prueba el equipo decide si se aplica utilizando prototipos de bajo nivel como prototipos en papel o prototipos de alto nivel como los funcionales. Una condición importante es que los desarrolladores deben estar presentes en el test. Los usuarios son alentados para expresar en voz alta las situaciones que experimentan con el software y como las van trabajando durante la prueba que es grabada de alguna forma de acuerdo a la disponibilidad de las herramientas para esto y la comodidad de los usuarios. Conforme se va desarrollando la prueba se va generando la lista de retroalimentación, los observadores de la prueba identifican y registran en la lista posibles mejoras derivadas de las características señaladas por el usuario al interactuar con el software desarrollando las tareas definidas para la prueba. Después de las pruebas la información obtenida y registrada de los usuarios se presenta a todo el equipo con el objetivo de evaluar las observaciones y **analizar la retroalimentación de usabilidad**. Durante el análisis de la retroalimentación obtenida, las características analizadas y consideradas como mejora se clasifican en la lista como “bugs” (errores) para identificar aquellas que pueden repararse durante el sprint que se está ejecutando o “enhancements” (mejoras) a considerar en el desarrollo de sprints posteriores. Esta lista de características puede ayudar al momento de la planificación de la siguiente iteración para considerar aquellas que deben ser integradas como historias de usuario dentro del product backlog del proyecto. Esta lista debe estar disponible para el equipo de desarrollo en cualquier momento como una evidencia del reporte de las pruebas de usabilidad.

En la figura C5-3 observamos dos diagramas en donde el proceso llamado “Usability Testing” en fondo gris indica el momento de aplicar las pruebas de usabilidad. El diagrama de la izquierda representa el proceso general de desarrollo ágil. El diagrama de la derecha muestra a detalle las actividades que corresponden al proceso “Development iteration”, identificado en el diagrama del proceso general con el símbolo “μ”, donde también se indica aplicar pruebas de usabilidad. Cabe recordar que el proceso “Usability Testing” abarca todas las actividades de la figura C5-1 descrita previamente.

Observamos que el proceso “Usability testing” se indica en diferentes momentos durante el proceso de desarrollo ágil. Se establece como opcional al principio del desarrollo ágil, obligatorio durante las iteraciones, opcional al finalizar las iteraciones y opcional después del último release. Es importante señalar que en aquellos momentos donde se indica como opcional aplicar las pruebas de usabilidad sí es altamente recomendable hacerlas. En particular durante el desarrollo de la iteración dependiendo del tipo de bloque que se trabaje (diseño o desarrollo) se indica aplicar las pruebas de usabilidad en dos momentos; este diagrama se describe a detalle más adelante.

Figura C5-3 Cuándo es aplicado el patrón "Agile Usability Testing"

Profundizando en el proceso solución sobre cuándo aplicar las pruebas de usabilidad en un desarrollo ágil que se presenta en la figura C5-3 observamos que al inicio del proceso el equipo **obtiene información** de los usuarios realizando una mínima investigación por medio de observaciones y entrevistas; esto proporciona al equipo un enfoque general que le permita comenzar con el desarrollo del software. Los datos obtenidos proporcionan solamente información representativa pero importante que se irá complementando y refinando a través de las iteraciones subsecuentes. Ya con un enfoque inicial, el equipo decide si requiere aplicar una primera etapa de **pruebas de usabilidad formativa**, creando prototipos de bajo nivel, para utilizar el resultado de retroalimentación de ésta en la **definición del product backlog** inicial. El equipo también puede decidir definir el product backlog sin necesidad de los resultados de las pruebas de usabilidad de tipo formativas. Si el equipo decide aplicar una primera etapa de pruebas de usabilidad formativa antes de la definición del product backlog inicial puede dedicar tiempo a estas actividades trabajando un ciclo 0 (Sy, 2007).

Durante la **planificación de la iteración** además de la priorización establecida de las historias de usuario la retroalimentación de usabilidad de iteraciones previas (en caso de no ser la primera) puede proporcionar información útil para determinar las historias de usuario a desarrollar y para ir actualizando el product backlog.

Es necesario aplicar pruebas de usabilidad en cada iteración por lo que una vez definido el iteration backlog el equipo ágil decide el tipo de pruebas de usabilidad que aplicará durante la iteración que se está planificando. Esta decisión depende del avance del proyecto, el tiempo de entrega establecido y los usuarios finales o representantes que evaluarán el producto.

Una vez completa la planificación de la iteración el desarrollo de esta inicia con el equipo de desarrollo **verificando la línea de proceso a aplicar** decidiendo realizar o no el diseño de interfaz (UI Design) inicialmente. Cada una de las opciones de la decisión representa bloques de actividades. La decisión de no definir un diseño de interfaz conlleva la implementación de historias de usuario para generar una aplicación funcional. Si el equipo cuenta con elementos del equipo enfocados en particular a cada uno de los bloques mencionados la solución puede trabajar ambos con actividades en paralelo (parallel track). Además, en caso de desarrollarse en paralelo el diseño puede definirse con iteraciones por adelantado a la implementación. De esta manera los equipos ágiles pueden visualizar más allá de la primera iteración y definir un esbozo sobre futuros iteration backlog.

Siguiendo la línea del bloque de diseño de la interacción en la figura C5-4 tenemos que durante el desarrollo de la iteración después de diseñar prototipos de bajo nivel el aplicar pruebas de usabilidad es opcional. Sin embargo, al finalizar con la última historia de usuario es necesario aplicar pruebas de usabilidad. Similarmente siguiendo la línea del bloque de desarrollo las pruebas de usabilidad se aplican necesariamente después de implementar el desarrollo de la última historia de usuario. La retroalimentación de usabilidad obtenida de cualquiera de los dos bloques se utiliza para la planificación de la siguiente iteración.

Figura C5-4 Momentos en los que puede aplicarse pruebas de usabilidad durante la iteración

Una vez que se lleva al cabo un release una nueva prueba de usabilidad puede ser aplicada y es usualmente más fácil y rápida que las pruebas realizadas durante las iteraciones. El reporte de las pruebas de usabilidad aplicadas al producto es parte de la retroalimentación de la iteración e influye en la redefinición del product backlog del proyecto. **Al final del proceso de desarrollo** es altamente recomendable aplicar otra prueba de usabilidad para evaluar que tan útil es el producto de software completo. En nuestra solución las pruebas de usabilidad en ese momento son representadas como opcionales.

En relación a los elementos del equipo que aplican las pruebas de usabilidad como mencionamos previamente, no necesariamente son expertos del área de diseño, incluso los mismos miembros del equipo de desarrollo pueden ser capacitados para adoptar las técnicas HCI y trabajar con roles enfocados al diseño a pesar de no ser

expertos (Dayton et al., 2009). Aunque no es mandatorio tener expertos HCI en el equipo de desarrollo sí es altamente recomendable.

Hemos mencionado que se identificaron cuatro enfoques en los que son incorporadas las técnicas de usabilidad en el proceso ágil. En relación a estos diferentes enfoques a continuación podemos observar en las figuras C5-5 a la C5-8 los diagramas donde se resalta con flechas en negritas el orden de los procesos para la integración de las pruebas de usabilidad en cada uno de ellos siguiendo nuestro patrón. Independientemente del enfoque el patrón establece dos actividades en común:

- Al inicio del desarrollo, la obtención de datos es importante para identificar con más exactitud las funcionalidades prioritarias para el usuario a partir de la información obtenida.
- Cada vez que se produce una liberación (release) y al final del proceso completo de desarrollo la aplicación de pruebas de usabilidad es opcional pero altamente recomendable.

La figura C5-5 muestra la integración utilizando un enfoque Interaction Design First mientras que la figura C5-6 muestra la integración de la técnica con el enfoque Minimal up-front design. Las figuras C5-7 y C5-8 muestran la aplicación de las pruebas de usabilidad en los enfoques Implicit Interaction Design y Parallel Track respectivamente.

Figura C5-5 Aplicando pruebas de usabilidad con el enfoque “Interaction Design First”

Siguiendo la estructura de nuestro patrón, aplicar pruebas de usabilidad con los enfoques Interaction Design First (Figura C5-5) o Minimal Up-front design (Figura C5-6) comienza con una obtención de datos exhaustiva o mínima respectivamente. En estos casos se utilizan pruebas de usabilidad formativas desarrollando prototipos de baja fidelidad (low-fi). La retroalimentación obtenida a partir de la aplicación de las pruebas se utiliza para definir el product backlog. En el caso del enfoque Interaction Design First (Figura C5-5) una vez aprobado el diseño de la interfaz y definido el product backlog éste es implementado en su totalidad por los desarrolladores a través de las iteraciones; al final de cada iteración se pueden aplicar pruebas de usabilidad para evaluación o validación.

Incorporar las pruebas de usabilidad con el enfoque de Minimal Up-front design (Figura C5-6) es un proceso parecido al enfoque Interaction Design First. La diferencia radica en que durante la iteración también se obtiene información para complementar la que se obtiene al principio por medio de una investigación mínima sobre el usuario. Posteriormente se crean los diseños de bajo nivel para utilizarlos en la prueba de usabilidad, una vez aplicada la(s) prueba(s) se puede continuar con la implementación de las historias de usuario tomando siempre en cuenta la retroalimentación obtenida previamente de las pruebas con el usuario.

Al finalizar la implementación del código de todas las historias de usuario consideradas en la iteración se aplica nuevamente la prueba de usabilidad. La retroalimentación de usabilidad es considerada para la revisión del product backlog y consecuentemente para definir el backlog de la siguiente iteración. Subsecuentemente el equipo de desarrollo decide aplicar o no las pruebas de usabilidad en cada liberación o en la liberación final.

Figura C5-6 Aplicando pruebas de usabilidad con el enfoque "Minimal Up Front Design"

Incorporando las pruebas de usabilidad a través del enfoque Implicit Interaction Design (Figura C5-7) no se aplican pruebas de usabilidad formativa para conocer el enfoque general del proyecto al inicio del proceso, la información necesaria relacionada con el usuario se obtiene durante las iteraciones.

En el desarrollo de la iteración se obtiene información del usuario y se aplican pruebas de usabilidad formativas creando prototipos de bajo nivel y probando estos con los usuarios.

Una vez aplicadas las pruebas se pueden ya desarrollar las historias de usuario esto es, implementar los códigos para generar las aplicaciones funcionales tomando en cuenta para ello los resultados de la retroalimentación obtenida.

Figura C5-7 Aplicando pruebas de usabilidad con el enfoque “Implicit Interaction Design”

Cuando las pruebas de usabilidad se incorporan en el desarrollo ágil utilizando el enfoque Parallel Track (Figura C5-8) las iteraciones se desarrollan involucrando dos bloques: el bloque de diseño de interfaz y el bloque de desarrollo.

Figura C5-8 Aplicando pruebas de usabilidad con el enfoque “Parallel Track”

Los datos obtenidos al inicio son esenciales para definir el diseño de interfaz inicial. Las pruebas de usabilidad durante el desarrollo de la iteración son aplicadas de acuerdo al bloque que se está trabajando, puede ser un bloque de diseño de la interacción, bloque de implementación o incluso los dos al mismo tiempo desarrollados por diferentes miembros de un mismo equipo ágil. En el bloque de diseño donde la interfaz de usuario es diseñada, las pruebas de usabilidad formativas se aplican derivadas de la información obtenida al inicio del bloque o en iteraciones previas. La retroalimentación obtenida es considerada para la siguiente iteración. Por otro lado, en el bloque de implementación se puede considerar pruebas de usabilidad de evaluación o validación. Las características a ser desarrolladas son derivadas de las historias de usuario definidas de iteraciones previas. En la figura C5-8 dentro de la iteración se representa el bloque de diseño con líneas continuas mientras que el bloque de desarrollo se muestra con líneas punteadas.

Continuando con la descripción del patrón después del campo solución descrito a detalle previamente tenemos la información que corresponde a los *roles* involucrados en el proceso de aplicación de las pruebas de usabilidad (Tabla C5-3 parte 3 de 5). Durante las pruebas de usabilidad además de los usuarios participan como observadores los desarrolladores y el scrum master; también pueden participar el experto en diseño (en caso de contar con este elemento), el product owner y aquellos stakeholders cuya participación se considere necesaria como observadores.

Para aplicar este patrón los elementos del equipo requieren tener *conocimientos básicos* sobre las pruebas de software y el enfoque general del proyecto de software en construcción. Las *habilidades* necesarias son la capacidad de aplicar investigación de usuario para obtener datos relevantes por medio de observación directa, capacidad de análisis para identificar necesidades del usuario estableciendo requerimientos del diseño de interacción y por último pero no menos importante la capacidad de interactuar con los usuarios de una manera amigable. La entrada que recibe el patrón es la lista de retroalimentación de usabilidad obtenida a partir de la aplicación de las pruebas y la salida es la lista revisada y clasificada de acuerdo al tipo de mejora identificada en la prueba (bug o característica a considerar en las iteraciones posteriores). En el campo *lecciones aprendidas* se indican las seis que el patrón tiene por objetivo.

Tabla C5-3 Patrón "Agile Usability Testing" (Parte 3 de 5)

Roles	<ul style="list-style-type: none"> Development Team (Agile Developers) Scrum Master Product owner (optional) Users (stakeholders, end-users, representative users) UI Designer, HCI expert, UX expert (optional)
Basic Knowledge	<p>Knowledge</p> <ul style="list-style-type: none"> Knowing basic concepts about software testing General scope of the software project under construction <p>Skills</p> <ul style="list-style-type: none"> Capability to apply user research to get data through direct observation. Capability to data Analysis identifying needs and establishing interaction requirements Capability to interact with the users in a friendly way.
Entries	Usability testing report
Exit	Reviewed usability Testing List per iteration
Lessons learned	<ul style="list-style-type: none"> Usability testing should be applied in a relaxed and friendly atmosphere aiming to get sincerely comments from the users. After the tests have been applied the results must be presented to the whole development team in a meeting to review the feedback gathered. The pattern "Agile Usability Testing" must be kept available for the team through the whole development process.

Continuando con la descripción del patrón tenemos los campos *Examples (Ejemplos)* y *Templates (plantillas)*. El *ejemplo* presenta una lista de revisión de una prueba de usabilidad con la información contenida en todos los campos de la plantilla (figure 8). Como podemos observar la primera parte de la lista contiene información sobre datos generales de la prueba de usabilidad: identificador el proyecto, identificador de la iteración a la que corresponde la prueba, identificador de la prueba de usabilidad, fecha y el objetivo de la prueba. La siguiente parte de la plantilla corresponde a la lista de las características observadas durante la prueba. A cada una de las

observaciones se asigna un identificador que la plantilla sugiere como numérico, se describe lo observado y la clasificación asignada posteriormente durante el análisis de esta lista. La clasificación que puede ser “B” para identificar los bugs o errores a reparar durante la iteración en curso o “FR” para aquellas características a considerar para futuras redefiniciones del product backlog en iteraciones posteriores.

Tabla C5-3 Patrón "Agile Usability Testing" (Parte 4 de 5)

Examples

Usability Testing Review List					
Project ID	TFSysstem				
ID Iteration:	03	ID Usability Testing:	UTR 0302	Date Testing:	20/Jan/2015
Objective of the test:					
This test aim to validate efficiency and functionality of the interfaces design of "Taxes register" and "Sales record" derived from the user story US10.					
ID	Observation			Classification	
01	Botton "cancel" hasnot the first letter uppercase			B	
02	There is no option to apply taxes by city			FR	
03	Date field allow to capture month 13			B	
04	There is no confirmation window when user save the information			B	
05	Principal window close when option "search by name" is selected			B	
06	The list of products doesn't show the subtotal and total including IVA			FR	
07					
08					
09					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					

*B = Bug or feature to fixed on the iteration in course
FR = Feature required to be considered on the project backlog

Figure 8 Usability Testing Review List

Templates

<https://drive.google.com/file/d/0B1YqXzQvZlQuYXljZTdvaJFKOTA/view?usp=sharing>

En el campo de *plantillas* se proporciona un link al espacio on-line en el que puede obtenerse el formato definido para llevar el registro de la lista de retroalimentación a utilizar en las pruebas de usabilidad. La figura C5-9 muestra dicha plantilla con los distintos campos.

Usability Testing Review List					
Project ID	(Project Identifier)				
ID Iteration:	(Identifier corresponding to the iteration when test is applied)	ID Usability Testing:	(Identifier assigned to the usability test)	Date Testing:	(Testing date)
Objective of the test:					
Objective applying the usability test					
ID	Observation			Classification	
01	(Observation's description)			(Feature Classification: "B" for bugs to be fixed during the actual iteration and "FR" to features to be considered onto next iterations)	
02					
03					

Figura C5-9 Plantilla de la lista de revisión para pruebas de usabilidad

Por último tenemos el dato de las *fuentes de información* que el equipo puede consultar para obtener mayor información sobre las pruebas de usabilidad y datos a fin que les ayuden en la aplicación de esta técnica.

Tabla C5-3 Descripción del patrón "Agile Usability Testing" (Parte 5 de 5)

Information Resources	<p>Dumas, J. S., & Redish, J. C. (1999). A practical guide to Usability Testing. (I. Books, Ed.). Intellect Ltd.</p> <p>Sharp, H., Rogers, Y., Preece J. (2007) , Interaction Design: beyond human-computer interaction. Ed John Wiley & Sons.</p> <p>Sanchez-Segura M., Medina-Dominguez F., Amescua A., Mora-Soto A. Improving the efficiency of use of software engineering practices using product patterns. Recovered May 2014 from http://www.sciencedirect.com</p> <p>Nielsen, J. (2008). Agile Development Projects and Usability. Jakob Nielsen's Alertbox. Retrieved from https://www.nngroup.com/articles/agile-development-and-usability.</p> <p>Barnum, C. M. (2010). Usability Testing Essentials. Morgan Kaufmann.</p>
-----------------------	---

5.2.2 Adaptaciones consideradas en el patrón “Agile Usability Testing”

Para la definición del patrón propuesto se consideraron las principales adaptaciones que se han aplicado a las pruebas de usabilidad para incorporarse en el entorno de desarrollo ágil. Estas fueron identificadas a través de la revisión de literatura y descritas en la sección 5.1.3. Además de estas adaptaciones se tomó en cuenta información derivada de los resultados obtenidos en estudios previos analizados también en la revisión de literatura. Resultados que aunque no reflejan una adaptación a la técnica para su incorporación en el proceso de desarrollo sí proporcionan información relevante que permite visualizar mecanismos, procesos y prácticas comunes que sirven de apoyo al integrar las pruebas de usabilidad en el contexto ágil. “Thinking aloud” y los diferentes tipos de prototipos forman parte de estos métodos y artefactos considerados en el patrón “Agile Usability Testing”.

Comenzando con el *enfoque al momento de aplicar las pruebas*. La solución propuesta considera la incorporación de las pruebas de usabilidad en un proyecto ágil como el desarrollo de pequeños estudios en los que al inicio, durante su desarrollo y al finalizar es posible aplicar pruebas de usabilidad sin que afecten el límite de tiempo para el proceso. Aunque en ciertos momentos el aplicar las pruebas se considera como opcional se hace hincapié en que es altamente recomendable realizarlas. En cuanto al *número de participantes* el patrón indica contar con tres o cinco usuarios para participar en cada prueba tomando en cuenta que en un entorno ágil se tiene como limitante principal el tiempo. Tener un mayor número de participantes en la prueba repercute en una mayor inversión de tiempo para aplicarla y analizar los resultados. Se recomienda reclutar como *participantes* de la prueba a usuarios reales pero abriendo la posibilidad de aplicar la prueba incluso con usuarios representativos de los reales.

El *sitio para aplicar la prueba* es una decisión que toma el equipo en conjunto para determinar de acuerdo a las condiciones actuales del proyecto al momento de aplicar la prueba cuál es el lugar más indicado para llevarlas al cabo. El patrón considera tres opciones: in-home, in-situ o a distancia de manera remota.

La plantilla proporcionada en el patrón para el registro de las observaciones obtenidas durante la prueba de usabilidad llamada “Usability Testing Review List” permite un *análisis más sencillo y rápido* de la retroalimentación obtenida clasificando los resultados de acuerdo a su prioridad y complejidad técnica. De esta manera durante la revisión de la lista de observaciones resulta para el equipo ágil más fácil determinar cuáles pueden ser corregidas e implementadas durante la iteración en curso y cuáles se anexan al backlog para considerarse en iteraciones posteriores.

Aunque el uso de bloques paralelos para diseño de interacción y desarrollo es el que utilizan principalmente los equipos ágiles en su desarrollo, como vimos previamente de acuerdo a nuestra revisión de literatura se han obtenido resultados exitosos de la incorporación de las técnicas HCI incluso realizándolo con otros enfoques. El patrón ofrece una solución que abarca la descripción de estos diferentes enfoques de los momentos en los que las técnicas HCI pueden incorporarse dentro de un desarrollo ágil. De acuerdo a la línea que sigue su desarrollo los equipos ágiles pueden determinar cuál es el que mejor se adapta a su proceso como apoyo para usar esta técnica. Es sabido que un patrón no proporciona la solución final pero sí aporta valiosa información para que las buenas prácticas documentadas sirvan como apoyo en la consecución de los resultados esperados.

En un desarrollo ágil el objetivo establecido para las pruebas de usabilidad considera *principalmente metas cualitativas*. No está limitado a la evaluación cualitativa exclusivamente sin embargo los datos cuantitativos se evalúan cuando el tiempo y presupuesto lo permite tal como otras cuestiones que los proyectos ágiles consideran debido al factor tiempo que es una principal limitante. La plantilla para el registro de las observaciones obtenidas en la prueba considera dos clasificaciones para estas: Bugs y nuevas características requeridas. “B” para aquellas características que se consideran “Bugs” o errores que pueden ser resueltos sin mayor complejidad técnica o de análisis por el equipo. Las nuevas características requeridas se identifican con “FR” para aquellas observaciones que implican incorporar nuevas historias de usuario al product backlog para realizarse en iteraciones posteriores. En la sección 5.1.1 se presenta una tabla comparativa entre las características que diferentes autores señalan en relación a las pruebas de usabilidad. La información contenida en el patrón cubre aquellas que resultan comunes entre los enfoques de dichos autores.

5.2.3 Estructura del Patrón “Agile Usability Testing”

Name	Agile Usability Testing	Collaboration pattern	None
Initial context	This pattern can be used in agile projects to validate how easy to use is the user interface design. This validation can be applied at the beginning of the process development and/or into the iterations. Low-fi prototypes, high-fi prototypes or even working software are tested with the users in order to identify interaction design features to improve. Tests are applied with real users or representative users from them.		
Result Context	Observations about user interface design and functionality are gathered from the user's feedback through tests applied directly to them. The team analyzes, prioritizes and decides which features can be fixed at the current iteration. The remaining features that require more time or analysis to be improved are moved to the backlog for future iterations.		
Problem	The agile development team must be able to identify and implement as soon as possible those features that assure delivering useful software.		
Forces	<p>Kind of the organization This pattern can be used in agile projects regardless of company size.</p> <p>Kind of system to be developed This pattern can be used in agile projects that use development methodologies following the agile philosophy.</p> <p>Kind of client The user must be available and always involved through the process of the persona's definition.</p> <p>Other</p> <ul style="list-style-type: none"> • Meetings for interviews, pattern's review or feedback between development team and user must be established in a friendly and trusting environment • Users, UI designers and developers must be involved in the presentation, reviews and feedbacks meetings • Pattern “Agile Usability Testing” must be kept available to the team through the whole development process at any moment 		
Solution	<p>The process of usability testing from the pattern (figure 1) involves three activities: planning the testing, applying the tests, report and analyzing the users' observations.</p> <ol style="list-style-type: none"> 1. Planning the tests, important features must be defined before applying usability testing in order to get valuable information from the users interaction: <ul style="list-style-type: none"> ▪ Firstly the objective of the tests: knowing the goal of applying the tests assures focus on tasks definition on the user's behavior target. ▪ Define the tasks to be applied on the usability testing: it must be focused on gathering information related to the testing's goal. ▪ The place to apply the test: it must be defined according to the capacity, availability, advancement of development and tools. Usability testing could be applied in-home, remotely or on-site. ▪ Define test participants: it is highly recommended to apply the tests with end users of the software; however it is well known that it is not always possible. Nevertheless usability testing must be applied with end users or representative users. Developers or team elements playing the user role are not recommended. Finally, the participants are recruited and the usability testing applied. 2. Applying usability tests, the usability testing must be recorded in some way, either video recording, sound recording or capture screen windows (by some software tool). Developers and those stakeholders involved in any way with the software development must be present on the usability testing. During the usability session the observations are registered in a list. 3. Analyzing usability feedback by the whole team, those features which do not represent major changes and can be corrected before the end of the iteration are entered into the tracking system and implemented. New features requested or those too large to be implemented are considered an enhancement required and classified in the review list as urgent, important or desirable to have. These features are considered to be entered into the project backlog for subsequent iterations. <p>Once the usability testing review has ended and all the features have been analyzed, then the shippable product of the iteration is delivered.</p>

Figure 1 - How "Usability Testing" is applied in the solution pattern

Figure 2 shows when usability testing is used through an agile software development process. The process named "development iteration" has a symbol " μ " to identify the relation to the iteration process depicted in figure 3 which shows when usability testing is applied. In the diagrams the process identified as "Usability Testing" involves the activities described previously and depicted in figure 1.

Figure 2 - When pattern "Usability Testing" is applied in agile process

Figure 3 - When product pattern “Usability Testing” is applied in agile iteration

At the beginning of the software process the development team gets a scope of the project gathering enough information from users. Depending on time and resources, this scope could be a whole interaction design or just a general scope defining enough interaction design with subsequent enhancements during iterations. Thus, we have a few options:

- a) If agile team gets information to make a whole interaction design before any other development tasks then formative usability testing is applied. Initial features can be designed by creating low-fi prototypes (like paper prototypes) to be evaluated with usability testing and then from the feedback of this review define the product backlog. Once the whole scope is designed and product backlog defined then the normal agile process continues defining iteration backlog with the user stories to develop. After the iteration development, the usability feedback gathered is applied to the product backlog update if there is another iteration to be developed. We call this approach “Interaction Design First”. Figure 4 shows the way applying usability testing marked with bold arrows through the development process.

Figure 4 - Applying usability testing during iteration with approach “Interaction Design First”

- b) If at the beginning of the development process the agile team gets sufficient information to make a general interaction design and decides to apply formative usability, then initial features can be designed by creating low-fi prototypes (like paper prototypes) to be evaluated with usability testing and subsequently from the feedback of this

review define the initial product backlog. Once the general scope is designed and initial product backlog defined the normal agile process continues defining iteration backlog with the user stories to develop. Through the iteration development interaction design is refined by gathering data, creating low-fi prototypes and applying usability testing to them. Usability feedback is used to define which features are fixed during the actual iteration or considered in next iterations. When iteration development is finished, the usability feedback gathered is used to update the product backlog. We call this approach as “Minimal Up-Front Design”. Figure 5 shows the way applying usability testing marked with bold arrows through the development process.

Figure 5 - Applying usability testing during iteration with approach “Minimal Up-Front Design”

- c) Agile team gets just enough information at the beginning of the process and decides not to define interaction design. The product backlog is defined without previous formative usability testing. Once initial product backlog is defined then the normal agile process continues defining iteration backlog with the user stories to develop. Through the iteration development interaction design is defined for the actual user stories and enhanced in subsequent tasks or even iterations. We call this approach as “Implicit interaction design”. Figure 6 shows the way applying usability testing marked with bold arrows through the development process.

Figure 6 - Applying usability testing during iteration with approach “Implicit Interaction Design”

- d) If the development team has enough elements to work user interface design and implementation activities at the same time on parallel tracks, then get data from the user by developing research activities previous product backlog definition working this activities in a first iteration focused just to get enough information about the user and the software project (it could be named “cycle 0”). Once this initial iteration is complete initial backlog can be defined and start with the iterations where interaction design is implemented, tested and enhanced. Through iterations development team work with design track and implement track; at the design track the interface design is defined ahead for subsequently iterations, at the implement track use the feedback from previous iterations to develop the interface. We call this approach “Parallel Track”. Figure 7 depicts both design track (bold arrows) and implement track (dotted lines).

Figure 7 - Applying usability testing during iteration with approach “Parallel Track”

At the end of the iteration if there is a release programmed, then usability testing is shown as optional but this test is strongly recommended and again at the end of entire project.

Roles	<ul style="list-style-type: none"> Development Team (Agile Developers) Scrum Master Product owner (optional) Users (stakeholders, end-users, representative users) UI Designer, HCI expert, UX expert (optional)
Basic Knowledge	<p>Knowledge</p> <ul style="list-style-type: none"> Knowing basic concepts about software testing General scope of the software project under construction <p>Skills</p> <ul style="list-style-type: none"> Capability to apply user research to get data through direct observation. Capability to data Analysis identifying needs and establishing interaction requirements Capability to interact with the users in a friendly way.
Entries	Usability testing report
Exit	Reviewed usability Testing List per iteration
Lessons learned	<ul style="list-style-type: none"> Usability testing should be applied in a relaxed and friendly atmosphere aiming to get sincerely comments from the users. After the tests have been applied the results must be presented to the whole development team in a meeting to review the feedback gathered. The product pattern “agile usability testing” must be kept accessible to the team through the whole development process.
Examples	

Usability Testing Review List					
Project ID	TFSysystem				
ID Iteration:	03	ID Usability Testing:	UTR 0302	Date Testing:	20/Jan/2015
Objective of the test:					
This test aim to validate efficiency and functionality of the interfaces design of "Taxes register" and "Sales record" derived from the user story US10.					
ID	Observation			Classification*	
01	Botton "cancel" hasnot the first letter uppercase			B	
02	There is no option to apply taxes by city			FR	
03	Date field allow to capture month 13			B	
04	There is no confirmation window when user save the information			B	
05	Principal window close when option "search by name" is selected			B	
06	The list of products doesn't show the subtotal and total including IVA			FR	
07					
08					
09					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					

*B = Bug or feature to fixed on the iteration in course
FR = Feature required to be considered on the project backlog

Figure 8 - Usability Testing Review List

Templates	https://drive.google.com/file/d/0B1YqXzQvZlQuYXljZTdvejFKOTA/view?usp=sharing
Information Resources	<p>Dumas, J. S., & Redish, J. C. (1999). A practical guide to Usability Testing. (I. Books, Ed.). Intellect Ltd.</p> <p>Sharp, H., Rogers, Y., Preece J. (2007) , Interaction Design: beyond human-computer interaction. Ed John Wiley & Sons.</p> <p>Sanchez-Segura M., Medina-Domínguez F., Amescua A., Mora-Soto A. Improving the efficiency of use of software engineering practices using product patterns. Recovered May 2014 from http://www.sciencedirect.com</p> <p>Nielsen, J. (2008). Agile Development Projects and Usability. Jakob Nielsen's Alertbox. Retrieved from https://www.nngroup.com/articles/agile-development-and-usability.</p> <p>Barnum, C. M. (2010). Usability Testing Essentials. Morgan Kaufmann.</p>

Capítulo 6 Evaluación de los patrones propuestos

6.1 Introducción

Los patrones Agile Persona y Agile Usability Testing que se han mostrado en los capítulos 4 y 5 respectivamente, son la versión final que se ha obtenido tras un proceso de aplicación y validación en diferentes contextos mismo que se detalla en este capítulo. La sección 6.1 presenta la estrategia de evaluación que se siguió para validar los patrones propuestos en proyectos de software académicos y reales. En la sección 6.2 se describe la evaluación académica realizada con grupos de estudiantes de diferentes cursos trabajando con uno de ellos como el proyecto piloto.

La sección 6.3 describe la evaluación aplicada a los patrones en proyectos reales de desarrollo ágil en los contextos universitario e industrial. Los aspectos analizados en este proceso son: facilidad/dificultad de entender los patrones, efectividad de los patrones como guía para incorporar técnicas de usabilidad y los resultados obtenidos al utilizarlos. Estos últimos se estudian cualitativamente enfocados en el nivel de usabilidad percibido de las aplicaciones cuyo desarrollo incluyó el uso de los patrones. Finalmente en la sección 6.4 se presentan las conclusiones sobre los resultados obtenidos de la evaluación.

6.2 Estrategia de evaluación

El proceso de evaluación de los patrones definidos se dividió en dos fases: evaluación académica y evaluación en proyectos reales. La evaluación académica se desarrolló con el objetivo de evaluar la factibilidad de aplicación del patrón en proyectos de software e identificar mejoras a partir de los problemas que enfrentaron los equipos que utilizaron en inicio los patrones. Durante esta primera fase de evaluación el patrón utilizado fue el de “Agile persona” ya que al momento de la impartición de las asignaturas en las que se utilizó el patrón para probar su adecuación, el patrón “Agile usability Testing” se encontraba en proceso de definición. Sin embargo, los resultados observados en la evaluación de “Agile persona” fueron aprovechados también para mejorar el patrón de “Agile usability testing” principalmente en cuanto al formato con el que se describen los patrones. Una vez aplicadas las mejoras identificadas a partir de los resultados obtenidos y analizados durante la primera etapa de evaluación, los patrones fueron proporcionados a equipos ágiles de desarrollo para su uso en proyectos reales de software, realizando de esta manera la segunda etapa de evaluación. Para efectos de nuestro estudio y una mejor trazabilidad al seguimiento de actualizaciones, los patrones se fueron identificando por medio de versiones de acuerdo a los resultados obtenidos en las evaluaciones. Así por ejemplo, PAPv0 y PAUTv0 corresponden a las versiones iniciales de los patrones “Agile Persona” y “Agile Usability Testing”, respectivamente.

Los equipos que participaron tanto en la evaluación académica como en la evaluación en proyectos reales utilizaron SCRUM como metodología ágil. Hasta antes del momento de la evaluación de los patrones propuestos no habían considerado utilizar en sus proyectos de desarrollo alguna técnica de usabilidad.

Evaluación académica

Durante la evaluación académica el patrón “Agile persona” definido fue proporcionado a grupos de estudiantes con el objetivo de identificar posibles mejoras en la definición inicial del patrón. La evaluación académica se realizó con tres grupos de estudiantes correspondientes a dos diferentes asignaturas impartidas en diferentes universidades. El primer grupo de estudiantes corresponde a una asignatura impartida dentro del programa de Máster Universitario en Ingeniería del Software (EMSE por sus siglas en inglés) de la Universidad Politécnica de Madrid (UPM). Los objetivos y temas de esta asignatura contemplan que los estudiantes desarrollen un proyecto final utilizando una metodología ágil incorporando usabilidad, características idóneas para evaluar los productos propuestos y razón por cual uno de los equipos en los que se dividió el grupo de dicho programa fue considerado nuestro proyecto “piloto”.

En este programa nuestra propuesta fue proporcionada a dicho equipo y utilizada durante todo el proceso de desarrollo de su proyecto de asignatura. El segundo y tercer grupo de estudiantes corresponden a una

de las asignaturas impartidas en la Facultad de Matemáticas (FMAT) de la Universidad Autónoma de Yucatán (UADY). Los dos grupos evaluaron la definición del patrón “Agile Persona” en diferentes periodos escolares. En este programa los alumnos no desarrollaron todas las etapas del proceso de construcción del software por lo que únicamente se evaluó el resultado de aplicar el patrón “Agile Persona” en la etapa inicial del desarrollo de software para identificar los usuarios finales y sus requisitos. Durante la descripción de nuestro proceso de evaluación haremos referencia a los proyectos de evaluación académica de la siguiente forma:

PEAP. Proyecto de Evaluación Académica Piloto (EMSE-UPM)

PEAA (1 – 8). Proyecto de Evaluación Académica “A” (FMAT-UADY)

Evaluación en proyectos reales de desarrollo de software

La evaluación en proyectos reales de software se llevó a cabo por equipos ágiles correspondientes a dos instituciones, una educativa y otra pública, que utilizan metodologías ágiles. En este caso fueron proporcionados a los equipos ambos patrones objetos de nuestro estudio “Agile Persona” y “Agile Usability Testing” para utilizarlos durante todo el proceso de desarrollo del software. El primer grupo de equipos ágiles estuvo conformado por tres equipos de desarrollo trabajando un proyecto real en un entorno universitario. El objetivo del proyecto de software es desarrollar juegos interactivos que sirvan como apoyo para la retroalimentación y refuerzo de los temas vistos con alumnos de los diferentes grados de educación básica primaria. Cada equipo estuvo conformado por cuatro alumnos, tres de ellos desempeñando los roles de desarrolladores y uno como experto en la enseñanza de las matemáticas. El software es parte de un proyecto social educativo realizado por la Universidad Autónoma de Yucatán (México) a través de la Facultad de Matemáticas en conjunto con el laboratorio de mantenimiento de equipos de cómputo y los alumnos de las diferentes licenciaturas de las áreas de computación y matemáticas.

Los equipos ágiles del segundo grupo pertenecen al departamento de Innovación e Implementación de sistemas adscrito al Poder Judicial del Estado de Yucatán. En este departamento cada uno de los equipos tiene a su cargo el mantenimiento de los sistemas existentes y el desarrollo de nuevos proyectos de software que requieran ser implementados para cubrir las necesidades de información que surgen en la empresa pública. Cada uno de los equipos estuvo conformado en promedio de seis miembros: tres desarrolladores (Development team), Scrum Master, tester y el Product Owner. Fueron evaluados seis proyectos, dos sistemas en su etapa de mantenimiento y cuatro proyectos nuevos de software desarrollando los módulos iniciales: Sistema de Administración de nóminas (mantenimiento), Sistema de Administración de fondo auxiliar (mantenimiento), Sistema informático de gestión de ejecución de sentencias, Sistema de Armonización contable, Sistema informático para justicia especializada para adolescentes (Módulo agenda de audiencias), Sistema Informático de Almacén, compras y activo fijo. Durante el desarrollo de este capítulo haremos referencia de la siguiente manera a los proyectos de software reales en los que fueron evaluados los patrones:

PRAU. Proyecto Real de Evaluación en un Entorno Universitario (FMAT- UADY).

PREI (1 – 6). Proyectos Reales de Evaluación en Industria. Tribunal de Justicia del Estado de Yucatán (México).

A cada grupo de los desarrolladores que participaron en los proyectos PRAU o PREI haremos referencia como grupo de evaluación PRAU y grupo de evaluación PREI respectivamente.

Períodos de evaluación

Los momentos de evaluación académica y evaluación en proyectos reales de desarrollo se ilustran con diferente color en la figura C6-1. Como podemos observar durante el tiempo en el que se utilizaba el patrón “Agile Persona” en el proyecto piloto de evaluación académica (PEAP) con el grupo de

estudiantes de la UPM el mismo patrón también era utilizado por el primer grupo de estudiantes de la UADY (PEAA).

Figura C6-1 Etapas de evaluación de los patrones definidos

La tabla C6-1 resume información sobre los grupos participantes durante la evaluación académica y en proyectos reales de los patrones “Agile Persona” y “Agile Usability Testing”. La columna “ID Eval” corresponde al identificador utilizado para cada grupo que evaluó los productos patrón. La columna “Descripción” corresponde a la descripción del grupo al que pertenecen los equipos que utilizaron las guías para su evaluación; “Equipos ágiles” es el número de equipos de desarrollo que participaron en la evaluación de los patrones. La columna “Período de evaluación” nos proporciona información sobre el momento durante el cual se realizó la evaluación. No todos los grupos tuvieron oportunidad de evaluar los dos productos patrón definidos; en la columna “Patrón producto evaluado” podemos consultar cuál fue el patrón evaluado durante el correspondiente período. Los proyectos evaluados en proyectos reales son aquellos identificados en la tabla con filas sombreadas.

Tabla C6–1 Grupos participantes en la revisión de los productos patrón

ID Eval	Descripción	Equipos ágiles	Período de evaluación	Patrón producto evaluado
PEAP	Grupo de Master Europeo en Ingeniería de Software (EMSE) impartido en la escuela de ingenieros técnicos en informática de la universidad Politécnica de Madrid (España).	1	Septiembre 2014 – Febrero 2014	Agile Persona
PEAA	Grupo conformado por alumnos del grado de Licenciatura en Ingeniería de Software impartido en la Facultad de Matemáticas de la Universidad Autónoma de Yucatán (México).	4	Septiembre 2014 – Febrero 2014	Agile Persona
PRAU	Grupo conformado por alumnos de las diferentes licenciaturas de las áreas de computación y matemáticas en la Facultad de Matemáticas de la Universidad Autónoma de Yucatán (México).	6	Septiembre 2015 – Febrero 2016	Agile Persona Agile Usability Testing
PREI	Departamento de Informática del Tribunal Superior de Justicia del estado de Yucatán (México).	6	Septiembre 2015 – Marzo 2016	Agile Persona Agile Usability Testing

Metodología para la evaluación de los productos patrón

Para la evaluación de los productos patrón definidos en nuestro estudio consideramos la siguiente metodología común en todos los proyectos que realizaron la evaluación:

- Conformación de los equipos. Para la evaluación académica los equipos de desarrollo fueron conformados por los mismos alumnos mientras que en la evaluación con proyectos reales tanto en ambiente universitario como en la industria los equipos de desarrollo trabajaron de acuerdo a la estructura definida previamente por la organización correspondiente.

- Taller. En ambas fases de evaluación se proporcionó un taller sobre conceptos de usabilidad y en particular sobre las técnicas de Persona y Usability Testing.
- Aplicación de los productos patrón. Se proporcionó a los equipos los productos patrón a evaluar así como las plantillas relacionadas con éstos. La información también se mantuvo disponible en un espacio on-line para consulta de los equipos en cualquier momento.
- Seguimiento y recabación de resultados. Los resultados de la evaluación se recabaron utilizando una o varias de las siguientes técnicas: observación, revisión y análisis de documentación, entrevista y encuesta.

6.3 Evaluación Académica

En esta etapa del proceso de evaluación, el patrón “Agile Persona” en su primera versión se proporcionó a grupos de estudiantes de diferentes cursos académicos para identificar posibles mejoras a implementar en el diseño definido para su posterior validación en proyectos reales. El patrón se evaluó por tres grupos de estudiantes en diferentes momentos.

6.3.1 Contexto y participantes

Tal como se ha indicado anteriormente, uno de los equipos del grupo cursando la asignatura del EMSE en la que se utilizó metodologías ágiles durante el desarrollo del proyecto final del curso fue considerado el proyecto piloto PEAP (ver tabla C6-2). Este equipo utilizó SCRUM como metodología ágil utilizando el patrón para incorporar la técnica persona durante todo el proceso de desarrollo. Los equipos en los que se dividió el grupo PEAA únicamente utilizaron el patrón en las etapas iniciales del desarrollo de software durante la identificación de los usuarios finales y los requerimientos. Esto es debido a que, aunque el objetivo de la asignatura en la que se evaluó el patrón remarca la importancia de la usabilidad durante todo el proceso de desarrollo de software independientemente de la metodología de desarrollo utilizada, el período de tiempo del curso en el que fue evaluado el patrón limitó únicamente el uso de técnicas de usabilidad en las etapas iniciales del desarrollo de software. Los proyectos van avanzando en el desarrollo durante asignaturas posteriores impartidas en cursos siguientes.

La tabla C6-2 muestra información sobre cada uno de los proyectos que participaron en la evaluación académica del patrón. La columna “ID Equipo” lista el identificador con el que hacemos referencia a cada uno de los equipos que participaron en la evaluación académica del patrón “Agile Persona”; “Descripción del proyecto” describe de manera general el proyecto en el que cada equipo de estudiantes aplicó el patrón; “Momento de aplicación del patrón” el momento del proceso de desarrollo de software en el que fue aplicado el patrón por los equipos de estudiantes.

Tabla C6-2 Equipos participantes en la evaluación académica de los patrones

ID Equipo	Descripción del proyecto	Momento de aplicación del patrón
PEAP	Sitio web de venta de software e intercambio de ideas sobre desarrollo de software	Todo el proceso de desarrollo
PEAA1	COMEME. Aplicación para el control y seguimiento de pedidos de comida a domicilio.	Identificación de usuarios finales y definición de requisitos
PEAA2	K-FMAT. Aplicación móvil para realizar pedidos a la cafetería vía equipos móviles.	
PEAA3	ProAccountant. Aplicación móvil como apoyo para empresas PYMES en la administración financiera de sus ingresos y egresos.	
PEAA4	ARopaT. Aplicación móvil con el objetivo de proporcionar sugerencias de “Outfits” diarios.	
PEAA5	BuscaPerro. Aplicación móvil para ayudar a la ciudadanía a encontrar mascotas extraviadas o en adopción.	
PEAA6	BuStop. Aplicación móvil con el objetivo de orientar correctamente a los usuarios en el uso del transporte público para llegar a su destino.	
PEAA7	CrowID. Aplicación móvil que permite encontrar y registrarse a eventos de corte social que se llevan a cabo dentro de México.	
PEEA8	CFE, Control Financiero Escolar.	

6.3.2 Proceso de evaluación

A continuación se describe cómo la metodología mencionada en la sección anterior se aplicó con los grupos que participaron en la evaluación académica:

Conformación de los equipos

Los equipos de desarrollo fueron integrados por estudiantes de los diferentes cursos en los que fue evaluado el patrón correspondiente. Los mismos estudiantes decidieron la organización y composición de los equipos. En el caso del grupo de la asignatura del EMSE, una vez conformados los equipos fue solicitada la participación voluntaria de uno de éstos para la evaluación del patrón.

Taller

Con el objetivo de que los estudiantes contaran con el conocimiento necesario sobre la técnica de usabilidad “Persona” y el patrón para que pudieran aplicarlas correctamente durante el proyecto, les fue impartido un taller sobre los conceptos básicos de dicha herramienta y sobre la forma en la que estaba definido y estructurado el patrón “Agile Persona” incluyendo las plantillas del mismo.

Aplicando los producto patrón como guías

Se proporcionó a los equipos participantes el documento del patrón “Agile Persona” versión PAPv0 así como la plantilla relacionada con este. La información se mantuvo disponible en un espacio on-line para consulta de los equipos en cualquier momento. En el caso del grupo PEAA la documentación relacionada con el patrón estuvo disponible en el espacio on-line destinado a la asignatura en plataforma Moodle durante todo el curso.

Seguimiento y recabación de resultados

La retroalimentación necesaria por parte de los alumnos sobre los resultados de aplicar las guías se obtuvo mediante observación, reportes del curso y/o entrevistas. En la tabla C6-3 tenemos los diferentes medios utilizados en cada uno de los grupos. La columna “Equipo” corresponde al identificador de cada uno de los grupos participantes en nuestro estudio y en la columna “Fuente” tenemos el método de retroalimentación utilizado para obtener información sobre el uso del patrón por parte de los grupos.

Tabla C6–3 Fuentes de retroalimentación durante la evaluación Académica

Equipo	Fuente
PEAP	Observación. Los alumnos presentaron informes semanales del avance del desarrollo de su proyecto semanalmente. Durante estas presentaciones también informaban de la forma en la que habían trabajado las definiciones de las personas relacionadas con sus proyectos.
	Reportes del curso. Al final del curso los equipos entregaron un informe final sobre el proceso de su proyecto de asignatura.
	Entrevista. Al inicio del desarrollo del proyecto se realizaron entrevistas con el equipo.
PEAA	Reportes del curso. Durante el curso los equipos entregaron informes sobre el avance de su proyecto. Estos informes fueron analizados para obtener información de la manera en la que iban utilizando el patrón.

La evaluación académica fue aplicada en diferentes proyectos que describimos en los apartados siguientes.

6.3.3 Proyecto Piloto

El objetivo de trabajar en nuestra investigación con un proyecto piloto es identificar las posibles mejoras de la primera versión de los productos patrón. Como mencionamos previamente el proyecto elegido como piloto, por sus características similares a la de aquellos proyectos reales en donde se pretende sean utilizadas las soluciones propuestas, fue desarrollado por un equipo de alumnos del EMSE impartido en la Escuela Técnica Superior de Ingenieros en Informática de la Universidad Politécnica de Madrid. Como se

observa en la tabla C6-2 el proyecto desarrollado por dicho equipo es un sitio web cuyo objetivo es servir como plataforma de venta de software e intercambio de ideas sobre desarrollo de software. El equipo adoptó la metodología SCRUM y adicionalmente TDD (Test Driven Development).

6.3.3.1 Recabación de información

La metodología mencionada en la sección 6.3.2 se aplicó de la siguiente forma:

Conformación de los equipos

El equipo voluntario participante en la evaluación del patrón fue uno del total de los cinco equipos que fueron conformados para fines de la asignatura. Dicho equipo estuvo integrado por cuatro estudiantes en el rol de equipo de desarrollo (Development Team) y uno de ellos se desempeñó adicionalmente como Scrum Master. La figura de Product Owner fué representada por el profesor.

Taller

Se impartió un taller al grupo completo de la asignatura sobre los conceptos básicos de la técnica “Persona” y sobre la forma en la que estaba definido y estructurado el patrón “Agile Persona” incluyendo las plantillas del mismo. De esta forma el equipo voluntario que utilizó el patrón adquirió los conocimientos necesarios para aplicarlos posteriormente durante el proyecto.

Aplicando los producto patrón como guías

Se proporcionó al equipo ágil participante el documento del patrón “Agile Persona” versión PAPv0 cuyo diagrama solución se muestra en la figura C6-2 así como la plantilla relacionada con este. El equipo mantuvo la información disponible en un espacio on-line para consulta de los elementos del equipo en cualquier momento.

Seguimiento y recabación de resultados

En la tabla C6-3 descrita previamente se especifican las fuentes de retroalimentación utilizadas con el equipo del proyecto piloto PEAP.

A partir de dichas fuentes se desarrollaron las diferentes actividades listadas en la tabla C6-4 para recabar la información relacionada con los resultados de uso del patrón “Agile persona” en el proyecto piloto. La columna “Actividad” corresponde a las actividades que se realizaron con el equipo, “Objetivo” es la descripción de la meta que se fijó para la actividad y “Fecha de realización” es la fecha en la que se llevó a cabo la actividad indicada.

Tabla C6-4 Actividades realizadas en el PEAP como parte del proceso de evaluación del patrón "Agile persona"

Actividad	Objetivo	Fecha realización
Taller usabilidad – Técnica persona	Presentar al grupo de estudiantes de la asignatura del EMSE en la que se evaluará el formato y contenido del patrón los conceptos generales sobre la técnica persona.	14 Noviembre 2014
Reunión inicial	Informar al equipo de desarrollo del proyecto piloto el objetivo de su participación en la evaluación del patrón “Agile Persona”.	14 Noviembre 2014
Segunda reunión	Obtener observaciones preliminares del equipo piloto sobre la estructura y contenido del patrón “Agile Persona”.	20 Noviembre 2014
Sprint Review – Sprint 1	Observar por medio de los informes de cada equipo en particular del equipo participante en nuestra investigación, la forma en la que utilizan el patrón para definir las personas y como solventan los	28 Noviembre 2014

	posibles problemas (en caso de haberlos) a los que se enfrentan al utilizarlo.	
Sprint Review – Sprint 2	Observar por medio de los informes de cada equipo en particular del equipo participante en nuestra investigación, la forma en la que utilizan el patrón para definir las personas y como solventan los posibles problemas (en caso de haberlos) a los que se enfrentan al utilizarlo.	12 Diciembre 2014
Reporte final de curso	Revisar la documentación final del proyecto piloto para obtener información relevante sobre el uso del patrón, beneficios y problemas con los que enfrentó el equipo al utilizarlo durante el proceso de desarrollo ágil del software.	Febrero 2015

6.3.3.2 Revisión preliminar del proyecto

Una vez impartido el taller sobre la técnica persona se le proporcionó al equipo del PEAP el documento del patrón a evaluar identificado para efectos de nuestro estudio, como mencionamos previamente, como PAPv0 cuyo diagrama solución se muestra en la figura C6-2. Siguiendo el proceso indicado en el patrón antes de desarrollar los sprints el equipo comenzó identificando y describiendo las definiciones iniciales de sus personas; posteriormente mediante entrevista no estructurada con el equipo del PEAP se obtuvieron sus primeras observaciones sobre el formato y contenido del mismo. Como resultado, el diagrama solución de la versión PAPv0 del patrón fue modificado en algunos de los procesos que se realizan durante el desarrollo del sprint, la figura C6-3 muestra el diagrama solución generado al aplicar dichos cambios e incluido en la nueva versión PAPv1 del patrón “Agile persona”.

En la versión actualizada PAPv1 el proceso “Identify users involved in the user story” fue eliminado del diagrama solución debido a que el proceso “Relate persona and user story” abarca el mismo objetivo de verificar que las personas definidas están relacionadas con las historias de usuario a desarrollar durante el sprint. La segunda adecuación al diagrama fue cambiar el nombre y objetivo del proceso “User Story development” por el de “Detailed definition of the user story”; el equipo identificó que una vez definidas y/o actualizadas las descripciones de las personas definidas era necesario revisar también las tareas y criterios de aceptación de la historias de usuario en desarrollo. Por último, se incorporó el proceso “Sprint Implementation” que corresponde a realizar aquellas tareas contempladas en el sprint para la historia de usuario en desarrollo.

Figura C6-2 Diagrama solución del patrón “Agile persona” versión PAPv0

Figura C6-3 Diagrama solución del patrón "Agile persona" versión PAPv1

6.3.3.3 Análisis de Datos

Una vez aplicadas las primeras modificaciones, la versión PAPv1 fue enviada al equipo para su utilización durante el desarrollo de los sprints. Siguiendo el proceso guía del patrón “Agile persona” el equipo describió y priorizó cinco personas al inicio de su proceso de desarrollo antes de definir su product backlog. Esta aplicación de la técnica abarca los procesos descritos previamente: “Minimal user research”, “describe and prioritize persona”, “review persona definition” y, en caso de ser necesario, “Update description and prioritize persona”. A partir de las personas definidas y clasificadas se obtuvo un product backlog inicial con 18 historias de usuario. La definición de las personas y el product backlog inicial se pueden consultar en el Anexo B de este documento.

Se desarrollaron tres sprints durante los cuales el equipo del proyecto piloto utilizó el patrón “Agile persona”. A continuación se presentan extractos de sus seguimientos en donde se observan las tareas definidas en cada sprint. La tabla C6-5 muestra las tareas correspondientes al sprint planning y al sprint 1 en el cuál, de acuerdo a su reporte final, consideraron el desarrollo de dos historias de usuario: “Create an account” y “Login to an account”. En el registro de sus tareas observamos (sombreado en gris) que consideraron como parte de su sprint planning la tarea “Personas definition” incorporando de esta manera la técnica persona en otro momento además del indicado en la solución propuesta en el product pattern “Agile persona” donde se indica aplicar la técnica antes de definir el product backlog inicial y al inicio del desarrollo de los sprints.

