



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



TFG del Grado en Ingeniería
Informática

Aplicación web de gestión de
PDI en universidades



Presentado por Ignacio Dávila García
en Universidad de Burgos — 28 de junio
de 2023

Tutores: Álgvar Arnaiz González y Carlos
Pardo Aguilar



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D. Álgvar Arnaiz González y D. Carlos Pardo Aguilar, profesores del departamento de Ingeniería Informática, área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. Ignacio Dávila García, con DNI 71755022J, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado *Aplicación web de gestión de PDI en universidades*.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 28 de junio de 2023

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

D. Álgvar Arnaiz González

D. Carlos Pardo Aguilar

Resumen

La gestión académica es un elemento fundamental en las instituciones educativas, ya que involucra la planificación, organización y supervisión de todas las actividades relacionadas con la enseñanza.

Este trabajo de fin de grado se enfocará en proponer una solución para optimizar dicha gestión, teniendo en cuenta el caso particular de la Universidad de Burgos y la forma en la que se trabaja en la misma.

Se creará una aplicación web que permita el mantenimiento académico, el mantenimiento del Personal Docente Investigador (PDI) y la asignación docente de horas, grupos y cursos académicos, siguiendo las necesidades conocidas y marcadas por los tutores de este trabajo para la gestión académica.

Descriptores

Gestión académica, área de conocimiento, departamento, PDI, curso académico, aplicación web.

Abstract

Academic management is a fundamental element in educational institutions as it involves the planning, organization, and supervision of all activities related to teaching.

This final degree project will focus on proposing a solution to optimize this management, taking into account the specific case of the University of Burgos and the way in which it operates.

A web application will be developed that allows for academic maintenance, maintenance of the Teaching and Research Staff (PDI), and the assignment of teaching hours, groups, and academic courses, following the known needs and requirements set by the supervisors of this project for academic management.

Keywords

Academic management, area of knowledge, department, Teaching and Research Staff (PDI), academic year, web application.

Índice general

Índice general	iii
Índice de figuras	v
Índice de tablas	vi
1. Introducción	1
2. Objetivos del proyecto	3
2.1. Objetivos del <i>software</i>	3
2.2. Objetivos técnicos	4
3. Conceptos teóricos	5
3.1. Seguridad	5
3.2. Organización de la Universidad de Burgos	6
4. Técnicas y herramientas	9
4.1. Técnicas	9
4.2. Herramientas	13
5. Aspectos relevantes del desarrollo del proyecto	19
5.1. Diseño	19
5.2. Desarrollo	21
6. Trabajos relacionados	23
7. Conclusiones y Líneas de trabajo futuras	25

Índice de figuras

4.1. <i>Proceso seguido en Scrum [2].</i>	11
4.2. <i>Explicación de los elementos del proceso [2].</i>	11

Índice de tablas

1. Introducción

La gestión del personal docente e investigador es una tarea crítica y compleja para cualquier institución de educación superior, ya que implica el manejo de información confidencial y la coordinación de múltiples actividades administrativas. Por lo tanto, contar con una herramienta informática eficiente y segura para gestionar esta tarea es fundamental.

El presente trabajo de fin de grado tiene como objetivo la creación de una aplicación web para la gestión del personal docente e investigador (PDI) de la Universidad de Burgos utilizando el lenguaje de programación Python y el *framework* Flask.

En este contexto, se ha desarrollado una aplicación web que permite la gestión de información de los docentes e investigadores, así como el acceso a diferentes funcionalidades de forma sencilla y eficiente.

Este trabajo describe el proceso de desarrollo de la aplicación, desde la definición de los requisitos y la arquitectura de la solución, hasta la implementación, despliegue y evaluación de la aplicación.

2. Objetivos del proyecto

Los objetivos del proyecto se centran en desarrollar una aplicación web de gestión académica y del personal docente e investigador (PDI) que satisfaga las necesidades específicas de la Universidad de Burgos. Estos objetivos se dividen en dos categorías principales: los objetivos del software a diseñar y los objetivos técnicos necesarios para llevar a cabo la implementación del proyecto.

2.1. Objetivos del *software*

Los objetivos del software a diseñar e implementar son los siguientes:

1. Desarrollar una aplicación web intuitiva y de fácil uso para la gestión del PDI de la Universidad de Burgos.
2. Centralizar la información del PDI en una base de datos segura y confiable, permitiendo un acceso eficiente y actualización de la información.
3. Agilizar las tareas administrativas relacionadas con el PDI y el mantenimiento académico, tales como la creación de cursos académicos y la asignación de cargas docentes.
4. Proporcionar a los usuarios diferentes funcionalidades, como la consulta, creación y modificación de centros, titulaciones, asignaturas, cursos académicos, plazas, departamentos, etc.
5. Desplegar la aplicación en la nube haciendo uso de alguna plataforma gratuita.

