



Universidad Politécnica de
Madrid

**Escuela Técnica de Ingenieros
Informáticos**



TRABAJO DE FIN DE GRADO

Desarrollo de una Plataforma
Integral y Cuestionarios
Interactivos para Evaluación de
Habilidades Interculturales en
Estudiantes Universitarios
Europeos del Siglo XXI.



Autora: Grettell Umpierrez Sardiñas

Tutora: Jelena Bobkina

Madrid, Junio 2024

Trabajo de Fin de Grado

Grado de Ingeniería Informática

Título: *Desarrollo de una Plataforma Integral y Cuestionarios Interactivos para Evaluación de Habilidades Interculturales en Estudiantes Universitarios Europeos del Siglo XXI.*

Junio 2024

Autora: *Grettell Umpierrez Sardiñas*

Tutora:

Jelena Bobkina

Departamento de Lingüística Aplicada a la Ciencia y la Tecnología

ETSI Informáticos

Universidad Politécnica de Madrid

RESUMEN

En el marco de la asignatura "English for Professional and Academic Communication (EPAC)", se plantea una investigación enfocada en explorar las habilidades comunicativas interculturales de los estudiantes universitarios europeos del siglo XXI a través del desarrollo de una plataforma web: DeStance, que permita tanto la administración y almacenamiento de subcorpora escritos en diferentes lenguas como la implementación y gestión de cuestionarios interactivos, adaptados para ser aplicados antes y después de un Curso Online Masivo y Abierto (MOOC), buscando medir dichas capacidades interculturales a distintos niveles.

DeStance ha sido diseñada e implementada siguiendo el patrón de pila tradicional de desarrollo web de tres niveles: MEAN, incluido el nivel de visualización, un servidor de Angular que actúa como cliente web del nivel de aplicación, constituido por un servidor de Express.js y Node.js que se encarga de interactuar con el nivel de base de datos, el cual consiste en un servidor de MongoDB. Además, tanto front-end como back-end, serán alojados en Firebase, mientras que para la base de datos se utilizarán los servidores designados en MongoDB Atlas.

El proyecto consta de cuatro fases que serán expuestas a través de los ocho capítulos que presenta este documento, de la siguiente manera:

- Fase de Requisitos:
 - **Capítulo 1:** Introducción.
 - Se expone la motivación que condujo al planteamiento de DeStance y qué se pretende conseguir.
 - **Capítulo 2:** Estado del Arte.
 - Se presentan las tecnologías y plataformas similares que existen actualmente que sirven de línea base para DeStance.
 - **Capítulo 3:** Especificación de Requisitos.
 - Se plantean las funcionalidades a desarrollar y con qué restricciones.

- Fase de Diseño:
 - **Capítulo 4:** Elección de Tecnologías Web.
 - Se explican las tecnologías escogidas para el desarrollo de la aplicación web y el razonamiento detrás de su elección, dada la especificación de requisitos del proyecto.
 - **Capítulo 5:** Diseño
 - Se realiza una descripción detallada del diseño de la aplicación partiendo de los casos de uso.
- Fase de Implementación:
 - **Capítulo 6:** Desarrollo de DeStance.
 - Se presenta la planificación inicial de DeStance, su progreso actual, el resultado final con capturas de la aplicación, así como la estructura final del código.
- Fase de Retrospectiva:
 - **Capítulo 7:** Resultados y Conclusiones.
 - Se plantean los resultados y conclusiones personales alcanzadas tras la culminación de DeStance.
 - **Capítulo 8:** Análisis de Impacto.
 - Se expone un análisis del impacto potencial del proyecto en relación a los objetivos de la Agenda 2030.

A día 3 de junio de 2024, al menos en las tareas objetivo de este Trabajo de Fin de Grado, DeStance se encuentra en un estado casi final de la desarrollo de cara a empezar su uso. En su estado actual la aplicación implementa todos sus requisitos funcionales y no funcionales, sin embargo le siguen faltando datos reales en las secciones de cuestionarios, preguntas frecuentes y conceptos, además del diseño de la página de bienvenida y del logo de la aplicación que actualmente hace referencia a su antiguo nombre: RACISMMAFF.

Palabras clave: Aplicación web, habilidades interculturales, cuestionarios interactivos, racismo e inmigración.

ABSTRACT

Within the framework of the subject "English for Professional and Academic Communication (EPAC)", a research project has been formulated. Its focus lay on exploring the intercultural communication skills of European university students of the 21st century through the development of a web platform: DeStance, which allows both the administration and storage of subcorpora written in different languages and the implementation and management of interactive questionnaires, adapted to be applied before and after a Massive Open Online Course (MOOC), seeking to measure said intercultural capabilities at different levels.

DeStance has been designed and implemented following the traditional three-tier web development stack pattern: MEAN, including the display tier, an Angular server acting as the application tier web client, consisting of an Express.js and Node.js server that is responsible for interacting with the database level, which consists of a MongoDB server. In addition, both front-end and back-end will be hosted in Firebase, while the servers designated in MongoDB Atlas will be used for the database.

The project consists of four phases that will be explained through the eight chapters presented in this document, as follows:

- Requirements Phase:
 - **Chapter 1:** Introduction.
 - The motivation that led to DeStance's approach and what it aims to achieve is explained.
 - **Chapter 2:** State of the Art.
 - Similar technologies and platforms that currently exist that serve as a baseline for DeStance are presented.
 - **Chapter 3:** Requirements Specification.
 - The functionalities to be developed and with what restrictions are proposed.
- Design Phase:
 - **Chapter 4:** Choice of Web Technologies.

- The technologies chosen for the development of the web application and the reasoning behind their choice are explained, given the project requirements specification.
- **Chapter 5: Design**
 - A detailed description of the application design is made based on the use cases.
- Implementation Phase:
 - **Chapter 6: Development of DeStance.**
 - The initial planning of DeStance, its current progress, the final result with screenshots of the application, as well as the final structure of the code are presented.
- Retrospective Phase:
 - **Chapter 7: Results and Conclusions.**
 - The results and personal conclusions reached after the completion of DeStance are presented.
 - **Chapter 8: Impact Analysis.**
 - An analysis of the potential impact of the project in relation to the objectives of the 2030 Agenda is proposed.

As of June 3, 2024, at least in the objective tasks of this Final Degree Project, DeStance is in an almost final state of development in order to begin its use. In its current state, the application implements all its functional and non-functional requirements, however it is still missing real data in the questionnaires, frequently asked questions and concepts sections, in addition to the welcome page design and the application logo that currently reference its former name: RACISMMAFF.

Keywords: Web application, intercultural skills, interactive questionnaires, racism and immigration.

TABLA DE CONTENIDOS

1. Introducción.....	11
1.1. Motivación del proyecto.....	11
1.2. Objetivos del proyecto.....	17
2. Estado del Arte.....	19
2.1. Google Forms.....	19
Figura 1: Gestión de Cuestionarios en la plataforma Google Forms.....	20
Figura 2: Interacción con cuestionarios en la plataforma Google Forms..	20
Figura 3: Dashboard con las respuestas a un cuestionario en la	
plataforma Google Forms.....	21
2.2. Moodle.....	21
Figura 4: Plataforma Moodle como guía para la interfaz de usuario de	
DeStance.....	22
3. Especificación de Requisitos.....	23
3.1. Requisitos Funcionales.....	23
3.1.1. Consulta de funcionalidades de DeStance.....	23
3.1.2. Comunicación bidireccional con los Usuarios.....	24
3.3. Requisitos de Interfaces Externos.....	26
3.2.1. Interfaces de Usuario.....	26
3.2.3. Interfaces de Software.....	26
3.2.3. Interfaces de Comunicación.....	26
3.3. Requisitos de Desarrollo.....	26
3.4. Requisitos Tecnológicos.....	27
3.4.1. Front-end.....	27
3.4.2. Back-end.....	27
3.4.3. Base de Datos.....	27
3.5. Requisitos No Funcionales.....	27
3.5.1. Disponibilidad y Rendimiento.....	27
3.5.2. Mantenibilidad.....	28
3.5.3. Accesibilidad.....	28
3.5.4. Seguridad.....	28
3.5.5. Compatibilidad.....	28
3.5.6. Idioma.....	28
4. Elección de Tecnologías Web.....	29
4.1. Tecnologías para aplicaciones web full-stack.....	29
4.1.4. MERN Stack:.....	29
4.1.2. LAMP Stack:.....	29
4.1.3. Django Stack.....	30
4.1.4. MEAN Stack:.....	30

4.2. Tecnología escogida.....	30
Figura 5: Encuesta de Stack Overflow a sus usuarios sobre frameworks y tecnologías web más usadas por estos [20].....	32
Figura 6: Encuesta de Stack Overflow a sus usuarios sobre bases de datos más usadas por estos [20].....	33
4.3. MEAN en detalle.....	35
Figura 7: Componentes pila MEAN y su interacción.....	35
4.4. Hospedaje (hosting) de DeStance.....	38
Figura 8: Encuesta de Stack Overflow a sus usuarios sobre plataformas cloud más usadas por estos [20].....	39
4.5. Resumen de la sección.....	40
5. Diseño.....	41
5.1. Actores.....	41
5.2. Casos de Uso.....	41
Figura 9: Diagrama de Casos de Uso de DeStance.....	42
5.3. Arquitectura.....	50
5.3.1. Cliente/Servidor.....	50
Figura 10: Arquitectura cliente/servidor aplicada a DeStance.....	52
5.3.2. Base de Datos.....	52
Figura 11: Diagrama Entidad/Relación de DeStance.....	55
Figura 12.1: Ejemplo datos de prueba de entidad User (colección users). 57	
Figura 12.2: Ejemplo datos reales de entidad UserAuth (colección userauths).....	57
Figura 12.3: Ejemplo datos de prueba de entidad Questionnaire (colección questionnaires).....	58
Figura 12.4: Ejemplo datos de prueba de entidad UserQuestionnaire (colección userquestionnaires).....	59
Figura 12.5.1: Ejemplo datos de prueba de entidad Dashboard (colección dashboards).....	62
Figura 12.5.2: Ejemplo datos de prueba de entidad Dashboard (colección dashboards).....	63
Figura 12.5.3: Ejemplo datos de prueba de entidad Dashboard (colección dashboards).....	63
Figura 12.6: Ejemplo datos reales de entidad ConceptPool (colección conceptpools).....	64
Figura 12.7: Ejemplo datos de prueba de entidad AboutUs (colección aboutus).....	64
5.4. Diseño del front-end.....	64
5.4.1. Componentes del front-end.....	64
Figura 13.1: Diseño de los componentes del front-end de DeStance....	66
Figura 13.2: Diseño de los componentes del front-end de DeStance....	66
Figura 13.3: Diseño de los componentes del front-end de DeStance....	67
Figura 13.4: Diseño de los componentes del front-end de DeStance....	67

5.4.2. Routing del front-end.....	67
Tabla 1: Rutas front-end de DeStance.....	69
Figura 14: Mapa de navegación del front-end de DeStance.....	69
5.5. Diseño del back-end.....	72
5.4.1. Estructura General.....	72
Figura 15: Diseño de los componentes del back-end de DeStance.....	72
5.5.2. Descripción de Carpetas y Archivos.....	72
Figura 16: Estructura de carpetas en el back-end de DeStance.....	73
5.5.3. Descripción de las API.....	75
Tabla 2: Recursos del back-end de DeStance.....	76
Tabla 3: Endpoints del back-end de DeStance.....	84
6. Desarrollo de DeStance.....	85
6.1. Planificación Inicial.....	85
Tabla 4: Distribución general de horas de DeStance.....	85
Tabla 5: Distribución tareas por horas de DeStance.....	86
Figura 17: Diagrama de Gantt planificado para DeStance.....	87
6.2. Progreso Actual.....	88
6.3. Resultado Final.....	90
Pantalla Inicial.....	90
Figura 18: Sección temporal de bienvenida de DeStance.....	91
I. Registrarse.....	92
Figura 19.1: Registro - Menú de Login/Sign-up.....	92
Figura 19.2: Registro - Iniciar-Sesión - El usuario no tiene cuenta todavía.....	92
Figura 19.3: Registro DeStance - Acuerdo de consentimiento del participante.....	93
Figura 19.4: Registro DeStance. Formulario de registro vacío.....	94
Figura 19.5: Registro DeStance. Formulario de registro con datos de prueba.....	94
Figura 19.6: Registro DeStance. Cuadro de diálogo. Confirmación de datos introducidos.....	95
Figura 19.7: Registro DeStance. Cuadro de diálogo. Bienvenida a DeStance.....	95
II. Iniciar Sesión.....	95
III. Recuperación de contraseña.....	95
Figura 20: Inicio Sesión DeStance. Formulario de inicio de sesión con datos de prueba.....	96
Figura 21.1: Recuperación de contraseña DeStance - Email de recuperación.....	96
Figura 21.2: Recuperación de contraseña DeStance. Confirmación email de recuperación.....	97
Figura 21.3: Recuperación de contraseña DeStance. Formulario de recuperación de contraseña.....	97
Figura 21.4: Recuperación de contraseña DeStance -Confirmación	

cambio de contraseña.....	98
IV. Cerrar Sesión.....	98
Figura 22: Cerrar sesión.....	98
V. Visualizar Introducción a DeStance.....	98
Figura 23: Botón de ocultar/mostrar menú de navegación lateral izquierdo.....	99
VI. Visualizar Conceptos.....	99
Figura 24.1: Concepts-pool. Conceptos no desplegados.....	99
Figura 24.2: Concepts-pool. Concepto de Empathy desplegado.....	100
VII. Visualizar Preguntas Frecuentes.....	100
Figura 25: About DeStance. Preguntas no desplegadas.....	101
VIII. Ver perfil.....	101
Figura 26: Account Details.....	102
IX. Actualizar perfil.....	102
Figura 27.1: Update Account.....	103
Figura 27.2: Cambiar imagen de avatar.....	103
Figura 27.3: Confirmación de datos introducidos. A la izquierda solo Imagen Avatar y a la derecha datos personales o de autenticación....	104
Figura 27.4: Confirmación de modificación.....	104
X. Visualizar preguntas/respuestas Cuestionarios.....	104
Figura 28.1: Pre-cuestionarios. Preguntas.....	105
Figura 28.2: Pre-cuestionarios. Pre-Cuestionario específico.....	105
Figura 28.3: Pre-cuestionarios. Pre-Cuestionario respondido anteriormente.....	106
Figura 28.4: Pre-cuestionarios. Respuestas dadas anteriormente por pregunta.....	106
Figura 28.5: Pre-cuestionarios. Resultado obtenido.....	107
XI. Responder Cuestionarios.....	107
Figura 29.1: Responder preguntas de cuestionarios.....	107
Figura 29.2: Responder TODAS las preguntas del cuestionario.....	108
Figura 29.3: Confirmación de respuestas introducidas.....	108
Figura 29.4: Confirmación de respuestas guardadas.....	108
XV. Visualizar Dashboard Cuestionarios.....	109
Figura 30.1: Pantalla inicial con opciones de administrador habilitadas.	109
Figura 30.2: Dashboard Post-Cuestionarios. Selección de cuestionarios.	110
Figura 30.3: Dashboard de Post-Cuestionarios. Selección de uno de los cuestionarios.....	110
Figura 30.4: Dashboard de Post-Cuestionarios. Selección de diferentes tipos de gráficos por preguntas.....	110
XVI. Gestionar Cuestionarios.....	111
Figura 31.1: Gestión de Pre-Cuestionarios. Menú de Selección.....	111
Figura 31.2: Gestión de Pre-Cuestionarios. Nuevo cuestionario.....	112

Figura 31.3: Nuevo Cuestionario. Preguntas y resultados.....	112
Figura 31.4: Nuevo Cuestionario. Confirmación de datos introducidos y de creación con éxito.....	113
Figura 31.5: Edición Cuestionario. Confirmación de datos introducidos.	114
Figura 31.6: Edición Cuestionario. Confirmación de modificación de cuestionario.....	114
Figura 31.7: Eliminación Cuestionario. Confirmación de eliminación de cuestionario.....	114
6.4. Estructura Final del Código.....	115
6.4.1. Estructura Frontend.....	115
Figura 32.1: Jerarquía inicial de DeStance.....	115
Figura 32.2: Jerarquía directorio src.....	116
Figura 32.3: Jerarquía directorio app.....	116
Figura 32.4: Componentes en el directorio app. About-Us, Authentication, Login, Sign-Up, Reset-Password, Concepts-Pool, Header, Not-Found.....	117
Figura 32.5: Componentes en directorio app. Dashboard de cuestionarios y sus subcomponentes. Cuestionarios y sus subcomponentes.....	118
Figura 32.6: Componentes en directorio app. Gestión de Cuestionarios y sus subcomponentes. Componentes, servicios y modelos que se utilizan en distintas partes de la aplicación. Sidebar.....	119
Figura 32.7: Componentes en directorio app. User y subcomponentes. Welcome.....	119
6.4.2. Estructura Backend.....	119
7. Resultados y Conclusiones.....	120
8. Análisis de Impacto.....	124
9. Líneas de Trabajo Futuras.....	127
10. Referencias Bibliográficas.....	129

1. INTRODUCCIÓN

1.1. Motivación del proyecto

En un mundo cada vez más globalizado, la movilidad humana se ha convertido en una característica distintiva de la era contemporánea. Esta, sin embargo, no está exenta de desafíos, especialmente en el contexto europeo, donde la diversidad cultural se ha convertido en una realidad innegable, dando lugar a la emergencia de ideologías xenófobas y prácticas de radicalización.

Las estructuras lingüísticas y las prácticas comunicativas con frecuencia transmiten estas ideologías, lo que hace necesario desarrollar metodologías de investigación lingüística que permitan identificar y abordar tales prácticas, señalándolas en el discurso de poderes institucionales, políticos y mediáticos, y abordándolas en la educación para enfrentarlas a través de metodologías pedagógicas afectivas destinadas a potenciar el pensamiento crítico y las capacidades interculturales [\[1\]](#).

En este contexto y en el marco de la asignatura "English for Professional and Academic Communication (EPAC)", se plantea una investigación enfocada en explorar las habilidades comunicativas interculturales de los estudiantes universitarios europeos del siglo XXI a través del desarrollo de una plataforma web: DeStance. Esta complementaría al proyecto de investigación RACISMMAFF [\[1\]](#), el cual pertenece a un grupo de más veinte investigadores y colaboradores de la Universidad Complutense de Madrid y la Universidad Politécnica de Madrid.

Tanto RACISMMAFF como DeStance, en su defecto, están financiados por el Ministerio de Ciencia e Innovación del Gobierno de España (MCIN), así como por el Fondo Europeo de Desarrollo Regional (FEDER). Este apoyo financiero es crucial para el desarrollo y ejecución de ambos proyectos, permitiendo que se disponga de los recursos necesarios para llevar a cabo investigaciones, desarrollar herramientas educativas y crear plataformas tecnológicas avanzadas. Estos proyectos no solo representan esfuerzos individuales, sino que forman parte de una continuidad en la línea de investigación y desarrollo

iniciada por proyectos previos: EUROEVIDMOD [\[2\]](#), EVIDISPRAG [\[3\]](#) y STANCEDISC [\[4\]](#).

The expression of Evidentiality and Modality in English and other European Languages: Cross-linguistic perspectives (EUROEVIDMOD) tiene como objetivo establecer un análisis unificado y crear una base de datos de marcadores evidenciales y modales en inglés en comparación con el español y otros idiomas en España, incluyendo el vasco, el catalán y el gallego. Este proyecto también extiende su análisis a varios otros idiomas europeos, tales como el holandés, el francés, el alemán, el griego moderno, el italiano, el lituano y el polaco [\[2\]](#).

El objetivo principal de EUROEVIDMOD es entender cómo se expresan la evidencialidad y la modalidad en diferentes lenguas, y cómo estos elementos lingüísticos varían entre ellas. La evidencialidad se refiere a los medios por los cuales los hablantes indican la fuente de su información, mientras que la modalidad trata sobre la actitud del hablante hacia la proposición expresada, como la certeza, la obligación o la posibilidad. Para lograrlo ha desarrollado una metodología rigurosa para identificar y categorizar los marcadores evidenciales y modales en cada idioma estudiado [\[2\]](#).

Lo anterior incluye tanto la recopilación de datos a través de corpus lingüísticos extensos como el análisis detallado de textos en contextos específicos. Los investigadores han trabajado en colaboración con expertos en cada una de las lenguas incluidas en el estudio, garantizando que los análisis sean precisos y contextualmente relevantes. Además de establecer una base de datos exhaustiva, EUROEVIDMOD también busca proporcionar una perspectiva comparativa que permita identificar patrones y diferencias entre los idiomas. Este enfoque comparativo es crucial para entender las particularidades de cada lengua y cómo los hablantes de diferentes culturas utilizan los recursos lingüísticos para expresar evidencialidad y modalidad [\[2\]](#).

Los resultados de EUROEVIDMOD no solo contribuyen al campo de la lingüística teórica y aplicada, sino que también tienen aplicaciones prácticas en áreas como la traducción, la enseñanza de idiomas y la inteligencia

artificial. Por ejemplo, una comprensión más profunda de cómo se expresan la evidencialidad y la modalidad en diferentes lenguas puede mejorar la precisión de los sistemas de traducción automática y proporcionar herramientas más efectivas para la enseñanza de segundas lenguas.

Por otro lado, como otro estudio que precede al proyecto en cuestión se encuentra: *Epistemic and non-epistemic modals: The key to interpreting the spirit of counter-terrorism United Nations Security Council resolutions* (EVIDISPRAG). Este estudio tiene como objetivo revelar la postura del Consejo de Seguridad hacia el terrorismo en todas sus formas y manifestaciones mediante la exploración de los patrones de distribución, frecuencia y funciones de nueve marcadores modales centrales en las cuarenta y siete resoluciones contra el terrorismo que se han publicado hasta la fecha [\[3\]](#).

Los modales epistemológicos expresan grados de certeza, posibilidad y conocimiento del emisor, mientras que los modales no epistemológicos abarcan aspectos como la obligación, la permisibilidad y la capacidad. Estos marcadores son cruciales para entender las intenciones y prioridades subyacentes en los textos de las resoluciones. Para alcanzar su objetivo, EVIDISPRAG utiliza un enfoque metodológico detallado y riguroso, que incluye el análisis de corpus lingüísticos específicos de las resoluciones del Consejo de Seguridad [\[3\]](#).

Este análisis permite identificar la frecuencia y los patrones de uso de los marcadores modales seleccionados, así como sus funciones en diferentes contextos. La investigación se basa en un examen minucioso de las resoluciones, buscando no solo la presencia de estos marcadores, sino también cómo contribuyen a la construcción del significado y la postura del Consejo de Seguridad [\[3\]](#).

Los resultados de EVIDISPRAG tienen implicaciones importantes para la interpretación de las resoluciones contra el terrorismo de las Naciones Unidas. Al comprender mejor los matices lingüísticos de estas resoluciones, se puede obtener una visión más clara de la postura del Consejo de Seguridad sobre el terrorismo, así como de las estrategias y enfoques que este organismo

internacional considera más efectivos. Esta comprensión puede, a su vez, informar la formulación de políticas y la implementación de medidas contra el terrorismo a nivel global.

Además, EVIDISPRAG contribuye al campo de la lingüística aplicada y la pragmática, demostrando cómo el análisis detallado de los modales puede ofrecer insights valiosos sobre la comunicación institucional y la formulación de políticas. Los hallazgos del estudio pueden aplicarse a otros contextos y documentos institucionales, ampliando su relevancia más allá del ámbito específico de las resoluciones contra el terrorismo.

Por último, *Posicionamiento y Subjetividad en el Discurso: Hacia un modelo integrado de análisis de la epistemicidad, efectividad, evaluación e inter/subjetividad desde la perspectiva del Análisis Crítico*, es un proyecto de investigación que tiene como objetivo principal el desarrollo de un marco integrado para el análisis de la postura y la inter/subjetividad en el discurso, desde una perspectiva analítica crítica de este. Este proyecto se propone abordar una serie de aspectos fundamentales del discurso que son cruciales para entender cómo los individuos y grupos expresan sus creencias, actitudes y evaluaciones a través del lenguaje [\[4\]](#).

El concepto de epistemicidad se refiere a cómo los hablantes expresan su conocimiento, creencias y grado de certeza sobre la información que comunican. En el análisis crítico del discurso, entender la epistemicidad es esencial para desentrañar cómo se construyen y comunican las verdades y certezas en diferentes contextos comunicativos. Este proyecto pretende desarrollar herramientas y metodologías para analizar cómo se manifiesta la epistemicidad en diversos tipos de discurso, y cómo esta influye en la persuasión y en la formación de opiniones [\[4\]](#).

La efectividad, por otro lado, se refiere a la capacidad del discurso para lograr ciertos objetivos comunicativos, como persuadir, informar, o motivar a la acción. Este aspecto del proyecto se centra en identificar los recursos lingüísticos y retóricos que contribuyen a la efectividad del discurso, así como en analizar su impacto en los destinatarios. A través de este análisis, el

proyecto busca ofrecer una comprensión más profunda de las estrategias discursivas que hacen que ciertos mensajes sean más influyentes y persuasivos [\[4\]](#).

La evaluación en el discurso implica la forma en que los hablantes expresan sus juicios de valor, opiniones y actitudes hacia los temas de los que hablan. Este componente del proyecto investiga cómo se construyen las evaluaciones en el lenguaje y cómo estas reflejan las ideologías y posiciones de los hablantes. El análisis de la evaluación es crucial para entender los mecanismos mediante los cuales se refuerzan o desafían las normas sociales y culturales a través del discurso [\[4\]](#).

La inter/subjetividad se refiere a cómo los hablantes se posicionan a sí mismos y a otros dentro del discurso, reflejando relaciones de poder, solidaridad, conflicto, y cooperación. Este aspecto del proyecto examina cómo los hablantes negocian significados y construyen identidades colectivas e individuales en la interacción discursiva. Desde la perspectiva del análisis crítico del discurso, entender la inter/subjetividad es clave para desentrañar las dinámicas sociales y las estructuras de poder que se manifiestan en el lenguaje [\[4\]](#).

En conjunto, este proyecto busca desarrollar un modelo integrado que combine estos diferentes aspectos del análisis del discurso en un marco coherente y sistemático. Al hacerlo, pretende proporcionar herramientas analíticas que sean útiles tanto para la investigación académica como para aplicaciones prácticas en áreas como la educación, la comunicación política, y la mediación intercultural. Desde una perspectiva de análisis crítico, el proyecto también aspira a contribuir a la comprensión y transformación de las prácticas discursivas que perpetúan la desigualdad y la injusticia social [\[4\]](#).

RACISMMAFF, por su parte, busca sumar sus resultados al conocimiento existente sobre las formas en que los usuarios del lenguaje se posicionan en el discurso sobre inmigración y racismo, contribuyendo significativamente a la literatura académica y práctica en estos campos. A través de un enfoque multidisciplinario que combina la lingüística, la sociología y la educación,

pretende comprender mejor las dinámicas y los matices del discurso que perpetúan estereotipos y prejuicios sobre la inmigración y el racismo. [\[1\]](#).

El objetivo preciso de este Trabajo de Fin de Grado es el desarrollo de DeStance como una plataforma integral que funcione como complemento a RACISMMAFF [\[1\]](#), y permita tanto la administración y almacenamiento de subcorpora escritos en diferentes lenguas, como la implementación y gestión de cuestionarios interactivos, asegurando la recopilación y almacenamiento de datos de manera segura y eficiente para su posterior análisis e interpretación.

Adicionalmente, busca desarrollar cuestionarios interactivos para medir las capacidades interculturales de los estudiantes universitarios europeos a distintos niveles, adaptados para ser aplicados antes y después de un Curso Online Masivo y Abierto (MOOC). Estos cuestionarios serán integrados en la aplicación web, actuando como herramienta central para la recolección y almacenamiento de los datos derivados de estas mediciones que darán lugar a diferentes trabajos de investigación posteriores.

Este enfoque ayudará a aumentar la conciencia de los estudiantes universitarios y futuros profesores de la UE sobre sus prácticas discursivas mediante el análisis de su uso de expresiones de posición de tres tipos (posición epistémica, posición afectiva y representación del discurso) para negociar y renegociar sus posiciones hacia temas relacionados con la inmigración y el racismo, aumentando así sus capacidades interculturales y sus niveles de empatía hacia la diversidad de idiomas, culturas y razas que derivan de la recepción de inmigrantes en Europa [\[1\]](#).

En el ámbito personal, DeStance representa una oportunidad de poner en práctica los conocimientos adquiridos dentro de Grado de Ingeniería Informática en diversas materias como Ingeniería de Software, Programación Orientada a Objetos, Interacción Persona-Ordenador, entre otras, así como mi interés en especializarme en el desarrollo de aplicaciones web. Luego, mi meta sería el integrar esta aplicación en mi currículum vitae como evidencia de mis competencias y habilidades en el sector correspondiente.

1.2. Objetivos del proyecto

El objetivo del proyecto es desarrollar una aplicación web atractiva e intuitiva para los usuarios, en su mayor parte estudiantes universitarios europeos, que integre cuestionarios interactivos para medir las capacidades interculturales de estos a distintos niveles, adaptados para ser aplicados antes y después de un Curso Online Masivo y Abierto (MOOC). Para ello, DeStance tiene que cumplir con los siguientes objetivos:

- Diseñar e implementar la arquitectura de la plataforma web, asegurando su usabilidad y funcionalidad para la implementación y gestión de cuestionarios.
- Desarrollar sistemas de recopilación y almacenamiento seguros para datos generados por herramientas de compilación y almacenamiento de corpus y cuestionarios.
- Asegurar la recolección y almacenamiento eficientes de datos implementando protocolos y procedimientos para la recopilación y almacenamiento eficiente y seguro de información previa y posterior al curso en línea.
- Garantizar la accesibilidad y utilización de datos para análisis futuros mediante el desarrollo de estructuras de almacenamiento y sistemas de fácil interpretación y análisis.

Adicionalmente, el desarrollo de la Creación de Herramientas de Evaluación tiene que cumplir con los siguientes objetivos:

- Diseñar e implementar los pre y post cuestionarios interactivos destinados a medir las capacidades interculturales de los estudiantes universitarios europeos del siglo XXI.
- Implementar estos cuestionarios en la aplicación web que gestione los procesos de (1) auto-medición y (2) almacenamiento de los datos resultantes de la auto-medición previa y posterior a

la realización de un curso en línea propuesto en la misma aplicación web.

- Diseñar un dashboard para monitorización del proceso de aprendizaje.

2. ESTADO DEL ARTE

En este capítulo se presenta una colección de herramientas actualmente existentes en materia de gestión de cuestionarios, así como plataformas en general en las que inspirarse para la realización de este proyecto.

2.1. Google Forms

Google Forms es una herramienta de encuestas y formularios en línea desarrollada por Google. Es una herramienta versátil y fácil de usar para la creación de formularios y encuestas en línea, especialmente adecuada para usuarios que buscan una solución gratuita y bien integrada dentro del ecosistema de Google. Google Forms ofrece robustas medidas de seguridad, como cifrado de datos y control de acceso, garantizando la privacidad de la información recopilada. Además, es altamente escalable y versátil, adaptándose fácilmente tanto a encuestas simples como a formularios complejos para recopilación de datos [\[5\]](#).

Algunas limitaciones incluyen su personalización limitada en comparación con herramientas especializadas, así como funciones de análisis básicas que pueden no satisfacer las necesidades de usuarios que requieren un análisis más profundo de los datos. Además, su dependencia de una conexión a Internet puede ser una limitación en entornos con conectividad limitada, y para usuarios con necesidades específicas de diseño o funcionalidad, Google Forms puede carecer de algunas características avanzadas disponibles en herramientas de encuestas más especializadas y de pago [\[6\]](#).

Esta herramienta es relevante al ser la guía para la sección de pre y post cuestionarios y sus dashboards. Se decidió no integrar directamente Google Forms en la aplicación para lograr una mayor personalización acorde a los requisitos del proyecto, incluyendo además las restricciones en la creación de estos. En la [Figura 1](#) y [Figura 2](#) se puede observar un ejemplo de la creación de cuestionarios y posterior interacción con este. Por último, en la [Figura 3](#) consiste en un ejemplo de dashboard generado automáticamente al responder el cuestionario online, lo cual será muy similar a la sección de dashboards de DeStance.

The screenshot shows the Google Forms editor interface. At the top, there's a header with the form title 'Formulario sin título', a folder icon, a star icon, and a purple 'Enviar' button. Below the header, there are three tabs: 'Preguntas' (selected), 'Respuestas', and 'Configuración'. The main area displays a question titled 'Pregunta sin título' with a linear scale type 'Escala lineal'. The scale ranges from 1 to 5. Below the scale, there are two labels: '1 Etiqueta (opcional)' and '5 Etiqueta (opcional)'. At the bottom right of the question card, there are icons for copy, delete, and a toggle for 'Obligatorio' (Required).

Figura 1: Gestión de Cuestionarios en la plataforma Google Forms.

The screenshot shows the Google Forms response interface. At the top, there's a header with the form title 'Formulario sin título'. Below the title, there's a section with the user's email 'grettellus99@gmail.com', a link to 'Cambiar de cuenta', and a status 'No compartido'. To the right, there's a cloud icon and the text 'Borrador guardado'. The main area displays a question titled 'Pregunta sin título' with a linear scale. The scale has five options: 1, 2, 3, 4, and 5. Option 3 is selected, indicated by a purple circle. Below the scale, there's a button 'Borrar selección'. At the bottom left, there's a purple 'Enviar' button, and at the bottom right, there's a link 'Borrar formulario'.

Figura 2: Interacción con cuestionarios en la plataforma Google Forms.

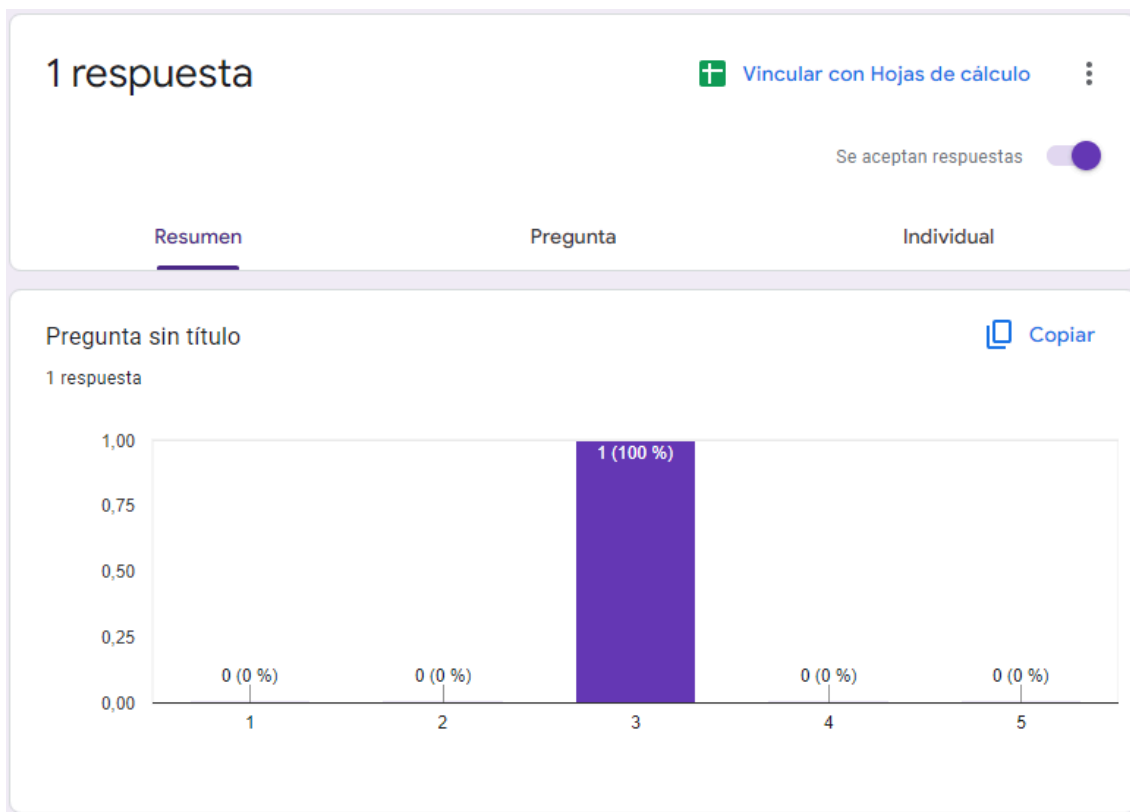


Figura 3: Dashboard con las respuestas a un cuestionario en la plataforma Google Forms.

2.2. Moodle

Moodle es una plataforma de gestión del aprendizaje versátil que permite a los usuarios crear cursos en línea y gestionar el contenido educativo de manera eficiente. Con herramientas para interactuar, realizar un seguimiento del progreso y personalizar el entorno de aprendizaje, Moodle ofrece una experiencia flexible tanto para instituciones educativas como para empresas que buscan ofrecer formación en línea [\[7\]](#).

La gratuidad y la comunidad activa de usuarios y desarrolladores son puntos destacados de Moodle. Al ser de código abierto, es accesible para organizaciones con presupuestos limitados, mientras que la contribución continua de la comunidad garantiza una amplia variedad de plugins, temas y recursos adicionales que amplían su funcionalidad y versatilidad [\[7\]](#).