Tabla C6–5 Tareas del sprint planning y el sprint backlog correspondiente al sprint 1 equipo PEAP

Backlog item	Task	Task Owner	Initial Estimation	Hours of Work Reaming on each day						
				Sat	Sun	Mon	Tus	Wed	Thu	Fri
Sprint planning	Personas definition	MD, TC, AZ, PN.	3h	3	3	0	0	0	0	0
	Project setup	MD, TC, AZ, PN.	8h	5	2	0	0	0	0	0
	Learning	MD, TC, AZ, PN.	8h	5	3	2	0	0	0	0
Create an account (5 points)	Create test cases	MD, TC	1 h	1	1	1	0	0	0	0
	Implement front-end	MD, TC	4 h	4	4	4	3	2	0	0
	Implement back-end	AZ, PN	3 h	3	3	3	1	0	0	0
	Complete testing	AZ, ON	2 h	2	2	2	2	2	0	0
Login to an account (8 points)	Create test cases	MD, TC	1 h	1	1	0	0	0	0	0
	Implement front-end	MD, TC	6 h	6	5	3	1	0	0	0
	Implement back-end	AZ, PN	5 h	5	2	0	0	0	0	0
	Complete testing	AZ, PN	2 h	2	2	2	1	0	0	0
Total			43 h	37	28	17	8	4	0	0

El equipo del proyecto PEAP menciona que durante el primer sprint de su proyecto observaron que las historias definidas reflejaban las funcionalidades requeridas prácticamente por todas las personas identificadas previamente, ellos atribuyeron esto al hecho de que las funcionalidades eran muy básicas por ser el inicio del proyecto.

“It is important to note that due to the very basic nature of the user stories and tasks involved in the first two weeks, user stories we incorporated were shared by practically all personas...”

A excepción del registro de seguimiento del sprint 1 donde claramente el equipo PEAP definió una tarea en particular para la definición de personas (Ver figura 6-4), en los seguimientos de sus sprints posteriores no se identifican aquellas tareas relacionadas con la revisión y/o actualización de las personas definidas o la creación de nuevas. Las historias de usuario consideradas en los sprint 2 y 3 del equipo PEAP las tenemos en el anexo B.3.

La tabla C6-6 muestra las observaciones relevantes para nuestra investigación obtenidas de las etapas descritas por el equipo del proyecto piloto en su reporte final. La columna “Proceso” corresponde a la fase del proceso ágil donde se aplicó el patrón y “Comentarios” son aquellos comentarios documentados por el equipo en relación al uso de la técnica persona siguiendo el patrón “Agile persona” durante la fase correspondiente.

Tabla C6-6 Observaciones del equipo del PEAP durante el proceso del desarrollo de software

Proceso	Comentarios
Definición de personas	<i>“Our method of applying the Personas technique consisted in taking the list of personas and descriptions that was established before the sprints began (though it suffered modifications as time went on) and during each sprint evaluating stories from the product backlog to better define them and their further establish goals and tasks/subtasks associated with each user story”</i>
Initial Product Backlog definition	<i>“...Based on the personas and guiding principles we established, we next aimed to use the personas to create product backlog for the agile process and to prioritize the backlog according to the guiding “driving factors” and the personas’ individual needs...”</i>
Sprint Planning sprint 1	<i>“...During Sprint 1, a sixth persona, Andrew Sartorius (Startup Entrepreneur), was split off from Spencer Johnson as we realized there could be two very different types of entrepreneurs when evaluating our progress up to that point”</i>
Sprint 1	<i>“...We used our personas to evaluate the final products of the sprint”</i> <i>“...taking into account the personas evaluation, we decided to plan for the second sprint the following user stories...”</i>
Sprint 2	<i>“We used our Personas to evaluate the final products of the sprint”</i> <i>“As the leader for IdeaXchange system, I want to plan for our next sprint. I wish to analyze the personas of my customers, achieve their user stories and divide the user stories into small tasks, then estimate how much time will be spent on each task”</i>
Sprint 3	<i>“We used our personas to evaluate the final products of the sprint”</i>

En la tabla C6-7 se presentan las cuestiones que consideramos relevantes para nuestro estudio como resultado de esta primera evaluación aplicada al producto patrón “Agile persona” por el equipo de

desarrollo PEAP. La columna “Situación” se refiere a la situación que se presentó durante el desarrollo del proyecto; “Comentarios” es un extracto de lo documentado por el equipo en su reporte final respecto al punto que surgió; “Solución” describe la forma en la que el equipo dio seguimiento a la situación mencionada.

Tabla C6–7 Situaciones relevantes identificadas a partir del reporte final del equipo y la forma en la que abordaron la situación

Situación	Comentarios	Solución
Durante el sprint planning se identifica una nueva persona	<i>“During Sprint 1, a sixth persona, Andrew Sartorius (Startup Entrepreneur), was split off from Spencer Johnson as we realized there could be two very different types of entrepreneurs...”</i>	Consideraron una tarea durante el sprint planning para definir la persona que surgió en esta etapa además de las definidas previamente al product backlog
Revisión del Demo (Sprint Review)	<p><i>“We found that the Personas technique was less useful in identifying user stories that needed to be completed at least in the first couple sprints and much more useful at this stage of development in evaluating the product as progress was made to identify rising deficiencies...”</i></p> <p><i>“Evaluating the final product from the point of view of all personas allowed us to see how the final product was being shaped after each user story was incorporated and provide critical feedback on the direction the product was taking and the direction in which it needed to head...”</i></p>	Utilizan las personas identificadas para para evaluar los productos potencialmente útiles con los usuarios finales al final de cada sprint.

Analizando la información proporcionada por el equipo del proyecto PEAP se identificaron mejoras a aplicar en el diagrama de la solución propuesta del patrón “Agile persona”. En la figura C6-4 se muestra el diagrama de los momentos en los que se propone aplicar la técnica persona, procesos sombreados y con diferente color para identificarlos mejor. El diagrama se dividió para hacerlo más claro y representar mejor los momentos en los que se propone aplicar la técnica y cómo aplicarla, incluyendo el sprint planning mencionado por el equipo PEAP. El proceso de aplicación de esta técnica en el diagrama se identificó como “Apply persona”, los momentos en los que se utiliza y que implican otros procesos se identifican con (*) y (**) para indicar el sub diagrama adjunto donde consultar las actividades. El segundo punto que mencionó el equipo sobre la utilidad que percibió al utilizar como referencia las personas definidas durante la revisión del demo en el sprint review se plasmó en el diagrama agregando un comentario al proceso “Shippable product delivery”. En la figura C6-5 tenemos a detalla el diagrama del proceso para aplicar persona que describe cómo aplicar la técnica persona durante el proceso ágil. Ambos modificados a partir de las mejoras obtenidas a partir de la validación del equipo del proyecto PEAP.

Figura C6-4 Diagrama solución modificado de acuerdo a observaciones finales del equipo PEAP. Patrón “Agile Persona” versión PAPv2

Figura C6-5 Cómo aplicar la técnica persona en un entorno ágil (proceso “Apply persona”)

El equipo PEAP no tuvo observaciones sobre la plantilla proporcionada en ambas versiones del patrón PAPv0 y PAPv1 y descrita en el capítulo 4 sección 4.3.1, por lo que no fue necesario aplicar cambios en el formato de la misma.

6.3.4 Proyectos Docentes

Como se ha indicado en la sección 6.1 la evaluación se desarrolló en una de las asignaturas perteneciente al programa de grado Licenciatura en Ingeniería de Software impartido en la Universidad Autónoma de Yucatán, México (UADY) durante el curso Agosto 2014 – Diciembre 2014. El grupo de este programa lo identificamos como mencionamos previamente para fines de nuestro estudio como el grupo de evaluación PEAA (tabla C6-1). El grupo participó en un taller sobre el uso del patrón “Agile persona”, posteriormente la información relacionada con el patrón les fue proporcionada y puesta a su disposición on-line a través de la plataforma web de la materia para su consulta en cualquier momento que así fuera requerido. En secciones previas indicamos que la evaluación sobre el patrón “Agile persona” por parte del grupo de estudiantes se limitó a aplicarse durante la primera etapa del desarrollo del software para identificar los usuarios finales y obtener el diseño inicial de lo que serán las aplicaciones a desarrollar e implementar. Los equipos utilizaron la parte inicial de la solución propuesta en el patrón y la plantilla proporcionada. Como observaremos más adelante algunos equipos optaron por no utilizar el patrón para identificar sus usuarios finales y otros utilizaron la plantilla aplicando ciertas modificaciones a la proporcionada como parte de nuestra solución.

Dado el alcance de la asignatura y el objetivo de nuestra evaluación en el caso del grupo PEAA en donde nuestro interés era verificar que los conceptos presentados en nuestro producto patrón eran claros, no fue necesario que los equipos utilizaran alguna metodología ágil en particular para la gestión y desarrollo de su proyecto final de software. El resultado de lo obtenido durante su proyecto fue registrado en el documento de Especificación de Requisitos de Software entregado al final del curso. Los proyectos inicialmente definidos en esta asignatura van avanzando en cuanto al desarrollo de las aplicaciones en cursos posteriores conforme los alumnos van cubriendo los cursos del programa de formación universitaria. Para ambos equipos el uso de la información contenida en el patrón fue considerado como opcional, de ahí que algunos equipos no utilizaran la plantilla proporcionada en él.

6.3.4.1 Recabación de la información

La metodología utilizada con el grupo PEAA para la evaluación del patrón “Agile Persona” se describe a continuación:

Conformación de los equipos

Fueron conformados un total de 8 equipos de trabajo en el grupo PEAA en donde los mismos estudiantes decidieron la organización y composición de cada equipo. La tabla C6-8 muestra la manera en la que estuvo conformado cada equipo. “ID Equipo” corresponde al identificador asignado a cada equipo durante nuestro estudio, “Descripción del proyecto” es la descripción del proyecto que el equipo trabajó durante el curso, “# estudiantes” indica el número de alumnos que conformaron el equipo.

Tabla C6-8 Descripción de los proyectos en el grupo de evaluación PEAA

ID Equipo	Descripción del proyecto	# elementos
PEAA1	Aplicación móvil COMEME. Aplicación para el control y seguimiento de pedidos de comida a domicilio. La finalidad de este sistema consta en la mejora en productividad de pequeñas y medianas empresas que involucren logística en pedidos día a día, facilitándoles la administración de encargos, entregas y pedidos.	4
PEAA2	K-FMAT. Aplicación móvil para realizar pedidos a la cafetería vía equipos móviles.	3
PEAA3	ProAccountant. Aplicación móvil como apoyo para empresas PYMES en la administración financiera de sus ingresos y egresos.	4
PEAA4	ARopaT. Aplicación móvil con el objetivo de proporcionar sugerencias de “Outfits” diarios.	4
PEAA5	BuscaPerro. Aplicación móvil para ayudar a la ciudadanía a encontrar mascotas extraviadas o en adopción.	4
PEAA6	BuStop. Aplicación móvil con el objetivo de orientar correctamente a los usuarios en el uso del transporte público para llegar a su destino.	4
PEAA7	CrowID. Aplicación móvil que permite encontrar y registrarse a eventos de corte social que se llevan a cabo dentro de México.	4
PEAA8	CFE, Control Financiero Escolar. Aplicación web cuyo objetivo es servir como una herramienta tecnológica de apoyo a las instituciones educativas para llevar el control de sus ingresos y egresos económicos.	4

Taller

Con el objetivo de que los estudiantes contaran con el conocimiento necesario sobre la técnica de usabilidad “Persona” y el patrón para que pudieran aplicarlas correctamente durante el proyecto, les fue impartido un taller sobre los conceptos básicos de dicha herramienta y sobre la forma en la que estaba definido y estructurado el patrón “Agile Persona” incluyendo las plantillas del mismo.

Aplicando los producto patrón como guías

Como mencionamos previamente el grupo de evaluación PEAA evaluó el patrón “Agile persona” durante el mismo período que el equipo del proyecto piloto PEAP por lo que utilizaron la misma versión inicial del patrón PAPv0. El documento correspondiente se proporcionó a los equipos ágiles participantes así como la plantilla relacionada con este. La información referente al producto patrón evaluado se mantuvo disponible en el espacio on-line de la asignatura a través de la plataforma Moodle para consulta de los equipos en cualquier momento.

Seguimiento y recabación de resultados

En la tabla C6-3 descrita previamente tenemos las fuentes de retroalimentación utilizadas con el grupo PEAA.

6.3.4.2 Grupo de Evaluación PEAA

Durante septiembre 2014 – Diciembre 2014, mismo período en el que el equipo del proyecto piloto PEPP utilizaba el patrón “Agile Persona”, el grupo PEAA, a su vez contaba con la información y documentación necesaria del patrón para utilizarlo durante la fase inicial del desarrollo de un proyecto de

software. Para cubrir el objetivo de la asignatura definido como *Aplicar los fundamentos y métodos utilizados para la construcción de interfaces de usuario que faciliten el uso de la computadora por los usuarios finales*, al final del curso los alumnos definen y aplican una prueba de usabilidad de acuerdo a escenarios definidos en base a los perfiles de usuarios identificados. Tomando en cuenta el objetivo de la asignatura y el resultado esperado de la misma consideramos que la evaluación de los equipos proporcionaría información relevante que nos permitiera identificar puntos de mejora durante la etapa inicial de la solución con la consecuente actualización de la definición de nuestra propuesta a ser utilizada posteriormente por equipos ágiles de desarrollo en proyectos reales.

En la figura C6-6 se muestra el diagrama solución que corresponde a la versión del patrón “Agile persona” versión PAPv0 misma que fue evaluada inicialmente por el equipo PEAP.

Figura C6-6 Diagrama solución del patrón “Agile persona” versión PAPv0

De acuerdo al objetivo y período del curso, la actividad principal del diseño de interfaz de los equipos se centró al inicio del proceso por lo que en este caso, como mencionamos previamente, para nuestra investigación no se tuvo la oportunidad de evaluar los resultados en el segundo momento indicado en la solución del patrón para aplicar la técnica persona. Sin embargo, la aplicación de la técnica persona por parte de los equipos del grupo PEAA para identificar a los usuarios finales y a partir de esto las funcionalidades necesarias a considerar para aplicar finalmente la prueba de usabilidad correspondiente proporciona datos a nuestra investigación.

La plantilla proporcionada para evaluación del grupo PEAA es la misma que se proporcionó al equipo del proyecto piloto PEAP descrita en el capítulo cuatro sección 4.2.1. La plantilla y ejemplo fueron publicados en el sitio on-line de la materia al igual que el documento completo del patrón “Agile persona” permaneciendo disponibles para aquellos equipos que utilizaran el modelo del patrón “Agile persona”.

6.3.4.3 Análisis de los datos

Los alumnos hicieron tres entregas de seguimiento durante el desarrollo de la materia. A partir del análisis de la información documentada y presentada por los equipos en cada una de las entregas se extrajeron

aquellos datos que se consideraron relevantes para nuestro estudio respecto al uso que dieron los equipos de la información proporcionada en el patrón. Parte del resultado de nuestro análisis sobre la documentación proporcionada por cada equipo en las entregas programadas se describe en la tabla C6-9. La columna “Equipo” corresponde al identificador asignado al equipo para efectos de la investigación (ver tabla C6-8) para consulta de la descripción de cada uno; las columnas “Obs. E1”, “Obs. E2” y “Obs. E3” nos proporcionan información de las observaciones sobre la documentación analizada en las primera, segunda y tercera entregas respectivamente.

Tabla C6-9 Observaciones relacionadas con el uso del patrón “Agile persona” por el grupo PEAA

Equipo	Obs. E1	Obs. E2	Obs. E3
PEAA1	Mencionan el nombre de las personas que identificaron pero no se presentan las definiciones.	Ninguna referencia a la definición de personas utilizando el patrón “Agile persona”.	No presentan definición alguna de las personas identificadas. Únicamente mencionan cuáles fueron las personas identificadas para el proyecto.
PEAA2	Mencionan a los dos tipos de personas identificadas pero no utilizan el formato proporcionado.	Se mencionan los mismos dos tipos de personas indicadas en la primera entrega.	Se hace referencia a los dos tipos de usuarios identificados y clasificados al principio del proyecto. No hay nuevas definiciones de usuarios potenciales que hayan sido identificados al final del proyecto.
PEAA3	Utilizaron la plantilla proporcionada definiendo una persona. Los campos utilizados en la definición son: Nombre, edad, ocupación, actividades y metas y necesidades. No asignaron la clasificación del tipo de persona definida.	Información No disponible	No hay definiciones adicionales de persona.
PEAA4	Utilizaron la plantilla de “Agile persona” para definir su persona “Estudiante”.		Se menciona a una segunda persona pero no se presenta la definición de la misma. La primera persona definida se mantuvo sin cambios en su descripción.
PEAA5	Utilizan la plantilla proporcionada definiendo dos personas sin embargo no asignaron una clasificación.	No mencionan nuevas definiciones de persona o modificaciones aplicadas a la persona definida inicialmente. Hacen referencia a “Perfil de usuario”.	

PEAA6	Utilizan la plantilla proporcionada en el patrón definiendo una persona sin embargo, no asignaron clasificación. De la plantilla de “Agile persona” utilizan para la definición los siguientes campos: nombre, edad, ocupación, actividades, metas y necesidades.	No mencionan nuevas definiciones de persona o modificaciones a la descrita en la primera entrega. Hacen referencia a “Perfil de usuario”	Hacen referencia a dos nuevas personas (supervisores y usuarios terciarios) se indica la clasificación de cada una sin embargo no presentan la descripción de éstas utilizando el patrón “Agile persona”.
PEAA7	Utilizan la plantilla proporcionada en el patrón “Agile persona”, definen una persona “Persona 1” definiendo como ocupación “Estudiante” y asignan la clasificación correspondiente.	Anexan la definición de una nueva persona “Persona 2” definiendo como ocupación “Organizador de eventos” utilizando la plantilla del patrón “Agile persona”. El equipo presenta el escenario de sus pruebas de usuario basados en las personas definidas.	Hacen mención de los perfiles de usuario sin definiciones de nuevas personas.
PEAA8	Utilizan la plantilla proporcionada en el patrón “Agile persona”, definen una persona y asignan la clasificación correspondiente.	Mencionan 4 usuarios, esto es tres tipos de usuarios más de los iniciales pero no se presenta la definición de los usuarios.	Hacen referencia a tres “Perfiles de usuario” adicionales a la persona definida inicialmente. Dichos perfiles no se describen como personas y tampoco se presenta la definición de las mismas utilizando la plantilla del patrón “Agile Persona”.

Podemos ver que la incorporación de la técnica desde el inicio resultó para los equipos una actividad considerada como parte del proceso de desarrollo y no un paso adicional. Algunos como el PEAA3 consideran en su calendario de actividades aquellas tareas relacionadas con la definición de personas. En la tabla C6-10 tenemos un extracto de su calendario de actividades. En ellas se puede observar filas sombreadas que corresponden a tareas relacionadas con la definición inicial de la persona y la revisión de ésta en base a los datos obtenidos a partir de la entrevista aplicada. Tareas que del equipo PEAA3 que coinciden con aquellas propuestas en la solución de nuestro producto patrón.

Actividad	Fecha	Hora	Entregable
Primera reunión para discutir sobre los métodos de recolección de información	14 de Octubre de 2014	14:00	Decisión sobre que métodos serian convenientes aplicar y cuáles serian los que más información aportarían al desarrollo
Redacción de la entrevista	16 de Octubre de 2014	16:00	La entrevista terminada y lista para ser aplicada
Redacción de la encuesta	16 de Octubre de 2014	19:00	La encuesta termina y lista para ser subida a GoogleDocs
Implementación de la entrevista	18 de Octubre de 2014	15:00	Las respuestas contestadas de la entrevista
Redacción preliminar del modelo persona	21 de Octubre de 2014	16:00	Modelo persona preliminar sin resultados de las entrevistas
Interpretación de los datos recolectados en la entrevista	22 de Octubre de 2014	19:00	La interpretación de los datos obtenidos en la entrevista y
Contrastación de los datos de la entrevista con el modelo persona preliminar	24 de Octubre de 2014	16:00	Resultados de la comparación de lo que creíamos correcto con lo que en realidad sucedió
Implementación de la encuesta	25 de Octubre de 2014	19:00	Encuesta lista para contestar en GoogleDocs y publicada en el foro de la plataforma
Investigación acerca del ERS	27 de Octubre de 2014	19:00	Información acerca de cómo se realiza un ERS y que debe de contener
Modelo persona final	29 de Octubre de 2014	19:00	Modelo persona terminado para la primera entrega y que describa correctamente los datos de la persona entrevistada

Tabla C6–10 Extracto del calendario de actividades del proyecto PEAA3

Los alumnos describen cómo el utilizar la técnica persona ayudó en el proceso de identificar a los usuarios finales de sus respectivos proyectos para poder definir los escenarios adecuados y aplicar la prueba de usabilidad.

El equipo PEAA2 documenta que *“Acotando a los usuarios primarios y secundarios logramos modelar personas que cumplen con los perfiles identificados en el primer análisis. Cada persona expresa a los usuarios regulares de la cafetería de la facultad de matemáticas, que son a quienes está dirigida la aplicación...”*.

“La metodología del análisis consiste en el uso de la herramienta Persona para determinar el tipo de usuarios a quienes va dirigido y cuáles son sus metas, esto nos ayudó al momento de determinar las características finales del proyecto...”

Por su parte el equipo PEAA5 en el documento de su tercera y última entrega:

“...Hemos contemplado un usuario primario que es la persona responsable del centro de adopción, el cual será el administrador de la aplicación. Los Usuarios secundarios son las personas que descarguen la aplicación para hacer uso de cualquiera de sus funcionalidades como usuario secundario...”

Comentarios del equipo PEAA6:

“Para los perfiles diseñamos 3 diferentes personas: Estudiantes, supervisores, conductor...”

Respecto a la relación de utilizar la técnica persona inicialmente para identificar las características de los usuarios finales el equipo PEAA7 comenta:

“Para determinar los perfiles de los usuarios que participaran en la prueba, tomamos en cuenta los siguientes factores:

- Los usuarios deducidos previamente en el Documento de Especificación de Requisitos.
- Que estén considerados dentro de los usuarios primarios de la aplicación”

Lo anterior corrobora los resultados ya estudiados y presentados en la literatura actual sobre la efectividad de la técnica persona para identificar a los usuarios finales.

De acuerdo a la documentación de cada equipo en la tabla C-11 se resume lo observado en relación a la plantilla utilizada para registrar la definición de sus personas. La columna “Equipo” corresponde al identificador de cada equipo y en la columna “Formato utilizado” se indica el formato que utilizó cada equipo y/o las adecuaciones que aplicó al utilizar la plantilla del patrón proporcionado. Las descripciones de las personas identificadas por los equipos en sus respectivos proyectos podemos consultarlas en el anexo C.

Tabla C6–11 Formato utilizado por cada equipo para definir las personas identificadas

Equipo	Formato utilizado
PEAA1	No utilizó la plantilla proporcionada en el patrón.
PEAA2	Utilizó una plantilla diferente a la proporcionada en el patrón
PEAA3	Tomó como base la plantilla proporcionada en el patrón aplicando las siguientes modificaciones: <ul style="list-style-type: none"> – Etiqueta “Modelo Persona de...” para el identificador de la persona descrita en lugar de “Persona”. – No asignó clasificación de la persona definida
PEAA4	Utilizó la plantilla proporcionada en el patrón modificando el nombre del campo para la clasificación de la persona asignando “Persona” en lugar de “Tipo de Persona”
PEAA5	Tomó como base la plantilla proporcionada en el patrón aplicando las siguientes modificaciones: <ul style="list-style-type: none"> – No asignó identificador a la persona definida – Etiqueta “Empleo” en lugar de “Ocupación” – No asignó clasificación de la persona definida
PEAA6	Tomó como base la plantilla proporcionada en el patrón aplicando las siguientes modificaciones: <ul style="list-style-type: none"> – No asignó identificador a la persona definida. – No asignó clasificación de la persona definida.
PEAA7	Utilizó la plantilla proporcionada en el patrón.
PEAA8	Utilizó la plantilla proporcionada en el patrón agregando dos campos: Hora de entrada y hora de salida.

Como mencionamos previamente el objetivo de evaluar el patrón con los grupos académicos es verificar que la información proporcionada en él es lo suficientemente clara. Mientras que parte del objetivo de la materia en la que fue proporcionado a los estudiantes para su uso busca que los estudiantes identifiquen los usuarios finales de sus proyectos de software. Por lo anterior los puntos a utilizar del patrón por los

equipos fueron dos: el procedimiento en el que se explica cómo aplicar la técnica persona y la plantilla proporcionada. De acuerdo a los comentarios generados por los estudiantes éstos aplicaron el procedimiento clasificando las personas y obteniendo los perfiles correspondientes aunque su uso fue opcional para los equipos, como podemos observar la mayoría utilizó la plantilla proporcionada como parte del patrón “Agile persona”. Algunos equipos aplicaron cambios mínimos no afectando el formato original.

Podemos considerar que los resultados a partir de las observaciones y uso que reportan los estudiantes en cuanto a la claridad de la información proporcionada para implementar la técnica los estudiantes cumplieron con el objetivo de su materia identificando los usuarios finales de sus respectivos proyectos y utilizaron la plantilla proporcionada con el patrón “Agile persona” prácticamente sin cambios importantes. Al mismo tiempo se amplió el resultado de nuestra evaluación confirmando la claridad del proceso y la plantilla.

6.3.5 Discusión de resultados en evaluación académica

Tomando en cuenta las observaciones y análisis de la documentación proporcionada por los grupos a los que fue proporcionado el patrón “Agile persona” a continuación realizamos un análisis de la información relevante para nuestra investigación.

Una vez comprendido el objetivo del patrón y la información contenida en éste, los equipos se enfocaron en el campo “Solution” donde se proporciona un diagrama con el proceso a seguir para aplicar la técnica en un contexto ágil. Como parte de la solución también se proporciona una plantilla para la descripción de la persona. Respecto al proceso, nos interesa saber si el patrón “Agile persona” proporciona información necesaria a los equipos de desarrollo para incorporar el uso de la técnica dentro del entorno ágil; en relación al formato nos interesa identificar posibles mejoras de tal manera que la información sea realmente la necesaria para enfocar el diseño a los usuarios finales.

La información de mayor relevancia para nuestra investigación sobre la solución propuesta se esperaría del proyecto piloto PEAP al haber utilizado el patrón durante todo el proceso de un desarrollo ágil. Sin embargo, los datos obtenidos de aquellos equipos pertenecientes a los grupos de evaluación PEAA que utilizaron el patrón únicamente en la etapa inicial del diseño para su proyecto también contribuyeron a ambos aspectos como veremos a continuación.

Proceso

Una primera aportación del equipo del proyecto piloto PEAP fue sobre el nombre y orden de los procesos contemplados durante el desarrollo del sprint y representados en el diagrama solución del patrón. Al inicio del sprint no era necesario considerar un proceso para identificar a los usuarios involucrados en la historia de usuario a ser desarrollada (“Identify users involved in the user story”) ya que resultaba redundante al existir un proceso posterior cuyo objetivo cubría ya la necesidad de información (“Relate persona and user story”). Una vez aplicada la técnica persona durante el desarrollo del sprint, el equipo identificó que era necesario revisar las tareas y criterios de aceptación de las historias de usuario del sprint backlog antes de iniciar con el desarrollo de estas. Para representar lo anterior en nuestra solución se agregó un proceso que indicara estas actividades (“Detailed Definition of the User Story”) y se cambió el nombre del proceso que representa el momento en el que las tareas planeadas son llevadas a cabo (“Sprint Implementation”).

El equipo PEAP inició su proceso de desarrollo, de acuerdo al diagrama solución del patrón, definiendo sus personas y a partir de estas generar el product backlog. Sin embargo, durante el sprint planning se toparon con la necesidad de definir una nueva persona para representar un potencial usuario más que previamente no se había considerado. De esto se deriva que en el diagrama solución haya la posibilidad de que incluso durante el sprint planning sea necesario revisar las descripciones de las personas para tener una mejor definición de las tareas antes de iniciar con el desarrollo del sprint. Siguiendo la misma línea

sobre el proceso algunos de los equipos de PEAA durante la primera entrega documentaron la definición de personas y conforme avanzaron las entregas mencionaron la definición de personas adicionales que fue necesario agregar para representar usuarios finales.

Otro punto relevante es el hecho de que el equipo que desarrolló el proyecto piloto PEAP resalta lo útil que fueron las definiciones de las personas en las validaciones que realizaban al final de cada sprint. Esto es, al tener identificadas ya las descripciones de las personas tenían la referencia de aquellos usuarios finales involucrados con la historia de usuario que se desarrollaba y la evaluación final la realizaban con estos. Para el equipo PEAP después de aplicar la técnica persona en los primeros sprints la definición de nuevas personas y/o modificaciones a aplicar son menos necesarias y si resultó más útil usar las definiciones previas ya revisadas al momento de evaluar el progreso del producto al final de cada sprint. Aunque los equipos de PEAA no puntualizan de la misma forma que el PEAP sí observamos en los documentos de sus avances que el número de personas no incrementa demasiado entre cada informe de avance en el proyecto y las modificaciones aplicadas a las definiciones iniciales también fueron mínimas o incluso nulas.

Formato

En cuanto al formato de la plantilla propuesta como parte de la solución en el patrón, el equipo del proyecto piloto PEAP utilizó el formato sin aplicar adecuación alguna para trabajarlo durante el proceso de desarrollo. Los equipos del grupo PEAA que utilizaron la plantilla podemos concluir que de manera general lo utilizaron sin aplicar mayor adecuación.

6.4 Evaluación en proyectos reales de desarrollo

En esta etapa de evaluación utilizamos como estrategia de investigación la metodología de casos de estudio. Esta técnica implica un estudio en profundidad de una instancia en particular (o pequeño número de instancias) dentro de un contexto real específico. En particular y derivado de las definiciones generales de Runeson et. al. (Runeson, Höst, Rainer, & Regnell, 2012) un caso de estudio en ingeniería de software es una investigación empírica que se basa en múltiples fuentes de evidencia para investigar un caso de un fenómeno de ingeniería de software contemporáneo dentro de su contexto real. Dado nuestro interés en evaluar la eficacia de los productos patrón en desarrollos ágiles reales, un examen minucioso a profundidad de casos individuales que puede ser utilizado para construir entendimiento, generar teorías e hipótesis, presentar evidencia de la existencia de cierto comportamiento (Lazar et al., 2010) nos permitirá entender y obtener una visión más cercana a lo que realmente aportan nuestros patrones propuestos.

Nuestro caso de estudio tiene un propósito exploratorio con el que buscamos conocer, a través del estudio de tres aspectos en particular, la utilidad que representa para los equipos ágiles utilizar productos patrón para incorporar durante el desarrollo de software el uso de técnicas de usabilidad. Los tres aspectos a considerar son: 1) la facilidad/dificultad que tuvieron los equipos ágiles para entender los productos patrón; 2) efectividad de los productos patrón como guía de los equipos ágiles para incorporar las técnicas de usabilidad y 3) las mejoras en usabilidad de las aplicaciones los resultados obtenidos al incorporar las técnicas de usabilidad dentro del proceso de desarrollo ágil. Los datos se recolectaron de manera cualitativa utilizando para ello observación *in situ* de los equipos ágiles, análisis de documentación, entrevistas semi estructuradas a los programadores y la aplicación de una encuesta on-line. Nuestro proceso de investigación tiene un diseño flexible con lo cual los parámetros principales de estudio definidos al inicio se fueron ajustando durante el transcurso del estudio de acuerdo a los descubrimientos que fueron surgiendo conforme avanzó la investigación. La flexibilidad en un caso de estudio se limita en la definición de los objetivos desde el inicio de la investigación, si el objetivo cambia durante el transcurso de la investigación es necesario definir un nuevo caso de estudio (Runeson et al., 2012).

Para nuestro proceso de investigación consideramos: diseño del caso de estudio, preparación de la colección de datos, recolección de la evidencia, análisis de los datos recolectados y reporte del estudio (Runeson et al., 2012).

6.4.1 Diseño del Estudio

Para nuestra investigación consideramos dos contextos: proyectos reales desarrollados en un ambiente universitario y proyectos desarrollados en la industria. En el contexto de ambiente universitario dada la organización y características de los elementos del equipo el proyecto desarrollado en la Facultad de Matemáticas (FMAT) se considera un caso de estudio holístico donde el caso es estudiado como un todo (Runeson et al., 2012). En el contexto de proyectos desarrollados en la industria consideramos la institución pública del Tribunal de Justicia del Estado de Yucatán (TSJYUC) un caso de estudio embebido dónde múltiples unidades de análisis son estudiadas (Runeson et al., 2012). Las unidades de análisis son los proyectos de software desarrollados en cada una de las instituciones mencionadas. En la figura C6-7 observamos las diferentes unidades de análisis para cada caso de estudio en sus diferentes contextos.

Figura C6-7 Contextos, casos y unidades de análisis

A continuación se describen con mayor detalle las unidades de análisis.

6.4.1.1 Caso de estudio 1. Proyecto desarrollado en un ambiente universitario

La Universidad Autónoma de Yucatán cuenta con diferentes proyectos sociales integrales que consideran el desarrollo e implementación de aplicaciones de software para alcanzar el objetivo general. Uno de sus proyectos es “Juega, aprende y diviértete con Linux” programa que apoya en la adopción de tecnologías de información para la enseñanza de educación básica nivel primaria a escuelas de bajos recursos urbanas o rurales. De manera integral el proyecto contempla proporcionar el hardware, apoyo en la instalación de red y software educativo necesario para que los alumnos y profesores de la escuela primaria beneficiada cuente con una infraestructura intranet óptima que les permita utilizar la tecnología para reforzar los temas de enseñanza correspondientes. El proyecto es desarrollado en la Facultad de Matemáticas bajo la supervisión de los profesores responsables del proyecto y con la participación de alumnos que llevan a cabo su prácticum desarrollando el software necesario de acuerdo al programa educativo al que se enfoque el proyecto durante el curso. Los equipos participantes en el proyecto de software adoptaron SCRUM como metodología ágil para el desarrollo de las aplicaciones.

Para aplicar la evaluación a la utilidad de los productos patrón como guía para incorporar técnicas de usabilidad en proyectos reales de desarrollo ágil, seguimos la misma metodología que para la evaluación académica pero considerando las características propias de la organización en donde fueron utilizados. A continuación se describe la forma en la que fueron abordados cada uno de los puntos.

El equipo de desarrollo

Los equipos de desarrollo fueron integrados con alumnos de los diferentes grados del área de informática adscritos al proyecto como parte de su prácticum a desarrollar en el laboratorio de mantenimiento de la

Facultad de Matemáticas de la UADY. La conformación de los equipos fue definida por los profesores responsables del proyecto asignando a cada equipo estudiantes de los grados de informática y un estudiante del grado de Licenciatura en la enseñanza de las matemáticas para desempeñar el rol de experto dentro del proyecto ágil.

Taller

Con el objetivo de presentar a los participantes información general sobre las técnicas de usabilidad involucradas en nuestra investigación y evaluación, le fue proporcionado a todo el equipo de trabajo, incluyendo los profesores involucrados y alumnos con el rol de “experto”, un taller sobre los productos patrón a utilizar incluyendo campos y plantillas necesarias.

Aplicando los productos patrón

Los productos patrón “Agile persona” y “Agile Usability testing” en su versión PAPv2 y PAUTv0 respectivamente junto con las plantillas relacionadas con éstos fueron proporcionados a los equipos ágiles participantes y se mantuvieron disponibles en un espacio on-line para consulta de los equipos en cualquier momento. Los patrones se pueden consultar en el Anexo D.

Seguimiento

La retroalimentación necesaria sobre los resultados de aplicar las guías se obtiene de la siguiente forma:

- Observación
- Revisión de la documentación generada (gestión de los sprints e informes finales de prácticum)
- Entrevistas
- Encuesta on-line

Como observadores se proporcionó apoyo a los equipos durante todo el proceso del desarrollo de software. Es necesario participar en las reuniones propias del proceso ágil como son el product backlog meeting (si las guías podrán ser implementadas dentro de un proyecto completo), sprint planning, daily meetings, etc. Se puso especial atención a participar sin interferir en las actividades del equipo de desarrollo.

Previo calendarización programada junto con los profesores responsables del proyecto y durante el curso del prácticum de los alumnos se aplicaron entrevistas y una encuesta a los elementos del equipo sobre la utilidad de los productos patrón objetos de nuestra investigación.

6.4.1.2 Caso de estudio 2. Proyectos desarrollados en la industria

Los productos patrón fueron usados por los equipos de desarrollo del Departamento de Innovación e implementación de sistemas del Tribunal Superior de Justicia del Estado de Yucatán (TSJYUC), México. El departamento tiene a su cargo el mantenimiento de los sistemas existentes así como el desarrollo de aquellas nuevas funcionalidades que se van requiriendo de acuerdo al crecimiento o cambios en los procesos de la institución. Los equipos de desarrollo utilizan SCRUM como marco de trabajo para el desarrollo de aquellos proyectos de software cuyas características permiten alcanzar los objetivos siguiendo la filosofía ágil. Los productos patrón “Agile Persona” y “Agile Usability Testing” fueron utilizados en un total de seis proyectos, dos proyectos en su proceso de adecuación incorporando nuevas funcionalidades y cuatro nuevos proyectos de software.

- PREI01. Sistema de Administración de Nóminas (mantenimiento)
- PREI02. Sistema de Administración del Fondo Auxiliar (mantenimiento)
- PREI03. Sistema de gestión de Ejecución de Sentencias
- PREI04. Sistema de Armonización Contable
- PREI05. Sistema Informático para Justicia Especializada para Adolescentes
- PREI06. Sistema Informático de Almacén, Compras y Activo Fijo

Cada uno de los proyectos fue desarrollado para determinadas áreas de la institución pública por lo que sus objetivos como herramienta de software fueron diferentes como observamos en la tabla C6-12. La

columna “ID Proyecto” corresponde al identificador con el que se hace referencia durante el estudio al proyecto; “Nombre del proyecto” nombre oficial del proyecto de software asignado por el departamento responsable del desarrollo; “Descripción del proyecto” información general sobre el proyecto incluido el objetivo del sistema.

Tabla C6–12 Descripción de los proyectos en el grupo de evaluación PREI

ID Proyecto	Nombre del proyecto	Descripción del proyecto
PREI01	Sistema Informático de Administración de Nóminas	Tiene por objetivo administrar el proceso de emisión de nóminas para los empleados del Tribunal Judicial del Estado de Yucatán, México. Se desarrolló con metodología SCRUM el módulo de becarios y asimilables a salario utilizando una plataforma cliente-servidor en el lenguaje Visual .NET con un equipo de tres desarrolladores, un tester y el product owner.
PREI02	Sistema de Administración del Fondo Auxiliar	Sistema cuyo objetivo es funcionar como herramienta automatizada para el control de los fondos ingresados y egresados como resguardos al poder judicial del estado. El sistema se desarrolla para plataforma cliente – servidor en el lenguaje Visual Basic por un equipo conformado por 4 desarrolladores, 1 tester y el product owner.
PREI03	Sistema Informático de Gestión de Ejecución de Sentencias	El objetivo del sistema es proporcionar una herramienta que permita la administración y gestión automatizada del proceso de ejecución de sentencia de cada uno de los expedientes que se registran en los juzgados de Ejecución de acuerdo a los procedimientos de justicia correspondientes. Agilizando de esta forma el seguimiento de los procesos penales asignados para los expedientes aperturados. El sistema se desarrolla para plataforma web utilizando el lenguaje ASP .Net por un equipo conformado por 4 desarrolladores, 1 tester y el product owner.
PREI04	Sistema de Armonización Contable	Sistema en plataforma cliente – servidor cuyo objetivo es gestionar y controlar de una manera más eficiente y eficaz la administración del presupuesto contable del Tribunal superior de justicia mediante la automatización e implementación de los procedimientos indicados en el manual contable vigente. El equipo está conformado por tres desarrolladores, un tester y el producto owner. El software se desarrolla utilizando como lenguaje de programación C Sharp.
PREI05	Sistema Informático para Justicia Especializada para Adolescentes	El módulo desarrollado siguiendo metodología ágil tiene por objetivo proporcionar un control y administración automatizado de las salas disponibles en los juzgados penales para los procesos que ahí se llevan a cabo diariamente. Equipo de desarrollo ágil conformado por tres desarrolladores, un tester y product owner. El software se desarrolló para una plataforma WEB utilizando como lenguaje de programación visual .NET.
PREI06	Sistema Informático de Almacén, Compras y Activo Fijo	El sistema tiene por objetivo ofrecer al usuario una herramienta automatizada para agilizar el proceso de adquisición, resguardo y entrega de material al personal del Poder Judicial. Las funcionalidades implementadas en principio fueron el registro de solicitudes, envío de solicitudes, seguimiento y entrega de material solicitado y actualización del inventario. El equipo está conformado por tres desarrolladores, un tester y el product owner. El software se desarrolla para una plataforma web utilizando como lenguaje de programación PHP.

Para llevar a cabo la evaluación con los equipos de desarrollo de la industria seguimos la misma metodología que para la evaluación académica pero considerando las características propias de la organización.

El equipo de desarrollo

Los equipos de desarrollo fueron integrados con elementos asignados por los responsables del departamento de innovación e implementación de sistemas encargados de esta tarea. Cada uno de los equipos estuvo conformado en promedio de seis elementos: tres desarrolladores (Development team), Scrum Master, tester y el product owner.

Taller

Con el objetivo de presentar a los participantes información general sobre las técnicas de usabilidad involucradas en la evaluación se proporcionó a todo el departamento de innovación e implementación de sistemas un taller de usabilidad. En los temas del taller se incluyeron los productos patrón..

Aplicando los productos patrón

Las guías se proporcionaron a los equipos ágiles participantes. Los productos patrón “Agile persona” y “Agile Usability Teting” en sus versiones PAPv2 y PAUTv0 respectivamente así como las plantillas relacionadas con estos se mantuvieron disponibles en un espacio on-line para consulta de los equipos en cualquier momento. Las guías pueden ser aplicadas por uno o más equipos ágiles al mismo tiempo ya sea en un mismo proyecto o en diferentes. Los patrones evaluados se pueden consultar en el Anexo D.

Seguimiento

La retroalimentación necesaria sobre los resultados de aplicar las guías se obtuvo utilizando los siguientes métodos:

- Observación
- Revisión de documentación relacionada con la gestión de los sprints
- Entrevistas
- Encuesta on-line

Previo calendarización programada junto con el responsable del área informática de la compañía se aplicaron entrevistas y encuestas al scrum master y los desarrolladores (development team).

6.4.2 Preparación para la Recolección de Datos

Los datos fueron recolectados a través de observación, revisión y análisis de la documentación generada durante los sprints, una entrevista estructurada aplicada de manera individual a los elementos del development team de cada unidad de análisis y una encuesta publicada en línea.

Para el proceso de observación se acordó con los responsables de las áreas correspondientes participar en las reuniones de los equipos ágiles al inicio (sprint planning) y final de cada iteración (sprint review y retrospective) obteniendo información sobre la experiencia y los resultados que cada equipo iba alcanzando al incorporar las técnicas persona y pruebas de usabilidad durante el desarrollo ágil. Con el objetivo de mantener disponible para consulta y actualización toda la documentación generada durante el avance del proyecto los equipos utilizaron un almacenamiento on-line en Drive de Google. Como parte de los documentos se almacenaron en archivos de formato Excel los registros del product backlog y los sprint backlogs con la información detallada de las tareas y el correspondiente avance de las mismas así como los documentos generados a partir de la aplicación de los productos patrón en donde los equipos hacían uso de plantillas.

Previo programación con los desarrolladores del development team se realizó una entrevista estructurada con cada uno de sus miembros buscando obtener del desarrollador entrevistado respuestas más amplias que nos proporcionaran más información y permitieran estudiar mejor los resultados relacionados con los aspectos a evaluar en nuestra investigación.

Las preguntas abiertas realizadas durante la entrevista estructurada agrupadas de acuerdo a los aspectos a evaluar son las siguientes:

A1. Facilidad/Dificultad de utilizar los patrones

- P.A1.1 Fue complicado entender la estructura del patrón “Agile persona”?
- P.A1.2 Fue complicado entender la estructura del patrón “Agile Usability Testing”?
- P.A1.3 Cuáles fueron los aspectos más positivos del patrón “Agile Persona”?
- P.A1.4 Cuáles fueron los aspectos más positivos del patrón “Agile Usability Testing”?

- P.A1.5 Cuáles fueron los aspectos más negativos del patrón “Agile Persona”?
- P.A1.6 Cuáles fueron los aspectos más negativos del patrón “Agile Usability Testing”?
- P.A1.7 Sugerencias para mejorar la utilidad del patrón “Agile Persona” como guía para implementar la técnica persona durante el proceso de desarrollo ágil
- P.A1.8 Sugerencias para mejorar la utilidad del patrón “Agile Usability Testing” como guía para implementar las pruebas de usabilidad durante el proceso de desarrollo ágil

A2. Efectividad de los patrones como guía para incorporar las técnicas de usabilidad en el proceso ágil

- P.A2.1. La información proporcionada en el documento del patrón facilitó la aplicación de la técnica “Persona” durante el proceso de desarrollo ágil?
- P.A2.2. Utilizar el patrón “Agile Persona” durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones?
- P.A2.3. Utilizar la solución propuesta en el patrón “Agile Persona” (Campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales?
- P.A2.4. La información proporcionada en el documento del patrón facilitó la aplicación de la técnica usability testing durante el proceso de desarrollo ágil?
- P.A2.5. Utilizar el patrón “Agile Usability Testing” durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones?
- P.A2.6. La solución propuesta en el patrón “Agile usability testing” (campo solution) facilitó aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil de software?

A3. Resultado de utilizar los patrones

- P.A3.1. Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios?
- P.A3.2. Durante las iteraciones hubo cambios en las funcionalidades debido a la mala definición de los perfiles de los usuarios?
- P.A3.3. Durante las iteraciones hubo cambios en las funcionalidades debido a la mala interpretación de los requerimientos del usuario?
- P.A3.4. En base a tu experiencia previa en desarrollo de software, las aplicaciones desarrolladas utilizando las técnicas de usabilidad en el proceso de desarrollo ágil fueron más fáciles de utilizar por los usuarios
- P.A3.5. Utilizarías de nuevo el patrón “Agile Persona” durante el desarrollo de software con metodologías ágiles?
- P.A3.6. Utilizarías de nuevo el patrón “Agile Usability Testing” durante el desarrollo de software con metodologías ágiles?

La información obtenida a partir de aplicar la entrevista descrita anteriormente se complementó con los resultados obtenidos a partir de una encuesta elaborada y publicada en línea utilizando la herramienta de software surveymonkey (www.surveymonkey.com). A través de una escala de Lickert los desarrolladores evaluaron diferentes preguntas relacionadas con nuestro objeto de estudio que nos permitieron capturar de manera rápida una “big picture” de su interacción con los patrones “Agile persona” y “Agile usability Testing”. Los resultados de la evaluación proporcionaron información sobre cómo percibieron la utilidad y facilidad de uso de patrones, los problemas a los que se enfrentaron al utilizarlos y cuáles fueron las acciones que tomaron durante el uso de ellos. Utilizar la técnica de la encuesta en línea resultó adecuado para los equipos ágiles debido a que les resultó más sencillo responder la encuesta en algún momento disponible sin presión de tiempo y/o espacio que interfiriera con sus diferentes actividades laborales. En la figura C6-8 se muestra la primera parte de la encuesta referente a la evaluación sobre la facilidad de uso del patrón “Agile Persona” y la escala de Lickert con la que fueron evaluadas cada una de las preguntas.. En el anexo F.2 se muestra el detalle de todos los cuestionarios utilizados.

1. Nombre del proyecto de software en el que utilizaste los patrones

* 2. Patrón "**Agile persona**"

	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en desacuerdo
Fué complicado entender la estructura del patrón "Agile Persona"	☐	☐	☐	☐	☐
La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil?	☐	☐	☐	☐	☐
Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	☐	☐	☐	☐	☐
Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales	☐	☐	☐	☐	☐
Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles	☐	☐	☐	☐	☐

* 3. Lista los aspectos más positivos del patrón "Agile persona"

a)

b)

c)

* 4. Lista los aspectos más negativos del patrón "Agile Persona"

a)

b)

c)

5. Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Sig.

Figura C6-8 Encuesta Facilidad/Dificultad de utilizar los patrones

Antes de realizar las entrevistas y publicar la encuesta en línea se realizaron pruebas piloto con el objetivo de verificar que las preguntas a considerar tanto en la encuesta y como en la entrevista eran las correctas y una vez aplicadas aportaban datos valiosos a nuestra investigación. Una de las pruebas fue aplicada solamente con los desarrolladores de los proyectos desarrollados en Industria y la segunda prueba fue aplicada con desarrolladores tanto del proyecto en ambiente universitario como de la industria. Como apoyo se definieron instrumentos de entrevista que contiene una serie de pasos que sirven como guía para obtener la información durante la misma. En el anexoE podemos consultar los instrumentos de entrevista aplicados en cada una de las pruebas piloto así como los registros de cada una de estas. Las tablas E-1 y E-2 de dicho anexo presentan información sobre las entrevistas aplicadas como el grupo al que pertenece el entrevistado; el ID del entrevistado; la fecha de aplicación (y hora de aplicación); tiempo de duración; y el medio de almacenamiento en donde fue resguardada cada una de las entrevistas. Como resultado de las pruebas piloto aplicadas se realizaron las mejoras necesarias resultando la definición de las preguntas finales a considerar como preguntas en la entrevista estructurada y como puntos en la encuesta.

6.4.3 Recolección de Datos

Durante el desarrollo de los sprints los equipos consideraron como parte de sus tareas aquellas destinadas a trabajar con las técnicas de usabilidad por lo que, en el seguimiento de gestión generado para cada sprint los equipos llevaron el registro de dichas tareas y los tiempos estimados para realizarlas. La documentación generada al utilizar los productos patrón y los artefactos propuestos en éstos fue almacenada en el espacio cloud seleccionado por los equipos. Al final del período establecido para el uso de los patrones se aplicó la entrevista de manera individual a cada uno de los elementos de los equipos para obtener información sobre sus percepciones al utilizar los productos patrón. Los registros de las entrevistas se almacenaron en audio y/o video para mantener resguardada la evidencia de los resultados.

La encuesta fue publicada en línea durante el mismo período en el que se realizó la entrevista a los desarrolladores. Para tener un mejor control sobre los momentos de respuesta a la encuesta se envió invitación personalizada por correo electrónico a cada desarrollador y el seguimiento a sus respuestas fue a través del mismo software con el que fue generada la encuesta.

6.4.3.1 Caso de Estudio 1. Software desarrollado en Ambiente Universitario.

Los programas institucionales para la educación primaria en México están organizados por bloques que abarcan diversas áreas de acuerdo al objetivo de enseñanza de cada uno de ellos. El software educativo a desarrollar para el proyecto “Juega, aprende y diviértete con Linux” durante el período de evaluación de los productos patrón considera bloques de temas diseñados para niños de cuarto grado de educación primaria contemplando dos juegos interactivos en cada bloque por medio de los cuáles los niños refuerzan sus conocimientos sobre las siguientes áreas: a) Operaciones básicas, b) Números decimales y fracciones y c) Recta numérica. Los estudiantes que se registraron para participar en el programa y llevar a cabo su prácticum fueron organizados en tres equipos con cuatro estudiantes cada uno para desarrollar el software didáctico correspondiente a cada una de las áreas mencionadas.

Los equipos participantes utilizaron SCRUM como marco de trabajo para el desarrollo ágil del software. Cada uno de los roles requeridos en la metodología se cubrió de la siguiente forma: el Development Team fue conformado por todos los elementos de cada equipo uno de ellos desempeñando adicionalmente el rol de Scrum Master; el rol de product Owner fue cubierto por el profesor responsable del proyecto apoyado por un stakeholder; rol representado por un estudiante del área de matemáticas y también elemento del equipo. El software fue diseñado y desarrollado para su uso en plataforma web, los equipos utilizaron para la generación de los juegos el lenguaje HTML5, JavaScript y el framework Phaser. La tabla C6-13 muestra información sobre los temas abarcados por cada equipo en cada juego desarrollado. La columna “Equipo” indica el identificador asignado a cada equipo ágil de estudiantes; “Área” corresponde al bloque de temas del programa educativo sobre el cuál trabajó el equipo correspondiente; “Juegos desarrollados / Temas abarcados” temas abarcados por cada juego como software educativo; y la columna “Iteraciones” corresponde al número total de sprints de desarrollo que realizó el equipo.

Tabla C6–13 Juegos desarrollados por los equipos

Equipo	Área	Juegos Desarrollados / Temas abarcados	Iteraciones
E1	Bloque 1	Juego 1 <ul style="list-style-type: none"> • Jerarquía de operaciones • Uso de signos de agrupación • Sumas y restas Juego 2 <ul style="list-style-type: none"> • Conversión de unidades • Expresiones con punto decimal • Fracciones decimales 	4
E2	Bloque 2	Juego 1 <ul style="list-style-type: none"> • Identificación y ubicación de elementos en la recta de números reales Juego 2 <ul style="list-style-type: none"> • Representación geométrica con fracciones 	4
E3	Bloque 3	Juego 1 <ul style="list-style-type: none"> • Representaciones Gráficas de Fracciones • Equivalencia de Fracciones • Suma y Resta de Fracciones Juego 2 <ul style="list-style-type: none"> • Fracciones equivalentes • Conversión de unidades 	4

En total participaron en el proyecto de software ocho desarrolladores y tres product owner. La especificación sobre la participación de cada uno de ellos la tenemos en la tabla C6-14 donde, “ID elemento” corresponde al identificador asignado a cada elemento del development team; “Rol” identifica

el rol desempeñado por el elemento; y en la columna “Equipo” se observa el identificador del equipo en el que participó cada elemento.