2.2. Objetivos técnicos

Los objetivos técnicos que se persiguen con la realización del proyecto son los siguientes:

1. Utilizar el lenguaje de programación Python y el framework Flask para el desarrollo de la aplicación web.
2. Diseñar una arquitectura modular y escalable, que permita el crecimiento y la evolución futura del sistema.
3. Integrar la aplicación con una base de datos relacional, como SQL, para el almacenamiento seguro y eficiente de la información.
4. Aplicar buenas prácticas de desarrollo de software, como el uso de control de versiones y la documentación adecuada del código.
5. Seguir los requisitos marcados para mejorar el software utilizado actualmente para este fin.
6. Simular una interacción real entre cliente y programador manteniendo reuniones con las que obtener los requisitos necesarios.
7. Documentar el diseño de la aplicación, tanto estructural como de prototipos antes de empezar la codificación.

3. Conceptos teóricos

En este apartado se presentarán los conceptos teóricos fundamentales para la correcta comprensión del trabajo.

3.1. Seguridad

Cross-Site Request Forgery

El *Cross-Site Request Forgery* (CSRF) o falsificación de petición en sitios cruzados es un tipo de ataque en el que un sitio web malicioso «engaña» al navegador de un usuario para que realice acciones no deseadas en otro sitio web en el que el usuario está identificado/a [6]. Por ejemplo, un atacante puede enviar una solicitud HTTP en nombre del usuario autenticado para cambiar su contraseña sin su consentimiento. Para mitigar este tipo de ataques al renderizar un formulario, se genera un token único y se almacenará en la sesión del usuario. Cuando el usuario envía el formulario, el valor del token se incluye en la solicitud HTTP. En el lado del servidor, se verifica si el token enviado coincide con el valor almacenado en la sesión del usuario. Si hay una coincidencia, se considera que la solicitud es válida, pero si el token no coincide o falta, se interpreta como una posible falsificación y se rechaza la solicitud.

Cross-site Scripting

El Cross-site Scripting (XSS) es una vulnerabilidad común en aplicaciones web que permite a un atacante inyectar código malicioso en páginas web visitadas por otros usuarios [7]. Este código malicioso se ejecuta en el navegador de la víctima mediante JavaScript, lo que puede llevar a robo de

información confidencial, manipulación de contenido o redirección a otros sitios web. Para protegerse de este tipo de vulnerabilidades es necesario validar y sanear las entradas que realizan los usuarios. Para ello se utiliza la biblioteca WTForms que proporciona mecanismos para validar y filtrar las entradas de los formularios, lo que ayuda a prevenir la ejecución de código malicioso. También es importante escapar caracteres especiales. En este caso, Flask y las plantillas Jinja2 realizan automáticamente el escape de variables al renderizar las páginas, aun así hay que tener cuidado con las vistas no renderizadas de esta forma.

3.2. Organización de la Universidad de Burgos

La Universidad de Burgos, institución a la que está enfocada la aplicación desarrollada, cuenta con una organización que es importante conocer debido a que la aplicación se basa en esta.

Podemos diferenciar la organización en los siguientes apartados:

Gestión académica

La universidad cuenta con diferentes facultades donde se imparten las titulaciones asignadas a estas, las cuales pueden ser grados o másteres. A su vez, estas titulaciones cuentan con un conjunto de asignaturas repartidas entre los diferentes cursos de la titulación y los diferentes semestres del curso académico. Estas asignaturas son escogidas en cada curso académico, donde se indica cuáles van a ser impartidas. Además, las asignaturas se dividen en grupos, de teoría o práctica, donde se hace un reparto de los alumnos para tener una organización de qué profesores van a impartir los diferentes contenidos de la asignatura, consiguiendo de esta manera mantener número de alumnos adecuado para poder ofrecer una buena docencia.

Gestión de profesorado

La gestión de profesorado incluye todo lo relacionado con los docentes, desde sus contratos hasta el reparto de sus horas en los diferentes grupos de las asignaturas.

La organización básica está compuesta por los docentes, los cuales tienen asignada una plaza, la cual depende de un tipo de contrato. Existen diferentes

tipos de contrato entre los que se encuentran doctor, ayudante doctor, catedrático, titular...

Las plazas de los docentes pertenecen a una de las distintas áreas de la universidad, que a su vez se encuentran integradas en los departamentos.

