



Universidad Politécnica
de Madrid



**Escuela Técnica Superior de
Ingenieros Informáticos**

Grado en Matemáticas e Informática

Trabajo Fin de Grado

Sistema de Entrega de Prácticas

Autor: Marco Prieto Gil

Tutor(a): Ángel Lucas González Martínez

Madrid, enero 2025

Este Trabajo Fin de Grado se ha depositado en la ETSI Informáticos de la Universidad Politécnica de Madrid para su defensa.

Trabajo Fin de Grado

Grado en Matemáticas e Informática

Título: Sistema de Entrega de Prácticas
enero 2025

Autor: Marco Prieto Gil

Tutor:

Ángel Lucas González Martínez

Lenguajes y Sistemas Informáticos e Ingeniería de Software

ETSI Informáticos

Universidad Politécnica de Madrid

Resumen

Este proyecto se basa en la implementación de un nuevo sistema de entrega de prácticas, ya que las últimas versiones, debido a su largo desarrollo durante diversos años, cuentan con factores redundantes y obsoletos. La solución propuesta se basa en un sistema multi asignatura, integrando funcionalidades esenciales como la autenticación con LDAP, gestión eficiente de asignaturas y prácticas, corrección automática de entregas y un diseño de base de datos optimizado.

El objetivo principal es la creación de un sistema que permite el acceso a 2 tipos de usuarios, alumnos y profesores, con prácticamente las mismas funcionalidades salvo algún permiso extra para los profesores. Por tanto, cada vez que se mencione “usuario” se estará teniendo en cuenta tanto alumno como profesor, y para las funcionalidades particulares se empleará solamente el tipo. Como es evidente, los usuarios podrán acceder a sus asignaturas y prácticas correspondientes, realizar entregas individuales o en grupo y recibir retroalimentación inmediata. Estas entregas se evalúan mediante un conjunto de scripts o programas (según los métodos de corrección deseados) que garantizan la validez y el cumplimiento de los criterios establecidos.

A nivel técnico, el desarrollo combina tecnologías robustas y accesibles, como PHP, MySQL y JavaScript, para asegurar su mantenimiento y escalabilidad. Además, se implementa en un entorno controlado basado en máquinas virtuales, garantizando la seguridad y fiabilidad del sistema antes de su despliegue. Una vez terminado el sistema, y teniendo en cuenta sus limitaciones, se pretende que esté en sistemas de producción de máquinas del CDC (Centro de Cálculo) de la escuela.

Abstract

This project focuses on implementing a new system for submitting assignments, as the latest versions, developed over several years, contain redundant and obsolete factors. The proposed solution is based on a multi-subject system, integrating essential functionalities such as LDAP authentication, efficient management of subjects and assignments, automated grading, and an optimized database design.

The main objective is to create a system that allows access to two types of users: students and teachers, with almost identical functionalities except for some additional permissions for teachers. Therefore, whenever "user" is mentioned, it refers to both students and teachers; specific functionalities will reference the relevant type. Users will be able to access their respective subjects and assignments, make individual or group submissions, and receive immediate feedback. These submissions are evaluated using a set of scripts or programs (depending on the desired grading methods) that ensure functionality and compliance with established criteria.

From a technical perspective, the development combines robust and accessible technologies, such as PHP, MySQL, and JavaScript, to ensure maintainability and scalability. Furthermore, it is implemented in a controlled environment based on virtual machines, ensuring the security and reliability of the system before deployment. Once the system is completed and considering its limitations, it is intended to run on production systems of the CDC (Centro de Cálculo) of the school.

Tabla de contenidos

1	Introducción	1
1.1	Objetivos Generales	1
1.2	Alcance	2
1.3	Estado de la Cuestión	3
2	Desarrollo	5
2.1	Análisis y Diseño	5
2.1.1	Secciones y flujo de usuarios	5
2.1.2	Base de Datos	7
2.1.2.1	Diseño Entidad-Relación	7
2.1.2.2	Ejemplos de Tablas	10
2.1.2.3	Funciones	14
2.1.3	Archivos de configuración de las Prácticas	15
2.1.4	Ficheros de corrección de las Prácticas	15
2.1.5	Formularios	16
2.1.5.1	Formulario de Inicio de Sesión	16
2.1.5.2	Formulario de Selección	16
2.1.5.3	Formulario de Entrega	17
2.1.5.4	Formulario de Creación de Grupo	18
2.1.6	Actores	20
2.2	Inicio de Sesión	21
2.3	Sección de Asignaturas	22
2.4	Sección de Prácticas	23
2.5	Entrega de Prácticas	24
2.5.1	Información de la Práctica	24
2.5.2	Opciones de la Práctica	25
2.5.2.1	Práctica obligatoriamente en Grupo	25
2.5.2.2	Práctica Opcionalmente Individual o en Grupo	25
2.5.2.3	Práctica obligatoriamente Individual	26

2.5.3	Formulario de Entrega de la Práctica	26
2.6	Corrección de la Entrega	27
2.7	Estructura del Sistema	29
2.8	Base del Sistema.....	32
2.9	Index.html y login.php	33
2.10	SeleccionAsignaturas.php	33
2.11	SeleccionPracticas.php.....	34
2.12	EntregaPractica.php.....	35
2.13	CreacionGrupos.php y grupos.php.....	36
2.14	SubirArchivos.php	38
2.15	Pruebas.....	39
2.15.1	Ingreso de Datos	39
2.15.2	Validación Bajo Diferentes Escenarios.....	41
3	Resultados y conclusiones	47
3.1	Resultados	47
3.2	Conclusiones.....	47
4	Análisis de Impacto	49
5	Bibliografía	50
6	Anexos.....	51
6.1	Instalación MV	51
6.2	Guía de Uso para Profesores	52
6.3	Archivo de configuración y Script de Corrección.....	53

1 Introducción

En el ámbito educativo, la implementación, entrega y gestión de prácticas representan momentos muy cotidianos. Hoy en día tenemos muchos medios para realizar dichas entregas, como el Moodle [1], que principalmente realiza correcciones manuales, aunque también cuenta con extensiones que permiten pruebas en el código de los alumnos, Deliverit [2], que ha ido recibiendo mejoras durante estos últimos años gracias a diferentes propuestas de TFG, u otros recursos del CDC.

1.1 Objetivos Generales

Por ello, es necesario el desarrollo de un sistema intuitivo, con fácil acceso y que cumpla con las funcionalidades básicas para la entrega de prácticas, siendo los objetivos más relevantes:

- **Diseño de la plataforma de inicio de sesión de usuarios:** Gestionando los casos que son válidos y los que no, y determinando el flujo que los usuarios seguirán una vez accedan al sistema. Uno de los aspectos más relevantes es que, junto con la BD, se hará uso del LDAP (protocolo ligero de acceso a directorios) [3] de la escuela o UPM para la primera fase de la autenticación de los usuarios. En la segunda fase, se hará uso de una BD para determinar si el usuario tiene acceso y a que tiene acceso.
- **Diseño y creación de la base de datos:** Prácticamente toda la información se almacena en la base de datos, por lo que hay que diseñar un modelo de entidad relación que almacene los datos que nos interesan evitando la redundancia y de manera organizada para facilitar el trabajo de futuros desarrolladores.

- **Funcionalidades del sistema:**

El sistema debe permitir a los profesores:

- Acceder a las asignaturas en las que están matriculados para la convocatoria actual
- Publicar prácticas para sus asignaturas
- Gestionar la información de las prácticas (el estado, las fechas, los errores establecidos para el corrector automático...)
- Ver y realizar entregas si la práctica está activa o en desarrollo y recibir información tras el procesado automático de todos los datos

- Corregir manualmente las entregas que ya han sido realizadas por otros usuarios y hayan superado el filtro de las pruebas obligatorias

El sistema debe permitir a los alumnos:

- Acceder a las asignaturas en las que están matriculados para la convocatoria actual
- Si la práctica es en grupo, avisar y dar la opción de crearlo con las credenciales correspondientes antes de la primera entrega, cumpliendo ciertas condiciones que se tratarán más adelante
- Ver y realizar entregas si la práctica esta activa y recibir información tras el procesado automático de todos los datos

1.2 Alcance

A pesar de que se vayan a permitir todas esas funcionalidades, este TFG plantea ciertas limitaciones en cuanto al uso del sistema.

El objetivo principal de este trabajo es que los usuarios (especialmente los alumnos) puedan acceder a las asignaturas y prácticas que les son asignadas, realizar entregas según las condiciones de la práctica (individual y/o grupal, el número máximo de entregas) y obtener información sobre su entrega.

En cuanto a los profesores, sus funcionalidades dentro de la aplicación serán muy parecidas a las de los alumnos, ya que a pesar de ser quienes tienen los privilegios para publicar, modificar y corregir manualmente las prácticas, todas estas acciones se harán directamente sobre la BD o sobre archivos de configuración para las prácticas con la información que no esté en la BD. Por lo que es importante destacar que no se plantea el desarrollo de una interfaz en la aplicación para poder realizar dichas tareas pues implicaría una carga de trabajo significativa, centrándose más en el uso del alumno, lo cual también requiere una gran dedicación de tiempo.

Hay otras limitaciones como la gestión de grupos de matriculación para poder aplicar diferentes fechas o correcciones a una misma práctica según el grupo, pero se comentarán más adelante en el apartado **Planes a Futuro** de

Conclusiones

1.3 Estado de la Cuestión

Este trabajo se plantea con el objetivo de desarrollar un nuevo sistema debido a la situación con el sistema actual de Programación II.

A pesar de la existencia de sistemas de entrega funcionales y prácticos, se busca la opción de que los usuarios no tengan que pasar por un registro directo y emplear únicamente el correo de la universidad (credenciales LDAP) para tener acceso al sistema. Por ejemplo, Deliverit emplea el número de matrícula y los usuarios pueden modificar su contraseña asociada. También cabe destacar las tecnologías empleadas en dicho sistema, como Elixir y Phoenix, PostgreSQL, Docker... Y debido a que el objetivo de este trabajo era realizar un nuevo sistema, se ha decidido usar tecnologías más conocidas o empleadas en versiones anteriores del sistema de entrega para Programación II, como PHP, MySQL y JavaScript, junto con otros servicios que ofrecen funcionalidades para la gestión y ejecución de prácticas en el entorno educativo

Como se comentó anteriormente, se espera que el sistema funcione en las máquinas del CDC, y como el acceso directo puede causar una situación crítica, se hará uso de MV (máquinas virtuales) que simulen el entorno y ofrezcan la mayoría de las herramientas necesarias para llevar a cabo el desarrollo y validación del sistema antes de pasarlo a producción en las máquinas del CDC. La separación del sistema de desarrollo y de producción son fundamentales para garantizar la funcionalidad, escalabilidad y seguridad de las aplicaciones, y el desarrollo de estos sistemas en entornos controlados permite simular condiciones reales, detectar errores y optimizar el rendimiento antes de finalizar la implementación. En este contexto, las MV desempeñan un papel crucial, ya que ofrecen la posibilidad de replicar ambientes aislados minimizando los riesgos asociados con el uso directo de los equipos del CDC

Hoy en día las MV son ampliamente utilizadas tanto educativa como profesionalmente gracias a la capacidad de simular entornos de manera segura como se acaba de comentar. Algunos de los más destacados son [4]:

- **VirtualBox** [5]: software gratuito y multiplataforma desarrollado por Oracle. Es conocido por su facilidad de uso y su compatibilidad con diversos sistemas operativos
- **VMware** [6]: software comercial que ofrece un rendimiento avanzado y múltiples herramientas para la gestión de entornos virtuales

- **Hyper-V** [7]: plataforma nativa de Windows que permite la creación y gestión de máquinas virtuales enfocándose en la integración con otros servicios de Microsoft

En este proyecto se ha optado por la opción de Hyper-V debido a varias razones:

- **Integración nativa:** al estar desarrollada por Microsoft, Hyper-V se integra perfectamente con sistemas Windows, lo que facilita la configuración y reduce problemas de compatibilidad. Y personalmente, al contar con un sistema de Windows, es más sencillo emplear una MV para simular un sistema UNIX en lugar de usar directamente una partición para Linux
- **Rendimiento:** ofrece un buen rendimiento en sistemas Windows, optimizando el uso de recursos en comparación con otras opciones
- **Uso:** las opciones que para crear una MV desde cero, importar y exportar MV, crear puntos de control antes de realizar acciones críticas, editar e inspeccionar discos...
- **Disponibilidad:** Hyper-V está incluido en ediciones profesionales de Windows, por lo que no es necesaria la instalación de softwares adicionales

2 Desarrollo

En este capítulo se tratará todo el desarrollo del trabajo, desde el análisis, estudiando que es necesario y por qué se necesita para poder tener un sistema funcional, hasta la implementación de todos los objetivos mínimos de desarrollo

2.1 Análisis y Diseño

Antes de empezar con el desarrollo se requiere saber las necesidades del sistema, como lo que se puede hacer, que información se quiere guardar, que se mostrará por pantalla en el sistema, como moverse a través de él...