Tabla C6–14 Participación de cada desarrollador en el grupo de evaluación PEAU

ID Elemento	Rol	Equipo
PRAU_E01	Programador 1	E1
PRAU_E02	Programador 2	E1
PRAU_E03	Product Owner	E1
PRAU_E04	Programador 4	E1
PRAU_E05	Programador 5	E3
PRAU_E06	Programador 6	E3
PRAU_E07	Programador 7	E3
PRAU_E08	Product Owner	E3
PRAU_E09	Programador 9	E2
PRAU_E10	Product Owner	E2
PRAU_E11	Programador 11	E2

El proyecto se desarrolló por los equipos durante el período Octubre 2015 – Febrero 2016. Al inicio los equipos trabajaron de manera independiente, sin embargo a partir del segundo sprint se establecieron días fijos en los cuales todos los equipos debían trabajar en el espacio físico destinado para tal efecto dentro de las instalaciones de la facultad. Previo al inicio de los sprints de desarrollo se les proporcionó a los equipos un taller de usabilidad con el objetivo de presentarles e involucrarles con los conceptos que estarían trabajando al aplicar los productos patrón de “Agile Persona” y “Agile Usability Testing”. Tomando en cuenta los artefactos mencionados en la sección previa para la preparación de la recolección de datos se definieron una serie de actividades, mostradas en la tabla C6-15, como parte del proceso para evaluar la efectividad de los productos patrón durante el desarrollo del proyecto de software educativo. Estuvieron involucrados en la validación los cuatro equipos desarrollando sus correspondientes sprints en paralelo durante un mismo periodo de tiempo. En la columna “Actividad” de la tabla C6-15 podemos consultar la descripción de las actividades llevadas al cabo, las columnas “Inicio” y “Fin” indican respectivamente las fechas en las que iniciaron y finalizaron las actividades.

Tabla C6–15 Actividades durante la evaluación en el proyecto de software educativo

Actividad	Inicio	Fin
Taller sobre las guías y conceptos generales de usabilidad	19 Oct 2015	21 Oct 2015
Desarrollo del proyecto	Oct 2015	Feb 2016
Entrevista	12 Feb 2016	23 Feb 2016
Publicación en línea de encuesta	10 Feb 2016	10 Mar 2016

Como parte de la recolección de datos durante el período de evaluación uno de los investigadores participó en todos los sprints como observador en las reuniones del sprint planning, sprint review y sprint retrospective indicadas como parte de la metodología SCRUM; también se proporcionó apoyo al equipo ágil durante el desarrollo de los sprints con respecto a dudas sobre la estructura o información contenida en los patrones y artefactos propuestos como parte de la solución para implementar las técnicas de usabilidad persona y usability testing durante el desarrollo del software. El objetivo de la participación del investigador fue el de verificar en cada sprint la manera en la que los equipos iban trabajando la incorporación de las técnicas de usabilidad durante desarrollo, desde el inicio con el sprint planning considerando como parte de sus tareas aquellas necesarias para aplicar alguna o ambas técnicas de usabilidad, utilizando los patrones hasta fin del sprint con la retroalimentación de los resultados en el sprint review y retrospective.

Los tiempos de desarrollo de los sprints así como todas las actividades se observan en la tabla C6-16. En la columna “Iteraciones” se listan los sprints desarrollados por los equipos a los que corresponde la

información; “Planning” indica la fecha en la que se llevó a cabo la reunión del sprint planning para definir los objetivos del sprint y verificar la priorización de las historias de usuario a desarrollar; las columnas “Inicio” y “Fin” corresponden a las fechas de inicio y fin respectivamente en las que se desarrolló el sprint; “Review” fecha en la que se realizó la reunión sprint review; y “Retrospective” fecha en la que se realizó la reunión del retrospective para el sprint correspondiente.

Tabla C6–16 Actividades realizadas durante los sprints

Iteraciones	Planning	Inicio	Fin	Review	Retrospective
Sprint 1		19 Oct 2015	19 Oct 2015		
Sprint 2	20 Oct 2015	21 Oct 2015	04 Nov 2015	05 Nov 2015	06 Nov 2015
Sprint 3	09 Nov 2015	10 Nov 2015	24 Nov 2015	25 Nov 2015	26 Nov 2015
Sprint 4	26 Nov 2015	27 Nov 2015	11 Dic 2015	14 Dic 2015	14 Dic 2015

Al finalizar el último release los desarrolladores y product owners fueron entrevistados de manera individual previa programación de acuerdo a la disponibilidad de estos. En la tabla C6-17 se observa información relacionada con la aplicación de la entrevista descrita previamente como artefacto para la recolección de datos, la columna “ID Desarrollador” corresponde al identificador asignado al desarrollador en el proyecto de investigación; “Fecha entrevista/Hora inicio” indica la fecha en la que se realizó y la hora de inicio de la entrevista; “Duración entrevista” tiempo que duró la entrevista; y “Registro de entrevista” espacio cloud de almacenamiento y nombre del archivo del video de la entrevista.

Tabla C6–17 Registro de las entrevistas realizadas a los equipos del proyecto PRAU

ID Desarrollador	Fecha entrevista/Hora inicio	Duración Entrevista	Registro de entrevista
PRAU_E01	10 Feb 2016 / 17:00	09:09 mins	Dropbox/Interview_PEAU_E01
PRAU_E02	10 Feb 2016 / 17:30	09:23 mins	Dropbox/Interview_PEAU_E02
PRAU_E03	10 Feb 2016 / 18:00	13:36 mins	Dropbox/Interview_PEAU_E03
PRAU_E04	10 Feb 2016 / 18:30	12:08 mins	Dropbox/Interview_PEAU_E04
PRAU_E05	11 Feb 2016 / 19:30	30 mins	Dropbox/Interview_PEAU_E05
PRAU_E06	11 Feb 2016 / 20:30	20:31 mins	Dropbox/Interview_PEAU_E06
PRAU_E07	11 Feb 2016 / 20:00	31:44 mins	Dropbox/Interview_PEAU_E07
PRAU_E08	02 Mar 2016 / 17:00	27:47 mins	Dropbox/Interview_PEAU_E08
PRAU_E09	12 Feb 2016 / 18:00	10:08 mins	Dropbox/Interview_PEAU_E09
PRAU_E10	12 Feb 2016 / 18:30	17:19 mins	Dropbox/Interview_PEAU_E10
PRAU_E11	13 Abr 2016 / 17:30	17:37 mins	Dropbox/Interview_PEAU_E11

Por otro lado la encuesta definida para complementar la información sobre la percepción de los desarrolladores en relación a los productos patrón evaluados fue publicada en línea y se envió invitación a cada uno de los elementos del development team vía correo electrónico. Los datos del seguimiento a las respuestas por cada uno de los desarrolladores se muestran en la tabla C6-18. La columna “ID Desarrollador” corresponde al identificador asignado al desarrollador durante el proyecto de investigación; “Fecha invitación encuesta” indica la fecha en la que fue enviado el correo electrónico de invitación para responder la encuesta; “Fecha respuesta encuesta” indica la fecha en la que el desarrollador dio respuesta a la encuesta en línea; por último la columna “Tiempo destinado a la encuesta” es el dato del tiempo registrado que dedicó el desarrollador a responder la encuesta. Cabe mencionar que el tiempo de respuesta a la encuesta en línea inicia una vez abierta y finaliza cuando el encuestado da por terminada con el botón “Listo”. En particular observamos registros con más de una hora de respuesta cuya razón de tiempo está relacionada con lo mencionado anteriormente.

Tabla C6–18 Seguimiento a las respuestas de la encuesta en línea para el proyecto PRAU

ID Desarrollador	Fecha Invitación Encuesta	Fecha respuesta Encuesta	Tiempo destinado a la encuesta
PRAU_E01	10-feb-16	10-feb-16	0:26:09
PRAU_E02	10-feb-16	10-feb-16	1:27:48
PRAU_E03	17-feb-16	18-feb-16	5:22:14
PRAU_E04	10-feb-16	10-feb-16	0:09:37
PRAU_E05	11-feb-16	12-feb-16	0:13:11
PRAU_E06	11-feb-16	11-feb-16	0:25:11
PRAU_E07	11-feb-16	11-feb-16	0:11:57
PRAU_E09	14-feb-16	14-feb-16	0:18:51
PRAU_E10	12-feb-16	12-feb-16	0:16:02

Adicionalmente cada equipo elaboró un documento final de su prácticum reportando información sobre todo el desarrollo del proyecto de software incluyendo los resultados de las técnicas de usabilidad implementadas, este documento se almacenó en el cloud Drive de Google junto con toda la documentación generada por los equipos relacionada al proyecto.

6.4.3.2 Caso de Estudio 2. Software desarrollado en la Industria

Los productos patrón fueron usados en el período Septiembre 2015 – Febrero 2016 en el Departamento de Innovación e implementación de sistemas del TSJYUC para el desarrollo de los seis proyectos descritos previamente como nuestras unidades de análisis. Antes de iniciar con el uso del patrón se realizaron una serie de actividades con todo el personal que participaría que utilizarían la herramienta para incorporar las técnicas de persona y usability Testing. Las actividades se pueden consultar en la tabla C6-19 donde la columna “Actividad” corresponde a la descripción de las actividades llevadas al cabo; las columnas “Inicio” y “Fin” indican respectivamente las fechas en las que iniciaron y finalizaron dichas actividades.

Tabla C6–19 Actividades durante la evaluación de los productos patrón en la Industria

Actividad	Inicio	Fin
Taller sobre las guías y conceptos generales de usabilidad	17 Ago 2015	21 Ago 2015
Desarrollo de proyecto(s)	31 Ago 2015	16 Feb 2016
Entrevistas	10 Feb 2016	23 Feb 2016
Publicación en línea de encuesta	10 Feb 2016	10 Mar 2016

Cada equipo de acuerdo a la metodología SCRUM estuvo constituido por el development team, scrum master y product owner, adicionalmente por cuestiones de organización previa del área informática en el equipo también participó un elemento en el rol de tester.

En la tabla C6-20 tenemos una visión general sobre cada uno de los proyectos que fueron evaluados. La columna ID Proyecto corresponde al identificador asignado a cada proyecto para efectos de nuestra investigación; “Nombre/Descripción” indica el nombre del proyecto y las siglas de éstos en paréntesis. La columna “Iteraciones” indica el número de iteraciones que se desarrollaron en cada proyecto.

Tabla C6–20 Proyectos de Software desarrollados en la Industria (Unidades de Análisis)

ID Proyecto	Nombre / Descripción	Iteraciones
PREI01	Sistema de Administración de Nóminas (SIANOM)	2
PREI02	Sistema de Administración de Fondo Auxiliar (SIAFA)	2
PREI03	Sistema de Gestión de Ejecución de Sentencias (SIGES)	4
PREI04	Sistema de Armonización Contable (ARCON)	4
PREI05	Sistema Informático para Justicia Especializada para Adolescentes (SIJEA)	2
PREI06	Sistema Informático de Almacén, Compras y Activo Fijo (SIACA)	3

En total participaron en los proyectos de software 12 desarrolladores, tres testers y seis product owner. La especificación sobre cada uno la tenemos en la tabla C6-21 donde “ID elemento” corresponde al identificador asignado a cada elemento del development team; “Rol” identifica el rol desempeñado por el elemento y en la columna “Proyectos” se observan los identificadores de los proyectos en los que participó cada elemento. Durante el proceso de desarrollo en algunos proyectos los programadores fueron movidos de un proyecto a otro debido a que finalizaban las iteraciones necesarias para alcanzar el objetivo y su apoyo se aprovechaba en aquellos proyectos en los que aún faltaban iteraciones por desarrollar.

Tabla C6–21 Proyectos en los que participó cada desarrollador

ID Elemento	Rol	Proyectos
PREI_E01	Programador 1	PREI01, PREI05, PREI06
PREI_E02	Programador 2	PREI01, PREI02, PREI05, PREI06
PREI_E03	Programador 3	PREI02, PREI03
PREI_E04	Programador 4	PREI01, PREI05, PREI06
PREI_E05	Programador 5	PREI03
PREI_E06	Programador 6	PREI03
PREI_E07	Programador 7	PREI03
PREI_E08	Programador 8	PREI04
PREI_E09	Programador 9	PREI04
PREI_E10	Programador 10	PREI04
PREI_E11	Programador 11	PREI02, PREI04
PREI_E12	Programador 12	PREI02, PREI06
PREI_E13	Tester 1	PREI03
PREI_E14	Tester 2	PREI01, PREI06
PREI_E15	Tester 3	PREI03
PREI_E16	Product Owner	PREI01
PREI_E17	Product Owner	PREI02
PREI_E18	Product Owner	PREI03
PREI_E19	Product Owner	PREI04
PREI_E20	Product Owner	PREI05
PREI_E21	Product Owner	PREI06

Uno de los investigadores participó como observador durante las reuniones iniciales (sprint planning) y las reuniones finales de cada sprint (sprint review y retrospective). Durante estas reuniones se realizaron apuntes sobre la información que exponían los equipos con particular atención a la experiencia que reportaban en relación al uso de los productos patrón. Como resultado del sprint planning los equipos definían las tareas a realizar durante la iteración que iniciaba generando el documento de seguimiento que se actualizaba de acuerdo al avance que los responsables tenían de éstas.

La tabla C6-22 nos muestra una visión general de los proyectos respecto a los tiempos y equipos de desarrollo en cada sprint; la columna “ID Proyecto” corresponde al identificador asignado al proyecto (el nombre completo del sistema lo podemos consultar en la tabla C6-16). “Sprint” es el número de sprint desarrollado; “Inicio” indica la fecha en la que inició el sprint correspondiente; “Fin” la fecha de término del sprint; “Sprint Review” la fecha en la que se llevó a cabo la reunión de revisión del sprint; “Sprint Retrospective” la fecha en la que se llevó a cabo la reunión de retrospectiva del equipo.

En la columna “Development Team” podemos consultar los identificadores de cada elemento que conformó el equipo de desarrollo incluyendo testers y product owner.

Tabla C6–22 Visión general de los sprints y elementos de cada proyecto

ID Proyecto	Sprint	Inicio	Fin	Sprint Review	Sprint Retrospective	Development Team
PREI01	1	31 Ago 2015	18 sept 2015	21 sept 2015	22 sept 2015	PREI_E01
	2	02 Oct 2015	23 Oct 2015	26 Oct 2015	26 Oct 2015	PREI_E02 PREI_E04 PREI_E14 PREI_E16
PREI02	1	14 Sep 2015	30 Sep 2015	06 Oct 2015	06 Oct 2015	PREI_E11 PREI_E12 PREI_E17
	2	29 Oct 2015	27 Nov 2015			PREI_E11 PREI_E12 PREI_E02 PREI_E03 PREI_E17
PREI03	1	31Ago 2015	11 sept 2015	15 sept 2015	15 sept 2015	PREI_E03
	2	07 Oct 2015	30 Oct 2015	03 Nov 2015	04 Nov 2015	PREI_E05
	3	30 Nov 2015	11 Dic 2015	16 Dic 2015	16 Dic 2015	PREI_E06
	4	15 Ene 2016	16 Feb 2016	17 Feb 2016	18 Feb 2015	PREI_E07 PREI_E13 PREI_E15 PREI_E18
PREI04	1	02 Sept 2015	15 Sept 2015	17 sept 2015	18 sept 2015	PREI_E08
	2	02 Oct 2015	03 Nov 2015	05 Nov 2015	06 Nov 2015	PREI_E09
	3	03 Dic 2015	16 Dic 2015			PREI_E10
	4	29 Ene 2016	15 Feb 2016			PREI_E19
PREI05	1	01 Dic 2015	11 dic 2015	14 Dic 2015	14 Dic 2015	PREI_E01 PREI_E02 PREI_E20
	2	13 Ene 2016	05 Feb 2016	12 Feb 2016	12 Feb 2016	PREI_E01 PREI_E02 PREI_E04 PREI_E20
PREI06	0	15 Dic 2015	13 Ene 2016	14 Ene 2016	14 Ene 2016	PREI_E08 PREI_E12 PREI_E04 PREI_E14 PREI_E21
	1	22 Ene 2016	24 Feb 2016	25 Feb 2016	25 Feb 2016	PREI_E08 PREI_E12 PREI_E04 PREI_E14 PREI_E21
	2	29 Feb 2016	11 Mar 2016	12 Mar 2016	12 Mar 2016	PREI_E08 PREI_E12 PREI_E04 PREI_E02 PREI_E14 PREI_E21

Las entrevistas fueron aplicadas a todos los desarrolladores con una duración promedio de 30 minutos, los resultados fueron resguardados en archivos de video y/o audio para su posterior análisis. Los testers y product owners no fueron entrevistados debido a su disponibilidad.

6.4.4 Análisis de datos

En las siguientes secciones analizamos los datos recolectados durante el desarrollo de los proyectos de software explorando la forma en la que los equipos ágiles consideran dentro de sus sprints aquellas tareas necesarias para implementar las técnicas de usabilidad durante el desarrollo del software utilizando los productos patrón y la manera en la que fueron solventando situaciones no contempladas durante su uso. Siguiendo con la línea de nuestra investigación nos enfocamos en los aspectos de facilidad/dificultad que tuvieron los desarrolladores para entender los productos patrón al implementar las técnicas de usabilidad durante el desarrollo del proyecto y la efectividad de éstos al ser utilizados como una guía para incorporar las dos técnicas de usabilidad. También se analiza la percepción sobre los resultados obtenidos identificando mejoras sugeridas en base a la experiencia de los desarrolladores trabajando con los productos patrón. Para efectos de nuestro análisis en general son consideradas las evaluaciones aplicadas por todos los elementos de los equipos. Sin embargo, en relación con los totales condensados y graficados únicamente se consideran las evaluaciones realizadas por los desarrolladores. Tomando en cuenta de esta forma que en el grupo de evaluación PREI únicamente fueron entrevistados y encuestados los desarrolladores y asegurando así que la percepción analizada es de los miembros del mismo tipo de rol. A continuación se describe el análisis de cada uno de los aspectos evaluados.

6.4.4.1 A1. Facilidad/Dificultad de utilizar los patrones

Para efectos de nuestro estudio las preguntas P.A1.1 y P.A1.2 en inicio dan pauta para identificar si los desarrolladores tuvieron algún grado de dificultad al utilizar los productos patrón, la evaluación de este aspecto se complementa con el análisis de aquellos puntos positivos, puntos negativos y sugerencias para cada uno de los productos patrón que los desarrolladores observaron y externaron tanto en sus respuestas a la encuesta como durante la entrevista. Extractos de las transcripciones de las entrevistas aplicadas a los desarrolladores se encuentran en el anexo G.2 y H.2 del presente documento.

En la tabla C6-23 se presenta un condensado de las frecuencias absolutas de las valoraciones aplicadas por los desarrolladores en la encuesta a las preguntas P.A1.1 y P.A1.2. La columna “ID” corresponde al identificador asignado durante nuestro estudio a la pregunta, “Pregunta” es la descripción de la pregunta, posteriormente en las columnas “Completamente de acuerdo”, “Muy de acuerdo”, “De acuerdo”, “En desacuerdo” y “Completamente en Desacuerdo” se indican las frecuencias observadas para cada una de las escalas valoradas. Para visualizar mejor las tendencias la gráfica de la figura C6-9 muestra la comparativa de los resultados globales de estas valoraciones y en donde observamos que en general la mayoría de los desarrolladores estuvo “En desacuerdo” con la pregunta de que haya sido complicado entender la estructura de los productos patrón “Agile persona” y “Agile usability testing”. Sin embargo, se observan valoraciones para las escalas de estar “De acuerdo”, “Muy de acuerdo” y “Completamente de acuerdo”.

Tabla C6-23. Condensado de los totales de las validaciones aplicadas en la encuesta a P.A1.1 y P.A1.2

ID	Pregunta	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en Desacuerdo
P.A1.1	Fue complicado entender la estructura del patrón "Agile persona"	0	3	2	10	6
P.A1.2	Fue complicado entender la estructura del patrón "Agile Usability Testing"	1	1	1	11	7

Figura C6-9 Distribución de la evaluación a la Facilidad/Dificultad de utilizar los patrones

La comparativa de opiniones para cada uno de los grupos de evaluación PRAU y PREI en relación a estas mismas preguntas P.A1.1 y P.A1.2 se muestra en las figuras C6-10 y C6-11 respectivamente. El comportamiento de las valoraciones aplicadas por cada grupo refleja que las validaciones de las escalas “De Acuerdo”, “Muy de acuerdo” y “Completamente de acuerdo” resultan principalmente por parte de los desarrolladores del grupo de evaluación PRAU. En dicho grupo los desarrolladores fueron estudiantes. Por el lado del grupo PREI se observa también valoraciones a la escala “Muy de acuerdo” para el patrón “Agile Persona”.

Figura C6-10 Distribución resultado de la evaluación a la Facilidad/Dificultad de utilizar los patrones. Grupo PRAU

Figura C6-11 Distribución resultado de la evaluación a la Facilidad/Dificultad de utilizar los patrones. Grupo PREI

A continuación analizamos más de cerca la información de acuerdo a las tendencias que muestran los resultados sobre la facilidad/dificultad para cada uno de los patrones propuestos.

6.4.4.1.1 P.A1.1. ¿Fue complicado entender la estructura del patrón “Agile Persona”?

De acuerdo a la distribución de las valoraciones que se muestra en la figura C6-12 la mayoría de los desarrolladores, reflejado por un 77%, no percibió complicado entender el patrón “Agile persona” valorando las escalas “En desacuerdo” (48%) y “Completamente en desacuerdo” (29%). Un 23% valoró las escalas “Muy de acuerdo” y “De acuerdo” lo cual indica que para este porcentaje de los desarrolladores si representó cierto grado de dificultad utilizarlo.

Figura C6-12 Distribución global de los resultados a la evaluación de los desarrolladores a la pregunta P.A1.1

Las distribuciones de valoración por cada grupo PRAU y PREI presentadas en la figura C6-13 incisos a y b respectivamente reflejan que la valoración “Muy de acuerdo” fue aplicada por desarrolladores en ambos grupos en tanto que la valoración “De acuerdo” solamente la aplicó el grupo de evaluación PRAU. Por otro lado, la evaluación del grupo PREI de los desarrollos en industria refleja que entender el patrón “Agile persona” no les resultó muy complicado al aplicar un 92% de sus valoraciones a las escalas “Completamente en desacuerdo” y “En desacuerdo”. La valoración que se refleja a la escala “Muy de acuerdo” corresponde a uno de los desarrolladores sin embargo el porcentaje de 8% que representa es importante.

Figura C6-13 Distribuciones de evaluaciones a la pregunta P.A1.1 de cada uno de los grupos de evaluación

Con el objetivo de identificar los factores que generaron dificultad a los desarrolladores para entender el patrón “Agile persona” analizamos las entrevistas de aquellos que percibieron complicación. La respuesta directa a la pregunta no aportó información relevante que indicara la razón por la que les fue complicado entender el patrón e incluso en la mayoría de los casos fue contradictoria a la valoración en la encuesta. Sin embargo, durante la entrevista los mismos desarrolladores hicieron comentarios relacionados que analizaremos más adelante.

Muy de acuerdo *“No estuvo tan complicado, básicamente era seguir la plantilla...para mí estuvo algo corto el contenido con el que describe el patrón...”* PRAU_E07

“Al inicio sí...” PRAU_E09

“No...” PREI_E11

De acuerdo *“No tanto porque te decía qué es lo que significaba cada campo entonces no fue muy complicado...”* PRAU_E06

“No, no lo fue...” PRAU_E11

En el caso del desarrollador PRAU_E09 indicó que al inicio se le hizo complicado entender el patrón y durante la entrevista en relación a otras preguntas externó que el patrón lo usaría de nuevo... *“...ya que le entendí sí (lo usaría) porque es más fácil...”* por lo que podemos deducir que la dificultad percibida al principio se debió a la falta de experiencia en el uso del patrón al comenzar a incorporarlo en su proceso de desarrollo. De manera similar durante la entrevista el desarrollador PRAU_E11 comentó que *“...estaba un poco confuso de entender el diagrama...”* lo que podemos interpretar como falta de claridad en la descripción del diagrama presentado en la solución del patrón.

6.4.4.1.2 P.A1.2 ¿Fue complicado entender la estructura del patrón “Agile Usability Testing”?

Continuando con el análisis de los resultados, en la figura C6-14 tenemos la distribución global de las valoraciones al punto P.A1.2 que corresponde a la facilidad/dificultad de entender el patrón “Agile usability testing”. En este caso un 85% del total de los desarrolladores no percibió complicación para comprender el patrón y un 15% sí.

Figura C6-14 Distribución global de los resultados a la evaluación de los desarrolladores a la pregunta P.A1.2

Vale la pena resaltar el comportamiento que se observa en el resultado de las evaluaciones de ambos grupos. Resulta interesante la diferencia en cuanto a las valoraciones asignadas por el grupo PRAU conformado por estudiantes y el grupo PREI conformado por desarrolladores profesionales. En la figura C6-15 incisos a y b tenemos las distribuciones de las valoraciones por grupo PRAU y PREI respectivamente.

a) Grupo de evaluación PRAU

b) Grupo de evaluación PREI

Figura C6-15 Distribuciones de evaluación a la pregunta P.A1.2 de cada uno de los grupos de evaluación

Para el grupo PREI definitivamente no significó complicación alguna entender el patrón no así para el grupo de validación PRAU del que un 37% de sus desarrolladores valoró las escalas “Completamente de acuerdo” (12%), “Muy de acuerdo” (12%) y “De acuerdo” (13%). Análogamente al caso de facilidad/dificultad de entender el patrón “persona” recurrimos a las entrevistas para obtener información que nos permitiera identificar factores que implicaron complicaciones para los desarrolladores al momento de comprender el patrón “usability testing”. Como podemos observar en los extractos de las entrevistas la respuesta a la pregunta específica de nuevo no proporciona información relevante y en algunos casos contradictoria para identificar porqué los desarrolladores percibieron complicación.

Completamente de acuerdo “Para mí no fue complicado...” PRAU_E07

Muy de acuerdo “No...fue más fácil...” PRAU_E09

De acuerdo “No, pero se me hizo más fácil el de persona...” PRAU_E11

Únicamente el desarrollador PRAU_E11 en su respuesta a otra pregunta durante la entrevista comentó “...el diagrama que estaba en el documento no lo sentía muy entendible...tuve que leerlo bastante hasta que logré entenderlo...”. De lo anterior podría considerarse la complicación que tuvo el desarrollador debido a la falta de claridad en el diagrama de la solución del patrón.

6.4.4.1.3 Aspectos positivos (P.A1.3) y aspectos negativos (P.A1.5) del patrón “Agile persona”

Las preguntas P.A1.3 y P.A1.5 proporcionaron información sobre los aspectos positivos y negativos respectivamente que los desarrolladores percibieron al utilizar el patrón “Agile persona”. De acuerdo a las respuestas de los desarrolladores en la encuesta y durante la entrevista de los desarrolladores, identificamos como más relevantes los que se presentan en las tablas C6-24 y C6-25. La columna “Característica” corresponde al aspecto relevante y “Comentarios (extractos de las entrevistas)” son extractos tomados de las entrevistas aplicadas a cada uno de los desarrolladores.

Tabla C6-24 Aspectos positivos patrón “Agile Persona”

Característica	Comentarios (extractos de las entrevistas)
Apoyo del patrón como guía	<p>“...nos dió el alcance del proyecto...para quién va dirigido y cómo los objetivos...es como una guía para que no nos salgamos del camino...” PRAU_E06</p> <p>“...a partir de ello uno ya tiene la idea de a qué tipo de usuario final va a desarrollar su software o para quién va a ir...” PRAU_E05</p> <p>“...facilitaba a manera de plantilla con seguir el proceso de qué dato debemos poner y no estar atormentándonos o analizar ‘este dato va o este</p>

	<p>dato no va...”...PRAU_E11</p> <p>“...como ya es parte de la información que se necesita...sólo tienes que imaginar a la persona y llenar esos campos con esa plantilla que te dan entonces ya es más rápido realizar la persona según sus características...” PRAU_E04</p> <p>“Tener de primera mano la información...nos ayudó con la definición de tiempos y de tareas, no directamente pero teníamos la noción al haber levantado la información...” PREI_E01</p> <p>“Te explica bastante bien cómo aplicar el proceso...te da una idea de cómo debes hacer tus formatos para definir las personas...” PREI_E03</p> <p>“Te da una guía para ver de qué manera puedes obtener información de los usuarios...” PREI_E10</p> <p>“...me hizo fácil considerar ciertas características...” PRAU_E09</p>
Solución propuesta en el patrón	<p>“Lo del diagrama me parece que es positivo...muy bien explicado...” PRAU_E08</p> <p>“La parte del diagrama me gustó mucho...te permite ver en qué fase lo tienes que hacer y de una manera visual te va mostrando en qué momento debes aplicar ese patrón...” PREI_E08</p>
Estructura del patrón	<p>“...la estructura que tiene el patrón...” PRAU_E10</p> <p>“...la estructura te permite hacer de una manera más fácil el desarrollo de todo el patrón...” PRAU_E03</p> <p>“Primero la facilidad de entenderlo, segundo la organización...lo que facilitó el entendimiento...” PRAU_E02</p> <p>“...el hecho de que lo tienes estructurado...” PREI_E11</p>
Ejemplo / Plantilla	<p>“...la parte donde puedes describir con palabras más o menos cómo es el perfil del usuario eso me gustó bastante porque lo describes de una manera natural”...PRAU_E11</p> <p>“Una de las partes más positivas es el ejemplo...si estás leyendo y se te complica un poco el hecho de entender a qué se está refiriendo cada cosa ya te vas al ejemplo y se queda claro lo que está tratando de decir el documento...”PREI_E08</p>

Tabla C6-25 Aspectos negativos patrón “Agile Persona”

Característica	Comentarios (extractos de las entrevistas)
Tiempo	<p>“A veces...como es nuevo para nosotros surgían dudas y teníamos que consultar ya sea la plantilla o de manera grupal, entonces esa parte como que sí nos quitaba un poco de tiempo...requiere un poco de tiempo, también para madurar el concepto...” PRAU_E06</p> <p>“...si no se planifica o no se sabe con precisión como se usa esto a final de cuentas puede ser muy tardado...” PRAU_E07</p> <p>“...el tiempo que no había considerado...” PRAU_E09</p> <p>“...el tiempo que lleva...” PRAU_E10</p>

	<p><i>“...toma un tiempo...” PREI_E04</i></p> <p><i>“...nos lleva un tiempo más grande que definir nuestros permisos, a veces estamos muy apretados con tiempo...a veces habíamos comido el tiempo para hacer el patrón persona en otras actividades...” PREI_E09</i></p>
Descripción de los campos del patrón	<p><i>“...explicar un poco más...sobre los objetivos, los porqués de cada cosa...” PRAU_E08</i></p> <p><i>“...la parte de solución debí poner más cuidado al leer principalmente por los diagramas...la parte de collaboration y capability label no sé si porque en esa parte no había ejemplo pero esa parte si fue la que realmente no comprendí...” PRAU_E02</i></p> <p><i>“...en el momento que estábamos definiendo a nuestra persona el proceso decía: el equipo de desarrollo hace su diseño de persona luego lo ve con su product owner...si siente (el product owner) que le falta algo tenemos que rediseñar (a la persona)...algo que no me gustó fue eso...me gustaría que desde el principio nuestro product owner esté con nosotros y nos esté diciendo sabes qué, esto no...para no tener que rediseñar otra vez...” PRAU_E05</i></p> <p><i>“...no le puedes poner ni la foto de la persona ni el rol con su nombre...desde mi punto de vista pienso que sí se puede cambiar, no perjudicaría y sería más claro...en la plantilla...” PREI_E11</i></p>

Como podemos observar en los extractos de las respuestas durante la entrevista de los desarrolladores, en particular a la pregunta sobre los aspectos negativos, alguna de las respuestas contienen a su vez sugerencias que pueden considerarse para mejorar la utilidad del patrón “Agile persona”. Como veremos más adelante lo mismo sucede con esta pregunta al ser cuestionados los desarrolladores sobre aspectos negativos del patrón “Agile usability testing”.

6.4.4.1.4 P.A1.7. Sugerencias para mejorar la utilidad del patrón “Agile persona”

Las respuestas a las preguntas definidas específicamente para obtener sugerencias por parte de los desarrolladores que contribuyan en mejorar la utilidad de los productos patrón propuestos también aportan datos importantes. En el caso de “Agile persona” la pregunta correspondiente es P.A1.7 de la cual se derivaron sugerencias de mejora relacionadas principalmente con la solución propuesta en el patrón, la plantilla, más ejemplos y la descripción de algunos puntos:

“...poner la foto de la persona como que está un poco de más, yo leyendo la descripción yo puedo saber quiénes, para quién va dirigido...esa parte siento que está un poco de más...” PRAU_E06

“...entre todo el equipo nos pusiéramos de acuerdo para ver lo que está ahí contenido, diagrama y todos los detalles del patrón...desglosarlo entre tareas o subprocesos...” PRAU_E07

“...otro ejemplo...” PRAU_E03

“...las dos últimas secciones que mencioné (collaboration y capability label) quizá sea conveniente revisarlas y si aplican el ejemplo que se ponga alguna información para que se entienda esa parte...” PRAU_E02

“...especificar más los roles...describir un poco qué características tiene esas personas...” PRAU_E04

“...no le puedes poner ni la foto de la persona ni el rol con su nombre...desde mi punto de vista pienso que sí se puede cambiar, no perjudicaría y sería más claro...en la plantilla...” PRAU_E11

“...dentro de las entrevistas que se van haciendo durante el patrón hacer ciertas visitas más cercanas a la parte operativa, más que sólo con el product owner o con los stakeholders sino que nosotros como gente de desarrollo ir y sensibilizarnos del procedimiento manual...” PRAU_E12

A partir de los comentarios de dos de los desarrolladores, durante sus respectivas entrevistas, se identificó que algunos equipos utilizaron artefactos de apoyo para definir sus personas. Uno de los equipos definió un pequeño cuestionario con algunas preguntas que le ayudó a obtener la información mínima para las primeras definiciones de sus personas. Por otro lado un segundo equipo considerando que era de inicio relativamente complicado para los usuarios establecer un tiempo para conceder una entrevista utilizó la misma plantilla proporcionada en el patrón para recabar información. El equipo imprimió el formato y lo entregó a sus usuarios para que la respondieran, aunque no les indicaron una fecha específica para entregarlo con las respuestas los usuarios enviaron de vuelta con la información necesaria al día siguiente de haberlo recibido en papel. La pronta respuesta resultó mejor de lo que esperaban los desarrolladores. El cuestionario utilizado por el primer equipo lo tenemos en el anexo H.3.

Extractos de sus comentarios los tenemos a continuación:

“...sólo fueron unas cuatro o cinco preguntas básicas de actividades que realiza, el nivel de puesto que tiene, cuáles son sus responsabilidades y cómo le gusta trabajar...en base a que fue llenando un parrafito por cada pregunta hicimos patrón...” PREI_E01

“...a nosotros sí se nos hizo buena idea lo de aplicarlo como cuestionario cuando son muchos usuarios...un sistema de uno o dos usuarios no hay mayor problema que tú lo entrevistes y te explayas pero cuando tienes muchos usuarios sobre todo en el caso de algunos sistemas que sus usuarios están en otro edificio es complicado poder coincidir con ellos...mandándolos (los cuestionarios) por correo o por estafeta...eso no te quitaría tiempo y tampoco les quitarías tiempo a ellos...” PREI_E10

6.4.4.1.5 Aspectos positivos (P.A1.4) y aspectos negativos (P.A1.6) del patrón “Agile Usability Testing”

P.A1.4 y P.A1.6 son las preguntas que se definieron para obtener información de los desarrolladores sobre características positivas y negativas percibidas al utilizar el patrón “Agile Usability testing”. De manera análoga al análisis descrito previamente para el patrón “Agile persona” en base a las respuestas en la encuesta y en la entrevista se identificaron aspectos que resultaron más relevantes. En las tablas C6-26 y C6-27 se presentan los positivos y negativos respectivamente; la columna “Característica” corresponde al aspecto relevante; y “Comentarios (extractos de las entrevistas)” son extractos tomados de las entrevistas.

Tabla C6-26 Aspectos positivos patrón “Agile Usability Testing”

Característica	Comentarios (extractos de las entrevistas)
Apoyo del patrón como guía	<p><i>“...era tener la información lo más rápido...una vez que tenía lo que va a hacer el usuario era capturar la información de lo que observabas...”</i> PRAU_E07</p> <p><i>“...fácil de entenderlo...me dio la solución más específica...”</i> PRAU_E09</p> <p><i>“...a pesar de que necesita una persona especializada para hacer las pruebas porque no cualquiera puede hacer las pruebas, es muy fácil gracias al patrón aprender cómo hacer esas pruebas...eso me parece una ventaja...hay una parte que dice que la persona que hace la prueba debe presentar los resultados, en nuestro caso el equipo de desarrollo fue el que hizo las pruebas y fuimos capaces de aplicar el patrón a pesar de eso...eso me parece que es</i></p>

	<p><i>una ventaja... ”...PRAU_E01</i></p> <p><i>“como ya está planeado te ayuda a automatizar más tu prueba...el formato igual para el llenado es bastante entendible y rápido...” PRAU_E04</i></p> <p><i>“Nos lleva de la mano, es muy sencillo, muy estructurado, organizado...” PRAU_E02</i></p> <p><i>“Es bastante claro...están bien definidas las tareas que se tienen que hacer...” PREI_E06</i></p> <p><i>“...te pone de manera concisa todo lo que tienes que considerar...como un pequeño manual que te permite saber específicamente qué es lo que tienes que hacer...es como un pequeño manualito para utilizar...” PREI_E08</i></p> <p><i>“Es una guía de cómo podíamos aplicar la técnica...te da una pauta de qué hacer...cómo aplicarlo...una guía...” PREI_E10</i></p>
Solución propuesta en el patrón	<p><i>“...lo que me gustó es que te pedían grabar audio o video y después analizar éste...” PRAU_E05</i></p> <p><i>“...los diagramas...te lo explica...” PRAU_E08</i></p> <p><i>“...el campo Solution porque te va indicando los momentos que va a seguir el proceso de usabilidad...” PRAU_E03</i></p> <p><i>“...si aplicas usabilidad al inicio o intermedia y tienes considerado el tiempo para reparar esos errores te ahorras una vuelta...un sprint al menos...” PREI_E01</i></p> <p><i>“...nuestro prototipo nos ayuda bastante...nosotros lo hicimos en papel...” PREI_E07</i></p> <p><i>“...el diagrama...me gustaron los dos diagramas...es muy entendible el diagrama” PREI_E08</i></p> <p><i>“...cuando usamos el testing con el software a través del video que generamos pudimos darnos cuenta de un error...al ver el video vimos que fue lo que movió el usuario para generar el error y con eso pudimos corregir...mejoró nuestros diseños...nos ayudó en la comunicación...” PREI_E09</i></p> <p><i>“Nos permitió probar los prototipos con diferentes personas, no solo con el product owner...también lo positivo fue que participó la gente de servicios (otro departamento) entonces nos adelantó sus observaciones porque son fundamentales...también lo que nos ayuda mucho es que nos permite identificar lo que es un Bug o una deficiencia del producto a terminar o un requerimiento nuevo...me ayudó a identificar y como que a negociar con el product owner...” PREI_E12</i></p>
Estructura del patrón	<p><i>“...la estructura...” PRAU_E10</i></p> <p><i>“...cómo se organiza la información y cómo este te ayuda a entenderlo mejor y por lo tanto invertir menos tiempo en entenderlo...” PRAU_E02</i></p> <p><i>“...orden...”PREI_E04</i></p>
Ejemplo / Plantilla	<p><i>“...los formatos que se proporcionan en él para llevar un registro de cómo realizar las pruebas de usabilidad ayudan bastante...” PREI_E03</i></p> <p><i>“...el checklist...el review list...” PREI_E05</i></p>

Tabla C6-27 Aspectos negativos producto patrón “Agile Usability Testing”

Punto negativo	Comentarios (extractos de las entrevistas)
Tiempo	<p>“...lo único negativo es que lleva tiempo el preparar como que el escenario...pero creo que eso no se puede evitar, entonces no creo tener nada negativo...” PRAU_E05</p> <p>“...el tiempo...al hacer la prueba...” PRAU_E10</p> <p>“...si no considera el tiempo adecuado (para las tareas relacionadas con la técnica) puede ser algo perjudicial...” PRAU_E09</p> <p>“...tiempo...” PRAU_E03</p> <p>“Es un tiempo...porque si lleva un tiempo hacer la prueba...cómo definir, saber con certeza que esto es un bug o el otro tipo...te va a al llevar tiempo hacer esa definición...es un tiempo más...” PREI_E02</p>
Descripción de los campos del patrón	<p>“...me quedo nuevamente con las partes de ‘Collaboration’ y ‘Capability label...’” PRAU_E02</p> <p>“Los diagramas...a veces está muy compactado...no se entienden muy bien las letritas que están ahí...está muy pequeño...” PRAU_E06</p> <p>“...la explicación que diste hace un rato no la encontré por ningún lado...” PREI_E03</p>

Los aspectos positivos y negativos del patrón “Agile Usability Testing” percibidos se adaptaron a la clasificación utilizada para estos mismos aspectos pero relacionados con el patrón “Agile persona”. No surgió alguna clasificación diferente derivada de las opiniones de los desarrolladores.

6.4.4.1.6 P.A1.8 Sugerencias para mejorar la utilidad del patrón “Agile Usability Testing”

Por otro lado, las sugerencias de mejora para el patrón “Agile Usability Testing” se obtuvieron a través de la pregunta P.A1.8 de las que derivaron sugerencias relacionadas con la solución propuesta en el patrón, la plantilla, ejemplos y la descripción de algunos puntos para que sean más claros. Los extractos de las entrevistas corresponden a los desarrolladores que proporcionaron sugerencia, los demás desarrolladores dijeron no tener ninguna:

“Los diagramas...que tengan cortes de cada actividad como fluye el proceso...que tenga cortes para que se entienda un poco mejor porque si esta todo ahí como que confunde un poco...” PRAU_E06

“...los detalles ponerlo en otro documento...” PRAU_E07

“...tiene espacios del lado izquierdo donde está cada apartado, siento que ocupa espacio...en cuanto a estructura a la izquierda hay muchos espacios debajo de cada apartado...” PRAU_E08

“Más ejemplos en la sección de ‘Examples’...” PRAU_E09

“...como en el anterior (“Agile persona”), revisar esas dos secciones “Collaboration” y “Capability label”...ya sea revisar si son útiles y en el caso de que si lo sean proporcionar información al respecto para que se entiendan mejor...” PRAU_E02

“...la descripción de los términos...tal vez poner una pequeña descripción...si lo ve una persona que nunca ha tomado el curso como nos lo diste siento que sí es un poquito más difícil de entender...” PREI_02

“...si pudiera haber un apartado en donde se aclare esa parte de las pruebas de usabilidad que son opcionales y que están al final estaría mucho mejor...” PREI_E03

“...solamente un puntito...no encontré en el diagrama un proceso como el que se nos presentó en una ocasión que solamente hacemos la prueba de usabilidad...dentro de tu diagrama veo que hablas de prueba de usabilidad pero hay que pasar a una parte de desarrollo... teníamos historias de usuario en base a las historias de usuario hicimos la prueba de usabilidad pero fue nuestro único producto entregable porque al final nos dedicamos a la investigación del tema...” – PREI_E08

En el caso de aquellos desarrolladores que indicaron no tener sugerencia para mejora, algunos dentro de la respuesta brevemente comentaron aspectos positivos como: “...está muy entendible y muy práctico” PREI_E09, “...está bastante completo” PREI_E11 y “...sencillo y rápido...” PREI_E12.

6.4.4.2 A2. Efectividad de los patrones como guía para incorporar las técnicas de usabilidad en el proceso ágil

Para evaluar la efectividad de los productos patrón como guías de apoyo a los desarrolladores ágiles para incorporar las técnicas persona y usability testing dentro del proceso de desarrollo ágil, en esta sección analizamos cada uno de las preguntas relacionadas con este aspecto. Extractos de las transcripciones de las entrevistas aplicadas a los desarrolladores se encuentran en el Anexo G.2 y H.2 del presente documento.

En la tabla C6-28 se presenta un condensado global de las valoraciones aplicadas en la encuesta por los desarrolladores. La columna “ID” corresponde al identificador asignado durante nuestro estudio a la pregunta aplicada; la columna “Pregunta” contiene la descripción de dicha pregunta; posteriormente en las columnas “Completamente de acuerdo”, “Muy de acuerdo”, “De acuerdo”, “En desacuerdo” y “Completamente en Desacuerdo” se indican las frecuencias observadas para cada una de las escalas valoradas en las preguntas correspondientes.

Tabla C6–28 Totales condensados preguntas aspecto A2

ID	Pregunta	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en Desacuerdo
P.A2.1	La información proporcionada en el documento del patrón "Agile persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil	7	11	3	0	0
P.A2.2	Utilizar el patrón "Agile Persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	1	5	4	9	2
P.A2.3	Utilizar la solución propuesta en el patrón "Agile persona" (Campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales	9	7	4	0	1
P.A2.4	La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil	9	11	1	0	0

P.A2.5	Utilizar el patrón "Agile Usability Testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	0	3	3	7	8
P.A2.6	La solución propuesta en el patrón "Agile Usability Testing" (Campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software	11	9	0	1	0

Las gráficas de las figuras C6-16 y C6-17 representan los resultados globales de la evaluación a las preguntas del aspecto A2 al producto patrón “Agile Persona” y “Agile Usability Testing” respectivamente.

Figura C6-16 Resultados globales de la evaluación sobre la efectividad del patrón “Agile Persona” como guía para implementar la técnica de pruebas de usabilidad durante el desarrollo del proyecto

Figura C6-17 Resultados globales de la evaluación sobre la efectividad del patrón “Agile Usability Testing” como guía para implementar la técnica de pruebas de usabilidad durante el desarrollo del proyecto

De forma general observamos que para ambas técnicas el utilizar la información contenida en los documentos de los productos patrón facilitó la aplicación de éstas. Los desarrolladores indican estar “De acuerdo”, “Muy de acuerdo” y “Completamente de acuerdo” para ambos casos, tanto con la información

del patrón “Agile persona” como del patrón “Agile usability Testing”. Sin embargo, para el caso de la pregunta sobre el atraso generado al implementar las técnicas tenemos resultados en donde algunos de los desarrolladores estuvieron “De acuerdo” con el hecho de que incorporar las técnicas generó atraso en los tiempos de las iteraciones. Incluso observamos opiniones en donde indican estar “Muy de acuerdo” en que incorporar las técnicas generó atraso en el proceso de desarrollo ágil (para el caso del patrón “Agile Persona” y “Agile usability Testing”) y “Completamente de acuerdo” (para el patrón “Agile Persona”). La mayoría de los desarrolladores valoró estar “De acuerdo”, “Muy de acuerdo” o “Completamente de acuerdo” en que la solución propuesta en los productos patrón tanto de “Agile persona” como de “Agile usability Testing” facilita aplicar estas técnicas durante el proceso de desarrollo; únicamente uno de ellos indicó estar completamente en desacuerdo.

Para efectos de visualizar mejor los resultados de la evaluación para cada uno de los productos patrón hemos graficado los datos por técnica de usabilidad y por grupo de evaluación. En las tablas C6-29 y C6-30 presentamos los totales condensados de los grupos de evaluación PRAU y PREI respectivamente, recabados a partir de las respuestas de los desarrolladores a las preguntas de la encuesta relacionados con el aspecto evaluado A2. Las columnas son las mismas que describimos previamente para la tabla C6-28.

Tabla C6-29 Totales condensados preguntas aspecto A2 grupo PRAU

ID	Pregunta	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en Desacuerdo
P.A2.1	La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil	2	5	1	0	0
P.A2.2	Utilizar el patrón "Agile Persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	0	4	1	3	0
P.A2.3	Utilizar la solución propuesta en el patrón "Agile persona" (Campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales	3	3	2	0	0
P.A2.4	La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil	2	5	1	0	0
P.A2.6	La solución propuesta en el patrón "Agile Usability Testing" (Campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software	4	3	0	1	0
P.A2.5	Utilizar el patrón "Agile Usability Testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	0	2	2	3	1

Tabla C6-30 Totales condensados preguntas aspecto A2 grupo PREI

ID	Pregunta	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en Desacuerdo
P.A2.1	La información proporcionada en el documento del patrón "Agile persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil	5	6	2	0	0
P.A2.2	Utilizar el patrón "Agile Persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	1	1	3	6	2
P.A2.3	Utilizar la solución propuesta en el patrón "Agile persona" (Campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales	6	4	2	0	1
P.A2.4	La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil	7	6	0	0	0
P.A2.6	La solución propuesta en el patrón "Agile Usability Testing" (Campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software	7	6	0	0	0
P.A2.5	Utilizar el patrón "Agile Usability Testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	0	1	1	4	7

A continuación enfocamos nuestro análisis más específico sobre cada una de las preguntas relacionadas con la evaluación del aspecto A2 de los productos patrón “Agile Persona” y “Agile Usability Testing”.

6.4.4.2.1 P.A2.1. La información proporcionada en el documento del patrón facilitó la aplicación de la técnica persona durante el proceso de desarrollo ágil.

Como podemos observar en la gráfica de la figura C6-18 la evaluación global muestra que en el 53% de las encuestas los desarrolladores reportan estar “Muy de acuerdo” con el hecho de que la información proporcionada en el documento del patrón “Agile persona” les facilitó aplicar la técnica persona durante el proceso de desarrollo ágil. Las evaluaciones fueron positivas con un 33% completamente de acuerdo y un 14% de acuerdo con el reactivo evaluado, ninguna de las encuestas reportó desarrolladores “En desacuerdo” o “Completamente en desacuerdo”.

Analizando los resultados por cada uno de nuestros casos de estudio observamos que un mayor porcentaje de los desarrolladores que opinaron estar “Completamente de Acuerdo” con el hecho de que la información del patrón les facilitó la aplicación de la técnica persona corresponde al grupo de evaluación de proyectos desarrollados en la industria PREI. El mayor porcentaje para el caso de los desarrolladores que reflejaron estar “Muy de Acuerdo” corresponde al grupo PRAU, recordemos que el número de desarrolladores en cada equipo varía entre un proyecto y otro sin embargo para el caso del valor “Muy de acuerdo” observamos en las tablas C6-29 y C6-30 que la diferencia entre porcentajes se incrementa considerablemente debido a la valoración de un solo desarrollador.

P.A2.1 - Global **Figura C6-18** Distribución global de los porcentajes obtenidos como resultado a la evaluación de los desarrolladores a la pregunta P.A2.1 correspondiente al aspecto de Efectividad de los productos patrón

Independientemente del número de desarrolladores en cada equipo, un porcentaje de evaluación similar se observa para aquellos que indicaron estar “De acuerdo” en que la información del patrón facilitó aplicar la técnica Persona con un 13% del grupo PRAU y un 15% del grupo PREI. La figura C6-19 muestra las distribuciones de los casos de estudio PRAU y PREI en las gráficas de los incisos a) y b) respectivamente.

Figura C6-19 Resultados de evaluación a la pregunta P.A2.1 de cada uno de los grupos de evaluación

Dentro de las características que los desarrolladores subrayan del patrón al facilitarles la aplicación de la técnica persona están la información contenida en el mismo y la estructura de organización con la que se presenta.

“...me pareció que la organización era bastante buena lo que ayudó con la comprensión y por lo tanto reducir el tiempo de leerlo para entenderlo...” – PRAU_E02

“...estaba cada punto específicamente qué es lo que había que hacer...era bastante específico...” – PRAU_E05

“...te decía para qué sirve y cómo debes enfocarte en tus objetivos y los campos te describían cómo hacer ese patrón y para qué sirve, entonces sí te da una idea...” - PRAU_E06

“...en algunos aspectos sí lo facilitó...ayudó en algunos casos a resolver dudas...” – PRAU_E09

6.4.4.2.2 P.A2.2. Utilizar el patrón “Agile Persona” durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones.

En general el porcentaje de evaluación de esta pregunta indica que los desarrolladores en su mayoría no percibieron atraso en sus tiempos de iteración al utilizar el patrón respaldando esta conclusión con un 43% de ellos “En desacuerdo” y un 9% “Completamente en desacuerdo”. Sin embargo es de llamar la atención el porcentaje que reflejan las otras tres escalas de evaluación. Un 5% de los desarrolladores

(representado por un solo desarrollador en términos de totales) estuvo “Completamente de acuerdo” en que utilizar durante el desarrollo ágil el patrón “Agile Persona” generó atraso, otro 24% estuvo “Muy de acuerdo” y 19% estuvo “De acuerdo”. Los totales se observan en la figura C6-20.

Figura C6-20 Distribución global de los porcentajes obtenidos como resultado a la evaluación de los desarrolladores a la pregunta P.A2.2

Para analizar mejor la evaluación de los desarrolladores en la figura C6-21 incisos a) y b) tenemos la distribución de los resultados de cada uno de los grupos PRAU y PREI respectivamente.

a) Grupo de evaluación PRAU

b) Grupo de evaluación PREI

Figura C6-21 Resultados de evaluación a la pregunta P.A2.2 de cada uno de los grupos de evaluación

Resulta evidente que los desarrolladores del grupo PRAU con un 50% de evaluación para la escala “Muy de acuerdo” y un 12% para la escala “De Acuerdo” fueron los que más percibieron un atraso en sus tiempos programados al utilizar el patrón durante el proceso de desarrollo. En el grupo PREI la escala “Completamente de acuerdo” obtuvo un 8% del porcentaje de evaluación total generado por la valoración de un desarrollador. Sin embargo incluso considerando la suma de las tres escalas “Completamente de acuerdo” con un 8%, “Muy de acuerdo” un 8% y un 23% “De acuerdo” se tiene un porcentaje que resulta menor (39%) comparándolo con la evaluación del grupo PRAU (62%). Los desarrolladores principalmente refieren que al comenzar a utilizar el patrón les resultó complicado en inicio, otro factor que resaltan es el hecho de que considerar las tareas necesarias minimiza el impacto en tiempos al incorporar la técnica.