Es importante conocer que cuando se crea la planificación de un nuevo curso académico, se escogen las titulaciones, y dentro de estas, las asignaturas que se van a impartir. Según el número previsto de alumnos se hace un reparto de grupos por cada asignatura y se indica que plazas van a impartir esos grupos y cuantas horas o créditos van a dedicar, ya que puede haber grupos que sean impartidos por varios profesores.

Por último, también es importante saber que, como he dicho anteriormente, los grupos son asignados a las plazas para ser impartidos. Esto es debido a que lo normal es que una plaza pertenezca a un docente, pero puede darse el caso de que la plaza se encuentre libre y se esté en búsqueda de un docente que la cubra y se encargue de impartir los créditos indicados en los grupos que tenga asignados dicha plaza.

4. Técnicas y herramientas

A continuación se mostrarán las técnicas y herramientas utilizadas para el desarrollo del proyecto. Tanto para la parte de documentación, como para la parte de desarrollo de la aplicación.

4.1. Técnicas

Scrum

Scrum es una metodología para realizar el seguimiento de proyectos que comparte los principios del desarrollo ágil de partir de un concepto, seguir un desarrollo incremental (en *sprints*) y cerrar el producto. [2]

Elementos

- **Pila del producto o *product backlog*:** Es una lista de requisitos desde el punto de vista del cliente que comienza con una visión inicial del producto, pero que se irá incrementando a lo largo del desarrollo.
- **Pila del sprint o *sprint backlog*:** Conjunto de requisitos o tareas desde el punto de vista del desarrollador que el equipo de desarrollo del proyecto espera realizar durante el *sprint*.
- **Incremento:** El incremento es el resultado del desarrollo de un *sprint*, es una parte terminada y probada.
- **Gráfico de avance o *burn-down chart*:** Sirve para representar tanto el trabajo pendiente como el trabajo realizado durante un *sprint*.

para ver la velocidad a la que se están completando las tareas. De este modo, se puede saber si se va dentro de la planificación o no.

- **Gráfico de producto o *burn-up chart*:** Gráfico utilizado para ver el trabajo realizado respecto al total.
- **Gráfico de velocidad o *velocity chart*:** Muestra la velocidad de trabajo durante el *sprint*. Es útil para ir ajustando la previsión de tiempo en futuros *sprints*.

Roles

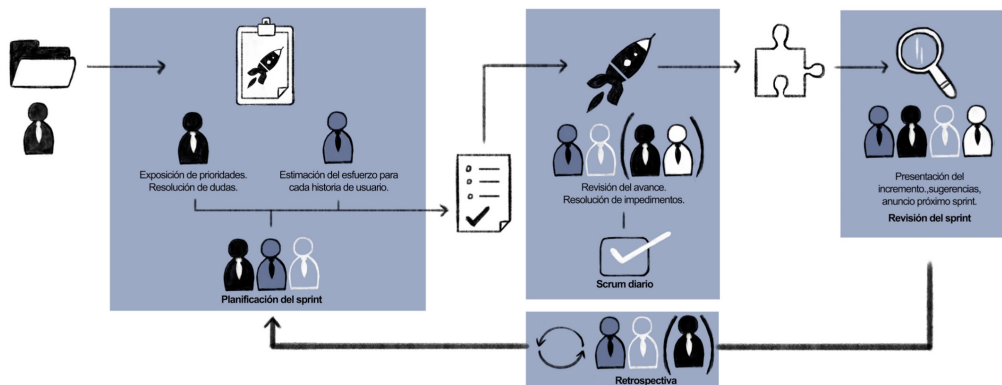
Las personas que participan en el proyecto tienen asignado un rol. En el siguiente listado se pueden ver los distintos roles junto a una breve descripción de los mismos.

- ***Scrum master*:** Persona con conocimientos avanzados en la metodología Scrum que se encarga de que se cumpla correctamente el procedimiento. También es el encargado de dar formación y asesorar al resto de personal sobre Scrum.
- **Propietario del producto o *product owner*:** Es la persona que hace de representación del cliente. Se encarga de la gestión de la pila del producto dando visión al producto con las historias de usuario y definiendo las prioridades de las mismas. Participa en cada reunión de planificación de los *sprints*.
- **Equipo de desarrollo:** Es recomendable que no sea un equipo con un gran número de personal (entre 4 y 8 personas). El equipo debe trabajar de forma simultánea para cumplir las tareas indicadas en la reunión de planificación del *sprint*.

En las figuras 4.1 y 4.2 se puede ver tanto un resumen del proceso seguido en Scrum como de los diferentes elementos que interactúan en el mismo.

De forma resumida, para seguir la metodología Scrum es necesario que el propietario del producto genere inicialmente las historias de usuario, que se encontrarán en la pila del producto. Estas historias podrán aumentar a lo largo del desarrollo.