2.1.1 Secciones y flujo de usuarios

Las secciones disponibles en el sistema vienen dadas en la siguiente figura

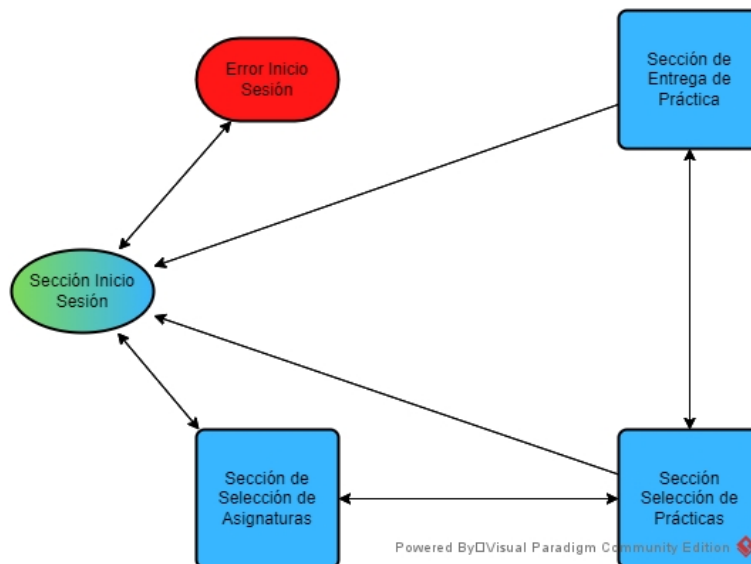


Figura 1 Secciones del Sistema

Se pueden apreciar 3 colores diferentes, el verde, que representa la conexión con el LDAP para el inicio de sesión, el rojo, en el caso de que ocurra algún error en el inicio de sesión, y el azul para las conexiones y otras operaciones en la BD, tanto en el inicio de sesión como en las secciones de selección y entrega. Cada una de ellas cuentan con acciones u errores que pueden producirse durante su uso, para más detalle se puede observar en la siguiente figura

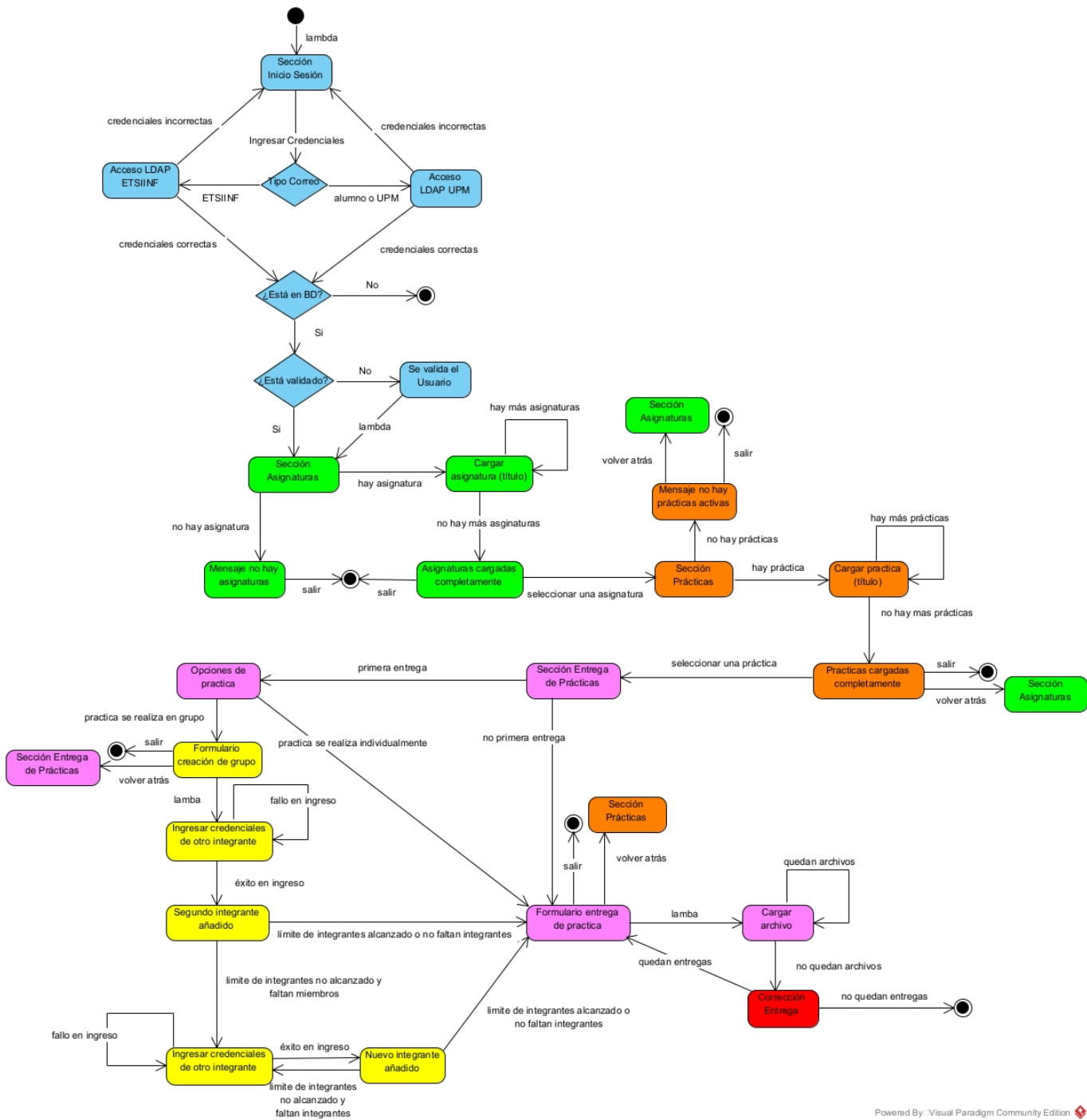


Figura 2 Diagrama de Flujo de Navegación

Aunque más adelante se tratarán estas acciones de manera más precisa, explicando su implementación y todos los elementos involucrados en la misma, se puede abordar esta figura de manera resumida. Cada color representa una sección del sistema, pudiendo volver atrás o cerrar sesión en cualquier momento

Cuando el usuario accede al sistema lo primero que debe hacer es iniciar sesión (no hay que registrarse por el uso del LDAP, aunque se tendrá en cuenta la validación del usuario para comprobar si ya accedió al sistema o las asignaturas previamente). En el caso de haber algún error, no permite al usuario acceder al resto del sistema, en caso contrario, se accede a la sección de asignaturas donde se cargan todas las asignaturas disponibles para el usuario.

Una vez mostradas todas las asignaturas, el usuario selecciona una de ellas, accediendo a la sección de prácticas, donde se repite un proceso similar al anterior.

Cuando una práctica es seleccionada y no se ha realizado ninguna entrega, puede haber 2 casos, que el usuario quiera (o deba) realizarla individual o en grupo, por tanto, según diferentes criterios se pasa directamente al formulario de entrega o al formulario de creación de grupo, donde se tiene la posibilidad de añadir desde un número mínimo hasta un número máximo de integrantes ingresando sus credenciales LDAP.

Y por último se pasa a la corrección automática de las entregas, donde habrá un límite de intentos, lo que bloqueará el envío de más entregas.

2.1.2 Base de Datos

Aquí se tratará el diseño y los elementos de la BD que se usarán en el sistema

2.1.2.1 Diseño Entidad-Relación

El sistema de entrega tendrá:

- **Usuarios:** de los cuales se debe conocer, el nombre, apellidos, dirección de correo, rol (alumno o profesor), último acceso al sistema y la validación. También contarán con su grupo de matriculación pero este no tendrá ningún efecto durante el proyecto
- **Asignaturas:** con el nombre y abreviatura de la correspondiente asignatura
- **Cursos:** formado por el año o curso académico (24/25,25/26...) y convocatoria (febrero, junio o julio)

Los usuarios están matriculados por lo menos en una **asignatura**, y en una asignatura, evidentemente, puede haber muchos usuarios. Las asignaturas pertenecen a uno o varios **cursos**.

Las “prácticas” se dividen en 2 partes:

- El **slot**, con *timestamps* de las fechas de inicio y fin (con hora, minuto y segundo) para permitir las entregas
- La **práctica**, que tiene un nombre, descripción, estado (activa, cerrada o desarrollo) de la práctica, número máximo de entregas, número máximo y mínimo de integrantes, peso en la asignatura.

Se debe conocer la asignatura y curso a la que pertenece.

- La práctica tendrá asociada un **baremo**, donde se guardarán los códigos de error, para el sistema de corrección automática y la corrección manual, junto con la descripción del error, el tipo de error (opcional u obligatorio), el tipo de corrección (tester o revisión) y la penalización de este

Si la práctica lo permite, los usuarios pueden formar un **grupo** para esa práctica, un usuario puede estar en ningún o varios grupos (siempre y cuando no sean de la misma práctica, además todos los miembros deben estar en la asignatura y misma convocatoria). Un grupo debe estar formado por 2 o más usuarios También se debe saber la práctica para la que se creó el grupo

- Cada **grupo** tendrá un representante, este será el único usuario del grupo que podrá realizar entregas, y uno o más miembros o integrantes

Tanto los usuarios como los grupos serán clasificados como **entregantes**, que se diferenciarán según el tipo: individual o grupo. Este punto se tratará con más profundidad en el **Formulario de Creación de Grupo** y la sección **Entrega de Prácticas**

- De los **entregantes** se necesita saber el tipo, el id del representante (el id del usuario si es individual o el id del representante del grupo) y la práctica a la que está asociado el entregante

Un entregante puede entregar una práctica 0 o M veces (número máximo), y una práctica se puede entregar por muchos entregantes (cada uno su entrega)

- En cada **entrega**, se guardará el *timestamp* de la fecha de entrega, la validez(válida o inválida) y el estado(apta o no apta) de la entrega. Si la entrega es válida, también aparecerán los códigos de error identificados por el corrector del baremo (los resultados del *tester*, tanto obligatorios como opcionales, junto comentarios del profesor, aunque estos comentarios solo en la última entrega, en la definitiva, tras la corrección manual de la entrega).

Una entrega será válida si se puede ejecutar correctamente la corrección automática de dicha entrega, e inválida si ocurre algún tipo de problema durante su ejecución. Y el estado de la entrega refleja si la práctica ha superado las pruebas mínimas u obligatorias

Se debe saber quién es el entregante y la asignatura de la práctica asociada a la entrega

Planteado el problema, se necesita un diagrama Entidad-Relación lo más simplificado posible para poder tener acceso a todo lo comentado.

Para ello, se puede partir de las 3 entidades básicas, **Usuarios**, **Asignaturas** y **Cursos**, cada una con sus respectivos atributos. Gracias a estas 3 entidades podemos crear sus respectivas tablas y formar las 2 relaciones, **Asignatura_Curso** y **Usuarios_Asignatura_Curso**, para conocer a que curso pertenece cada asignatura y donde este matriculado el usuario.

Asignatura_Curso también sirve para la definición de la tabla **Practicas**, lo que permite saber la asignatura y el curso de la práctica.

Mediante una relación 1:1 (se presupone que no hay distinción entre grupos de matriculación de la misma asignatura) y la tabla Practicas definidas, se puede relacionar con ella tanto el **Slot**, como el **Baremo**. Ambos tendrán identificadores para saber a la práctica a la que pertenecen.

La entidad **Grupos** precisa de una relación para saber que usuarios pertenecen a un mismo grupo y también tendrán un campo identificador para saber la práctica para la que se creó el grupo, por lo que se empleará la tabla **Grupo_Usuario**.

Y por último las tablas **Entregantes** y **Entregas**. El entregante tendrá asociado un identificador para conocer la práctica a la que está asociado, por lo que se podrá relacionar cada una de sus entregas con la práctica correspondiente.

Gracias a esto es posible representar la base de datos con un esquema de entidad-relación como en la Figura 3

Aunque se apreciará mas adelante, en la medida de lo posible se evitará trabajar con simbolos o caracteres especiales, es decir, cuando se traten de atributos o entradas de la BD, no se emplearán caracteres como 'ñ' o 'á', 'í'... Aun así, se se respetarán dichos simbolos para ciertos campos, como nombres o apellidos de usuarios, ya que estos no deben ser modificados por la codificación

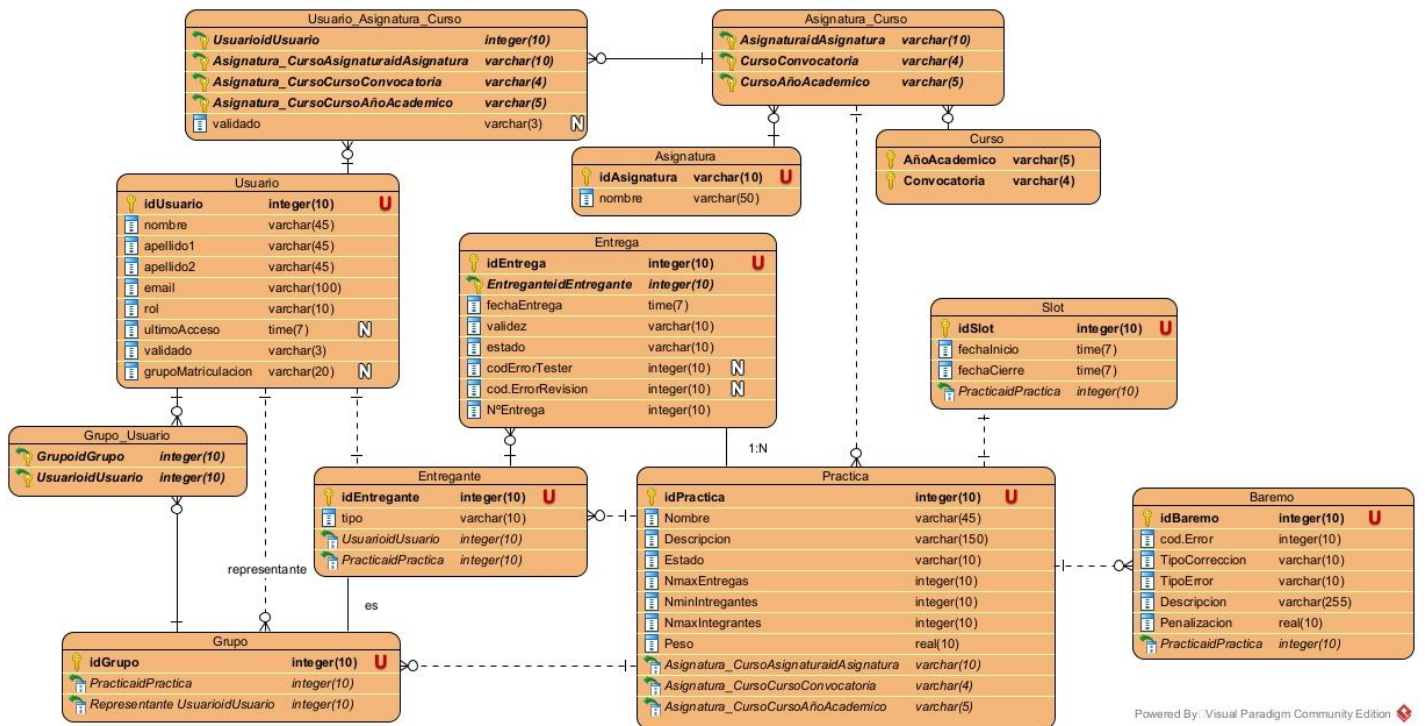


Figura 3 Esquema Entidad-Relación

2.1.2.2 Ejemplos de Tablas

A continuación, se muestran algunos ejemplos con datos ficticios de las tablas anteriormente mencionadas

idUsuario	nombre	apell1	apell2	correo	rol	ult. acc	valid
1	Pedro	Gutiérrez	...	pedro@alumnos.upm.es	alumno	02/10/2024 13:55:10	si
2	Marta	García	...	marta@upm.es	profesor	04/10/2024 13:55:10	si
3	Mario	Jiménez	...	mario@alumnos.upm.es	alumno		no
4	Patricia	Santos	...	patricia@alumnos.upm.es	alumno	02/10/2024 13:55:10	si
5	Mario	García	...	mario@upm.es	profesor	04/10/2024 13:55:10	si

Tabla 1 Usuarios

- **IdUsuario**: identificador de usuario único e autoincremental
- **Nombre, apell1, apell2**: nombre y apellidos del usuario
- **Correo**: correo electrónico de la escuela o UPM asociado al usuario
- **Rol**: rol del usuario, puede ser alumno o profesor
- **Ult. Acc**: *timestamp* del último acceso al sistema, puede ser null si no se ha accedido
- **Validado**: indica si se ha accedido al sistema al menos una vez

idAsignat.	nombre
PROGII	Programación II
AED	Algoritmos y Estructuras de Datos
CONCU	Concurrencia