Con el objetivo de identificar factores por los que los desarrolladores percibieron atraso analizamos las entrevistas en particular de aquellos que externaron que utilizar el patrón generó atraso en los tiempos programados de las iteraciones. Extractos de las entrevistas los tenemos a continuación de acuerdo a la escala de valoración asignada:

Completamente de acuerdo	<i>“No...de alguna manera en nuestro caso solamente fue un rol y las actividades que realizaba son actividades orientadas a un solo rol...como fue solamente uno no impacto de manera directa los tiempos...” PREI_E05</i>
Muy de acuerdo	<p><i>“...un poco porque una persona se centraba en realizar toda esa documentación pero a veces siempre habían dudas y teníamos que consultarlo entre los tres y llegábamos a un acuerdo, entonces un poco, no mucho pero sí un poco...” – PRAU_E06</i></p> <p><i>“...fue un poco atraso mientras estábamos aprendiendo a aplicar el patrón porque no sabíamos exactamente cómo aplicarlo de inmediato...conforme avanzamos siento que ya nos empezamos a acostumbrar poco a poco pero tampoco fue tan continuo...” – PRAU_E07</i></p> <p><i>“...no, de hecho creo que el no usarlos generaría los atrasos...” PRAU_E05</i></p> <p><i>“No...” PRAU_E11</i></p> <p><i>“Un poco... al principio...costó un poco de trabajo comprenderlo y aplicarlo...” PREI_E04</i></p>
De acuerdo	<p><i>“...sí nos afectó en algo en lo que son los tiempos porque no teníamos considerado ciertas cosas y cuando comenzamos a utilizar el patrón...fue poco (el tiempo de atraso) pero sí nos atrasó...no sacamos tiempo para las tareas (relacionadas con la incorporación de la técnica)...” – PRAU_E09</i><i>“Sí es cierto que nos lleva un tiempo...si separas el tiempo como tal no creo que sea realmente un atraso...considero que si lo haces bien y lo consideras bien en tus tareas no es un atraso...” PREI_E02</i></p> <p><i>“...un poquito porque no lo habíamos entendido...pero ya entendiéndolo bien no creo que genere atrasos en los siguientes proyectos...” PREI_E03</i></p> <p><i>“Sí...realmente no lo considerábamos tan importante entonces digamos que no lo teníamos tan en cuenta y una vez que se incluyó pues hubo que atrasar un poquito otras cosas...una vez que lo teníamos ya en cuenta realmente no nos impactaba...” PREI_E11</i></p>

Causan confusión respuestas que contradicen por completo la valoración asignada a la escala. Estas son las mínimas por lo que la asignación a la valoración puede atribuirse a algún error al momento de responder la encuesta o confusión del desarrollador durante la entrevista. Por otro lado, las respuestas indican que la razón de atraso percibida por parte de los desarrolladores fue principalmente cuando comenzaron a utilizar los productos patrón durante el proceso de desarrollo lo cual puede interpretarse la causa como falta de experiencia en el uso del patrón.

6.4.4.2.3 P.A2.3. Utilizar la solución propuesta en el patrón “Agile persona” (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales

El término “perfil” para efectos de nuestro estudio considera la identificación y descripción de las características, actitudes y actividades necesarias e importantes de los usuarios finales involucrados en el proyecto de software que se está desarrollando o está por desarrollar. En relación a esta pregunta el resultado global de la distribución de valoraciones que podemos observar en la figura C6-22 muestra que 43% estuvo “completamente de acuerdo”, un 33% “Muy de acuerdo” y un 19% “De acuerdo” haciendo un total de 95% de los desarrolladores consideran que la solución de nuestro producto patrón les permite definir de manera más exacta los perfiles de los usuarios finales.

P.A2.3. - Global

Figura C6-22 Distribución global de los porcentajes obtenidos como resultado a la evaluación de los desarrolladores a la pregunta P.A2.3

En la figura C6-23 parte a) y b) tenemos la distribución de valoraciones aplicadas por cada uno de los grupos de evaluación PRAU y PREI respectivamente. Observamos que el grupo de evaluación PREI refleja un mayor porcentaje de valoración por parte de los desarrolladores que están “Completamente de acuerdo” con la afirmación de que utilizar la solución propuesta en el producto patrón “Agile persona” les permite definir de mejor manera los perfiles de usuario. Por otro lado en este grupo es donde también se refleja la valoración “Completamente en desacuerdo” de uno de los desarrolladores que se traduce en un 8% del total.

a) Grupo de evaluación PRAU

b) Grupo de evaluación PREI

Figura C6-23 Resultados de evaluación a la pregunta P.A2.3 de cada uno de los grupos de evaluación

Algunas de las características que resaltaron los desarrolladores de la solución propuesta en el patrón fue la información y organización de la solución, otro más refirió la participación del product owner durante el diseño de las personas.

“...te decía cómo definir a la persona y te dice el objetivo de para qué se está definiendo...” – PRAU_E06

“...por cómo estaba organizado todo fue bastante fácil definir ese perfil...” – PRAU_E02

“Sí porque ya tenía los pasos definidos de cómo había que aplicar el patrón y complementó la clase que tuvimos (el taller)...” – PRAU_E01

“...decía haz tu diseño de persona y luego haz que tu product owner lo vea entonces cuando él lo veía decía ‘esto está mal’ o ‘te falta esto’...yo creo que sí ayudó a hacerlo más específico...” – PRAU_E05

Uno de los desarrolladores percibió que más comunicación entre el equipo al momento de trabajar con la solución del patrón podría aportar mayor rapidez en su aplicación y seguimiento.

“...sentimos que nos hizo falta un poco más de seguimiento en este caso y poder así acostumbrarnos...entre nosotros mismos si alguien no lo entendió no se nos ocurrió muy seguido tratar de resolver las dudas entre nosotros por si alguien entendió mejor que el otro o si otro no lo entendió de plano para que no haya problemas a la hora de seguirlo ...” – PRAU_E07

En el caso de la valoración que reflejó un 8% de la validación total corresponde a un desarrollador. Concluimos que su percepción se derivó por el hecho de que durante el desarrollo del primer módulo en el que participó al ser únicamente un usuario no tuvieron mayor problema al identificar al usuario final pero tampoco utilizó lo suficiente el patrón para asegurar que la información proporcionada en el campo “Solution” le permitía definir de manera más exacta los perfiles de los usuarios finales. Un extracto de los comentarios del desarrollador son los siguientes:

Completamente de acuerdo *“...era uno nada más (el usuario final)...estamos sobre un enfoque...sólo estamos viendo la forma como lo ve el secretario de acuerdos y las necesidades de él, no estamos extendiéndonos al proceso completo...tendrías que ver la forma en que lo va a operar o lo va a usar el técnico judicial, el secretario auxiliar, el juez, el proyectista...eso serían las otras formas de mirar o de usar el sistema...” - PREI_05*

En general de acuerdo a los resultados observados, tanto globales como de cada grupo, en los porcentajes de valoración a las diferentes escalas podemos considerar que la solución propuesta en el patrón permitió a los desarrolladores definir de manera más exacta los perfiles de los usuarios finales.

6.4.4.2.4 P.A2.4. La información proporcionada en el documento del patrón “Agile Usability Testing” facilitó la aplicación de la técnica “Usability Testing” durante el proceso de desarrollo ágil

Enfocaremos ahora nuestro análisis a la efectividad del patrón “Agile usability testing”. De manera similar a lo descrito en secciones previas visualizamos los resultados de manera global (Figura C6-24) y por grupo de evaluación (Figura C6-19 incisos a y b). En este caso en particular observamos que los resultados fueron muy positivos en ambos grupos, en ninguno de ellos se tuvo una valoración “En desacuerdo” o “Completamente en desacuerdo”. En general un 95% tuvo la valoración “Completamente de acuerdo” y “Muy de acuerdo” con el hecho de que el documento “Agile usability testing” les facilitó aplicar la técnica “Usability Testing”.

Figura C6-24 Distribución global de los porcentajes obtenidos como resultado a la evaluación de los desarrolladores a la pregunta P.A2.4

Observando la evaluación por grupo que se muestra en la figura C6-25 incisos a y b de cada uno de los grupos de evaluación tenemos que las valoraciones de los desarrolladores el grupo PREI únicamente aplicaron para las escalas “Completamente de acuerdo” y “Muy de acuerdo” lo que podemos interpretar como un muy buen resultado de la percepción que tuvieron del patrón como apoyo para facilitarles la

incorporación de la técnica durante su proceso de desarrollo. También es de destacar la diferencia de valoración aplicada por cada uno de los grupos para la escala “Completamente de acuerdo” la cual tiene mayor valoración por parte del grupo PREI con un 54% en comparación con el 25% de valoración del grupo PRAU. Este último grupo valoró otras dos escalas siendo “Muy de acuerdo” su mayor porcentaje calificado con un 62% de los resultados a comparación del 46% del grupo PREI, la escala “De acuerdo” recibió valoración de un 13% de los desarrolladores por parte de PRAU.

a) Grupo de evaluación PRAU

b) Grupo de evaluación PREI

Figura C6-25 Resultados de evaluación a la pregunta P.A2.4 de cada uno de los grupos de evaluación

En este punto los comentarios de los desarrolladores resaltaron la aportación del documento producto patrón como una guía. También apuntan que les ayudó la información que les fue proporcionada en el taller.

“...de cierta manera nos servía como guía, si teníamos dudas pues consultábamos y veíamos para qué sirve y cómo implementarlo...” - PRAU_E06

“...más que nada por las pláticas que nos había dado...y los ejemplos que nos había mostrado en las presentaciones y el vídeo...” - PRAU_E08

“...nos servía como guía...” – PRAU_E11

“...definitivamente...de no ser por el patrón no hubiéramos obtenido la información completa de lo que necesitábamos para que pudiéramos hacer los cambios...” – PRAU_E01

Ambos equipos percibieron la efectividad del patrón “Agile Usability Testing” al utilizarlo para incorporar la técnica dentro del proceso de desarrollo ágil sin embargo, en base a la valoración de ambos grupos PRAU y PREI, se refleja que los desarrolladores de la industria fueron los más convencidos del beneficio que resultó para ellos el utilizar el patrón para aplicar la técnica Usability Testing durante su proceso de desarrollo ágil. Como mencionamos anteriormente las valoraciones de este grupo fueron únicamente en las escalas de “Completamente de acuerdo” y “Muy de acuerdo”.

6.4.4.2.5 P.A2.5. Utilizar el patrón “Agile Usability Testing” durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones

Como podemos observar en la distribución de valoraciones globales en la gráfica de la figura C6-26, los resultados muestran que un total de 72% de los desarrolladores no percibieron atraso en sus tiempos programados al valorar las escalas “En desacuerdo” con un 34% y “Completamente en desacuerdo” con un 38%. Por otro lado un reducido porcentaje total del 28% de los desarrolladores valoran estar “Muy de acuerdo” (14%) o “De acuerdo” (14%) con el hecho de que utilizar el patrón les generó atraso en los tiempos de sus iteraciones.

Figura C6-26 Distribución global de los porcentajes obtenidos como resultado a la evaluación de los desarrolladores a la pregunta P.A2.5

La figura C6-27 incisos a y b nos muestra la distribución de los resultados de evaluación por cada uno de los grupos PRAU y PREI respectivamente.

Analizando de manera individual cada grupo de evaluación se identifica notables diferencias entre las valoraciones. Observamos que el grupo PRAU calificó en mayor porcentaje las escalas “Muy de acuerdo” y “De acuerdo” haciendo un total del 50% de los desarrolladores de este grupo de evaluación quienes consideran que utilizar el patrón les generó atraso en su proceso de desarrollo ágil. En el caso del grupo PREI únicamente un 15% de los desarrolladores refleja en sus valoraciones el haber percibido un atraso durante su desarrollo al utilizar el patrón “Agile Usability Testing”. Ninguno de los grupos calificó la escala “Completamente de acuerdo”.

a) Grupo de evaluación PRAU

b) Grupo de evaluación PREI

Figura C6-27 Resultados de evaluación a la pregunta P.A2.5 de cada uno de los grupos de evaluación

Para la mayoría de las escalas evaluadas se observan diferencias importantes de valoración entre un grupo y otro. La escala “Muy de acuerdo” fue valorada por el grupo PRAU con un 25% en tanto que el grupo PREI únicamente calificó esta escala con un 7%, la escala “De acuerdo” recibió una valoración del 25% de los desarrolladores del grupo PRAU mientras que el grupo PREI solamente asignó un 8%. Únicamente la escala “En desacuerdo” tuvo un menor rango de diferencia en las valoraciones con un 37% del grupo PRAU y un 31% del PREI. La escala “Completamente en desacuerdo” también refleja una notable diferencia de valoración entre los desarrolladores de cada grupo con un 13% de PRAU y un 54% de PREI.

De acuerdo a los resultados observados resulta evidente que el grupo de evaluación PRAU fue el que más percibió atraso en sus tiempos programados al utilizar el patrón “Agile Usability Testing” con un 50% de

la validación total de los desarrolladores para las escalas “Muy de acuerdo” y “De acuerdo”. El otro 50% de ellos valoró estar “En desacuerdo” y “Completamente en desacuerdo” con el hecho de que se haya generado atraso al utilizar el patrón. Mientras que, en el grupo de evaluación PREI el 85% de los desarrolladores valoró estar “En desacuerdo” y “Completamente en desacuerdo” con haber percibido algún atraso al utilizar el patrón para incorporar la técnica Usability Testing.

Con el objetivo de identificar factores relevantes por los que se percibió atraso al utilizar el patrón “Agile Usability Testing” analizamos las respuestas en la entrevista de los desarrolladores, algunos extractos se presentan a continuación:

Muy de acuerdo *“Un poco, no mucho pero sí. Debido a que se tiene que documentar cada cosa que observamos...tener que documentar todo...todo lo que tenemos que hacer es tiempo que tenemos que emplear...es parte del desarrollo pero considero que atrasa un poco...” PRAU_E06*

“Siento que es más problema de organización (interna) que problema de la técnica tal cual...” PRAU_E11

“No, nada...esta puesto dentro de las tareas...” PREI_E06

De acuerdo *“...fue un mínimo atraso para mí en el desarrollo...fue un ligero atraso...” PRAU_E07*

“No, de hecho...lo único que nos llevó tiempo fue leer el patrón y hacer nuestro documento con pruebas de usabilidad pero fuera de ello no, de hecho nos ayudó...” PRAU_E05

“No...de hecho hubiera sido un atraso no hacerlo...” PREI_E02

Observamos que hay respuestas que son contrapuestas a lo validado en la encuesta y no proporcionan información relevante que nos permita entender algún motivo en particular por el que hayan percibido que utilizar el patrón “Agile usability” generó atraso. Sin embargo aquellas que sí aportan datos que contribuye a identificar la razón de la valoración indican que fueron factores como la documentación que se tiene que realizar y organización.

Tomando en cuenta los resultados globales que reflejan un 72% total de los desarrolladores “En desacuerdo” y “Completamente en desacuerdo” con la pregunta P.A2.5 se observa que en general utilizar el patrón “Agile usability testing” durante el desarrollo ágil de los proyectos no generó atraso en los tiempos programados de las iteraciones.

6.4.4.2.6 P.A2.6. ¿La solución propuesta en el patrón “Agile Usability testing” (campo solution) facilitó aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil de software?

La valoración global que podemos observar en la figura C6-28 nos muestra que un total de 95% de los desarrolladores percibieron que la solución propuesta en el patrón “Agile usability testing” les facilitó aplicar las pruebas de usabilidad en el momento adecuado. Con valoraciones totales para las escalas “Completamente de acuerdo” de un 52% y “Muy de acuerdo” con 43% hacen evidente la evaluación positiva sobre la solución considerada en el patrón. Sin embargo una valoración “En desacuerdo” a la pregunta P.A2.6 de un desarrollador refleja un 5% en esta escala.

P.A2.6 - Global **Figura C6-28** Distribución global de los porcentajes obtenidos como resultado de la evaluación a la pregunta P.A2.6

Como observamos en la figura C6-29 incisos a y b de los grupos PRAU y PREI respectivamente, las evaluaciones muestran poca diferencia entre las valoraciones de la escala “Completamente de Acuerdo” en donde un 50% de los desarrolladores del grupo PRAU y un 54% de los desarrolladores de PREI asignaron su valoración a esta escala. La siguiente escala más valorada es “Muy de acuerdo” con mayor porcentaje por los desarrolladores de PREI con un 46% y un 37% por parte de los desarrolladores de PRAU.

a) Grupo de evaluación PEAU

b) Grupo de evaluación PEEI

Figura C6-29 Resultados de evaluación a la pregunta P.A2.6 de cada uno de los grupos de evaluación

El grupo de evaluación PREI claramente percibió en un 100% el beneficio de utilizar la solución propuesta en el patrón para incorporar la técnica usability testing durante el proceso ágil de desarrollo al valorar únicamente en las escalas “Completamente de acuerdo” y “Muy de acuerdo”.

En el caso del grupo de evaluación PRAU el porcentaje de 13% que refleja valoraciones a la escala “En desacuerdo” corresponde a la validación de un solo desarrollador. Recurriendo a la entrevista para tener información sobre la respuesta relacionada con este punto e identificar algún factor relevante tenemos el siguiente extracto de su entrevista:

En desacuerdo *“El diagrama que estaba en el documento no lo sentía muy entendible...siento que tuve que leerlo bastante hasta que logré entenderlo, no sé si porque es la primera vez que hago la técnica o porque realmente el diagrama pueda ser simplificado...”*
PRAU_E11

Se deduce que la complicación que tuvo el desarrollador para entender en un principio el patrón para poder aplicarlo dio pie a su valoración para la escala “En desacuerdo”. En general podemos considerar como positiva la percepción del grupo de evaluación PRAU respecto al beneficio que representó utilizar la solución proporcionada en el patrón con un respaldo del 87% de los desarrolladores (50% “Completamente de acuerdo” y 37% “Muy de acuerdo”) en la validación total.

6.4.4.3 A3. Resultados de utilizar los patrones en el proceso de desarrollo ágil

Otro aspecto evaluado fue el resultado percibido por los desarrolladores una vez aplicadas las técnicas de usabilidad dentro del proceso de desarrollo ágil utilizando la información contenida en los productos patrón. Como hemos visto previamente el atributo usabilidad está relacionado con asegurar que los productos interactivos son fáciles de aprender, efectivos para usar y agradables en su uso desde la perspectiva del usuario (Sharp et al., 2007). En nuestro estudio consideramos en la encuesta y entrevista los siguientes puntos para evaluar los resultados de usabilidad:

P.A3. 1. Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios

P.A3.4. En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil crees que dio como resultado aplicaciones más fáciles de utilizar por los usuarios

Otro punto considerado en nuestra evaluación de resultados es la percepción de los desarrolladores después de aplicar las técnicas persona y usability testing durante su proceso de desarrollo ágil. La técnica persona permite identificar las expectativas de los usuarios finales sobre el software conociendo y aprendiendo sobre el comportamiento y metas de los usuarios (Cooper et al., 2007), información que se puede considerar la colección de atributos para un usuario típico llamado perfil de usuario (Sharp et al., 2007). Por otro lado siendo un proceso iterativo permite conocer y aprender constantemente sobre las necesidades de los usuarios y corroborar si nuestro desarrollo lo está cubriendo de la mejor forma. Derivado de ello definimos los siguientes puntos:

P.A3.2. Durante las iteraciones hubieron cambios en las funcionalidades debido a la mala definición de los perfiles de usuario?

P.A3.3. Durante las iteraciones hubieron cambios en las funcionalidades debido a la mala interpretación de los requerimientos del usuario?

Finalmente otros dos puntos considerados para verificar la utilidad percibida de los productos patrón se definieron en relación a la reutilización que darían los desarrolladores a estos:

P.A3.5. ¿Utilizarías de nuevo el patrón “Agile persona” durante el desarrollo de software con metodologías ágiles?

P.A3.6. ¿Utilizarías de nuevo el patrón “Agile usability testing” durante el desarrollo de software con metodologías ágiles?

Extractos de las transcripciones de las entrevistas aplicadas a los desarrolladores se encuentran en los anexos G.2 y H.2 del presente documento.

En la tabla C6-31 se presentan el condensado de las frecuencias de valoración aplicada por los desarrolladores a cada uno de los puntos anteriores. La columna “ID” corresponde al identificador asignado al reactivo; “Pregunta” es la descripción de la pregunta evaluada, las cinco columnas posteriores corresponden a las frecuencias observadas en cada una de las escalas valoradas “Completamente de Acuerdo”, “Muy de acuerdo”, “De acuerdo”, “En desacuerdo” y “Completamente en desacuerdo”.

Tabla C6-31 Totales condensados preguntas aspecto A3

ID	Pregunta	Completamente de Acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en desacuerdo
P.A3.1	Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios	7	8	6	0	0

P.A3.2	Durante las iteraciones hubieron cambios en las funcionalidades debido a la mala definición de los perfiles de los usuarios	0	3	3	9	6
P.A3.3	Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos de los usuarios	3	3	6	5	4
P.A3.4	En base a tu experiencia previa en desarrollo de software, consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios	10	9	1	0	1
P.A3.5	Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles	8	6	4	3	0
P.A3.6	Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles	14	6	0	1	0

En la gráfica de la figura C6-30 se muestra el comportamiento de las evaluaciones para las preguntas P.A3.1, P.A3.2, P.A3.3 y P.A3.4. Consideramos que la evaluación global muestra resultados positivos que respaldan nuestros objetivos. Las preguntas P.A3.5 y P.A3.6 en relación con la reutilización de los productos patrón por parte de los desarrolladores se analizan posteriormente.

Resultados percibidos - Global

Figura C6-30 Evaluación global sobre los resultados percibidos por los desarrolladores al aplicar las técnicas de usabilidad

A la pregunta P.A3.1 la valoración global de los desarrolladores fue para las escalas “Completamente de acuerdo”, “Muy de acuerdo” y “De acuerdo”, siendo las más valoradas las dos últimas.

Para P.A3.2 los desarrolladores valoraron con mayor frecuencia las escalas “Completamente en Desacuerdo” y “En desacuerdo” lo cual indica tomando en cuenta resultados globales que no percibieron

mayores cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de usuario.

Aunque de forma global observamos en la gráfica que la segunda escala de mayor valoración para P.A3.3 fue “En desacuerdo” la más valorada fue la escala “De acuerdo”, lo cual indica que en general la mayoría de los desarrolladores estuvo de acuerdo con el hecho de que hubieron cambios durante las iteraciones debido a la mala interpretación de los requerimientos.

Para la pregunta P.A3.4 relacionada a la facilidad de uso percibida por parte de los usuarios en las aplicaciones desarrolladas el resultado global muestra valoraciones para las escalas “Completamente en desacuerdo”, “De acuerdo”, “Muy de acuerdo” y “Completamente de acuerdo”. Las escalas más valoradas fueron las dos últimas. El porcentaje de valoración para la escala “Completamente en desacuerdo” que se refleja en la gráfica corresponde a la opinión de un desarrollador.

Las dos últimas preguntas P.A3.5 Y P.A3.6 relacionados con la utilidad de los productos patrón fueron enfocados a la reutilización de estos por parte de los desarrolladores como apoyo para la incorporación de las técnicas de usabilidad persona y usability testing. La comparativa global de la evaluación por parte de los desarrolladores a estas dos preguntas la tenemos en la figura C6-31.

Observamos que la opinión de los desarrolladores en reutilizar el patrón “Agile Persona” no es tan contundente. Sin embargo la frecuencia de las valoraciones muestra que la mayoría lo volvería a utilizar. Resulta más clara la preferencia de reutilizar de nuevo el patrón “Agile Usability” durante un desarrollo ágil de software. Aunque en ambos casos se tuvieron opiniones de desarrolladores que externaron estar “En desacuerdo” en general la proporción de los resultados nos indica que la mayoría de los desarrolladores considera utilizar ambos patrones en futuros proyectos siendo la escala más valorada para P.A3.5 Y P.A3.6 “Completamente de acuerdo” seguida de la escala “Muy de acuerdo”.

Resultados Reutilización - Global

Figura C6-31 Evaluación global sobre los resultados percibidos por los desarrolladores al aplicar las técnicas de usabilidad

En las tablas C6-32 y C6-33 presentamos las frecuencias valoradas por los grupos de evaluación PRAU y PREI respectivamente, totales recabados a partir de las respuestas de los desarrolladores a las preguntas de la encuesta relacionados con el aspecto evaluado A3. La descripción de las columnas corresponde a la misma que se especifica para la tabla C6-31.

Tabla C6-32 Condensado de las frecuencias totales de las preguntas del aspecto A3 grupo PREI

ID	Pregunta	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en desacuerdo
P.A3.1	Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios	1	2	5	0	0
P.A3.2	Durante las iteraciones hubieron cambios en las funcionalidades debido a la mala definición de los perfiles de los usuarios	0	3	2	1	2
P.A3.3	Durante las iteraciones hubieron cambios en las funcionalidades debido a la mala interpretación de los requerimientos de los usuarios	2	2	2	1	1
P.A3.4	En base a tu experiencia previa en desarrollo de software, consideras que al incorporar las técnicas de usabilidad persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios	1	5	1	0	1
P.A3.5	Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles	1	4	1	2	0
P.A3.6	Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles	3	4	0	1	0

Tabla C6-33 Condensado de las frecuencias totales de las preguntas del aspecto A3 grupo PREI

ID	Pregunta	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en desacuerdo
P.A3.1	Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios	6	6	1	0	0
P.A3.2	Durante las iteraciones hubieron cambios en las funcionalidades debido a la mala definición de los perfiles de los usuarios	0	0	1	8	4
P.A3.3	Durante las iteraciones hubieron cambios en las funcionalidades debido a la mala interpretación de los requerimientos de los usuarios	1	1	4	4	3
P.A3.4	En base a tu experiencia previa en desarrollo de software, consideras que al incorporar las técnicas de usabilidad persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios	9	4	0	0	0
P.A3.5	Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles	7	2	3	1	0
P.A3.6	Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles	11	2			

De manera análoga a secciones previas para un mejor entendimiento de los resultados realizamos un análisis por cada una de las preguntas consideradas para obtener información sobre los resultados percibidos por los desarrolladores en sus aplicaciones una vez aplicadas las técnicas persona y usability testing.

6.4.4.3.1 P.A3.1. ¿Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios?

La distribución global de las valoraciones a esta pregunta se muestra en la figura C6-32 en donde la visión general indica que los desarrolladores sí percibieron mayor precisión en las funcionalidades identificadas para sus aplicaciones en relación a las necesidades reales de los usuarios. Para afirmar lo anterior nos basamos en que únicamente asignaron valoración a las escalas “Completamente de acuerdo”, “Muy de acuerdo” y “De acuerdo”.

Figura C6-32 Distribución global de las valoraciones a la pregunta P.A3.1

Por otro lado, profundizando un poco más el análisis la figura C6-33 incisos a y b muestra la distribución de las evaluaciones de los desarrolladores del grupo PRAU y PREI para P.A3.1.

a) Grupo de evaluación PRAU

b) Grupo de evaluación PREI

Figura C6-33 Resultados de evaluación a la pregunta P.A3.1 por grupo de evaluación

Observamos que el grupo PRAU valoró con mayor porcentaje la escala “De Acuerdo” con un 63% a diferencia con las valoraciones del grupo PREI en donde esta última tuvo la menor valoración de los desarrolladores (8%) y la mayor valoración de este grupo fue por igual para dos escalas: “Muy de Acuerdo” y “Completamente de Acuerdo” con un 46% cada una haciendo un total del 92%. Aunque todos los desarrolladores percibieron mayor precisión al identificar las necesidades de los usuarios el grupo PREI refleja la mejor valoración a este punto. Algunos comentarios de los desarrolladores del grupo PRAU son los siguientes:

PRAU_E02

“...antes de aplicar estos patrones no quedaban tan claras cuáles eran las funcionalidades que querían (los usuarios), definitivamente ayudó a dejarlas

realmente como querían porque si no eso se queda a interpretación nuestra y no necesariamente es lo que ellos buscaban...”

PRAU_E01 *“Sí...de no ser por haber usado las técnicas de usabilidad otras veces cuando se hacen las pruebas aparecen demasiados problemas en la usabilidad...definitivamente ayudaron...”*

PRAU_E04 *“...planificando en base a los perfiles ya tienes una idea sobre a quién va dirigido y no sobrepasarte un poco (de nivel) en cuanto a la definición de las preguntas o las operaciones...”*

PRAU_E05 *“Después de aplicar las pruebas de usabilidad sí...antes de aplicarlas no...porque habían muchas cosas que nosotros asumíamos que ellos sabían...a pesar de tener su perfil alguna que otra cosa lo asumíamos...”*

6.4.4.3.2 P.A3.2. ¿Durante las iteraciones hubieron cambios en las funcionalidades debido a la mala definición de los perfiles de los usuarios?

Considerando las valoraciones a las escalas “En desacuerdo” y “Completamente en desacuerdo” en la gráfica de la figura C6-34 donde se presenta la distribución general, indican que un 72% de los desarrolladores no percibió que haya sido necesario realizar cambios en las funcionalidades debido a la mala definición de los usuarios. Sin embargo también se valoraron las escalas “Muy de acuerdo” y “De acuerdo” lo cual es importante analizar a más detalle.

Figura C6-34 Distribución global de las valoraciones a la pregunta P.A3.1

La distribución de las evaluaciones para P.A3.2 por cada uno de los grupos de evaluación PRAU y PREI las tenemos en la figura C6-35 incisos a y b respectivamente. Una primera diferencia notable entre las valoraciones de cada uno de los grupos es la escala “Muy de acuerdo” a la que el grupo PRAU le asignó su valoración más alta con un 37% mientras que el grupo PREI no asignó valoración alguna a esta escala.

a) Grupo de evaluación PRAU

b) Grupo de evaluación PREI

Figura C6-35 Distribución valoraciones a la pregunta P.A3.2 por grupo de evaluación

La valoración más alta del grupo PREI la asignó a las escalas “En desacuerdo” con 61% y “Completamente en desacuerdo” con 31% y la valoración de un solo desarrollador refleja un 8% de la escala “De acuerdo”; de lo cual se interpreta que durante el desarrollo no percibieron cambios o demasiados cambios en las funcionalidades debido a la mala definición de los perfiles de usuarios. A diferencia del grupo PRAU cuyas valoraciones más altas asignadas a las escalas “Muy de acuerdo” con 37% y “De acuerdo” con “25% suman un total del 62% de los desarrolladores del equipo que indican haber percibido la necesidad de realizar cambios en funcionalidades durante el desarrollo del software debido a la mala definición de los perfiles de sus usuarios. Recurriendo a las entrevistas para identificar los posibles factores por los que se refleja ese porcentaje en la valoración de los desarrolladores tenemos los siguientes extractos clasificados de acuerdo a las escalas valoradas.

Muy de acuerdo *“Sí...había una parte donde se hacían operaciones entonces dijeron que por ejemplo, creo que tenían puesto la operación multiplicación con asterisco, entonces según ahí eso es más para personas con preparatoria y ahí se sugirió el cambio...entonces ahí si nos ayudó bastante en esa parte...”* PRAU_E04

“Creo que en el juego de los barcos hubo un pequeño cambio...quizá le bajamos un poco de nivel (de complejidad) a las operaciones...se definió que eran alumnos de primaria pero pues a lo mejor nosotros tuvimos un poco que ver ahí por poner operaciones tan complicadas que a lo mejor no se cubren (en los programas de primaria)...” PRAU_E06

“...algunos cambios...básicamente los maestros nos decían ‘cambia’ porque no les parecía bien el diseño...pero cuando hicimos la prueba de usabilidad el niño si lo entendió...fue un poco más drástico (el cambio) por los maestros que nos recomendaron...si volviésemos a aplicar las pruebas tal vez al niño le iba a gustar un poco más...” PRAU_E07

De Acuerdo *“Sí...por lo mismo, en que asumimos cosas. Ya en la prueba de usability testing fue cuando nos dimos cuenta de que estaba mal...”* PRAU_E05

“Algunas...” PRAU_E09

“No... el cuestionario, de ahí salió todo lo que necesitábamos...realmente se llenó como se necesitaba y en base a eso se desarrolló...” PREI_01

Las respuestas de los desarrolladores PRAU_E04, PRAU_E05 y PRAU_E06 reflejan que a pesar de haber definido el perfil del usuario, no dimensionaron correctamente de acuerdo a las necesidades de ese perfil y consideraron en el diseño de la interfaz elementos que finalmente no resultaron útiles para el usuario repercutiendo en la necesidad de aplicar cambios. El desarrollador PRAU_E04 comenta que para una operación aritmética consideraron un operador que en realidad es para un usuario con un nivel de estudios más avanzado que el del usuario final de la aplicación que es nivel de educación primaria. De manera similar el desarrollador PRAU_E06 señala que el nivel de complejidad considerado en las operaciones del juego era superior al que realmente era el que se necesitaba por el usuario final. En el caso del desarrollador PRAU_E05 reitera el impacto que tuvo sobre su desarrollo el hecho de asumir situaciones para el diseño de la interfaz sin consultarlas con el usuario. Previamente en su respuesta a la pregunta P.A3.1 indica que a pesar de que tenían ya definido el perfil del usuario asumieron por cuenta propia ciertas situaciones, por ejemplo, que el usuario final contaba con ciertos conocimientos mismos que realmente el usuario no había adquirido, situación que les generó problemas en su diseño y que identificaron al aplicar las pruebas de usabilidad. La falta de enfoque en la definición del perfil de usuario final al momento de diseñar la interfaz puede atribuirse a la falta de costumbre de utilizar durante el proceso de desarrollo artefactos que les permiten tener siempre presente y disponibles las necesidades del usuario como en el caso de la técnica persona que fue la que utilizaron para identificar a los usuarios finales del proyecto.

PRAU_E07 considera que los cambios que tuvieron que aplicar se debieron principalmente a las opiniones de los maestros como product owner. Es importante señalar que durante el desarrollo del proyecto del grupo PRAU se dio un cambio en los elementos que inicialmente desempeñaron el rol de product owner, a partir del segundo sprint tomaron el rol otros elementos y aquellos que eran originalmente product owner desempeñaron el papel de expertos. Sin embargo a pesar de la situación anterior se deduce de la respuesta del desarrollador que los cambios percibidos en las funcionalidades no se debieron a una mala definición del perfil del usuario.

La respuesta del desarrollador PREI_01 resulta contradictoria a la valoración asignada en la encuesta, en sus comentarios hace referencia al cuestionario utilizado por este equipo como apoyo para definir sus personas por lo que la valoración en la encuesta puede atribuirse a un error al momento del llenado de la encuesta o interpretación errónea de la pregunta.

6.4.4.3.3 P.A3.3. Durante las iteraciones ¿hubieron cambios en las funcionalidades debido a la mala interpretación de los requerimientos del usuario?

En la figura C6-36 observamos la distribución general de las valoraciones a esta pregunta. Un 14% de los desarrolladores estuvo “Completamente de acuerdo” otro 14% “Muy de acuerdo” y un 29% “De acuerdo” haciendo un total del 57% que consideró que sí hubieron cambios debido a la mala interpretación de los requerimientos del usuario.

Figura C6-36 Distribución global de las valoraciones a la pregunta P.A3.3

Analizando la distribución de los totales evaluados por los grupos PRAU y PREI mostrados en la figura C6-37 inciso a y b respectivamente tenemos que en total un 75% de los desarrolladores del grupo PRAU valoró las escalas “Completamente de acuerdo” (25%), Muy de acuerdo (25%) y De acuerdo (25%). Los desarrolladores del grupo PREI valoraron las escalas mencionadas haciendo un total de 46%.

a) Grupo de evaluación PRAU

b) Grupo de evaluación PREI

Figura C6-37 Distribución valoraciones a la pregunta P.A3.3 por grupo de evaluación

Analizando las respuestas en las entrevistas los desarrolladores de acuerdo a sus valoraciones comentaron lo que observamos en los extractos siguientes:

Completamente de acuerdo de “Sí...” PRAU_E01

“...sí hicimos unos cuantos cambios en la parte de requisitos, más que nada en la descripción de la historia de usuario...los cambios que nos hacían los maestros, sugerencias eso repercutió en cambios en las historias de usuario...” PRAU_E07

“Sí...lo bueno es que te das cuenta a tiempo...permite no verse afectado las tareas o inclusive los alcances del producto final porque muchas veces cuando no se entiende bien pues hay que disminuir alcances para poder terminar pero sí hubo malas interpretaciones...por eso te comentaba lo de las visitas porque de repente el product owner o nosotros no comprendemos bien...” PREI_E12

Muy de acuerdo

“Sí hubieron cambios...uno puede pensar un requisito como uno propiamente cree pero a la vez no es tan así...sí hubieron cambios en la complejidad... y en la interactividad...” PRAU_E02

“Sí...en ocasiones hubieron planeaciones de que product owner lo quería de esta manera entonces no se podía realizar de esa forma, entonces tuvimos que buscar otras maneras para hacerlo y no se realizó como realmente se esperaba para el product owner sino con las necesidades que se tuvo en el lenguaje...” PRAU_E04

“Sí...fueron cambios pequeños...tal vez en etiquetas, en despliegue de datos...no fueron grandes cambios, fueron más bien de visualización...pero sí hubo cambios...” PREI_E02

De Acuerdo

“Sí, principalmente por los criterios de aceptación...” PRAU_E05

“Sí...” PRAU_E09

“No...sólo un detalle pero fue así como que un bug una configuración de la agenda en formato de 12 horas daba las horas en a.m y p.m pero el manejo del juzgado está en formato de 24 horas entonces había que cambiar...siempre ha estado así y no nos había dicho nada...se cambió a formato de 24 horas...una petición que nos hizo un criterio de aceptación nos hizo ver que necesitaba ver la agenda en los horarios completos porque solo tenía horarios de oficina de ocho a tres...hasta que aplicamos usabilidad nos dijo es que las horas son en formato de 24 horas si llego a programar a la 1 de la tarde voy a poner 13:00 hrs...eso fue como un bug al final...no fue así importante que hubiera que cambiar toda la codificación...” PREI_E01

“En SIAFA recuerdo que hubo un cambio...partir del segundo sprint que es en cual nos involucramos y de a como nos lo explicaron los que previamente ya habían estado en el proyecto a lo que se hizo finalmente en el segundo sprint si hubo una modificación pero no estoy seguro de que haya sido por una mala interpretación o porque el usuario cambió de opinión...” PREI_E03

“Sí...hubo varias cosas que no han quedado claras y hubo que regresarse un poquito a lo anterior...al final de cuentas cuando se terminaba una iteración él (product owner) debería haber dicho si algo no estaba completamente entendido y no esperar hasta que otras personas criticaran para que él pudiera decir...” PREI_E11

“Si...te das cuenta a tiempo que permite no verse afectado las tareas o inclusive los alcances del producto final porque muchas veces cuando no se entiende bien pues hay que disminuir alcances para poder terminar...”
PREI_E12

Las respuestas de los desarrolladores PRAU_E07 y PRAU_E05 refieren situaciones relacionadas con cambios generados por los Product Owners en las historias de usuario y criterios de aceptación. Aunque consideran que fue por mala interpretación de los requerimientos tener constantes cambios en los requerimientos durante el desarrollo es una situación en cierta forma normal al desarrollar software utilizando metodologías ágiles. El desarrollador PRAU_E02 atribuye los cambios a que asumieron situaciones que no necesariamente eran las que el usuario necesitaba y esperaba, es decir, en algún momento priorizaron su propia percepción como desarrolladores al implementar las aplicaciones.

Además de los comentarios de aquellos desarrolladores que aplicaron valoraciones que reflejan su percepción en cuanto a los cambios debido a la mala interpretación de los requisitos del usuario llama la atención la respuesta a esta misma pregunta por parte de uno de aquellos desarrolladores que no percibieron cambios por no interpretar bien los requerimientos.

“...hubieron luego dos product owners primero fueron los LEMs y luego los maestros... obviamente en que hubo el cambio de LEMs a maestros pues si se cambiaron varias cosas porque era otro product owner pero por causas de no entender al product owner no (no se generaron cambios)...” PRAU_E11

En el caso del desarrollador PRAU_E11 atribuye los cambios a la diferencia de puntos de vista entre el product owner inicial y el que adoptó el rol posteriormente.

Tanto PREI_E01 como PREI_E02 percibieron que tuvieron que hacer cambios en sus aplicaciones debido a malas interpretaciones de los requerimientos del usuario y valoraron en diferentes escalas esta pregunta. Aunque en inicio la respuesta de PREI_E01 parece contradictoria conforme va desarrollando su explicación expone la misma situación que PREI_E02 comentó en su respuesta a la pregunta P.A3.2 al cuestionarle sobre los cambios debido a mala definición del perfil del usuario a lo cual el desarrollador señaló:

“No...en el caso de la persona pues no cambió, seguimos con el mismo...como aplicamos varias pruebas (usability testing) de una salieron unas cosas que nos afectaron en las tareas...a la mitad o casi terminando se volvió a hacer otra (prueba) con Cinthia (el product owner) y volvieron a salir detallitos claro que no muy grandes pero requirieron corrección...tiempo adicional que como estábamos holgados de tiempo nos dio chance de hacerlos.....en la funcionalidad si...habían cosas como por ejemplo lo que dijo ella de ponerle el tipo de sala a la audiencia...que se vea que sala es porque son más de una eso si nos llevó (tiempo) y eso se vió en la segunda prueba...eso sí afectó...ella quería sea de 0 horas a 23...eso no lo teníamos pensado, lo corregimos...” PREI_E02

Analizando los datos que proporcionan en su respuesta no se observa algún factor que indique haya sido necesario realizar cambios debido a la mala interpretación de los requerimientos. De hecho ambos comentan que al aplicar las pruebas de usabilidad se identificaron los detalles que les fueron observados por el product owner que era necesario cambiar.

Las respuestas de PREI_E11 y PREI_E12 sugieren que los cambios que tuvieron que aplicar durante el desarrollo de sus proyectos no se debieron a mala interpretación de los requerimientos. PREI_E11 refiere cambios de opinión del product owner influenciados por observaciones de los stakeholders involucrados situación que fue ya de corte organizacional y administrativo del product owner que afectó al equipo de desarrollo. Por su parte PREI_E12 indica que si hubo malas interpretaciones más no afectaron las tareas o alcances del producto final debido a que se pueden identificar con anticipación.

De acuerdo a lo expresado por los desarrolladores se deduce que los cambios que tuvieron que aplicarse no se debieron realmente a una mala interpretación de los requerimientos del usuario sino por omisión de

éstos durante el desarrollo y por cambios de opinión del product owner. En resumen los cambios fueron los esperados en un proceso ágil de desarrollo.

6.4.4.3.4 P.A3.4. En base a tu experiencia previa en desarrollo de software ¿consideras que las aplicaciones desarrolladas utilizando las técnicas de usabilidad en el proceso de desarrollo ágil fueron más fáciles de utilizar por los usuarios?

Considerando el concepto de usabilidad mencionado previamente el objetivo de esta pregunta fue saber si mejoró el nivel de usabilidad en el software desarrollado por los grupos de evaluación. La figura C6-38 nos presenta la distribución global de las valoraciones de los desarrolladores en cuanto al nivel de usabilidad percibido incorporando las técnicas persona y usability testing. El 95% de los desarrolladores consideró que sus aplicaciones fueron más fáciles de utilizar por los usuarios, es decir tuvieron un mejor nivel de usabilidad que proyectos previos en los que habían participado.

Figura C6-38 Distribución global de las valoraciones a la pregunta P.A3.4

En la figura C6-39 inciso a) y b) tenemos las valoraciones de los grupos PRAU y PREI respectivamente aplicadas a la pregunta P.A3.4. En la figura observamos de acuerdo a la distribución de resultados que el grupo de evaluación PREI calificó únicamente con dos escalas la pregunta P.A3.4: “Completamente de acuerdo” y “Muy de acuerdo” lo que indica que todos los desarrolladores percibieron como resultado que sus aplicaciones fueron más fáciles de utilizar por los usuarios después de incorporar en su proceso de desarrollo las técnicas persona y usability testing. En el caso del grupo de evaluación PRAU la valoración a la escala “Completamente en desacuerdo” refleja que para 13% de los desarrolladores el incorporar las técnicas de usabilidad no significó desarrollar aplicaciones más fáciles de utilizar. Sin embargo un total del 87% valoró para las escalas “Completamente de acuerdo” (12%), “Muy de acuerdo” (62%) y “De acuerdo” (13%) por lo que, en general la mayoría de los desarrolladores del grupo PRAU considera que el incorporar las técnicas dio como resultado aplicaciones más fáciles de utilizar por sus usuarios.

Figura C6-39 Distribución valoraciones a la pregunta P.A3.4 por grupo de evaluación

Resulta más contundente la percepción del equipo PREI en cuanto a un nivel de usabilidad percibido, así lo reflejan en sus valoraciones y respuestas durante la entrevista. Lo anterior posiblemente puede atribuirse en parte a la experiencia en desarrollo con la que cuenta el grupo PREI que les da mayor rango de visión en cuanto a la apreciación del nivel de utilidad en su software después de haber incorporado las técnicas de usabilidad.

Completamente de acuerdo *“Yo considero que sí...” PRAU_E02*

“Si...sin duda...a ellos (los usuarios) les gusta que los involucre... se sienten integrados, se sienten escuchados...antes pensábamos que era así y casi casi imponíamos...a ellos les gusta saber que el producto está hecho a su medida...” PREI_02

“Si, sobre todo por el usability testing... fue fácil para él entender cómo iba a estar organizado...nos ayudó mucho porque también a él le ayudó a saber que quería de funcionalidad y que no...a veces le metemos tanto al sistema...esta tan cargado nuestro sistema y es tanto lo que nosotros como departamento queremos controlar...cuando en realidad solo quiere algo tan sencillo como un registro...” PREI_E05

“Si...definitivamente sí...” PREI_E06

“Si...fue más fácil...como hay pruebas de usabilidad él ya lo está viendo de alguna manera... ya el usuario espera como viene el sistema...” PREI_E07

“...el hecho de hacer que los usuarios primero vean el diseño, los prototipos y se empiecen a involucrar más en el sistema mejora la experiencia con el usuario...” PREI_E08

“Si...con todos los bugs que corregimos, si notamos que al final el producto fue más intuitivo para el usuario que lo que presentamos inicialmente y afortunadamente los cambios fueron pequeños pero de gran impacto...” PREI_E09

“Por supuesto...la aplicación de este patrón si te ayuda bastante...ambos patrones...creo que ambos son muy buenos para definir correctamente funciones, actividades que hace cada usuario...hablando de sistemas nuevos o de funciones nuevas...hemos aplicado el patrón persona en dos sistemas en el de UPI y en el de ARCON...en el de UPI solo teníamos por el momento a la psicóloga y al juzgado que sus actividades ya sabíamos que era lo que iban a hacer...sí tiene muchas funciones pero no era parte del sistema... ..claramente percibí qué es lo que tenía que hacer el juzgado...en el caso de ARCON nos permitió ver que teníamos a dos o tres administradores y los tres tenían completamente diferentes actividades a pesar de ser la misma categoría...” PREI_E10

“Si...por ese conocimiento previo de pantallas...como tradujimos el proceso que ellos nos piden automatizar hacia la computadora, siento que fue más sencillo entendieron de alguna otra manera la funcionalidad del producto ...” PREI_E12

Muy de acuerdo *“Sí...ayudaron bastante...” PRAU_E01*

“Sí, porque como dije, en lo que era de desarrollo como que el usuario si estaba involucrado. En cambio, como está involucrado pues puede ver lo que está pasando...puede ver de una vez “esto está mal, no es así”, en cambio anteriormente en proyectos anteriores pues no, era hazlo y se lo entregas al final y obviamente si no le gusta tienes que volver a hacerlo, entonces sí se me hizo más fácil... PRAU_E05

“Sí...en el otro (proyecto desarrollado) el usuario fue a su suerte y como él lo haya entendido lo va a usar, sin embargo con la metodología se da como una versión beta de lo que estás haciendo y compruebas que sea intuitivo para el usuario y así se

entiende que cuando se libere el proyecto pues ya será intuitivo para los usuarios...”
PRAU_E06

“...pienso que sí...” PRAU_E07

“Sí...realmente sí porque como ya están involucrados ya saben más o menos como está el sistema desde antes...se les facilita mucho...ya lo probaron...saben cómo funciona...aplicar el patrón y la metodología sí es muy útil...” PREI_01

“Sí, mucho, mucho más. Nos dio una idea clara de cómo el usuario pues de repente utiliza ciertas cosas de la interfaz q uno como programador no alcanza a percibir o las ve de otro modo...” PREI_03

“Sí...” PREI_04

“En mi caso de SIAFA...no tanto...al final de cuentas mi usuaria de repente se empeñó en hacer algo que para nosotros no era correcto...lo hicimos porque al final ella es la que define pero para los demás usuarios no fue tan fácil entenderlo sin embargo aún sigo pensando que es bueno...quizá nos pudimos haber topado con muchísimas más cosas...” PREI_E11

De acuerdo *“Sí...”* PRAU_E09

Completamente en desacuerdo *“...en general sí... normalmente en las interfaces gráficas que el usuario no ve tal botón porque se pierde con tal color o que la letra es muy chica o ese tipo de cosas si se identifican en las pruebas...la respuesta es sí...”* PRAU_E11

En los comentarios anteriores la mayoría de los desarrolladores refieren principalmente al momento las pruebas de usabilidad. Únicamente PRAU_E10 hace referencia a la técnica persona.

La valoración general del 5% y del 13% “Completamente en desacuerdo” el grupo PRAU refleja la percepción de un desarrollador del grupo PRAU cuya respuesta como podemos observar en los extractos anteriores resulta un poco contradictoria a su valoración en la encuesta, el mismo desarrollador responde a la pregunta P.A3.1 referente a la precisión de las funcionalidades respecto a las necesidades reales de los usuarios lo siguiente:

“...como es un juego... si fuera un sistema normal como que si puedes ver más o menos como va pero siendo un juego cada quien...los video juegos una persona no ve esto luego otra persona si...los juegos son tan distintos a los otros tipos de sistemas...siento que para un juego no nos funcionó tanto...los niños si entendieron los juegos pero no siento que haya sido por las pruebas...los juegos son tan distintos a los sistemas normales...tomando en cuenta usuarios reales...sí...acotándolo sí...” PRAU_E11

Estas dos respuestas nos indican que para el desarrollador las funcionalidades requeridas para el proyecto de software en el que participó no pueden ser evaluadas de la misma forma que para otros tipos de software ya que al ser un juego, las opiniones de los usuarios pueden diferir mucho al utilizarlo. En base a este criterio en su opinión usar las técnicas durante el desarrollo no necesariamente incrementó el nivel de usabilidad del software, en particular refiere las pruebas de usabilidad. Una mejor definición de las pruebas de usabilidad y la interacción con más usuarios reales del proyecto pueden ser factores para que PRAU_E11 tenga mayor objetividad en lo referente a la aportación de las pruebas de usabilidad en cualquier tipo de desarrollo de software o producto.

Resumiendo el resultado la percepción de los desarrolladores coincide con la teoría ya probada relacionada con la usabilidad. Sus aplicaciones fueron más útiles para los usuarios debido al involucramiento de éstos durante el proyecto.

6.4.4.3.5 P.A3.5. ¿Utilizarías de nuevo el patrón “Agile persona” durante el desarrollo de software con metodologías ágiles?

Para obtener un panorama más claro sobre la utilidad que significó para los desarrolladores cada uno de los patrones, se definieron preguntas que nos proporcionarían información suficiente para evaluar este punto. En esta sección en particular analizamos más de cerca sus opiniones sobre el patrón persona. De manera similar a los análisis previos visualizamos de manera general las valoraciones y en particular por cada uno de los grupos que utilizaron los productos patrón. La figura C6-40 muestra la distribución global de las evaluaciones aplicadas en relación al producto patrón “Agile Persona”. Tomando en cuenta las valoraciones para las escalas “Completamente de acuerdo” con un 38%, “Muy de acuerdo” con 29% y “De acuerdo” con 19% nos indican que un total de 86% de los desarrolladores volvería a utilizar el patrón “Agile persona” durante el desarrollo ágil de un proyecto de software.

Figura C6-40 Distribución global de los porcentajes obtenidos como resultado de la evaluación a la pregunta P.A3.5

En la figura C6-41 incisos a y b se presentan las distribuciones de valoraciones para cada uno de los grupos de evaluación PRAU y PREI respectivamente. Al observar las gráficas por grupo se notan diferencias importantes entre las valoraciones de cada uno de ellos. Por un lado un 12% de los desarrolladores del grupo PRAU valoraron la escala “Completamente de acuerdo” mientras que en el grupo PREI un porcentaje mayor con 54% de los desarrolladores asignaron su validación a esa escala. “Muy de acuerdo” tiene el 50% de las valoraciones de los desarrolladores de PRAU y un 15% del grupo PREI.

Otra diferencia importante entre las valoraciones de los grupos es el 25% para la escala “En desacuerdo” por el grupo PRAU interpretando con ese resultado que ese porcentaje de los desarrolladores no consideran utilizar de nuevo el patrón “Agile persona”.

a) Grupo de evaluación PRAU

b) Grupo de evaluación PREI

Figura C6-41 Distribución de valoraciones por grupo de evaluación a la pregunta P.A3.5

A diferencia del resultado en el grupo PRAU solamente el 8% de los desarrolladores del grupo PREI valoró estar “En desacuerdo” con utilizar de nuevo el patrón. Las escalas más valoradas por el grupo PRAU fueron “Muy de acuerdo” (50%) y “En desacuerdo” (25%), por otro lado las escalas más valoradas por el grupo PREI fueron “Completamente de acuerdo” (54%) y “De acuerdo” (23%). En general un 75% de los desarrolladores del grupo de evaluación PRAU considera utilizar de nuevo el patrón “Agile persona” a comparación de un 92% de los desarrolladores del grupo PREI.

Enfocamos un análisis en particular a las respuestas en sus respectivas entrevistas de aquellos desarrolladores que validaron estar “En desacuerdo” de volver a utilizar el patrón. Como vemos en los extractos siguientes sus respuestas son contradictorias a la validación aplicada en la encuesta y no aportan datos suficientes que nos permitan identificar factores que hayan influido en la valoración aplicada.

En desacuerdo “Sí...” PRAU_E05

“Sí...” PRAU_E11

“Sí...” PREI_E02

Por otro lado algunas de las respuestas a esta pregunta de los desarrolladores que sí consideran utilizar de nuevo el patrón “Agile persona” resaltan ciertas características que a nuestro criterio vale la pena remarcar.