Con la pila del producto creada, todo el personal se reúne para planificar el *sprint*. Estas reuniones son las que general la pila del *sprint*. Después de esta reunión, el equipo de desarrollo comienza a trabajar en las tareas que

Figura 4.1: *Proceso seguido en Scrum [2].*Figura 4.2: *Explicación de los elementos del proceso [2].*

se ha comprometido para completarlas en el tiempo acordado. Cada día de trabajo se realiza una pequeña reunión de aproximadamente 15 minutos donde se expone el trabajo realizado, el trabajo que se va a hacer en el día, las dudas, etc., esta reunión sirve para ir actualizando la pila del *sprint*.

Al final del *sprint*, todo el personal se vuelve a reunir para presentar el incremento realizado y ver posibles cambios. Además, se fija el anuncio del próximo *sprint* que comenzará de nuevo con una reunión de planificación.

Web

La web (*World Wide Web*) es un sistema que permite la transmisión y acceso a datos y documentos a través de internet utilizando el protocolo HTTP para la comunicación y el lenguaje HTML para la representación de las páginas web [5].

La tecnología web es una de las que más desarrollo ha sufrido en los últimos años. Esto ha provocado un cambio completo en nuestra forma de vida, afectando a aspectos como la búsqueda de información por Internet, realizar compras online, hacer trámites sin necesidad de acudir a una oficina, etc.,

Plataformas de aprendizaje (LMSs)¹

Un LMS es un sistema de gestión de aprendizaje *online* diseñado para administrar formación en línea. Este tipo de plataformas permiten que las instituciones educativas creen, gestionen y distribuyan distintas herramientas y contenidos a los estudiantes mediante internet.

Normalmente, un LMS proporciona un entorno virtual en el cual los estudiantes pueden acceder a diferentes materiales de estudio, como presentaciones, vídeos, documentos, tareas o incluso exámenes. Además, permite a los docentes llevar un seguimiento el progreso de los estudiantes, aportar retroalimentación de las tareas, administrar calificaciones y fomentar la participación mediante el uso de, por ejemplo, foros en la asignatura.

Un LMS estándar cuenta con las siguientes características:

1. **Gestión de cursos académicos:** Un usuario con permisos superiores, normalmente un docente, puede crear y administrar cursos académicos. Se entiende por esto establecer las asignaturas o contenidos del curso, la duración del mismo, los alumnos matriculados, etc.
2. **Distribución de contenido:** Los docentes tienen la posibilidad de subir diferentes contenidos a la plataforma como puede ser el temario de la asignatura impartida, vídeos de apoyo o cualquier otro tipo de material necesario o útil para el aprendizaje.
3. **Seguimiento y evaluación:** Los LMSs proporcionan herramientas para poder llevar un seguimiento del progreso de los estudiantes. Además, permiten evaluar las tareas pedidas a través de la plataforma y los exámenes realizados, tanto de forma presencial como *online*. Gracias a estas herramientas se puede dar una retroalimentación personalizada para cada estudiante en caso de ser necesario.
4. **Interacción en línea:** Gracias a estas plataformas los estudiantes pueden participar e interactuar entre ellos mediante, por ejemplo, foros de discusión dentro del espacio de cada asignatura.

¹LMS: Learning Management Systems

5. **Notificaciones y comunicación:** Estos sistemas cuentan con servicios de mensajería integrados que permiten a los docentes mandar un mensaje en forma de notificación a los estudiantes con información importante que deben ver. De esta forma se consigue dar una mayor importancia y visibilidad a los mensajes que los docentes o servicios administrativos consideren necesarios.
6. **Personalización de la plataforma:** Generalmente, los LMSs permiten un alto grado de personalización. Esto permite que diferentes entidades educativas utilicen el mismo sistema adaptándolo a sus necesidades.

4.2. Herramientas

GitHub

GitHub es una plataforma de alojamiento en la nube que permite llevar un control de versiones Git para los proyectos subidos. Además del control de versiones, GitHub posee distintas características como la creación de repositorios, el uso de *issues*, *Pull requests*, las páginas de *wiki*, añadir distintos usuarios al repositorio con diferentes permisos, etc. [8].

ZenHub

Extensión para navegador web que se integra con GitHub para permitir añadir los elementos de Scrum al repositorio. De esta manera se puede llevar un mejor seguimiento del trabajo realizado en el proyecto.

Python

Python es un popular lenguaje de programación que puede ser utilizado en diferentes ámbitos, entre ellos el desarrollo web. Algunas de las razones para utilizar este lenguaje es su facilidad de uso, la posibilidad de programar con orientación a objetos y la amplia variedad de bibliotecas y frameworks, como puede ser Flask [3].