Tabla 2 Asignaturas

- **IdAsignat:** identifica la asignatura mediante su abreviatura
- **Nombre:** Nombre de la asignatura

anioAcad	convocatoria
24/25	Feb
24/25	Jun
24/25	Jul

Tabla 3 Cursos

- **AnioAcad:** año académico 24/25, 25/26, 26/27...
- **Convocatoria:** convocatoria del curso, puede ser feb, jun o jul

idGrupo	idPractica	Representante
1	1	1
2	2	2
3	3	3

Tabla 4 Grupos

- **IdGrupo:** identificador del grupo, único y autoincremental
- **IdPractica:** identificador de la práctica a la que está asociado el grupo
- **Representante:** identificador del usuario representante del grupo

idGrupo	idUser
1	1
1	4
2	2
2	5
3	1
3	4

Tabla 5 Grupo-Usuarios

- **IdGrupo:** identificador del grupo
- **IdUser:** identificador de usuarios que pertenecen a dicho grupo

idEntregante	tipo	idRepresentante	idPractica
1	usuario	2	1
2	grupo	1	1
3	grupo	2	2
4	grupo	3	3

Tabla 6 Entregantes

- **IdEntregante:** identificador del entregante, único y autoincremental
- **Tipo:** el tipo de entregante, puede ser usuario o grupo
- **IdRepresentante:** identificador que representa al entregante, si el tipo es individual, será el id del mismo usuario, si el tipo es grupal, será el id del representante del equipo
- **IdPractica:** identificador de la práctica a la que está asociado el entregante

idEntrega	idEntregante	fechaEntr.	validez.	estado.	cod. Error Tester	cod. Error Revisión
1	1	01/10/2024 13:55:10	valida	noApta	1,3,7,11,15,20	
2	1	02/10/2024 10:19:01	valida	apta	1,2,3	21,22,23
3	2	03/10/2024 17.12:22	valida	apta	3,4	10
4	3	03/10/2024 17:14:551	valida	noApta	23	
5	4	10/10/2024 20:32:11	valida	apta	1,2,4	21

Tabla 7 Entregas

- **IdEntrega:** identificador de la entrega que realiza un entregante, único y autoincremental
- **IdEntregante:** identificador del entregante al que está asociado la entrega
- **FechaEntr:** *timestamp* de la fecha en la que se realizó la entrega
- **Validez:** determina si se pudo corregir la práctica (válida) o no
- **Estado:** determina si ha superado las pruebas mínimas obligatorias (apta) o no
- **Cod.Error.Tester:** códigos de error generados en las pruebas automáticas
- **Cod. Error Revision:** códigos de error tras la corrección de la entrega por un profesor

idPractica	nombre	descrip	estado	maxEntregas	min integr.	max integr.	peso	id Asignatura	anioAcad	convocatoria
1	SmartParking		activa	10	1	3	0.2	PROGII	24/25	feb
2	Ajedrez		activa	3	1	3	0.1	PROGII	24/25	feb
3	Hashmaps		activa	5	1	3	0.15	AED	24/25	feb
5	C. Carrera 1		cerrada	1	1	2	0.1	CONCU	24/25	feb
4	C. Carrera 2		desarrollo	1	1	2	0.05	CONCU	24/25	jul

Tabla 8 Prácticas

- **IdPractica:** identificador de la práctica, único y autoincremental
- **Nombre:** nombre de la práctica
- **Descripción:** descripción de la práctica
- **Estado:** estado de la práctica, puede ser activa (todos los usuarios pueden entregar), en desarrollo (solo profesores pueden verlas) y cerrada (nadie puede verlas)
- **maxEntregas:** número máximo de entregas permitidas para la práctica. Debe ser mayor que 0
- **minIntegr:** número mínimo de usuarios para hacer la práctica juntos. Debe ser mayor que 0
- **maxIntegr:** número máximo de usuarios para hacer la práctica juntos. Debe ser mayor o igual que N° min integr.
- **Peso:** peso en la asignatura correspondiente
- **IdAsignatura, anioAcad, Convocatoria:** identificador, curso y convocatoria de la asignatura a la que está asociada la práctica

idSlot	idPractica	fecha inicio	fecha fin
1	1	20/09/2024	20/10/2024
2	2	25/09/2024	25/10/2024
3	3	01/10/2024	15/10/2024

Tabla 9 Slots

- **IdSlot:** identificador del *slot*, único y autoincremental
- **IdPractica:** identificador de la práctica a la que está asociada el *slot*
- **Fecha inicio/fin:** fechas de apertura/cierre para la entrega de la practica

idPractica	codigo.err	tipo Corrección	tipo Error	descrip	penalizac.
1	1	tester	obligatorio	...	0.05
1	2	tester	opcional	...	0.05
1	3	revisión	opcional	...	0.1
2	1	tester	obligatorio	...	0.05
2	2	revisión	opcional	...	0.05
3	1	tester	obligatorio	...	0.1

Tabla 10 Baremos

- **idPractica:** identificador de la práctica a la que está asociada el baremo
- **Código. Err:** código de error (de un caso dentro de la práctica)
- **Tipo Corrección:** puede ser *tester* (corrector automático) o manual (por profesor)
- **Tipo Error:** puede ser opcional u obligatorio
- **Descripción:** descripción del error
- **Penalización:** penalización del error dentro de la práctica

2.1.2.3 Funciones

En la BD también será de utilidad tener diferentes funciones para asegurar que a los usuarios se les muestre solo la información relevante o disponible en cada momento.

Entre estos elementos destacan funciones para recuperar el año académico y la convocatoria actual, pues serán necesarias para obtener solo la información actual de los usuarios ya que no necesitamos datos de cursos anteriores, y como los profesores interactuarán con la BD, es preferible localizarlas en ella para el momento que se necesite su modificación

2.1.3 Archivos de configuración de las Prácticas

Como se mencionó en la **Introducción**, hay cierta información relacionada a las prácticas que no se encuentra en la BD, y está ubicada en diferentes archivos asociados a cada práctica.

La información de utilidad de estos archivos es el nombre de los ficheros que se deben entregar, el tamaño máximo permitido de esos ficheros y el script de corrección de la práctica. El resto de información será ignorada.

En estos archivos cada línea representa un único dato, se pueden comentar (ignorar) líneas específicas poniendo “//” al principio y están formateadas de la siguiente manera: Dato = valor. Deben nombrarse igual que las prácticas (sustituyendo espacios con guiones bajos) seguidos de “.conf”.

2.1.4 Ficheros de corrección de las Prácticas

Estos archivos son de suma importancia ya que serán los encargados de clasificar la validez y el estado de las entregas entre otras tareas

Serán unos ficheros *Bash* que deben recibir cada archivo (es decir, su ruta temporal absoluta) que se requiere en la práctica y el identificador de la práctica. Aunque se podría aceptar cualquier comando que se ejecute en entornos UNIX, pero gracias a la flexibilidad y libertad de estos ficheros se ha elegido esta opción

Una vez recibida toda la información correctamente, el fichero debe compilar u obtener el ejecutable (lo que cada práctica demande) de la entrega. Si este proceso no funciona correctamente, supera el tiempo límite de espera u ocurre cualquier otro problema, se debe clasificar la entrega como no válida, en caso contrario se clasifica como válida y se sigue con el proceso de corrección

Después se debe proceder con la ejecución de los *testers* o correctores, que, según las entregas de los usuarios, generarán unos códigos de errores u otros. Gracias al identificador de la práctica estos ficheros pueden obtener los códigos de errores de la corrección automática (*tester*) guardados en la tabla de baremos de la BD y trabajar con los que coincidan (por lo que es necesario que estén bien definidos en la BD). Según los errores que genere la entrega, se debe clasificar como apta, si ha superado las pruebas obligatorias o no apta en caso contrario

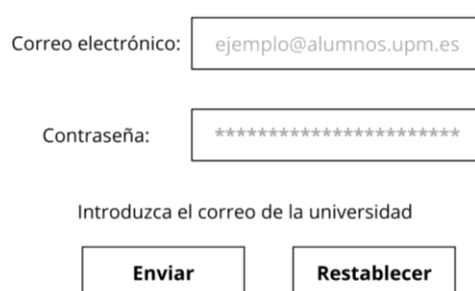
Antes de finalizar la ejecución de estos ficheros, debe devolver (escribiendo por consola) las clasificaciones (validez y estado de la entrega) y si los hubiese los códigos de error generados

2.1.5 Formularios

Los formularios serán esenciales durante el flujo del usuario, ya que permitirán realizar muchas de las tareas indispensables como el acceso al sistema, seleccionar asignaturas o prácticas, crear grupos para las prácticas y realizar entregas

2.1.5.1 Formulario de Inicio de Sesión

Este es el más sencillo de todos los formularios, solo contará con 2 campos, el correo electrónico y la contraseña del usuario, es decir, las credenciales LDAP, su diseño será:



Correo electrónico:

Contraseña:

Introduzca el correo de la universidad

Figura 4 Formulario Inicio Sesión

Con un botón para limpiar el formulario y otro para enviar los datos. Este último no se activará sin rellenar los datos del formulario ya que son indispensables para el acceso.

Debido a la privacidad y sensibilidad de estos datos, no podrán ser vistos en ningún instante, por lo que se enviarán mediante un método POST

2.1.5.2 Formulario de Selección

Los formularios de selección como se acaba de comentar serán de utilidad para mostrar las asignaturas y prácticas, y que el usuario elija una entre todas las disponibles. Estarán diseñados según las siguientes figuras

CABECERA



Seleccione el campo que desee

CAMPO 1	v
---------	---

Figura 5 Formulario de Selección 1

CABECERA

Seleccione el campo que desee:

CAMPO 1	v
CAMPO 2	
CAMPO 3	
CAMPO 4	

Enviar

Figura 6 Formulario de Selección 2

Estos formularios estarán formados por una cabecera, que tendrá alguna información sobre el formulario o sobre cómo debe usarlo un usuario, los campos, que serán los nombres completos de las asignaturas o prácticas y las abreviaturas de los campos

Cuando haya que manipular e interpretar estos campos, al tratarse de cadenas de caracteres considerablemente largas, se emplearán las abreviaturas, que serán cadenas simplificadas para permitir un mejor manejo de los datos. Se definirán los formatos cada abreviatura más adelante

Esta información se envía mediante un GET, por lo que será visible en la URL

2.1.5.3 Formulario de Entrega

Estos formularios permitirán la subida de archivos para la corrección de una práctica. Tendrán un diseño similar al de la siguiente figura

Seleccione todos los archivos:

Archivo1

Arrastre y suelte aquí

Seleccione Archivo

Archivo2

Arrastre y suelte aquí

Seleccione Archivo

Enviar

Seleccione todos los archivos:

Archivo1

Arrastre y suelte aquí

Archivo1 ✓

Seleccione Archivo

Archivo2

Arrastre y suelte aquí

Archivo1 ✗

Seleccione Archivo

Enviar

Seleccione todos los archivos:

Archivo1

Arrastre y suelte aquí

Archivo1 ✓

Seleccione Archivo

Archivo2

Arrastre y suelte aquí

Archivo2 ✓

Seleccione Archivo

Enviar

Figura 7 Formulario de Entrega

Se generarán tantos campos de carga como archivos requiera la práctica y se comprobará que el nombre del fichero seleccionado sea igual que el del fichero requerido, mientras no se cumpla esta condición y falten archivos por seleccionar el botón de enviar no estará activo

2.1.5.4 Formulario de Creación de Grupo

Gracias a este formulario se podrán formar grupos para una práctica si esta lo permite. Sus campos serán similares al formulario de inicio de sesión, ya que serán necesarias las credenciales de los posibles integrantes del grupo (y comprobaciones que se comentarán más adelante), pero una vez introducidas, se mostrará información que identifique a los usuarios seleccionados correctamente, dando la opción de eliminarlos antes de crear el grupo definitivamente. No se podrá borrar al usuario (el representante) que va a crear el grupo, por lo que siempre se mostrará. Si en el grupo no se ha igualado el número mínimo de integrantes (al menos 2 para todas las prácticas) no se activará el botón crear grupo, y debe permitir introducir tantos integrantes como el número máximo de integrantes la práctica a la que estará asociada el grupo. Tendrá un diseño similar al de las siguientes figuras:

Numero mínimo de Integrantes: 2
Numero máximo de Integrantes: 3

Correo electrónico:

Contraseña:

Integrantes:

UsuarioRepresentante

Figura 8: Formulario de Creación de Grupo Inicial

Numero mínimo de Integrantes: 2
Numero máximo de Integrantes: 3

Correo electrónico:

Contraseña:

Integrantes:

UsuarioRepresentante	
usuario1	Eliminar

Figura 9: Formulario de Creación de Grupo con 2 Integrantes

Numero mínimo de Integrantes: 2
 Numero máximo de Integrantes: 3

Correo electrónico:

Contraseña:

Intregrantes:

UsuarioRepresentante	
usuario1	Eliminar
usuario2	Eliminar

Figura 10: Formulario de Creación de Grupo Completo

Como se muestra en los ejemplos, una vez se selecciona un nuevo integrante aparece la opción de eliminarlo, dando a entender que el representante no puede eliminarse

Los campos para introducir credenciales permanecerán activos mientras no se supere el número máximo de integrantes de su práctica asociada, por lo que, si el grupo está completo y se elimina un miembro, volverá a activarse y se podrán volver a introducir otras credenciales. Y el botón crear grupo se activará una vez se llegue al número mínimo de integrantes

Después de pulsar el botón de crear grupo aparecerá un cuadro de diálogo o de aviso para confirmar la creación mediante un *checkbox*, mencionando que el grupo será definitivo y no se podrá cambiar ni eliminar

Numero mínimo de Integrantes: 2
 Numero máximo de Integrantes: 3

Correo electrónico:

Contraseña:

¿Realmente quiere confirmar el grupo?
 Una vez se cree el grupo, este será definitivo y no se podrá modificar

Deseo crear el grupo

UsuarioRepresentante	
usuario1	Eliminar
usuario2	Eliminar

Figura 11: Cuadro Diálogo Confirmación Grupo 1

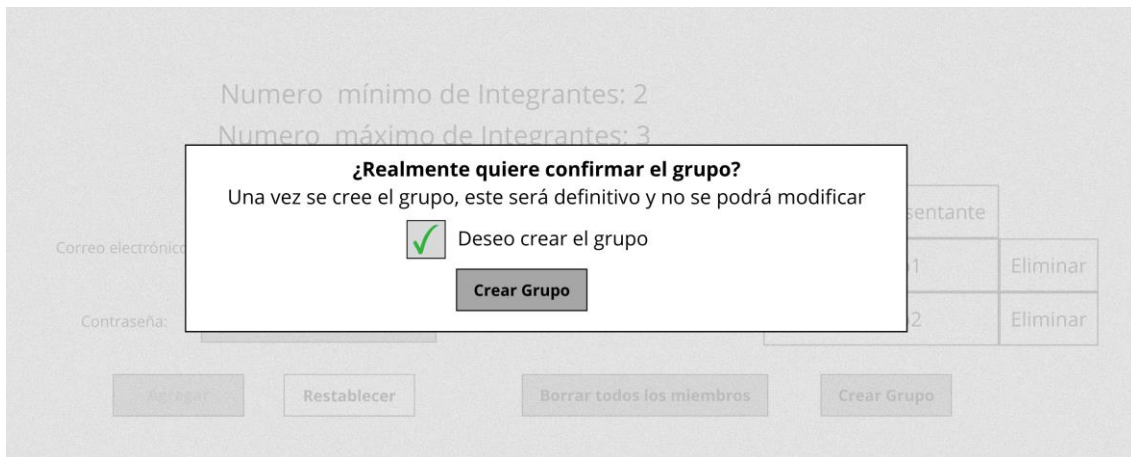


Figura 12: Cuadro Diálogo Confirmación Grupo 2

2.1.6 Actores

En el diseño de un sistema, es fundamental identificar y describir las componentes clave que lo conforman, conocidas como actores. Estos actores representan el conjunto de partes funcionales y técnicas que permiten alcanzar los objetivos de desarrollo. En este caso, los actores desempeñarán funciones comunes o específicas que abarcan desde la interacción con usuarios hasta la gestión de datos, garantizando un funcionamiento correcto y coherente.

En este apartado se muestran los actores principales, describiendo su labor, responsabilidad y contribución al diseño general del sistema

- **Base de Datos(MySQL):** como ya se ha mencionado, es necesario una base de datos relacional para poder almacenar toda información de las tablas definidas
- **Frontend(HTML, CSS y JavaScript):** para poder proporcionar una interfaz gráfica con la que los usuarios puedan interactuar. Con estas herramientas se puede estructurar las páginas del sistema, darles estilo, mostrar formularios y validar información básica de dichos formularios, mostrar mensajes o avisos en tiempo real...
- **Backend(PHP):** con PHP se podrán crear plantillas de formularios dinámicos, procesar la información que envían... También será de gran utilidad a la hora de recuperar o modificar información de la base de datos, ya que PHP cuenta con la extensión MySQLi integrada desde versiones anteriores, diseñada para trabajar con MySQL de manera más segura y eficiente que versiones obsoletas

- **Gestor de Base de Datos:** como se acaba de mencionar, PHP permite realizar consultas u operaciones a la BD de manera sencilla, por lo que, para una mejor gestión de dichas operaciones, se empleará un gestor para comunicarse con la BD, logrando así tener todas las interacciones dentro de una sola clase para facilitar un posible cambio de implementación o de sistema de gestión de bases de datos. Todas estas acciones serán llevadas a cabo por un usuario específico, con los permisos SELECT, INSERT, UPDATE y EXECUTE, sin contemplarse DELETE ya que no se plantea borrar entradas de la BD

2.2 Inicio de Sesión

La página de acceso al sistema es la encargada de mostrar el formulario de inicio de sesión que se ha tratado anteriormente

Como se mencionó el apartado Secciones y flujo de usuarios, el proceso de inicio de sesión es el siguiente:

- introducción de credenciales
- verificación con LDAP
- verificación con la BD

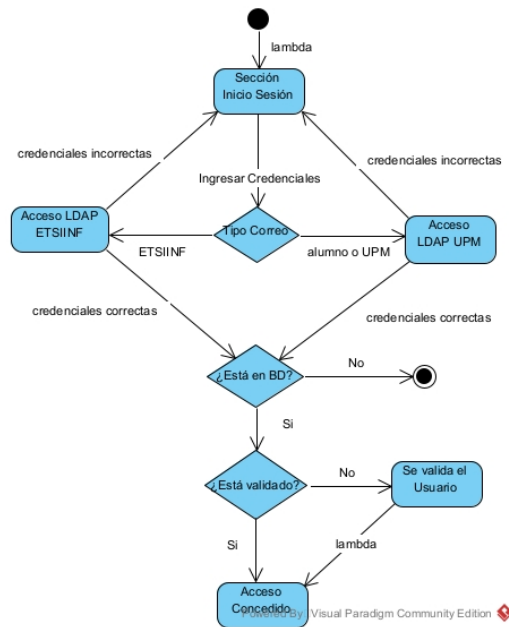


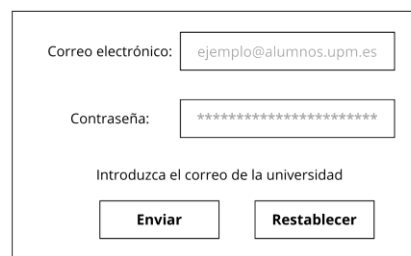
Figura 13: Proceso de Inicio de Sesión

Primero, gracias a un desarrollo previo [8], podemos obtener con facilidad si el usuario es de la UPM, ETSIIINF, profesor o no identificado (ninguno de los anteriores) y obtener una respuesta del LDAP. Se contempla que esta tarea

pueda dar lugar a errores durante la conexión, introduciendo credenciales incorrectas o un correo no identificado

Después, será necesaria una consulta para buscar al usuario en la BD, haciendo uso del correo electrónico proporcionado, y recuperar sus datos (como el rol, identificador de usuario...) en caso de que haya éxito en la consulta. Esta información estará disponible durante toda la sesión (tiempo entre el correcto acceso y la salida de un usuario en el sistema)

Acceso al Sistema de Entrega



Formulario de inicio de sesión con los siguientes campos:

- Correo electrónico: ejemplo@alumnos.upm.es
- Contraseña: *****
- Introduzca el correo de la universidad
- Botones: Enviar, Restablecer

Sistema de Entrega de Practicas @2024

Figura 14 Sección de Inicio de Sesión

2.3 Sección de Asignaturas

En esta sección, se cargarán las asignaturas en las que un usuario está matriculado en el curso y convocatoria actual (se usarán funciones de la BD comentadas anteriormente) y se hará uso del formulario de selección para mostrar dichas asignaturas.

Sistema de Entrega de Practicas @2024

Cerrar Sesión

Volver

Sección de Asignaturas

Bienvenido NombreUsrio

Ha accedido a la sección de asignaturas

NO SE ENCONTRARON ASIGNATURAS ✘
Póngase en contacto con el responsable correspondiente

Figura 15 Error en Sección de Asignaturas

Si a la hora de recuperar las asignaturas no se obtiene ninguna, debido a la suposición inicial de que todo usuario debe estar en al menos una asignatura, se mostrará un mensaje de error similar al de la Figura 15

Si todo sale correctamente, el usuario podrá ver todas las asignaturas que le correspondan, escoger una de ellas y enviar la selección a la Sección de Prácticas



Sección de Asignaturas

Bienvenido NombreUsuario

Ha accedido a la sección de asignaturas

Seleccione la asignatura que desee

Concurrencia

v

Enviar

Figura 16 Sección de Asignaturas

2.4 Sección de Prácticas

Esta sección realiza la búsqueda de prácticas de forma análoga a las asignaturas con algunas diferencias.

Se deben obtener las practicas activas (y en desarrollo para los profesores) de la asignatura y convocatoria actual, pero en el caso de que no se encuentre ninguna práctica, el mensaje que se mostrará, simplemente indicará que no hay prácticas publicadas por el momento.

Si todo sale correctamente, el usuario podrá ver las prácticas de las asignaturas que le correspondan en ese instante determinado, seleccionar una de ellas y enviar la selección a la siguiente sección, la Sección de Entrega de Prácticas



Sección de Prácticas

Ha accedido a la sección de prácticas

Seleccione la práctica que desee

Practica 1

v

Enviar

Figura 17 Sección de Prácticas

2.5 Entrega de Prácticas

Este apartado, junto con el de corrección de prácticas, constituye uno de los más importantes, donde se tienen que llevar a cabo la mayoría de las acciones fundamentales del sistema como la comprobación y creación de grupos, de entregantes, gestión de entregas...

Se puede entender de manera sencilla el flujo de los siguientes procesos mediante la figura:

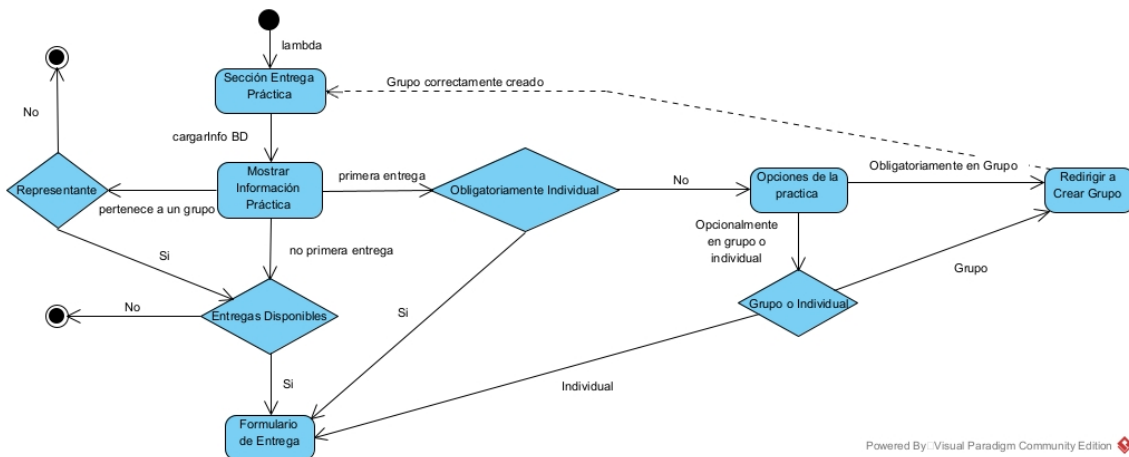


Figura 18: Flujo Proceso de Entrega

2.5.1 Información de la Práctica

Lo primero que debe ver un usuario será la información de la práctica, como el nombre, descripción, número máximo de entregas, número de entregas restantes, integrantes mínimos y máximos permitidos, fechas... Y a pesar de tener el tipo de práctica para determinar quién puede verla también se usarán las fechas de inicio y fin de la práctica de modo que, si por algún casual, la práctica permanece como activa, y el usuario no se encuentra entre estas fechas, no se podrá realizar ninguna acción, indicando que está fuera del rango de fechas permitidas

Sistema de Entrega de Practicas ©2024
Cerrar Sesión
Volver

Información de la Práctica
Nombre: SmartParking
...
Fecha inicio: 2024-11-01 08:00:00
Fecha inicio: 2024-12-01 08:00:00

También:

- Descripción
- NºmaxEntregas
- NºintentosDisponibles (si ya se ha entregado)
- NºminEntregantes
- NºmaxEntregantes

Figura 19: Información de la Práctica

2.5.2 Opciones de la Práctica

Después, según los números máximos y mínimos de integrantes, y antes de la primera entrega (y de pertenecer a un grupo), se mostrará un cuadro de opciones de la práctica, donde se decidirá si la práctica se hace individualmente, opcionalmente en grupo u obligatoriamente en grupo.

2.5.2.1 Práctica obligatoriamente en Grupo

Se mostrará un botón para redirigir al formulario de creación de grupo que permitirán incluir al resto de miembros del grupo, hasta que se alcance el número máximo de integrantes, o hasta que se supere el número mínimo y no quieran añadir más integrantes. Para añadir a un integrante al grupo ese integrante, al igual que el representante, debe estar validado en la BD, no deben de haber realizado ninguna entrega de esa práctica, ni pertenecer a ningún otro grupo de esa misma práctica y deben pertenecer a la asignatura para la convocatoria actual. Si un posible integrante no cumple alguna de estas características no se le permitirá ingresar al grupo, indicando el motivo, y se volverá a ofrecer añadir a un nuevo integrante

El usuario que rellene este formulario será considerado el representante del grupo y solo él podrá realizar entregas.

Una vez creado el grupo no se podrá modificar de ninguna manera, por lo que los integrantes y el representante de dicho grupo serán definitivos para toda la práctica

2.5.2.2 Práctica Opcionalmente Individual o en Grupo

Si el usuario puede elegir entre realizar la práctica individualmente o en grupo, el cuadro de opciones de la práctica mostrará el botón de la práctica obligatoriamente en grupo, y un *checkbox* inicialmente desmarcado para indicar



Opciones de la Práctica	
El usuario no pertenece a un grupo ¿Desea crear un grupo?	
<input type="button" value="Crear Grupo"/>	
También puede realizar la práctica de manera individual	
<input type="checkbox"/>	Deseo realizar la practica individualmente

Figura 21: Práctica con Opción de Crear Grupo o Individual

que el usuario quiere realizar la entrega individualmente. Si se marca ese *checkbox*, desaparecerá el botón de crear grupo y se mostrará el formulario de entrega

2.5.2.3 Práctica obligatoriamente Individual

Y en el último caso, si el usuario solo puede realizar entregas de manera individual, no se mostrarán las opciones de la práctica (ya que no hay opciones) y se mostrará directamente el formulario de entrega

2.5.3 Formulario de Entrega de la Práctica

Una vez superado el posible proceso de creación de grupo, se necesita saber que archivos son necesarios para la entrega de la práctica y que archivo permite la ejecución de la corrección automática. Esta información se encuentra en el fichero de configuración de la práctica, donde, como se ha mencionado anteriormente, deben estar incluidos tanto los nombres de los archivos como sus tamaños máximos permitidos

Si no se ha alcanzado el límite de entregas y todo sale correctamente (se cumplen las demás condiciones como ser el representante de un grupo, haberlo definido si es obligatoriamente en grupo...) se mostrará el formulario de entrega, para que se puedan procesar los archivos correspondientes.

Y si es el último intento o entrega disponible el formulario mostrará un cuadro de diálogo informando de las consecuencias correspondientes, como que no se podrá entregar otra vez y que esa entrega es la definitiva y será corregida manualmente independientemente de la diferencia de “notas” o errores con anteriores entregas

2.6 Corrección de la Entrega

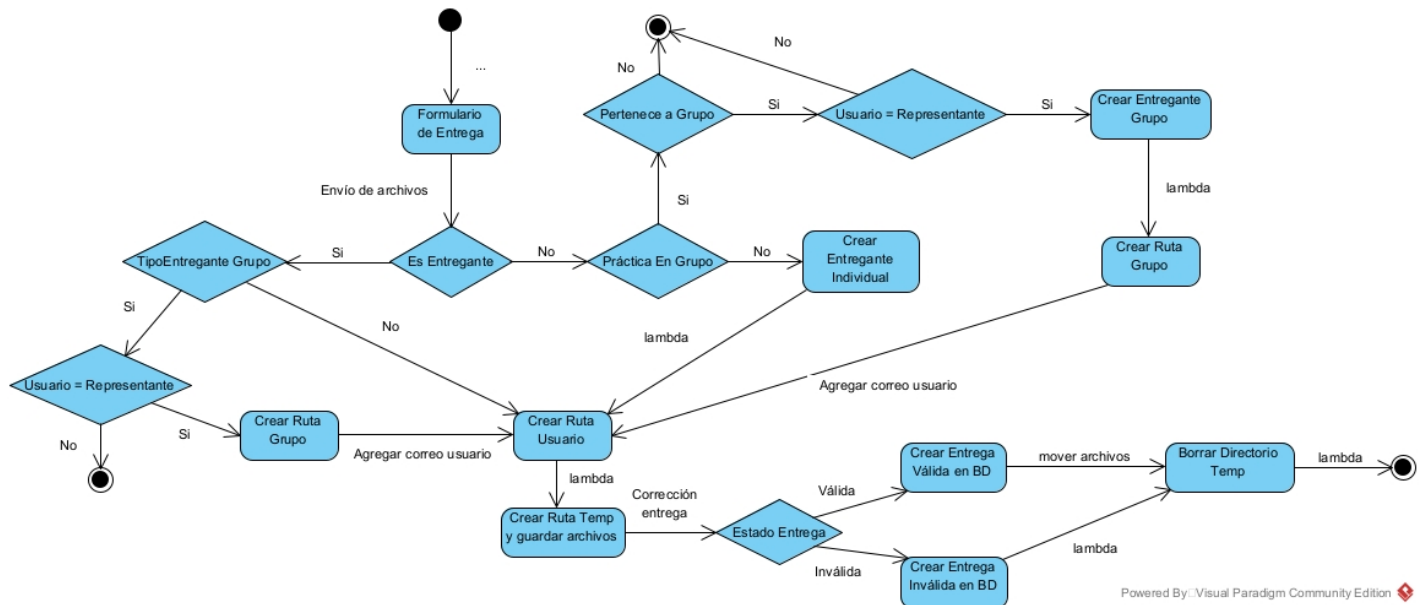


Figura 22: Flujo de Corrección de Entrega

Antes de realizar la corrección de la entrega, se deben hacer las comprobaciones correspondientes de entregante y/o grupo, y aprovechando estos datos se puede definir la ruta donde deben ser guardados los archivos que sube un usuario

Primero, si el usuario no es entregante se debe crear el entregante, siempre y cuando las circunstancias lo permitan. Es decir, si la práctica es:

- Individual, se crea el entregante de tipo individual
- En grupo, se comprueba que pertenezca a un grupo y que sea el representante, entonces se crea el entregante de tipo grupo

Teniendo los entregantes definidos se pueden definir la ruta (...)/Asignatura/Practica/[G_idGrupo]/correoEntregante

Cuando el usuario envía todos los archivos que requiere el formulario de entrega, estos se guardan en un directorio temporal ubicado en la ruta que se acaba de mencionar.