“Sí, es muy útil porque si no tienes idea de cómo definirlo pues eso te ayuda...a no empezar desde cero con eso ya tienes una base para hacer tu persona...” PRAU_E04

“Sí...al principio, es muy útil el tenerlo al principio...sobre todo cuando estás codificando con niveles de roles y privilegios...el patrón persona permite ahondar un poco sobre cómo es nuestro usuario, de ahí podemos sacar si le gusta tener un ambiente ordenado, si es metódico, si se acostumbra a tener un camino a seguir...por ejemplo nuestro usuario nos dijo que le gusta tener todo ordenado de ahí dijimos que presentarle un menú en forma de árbol se le sería mucho más sencillo a él utilizar las opciones. En el caso de Cinthia nos dijo que le gusta todo más ágil entonces le presentamos las opciones de primera mano...estaba en la pantalla principal las opciones y ella nada más elegía la opción que quería...si ayudó un poco a definir como ir presentando la aplicación ante el usuario, la parte de la vista...” PREI_E01

“Dependiendo del proyecto...lo usaría en proyectos nuevos...para conocer al usuario...en proyectos iniciados ya se tienen los perfiles...” PREI_E04

“Sí...es claro...de hecho sería necesario sobre todo cuando sea un proyecto nuevo...definir los roles y los enfoques...inclusive eso te dice en cuanto a nosotros manejamos la seguridad, cómo definir el rol dentro del sistema para saber a donde pueda tener acceso, cuales son las partes del proceso que él puede mirar, etc...” PREI_E05

“Si...te ahorra mucho tiempo...aquí (en el departamento de innovación) cuando desarrollamos cada elemento que tú tienes puedes definirlo como una opción dentro del sistema de seguridad...con el patrón persona identificas los roles que vas a poner en el sistema de seguridad y vas colocando las opciones que tu consideras que deben de ser para qué rol y para qué rol...la definición persona te va a permitir definir los niveles de seguridad...” PREI_E08

“Si...si nos permitió visualizar más de un rol...eso no lo habíamos contemplado, ni el product owner nos lo había externado explícitamente hasta ese momento fue que nos dimos cuenta...basado en el patrón persona que dice qué datos debe tener el formatito (la plantilla del patrón) ...se lo entregamos (un cuestionario) a todos los miembros del departamento y de ahí fue como cada uno lo contestó con calma

sin presión ni nada ahí vimos que cada uno tiene actividades diferentes aun teniendo la misma categoría jerárquicamente...” PREI_E09

PRAU_E04 señala de nuevo la importancia que tuvo el patrón para él como una guía a seguir en la definición de las personas. El desarrollador PREI_E09 menciona que se apoyaron de la información contenida en la plantilla del patrón para generar un cuestionario que les sirvió de artefacto adicional para la definición de sus personas.

Cabe mencionar que el grupo PREI controla el acceso de sus usuarios a los diferentes sistemas por medio de la asignación de permisos a través de un sistema de seguridad interno en donde controlan los privilegios de cada usuario y para ello requieren identificar cuáles son necesarios controlar en las aplicaciones a implementar. El desarrollador PREI_E01, además de los resultados conocidos y esperados de esta técnica, indica que fue muy útil al principio y cuando requieren identificar los niveles de roles y privilegios de los usuarios. Este mismo punto también lo mencionan PREI_05 y PREI_E08 como podemos observar en los extractos de sus respuestas durante la entrevista.

PREI_E04 y PREI_E05 subrayan que el patrón lo usarían principalmente en proyectos nuevos. Incluso el desarrollador PREI_E06 en respuesta a otra pregunta la P.A1.3 en relación a los aspectos positivos del patrón hace mención de este punto:

“...cuando estamos trabajando con sistemas sobre todo que son nuevos y en el cual están involucrados muchas personas o muchos roles pues si conviene tener la mayor claridad posible...hay otras ocasiones en las que como son sistemas ya bien definidos o roles bien definidos...si puede resultar un poquito tedioso porque es información que ya se tiene...es más como si fuera vamos a documentarlo no porque se desconoce sino como parte del proceso” PREI_E06

Concretando un poco de esta pregunta identificamos que los desarrolladores volverían a utilizar producto patrón persona por que representa una guía para ellos, les ayuda a definir sus perfiles de permisos y lo utilizarían principalmente en del desarrollo de sistemas de software nuevos.

6.4.4.3.6 P.A3.6. ¿Utilizarías de nuevo el patrón “Agile Usability Testing” durante el desarrollo de software con metodologías ágiles?

De manera análoga al análisis de la utilidad que significó el patrón “Agile persona” en esta sección analizamos el mismo punto pero en relación al producto patrón “Agile Usability Testing”. Para la pregunta P.A3.6 en la gráfica de la figura C6-42 observamos las valoraciones globales a la posibilidad de utilizar de nuevo el patrón “Agile Usability testing”. El resultado indica, considerando las valoraciones a las escalas “Completamente de acuerdo” (67%) y “Muy de acuerdo” (28%), que en promedio un 95% de los desarrolladores volvería a utilizar el patrón durante el desarrollo de software utilizando alguna metodología ágil.

Figura C6-42 Distribución global de los porcentajes obtenidos como resultado de la evaluación a la pregunta P.A3.6

Aunque en términos generales la mayoría de los desarrolladores considera volver a utilizar de nuevo el patrón “Agile Usability Testing” las valoraciones de cada grupo de evaluación como en secciones anteriores muestra diferencias en las validaciones de cada uno de ellos. Las distribuciones de las valoraciones de cada grupo las tenemos en la figura C6-43 incisos a y b donde la diferencia más evidente es el resultado en la valoración de la escala “Completamente de acuerdo” a la que los desarrolladores del grupo PRAU valoraron con un 37% mientras que en el grupo PREI un 85% de los desarrolladores valoró esta escala. Por otro lado la escala “Muy de acuerdo” fue valorada por el 50% de los desarrolladores del grupo PRAU y un 15% del grupo PREI.

El porcentaje del 13% de validación para la escala “En desacuerdo” que se refleja en la distribución total del grupo de evaluación PRAU (figura C6-43 inciso a) corresponde a la valoración de uno de los desarrolladores del grupo. Por ende al no tener el grupo PREI valoración alguna para esa escala, en la distribución global a la pregunta P.A3.6 esa misma validación es la que genera el 5% en la valoración general de la escala mencionada.

a) Grupo de evaluación PRAU

b) Grupo de evaluación PREI

Figura C6-43 Distribución de valoraciones por grupo de evaluación a la pregunta P.A3.6

La respuesta del desarrollador que valoró la escala “En Desacuerdo” resulta contradictoria a su valoración en la encuesta y muy escueta para identificar factores que hayan influido en ella.

En desacuerdo “Sí...” PRAU_E11

A partir de las respuestas de los desarrolladores que volverían a utilizar el patrón “Agile usability testing” también se identifican factores a remarcar.

“Si...es como una referencia, siempre van a haber dudas entonces yo puedo acudir a eso (al producto patrón) para apoyarme...” PRAU_E06

“Yo pienso que sí porque es para mí algo rápido de hacer, un poco más a la mano...con información así escorada con apuntar las observaciones y con apuntar los pasos a seguir...no hay que escribir demasiado...es una forma de tabla que facilita mucho al momento de analizar lo que estamos escribiendo y ya tener una idea más visible a lo quiere el usuario...yo pienso que sí lo volvería a usar” PRAU_E07

“Sí...sin duda...nos lleva de la mano” PREI_E02

“Sí... Primero que se puede utilizar bajo diversos esquemas...es como que genérico y dependiendo de tu necesidad tú vas siguiendo camino. Y por último porque no está limitado a una forma de hacerlo sino que es adaptable...” PREI_E06

“Si... el diagrama...te permite mostrar y ver específicamente en qué momento...te da un orden de lo que tienes que hacer e inclusive...si aparecieron algún detalle, algún problema o algo...el mismo patrón te sugiere o te marca el lineamiento de que resuelve lo que puedas resolver de acuerdo al tiempo y lo que quede te vas a un siguiente sprint...este mismo diagrama presentándolo ante superiores ante personas no

enfocadas a desarrollo de software les puede dar una idea...hay un orden que estamos llevando y este es nuestro orden...aquí están estos momentos...sepan qué se hace...les queda la duda que estás haciendo...involucrarlos y mostrarles procesos ya definidos...les da una idea de que no lo estás inventando...lo estás haciendo en base a algo...” PREI_E08

Podemos resumir que las respuestas de los desarrolladores apuntan como característica principal por la que utilizarían de nuevo el patrón “Agile usability testing” el hecho que representó una guía para ellos al momento de usar la técnica usability testing dentro de su proceso de desarrollo. Adicionalmente PREI_E08 menciona que el diagrama que se presenta en la solución del patrón puede servir para involucrar más a los usuarios sobre el proceso que se sigue durante las pruebas de usabilidad e incluso servirles a ellos también de guía.

De los dos patrones evaluados el que tuvo mayor porcentaje de evaluación por parte de los desarrolladores para ser utilizado de nuevo es el patrón “Agile usability testing” con un promedio de 95% de aceptación de los desarrolladores a utilizarlo de nuevo a diferencia con el “Agile persona” con un 86% de aceptación de los desarrolladores.

6.4.5 Discusión resultados de evaluación en proyectos reales

Las valoraciones aplicadas en la encuesta por los desarrolladores reflejan resultados que podemos interpretar como positivos para respaldar nuestra propuesta de patrones como una herramienta reutilizable de apoyo a los desarrolladores para incorporar técnicas de usabilidad durante el proceso ágil. Por otro lado, las respuestas a las entrevistas proporcionan información que complementa las valoraciones aplicadas por los desarrolladores y permite identificar áreas de oportunidad para mejorar la utilidad de los patrones. Para una visión más clara de los resultados consideramos en nuestra discusión de resultados la percepción sobre la claridad del formato e información contenida en el patrón que contribuye al análisis de la facilidad o dificultad que representa utilizarlos.

Por otro lado también se desarrolla una comparativa de los resultados entre cada uno de los patrones como guías para los desarrolladores en la incorporación efectiva del uso de las técnicas de usabilidad y el nivel de usabilidad obtenido en el software al aplicarlas siguiendo los patrones en su entorno ágil.

Estructura y contenido de los patrones

En relación a la organización del patrón a manera de tabla con el que se presenta la información del patrón no hubo mayor comentario, la estructura del patrón no representó problema para ellos. No así algunos contenidos, se identifica que algunos campos no son lo suficientemente claros en su descripción y causaron cierta confusión para el desarrollador, esta deficiencia se cubrió en parte con el taller que se impartió antes de que utilizaran los patrones. No obstante, durante el desarrollo de los sprints ciertos términos que en su momento fueron mencionados y explicados en el taller causaron cierta confusión al consultarlos en el patrón por lo que los desarrolladores sugieren especificarlos un poco más al momento de hacer referencia a ellos en el patrón. En cuanto al proceso descrito en el campo “solution” particularmente del patrón “Agile Usability Testing” los equipos utilizaron diferentes enfoques al momento de incorporar las técnicas de usabilidad en su proceso de desarrollo. Aunque el diagrama presentado en la solución como mencionan en sus comentarios los mismos desarrolladores les va guiando independientemente del enfoque utilizado, se identifica la necesidad de indicar en el diagrama de manera más clara los posibles caminos a seguir de acuerdo al enfoque aplicado y por ende una descripción más detallada. En esta misma línea una observación relevante fue la relacionada al proceso a seguir durante un sprint en el que únicamente se realizarán tareas de diseño de la interfaz.

Los formatos de las plantillas propuestas en “Agile persona” y “Agile Usability Testing” resultaron muy útiles para los desarrolladores y al igual que los ejemplos fueron dos de los aspectos positivos que mencionaron reiterativamente en la encuesta y en las entrevistas. El grupo PREI personalizó la plantilla de “Agile Usability Testing” agregando el logotipo de la institución al momento de utilizar los impresos

durante las pruebas de usabilidad. Cabe mencionar el hecho que durante el uso de la plantilla en los desarrollos derivado de la observación que se realizó como seguimiento a la evaluación en los proyectos reales se aplicó un cambio en la plantilla de la lista de revisión utilizada durante las pruebas de usabilidad, posteriormente no se identificó la necesidad de otro cambio. La versión mejorada de este formato se puso a disposición de los grupos de evaluación y lo podemos consultar en el la sección 5.2.1.

Comparativa entre los patrones “Agile persona” y “Agile Usability Testing”

A continuación presentamos un análisis comparativo entre los dos productos patrón que nos amplía el panorama general sobre la efectividad percibida de cada uno de ellos como guías para usar las técnicas persona y usability testing en su ambiente ágil de desarrollo.

En relación con el primer aspecto: “A1. Facilidad/Dificultad de utilizar los productos patrón” los porcentajes de valoración para cada uno de los patrones evaluados se presentan en la tabla C6-34. La columna “ID” corresponde al identificador de la pregunta; “Pregunta” es la descripción de la pregunta definida y publicada en la encuesta; por último las cinco columnas posteriores corresponden a las escalas de valoración para cada pregunta “Completamente de acuerdo”, “Muy de acuerdo”, “De acuerdo”, “En desacuerdo” y “Completamente en Desacuerdo”.

En la tabla están sombreadas las columnas de las escalas “Completamente de acuerdo”, “Muy de acuerdo” y “En desacuerdo” consideradas para nuestras interpretaciones y se resaltan en negritas los porcentajes relevantes. El patrón “Agile persona” obtuvo un total del 23% de las validaciones de los desarrolladores mientras que el patrón “Agile Usability Testing” tuvo un 15% de la valoración en las escalas mencionadas. “Agile persona” tuvo un 9% más de los desarrolladores que percibieron dificultad en entender su estructura. Comparando las valoraciones reflejadas se puede deducir que el patrón “Agile Persona” resultó más complicado de entender para los desarrolladores.

Tabla C6-34 Porcentajes globales para las preguntas P.A1.1 y P.A1.2

ID	Pregunta	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en Desacuerdo
P.A1.1	Fue complicado entender la estructura del patrón “Agile Persona”	0%	14%	9%	48%	29%
P.A1.2	Fue complicado entender la estructura del patrón “Agile Usability Testing”	5%	5%	5%	52%	33%

Resulta interesante la diferencia de valoraciones que muestran los resultados de cada uno de los grupos de evaluación PRAU y PREI. En la tabla C6-35 podemos observar los resultados por cada equipo para las dos preguntas P.A1.1 y P.A1.2, las columnas de la tabla corresponden a la misma descripción de la tabla C6-32.

Tabla C6-35 Porcentajes globales para las preguntas P.A1.1 y P.A1.2 por grupo de evaluación

ID	Pregunta	Grupo	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en Desacuerdo
P.A1.1	Fue complicado entender la estructura del patrón “Agile Persona”	PRAU	0%	25%	25%	37%	13%
		PREI	0%	8%	0%	54%	38%
P.A1.2	Fue complicado entender la estructura del patrón “Agile Usability Testing”	PRAU	12%	12%	13%	50%	13%
		PREI	0%	0%	0%	54%	46%

Tomando en cuenta las escalas representadas en las columnas sombreadas observamos que un 50% del grupo PRAU considera que fue complicado entender la estructura del patrón “Agile Persona” mientras que en el grupo PREI un 8% lo consideró complicado. Por otro lado un 37% de los desarrolladores de PRAU consideraron complicado entender el patrón “Agile usability testing” a diferencia del grupo PREI en donde el 0%, es decir ningún desarrollador, lo consideró complicado de entender. Los productos patrón “Agile persona” y “Agile Usability Testing” resultaron más complicados de entender para el grupo de evaluación PRAU.

Continuando con los aspectos evaluados en el estudio, en relación con el A2: Efectividad de los productos patrón como guía para incorporar las técnicas de usabilidad en el proceso ágil, se definieron seis reactivos para obtener información y analizar este punto. Las tablas C6-36, C6-37 y C6-38 muestran los porcentajes de valoración aplicados por los desarrolladores que nos permiten comparar los resultados para cada uno de los productos patrón evaluados. Las columnas de la tabla corresponden a la misma descripción de la tabla C6-32. Para el análisis de los resultados relacionados a los puntos del aspecto A2 consideraremos las valoraciones a las escalas “Completamente de acuerdo”, “Muy de acuerdo” y “De acuerdo”.

Tabla C6-36 Distribución de porcentajes globales para las preguntas P.A2.1 y P.A2.4

ID	Pregunta	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en Desacuerdo
P.A2.1	La información proporcionada en el documento del patrón "Agile persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil	33%	53%	14%	0%	0%
P.A2.4	La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil	43%	52%	5%	0%	0%

Observamos en la tabla C6-36 que la información de ambos productos patrón facilitó la aplicación de las respectivas técnicas de usabilidad. Sin embargo, analizando más de cerca las valoraciones, el patrón “Agile usability testing” tuvo mayor validación por parte de los desarrolladores en la escala “Completamente de acuerdo” con un 43% sobre el 33% del patrón “Agile persona”. En base a este resultado podemos considerar que dicho patrón, de acuerdo a la percepción de los desarrolladores, fue el que mejor ayudó para incorporar la técnica de usabilidad correspondiente, en este caso las pruebas de usabilidad.

Los puntos P.A2.2 y P.A2.5 nos ayudan a evaluar si la incorporación de las técnicas de usabilidad durante el proceso de desarrollo ágil significó atraso para los equipos ágiles. Los resultados globales de valoración de los desarrolladores para estos puntos los tenemos en la tabla C6-37.

Tabla C6-37 Distribución de porcentajes globales para las preguntas P.A2.2 y P.A2.5

ID	Pregunta	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en Desacuerdo
P.A2.2	Utilizar el patrón “Agile Persona” durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	5%	24%	19%	43%	9%
P.A2.5	Utilizar el patrón “Agile Usability Testing” durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	0%	14%	14%	34%	38%

Los dos últimos puntos evaluados para analizar el aspecto A2 son P.A2.3 y P.A2.6 relacionados con la solución propuesta en los productos patrón. Las valoraciones a estos dos puntos las tenemos a continuación en la tabla C6-38.

Tabla C6-38 Distribución de porcentajes globales para las preguntas P.A2.3 y P.A2.6

ID	Pregunta	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en Desacuerdo
P.A2.3	Utilizar la solución propuesta en el patrón “Agile Persona” (Campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales	43%	33%	19%	0%	5%
P.A2.6	La solución propuesta en el patrón “Agile Usability Testing” (Campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software	52%	43%	0%	5%	0%

Interpretamos que las soluciones propuestas de ambos productos patrón fueron valoradas positivamente con un porcentaje total, tomando en cuenta las tres escalas mencionadas previamente, del 95% cada una. Analizando más de cerca la escala “Completamente de acuerdo” el patrón “Agile usability testing” obtuvo mayor porcentaje de valoración con un 52% a comparación del 43% del patrón “Agile persona”, diferencia que se puede interpretar como mejor validación para el patrón “Agile usability Testing”.

Profundizando por grupo de evaluación la tabla C6-39 nos presenta los resultados de valoración por cada equipo para cada una de las preguntas definidas para analizar el aspecto A2 del estudio. En la tabla se han sombreado las columnas de las escalas cuyas valoraciones dan pauta a nuestro análisis y conclusiones. Los valores resaltados representan valores que consideramos relevantes para nuestro análisis y a los que hacemos particular referencia durante la descripción de éste. Las columnas que conforman la tabla son las mismas descritas previamente para la tabla C6-34 y posteriores.

A la pregunta P.A2.1 todos los desarrolladores de ambos grupos percibieron que la información del patrón “Agile persona” les facilitó aplicar la técnica persona sin embargo un mayor porcentaje de los desarrolladores del grupo PREI estuvo completamente de acuerdo con esta afirmación. El 25% del grupo PRAU y 39% de los desarrolladores del grupo PREI.

En relación a la pregunta P.A2.2 y a pesar de que se refleja un mayor porcentaje de validación a la escala “Completamente de acuerdo” por parte del grupo PREI con un 8%, el resultado global muestra que los desarrolladores de ese mismo grupo de validación evaluaron con menor porcentaje el hecho que aplicar el patrón haya generado atraso en las iteraciones programadas. Un 62% de los desarrolladores de PRAU consideró que sí genero atraso utilizarlo mientras que en el grupo PREI un 39%, interpretando de esta forma que usar el patrón significó atraso en mayor porcentaje para el grupo PRAU con más de la mitad de sus desarrolladores respaldando esta afirmación. A pesar de que la opinión de uno de los desarrolladores del grupo PREI representa un 8% de ellos “Completamente en desacuerdo” con P.A2.3, entre los dos grupos un mayor porcentaje de los desarrolladores del grupo PREI (46%) estuvo completamente de acuerdo con este punto comparado con un 37% del grupo PRAU.

Ambos grupos percibieron que la información proporcionada en el patrón “Agile usability testing” les facilitó aplicar la técnica usability testing durante el proceso ágil de desarrollo. Como observamos en la tabla ninguno de los desarrolladores valoró alguna de las dos últimas escalas. Se refleja en los resultados que los desarrolladores del grupo PREI validó en mayor porcentaje (54%) estar “Completamente de acuerdo” con P.A2.4. de nuevo PREI asignó la mejor validación. Para el caso de la pregunta P.A2.5 el resultado de la validación del equipo PRAU fue la más alta entre los dos grupos de evaluación. El utilizar el patrón “Agile usability testing” significó para ellos atraso en las iteraciones valorando esto un total del

75% de los desarrolladores de PRAU en tanto que del grupo PREI únicamente el 15% consideró que se generó atraso al utilizar el patrón.

Tabla C6-39 Porcentajes globales para las preguntas del aspecto A2 por grupo de evaluación

ID	Pregunta	Grupo	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en Desacuerdo
P.A2.1	La información proporcionada en el documento del patrón "Agile persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil	PRAU	25%	62%	13%	0%	0%
		PREI	39%	46%	15%	0%	0%
P.A2.2	Utilizar el patrón "Agile Persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	PRAU	0%	50%	12%	38%	0%
		PREI	8%	8%	23%	46%	15%
P.A2.3	Utilizar la solución propuesta en el patrón "Agile Persona" (Campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales	PRAU	37%	38%	25%	0%	0%
		PREI	46%	31%	15%	0%	8%
P.A2.4	La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil	PRAU	25%	62%	13%	0%	0%
		PREI	54%	46%	0%	0%	0%
P.A2.5	Utilizar el patrón "Agile Usability Testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	PRAU	25%	25%	25%	37%	13%
		PREI	0%	7%	8%	31%	54%
P.A2.6	La solución propuesta en el patrón "Agile Usability Testing" (Campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software	PRAU	50%	37%	13%	0%	0%
		PREI	54%	46%	0%	0%	0%

Los resultados anteriores nos indican que el grupo PRAU percibió mayor afectación en sus tiempos programados. Con muy poca diferencia de validación el punto P.A2.6 fue mejor valorado por los desarrolladores del grupo PREI con un 54% de ellos calificando estar "Completamente de acuerdo" contra un 50% del grupo PRAU, la solución del patrón "Agile usability testing" tuvo mejor percepción por parte del grupo PREI.

Las distribuciones globales de valoración a cada una de las preguntas definidas para analizar el aspecto A3 se presenta en la tabla C6-40. De forma similar al formato de la tabla C6-39 para nuestro análisis las columnas sombreadas corresponden a las escalas cuyas valoraciones dieron pauta a las conclusiones sobre la evaluación a este aspecto.

Únicamente dos de las preguntas, P.A3.3 y P.A3.5, muestran resultados que en particular merecen un análisis más específico. La mayoría de los desarrolladores percibió que hubo cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos de los usuarios.

Tabla C6–40 Porcentajes globales para las preguntas del aspecto A3

ID	Pregunta	Completamente de Acuerdo	Muy de Acuerdo	De Acuerdo	En Desacuerdo	Completamente en Desacuerdo
P.A3.1	Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios	33%	38%	29%	0%	0%
P.A3.2	Durante las iteraciones hubieron cambios en las funcionalidades debido a la mala definición de los perfiles de los usuarios	0%	14%	14%	43%	29%
P.A3.3	Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos de los usuarios	14%	14%	29%	24%	19%
P.A3.4	En base a tu experiencia previa en desarrollo de software, consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios	47%	43%	5%	0%	5%
P.A3.5	Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles	38%	29%	19%	14%	0%
P.A3.6	Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles	67%	28%	0%	5%	0%

En cuanto a la reutilización de los patrones, el patrón “Agile persona” tiene menos valoración para ser utilizado de nuevo.

La tabla C6-41 nos presenta los resultados de valoración por cada equipo para cada uno de las preguntas definidos para analizar el aspecto A3 del estudio. Las columnas sombreadas corresponden a las escalas cuyas valoraciones dieron pauta a las comparaciones de resultados entre los grupos y, los porcentajes resaltados representan a los que hacemos particular referencia durante la descripción del análisis.

Todos los desarrolladores estuvieron de acuerdo en que las funcionalidades identificadas fueron más precisas en relación a las necesidades de los usuarios (P.A3.1) sin embargo la valoración a nivel de grupos muestra que un mayor porcentaje de los desarrolladores del grupo PREI (46%) estuvo “Completamente de acuerdo” con este punto. Mucha diferencia entre las validaciones de cada equipo tomando en cuenta el 12% de los desarrolladores del grupo PRAU. La mejor valoración fue del grupo PREI.

Para el punto P.A3.2 la mejor valoración de nuevo fue por parte del equipo PREI valorando con un 8% en total comparado con un 62% del grupo PRAU. Este resultado refleja el grado en que afectó al equipo ciertas situaciones organizativas que se presentaron durante el desarrollo como el cambio de product owner. De manera similar P.A3.3 recibió mejor valoración por parte del grupo PREI con un menor porcentaje de los desarrolladores validando las tres primeras escalas con un total de 46% a diferencia del 75% del grupo PRAU, la valoración global indica que la mayoría de los desarrolladores percibió que tuvieron que hacerse cambios a las funcionalidades debido a la mala interpretación de los requerimientos.

Todos los desarrolladores de PREI percibieron que sus aplicaciones fueron más fáciles de utilizar por sus usuarios, de este 100% un 69% de los desarrolladores del grupo estuvo “Completamente de acuerdo”. En el grupo PRAU 87% de los desarrolladores percibieron aplicaciones más fáciles después de utilizar las técnicas de usabilidad, de ese total únicamente el 12% de los desarrolladores validó estar

“Completamente de acuerdo”. Dado que al validar este punto nos interesa que los desarrolladores hayan percibido aplicaciones más fáciles de utilizar podemos considerar que la mejor valoración fue del equipo PREI. Para P.A3.5 un 75% de PRAU considera volver a utilizarlo y un 92% de PREI.

Tabla C6-41 Distribución de valoraciones para los cuestionamientos del aspecto A3 por grupo de evaluación

ID	Pregunta	Grupo	Completamente de Acuerdo	Muy de Acuerdo	De Acuerdo	En Desacuerdo	Completamente en Desacuerdo
P.A3.1	Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios	PRAU	12%	25%	63%	0%	0%
		PREI	46%	46%	8%	0%	0%
P.A3.2	Durante las iteraciones hubieron cambios en las funcionalidades debido a la mala definición de los perfiles de los usuarios	PRAU	0%	37%	25%	13%	25%
		PREI	0%	0%	8%	61%	31%
P.A3.3	Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos de los usuarios	PRAU	25%	25%	25%	12%	13%
		PREI	7%	8%	31%	31%	23%
P.A3.4	En base a tu experiencia previa en desarrollo de software, consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios	PRAU	12%	62%	13%	13%	0%
		PREI	69%	31%	0%	0%	0%
P.A3.5	Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles	PRAU	12%	50%	13%	25%	0%
		PREI	54%	15%	23%	8%	0%
P.A3.6	Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles	PRAU	37%	50%	13%	0%	0%
		PREI	85%	15%	0%	0%	0%

Además de que un mayor número de desarrolladores de PREI está dispuesto a utilizar el patrón “Agile persona”, los resultados reflejan que más de la mitad (54%) valoró estar “Completamente de acuerdo”. Por otro lado todos los desarrolladores (100%) consideran volver a utilizar en un desarrollo ágil el patrón “Agile Usability Testing”, las validaciones por grupo indican que un 85% del equipo PREI registró estar “Completamente de acuerdo” en volver a utilizarlo mientras que en el grupo PRAU solamente un 37% de los desarrolladores calificó en esta escala al P.A3.6. No obstante, dados los resultados de PRAU en las otras dos escalas “Muy de acuerdo” con 50% y “De acuerdo” con 13% indican que todos los desarrolladores de PRAU volverían a utilizar el patrón. De nuevo la mejor valoración fue por parte del grupo PREI.

En general, de acuerdo a los parámetros de evaluación considerados para nuestro análisis, las valoraciones en los tres aspectos definidos para los productos patrón propuestos fueron positivas. Existen diferencias entre cada uno de los patrones sin embargo en base a los resultados de las valoraciones podemos resaltar las siguientes:

A1. Facilidad/Dificultad de utilizar los patrones.

- La estructura del patrón “Agile persona” resultó la más complicada de entender para los desarrolladores.
- El grupo de evaluación PRAU percibió mayor complicación para entender ambos productos patrón “Agile persona” y “Agile usability testing”.

A2. Efectividad de los patrones como guía para incorporar las técnicas de usabilidad en el proceso ágil

- La información del patrón “Agile usability testing” tuvo mejor percepción como facilitadora para aplicar la técnica de usabilidad correspondiente durante el proceso de desarrollo ágil.
- Los desarrolladores percibieron menos atraso durante su proceso de desarrollo ágil al utilizar el patrón “Agile usability testing”.
- La solución del patrón “Agile usability testing” se percibió como la más efectiva.

A3. Resultado de utilizar los patrones en el proceso de desarrollo ágil

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios.
- El grupo de evaluación PREI percibió funcionalidades más precisas en relación a las necesidades reales de los usuarios
- El grupo PREI fue el que menos cambios percibió en sus funcionalidades debido a mala definición de perfiles de usuarios
- Los desarrolladores tuvieron que realizar cambios en sus funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario. Sin embargo, el grupo PREI fue el que menos percibió este factor.
- El grupo de evaluación PREI asignó mayor valoración al hecho de que incorporar técnicas de usabilidad durante el proceso de desarrollo ágil dio como resultado aplicaciones más fáciles de utilizar por los usuarios.
- El patrón “Agile usability testing” es el más valorado para utilizar de nuevo durante el proceso de desarrollo ágil.
- En promedio el grupo de evaluación PREI es el que más validación asignó a la escala “Completamente de acuerdo” para utilizar los producto patrón “Agile persona” y “Agile usability Testing”.

6.5 Conclusiones sobre la evaluación

Cada una de las etapas de evaluación aplicadas a los patrones proporcionó información relevante a nuestro estudio. Como se describió previamente el objetivo específico fue diferente para cada una dependiendo de los datos que eran importantes recabar en el momento de su aplicación.

Durante la evaluación académica uno de los puntos de interés fue verificar que la descripción de los patrones basándose en el concepto y formato de un patrón de producto (Sanchez-Segura et al., 2010) era lo suficientemente clara de entender para el equipo ágil, en particular los desarrolladores. También fue importante conocer las complicaciones que significaba para los desarrolladores seguir como guía la información contenida en el patrón para incorporar las técnicas de usabilidad en su proceso de desarrollo ágil. En esta etapa únicamente fue evaluado el patrón “Agile persona”. Sin embargo, los datos obtenidos contribuyeron de manera relevante a identificar modificaciones importantes para mejorar la definición de ambos patrones tanto en el formato como en su contenido antes de evaluar su uso en proyectos reales de desarrollo. Uno de los dos grupos de estudiantes que utilizaron el patrón fue considerado el proyecto piloto al cubrir las características propias de aquellos equipos en los que serían aplicados los patrones en desarrollos reales. El tomar en cuenta un proyecto piloto significó una importante fuente de retroalimentación en cuanto al proceso de solución considerado en el patrón. El segundo grupo de

estudiantes proporcionó retroalimentación sobre todo en cuanto a la claridad de la información contenida en el patrón.

Las adecuaciones identificadas como relevantes fueron aplicadas durante el proceso de desarrollo en curso de los grupos de estudio, lo cual dio la oportunidad que los equipos lo usaran con varias mejoras ya incluidas dando lugar a retroalimentación incluso sobre el patrón actualizado. Las principales modificaciones derivadas de los resultados de esta evaluación se aplicaron en la solución propuesta del patrón. El diagrama solución fue modificado renombrando procesos, cambiando el orden en que se realizaban, eliminando y agregando unos más para que resultará más sencillo de entender y seguirlo por el equipo para usar la técnica persona. Esta mejora implicó también modificar la descripción de la solución propuesta en el patrón.

Analizando cualitativamente los datos obtenidos de las revisiones e informes de avance de los equipos y comentarios expresos de los participantes esta primera etapa de validación mostro que la descripción, utilizando el formato de un patrón de producto, resultó adecuada para los equipos que lo utilizaron. Por otro lado, conocer las complicaciones que los equipos enfrentaron al utilizar el patrón como guía y aplicar las mejoras pertinentes durante el mismo proceso dio como resultado que el contenido del patrón se validara dentro de la misma evaluación y fuera más claro para los desarrolladores ágiles.

En la segunda etapa de la validación ambos patrones “Agile Persona” y “Agile Usability Testing” fueron usados en proyectos reales de desarrollo ágil. El objetivo de esta fase fue verificar que los patrones propuestos efectivamente proporcionan una guía fácil y sencilla a los desarrolladores para usar las técnicas persona y usability testing. Contribuyendo con lo anterior a la mejora en el nivel de usabilidad del software en construcción. Además, nuevas mejoras para aplicar a los patrones y posibles líneas futuras de estudio fueron identificados a partir de los resultados de esta validación en desarrollos ágiles de software. Se definieron tres aspectos que permitieran analizar cualitativamente el cumplimiento del objetivo para esta fase de evaluación.

El primer aspecto se enfocó a evaluar lo fácil de entender que resultan los patrones para los desarrolladores, la evaluación académica proporcionó evidencia que el producto patrón “Agile persona” fue fácil de entender sin embargo se consideró importante evaluar este punto en proyectos reales para ambos patrones. Al analizar los resultados fue evidente que se presenta cierta complicación de entender la estructura de los patrones al inicio de su uso independientemente del contexto en el que se utilizaron. El patrón “Agile persona” fue evaluado por un 23% de los desarrolladores como complicado de entender y el patrón “Agile Usability Testing” fue percibido de esta manera por un 15% de los desarrolladores. Por lo tanto, el taller previo a la utilización de los patrones es un apoyo importante para la correcta comprensión tanto de su estructura como de la importancia y objetivo de las técnicas de usabilidad que abarca el uso de cada uno de ellos. Como analizamos en la sección anterior los equipos ágiles que desarrollaron en un ambiente universitario reportaron mayor dificultad en entender los patrones. Derivado del análisis a este aspecto se observa que los desarrolladores ágiles se enfocan principalmente en ciertos campos como: Solution, example y template al ser estos los que contienen la información sobre quién, cómo y cuándo aplicar la técnica. Buscando que toda la información contenida en los patrones tenga relevancia para el desarrollador que los utiliza, es necesario revisar la descripción de algunos campos y analizar la necesidad de mejorarla para que su comprensión sea más clara y precisa o prescindir de ellos. También observamos que una vez que los desarrolladores se familiarizan con los patrones adquieren mayor habilidad para aplicarlos y aunque recurren menos a ellos no dejan de ser una guía a la que acuden para consultar cualquier duda o información.

La efectividad de los patrones como guías fue el segundo aspecto evaluado. A diferentes escalas todos los desarrolladores que participaron en la evaluación consideran que ambos patrones facilitaron la aplicación de la técnica de usabilidad para la que cada uno fue definido, En cuanto al atraso generado en el proceso de desarrollo al utilizar los patrones, un 48% de los desarrolladores consideró que el patrón “Agile persona” genera atraso y un 28% percibió atraso al utilizar el patrón “Agile Usability Testing”. A pesar del atraso que pudiera generar el utilizar los patrones estos facilitaron la aplicación de las técnicas

obteniendo el resultado objetivo de cada una de ellas. Un 95% percibió que el patrón “Agile Persona” permitió definir de manera más exacta los perfiles de los usuarios finales. Por otro lado 95% de los desarrolladores aplicó las pruebas en el momento adecuado al utilizar el patrón “Agile Usability Testing”. Nuevas mejoras en el proceso de solución fueron identificadas para que la descripción de esta sea más clara al seguirla.

Un tercer aspecto fue definido para analizar la mejora en el nivel de usabilidad del software en cuyo proceso de desarrollo se incorporaron las técnicas persona y usability testing utilizando los patrones correspondientes. De acuerdo al análisis de resultados discutido en la sección anterior, 95% de los desarrolladores percibieron que al incorporar las técnicas de usabilidad en su entorno ágil las aplicaciones fueron más fáciles de utilizar por los usuarios. Por otro lado, todos los desarrolladores indicaron que las funcionalidades implementadas cubrieron de manera más precisa las necesidades reales del usuario. Aunque esto último pudiera parecer contradictorio con el hecho de que la mayoría percibió cambios en las funcionalidades por mala interpretación de requerimientos cabe recordar la dificultad de uso que significó el utilizar los patrones en inicio, situación que pudiera haber influido. Los resultados en general indican un buen nivel de usabilidad percibido en las aplicaciones del software construido.

Aunque el patrón “Agile usability testing” no fue utilizado durante la evaluación académica fue el que mejores valoraciones y comentarios obtuvo por parte de los desarrolladores en proyectos reales como pudimos observar en la sección 6.4.5.

La etapa de evaluación en proyectos reales proporcionó un panorama más palpable sobre la contribución del uso de los patrones propuestos. Permitió corroborar su efectividad como guías para usar las dos técnicas de usabilidad en un entorno de desarrollo ágil mejorando o incorporando un nivel de usabilidad en los productos software construidos bajo este enfoque. La retroalimentación por parte de los desarrolladores proporciona información suficiente para aplicar nuevas mejoras que permitan todavía mayor claridad y facilidad al seguir los patrones. También dio lugar a puntos de interés que podrían tomarse en cuenta para líneas de estudio futuras. Los desarrolladores que utilizaron los patrones en proyectos reales pero construidos en un ambiente universitario percibieron en menor grado los beneficios del uso de los patrones para usar las técnicas de usabilidad. El estudio de los factores que pudieran influir en este resultado es un tema cuyos resultados aportarían información para un mayor conocimiento sobre posibles restricciones (forces) para aplicar los patrones. También permitirían identificar áreas de oportunidad para un mayor aprovechamiento de los patrones como guías reutilizables en un entorno ágil de cualquier contexto.

Capítulo 7 Conclusiones y futuras líneas de investigación

7.1 Conclusiones

El presente trabajo de investigación busca aportar soluciones que contribuyan a abordar el problema de la falta de mecanismos de información que apoyen a los desarrolladores en la integración de prácticas de usabilidad en sus desarrollos ágiles. Para ello el objetivo de esta investigación ha sido verificar que es posible identificar y definir patrones que sirvan a los equipos ágiles para incorporar determinadas técnicas de usabilidad en sus desarrollos ágiles.

Este trabajo de investigación se ha centrado en dos técnicas de usabilidad. Para seleccionarlás la revisión de la literatura realizada corroboró los resultados presentados por otros autores identificando las técnicas HCI más utilizadas dentro del contexto ágil. El estudio detallado de dichas técnicas abordó características como el momento de uso de la técnica, nivel de involucramiento de usuarios finales al utilizarla, perfil del diseñador de interfaz responsable de aplicar la técnica y el objetivo de su uso. Como resultado las técnicas HCI que se han abordado en este trabajo de investigación son:

- Persona
- Usability Testing

La técnica persona tiene especial interés por ser una de las herramientas que sirven como apoyo para conocer las necesidades y metas del usuario permitiendo documentar información relevante sobre éste en inicio y para identificar requisitos durante el desarrollo del proyecto. Inicialmente permite conocer al usuario y definir el perfil correspondiente sobre el que se enfocarán los esfuerzos de interacción en un diseño con el usuario en mente. Durante el desarrollo del proyecto persona también es útil para verificar que el diseño de la interacción considerada para las funcionalidades del software está de acuerdo a lo que el usuario realmente espera y necesita.

Por otro lado, las pruebas de usabilidad durante el proceso del desarrollo son una actividad medular para la disciplina HCI. Esta técnica destaca una participación importante del usuario al momento de aplicarla, usuarios finales de preferencia. Aunque es evidente la dificultad que representa la participación de usuarios reales que finalmente interactuaran con el software, esta característica es un factor relevante en dicha técnica.

Como método para documentar las buenas prácticas al incorporar estas dos técnicas en un desarrollo ágil utilizamos el concepto de patrón. Los patrones definidos para las dos técnicas persona y usability testing se describieron utilizando inicialmente el formato propuesto por (Sanchez-Segura et al., 2010) de esta manera se dio forma a las propuestas de reutilización para incorporar estas técnicas dentro de un entorno ágil.

Las validaciones de los patrones aplicadas en un ambiente real de desarrollo aunque en contextos diferentes proporcionaron información relevante para nuestro estudio que nos permite afirmar:

- Es posible identificar y describir patrones que sirvan de apoyo como guía al equipo de desarrollo ágil para incorporar técnicas HCI durante el proceso de desarrollo de software.
- El contar con líneas guía para implementar técnicas HCI resulta un apoyo significativo para los desarrolladores ágiles en la incorporación del atributo usabilidad durante la construcción del producto software. Independientemente de la organización de roles definida en el equipo, el uso de las guías permite a todos los elementos de éste conocer la utilidad de cada una de las técnicas descritas y aplicarlas sin mayor complicación.
- Utilizando información guía los desarrolladores son capaces de usar técnicas HCI en el desarrollo ágil de software sin que sea un impedimento el no contar con la preparación o experiencia de un diseñador experto HCI.

Adicionalmente mediante esta investigación se observaron los siguientes puntos:

- Una vez que el equipo utiliza y va reutilizando los patrones, la información contenida en ellos resulta más común por lo que la dificultad para incorporar las técnicas de usabilidad se minimiza en sobremanera.
- En determinado momento la incorporación de las técnicas se va dando de manera natural dentro del proceso ágil de software sin necesidad de que los patrones sean consultados de manera constante.
- Los desarrolladores del equipo ágil se apoyan en los patrones para proporcionar a los usuarios que participan en el proceso información que les permita visualizar con mayor claridad el objetivo y la forma en la que se realizan las actividades al aplicar las técnicas HCI.
- La experiencia de los desarrolladores interactuando con usuarios reales muestra impacto en el grado de importancia atribuida a la necesidad de considerar un buen nivel de usabilidad y por ende el uso de las técnicas HCI necesarias para ello.
- El apoyo organizacional resultó primordial para incorporar las técnicas en el proceso de desarrollo y para motivar la participación de los desarrolladores y usuarios.

En resumen, los patrones sirven como guía a los desarrolladores ágiles en la aplicación de técnicas HCI durante el proceso de desarrollo ágil. Si bien el uso de los patrones implica un sobre coste de tiempo y esfuerzo principalmente cuando empiezan a utilizarse, a medida que los desarrolladores avanzan en sus proyectos obteniendo experiencia en el uso de las técnicas, los beneficios se ven reflejados en esos mismos factores. Es importante señalar que el nivel de usabilidad de un software no depende únicamente de la aplicación de las dos técnicas descritas en nuestro estudio: persona y usability testing. Además de la variedad de técnicas que pueden aplicarse para contribuir a que un software sea usable también influyen aspectos organizativos como el presupuesto e interés por parte de la organización para promover la usabilidad.

7.2 Líneas futuras de investigación

Dado el amplio panorama que aún se tiene para lograr la incorporación de usabilidad en el proceso de desarrollo utilizando metodologías ágiles se planean varias líneas de investigación a desarrollar a partir de lo estudiado y obtenido en el presente trabajo:

Validar las actualizaciones aplicadas a los patrones persona y usability testing. Una vez actualizados los patrones de acuerdo a la retroalimentación obtenida en una primera validación en proyectos reales, descrita en este trabajo, el aplicar un siguiente proceso de validación aportará mayor información para confirmar la efectividad y eficiencia de los patrones descritos. Con el objetivo de continuar con la obtención de información a partir del contexto real donde se visualiza el uso final de los patrones la validación se aplicará de nuevo en industria utilizando instrumentos y métricas definidos de acuerdo a lo observado en los resultados descritos en este trabajo.

Evaluar nuevos formatos para describir los patrones. Describir las técnicas utilizando el formato inicial elegido para este trabajo no significa que sea el más útil y recomendable para utilizar como guía. Las observaciones señalan que fue adecuado y no representó mayor problema el utilizarlo, sin embargo no está limitado a ser el más eficiente para los equipos ágiles.

Verificar el funcionamiento de los patrones en diferentes metodologías de desarrollo ágil. La estructura de la información contenida en los productos patrón fue diseñada tomando en cuenta características de diferentes metodologías ágiles por lo que su aplicación independientemente de la metodología ágil utilizada debe proporcionar el mismo nivel de resultados.

Aplicar una validación cuantitativa. Resulta de interés aplicar validaciones que en base a variables y parámetros medibles proporcionen datos cuantificables sobre los cuales se realicen comparativas para obtener un panorama más exacto sobre la efectividad y eficiencia de los patrones al incorporarlos en el entorno ágil.

Definir un lenguaje de patrones. Describir patrones de manera individual es muy útil, sin embargo al ser parte de un lenguaje de patrones se complementa el concepto y se potencia aún más los resultados dentro del contexto determinado. En este caso los proyectos ágiles de software. Por lo tanto es necesario identificar y describir patrones para otras técnicas HCI utilizadas en dicho entorno que contribuyan a la definición de un lenguaje de patrones y, de esta forma se obtenga una aportación integral para la incorporación de usabilidad dentro del contexto de desarrollo ágil.

Validar el lenguaje de patrones. Una vez definido el lenguaje de patrones obtener la retroalimentación correspondiente es de vital importancia. Aunque la validación es igual de importante y relevante cuando se aplica en un ambiente académico realizar la validación en proyectos reales aporta un panorama más amplio con mayor relevancia en la práctica sobre los resultados obtenidos.

Referencias

- Alexander, C. (1979). *A timeless Way of Building*. New York: Oxford University.
- Alexander, C., Ishikawa, S., & Silverstein, M. (1977). *A Pattern Language: Towns, Buildings, Construction*. Oxford University Press USA.
- Ambler, S. (2008). Tailoring Usability into Agile Software Development Projects. In E.-C. Law, E. Hvannberg, & G. Cockton (Eds.), *Maturing Usability SE - 4* (pp. 75–95). Springer London. http://doi.org/10.1007/978-1-84628-941-5_4
- Ambler, S. W. (2006). Agile Unified Process. Retrieved from <http://www.ambysoft.com/unifiedprocess/agileUP.html>
- Anderson, D., Augustine, S., Avery, C., Cockburn, A., Cohn, M., DeCarlo, D., ... Wysocki, R. (2005). Declaration of Interdependence. Retrieved June 20, 2011, from <http://www.pmdoi.org/>
- Anwar, S., Motla, Y. H., Siddiq, Y., Asghar, S., Hassan, M. S., & Khan, Z. I. (2014). User-centered design practices in scrum development process: A distinctive advantage? In *17th IEEE International Multi Topic Conference 2014* (pp. 161–166). <http://doi.org/10.1109/INMIC.2014.7097330>
- Assael, H. (1999). *Comportamiento del consumidor. Comportamiento del consumidor*. México, D.F: International Thomson Editores.
- Barksdale, J. T., & McCrickard, D. S. (2012). Software Product Innovation in Agile Usability Teams: An Analytical Framework of Social Capital, Network Governance, and Usability Knowledge Management. *Int. J. Agil. Extrem. Softw. Dev.*, 1(1), 52–77. <http://doi.org/10.1504/IJAESD.2012.048302>
- Barnum, C. M. (2010). *Usability Testing Essentials*. Morgan Kaufmann.
- Beck, K., & Andres, C. (2004). *Extreme Programming Explained: Embrace Change, second edition*. Addison-Wesley Professional.
- Beck, K., Beedle, M., Bennekum, A. van, Cockburn, A., Cunningham, W., & Fowler, M. (2000). Agile Manifesto. Retrieved from www.agilemanifesto.org
- Belchev, B., & Baker, P. (2009). Improving Obama Campaign Software: Learning from Users. In *Agile Conference, 2009. AGILE '09*. (pp. 395–399). <http://doi.org/10.1109/AGILE.2009.46>
- Bertholdo, A. P. O., da Silva, T. S., de O. Melo, C., Kon, F., & Silveira, M. S. (2014). Agile Usability Patterns for UCD Early Stages BT - Design, User Experience, and Usability. Theories, Methods, and Tools for Designing the User Experience: Third International Conference, DUXU 2014, Held as Part of HCI International 2014, Heraklion, Crete. In A. Marcus (Ed.), (pp. 33–44). Cham: Springer International Publishing. http://doi.org/10.1007/978-3-319-07668-3_4
- Bertholdo, A. P. O., Kon, F., & Gerosa, M. A. (2016). Agile Usability Patterns for User-Centered Design Final Stages. In *International Conference on Human-Computer Interaction* (pp. 433–444).
- Beyer, H., & Holtzblatt, K. (1998). *Contextual Design: A Customer-Centered Approach to*

- Systems Design* (Morgan Kau). San Diego, CA, USA, CA, USA.
- Bias, R. G., & Mayhew, D. J. (2005). *Cost-justifying Usability An Update for the Internet Age, Second Edition*. (M. Kaufmann, Ed.) (2nd ed.). San Francisco, CA, USA: Elsevier.
- Bornoe, N., & Stage, J. (2014). Usability Engineering in the Wild: How Do Practitioners Integrate Usability Engineering in Software Development? In S. Sauer, C. Bogdan, P. Forbrig, R. Bernhaupt, & M. Winckler (Eds.), *Human-Centered Software Engineering: 5th IFIP WG 13.2 International Conference, HCSE 2014, Paderborn, Germany, September 16-18, 2014. Proceedings* (pp. 199–216). Berlin, Heidelberg: Springer Berlin Heidelberg. http://doi.org/10.1007/978-3-662-44811-3_12
- Brhel, M., Meth, H., & Maedcher, A. (2015). Exploring principles of user-centered agile software development: A. *Information and Software Technology*, 61(C), 163–181.
- Broschinsky, D., & Baker, L. (2008). Using Persona with XP at LANDesk Software, an Avocent Company. In *Agile 2008 Conference* (pp. 543–548). IEEE. <http://doi.org/10.1109/Agile.2008.91>
- Brown, D. D. (2013). Five Agile UX Myths. *J. Usability Studies*, 8(3), 55–60.
- Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., & Stal, M. (1996). Pattern-oriented software architecture: a system of patterns. Chichester;New York; Wiley.
- Caballero, L., Moreno, A. M., & Seffah, A. (2016). How Agile Developers Integrate User-Centered Design Into Their Processes: A Literature Review. *International Journal of Software Engineering and Knowledge Engineering*, 26(8), 1175–1201. <http://doi.org/10.1142/S0218194016500418>
- Caballero, L., Moreno, A., & Seffah, A. (2014). Persona as a Tool to Involving Human in Agile Methods: Contributions from HCI and Marketing. In S. Sauer, C. Bogdan, P. Forbrig, R. Bernhaupt, & M. Winckler (Eds.), *Human-Centered Software Engineering SE - 20* (Vol. 8742, pp. 283–290). Springer Berlin Heidelberg. http://doi.org/10.1007/978-3-662-44811-3_20
- Carbon, R., Dörr, J., & Trapp, M. (2004). Focusing Extreme Programming on Usability. *Lecture Notes in Informatics*, 51, 147–152.
- Carroll, J. M. (2000). *Scenario-Based Design of Human-Computer Interactions*. (Mit Pr, Ed.).
- Carvajal, L. (2009). Usability-enabling Guidelines: A Design Pattern and Software Plug-in Solution. In *Proceedings of the Doctoral Symposium for ESEC/FSE on Doctoral Symposium* (pp. 9–12). New York, NY, USA: ACM. <http://doi.org/10.1145/1595782.1595786>
- Carvajal, L. (2012). *Usability-oriented software development process*. Universidad Politécnica de Madrid.
- Carvajal, L., Moreno, A. M., Sanchez-Segura, M.-I., & Seffah, A. (2013). Usability through Software Design. *Software Engineering, IEEE Transactions on*, 39(11), 1582–1596. <http://doi.org/10.1109/TSE.2013.29>
- Chamberlain, S., Sharp, H., & Maiden, N. (2006). Towards a framework for integrating agile development and user-centred design. In P. Abrahamsson, M. Marchesi, & G. Succi (Eds.), *Extreme Programming and Agile Processes in Software Engineering* (Vol. 4044, pp. 143–153). Springer Berlin. Retrieved from <http://oro.open.ac.uk/4928/>
- Cho, L. (2009). Adopting an Agile Culture A User Experience Team's Journey. In *2009 Agile*

- Conference (pp. 416–421). IEEE. <http://doi.org/10.1109/AGILE.2009.76>
- Coad, P., Luca, J. de, & Lefebvre, E. (1999). *Java Modeling Color with Uml: Enterprise Components and Process with Cdrom* (1st ed.). Upper Saddle River, NJ, USA: Prentice Hall PTR.
- Coatta, T., & Gosper, J. (2010). UX Design and Agile: A Natural Fit? *Queue*, 8(11), 50:50--50:58. <http://doi.org/10.1145/1874534.1891739>
- Cockburn, A. (2007). *Agile software development. The Cooperative Game*. Indianapolis: Addison Wesley.
- Constantine, L. L. (2002). Process agility and software usability: Toward lightweight usage-centered design. *Information Age*, 1(June 2001), 1–10. Retrieved from <http://informatix.ida.liu.se/~TDP010/info/material/agiledesign4.pdf>
- Cooper, A. (1999). *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity*. (Sams, Ed.)*The Inmates are Running the Asylum Why High Tech Products Drive us Crazy and How to Restore the Sanity*. Sams. http://doi.org/10.1007/978-3-322-99786-9_1
- Cooper, A., Reimann, R., & Cronin, D. (2007). *About Face 3. The essentials of Interaction Design*. Wiley Publishing. Retrieved from www.wiley.com
- d. Silva, T. S., Silveira, M. S., & Maurer, F. (2015). Usability Evaluation Practices within Agile Development. In *2015 48th Hawaii International Conference on System Sciences* (pp. 5133–5142). <http://doi.org/10.1109/HICSS.2015.607>
- Da Silva, T., Martin, A., Maurer, F., & Silveira, M. (2011). User-Centered Design and Agile Methods: A Systematic Review. *2011 AGILE Conference*, 77–86. <http://doi.org/10.1109/AGILE.2011.24>
- Da Silva, T. S., Silveira, M., & Maurer, F. (2013). Ten Lessons Learned from Integrating Interaction Design and Agile Development. *Agile Conference (AGILE), 2013*. <http://doi.org/10.1109/AGILE.2013.11>
- Dayton, D., & Barnum, C. (2009). The impact of agile on user-centered design: Two surveys tell the story. *Technical Communication*, 56(3), 219–234. Retrieved from <http://www.scopus.com/inward/record.url?eid=2-s2.0-70349416876&partnerID=40&md5=c58457263fe68dce6d73ea713782437c>
- de Carvalho, J. M. I., da Silva, T. S., & Silveira, M. S. (2016). Agile and UCD Integration Based on Pre-development Usability Evaluations: An Experience Report. In M. Kurosu (Ed.), *Human-Computer Interaction. Theory, Design, Development and Practice : 18th International Conference, HCI International 2016, Toronto, ON, Canada, July 17-22, 2016. Proceedings, Part I* (pp. 586–597). Cham: Springer International Publishing. http://doi.org/10.1007/978-3-319-39510-4_54
- de Freitas, R., Rodrigues, L. A., & da Cunha, A. (2016). AGILUS: A Method for Integrating Usability Evaluations on Agile Software Development. In M. Kurosu (Ed.), *Human-Computer Interaction. Theory, Design, Development and Practice : 18th International Conference, HCI International 2016, Toronto, ON, Canada, July 17-22, 2016. Proceedings, Part I* (pp. 545–552). Cham: Springer International Publishing. http://doi.org/10.1007/978-3-319-39510-4_50
- Detweiler, M. (2007). Managing UCD within agile projects. *Interactions*, 14(3), 40–42. <http://doi.org/10.1145/1242421.1242447>