Se estudió la posibilidad de uso de PHP para realizar la aplicación web, pero al final se decidió trabajar con Python debido a que es más conocido para los tutores del proyecto y a que, al utilizarlo durante el grado, es más fácil la adaptación para alumnos que en un futuro hagan nuevas versiones partiendo de este proyecto.

Flask

Como se ha comentado en el apartado anterior, Flask es un framework de Python que facilita la creación de aplicaciones web. Flask incluye el motor de plantillas Jinja2 que permite implementar código en las vistas de la aplicación de una forma cómoda y sencilla.

JavaScript

JavaScript es un lenguaje de programación interpretado débilmente tipado. Principalmente es conocido al ser utilizado en el lado del cliente para dar mejor interacción a páginas web estáticas, pero es mucho más potente de lo que se puede pensar y es utilizado en muchos más ámbitos. Incluso en la misma web, puede ser utilizado en el lado servidor mediante el entorno de ejecución Node.js. Esto permite que una aplicación web pueda estar completamente escrita en JavaScript [1].

HTML

HTML es el «Lenguaje de Marcas de Hipertexto» utilizado por la *World Wide Web* para representar las páginas web. Puede ser combinado con diferentes tecnologías como CSS para dar una apariencia diferente o JavaScript para poder interactuar de forma dinámica.

CSS

CSS, también conocido como «Hojas de Estilo en Cascada», es un lenguaje de estilos utilizado para mejorar la apariencia de los ficheros HTML. Gracias a CSS se puede dar rienda suelta al diseño y hacer páginas web completamente diferentes.

Bibliotecas / *plugins*

- **SQLAlchemy**

Biblioteca de SQL para Python que incluye una versión para el *framework* Flask que permite su completa integración. SQLAlchemy contiene un conjunto de herramientas SQL que permiten el mapeo objeto-relacional, *Object Relational Mapper (ORM)*. Esto permite que con un conjunto de expresiones requeridas por la biblioteca, se creen clases de objetos en Python que serán transformadas a tablas de una forma prácticamente abstracta para el desarrollador. Además,

al estar los objetos vinculados a registros de una tabla SQL, cuenta con métodos para realizar distintas operaciones SQL como búsquedas, inserciones, eliminaciones, etc., sin tener que escribir sentencias SQL [4].

■ Grid.js

Grid.js es un *plugin* de JavaScript que ayuda con el tratamiento de tablas. Gracias a esta biblioteca se pueden crear tablas con búsqueda, ordenamiento, redimensión de columnas, etc., de una forma sencilla gracias a sus herramientas. Además, Grid.js se puede incluir con *frameworks* como React, Angular o Vue, pero también puede ser utilizado con JavaScript *vanilla*.

Antes de la elección de este *plugin* se estuvo valorando el uso de otro llamado DataTables. Al final se decidió utilizar Grid.js a pesar de que DataTables es más completo.

Algunas de las razones son la siguientes:

- Para utilizar Grid.js no es necesaria la biblioteca jQuery mientras que para DataTables sí. La dependencia de otras bibliotecas puede dar problemas en un futuro, además de que el rendimiento de JavaScript *vanilla* es algo superior.
- Aunque esto se pueda cambiar, la estética de Grid.js me parece más moderna.
- Para este proyecto no eran necesarias muchas de las funcionalidades que aporta DataTables.
- En su repositorio en GitHub, se puede ver por los *commits* realizados que es un proyecto en continuo desarrollo.

■ Select2

Biblioteca de JavaScript que proporciona nuevas funcionalidades al campo *select* de HTML. Gracias a esta biblioteca es posible escribir en el campo y realizar búsquedas más cómodas sobre los datos cargados en el campo. Además, se pueden realizar consultas Ajax de los datos para no tener que realizar la carga completa de la información desde el inicio, sino que la información se va cargando según sea pedida por el usuario que maneja la aplicación.

■ Sortable

Se trata de una biblioteca de JavaScript que facilita la creación de listas con elementos arrastrables. También permite la ordenación en las listas creadas.

Esta biblioteca permite de una forma muy sencilla convertir listas estáticas de HTML en listas interactivas, lo que ofrece una gran personalización y posibilidades de manejo de la interfaz al poder gestionar los eventos de arrastrar o soltar elementos entre muchos otros.

Software

■ Pincel Project

Es un programa de ordenador gratuito y de código abierto que permite la creación de los prototipos de las vistas de una aplicación de una forma cómoda, arrastrando distintos elementos a la pantalla y modificándolos para que queden al gusto del creador. Cuenta con versiones para Windows, Linux y macOS. Por defecto, el software cuenta con objetos suficientes para crear los prototipos de las vistas de una aplicación. Sin embargo, es posible cargar más de estos objetos creados por la comunidad o por uno mismo, como pueden ser nuevos botones, campos de entrada, menús, etc.