Sin embargo, Moodle también presenta desafíos, como una curva de aprendizaje inicial y posibles problemas de rendimiento en entornos con alta

demanda. El soporte técnico oficial puede ser limitado, lo que puede representar un obstáculo para las organizaciones que necesitan asistencia inmediata. A pesar de estas limitaciones, sigue siendo una opción popular y efectiva para la gestión del aprendizaje en línea [7].

Moodle, específicamente la implementación de la Universidad Politécnica de Madrid, es relevante para este proyecto como plataforma base por su simplicidad y popularidad con los estudiantes universitarios europeos, usuarios objetivo de DeStance, destacando el menú de navegación izquierdo que permite acceder a todas las partes de la aplicación, concepto que se puede observar en la [Figura 4](#) y se pretende usar como guía para la interfaz de usuario.

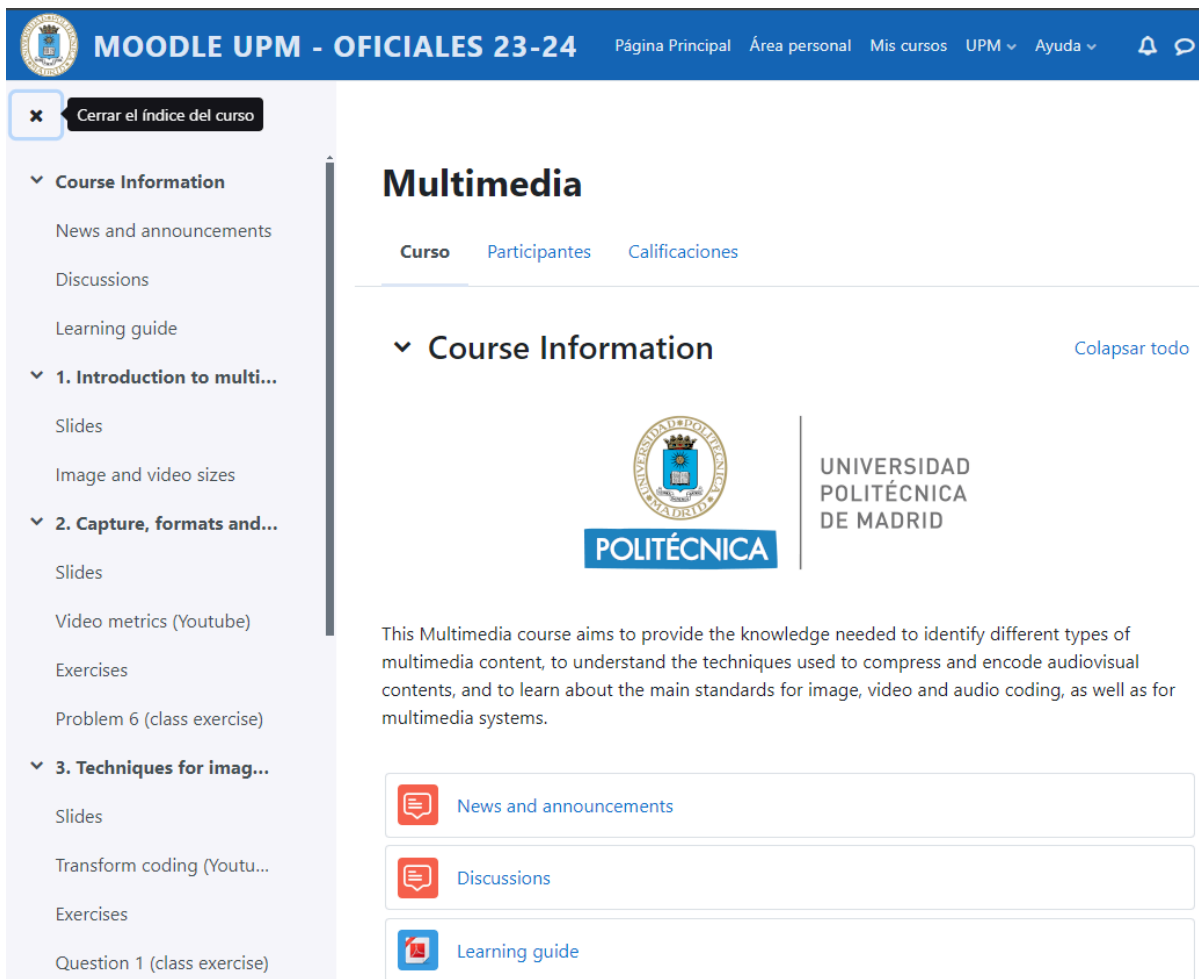


Figura 4: Plataforma Moodle como guía para la interfaz de usuario de DeStance.

3. ESPECIFICACIÓN DE REQUISITOS

En este capítulo se presentan los requisitos funcionales, es decir, servicios, prestaciones, features, puntos de menú, y no-funcionales como requisitos de calidad, fiabilidad, seguridad, usabilidad, etc., que deberán ser satisfechos por el sistema. Cada uno de ellos será identificado con un número. Además, los funcionales poseen una etiqueta breve.

Cada requisito tiene asignada una prioridad de implementación, que puede ser alta, media o baja. La prioridad alta es para aquellas funciones cuya ausencia no sería tolerable en el futuro sistema. La prioridad media es para aquellas que necesitan más discusión. Finalmente, la prioridad baja es para las funciones que no son necesarias ahora, pero deberán incorporarse en el medio-largo plazo.

3.1. Requisitos Funcionales

3.1.1. Consulta de funcionalidades de DeStance

- 1) Consulta. Bienvenida. Prioridad: Alta. El sistema mostrará a los usuarios una descripción de la herramienta y forma de uso en una serie de pasos intuitivos.
- 2) Consulta. Conceptos. Prioridad: Alta. El sistema mostrará a los usuarios una descripción de los siguientes conceptos: Postura Epistémica, Postura Efectiva, Postura Emocional, Empatía, Enseñanza y Aprendizaje Afectivo, Competencia Intercultural y Rutinas de Pensamiento Socioemocional.
- 3) Consulta. About the tool. Prioridad: Alta. El sistema mostrará a los usuarios una serie de respuestas a preguntas frecuentes.
- 4) Consulta. Cuestionarios. Preguntas. Prioridad: Alta. El sistema tendrá disponibles inicialmente para los usuarios tres pre-cuestionarios y tres post-cuestionarios, los cuales serán confeccionados por el equipo de investigadores del proyecto RACISMMAFF [\[1\]](#). Estos consisten en una

serie de preguntas con cinco respuestas posibles etiquetadas en una escala de 1 a 5.

- 5) Consulta. Cuestionarios. Respuestas. Prioridad: Alta. El sistema mostrará las respuestas a cada pregunta de aquellos cuestionarios que el usuario haya respondido anteriormente. Además mostrará el título del resultado obtenido y un contenido asociado que lo describe.
- 6) Consulta. Dashboard. Cuestionarios. Prioridad: Alta. El sistema mostrará a los usuarios administradores estadísticas de cada pre y post cuestionario a partir de las respuestas de los usuarios guardadas en la base de datos. Se mostrará adicionalmente estadísticas según los criterios de nacionalidad, edad, género, lengua materna y lenguaje de respuesta del Curso Online Masivo y Abierto (MOOC - acrónimo en inglés) incluyendo su nivel de competencia.
- 7) Consulta. MOOC. Cursos. Prioridad: Alta. El sistema permitirá a los usuarios ver detalles específicos de cada curso, incluyendo contenidos del curso y las preguntas.
- 8) Consulta. MOOC. Dashboard. Prioridad: Alta. El sistema mostrará las respuestas escritas por los usuarios y facilitará en gráfica los tipos de las palabras utilizadas basado en minería de texto.
- 9) Consulta. Autorización. Prioridad: Alta. Se deshabilitarán las opciones de administración: dashboards y mantenimiento de cuestionarios a los usuarios no administradores, por lo que es necesario distinguir los tipos de usuario en la base de datos.

3.1.2. Comunicación bidireccional con los Usuarios

- 10) Cuestionarios. Respuesta. Prioridad: Alta. El sistema permitirá a los usuarios responder cada pregunta de cada cuestionario seleccionando la respuesta que crean conveniente. Una vez respondidas todas las preguntas de un cuestionario se almacenarán las respuestas en la base de datos usando la escala de 1 a 5. Además, se inferirá una conclusión

dependiendo de las respuestas dadas a través de la media de los valores de la escala. Solo se permitirá responder cada cuestionario una vez.

- 11) Cuestionarios. Elaboración. Automatización. Prioridad: Baja. El sistema permitirá a los usuarios administradores crear nuevos cuestionarios de tipo pre o post. Permitirá introducir cada pregunta con las cinco etiquetas de las posibles respuestas que representan una escala de 1 a 5, así como introducir una conclusión dependiendo del rango de la escala con una descripción de cada posible resultado.
- 12) Cuestionarios. Mantenimiento. Automatización. Prioridad: Baja. El sistema permitirá a los usuarios administradores modificar y borrar cuestionarios ya existentes.
- 13) MOOC. Preguntas. Prioridad: Alta. El sistema mostrará preguntas específicas en diferentes etapas de cada curso, las cuales serán confeccionadas por el equipo de investigadores del proyecto RACISMMAFF [\[1\]](#).
- 14) MOOC. Respuestas. Prioridad: Alta. El sistema permitirá a los usuarios responder a las preguntas del curso en forma de texto, validando que se cumpla con un mínimo de 200 palabras, lo cual deberá ser almacenado en la base de datos.
- 15) MOOC. Respuestas. Retroalimentación. Prioridad: Alta. El sistema deberá analizar la respuesta del usuario para ofrecer retroalimentación sobre las palabras más utilizadas.
- 16) MOOC. Descargar Archivos. Prioridad: Alta. El sistema permitirá descargar archivos PDF con el resumen de cada curso.
- 17) Usuarios. Datos. Prioridad: Alta. El sistema creará una cuenta a cada usuario al registrarse recogiendo nombre, correo electrónico, contraseña, fecha de nacimiento, género, nacionalidad, lengua materna, lenguaje que utilizará para responder el MOOC y nivel de competencia que tiene en dicho lenguaje, de A1 a C2.

- a) Usuarios. Datos. Modificación. Prioridad: Baja. El sistema permitirá a los usuarios modificar sus datos personales introducidos anteriormente al crearse su cuenta.
- b) Usuarios. Datos. Contraseña. Recuperación. Prioridad: Baja El sistema permitirá a los usuarios recuperar su contraseña. Para ello deberán introducir un email de recuperación de contraseña que debe coincidir con el del propio usuario. El sistema enviará un código al email introducido, válido por máximo una hora, que deberá ser utilizado para validar la nueva contraseña.

3.3. Requisitos de Interfaces Externos

3.2.1. Interfaces de Usuario

- 18) Prioridad: Alta. La interfaz de usuario deberá ser accesible a través de un navegador web y atractiva para lograr una buena experiencia de usuario.

3.2.3. Interfaces de Software

- 19) Prioridad: Baja. Se necesita conectar el sistema con la aplicación externa como SendGrid [\[8\]](#) para el envío de emails de recuperación de contraseña.

3.2.3. Interfaces de Comunicación

- 20) Prioridad: Alta. Uso de los servicios e infraestructura proporcionada por la aplicación web Firebase [\[9\]](#) para el hosting de los servidores web (front-end y back-end) de DeStance.

3.3. Requisitos de Desarrollo

- 21) Prioridad: Alta. El ciclo de vida elegido para desarrollar el producto será iterativo incremental, de manera que se puedan incorporar fácilmente cambios y nuevas funciones. Habrá que definir, por tanto, en qué orden se implementarán las funcionalidades en cada ciclo.

3.4. Requisitos Tecnológicos

En cuanto a la infraestructura informática necesaria se seguirá el patrón MEAN de desarrollo web (MongoDB, Express.js, Node.js, Angular):

3.4.1. *Front-end*

22)Prioridad: Alta. Un servidor web reactivo *Single Web Application* (SPA) utilizando el Framework de Javascript Angular [\[10\]](#).

3.4.2. *Back-end*

23)Prioridad: Alta. Un servidor REST API que realice las principales operaciones de la aplicación, incluyendo TODAS las operaciones con la base de datos. Se utilizará el entorno de ejecución de javascript Node.js [\[11\]](#) y adicionalmente el framework complementario a este Express.js [\[12\]](#) para facilitar el desarrollo.

3.4.3. *Base de Datos*

24)Prioridad: Alta. Un servidor donde alojar la base de datos de DeStance. Este recibirá peticiones del servidor de back-end correspondientes a las operaciones de consulta, modificación, creación y eliminación de datos. Se utilizará para ello MongoDB Atlas [\[13\]](#).

3.5. Requisitos No Funcionales

3.5.1. *Disponibilidad y Rendimiento*

25)Prioridad: Alta. El sistema estará accesible 24/7. Debe poseer además un tiempo de respuesta menor a 2 segundos para el 95% de las solicitudes y soportar al menos 200 usuarios concurrentes sin degradación significativa del rendimiento.

3.5.2. Mantenibilidad

26)Prioridad: Alta. El sistema requerirá el mínimo posible de mantenimiento. El código debe estar estructurado de manera modular para facilitar las actualizaciones y el mantenimiento.

3.5.3. Accesibilidad

27)Prioridad: Alta. La interfaz de usuario debe ser intuitiva y fácil de usar, con una navegación clara y coherente. Además debe cumplir con los estándares WCAG 2.1 AA [\[14\]](#) para asegurar que sea accesible para usuarios con discapacidades.

3.5.4. Seguridad

28)Prioridad: Alta. Implementar autenticación con email y contraseña y permisos basados en los roles de usuario y administrador para asegurar que solo usuarios autorizados puedan acceder a determinadas funcionalidades. Todos los datos sensibles deben ser encriptados tanto en tránsito (usando HTTPS) como en reposo. La aplicación debe cumplir con las regulaciones pertinentes como GDPR [\[15\]](#), y otros estándares de privacidad y seguridad de datos aplicables.

3.5.5. Compatibilidad

29)Prioridad: Alta. La aplicación debe ser compatible con las versiones actuales y anteriores de los navegadores más utilizados (Chrome, Firefox, Safari, Edge).

3.5.6. Idioma

30)Prioridad: Alta. La aplicación debe estar disponible en el idioma inglés y soportar la traducción ofrecida por los navegadores.

4. ELECCIÓN DE TECNOLOGÍAS WEB

Este capítulo ofrece una recopilación de tecnologías destinadas al desarrollo de aplicaciones web, subrayando sus características más relevantes, la conexión entre sus funcionalidades y la plataforma en desarrollo, es decir, su pertinencia para el proyecto, así como un resumen de las tecnologías seleccionadas.

4.1. Tecnologías para aplicaciones web full-stack

En el desarrollo de aplicaciones web modernas, la elección de un stack tecnológico adecuado es crucial para garantizar el éxito y la eficiencia del proyecto. Un stack tecnológico es un conjunto de tecnologías y herramientas que se utilizan conjuntamente para construir y ejecutar una aplicación. Algunas de las opciones más populares se encuentran los stacks *MEAN*, *MERN*, *LAMP* y *Django*, cada uno con sus propias características y ventajas.

4.1.4. *MERN Stack*:

El MERN Stack está compuesto por MongoDB, Express.js, React y Node.js. Este conjunto de tecnologías permite desarrollar aplicaciones web completas utilizando JavaScript tanto en el front-end como en el back-end. MongoDB ofrece flexibilidad en el almacenamiento de datos no relacionales, mientras que Express.js y Node.js facilitan la creación de servidores robustos y escalables. React, por su parte, es conocido por su rendimiento y modularidad en la creación de interfaces de usuario dinámicas y reactivas. El uso unificado de JavaScript en todo el stack simplifica el desarrollo y el mantenimiento del código [\[16\]](#).

4.1.2. *LAMP Stack*:

El LAMP Stack incluye Linux, Apache, MySQL y PHP. Es una combinación clásica y ampliamente utilizada para el desarrollo de aplicaciones web dinámicas. Linux proporciona una plataforma estable y segura; Apache es un servidor web altamente configurable y eficiente; MySQL es un sistema de gestión de bases de datos relacional confiable; y PHP es un lenguaje de

scripting flexible y fácil de aprender, especialmente adecuado para la creación rápida de páginas web. La principal ventaja del LAMP Stack es su solidez y la vasta cantidad de recursos y comunidades disponibles para soporte [\[17\]](#).

4.1.3. Django Stack

El Django Stack se basa en Python, el framework web Django, PostgreSQL y Nginx. Django es un framework de alto nivel que promueve el desarrollo rápido y limpio del código, siguiendo el principio de "no te repitas" (DRY). PostgreSQL es una base de datos relacional potente y escalable que soporta complejas consultas y operaciones de datos. Nginx actúa como un servidor web eficiente y un proxy inverso. Los beneficios de este stack incluyen la facilidad de desarrollo y mantenimiento, la robustez y seguridad de las aplicaciones, y la capacidad de manejar grandes volúmenes de tráfico [\[18\]](#).

4.1.4. MEAN Stack:

El MEAN Stack, similar a MERN Stack, está compuesto por MongoDB, Express.js, Angular (en lugar de React) y Node.js. Esta combinación de tecnologías permite el desarrollo de aplicaciones web completas usando JavaScript en todas las capas de la aplicación. MongoDB, una base de datos NoSQL, facilita el manejo de datos en formato JSON, lo que se alinea perfectamente con el manejo de datos en JavaScript [\[19\]](#).

Express.js y Node.js permiten crear servidores y APIs robustos y escalables, mientras que Angular proporciona un framework potente para el desarrollo de interfaces de usuario dinámicas y reactivas. Los beneficios del MEAN Stack incluyen la uniformidad del lenguaje de programación en todo el stack, lo cual simplifica el desarrollo y la colaboración entre equipos, así como la flexibilidad y escalabilidad que ofrece para proyectos modernos y de gran envergadura [\[19\]](#).

4.2. Tecnología escogida

Cabe destacar que este proyecto no será desarrollado de forma aislada (en este caso, por dos estudiantes y en un período específico de tiempo, por ejemplo,

los 3-4 meses durante los que cursa la asignatura), sino que suelen ser mejorados por nuevos alumnos o profesionales en años venideros. Por tanto, se considera de gran relevancia escoger un conjunto de tecnologías con vistas a futuro, que sea relativamente fácil de aprender y de gran popularidad en el mercado laboral para que colaborar en el proyecto sea una experiencia deseable para nuevos alumnos o contratar profesionales no sea complicado.

Tomando como referencia la encuesta anual a desarrolladores en su versión de 2023 [20] realizada por Stack Overflow [21], popular sitio web de preguntas y respuestas para programadores profesionales y aficionados que contiene más de 50 millones de preguntas y respuestas sobre muchísimos temas relacionados con la tecnología, se obtienen los siguientes hallazgos mostrados en la [Figura 5](#) y la [Figura 6](#) respecto al uso de tecnologías para el desarrollo de aplicaciones web:

Figura 5: ¿En qué frameworks web y tecnologías web ha realizado un trabajo de desarrollo extenso durante el año pasado y en cuáles desea trabajar durante el próximo año?

Figura 6: ¿En qué entornos de bases de datos ha realizado un extenso trabajo de desarrollo durante el año pasado y en cuáles desea trabajar durante el próximo año?

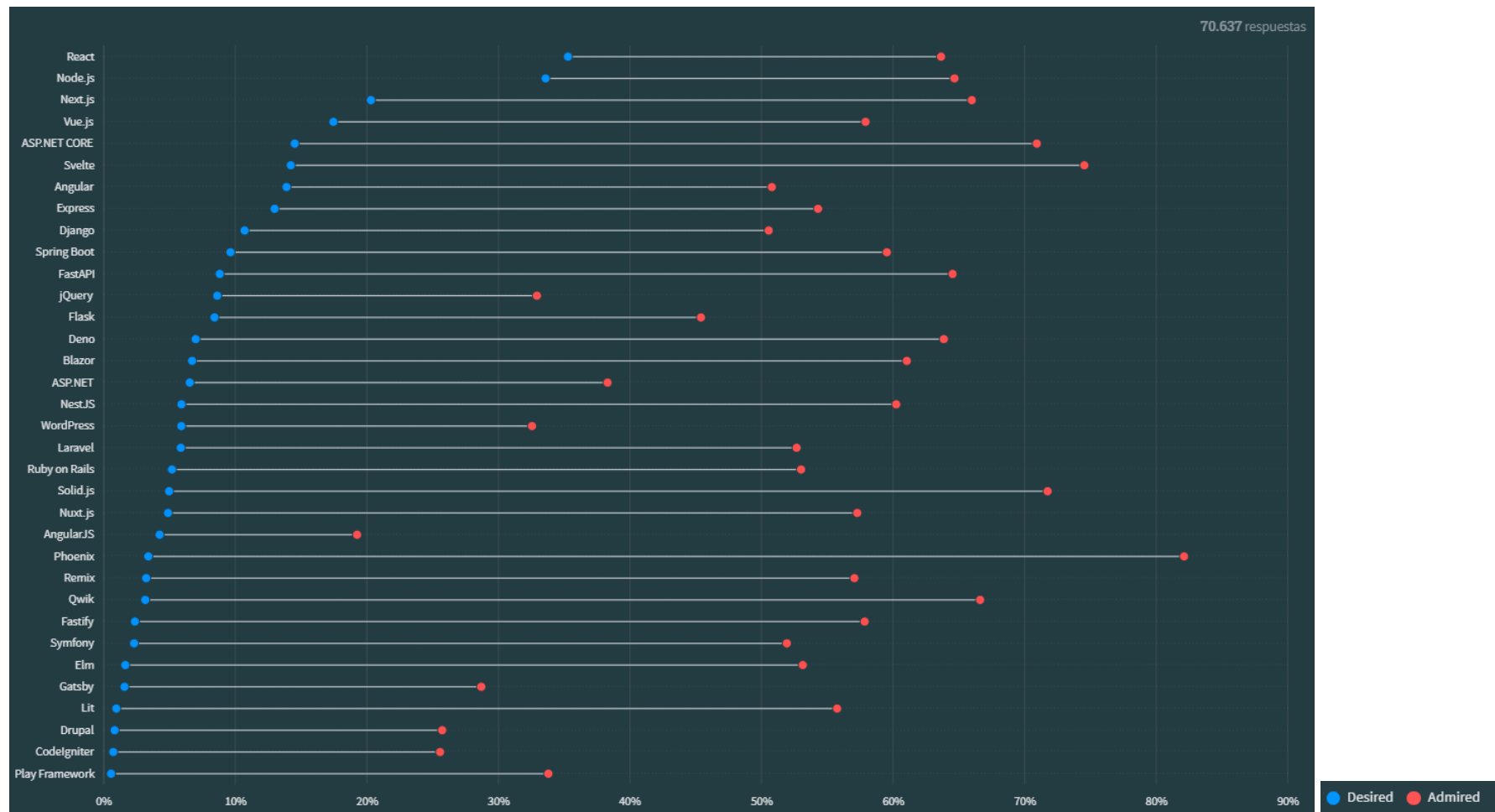


Figura 5: Encuesta de Stack Overflow a sus usuarios sobre frameworks y tecnologías web más usadas por estos [20].

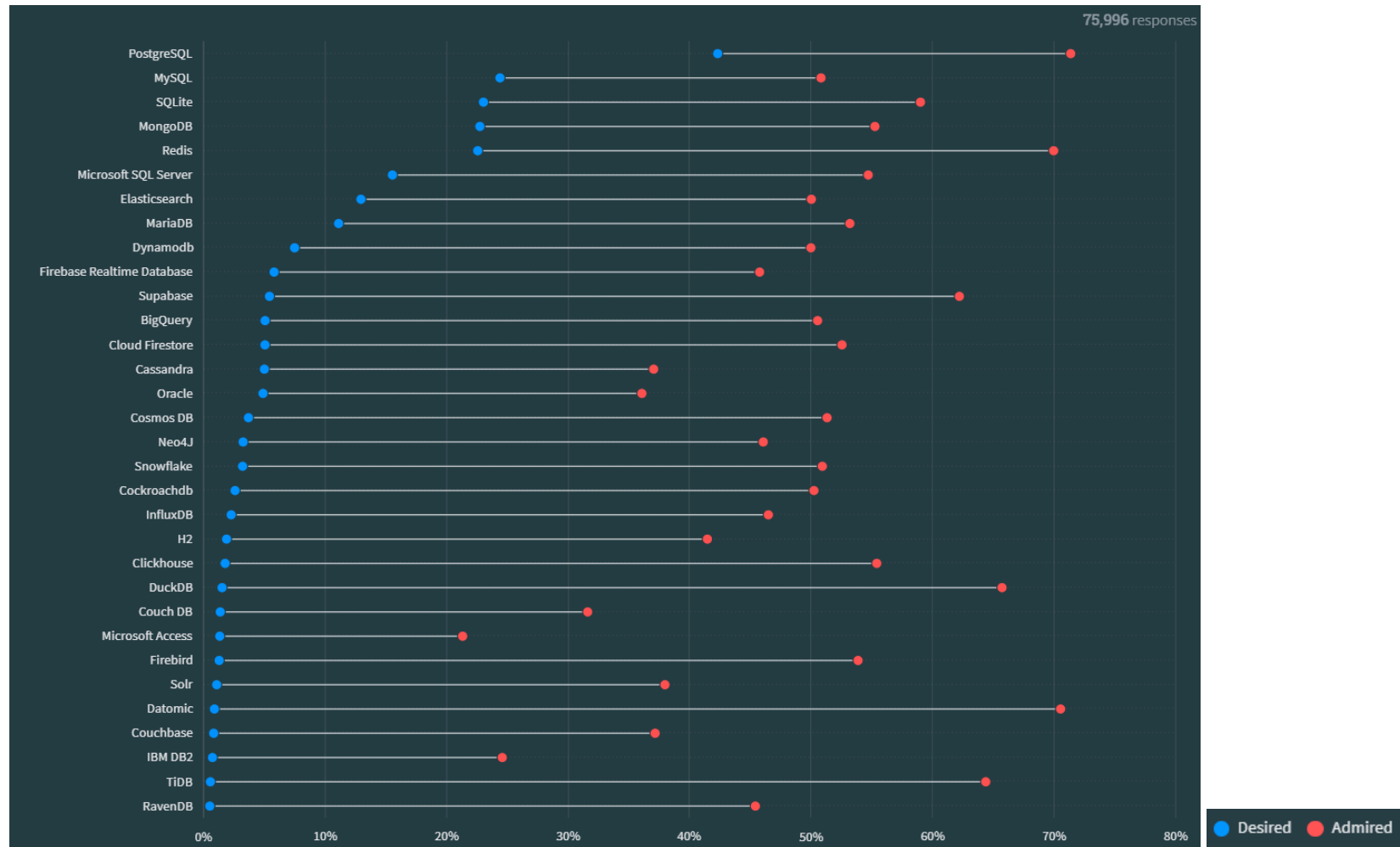


Figura 6: Encuesta de Stack Overflow a sus usuarios sobre bases de datos más usadas por estos [20].

En cuanto al desarrollo de front-end, como se aprecia en la [Figura 5](#), React es el framework más deseado por los encuestados, seguido por Next.js, Vue.js y Angular en la tercera, cuarta y séptima posición general. Por otra parte, Node.js es, en cuanto al desarrollo de back-end, el más demandado, segundo en general. En la [Figura 6](#), PostgreSQL, MySQL SQLite y MongoDB destacan en los cuatro primeros lugares de los entornos de bases de datos más populares.

Como se mencionó anteriormente en la [sección 3.4](#) del presente documento se opta por MEAN stack como el conjunto base de tecnologías de desarrollo. Una de las razones principales que decantó la decisión fue el conocimiento previo de desarrollo web con estas tecnologías por parte de la autora de este Trabajo de Fin de Grado, suponiendo menor complejidad en la curva de aprendizaje personal. Además, dada la popularidad de la solución, existe gran cantidad de documentación y tutoriales de ayuda en la web, incluyendo cursos especializados en plataformas online como Udemey Academy [\[22\]](#).

Otro factor determinante, dado el poco tiempo disponible para desarrollar DeStance y la necesidad de lograr una interfaz de usuario llamativa, fue la experiencia personal previa en el desarrollo de front-end con la librería de componentes UI diseñada específicamente para Angular, Angular Material [\[23\]](#), que ofrece una amplia gama de elementos de diseño y funcionalidades predefinidas para la creación rápida y consistente de interfaces de usuario modernas y atractivas.

Los beneficios de Angular Material son numerosos: en primer lugar, proporciona un conjunto coherente de componentes visuales que siguen las directrices de Material Design, lo que garantiza una experiencia de usuario intuitiva y familiar. Además, al ser una biblioteca oficial de Angular, Angular Material se integra perfectamente con el framework, lo que facilita su uso y mantenimiento en proyectos de Angular. Otra ventaja clave es la personalización y extensibilidad de los componentes, que permiten a los desarrolladores adaptar fácilmente el aspecto y la funcionalidad de los elementos a las necesidades específicas de su aplicación [\[24\]](#).

La alternativa con React y la opción MERN era Material UI [\[25\]](#). Esta es una librería de componentes de React que implementa, como en el caso de Angular Material, el lenguaje de diseño de Google, Material Design. Material UI es una librería de código abierto que se puede utilizar de forma gratuita y que cuenta con una gran comunidad de desarrolladores que la utilizan y la mantienen. Sin embargo, dada la mayor experiencia en el trabajo con Angular Material, se decanta la decisión por MEAN.

4.3. MEAN en detalle

La arquitectura MEAN está diseñada para hacer que la creación de aplicaciones web en JavaScript y el manejo de JSON sean increíblemente fáciles. Esta sigue el patrón de pila tradicional de tres niveles, incluido el nivel de visualización (Angular), el nivel de aplicación (Express.js y Node.js) y el nivel de base de datos (MongoDB). A continuación ([Figura 7](#)) se muestra como interactúan sus componentes:

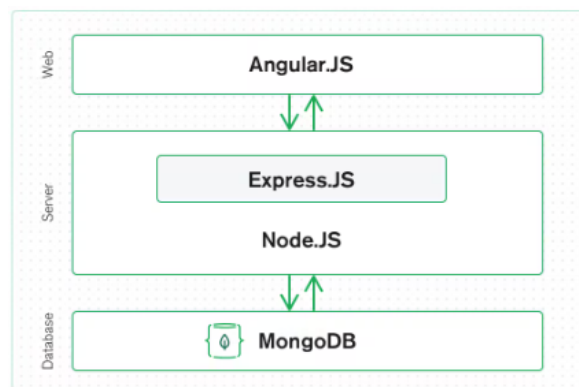


Figura 7: Componentes pila MEAN y su interacción.

En la parte superior de la pila MEAN se encuentra Angular.js o Angular (versión más reciente), el autodenominado “JavaScript MVW Framework” (MVW significa “Model View and Anyway”). Angular es un framework de desarrollo frontend mantenido por Google, que permite construir interfaces de usuario dinámicas y reactivas. Utiliza un enfoque basado en componentes, lo que promueve la reutilización del código y facilita el mantenimiento y la escalabilidad de las aplicaciones [\[19\]](#).

Angular permite ampliar sus etiquetas HTML con metadatos para crear experiencias web dinámicas e interactivas de manera mucho más poderosa que, por ejemplo, crearlas con HTML estático y JavaScript (o jQuery) [\[26\]](#). En este sentido, ofrece un potente sistema de enlace de datos bidireccional, que sincroniza automáticamente los cambios entre el modelo y la vista, reduciendo la cantidad de código necesario para mantener la consistencia de los datos. Además, su sólido sistema de directivas y servicios permite a los desarrolladores crear aplicaciones modulares y altamente personalizables [\[19\]](#).

El siguiente nivel inferior es Express.js , que se ejecuta en un servidor Node.js. Express.js se autodenomina un “framework web minimalista, rápido y sin opiniones para Node.js”, y eso es exactamente lo que es. Express.js tiene modelos potentes para el enrutamiento de URL (haciendo coincidir una URL entrante con una función de servidor) y el manejo de solicitudes y respuestas HTTP. Al realizar solicitudes HTTP XML (XHR), GET o POST desde la interfaz Angular, puede conectarse a las funciones de Express.js que impulsan la aplicación. Esas funciones, a su vez, utilizan los controladores Node.js de MongoDB, ya sea mediante devoluciones de llamada o promesas, para acceder y actualizar datos en su base de datos MongoDB [\[26\]](#).

Si la aplicación almacena datos (perfiles de usuario, contenido, comentarios, cargas, eventos, etc.), entonces requerirá una base de datos con la que sea tan fácil trabajar con Angular, Express y Node. Ahí es donde entra en juego MongoDB: los documentos JSON creados en su interfaz Angular.js se pueden enviar al servidor Express.js, donde se pueden procesar y (suponiendo que sean válidos) almacenar directamente en MongoDB para su posterior recuperación. Si se desea obtener fácilmente lo mejor de MongoDB, se deberá consultar MongoDB Atlas [\[13\]](#) [\[26\]](#). Esto permitirá integrar seguridad completa de la base de datos y escalabilidad en la nube fácilmente.

MongoDB almacena datos en un formato similar a JSON (BSON, una extensión JSON binaria), las API de consulta de MongoDB están definidas en JSON y su interfaz de línea de comandos (CLI) es un intérprete de JavaScript. MongoDB no solo es esencialmente un almacén de datos JavaScript/JSON,

sino que también está lleno de funciones avanzadas como indexación y consultas profundas en documentos JSON, tiene potentes controladores nativos Node.js y está diseñado para escalamiento horizontal. Es aún más fácil desarrollar aplicaciones en la nube utilizando MongoDB Atlas, la base de datos nativa de la nube como servicio de los creadores de MongoDB [\[26\]](#).

Las bases de datos de Atlas se implementan y escalan fácilmente, lo que le proporciona un URI coherente para conectarse. En el camino, las conexiones Atlas vienen con nombre de usuario/contraseña integrados y cifrado TLS de extremo a extremo de forma predeterminada. Además, estas conexiones le permiten utilizar funciones de seguridad avanzadas de MongoDB, como autenticación de certificado/IAM, LDAP, cifrado en reposo y auditoría con solo hacer clic en un botón. Además, un proyecto Atlas puede utilizar la plataforma de aplicaciones Atlas App Services para integrar fácilmente muchos proveedores de autenticación como Google, Facebook, JWT y autenticación personalizada. Escalar y administrar Atlas es muy fácil; su mayor beneficio es que admite y protege la capa más valiosa de la pila MEAN: la capa de datos [\[26\]](#).

Además, la pila MEAN tiene una separación concreta de tres niveles que, si se utiliza con las mejores prácticas y un aislamiento de red correcto, debería impedir que los usuarios finales tengan acceso a la lógica empresarial y, además, a la capa de su base de datos. Por lo tanto, la aplicación está diseñada de forma predeterminada para evitar que la interacción maliciosa del usuario ponga en riesgo su aplicación (inyección de consultas, manipulación de código, suplantación de puertos, etc.) [\[26\]](#).

En conjunto, el MEAN Stack ofrece una solución cohesiva y eficiente para el desarrollo de aplicaciones web modernas, proporcionando una experiencia de desarrollo unificada y simplificada mediante el uso de JavaScript en todas las capas de la aplicación.

4.4. Hospedaje (hosting) de DeStance

Tomando como referencia nuevamente la encuesta anual a desarrolladores en su versión de 2023 [\[20\]](#) realizada por Stack Overflow [\[21\]](#), en el apartado de plataformas en la nube más utilizadas, las cuales proveen múltiples servicios incluyendo hosting, se obtienen los resultados mostrados en la [Figura 8](#):

Figura 8: ¿En qué plataformas en la nube ha realizado un extenso trabajo de desarrollo durante el año pasado y en cuáles desea trabajar durante el próximo año?

Amazon Web Services (AWS) parece ser la plataforma cloud más utilizada entre los encuestados, mientras que Firebase (escogida para este proyecto) se halla en quinta posición en popularidad. Aunque AWS era una de las opciones más sólidas contempladas, finalmente se escoge Firebase por tener una mayor experiencia y por tanto minimizar la curva de aprendizaje, en favor de maximizar el tiempo de desarrollo de las funcionalidades de la plataforma, considerando que se trata de un nuevo proyecto y debe ser funcional en su totalidad a final de semestre (junio 2024). En cualquier caso el servicio de hosting (plataforma cloud) es migrable, en caso de que así se deseara en un futuro.

Firebase es una plataforma de desarrollo de aplicaciones móviles y web que ofrece una amplia gama de herramientas y servicios, incluyendo bases de datos en tiempo real, autenticación de usuarios, almacenamiento de archivos, hospedaje web y más. Sus beneficios son diversos: en primer lugar, su facilidad de uso y su rápida configuración permiten a los desarrolladores comenzar a construir y lanzar aplicaciones en poco tiempo, sin la necesidad de configurar y mantener servidores [\[27\]](#).