- Dubey, S., & Rana, A. (2010). Analytical roadmap to usability definitions and decompositions. *International Journal of Engineering Science and Technology*, 2(9), 4723–4729.
- Dumas, J. S., & Redish, J. C. (1999). *A practical guide to Usability Testing*. (I. Books, Ed.). Intellect Ltd.
- Eriksson, E., Cajander, Å., & Gulliksen, J. (2009). Hello World! -- Experiencing Usability Methods without Usability Expertise. In T. Gross, J. Gulliksen, P. Kotzé, L. Oestreicher, P. Palanque, R. O. Prates, & M. Winckler (Eds.), *Human-Computer Interaction -- INTERACT 2009: 12th IFIP TC 13 International Conference, Uppsala, Sweden, August 24-28, 2009, Proceedings, Part II* (pp. 550–565). Berlin, Heidelberg: Springer Berlin Heidelberg. http://doi.org/10.1007/978-3-642-03658-3_60
- Federoff, M., & Courage, C. (2009). Successful User Experience in an Agile Enterprise Environment. *Proceedings of the Symposium on Human Interface 2009*, 5617(October 2006), 233–242. <http://doi.org/10.1007/978-3-642-02556-3>
- Felker, C., Slamova, R., & Davis, J. (2012). Integrating UX with scrum in an undergraduate software development project. In *Proceedings of the 43rd ACM technical symposium on Computer Science Education* (pp. 301–306). New York, NY, USA: ACM. <http://doi.org/10.1145/2157136.2157226>
- Ferreira, J., Noble, J., & Biddle, R. (2007a). Agile Development Iterations and UI Design. *Agile 2007 Agile 2007*, 50–58. <http://doi.org/10.1109/AGILE.2007.8>
- Ferreira, J., Noble, J., & Biddle, R. (2007b). Interaction Designers on eXtreme Programming Teams : Two Case Studies from the Real World. In *NZCSRSC '07 5th New Zealand Computer Science Research Student Conference*, 1–8. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.89.4453>
- Fisher, K. G., & Bankston, A. From Cradle to Sprint: Creating a Full-Lifecycle Request Pipeline at Nationwide Insurance, 2009 Agile Conference 223–228 (2009). Ieee. <http://doi.org/10.1109/AGILE.2009.72>
- Fowler, M. (1996). *Analysis patterns: reusable object models*. Boston, Massachusetts: Addison-Wesley.
- Fox, D., Sillito, J., & Maurer, F. (2008). Agile methods and user-centered design: How these two methodologies are being successfully integrated in industrys. In *Proceedings - Agile 2008 Conference* (pp. 63–72). <http://doi.org/10.1109/Agile.2008.78>
- Frank, A., & Hartel, C. (2009). Feature Teams Collaboratively Building Products from READY to DONE. In *Agile Conference, 2009. AGILE '09*. (pp. 320–325). <http://doi.org/10.1109/AGILE.2009.51>
- Gamma, E., Helm, R., Johnson, R., Vlissides, J. M., & Booch, G. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Old Tappan;Boston; Addison Wesley Professional.
- Gonçalves, J., & Santos, C. (2011). POLVO - Software for prototyping of low-fidelity interfaces in agile development. In J. A. Jacko (Ed.), *Human-Computer Interaction, Part I, HCI 2011, LNCS 6761* (Vol. 6761 LNCS, pp. 63–71). Springer-Verlag Berlin Heidelberg.
- Goodwin, N. C. (1987). Functionality and Usability. *Commun. ACM*, 30(3), 229–233. <http://doi.org/10.1145/214748.214758>
- Gould, J. D., & Lewis, C. (1985). Designing for Usability: Key Principles and What Designers Think. *Commun. ACM*, 28(3), 300–311. <http://doi.org/10.1145/3166.3170>

- Gulliksen, J., Göransson, B., Boivie, I., Blomkvist, S., Persson, J., & Cajander, Å. (2003). Key principles for user-centred systems design. *Behaviour Information Technology*, 22(6), 397–409. <http://doi.org/10.1080/01449290310001624329>
- Haikara, J. (2007). Usability in Agile Software Development: Extending the Interaction Design Process with. *Lecture Notes in Computer Science*.
- Hakim, J., Spitzer, T., & Armitage, J. (2003). Sprint: Agile specifications in Shockwave and Flash. In *Proceedings of the 2003 Conference on Designing for User Experiences DUX 03* (pp. 1–14). <http://doi.org/10.1145/997078.997111>
- Hao, H. M., & Jaafar, A. (2011). Usability in practice: Perception and practicality of management and practitioners. In *2011 International Conference on Pattern Analysis and Intelligence Robotics* (Vol. 2, pp. 211–216). <http://doi.org/10.1109/ICPAIR.2011.5976928>
- Hudson, W. (2003). Adopting user-centered design within an agile process: A conversation. *Cutter IT Journal*, 16(10), 5–12.
- Hussain, Z., Lechner, M., Milchrahm, H., Shahzad, S., Slany, W., Umgeher, M., & Wolkerstorfer, P. (2008a). Agile User-Centered Design Applied to a Mobile Multimedia Streaming Application. *HCI and Usability for Education and Work*, 5298(2008), 313–330. <http://doi.org/10.1007/978-3-540-89350-9>
- Hussain, Z., Lechner, M., Milchrahm, H., Shahzad, S., Slany, W., Umgeher, M., & Wolkerstorfer, P. (2008b). Integrating Extreme Programming and User-Centered Design.
- Hussain, Z., Lechner, M., Milchrahm, H., Shahzad, S., Slany, W., Umgeher, M., ... Wolkerstorfer, P. (2008). User Interface Design for a Mobile Multimedia Application: An Iterative Approach. In *Advances in Computer-Human Interaction, 2008 First International Conference on* (pp. 189–194). <http://doi.org/10.1109/ACHI.2008.24>
- Hussain, Z., Milchrahm, H., Shahzad, S., Slany, W., Tscheligi, M., & Wolkerstorfer, P. (2009). Integration of Extreme Programming and User-Centered Design: Lessons Learned. *Agile Processes in Software Engineering and Extreme Programming*, 31(3), 174–179. <http://doi.org/10.1007/978-3-642-01853-4>
- IEEE. (1998). Standard for a Software Quality Metrics Methodology. IEEE Std 1061.
- Illmensee, T., & Muff, A. 5 Users Every Friday: A Case Study in Applied Research, 2009 Agile Conference 404–409 (2009). Ieee. <http://doi.org/10.1109/AGILE.2009.45>
- ISO/IEC 25010. (2011). ISO/IEC 25010:2011 Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE). ISO/IEC 25010:2011.
- ISO 9241-11. (1998). ISO 9241-11:1998 Ergonomic requirements for office work with visual display terminals (VDTs). Part 11. Guidance on Usability.
- ISO 9241-210. (2010). ISO 9241-210:2010 Ergonomics for human-system interaction. Part 210. Human-Centered Design for Interactive Systems.
- Isomursu, M. A., Sirotkin, A. A., Voltti, P. B., & Halonen, M. B. (2012). User experience design goes agile in lean transformation - A case study. In *Proceedings 2012 Agile Conference Agile 2012* (pp. 1–10). <http://doi.org/10.1109/Agile.2012.10>
- Jeffrey, R., Chisnell, D., & Spool, J. (2008). *Handbook of Usability Testing: How to Plan, Design and Conduct effective tests*. John Wiley & Sons.
- Jenkinson, A. (1994). Beyond Segmentation. *Journal of Targeting, Measurement and Analysis*

- for Marketing*, 3(1), 60–72.
- Jenkinson, A. (2009). What happened to strategic segmentation? *Journal of Direct, Data and Digital Marketing Practice*, 11(12), 124.
- Jia, Y., Larusdottir, M. K., & Cajander, A. (2012). The Usage of Usability Techniques in Scrum Projects. In *Proceedings of the 4th International Conference on Human-Centered Software Engineering* (pp. 331–341). Berlin, Heidelberg: Springer-Verlag. http://doi.org/10.1007/978-3-642-34347-6_25
- Jokela, T., & Abrahamsson, P. (2004). Usability Assessment of an Extreme Programming Project: Close Co-operation with the Customer Does Not Equal to Good Usability. In F. Bomarius & H. Iida (Eds.), *Product Focused Software Process Improvement SE - 28* (Vol. 3009, pp. 393–407). Springer Berlin Heidelberg. http://doi.org/10.1007/978-3-540-24659-6_28
- Jurca, G., Hellmann, T. D., & Maurer, F. (2014). Integrating Agile and User-Centered Design: A Systematic Mapping and Review of Evaluation and Validation Studies of Agile-UX. In *Agile Conference (AGILE), 2014* (pp. 24–32). <http://doi.org/10.1109/AGILE.2014.17>
- Juristo, N., Moreno, A. M., & Sanchez-Segura, M.-I. (2007). Analysing the impact of usability on software design. *Journal of Systems and Software*, 80(9), 1506–1516. <http://doi.org/http://dx.doi.org/10.1016/j.jss.2007.01.006>
- Kane, D. (2003). Finding a Place for Discount Usability Engineering in Agile Development: Throwing Down the Gauntlet. In *Proceedings of the Conference on Agile Development* (p. 40--). Washington, DC, USA: IEEE Computer Society. <http://doi.org/10.1109/ADC.2003.1231451>
- Kodaganallur, V., & Shim, S. (2006). Analysis Patterns: A Taxonomy and its Implications. *Information Systems Management*, 23(3), 52–61.
- Kropp, E., & Koischwitz, K. (2016). Experiences with User-Centered Design and Agile Requirements Engineering in Fixed-Price Projects. In A. Ebert, S. R. Humayoun, N. Seyff, A. Perini, & S. D. J. Barbosa (Eds.), *Usability- and Accessibility-Focused Requirements Engineering: First International Workshop, UsARE 2012, Held in Conjunction with ICSE 2012, Zurich, Switzerland, June 4, 2012 and Second International Workshop, UsARE 2014, Held in Conjunction with RE 2014, Karlskrona, Sweden, August 25, 2014, Revised Selected Papers* (pp. 47–61). Cham: Springer International Publishing. http://doi.org/10.1007/978-3-319-45916-5_4
- Krug, S. (2005). *Don't Make me Think! A common Sense Approach to web usability, Second edition*. New Riders.
- Laubheimer, P., & Loranger, H. (2017). Accounting for UX Work with User Stories in Agile Projects. Retrieved March 15, 2017, from <https://www.nngroup.com/articles/ux-user-stories/>
- Lazar, J., Feng, J. H., & Hochheiser, H. (2010). *Research Methods in Human-Computer Interaction*. Wiley Publishing.
- Lee, J. C. (2011). Evaluating eXtreme Scenario-based Design in a Distributed Agile Team. *Development*, 863–877. <http://doi.org/10.1145/1979742.1979681>
- Lee, J. C., & McCrickard, D. S. (2007). Towards Extreme(ly) Usable Software: Exploring Tensions Between Usability and Agile Software Development. *Agile 2007 Agile 2007*, 59–71. <http://doi.org/10.1109/AGILE.2007.63>

- Lee, J. C., McCrickard, D. S., & Stevens, K. T. (2009). Examining the Foundations of Agile Usability with eXtreme Scenario-Based Design. *2009 Agile Conference*. Ieee. <http://doi.org/10.1109/AGILE.2009.30>
- Losada, B., Urretavizcaya, M., López, J.-M., & Fernández-Castro, I. (2013). Applying usability engineering in InterMod agile development methodology. A case study in a mobile application. *Journal of Universal Computer Science*, 19(8), 1046–1065.
- Loudon, D. L., & Della Bitta, A. J. (1979). *Consumer behavior: concepts and applications*. New York: McGraw-Hill.
- Magües, D. A., Castro, J. W., & Acuña, S. T. (2016a). HCI usability techniques in agile development. In *2016 IEEE International Conference on Automatica (ICA-ACCA)* (pp. 1–7). <http://doi.org/10.1109/ICA-ACCA.2016.7778513>
- Magües, D. A., Castro, J. W., & Acuña, S. T. (2016b). Usability in agile development: A systematic mapping study. In *2016 XLII Latin American Computing Conference (CLEI)* (pp. 1–8). <http://doi.org/10.1109/CLEI.2016.7833347>
- Martin, R. C. (2003). *Agile Software Development. Principles, patterns and practices*. Prentice Hall.
- Mayhew, D. J. (1999). *The usability engineering LifeCycle*. Morgan Kaufmann Publishers, Inc.
- McGinn, J. J., & Chang, A. R. (2013). RITE+Krug: A Combination of Usability Test Methods for Agile Design. *J. Usability Studies*, 8(3), 61–68. Retrieved from <http://dl.acm.org/citation.cfm?id=2817915.2817917>
- McInerney, P., & Maurer, F. (2005). UCD in Agile Projects: Dream Team or Odd Couple? *Interactions*, 12(6), 19–23. <http://doi.org/10.1145/1096554.1096556>
- Medlock, M., Dixon D., T. M., Romero, R., & Fulton, B. (n.d.). Using the RITE method to improve products; a definition and a case study. Retrieved from <https://www.microsoft.com/en-us/Search/result.aspx?q=rapid+iterative&x=0&y=0>
- Mommel, T., Gundelsweiler, F., & Reiterer, H. (2007). Agile Human-centered Software Engineering. In *Proceedings of the 21st British HCI Group Annual Conference on People and Computers: HCI...But Not As We Know It - Volume 1* (pp. 167–175). Swinton, UK, UK: British Computer Society. Retrieved from <http://dl.acm.org/citation.cfm?id=1531294.1531317>
- Meszaros, G., & Aston, J. (2006). Adding usability testing to an agile project. *AGILE 2006 AGILE06*. IEEE. <http://doi.org/10.1109/AGILE.2006.5>
- Miller, L. Case study of customer input for a successful product, Agile Development Conference ADC05 225–234 (2005). IEEE Comput. Soc. <http://doi.org/10.1109/ADC.2005.16>
- Moreno, A. M., Seffah, A., Capilla, R., & Sanchez-Segura, M.-I. (2013). HCI Practices for Building Usable Software. *Computer*, 46(4), 100–102. <http://doi.org/10.1109/MC.2013.133>
- Najafi, M., & Toyoshiba, L. (2008). Two Case Studies of User Experience Design and Agile Development. In *Agile 2008 Conference* (pp. 531–536). IEEE Press. <http://doi.org/10.1109/Agile.2008.67>
- Nielsen, J. (1993). *Usability Engineering*.

- Nielsen, J. (2000). Why you only need to test with five users. Retrieved from <https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>
- Nielsen, J. (2008). Agile Development Projects and Usability. Retrieved from <https://www.nngroup.com/articles/agile-development-and-usability/>
- Nielsen, J., Berger, J. M., Gilutz, S., & Whitenon, K. (2012). *Return on Investment (ROI) for Usability* (4th ed.). Nielsen Norman Group.
- Nielsen, J., & Hinderer Sova, D. (2003). How to recruit participants for usability studies. Retrieved from <http://www.nngroup.com/reports/how-to-recruit-participants-usability-studies/>
- Nielsen, L., & Madsen, S. (2012). The Usability Expert's Fear of Agility: An Empirical Study of Global Trends and Emerging Practices. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design* (pp. 261–264). New York, NY, USA: ACM. <http://doi.org/10.1145/2399016.2399057>
- Norman, D. (2016). Don Norman: The term “UX.” Retrieved July 1, 2016, from <https://www.youtube.com/watch?v=9BdtGjoIN4E>
- Norman, D., & Nielsen, J. (2014). The definition of “User Experience.” Retrieved June 20, 2007, from <https://www.nngroup.com/articles/definition-user-experience/>
- Obendorf, H., & Finck, M. (2008). Scenario-based usability engineering techniques in agile development processes. *Proceeding of the Twentysixth Annual CHI Conference Extended Abstracts on Human Factors in Computing Systems CHI 08*, 2159. <http://doi.org/10.1145/1358628.1358649>
- Ogunyemi, A. A., Lamas, D., Adagunodo, E. R., Loizides, F., & Rosa, I. B. Da. (2016). Theory, Practice and Policy: An Inquiry into the Uptake of HCI Practices in the Software Industry of a Developing Country. *International Journal of Human-Computer Interaction*, 32(9), 665–681. <http://doi.org/10.1080/10447318.2016.1186306>
- Ovad, T., & Larsen, L. (2015). The Prevalence of UX Design in Agile Development Processes in Industry. In *2015 Agile Conference* (pp. 40–49). <http://doi.org/10.1109/Agile.2015.13>
- Patton, J. (2002). Designing Requirements: Incorporating Usage-Centered Design into an Agile SW Development Process. In D. Williams (Ed.), *XP/Agile Universe, LNCS 2418* (Vol. 2418, pp. 1–12). Springer-Verlag Berlin Heidelberg.
- Patton, J. (2005). Finding the forest in the trees. In *Companion to the 20th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications* (pp. 266–274). New York, NY, USA: ACM. <http://doi.org/10.1145/1094855.1094961>
- Peres, A. L., Silva, T. S. D., Silva, F. S., Soares, F. F., Carvalho, C. R. M. D., & Meira, S. R. D. L. (2014). AGILEUX Model: Towards a Reference Model on Integrating UX in Developing Software Using Agile Methodologies. In *2014 Agile Conference* (pp. 61–63). <http://doi.org/10.1109/AGILE.2014.15>
- Promise - PPL. (2009). Retrieved from <http://kovachi.sel.inf.uc3m.es/800-spanish>
- Pruitt, J., & Adlin, T. (2010). *The Essential Persona Lifecycle: Your Guide to Building and Using Personas*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Rahim, W. A., Isa, W. M., Lokman, A. M., Taharim, N. F., & Wahid, N. D. (2014). Engineering M-Learning Using Agile User-Centered Design. In *2014 Eighth International Conference*

- on Next Generation Mobile Apps, Services and Technologies (pp. 60–65).
<http://doi.org/10.1109/NGMAST.2014.46>
- Ring, A., & Benvenuto, A. (2008). The power of nuance Admap. *World Advertising Centre*, 43(495), 28–30. Retrieved from <http://www.rga.com/news/articles/power-of-nuance/>
- Ritter, F. E., Baxter, G. D., & Churchill, E. F. (2014). Foundations for Designing User-Centered Systems: What System Designers Need to Know about People. London: Springer London.
<http://doi.org/10.1007/978-1-4471-5134-0>
- Rivero, J. M., Luna, E. R., Grigera, J., & Rossi, G. (2013). Improving user involvement through a model-driven requirements approach. In *2013 3rd International Workshop on Model-Driven Requirements Engineering (MoDRE)* (pp. 20–29).
<http://doi.org/10.1109/MoDRE.2013.6597260>
- Rubin, K. S. (2012). Essential Scrum: a practical guide to the most popular agile process . Upper Saddle River, NJ: Addison-Wesley .
- Runeson, P., Höst, M., Rainer, A., & Regnell, B. (2012). *Background and Definition of Concepts, in Case Study Research in Software Engineering: Guidelines and Examples*. (I. John Wiley & Sons, Ed.). Hoboken, NJ, USA. <http://doi.org/10.1002/9781118181034.ch2>
- Salah, D., Paige, R., & Cairns, P. (2015). Patterns for Integrating Agile Development Processes and User Centred Design. In *Proceedings of the 20th European Conference on Pattern Languages of Programs* (p. 19:1--19:10). New York, NY, USA: ACM.
<http://doi.org/10.1145/2855321.2855341>
- Salah, D., Paige, R. F., & Cairns, P. (2014). A Systematic Literature Review for Agile Development Processes and User Centred Design Integration. In *Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering* (p. 5:1--5:10). New York, NY, USA: ACM. <http://doi.org/10.1145/2601248.2601276>
- Salah, D., Petrie, H., & Paige, R. (2009). Towards a Framework for Bridging User-Centred Design and Agile Software Development Processes. In *3rd Irish HCI Conference 2009*.
- Salvador, C., Nakasone, A., & Pow-Sang, J. A. (2014). A Systematic Review of Usability Techniques in Agile Methodologies. In *Proceedings of the 7th Euro American Conference on Telematics and Information Systems* (p. 17:1--17:6). New York, NY, USA: ACM.
<http://doi.org/10.1145/2590651.2590668>
- Sanchez-Segura, M.-I., Medina-Dominguez, F., de Amescua, A., & Mora-Soto, A. (2010). Improving the efficiency of use of software engineering practices using product patterns. *Information Sciences*, 180(14), 2721–2742. <http://doi.org/10.1016/j.ins.2010.03.028>
- Scheiber, F., Wruk, D., Oberg, A., Britsch, J., Woywode, M., Maedche, A., ... Plach, M. (2012). Software Usability in Small and Medium Sized Enterprises in Germany: An Empirical Study. In A. Maedche, A. Botzenhardt, & L. Neer (Eds.), *Software for People: Fundamentals, Trends and Best Practices* (pp. 39–52). Berlin, Heidelberg: Springer Berlin Heidelberg. http://doi.org/10.1007/978-3-642-31371-4_3
- Schwaber, K., & Beedle, M. (2002). *Agile Software Development with Scrum*. Prentice Hall.
- Seffah, A. (2015). *Patterns of HCI Design and HCI design of Patterns*. Springer.
<http://doi.org/10.1007/978-3-319-15687-3>
- Seffah, A., & Metzker, E. (2004). The Obstacles and Myths of Usability and Software Engineering. *Communications of the ACM*, 47, 71–76.

- Seffah, A., & Taleb, M. (2012). Tracing the evolution of HCI patterns as an interaction design tool. *Innovations in Systems and Software Engineering*. London : Springer-Verlag . <http://doi.org/10.1007/s11334-011-0178-8>
- Seffah, A., Vanderdonckt, J., & Desmarais, M. (2009). Human-Centered Software Engineering: Software Engineering Architectures, Patterns, and Models for Human Computer Interaction. In A. Seffah, J. Vanderdonckt, & M. Desmarais (Eds.), *Human-Centered Software Engineering SE - I* (pp. 1–6). Springer London. http://doi.org/10.1007/978-1-84800-907-3_1
- Sharp, H., Rogers, Y., & Preece, J. (2007). *Interaction Design beyond human-computer interaction*. John Wiley & Sons.
- Sharp, H., Rogers, Y., & Preece, J. (2015). *Interaction Design Beyond human-computer interaction*. John Wiley & Sons.
- SIGCHI. (n.d.). Curricula for Human-Computer Interaction. ACM. Retrieved from http://old.sigchi.org/cdg/cdg2.html#2_1
- Singh, M. (2008). U-SCRUM: An Agile Methodology for Promoting Usability. In *Agile 2008 Conference* (pp. 555–560). IEEE. <http://doi.org/10.1109/Agile.2008.33>
- Sohaib, O., & Khan, K. (2010). Integrating usability engineering and agile software development: A literature review. In *Computer Design and Applications (ICCD A), 2010 International Conference on* (Vol. 2, pp. V2-32-V2-38). <http://doi.org/10.1109/ICCD A.2010.5540916>
- Sommerville, I. (2011). *Ingeniería del Software*. Pearson Educación.
- Sy, D. (2007). Adapting Usability Investigations for Agile User-centered Design. *Journal of Usability Studies*, 2(3), 112–132.
- Tidwell, J. (2005). *Designing interfaces*. Sebastopol, CA: O'Reilly Media.
- Ungar, J., & White, J. (2008). Agile user centered design: enter the design studio - a case study. *Design*, 2167–2178. <http://doi.org/10.1145/1358628.1358650>
- VersionOne. (2017). *The 11th Annual State of Agile*. Retrieved from www.versionone.com
- Wale-Kolade, A., & Nielsen, P. A. (2016). Apathy Towards the Integration of Usability Work: A Case of System Justification. *Interacting with Computers*, 28(4), 437. <http://doi.org/10.1093/iwc/iwv016>
- Welie, M. Van. (2008). A Pattern Library for Interaction Design. Retrieved from www.welie.com
- Williams, H., & Ferguson, A. (2007). The UCD Perspective: Before and After Agile. In *Agile Conference (AGILE), 2007* (pp. 285–290). <http://doi.org/10.1109/AGILE.2007.61>
- Wolkerstorfer, P., Tscheligi, M., Sefelin, R., Milchrahm, H., Hussain, Z., Lechner, M., & Shahzad, S. (2008). Probing an agile usability process. *Proceeding of the Twentysixth Annual CHI Conference Extended Abstracts on Human Factors in Computing Systems CHI 08*, 2151. <http://doi.org/10.1145/1358628.1358648>

Anexos

Anexo A. Lista de estudios primarios de la Revisión de Literatura

Primary studies of the systematic literature review

Agile teams integrating HCI techniques into their development process

User Centered Design techniques working together with agile methodologies has been a widely issue studied over the last few years. As an outcome of this adoption of HCI techniques, agile developers had adapted these techniques as they had visualized the necessity in order to get better and most usable agile software. Our study aims to identify these different ways defined and adopted by agile teams integrating HCI techniques into their development process. Regarding our primary papers studied the table below provides detailed information about them.

Period of time: From 2000 – first trim 2017

First Author (Year)	Short Summary	Source	Classification	Agile Methodology	HCI technique(s) or artifacts	Application moment in agile	Team Collaboration
Patton (2002)	Author refers Alan Cooper definition about interaction design when he said it concept refers to the selection of behavior, function and information and their presentation to users. Patton presents a short list of interaction-design-centric guidelines.	Springer	IR	XP	Index cards Persona Prototype	Interaction design first	Cross-functional team
Constantine (2002)	This article describes a flexible, model-driven approach for engineering Web applications that succeeds through a focus on user interface design and usability.	IEEE / Scopus	IR	XP	Index cards Prototype Heuristic evaluation	Minimal Up-front Design	Independent Close Collaboration Team
Kane (2003)	Author explores in this paper the challenge for the agile development community to find ways to incorporate and gain value from such discount usability engineering practices.	IEEE	IR	Not specified	Usability Testing Scenarios Prototype Heuristic Evaluation	Not specified	Not specified
Constantine (2003)	The tutorial introduces the models and a method of usage-centered design and explores the integration of usage-centered approaches into agile approaches.	IEEE	IR	Not specified	Not specified	Not specified	Not specified
Beyer (2004)	This paper shows how one user-centered design method, Contextual Design (CD), forms a natural fit with agile methods. The authors describe a list of axioms-principles of agile process modified taking in account customer-centered design.	Springer	RP	XP	User stories Contextual inquiry Prototype Contextual design Story boarding	Overlapped Track	Close Collaboration
Jokela (2004)	The authors discuss the implications observed from their analysis which shows that in XP the team transfers the responsibility of the product's usability to the customer.	Scopus	RP	XP	User stories Acceptance test	Minimal Up-front Design	Independent Close Collaboration Team
Hodgetts (2005)	In this experience report, the author discusses his coaching	IEEE	IR	SCRUM & XP	User stories Scenarios Prototype	Overlapped Track	Cross-functional team

	experiences integrating sophisticated UED practices into the agile process initiatives of organizations.						
Miller (2005)	This paper describes one company's efforts to merge agile development and User Centered Design by creating interconnected parallel design and development tracks.	IEEE	IR	SCRUM & XP	User stories Contextual Inquiry Prototype	Overlapped Track	Close Collaboration
Maurer (2005)	The workshop looked at software engineering from a number of perspectives, including those of agile methods and communication theory, in order to point out solutions and conditions for human-centred software engineering.	Scopus	IR	Not specified	Not specified	Not specified	Not specified
Holzinger (2005)	The authors present ideas in how to combine XP and Usability engineering in an even more powerful development method called Extreme Usability (XU)	IEEE	RP	XP	User stories Scenarios Prototype	Overlapped Track	Cross-functional
Meszaros (2006)	This paper describes the process of deviated from the tried and accepted company practices for developing software, first by using agile methods, and second by introducing usability testing into said agile methods.	IEEE	IR	XP	Usability Testing Scenarios Prototype Story Boarding	Interaction design first	Close Collaboration
Chamberlain (2006)	The aim of the study was to develop a framework for use by project teams wishing to integrate UCD practices with agile development. The study, its findings and five principles for integrating UCD and agile development arising from this work are discussed.	Scopus	RP	SCRUM XP	User stories Persona Prototype	Interaction design first /Minimal up-front Design/Implicit interaction design	Close collaboration / None collaboration
Santana (2006)	This paper presents a streamlined approach to human-computer interaction design called extreme designing. Proposes to combine user interface sketches with a more structured representation such as an interaction model.	IEEE	RP	XP	User stories Scenarios Prototype	Implicit Interaction design	Cross-functional
Ferreira (2007)	The key results from this study are that agile iterations facilitates usability testing; allows software developers to incorporate results of those tests into subsequent iterations; and crucially, can significantly improve the quality of the relationship between UI designers and software	IEEE	RP	SCRUM XP	Usability Testing User stories Prototype	Interaction design first /Implicit interaction design /Overlapped track	Close collaboration/None collaboration

	developers.						
Detweiler (2007)	The author presents three iterative UCD phases and describes challenges and tips to each phase in order to improve the outcomes.	Scopus	IR	Not specified	Usability Testing Prototype	Overlapped Track	Close Collaboration
Lee (2007)	This work presents a development approach that draws from extreme programming (XP), a widely practiced agile software development process, and scenario-based design (SBD), an established usability engineering process	IEEE	RP	XP	User stories Scenarios	Minimal up-front design	Close Collaboration
Haikara (2007)	In this study, the interaction design process of Mobile-D of which practices are based on XP, is extended with personas. Integrating an explicit interaction design process into Agile Software Development.	Scopus	RP	XP	Persona	Interaction design first	Not specified
Budwig (2007)	This case study details our UX teams' experiences working with scrum, describing the challenges and issues that we faced, and the solutions that we implemented to resolve those issues.	Scopus	IR	SCRUM	User Stories	Overlapped Track	Cross-functional team
Duechting (2007)	The approach presented in this paper evaluates how agile software engineering models consider activities of usability engineering to ensure the creation of usable software products.	Scopus	RP	SCRUM&XP	Not specified	Not specified	Not specified
Ferguson (2007)	The authors describe their successfully experience demonstrating how two UCD teammates on the agile team can aid in upfront and continual user input.	IEEE	IR	Not specified	Usability Testing User stories Scenarios Prototype	Implicit Interaction design	Independent close collaboration
Ferreira (2008)	How Interaction design and agile development work together with a focus on the issue of interaction design were being done "up-front", before software development begins.	ACM	RP	XP	Prototype	Minimal up-front design	Close Collaboration/None collaboration
Fox (2008)	The research is an analysis of the results gathering from interviews conducted with members of teams that have integrated UCD practices into an Agile Methods development process. Propose three different approaches to better understand the roles the various team member have (Generalist, specialist and Hybrid).	IEEE	RP	Not specified	Usability Testing Prototype	Overlapped Track	Cross-functional team

Ungar (2008)	The researchers propose use the design studio as a bridge to avoid the gap between UCD and Agile methodologies. They describe benefits of up front design.	Scopus	RP	SCRUM	Usability Testing Contextual Inquiry Prototype Story Boarding	Overlapped Track	Close Collaboration
Hussain (2008)	The paper proposes and describes an integrated process for Extreme programming and User centered Design. Shows how this process facilitates user-orientation and at the same time preserves the social values of the development team.	Scopus	RP	XP	Usability Testing Persona Prototype	Implicit Interaction Design	Close Collaboration
Evnin (2008)	Apply a user interface technique in order to help development teams create experiences and interfaces that match user needs and expectations.	IEEE	IR	Not specified	User Stories Contextual Inquiry	Minimal up-front design	Cross-functional team
Wolkerstorfer (2008)	This case study sums up the setup of an adapted Extreme Programming (XP) process. The goal of the adaptation was to examine the applicability of a usability-aware agile software development process.	Scopus	RP	XP	Usability Testing Persona	Implicit Interaction Design	Close Collaboration
Obendorf (2008)	To address the usability weaknesses of XP, authors adopted techniques from Scenario-Based-Engineering (SBD) and Rapid Contextual Design (RCD).	Scopus	IR	XP	User stories Scenarios	Interaction design first	Close Collaboration
Ungar (2008)	In this report authors describe their experience with the merger of user centered design (UCD) into agile (team) development practice as manifest in a one day design studio.	IEEE	IR	SCRUM	Scenarios Contextual Inquiry	Minimal up-front design	Close Collaboration
Najafi (2008)	Agile development and UCD techniques were implemented in two projects with certainly differences to evaluate the impact on these projects with different characteristics.	IEEE	IR	SCRUM	Scenarios Persona Prototype	Overlapped track/Interaction design first	Cross functional
Singh (2008)	The authors propose U-SCRUM methodology each basically propose two ideas: Work with usability product owner and use of main usability artifacts.	IEEE	RP	SCRUM	Usr Stories Persona	Interaction design first	Close Collaboration
Federoff (2008)	This report panel describes the panelist Statements focus on adaptations to research methodology and strategy that make UCD possible to attain within Agile cultures.	Scopus	IR	Not specified	Not specified	Not specified	Not specified
Sy (2008)	In this workshop senior UX practitioners with prior experience on an	Scopus	IR	Not specified	Not specified	Not specified	Not specified

	Agile project shared their knowledge in order to improve future Agile user-centred design (UCD) experiences for User Experience (UX) practitioners.						
Hussain (2008)	This paper describe an approach to mobile application development focuses on the adoption of agile software development methodologies and user-centered design, emphasizing iterative user-interface development involving usability engineers and non-technical users.	IEEE	IR	XP	Usability Testing Index Cards Scenarios Prototype	Implicit Interaction design	Independent close collaboration
Brown (2008)	In this paper, we present a study of collaboration between user interaction designers and developers, with a particular emphasis on the role of artefacts in the process.	IEEE	RP	Not specified	User Stories Scenarios	Implicit Interaction design	Independent close collaboration
Broschinsky (2009)	The authors report their experience improving the communications team with the customer and developers' understanding of the customer needs by developing personas to integrate the user into the development process.	IEEE	IR	XP	User Stories Scenarios Persona Contextual Inquiry Prototype	Minimal up-front design	Close Collaboration
Illmensee (2009)	This paper presents a case study describing successes and failures while integrating continuous research into Agile projects..	IEEE	RP	SCRUM	Usability Testing Prototype	Minimal up-front design	Close Collaboration
Cho (2009)	The authors report an enterprise experience adopting an agile culture. Observing the organization's ability to adapt and establishing common objectives and principles across the organization.	IEEE	IR	SCRUM	Scenarios Persona Prototype	Overlapped Track	Close Collaboration
Barksdale (2009)	Team politics complicate software projects. They cause internal conflicts that can not only cost a software team time and money, but may also detract from the needs of the product's end users. In this paper, authors explore the use of concept maps as a means of mitigating such team conflicts.	IEEE	RP	XP	User stories Scenarios	Minimal up-front design	Not specified
Lee (2009)	Authors report the results of using XSBD approach to develop a product, summarizing how it addresses those requirements and draw conclusions about the effectiveness of the approach.	IEEE	IR	XP	Usability Testing User stories Scenarios	Overlapped Track	Close Collaboration
Adikari	This paper presents a	Scopus	RP	Not specified	Usability	Minimal up-	Close Collaboration

(2009)	design science approach using “Little Design Up Front” to integrate the User-Centred Design perspective into Agile Requirements Engineering.				Testing Prototype	front design	
Federoff (2009)	This paper discusses how the team and management responded to their satisfaction arising from agile transition and as a result reached an 85% satisfaction rate a year later.	Scopus	IR	SCRUM	Usability Testing User stories Prototype	Overlapped Track	Close Collaboration
Kollmann (2009)	The authors report that two principal issues to facilitate the integration of UCD and Agile approaches are the identity and a global vision for UX designers an agile. Another important issue is the communication between the team elements.	Scopus	RP	SCRUM	User stories Prototype	Not specified	Close Collaboration
Krohn (2009)	Authors present the application myPIM as an application web developed with FDD agile methodology and UCD in order to obtain usability since the beginning of the project.	Scopus	RP	FDD	Prototype	Minimal up-front design	Not specified
Hussain (2009)	The paper presents the preliminary study setup of a usability evaluation study. The study was applied about an application that enables a user to perform content-based search for audio and video content and play it via streaming on a mobile phone.	IEEE	RP	XP	Usability Testing	Implicit Interaction Design	Close Collaboration
Peixoto (2009)	Semantic network is used to represent main concepts in HCI design. The intention is to use this conceptual modeling to implement an expert system to guide agile developers during HCI design.	IEEE	RP	SCRUM	Prototype Heuristic Evaluation	Implicit Interaction design	Cross-functional
Miller (2009)	Authors describe a SIG. The goal of this SIG is to draw upon the shared experience of these practitioners to uncover the best practices for agile user centered design to facilitate optimal product development.	Scopus	IR	Not specified	Not specified	Not specified	Not specified
Schwartz (2009)	En este trabajo, se propone una metodología de gestión de proyectos de desarrollo para combinar la usabilidad y la funcionalidad del producto final a través de la utilización de UCD y los métodos ágiles.	Scopus	IR	Not specified	Not specified	Not specified	Not specified
Peixoto	This article presents	IEEE	RP	SCRUM	Usability	Overlapped	Not specified

(2009)	some limitations of Agile Methods and the results of using Scrum in a specific project and proposes an Expert System to help developers design Human-Computer Interfaces.				Testing Heuristic Evaluation	Track	
Belchev (2009)	Authors describe what usability activities they did and outline some techniques that your Agile team may wish to consider for your projects.	IEEE	IR	Not specified	Usability Testing Scenarios Contextual Inquiry	Minimal Up-front Design	Not specified
Hussain (2009)	This paper present the lessons learned from the adoption of HCI instruments into XP methodology.	Springer	RP	XP	Usability Testing User stories Prototype	Implicit Interaction Design	Close collaboration
Barksdale (2010)	The authors argue that the recommended interaction modifications through the concept mapping approach will improve strategic weaknesses that significantly burden the current interaction in an agile usability environment.	Scopus	RP	Not specified	User stories Scenarios Prototype	Implicit Interaction design	Cross-functional
Sohaib (2010)	The research aim is to identify the common approach of agile methods and usability engineering by surveying extensive related work on integration of usability and agile methods.	IEEE	RP	Not specified	Not specified	Not specified	Not specified
Hellmann (2010)	This paper introduces a new approach to user-interface test-driven development. This allows GUI tests to be written simply and without requiring that the GUI exist first.	IEEE	RP	Not specified	Usability Testing User stories Prototype	Implicit Interaction design	Not specified
Benigni (2010)	The research presents an agile methodology designed for the Web, to design or to re-engineer a software product by analyzing the product's architecture, developing new user interface's prototypes, and testing them using detailed usability criteria.	Scopus	RP	Not specified	Prototype Acceptance Test	Implicit Interaction design	Not specified
Meckem (2010)	Paper summarizes how "Rapid experimentation" approach was successfully applied in the customer service experience design.	Scopus	RP	Not specified	Prototype	Implicit Interaction design	Not specified
Albisetti (2010)	This paper is about the author experience shifting the launchpads teams' focus to a more user-friendly mindset.	Springer	IR	Not specified	Usability Testing	Minimal Up-front Design	Cross-functional
Ferreira (2010)	The results gathering from this study suggest that the values and assumptions of	Springer	RP	SCRUM	Prototype	Minimal Up-front Design/Implicit interaction	Independent none collaboration

	decision-makers external to the teams shape UX/Agile Practice.					design	
Rukh (2011)	Present a framework that incorporates user-centered design (UCD) philosophy into agile software development through a three-fold integration approach.	Scopus	RP	Not specified	Prototype	Overlapped Track	Close Collaboration
Lester (2011)	This paper presents the results of an empirical investigation that combined one activity of human-computer interaction, user-centered design and agile development into a small scale development exercise focused on website development.	Scopus	RP	XP	User stories Prototype	Implicit Interaction Design	Close Collaboration
Gonzalves (2011)	The authors propose a software that is able to build low-fidelity prototypes, document them and support user testing, to aid the process of interface building in the Scrum methodology.	IEEE	RP	SCRUM	Usability Testing User stories Scenarios Persona Prototype	Overlapped Track	Close Collaboration
Brown (2011)	This study focused on designer and developer collaborations in the early stages of project work at four workplaces.	Scopus	RP	Scrum XP Not Specified	Prototype	Not specified	Cross-functional team
Ferreira (2012)	This paper reports on three ethnographically-informed field studies of the day-to-day practice of developers and designers in organizational settings.	Scopus	RP	SCRUM	Prototype	Implicit Interaction Design	Close Collaboration
Isomursu (2012)	This paper describes the results of a single-case study, exploring the role of user experience (UX) work in agile software development.	Scopus	IR	SCRUM	User stories Scenarios Prototype	Overlapped Track	Close Collaboration
Felker (2012)	Authors report their experiences using the Scrum agile software development method in an undergraduate user-centered web development project.	Scopus	RP	SCRUM	Contextual Inquiry Prototype	Interaction design first	Close Collaboration
da Silva (2013)	In the paper, authors present a set of ten lessons learned from the multiple case study of agile teams working with interaction designers have been performed in order to better understand how these approaches are combined in practice.	IEEE	RP	SCRUM	Usability Testing User stories Prototype	Overlapped Track	Close Collaboration
Prior (2013)	This paper reports on how the adoption of an adapted agile approach involving the incorporation of a user advocate on the research team helped in meeting this challenge in one software project and offers suggestions	Scopus	RP	SCRUM	User stories Prototype	Minimal up-front design	Close Collaboration

	for how this could be used by other development teams.						
Losada (2013)	This paper explains when and how to integrate aspects of usability engineering in the agile development process proposed by the InterMod methodology.	Scopus	RP	InterMode	Usability Testing Prototype Heuristic Evaluation	Overlapped Track	Not specified
Abdallah (2013)	Authors propose the Quantified eXtreme Scenario Based Design (QXSBD) approach, which complements XSBD with a set of usability metrics collected from the agile architecture-centric approach and the Quality in Use Integrated Model (QUIM).	Scopus	RP	Not specified	Scenarios Prototype	Minimal up-front design	Close Collaboration
McGinn (2013)	In this paper, authors describe why combined RITE+Krug methods and which elements were incorporated from each. The impact of using this new RITE+Krug combination has been remarkable.	ACM	IR	SCRUM	Usability Testing Prototype	Overlapped Track	Close-Collaboration
Grigoreanu (2013)	The primary contribution of this paper is to present a new agile UX research methodology, the informal cognitive walkthrough (ICW), which combines simplified versions of SCWs and PWs, tailored for software development practitioners on fast-moving agile projects.	ACM	RP	Not Specified	Scenarios Prototype Cognitve Walkthrough	Implicit Interaction Design	Cross-functional
Rivero (2013)	In this paper authors propose a model-driven development approach called MockRE that captures requirements using User Interface prototypes (mockups) that end-users can understand completely.	IEEE	RP	MockRE	Prototype	Implicit Interaction Design	Not specified
Krusche (2014)	The Tornado model is a light-weight scenario-based approach for producing executable prototypes. Rugby, an agile process model based on Scrum, allows the developer to continuously deliver these prototypes at any time during a sprint to obtain feedback	ACM	RP	Rugby	Scenarios Prototype	Implicit Interaction Design	Not Specified
Anwar (2014)	This research focuses on Agile and UCD integration, providing a few practices and artifacts to guide designer's and Agile teams to overcome the challenges. How UCD techniques can support Agile in order to deal with usability issues. The authors identify the	IEEE	RP	SCRUM	Usability Testing Scenarios Persona Contextual Inquiry Prototype Heuristic Evaluation	Minimal Up-front design	Close collaboration

	levels and phases where stakeholders, especially the user involvement and interaction are required.						
Rahim (2014)	This study demonstrates the applicability of agile user centered design to engineer m-learning application in informal setting.	IEEE	RP	XP	Scenarios Persona Contextual Inquiry Prototype Contextual Design Story Boarding	Interaction Design First	Not specified
Peres (2014)	This paper presents a maturity model for integrating agile methods and user experience in the software development cycle that may be suitable for small companies. The proposal is in line with CMMI, MPS. BR and ISO18529.	IEEE	RP	SCRUM	Persona Contextual Inquiry Prototype Story Boarding	Overlapped Track	Close Collaboration
Choma (2015)	The contribution of this work was to propose a protocol based on the concepts of personas and Nielsen's heuristics that improves the communication of usability aspects between the UX team and SCRUM team	SPRINGER	RP	SCRUM	Persona Heuristics Evaluation Usability Testing Prototype	Implicit Interaction Design	Close collaboration
Belluci (2015)	Authors present an approach called Extreme Co-design to denote how design is inscribed in Extreme Programming sessions with rapid cycles of use, design and development that allow extensive exploration and experiencing of appropriation scenarios.	SPRINGER	RP	XP	Scenarios Prototype	Implicit Interaction-Design	Not specified
D Silva (2015)	In order to address the issue about the integration between agile and User-Centered Design in real-life projects, the authors carried out multiple case studies in two large agile companies from Canada and Brazil. One of their goal was to better understand how usability evaluation occurs within agile environments.	IEEE	RP	SCRUM	Usability Testing Prototype	Minimal Up-front design	Close collaboration Cross functional
Choma (2015b)	Authors proposal supports the conduction of projects integrating the view points of industry and academia, aiming to add User experience (UX) methods into agile practices.	IEEE	RP	SCRUM	Usability Testing Persona Prototype Heuristic Evaluation	Implicit Interaction Design	Close collaboration
Kropp (2016)	Authors introduce a team role called 'On-site User Experience Consultant' (osUX consultant) which supports integrating	SPRINGER	IR	SCRUM	Usability Testing Scenarios Persona Contextual Inquiry	Overlapped Track	Close Collaboration

	user-centered design (UCD) firmly in agile RE under the constraints of fixed-price software development projects.				Prototypes		
De Freitas (2016)	This paper presents a method for integrating usability evaluations on agile software development aiming to improve software quality and user satisfaction.	SPRINGER	RP	SCRUM	Prototypes Heuristic Evaluation	Overlapped Track	Cross functional
De Carvalho (2016)	This study is focused on understanding the integration of Agile and User-Centered Design through usability evaluations before the development cycles. Based on a pilot study carried out in a Scrum web-based software development team.	SPRINGER	RP	SCRUM	Persona Prototype Heuristic Evaluation Cognitive walkthrough	Minimal Up- front Design	Cross-functional
Choma (2016c)	Authors present a grammar for stories of interaction called UserX Story, in order to remedy the difficulties encountered by teams to insert UX aspects and usability requirements in the first steps of software conception. An action research approach was applied to carry out the research within the industry, allowing to work closely with the agile teams.	SPRINGER	RP	SCRUM	Usability Testing Scenarios Persona Prototype Heuristic Evaluation	Not Specified	Close collaboration
Sfetsos (2016)	In this study authors empirically investigate the integration of User-Centered Design (UCD) practices into agile Web development. They propose a hybrid process framework which extends the agile development process, adding lightweight usability engineering practices.	IEEE	RP	Other	Usability Testing Scenarios Persona Prototypes	Overlapped Track	Close Collaboration
Joyce (2016)	The aim of this work is to define a conceptual method, namely Contextual Usability Evaluation (CLUE). The proposed Contextual Usability Evaluation method promises to allow Human-Computer Interaction experts to evaluate the perceived impact of varying contexts over time on the usability of mobile Applications.	IEEE	RP	Not specified	Usability Testing	Overlapped Track	Not Specified
Bhaskar (2016)	This paper explores a new combined process for building usable APIs that combines concepts from a usability evaluation method (Cognitive Dimensions Framework) and an Agile development methodology (eXtreme Programming).	IEEE	IR	Other	Usability Testing Scenario Persona Contextual Inquiry Prototypes	Implicit Interaction Design	None collaboration

Kropp (2016)	This paper is about experiences with user-centered design (UCD) and agile requirements engineering (RE) in fixed price projects. Efficiently incorporating usability and UCD into RE remains a challenge.	IEEE	IR	Not specified	Usability Testing Scenario Persona	Minimal up-front Design	Close collaboration
---------------------	---	------	----	---------------	---	-------------------------	---------------------

Anexo B. Proyecto Piloto PEAP

B.1. Definiciones de Persona del proyecto piloto PEAP

Persona
<p>Name: <i>Ming Zhao</i> Age: 35 Occupation: <i>Administrator</i></p>
Activities	
<p><i>What the user does; frequency and volume Why the user is engaged in the product domain</i></p>	<p><i>Zhao is 35 years old. He is a site administrator at IdeaXchange. He is in charge of user and transaction management of the system. Zhao verifies that he need to make sure the ideas in the system is original and legal. He wishes to eliminate those ideas provided by plagiarizing others' idea or illegal according to the laws. Similarly, he wants to close the accounts of malicious users and reduce the possibility of illegal transaction on the system. Zhao expects that he could delete some of comments including abuses, profanity, and personal attacks, so that the users are able to enjoy the good purchasing environment in the system.</i></p>
Goals in using the solution	
<p><i>How the user thinks about the product domain and technology Desired results</i></p>	<p><i>Zhao expects that he is able to view all the users, the ideas, the transactions and the users in the system. If he finds some comments including abuses, profanity and personal attacks, he wishes he is able to delete them. If he find an unoriginal or illegal idea, he wishes to delete them. And when he fails to identify and delete an unoriginal or illegal idea, he wishes he could forbid the transaction of the idea when the transaction is processing. For the users provide these ideas and make inappropriate comments, he has the right to close their account.</i></p>
Type of Persona	
	<p><i>Primary</i></p>

Figura B-1. Persona “Ming Zhao”

Persona
Name: Kogoro Mori Age: 48 Occupation: Academic Researcher
Activities	
<i>What the user does; frequency and volume Why the user is engaged in the product domain</i>	<p>Mori is 48 years old. He is an academic researcher working in a university. His major is finance. Sometimes, he comes up with great business ideas and he can't afford the idea by himself. However, he is willing to describe these ideas and does some financial analysis about them. After that, he wishes to introduce and sell these ideas to some entrepreneurs or managers. He expects to use an idea transaction system, in which he could introduce the intriguing part of his ideas and finally persuade others to buy it. If someone is willing to comment his ideas and give him advices, he wants to read and consider about them. He hopes he can put some of his information on the system so the potential buyer could know his experience and works. He wishes he has the right to delete all the datas of his account, if he is not willing to use this system anymore, because he want to protect his intellectual property.</p>
Goals in using the solution	
<i>How the user thinks about the product domain and technology Desired results</i>	<p>Mori expects that he could register in the idea transaction system. After that, he hopes that he can login the system, create and sell ideas in the system, read the comments on his ideas and give reply to them. When adding ideas he wants to be able to sort the idea into either new or existing groups and apply labels that will let him keep his ideas organized. As he creates new ideas, he wish to mark them as being ready for sale or keep them private as he develops them before he make them public. He can come back to his idea in progress and continue his progress on it later. He hope he could list and classify all the ideas in the system, and he himself can buy some of them. When looking for areas of general industry interest he can simply go on to the idea platform and sort ideas by popularity. He can also see which ideas have the highest score and vote on those that seem most innovative for him as a researchers. Then He can read more about each idea and mark as favourites ideas that are of interest to him for future reference. Later on, he can come back to these ideas and leave comments of his thoughts of the ideas as related to other ideas on the platform. He wishes to communicate with potential buyers so that he can persuade them to buy his ideas. Mori needs to be able to edit his personal information in the system, so that his potential customer could know him better. Mori wants to delete all his information and data if he doesn't want to use this system any more.</p>
Type of Persona	Primary

Figura B-2. Definición de Persona “Kogoro Mori”

Persona
<p>Name. <i>Spencer Johnson</i> Age: 58 Occupation: <i>Entrepreneur</i></p>
Activities	
<p><i>What the user does; frequency and volume</i> <i>Why the user is engaged in the product domain</i></p>	<p><i>John is the CEO of a consultant firm. He oversees all the work of his company and want to be inform of all the issues that come up in the different projects of his company. As the CEO of a successful company, John will want to provide to the community some insight of what he has done in creating his company.</i></p>
Goals in using the solution	
<p><i>How the user thinks about the product domain and technology</i> <i>Desired results</i></p>	<p><i>John expects to be able to visit the idea store of 'IdeaXchange' in which he will search for the ideas that address his current problem by using keywords related to his issue. John will need to be able to classify those ideas to determine which one are the most effective according to the point of view of others users that have already applied them. Once John has found the best idea for resolving his problem he will need to be able to obtain more details about the idea by contacting with the seller. John will need this extra information to pitch the idea at his company before committing to purchase it.</i> <i>John also expects to be able to create his own ideas that have been proven useful in his company.</i></p>
Type of Persona	
	<p><i>Secondary</i></p>

Figura B-3. Definición de persona “Spencer Johnson”

Persona
<p>Name. <i>Rick Richardson</i> Age: <i>31</i> Occupation: <i>Team Manager</i></p>
Activities	
<p><i>What the user does; frequency and volume</i> <i>Why the user is engaged in the product domain</i></p>	<p><i>Rick is the team manager at a software company focused in developing mobile phone apps. Rick is in charge of a multidisciplinary team including software developers, designers, ... in which he has to plan all the work of the different projects his team has been assigned</i></p>
Goals in using the solution	
<p><i>How the user thinks about the product domain and technology</i> <i>Desired results</i></p>	<p><i>As the team manager, Rick has detected some issues regarding the working process of his team. Rick sees 'IdeaXchange' as the place where he could find useful ideas to address his team issues. Rick will want to be able to save to his profile all the different ideas that he has found interesting. Rick will want to review those ideas saved in his profile in order to choose the one that he wants to buy. Once he had buy the idea he will want to receive a full disclosure of the idea in order to apply it to his team. As a feedback, Rick will want to use 'IdeaXchange' to comment his experience on the idea he bought to facilitate other users the finding of effective ideas.</i></p>
Type of Persona	
	<p><i>Secondary</i></p>

Figura B-4. Definición de persona “Rick Richardson”

Persona
Name. Joaquin Reynolds Age: 46 Occupation: Operations Manager
Activities	
<i>What the user does; frequency and volume Why the user is engaged in the product domain</i>	<i>Chad is an operations manager at Illumina, which is a biotechnology company specializing in products relating to genetics. He is currently assigned to overseeing the development of a new line of genome sequencers designed to be very low cost and affordable for hospitals and clinics to use for serving their patients. Chad oversees progress on this project and is in charge of deciding how to allocate resources to project and how to structure all those who will be working under him (which includes about 120 individuals ranging from engineers, designers, to software developers). Work on this has just begun, so he is deciding how best to organize those under him to allow for efficient communication between groups so as to avoid unforeseen problems arising as was the case with the last project he managed.</i>
Goals in using the solution	
<i>How the user thinks about the product domain and technology Desired results</i>	<i>Chad is very interested in getting new management ideas and new ideas for reducing costs, especially those that will help him improve cross-disciplinary communication. He is viewing IdeaXchange as a prime opportunity for getting lots of new ideas that will help him achieve his goals in these areas. Chad is an experienced manager and trusts the voice of other experienced managers, much more so than that of academics and unproved "expertise" givers. He views the IdeaXchange as particularly useful because he hopes that he will be able to use it to get the informed opinion of others like him who have applied different ideas and who have experience and stories to share. He hopes that from this he will be able to obtain several ideas from users whose opinions he trusts and will be able to learn more and apply these techniques at Illumina.</i>
Type of Persona	
	Primary

Figura B-5. Definición de persona "Joaquín Reynolds"

B.2. Product Backlog inicial del equipo PEAP definido a partir de las personas identificadas utilizando el patrón “Agile Persona”.