■ Draw.io

Aplicación web que permite la creación de diagramas utilizando diferentes figuras geométricas. Además cuenta con objetos ya creados para crear diagramas utilizando UML, diagramas entidad-relación, diagramas relacionales, etc. Esta aplicación permite una personalización avanzada por lo que es muy útil para crear cualquier tipo de diagrama. Además, al ser una aplicación web no requiere de ningún tipo de instalación aunque sí que cuenta con una versión de escritorio para Windows, Linux, macOS y Chrome OS.

■ PyCharm

Es un entorno de desarrollo integrado (IDE) que pertenece a la empresa JetBrains y que está creado para trabajar en proyectos basados en el lenguaje de programación Python. Cuenta con una licencia de uso gratuita con funciones limitadas, aunque para estudiantes regalan un año de licencia completa para todos los productos de la empresa.

PyCharm es uno de los IDE más completos y avanzados para el desarrollo con Python, ya que cuenta con una gran cantidad de características

que facilitan el trabajo y ayudan a ahorrar tiempo. Además, a la hora de crear proyectos se puede indicar el tipo de *framework* a utilizar, lo que hará que cargue de forma automática todos los paquetes necesarios y creará una versión simple del proyecto desde la que empezar a trabajar. Esto ahorra mucho tiempo al no tener que ir buscando e instalando los paquetes necesarios desde distintos repositorios.

- **Postman**

Postman es una herramienta pensada para el desarrollo de API que permite crear y enviar solicitudes HTTP, realizar pruebas automatizadas y depurar errores. Además, la herramienta proporciona una interfaz fácil de usar. Gracias a esta herramienta se pueden realizar distintas peticiones a una web y ver si los resultados son los esperados. Esto permite ver de forma más sencilla la información devuelta por la aplicación, lo que aumenta la facilidad para detectar errores o ver la estructura de la información.

- **T_EXmaker**

Editor de documentos de código abierto diseñado para la creación y edición de documentos en formato L^AT_EX.

- **Moodle**

Moodle es una plataforma de aprendizaje en línea utilizada por la Universidad de Burgos para la gestión académica y la interacción entre docentes y alumnos. Permite realizar cuestionarios, exámenes, entregas y valoración de tareas, comunicar calificaciones y participar en foros, entre otras funciones.

En este proyecto, se ha utilizado Moodle para validar el inicio de sesión en la aplicación. Se ha hecho uso de un *endpoint*² al cual se envía el correo electrónico y la contraseña. Si el inicio de sesión es correcto, se devuelve un token de acceso; en caso contrario, se muestra un mensaje de error. De esta manera, se puede proporcionar acceso sin tener que almacenar contraseñas en la base de datos de la aplicación desarrollada.

Además, durante el inicio de sesión, se verifica primero si la dirección de correo electrónico se encuentra en la base de datos de la aplicación web desarrollada. Si se encuentra, se realiza la validación contra Moodle; de lo contrario, se muestra directamente un mensaje de error sin llegar a realizar la petición.

²URL que permite enviar una solicitud para obtener un recurso concreto

5. Aspectos relevantes del desarrollo del proyecto

Este apartado pretende mostrar aquellos aspectos más relevantes del proyecto realizado.

5.1. Diseño

Requisitos y casos de uso

El proyecto realizado podría ser un buen ejemplo de un proyecto real en el que un jefe de proyecto debe comunicarse con el cliente para obtener los requisitos necesarios de la aplicación y trasladar la información requerida a desarrollar a los programadores (en este caso ambos roles los cumplía la misma persona).

La idea descrita anteriormente es debida a que este proyecto se basa en una aplicación web que se pretende utilizar en la Universidad de Burgos. Por ello, los tutores del proyecto han actuado de una forma similar a clientes y durante las primera reuniones del proyecto daban la idea de como debía funcionar la aplicación a desarrollar.

El proceso de obtención de requisitos y diseño de casos de uso fue uno de los apartados de mayor duración del proyecto. Esto es debido a que las primeras reuniones fueron dedicadas íntegramente a explicar la idea se tenía de cómo debía ser la aplicación web y qué requisitos era necesario cumplir.

En un principio se comenzó por ir diseñando el diagrama de entidad-relación. De esta manera se veía de una forma más clara que entidades iba a tener la aplicación y cómo estas debían interaccionar entre ellas. Este

proceso duró varias semanas hasta dar con un diagrama que cumpliera de forma correcta con la idea de los tutores del proyecto y la idea que tenían de la aplicación web.

Con el diagrama de entidad-relación finalizado, se comenzaron a crear los diferentes casos de uso. Este apartado del proyecto también tuvo una duración de varias semanas. Durante este proceso se fueron creando todos los casos de uso de la aplicación, los cuales se pueden ver en los anexos. Además, con la creación de estos se vieron algunos detalles que no estaban todavía bien especificados y debían determinar los tutores del proyecto.

Al cabo de varias semanas de diseño, cuando se tuvo una base correcta de trabajo, se comenzó con el prototipado de las vistas y, con el visto bueno de estas, el desarrollo del *software*.

Cabe destacar que a pesar de haber hecho un largo proceso de diseño cuidando cada detalle, a lo largo del desarrollo hubo que retocar algún caso de uso o requisito para adaptarlo a nuevas necesidades surgidas que no se habían contemplado anteriormente o no se había profundizado lo suficiente en ellas.

Interfaz

Otro aspecto que se ha tenido muy en cuenta a la hora de desarrollar la aplicación es el diseño de la interfaz.

El diseño de las vistas ha intentado cuidarse para que fuese minimalista sin dejar que esto hiciese que se tuviese una *web* poco atractiva. La idea de hacer un diseño minimalista parte principalmente de la idea de hacer más sencillo el uso de la aplicación al evitar elementos que desvíen la atención del usuario. Además, el uso de un diseño minimalista ayuda a que la velocidad de carga de las páginas sea más rápida. Estos dos elementos hacen que la experiencia del usuario al utilizar la aplicación *web* sea mayor.

Otro aspecto que se ha cuidado al realizar la aplicación es que la web se adaptase lo máximo posible a cualquier tipo de resolución de pantalla, aspecto importante, ya que en esto es una mejora respecto a la aplicación SIGMA utilizada por la universidad.

Es necesario destacar que la aplicación *web* desarrollada es funcional a través de cualquier dispositivo móvil, pero no se recomienda su uso de esta manera ya que la aplicación cuenta con grandes tablas que a pesar de adaptarse a la pantalla, no son cómodas de manipular a través de este tipo de dispositivos.

5.2. Desarrollo

Inicio de sesión y las sesiones

El inicio de sesión de la aplicación funciona mediante un formulario de correo electrónico y contraseña. En un primer lugar se valida que el correo electrónico se encuentre en la base de datos propia y después se manda al Moodle de la Universidad de Burgos la petición mediante un *endpoint* de su API. Si los datos son correctos se devuelve un *token* que sirve para acceder a los recursos de Moodle. En el caso de esta aplicación, el *token* se guarda en la sesión para poder dar acceso al contenido de la misma. Si en la sesión está el *token* se permite el acceso, en caso contrario se redirige a la ventana de inicio de sesión.

La aplicación cuenta con un sistema de permisos que sólo permite acceder a aquellos usuarios con permisos de lectura y sólo permite modificar y ver cierta información a aquellos usuarios con permisos de modificación.

Para almacenar tanto el *token* como el identificador del usuario, y así poder comprobar sus permisos, se hace uso de las sesiones.

Por defecto, la sesión se almacena en el cliente mediante una *cookie*. El problema que tiene esta forma de trabajar es que la *cookie* podría ser robada y con ello, el ladrón podría tener acceso a la aplicación. En caso de guardar información sensible en la *cookie* lo mejor es hacer el almacenamiento en el lado del servidor, y eso es lo que se ha hecho en esta aplicación para mejorar la seguridad.

Creación de cursos académicos

La creación de cursos académicos en la aplicación es una de las partes que más se ha intentado cuidar debido a su importancia, ya que mientras algunos elementos, como por ejemplo los centros, lo normal es que se creen y no se vuelvan a tocar, la creación de cursos se realiza todos los años. Además, crear un curso académico supone una carga de trabajo importante por parte del usuario ya que debe seleccionar todas aquellas asignaturas deseadas para el curso, lo que puede ser un número elevado de elementos que se deben seleccionar de una forma cómoda.

Para resolver este problema en un principio se pensó en poner una tabla con todas las asignaturas existentes y un campo de selección, pero tras hacer una primera implementación de prueba se vio que no iba a ser algo cómodo para un uso real. Tras investigar diferentes opciones se pensó en utilizar la

biblioteca de JavaScript llamada «Sortable.js» que permite, entre muchas otras cosas, arrastrar elementos de la pantalla entre contenedores. De esta manera se pueden tener las asignaturas que se desean seleccionar en un contenedor y las asignaturas seleccionadas en otro.

Finalmente, se decidió añadir un filtro por titulación y curso para que el usuario tenga un mayor control de las asignaturas a buscar y utilizar la biblioteca Sortable.js para permitir arrastrar las asignaturas requeridas de forma simultanea. Al arrastrar, estas son seleccionadas para añadir al curso.

Esta forma de trabajar hace que la aplicación se maneje de una forma más cómoda y dinámica que si se hubiesen usado otras opciones. Además, se añadió la opción de duplicar un curso académico completo para que de un año a otro no haya que crear todas las vinculaciones si no que se pueda partir de las asignaciones realizadas en cursos previos.

6. Trabajos relacionados

A continuación se presentarán diferentes trabajos que mantienen relación con la aplicación desarrollada en este trabajo de fin de grado. Estos trabajos han abordado diferentes aspectos relacionados con la gestión del personal, principalmente desde el punto de vista la planificación y asignación de recursos.

- **SIGMA**³

SIGMA es un *software* diseñado para la gestión académica y es utilizado por un gran número de universidades españolas, entre ellas la Universidad de Burgos. Está diseñado para ayudar a las instituciones educativas a gestionar tanto sus procesos administrativos como académicos. Esto incluye, entre otras, la gestión de estudiantes, docentes, matrículas, cursos académicos y calificaciones.

Esta plataforma es muy completa ya que permite gestionar prácticamente todo el ciclo de vida de la gestión académica. Sin embargo, cada institución tiene sus propias particularidades y ahí es donde la aplicación de este trabajo destaca por completo al estar hecha a medida para lo que se requiere.

A día de hoy SIGMA cuenta con algunas desventajas que hacen que su uso no sea del todo cómodo. Algunas de estas son que cuenta con un diseño pensado para pantallas grandes que no se adapta bien a otro tipo de resoluciones, es decir, un diseño no *responsive*, se encuentra migrada a una aplicación web de una mala manera ya que su funcionamiento es lanzando procesos en una web, no visualiza fácilmente la información

³<https://www.sigmaaie.org/es>

en pantalla sino que se basa en listados de generación asíncrona y cuenta con problemas de integración.

- **Classter**⁴

Es una plataforma de gestión académica que permite controlar el proceso educativo al completo. Incluye un sistema para la gestión de estudiantes, profesorado, aulas, calificaciones, etc.

Los puntos fuertes de esta plataforma podrían ser su modularidad, que permite añadir sólo aquellas partes que se deseen, la personalización, que permite adaptar la aplicación a cada institución y la facilidad de comunicación que proporciona entre docentes y alumnos.

Entre sus principales desventajas se encuentran una gran curva de aprendizaje, es decir, que cuesta conocer y trabajar con normalidad con el sistema desde un principio y requiere de un proceso de aprendizaje previo para aprovecharlo al completo, la falta o mala integración con otras herramientas para la gestión académica y por último, el costo.

- **Constructor**⁵

Constructor es una plataforma pensada para instituciones de educación superior. Ofrece diferentes soluciones entre las que se encuentran la gestión del PDI, la administración de cursos, una plataforma de enseñanza *online* y la gestión de servicios estudiantiles como pueden ser matrículas, biblioteca, gestión de becas, etc.

⁴<https://www.classter.com/>

⁵<https://constructor.tech/solutions/higher-education>

7. Conclusiones y Líneas de trabajo futuras

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.

Bibliografía

- [1] Amazon. ¿qué es javascript? <https://aws.amazon.com/es/what-is/javascript/>, 2023. <https://aws.amazon.com/es/what-is/javascript/>.
- [2] Claudia Ruata Juan Palacio. *Scrum Manager. Gestión de Proyectos*. 2011.
- [3] Phyton. Documentación de python. <https://docs.python.org/3/tutorial/index.html>, 2023. <https://docs.python.org/3/tutorial/index.html>.
- [4] SQLAlchemy. Documentación de sqlalchemy. <https://docs.sqlalchemy.org/en/20/intro.html>, 2023. <https://docs.sqlalchemy.org/en/20/intro.html>.
- [5] Wikipedia. World wide web — wikipedia, la enciclopedia libre. https://es.wikipedia.org/w/index.php?title=World_Wide_Web&oldid=149753290, 2023. [Internet; descargado 3-abril-2023].
- [6] Wikipedia contributors. Cross-site request forgery — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Cross-site_request_forgery&oldid=1154181612, 2023. [Online; accessed 22-May-2023].
- [7] Wikipedia contributors. Cross-site scripting — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Cross-site_scripting&oldid=1151071276, 2023. [Online; accessed 22-May-2023].

- [8] Wikipedia contributors. Github — Wikipedia, the free encyclopedia. <https://en.wikipedia.org/w/index.php?title=GitHub&oldid=1147528749>, 2023. [Online; accessed 2-April-2023; <https://en.wikipedia.org/w/index.php?title=GitHub&oldid=1147528749>].