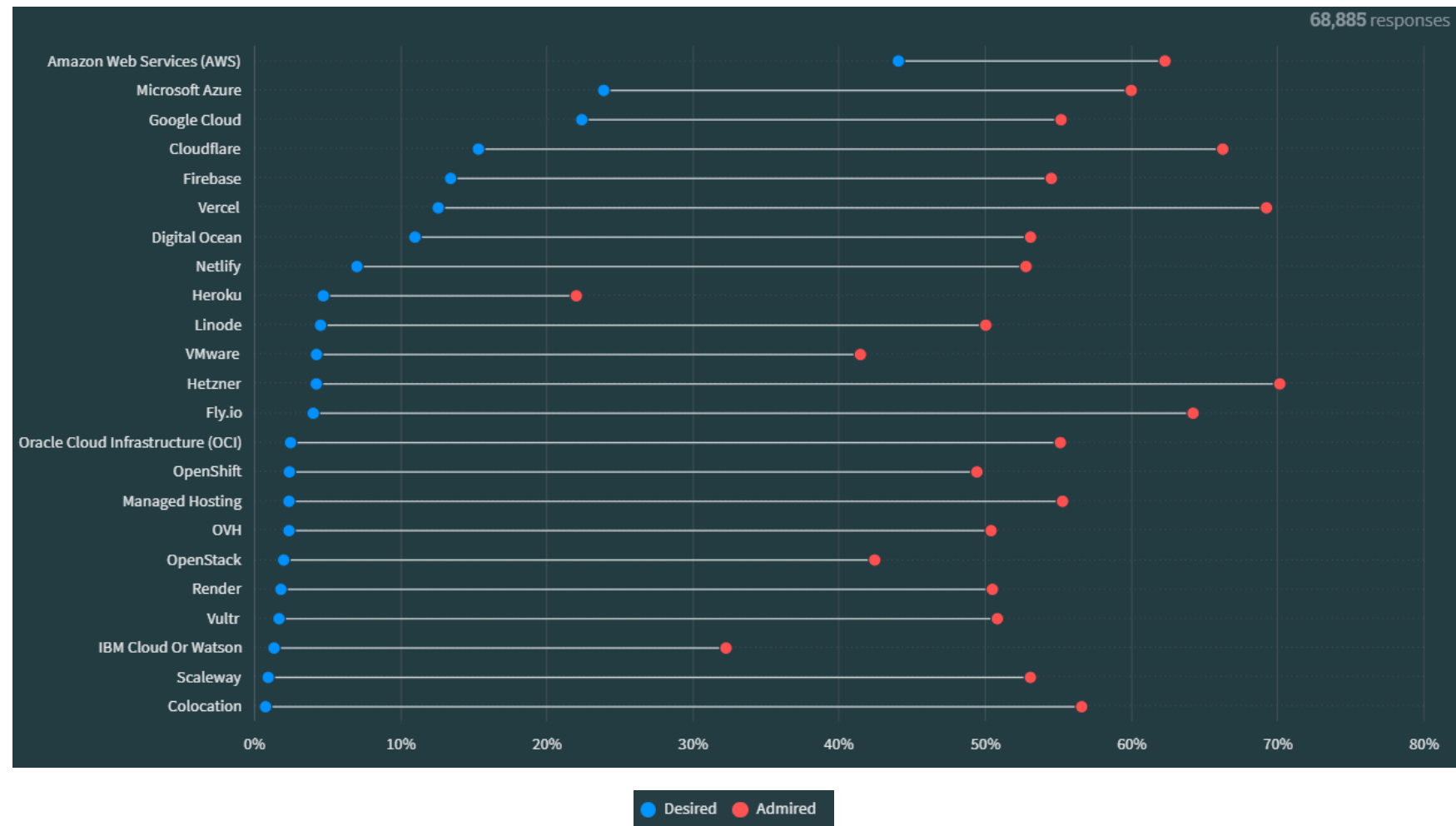


Figura 8: Encuesta de Stack Overflow a sus usuarios sobre plataformas cloud más usadas por estos [20].

Firestore ofrece una base de datos en tiempo real que sincroniza automáticamente los datos entre los clientes y el servidor, lo que permite construir aplicaciones colaborativas y en tiempo real de forma sencilla. Otra ventaja clave es su integración con otras plataformas y servicios de Google, lo que facilita el acceso a herramientas como Google Analytics y Google Cloud Messaging. En resumen, simplifica el desarrollo de aplicaciones al proporcionar una plataforma unificada y escalable para la construcción, el lanzamiento y el crecimiento de aplicaciones móviles y web [\[27\]](#).

Específicamente DeStance utilizará el servicio de hosting estático de Firebase para el alojamiento del servidor web (front-end), el servicio de hosting dinámico (denominado Functions) para el servidor de back-end y el servicio de Storage para alojar imágenes u otros recursos necesarios para el front-end. Esto forma parte de un proyecto colaborativo de Firebase de inicialmente los dos estudiantes encargados del desarrollo de DeStance, denominado homónimamente. Adicionalmente se le dará acceso al proyecto a los clientes de DeStance, utilizando un tercer colaborador (cuenta de Google) para asegurar que se pueda continuar el desarrollo o bien el mantenimiento de la aplicación sin depender de los programadores iniciales.

4.5. Resumen de la sección

El conjunto de tecnologías escogido, por tanto, será la arquitectura MEAN con Angular [\[10\]](#), Node.js [\[11\]](#), Express.js [\[12\]](#) y MongoDB [\[13\]](#), con la aplicación alojada en Firebase [\[28\]](#), excepto la base de datos, que se aloja en los servidores designados en MongoDB Atlas [\[13\]](#).

5. DISEÑO

Una vez establecidas las herramientas, los lenguajes y los entornos que conformarán la base del proyecto, el siguiente paso es el diseño e implementación de la aplicación web. Para llevar a cabo un análisis adecuado, es esencial identificar casos de uso que satisfagan las funcionalidades especificadas en los requisitos ([sección 3](#)), así como considerar la experiencia y el contexto de los usuarios al interactuar con nuestra aplicación.

5.1. Actores

Para definir las funcionalidades y el acceso a ellas, es necesario aclarar quienes son los posibles actores que podrán operar con el sistema. Se considerará usuario a cualquiera que acceda a la aplicación independientemente de su rol asignado y de si está identificado o no. Los usuarios identificados son aquellos ya tienen una cuenta en el sistema y se han logueado satisfactoriamente, incluyendo los usuarios no administradores y los administradores.

Los usuarios no identificados serán invitados. Estos últimos tendrán acceso limitado a DeStance, pudiendo acceder a las funcionalidades de introducción a la aplicación, preguntas frecuentes, conceptos, nueva cuenta, inicio de sesión y recuperación de contraseña. Los usuarios identificados podrán acceder a las funcionalidades de introducción a la aplicación, preguntas frecuentes, conceptos, cerrar sesión, detalles de su cuenta (incluyendo la posibilidad de modificarlos), a los pre-cuestionarios, el curso MOOC y los post-cuestionarios, mientras que los administradores podrán adicionalmente acceder a los dashboards de los cuestionarios y del MOOC, además de la función de gestión de cuestionarios para la creación, modificación o eliminación de estos.

5.2. Casos de Uso

Para facilitar la comprensión de los casos de uso se ofrece en la [Figura 9](#) un Diagrama de Casos de Uso que muestra la interacción de los actores con la aplicación.

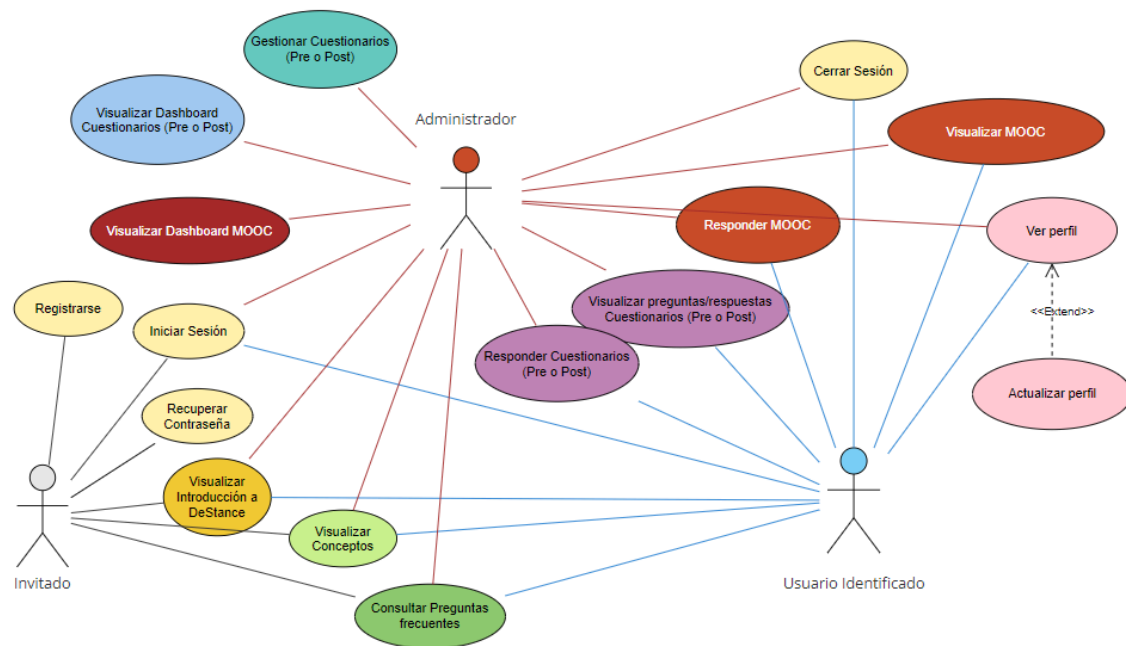


Figura 9: Diagrama de Casos de Uso de DeStance

A continuación se detallan los distintos casos de usos presentados en la [Figura 9](#). Se describe su propósito, quién lo ejecuta, las condiciones previas que se necesitarán, los resultados que podrían generarse, los datos de entrada y salida, así como la secuencia de eventos que ocurren.

I Registrarse

- a. Propósito: Crear nuevo usuario.
- b. Actor Iniciador: Invitado.
- c. Precondición: El usuario no existe en el sistema.
- d. Postcondición: Se añade nuevo usuario con su contraseña cifrada.
- e. Datos de entrada: Datos del nuevo usuario.
- f. Datos de salida: Confirmar si tuvo éxito o mostrar mensaje de error si no lo hay.
- g. Eventos:
 - i. El usuario completa el formulario de registro.
 - ii. El sistema comprueba que no hayan usuarios duplicados manteniendo el email único, cifra la contraseña y almacena la información en la base de datos.

- iii. El sistema notifica al usuario del éxito o error de la operación.

II Iniciar Sesión

- a. Propósito: Validar e identificar al usuario.
- b. Actor Iniciador: Invitado.
- c. Precondición: El usuario existe en el sistema y no está identificado.
- d. Postcondición: El usuario se mantiene identificado por máximo una hora. El sistema debe cerrar la sesión de manera automática una vez alcance dicho tiempo.
- e. Datos de entrada: Email del usuario y contraseña.
- f. Datos de salida: Confirmar si tuvo éxito o mostrar mensaje de error si no lo hay.
- g. Eventos:
 - i. El usuario introduce su email y contraseña.
 - ii. El sistema comprueba si los datos corresponden con alguno de los usuarios existentes.
 - iii. El sistema notifica al usuario del éxito o error de la operación. En el primer caso se identifica al usuario por un período máximo de una hora.

III Recuperación de contraseña

- a. Propósito: Cambiar la contraseña cuando el usuario no la recuerda.
- b. Actor Iniciador: Invitado.
- c. Precondición: El usuario existe en el sistema y no está identificado. El email de recuperación introducido debe coincidir con el email de algún usuario existente.
- d. Postcondición: Se actualiza la contraseña del usuario en la base de datos con la nueva contraseña introducida. Esta se cifra. Si tiene éxito la operación se manda al usuario a la “página” de iniciar sesión.

- e. Datos de entrada: Email de recuperación de contraseña, código de seguridad enviado al email y nueva contraseña.
- f. Datos de salida: Confirmar si tuvo éxito o mostrar mensaje de error si no lo hay.
- g. Eventos:
 - i. El usuario introduce el email de recuperación de contraseña.
 - ii. El sistema comprueba que existe el email introducido en la base de datos y envía un correo con un código de seguridad.
 - iii. El usuario introduce el código de seguridad y la nueva contraseña.
 - iv. El sistema notifica al usuario del éxito o error de la operación. En el primer caso se actualiza la contraseña del en la base de datos con la nueva contraseña introducida. Esta se cifra.

IV Cerrar Sesión

- a. Propósito: Cerrar la sesión activa del usuario.
- b. Actor Iniciador: Usuario Identificado.
- c. Precondición: El usuario existe en el sistema, está identificado por un período menor a una hora.
- d. Postcondición: El usuario cambia su rol a invitado.
- e. Datos de entrada: -
- f. Datos de salida: Confirmar si tuvo éxito o mostrar mensaje de error si no lo hay.
- g. Eventos:
 - i. El usuario selecciona la opción de Cerrar Sesión.
 - ii. El sistema cierra la sesión activa del usuario cambiando su rol a invitado.

V Visualizar Introducción a DeStance

- a. Propósito: Obtener información general del sitio incluyendo modo de uso.

- b. Actor Iniciador: Invitado, Usuario Identificado o Administrador.
- c. Precondición: -
- d. Postcondición: -
- e. Datos de entrada: -
- f. Datos de salida: Información del sitio.
- g. Eventos:
 - i. El actor selecciona esta “página” en el menú de navegación o simplemente accede al sitio web.
 - ii. El sistema muestra información de bienvenida a DeStance.

VI Visualizar Conceptos

- a. Propósito: Obtener detalles de los conceptos más relevantes del contenido instructivo de DeStance.
- b. Actor Iniciador: Invitado, Usuario Identificado o Administrador.
- c. Precondición: -
- d. Postcondición: -
- e. Datos de entrada: -
- f. Datos de salida: En la versión actual de DeStance se proporcionarían los conceptos de Postura Epistémica, Postura Efectiva, Postura Emocional, Empatía, Enseñanza y Aprendizaje Afectivo, Competencia Intercultural y Rutinas de Pensamiento Socioemocional.
- g. Eventos:
 - i. El actor selecciona esta “página” en el menú de navegación.
 - ii. El sistema muestra los conceptos más relevantes del contenido instructivo de DeStance.

VII Visualizar Preguntas Frecuentes

- a. Propósito: Obtener respuesta a preguntas frecuentes que pudiesen tener los usuarios de DeStance.
- b. Actor Iniciador: Invitado, Usuario Identificado o Administrador.
- c. Precondición: -

- d. Postcondición: -
- e. Datos de entrada: -
- f. Datos de salida: Respuestas a preguntas frecuentes.
- g. Eventos:
 - i. El actor selecciona esta “página” en el menú de navegación.
 - ii. El sistema muestra respuesta a preguntas frecuentes que pudiesen tener los usuarios de DeStance.

VIII Ver perfil

- a. Propósito: Consultar los datos personales introducidos anteriormente en la aplicación.
- b. Actor Iniciador: Usuario Identificado o Administrador.
- c. Precondición: El usuario existe en el sistema y está identificado.
- d. Postcondición: -
- e. Datos de entrada: -
- f. Datos de salida: Datos personales anteriormente introducidos.
- g. Eventos:
 - i. El actor selecciona esta “página” en el menú de navegación.
 - ii. El sistema muestra sus datos personales anteriormente introducidos.

IX Actualizar perfil

- a. Propósito: Modificación de los datos personales introducidos anteriormente en la aplicación.
- b. Actor Iniciador: Usuario Identificado o Administrador.
- c. Precondición: El usuario existe en el sistema y está identificado.
- d. Postcondición: Los datos personales del usuario son modificados con los nuevos valores.
- e. Datos de entrada: Nuevos datos en cualquiera de los apartados del [requisito 11](#) incluyendo cambio de contraseña.
- f. Datos de salida: Confirmar si tuvo éxito o mostrar mensaje de error si no lo hay.

g. Eventos:

- i. El actor selecciona la opción de modificación una vez consulta sus datos personales actuales.
- ii. El usuario introduce nuevos valores en el formulario de datos personales.
- iii. En caso de cambiar el email, el sistema comprueba que no hayan usuarios duplicados manteniendo el email único. Si se cambia la contraseña, esta se cifra. Se almacena la información en la base de datos modificando los valores correspondientes.
- iv. El sistema notifica al usuario del éxito o error de la operación.

X Visualizar preguntas/respuestas Cuestionarios

- a. Propósito: Consulta de las preguntas de un cuestionario (pre-cuestionario o post-cuestionario) o en caso de haberse respondido anteriormente, las respuestas dadas al cuestionario y su resultado.
- b. Actor Iniciador: Usuario Identificado o Administrador.
- c. Precondición: El usuario existe en el sistema y está identificado. Si el usuario no ha respondido el cuestionario sólo se podrán visualizar las preguntas para proceder a responderlas. Si el usuario ha respondido el cuestionario anteriormente se mostrarán las respuestas dadas a cada pregunta y el resultado obtenido.
- d. Postcondición: -
- e. Datos de entrada: -
- f. Datos de salida: Las preguntas del cuestionario o las respuestas dadas a cada pregunta incluyendo el resultado obtenido.
- g. Eventos:
 - i. El actor selecciona esta “página” en el menú de navegación.

- ii. El sistema muestra las preguntas del cuestionario o las respuestas dadas a cada pregunta incluyendo el resultado obtenido.

XI Responder Cuestionarios

- a. Propósito: Responder a las preguntas de un cuestionario (pre-cuestionario o post-cuestionario).
- b. Actor Iniciador: Usuario Identificado o Administrador.
- c. Precondición: El usuario existe en el sistema y está identificado. Si el usuario ha respondido el cuestionario sólo se podrán visualizar las respuestas guardadas y no se ofrecerá opción de modificar estas respuestas.
- d. Postcondición: Las respuestas a las preguntas del cuestionario se guardarán en la base de datos.
- e. Datos de entrada: Opción seleccionada en una escala de 1 a 5 para cada pregunta del cuestionario.
- f. Datos de salida: Confirmar si tuvo éxito o mostrar mensaje de error si no lo hay.
- g. Eventos:
 - i. El actor selecciona esta “página” en el menú de navegación.
 - ii. El sistema muestra las preguntas del cuestionario sin responder.
 - iii. El usuario responde todas las preguntas.
 - iv. El sistema transforma las respuestas en una escala de números del 1 al 5, infiere un resultado a través de la media de los valores, y guarda todo en la base de datos.
 - v. El sistema notifica al usuario del éxito o error de la operación.

XII Responder MOOC (no se incluye la descripción en este TFG por ser una funcionalidad que debe desarrollar otro estudiante).

XIII Visualizar MOOC (no se incluye la descripción en este TFG por ser una funcionalidad que debe desarrollar otro estudiante).

XIV Visualizar Dashboard MOOC (no se incluye la descripción en este TFG por ser una funcionalidad que debe desarrollar otro estudiante).

XV Visualizar Dashboard Cuestionarios

- a. Propósito: Consultar las estadísticas resultantes de todas las respuestas dadas por los usuarios a los cuestionarios.
- b. Actor Iniciador: Administrador.
- c. Precondición: El usuario existe en el sistema y es administrador.
- d. Postcondición: -
- e. Datos de entrada: -
- f. Datos de salida: Estadísticas de cantidad de usuarios que han respondido cada respuesta a cada pregunta de cada cuestionario tanto de forma general, como por edad, nacionalidad, lengua materna y lenguaje incluyendo nivel de competencia. Cantidad de usuarios que han contenido cada resultado de cada cuestionario.
- g. Eventos:
 - i. El usuario administrador selecciona esta “página” en el menú de navegación.
 - ii. El sistema muestra las estadísticas.

XVI Gestionar Cuestionarios

- a. Propósito: Crear, modificar o eliminar un cuestionario.
- b. Actor Iniciador: Administrador.
- c. Precondición: El usuario existe en el sistema y es administrador.
- d. Postcondición: Se añade un nuevo cuestionario, se modifica o se elimina uno existente.
- e. Datos de entrada: Si es un nuevo cuestionario se debe introducir una pregunta general, cada pregunta con sus posibles respuestas, posibles resultados incluyendo un título, descripción

- y rango de valores a los que pertenece dicho resultado. Datos a modificar en caso de que sea modificar cuestionario.
- f. Datos de salida: Confirmar si tuvo éxito o mostrar mensaje de error si no lo hay.
- g. Eventos:
- i. El usuario administrador selecciona esta “página” en el menú de navegación.
 - ii. El usuario selecciona crear nuevo cuestionario o uno existente para modificarlo o eliminarlo. En los dos primeros casos introduce o modifica los datos pertinentes.
 - iii. El sistema guarda en la base de datos el nuevo cuestionario o modifica o elimina uno existente.
 - iv. El sistema notifica al usuario del éxito o error de la operación.

5.3. Arquitectura

Una vez establecidos los tipos de usuarios y las funcionalidades que pueden realizar en el sistema, se puede referenciar la arquitectura tradicional de cualquier software web, es decir, el modelo cliente/servidor.

5.3.1. Cliente/Servidor

La arquitectura cliente-servidor es un modelo de diseño de software en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, quien le da respuesta. La separación es una de tipo lógico, donde el servidor no se ejecuta necesariamente sobre una sola máquina ni es necesariamente un solo programa. Los tipos específicos de servidores incluyen los servidores web, los servidores de archivo, los servidores del correo, etc. Mientras que sus propósitos varían de unos servicios a otros, la arquitectura básica seguirá siendo la misma [\[29\]](#).

De Stance, como se menciona en anteriores secciones de este documento (véase [sección 4.5](#) por ejemplo) seguirá la arquitectura MEAN que consiste en

un servidor web de Angular en forma de aplicación de una sola página. Una single-page application (SPA), o aplicación de página única, es una aplicación web o sitio web que cabe en una sola página con el propósito de dar una experiencia más fluida a los usuarios, como si fuera una aplicación de escritorio [\[30\]](#). El servidor web SPA de DeStance despliega la interfaz de usuario y provee la navegación entre las distintas funcionalidades de la aplicación.

En un SPA todos los códigos de HTML, JavaScript, y CSS se cargan una sola vez, los recursos necesarios se cargan dinámicamente cuando lo requiera la página, normalmente como respuesta a las acciones del usuario. La página no tiene que cargarse de nuevo en ningún punto del proceso y tampoco es necesario transferir a otra página las peticiones [\[30\]](#). En el caso de DeStance el servidor web SPA de Angular se comporta como cliente cuando realiza peticiones al servidor API REST de Node.js y Express.js en los casos en los que determinada funcionalidad requiere de la base de datos o algún cálculo complejo.

Una API REST es una interfaz de comunicación entre sistemas de información que usa el protocolo de transferencia de hipertexto (hypertext transfer protocol o HTTP, por su siglas en inglés) para obtener datos o ejecutar operaciones sobre dichos datos en diversos formatos, como pueden ser XML o JSON [\[31\]](#). El servidor API de DeStance actúa como servidor cliente cuando realiza peticiones al servidor de MongoDB Atlas (base de datos). En la siguiente figura se muestra cómo la arquitectura cliente/servidor se aplica a los componentes de DeStance mencionados anteriormente:

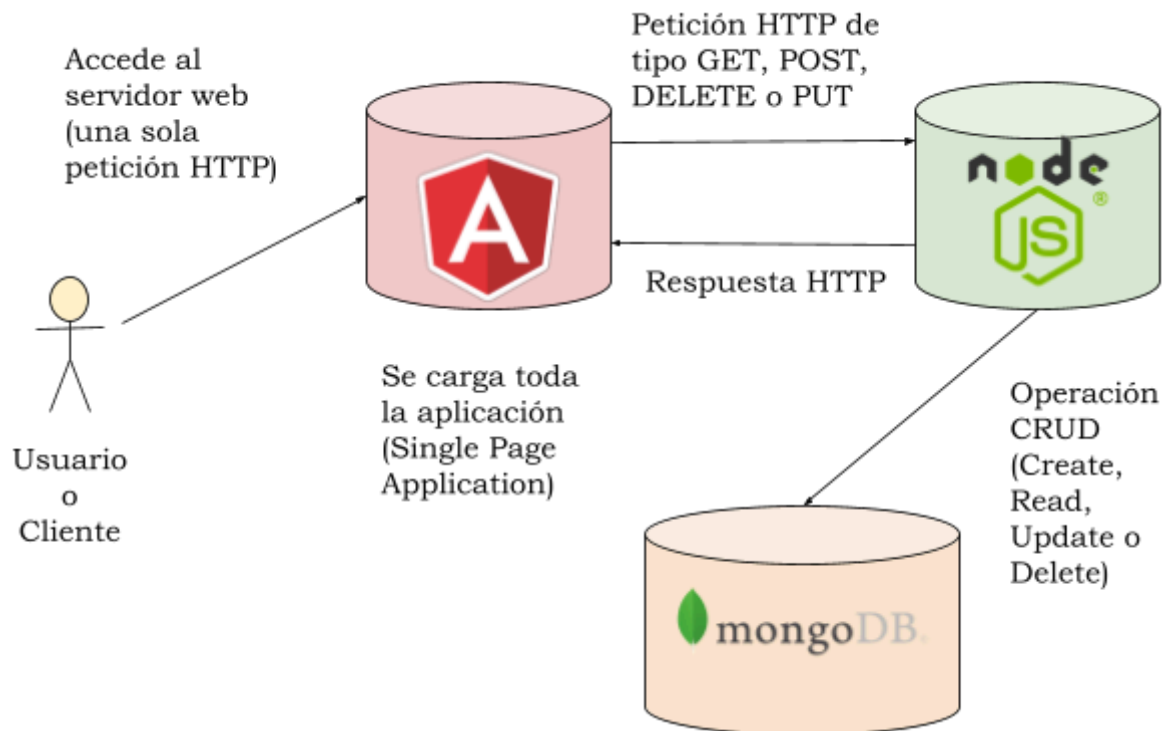


Figura 10: Arquitectura cliente/servidor aplicada a DeStance.

5.3.2. Base de Datos

DeStance usa el modelo no relacional (noSQL) como tipo de base de datos. Una de las características más importantes de las bases de datos no relacionales, como MongoDB, es su modelo de datos flexible. MongoDB almacena datos en documentos BSON (una extensión binaria de JSON), lo que permite una representación de datos más natural y cercana a los objetos en los lenguajes de programación modernos. Además, no requiere un esquema predefinido, lo que permite almacenar documentos con diferentes estructuras en la misma colección, facilitando así la evolución y flexibilidad de los datos [\[13\]](#).

En términos de consultas y procesamiento de datos, las bases de datos NoSQL proporcionan capacidades potentes y flexibles. Permiten realizar búsquedas y filtrados complejos, y muchas de ellas ofrecen frameworks de agregación que facilitan la creación de consultas avanzadas. Estas capacidades son esenciales para aplicaciones que requieren un análisis de datos robusto y eficiente. Las

bases de datos NoSQL están diseñadas para escalar horizontalmente, a través de la distribución de datos en múltiples servidores, lo que mejora la capacidad de manejar grandes volúmenes de datos y aumenta la performance general del sistema. La escalabilidad horizontal es esencial para aplicaciones que requieren un crecimiento rápido y deben manejar cargas de trabajo intensivas [\[32\]](#).

Por otro lado, las bases de datos relacionales tienen un modelo de datos estructurado. En estas bases de datos, los datos se organizan en tablas con filas y columnas, y cada tabla tiene un esquema predefinido que describe la estructura de los datos que contiene. Esto garantiza la consistencia e integridad de los datos, ya que cada entrada debe cumplir con el esquema definido. Las relaciones entre tablas se establecen mediante claves primarias y foráneas, lo que permite una organización eficiente y coherente de los datos [\[33\]](#).

Otra característica clave de las bases de datos relacionales es su capacidad para realizar consultas complejas. Utilizando el lenguaje de consulta estructurado (SQL), los usuarios pueden ejecutar una amplia variedad de operaciones sobre los datos, incluyendo selección, inserción, actualización y eliminación de registros. Además, SQL permite la creación de consultas avanzadas que pueden combinar datos de múltiples tablas mediante operaciones como JOIN, lo que facilita el análisis y la manipulación de los datos [\[33\]](#).

A pesar de que en las bases de datos NoSQL no se requiere un esquema predefinido o relaciones entre los datos, esto no quiere decir que no lo pueda tener. En el caso de DeStance, se aprovecha la flexibilidad para determinadas colecciones como los conceptos y se proveen relaciones entre otras colecciones más complejas a través de un ID, que actúa como clave primaria y foránea. Para lograr esto se utiliza la librería de MongoDB: Mongoose.

Mongoose es una biblioteca de JavaScript que proporciona una solución elegante y poderosa para trabajar con MongoDB desde aplicaciones Node.js. Actúa como un ODM (Object Data Modeling), facilitando la gestión de datos

mediante la definición de esquemas que mapean documentos de MongoDB a modelos de la aplicación. Esto permite trabajar con datos de manera estructurada y aplicar validaciones, ganchos (hooks) y otras funcionalidades adicionales de manera más sencilla [\[34\]](#).

Como las colecciones en la base de datos de DeStance siguen un esquema y algunas tienen relación entre sí, se representará en la [Figura 11](#) a través de un Modelo Entidad-Relación (MER). Este último está formado por tres elementos básicos:

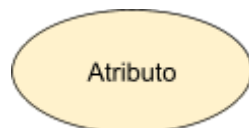
- **Entidad:** Representan los distintos modelos que se almacenan y corresponden con el nombre de la tabla (colección en este caso). En el diagrama están representados por su nombre dentro de un cuadrado [\[35\]](#).



- **Relación:** Representan las conexiones entre las distintas entidades y los conjuntos que están relacionados por cada entidad. En el diagrama están representados dentro de un romboide [\[35\]](#).



- **Atributo:** Representan las características asociadas a cada entidad, corresponden con cada propiedad de una colección. En el diagrama están representados por su nombre dentro de un óvalo [\[35\]](#).



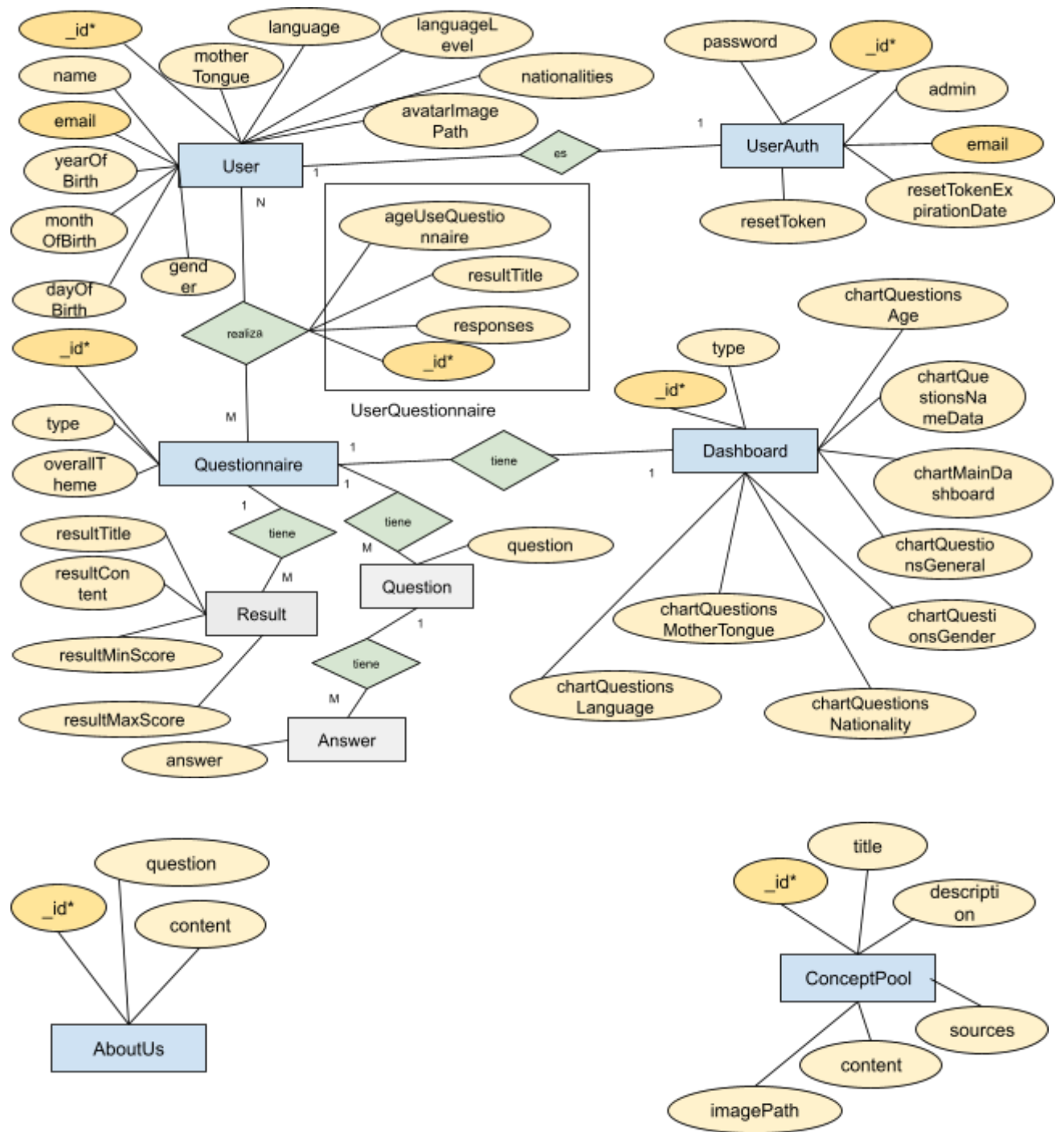


Figura 11: Diagrama Entidad/Relación de DeStance.

Como se puede observar en el diagrama anterior, existen un total de seis entidades que serán las colecciones en la base de datos de DeStance (solamente se modelan las funcionalidades a desarrollar por la autora de este

TFG). Estas serían: User, UserAuth, Dashboard, Questionnaire, AboutUs y ConceptPool. Adicionalmente, la colección resultante de la relación (mucho a mucho) entre User y Questionnaire se denominará UserQuestionnaire. En cuanto a Questionnaire, las “entidades” de color gris relacionadas con este: Results, Question y Answer se modelarán como atributos en forma de objetos BSON dentro de la propia colección Questionnaire utilizando el plural un minúsculas como el nombre, aprovechando con ello la flexibilidad de NoSQL.

A continuación se proveerá una descripción más detallada de cada entidad (colección) y sus atributos, así como capturas ejemplo de los datos reales o de prueba en la base de datos:

- **User:** Contiene los datos personales de los usuarios de la aplicación.
 - `_id`: Identificador del documento (Moongose genera este campo automáticamente).
 - `name`: Nombre de usuario del usuario (no tiene por que ser su nombre real).
 - `email`: Correo electrónico del usuario (debe ser único en la aplicación por lo que consiste en un identificador más funcional).
 - `yearOfBirth`: Año de nacimiento.
 - `monthOfBirth`: Mes de nacimiento.
 - `dayOfBirth`: Día de nacimiento.
 - `gender`: Género del usuario (masculino, femenino u otro).
 - `nationalities`: Nacionalidad(es) del usuario.
 - `motherTongue`: Lengua materna del usuario.
 - `language`: Lenguaje en el que el usuario va a responder el MOOC.
 - `languageLevel`: Nivel de competencia del lenguaje en el que el usuario va a responder el MOOC.
 - `avatarImagePath`: Dirección URL de la imagen de avatar del usuario.

```

_id: ObjectId('663f7982151a06b71b9436aa')
name: "User 2"
email: "user2@email.com"
yearOfBirth: "1988"
monthOfBirth: "06"
dayOfBirth: "25"
gender: "Male"
nationalities: Array (1)
  0: "United Kingdom"
motherTongue: "English"
language: "English"
languageLevel: "C2"
avatarImagePath: "https://firebasestorage.googleapis.com/v0/b/rascimaff.appspot.com/o/a..."

```

Figura 12.1: Ejemplo datos de prueba de entidad User (colección users).

- **UserAuth:** Utilizada para separar los datos de autenticación de los datos personales del usuario.
 - `_id`: Identificador del documento (Moongose genera este campo automáticamente).
 - `email`: Correo electrónico del usuario (debe ser único en la aplicación por lo que relaciona cada documento de UserAuth con el correspondiente documento en User).
 - `password`: Contraseña cifrada del usuario.
 - `admin`: “flag” que verifica si el usuario es administrador o no.
 - `resetToken`: Código de recuperación de contraseña. Se actualiza cada vez que se solicita una nueva contraseña y es válido por máximo una hora.
 - `resetTokenExpirationDate`: Es la fecha y hora máxima hasta la cual el código en `resetToken` es válido.

```

_id: ObjectId('662bcd960c066cb68c2cfdb0')
email: "grettell.umpierrez@alumnos.upm.es"
password: "$2a$10$wCPUiMUbXJGt2tbvKapSP.Fs5g37j7IsLls/uGmcqQQ2dPIMvgjAm"
admin: true
__v: 0
resetToken: "929fd203343c2a59c52c43100393774929a602fa60031f0eeb83e69636b18b5e"
resetTokenExpirationDate: 2024-05-18T12:27:00.169+00:00

```

Figura 12.2: Ejemplo datos reales de entidad UserAuth (colección userauths).

- **Questionnaire:** Es la colección que representa los cuestionarios en la base de datos.

- `_id`: Identificador del documento (Moongose genera este campo automáticamente).
- `overallTheme`: Título general del cuestionario.
- `type`: Tipo de cuestionario: pre-cuestionario o post-cuestionario.
- `questions`: Simula un objeto JSON (BSON) en forma de conjunto de preguntas.
 - `question`: Título de la pregunta.
 - `answers`: Simula un objeto JSON (BSON) en forma de conjunto de 5 etiquetas de respuestas.
- `results`: Simula un objeto JSON (BSON) en forma de conjunto de resultados.
 - `resultTitle`: Título del resultado
 - `resultContent`: Contenido descriptivo del resultado.
 - `resultMinScore`: Valor mínimo que debe tener la media de respuestas para que sea este el resultado.
 - `resultMaxScore`: Valor máximo que debe tener la media de respuestas para que sea este el resultado.

```

_id: ObjectId('663f79a2151a06b71b943719')
overallTheme: "Ready to put your intercultural skills to the test?"
questions: Array (5)
  0: Object
    question: "Question 1"
    answers: Array (5)
      0: "Answer 1"
      1: "Answer 2"
      2: "Answer 3"
      3: "Answer 4"
      4: "Answer 5"
    _id: ObjectId('663f79a2151a06b71b94371a')
  1: Object
  2: Object
  3: Object
  4: Object
results: Array (3)
  0: Object
    resultTitle: "Result 1"
    resultContent: "Content R1"
    resultMinScore: 1
    resultMaxScore: 2
    _id: ObjectId('663f79a2151a06b71b94371f')
  1: Object
  2: Object
type: "pre-q"

```

Figura 12.3: Ejemplo datos de prueba de entidad Questionnaire (colección questionnaires).