Después se pasan esos archivos al script de corrección para que devuelva la validez (válida o inválida) y estado (apta, no apta) de la entrega junto con los errores encontrados en la corrección automática

Según el estado de la entrega se crean diferentes entradas en la BD, por ejemplo, si la entrega no es válida, se guardará una entrada donde no se guardará más información que la validez de la entrega y no afectará al número de intentos

disponibles. Si es válida, se crea una entrada con su correspondiente estado (apta o no apta) junto los errores generados y se trasladan los archivos fuera del directorio temporal, que será eliminado independientemente de la validez de la entrega

En cualquier caso, se debe comunicar a los usuarios toda la información recopilada de la práctica.

En el caso de ser inválida, se mostrará un mensaje mencionando que no se pudo corregir su entrega y que no se tendrá en cuenta para el número de intentos restantes.

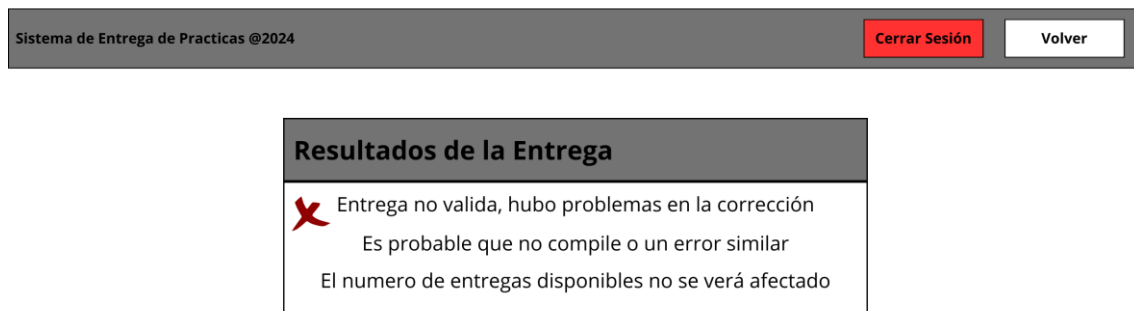


Figura 23: Entrega Inválida

Si es válida, se deben comunicar si la entrega es o no apta y mostrar todos los errores generados en la corrección junto con la descripción de estos, que se encuentra en la tabla Baremos de la BD

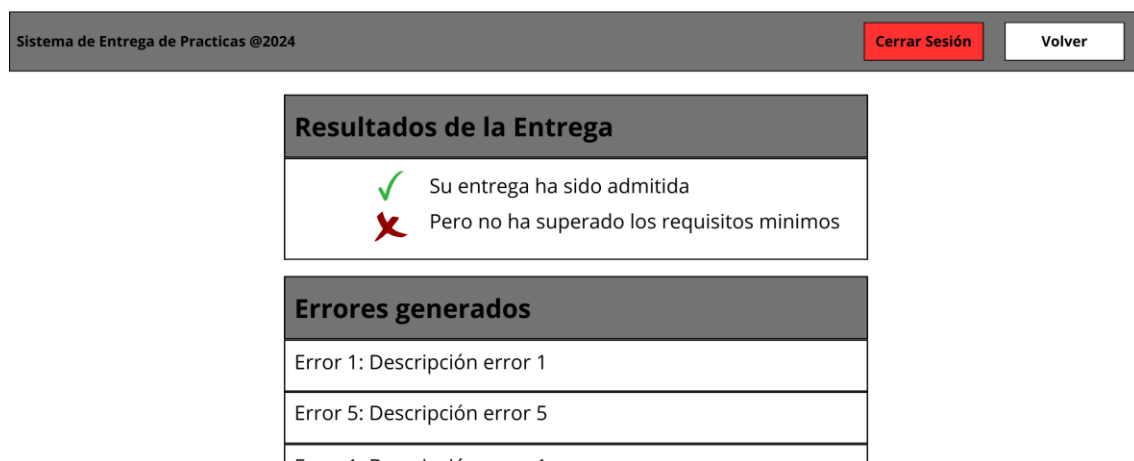


Figura 24: Entrega no Apta

Resultados de la Entrega

- ✓ Su entrega ha sido admitida
- ✓ Y supera todos los requisitos minimos

Errores generados

Error 12: Descripción error 12

Error 20: Descripción error 20

Error 1: Descripción error 1

Figura 25: Entrega Apta

2.7 Estructura del Sistema

El sistema desarrollado se organiza en una estructura de directorios diseñada para facilitar la comprensión, el mantenimiento y la escalabilidad del código.

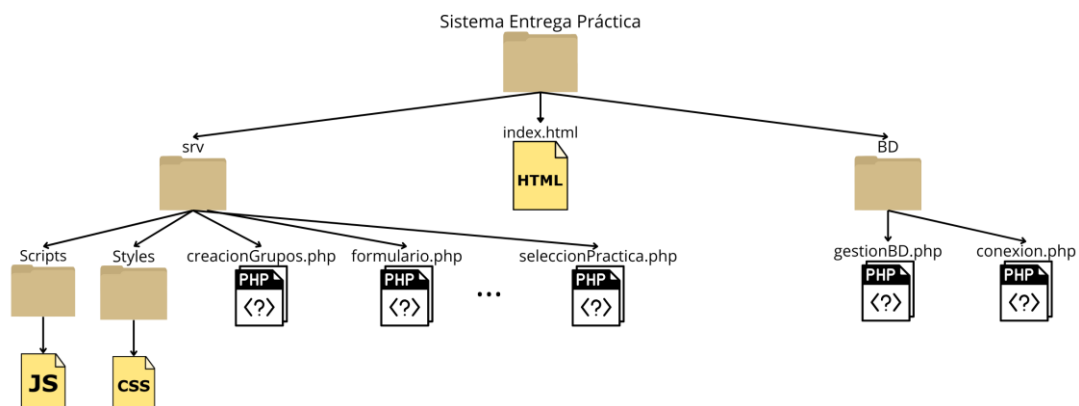


Figura 26: Árbol de Directorios del Sistema

El directorio principal está compuesto por:

- **Index.html:** el archivo principal que sirve como punto de entrada al sistema, es la página donde los usuarios deben iniciar sesión
- **/srv:** donde se encuentran los elementos relacionados con la lógica del sistema y la interfaz del usuario:
 - **/Scripts:** con los archivos JavaScript para gestionar diversas interacciones del usuario con el sistema, como marcar *checkboxes*, comprobaciones de formularios...
 - **/Styles:** hojas de estilo para definir el diseño y apariencia del sistema

- **Resto de archivos PHP:** archivos que implementan la lógica, los componentes, y el flujo del sistema
 - **creacionGrupos.php:** sección donde se crearán los grupos
 - **entregaPractica.php:** sección para realizar las entregas
 - **fichero.php:** clase Fichero para leer los ficheros de configuración de las prácticas
 - **formulario.php:** archivo donde se implementan los formularios de selección y de entrega
 - **funcionGetEmailType.php:** para obtener el tipo de un email dado [8]
 - **general.php:** acciones comunes a todas las secciones como controlar que el usuario tenga la sesión iniciada, pintar la cabecera del sistema...
 - **grupos.php:** con el formulario para la creación de grupos
 - **login.php:** para realizar el inicio de sesión al sistema
 - **loginLDAPUPM.php:** para realizar el inicio sesión con el LDAP [8]
 - **logs.php:** archivo encargado de registrar los errores del sistema
 - **seleccionAsiganturas.php:** sección donde se muestran las asignaturas del usuario
 - **seleccionPracticas.php:** sección donde se muestran las prácticas de una asignatura
 - **subirArchivos.php:** sección que maneja la subida de archivos y muestra el resultado de la corrección de la entrega
- **/BD:** para almacenar todas las acciones que requieran interacción con la BD, incluyendo conexión, consultas, inserciones... (gestionBD.php y conexión.php)

El flujo del sistema, ya mencionado anteriormente, se sigue a través del siguiente orden:

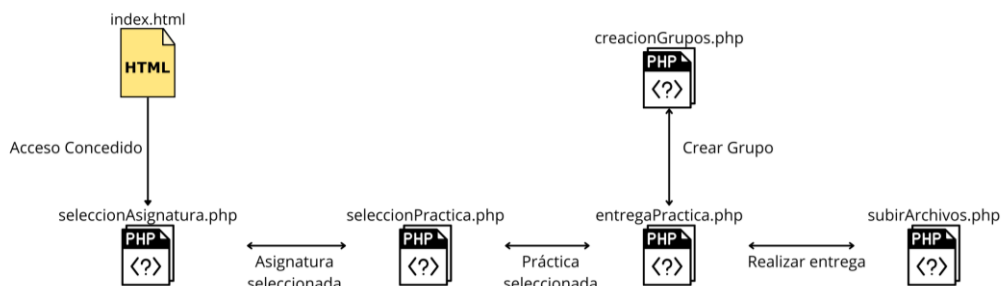


Figura 27: Implementación del Flujo del Sistema

Además de la estructura de directorios del sistema, como se trató en el apartado **Corrección de la Entrega**, según las asignaturas y las prácticas, se definirá otro árbol de directorios ajeno al sistema de la siguiente manera:

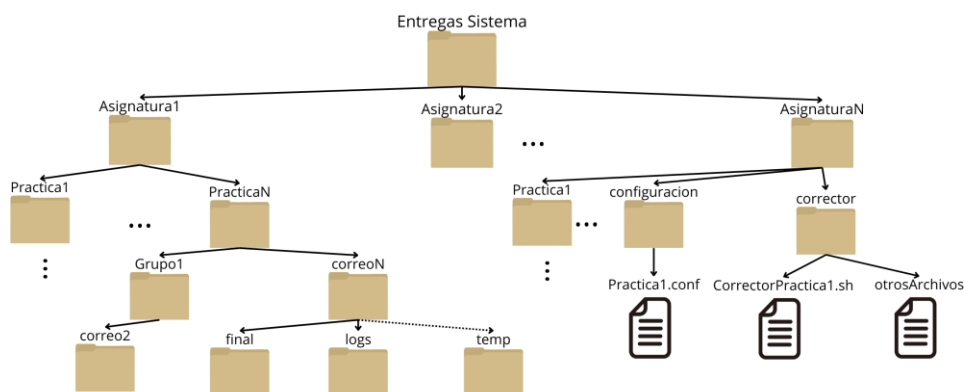


Figura 28: Árbol de Directorios del Almacenamiento de Archivos

Donde cada práctica tendrá su directorio en su correspondiente asignatura, y según el tipo de entregante se generará el directorio del usuario que realiza la entrega como hijo directo de la práctica o como hijo del grupo si es el representante de uno.

En cada asignatura, también se encontrarán otros 2 directorios:

- Configuración: para guardar los archivos de configuración de las prácticas
- Corrector: para guardar los scripts de corrección y todos los archivos requeridos por cada profesor y cada práctica

2.8 Base del Sistema

Una vez definida toda la estructura del sistema, y antes de explicar la implementación se deben tratar varios temas para poder usar el sistema en un entorno de desarrollo

Como en mi caso estoy usando el sistema operativo Windows 10, lo primero que hay que hacer es la instalación de la MV Linux con todos los archivos y aplicaciones necesarias para la ejecución del sistema. Para más información ver anexo **Instalación MV**.

Una vez se tenga acceso a la MV, serán necesarios diferentes entradas o información en la BD para poder hacer uso del sistema.

Primero, se crea el esquema 'SistemaEntregaPracticas', donde se incorporan todas las entidades y relaciones que se han visto anteriormente. Por ahora la BD solamente tendrá los usuarios con correo electrónico asociado: marco.prieto@alumnos.upm.es, alumno@prueba.es, profesor@prueba.es, inigo@prueba.es.

Como estos últimos no son considerados como válidos, ya que no están el LDAP, si se accede con dichos correos, solo se comprobará la BD en lugar de la comprobación completa. Esto se hace para poder definir pruebas del sistema sin necesitar credenciales LDAP de terceros.

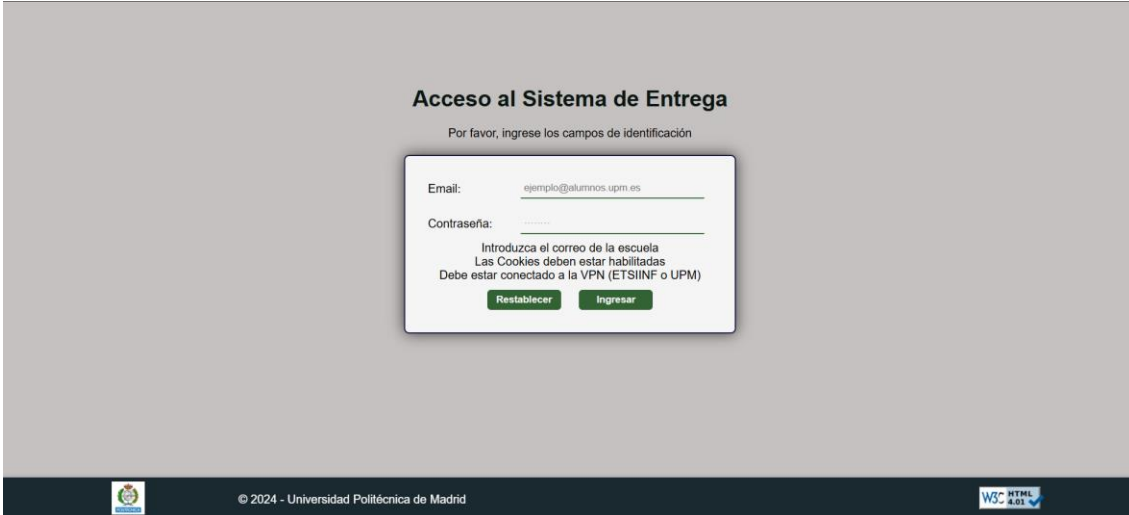
También es importante mencionar la simulación de scripts de corrección de las prácticas. Como se especificó anteriormente, cada práctica tiene un script de corrección que realiza diversas tareas, y para un uso y comprobaciones prácticas, en lugar de corregir realmente las entregas se generarán salidas y errores específicos para emular diferentes casos

Estos temas se tratarán más concretamente en el aparato **Pruebas**

Con todo esto planteado, ya se puede pasar a la implementación de todos los elementos que forman parte del flujo del sistema

2.9 Index.html y login.php

Este archivo HTML será de utilidad para que el usuario intente acceder al sistema, por tanto, seguirá el diseño que se ha mencionado en el apartado **Inicio de Sesión**.



The screenshot shows a login form titled "Acceso al Sistema de Entrega". Below the title is the instruction "Por favor, ingrese los campos de identificación". The form contains two input fields: "Email:" with the placeholder "ejemplo@alumnos.upm.es" and "Contraseña:". Below the password field, there are three lines of text: "Introduzca el correo de la escuela", "Las Cookies deben estar habilitadas", and "Debe estar conectado a la VPN (ETSINF o UPM)". At the bottom of the form are two buttons: "Restablecer" and "Ingresar". The footer of the page includes the logo of the Universidad Politécnica de Madrid, the copyright notice "© 2024 - Universidad Politécnica de Madrid", and a "W3C HTML 4.01" logo.

Figura 29: Inicio de Sesión del Sistema en "index.html"

Se carga el formulario de inicio de sesión y se envían las credenciales introducidas a "login.php", donde se maneja el acceso del usuarios. Primero con el desarrollo previo a este TFG, se usan los archivos "funcionGetEmailType.php" y "LoginLDAPUPM.php" para verificar que el usuario pertenece a la universidad. En caso afirmativo, mediante el método "validación" de "gestionBD.php", se busca una entrada que coincida con el correo electrónico proporcionado en la tabla Usuarios de la BD, lo verifica (si no lo estaba), se actualiza el último acceso y se recuperan los datos del usuario (nombre, apellidos, idUsuario, rol...) mediante el método "datosBasicosUsuarios" para introducirlos en la sesión PHP. Si todo salió correctamente el usuario pasa a la siguiente sección, en caso contrario, se reenvía al usuario a "index.php" nuevamente, mostrando un mensaje de error debajo del formulario.

2.10 SeleccionAsignaturas.php

Este documento será el encargado de mostrar el formulario de asignaturas para que los usuarios puedan seleccionar una de ellas. Para ello será necesario el método "cargarAsignaturas" que mediante el idUsuario (de la sesión) busca las asignaturas en las que dicho usuario está matriculado en el curso y convocatoria actuales.

Este método devuelve los nombres, abreviaturas y número de asignaturas encontradas.

Si este número es cero, se pide al usuario que se comunice con el profesor correspondiente, ya que, si tiene acceso al sistema, se espera que al menos haya una asignatura en la que se encuentre matriculado

Si el número es mayor que cero, se llama al método “FormularioSeleccion” del archivo “formulario.php”

Como se mencionó anteriormente, el formulario envía las abreviaturas, en este caso, las abreviaturas son los idAsignatura, por lo que, si un usuario selecciona, por ejemplo, “Programación II”, se enviará la selección “asignatura = PROGII”

2.11 SeleccionPracticas.php

La implementación de este archivo es prácticamente análoga a la del anterior apartado.

De la sesión se recuperan el idUsuario y el rol, y se pasan como parámetro al método “cargarPracticas”, el cual se encarga de recuperar las prácticas de la asignatura anteriormente seleccionada según el rol, es decir, solo las activas para los alumnos y para los profesores también las que están en desarrollo.

Y al igual que el método en el anterior apartado, este devuelve las abreviaturas, nombres y número de prácticas encontradas en cada caso

A diferencia de **SeleccionAsignaturas.php**, se prevé que pueda no haber prácticas asignadas en un momento determinado, por lo que, si el número de prácticas encontradas es cero, solo se mostrará un mensaje indicando que no hay prácticas en ese instante.

Nuevamente, si el número es mayor que cero se llama al método “FormularioSeleccion” del archivo “formulario.php” para enviar la selección al siguiente archivo

Al igual que en el anterior apartado, el formulario emplea las abreviaturas para identificar a las prácticas, usando el idPractica para este formulario. Por tanto la selección enviada a la siguiente sección será de la forma “idPractica=23”

2.12 EntregaPractica.php

Lo primero que se debe hacer es, recuperar la información de la práctica y, gracias al identificador de la práctica recibido del anterior formulario y los métodos “cargarPractica” y “cargarSlot” del gestor de BD se puede obtener toda la información relacionada con dicha práctica. Con dichos datos, y si el rol que se ha guardado en la sesión previamente es alumno se procede a la comprobación tanto del estado de la práctica como el rango de fechas para la entrega de la práctica, y si la práctica no está activa o el alumno se encuentra fuera del rango de fechas permitido no se permiten más acciones

Después, como se comentó en el apartado **Entrega de Prácticas**, hay varios casos según los números máximos y mínimos de integrantes, si se ha entregado anteriormente....

- Veamos primero si el usuario no ha entregado, es decir, no tiene entregante asociado para esa práctica. Para ello empleamos el método “buscarEntregante” del gestor de BD, que mediante los identificadores de usuario y de la práctica busca alguna coincidencia en la tabla de Entregantes, que devolverá NULL o los datos del entregante si la búsqueda fue exitosa, en este caso devolvería NULL.
 - Si la práctica es obligatoriamente individual, se muestra directamente el formulario de entrega.
 - En cambio, si es obligatoriamente en grupo o se puede hacer de ambas maneras, se debe comprobar si pertenece a un grupo. Gracias al método “buscarGrupo”, con los mismos parámetros que el anterior, se puede saber si el usuario pertenece o no a un grupo. Si pertenece a un grupo y no es el representante no se permite avanzar más al usuario. Y si no pertenece a ningún grupo para dicha práctica, en función del tipo de práctica se muestra solamente el botón de crear grupo o el botón junto al *checkbox* para confirmar la entrega individual y mostrar el formulario de entrega
- Ahora veamos si el usuario ya es entregante, es decir, ya ha realizado entregas anteriormente. Mediante el método “entregasDisponibles” se obtienen y se muestran las entregas restantes que puede realizar el usuario. Si dicho número es cero se detiene la ejecución del código, si no se carga el formulario de entrega

Para poder mostrar el formulario son necesarios los nombres de los archivos que se van a corregir en cada práctica, ubicados en el fichero de configuración de la práctica, junto con el nombre del script de corrección como se comentó en el apartado **Archivos de configuración de las Prácticas**. Y para obtener dicha información se emplea el método “leerFichero” de la clase Fichero(“fichero.php”), que guarda en un array asociativo los nombres y tamaños máximos permitidos de los ficheros y el nombre del script de corrección

El formulario de entrega se cargará, mediante el método “formularioEntrega” del archivo “formulario.php”, de manera oculta al principio, y se mostrará según se requiera y se permita mediante JS con el método “toggleFormulario” para el *checkbox*, y otras funciones de comprobación automática

2.13 CreacionGrupos.php y grupos.php

Al pulsar el botón Crear Grupo de la anterior sección, se redirige al usuario a “creacionGrupos.php”, donde se debe mostrar el formulario para crear un grupo

Evidentemente se tienen que hacer las comprobaciones previas de que el usuario que va a crear el grupo todavía no sea entregante, y que no pertenezca a ningún grupo para dicha práctica (igual para cada miembro del grupo que se desea añadir)

Debido a la cantidad de acciones de este formulario (insertar miembros, borrar miembros, eliminar todos los miembros... vistas en el apartado **Formulario de Creación de Grupo**), en lugar de incluir este formulario en el archivo “formulario.php” como el resto, se ha decidido mantenerlo aislado en el archivo “grupos.php”, dividido en las funciones “permisoIngresarGrupo” y “formularioGrupo”

La primera función devuelve si el usuario que ha ingresado sus credenciales tiene permiso para acceder al grupo, o en cambio el motivo por el que no puede acceder por ejemplo, no tiene esa asignatura (con el método “tieneAsignaturaPractica” del gestor de BD que comprueba si un usuario pertenece a la asignatura asociada a la práctica dada), ya pertenece a un grupo, ya ha entregado...

Y la segunda función, “formularioGrupo” se encarga de manejar todas las acciones mencionadas anteriormente. Cada acción está dividida en diferentes formularios o botones que envían la información al mismo archivo, es decir,

cada vez que se realiza alguna acción, se envían así mismos los datos y se recarga la página, actualizando tanto el formulario como los datos que representa (correos electrónicos).

- **Formulario de “Inicio de sesión”**: (para ingresar credenciales) llama a la función “permisoIngresarGrupo” con el identificador de la práctica, y correo y la contraseña introducidos y según el valor devuelto imprime un mensaje de error o de éxito. Si se concede permiso a un usuario, se toma una lista de la sesión en la que se añade su correo electrónico (el usuario que está creando el grupo siempre está).

Como la lista se guarda en la sesión es necesario hacer una lista por cada práctica para que no solape los grupos en caso de uso simultaneo del sistema, así un mismo usuario podrá crear varios grupos seguidos sin problema.

- **Los botones de “Eliminar” y “Eliminar todos los miembros”**: que eliminan a un miembro específico del grupo o a todos los miembros (sin contar el usuario que está creando el grupo). Solo se activan si ya se ha introducido algún miembro a parte del representante del grupo
- **El botón de “Confirmar Grupo”**: se activa una vez se han alcanzado el mínimo de integrantes requerido y permite crear el grupo, mostrando antes un mensaje de que esta acción será definitiva y no se podrá modificar el grupo en ningún momento una vez esté creado. Cuando es pulsado se activa el cuadro de dialogo pidiendo confirmar el grupo mediante un *checkbox*. Una vez se recibe la confirmación del grupo se llama al método del gestor de BD “crearGrupo”, con el identificador de la práctica y la lista de correos y redirige al usuario de nuevo a “entregaPractica.php”

Para evitar problemas o ataques de concurrencia, como que un usuario intente ingresar a la vez en dos grupos diferentes para una misma práctica u otros problemas similares, tanto en el método “crearEntregante”(mencionado más adelante) como en “crearGrupo” se emplearán transacciones (si ocurre algún problema deshace los cambios realizados durante la ejecución de la transacción) con bloqueos para evitar que diferentes usuarios accedan a la misma información a la vez

2.14 SubirArchivos.php

Los usuarios accederán a esta sección una vez se han enviado los archivos correspondientes a través del formulario de entrega

Antes de empezar a corregir, de manera similar al apartado **Entrega de Prácticas**, se deben hacer las comprobaciones de entregante y grupo para comprobar que puede realizar la entrega o para asociar dicha entrega a un entregante. También servirá para crear las rutas de guardado de archivos como se trató en los apartados **Corrección de la Entrega** y **Estructura del Sistema**

- Si el usuario no es entregante, la práctica puede ser obligatoriamente en grupo o no.
 - Si no tiene grupo y es obligatoriamente en grupo, no se permite la entrega, en cambio si no es obligatoriamente en grupo, se crea un entregante individual
 - Si tiene grupo, se comprueba que sea el representante de dicho grupo y se crea entregante grupal
- Si el usuario ya es entregante solo se comprueba si es entregante grupal para incluir su identificador en la ruta de guardado

Una vez se han hecho las comprobaciones, las posibles creaciones de entregantes, y se define la ruta de guardado, se pueden guardar los archivos en una carpeta temporal y proceder a la corrección de la entrega

En ese momento se debe llamar al script de corrección(con los archivos y el identificador de la práctica), que es el encargado de ejecutar las pruebas correspondientes, y en función de esas pruebas, debe devolver la validez o estado de la entrega y los correspondientes códigos de error

- Si la entrega es no válida, se informa que la entrega no ha sido aceptada
- Si es válida, tanto como si es apta o no apta, se informa de la validez y estado de la entrega y se muestran los errores generados en la corrección

Para mostrar la descripción de los errores, se emplea el método “descripcionError” del gestor de BD, que recibe el código de error y el identificador de la práctica. Si lo encuentra, devuelve la cadena de texto que describe el error, si no, se escribe la cadena “Error desconocido”

2.15 Pruebas

En este apartado realizaremos diferentes pruebas para simular de la manera más realista posible el funcionamiento del sistema, como si se tratase de un nuevo usuario que accede al sistema y quiere realizar las diferentes entregas que tiene disponibles

2.15.1 Ingreso de Datos

Como se trató en el apartado **Base del Sistema**, para poder usar y probar el sistema se necesita una BD con entradas que permitan acceder y navegar por el sistema, y otros archivos como archivos de configuración de las prácticas, los scripts de corrección...

Para ello, empezaremos exportando el esquema de la BD actual, para tener solo las entidades, relaciones y sus tablas correspondientes. Antes de ello, para evitar cualquier conflicto o error, se crea un punto de control en la MV por si al exportar el esquema surge algún problema poder deshacer dichas acciones

Una vez exportada la estructura, la importamos en un nuevo esquema llamado 'SistemaEntregaPracticasPrueba1'. Como se comentó en el apartado **Actores**, más concretamente en la sección del gestor de BD, el sistema trabaja bajo un usuario con ciertos privilegios o permisos. Y al igual que en el anterior esquema, 'SistemaEntregaPracticas', se le deben conceder los permisos de INSERT, SELECT, UPDATE y EXECUTE al usuario que realizará todas las acciones en la BD del sistema. Esto se tendrá que repetir en todos los esquemas que se repliquen en el futuro

A este esquema se le añadirán los mismos usuarios que se mencionaron en el apartado **Base del Sistema**, además del curso académico 24/25 con todas sus convocatorias (febrero, junio y julio) junto con diversas asignaturas.

Para las diferentes convocatorias se han añadido las siguientes asignaturas

- Febrero: Programación I y II, Algoritmos y Estructuras de Datos, y Redes y Comunicaciones
- Junio: Sistemas Operativos
- Julio: todas las asignaturas anteriores

Para poder visualizar rápidamente a que asignatura se ha matriculado cada usuario vemos la siguiente tabla

usuario	idAsignatura	anioAcademico	convocatoria
marco.prieto	AED	24/25	feb
marco.prieto	PROGII	24/25	feb
marco.prieto	RC	24/25	feb
marco.prieto	SO	24/25	jun
alumnoFicticio	PROGI	24/25	feb
alumnoFicticio	RC	24/25	feb
alumnoFicticio	SO	24/25	jun
alumnoFicticio	AED	24/25	feb
Íñigo	AED	24/25	feb
Íñigo	PROGII	24/25	feb
Íñigo	SO	24/25	jun

Tabla 11: Asignaturas en las que hay usuarios matriculados de 'SistemaEntregaPracticasPrueba1'

El profesor ficticio no se representa en esta tabla ya que está matriculado en todas las asignaturas y convocatorias posibles

Ahora veamos las prácticas existentes en cada asignatura

idAsignatura	nombre	estado	maxEntregas	minIntegrantes	maxIntegrantes
AED	Listas Indexadas	activa	5	1	1
AED	HashMaps	activa	5	1	2
AED	Arboles	desarrollo	5	1	2
PROGI	Tipos de Datos	activa	10	1	1
PROGI	Bucles	activa	10	1	1
PROGI	Lectura de Ficheros	desarrollo	10	1	1
PROGII	Introduccion POO	activa	10	1	1
PROGII	Herencia	activa	10	2	2
PROGII	Interfaces	desarrollo	10	1	1
RC	Direccionamiento IP	activa	10	1	1
RC	Diseño de Red	activa	8	2	3

Tabla 12: Prácticas de las Asignaturas Anteriores

También se definen los Slots para establecer las fechas de inicio y fin de las prácticas. Todas tendrán la misma fecha de inicio (2025/01/01) y de fin (2025/02/28), excepto la primera práctica de AED (Práctica de Listas Indexadas) que tiene la fecha de cierre en 2025/01/10 (aaaa/mm/dd).

Y por último una lista de errores genéricos para el baremo de todas las prácticas

Ahora, antes de empezar a probar el sistema, exportamos este esquema de la BD para tener un “punto de control” al que regresar en el caso de que haya un error, y así poder replicar más fácilmente las condiciones que se dieron para generar dicho error y poder corregir el comportamiento indeseado

Es importante mencionar también el modo de empleo de los archivos de configuración y de corrección de la práctica. Para simular su implementación se usarán archivos comunes para todas las prácticas, contando con las siguientes características:

- **Archivo de configuración:** su información más relevante es el nombre de los ficheros que se van a usar (“archivo1.java” y “archivo2.java”) y el nombre del script de corrección (“corrector.sh”)
- **Script de Corrección:** un sencillo script de Bash que escribe por terminal, según se requiera, el estado, validez y errores de la entrega. Es decir, simula una corrección automática mediante salidas preconfiguradas

Estos archivos se encuentran en el anexo **Archivo de configuración y Script de Corrección**

2.15.2 Validación Bajo Diferentes Escenarios

Para probar el sistema veamos las diferentes fases del flujo de usuarios mencionado al principio

- **Asignaturas y Prácticas:** para poder usar el sistema, lo primero que se debe comprobar es que se muestren las asignaturas y prácticas correspondientes a cada usuario (además de que permita ingresar a los usuarios que deben ingresar y rechazar a los que no) Se muestra el caso del profesor ficticio, usuario al cual se le deben mostrar todas las asignaturas y prácticas (activas o en desarrollo) de la convocatoria actual (febrero 24/25)

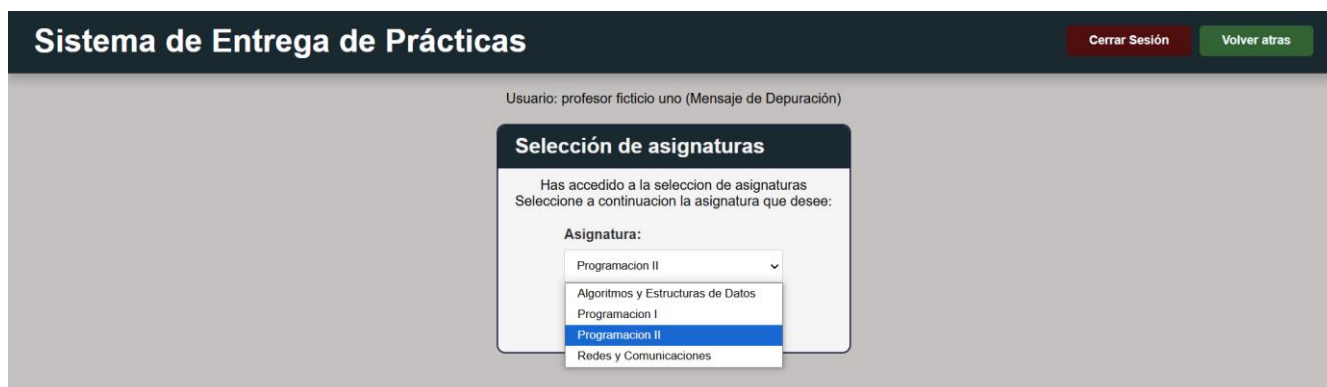


Figura 30: Asignaturas Profesor Ficticio

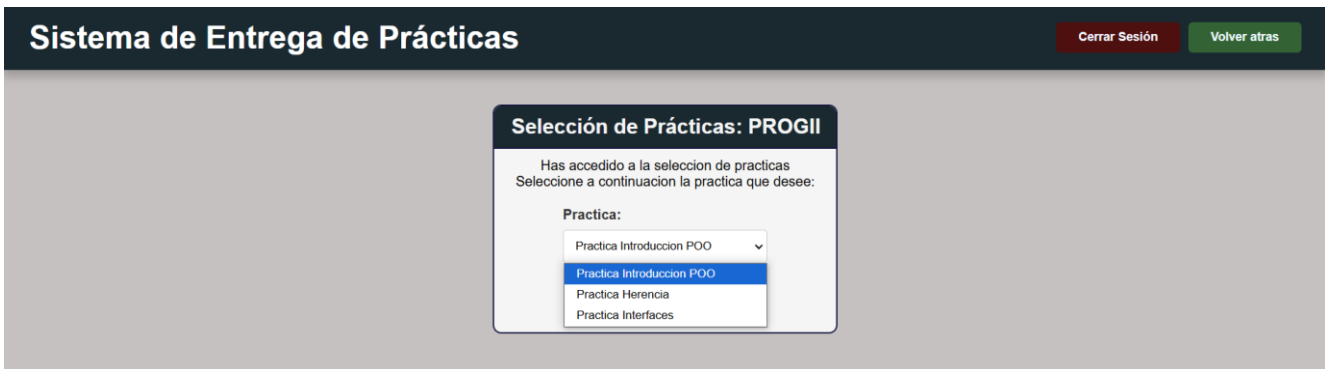


Figura 31: Prácticas de PROG II para Profesor Ficticio

- Información de la Práctica:** cada práctica tiene su información específica, pero al emplear un archivo de configuración común, el formulario de entrega será igual en todas las prácticas. Aquí se debe emplear cualquier usuario de tipo alumno para comprobar el rango de fechas, ya que los profesores tienen acceso al resto del archivo “entregaPracticas.php” independientemente de las fechas de inicio o fin. Por ejemplo, como hemos visto, la práctica “Listas Indexadas” de AED tiene una fecha de cierre el 10 de enero de 2025 por lo que no está disponible, mientras que el resto de las prácticas son accesibles por el momento.

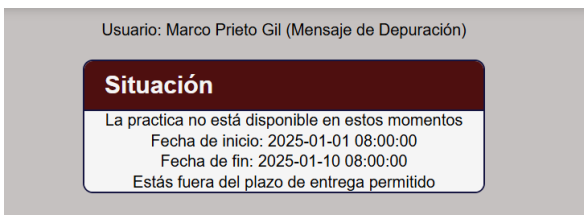


Figura 32: Práctica Fuera del Plazo Permitido

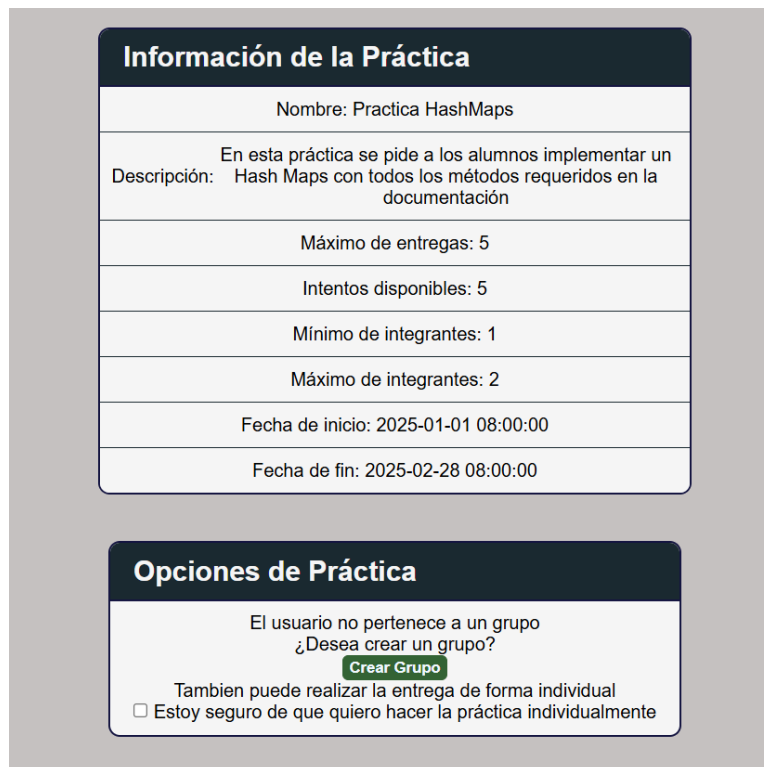


Figura 33: Información de la Práctica

- **Crear Grupos:** antes de realizar entregas, vamos a definir algunos grupos para las prácticas que lo permiten u obligan su creación. El representante será el primer usuario y aparecerá resaltado

Práctica	Usuarios
HashMaps (AED)	Marco Prieto e Íñigo
Herencia (PROG II)	Profesor Ficticio e Íñigo
Diseño de Red (RC)	Marco Prieto , Profesor Ficticio y Alumno Ficticio

También se puede aprovechar para comprobar el ingreso de integrantes al grupo, por ejemplo, Alumno Ficticio no puede ingresar a ningún grupo de AED

Figura 34: Intento Incorrecto de Ingreso al Grupo 1

Figura 35: Intento Incorrecto de Ingreso al Grupo 2

Y una vez creados los grupos comprobamos que solo los representantes tengan acceso al formulario de entrega

The screenshot shows the 'Sistema de Entrega de Prácticas' interface. At the top right, there are buttons for 'Cerrar Sesión' and 'Volver atrás'. The user is identified as 'Usuario: Íñigo Ficticio Uno (Mensaje de Depuración)'. The main content is a table titled 'Información de la Práctica' with the following details:

Información de la Práctica
Nombre: Practica HashMaps
Descripción: En esta práctica se pide a los alumnos implementar un Hash Maps con todos los métodos requeridos en la documentación
Máximo de entregas: 5
Intentos disponibles: 5
Mínimo de integrantes: 1
Máximo de integrantes: 2
Fecha de inicio: 2025-01-01 08:00:00
Fecha de fin: 2025-02-28 08:00:00

Below the table is a 'Situación' box with the message: 'Ya pertenece a un grupo. No puede realizar entregas. Solo el representante del grupo tiene permiso.'

Figura 36: Miembro del Grupo Sin Acceder al Formulario de Entrega

- **Crear Entregantes y Realizar Entregas:** por último, realicemos algunas entregas con diferentes estados y validez, comprobando la interfaz de usuario desplegada tanto en la entrega como en la corrección. Primero veamos la reacción del formulario ante la selección de diferentes archivos.

The figure shows two side-by-side screenshots of the 'Formulario de Entrega' interface. Both screens show a header 'Formulario de Entrega' and a prompt 'Seleccione a continuación todos los archivos:'. The interface is divided into two sections for file selection.

Left Screenshot: The first section shows 'archivo1.java' with a green checkmark and a 'Seleccionar archivo' button. The second section shows 'archivo2.java' with a red 'X' and a 'Seleccionar archivo' button. At the bottom, there is a 'Subir todos los archivos' button.

Right Screenshot: The first section shows 'archivo1.java' with a green checkmark and a 'Seleccionar archivo' button. The second section shows 'archivo2.java' with a green checkmark and a 'Seleccionar archivo' button. At the bottom, there is a 'Subir todos los archivos' button.

Figura 37: Formulario de Entrega del Sistema

Y una vez se selecciona “subir todos los archivos”, probaremos las 2 posibilidades:

- **Entrega Inválida** como se trató anteriormente, si no es válida no se tiene en cuenta para el resto de las entregas, aunque sí se registra en la BD

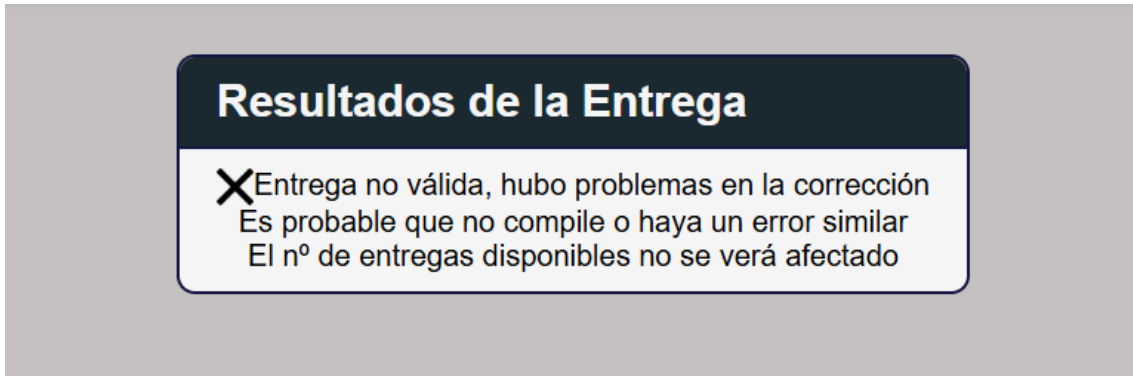


Figura 38: Entrega Inválida

- **Entrega Válida:** estas entregas sí que restan intentos en las prácticas, y puede haber otros 2 casos, aunque siempre se mostrarán los errores generados si existen
 - **Entrega No Apta:**