1	As an academic researcher, I want to create a new idea in order to sell it.
2	As an entrepreneur, I want to edit my own idea in order to make it better.
3	As an entrepreneur, I want to comment on an idea in order to share my experience with it.
4	As an team manager, I want to vote on an idea in order to express my opinion.
5	As an academic researcher, I want to classify / list ideas in order to keep them arranged.
6	As a startup entrepreneur, I want to buy an idea in order to read the whole idea which I like.
7	As an operations manager, I want to able to view published ideas in order to find the ideas I am interested in.
8	As an academic researcher, I want to sell an idea in order to make money.
9	As an operations manager, I want to login into the system in order to access my account.
10	As an operations manager, I want to register in the system in order to create my own account.
11	As an academic researcher, I want to be able to edit my account with personal Information, in order to provide my information to my potential customer
12	As an academic researcher, I want to be able to reply to comments on my ideas In order provide readers with additional information concerning negative or positive comments.
13	As a project manager, I want to receive messages from other users about m y ideas in order to communicate with potential buyer.
14	As a project manager, I want to send messages to the owner of ideas that I like in order to get more information.
15	As an administrator, I want to be able to view all the users, ideas and comments, in order to manage them.
16	As an administrator, I want to be able to delete an idea in order to enforce integrity of the system.
17	As an administrator, I want to delete comment on an idea in order to enforce integrity of the system.
18	As an administrator, I want to delete users in order to remove illegal users.

Tabla B-1. Product Backlog equipo del proyecto piloto PEAP

B.3. Historias de usuario del sprint 2 y Sprint Backlog del sprint 3 PEAP

User stories	Points	Priority
Plan for Sprint 2	13	High Priority
Create a New Idea	5	High priority
Edit an Idea	3	High priority
List/Classify Ideas	8	Medium priority
View Published Ideas	3	High priority

Figura B-6. Historias de usuario del proyecto PEAP desarrolladas en el sprint 2

Backlog Item	Task	Task Owner	Initial Estimate	Hours of Work Remaining on Each day of the sprint							
				Fri	Sat	Sun	Mon	Tus	Wed	Thu	Fri
Idea Classification (5 points)	Create test cases	MD, TC	1	1	1	1	0	0	0	0	0
	Implement front-end code	MD, TC	1	1	1	1	1	1	0	0	0
	Implement back-end	AZ, PN	3	3	3	0	0	0	0	0	0
	Complete testing	AZ, PN	2	2	2	2	1	1	0	0	0
Edit User Account(3 points)	Create test cases	MD, TC	1	1	1	0	0	0	0	0	0
	Implement front-end code	MD, TC	2	2	2	2	2	1	1	1	1
	Implement back-end	AZ, PN	2	2	2	2	0	0	0	0	0
	Complete testing	AZ, PN	1	1	1	1	1	1	1	1	1
Delete User Account (3 points)	Create test cases	MD, TC	1	1	1	0	0	0	0	0	0
	Implement front-end code	MD, TC	1	1	1	1	1	0	0	0	0
	Implement back-end	AZ, PN	2	2	0	0	0	0	0	0	0
	Complete testing	AZ, PN	1	1	1	1	1	1	1	0	0
View Comments(8 points)	Create test cases	MD, TC	1	1	0	0	0	0	0	0	0
	Implement front-end code	MD, TC	2	2	2	2	2	2	2	2	1
	Implement back-end code	AZ, PN	5	5	5	5	3	2	1	0	0
	Complete testing	AZ, PN	2	2	2	2	2	1	1	1	1
Make Comments (8 points)	Create test cases	MD, TC	1	0	0	0	0	0	0	0	0
	Implement front-end code	MD, TC	2	2	2	2	1	1	1	1	1
	Implement back-end code	AZ, PN	5	5	3	2	0	0	0	0	0
	Complete testing	AZ, PN	2	2	2	2	2	1	1	1	1
Navigation and Design (8 points)	Design User Interface	MD	5	5	5	5	5	0	0	0	0
	Complete front-end code	AZ, PN, TC	6	6	6	6	6	2	0	0	0
Total			49	48	43	37	28	14	9	7	6

Figura B-7. Sprint Backlog del proyecto PEAP del sprint 3

Anexo C. Proyecto de Evaluación Académica PEAA

C.1. Definiciones de Persona de los equipos del grupo PEAA

PERSONA 1. Mario Duran Lopez

Edad: 20 años

Sexo: Hombre.

Puesto de trabajo: Estudiante universitario.

Horas de trabajo: No aplica.

Educación: Bachillerato.

Localización: Francisco de Montejo.

Ingreso: Depende del ingreso de sus padres.

Interacción con Tecnología: En la universidad tiene contacto con la tecnología a diario, también cuenta con acceso a internet desde su casa, cuenta con dispositivos personales que le permiten acceso a medios digitales y por su cuenta busca siempre nuevas formas de interactuar con la tecnología.

Limitaciones: Olvida con facilidad las cosas, tiene muchas distracciones y casi siempre olvida organizar su tiempo.

Familia: Vive con su padre pero pasa la mayor parte del tiempo en la escuela.

Hobbies: Estudiar mucho, leer, ir a actividades culturales.

Metas: Su mayor meta es llegar a ser challenger, también desea graduarse en ingeniería de software, así como conseguir un buen trabajo e independizarse. Desea aprender nuevas cosas e incrementar sus habilidades como ingeniero de software

Figura C-1. Definición de persona “Persona 1” del equipo PEAA2

PERSONA 2. Victor Hugo Menendez

Edad: 30 años.

Sexo: Hombre.

Puesto de trabajo: Profesor de la Facultad de Matemáticas.

Horas de trabajo: 8 Horas.

Educación: Universitario con postgrado.

Localización: Mérida.

Ingreso: \$800 diarios.

Interacción con Tecnología: Debido a la rama de estudios que imparte en la facultad, su nivel de interacción con la tecnología es bastante elevado, también se mantiene actualizado sobre las nuevas tendencias y/o tecnologías, para poder dar a sus alumnos información actualizada y fresca.

Limitaciones: Su principal limitante es el tiempo disponible, ya que pasa la mayor parte de su tiempo impartiendo clases, casi no dispone de tiempo libre para investigar sobre nuevas y más avanzadas tecnologías.

Familia: Tiene dos hijos, y una esposa. Sus hijos tienen 17 y 14 años respectivamente. El primero de ellos con gran afición por el uso de internet y las tecnologías web.

Hobbies: Diseñar nuevos sistemas y aplicaciones, leer y documentarse sobre nuevas tendencias tecnológicas.

Metas: Su principal meta es llegar a ser director de la facultad de matemáticas y posteriormente ser rector de la universidad, también desea establecer una empresa y comprarse una nueva casa más grande.

Figura C-2. Definición de persona “Persona 2” equipo PEAA2

Modelo Persona De ProAccountant

Nombre: Maria Pancha.

Edad: 30 años.

Ocupación: Vendedora.

Actividades

Maria pancha se levanta todos los días a las 8 a.m. a abrir su pequeño local que tiene cerca de su casa, en el este local Maria Pancha cuenta con diversos artículos, algunos de ellos elaborados por ella misma, como vienen siendo ropa, calzado, accesorios e incluso accesorios del hogar. En su mismo lugar de trabajo Maria Pancha desayuna mientras trata de arreglar todo para empezar su día, revisando pedidos, intentando hacer todos, pero no siempre son posibles pues a veces los apuntes se pierden o ella olvida donde apuntarlos, ella es la única trabajadora de este pequeño local pues no tiene dinero suficiente para los empleados, ella trabaja en el local desde las 9 de la mañana hasta las 9 de la noche, pero para Maria Pancha su trabajo no termina ahí, pues ella debe sacar el inventario de lo que se ha vendido y le toma demasiado tiempo y a veces el resultado final no siempre coincide con lo que debería.

Metas y Necesidades.

Maria Pancha necesita ser más ágil en su trabajo, sin perder nada de información ni encargos ni apuntes de sus ventas, ella quisiera tener organizado sus ventas así como quisiera tener al final del día una tabla con todas las ganancias, pero tiene que sacarla manualmente y a veces no la hace. Ella está acostumbrada a tener todo apuntado en una libreta pero ella siente que es demasiado lento. Maria Pancha cuenta actualmente con un teléfono inteligente que se conecta a internet y ella quisiera que esa información sea fácil de llevar a todos lados y que no se le pierda la información en un momento dado.

Figura C-3. Definición de persona del equipo PEAA3

Persona
<p>Nombre: Ana</p> <p>Edad: 16</p> <p>Ocupación: Estudiante</p>
Actividades	
<p>Ana es una estudiante de 16 años. Todos los días Ana se levanta temprano para ir a la escuela. Ana es muy indecisa, y debido a que cuenta con demasiada ropa, pasa aproximadamente entre 10 y 15 minutos escogiendo su vestuario, tratando de que las prendas combinen y que le den el abrigo suficiente para hacerle frente a los recientes cambios climáticos. Acto seguido, Ana desayuna y se va a la escuela. Ana realiza actividades extracurriculares, para las cuales debe vestir de acuerdo a la ocasión. Ana siempre sale los fines de semana con sus amigas, y suele lavar su ropa cada dos semanas.</p>	
Metas y necesidades al utilizar la aplicación	
<p>Ana necesita una aplicación que le proporcione un atuendo de manera inmediata, tomando en cuenta criterios como color, clima, entre otros. Además, se le indicará si debe llevar paraguas. Por otra parte, Ana espera contar con un sistema calendárico que le permita reservar ropa para las fechas que ella desee. Ana quiere poder compartir sus vestuarios y así mismo ver los vestuarios de sus amigas. Ana también requiere que la aplicación le indique la cantidad de ropa limpia restante para así saber cuándo sea necesario lavar. Ana quiere optimizar el uso de su tiempo.</p>	
Persona	Primaria

Figura C-4. Definición de persona del equipo PEAA4

PERFIL DE USUARIO	
1. Nombre del Producto:	Aropate
2. Características Generales de la población de usuarios.	
Mujeres (15-45 años) de la ciudad de Mérida, YUC.	
3. Características de los Usuarios que son relevantes para el Producto.	
Tener un dispositivo móvil. Experiencia y uso de tecnologías informáticas. Tener un amplio guardarropa	
4. ¿Cuáles de las Características listadas en 3, deberían los participantes en el Test tener en común, y cómo están definidas?	
Tener un dispositivo móvil.	
5. ¿Cuáles de las características que fueron listadas en 3, variara en la Prueba, y como están definidos?	
<p>Experiencia en el manejo de aplicaciones web.</p> <p>Nulo = ningún uso de aplicaciones web</p> <p>Bajo = Utiliza aplicaciones web sin saberlo (Youtube, Facebook)</p> <p>Medio = Utiliza aplicaciones web y sabe que existen otras de su índole, además se desenvuelve con facilidad en las usadas por los usuarios con nivel bajo.</p> <p>Alto = Conoce el significado de aplicación web, o puede diferenciar prácticamente entre una aplicación y una simple página web.</p> <p>Avanzado = Conoce el significado del término y aspectos técnicos o más.</p> <p>Experiencia y percepción de internet.</p> <p>Nulo = Percepción negativa y reacio a usarla</p> <p>Bajo-Entusiasta = con pocos conocimientos pero visión positiva</p> <p>Bajo-Pesimista = pocos conocimientos y visión negativa</p> <p>Medio-Entusiasta = conocimientos suficientes para efectuar navegación sin problemas mayores y visión positiva</p> <p>Medio-Pesimista = conocimientos suficientes para efectuar navegación casi natural pero con visión negativa</p> <p>Alto-Entusiasta = facilidad de manejo y navegación casi natural, deseo de aprender más.</p> <p>Alto-Pesimista = facilidad de manejo de la web y navegación natural, con una visión negativa del impacto de internet.</p>	

Figura C-5. Perfil de usuario derivado de la definición de persona equipo PEAA4

<p>Nombre: Martina Rosique Campos</p> <p>Edad: 53</p> <p>Empleo: Responsable de un centro de adopción.</p>
<p>Actividades</p> <p>Martina es la responsable de un centro de adopción junto con su hermana. Es una de las personas que están a cargo del lugar y está involucrada en todas las actividades que se realizan en este sitio, desde atender a los animales hasta el papeleo de las adopciones. Acude al lugar durante más de seis horas diarias, seis días a la semana. Algunos días cuenta con el apoyo de voluntarios para realizar las tareas que no son administrativas.</p>	
<p>Metas y necesidades</p> <p>Martina espera que las personas puedan llegar al albergue con una idea de cuál es el animal que desean adoptar y cuál es el proceso requerido para la adopción, incluyendo los papeles necesarios y el precio que tiene para no desperdiciar tiempo que sería útil en otras actividades.</p>
<p>Nombre: Nilma Cortes Ancona</p> <p>Edad: 20-60 años</p> <p>Empleo: Infinidad de opciones</p>
<p>Actividades</p> <p>Nilma es una mujer que tiene dos perros a los que quiere mucho. Los perros están en la tarraza, la cual está a un lado del garaje y tiene una reja para evitar que salgan a la calle. Vive con la preocupación de que sus mascotas puedan extraviarse por un descuido suyo o de alguien en casa.</p>	
<p>Metas y necesidades</p> <p>Nilma espera que si alguno de sus perros escapa su localización sea fácil y rápida sabiendo de las tecnologías de la actualidad que pueden simplificar el problema. Espera que si alguna persona llega a encontrar o ver a sus animales tenga un espacio para encontrar al dueño de estos, y así poder contactarla y que regresen a casa.</p>	

Figura C-6. Definición de personas equipo PEAA5

PERSONA

Nombre: Ruperto Salazar.

Edad: 19 años.

Ocupación: Estudiante (persona que frecuenta el transporte público).

Actividades.

Ruperto se levanta todos los días para ir a la escuela 7:30 am, generalmente él siempre llega tarde a la escuela debido a que no aborda el camión a tiempo a causa de que no sabe el tiempo aproximado en que pasa el transporte público y a veces porque aborda un camión erróneo a causa de la prisa. A él le gustaría visitar otros lugares de la ciudad, como plazas, museos, cines, etc. Pero él necesita que lo lleven porque no conoce distintas maneras de llegar a su destino. Además él cuenta con una credencial de estudiante del transporte público, el cual reduce tarifa a pagar, pero él no suele usarla ya que desconoce en que momentos pasan los autobuses que aceptan este tipo de credenciales.

Metas y Necesidades.

Ruperto como muchos estudiantes de la facultad, tiene que medir su tiempo para poder tomar el transporte público y sacar el máximo provecho de la credencial con la que dispone, él espera por lo tanto tener los horarios en que pasa el autobús por su paradero, antes de que estos lleguen, también necesita buscar de manera rápida las formas en las que él pueda llegar al lugar al que quiera ir mediante el uso del transporte público por lo que él esperaría poder ingresar una palabra clave en un buscador y obtener una lista de los autobuses que puede tomar así como la ubicación de donde puede abordarlos.

Figura C-7. Definición de persona equipo PEAA6

Persona 1.
<p>Nombre: José Luis Rodríguez Estrada. Edad: 21 años. Ocupación: Estudiante de Universidad.</p>
Actividades.	
<p>José es un estudiante de universidad de 21 años. José asiste a clases de universidad de lunes a viernes en horario de 7:00am a 2:00pm, aunque a veces los trabajos en equipo o investigaciones en la biblioteca lo obligan a quedarse más tiempo. José es un estudiante muy dedicado y le preocupan mucho sus calificaciones, por lo que las rondas de estudio pueden ser de horas prolongadas. Además, José es muy extrovertido y durante los fines de semana asiste a los eventos que se realicen en su ciudad cuando se entera de ellos. José se reúne los viernes con amigos y juntos deciden a que evento asistir. Cuando deciden, José se encarga de revisar el sitio web o el evento en Facebook para obtener más información. José consulta Google Maps para encontrar la ruta más adecuada para llegar al lugar del evento en automóvil, además, de ser posible reserva boletos en Ticketmaster o realiza preguntas al organizador del evento.</p>	
Metas y necesidades.	
<p>José necesita una aplicación que le permita explorar su alrededor y saber que eventos se realizarán cerca de él. José espera que todas las actividades antes de asistir al evento, como búsqueda, preguntas y resolución de dudas, consulta de información, reservación de boletos, entre otras, pueda realizarlas en un solo lugar para ahorrar tiempo. Además, José necesita que la aplicación pueda realizar conexión con las redes sociales para invitar a sus amigos y compartir que asistirá a un evento.</p>	
Tipo de Persona.	Primaria.

Figura C-8. Definición de persona “Persona 1” del equipo PEAA7

Escenario 1
<p>Tarea: Consultar la ubicación de un evento de música realizado en la ciudad. Personaje: José Luis Rodríguez Estrada. Grupo de usuarios: Estudiantes.</p>
Contexto.
<p>CrowdID es una aplicación móvil que permite buscar eventos que se realizan alrededor del usuario basado en su ubicación geográfica y sus intereses personales. La aplicación presenta los datos más relevantes de todos los eventos cercanos en forma de tarjetas interactivas.</p>
Escenario
<p>José llega a su casa un viernes por la tarde luego de haber estado en sus clases de la universidad. José está emocionado porque es fin de semana y tiene ganas de salir para relajarse un rato y convivir con sus amigos. José es amante de las nuevas propuestas musicales, por lo que se dispone a buscar un evento de música en su ciudad al cual pueda asistir.</p>

Figura C-9. Definición del “Escenario 1” derivado de la persona “Persona 1” del equipo PEAA7

Persona 2.
<p>Nombre: Elvis Josue Tec Ramirez Edad: 34 años Ocupación: Organizador de evento</p>
Actividades.	
<p>Josue es un organizador de eventos por cuenta propia, dedica día a día a organizar eventos para otras personas y siempre presta cuidado a cualquier detalle de estos. Cada que Josue realiza un evento se ve obligado a elegir entre que sus clientes deban encargarse de la divulgación de las invitaciones o que el cree un evento a través de Facebook y lo divulgue por redes sociales. Siempre se le presentan problemas al momento de divulgarlo en Facebook ya que muchos de los invitados de sus clientes no se enteran del evento provocando que no asista la gente que sus clientes esperan.</p>	
Metas y necesidades.	
<p>Josue quiere ofrecerles a sus clientes una forma más efectiva para divulgar sus eventos, para ello necesita una plataforma en la cual sus invitados puedan enterarse de forma fácil y sencilla de los eventos a los que son invitados y que además no se pierda entre la demás información que suele encontrarse en otros medios.</p>	
Tipo de Persona.	Primaria.

Figura C-10. Definición de la persona “Persona 2” del equipo PEAA7

Persona
Nombre: Verónica Medina Carrillo
	Edad: 35
	Ocupación: Contador
	Hora de entrada: 7am
	Hora de salida: 3:30pm
Actividades	
Verónica tiene 35 años de edad, ella es la encargada de llevar el control de cuantas personas se inscriben a la escuela donde ella trabaja así como los gastos de servicios utilizados en la misma. Cuando Verónica llega a su oficina a las 7 am verifica los movimientos registrados en días pasados, así como los gastos sin pagar de la escuela. Verónica revisa su calendario para saber que días hay que pagar los servicios que se le brindan a la escuela (Agua, luz, mantenimiento, etc) para así poder llevar un control del dinero que se utiliza para dichos servicios. En los días de inscripción Verónica apunta todos los alumnos inscritos para poder saber quien a pagado su colegiatura y quien no. Verónica verifica el total de dinero utilizado para el pago de servicios y pago a los maestros, para después restarlo del dinero que la escuela ha ganado con las colegiaturas y así saber la ganancia total de la escuela por mes.	
Metas y Necesidades	
Verónica experta en la organización y manipulación de información necesita un programa que le ayude a automatizar sus procesos para el rápido manejo de dicha información. Verónica necesita que sus tiempos de resolución trabajo sean más rápidos, debido a que tiene que realizar otras actividades. Verónica espera poder localizar en cualquier momento a un alumno determinado para saber si ya pago su colegiatura o no. Verónica necesita saber el total de ingreso mensual en cualquier momento, así como el dinero gastado quincenalmente en los servicios que la escuela utiliza.	
Tipo de Persona: Primaria	

Figura C-11. Definición de persona del equipo PEAA8

Anexo D. Patrones evaluados

D.1. Patrón evaluado “Agile Persona” PAPv0

Name	Agile persona
Initial context	This pattern can be used in agile projects to help development team to learn and understand about the user's needs, goals and behavior in order to identify the final users and develop the right functionalities for them with useful and usable interfaces.
Result Context	Specific descriptions of the final users' features will be obtained. Such descriptions allow the development team to know about the users for whom the software will be developed. Those descriptions will be available to review at any moment through the agile development process.
Problem	The agile development team must be able to identify and define as soon as possible the final users to whom the software will be developed avoiding by this way waste of time in reworking through the development process.
Forces	<p>Kind of the organization This pattern can be used in agile projects regardless of company size. In order to get better results it can be implemented for clients who have well-defined their organizational processes.</p> <p>Kind of system to be developed This pattern can be used in agile projects where the user's requirements would change through the development. This product would be evaluated and updated according to the feedbacks received from the iterations' release reviews.</p> <p>Kind of client The user must be available and always involved through the process of the persona's definition.</p> <p>Rationale</p> <ul style="list-style-type: none"> Meetings for interviews, product pattern's review or feedback between development team and user must be established in a friendly and trusted environment. The product pattern persona must be keep available for the team through the whole development process at any moment.
Solution	<ul style="list-style-type: none"> The development team receives information previously gathered from the Minimal User Research (observation and interviews). The development team defines persona as user models with hypothetical archetypes describing the user's features identifying needs, goals and behavior variables. The development team prioritizes personas identified. The user review together with development team, the persona's description and prioritization. If is necessary the development team refines persona's definitions by creating a new persona product pattern or by updating existing products patterns after each iteration feedback or product owner's review. <p>a) Who are involved and how</p> <p>Agile development process</p> <ul style="list-style-type: none"> At the beginning of the development process the development team gets information through a Minimal User Research using observations and interviews. The development team describes the personas identified using the template proposed and

	<p>prioritize the type of persona identified (primary or secondary)</p> <ul style="list-style-type: none"> Persona's definition is reviewed by both product owner and team development If refinement is necessary development team update the persona's definition according to the product owner's review and verify the prioritization of persona updating it if is necessary Product backlog is defined and the user stories are related with the personas involved with them. Define the sprint backlog <ul style="list-style-type: none"> Develop the Sprint <ul style="list-style-type: none"> Identify the users involved in the user story in order to know if the personas defined previously cover the archetypes needed in the user story. Relate the personas' definition with the user stories to develop. Verify definitions with the personas related with the users involved in the user story. Validate the refinement or creation for a new persona definition If is necessary create new persona then minimal user research is applied Else if is identified a necessary refinement to persona then we update the definition according the feedback gathered from the previous iteration or product owner's review. Verified the prioritization of persona and updated it if is necessary. Develop the user story Deliver a shippable product Gathered feedback If is not the last Sprint then verify the product backlog and determine the next sprint backlog <p style="text-align: center;">b) When product pattern persona is applied in agile</p>
Roles	<ul style="list-style-type: none"> UI Designer (optional) Agile Developer User (product owner)
Basic Knowledge	<p>Knowledge</p> <ul style="list-style-type: none"> Knowing about requirements gathering <p>Skills</p> <ul style="list-style-type: none"> Capability to apply user research to get data through direct observation, interviews and questionnaires. Capability to Data Analysis identifying needs and establishing requirements
Entries	User information
Exit	Persona's descriptions as user Models
Collaboration	---
Capability label	---
Lessons	<ul style="list-style-type: none"> The whole development team must be involved in the personas definition.

learned	<ul style="list-style-type: none"> After the UI designers and/or developers have prioritized the personas described the models have to be presented to review in a meeting to all the development team and users. Users, UI designers and developers must be involved in the presentation, reviews and feedbacks meetings. 										
Examples	<table border="1"> <thead> <tr> <th colspan="2">Persona</th> </tr> </thead> <tbody> <tr> <td> </td><td> <p>Name: Michael Age: 35 Occupation: Accountant</p> </td></tr> <tr> <td colspan="2"> <p>Activities</p> <p>Michael is a 35 years old, he is a management accountant in charge of the company's store. Every day when Michael arrives to his office verify the reports of the movements registered on previous day by his assistants and the schedule of payments, purchases and deliveries programming for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in a shorter time. When the number of items requested exceeds the minimum the requesting area is informed an approximate date on which would receive the products requested. There are specific processes to the purchases depending of the kind of product required by the area of the company and Michael applies, helped by their assistants, the corresponding process.</p> </td></tr> <tr> <td colspan="2"> <p>Goals and Needs using the solution</p> <p>Michael expects to have information about the processes of his department at any moment he need it or he want to verify. Michael needs to know ahead of time the department's requests to verify if the products required are available in the store or, otherwise, buy them to deliver on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants needs to know amount, price and total that at the end of the day were implied in the movements, he expects has a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with their suppliers and the company's areas.</p> <p>"the main thing is to save without compromising the quality of the products purchased"</p> <p>"Delivery on time"</p> </td></tr> <tr> <td>Type of Persona</td><td>Primary</td></tr> </tbody> </table>	Persona			<p>Name: Michael Age: 35 Occupation: Accountant</p>	<p>Activities</p> <p>Michael is a 35 years old, he is a management accountant in charge of the company's store. Every day when Michael arrives to his office verify the reports of the movements registered on previous day by his assistants and the schedule of payments, purchases and deliveries programming for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in a shorter time. When the number of items requested exceeds the minimum the requesting area is informed an approximate date on which would receive the products requested. There are specific processes to the purchases depending of the kind of product required by the area of the company and Michael applies, helped by their assistants, the corresponding process.</p>		<p>Goals and Needs using the solution</p> <p>Michael expects to have information about the processes of his department at any moment he need it or he want to verify. Michael needs to know ahead of time the department's requests to verify if the products required are available in the store or, otherwise, buy them to deliver on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants needs to know amount, price and total that at the end of the day were implied in the movements, he expects has a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with their suppliers and the company's areas.</p> <p>"the main thing is to save without compromising the quality of the products purchased"</p> <p>"Delivery on time"</p>		Type of Persona	Primary
Persona
<p>Name: Michael Age: 35 Occupation: Accountant</p>										
<p>Activities</p> <p>Michael is a 35 years old, he is a management accountant in charge of the company's store. Every day when Michael arrives to his office verify the reports of the movements registered on previous day by his assistants and the schedule of payments, purchases and deliveries programming for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in a shorter time. When the number of items requested exceeds the minimum the requesting area is informed an approximate date on which would receive the products requested. There are specific processes to the purchases depending of the kind of product required by the area of the company and Michael applies, helped by their assistants, the corresponding process.</p>											
<p>Goals and Needs using the solution</p> <p>Michael expects to have information about the processes of his department at any moment he need it or he want to verify. Michael needs to know ahead of time the department's requests to verify if the products required are available in the store or, otherwise, buy them to deliver on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants needs to know amount, price and total that at the end of the day were implied in the movements, he expects has a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with their suppliers and the company's areas.</p> <p>"the main thing is to save without compromising the quality of the products purchased"</p> <p>"Delivery on time"</p>											
Type of Persona	Primary										
Templates	https://drive.google.com/file/d/0B1YqXzQvZlQuZ2ZJTzZJU0YwVzg/edit?usp=sharing										
Information Resources	<p>Cooper, A. (1999), Inmates are running the asylum, The: Why high-tech products drive us crazy and How to Restore the Sanity. Ed Sams.</p> <p>Cooper, A. ,Reimann R., Cronin D. (2007), About face 3: The essentials of Interaction Design. Ed John Wiley & Sons.</p> <p>Sharp, H., Rogers, Y., Preece J. (2007) , Interaction Design: beyond human-computer interaction. Ed John Wiley & Sons.</p> <p>Sanchez-Segura M., Medina-Domínguez F., Amescua A., Mora-Soto A. Improving the efficiency of use of software engineering practices using product patterns. Recovered May 2014 from http://www.sciencedirect.com</p>										

D.2. Patrón evaluado “Agile Persona” PAPv1

Name	Agile persona
Initial context	This pattern can be used in agile projects to help development team to learn and understand about the user's needs, goals and behavior in order to identify the final users and develop the right functionalities for them with useful and usable interfaces.
Result Context	Specific descriptions of the final users' features will be obtained. Such descriptions allow the development team to know about the users for whom the software will be developed. Those descriptions will be available to review at any moment through the agile development process.
Problem	The agile development team must be able to identify and define as soon as possible the final users to whom the software will be developed avoiding by this way waste of time in reworking through the development process.
Forces	<p>Kind of the organization This pattern can be used in agile projects regardless of company size. In order to get better results it can be implemented for clients who have well-defined their organizational processes.</p> <p>Kind of system to be developed This pattern can be used in agile projects where the user's requirements would change through the development. This product would be evaluated and updated according to the feedbacks received from the iterations' release reviews.</p> <p>Kind of client The user must be available and always involved through the process of the persona's definition.</p> <p>Rationale</p> <ul style="list-style-type: none"> ▪ Meetings for interviews, product pattern's review or feedback between development team and user must be established in a friendly and trusted environment. ▪ The product pattern persona must be keep available for the team through the whole development process at any moment.
Solution	<p>Agile development process</p> <ul style="list-style-type: none"> ▪ At the beginning of the development process the development team gets information through a Minimal User Research using observations and interviews. ▪ The development team describes the personas identified using the template proposed and prioritize the type of persona identified (primary or secondary) ▪ Persona's definition is reviewed by both product owner and team development ▪ If refinement is necessary development team update the persona's definition according to the product owner's review and verify the prioritization of persona updating it if is necessary ▪ Product backlog is defined and the user stories are related with the personas involved with them. ▪ Define the sprint backlog <ul style="list-style-type: none"> ▪ For each user story to develop in the Sprint we are going to: <ul style="list-style-type: none"> • Identify the users involved in the user story in order to know if the personas defined previously cover the archetypes needed in the user story. • Relate the personas' definition with the user stories to develop. Verify definitions with the personas related with the users involved in the user story. • Validate the refinement needed or creation for a new persona definition • If is necessary create new persona then minimal user research is applied • Else if is identified a necessary refinement to persona then we update the definition according the feedback gathered from the previous iteration or product owner's review. Verified the prioritization of persona and updated it if is necessary. • Define the user story details by specifying tasks and acceptance criteria related with it. ▪ When the last user story is detailed then we start with the sprint implementation (code, users' tests, etc). ▪ Deliver a shippable product ▪ Gathered feedback ▪ If is not the last Sprint then verify the product backlog and determine the next sprint backlog

Examples	<table border="1"> <thead> <tr> <th colspan="2">Persona</th> </tr> </thead> <tbody> <tr> <td data-bbox="549 215 724 416"> </td><td data-bbox="724 215 1270 416"> Name: Michael Age: 35 Occupation: Accountant </td></tr> <tr> <td colspan="2">Activities</td></tr> <tr> <td colspan="2"> Michael is a 35 years old, he is a management accountant in charge of the company's store. Every day when Michael arrives to his office verify the reports of the movements registered on previous day by his assistants and the schedule of payments, purchases and deliveries programming for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in a shorter time. When the number of items requested exceeds the minimum the requesting area is informed an approximate date on which would receive the products requested. There are specific processes to the purchases depending of the kind of product required by the area of the company and Michael applies, helped by their assistants, the corresponding process. </td></tr> <tr> <td colspan="2">Goals and Needs using the solution</td></tr> <tr> <td colspan="2"> Michael expects to have information about the processes of his department at any moment he need it or he want to verify. Michael needs to know ahead of time the department's requests to verify if the products required are available in the store or, otherwise, buy them to deliver on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants needs to know amount, price and total that at the end of the day were implied in the movements, he expects has a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with their suppliers and the company's areas. "the main thing is to save without compromising the quality of the products purchased" "Delivery on time" </td></tr> <tr> <td>Type of Persona</td><td>Primary</td></tr> </tbody> </table>	Persona			Name: Michael Age: 35 Occupation: Accountant	Activities		Michael is a 35 years old, he is a management accountant in charge of the company's store. Every day when Michael arrives to his office verify the reports of the movements registered on previous day by his assistants and the schedule of payments, purchases and deliveries programming for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in a shorter time. When the number of items requested exceeds the minimum the requesting area is informed an approximate date on which would receive the products requested. There are specific processes to the purchases depending of the kind of product required by the area of the company and Michael applies, helped by their assistants, the corresponding process.		Goals and Needs using the solution		Michael expects to have information about the processes of his department at any moment he need it or he want to verify. Michael needs to know ahead of time the department's requests to verify if the products required are available in the store or, otherwise, buy them to deliver on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants needs to know amount, price and total that at the end of the day were implied in the movements, he expects has a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with their suppliers and the company's areas. "the main thing is to save without compromising the quality of the products purchased" "Delivery on time"		Type of Persona	Primary
Persona
Name: Michael Age: 35 Occupation: Accountant														
Activities															
Michael is a 35 years old, he is a management accountant in charge of the company's store. Every day when Michael arrives to his office verify the reports of the movements registered on previous day by his assistants and the schedule of payments, purchases and deliveries programming for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in a shorter time. When the number of items requested exceeds the minimum the requesting area is informed an approximate date on which would receive the products requested. There are specific processes to the purchases depending of the kind of product required by the area of the company and Michael applies, helped by their assistants, the corresponding process.															
Goals and Needs using the solution															
Michael expects to have information about the processes of his department at any moment he need it or he want to verify. Michael needs to know ahead of time the department's requests to verify if the products required are available in the store or, otherwise, buy them to deliver on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants needs to know amount, price and total that at the end of the day were implied in the movements, he expects has a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with their suppliers and the company's areas. "the main thing is to save without compromising the quality of the products purchased" "Delivery on time"															
Type of Persona	Primary														
Templates	https://drive.google.com/file/d/0B1YqXzQvZlQuZ2ZJTzJU0YwVzg/edit?usp=sharing														
Information Resources	<p>Cooper, A. (1999), Inmates are running the asylum, The: Why high-tech products drive us crazy and How to Restore the Sanity. Ed Sams.</p> <p>Cooper, A. ,Reimann R., Cronin D. (2007), About face 3: The essentials of Interaction Design. Ed John Wiley & Sons.</p> <p>Sharp, H., Rogers, Y., Preece J. (2007) , Interaction Design: beyond human-computer interaction. Ed John Wiley & Sons.</p> <p>Sanchez-Segura M., Medina-Dominguez F., Amescua A., Mora-Soto A. Improving the efficiency of use of software engineering practices using product patterns. Recovered May 2014 from http://www.sciencedirect.com</p> <p>Caballero, L., Moreno, A., & Seffah, A. (2014). Persona as a Tool to Involving Human in Agile Methods: Contributions from HCI and Marketing. In S. Sauer, C. Bogdan, P. Forbrig, R. Bernhaupt, & M. Winckler (Eds.), Human-Centered Software Engineering SE - 20 (Vol. 8742, pp. 283–290). Springer Berlin Heidelberg. doi:10.1007/978-3-662-44811-3_20</p>														

D.3. Patrón evaluado “Agile Persona” PAPv2

Name	Agile persona
Initial context	This pattern can be used in agile projects to help development team to learn and understand about the user’s needs, goals and behavior in order to identify the final users and develop the right functionalities for them with useful and usable interfaces.
Result Context	Specific descriptions of the final users’ features will be obtained. Such descriptions allow the development team to know about the users for whom the software will be developed. Those descriptions will be available to review at any moment through the agile development process.
Problem	The agile development team must be able to identify and define as soon as possible the final users to whom the software will be developed avoiding by this way waste of time in reworking through the development process.
Forces	<p>Kind of the organization This pattern can be used in agile projects regardless of company size.</p> <p>Kind of system to be developed This pattern can be used in software projects that use development methodologies following the agile philosophy.</p> <p>Kind of client The user must be available and always involved through the process of the persona’s definition.</p>
Solution	<p>In agile environment, incorporating persona involves different activities: develop minimal user research, describe and prioritize persona descriptions, review persona definitions, update description/prioritize persona and obtain a detailed definition of the user story. Figure 1 depicts how persona is applied through these processes.</p> <p>In order to construct the user models we applied persona technique by following the next steps:</p> <ul style="list-style-type: none"> • Minimal User Research: development team develops minimal user research using artifacts such as interviews and observation in order to get just basic information about the user in order to identify relevant information to define user models. These models will be improve through subsequent iterations. • Describe and prioritize persona: using a template provided as part of the pattern the development team describe the user’s features, needs, goals and behaviors. Then, development team prioritizes personas identified. • Review persona definition: product owner and development team review the persona’s descriptions and prioritization assigned. • Update description/ prioritize persona: if a refinement is necessary, the personas’ descriptions are updated and prioritized again according to the feedback from previous review or product owner’ observations. • Detailed definition of the user story: once the persona’ description is finished then, tasks and acceptance criteria related with this can be detailed. In this way the user stories remain relative to their corresponding persona. <pre> graph TD Start(()) --> CreatePersona{Create persona?} CreatePersona -- No --> RefinementPersona{Refinement persona?} CreatePersona -- Yes --> MinimalUserResearch([Minimal User Research (Interviews, observation)]) MinimalUserResearch --> DescribePrioritize([Describe and prioritize persona]) DescribePrioritize -.-> Note1[Fill the persona's template: -Name, age, occupation -Activities -Goals using the solution -Type of persona] DescribePrioritize --> ReviewDefinition([Review persona Definition]) ReviewDefinition -.-> Note2[Product Owner and Development Team are involved in the review] ReviewDefinition --> RefinementPersona RefinementPersona -- Yes --> UpdatePrioritize([Update description and prioritize persona]) UpdatePrioritize -.-> Note3[Update the characteristics of persona according to the observations derived from the previous iteration's feedback or from the product owner's review] UpdatePrioritize --> DetailedUserStory([Detailed definition of the User Story]) DetailedUserStory -.-> Note4[Specify tasks and acceptance criteria] DetailedUserStory --> End((())) RefinementPersona -- No --> End </pre> <p>Figure 1- How to Apply persona at Agile environment</p> <p>Figure 2 depicts an activity diagram which shows who develops the corresponding activities regarding the roles involved. In general the activities are performed as described below:</p> <ul style="list-style-type: none"> ▪ The development team receives information previously gathered from the Minimal User Research (observation and interviews). ▪ The development team defines persona as user models with hypothetical archetypes describing

- the user's features identifying needs, goals and behavior variables.
- The development team prioritizes personas identified.
- The user review together with development team, the persona's description and prioritization.
- If it is necessary the development team refines persona's definitions by creating a new persona product pattern or by updating existing patterns after each iteration feedback or product owner's review.
- User stories can then be detailed according to the personas definition. This activity is developed by both product owner (establishing acceptance criteria) and development team (defining tasks).

Figure 2 Who are involved defining persona pattern

Figure 3 depicts a general way of the agile development process highlighting the moment to apply persona with a gray background. In this diagram sprint planning and sprint development has an asterisk (*) and a double asterisk (**) respectively to indicate a reference to the processes shown in figure 4 and 5 where persona technique is also applied. In the diagrams the process named “Apply persona” involves the activities described previously and depicted in figure 1.

Persona technique is applied at the beginning of the development process in an explorative way in order to get information about the users visualizing the general scope of the project considering this information to define initial product backlog and interface design. The agile team decides how deeply to conduct the user research to get necessary information.

Figure 3 When to apply persona through agile development process

At this moment, *the beginning of the development process*, “Apply persona” implies doing the following tasks according to the processes described previously with reference to figure 1:

- Do a minimum user research to get relevant information in order to define initial user’s descriptions.
- Once possible target users are identified then describe them and prioritize these descriptions.
- Review the persona definitions. This task is performed by the development team together with the product owner.
- If a necessary refinement to persona is identified an update to the definition is applied according to the feedback gathered from previous iterations or product owner’s review. Verify the prioritization of persona again and update it if necessary.
- Define the user story details by specifying tasks and acceptance criteria related to it.

Persona is applied in an explorative way at sprint planning and sprint development. At this time applying this technique aims to verify and improve previous persona’s descriptions. Figure 4 depicts when persona is applied in iteration planning and figure 5 when persona is applied in iteration development.

Figure 4 - When apply persona at iteration planning

During *iteration planning* the technique is applied by developing the following tasks:

- Review feedback from previous iterations (if there have been previous iterations)
- Relate the personas’ definition with the user stories to develop
- Determine if refinement is needed or creation for a new persona definition
 - If it is necessary to create a new persona, then minimal user research is applied and persona’s description defined, prioritized and reviewed.
 - If refinement to persona is necessary then update the definition according to the feedback gathered from the previous iteration or product owner’s review. Verify the prioritization of persona and updated it if necessary. This verification is applied by product owner and development team.
- Define the user story details by specifying tasks and acceptance criteria related to it.

Figure 5 - When apply persona at iteration development

At *iteration development* persona is applied by developing the following tasks. For each user story to develop in the iteration we are going to:

- Relate the personas' definition with the user stories to develop. Verifying definitions with the personas related with the users involved in the user story.
- Validate the refinement needed or creation of a new persona definition
 - If it is necessary create a new persona, then minimal user research is applied
 - If a necessary refinement to persona is identified we update the definition according to the feedback gathered from the previous iteration or product owner's review. Verify the prioritization of persona and updated it if is necessary.
- Define the user story details by specifying tasks and acceptance criteria related to it.
- Development team decides to implement the user story or continue to design next user story.

During iteration development parallel tracks of user stories implementation or design can be worked at the same time if the development team has enough elements and timeboxing.

When the last user story of the iteration backlog is done a shippable product is ready to be delivered. Feedback from persona incorporation through the iteration is available to be used at the iteration backlog for the next iteration(s).

Roles	<ul style="list-style-type: none"> ▪ UI Designer (optional) ▪ Agile Developer ▪ User (product owner)
Basic Knowledge	<p>Knowledge</p> <ul style="list-style-type: none"> ▪ Knowing about requirements gathering <p>Skills</p> <ul style="list-style-type: none"> ▪ Capability to apply user research to get data through direct observation, interviews and questionnaires. ▪ Capability to Data Analysis identifying needs and establishing requirements
Entries	User information
Exit	Persona's descriptions as user Models
Collaboration	---
Capability label	---
Lessons learned	<ul style="list-style-type: none"> ▪ The whole development team must be involved in the personas definition. ▪ After the UI designers and/or developers have prioritized the personas described the models have to be presented to review in a meeting to all the development team and users. ▪ Users, UI designers and developers must be involved in the presentation, reviews and feedbacks

	meetings.														
Examples	<table border="1"> <thead> <tr> <th colspan="2">Persona</th> </tr> </thead> <tbody> <tr> <td> </td><td> Name: Michael Age: 35 Occupation: Accountant </td></tr> <tr> <td colspan="2">Activities</td></tr> <tr> <td colspan="2"> Michael is a 35 years old. He is a management accountant in charge of a company's store. Every day when Michael arrives at his office he verifies the reports of movements registered on the previous day by his assistants and the schedule of payments, purchases and deliveries programed for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in the least time. When the number of items requested exceeds the minimum the requesting area is informed for an approximate date on which they will receive the products requested. There are specific processes for the purchases depending on the kind of product required by the company and Michael applies, helped by his assistants, the corresponding process. </td></tr> <tr> <td colspan="2">Goals and Needs using the solution</td></tr> <tr> <td colspan="2"> Michael expects to have information about the processes of his department at any moment. He needs it and wants to verify. Michael needs to know ahead of time the departments' requests to verify if the products required are available in the store or, otherwise, buy them for delivery on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants need to know the amount, price and total that at the end of the day were implied in the movements. He expects to have a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with his suppliers and the company's departments. "the main thing is to save without compromising the quality of the products purchased" "Delivery on time" </td></tr> <tr> <td>Type of Persona</td><td>Primary</td></tr> </tbody> </table>	Persona			Name: Michael Age: 35 Occupation: Accountant	Activities		Michael is a 35 years old. He is a management accountant in charge of a company's store. Every day when Michael arrives at his office he verifies the reports of movements registered on the previous day by his assistants and the schedule of payments, purchases and deliveries programed for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in the least time. When the number of items requested exceeds the minimum the requesting area is informed for an approximate date on which they will receive the products requested. There are specific processes for the purchases depending on the kind of product required by the company and Michael applies, helped by his assistants, the corresponding process.		Goals and Needs using the solution		Michael expects to have information about the processes of his department at any moment. He needs it and wants to verify. Michael needs to know ahead of time the departments' requests to verify if the products required are available in the store or, otherwise, buy them for delivery on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants need to know the amount, price and total that at the end of the day were implied in the movements. He expects to have a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with his suppliers and the company's departments. "the main thing is to save without compromising the quality of the products purchased" "Delivery on time"		Type of Persona	Primary
Persona
Name: Michael Age: 35 Occupation: Accountant														
Activities															
Michael is a 35 years old. He is a management accountant in charge of a company's store. Every day when Michael arrives at his office he verifies the reports of movements registered on the previous day by his assistants and the schedule of payments, purchases and deliveries programed for the day. Michael verifies that every payment was registered correctly controlling and comparing the budget. The store has established a minimum quantity to supply in the least time. When the number of items requested exceeds the minimum the requesting area is informed for an approximate date on which they will receive the products requested. There are specific processes for the purchases depending on the kind of product required by the company and Michael applies, helped by his assistants, the corresponding process.															
Goals and Needs using the solution															
Michael expects to have information about the processes of his department at any moment. He needs it and wants to verify. Michael needs to know ahead of time the departments' requests to verify if the products required are available in the store or, otherwise, buy them for delivery on time. Michael expects to be informed in advance about those products stocks available below the minimum established to acquire more and fill the store. Michael and his assistants need to know the amount, price and total that at the end of the day were implied in the movements. He expects to have a better budget control. Michael expects to generate savings in procurement purchasing lower-priced products and equal or better quality. He wants to improve the communication with his suppliers and the company's departments. "the main thing is to save without compromising the quality of the products purchased" "Delivery on time"															
Type of Persona	Primary														
Templates	https://drive.google.com/file/d/0B1YqXzQvZlQuZ2ZJTzJU0YwVzg/edit?usp=sharing														
Information Resources	<p>Cooper, A. (1999), Inmates are running the asylum, The: Why high-tech products drive us crazy and How to Restore the Sanity. Ed Sams.</p> <p>Cooper, A. ,Reimann R., Cronin D. (2007), About face 3: The essentials of Interaction Design. Ed John Wiley & Sons.</p> <p>Sharp, H., Rogers, Y., Preece J. (2007) , Interaction Design: beyond human-computer interaction. Ed John Wiley & Sons.</p> <p>Sanchez-Segura M., Medina-Dominguez F., Amescua A., Mora-Soto A. Improving the efficiency of use of software engineering practices using product patterns. Recovered May 2014 from http://www.sciencedirect.com</p> <p>Caballero, L., Moreno, A., & Seffah, A. (2014). Persona as a Tool to Involving Human in Agile Methods: Contributions from HCI and Marketing. In S. Sauer, C. Bogdan, P. Forbrig, R. Bernhaupt, & M. Winckler (Eds.), Human-Centered Software Engineering SE - 20 (Vol. 8742, pp. 283–290). Springer Berlin Heidelberg. doi:10.1007/978-3-662-44811-3_20</p>														

D.3. Patrón evaluado “Agile Usability Testing” PAUTv0

Name	Agile Usability Testing
Initial context	This pattern can be used in agile projects to help development team to validate design and functionality decisions with the end users aiming to identify those features to improve and/or include on the development software in order to deliver an useful software.
Result Context	Observations about user interface and functionality are gathered from the user through tests applied directly with them. Such observations are analyzed, prioritized and implemented on the software allowing delivering a useful software.
Problem	The agile development team must to be able to identify and implement as soon as possible those features that assure delivering useful software.
Forces	<p>Kind of the organization This pattern can be used in agile projects regardless of company size.</p> <p>Kind of system to be developed This pattern can be used in agile projects where the user’s requirements would change through the development. This product would be evaluated and updated according to the usability testing feedbacks received from the beginning, iterations, releases and implementation reviews.</p> <p>Kind of client The end-users or representative users must be available and always involved in usability testing.</p> <p>Rationale</p> <ul style="list-style-type: none"> ▪ Meetings for interviews, product pattern’s review or feedback between development team and user must be established in a friendly and trusted environment. ▪ The product pattern “agile usability testing” must be keep available for the team through the whole development process at any moment.
Solution	<ul style="list-style-type: none"> ▪ The process of usability testing from the pattern (figure a) involves planning the testing, apply the tests, report and analyze the users’ observations. <ul style="list-style-type: none"> ○ Planning the tests, in order to get valuable information from the usability testing important features must be defined for each usability testing: firstly the objective of the tests; knowing the goal why is necessary apply the tests to the users assure define the tasks focused on the behavior target from the users that the testers and developers will be looking to observe from them. The place to apply the test must be defined according to the capacity, availability of spaces, advance of development and tools. Usability testing could be applied in-home, remotely or on-site. Is highly recommended apply the tests with end users of the software; however is well know that it is not always possible. Nevertheless usability testing must be applied with end users or representative users developers or team elements like users is not recommended. Next, define the tasks to be applied on the usability testing. It must be focused to gathering information related with the testing’s goal. Finally the users are recruiting and the usability testing applied. ○ Apply usability tests, the usability testing must be recorded in some way, either video recording, sound recording or capture screen windows (by some software tool). Developers must be present on the usability testing and those stakeholders involved in some way with the software development. During the usability session the observations are registered in a list. ○ Subsequently the usability feedback is analyzed by the whole team. Those features which do not represent major changes and can be corrected before the end of the iteration are entered into the tracking system and implemented. New feature requested or too large to be implemented are considered an enhancement required and classified in the review list as urgent, important or desirable to have. These features are considered to be entered into the project backlog to subsequent iterations. ○ Once the usability testing review ended and all the features have been analyzed then the shippable product of the iteration is delivered.

- At the beginning of the development software is gathering enough information from users in order to have a general scope of the project.
- Usability testing is applied through the iterations, end of release and after software launch.
- If agile team decides get feedback from formative usability testing, initial features can be designed with prototypes (low-fi prototypes like paper prototypes) to be evaluated with usability testing and then from the feedback of this review define the project backlog.
- At the beginning of the iterations development, formative usability testing can be considered or apply usability testing at the end of the iteration.
- Usability feedback gathered is considered and available to the whole development team.
- Usability testing can be applied at the end of each release and highly recommended to apply when the development process is complete. This test would be applied through questionnaires or surveys provided to the users which are actually working the software.