- **UserQuestionnaire:** Colección resultado de la relación entre User y Questionnaire donde cada usuario puede responder muchos cuestionarios y un cuestionario puede ser respondido por muchos usuarios. Permite actualizar las estadísticas del Dashboard de cuestionarios de forma eficiente, así como mantener las respuestas en la base de datos.
 - `_id`: Identificador del documento (Moongose genera este campo automáticamente).
 - `idUser`: identificador del usuario que realiza el cuestionario.
 - `idQuestionnaire`: identificador del cuestionario al que pertenece la respuesta.
 - `type`: Tipo de cuestionario: pre-cuestionario o post-cuestionario.
 - `response`: Respuesta del usuario a cada pregunta del cuestionario. Cada respuesta consiste en un número del 1 al 5.
 - `resultTitle`: Título del resultado obtenido.
 - `ageUserDayOfQuestionnaire`: Edad que tiene el usuario en el momento que responde el cuestionario.

```
_id: ObjectId('663f79b8151a06b71b9437cb')
idUser: "0"
idQuestionnaire: "663f79a2151a06b71b943719"
type: "pre-q"
▼ responses: Array (5)
  0: 5
  1: 3
  2: 1
  3: 3
  4: 4
resultTitle: "Result 2"
ageUserDayOfQuestionnaire: "27"
```

Figura 12.4: Ejemplo datos de prueba de entidad UserQuestionnaire (colección userquestionnaires).

- **Dashboard:** Consiste en una colección que contiene el conteo de respuestas para cada tipo de gráfico y se actualiza cada que un usuario responde un cuestionario. De esta forma se gana en eficiencia porque no sería necesario recomtar todo cada vez que se ejecute una petición.

- `_id`: Identificador del documento (Moongose genera este campo automáticamente)
- `idQuestionnaire`: identificador del cuestionario al que pertenece el dashboard.
- `type`: Tipo de cuestionario, pre-cuestionario o post-cuestionario.
- `chartMainDashboard`: Simula un objeto JSON (BSON) y contiene los datos necesarios para construir el gráfico de cantidad de usuarios por cada posible resultado del cuestionario.
 - `data`: Conteo de cantidad de usuarios que han obtenido cada resultado del cuestionario (si hubiese cuatro posibles resultados este campo sería un conjunto de cuatro números que corresponden al conteo por resultado).
 - `nameData`: Título de cada resultado.
- `chartQuestionsNameData`: Simula un objeto JSON (BSON) y contiene las etiquetas de las respuestas a cada pregunta.
- `chartQuestionsGeneral`: Simula un objeto JSON (BSON) y contiene los datos necesarios para construir el gráfico de cantidad de usuarios por cada respuesta de cada pregunta del cuestionario.
 - `data`: Conjunto del conteo de la cantidad de usuarios que han respondido cada respuesta para cada pregunta.
- `chartQuestionsAge`: Simula un objeto JSON (BSON) y contiene los datos necesarios para construir el gráfico de cantidad de usuarios de cada categoría de edad (<18, 18-29, 30-49, 50-79, >80) que han respondido cada respuesta por cada pregunta del cuestionario.
 - `data`: Conjunto del conteo de cantidad de usuarios de cada categoría de edad (<18, 18-29, 30-49, 50-79, >80) que han respondido cada respuesta por cada pregunta del cuestionario.
- `chartQuestionsGender`: Simula un objeto JSON (BSON) y contiene los datos necesarios para construir el gráfico de cantidad de usuarios de cada género (masculino, femenino, otro)

que han respondido cada respuesta por cada pregunta del cuestionario.

- data: Conjunto del conteo de cantidad de usuarios de cada género (masculino, femenino, otro) que han respondido cada respuesta por cada pregunta del cuestionario.
- chartQuestionsNationality: Simula un objeto JSON (BSON) y contiene los datos necesarios para construir el gráfico de cantidad de usuarios de nacionalidad que han respondido cada respuesta por cada pregunta del cuestionario.
 - data: Conjunto del conteo de cantidad de usuarios de cada nacionalidad que han respondido cada respuesta por cada pregunta del cuestionario.
 - captionData: Conjunto de todas las nacionalidades de los usuarios por cada pregunta del cuestionario.
- chartQuestionsNationality: Simula un objeto JSON (BSON) y contiene los datos necesarios para construir el gráfico de cantidad de usuarios de nacionalidad que han respondido cada respuesta por cada pregunta del cuestionario.
 - data: Conjunto del conteo de cantidad de usuarios de cada nacionalidad que han respondido cada respuesta por cada pregunta del cuestionario.
 - captionData: Conjunto de todas las nacionalidades de los usuarios por cada pregunta del cuestionario.
- chartQuestionsMotherTongue: Simula un objeto JSON (BSON) y contiene los datos necesarios para construir el gráfico de cantidad de usuarios de cada lengua materna que han respondido cada respuesta por cada pregunta del cuestionario.
 - data: Conjunto del conteo de cantidad de usuarios de cada lengua materna que han respondido cada respuesta por cada pregunta del cuestionario.
 - captionData: Conjunto de todas las lenguas maternas de los usuarios por cada pregunta del cuestionario.

- chartQuestionsLanguage: Simula un objeto JSON (BSON) y contiene los datos necesarios para construir el gráfico de cantidad de usuarios de cada lenguaje y nivel de competencia que han respondido cada respuesta por cada pregunta del cuestionario.
 - data: Conjunto del conteo de cantidad de usuarios de cada lenguaje y nivel de competencia que han respondido cada respuesta por cada pregunta del cuestionario.
 - captionData: Conjunto de todos los lenguajes y nivel de competencia de los usuarios por cada pregunta del cuestionario.
 - groupData: Conjunto de todos los niveles de competencia de los usuarios por cada pregunta del cuestionario.

```

_id: ObjectId('663f79a2151a06b71b943755')
idQuestionnaire: "663f79a2151a06b71b943719"
type: "pre-q"
▼ chartMainDashboard: Object
  ▼ data: Array (3)
    0: 6
    1: 16
    2: 9
  ▼ nameData: Array (3)
    0: "Result 1"
    1: "Result 2"
    2: "Result 3"
  ▼ chartQuestionsNameData: Array (5)
    ▼ 0: Array (5)
      0: "Answer 1"
      1: "Answer 2"
      2: "Answer 3"
      3: "Answer 4"
      4: "Answer 5"
    ▶ 1: Array (5)
    ▶ 2: Array (5)
    ▶ 3: Array (5)
    ▶ 4: Array (5)
  ▶ chartQuestionsGeneral: Object
  ▶ chartQuestionsAge: Object
  ▼ chartQuestionsGeneral: Object
    ▼ data: Array (5)
      ▶ 0: Array (5)
      ▶ 1: Array (5)
      ▶ 2: Array (5)
      ▶ 3: Array (5)
      ▼ 4: Array (5)
        0: 7
        1: 5
        2: 5
        3: 7
        4: 7
    ▼ chartQuestionsAge: Object
      ▼ data: Array (5)
        ▼ 0: Array (5)
          ▶ 0: Array (5)
          ▼ 1: Array (5)
            0: 5
            1: 2
            2: 4
            3: 4
            4: 4
          ▶ 2: Array (5)
          ▶ 3: Array (5)
          ▶ 4: Array (5)
        ▶ 1: Array (5)

```

Figura 12.5.1: Ejemplo datos de prueba de entidad Dashboard (colección dashboards).

```
▼ chartQuestionsGender : Object
  ▼ data : Array (5)
    ▼ 0: Array (3)
      ▼ 0: Array (5)
        0: 3
        1: 1
        2: 2
        3: 5
        4: 2
      ▶ 1: Array (5)
      ▶ 2: Array (5)
    ▶ 1: Array (3)
    ▶ 2: Array (3)
    ▶ 3: Array (3)
    ▶ 4: Array (3)
  ▼ chartQuestionsNationality : Object
    ▼ data : Array (9)
      ▼ 0: Array (5)
        ▼ 0: Array (5)
          0: 1
          1: 0
          2: 1
          3: 2
          4: 0
        ▶ 1: Array (5)
        ▶ 2: Array (5)
      ▼ 1: Array (5)
        0: 0
        1: 0
        2: 0
        3: 0
        4: 0
      ▼ 2: Array (5)
        0: 0
        1: 0
        2: 0
        3: 0
        4: 0
      ▼ 3: Array (5)
        0: 0
        1: 0
        2: 0
        3: 0
        4: 0
      ▼ 4: Array (5)
        0: 0
        1: 0
        2: 0
        3: 0
        4: 0
      ▼ 5: Array (5)
        0: 0
        1: 0
        2: 0
        3: 0
        4: 0
      ▼ 6: Array (5)
        0: 0
        1: 0
        2: 0
        3: 0
        4: 0
      ▼ 7: Array (5)
        0: 0
        1: 0
        2: 0
        3: 0
        4: 0
      ▼ 8: Array (5)
        0: 0
        1: 0
        2: 0
        3: 0
        4: 0
    ▼ chartQuestionsMotherTongue : Object
      ▼ data : Array (6)
        ▼ 0: Array (5)
          ▼ 0: Array (5)
            0: 3
            1: 1
            2: 1
            3: 4
            4: 1
          ▼ 1: Array (5)
            0: 0
            1: 0
            2: 0
            3: 0
            4: 0
          ▼ 2: Array (5)
            0: 0
            1: 0
            2: 0
            3: 0
            4: 0
          ▼ 3: Array (5)
            0: 0
            1: 0
            2: 0
            3: 0
            4: 0
          ▼ 4: Array (5)
            0: 0
            1: 0
            2: 0
            3: 0
            4: 0
          ▼ 5: Array (5)
            0: 0
            1: 0
            2: 0
            3: 0
            4: 0
        ▼ 1: Array (5)
          0: 0
          1: 0
          2: 0
          3: 0
          4: 0
        ▼ 2: Array (5)
          0: 0
          1: 0
          2: 0
          3: 0
          4: 0
        ▼ 3: Array (5)
          0: 0
          1: 0
          2: 0
          3: 0
          4: 0
        ▼ 4: Array (5)
          0: 0
          1: 0
          2: 0
          3: 0
          4: 0
        ▼ 5: Array (5)
          0: 0
          1: 0
          2: 0
          3: 0
          4: 0
```

Figura 12.5.2: Ejemplo datos de prueba de entidad Dashboard (colección dashboards).

```

▼ chartQuestionsMotherTongue : Object
  ▶ data : Array (6)
  ▼ captionData : Array (5)
    ▼ 0: Array (6)
      0: "Spanish"
      1: "English"
      2: "Polish"
      3: "German"
      4: "French"
      5: "Italian"
    ▶ 1: Array (6)
    ▶ 2: Array (6)
    ▶ 3: Array (6)
    ▶ 4: Array (6)
▼ chartQuestionsLanguage : Object
  ▼ data : Array (7)
    ▼ 0: Array (5)
      ▼ 0: Array (5)
        0: 1
        1: 0
        2: 0
        3: 2
        4: 0
      1: "C2 Spanish"
      2: "C2 English"
      3: "C1 English"
      4: "C2 Polish"
      5: "C2 German"
      6: "C2 French"
    ▶ 1: Array (7)
    ▶ 2: Array (7)
    ▶ 3: Array (7)
    ▶ 4: Array (7)
  ▼ groupData : Array (5)
    ▼ 0: Array (7)
      0: "C2"
      1: "C2"
      2: "C1"
      3: "C2"
      4: "C2"
      5: "C2"
      6: "B2"

```

Figura 12.5.3: Ejemplo datos de prueba de entidad Dashboard (colección dashboards).

- **ConceptPool:** Consiste en una colección que contiene los conceptos más importantes de DeStance.
 - `_id`: Identificador del documento (Moongose genera este campo automáticamente).
 - `title`: Título del concepto.
 - `description`: Descripción resumen del concepto.
 - `content`: Contenido del concepto.
 - `imagePath`: URL de la imagen que representa el concepto.
 - `sources`: Referencia bibliográfica del contenido del concepto.

```
_id: ObjectId('6634bf3ac15cd9d9ff108b2a')
title: "Empathy"
description: "Understanding and sharing the feelings of others"
imagePath: "https://firebasestorage.googleapis.com/v0/b/rascimaff.app...
content: "Empathy is the set of skills required to understand and
        relate to othe..."
sources: "Competences for Democratic Culture: Living together as equal
        in cultur..."
```

Figura 12.6: Ejemplo datos reales de entidad ConceptPool (colección conceptpools).

- **AboutUs:** Consiste en una colección que contiene las respuestas a preguntas frecuentes que pudiesen tener los usuarios de DeStance.
 - `_id`: Identificador del documento (Moongose genera este campo automáticamente).
 - `question`: Pregunta frecuente.
 - `content`: Respuesta a la pregunta.

```
_id: ObjectId('6634bd9cf46f089e4d13bbc0')
question: "Question 1"
content: "Lorem ipsum dolor sit amet, consectetur adipiscing elit.
        Adipiscing co..."
```

Figura 12.7: Ejemplo datos de prueba de entidad AboutUs (colección aboutus).

5.4. Diseño del front-end

5.4.1. Componentes del front-end

Angular es un framework que facilita la creación de aplicaciones web dinámicas mediante una estructura robusta y clara, dividiendo la

funcionalidad en componentes, gestionando la interacción del usuario y manejando el enrutamiento de manera eficiente. Los componentes son las piezas fundamentales, compuestos por un archivo TypeScript que define su lógica, una plantilla HTML para su estructura visual y un archivo de estilos para su apariencia. La interacción del usuario se maneja mediante eventos y enlazado de datos (data binding) [\[19\]](#).

El enrutamiento se gestiona con Angular Router, que configura las rutas en un archivo de módulo y define las rutas en una matriz de objetos *Routes*, especificando el path y el componente asociado. En la plantilla principal, se usa `<router-outlet></router-outlet>` para mostrar los componentes según la ruta activa, y se puede navegar programáticamente usando *Router* en la lógica del componente. Además, Angular proporciona hooks del ciclo de vida como *ngOnInit*, *ngOnChanges* y *ngOnDestroy* para ejecutar lógica en diferentes momentos del ciclo de vida de un componente [\[19\]](#).

Los servicios en Angular son una manera de encapsular y compartir datos o lógica entre diferentes componentes. Estos servicios son generalmente clases que se pueden inyectar en los componentes usando la inyección de dependencias. Esto permite que múltiples componentes accedan a la misma instancia del servicio, facilitando la gestión de estados y la comunicación entre componentes. Los servicios son ideales para manejar operaciones como la obtención de datos de una API, almacenamiento en caché de datos y cualquier otra lógica de negocio que necesite ser reutilizada [\[19\]](#).

El modelo, en el contexto de Angular, se refiere a las estructuras de datos que representan la información de la aplicación. Los modelos son generalmente clases TypeScript que definen la forma de los datos y se utilizan para tipar los datos que se manejan en la aplicación, garantizando así la coherencia y facilitando el manejo de los datos [\[19\]](#). Conectan, por tanto, los modelos de la base de datos con los componentes del front-end.

A continuación se ofrece el diseño general de los componentes, servicios y modelos que forman parte del servidor web de front-end de DeStance.

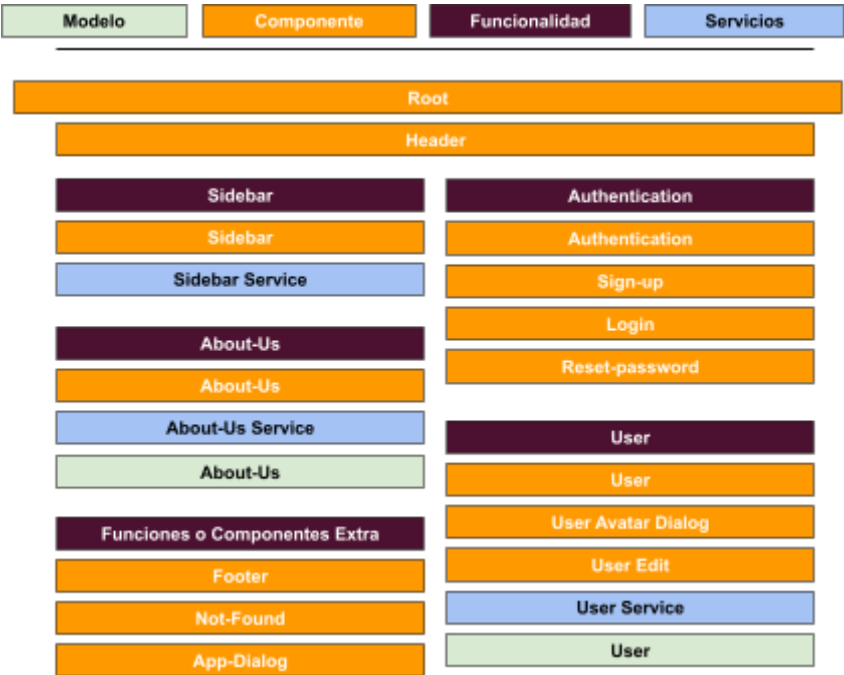


Figura 13.1: Diseño de los componentes del front-end de DeStance.

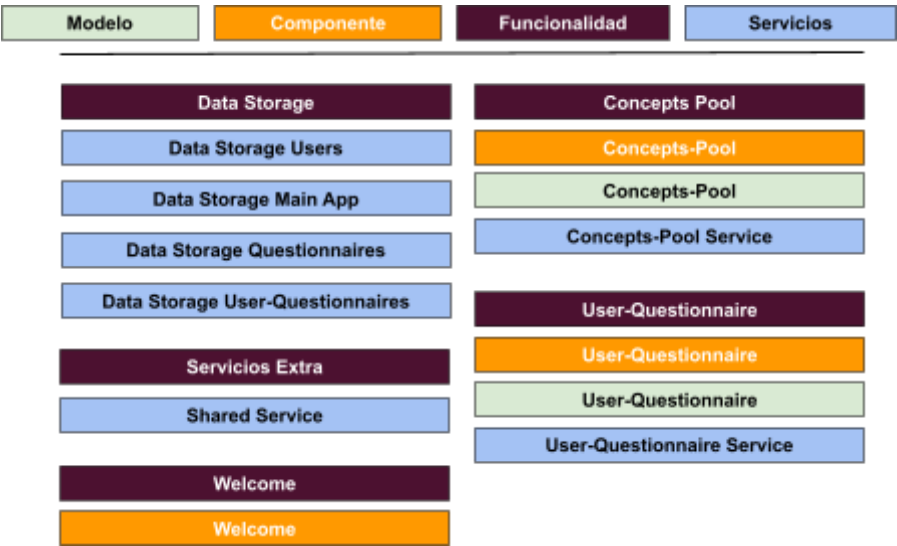


Figura 13.2: Diseño de los componentes del front-end de DeStance.

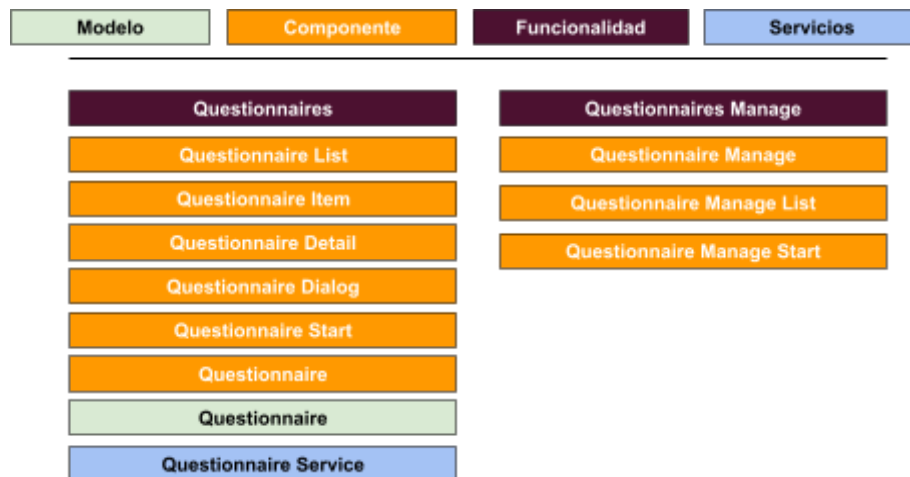


Figura 13.3: Diseño de los componentes del front-end de DeStance.

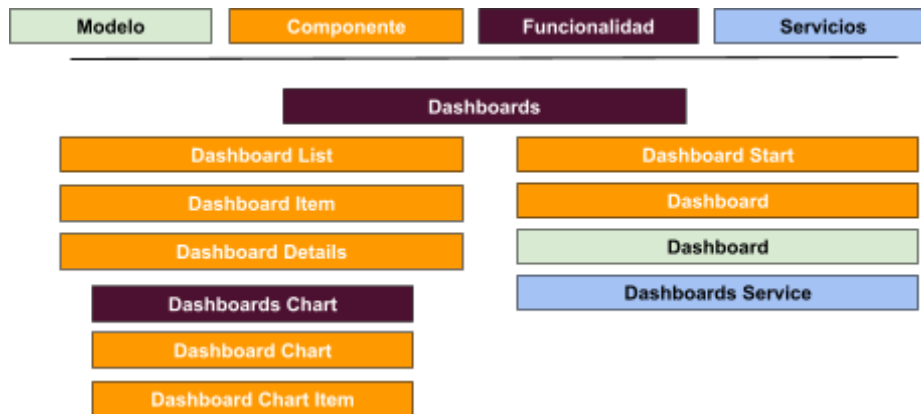


Figura 13.4: Diseño de los componentes del front-end de DeStance.

DeStance cuenta, como se puede observar desde la [Figura 13.1](#) hasta la [Figura 13.4](#), con seis modelos de datos, doce servicios y treinta y tres componentes que se integran para conformar el front-end de la aplicación.

5.4.2. Routing del front-end

A continuación se muestran las rutas internas del servidor web de front-end:

Ruta	Componente
'/welcome'	Welcome
'/about-us'	About-Us
'/concepts-pool'	Concepts-Pool
'/user-details'	User

‘/user-details/edit’			User + User-Edit
‘/auth’			Authentication
‘/auth/login’			Login
‘/auth/sign-up’			Sign-Up
‘/auth/reset-password’			Reset-Password
‘/questionnaires/:type’			Questionnaires + QuestionnairesStart
Parámetros	Explicación	Valores	
type	Tipo de cuestionario	pre-q	
		post-q	
‘/questionnaires/:type/:id’			Questionnaires + Questionnaires-Detail
Parámetros	Explicación	Valores	
id	Identificador del cuestionario	Valores numéricos (0,1...)	
‘/manage/:type’			
Parámetros	Explicación	Valores	Questionnaires Manage + Questionnaires Manage Start
type	Tipo de cuestionario	pre-q	
		post-q	
‘/manage/:type/new’			
‘/manage/:type/:id/edit’			Questionnaires Manage + Questionnaires Edit
Parámetros	Explicación	Valores	
id	Identificador del cuestionario	Valores numéricos (0,1...)	

‘/not-found’ (Todas las rutas que el usuario introduzca directamente que no estén definidas en esta tabla serán redirigidas a esta)	Not-Found
‘/mooc’	ESTA FUNCIONALIDAD PERTENECE AL TFG DE OTRO ESTUDIANTE
‘/dashboard-mooc’	ESTA FUNCIONALIDAD PERTENECE AL TFG DE OTRO ESTUDIANTE

Tabla 1: Rutas front-end de DeStance.

Como complemento a la [Tabla 1](#), en la siguiente figura se muestra el mapa de navegación de la aplicación:

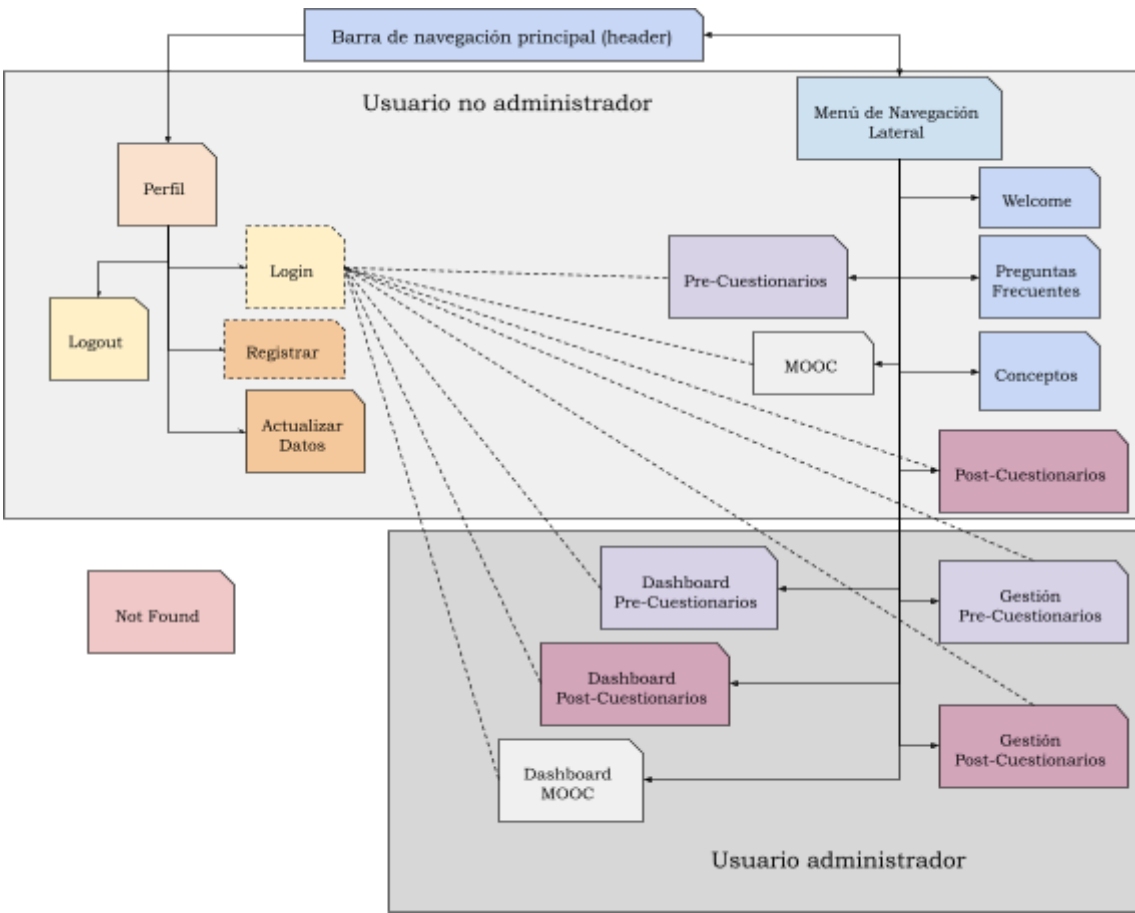


Figura 14: Mapa de navegación del front-end de DeStance.

La navegación de DeStance expuesto en la [Figura 14](#) y la [Tabla 1](#) está diseñada para proporcionar una experiencia de usuario organizada y accesible, tanto para usuarios no administradores como para administradores. A continuación, se describe la estructura y funcionalidad del menú de navegación según se muestra en el diagrama:

Barra de Navegación Principal (Header)

La barra de navegación principal se encuentra en la parte superior de la interfaz y está disponible para todos los usuarios. Incluye enlaces o botones que permiten el acceso rápido a secciones importantes de la aplicación, tales como el perfil del usuario, la opción de cerrar sesión (logout), y los formularios de iniciar sesión (login), registro y actualización de datos.

Usuario No Administrador

Los usuarios no administradores tienen acceso a las siguientes secciones a través del menú de navegación lateral:

- Perfil: Acceso a la página de perfil del usuario donde pueden ver y editar su información personal.
- Logout: Opción para cerrar sesión en la aplicación.
- Login: Formulario para iniciar sesión en la aplicación. Este enlace también aparece en varias secciones, permitiendo a los usuarios sin autenticación ser redirigidos al login cuando intentan acceder a áreas protegidas.
- Registrar: Formulario para crear una nueva cuenta de usuario.
- Actualizar Datos: Formulario para que el usuario pueda actualizar su información personal.

Menú de navegación lateral

El menú de navegación lateral incluye:

- Welcome: Página de bienvenida que proporciona una visión general o mensajes de introducción.

- Preguntas Frecuentes: Sección con respuestas a las preguntas más comunes sobre el uso de la aplicación.
- Conceptos: Información sobre los conceptos clave que son relevantes para el uso de la aplicación o el contexto de sus funcionalidades.
- Pre-Cuestionarios: Acceso a los cuestionarios iniciales que el usuario debe completar.
- MOOC: Acceso a los cursos en línea abiertos y masivos ofrecidos por la plataforma.
- Post-Cuestionarios: Acceso a los cuestionarios finales que el usuario debe completar después del MOOC.

Usuario Administrador

Además de tener acceso a todas las secciones disponibles para los usuarios no administradores, los administradores tienen acceso a funcionalidades adicionales para gestionar la aplicación:

- Dashboard Pre-Cuestionarios: Panel de control para ver y gestionar los resultados de los pre-cuestionarios completados por los usuarios.
- Gestión Pre-Cuestionarios: Herramientas para crear, editar y eliminar pre-cuestionarios.
- Dashboard Post-Cuestionarios: Panel de control para ver y gestionar los resultados de los post-cuestionarios completados por los usuarios.
- Gestión Post-Cuestionarios: Herramientas para crear, editar y eliminar post-cuestionarios.
- Dashboard MOOC: Panel de control para gestionar los cursos en línea, incluyendo progreso de los usuarios y análisis de datos.

Not Found

En caso de que un usuario intente acceder a una página o recurso que no existe, se redirige a una página "Not Found" que indica que el recurso no fue encontrado.

Esta estructura de navegación asegura que los usuarios puedan encontrar y acceder fácilmente a las secciones relevantes según sus roles y permisos, proporcionando una experiencia de usuario eficiente y bien organizada.

5.5. Diseño del back-end

5.4.1. Estructura General

La estructura de back-end de DeStance sigue una organización modular típica de un proyecto Node.js con Express.js, dividiéndose en varias carpetas según su funcionalidad: controllers, middleware, models, routes, shared, además de los archivos de configuración y el servidor. En la siguiente figura se muestra lo descrito anteriormente:

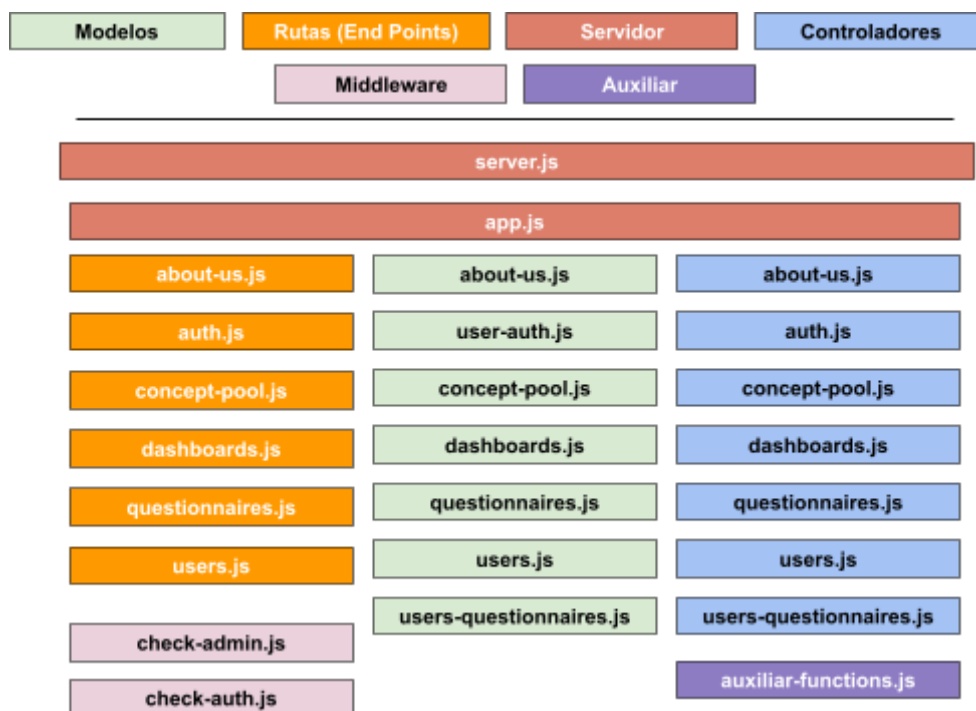


Figura 15: Diseño de los componentes del back-end de DeStance.

5.5.2. Descripción de Carpetas y Archivos

A continuación se muestra la estructura de las carpetas y archivos en el back-end de DeStance, así como su posterior descripción.

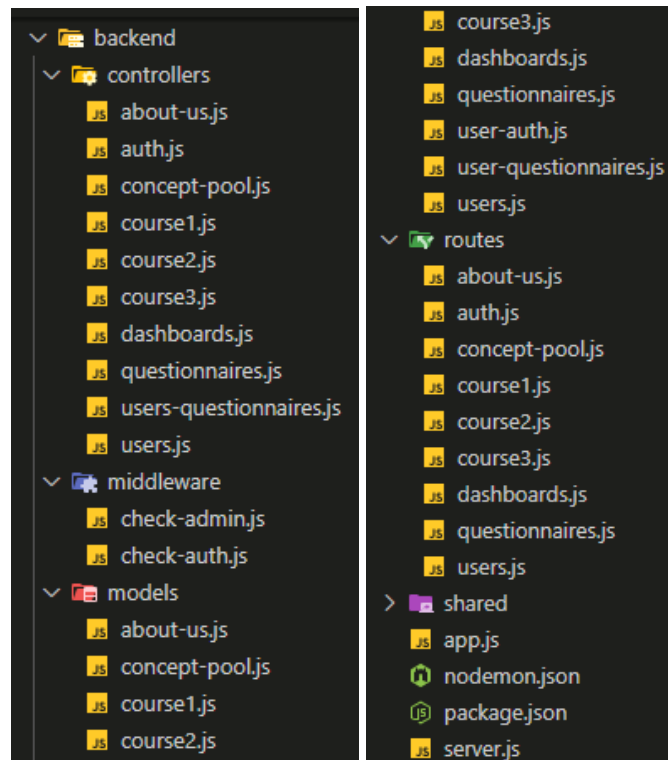


Figura 16: Estructura de carpetas en el back-end de DeStance.

controllers

Esta carpeta contiene los controladores, que manejan la lógica de negocio y las operaciones que responden a las solicitudes HTTP.

- about-us.js: Controlador para manejar la información sobre preguntas frecuentes.
- auth.js: Controlador para manejar la autenticación de usuarios (login, registro, etc).
- concept-pool.js: Controlador para gestionar el conjunto de conceptos.
- course1.js, course2.js, course3.js: ESTA FUNCIONALIDAD PERTENECE AL TFG DE OTRO ESTUDIANTE.
- dashboards.js: Controlador para gestionar la lógica relacionada con los tableros de control (dashboards).
- questionnaires.js: Controlador para manejar cuestionarios.
- users-questionnaires.js: Controlador para gestionar la relación entre usuarios y cuestionarios.

- users.js: Controlador para operaciones relacionadas con la gestión de usuarios.

middleware

Esta carpeta contiene middleware, que son funciones que procesan las solicitudes antes de que lleguen a los controladores.

- check-admin.js: Middleware para verificar si un usuario tiene privilegios de administrador.
- check-auth.js: Middleware para verificar si un usuario está autenticado.

models

Esta carpeta contiene los modelos de datos, que representan las estructuras de datos y la lógica asociada a la base de datos.

- about-us.js: Modelo para la información sobre preguntas frecuentes.
- concept-pool.js: Modelo para el conjunto de conceptos.
- course1.js, course2.js, course3.js: ESTA FUNCIONALIDAD PERTENECE AL TFG DE OTRO ESTUDIANTE.
- questionnaires.js: Modelo para los cuestionarios.
- user-auth.js: Modelo para la autenticación de usuarios.
- user-questionnaires.js: Modelo para la relación entre usuarios y cuestionarios.
- users.js: Modelo para los datos de los usuarios.

routes

Esta carpeta contiene las rutas de la aplicación, que mapean las URLs a los controladores correspondientes.

- about-us.js: Define las rutas para las operaciones sobre preguntas frecuentes.
- auth.js: Define las rutas para la autenticación de usuarios.
- concept-pool.js: Define las rutas para el conjunto de conceptos.

- course1.js, course2.js, course3.js: ESTA FUNCIONALIDAD PERTENECE AL TFG DE OTRO ESTUDIANTE.
- dashboards.js: Define las rutas para los tableros de control.
- questionnaires.js: Define las rutas para los cuestionarios.
- users.js: Define las rutas para la gestión de usuarios incluyendo las rutas de la relación de usuarios y cuestionarios.

shared

Esta carpeta contiene funciones auxiliares y compartidas que pueden ser utilizadas por diferentes partes de la aplicación.

- auxiliar-functions.js: Archivo con funciones auxiliares que pueden ser utilizadas en distintos lugares del backend.

Archivos raíz

- app.js: Archivo principal de configuración de la aplicación Express. Aquí se configuran los middleware globales, las rutas, la conexión a la base de datos y otros ajustes generales de la aplicación.
- package.json: Archivo de configuración de Node.js que incluye las dependencias, scripts y metadatos del proyecto.
- server.js: Archivo que inicia el servidor y hace que la aplicación comience a escuchar en un puerto específico.
- nodemon.js: Archivo que contiene las variables de entorno necesarias para la conexión a la base de datos.

5.5.3. Descripción de las API

Como se explicó en secciones pasadas, el back-end de DeStance consiste en un servidor REST API que sigue los principios del diseño RESTful para permitir la interacción entre el cliente y el servidor. Este expone varios endpoints¹ que permiten a los clientes interactuar con los datos y

¹ Un endpoint es un término utilizado en el contexto de las aplicaciones web y las APIs (Application Programming Interfaces) para referirse a una URL específica a la que puede enviarse una solicitud para interactuar con el servidor. En otras palabras, un endpoint es una dirección en un servidor donde se pueden realizar operaciones como obtener datos, enviar datos, actualizar información o eliminar datos [\[36\]](#).

funcionalidades de la aplicación. Cada endpoint está asociado con un recurso específico y responde a métodos HTTP como GET, POST, PUT y DELETE. En la siguiente tabla se muestra un resumen de los recursos asociados a los endpoints de DeStance:

Nombre del Recurso	Explicación
users	Representa a los usuarios de la aplicación
questionnaires	Representa los cuestionarios de la aplicación
concept-pool	Representa el conjunto de conceptos de la aplicación
dashboards	Representa los dashboards de la aplicación
about-us	Representa el conjunto de preguntas frecuentes de la aplicación
auth	Agrupar las tareas de autenticación de la aplicación
signup	Dentro de las tareas de autenticación, corresponde al registro de un nuevo usuario
login	Dentro de las tareas de autenticación, corresponde al inicio de sesión de un usuario
reset-request	Dentro de las tareas de autenticación, corresponde a la petición de cambio de contraseña
reset-password	Dentro de las tareas de autenticación, corresponde a la actualización de la contraseña
user	Dentro de las tareas de autenticación, corresponde a la gestión de los datos de autenticación del usuario (email, contraseña)

Tabla 2: Recursos del back-end de DeStance.

Cuando un cliente (front-end de DeStance) realiza una solicitud a uno de los endpoints de la API, esta es manejada por la ruta correspondiente que invoca el controlador adecuado. El controlador realiza la lógica de negocio necesaria, interactúa con los modelos para acceder o modificar los datos en la base de datos, y finalmente envía una respuesta al cliente con los datos solicitados o un mensaje de confirmación.

Por ejemplo, si un cliente realiza una solicitud GET a /users, la ruta definida en users.js invoca el controlador de usuarios, que a su vez usa el modelo de usuarios para obtener la lista de todos los usuarios y devuelve esta lista al cliente. A continuación se muestra un resumen de los endpoints del back-end de DeStance, así como las operaciones que permiten:

ENDPOINTS			GET	POST	PUT	DELETE
CUESTIONARIOS						
questionnaires			1	-	-	-
1) Devuelve todos los cuestionarios existentes en la base de datos.			Códigos ²			
			200 (OK)	-	-	-
questionnaires/{ type }			1	2	-	-
Parámetros	Explicación	Valores				
type	Tipo de cuestionario	pre-q				
		post-q				
1) Obtener todos los cuestionarios de determinado tipo (determinado por <i>type</i>).			Códigos			
			200 (OK)	201 (Created)	-	-

² Los códigos de respuesta HTTP son fundamentales para la comunicación entre clientes y servidores, y algunos de los más comunes son el 200, 201, 401, 404 y 500. El código 200 (OK) indica que la solicitud ha tenido éxito, siendo utilizado para respuestas exitosas de solicitudes GET, POST, PUT, DELETE, entre otras. El código 201 (Created) señala que la solicitud ha sido cumplida y ha resultado en la creación de un nuevo recurso, usándose principalmente en respuestas a solicitudes POST. El código 401 (Unauthorized) indica que la solicitud no ha sido aplicada porque carece de credenciales de autenticación válidas para el recurso solicitado. El código 404 (Not Found) se utiliza cuando el servidor no puede encontrar el recurso solicitado. Finalmente, el código 500 (Internal Server Error) indica que el servidor ha encontrado una situación que no sabe cómo manejar, siendo utilizado para errores genéricos del servidor, como cuando una operación en la base de datos falla inesperadamente y no se puede procesar la solicitud. Estos códigos permiten a los clientes comprender el resultado de sus solicitudes y actuar en consecuencia [37].

2) Añade un nuevo pre o post cuestionario (determinado por <i>type</i>) a la base de datos.						
questionnaires/{ id }			1	-	2	3
Parámetros	Explicación	Valores				
id	Identificador del cuestionario	Cadena ('abc..')				
1) Devuelve el cuestionario cuyo identificador coincide con <i>id</i> .			Códigos			
2) Modifica el cuestionario cuyo identificador coincide con <i>id</i> con nuevos datos.			200 (OK)	-	200 (OK)	
3) Elimina el cuestionario cuyo identificador coincide con <i>id</i> .			404 (Not Found)		404 (Not Found)	-
					500 (Internal Server Error)	
USUARIOS						
users			1	2	-	-
1) Devuelve todos los usuarios de la base de datos.			Códigos			
2) Crea un nuevo usuario en la base de datos.			200 (OK)	201 (Created)	-	-
users/{ email }			1	-	-	-
Parámetros	Explicación	Valores				
email	Email del usuario	Cadena ('abc@aa.aa')				
1) Devuelve el usuario cuyo correo coincide con el valor de <i>email</i> .			Códigos			
			200 (OK)	-	-	-
			404 (Not			

			Found)			
users/{ id }			-	-	1	2
Parámetros	Explicación	Valores				
id	Identificador del usuario	Cadena ('abc..')				
1) Modifica el usuario cuyo identificador coincide con <i>id</i> .			Códigos			
2) Elimina el usuario cuyo identificador coincide con <i>id</i> .			-	-	200 (OK)	200 (OK)
USUARIOS-CUESTIONARIOS						
users/questionnaires/{ type }			1	-	-	-
Parámetros	Explicación	Valores				
type	Tipo de cuestionario	pre-q				
		post-q				
1) Devuelve todas las respuestas de los usuarios a los cuestionarios de determinado tipo (determinado por <i>type</i>).			Códigos			
			200 (OK)	-	-	-
users/{ user_id }/questionnaires			1	-	-	-
Parámetros	Explicación	Valores				
user_id	Identificador del usuario	Cadena ('abc..')				
1) Devuelve todas las respuestas a los cuestionarios del usuario cuyo identificador coincide con <i>user_id</i> .			Códigos			
			200 (OK)	-	-	-
users/{ users_id }/questionnaires/{ type }			1	-	-	-

Parámetros	Explicación	Valores				
user_id	Identificado r del usuario	Cadena ('abc..')				
type	Tipo de cuestionario	pre-q				
		post-q				
1) Devuelve todas las respuestas a los cuestionarios de determinado tipo (determinado por <i>type</i>) del usuario cuyo identificador coincide con <i>user_id</i> .			Códigos			
			200 (OK)	-	-	-
users/{ users_id }/questionnaires/{ quest_id }			1	2	-	3
Parámetros	Explicación	Valores				
user_id	Identificador del usuario	Cadena ('abc..')				
quest_id	Identificador del cuestionario	Cadena ('abc..')				
1) Devuelve todas las respuestas al cuestionario cuyo identificador coincide con <i>quest_id</i> , del usuario cuyo identificador coincide con <i>user_id</i> . 2) Crea una nueva respuesta del usuario cuyo identificador coincide con <i>user_id</i> al cuestionario cuyo identificador coincide con <i>quest_id</i> . 3) Elimina la respuesta del usuario cuyo identificador coincide con			Códigos			
			200 (OK)	201 (Created)	-	200 (OK)
			404 (Not Found)	500 (Internal Server Error)		404 (Not Found)

user_id al cuestionario cuyo identificador coincide con quest_id.						500 (Intern al Server Error)
DASHBOARDS						
dashboards/ { type }			1	-	-	-
Parámetros	Explicación	Valores				
type	Tipo de dashboard	pre-q				
		post-q				
1) Devuelve los dashboards de determinado tipo (determinado por type) de la base de datos.			Códigos			
			200 (OK)	-	-	-
dashboards/{ quest_id }			1	-	-	2
Parámetros	Explicación	Valores				
quest_id	Identificador del cuestionario al que pertenece el dashboard	Cadena ('abc..')				
1) Devuelve el dashboard asociado al cuestionario de identificador quest_id.			Códigos			
2) Elimina el dashboard asociado al cuestionario de identificador quest_id.			200 (OK)	-	-	200 (OK)
			404 (Not Found)			
GENERAL						
about-us			1	2	-	3

1) Devuelve todas las preguntas frecuentes. 2) Crea una nueva pregunta frecuente en la base de datos. 3) Elimina todas las preguntas frecuentes.	Códigos								
	200 (OK)	201 (Created)	-	200 (OK)					
about-us/{ id }	1	-	-	-					
<table><tr><th>Parámetros</th><th>Explicación</th><th>Valores</th></tr><tr><td>id</td><td>Identificador de la pregunta frecuente</td><td>Cadena ('abc..')</td></tr></table>		Parámetros	Explicación	Valores	id	Identificador de la pregunta frecuente	Cadena ('abc..')		
Parámetros	Explicación	Valores							
id	Identificador de la pregunta frecuente	Cadena ('abc..')							
1) Devuelve la pregunta frecuente cuyo identificador coincide con <i>id</i> .	Códigos								
	200 (OK)	-	-	-					
	404 (Not Found)								
concepts-pool	1	2	-	3					
1) Devuelve todos los conceptos de la base de datos. 2) Crea un nuevo concepto en la base de datos. 3) Elimina todos los conceptos de la base de datos.	Códigos								
	200 (OK)	201 (Created)	-	200 (OK)					
concepts-pool/{ id }	1	-	-	-					
<table><tr><th>Parámetros</th><th>Explicación</th><th>Valores</th></tr><tr><td>id</td><td>Identificador del concepto</td><td>Cadena ('abc..')</td></tr></table>		Parámetros	Explicación	Valores	id	Identificador del concepto	Cadena ('abc..')		
Parámetros	Explicación	Valores							
id	Identificador del concepto	Cadena ('abc..')							
1) Devuelve el concepto cuyo identificador coincide con <i>id</i> .	Códigos								
	200 (OK)	-	-	-					

	404 (Not Found)			
<i>AUTENTICACIÓN</i>				
auth/signup	-	1	-	-
1) Crea un nuevo usuario (solo datos de autenticación) en la base de datos.	Códigos			
	-	201 (Created)	-	-
		500 (Server Internal Error)		
auth/login	-	1	-	-
1) Permite el inicio de sesión por parte del usuario.	Códigos			
	-	200 (OK)	-	-
		401 (Unauthorized)		
auth/reset-request	-	1	-	-
1) Permite comenzar el proceso de recuperación de contraseña.	Códigos			
	-	200 (OK)	-	-
		401 (Unauthorized)		
		500 (Internal Server Error)		
auth/reset-password	-	1	-	-
1) Permite el cambio de contraseña dentro del proceso de recuperación de contraseña.	Códigos			
	-	200 (OK)	-	-

		401 (Unauth orized)		
		500 (Internal Server Error)		
auth/user	-	-	1	-
1) Permite modificar los datos de autenticación de un usuario.	Códigos			
	-		200 (OK)	-
			500 (Intern al Server Error)	

Tabla 3: Endpoints del back-end de DeStance.

6. DESARROLLO DE DeSTANCE

6.1. Planificación Inicial

Para el desarrollo de DeStance se contaba con la siguiente distribución de horas en tareas generales:

Tareas Generales	Horas totales
Comprensión y Análisis del problema	27
Diseño e Implementación	200
Pruebas	30
Documentación, memorias y presentación	40

Tabla 4: Distribución general de horas de DeStance.

Teniendo en cuenta la [Tabla 4](#), se definen las siguientes tareas dados los requisitos obtenidos en la [Sección 3](#):

Nombre de la tarea	Horas a dedicar
Reuniones recurrentes	12
Obtener y Analizar Requisitos	10
Documentar requisitos	2
Realizar planificación de trabajo	5
Documentar planificación de trabajo	2
Realizar Diseño de Alto Nivel	10
Realizar Diseño de Bajo Nivel	10
Documentar Diseño	4
Crear Base de Datos	10
Instalar herramientas necesarias	5
Setup de Frontend base	30

Consulta About the tool	5
Consulta Conceptos	5
Pre-Cuestionarios Frontend	15
Post-Cuestionarios Frontend	10
Pre-Cuestionarios Backend	15
Post-Cuestionarios Backend	10
Sign-up/Login Frontend	15
Sign-up/Login Backend	10
Dashboard Cuestionarios Frontend	10
Dashboard Cuestionarios Backend	20
Gestor Cuestionarios Frontend	10
Gestor Cuestionarios Backend	10
Documentar dificultades y descripción de progreso	32
Pruebas Cuestionarios Frontend	4
Pruebas Cuestionarios Backend	4
Pruebas Dashboard Frontend	4
Pruebas Dashboard Backend	4
Pruebas Gestor Frontend	4
Pruebas Gestor Backend	4
Documentar pruebas	6
Horas Totales	297

Tabla 5: Distribución tareas por horas de DeStance.

En la [Figura 17](#) se distribuyen las tareas de la [Tabla 5](#) desde el día 28 de febrero de 2024 hasta el 3 de junio de 2024 a través de un Diagrama de Gantt.

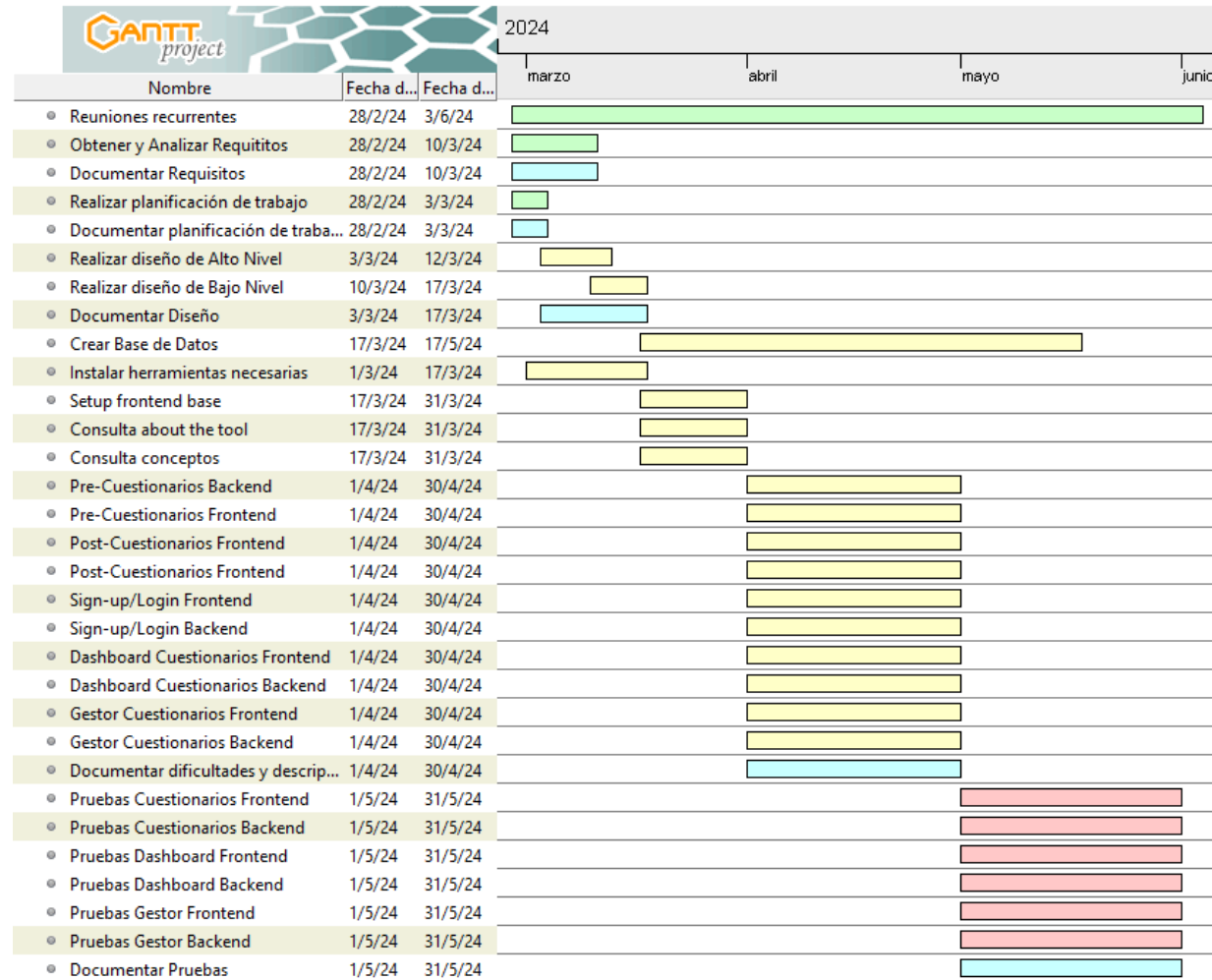


Figura 17: Diagrama de Gantt planificado para DeStance.

Este cronograma abarca desde la obtención y análisis de requisitos hasta las pruebas y documentación de la aplicación, cubriendo todas las fases esenciales del desarrollo, desde la planificación y diseño hasta la implementación y validación final.

6.2. Progreso Actual

Durante el mes de marzo hasta la primera semana de abril, el trabajo estuvo enfocado en el front-end de la aplicación. Durante este período, se crea la interfaz de usuario para todas las funcionalidades de la aplicación. Esto incluyó la creación de componentes visuales, la implementación de formularios y la integración de estilos y diseños para asegurar una experiencia de usuario atractiva y cohesiva. El objetivo era tener una versión funcional y visualmente completa del front-end que pudiera ser utilizada para futuras pruebas y mejoras.

A partir de la segunda semana de abril hasta mediados de mayo, el enfoque cambió hacia el servidor API del back-end, la base de datos y la integración con el servidor web de Angular. En esta fase, se trabajó en la implementación de la lógica de negocio del servidor, desarrollando las API necesarias para manejar las solicitudes del cliente. Se crearon los modelos de datos y se configuró la base de datos para almacenar y gestionar la información de manera eficiente. Además, se trabajó en la integración del front-end con el back-end, asegurando que las llamadas a la API se manejaran correctamente y que los datos se presentaran de manera adecuada en la interfaz de usuario.

Sin embargo, esta estrategia de desarrollo no resultó ser la más eficiente debido a que para probar algunas funcionalidades del front-end se tuvo que implementar código de prueba para simular cómo funcionaría cada característica con un back-end y una base de datos. Este enfoque resultó en una pérdida de tiempo, pues implicaba crear soluciones de código temporales para probar la interfaz, para luego ser descartadas una vez la lógica del back-end fue implementada.

Hubiera sido más eficiente adoptar un enfoque iterativo, desarrollando y probando cada funcionalidad de manera completa y simultánea, tanto en el front-end como en el back-end. Este enfoque habría permitido identificar y resolver los problemas del código final, y evitar perder el tiempo en problemas del código temporal, mejorando la eficiencia del proceso y asegurando una mejor integración entre el front-end y el back-end desde el principio.

DeStance no actúa como una herramienta educativa independiente, sino que sirve como una plataforma complementaria para fortalecer los objetivos de RACISMMAFF [\[1\]](#). Luego, para afectar de manera positiva el discurso relacionado con la inmigración y el racismo de los estudiantes universitarios y futuros profesores de la Unión Europea (UE), DeStance, necesita que sus cuestionarios, conceptos, contenido del MOOC, etcétera, sean desarrollados por los investigadores encargados de RACISMMAFF.

Los investigadores de RACISMMAFF son responsables de proporcionar contenido especializado y bien investigado que será incorporado en los diferentes módulos de DeStance. Esto incluye la elaboración de cuestionarios diseñados para evaluar y modificar las percepciones de los estudiantes sobre la inmigración y el racismo, la creación de conceptos clave que deben ser comprendidos y asimilados, y la producción de contenido educativo para el curso en línea masivo y abierto (MOOC). Este contenido debe ser meticuloso y estar basado en evidencias para asegurar que la información presentada a los usuarios sea precisa y efectiva en el cambio de actitudes y comportamientos.

Debido a esta dependencia del contenido proporcionado por los investigadores, la fecha del despliegue final de DeStance está intrínsecamente ligada al momento en que los datos reales y el contenido educativo estén completos. Hasta que todo el material necesario no haya sido desarrollado y validado por los expertos, la aplicación no puede ser lanzada al público. Esto significa que si bien las funcionalidades pedidas en la [Sección 3](#) se completaron y probaron satisfactoriamente, los datos reales en algunas partes de la aplicación no están disponibles.

Específicamente, los dashboards de la aplicación DeStance son los más afectados por la falta de datos reales. En su estado actual, muestran estadísticas basadas en datos de prueba, dado que aún no se han recopilado respuestas reales de los usuarios a los cuestionarios. Este uso de datos simulados limita su capacidad para reflejar la verdadera eficacia y el impacto de la aplicación en tiempo real, ya que no están basadas en interacciones genuinas con el contenido educativo de DeStance. Por tanto, antes de desplegar la aplicación, se eliminarán estas respuestas de prueba.

Desde la fecha de entrega de este documento hasta su exposición, se planea continuar trabajando diligentemente en el desarrollo de los cuestionarios reales, conceptos y preguntas frecuentes. Este trabajo incluye la elaboración de preguntas que evalúen con precisión el conocimiento y las actitudes de los usuarios sobre temas relacionados con la inmigración y el racismo. También se están afinando los conceptos clave que deben ser comunicados eficazmente a través del contenido del MOOC y las preguntas frecuentes, las cuales deben abordar las dudas comunes y proporcionar claridad sobre los objetivos y el funcionamiento de la aplicación.

6.3. Resultado Final

Teniendo en cuenta que, a pesar de haber implementado todas las funcionalidades solicitadas, el proyecto sigue abierto para completar los cuestionarios, conceptos y preguntas frecuentes, así como decisiones de diseño como el logo y la página principal, a continuación se mostrarán las capturas más recientes a la interfaz de usuario, asociándolas con cada caso de uso de [Sección 5.2](#).

Pantalla Inicial

Al acceder a DeStance por primera vez te llevará a la sección de bienvenida, lo cual se muestra en [Figura 18](#). Aclarar que no es la última versión de esta, lo cual se evidencia en el logo de la aplicación, que por ahora hace referencia a su antiguo nombre (RACISMMAFF), y al texto de prueba.

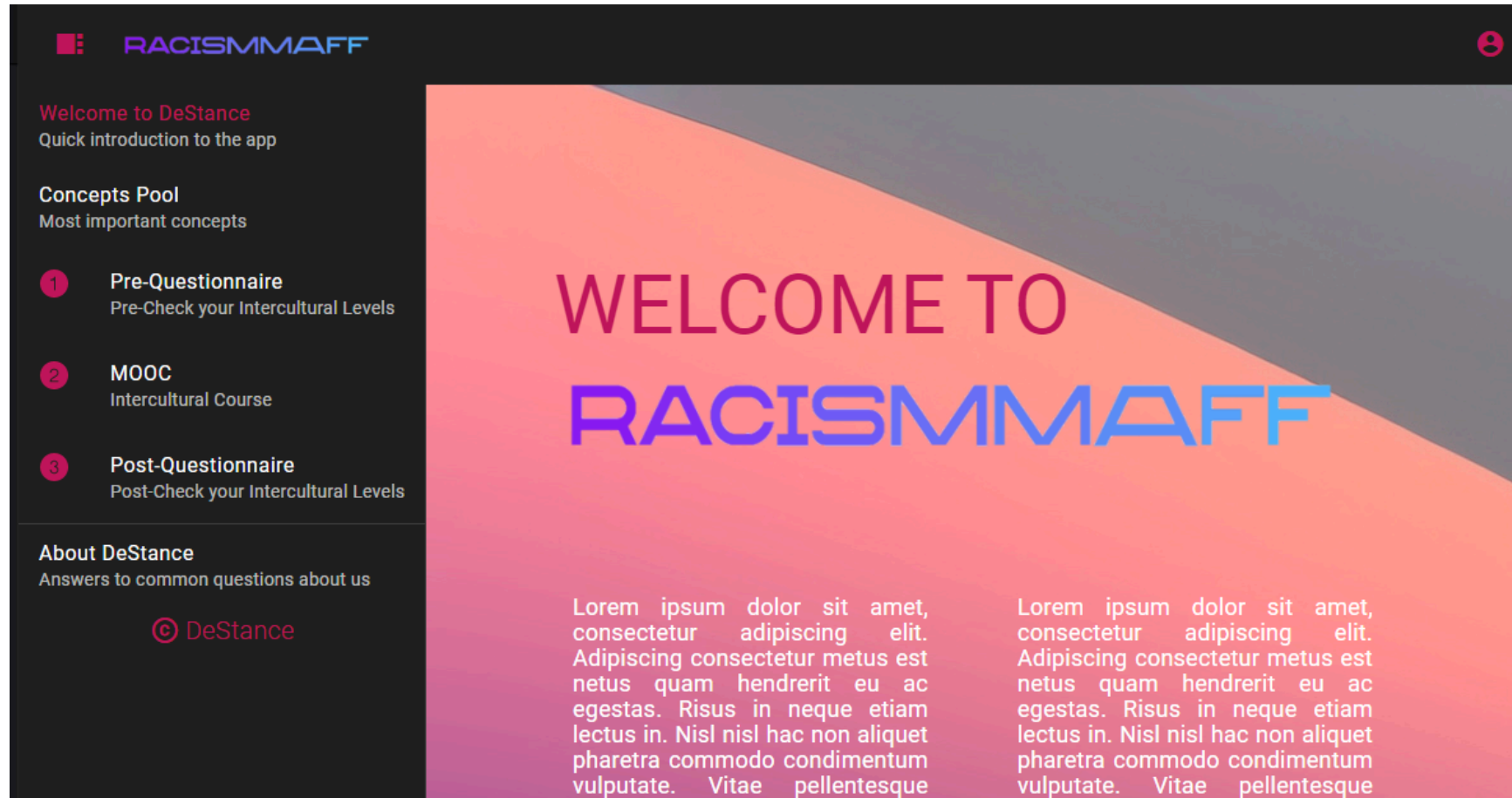


Figura 18: Sección temporal de bienvenida de DeStance.

I. Registrarse

Tomando como referencia la [Figura 18](#), para acceder al registro se debe hacer click en el botón de perfil de usuario, en la barra de navegación superior, a la derecha del todo, y luego seleccionar la opción Login/Sign-up del menú mostrada en la [Figura 19.1](#).

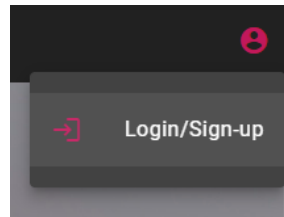


Figura 19.1: Registro - Menú de Login/Sign-up.

Una vez realizado esto, la aplicación lleva al usuario a la sección de Iniciar Sesión, por lo que para acceder a Registro se debe seleccionar el link mostrado en la [Figura 19.2](#) que indica que el usuario no tiene una cuenta todavía. Esta acción lleva a la pantalla de la [Figura 19.3](#) donde se muestra un mensaje que advierte al usuario sobre cómo se usarán sus datos, pidiendo su consentimiento antes de registrarse.

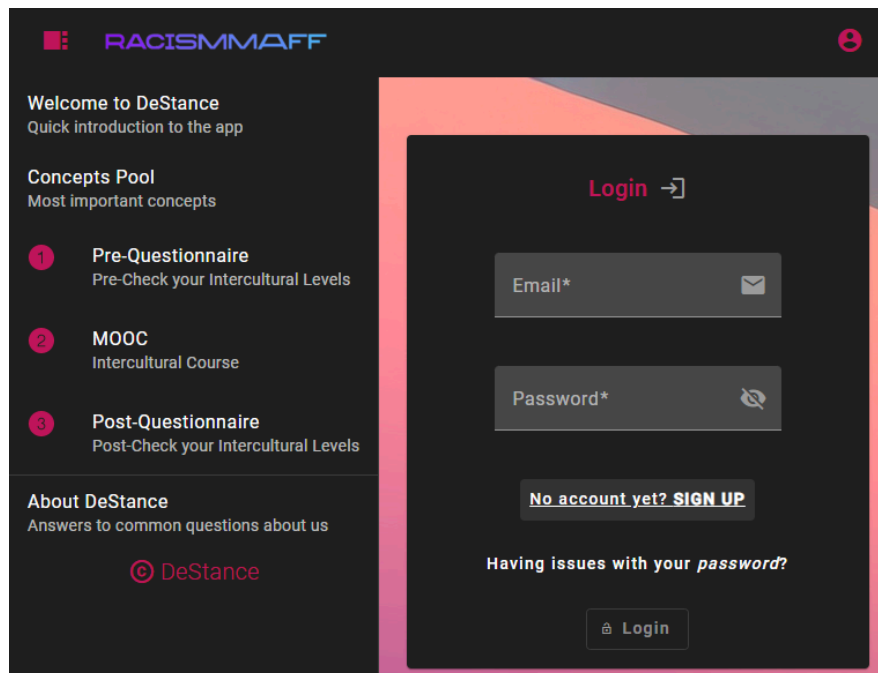


Figura 19.2: Registro - Iniciar-Sesión - El usuario no tiene cuenta todavía.

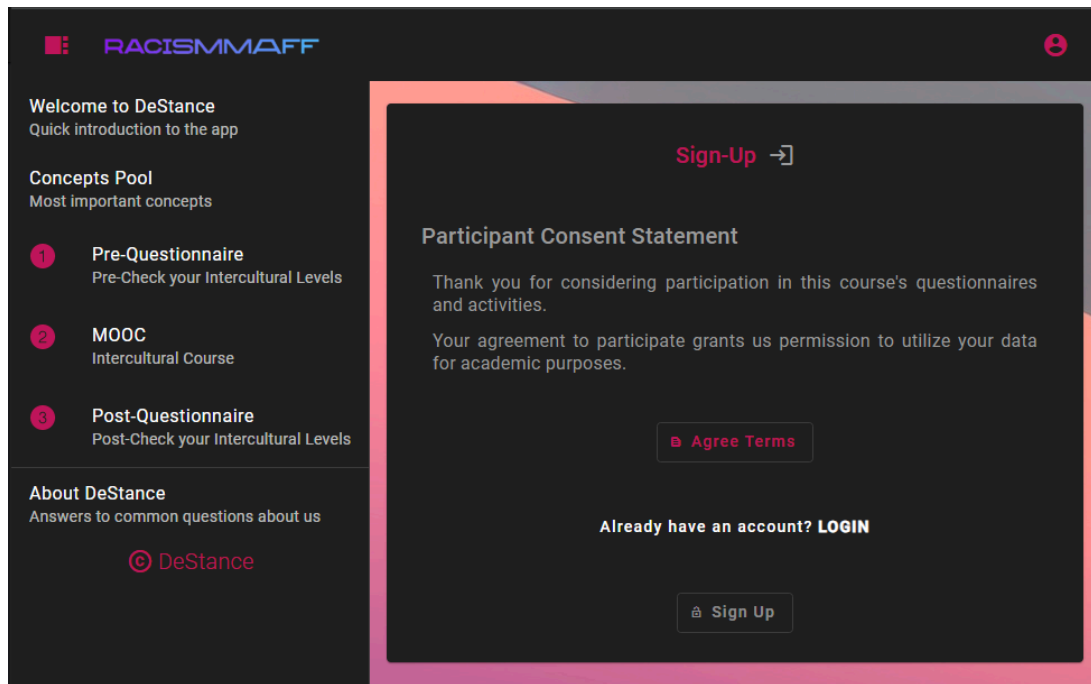


Figura 19.3: Registro DeStance - Acuerdo de consentimiento del participante.

En esta pantalla se podrá retornar a la página de Inicio de Sesión seleccionando “*Already have an account? LOGIN*”. El botón *Sign Up* permanece deshabilitado hasta que se completen todos los pasos, comenzando por el siguiente a realizar, seleccionar el botón *Agree Terms*. En la [Figura 19.4](#) se muestra el formulario de registro, donde todos los campos son obligatorios y hasta que no se completen no se habilita el botón de Sign Up. En la [Figura 19.5](#) se observan todos los campos rellenos con datos de prueba.

El siguiente paso sería seleccionar el botón *Sign Up*, lo que desencadena el cuadro de diálogo de la [Figura 19.6](#), que pregunta al usuario si está seguro de los datos que introdujo. Responder Ok debería satisfactoriamente registrar el nuevo usuario e iniciar la sesión de este automáticamente por un plazo máximo de una hora, mostrando como mensaje de confirmación al usuario el cuadro de diálogo de bienvenida a DeStance de la [Figura 19.7](#).

Welcome to DeStance
Quick introduction to the app

Concepts Pool
Most important concepts

- 1 Pre-Questionnaire
Pre-Check your Intercultural Levels
- 2 MOOC
Intercultural Course
- 3 Post-Questionnaire
Post-Check your Intercultural Levels

About DeStance
Answers to common questions about us

DeStance

Sign-Up →

Change Avatar

Name* Email*

Password* Confirm Password*

Date of Birth* Gender*

MM/DD/YYYY

Mother Tongue* Language*

Language in which you will complete the MOOC

Language Level*

Language proficiency level in which you will complete the MOOC

+ Add a Nationality

Nationality*

Already have an account? LOGIN

Sign Up

Figura 19.4: Registro DeStance. Formulario de registro vacio.

Welcome to DeStance
Quick introduction to the app

Concepts Pool
Most important concepts

- 1 Pre-Questionnaire
Pre-Check your Intercultural Levels
- 2 MOOC
Intercultural Course
- 3 Post-Questionnaire
Post-Check your Intercultural Levels

About DeStance
Answers to common questions about us

DeStance

Sign-Up →

Change Avatar

Name* User 13 Email* user13@gmail.com

Password* Confirm Password*

Date of Birth* 6/24/1997 Gender* Male

MM/DD/YYYY

Mother Tongue* Spanish - español Language* Spanish - español

Language in which you will complete the MOOC

Language Level* C2 (Proficiency)

Language proficiency level in which you will complete the MOOC

+ Add a Nationality

Nationality* United States

Nationality* Spain

Already have an account? LOGIN

Sign Up

Figura 19.5: Registro DeStance. Formulario de registro con datos de prueba.

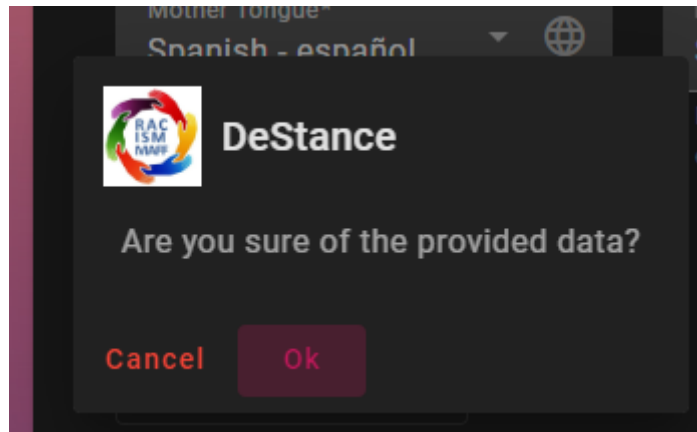


Figura 19.6: Registro DeStance. Cuadro de diálogo. Confirmación de datos introducidos.

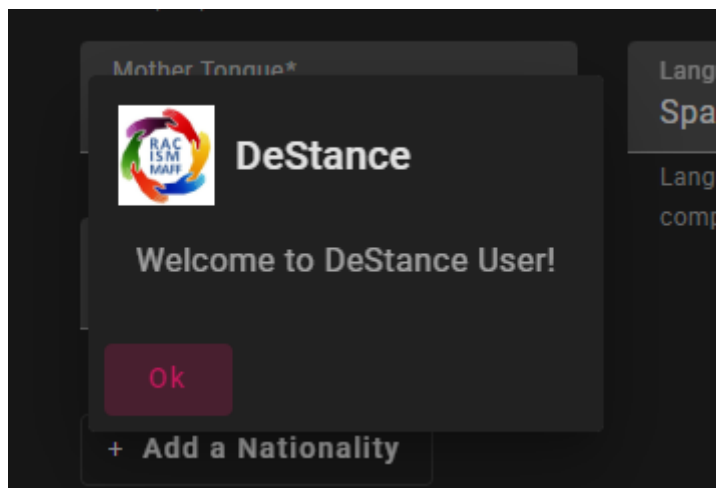


Figura 19.7: Registro DeStance. Cuadro de diálogo. Bienvenida a DeStance.

II. Iniciar Sesión

Para iniciar sesión el usuario debe seguir los mismos pasos de Registro hasta la pantalla de la [Figura 19.2](#), procediendo luego a introducir su email y contraseña en el formulario y seleccionar el botón *Login*, habilitado una vez se completen ambos campos con valores válidos. Esto se evidencia en la [Figura 20](#).

III. Recuperación de contraseña

Para comenzar el proceso de cambio de contraseña, tal como en el casos anterior, el usuario debe seguir los mismos pasos de Registro hasta la pantalla

de la [Figura 19.2](#), seleccionando luego el link “*Having issues with your password*”, lo que lo llevará a la pantalla de la [Figura 21.1](#).

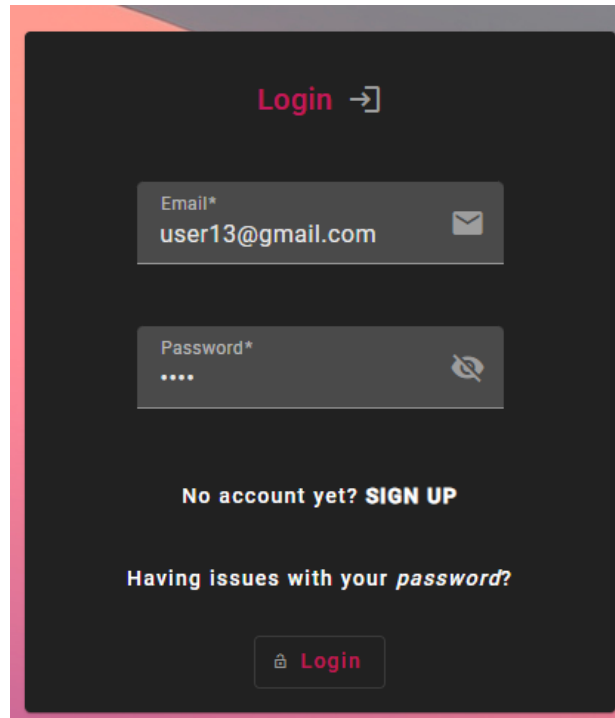
The image shows a login screen for the DeStance app. At the top, there is a 'Login' button with a right-pointing arrow. Below it are two input fields: 'Email*' with the text 'user13@gmail.com' and an envelope icon, and 'Password*' with four dots and an eye icon. Under the password field, there is a link that says 'No account yet? SIGN UP'. Below that is another link that says 'Having issues with your password?'. At the bottom, there is a 'Login' button with a key icon.

Figura 20: Inicio Sesión DeStance. Formulario de inicio de sesión con datos de prueba.

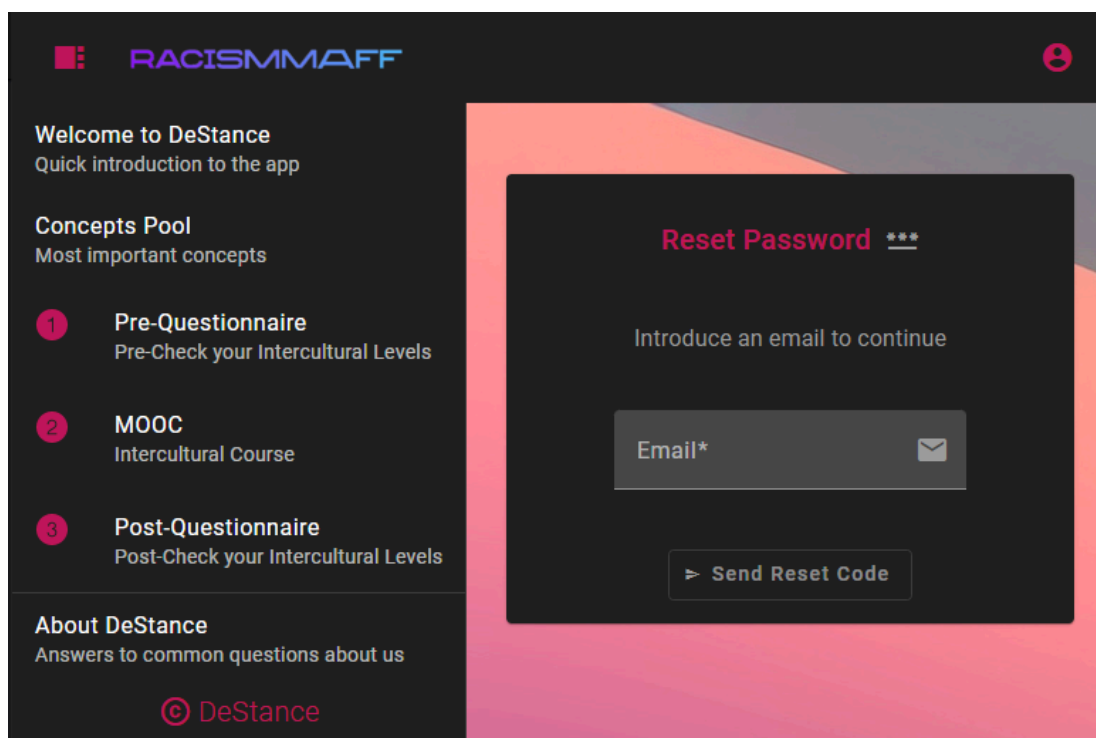
The image shows the home screen of the DeStance app. The top bar has the 'RACISMMAFF' logo and a user profile icon. The main content area is divided into two sections. On the left, there is a 'Welcome to DeStance' section with a 'Quick introduction to the app' link, followed by a 'Concepts Pool' section with 'Most important concepts' and a list of three items: '1 Pre-Questionnaire Pre-Check your Intercultural Levels', '2 MOOC Intercultural Course', and '3 Post-Questionnaire Post-Check your Intercultural Levels'. Below this is an 'About DeStance' section with a link 'Answers to common questions about us'. At the bottom left is the '© DeStance' logo. On the right, there is a large overlay for the 'Reset Password' process. It has a title 'Reset Password ***', a subtitle 'Introduce an email to continue', an 'Email*' input field with an envelope icon, and a 'Send Reset Code' button with a right-pointing arrow.

Figura 21.1: Recuperación de contraseña DeStance - Email de recuperación.

Una vez introducido el email de recuperación se debe seleccionar el botón *Send Reset Code* para recibir el código de validación en dicha dirección de correo. En la [Figura 21.2](#) se muestra un mensaje de confirmación al usuario de que fue enviado el código. En la [Figura 21.3](#) se observa la pantalla siguiente donde el usuario completaría este caso de uso introduciendo el código, y la nueva contraseña, finalizando el proceso con la selección del botón *Reset Password*. A su vez el botón *Reset Code Email*, llevaría al usuario a la pantalla anterior ([Figura 21.1](#)) para volver a enviar el código a otro email o al mismo.

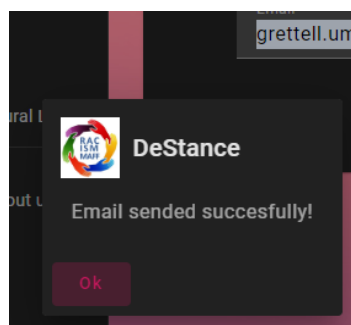


Figura 21.2: Recuperación de contraseña DeStance. Confirmación email de recuperación.

A screenshot of a mobile application interface for password reset. The title is 'Reset Password ***'. Below the title, there is a text instruction: 'Check the inbox of the email provided and introduce the last code sent to reset the password.' There are three input fields: 'Code*' with the value 'd97814d9476f2c7c81bt', 'Password*', and 'Confirm Password*'. Each field has a red eye icon for toggling visibility. At the bottom, there are two buttons: 'Reset Code Email' and 'Reset Password'.

Figura 21.3: Recuperación de contraseña DeStance. Formulario de recuperación de contraseña.

En la [Figura 21.4](#) se muestra el cuadro de diálogo de confirmación del cambio satisfactorio de la contraseña.

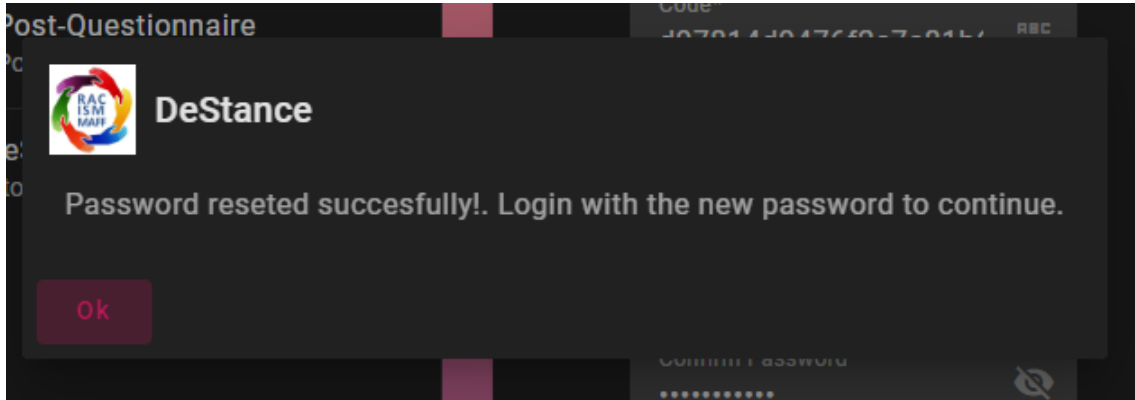


Figura 21.4: Recuperación de contraseña DeStance -Confirmación cambio de contraseña.

IV. Cerrar Sesión

Para cerrar una sesión activa se debe hacer click en el botón de perfil de usuario, en la barra de navegación superior, a la derecha del todo, y seleccionar Logout en el menú de usuario mostrado a continuación.

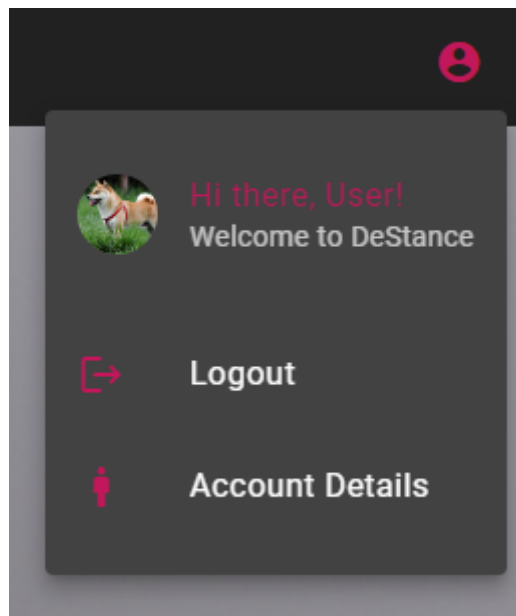


Figura 22: Cerrar sesión.

V. Visualizar Introducción a DeStance

Para acceder a la pantalla de bienvenida a DeStance de la [Figura 18](#) se debe seleccionar la opción *Welcome to DeStance* en el menú de navegación lateral izquierdo que se observa en dicha figura, donde se aprecia además esta opción como seleccionada mostrándola en color rosado en lugar de blanco. Para ocultar o mostrar el menú de navegación lateral se debe seleccionar el botón que se halla a la izquierda del logo de la aplicación, en la barra de navegación superior. Esto se observa en la figura siguiente:

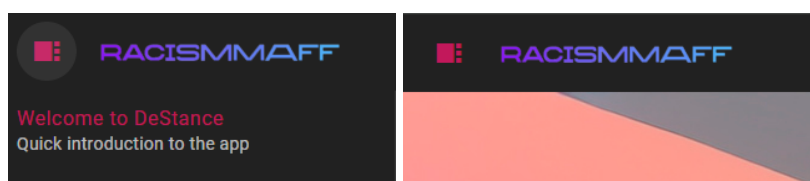


Figura 23: Botón de ocultar/mostrar menú de navegación lateral izquierdo.

VI. Visualizar Conceptos

Tomando como referencia la [Figura 18](#) (pantalla inicial), a la sección de Conceptos se podrá acceder seleccionando la opción *Concepts Pool* en el menú de navegación lateral izquierdo, acción cuyo resultado se muestra a continuación:

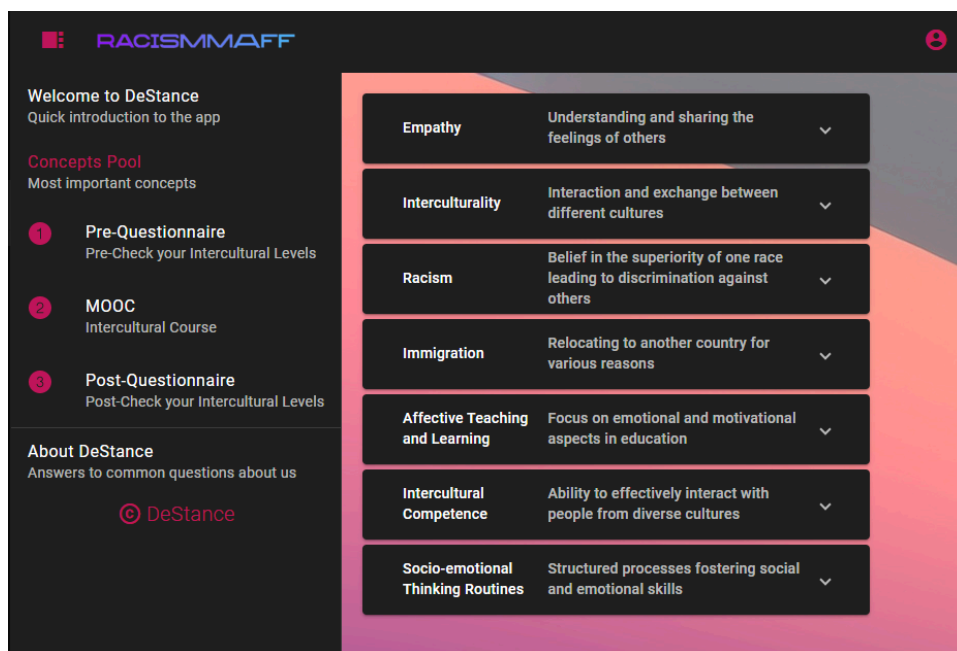


Figura 24.1: Concepts-pool. Conceptos no desplegados.

En la [Figura 24.1](#) se observa la vista general de algunos de los conceptos de la aplicación (serán añadidos más conceptos en la versión final). Para obtener la vista detallada de cualquiera de ellos bastaría con hacer click sobre ellos para desplegarlos y luego hacer click nuevamente para volver al estado no detallado. En la figura siguiente se despliega el concepto *Empathy*:

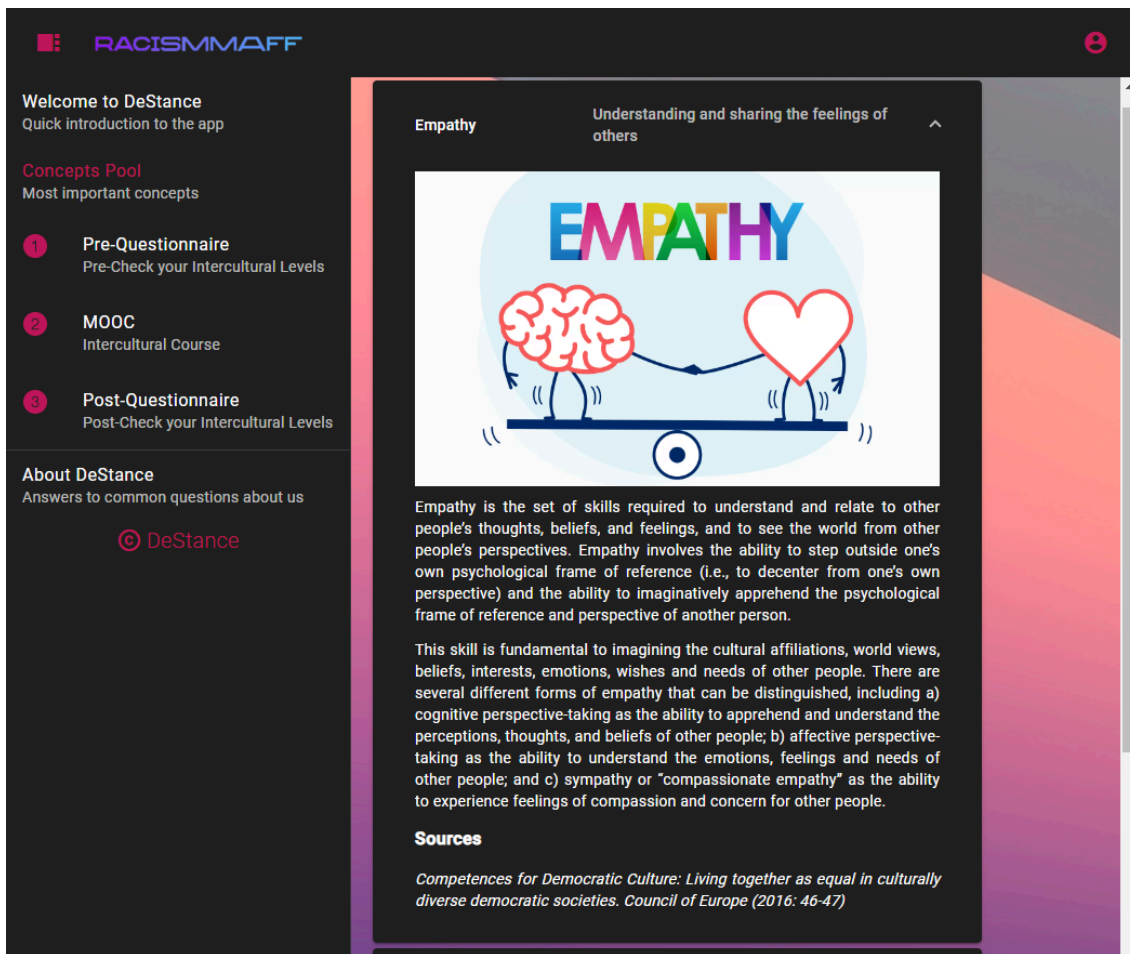


Figura 24.2: Concepts-pool. Concepto de Empathy desplegado.

VII. Visualizar Preguntas Frecuentes

Tomando como referencia la [Figura 18](#) (pantalla inicial), a la sección de Preguntas Frecuentes se podrá acceder seleccionando la opción *About DeStance* en el menú de navegación lateral izquierdo, acción cuyo resultado se muestra a continuación:

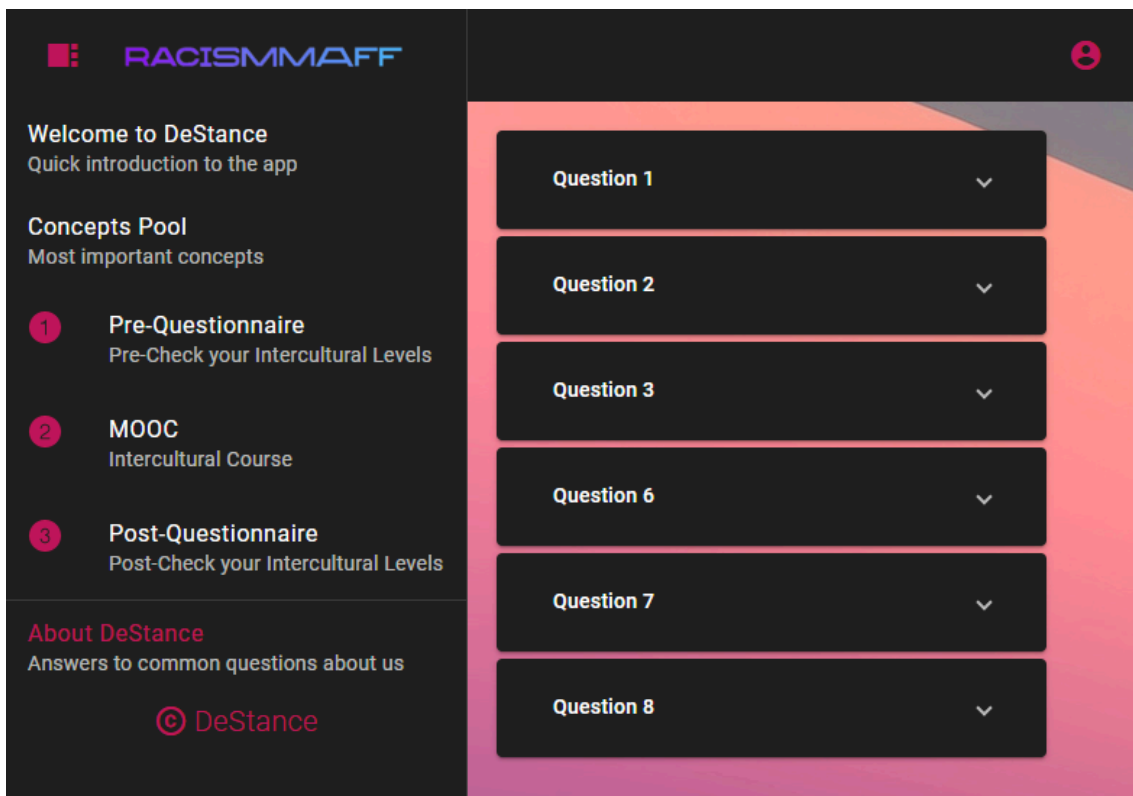


Figura 25: About DeStance. Preguntas no desplegadas.

En la [Figura 25](#) se observa la vista general de las preguntas frecuentes de la aplicación (en este caso son datos de prueba). Para obtener la vista detallada de cualquiera de ellas bastaría con hacer click sobre ellas para desplegarlas y luego hacer click nuevamente para volver al estado no detallado.

VIII. Ver perfil

Para consultar sus datos personales el usuario debe primeramente iniciar la sesión ([Caso de Uso II](#)). Tomando como referencia la [Figura 18](#) de pantalla inicial, se debe seleccionar el botón de perfil de usuario que se observa a la derecha del todo, en la barra de navegación superior. En el menú que se muestra en la [Figura 22](#) se debe hacer click en *Account Details* . En la [Figura 26.1](#) se aprecian los detalles actuales del usuario una vez se realiza la acción anterior.

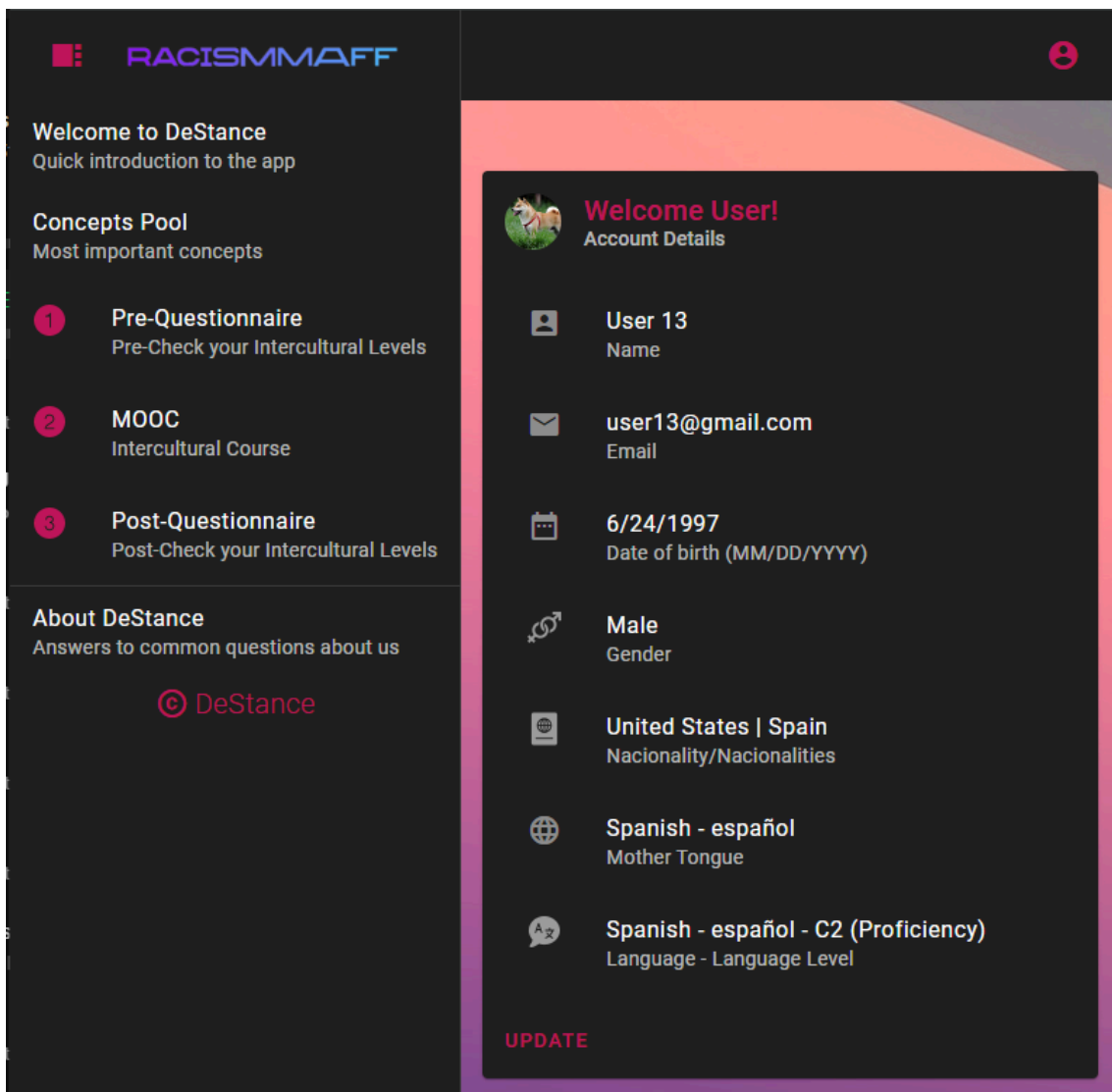


Figura 26: Account Details.

IX. Actualizar perfil

Hacer click en el botón *Update* de la [Figura 26](#) permite al usuario editar sus datos personales. En la [Figura 27.1](#) se muestra la sección de modificación de datos del usuario. En ella se puede optar por modificar solo la imagen de avatar, cambiándola con el botón *Update Avatar*, seleccionando la imagen deseada en el cuadro de diálogo de la [Figura 27.2](#) y guardando la opción marcada haciendo click en *Save Choice*. Para guardar la modificación en la base de datos, se debe seleccionar el botón *Save Only Avatar* o *Save Changes* en la sección de la izquierda (modificación de datos personales). El primero

solo cambiará la imagen del avatar, mientras que el segundo guardará todas las modificaciones realizadas a los datos personales.

The screenshot displays the 'Update Account' interface with two main panels. The left panel, titled 'Update User Details', contains fields for Name (User 13), Date of Birth (6/24/1997), Gender (Male), Mother Tongue (Spanish - español), Language (Spanish - español), Language Level (C2 (Proficiency)), and Nationality (United States and Spain). The right panel, titled 'Update Authentication Details', contains fields for Email, Password, and Confirm Password. Both panels have 'Save Changes', 'Save Only Avatar', and 'Cancel' buttons at the bottom.

Figura 27.1: Update Account.

The screenshot displays the 'Choose an Avatar' interface, which shows a grid of eight animal images. The first row contains a dog, a cat, a dog in a Santa hat, a cat, a dog, and a cat. The second row contains a horse and a dog. The 'Save Choice' button is highlighted in red.

Figura 27.2: Cambiar imagen de avatar.

Modificar cualquier otro dato personal requerirá cambiar el valor actual en el campo correspondiente y luego seleccionar el botón *Save Changes* en la sección de la izquierda (*Update User Details*). En cuanto a los datos de

autenticación, se permite al usuario cambiar su correo electrónico y/o actualizar su contraseña, sin embargo será necesario introducir ambos valores, aunque no se quiera cambiar uno de los dos. Simplemente, en dicho caso, se introduce el mismo valor en el campo correspondiente. Una vez el usuario introduzca los valores actualizados de su email y/o contraseña, deberá seleccionar el botón *Save Changes* de la sección derecha (*Update Authentication Details*).

A continuación, tanto en este caso de modificación de los datos de autenticación, como en el de modificación de datos personales, aparecerá un cuadro de diálogo preguntando al usuario si está seguro de los datos introducidos, lo cual se observa en la [Figura 27.3](#). Seleccionar *OK* guardará los valores actualizados en la base de datos, indicando luego, mediante otro cuadro de diálogo que se realizó la operación correctamente ([Figura 27.4](#)), o en su defecto ha ocurrido un error.

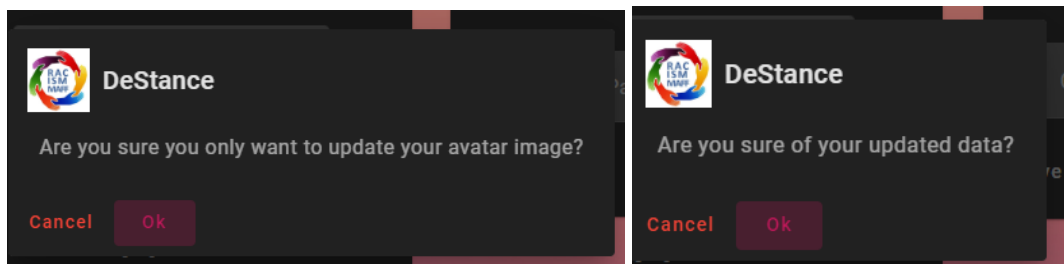


Figura 27.3: Confirmación de datos introducidos. A la izquierda solo Imagen Avatar y a la derecha datos personales o de autenticación.

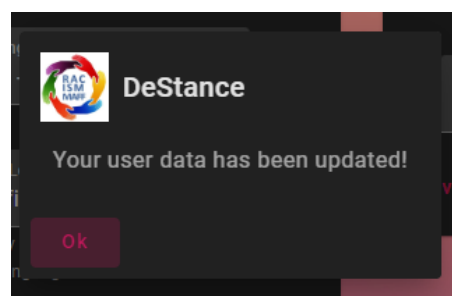


Figura 27.4: Confirmación de modificación.

X. Visualizar preguntas/respuestas Cuestionarios

Para acceder a la sección de pre o post-cuestionarios se debe seleccionar la opción correspondiente en el menú de navegación lateral izquierdo. En la

[Figura 28.1](#) se aprecia la sección de Pre-Cuestionarios en su pantalla inicial. A la izquierda se despliega una lista con los nombres de los cuestionarios. Para acceder a uno específico se debe hacer click sobre este. En la [Figura 28.2](#) se muestra el resultado de seleccionar el primer pre-cuestionario. Como el usuario no ha respondido dicho cuestionario todavía, se muestran las preguntas con las posibles respuestas para que este seleccione la que mejor se ajuste a su punto de vista. Para cambiar de cuestionario o cancelar las respuestas se debe seleccionar el botón *Change Questionnaire*, en la esquina inferior izquierda del menú de selección de cuestionarios.

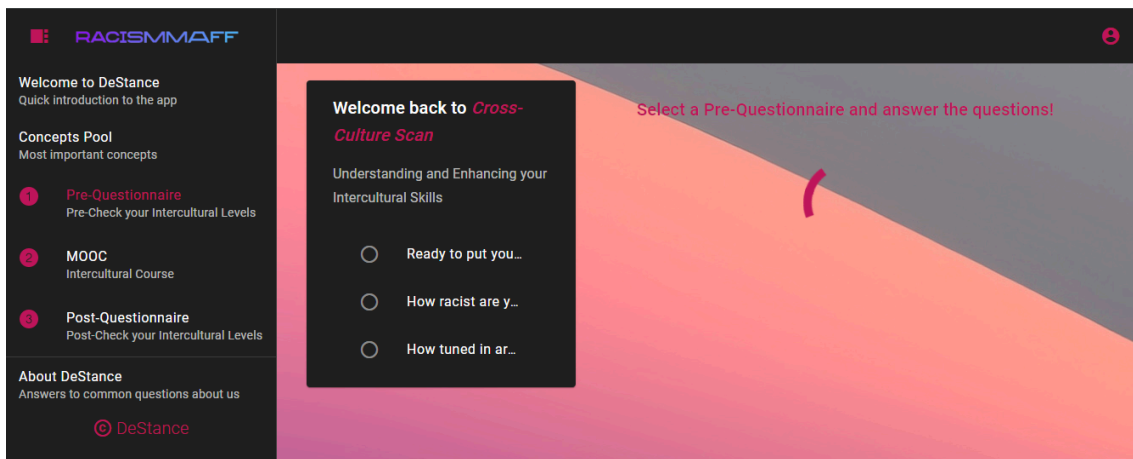


Figura 28.1: Pre-cuestionarios. Preguntas.

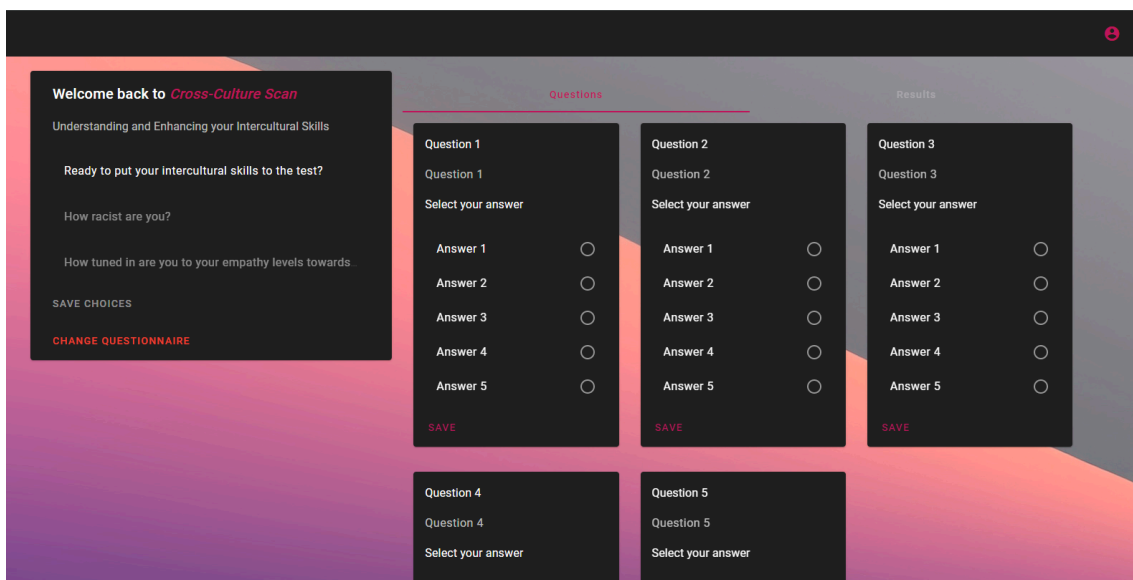


Figura 28.2: Pre-cuestionarios. Pre-Cuestionario específico.

Si el usuario hubiese respondido algún cuestionario, el icono del menú de selección de cuestionarios para dicho cuestionario cambia (de círculo vacío a comprobado), tal como se presenta en la [Figura 28.3](#). Para acceder a las respuestas dadas anteriormente al cuestionario basta con seleccionarlo. En la [Figura 28.4](#) se aprecian las respuestas anteriormente dadas por el usuario al cuestionario. Por defecto se muestra la subsección de las preguntas (*Questions*), por lo que para acceder a los resultados se debe seleccionar la opción *Results* tal como presenta la [Figura 28.5](#). Finalmente, es necesario aclarar que en las imágenes proporcionadas se observan datos de prueba.

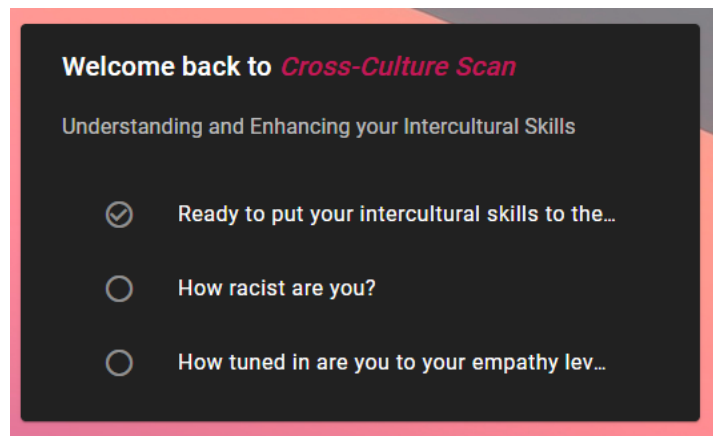


Figura 28.3: Pre-cuestionarios. Pre-Cuestionario respondido anteriormente.

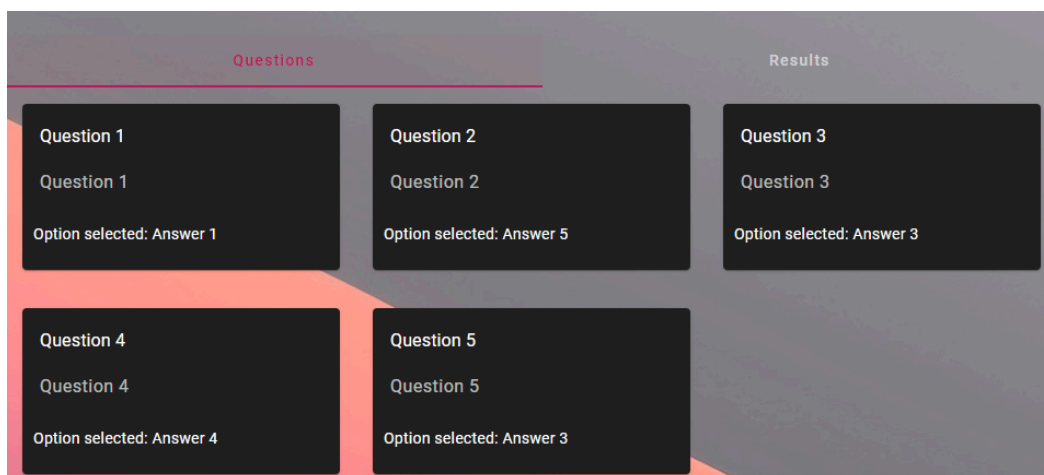


Figura 28.4: Pre-cuestionarios. Respuestas dadas anteriormente por pregunta.

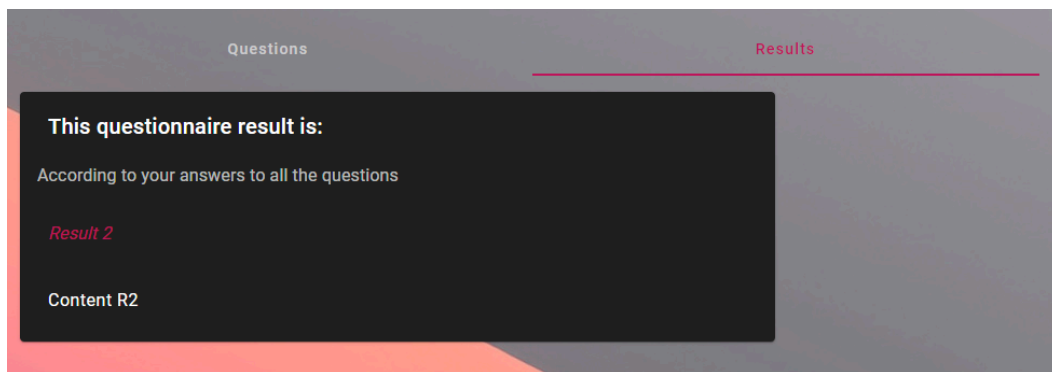


Figura 28.5: Pre-cuestionarios. Resultado obtenido.

XI. Responder Cuestionarios

Tomando como referencia la [Figura 28.2](#), para guardar una respuesta se debe seleccionar el botón *Save* que cada pregunta tiene en la parte inferior izquierda una vez elegida una opción válida, acción que se muestra en la [Figura 29.1](#). Si el usuario deseara cambiar su respuesta debe seleccionar el botón *Edit*.

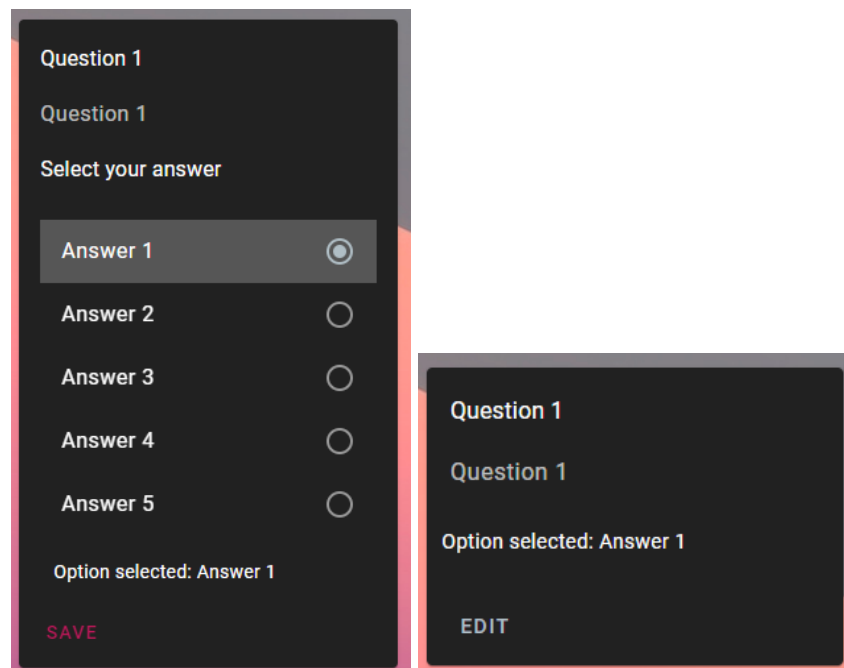


Figura 29.1: Responder preguntas de cuestionarios.

Una vez sean respondidas todas las preguntas el usuario debe seleccionar el botón *Save Choices*, ahora habilitado, en la esquina inferior izquierda del menú de selección de cuestionario, justo encima del botón *Change*

Questionnaire, tal como se muestra en la [Figura 29.2](#). Una vez hecho esto se proporcionará un cuadro de diálogo de confirmación de todas las respuestas introducidas ([Figura 29.3](#)), informando al usuario que no podrá modificarlas posteriormente. Seleccionar *Ok* guardará las respuestas en la base de datos, proporcionando la confirmación de éxito observada en la [Figura 29.4](#), o en su defecto error.

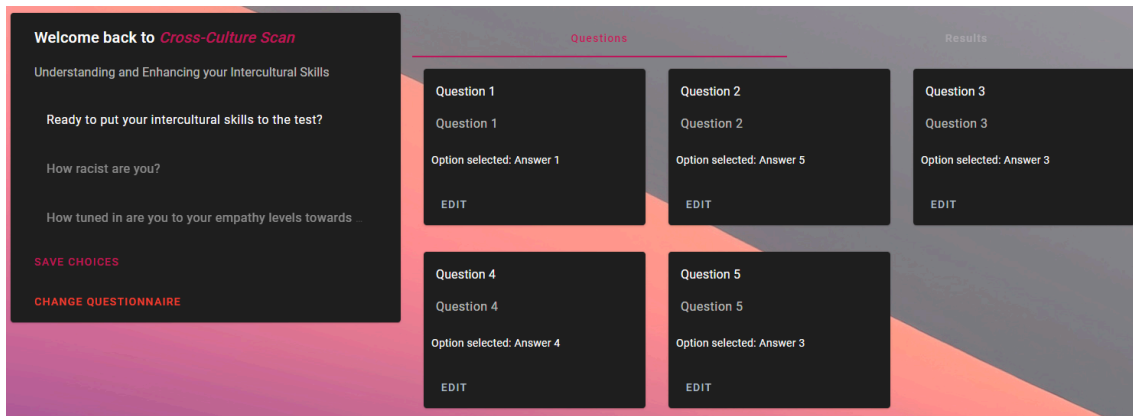


Figura 29.2: Responder TODAS las preguntas del cuestionario.

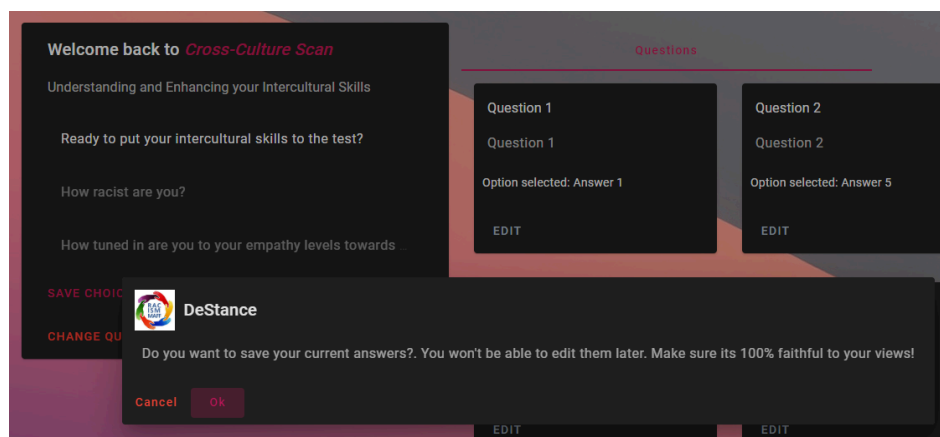


Figura 29.3: Confirmación de respuestas introducidas.

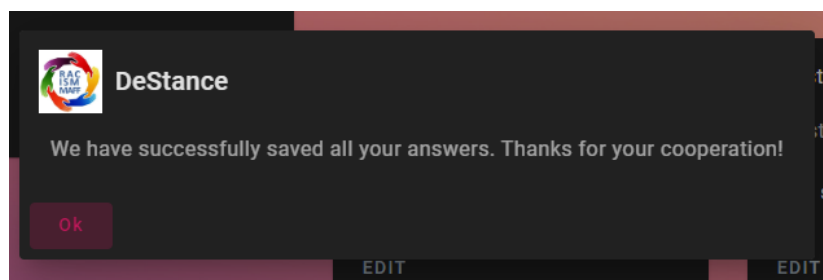


Figura 29.4: Confirmación de respuestas guardadas.

XV. Visualizar Dashboard Cuestionarios

Para poder visualizar el dashboard de los cuestionarios, el usuario deberá ser administrador y tener la sesión iniciada. El menú de navegación lateral ahora tendrá las opciones de administrador habilitadas ([Figura 30.1](#)). En dicho menú, en la sección de *Administration Dashboards* el usuario deberá seleccionar *Pre-Questionnaires* o *Post-Questionnaires*. El desarrollo de la opción *MOOC* le pertenece a otro estudiante.

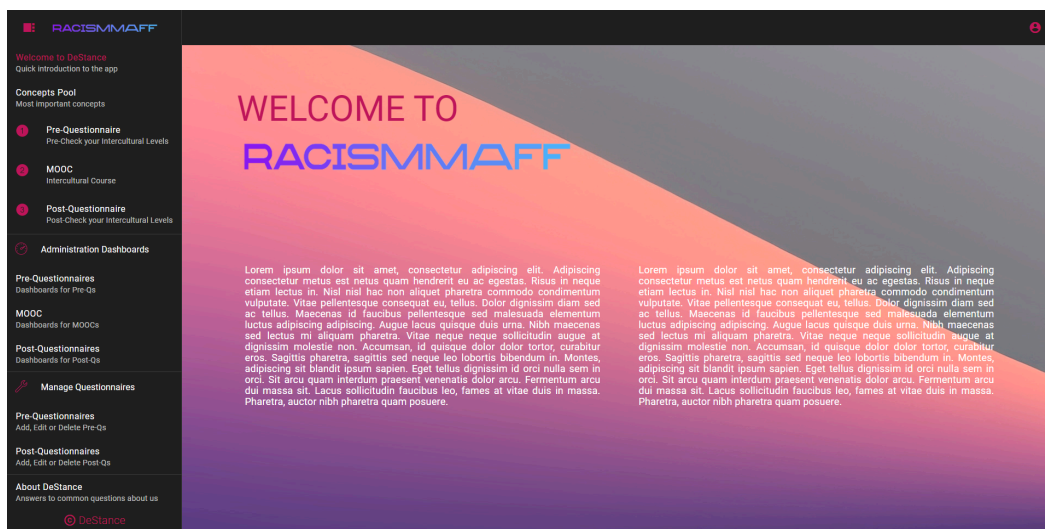


Figura 30.1: Pantalla inicial con opciones de administrador habilitadas.

Una vez seleccionado, por ejemplo *Post-Questionnaires*, se mostrará un menú de selección de cuestionarios ([Figura 30.2](#)) donde se deberá hacer click sobre uno de ellos para visualizar su dashboard. En la [Figura 30.3](#) se aprecia los gráficos correspondientes al segundo post-cuestionario. Los datos presentados corresponden a datos de prueba, dado que el objetivo es mostrar las estadísticas de las respuestas de los usuarios reales, para lo cual la aplicación debe estar disponible al público.

De cada cuestionario se presenta ([Figura 30.3](#)) un gráfico general de los resultados obtenidos por los usuarios, y seis gráficos por preguntas, a los cuales se puede acceder seleccionando las opciones de: *General*, *Age*, *Gender*, *Nationality*, *Mother Tongue* o *Language*, tal como se aprecia en la [Figura 30.4](#).

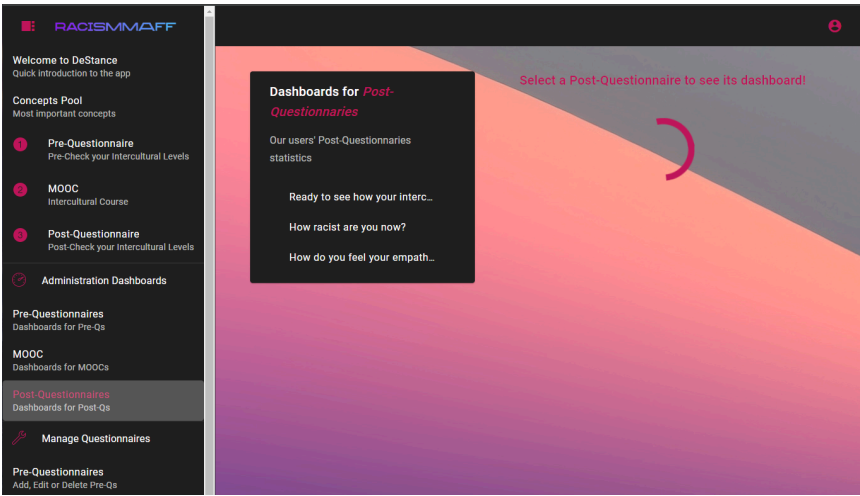


Figura 30.2: Dashboard Post-Cuestionarios. Selección de cuestionarios.

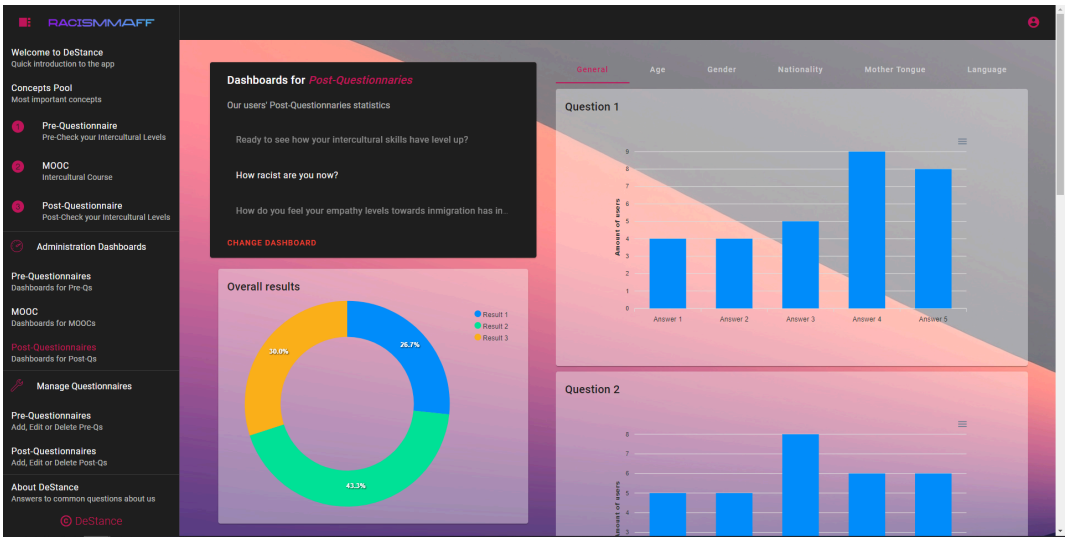


Figura 30.3: Dashboard de Post-Cuestionarios. Selección de uno de los cuestionarios.

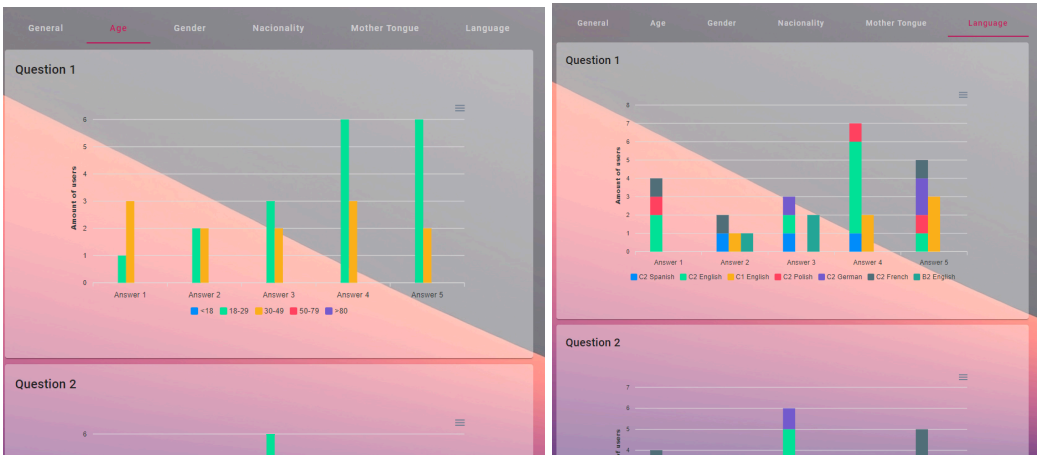


Figura 30.4: Dashboard de Post-Cuestionarios. Selección de diferentes tipos de gráficos por preguntas.

XVI. Gestionar Cuestionarios

Como en la mayoría de los casos anteriores, para acceder a la gestión de cuestionarios se debe seleccionar la opción correspondiente en el menú de navegación lateral izquierdo, en este caso en la subsección de *Manage Questionnaires*, escoger *Pre-Questionnaires* o *Post-Questionnaires*. En la [Figura 31.1](#) se muestra la acción anterior seleccionando *Pre-Questionnaires*.

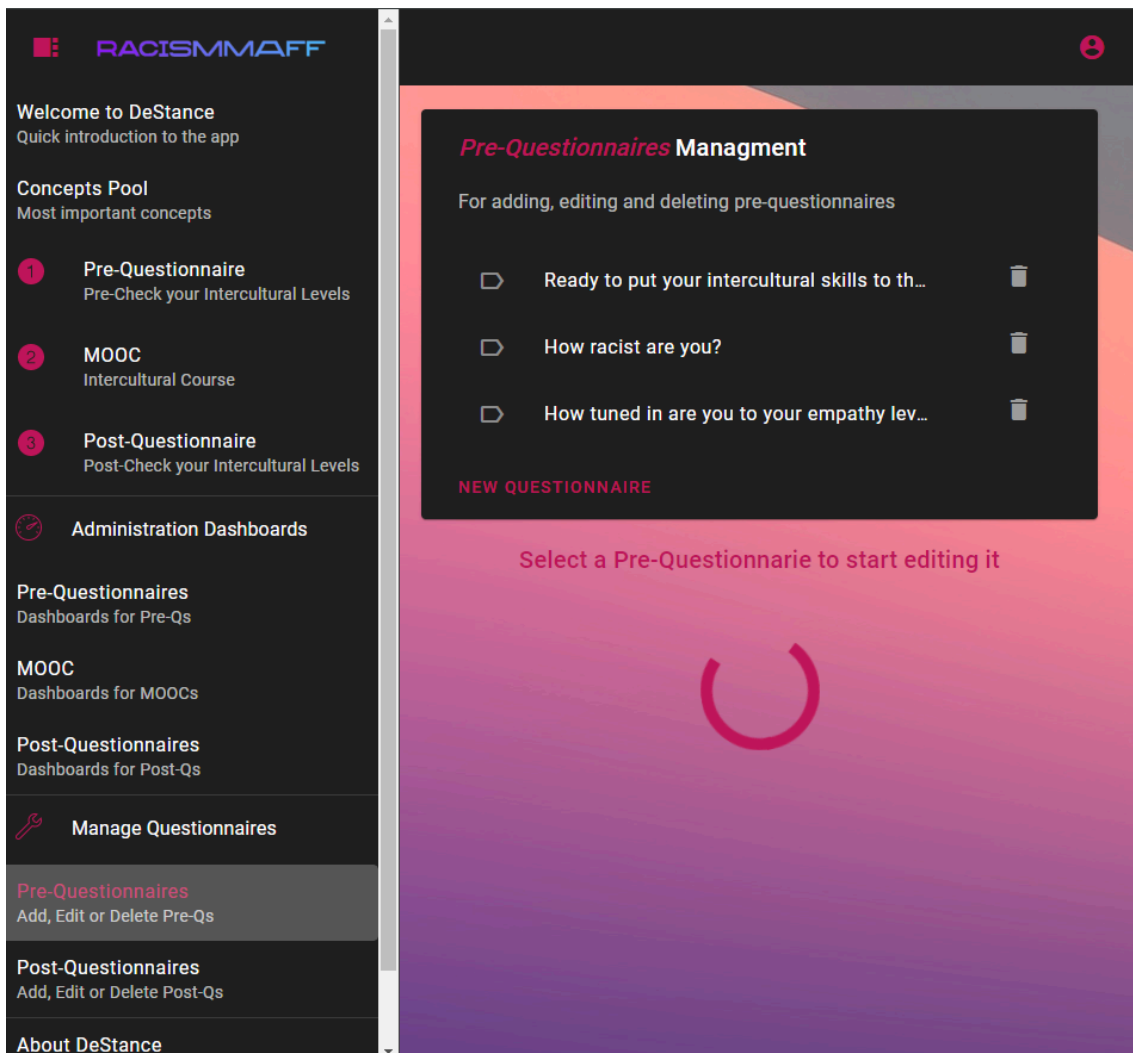


Figura 31.1: Gestión de Pre-Cuestionarios. Menú de Selección.

A partir de aquí el usuario puede optar por crear, editar o borrar un cuestionario. Para crear un nuevo cuestionario debe hacer click en el botón *New Questionnaire*, acción que lleva al usuario a la [Figura 31.2](#), donde tendrá que rellenar los datos pedidos, tanto en la sección de *Questions* como en la de

Results observadas en la [Figura 31.3](#). Para agregar nuevas preguntas y resultados se debe hacer click en el botón *Add a Question* o *Add a Result* en la sección correspondiente.

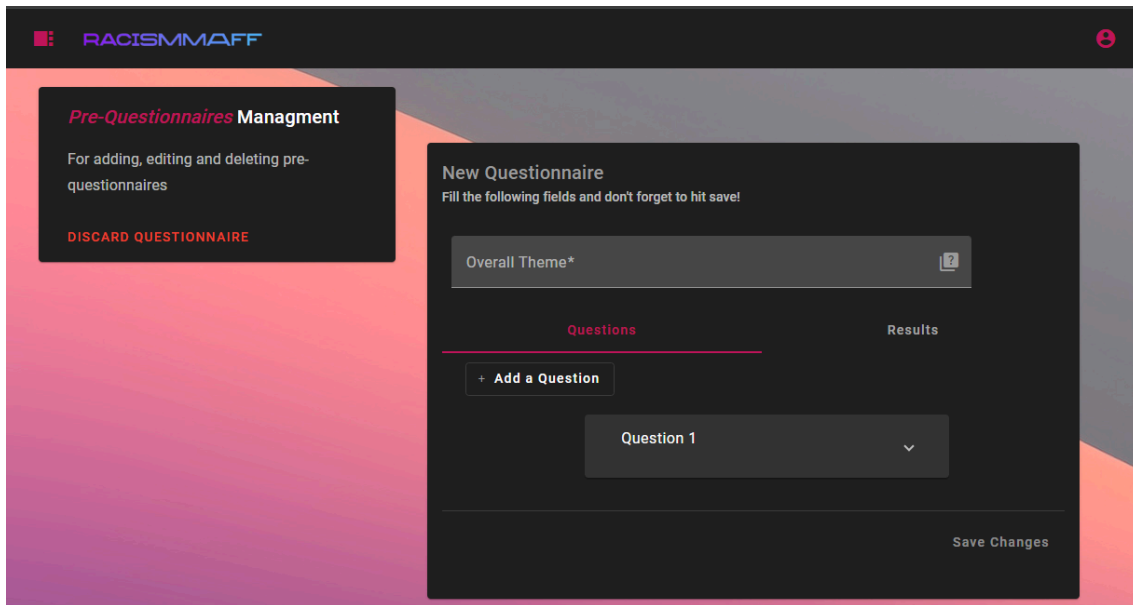


Figura 31.2: Gestión de Pre-Cuestionarios. Nuevo cuestionario.

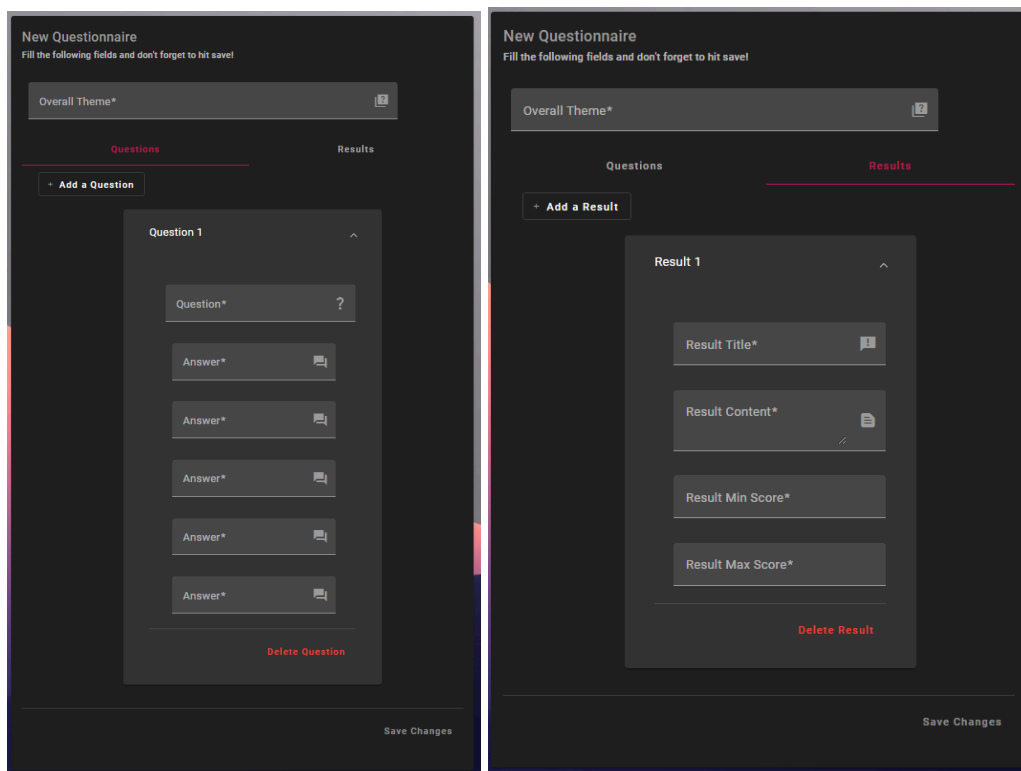


Figura 31.3: Nuevo Cuestionario. Preguntas y resultados.

Para desplegar/ocultar las preguntas o resultados basta con hacer click sobre ellos, y en el caso de querer eliminarlos seleccionar *Delete Question/Delete Result*. Una vez rellenos todos los campos se debe seleccionar el botón *Save Changes* (habilitado una vez se introduzcan valores válidos) para guardar el nuevo cuestionario en la base de datos, lo que conduce al cuadro de diálogo de la [Figura 31.4](#) (izquierda). Luego de seleccionar *Ok* el usuario podrá comprobar el éxito de la operación a través de su inclusión en el menú de selección de cuestionarios [Figura 31.4](#) (derecha). En caso de error se mostrará un cuadro de diálogo de error.

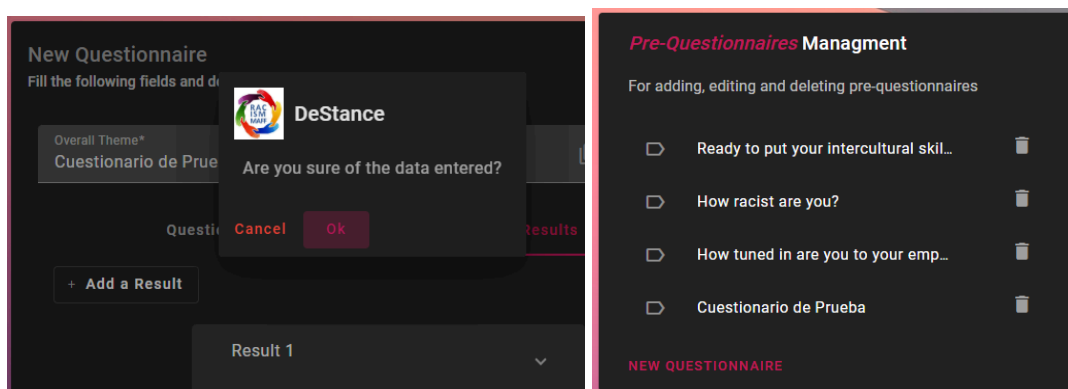


Figura 31.4: Nuevo Cuestionario. Confirmación de datos introducidos y de creación con éxito.

Para editar un cuestionario se debe seleccionar en el menú de gestión de cuestionarios haciendo click sobre este (evitar el icono de cesto de basura que es para eliminarlo). Esta acción conduce a una pantalla similar a la [Figura 31.2](#), esta vez rellenando los distintos campos con los datos actuales del cuestionario. Luego de modificar los campos deseados el usuario debe seleccionar el botón *Save Changes* y posteriormente *Ok* en el subsecuente cuadro de diálogo de confirmación ([Figura 31.5](#)). En la [Figura 31.6](#) se observa el resultado satisfactorio de la modificación del cuestionario respecto a [Figura 31.4](#) (derecha).

Por último, para eliminar un cuestionario basta con hacer click en el botón con icono de cesto de basura a la derecha del cuestionario ([Figura 31.6](#)). Posteriormente el usuario deberá seleccionar *OK* en el cuadro de diálogo de la [Figura 31.7](#) (izquierda), lo que deberá llevar a la [Figura 31.7](#) (derecha) donde se observaría que el cuestionario ha eliminado.

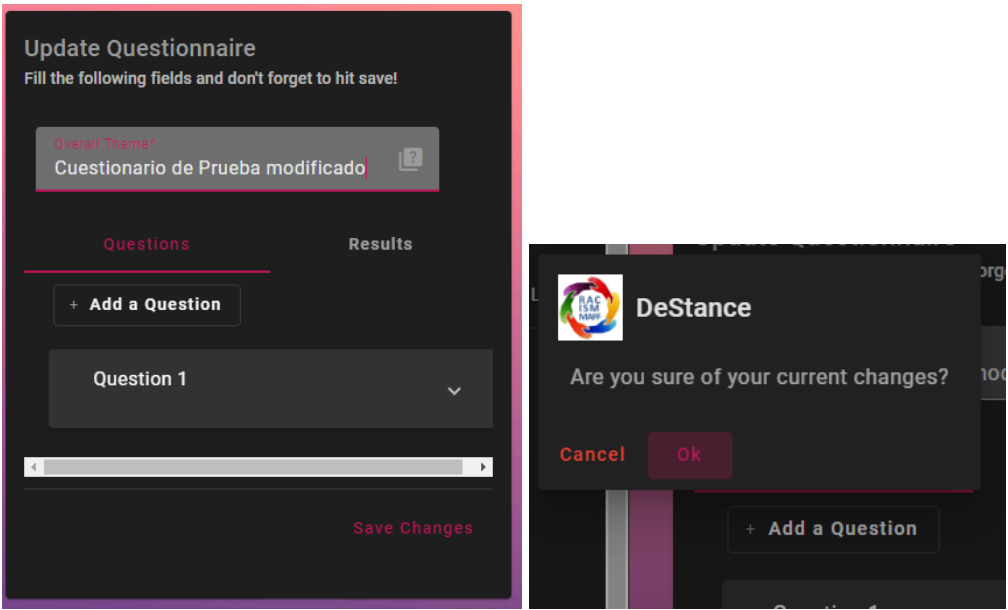


Figura 31.5: Edición Cuestionario. Confirmación de datos introducidos.

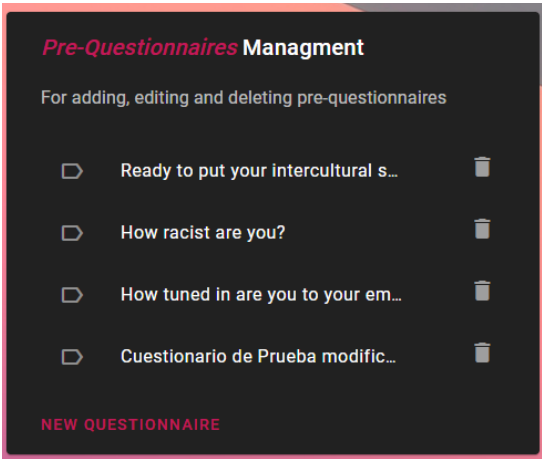


Figura 31.6: Edición Cuestionario. Confirmación de modificación de cuestionario.

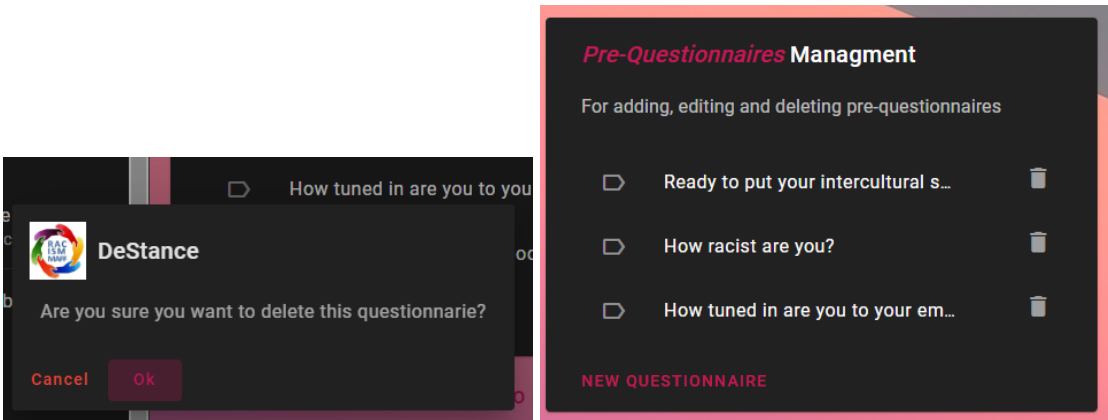


Figura 31.7: Eliminación Cuestionario. Confirmación de eliminación de cuestionario.

6.4. Estructura Final del Código

En este apartado se expone la estructura del código del proyecto a día 3 de junio de 2024. Dividido en dos proyectos (front-end y back-end) incluidos en la misma carpeta durante la fase de desarrollo, DeStance sigue la jerarquía de directorios y archivos que se presentan a continuación. Es importante tener en cuenta que el código seguirá evolucionando dado que todavía no está en su versión final.

6.4.1. Estructura Frontend

En la [Figura 32.1](#) se observa la jerarquía de directorios inicial de ambos proyectos (front-end y back-end). Al cliente web de DeStance (front-end) le corresponden todas las carpeta excepto *backend*. Todas las demás carpetas, excepto *src* corresponden a la configuración automática de un proyecto de Angular, donde destaca *node_modules* que contiene todas las dependencias instaladas del proyecto recogidas en el archivo *package.json*.

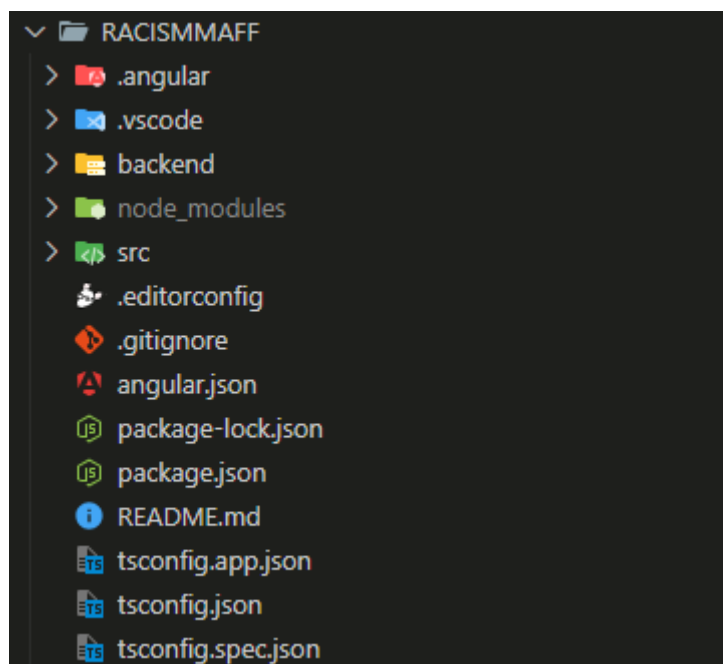


Figura 32.1: Jerarquía inicial de DeStance.

Dentro de la carpeta *src* ([Figura 32.2](#)) se encuentran los archivos *main.ts* y *styles.css*, así como la carpeta *assets* que no han sido modificados y forman

parte de la configuración inicial de Angular, mientras que *logo_RACISMMAFF.ico* corresponde al logo actual (3 de junio de 2024) de la aplicación y en el archivo *index.html* solo ha sido añadido la referencia a este último. La carpeta *app* ([Figura 32.3](#)) contiene los archivos de typescript, html y css que conforman los distintos componentes mencionados anteriormente en la [sección 5.4](#).

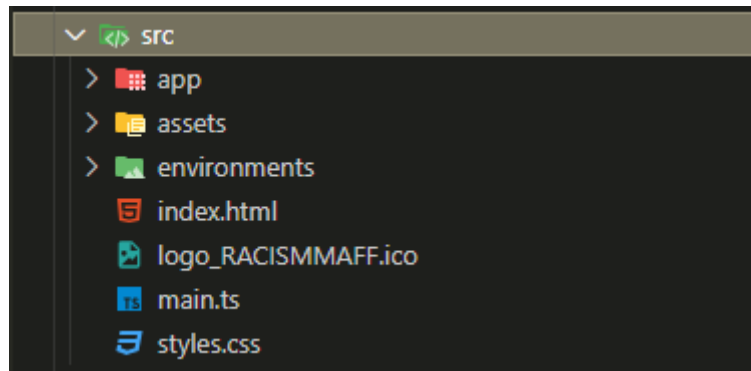


Figura 32.2: Jerarquía directorio src.

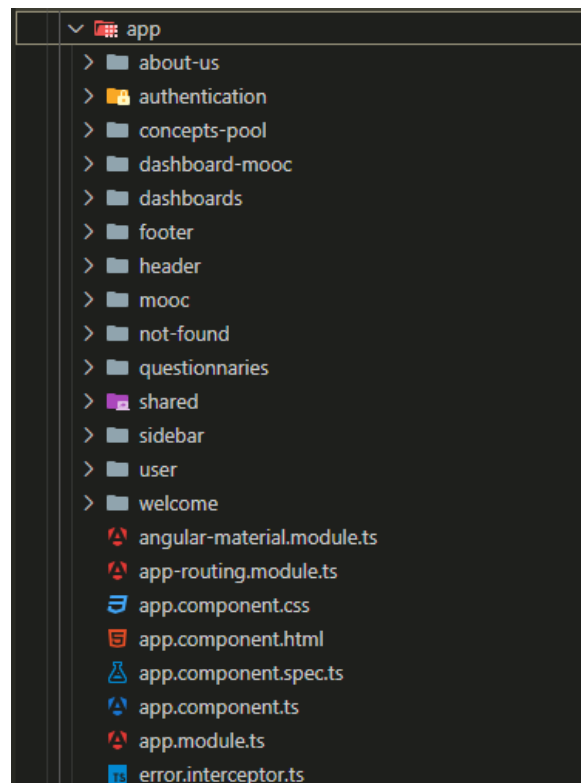


Figura 32.3: Jerarquía directorio app.

El componente *root* está formado por *app.component.ts*, *app.component.html*, *app.component.css* y *app.component.spec.ts*³. Además, *app.module.ts* contiene las dependencias internas de los módulos y componentes de la aplicación, mientras que *angular-material.module.ts* hace referencia a los módulos y componentes de Angular Material como forma de optimización en legibilidad del código. Por último *error.interceptor.ts* se utiliza para obtener los distintos mensajes de error de la aplicación referentes a peticiones HTTP. De la [Figura 32.4](#) a la [Figura 32.7](#) se muestran la estructura de directorios de los distintos componentes de la aplicación.

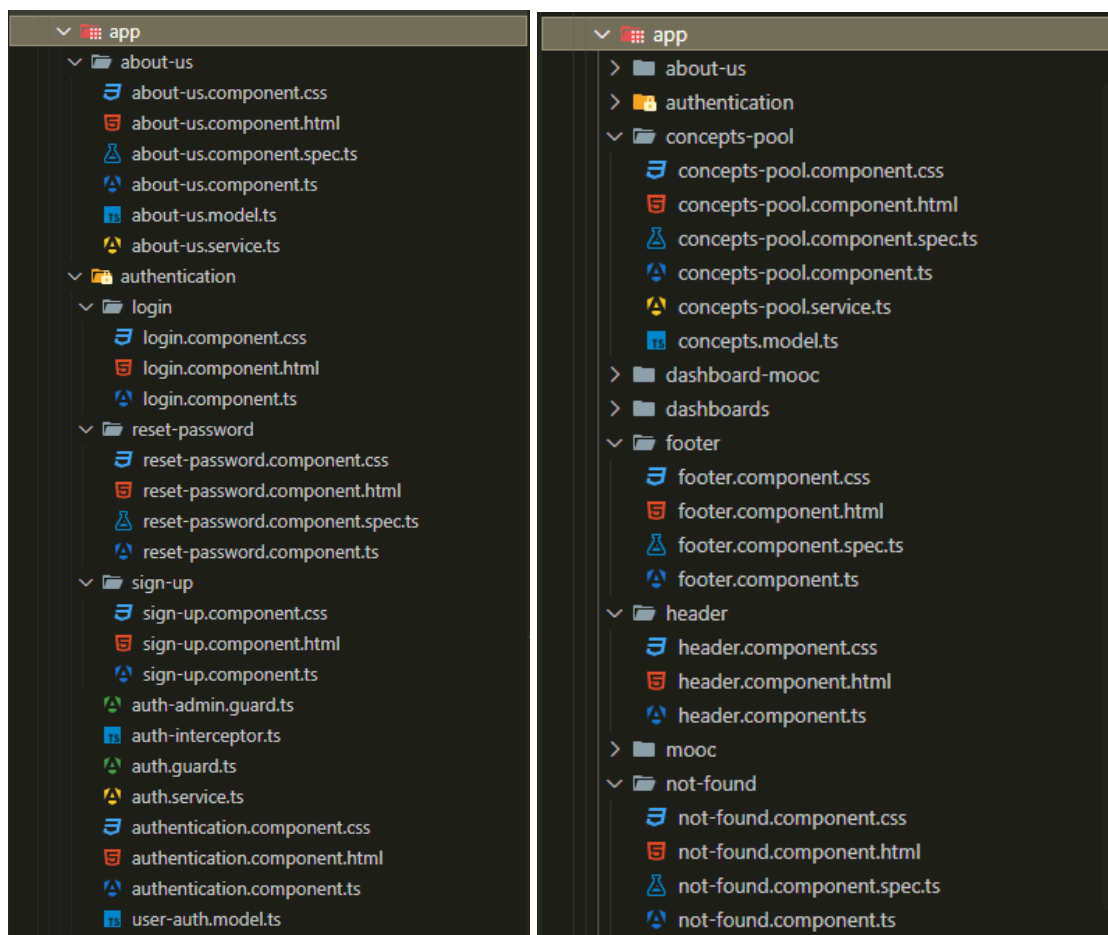


Figura 32.4: Componentes en el directorio app. About-Us, Authentication, Login, Sign-Up, Reset-Password, Concepts-Pool, Header, Not-Found.

³ Los archivos *.spec.ts* son necesarios para realizar pruebas a los componentes de Angular. Sin embargo, en esta versión de DeStance, no se modificaron los archivos por defecto porque las pruebas fueron realizadas al funcionamiento general de la aplicación durante su desarrollo.

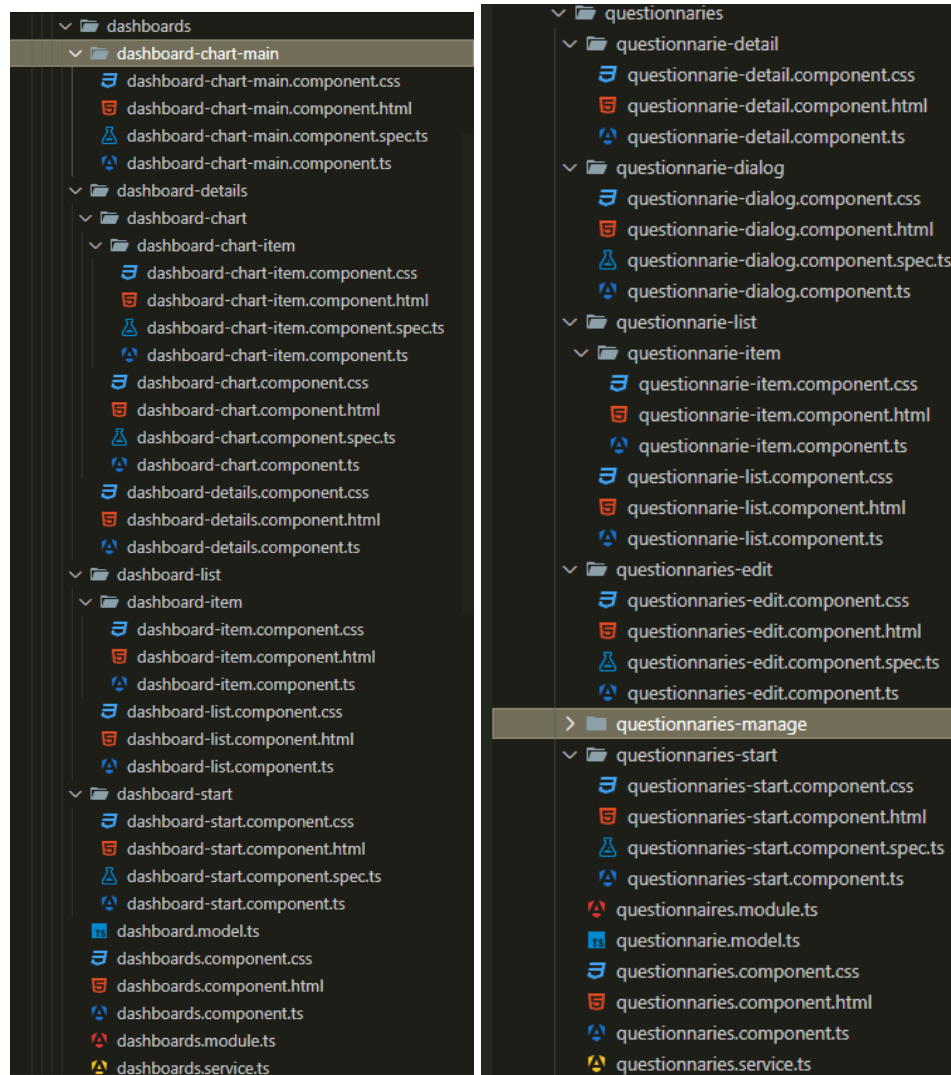


Figura 32.5: Componentes en directorio app. Dashboard de cuestionarios y sus subcomponentes. Cuestionarios y sus subcomponentes.

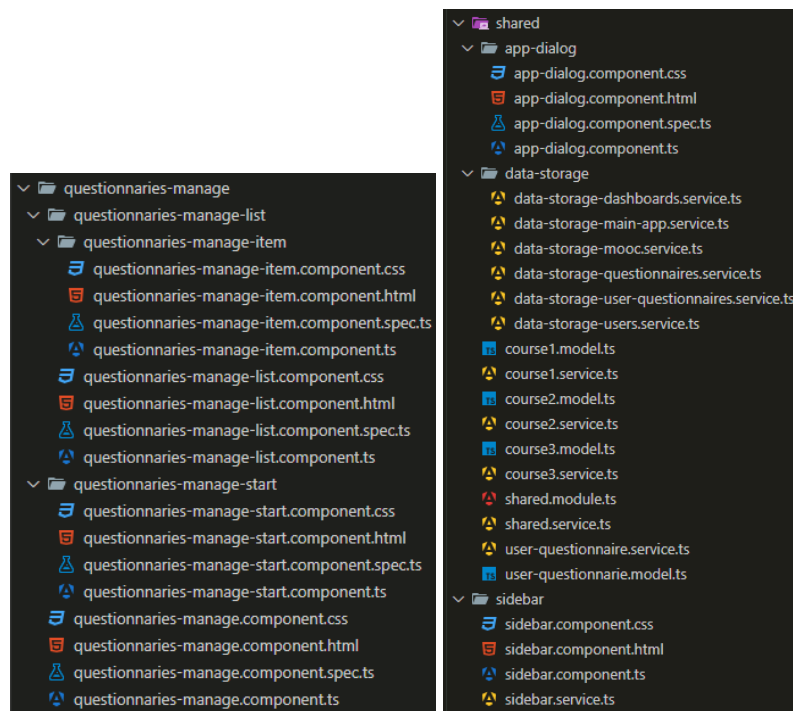


Figura 32.6: Componentes en directorio app. Gestión de Cuestionarios y sus subcomponentes. Componentes, servicios y modelos que se utilizan en distintas partes de la aplicación. Sidebar.

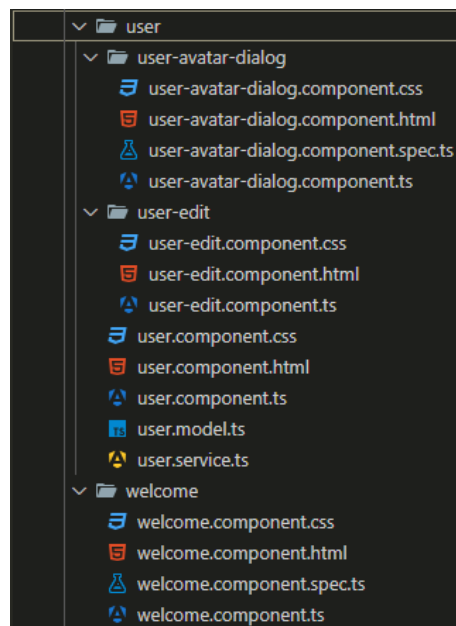


Figura 32.7: Componentes en directorio app. User y subcomponentes. Welcome.

6.4.2. Estructura Backend

La estructura de carpetas del backend fue comentada anteriormente en la [sección 5.5.2](#).

7. RESULTADOS Y CONCLUSIONES

Tomando en cuenta el apartado anterior ([Sección 6](#)), se puede concluir que, al menos en las tareas objetivo de este Trabajo de Fin de Grado, DeStance se encuentra en un estado casi final de la implementación de cara a empezar su uso. En su estado actual la aplicación permite consultar las páginas de bienvenida, de conceptos, preguntas frecuentes, dashboards de cuestionarios, cuestionarios, gestionarlos y responderlos, acceder a los datos personales y modificarlos, así como darse de alta en la aplicación y autenticarse, incluido la diferenciación de acceso a funcionalidades entre los distintos roles de usuario (no identificado, usuario identificado y administrador).

A día 3 de junio de 2024, DeStance le siguen faltando datos reales en las secciones de cuestionarios, preguntas frecuentes y conceptos, además del diseño de la página de bienvenida y del logo de la aplicación que actualmente hace referencia a su antiguo nombre RACISMMAFF. Durante las próximas dos semanas, a través de reuniones con la tutora y otros investigadores del proyecto RACISMMAFF [\[1\]](#) se deberían proporcionar estos datos para finalmente hacer disponible el proyecto al público. Las respuestas reales de usuarios reales serán las que se tomarán en cuenta para calcular las estadísticas de los dashboards, por lo que estos deberán ser reseteados respecto a los datos actuales que muestran respuestas de prueba a los cuestionarios.

Hasta ahora, el desarrollo del proyecto general se ha realizado por dos estudiantes distintos, la autora de este Trabajo de Fin de Grado y un compañero del Grado de Ingeniería Informática, ambos con tareas distintas pero complementarias, por lo que se ha utilizado un repositorio privado conjunto en GitHub [\[38\]](#) para mantener el código sincronizado. Esta plataforma ha permitido trabajar independientemente, así como documentar y comprender los avances de cada uno en el código a través de los mensajes de commit y de la propia página web del repositorio que ofrece un historial de estos, con los cambios realizados en cada momento.

Como se mencionó anteriormente y además en la [sección 6.2](#) uno de los puntos críticos en DeStance ha sido la dependencia de los datos reales que deben proporcionar los investigadores de RACISMMAFF [\[1\]](#). Respecto al trabajo colaborativo con ellos, se ha ido exponiendo los avances en el desarrollo de DeStance a través de reuniones cada dos semanas aproximadamente, obteniendo adicionalmente los datos reales actuales por dicha vía o por intercambio de correos.

Otra dificultad presentada fue la decisión de codificar primero el front-end, y luego enfocarse en el back-end. Resultó en una pérdida de tiempo porque se necesitó codificar de más para probar las funcionalidades de la página web. Esto ocurrió porque los conocimientos de Node.js y Express.js eran más bajos que de Angular, además de que en todo momento fueron consultados tutoriales online de la plataforma Udemy [\[22\]](#) con una estructura predefinida, empezando por uno de sólo Angular, para pasar a uno de MEAN stack complementado con otro de Node.js.

En cuanto a dificultades en el desarrollo de las funcionalidades, no se produjeron grandes baches de implementación, siendo esta bastante dinámica gracias a los tutoriales antes mencionados, y la experiencia previa tanto de programación web como en general. Quizás lo más difícil fue conseguir un diseño que fuera atractivo a opinión de la tutora y el resto de los clientes (investigadores de RACISMMAFF [\[1\]](#)). Primero se opta por uno de los estilos predeterminados de Angular Material v16, *Deep Purple & Amber*, a lo que la tutora pide más personalización.

La tutora envía como guía una plantilla de colores y diseño muy general. Fue imposible integrar esos colores porque Angular Material necesita mucho más nivel de detalle como, por ejemplo, cuatro tipos de colores que deben contrastar lo suficiente para cumplir las reglas de accesibilidad. Esto último se denomina *theme* y debe ser programado de forma global, no componente a componente, los cuales además lucen de una forma predeterminada que no se puede modificar. Precisamente fue escogida esta librería para no perder mucho

tiempo en personalización y enfocarse en que al final se tuviese un producto funcional.

Lo anterior es explicado en forma de correo electrónico a la tutora y demás investigadores de RACISMMAFF, lo que conduce sin problemas al diseño actual de la aplicación, con la utilización del *theme* de Angular Material v16 denominado *Pink & Blue-grey*. Si bien se opta por oscurecer la mayoría de componentes al color #212121 para mejorar el contraste y obtener un diseño más atractivo a opinión subjetiva de la autora de este TFG, pero que luego fue aprobado y cumplimentado por los clientes.

Actualmente, cada vez que un usuario responde un cuestionario, el dashboard correspondiente se actualiza automáticamente en tiempo real. Este proceso asegura que los administradores y usuarios obtengan información actualizada inmediatamente después de cada interacción. Sin embargo, existe una posibilidad muy baja, pero no nula, de que dos o más usuarios respondan simultáneamente, en el mismo milisegundo, lo que podría causar problemas de concurrencia. En tales casos, podría suceder que el sistema solo registre una de las respuestas debido a que todos los usuarios comparten la misma versión del dashboard en ese momento.

Este problema de concurrencia es altamente improbable dado que se espera que el tráfico de la aplicación sea relativamente bajo en términos de usuarios concurrentes. Aun así, para asegurar la integridad y precisión de los datos, se podría implementar un método para contabilizar todas las respuestas de los usuarios directamente desde la base de datos. Esto implicaría diseñar una función específica en el servidor de back-end que recopile y recuento las respuestas de los usuarios de manera periódica.

Para minimizar la carga en el sistema y garantizar que los cálculos se realicen de manera eficiente, esta función podría programarse para ejecutarse en intervalos regulares, como una vez al día. Este enfoque no sólo resolvería cualquier posible problema de concurrencia, sino que también proporcionaría un método robusto para manejar grandes volúmenes de datos a medida que la cantidad de respuestas de los usuarios aumenta con el tiempo. Implementar

esta solución aseguraría que los dashboards reflejen siempre la información más precisa y actualizada, beneficiando a los administradores de la aplicación.

En general, este ha sido el proyecto más ambicioso en cuanto a la cantidad de líneas de código, horas dedicadas y esfuerzo requerido. Aunque se han agotado los plazos disponibles para desarrollar la plataforma, confío en que el departamento esté satisfecho con el resultado y que este sea útil para RACISMMAFF [\[1\]](#). Finalmente, quiero expresar mi agradecimiento a mi tutora y a los colaboradores por su excelente disposición, la cual ha hecho que la realización de este Trabajo Fin de Grado haya sido una experiencia muy gratificante y enriquecedora.

8. ANÁLISIS DE IMPACTO

DeStance está concebida como una plataforma integral que permite la administración y almacenamiento de subcorpora escritos en diferentes lenguas, así como la implementación y gestión de cuestionarios interactivos para medir las capacidades interculturales de los estudiantes universitarios europeos a distintos niveles, adaptados para ser aplicados antes y después de un Curso Online Masivo y Abierto (MOOC) que busca afectar de manera positiva el discurso relacionado inmigración y racismo por parte de los estudiantes universitarios y futuros profesores de la Unión Europea (UE).

DeStance surge como un complemento al proyecto RACISMMAFF [\[1\]](#) liderado por investigadores de la Universidad Complutense de Madrid y la Universidad Politécnica de Madrid. Ambos, por tanto, están estrechamente relacionados con el objetivo de reducción de desigualdades de la Agenda 2030 para el Desarrollo Sostenible. Este objetivo, conocido como ODS 10, busca reducir la desigualdad dentro de los países y entre ellos, promoviendo la inclusión social, económica y política de todas las personas, independientemente de su edad, sexo, discapacidad, raza, etnia, origen, religión, situación económica u otra condición.

En este contexto, RACISMMAFF y DeStance se centran en abordar y mitigar las desigualdades derivadas de la inmigración y el racismo, dos factores que contribuyen significativamente a la desigualdad social y económica. RACISMMAFF, a través de sus investigaciones y actividades, busca sensibilizar y educar a los estudiantes universitarios y futuros profesores sobre las realidades y desafíos de la inmigración y el racismo. DeStance, como plataforma complementaria, proporciona herramientas y recursos educativos que refuerzan estos objetivos, facilitando el aprendizaje y la reflexión crítica sobre estos temas.

Además, DeStance y RACISMMAFF también contribuyen al ODS 10 al influir en el discurso académico y social sobre la inmigración y el racismo. Al educar a los estudiantes y futuros profesores, estos proyectos ayudan a crear un entorno más informado y sensible, donde se reconocen y abordan las

desigualdades de manera proactiva. Esto no solo beneficia a los individuos directamente involucrados, sino que también tiene un impacto positivo en la sociedad en general, promoviendo la cohesión social y reduciendo las tensiones relacionadas con la inmigración y la diversidad.

Adicionalmente, los cuestionarios interactivos de DeStance permiten medir el impacto que ha tenido el MOOC en los estudiantes que lo han completado. En primer lugar, a través de los pre-cuestionarios se obtiene una visión general de la posición del estudiante respecto a la inmigración y el racismo, brindando una línea base que permite evaluar sus actitudes y conocimientos iniciales. Estos pre-cuestionarios son fundamentales para identificar las percepciones y posibles sesgos que los estudiantes pueden tener antes de iniciar el curso.

Una vez completado el MOOC, los estudiantes responden a los post-cuestionarios, los cuales están diseñados para evaluar cualquier cambio en sus actitudes y conocimientos. Esto permite medir cuánto ha cambiado la perspectiva del estudiante sobre la inmigración y el racismo gracias al contenido educativo y las actividades interactivas del MOOC. Los post-cuestionarios ofrecen una valiosa comparación con los resultados iniciales, proporcionando datos concretos sobre la efectividad del curso.

Además, los dashboards de la aplicación permiten a los administradores, es decir, los investigadores de RACISMMAFF, obtener estadísticas detalladas sobre los resultados de los pre y post-cuestionarios. Estas herramientas visuales facilitan la interpretación de los datos recopilados, mostrando tendencias y patrones en el cambio de actitudes y conocimientos entre los estudiantes. Las estadísticas generadas por los dashboards son de gran relevancia, ya que permiten a los investigadores realizar un análisis profundo de la eficacia del MOOC.

La recolección de estos datos no solo es crucial para evaluar el impacto inmediato del curso, sino que también proporciona una base sólida para futuros trabajos de investigación. Los datos derivados de los cuestionarios interactivos y su análisis a través de los dashboards permitirán a los investigadores de RACISMMAFF desarrollar estudios detallados sobre la

educación en temas de inmigración y racismo. Estos pueden explorar diversas dimensiones, como la variación en el impacto del MOOC según diferentes grupos demográficos, la durabilidad de los cambios de actitud a lo largo del tiempo, y la identificación de los elementos del curso que resultaron más efectivos.

Es por ello que se evidencia que los cuestionarios interactivos de DeStance y los dashboards asociados no solo miden el impacto del MOOC en los estudiantes, sino que también proporcionan datos valiosos para la investigación continua. Esta capacidad de evaluación y análisis es fundamental para mejorar continuamente los contenidos educativos y maximizar su impacto positivo en la reducción de desigualdades relacionadas con la inmigración y el racismo, alineándose así con el objetivo ODS 10 de la Agenda 2030 para el Desarrollo Sostenible.

9. LÍNEAS DE TRABAJO FUTURAS

En este punto de implementación de DeStance, si bien se ha conseguido desarrollar todas las prestaciones requeridas, la aplicación cuenta con gran potencial para mejorar y ampliar sus funcionalidades. A continuación se describen unas líneas de trabajo futuras que puedan servir de guía para estos desarrollos:

- **Mejora del estilo de la interfaz:** Aunque se ha realizado un gran esfuerzo en este aspecto, y este ha sido cumplimentado por los clientes de DeStance, así como la tutora, siempre se puede lograr más personalización abandonando los componentes de Angular Material y creando unos propios utilizando el Lenguaje de Hojas de Estilos de Cascada (CSS). Desde luego esta tarea tiene un nivel alto de complejidad, por lo que se necesitaría a personas con mucha experiencia en el lenguaje css.
- **Acceso a respuestas de usuarios:** Actualmente la aplicación guarda las respuestas a los cuestionarios de los usuarios en la base de datos, y las utiliza para calcular las estadísticas pertinentes en los dashboards. Sin embargo, no hay una forma de acceder a las respuestas individuales desde DeStance. Se necesitaría crear una nueva sección para administradores donde se pueda acceder a las respuestas de cada usuario, o integrar esto como otra funcionalidad en el dashboard de los cuestionarios.
- **Gestión de Conceptos y Preguntas Frecuentes:** Para facilitar la mejora de los contenidos de la aplicación se podría considerar añadir una sección de gestión de Conceptos y Preguntas Frecuentes que permita a los administradores (investigadores de RACISMMAFF) crear, modificar o eliminar conceptos y preguntas frecuentes.
- **Relación más estricta entre cuestionarios:** Una vez se desarrolle el contenido de los cuestionarios por parte de los investigadores de RACISMMAFF, podría añadirse una relación de dependencia entre pre y

post-cuestionarios, de forma que se necesite haber respondido el pre-cuestionario antes de iniciar a responder el post-cuestionario. Para ello se necesitaría añadir cuidadosamente el nuevo atributo en la base de datos a los cuestionarios ya creados que han sido respondidos, de forma que no haya que eliminarlos para evitar perder las respuestas. Podría conseguirse a través de la actualización de los campos a rellenar en la sección de modificación de cuestionarios, permitiendo actualizar los cuestionarios desde la misma aplicación.

- **Vista de móvil:** Aunque la aplicación está pensada para navegadores web y se ha tratado de adaptar los componentes a pantallas pequeñas tamaño móvil, no estaría de más dedicar más tiempo a asegurarse de que la experiencia del usuario en móvil sea tan satisfactoria como en ordenador (pantallas más grandes).
- **Actualizar datos estadísticos de los dashboards:** Actualmente, cada vez que un usuario responde un cuestionario, el dashboard correspondiente se actualiza automáticamente. Existe una probabilidad muy baja de que dos o más usuarios respondan en el mismo milisegundo o menos, lo que podría causar problemas de concurrencia y actualizar solo una respuesta debido a la misma versión del dashboard. Aunque este escenario es improbable debido al bajo tráfico concurrente, se puede evitar mediante un recuento periódico de todas las respuestas en la base de datos. Para ello, se debe crear una función en el servidor de back-end que se ejecute, por ejemplo, una vez al día, para manejar estos cálculos de manera eficiente.

10. REFERENCIAS BIBLIOGRÁFICAS

- [1] Universidad Complutense Madrid. *"Stance strategies in immigration and racism-related discourse: Analysis and applications in affective learning practices (RACISMMAFF)"*. [Online]. Available: <https://www.ucm.es/racismmaff/about-the-project>. [Accessed: 2024, 04, 18]
- [2] Universidad Complutense Madrid. *"The expression of Evidentiality and Modality in English and other European Languages: Cross-linguistic perspectives. (EUROEVIDMOD)"*. [Online]. Available: <https://www.ucm.es/euroevidmod/objectives>. [Accessed: 2024, 04, 18].
- [3] V. Martín de la Rosa y E. Domínguez Romero. *Epistemic and non-epistemic modals: The key to interpreting the spirit of counter-terrorism United Nations Security Council resolutions*, Journal of Pragmatics, Volume 180, 2021, pp. 89-101, ISSN 0378-2166. [Online]. Available: <https://doi.org/10.1016/j.pragma.2021.04.009>.
- [4] J. I. Marín Arrese. "STANCEDISC," 16-Sep-2021. [Online]. Available: osf.io/2qzrm. [Accessed: 2024, 04, 18].
- [5] B. Rodríguez-Romero, F. R. Gómez, y A. Quintela-del-Río, *"Formularios Google como recurso educativo"* en Contextos universitarios transformadores: Boas prácticas no marco dos GID. Servizo de Publicacións, 2020. pp. 575-578.
- [6] S. Melo, *"¿Cuáles son las ventajas y desventajas de los formularios de Google?"* DataScope, 2021. [Online]. Available: <https://datascope.io/es/blog/ventajas-y-desventajas-de-formularios-google/>. [Accedido: 2024,04,20].
- [7] Colaboradores de Wikipedia. *"Moodle"*. Wikipedia, La enciclopedia libre, 2024. Disponible en <https://es.wikipedia.org/w/index.php?title=Moodle&oldid=159536476>. [Online]. [Disponible en: 2024, 04, 18].
- [8] SendGrid, *"SendGrid: Cloud-Based Email Delivery Service,"*. [Online]. Available: https://sendgrid.com/en-us?utm_source=google&utm_medium=cpc&utm_term=sendgrid&utm_campaign=G_S_EMEA_TSG_Brand_T2_NV&utm_plac=&utm_net=g&utm_pos=&utm_med=&utm_plt=gp&utm_gad_source=1&utm_gclid=CjwKCAjwo6GyBhBwEiwAzQTmc7BohyypBtK1sB2bwyGu2Z6KsrzvMILO8MKr9t6CONRK_OqEQsN7vRoC46EQAvD_BwE. [Accedido: 2024, 05, 18].
- [9] Google Firebase, *"Firebase: App Development Platform,"*. [Online]. Available: <https://firebase.google.com/?hl=es>. [Accedido: 2024, 05, 18].
- [10] Angular, *"Angular - The modern web developer's platform,"*. [Online]. Available: <https://angular.io/>. [Accedido: 2024, 05, 18].
- [11] Node.js, *"Node.js,"*. [Online]. Available: <https://nodejs.org/en>. [Accedido: 2024, 05, 18].
- [12] Express.js, *"Express - Node.js web application framework,"*. [Online]. Available: <https://expressjs.com/es/>. [Accedido: 2024, 05, 18].

[13] MongoDB, "MongoDB Atlas - Database as a Service". [Online]. Available: https://www.mongodb.com/es/lp/cloud/atlas/try4?utm_source=google&utm_campaign=search_gs_pl_evergreen_atlas_core-high-int_prosp-brand_gic-null_emea-es_ps-all_desktop_eng_lead&utm_term=mongodb%20atlas&utm_medium=cpc_paid_search&utm_ad=e&utm_ad_campaign_id=19609093152&adgroup=145580477757&cq_cmp=19609093152&gad_source=1&gclid=CjwKCAjwo6GyBhBwEiwAzQTmc9RTU2q22WHegyiYJ27JoIm7OTlolF4R3mcTe9MDxI10bY09SpVMAxoCONQQAxD_BwE?utm_source=google&utm_campaign=search_gs_pl_evergreen_atlas_core-high-int_prosp-brand_gic-null_emea-es_ps-all_desktop_eng_lead&utm_term=mongodb%20atlas&utm_medium=cpc_paid_search&utm_ad=e&utm_ad_campaign_id=19609093152&adgroup=145580477757&cq_cmp=19609093152&gad_source=1&gclid=CjwKCAjwo6GyBhBwEiwAzQTmc9RTU2q22WHegyiYJ27JoIm7OTlolF4R3mcTe9MDxI10bY09SpVMAxoCONQQAxD_BwE.

[Accedido: 2024, 05, 18].

[14] W3C, "Pautas de Accesibilidad para el Contenido Web (WCAG) de un vistazo,". [Online]. Available: <https://www.w3.org/WAI/standards-guidelines/wcag/glance/es>.

[Accedido: 2024, 05, 18].

[15] "Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo de 27 de abril de 2016 relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos (Reglamento general de protección de datos)," EUR-Lex. [Online]. Available: <https://eur-lex.europa.eu/legal-content/ES/TXT/?uri=celex%3A32016R0679>.

[Accedido: 2024, 05, 18].

[16] HOQUE, Shama. *Full-Stack React Projects: Learn MERN stack development by building modern web apps using MongoDB, Express, React, and Node.js*. Packt Publishing Ltd, 2020.

[17] LEE, James; WARE, Brent. *Open Source Web Development with LAMP: Using Linux, Apache, MySQL, Perl, and PHP*. Addison-Wesley Professional, 2003.

[18] MELÉ, Antonio. *Django 5 By Example: Build powerful and reliable Python web applications from scratch*. Packt Publishing Ltd, 2024.

[19] HAVIV, Amos Q. *MEAN Web Development*. Packt Publishing Ltd, 2016.

[20] Stack Overflow, "Stack Overflow Developer Survey 2023,". [Online]. Available: <https://survey.stackoverflow.co/2023/#overview>. [Accedido: 2024, 05, 19].

[21] L. Blanchard, "Stack Overflow CEO on how it became the world's most popular programming site," ZDNet, Feb. 28, 2022. [Online]. Available: <https://www.zdnet.com/article/stack-overflow-ceo-on-how-it-became-the-worlds-most-popular-programming-site/>. [Accedido: 2024, 05, 19].

[22] Udemy, "Udemy: Online Courses,". [Online]. Available: <https://www.udemy.com/>. [Accedido: 2024, 05, 19].

[23] Angular Material, "Angular Material,". [Online]. Available: <https://material.angular.io/>. [Accedido: 2024, 05, 19].

- [24] M. Salazar, "To Angular Material or Not to Angular Material," Medium. [Online]. Available: <https://marasalazar.medium.com/to-angular-material-or-not-to-angular-material-940a616ac22d>. [Accedido: 2024, 05, 19].
- [25] Material-UI, "*Material-UI: React components for faster and easier web development*,". [Online]. Available: <https://mui.com/material-ui/>. [Accedido: 2024, 05, 19].
- [26] MongoDB, "The MEAN Stack: MongoDB, Express.js, AngularJS (or Angular), and Node.js,". Available: <https://www.mongodb.com/resources/languages/mean-stack>. [Accedido: 2024, 05, 19].
- [27] CHOUGALE, Pankaj, et al. Firebase-overview and usage. International Research Journal of Modernization in Engineering Technology and Science, 2021, vol. 3, no 12, p. 1178-1183.
- [28] Google Firebase, "Firebase Console,". [Online]. Available: <https://console.firebase.google.com/>. [Accedido: 2024, 05, 19]
- [29] Colaboradores de Wikipedia. Cliente-servidor [Online]. Wikipedia, La enciclopedia libre, 2024. Available: <https://es.wikipedia.org/w/index.php?title=Cliente-servidor&oldid=160239109>. [Accedido: 2024, 05, 20]
- [30] Colaboradores de Wikipedia. Single-page application [en línea]. Wikipedia, La enciclopedia libre, 2024. Available: https://es.wikipedia.org/w/index.php?title=Single-page_application&oldid=160013966. [Accedido: 2024, 05, 20]
- [31] Colaboradores de Wikipedia. Transferencia de Estado Representacional [Online]. Wikipedia, La enciclopedia libre, 2024. Available: https://es.wikipedia.org/w/index.php?title=Transferencia_de_Estado_Representacional&oldid=157265153. [Accedido: 2024, 05, 20]
- [32] Amazon Web Services, "Bases de Datos NoSQL,". [Online]. Available: <https://aws.amazon.com/es/nosql/>. [Accedido: 2024, 05, 20]
- [33] Colaboradores de Wikipedia. Base de datos relacional [Online]. Wikipedia, La enciclopedia libre, 2024. Available: https://es.wikipedia.org/w/index.php?title=Base_de_datos_relacional&oldid=157283674. [Accedido: 2024, 05, 20]
- [34] FreeCodeCamp, "Introducción a Mongoose para MongoDB,". [Online]. Available: <https://www.freecodecamp.org/espanol/news/introduccion-a-mongoose-para-mongodb/#:~:text=Mongoose%20es%20una%20biblioteca%20de,de%20esos%20objetos%20en%20MongoDB>. [Accedido: 2024, 05, 20]

[35] Colaboradores de Wikipedia. Modelo Entidad-Relación [Online]. Wikipedia, La enciclopedia libre, 2024. Available: https://es.wikipedia.org/w/index.php?title=Modelo_entidad-relaci%C3%B3n&oldid=159937915. [Accedido: 2024, 05, 20]

[36] Cloudflare, "¿Qué es un endpoint de API?,". [Online]. Available: <https://www.cloudflare.com/es-es/learning/security/api/what-is-api-endpoint/>. [Accedido: 2024, 05, 24]

[37] Colaboradores de Wikipedia. Anexo: Códigos de estado HTTP. Wikipedia, La enciclopedia libre, 2024. [Online]. Available: https://es.wikipedia.org/w/index.php?title=Anexo:C%C3%B3digos_de_estado_HTTP&oldid=160318057. [Accedido: 2024, 05, 24]

[38] GitHub, "GitHub: Where the world builds software," [Online]. Available: <https://github.com/>. [Accedido: 2024, 06, 02]

Este documento esta firmado por



Firmante	CN=tfgm.fi.upm.es, OU=CCFI, O=ETS Ingenieros Informaticos - UPM, C=ES
Fecha/Hora	Mon Jun 03 18:54:44 CEST 2024
Emisor del Certificado	EMAILADDRESS=camanager@etsiinf.upm.es, CN=CA ETS Ingenieros Informaticos, O=ETS Ingenieros Informaticos - UPM, C=ES
Numero de Serie	561
Metodo	urn:adobe.com:Adobe.PPKLite:adbe.pkcs7.sha1 (Adobe Signature)