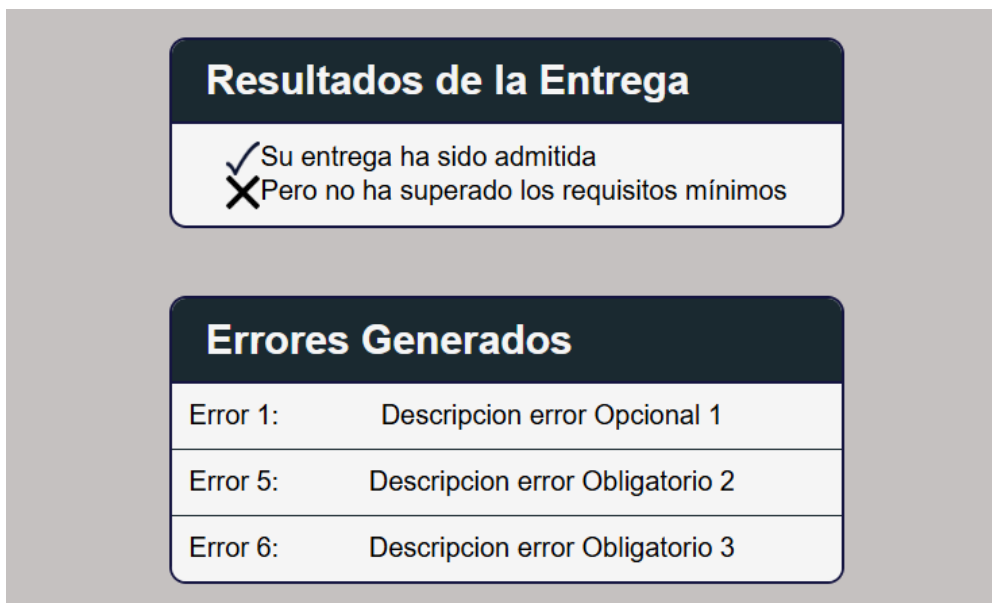


Figura 39: Entrega Válida pero No Apta

- **Entrega Apta:**

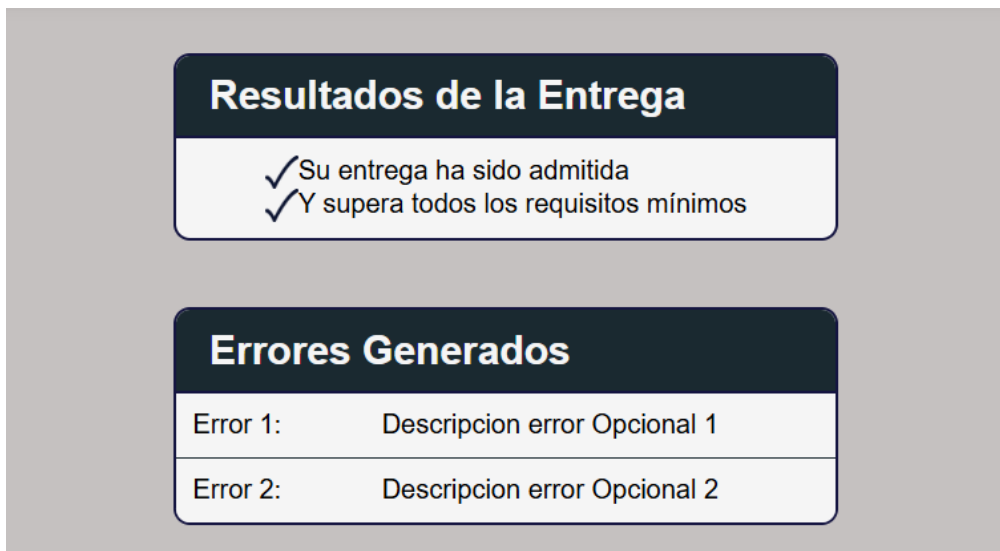


Figura 40: Entrega Válida y Apta

Para concluir, las pruebas aquí presentadas son solo una muestra representativa de las verificaciones realizadas. Durante el desarrollo, el sistema ha sido sometido a un proceso de pruebas y ajustes, asegurando su correcto funcionamiento en cada etapa de implementación

3 Resultados y conclusiones

3.1 Resultados

El desarrollo del sistema de entrega de prácticas ha alcanzado los objetivos planteados, logrando diseñar e implementar una solución funcional, intuitiva y adaptable para gestionar la entrega y corrección de prácticas académicas.

Entre los resultados más destacados se encuentran:

1. **Sistema de autenticación robusto:** Gracias a trabajos anteriores, la integración con LDAP para autenticar usuarios de manera segura y eficiente fue muy sencilla
2. **Gestión eficiente de asignaturas y prácticas:** Creación de una interfaz con la que los usuarios pueden interactuar fácilmente con el sistema, mostrando asignaturas y prácticas relevantes de manera dinámica
3. **Automatización de correcciones:** El sistema permite a los profesores cargar sus scripts personalizados que evalúan automáticamente las prácticas entregadas, proporcionando retroalimentación inmediata
4. **Modelado de base de datos optimizado:** Diseño de un modelo relacional que minimiza la redundancia y garantiza la integridad de los datos
5. **Adaptabilidad y escalabilidad:** Uso de tecnologías accesibles y entornos virtuales para garantizar un despliegue seguro y eficiente

3.2 Conclusiones

Gracias al apoyo en el anterior sistema, y sobre todo, a la guía de mi tutor, Ángel Lucas González Martínez, la realización de este proyecto ha supuesto una experiencia enriquecedora, tanto a nivel técnico como personal.

Entre las principales conclusiones destacan:

1. **Crecimiento profesional:** La implementación del sistema ha consolidado habilidades en programación, diseño de bases de datos y gestión de proyectos proporcionando experiencia práctica y de utilidad
2. **Impacto potencial:** Imaginar cómo el sistema, en el futuro, podría evolucionar e implementarse en la escuela ha sido una motivación constante.

3. **Planes a futuro:** Aunque los objetivos principales se han cumplido, como se mencionó en el apartado **Alcance**, aún faltan diferentes objetivos para que permitir a los usuarios (principalmente profesores) trabajar de forma completamente autónoma en el sistema. Entre los planes a futuro más destacados se encuentran:
- i. **Interfaz de profesores:** Para poder publicar prácticas, modificar fechas de entrega, ver otras convocatorias más allá de la actual... Sin necesidad de hacerlo mediante modificaciones a mano en la BD
 - ii. **Interfaz de usuarios:** Para que los usuarios puedan ver la información de entregas realizadas anteriormente, no solo de la práctica recién entregada
 - iii. **Servicio de correo electrónico:** Comunicar los resultados de la entrega mediante correo electrónico, dando opciones para en el caso de ser un grupo enviar la información al resto de los miembros
 - iv. **Implementación de Grupos de Matriculación:** Así se podría definir una misma práctica para diferentes grupos de matriculación, asignando fechas de inicio y fin diferentes para cada uno
 - v. **Mejora en la Gestión de Concurrencia:** Actualmente, la concurrencia se ha implementado de manera limitada, únicamente en las funciones “crearEntregante” y “crearGrupo” del gestor de la BD, mediante el uso de transacciones con bloqueos. Sin embargo, no ha sido posible realizar pruebas exhaustivas para evaluar su eficacia en escenarios reales, por lo que sería ideal ampliar esta funcionalidad a otros puntos críticos del sistema y validar su comportamiento en un entorno más exigente

En conclusión, este Trabajo de Fin de Grado cumple con los objetivos planteados y abre la puerta a nuevas posibilidades y evoluciones en el sistema orientadas a la digitalización y optimización de procesos educativos

4 Análisis de Impacto

El sistema de entrega de prácticas propuesto en este TFG no solo aborda una necesidad tecnológica, sino que también genera un impacto notable en diversos ámbitos.

A continuación, se detallan dichos impactos en diferentes contextos:

1. **Impacto Personal:** Como se ha comentado anteriormente, el desarrollo del proyecto me ha permitido adquirir y consolidar conocimientos técnicos en tecnologías como PHP, MySQL y JavaScript, además de experiencia en metodologías de desarrollo de software, fomentando competencias clave como la gestión de proyectos, la resolución de problemas y la colaboración interdisciplinaria
2. **Impacto Empresarial:** El sistema tiene potencial para integrarse en otras instituciones o corporaciones educativas gracias al uso del LDAP para la autenticación de usuarios
3. **Impacto Económico:** Al reducir la dependencia de sistemas comerciales costosos y emplear tecnologías accesibles y de uso local, el sistema propuesto podría generar ahorros significativos para instituciones educativas
4. **Consideraciones en el Marco de los ODS:** El sistema está enfocado en alinearse con los Objetivos de Desarrollo Sostenible (ODS), más concretamente con el ODS 4 (Educación de Calidad), al mejorar el acceso y la gestión de herramientas de aprendizaje, el ODS 9 (Industria, Innovación e Infraestructura), al fomentar el uso de tecnologías innovadoras en el sector educativo, y el ODS 12 (Producción y Consumo Responsables) y 13 (Acción por el Clima), minimizando el impacto ecológico de los procesos educativos ya que la digitalización de entregas disminuye el uso de papel y otros recursos físicos

A lo largo del desarrollo, se priorizaron decisiones que favorecieran la accesibilidad y sostenibilidad del sistema, como la elección de tecnologías familiares, que facilita el mantenimiento por futuros desarrolladores. Además, el uso de máquinas virtuales asegura un entorno controlado, minimizando riesgos operativos y garantizando un desarrollo seguro y escalable

5 Bibliografía

- [1] (2024, octubre 6) Actividad de Tarea [Online] Available: [https://docs.moodle.org/all/es/Actividad de Tarea](https://docs.moodle.org/all/es/Actividad_de_Tarea)
- [2] Elías Herrero Lázaro. “Monitorización de Entregas” TFG Grado de Ingeniería Informática, Universidad Politécnica de Madrid, Escuela Técnica Superior de Ingenieros Informáticos, Madrid, España, junio 2023
- [3] Sean Blanton. (2023, febrero 17). ¿Qué es la autenticación LDAP? [Online]. Available: <https://jumpcloud.com/es/blog/what-is-ldap-authentication>
- [4] Rubén Velasco (2024, enero 22) VMware, VirtualBox o Hyper-V - ¿Qué programa es mejor? [Online]. Available: <https://www.softzone.es/programas/utilidades/diferencias-vmware-virtualbox-hyper-v/>
- [5] (2024, diciembre)VirtualBox | Virtualización Available: <https://www.oracle.com/es/virtualization/virtualbox/>
- [6] Stephanie Susnjara, Ian Smalley (2024 agosto 15) ¿Qué es VMware? Available: <https://www.ibm.com/es-es/topics/vmware>
- [7] Makenzie Buenning (2024 abril 12) ¿Qué es Hyper-V?. Available: <https://www.ninjaone.com/es/blog/hyper-v-caracteristicas-y-capacidades/>
- [8] Ignacio Manjón Baranda. “Actualizar el Sistema de Entrega de Prácticas de la Asignatura Programación II” TFG Grado de Ingeniería Informática, Universidad Politécnica de Madrid, Escuela Técnica Superior de Ingenieros Informáticos, Madrid, España, enero 2024

6 Anexos

6.1 Instalación MV

Primero se debe instalar o habilitar Hyper-V en Windows 10 o Windows 11.

En mi caso, tengo la versión Windows 10, y los pasos para instalar la MV Linux son los siguientes:

1. Evidentemente, se deben tener los archivos de máquina virtual y de disco virtual preparados
2. En Hyper-V seleccionar editar disco (a la derecha en menú vertical Acciones)
3. Seleccionar el disco (Virtual Hard Disk, extensión .vhdx)
4. Pulsar Siguiente y en las acciones elegir 'Convertirlo'
5. Pulsar Siguiente y elegir formato VDHX
6. Pulsar Siguiente y elegir 'Expansión Dinámica'
7. Pulsar Siguiente y elegir la ruta donde estará la MV, por ejemplo, C:\ProgramData\Microsoft\Windows\Virtual Hard Disks\Linux
8. Tras terminar la importación hay que crear una nueva máquina virtual, darle nombre, por ejemplo, Linux
9. Pulsar Siguiente y seleccionar Generación 2
10. Pulsar Siguiente y seleccionar 4096 MB de memoria RAM
11. Pulsar Siguiente y, en configuración de red seleccionar 'Default Switch'
12. **IMPORTANTE.** EN 'Conectar disco duro virtual' hay que seleccionar 'Usar disco duro virtual existente' y poner la ruta (paso 7) al archivo .vhdx que se ha realizado anteriormente, por ejemplo, C:\ProgramData\Microsoft\Windows\Virtual Hard Disks\Linux.vhdx
13. Pulsar Finalizar, pero **IMPORTANTE**, NO ARRANCAR LA MÁQUINA VIRTUAL TODAVÍA
14. Seleccionar la máquina, y pulsar 'Configuración'
15. Ir a la sección 'Seguridad' y asegurarse de que el *checkbox* de 'Habilitar arranque seguro' está desmarcado
16. Por último, arrancar la MV para comprobar que se ha instalado correctamente

6.2 Guía de Uso para Profesores

Como se trató en los apartados **Alcance** y **Conclusiones**, ante la falta de interfaz para los profesores, estos deben realizar diferentes tareas fuera de la aplicación para poder definir las prácticas, sus archivos de configuración, de corrección...

Para ello veamos las diferentes acciones que se deben llevar a cabo para poder definir correctamente las prácticas

1. Lo primero de todo es definir los **cursos** y las **asignaturas** en la **BD**, como se ha visto anteriormente, los cursos con año académico (24/25,25/26) y convocatoria (feb, jun o jul) y las asignaturas con identificador (abreviatura) y nombre. También se debe relacionar la asignatura con el curso deseado
2. Crear la **práctica** en la **BD**: definiendo las entradas en las tablas Practicas, *Slots*(fecha inicio y fin) y Baremos (errores para la corrección)
3. Lo siguiente es definir los archivos de configuración y scripts de corrección para la práctica. Será necesario entonces construir la ruta de directorios donde se guardarán los archivos de las entregas, como se comentó en el apartado **Estructura del Sistema**. Por tanto habrá que definir el directorio de la asignatura (nombrando con su abreviatura) y dentro los directorios “corrector” y “configuracion”, en este último se debe guardar el archivo de configuración de la práctica nombrado con el mismo nombre de la asignatura (sustituyendo, si hubiera, los espacios por guiones bajos) seguido de “.conf”. Y evidentemente, en el directorio “corrector”, guardar el script y/o archivos de corrección con el nombre especificado en el archivo de configuración

Con estas definiciones, el resto de los usuarios podrá realizar entregas y utilizar el sistema correctamente. No obstante, existen otras acciones que podrían resultar útiles, como cambiar las fechas de los *slots* para una práctica, modificar las funciones para recuperar curso y convocatoria de la BD, o consultar información sobre las entregas realizadas por los alumnos. Sin embargo, como se ha mencionado previamente, estas acciones deben realizarse directamente de forma manual sobre la base de datos.

6.3 Archivo de configuración y Script de Corrección

- Archivo de configuración: "prueba.conf"

```
N_fichero = archivo1.java
Tamano_Maximo=1048576
N_fichero = archivo2.java
Tamano_Maximo=1048576

Respetar_Mayus = Si

//Numero de entregas máximas (se ignora)
N_Entregas= 20


//Nombre del script
Script = PR2AjedrezTest
```

- Script de corrección: "corrector.sh"

```
#!/bin/bash

# Devuelve la validez
validez="valida"
echo $validez
#Si resultado es invalida salir
if [[ "$validez" == "invalida" ]]; then
    exit 0
fi
estado="apta"
echo $estado
echo "1,2"
```

Este documento esta firmado por



Firmante	CN=tfgm.fi.upm.es, OU=CCFI, O=ETS Ingenieros Informaticos - UPM, C=ES
Fecha/Hora	Tue Jan 14 21:36:46 CET 2025
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)