Leydi Ofelia Caballero Chi

	<ul style="list-style-type: none">▪ Capability to apply user research to get data through direct observation, interviews and questionnaires.▪ Capability to Data Analysis identifying needs and establishing requirements▪ Capability to interact with the users in a friendly way.																																																																																																									
Entries	Usability testing report																																																																																																									
Exit	Reviewed usability Testing List per iteration																																																																																																									
Collaboration	---																																																																																																									
Capability label	---																																																																																																									
Lessons learned	<ul style="list-style-type: none">▪ User tests could be applied either an usability expert or team element with previous education on how to apply tests.▪ Developers must be present on the usability testing▪ After the tests have been applied the results must be presented to the development team in a meeting.▪ The whole development team must be involved in the usability testing reporting to review and analyzing it to prioritize the users' observations into bugs to fix at the iteration or enhancements required to be considered for the next iterations.▪ Usability testing should be applied in a relaxed and friendly atmosphere aiming to get sincerely comments from the users.▪ Design and development must be worked on parallel tracks with designs to evaluate available at least one cycle ahead.																																																																																																									
Examples	<table><tr><th colspan="3">Usability Testing Review List</th></tr><tr><td colspan="2">ID Usability Testing:</td><td>UTR 0302</td></tr><tr><td colspan="2">Date Review:</td><td>20/Jan/2015</td></tr><tr><td colspan="2">ID Iteration:</td><td>03</td></tr><tr><th>ID</th><th>Observation</th><th>Classification*</th></tr><tr><td>01</td><td>Bottom "cancel" has not a the first letter on Uppercase</td><td>B</td></tr><tr><td>02</td><td>There is no option to apply taxes per city</td><td>FR</td></tr><tr><td>03</td><td>Date field allow to capture month 13</td><td>B</td></tr><tr><td>04</td><td>There is no confirmation window when user save the information</td><td>B</td></tr><tr><td>05</td><td>Principal window close when option "search name" is selected</td><td>B</td></tr><tr><td>06</td><td>The list of products doesn't show the subtotal and total including IVA</td><td>FR</td></tr><tr><td>07</td><td></td><td></td></tr><tr><td>08</td><td></td><td></td></tr><tr><td>09</td><td></td><td></td></tr><tr><td>10</td><td></td><td></td></tr><tr><td>11</td><td></td><td></td></tr><tr><td>12</td><td></td><td></td></tr><tr><td>13</td><td></td><td></td></tr><tr><td>14</td><td></td><td></td></tr><tr><td>15</td><td></td><td></td></tr><tr><td>16</td><td></td><td></td></tr><tr><td>17</td><td></td><td></td></tr><tr><td>18</td><td></td><td></td></tr><tr><td>19</td><td></td><td></td></tr><tr><td>20</td><td></td><td></td></tr><tr><td>21</td><td></td><td></td></tr><tr><td>22</td><td></td><td></td></tr><tr><td>23</td><td></td><td></td></tr><tr><td>24</td><td></td><td></td></tr><tr><td>25</td><td></td><td></td></tr><tr><td>26</td><td></td><td></td></tr><tr><td>27</td><td></td><td></td></tr><tr><td>28</td><td></td><td></td></tr><tr><td>29</td><td></td><td></td></tr><tr><td>30</td><td></td><td></td></tr></table> <p>*B = Bug to fixed on the iteration in course FR = Feature required to be considered on the project backlog</p>	Usability Testing Review List			ID Usability Testing:		UTR 0302	Date Review:		20/Jan/2015	ID Iteration:		03	ID	Observation	Classification*	01	Bottom "cancel" has not a the first letter on Uppercase	B	02	There is no option to apply taxes per city	FR	03	Date field allow to capture month 13	B	04	There is no confirmation window when user save the information	B	05	Principal window close when option "search name" is selected	B	06	The list of products doesn't show the subtotal and total including IVA	FR	07			08			09			10			11			12			13			14			15			16			17			18			19			20			21			22			23			24			25			26			27			28			29			30		
Usability Testing Review List																																																																																																										
ID Usability Testing:		UTR 0302																																																																																																								
Date Review:		20/Jan/2015																																																																																																								
ID Iteration:		03																																																																																																								
ID	Observation	Classification*																																																																																																								
01	Bottom "cancel" has not a the first letter on Uppercase	B																																																																																																								
02	There is no option to apply taxes per city	FR																																																																																																								
03	Date field allow to capture month 13	B																																																																																																								
04	There is no confirmation window when user save the information	B																																																																																																								
05	Principal window close when option "search name" is selected	B																																																																																																								
06	The list of products doesn't show the subtotal and total including IVA	FR																																																																																																								
07																																																																																																										
08																																																																																																										
09																																																																																																										
10																																																																																																										
11																																																																																																										
12																																																																																																										
13																																																																																																										
14																																																																																																										
15																																																																																																										
16																																																																																																										
17																																																																																																										
18																																																																																																										
19																																																																																																										
20																																																																																																										
21																																																																																																										
22																																																																																																										
23																																																																																																										
24																																																																																																										
25																																																																																																										
26																																																																																																										
27																																																																																																										
28																																																																																																										
29																																																																																																										
30																																																																																																										
Templates	https://drive.google.com/file/d/0B1YqXzQvZlQuYXljZTdvaJFKOTA/view?usp=sharing																																																																																																									
Information Resources	<p>Dumas, J. S., & Redish, J. C. (1999). A practical guide to Usability Testing. (I. Books, Ed.). Intellect Ltd.</p> <p>Sharp, H., Rogers, Y., Preece J. (2007) , Interaction Design: beyond human-computer interaction. Ed John Wiley & Sons.</p> <p>Sanchez-Segura M., Medina-Dominguez F., Amescua A., Mora-Soto A. Improving the efficiency of use of software engineering practices using product patterns. Recovered May 2014 from http://www.sciencedirect.com</p> <p>Nielsen, J. (2008). Agile Development Projects and Usability. Jakob Nielsen's Alertbox. Retrieved from http://www.useit.com/alertbox/agile-methods.htm</p> <p>Barnum, C. M. (2010). Usability Testing Essentials. Morgan Kaufmann.</p>																																																																																																									

Anexo E. Pruebas piloto a la estructura de la entrevista

E.1. Ejemplo del instrumento entrevista EP01 utilizado en las pruebas piloto

Paso 1. Explicación sobre el estudio, el beneficio que representa para ellos y la estructura de la entrevista.

Paso 2. Preguntas sobre la facilidad/dificultad de uso de las técnicas de usabilidad

- Ha sido fácil o difícil utilizar la técnica persona?
- Ha sido fácil o difícil utilizar la técnica “Usability Testing”?

Paso 3. Preguntas sobre la utilidad de las técnicas de usabilidad

- Utilizaste el patrón “Agile Persona”?
- El patrón “Agile Persona” se entiende?
- Ha sido útil la herramienta “Persona”?
- La plantilla para definir persona te proporciona la información necesaria para identificar a tus usuarios finales?
- Volverías a utilizar la técnica “Persona”?
- Utilizaste el patrón “Agile Usability Testing”?
- El patrón “Agile Usability Testing” se entiende?
- Ha sido útil la herramienta “Usability Testing”?
- La plantilla del feedback para aplicar las pruebas de usabilidad te permite realizarlo de una manera sencilla?
- Volverías a utilizar la técnica “Usability Testing”?
- Cuál de las dos herramientas te ha resultado más útil para el proceso de desarrollo de software que utilizas?

Paso 4. Preguntas sobre la experiencia con el uso de metodologías ágiles

- Cuál fue el enfoque de implementación de las herramientas de usabilidad que utilizaron durante su desarrollo ágil? (SCRUM)
- Cuál es tu experiencia en general sobre utilizar una metodología ágil para el desarrollo de software?
- Cuál ha sido tu experiencia implementando usabilidad durante el desarrollo de software?
- Cuál ha sido tu experiencia incorporando herramientas de usabilidad durante el desarrollo ágil de software?

E.2. Ejemplo del instrumento entrevista EP02 utilizado en las pruebas piloto

Paso 1. Explicación sobre el contenido y estructura de la entrevista.

Paso 2. Preguntas sobre la facilidad/dificultad de utilizar los productos patrón

- Cuáles fueron las complicaciones que se tuvieron para entender los patrones “persona” y “Usability Testing”?
- Cuáles fueron las complicaciones que se tuvieron para seguir las instrucciones de los patrones?
- Aspectos positivos de los patrones utilizados (persona y usability testing)

Paso 3. Preguntas sobre la utilidad de los productos patrón

- El patrón sirvió como guía para implementar las técnicas de usabilidad durante el proceso del desarrollo de software?
- La información proporcionada por los patrones facilitó que usaran las técnicas de usabilidad durante el proceso ágil?
- Alguna mejora que se sugiera para que sea más fácil utilizar el patrón como guía?
- Aplicarían de nuevo las técnicas siguiendo las instrucciones del patrón?

E.3. Registro de aplicaciones de las entrevistas correspondientes a las prueba piloto EP01 y EP02

Las tablas E-1 y E -2 presentan los registros de las entrevistas que fueron aplicadas como parte de las pruebas piloto para evaluar el contenido del instrumento final. La columna “Grupo de Evaluación” corresponde al identificador correspondiente a cada uno de los grupos que utilizaron los productos patrón (PRAU = Proyecto de evaluación en un ambiente universitario y PREI = Proyecto de evaluación en la industria), “ID Entrevistado” es el identificador asignado a cada elemento de los equipos, “Fecha entrevista / Hora inicio” dato sobre la fecha de la entrevista y la hora en la que fue aplicada, “Duración de la entrevista” corresponde al tiempo que duró en aplicarse la entrevista y “Registro / Almacenamiento” es el tipo de archivo en el que se almacenó la entrevista y el sitio donde se resguardó para su posterior consulta y análisis.

Grupo de Evaluación	ID Entrevistado	Fecha entrevista/Hora inicio	Duración de la entrevista	Registro / Almacenamiento
PREI	PEEI_E1_E1	17 Nov 2015	30 mins	Excel / Dropbox
PREI	PEEI_E1_E2	17 Nov 2015	30 mins	Excel / Dropbox
PREI	PEEI_E1_E3	17 Nov 2015	20 mins	Excel / Dropbox
PREI	PEEI_E2_E1	17 Nov 2015	20 mins	Excel / Dropbox
PREI	PEEI_E2_E2	17 Nov 2015	30 mins	Excel / Dropbox
PREI	PEEI_E3_E1	17 Nov 2015	30 mins	Excel / Dropbox
PREI	PEEI_E3_E2	17 Nov 2015	30 mins	Excel / Dropbox
PREI	PEEI_E3_E3	17 Nov 2015	30 mins	Excel / Dropbox
PREI	PEEI_E4_E1	17 Nov 2015	30 mins	Excel / Dropbox
PREI	PEEI_E4_E2	17 Nov 2015	30 mins	Excel / Dropbox
PREI	PEEI_E4_E3	17 Nov 2015	30 mins	Excel / Dropbox
PREI	PEEI_E4_E4	17 Nov 2015	30 mins	Excel / Dropbox

Tabla E-1. Registro de las entrevistas realizadas como parte de la prueba piloto EP01

Grupo de Evaluación	ID Entrevistado	Fecha entrevista/Hora inicio	Duración de la entrevista	Registro Almacenamiento /
PRAU	PEAU_E3_E1	03 Feb 2016 / 16:30	40 mins	Audio – Google Drive
PRAU	PEAU_E3_E2	03 Feb 2016 / 16:30	40 mins	Audio – Google Drive
PRAU	PEAU_E3_E3	03 Feb 2016 / 16:30	40 mins	Audio – Google Drive
PRAU	PEAU_E3_E4	03 Feb 2016 / 16:30	40 mins	Audio – Google Drive
PRAU	PEAU_E1_E1	04 Feb 2016 / 16:30	30 mins	Audio – Dropbox
PRAU	PEAU_E1_E2	04 Feb 2016 / 16:30	30 mins	Audio – Dropbox
PRAU	PEAU_E1_E3	04 Feb 2016 / 16:30	30 mins	Audio – Dropbox
PRAU	PEAU_E1_E4	04 Feb 2016 / 16:30	30 mins	Audio – Dropbox
PREI	PEEI_E1_E1	04 Feb 2016 / 20:00	32 mins	Video – Google Drive
PREI	PEEI_E1_E2	04 Feb 2016 / 20:30	14:50 mins	Video – Google Drive
PRAU	PEAU_E2_E1	05 Feb 2016 / 17:00	30 mins	Audio – Dropbox
PRAU	PEAU_E2_E2	05 Feb 2016 / 17:00	30 mins	Audio – Dropbox

Tabla E-2. Registro de las entrevistas realizadas como parte de las pruebas piloto EP02

Anexo F. Instrumentos utilizados: Entrevista y Encuesta

F.1. Ejemplo del instrumento entrevista aplicado a los equipos ágiles de proyectos reales.

Paso 1. Explicación sobre el contenido y estructura de la entrevista.

Paso 2. Preguntas sobre la facilidad/dificultad de utilizar el producto patrón “Agile Persona”

- Fue complicado entender la estructura del patrón “Agile Persona”?
- La información proporcionada en el documento del patrón “Agile Persona” facilitó la implementación de la técnica “persona” durante el proceso de desarrollo ágil?
- Utilizar el patrón “Agile persona” durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones?
- Utilizar la solución propuesta en el patrón “Agile Persona” (Campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales?
- Utilizarías de nuevo el patrón “Agile Persona” durante el desarrollo de software con metodologías ágiles?
- Aspectos más positivos del patrón “Agile Persona”?
- Aspectos negativos del patrón “Agile Persona”?
- Sugerencias para mejorar la utilidad del patrón “Agile Persona”?

Paso 3. Preguntas sobre la facilidad/dificultad de utilizar el producto patrón “Agile Usability Testing”

- Fue complicado entender la estructura del patrón “Agile Usability Testing”?
- La información proporcionada en el documento del patrón “Agile Usability Testing” facilitó la implementación de la técnica “Usability Testing” durante el proceso de desarrollo ágil?
- Utilizar el patrón “Agile Usability Testing” durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones?
- Utilizar la solución propuesta en el patrón “Agile Usability Testing” (Campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales?
- Utilizarías de nuevo el patrón “Agile Usability Testing” durante el desarrollo de software con metodologías ágiles?
- Aspectos más positivos del patrón “Agile Usability Testing”?
- Aspectos negativos del patrón “Agile Usability Testing”?
- Sugerencias para mejorar la utilidad del patrón “Agile Usability Testing”?

Paso 4. Preguntas sobre el resultado de la aplicación de las técnicas de usabilidad en el proceso de desarrollo ágil. Se resaltó a los entrevistados el hecho de responder considerando lo percibido a partir de la incorporación de las técnicas de usabilidad durante el desarrollo.

“Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...”

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios?
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios?
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario?
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios?

F.2. Formato de la encuesta publicada on-line para recabar datos de los desarrolladores.

Las figuras F.2.1, F.2.2 y F.2.3 muestran respectivamente los cuestionamientos definidos en la encuesta en relación con la facilidad/dificultad de utilizar el producto patrón “Agile Persona”, facilidad/dificultad de utilizar el producto patrón “Agile Usability Testing” y los resultados percibidos a partir de la incorporación de las técnicas de usabilidad.

Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

1. Nombre del proyecto de software en el que utilizaste los patrones

* 2. Patrón **"Agile persona"**

	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en desacuerdo
Fué complicado entender la estructura del patrón "Agile Persona"	☐	☐	☐	☐	☐
La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil?	☐	☐	☐	☐	☐
Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	☐	☐	☐	☐	☐
Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales	☐	☐	☐	☐	☐
Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles	☐	☐	☐	☐	☐

* 3. Lista los aspectos más positivos del patrón "Agile persona"

a)

b)

c)

* 4. Lista los aspectos más negativos del patrón "Agile Persona"

a)

b)

c)

5. Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Sig.

Figura F.2.1. Cuestionamientos de la encuesta para obtener información sobre la facilidad/dificultad de aplicar la técnica persona utilizando el producto patrón “Agile Persona”

Utilizando patrones para incorporar técnicas de usabilidad dentro el proceso de desarrollo ágil de software

Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

* 6. Patrón "Agile Usability Testing"

	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En desacuerdo	Completamente en desacuerdo
Fué complicado entender la estructura del patrón "Agile Usability Testing"	☐	☐	☐	☐	☐
La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil	☐	☐	☐	☐	☐
La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software	☐	☐	☐	☐	☐
Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones	☐	☐	☐	☐	☐
Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles	☐	☐	☐	☐	☐

* 7. Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a)
- b)
- c)

* 8. Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a)
- b)
- c)

9. Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Ant.

Sig.

Figura F.2.2. Cuestionamientos de la encuesta para obtener información sobre la facilidad/dificultad de aplicar la técnica Usability Testing utilizando el producto patrón "Agile Usability Testing"

Utilizando patrones para incorporar técnicas de usabilidad dentro el proceso de desarrollo ágil de software

Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

* 10. Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

	Completamente de acuerdo	Muy de acuerdo	De acuerdo	En Desacuerdo	Completamente en Desacuerdo
Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios	☐	☐	☐	☐	☐
Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios	☐	☐	☐	☐	☐
Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario	☐	☐	☐	☐	☐
En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios	☐	☐	☐	☐	☐

Ant.

Listo

Figura F.2.3. Resultados percibidos a partir de la incorporación de las técnicas de usabilidad persona y Usability Testing

Anexo G. Proyecto real de Evaluación en Ambiente Universitario PRAU

G.1. Encuestas grupo PRAU

COMPLETAS (Encuestado 3)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

Juego 1 - Equipo 1

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Completamente de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Completamente de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Facilidad de entender el patrón
- b) Organización de la información
- c) Facilidad para definir los perfiles

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Hubo necesidad de leer cuidadosamente la parte de "Solution"
- b) No se comprendieron las partes de "Collaboration" y "Capability Label"
- c) Prestar atención a los diagramas ya que estos pueden dificultar el entendimiento del patrón

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Revisar las partes de "Collaboration" y "Capability Label" y verificar si realmente son de utilidad

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Completamente de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Completamente de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Organización de la información
- b) Facilidad de entender el patrón
- c) Facilidad de aplicar las pruebas

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Los diagramas pueden complicar un poco el entendimiento
- b) La sección "Capability label" no proporciona mucha información
- c) Hubo necesidad de leer cuidadosamente la parte de "Solution"

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Revisar la sección de "Capability label" para añadirle información

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Completamente de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios En Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario En Desacuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Completamente de acuerdo

COMPLETAS (Encuestado 4)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

Equipo 1 - Juego 1

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Muy de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Muy de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Buena redacción
- b) Buena estructura
- c) Fácil de entender

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Sacar una versión en español

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Describir en un poco mas algunos campos que no tienen algún ejemplo

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Muy de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Muy de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Buena estructura
- b) Buena redaccion
- c) Buenos ejemplos

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Ninguno

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Describir un poco mas los campos

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Muy de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios Muy de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario Muy de acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Muy de acuerdo

COMPLETAS (Encuestado 5)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

Equipo 1 Juego 1

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Completamente de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Completamente de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Es posible usarlo en equipos de desarrollo tan grandes como los de empresas completas como en equipos pequeños
- b) Es un patrón fácil de entender y aplicar
- c) Al haber cambios en los requerimientos es más fácil de realizarlos gracias al patrón

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Es un poco más difícil de aplicar al desarrollo de videojuegos
- b) Ninguna
- c) Ninguna

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Ninguna

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Completamente de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Completamente de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Completamente en desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Permiten realizar cambios fácilmente en el desarrollo cuando los requisitos se modifican de cualquier forma
- b) Permiten realizar cambios sin retrasos en el desarrollo cuando los requisitos se modifican de cualquier forma
- c) Ninguno

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Ninguno
- b) Ninguno
- c) Ninguno

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Ninguna

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Completamente de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios Completamente en Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario Completamente de acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Muy de acuerdo

COMPLETAS (Encuestado 6)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

Aprende a leer y a escribir con Linux

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" De acuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Muy de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Muy de acuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Muy de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles De acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Nos da el ámbito del proyecto o alcance del proyecto
- b) Nos ayuda a recabar especificación de requisitos
- c) Con base a persona, el equipo de trabajo se centra en los objetivos del proyecto , a quien va dirigido

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Requiere de tiempo extra para documentar
- b) Adoptar el patrón puede ser molesto para un maquilador de software
- c) El nombre es ambiguo, no da a entender el objetivo del patrón

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Cambiar el nombre de persona por un nombre mas significativo a lo que en realidad significa

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Muy de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Completamente de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Muy de acuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Trata de garantizar un software que es a la medida a lo que el usuario pide
- b) Ayuda al equipo de desarrollo a tener una retroalimentación de las observaciones dadas en una prueba
- c) Es de mejora continua y se apega mucho a la metodología ágil SCRUM

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Requiere de tiempo para diseñar los patrones de usabilidad sea este por ejemplo un prototipo
- b) No hay la cultura de hacerlo cuando se desarrolla un programa
- c) Estamos acostumbrados a hacer pruebas demasiado tarde e ignoramos el concepto de pruebas de usabilidad

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Facilitar la elaboración de los prototipos, apoyo de más personal dedicado solo a la documentación de las pruebas de usabilidad

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Muy de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios Muy de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario Muy de acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Muy de acuerdo

COMPLETAS (Encuestado 7)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

Aprende a leer y a escribir con Linux

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" Muy de acuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? De acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones De acuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Muy de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Plantilla con elementos necesarios
- b) Obtención de información consistente
- c) Información ordenada

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Información confusa al principio

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Omitir detalles más lagos y anexar a otro documento

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" Completamente de acuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Muy de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Muy de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones De acuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Información precisa de contener
- b) Los datos son fáciles de obtener
- c) Agilidad de obtener información eficiente

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Ninguna

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Evitar tantas actividades o llenar de información confusa o innecesaria

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios De acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios Muy de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario Completamente de acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Muy de acuerdo

COMPLETAS (Encuestado 8)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

Equipo3

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Muy de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Muy de acuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Completamente de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles En desacuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Abstractar características del usuario final.
- b) Mejor entendimiento del comportamiento del usuario final.
- c) N/A

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Tener que validar si está bien o no el patrón una vez realizado.
- b) Suponer varias cosas sobre el usuario final a veces es malo.
- c) N/A

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Que desde el principio se integre al Product Owner para guiarnos en el desarrollo del perfil de la Persona.

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Muy de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Completamente de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones De acuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Detección de errores a tempranas etapas del desarrollo del software.
- b) Visualización de cómo respondían los usuarios ante la usabilidad del software.
- c) Recursos extras como grabación de audio y video.

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) A veces es complejo reunir a las personas indicadas (en nuestro caso, niños de primaria)
- b) N/A
- c) N/A

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Ninguna :)

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Completamente de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios De acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario De acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Muy de acuerdo

COMPLETAS (Encuestado 10)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

Se omitió esta pregunta

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" De acuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? De acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones De acuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales De acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles De acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) permite identificar aspectos importantes para ser tomados en cuenta a la hora de la planeacion
- b) permite ir de acuerdo con los objetivos propuestos
- c) la estructura que tiene

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) el tiempo
- b) la estructura
- c) las sub-etapas

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Se omitió esta pregunta

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" De acuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil De acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software De acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones De acuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles De acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) permite ir paso a paso
- b) detectar errores y componerlos
- c) que el juego este bien en todos los aspectos

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) el tiempo
- b) la poblacion a la que va dirigida
- c) la estructura

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Se omitió esta pregunta

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios De acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios De acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario De acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios De acuerdo

COMPLETAS (Encuestado 11)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

SCRUM

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" Muy de acuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Muy de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Muy de acuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales De acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Facilita perfilar a la persona
- b) Me da una solución más exacta

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Tarda un poco el tiempo de aplicación

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Se debe considerar poner más ejemplos

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" Muy de acuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil De acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Muy de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Muy de acuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Me da una solución más específica
- b) Es fácil de entender

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) hay que dedicarle un determinado tiempo

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

ninguna

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios De acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios De acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario De acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios De acuerdo

COMPLETAS (Encuestado 23)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

Equipo 1

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? De acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Muy de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Organización
- b) Pertinente
- c) Eficiente

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) No muy precioso en algunas partes

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Se omitió esta pregunta

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil De acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software De acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles De acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Organización
- b) Pertinencia

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) No es preciso en algunas partes

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Se omitió esta pregunta

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios De acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios De acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario De acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Muy de Acuerdo

COMPLETAS (Encuestado 26)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

Juegos Tux

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona"De acuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil?Muy de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteracionesMuy de acuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finalesDe acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágilesEn desacuerdo
- P3: Lista los aspectos más positivos del patrón "Agile persona"
- a)Definir el perfil de usuario de una manera natural
- P4: Lista los aspectos más negativos del patrón "Agile Persona"
- a)None
- P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Se omitió esta pregunta

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing"De acuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágilMuy de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del softwareEn desacuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteracionesEn desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágilesEn desacuerdo
- P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"
- a)Listas los pasos de las tareas
- P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"
- a)None
- P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Se omitió esta pregunta

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuariosDe acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuariosCompletamente en Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuarioCompletamente en Desacuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuariosCompletamente en Desacuerdo

G.2. Extractos de Entrevistas PRAU

Entrevistado PRAU_E05	
P.A1.1.	“Fue la más fácil de entender entre las dos”
P.A2.1.	“Sí, estaba cada punto específicamente que es lo que había que hacer...era bastante específico...”
P.A2.2.	“No, de hecho creo que el no usarlos generaría los atrasos...con el patrón persona como que pudimos ver al usuario...sus características y a partir de ello diseñar la interfaz”
P.A2.3.	“...sí, porque en esa parte donde decía “Solution”...decía haz tu diseño de persona y luego haz que tu product owner lo vea...yo creo que sí ayudó a hacerlo más específico...”
P.A1.3.	“...A partir de ello uno ya tiene la idea de a que tipo de usuario final va a desarrollar su software o para quién va a ir...y a partir de ello puede diseñar mejor las interfaces o mejorar la usabilidad del software”
P.A2.6.	“Sí, de hecho ahí decía que el momento era principio, en medio y al final. Y pues, obviamente si lo hago al principio voy a saber qué está mal, si lo hago en medio voy a saber qué está mal y está mejor así porque no espero al final a que me diga esta mal vuélvelo a hacer...esta mejor así como que por intervalos...”
P.A2.5.	“No, de hecho...lo único que nos llevó tiempo fue leer el patrón y hacer nuestro documento con pruebas de usabilidad pero fuera de ello no, de hecho nos ayudó”.
P.A3.6	“Sí...preferiría el de usability testing que el de persona...porque el de usability testing como que yo estoy con el usuario viendo como maneja mi software, en cambio con el de persona pues solamente tengo una idea de cómo es mi usuario final...con el de usability testing yo puedo ver cómo reacciona este y además lo uso tanto al principio, como en medio y al final se me hace más interactivo”
P.A1.4.	“...lo que me gustó es que te pedían grabar audio o video y después analizar este y estaba bien porque en el momento en el que tú estabas presente habían cosas que no analizabas, ya más adelante podías ver el video más detenidamente y ver cosas que no habías visto antes...entonces eso me gustó, lo de grabar”
P.A3.4.	“Sí, porque como dije, en lo que era de desarrollo como que el usuario si estaba involucrado. En cambio, como está involucrado pues puede ver lo que está pasando...puede ver de una vez “esto está mal, no es así”, en cambio anteriormente en proyectos anteriores pues no, era hazlo y se lo entregas al final y obviamente si no le gusta tienes que volver a hacerlo, entonces sí se me hizo más fácil...”

Entrevistado PRAU_E06	
P.A1.1.	“No tanto porque te decía que es lo que significaba cada campo entonces no fue muy complicado”
P.A2.1.	“Sí porque te decía para qué sirve y cómo debes enfocarte en tus objetivos y los campos te describían como hacer ese patrón y para qué sirve, entonces sí te da una idea”
P.A1.5.	“En lo personal como que nos dio el alcance del proyecto...para quién va dirigido y cómo los objetivos...es como una guía para que no nos salgamos del camino al momento de desarrollar”
P.A1.6	“A veces...como es nuevo para nosotros surgían dudas y teníamos que consultar ya sea la plantilla o de manera grupal, entonces esa parte como que sí nos quitaba un poco de tiempo...requiere un poco de tiempo, también para madurar el concepto...”
P.A1.7.	“...poner la foto de la persona como que está un poco de más, yo leyendo la descripción yo puedo saber quiénes para quién va dirigido...habían comentado igual que a lo mejor no es la foto del usuario o puede ser la foto entonces esa parte siento que está un poco de más...”
P.A1.2	“No, no mucho...nos daban una descripción de los objetivos de esas pruebas y cómo aplicarlas...hacer prototipos, para qué va a servir, hacer las planeaciones...estaba muy descrito ahí cómo se debe implementar...estaba claro cómo se tiene que hacer...”
P.A2.4.	“Sí, porque de cierta manera nos servía cómo una guía, si teníamos dudas pues consultábamos y veíamos para que sirve y cómo implementarlo”
P.A3.6	“Sí lo usaría porque con esta experiencia vimos que a veces programamos y hacemos algo pero cuando se hacen los casos de prueba pues vemos que no es como lo creímos...nos ayuda a que tengamos un programa que sea a la medida a lo que el cliente está pidiendo” “si...es como una referencia, siempre van a haber dudas entonces yo puedo acudir a eso (al producto patrón) para apoyarme”.
P.A1.4.	“Nos sirve como retroalimentación para la creación del juego...”
P.A1.6.	“Los diagramas...a veces está muy compactado...no se entienden muy bien las letras que están ahí...están muy pequeño”
P.A1.8.	“Los diagramas...que tengan cortes de cada actividad como fluye el proceso...que tenga cortes para que se entienda un poco mejor porque si está todo ahí como que confunde un poco”

Entrevistado PRAU_E07	
P.A1.1.	“No estuvo tan complicado, básicamente era seguir la plantilla que esta allá y hacer una breve descripción de la persona, en este caso al niño...para mí estuvo algo corto el contenido con el que describe el patrón...no fue complicado”
P.A2.1.	“...Para mí fue algo sencillo porque con esos datos pudimos observar a las personas cuál es su comportamiento en sí...”
P.A2.2.	“...para mí al principio...fue un poco de atraso mientras estábamos aprendiendo a aplicar el patrón porque no sabíamos exactamente cómo aplicarlo de inmediato estábamos aprendiendo a aplicarlo pues sentimos que llevó un poco de tiempo, ya después conforme avanzamos siento que ya nos empezamos a acostumbrar poco a poco pero tampoco fue tan continuo...”
P.A1.5.	“...facilitaba a manera de plantilla con seguir el proceso de que dato debemos poner y no estar atormentándonos o analizar ‘este dato va o este dato no va’...tenía un orden...estaba bien estructurado para que uno lo entienda...siento que fue más fácil aplicar esto. Otra cosa positiva siento que puede servir para poner información adicional o poner algo que no estuviera contemplado”
P.A1.2	“...para mí no fue complicado...”
P.A2.6.	“Yo considero que sí...teníamos como una información más a la mano no teníamos que estar buscando otros papeles u otro documento que realmente nos iba a servir o no...con esa información para mí era suficiente...”
P.A2.5.	“...fue un mínimo atraso para mí en el desarrollo...fue un ligero atraso...”
P.A3.6	“Yo pienso que sí porque es para mí algo rápido de hacer, un poco más a la mano...con información así escorada con apuntar las observaciones y con apuntar los pasos a seguir...no hay que escribir demasiado...es una forma de tabla que facilita mucho al momento de analizar lo que estamos escribiendo y ya tener una idea más visible a lo quiere el usuario...yo pienso que sí lo volvería a usar”
P.A1.4.	“...era tener la información lo más rápido...una vez que tenía lo que va a hacer el usuario era capturar la información de lo que observabas...podías escorar la información con todo lo que contenía ahí y con lo que sacamos...”

Entrevistado PRAU_E08	
P.A1.1.	“No, creo que por las instrucciones que usted nos había dado estaba claro”
P.A2.1.	“Si porque de cierta manera como que teníamos más pendiente hacia qué grupo de personas iba dirigido nuestro programa”
P.A2.2.	“...hacer la documentación...de ahí como que se agilizó...ya tenías pendiente hacia dónde va dirigida hacia qué audiencia...”
P.A1.3.	“Es algo que muchas personas no consideran...en cualquier proyecto prácticamente uno hace un trabajo pero no siempre tiene pendiente hacia qué audiencia va dirigido entonces eso es importante para cualquier proyecto...” “Lo del diagrama me parece que es positivo...muy bien explicado...”
P.A1.5.	“No...la verdad no...”
P.A2.4.	“Sí...más que nada por las pláticas que usted nos había dado...y los ejemplos que usted nos había mostrado en las presentaciones y el video...”
P.A2.6.	“...más bien me basé en el video (de ejemplo) un poco...”
P.A2.5.	“No, yo creo que no...es más sencillo que crear la persona...”
P.A3.6	“Claro...porque de esa forma sabes y consideras cosas que tu creías que estaban bien a lo mejor desde la perspectiva de otra persona no...te retroalimentas...”
P.A1.4.	“Los diagramas...te lo explica...”
P.A3.4.	“El product owner fue la primera vez que participaba en un desarrollo de software”

Anexo H. Proyecto real de Evaluación en Industria PREI

H.1. Encuestas grupo PREI

COMPLETAS (Encuestado 2)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

Arcon

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Completamente de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Completamente de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Conocer las actividades de cada una de las personas en el depto
- b) poder identificar las opciones del sistema a personalizar en cuanto a permisos
- c) identificar que aunque los usuarios tengan la misma categoría, tienen actividades diferentes

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Es un poco tardado
- b) se nos dificultó encontrar su utilidad, debido a que tenemos unas librerías para la definición de privilegios y permisos por usuario

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Utilizar cuestionarios en vez de entrevistas, ya que el encontrar el momento adecuado para entrevistar a los usuarios es un poco complicado

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Completamente de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Completamente de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Completamente en desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Ayuda a simplificar el funcionamiento de las pantallas
- b) permite solucionar errores, complicados de reproducir
- c) disminuye el tiempo perdido de programación, ya que permite ver los conflictos de funcionamiento de una forma temprana

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) en ocasiones el usuario desea hacer cambios que afectan los estándares del equipo de desarrollo

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Se omitió esta pregunta

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Muy de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios En Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario En Desacuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Completamente de acuerdo

COMPLETAS (Encuestado 12)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones ARCON

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Muy de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Completamente de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Permitted identificar los diferentes roles o categorías de las personas que iban a usar el sistema
- b) Permitted identificar con claridad las actividades de cada rol o persona

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Posiblemente el tiempo para realizar las entrevistas si hay diversos roles, categorías o personas que van a usar el sistema
- b) A fin de cuentas es una actividad e implica un coste: tiempo.
- c) A veces no se tiene a la mano a las diversas personas que van a participar

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Pues para minimizar los aspectos negativos generamos un cuestionario y se lo dejamos a las personas involucradas para que lo respondieran cuando tuvieran tiempo. No les perjudicamos en sus actividades y nos dieron una pronta respuesta. Así tampoco nos atrasamos.

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Completamente de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Muy de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Nos permitió evaluar qué tan intuitivo era el sistema para el usuario
- b) Nos permitió identificar, y mejorar, algunas formas de solicitar la información al usuario

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Pues no se si sea el aspecto negativo... pero pues implica tiempo y si fuéramos un equipo con un solo proyecto ágil no sería tan preocupante.

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

En realidad el patrón se me hace perfecto para minimizar cambios al momento de implementar sistema. El ver qué complica el uso del sistema o qué se puede mejorar, inclusive detectar errores ayuda grandemente

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Completamente de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios Completamente en Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario Completamente en Desacuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Completamente de acuerdo

COMPLETAS (Encuestado 13)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

SIJEA Agenda

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? De acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones De acuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales De acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles En desacuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Sencillo
- b) Practico
- c) Visual

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) No muy util en mi institucion
- b) Requiere tiempo la entrevista
- c) No todos los usuarios se sienten comodoss al ser entrevistados

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Que sea en español y con los terminos mas desgl

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Completamente de acuerdo
- La solución propuesta en el patron "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Completamente de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones De acuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Muy util para contemplar tareas despues
- b) Establece un orden al definir tareas
- c) Da una vision anticipada del proyecto al usuario final

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Requiere tiempo al preparar la prueba y el analisis
- b) Pueden surgir muchos cambios que no se hayan contemplado en el backlok y si son grandes cambios atrasa el proyecto

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

En español y con los terminos mas descriptivos

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Muy de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios En Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario Muy de acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Completamente de acuerdo

COMPLETAS (Encuestado 14)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

Siafa (Sistema del Fondo Auxiliar) y Siaca (Sistema de Adquisiciones)

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Completamente de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Completamente en desacuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Completamente de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Conocimiento mas amplia de los diferentes roles
- b) Se realizan adecuaciones durante el ciclo a los requerimientos que garantizan la aceptacion del producto final
- c) Mejor entendimiento entre lo técnico y lo operativo, es decir la negociación entre product owner y team es mas transparente

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) mayor tiempo al sprint
- b) menor alcance en la entrega de productos

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Realizar entrevistas y pruebas en los sitios de operación donde se va a utilizar el producto final

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Completamente de acuerdo
- La solución propuesta en el patron "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Completamente de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Completamente en desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Definición clara de roles y funciones para el proceso a automatizar
- b) mejor entendimiento entre el product owner y el team
- c) mayor garantía de aceptación del producto terminado

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Mayor gestión administrativa del proyecto
- b) resistencia de uso por parte de los programadores por inversión de diseños de prototipo
- c)

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Realizar dentro del patrón entrevistas y visitas al sitio donde se opera actualmente el proceso a automatizar

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Completamente de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios En Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario Completamente de acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Completamente de acuerdo

COMPLETAS (Encuestado 15)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

Siafa y Arcon

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" Muy de acuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Muy de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones De acuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Muy de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles De acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Perfil de roles mas estructurados
- b) Comprensión de las actividades de cada rol

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Demasiados detalles personales

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Se omitió esta pregunta

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Muy de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Muy de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Comprender la forma en que el usuario espera su interfaz
- b) Menos errores de diseño
- c) Agilización de los procesos a partir del diseño

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) No siempre se puede tener la infraestructura requerida

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Se omitió esta pregunta

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios De acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios En Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario De acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Muy de acuerdo

COMPLETAS (Encuestado 16)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

SIJEA_Agenda (Agenda Juzgado Adolescentes)

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Muy de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Completamente de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Se obtienen datos importantes
- b) Ayuda a definir los roles
- c) Oriente el diseño de la vista en el sistema

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Ninguno

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

pudiera ser el lenguaje

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Muy de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Muy de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Completamente en desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Ayuda a obtener un release del sistema
- b) Proporciona informacion trascendental para el desarrollo
- c) Ayuda al Product Owner a conocer el sistema

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Depene de la Disponibilidad del Product Owner y los Stakeholder

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Pudiera ser el lenguaje

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Muy de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios De acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario De acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Muy de acuerdo

COMPLETAS (Encuestado 17)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

SIGES, SIAFA

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? De acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones De acuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales De acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles De acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Permite identificar correctamente las historias de usuario
- b) Ayuda a establecer las prioridades de estas historias
- c) Permite definir los perfiles de usuario para el sistema

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Está en inglés

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Hacer una versión en español.

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Muy de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Muy de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Ayuda a definir correctamente las funcionalidades del sistema
- b) Las pruebas de usabilidad permiten identificar las funcionalidades faltantes
- c) En las pruebas de usabilidad se observan posibles mejoras en la interfaz del sistema

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Demasiadas pruebas de usabilidad

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Ninguna

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Muy de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios Completamente en Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario De acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Muy de acuerdo

COMPLETAS (Encuestado 18)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

SIGES

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Completamente de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Completamente de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Permite comprender mejor los roles de los usuarios
- b) Facilita la delimitación de tareas para cada rol

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Pérdida de tiempo cuando los roles son ampliamente conocidos

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Ninguna

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Completamente de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Completamente de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Muy de acuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Entender mejor lo que el usuario espera al interactuar con el sistema

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) La inversión de tiempo inicial

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Ninguna

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Completamente de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios En Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario En Desacuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Completamente de acuerdo

COMPLETAS (Encuestado 20)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

SIACA Almacén

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Muy de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Muy de acuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Muy de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles De acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) conocer al usuario
- b) definir el rol del usuario

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) tiempo para emplearlo

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Se omitió esta pregunta

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Muy de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Muy de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Muy de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) obtener la información que el usuario espera del proyecto
- b) evita programar algo que no será necesario

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) preparar la prueba

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Se omitió esta pregunta

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Muy de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios En Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario De acuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Muy de acuerdo

COMPLETAS (Encuestado 21)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

SIGES

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Completamente de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Completamente en desacuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Muy de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) me sirvió para conocer las habilidades del usuario
- b) me sirvió para saber como hacerle entender ciertos aspectos técnicos
- c) me sirvió para tener una relación más cercana con el usuario

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) Hacerle la entrevista para conocerlo, ya que en ocasiones el usuario no dispone de mucho tiempo
- b) la distancia entre los edificios y cierta restricción que se tiene con el teléfono

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Se omitió esta pregunta

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Completamente de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Completamente de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Completamente en desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Ayuda para evitar los retrasos en la programación de la vista del programa
- b) permite conocer como al usuario le gusta manejar los controles
- c) y con los criterios de aceptación se evitan los conflictos de que luego no lo solicite el usuario

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) el tiempo que disponen los usuarios
- b) cuando son más de uno se complica, ya que en ocasiones solo se tiene una pc

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Se omitió esta pregunta

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Completamente de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios Completamente en Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario Completamente en Desacuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Completamente de acuerdo

COMPLETAS (Encuestado 23)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

ARCON

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Muy de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Completamente en desacuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Completamente de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) permite identificar las actividades de cada persona
- b) permite identificar que tipo de persona es
- c) puede ser utilizado para la definicion de los niveles de seguridad en un sistema

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) ninguno

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

En la seccion de actividades y la de necesidades, podria colocarse a través de incisos

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Muy de acuerdo
- La solución propuesta en el patron "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Muy de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Completamente en desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Permite conocer las experiencias de los usuarios acerca de la facilidad de uso del sistema
- b) Permite identificar errores ocultos que no se habian visualizado en su momento
- c) Permite conocer a las impresiones del usuario al momento de presentar el prototipo o la aplicación

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Ninguno

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Se omitió esta pregunta

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Muy de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios En Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario En Desacuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Completamente de acuerdo

COMPLETAS (Encuestado 24)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

SIACA

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" En desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Muy de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones En desacuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Muy de acuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) permite identificar los roles de cada persona
- b) permite conocer cual es la actividad que realizan
- c) permite definir los niveles de seguridad

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) ninguno

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

En la seccion de actividades y la de necesidades, podria colocarse a través de incisos

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Muy de acuerdo
- La solución propuesta en el patron "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Completamente de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Completamente en desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Permite conocer las experiencias de los usuarios acerca de la facilidad de uso del sistema
- b) Permite identificar errores ocultos que no se habian visualizado en su momento
- c) Permite conocer a las impresiones del usuario al momento de presentar el prototipo o la aplicación

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Ninguno

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Se omitió esta pregunta

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Completamente de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios En Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario En Desacuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dió como resultado aplicaciones más fáciles de utilizar por los usuarios Completamente de acuerdo

COMPLETAS (Encuestado 25)

PÁGINA 1: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P1: Nombre del proyecto de software en el que utilizaste los patrones

SIJES

P2: Patrón "Agile persona"

- Fué complicado entender la estructura del patrón "Agile Persona" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Persona" facilitó la aplicación de la técnica "Persona" durante el proceso de desarrollo ágil? Completamente de acuerdo
- Utilizar el patrón "Agile persona" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Completamente de acuerdo
- Utilizar la solución propuesta en el patrón "Agile Persona" (campo solution) permitió definir de manera más exacta los perfiles de los usuarios finales Completamente en desacuerdo
- Utilizaré de nuevo el patrón "Agile persona" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P3: Lista los aspectos más positivos del patrón "Agile persona"

- a) Permite Identificar los roles que se desempeñan
- b) Permite establecer las estrategias para el diseño de la interfaz
- c) Permite conocer el tipo de usuario final que tendrá el sistema

P4: Lista los aspectos más negativos del patrón "Agile Persona"

- a) No encontré ninguno
- b) No encontré ninguno
- c) No encontré ninguno

P5: Sugerencias para mejorar la utilidad del patrón "Agile Persona"

Ninguna

PÁGINA 2: Facilidad / Dificultad de aplicar las Técnicas de usabilidad siguiendo el patrón

P6: Patrón "Agile Usability Testing"

- Fué complicado entender la estructura del patrón "Agile Usability Testing" Completamente en desacuerdo
- La información proporcionada en el documento del patrón "Agile Usability Testing" facilitó la aplicación de la técnica "Usability Testing" durante el proceso de desarrollo ágil Completamente de acuerdo
- La solución propuesta en el patrón "Agile usability testing" (campo solution) me facilita aplicar las pruebas de usabilidad en el momento adecuado durante el desarrollo ágil del software Completamente de acuerdo
- Utilizar el patrón "Agile usability testing" durante el desarrollo ágil del proyecto generó atraso en los tiempos programados de las iteraciones Completamente en desacuerdo
- Utilizaré de nuevo el patrón "Agile Usability Testing" durante el desarrollo de software con metodologías ágiles Completamente de acuerdo

P7: Lista los aspectos más positivos del patrón "Agile Usability Testing"

- a) Permite que el usuario conozca desde el principio la interfaz
- b) Permite identificar que tan útil es la interfaz diseñada
- c) Permite identificar aspectos que son importantes para el usuario al momento de usar la interfaz

P8: Lista los aspectos más negativos del patrón "Agile Usability Testing"

- a) Ninguno
- b) Ninguno
- c) Ninguno

P9: Sugerencias para mejorar la utilidad del patrón "Agile Usability Testing"

Ninguna

PÁGINA 3: Resultado de la aplicación de las técnicas de usabilidad dentro del proceso de desarrollo ágil

P10: Incorporando las técnicas de usabilidad en el proceso de desarrollo ágil...

- Las funcionalidades identificadas fueron más precisas en relación a las necesidades reales de los usuarios Completamente de acuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala definición de los perfiles de los usuarios Completamente en Desacuerdo
- Durante las iteraciones hubieron cambios en las funcionalidades desarrolladas debido a la mala interpretación de los requerimientos del usuario Completamente en Desacuerdo
- En base a tu experiencia previa en desarrollo de software consideras que al incorporar persona y usability testing durante el proceso ágil dio como resultado aplicaciones más fáciles de utilizar por los usuarios Completamente de acuerdo

H.2. Extractos de Entrevistas PREI

Entrevistado PREI_E01	
P.A1.1.	“No, la estructura fue sencilla. Ayuda mucho el diagrama de flujo”
P.A2.1.	“Sí, la aplicamos al principio. Vemos que es más factible obtener mejor resultado aplicando al principio de nuestros proyectos el patrón... en el proyecto Agenda sí lo aplicamos al principio y fue más sencillo implementarlo y fue más útil tener la información con el patrón...”
P.A2.2.	“No, no genera atraso...cuando tú te programas”
P.A2.3.	“Sí...básicamente para eso nos sirvió. Pudimos sacar por ejemplo con nuestro product owner el rol principal...vimos un rol más que no era necesario tenerlo físicamente porque tendría las mismas funcionalidades que tendría el stakeholder...”
P.A1.2	“No, el diagrama de flujo que tienes si está un poco complejo por todos los caminos que se puede seguir el patrón pero si pones atención si te ubicas en cada etapa del patrón, realmente no está tan difícil y como ya tienes el ejemplo...y haces el análisis de usabilidad...está bastante completo, no tuvimos que implementar nada adicional...es muy sencillo”
P.A2.4.	“Sí, totalmente. Te dice que es lo que se va a hacer...qué es lo que se debe hacer, cómo se planifica. Es más ágil...no estás buscando el término exacto o lo el que tu creas que debe ser, te enfocas a lo que dice el patrón, lo sigues y sacas mejor resultado si lo aplicas por tu cuenta”
P.A2.6.	Sí...de hecho quisimos ver primero como lo aplicaríamos con ese patrón y en base a eso hicimos las tareas...resulta que aplicamos el patrón en diferentes momentos en diferentes sprints...en el primer sprint fue al inicio como está el diagrama en la primera parte...en el segundo sprint fue al final...
P.A3.2.	No... el cuestionario de ahí salió todo lo que necesitábamos...realmente se llenó como se necesitaba y en base a eso se desarrolló...

Entrevistado PREI_E02	
P.A1.1.	“...el ejemplo está muy bien, muy entendible...se me hizo bastante claro...y los incisos...ponerles alguna numeración si es en el orden en el que van o es otro orden...más ordenado...sería bueno ponerle numeraciones...del diagrama de flujo...hacer más claro los términos...”
P.A2.2.	“Sí es cierto que nos lleva un tiempo...hacer una tarea específica, si separas el tiempo como tal no creo que sea realmente un atraso...al contrario siento que es una inversión...si inviertes en definir bien a tu usuario, su necesidad, sus características como q te involucras más con el y el contigo y a fin de cuentas es un beneficio para el proyecto...como nos pasó con Cinthia el hecho de intimar más con ella, de llevarnos más con ella nos hizo entender más su tarea, su proceso...creo que no es un retraso, considero que si lo haces bien y lo consideras bien en tus tareas no es un atraso...”
P.A1.3.	“Se me hace una herramienta que te acerca al usuario...para intimar con el usuario...entenderlo y conocerlo mejor...”
P.A1.2	“No...de hecho es el que más usamos...el ejemplo de la template...nos ayudó mucho...cuando hicimos nuestra primera usabilidad fue muy útil...”
P.A2.4.	“Si...de hecho completándolo con la clase que tú nos diste...nos ayudó bastante...fue más usable...”
P.A2.5.	“No...con certeza te puedo decir que no. De hecho hubiera sido un atraso no hacerlo...”
P.A3.6	“Si...sin duda...nos lleva de la mano...”
P.A1.6.	“Es un tiempo...porque si lleva un tiempo hacer la prueba...aplicarla también es un tiempo...siento que está lleva más tiempo que la otra (persona)...en cambio aquí tienes que preparar los casos adivinar que va a pensar y si nos dice esto que le vamos a decir o no le vamos a decir, que tal si se ofende...luego hacer el feedback...como definir, saber con certeza que esto es un bug o el otro tipo...te va a llevar tiempo hacer esa definición...es un tiempo más...”
P.A3.1.	“Si...definitivamente sí...”

Entrevistado PREI E08	
P.A1.1.	“No fue complicado...desde mi punto de vista es entendible”
P.A2.1.	“Sí, si me ayudó...te da ciertos lineamientos, te da una plantilla, te da un ejemplo que te permite hacerlo”
P.A2.2.	“El patrón persona normalmente lo manejamos en la parte de las definiciones de las tareas, no lo trabajamos tanto dentro de la iteración...antes del sprint tenemos las entrevistas y ahí yo maneje la parte del patrón persona...en el momento de continuar con el desarrollo apareció un usuario, este usuario se analizó que tanto impactaba dentro de la aplicación...se determinó que no tenía mucha influencia sobre el sistema y se consideró como parte de los ya definidos...”
P.A1.3.	“Una de las partes más positivas es el ejemplo...si no te acuerdas...vas directamente al ejemplo...si estás leyendo y se te complica un poco el hecho de entender a qué se está refiriendo cada cosa ya te vas al ejemplo y se queda claro lo que está tratando de decir el documento. La parte del diagrama me gustó mucho...te permite ver en qué fase lo tienes que hacer y de una manera visual te va mostrando en qué momento debes aplicar ese patrón...en lo particular si yo agarro el documento para un uso me iría inmediatamente al diagrama...para ver en qué momento se va a hacer esa actividad...y me iría al ejemplo para recordar cómo tengo que hacer el patrón ya si me quedan dudas o algo empezaría a subir en el documento para ver las definiciones de cada cosa...”
P.A2.6.	“Sí, si me sirvió...el diagrama si me ayudó para definir en qué momento se debería aplicar...”
P.A1.4.	“El diagrama...me gustaron los dos diagramas...es un documento chiquito, no es largo...te pone de manera concisa todo lo que tienes que considerar...como un pequeño manual que te permite saber específicamente qué es lo que tienes que hacer...es como un pequeño manualito para utilizar...”
P.A3.4.	“...el hecho de formalizarlo el hecho de hacer que los usuarios primero vean el diseño, los prototipos y se empiecen a involucrar más en el sistema mejora la experiencia con el usuario...”

Entrevistado PREI E10	
P.A1.1.	“No”
P.A2.1.	“Sí”
P.A2.2.	“No porque nosotros hicimos una pequeña variante...nosotros hicimos un cuestionario y se los entregamos entonces en el momento que tuvieran tiempo lo responderían...se los entregamos no se, un martes y al día siguiente ya teníamos las respuestas...ni se atrasaron ellos ni nos atraso a nosotros...de hecho son las mismas que tenía la plantilla solo que las pasamos a español y las hicimos como cuestionario...les dimos formato de cuestionario...”
P.A2.3.	“Por supuesto...sí...”
P.A3.5.	“Sí...de hecho no solo para metodologías ágiles también para metodologías tradicionales...”
P.A1.7.	“...a nosotros sí se nos hizo buena idea lo de aplicarlo como cuestionario cuando son muchos usuarios cuando tienes un solo usuario del sistema no sé, un sistema de uno o dos usuarios no hay mayor problema que tu lo entrevistes y te explayas pero cuando tienes muchos usuarios sobre todo en el caso de algunos sistemas que sus usuarios están en otro edificio es complicado poder coincidir con ellos ya sea mandándolos (los cuestionarios) por correo o por estafeta...eso no te quitaría tiempo y tampoco les quitarías tiempo a ellos...y la aplicación temprana de esa técnica sería lo ideal para que tu de entrada empieces a ver las actividades que tiene cada quien y empezar a definir las...”
P.A1.4.	“...es una guía de cómo podíamos aplicar la técnica...si te da una pauta de qué hacer, cómo informarle al usuario qué es lo que se debe de hacer y ciertos aspectos que a veces nosotros no consideramos, como indicarle al usuario de que en realidad no se le está evaluando...que se dé cuenta que en realidad el nos está examinando a nosotros...nos dio la pauta para que el usuario se sintiera cómodo...cómo aplicarlo...una guía...”
P.A3.1.	“Sí...por supuesto...”
P.A3.4.	“Por supuesto...la aplicación de este patrón si te ayuda bastante...ambos patrones...creo que ambos son muy buenos para definir correctamente funciones, actividades que hace cada usuario...hablando de sistemas nuevos o de funciones nuevas...hemos aplicado el patrón persona en dos sistemas en el de UPI y en el de ARCON...en el de UPI solo teníamos por el momento a la psicóloga y al juzgado que sus actividades ya sabíamos que era lo que iban a hacer...sí tiene muchas funciones pero no era parte del sistema...qué actividades iba a realizar la psicóloga...claramente percibí qué es lo que tenía que hacer el juzgado...en el caso de ARCON nos permitió ver que teníamos a dos o tres administradores y los tres tenían completamente diferentes actividades a pesar de ser la misma categoría...”

H.3. Cuestionario definido por el equipo del proyecto PREI01 utilizado como artefacto de apoyo para la definición de persona

Entrevista a Personas

1. ¿Cuál es tu puesto o nombre del Puesto?

2. ¿A quién le Reportas o quien es tu Jefe?

3. ¿Cuáles son las Actividades de tu Puesto, de manera general?

4. Describe brevemente las características o virtudes de tu trabajo

5. Menciona algunos de tus pasatiempos

6. ¿Tienes Alguna sugerencia para el sistema?
Si, Describe la: