

**Desarrollo de recursos educativos para Ingeniería del Software
utilizando repositorios de Open Data y Linked Data.**



**UNIVERSIDAD
COMPLUTENSE
MADRID**

Trabajo de fin de grado del Grado en Ingeniería Informática, Facultad de
Informática, Universidad Complutense de Madrid

Manuel Martín Canora y David Limón Miralles

Director

Antonio Sarasa Cabezuelo

Junio 2019

Resumen

En este trabajo se desarrolla una herramienta capaz de gestionar recursos educativos de la asignatura Ingeniería del Software. Existen tres roles diferentes, alumno, profesor y administrador donde cada uno tiene su propia interfaz.

El administrador es el encargado de gestionar la herramienta y las cuentas de los profesores.

El profesor es quien genera recursos en la herramienta, se le ofrece la posibilidad de completar dichos recursos con repositorios obtenidos a través de Open Data y Linked Data. Además, puede gestionar los alumnos pertenecientes a su grupo.

El alumno tiene la capacidad de interactuar con estos recursos, visualizando y descargando el contenido de estos.

Palabras clave: Ingeniería del Software, recursos, repositorio, Open Data y Linked Data.

Abstract

In this work, a tool capable of managing educational resources of the Software Engineering subject is developed. There are three different roles, student, teacher and administrator where each one has its own interface.

The administrator is in charge of managing the tool and the teachers' accounts.

The professor is the one who generates resources in the tool, he / she is offered the possibility of completing those resources with repositories obtained through Open Data and Linked Data. In addition, you can manage the students belonging to your group.

The student has the ability to interact with these resources, visualizing and downloading the content of these.

Keywords: Software engineering, Resources, Repository, Open data and Linked data.

Índice general.

Resumen.....	3
Abstract.....	4
Índice general.....	5
Índice de figuras.....	6
1. Introducción.....	9
1.1 Motivación.....	9
1.2 Objetivos.....	9
1.3 Estructura de la memoria.....	9
1. Inroduction.....	11
1.1 Motivation.....	11
1.2 Objectives.....	11
1.3 Structure of memory.....	11
2. Estado del arte.....	13
2.1 Moodle.....	13
2.2 Google Classroom.....	13
2.3 BlackBoard.....	13
2.4 Edmodo.....	14
3. Especificación de Requisitos.....	15
3.1 Diagramas de casos de uso.....	15
3.2 Casos de uso referidos al profesor.....	16
3.3 Casos de uso referidos al estudiante.....	18
3.4 Casos de uso referidos al administrador.....	19
3.5 Casos de uso comunes.....	20
4. Tecnologías aplicadas.....	24
4.1. Tecnologías de la parte cliente.....	24
4.2 Tecnologías de la parte del servidor.....	25
4.3 Tecnologías semánticas.....	25
4.4 Bases de datos.....	26
4.5 Otras tecnologías.....	26
5. Modelado de datos.....	29
5.1 Modelo Entidad-Relación.....	29
5.2 Base de datos.....	29
6. Arquitectura de la aplicación.....	33
7. Diseño e implementación.....	34
7.1 Principios de usabilidad.....	34
7.2 Funcionalidad.....	34
8. Evaluación.....	50
8.1 Guion de pruebas.....	50
8.2 Resultados de la evaluación.....	53
9. Conclusiones y trabajos futuros.....	62
9.1 Conclusiones del trabajo.....	62
9.2 Trabajos de futuro.....	62

9.3 Aportaciones de David Limón Miralles.....	63
9.4 Aportaciones de Manuel Martín Canora.....	64
9. Conclusions and future work.....	65
9.1 Conclusions of the work.....	65
9.2 Future Jobs.....	65
9.3 Contributions of David Limón Miralles.....	66
9.4 Contributions of Manuel Martín Canora.....	67
Bibliografía.....	69
Anexo I: Guía de instalación de la aplicación.....	71
Anexo II: Guía de uso de la aplicación.....	78

Índice de figuras

Figura 1: Caso de uso del estudiante.....	15
Figura 2: Caso de uso del profesor.....	16
Figura 3: Caso de uso del administrador.....	16
Figura 4: Diagrama entidad – relación.....	29
Figura 5: Diagrama de la base de datos.....	30
Figura 6: Funcionamiento de PHP, MySQL y apache con cliente.....	33
Figura 7: Diagrama de clases.....	35
Figura 8: Inicio de sesión.....	36
Figura 9: Confirmar inicio de sesión en código.....	36
Figura 10: Registro de usuario.....	37
Figura 11: Página principal para el tipo de usuario alumno	38
Figura 12: Mostrar los recursos de la página principal.....	38
Figura 13: Cabecera de la página principal para el tipo de usuario profesor.....	39
Figura 14: Consulta a la tabla “Perfil” de la base de datos.....	39
Figura 15: Página principal para el tipo de usuario administrador.....	40
Figura 16: Buscador de la página principal.....	40
Figura 17: Página del perfil para un alumno.....	41
Figura 18: Página del perfil para un profesor.....	42
Figura 19: Editar perfil.....	42
Figura 20: Gestionar cuentas para un profesor.....	43
Figura 21: Activar cuentas.....	44
Figura 22: Subir archivos por un profesor.....	44
Figura 23: Subir archivos.....	45
Figura 24: Crear recurso por un profesor.....	46
Figura 25: Crear recurso.....	47
Figura 26: Visualizar archivos.....	47
Figura 27: Recorrer directorio.....	48
Figura 28: Previsualización de un recurso.....	48
Figura 29: Consulta SPARQL	49
Figura 30: Generar consulta.....	49
Figura 31: Encuesta.....	50
Figura 32: Resultado de la encuesta, gráfico de sexo.....	54
Figura 33: Resultado de la encuesta, gráfico de crear cuenta.....	54
Figura 34: Resultado de la encuesta, gráfico de activar cuenta.....	55

Figura 35: Resultado de la encuesta, gráfico de cierre de sesión.....	55
Figura 36: Resultado de la encuesta, gráfico de estilo recurso.....	56
Figura 37: Resultado de la encuesta, gráfico de perfil.....	56
Figura 38: Resultado de la encuesta, gráfico de recursos.....	57
Figura 39: Resultado de la encuesta, gráfico de descarga.....	57
Figura 40: Resultado de la encuesta, gráfico web.....	58
Figura 41: Resultado de la encuesta, gráfico de crear recurso.....	58
Figura 42: Resultado de la encuesta, gráfico de guardar recurso.....	59
Figura 43: Resultado de la encuesta, gráfico de modificar recurso guardado.....	60
Figura 44: Resultado de la encuesta, gráfico borrar alumno desde perfil.....	61
Figura 45: Resultado de la encuesta, gráfico eliminar recursos.....	61
Figura 46: Descarga de Xampp.....	71
Figura 47: Inicio de instalación de Xampp.....	72
Figura 48: Instalación de Xampp.....	72
Figura 49: Ejecutable de Xampp.....	73
Figura 50: Panel de control de Xampp	74
Figura 51: Panel de control de Xampp (Activado).....	74
Figura 52: Buscador con localhost.....	75
Figura 53: Página de bienvenida de Xampp	75
Figura 54: Inicio de base de datos PHPMyAdmin	76
Figura 55: Creación de una base de datos	76
Figura 56: Importar una base de datos.....	77
Figura 57: Resultado de la importación	77
Figura 58: Inicio de sesión desde profesor.....	78
Figura 59: Registro de cuenta desde profesor.....	78
Figura 60: Activar Alumno desde profesor.....	79
Figura 61: Perfil desde profesor.....	80
Figura 62: Modificar perfil desde profesor.....	80
Figura 63: Mis recursos desde profesor.....	80
Figura 64: Mis alumnos desde profesor.....	81
Figura 65: Gestionar alumno desde perfil desde profesor.....	81
Figura 66: Visualizar Recursos, desde profesor.....	82
Figura 67: Recurso desde profesor.....	82
Figura 68: Descarga de archivo desde profesor.....	82
Figura 69: Descarga de archivo PDF desde profesor.....	83
Figura 70: Menú desplegable desde profesor.....	83
Figura 71: Selección de archivo desde profesor.....	84
Figura 72: Borrar archivo desde profesor.....	84
Figura 73: No se puede borrar desde profesor.....	84
Figura 74: Mensaje de confirmación desde profesor.....	84
Figura 75: Título y descripción desde profesor.....	85
Figura 76: Elección de archivos desde profesor.....	85
Figura 77: Selección de links desde profesor.....	86
Figura 78: Selección de opción desde profesor.....	86
Figura 79: Cerrar sesión desde profesor.....	86
Figura 80: Registro de cuenta del alumno.....	87
Figura 81: Inicio de sesión del alumno.....	87
Figura 82: Modificar perfil del alumno.....	88

Figura 83: Página principal del alumno.....	88
Figura 84: Visualizar grupos desde alumno.....	89
Figura 85: Descarga del archivo desde alumno.....	89
Figura 86: Descarga del archivo PDF desde alumno.....	90
Figura 87: Cerrar sesión desde alumno.....	90
Figura 88: Inicio de sesión del administrador.....	91
Figura 89: Modificar perfil del administrador.....	91
Figura 90: Activar cuentas de profesores.....	92
Figura 91: Visualizar Recursos de los profesores desde administrador.....	92
Figura 92: Recurso visto desde administrador.....	93
Figura 93: Descarga de archivo desde administrador.....	93
Figura 94: Descarga de archivo PDF desde administrador.....	94
Figura 95: Cerrar sesión desde administrador.....	94

1. Introducción.

En este capítulo se explica la necesidad de una herramienta como esta, los objetivos planteados y una guía estructurada de la memoria.

1.1. Motivación.

Para los alumnos de la asignatura de Ingeniería del Software buscar contenido relacionado con dicha asignatura acaba generando problemas, esta información se encuentra dispersa por toda la red y puede proceder de fuentes poco fiables.

Por ello surge la necesidad de unificar todo el contenido existente en una única herramienta, una herramienta capaz de facilitar el acceso a todo el contenido disperso, obteniendo dicha información con consultas sobre los repositorios de Open y Linked Data.

Crear un repositorio con un contenido amplio, estructurado y de fácil acceso para los alumnos de la asignatura, con recursos generados por los propios profesores es la solución.

1.2. Objetivos.

El objetivo principal es desarrollar una herramienta de aspecto educativo para la asignatura Ingeniería del Software, con repositorios de Open Data y Linked Data.

A continuación, se listan otros objetivos.

- Diseñar una interfaz para cada usuario existente.
- Proporcionar contenido generado por consultas SPARQL para la creación de recursos.
- Gestionar alumnos y profesores según el grupo al que pertenezcan.
- Unificar la información relacionada con Ingeniería del Software en esta herramienta.

1.3. Estructura de la memoria.

La memoria de este trabajo sigue la siguiente estructura. En el capítulo 1 se realiza una breve introducción al trabajo realizado, se expone la motivación, los objetivos planteados y se describe la estructura de la memoria.

A continuación, en el capítulo 2 se describe el estado del arte, analizando herramientas similares al proyecto; en el capítulo 3 se presenta la especificación de requisitos; en el capítulo 4 una explicación de las tecnologías utilizadas; en el capítulo 5 se expondrá la funcionalidad y significado de cada tabla de la base de datos; en el capítulo 6 una breve

explicación de la aplicación cliente/servidor, que es y cómo se adapta; en el capítulo 7 una explicación de la funcionalidad implementada.

Después en el capítulo 8 se mostrarán los resultados de la evaluación realizada sobre esta aplicación. En el último, el capítulo 9, se explicarán las aportaciones realizadas por cada miembro del grupo, conclusiones y una pequeña explicación de cómo se podría continuar.

En los anexos del trabajo se dispone de una guía de uso de la aplicación y una guía de instalación de la herramienta.

1. Introduction

This chapter explains the need for a tool like this, the objectives set and a structured memory guide.

1.1 Motivation.

For the students of the Software Engineering course, looking for content related to this subject ends up generating problems, this information is scattered throughout the network and may come from unreliable sources.

Therefore, there is a need to unify all the existing content in a single tool, a tool capable of facilitating access to all the dispersed content, obtaining this information with queries about the Open and Linked Data repositories.

Create a repository with a broad, structured and easily accessible content for students of the subject, with resources generated by the teachers themselves is the solution.

1.2 Objectives.

The main objective is to develop an educational aspect tool for the Software Engineering subject, with Open Data and Linked Data repositories.

Next, other objectives to be fulfilled are listed.

- Design an interface for each existing user.
- Provide content generated by SPARQL queries for the creation of resources.
- Manage students and teachers according to the group they belong to.
- Unify the information related to Software Engineering in this tool.

1.3 Structure of memory.

The memory of this work follows the following structure. In chapter 1 a brief introduction to the work done is made, the motivation is exposed, the objectives set out and the structure of the memory is described.

Then, in chapter 2, the state of the art is described, analyzing tools similar to the project; Chapter 3 presents the specification of requirements; in chapter 4 an explanation of the technologies used; Chapter 5 will explain the functionality and meaning of each table in the database; in chapter 6 a brief explanation of the client / server application, which is and how it is adapted; in chapter 7 an explanation with captures of the implemented functionality.

Then in chapter 8 the results of the evaluation made about our application will be shown. In the last one, chapter 9, the contributions made by each member of the group, conclusions and a small explanation of how it could be continued will be explained.

In the annexes of the work there is a guide to use the application and a guide to install the tool.

2. Estado del arte.

En este capítulo se realiza un breve estudio sobre herramientas similares a este proyecto. Analizando funcionalidad, finalidad y características principales de cada una de ellas.

2.1 Moodle.

Es una herramienta de gestión de aprendizaje de alcance mundial. Se establece como fuente de ayuda a los educadores creando una comunidad basada en el aprendizaje en línea.

Aquí <https://moodle.org/?lang=es> [5], podemos encontrar el enlace a su web. Una web que permite a los profesores la gestión de cursos virtuales para sus alumnos, o el uso de un espacio en línea que dé apoyo a la presencialidad. Respaldada la interacción grupal permitiendo la conversación privada entre los estudiantes, generando un medio para realizar evaluaciones del curso.

Es un sistema gratuito, seguro, personalizable y capaz de adaptarse a las necesidades que aparecen en el transcurso del tiempo. Distribuido bajo la licencia GNU [38] y utilizando lenguaje web destaca por su interoperabilidad, porque permite ejecutarse en diferentes entornos.

2.2 Google Classroom.

Es una plataforma educativa gratuita, que forma parte de las aplicaciones de Google para la educación, <https://classroom.google.com/u/0/h?hl=es> [6] incluye otras aplicaciones de Google como Google Docs, Gmail , y Google Calendar.

Sus funciones son simplificar y distribuir tareas, así como evaluar el contenido. Permite la creación de aulas virtuales dentro de una misma institución educativa, sirviendo como nexo entre profesores, padres y alumnos.

Esta herramienta permite que los alumnos sean capaces de trabajar en un mismo documento desde diferentes computadoras en tiempo real fomentando así el trabajo colaborativo. También optimiza el tiempo permitiendo a los profesores proporcionar un código para que se unan a su aula, generando una comunicación entre ambos.

2.3 Blackboard.

Es una compañía estadounidense de tecnología educativa, cuyo enlace es <https://ush.blackboard.com/>.

La compañía proporciona *software* de educación, para móviles, de comunicación y de comercio, así como también servicios a clientes como instituciones de educación, empresas y entidades gubernamentales.

Blackboard destaca por su gestión de *hosting* [33] y de programas, proporcionando asesoría, formación y servicios a cursos estudiantiles en línea.

2.4 Edmodo.

Es una plataforma tecnológica, social, educativa y gratuita que permite la comunicación entre alumnos y profesores en un entorno cerrado y privado. Este es su enlace <https://www.edmodo.com/?language=es> [10].

Utiliza la educación para organizar contenidos, hacer asignaciones y mantener una comunicación e interacción activa y constante entre alumnos, profesores y padres. Consiguiendo una comunicación sincrónica y asincrónica, proporcionando una flexibilidad de horarios a un aprendizaje colaborativo, donde el conocimiento es dinámico y compartido.

3. Especificación de requisitos.

En este capítulo se describe la especificación de las funciones implementadas en el proyecto y los diferentes casos de uso para cada uno de los roles existentes en la herramienta: alumno, profesor y administrador.

3.1. Diagrama de casos de uso.

El usuario final de la herramienta es el alumno, permitiéndole interactuar con ella. Teniendo la posibilidad de registrarse como alumno de un determinado profesor escogido por él mismo. Una vez registrado puede iniciar sesión rellenando los campos que se le facilitan.

Con la sesión iniciada tiene acceso a todos los recursos existentes en el sistema independientemente del grupo al que pertenece, además, se facilitan opciones de personalización del perfil y un buscador de recursos. Véase la relación entre el actor principal, el alumno y sus casos de uso en la Figura 1.

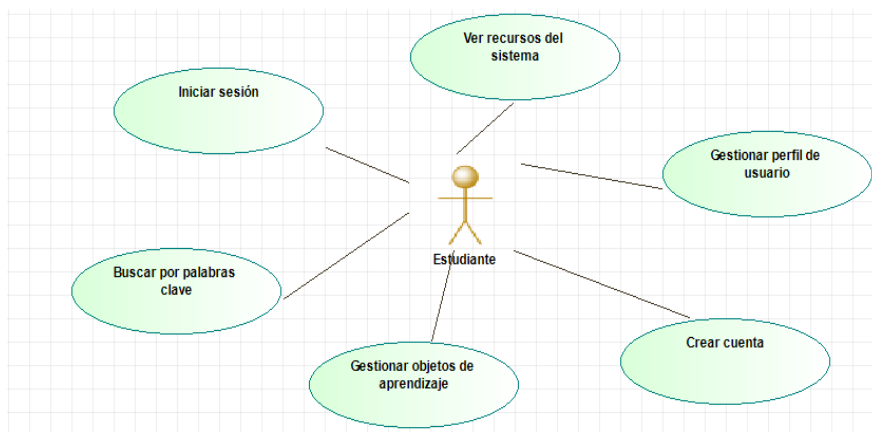


Figura 1: Caso de uso del estudiante.

El profesor, además de poseer la mayoría de las capacidades del alumno se caracteriza por tener la autoridad de gestionar el contenido de la herramienta. Destacando casos de uso como crear y gestionar recursos.

Es el encargado de gestionar las cuentas de los alumnos desempeñando el papel de administrador. En la Figura 2, aparece la relación entre el actor principal, el profesor y sus casos de uso.

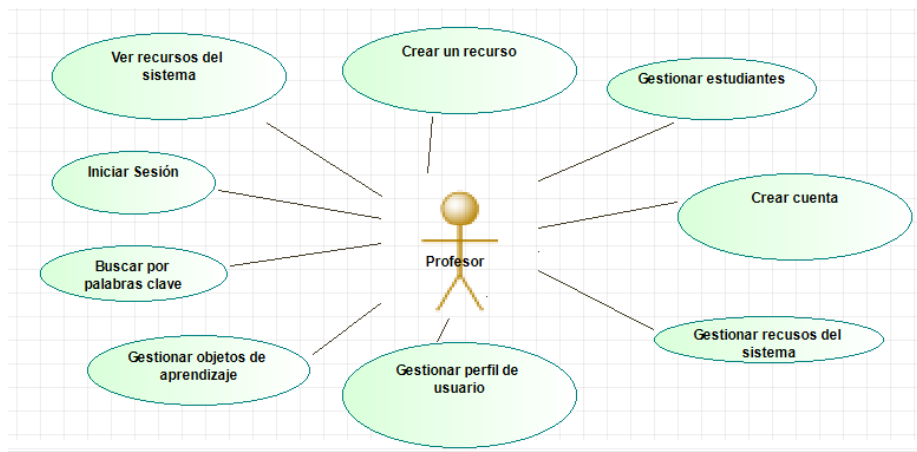


Figura 2: Caso de uso del profesor.

En la Figura 3, se relaciona el administrador y sus casos de uso. Denominado así, porque sus funciones en esta herramienta se caracterizan por poseer la capacidad de gestionar tanto recursos como a los creadores de estos, los profesores.

Cuenta también con el poder de interactuar con la herramienta, donde se incluye el inicio de sesión, el uso de buscadores y la visualización de contenido.

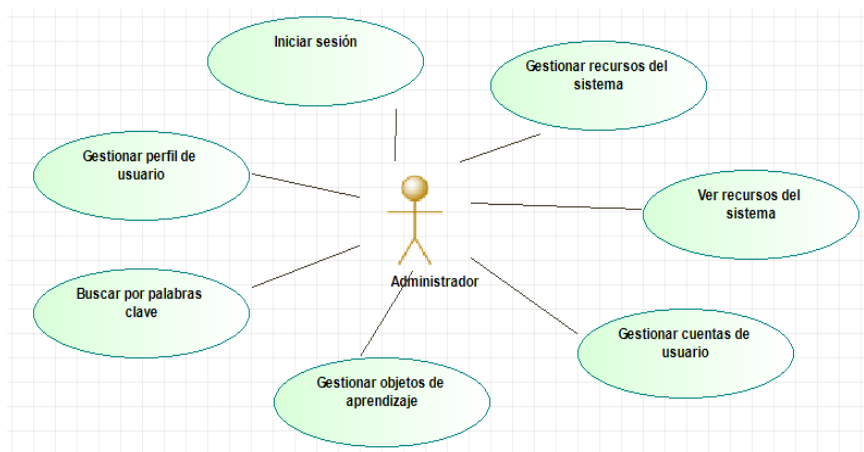


Figura 3: Caso de uso del administrador.

3.2. Casos de uso referidos al profesor.

Se describen las diferentes actividades que debe realizar el profesor para llevar a cabo los siguientes procesos: crear cuenta, gestionar estudiantes y crear recursos.

Caso de uso	P-1: Crear cuenta.
Actores	Profesor.
Objetivo	Registro de un nuevo profesor.
Entrada	Datos personales del profesor: nombre completo, correo electrónico, contraseña y grupo.
Salida	Una solicitud de registro queda almacenada en la base de datos pendiente de activación por el administrador.
Secuencia normal	<ol style="list-style-type: none"> 1. El profesor pulsa en el botón “Registrarse”. 2. El profesor rellena los campos obligatorios necesarios para el registro. 3. El profesor envía la solicitud. 4. Se muestra un mensaje indicando que la solicitud ha quedado pendiente de activación por el administrador.
Secuencia alternativa	<p>A) El profesor ya se encuentra registrado en la base de datos y se le notifica que no puede crear una nueva cuenta con los datos introducidos.</p> <p>B) El formato de algún campo obligatorio no es correcto y se le notifica que no puede finalizar el proceso hasta que corrija los errores.</p> <p>C) El profesor decide cancelar el proceso pulsando sobre “Volver a la página de inicio”.</p>

Caso de uso	P-2: Gestiona estudiantes.
Actores	Profesor.
Objetivo	Eliminar o activar las cuentas de estudiante asociados a un profesor en concreto.
Entrada	Cuentas de los estudiantes que van a ser eliminados o activadas.
Salida	Se reflejan los cambios realizados en la base de datos.
Secuencia normal	<ol style="list-style-type: none"> 1. El profesor selecciona la opción de “Gestionar alumnos”. 2. Se muestra un listado de cuentas de estudiantes. Cada cuenta tiene asociado dos opciones: rechazar o activar.

	3. El profesor realiza sobre las cuentas la operación que desea realizar.
Secuencia alternativa	El profesor decide cancelar el proceso pulsando sobre cualquier otra opción disponible.

Caso de uso	P-3: Crear un recurso.
Actores	Profesor.
Objetivo	Crear un recurso que contiene diferentes objetos de aprendizaje relacionados con una categoría en concreto.
Entrada	Recurso que se desea crear.
Salida	Se modifica la base de datos con el nuevo recurso creado.
Secuencia normal	<ol style="list-style-type: none"> 1. El profesor pulsa sobre su nombre de usuario. 2. El profesor selecciona la opción “Crear recursos”. 3. Se abre una nueva ventana donde puede añadir los datos necesarios para crear un recurso. 4. Escoge los objetos de aprendizaje que desea añadir al recurso. 5. Escoge los enlaces relacionados el recurso. 6. Rellena los metadatos asociados al recurso. 7. Pulsar sobre “Crear recurso”.
Secuencia alternativa	<p>El profesor decide previsualizar el recurso pulsando sobre “Previsualizar recurso”.</p> <p>El profesor decide guardar el recurso pulsando sobre “Guardar recurso”.</p>

3.3. Casos de uso referidos al estudiante.

Se describen las diferentes actividades que debe realizar el estudiante para llevar a cabo el proceso de crear cuenta.

Caso de uso	P-1: Crear cuenta.
Actores	Estudiante.
Objetivo	Registro de un nuevo estudiante asociado a un profesor existente.

Entrada	Datos personales del usuario: nombre completo, contraseña, correo electrónico. Profesor al que desea estar asociado.
Salida	Una solicitud de registro queda almacenada en la base de datos pendiente de activación por el profesor.
Secuencia normal	<ol style="list-style-type: none"> 1. El estudiante pulsa en el botón "Registrarse". 2. El estudiante rellena los campos obligatorios necesarios para el registro. 3. El estudiante selecciona de la lista al profesor al que desea asociarse. 4. El estudiante envía la solicitud. 5. Se muestra un mensaje indicando que la solicitud ha quedado pendiente de activación por el profesor.
Secuencia alternativa	<p>A) El estudiante ya se encuentra registrado en la base de datos y se le notifica que no puede crear una nueva cuenta con los datos indicados.</p> <p>B) El formato de algún campo obligatorio no es correcto y se le notifica que no puede finalizar el proceso hasta que corrija los errores.</p> <p>C) El estudiante decide cancelar el proceso pulsando sobre "Volver a la página de inicio".</p>

3.4 Casos de uso referidos al administrador.

Se describen las diferentes actividades que debe realizar el estudiante para llevar a cabo el proceso de gestionar cuentas de usuario.

Caso de uso	P-1: Gestionar cuentas de usuario.
Actores	Administrador.
Objetivo	Activar cuentas de profesores o rechazar cuentas de profesores.
Entrada	Cuentas de los profesores pendientes de activación.
Salida	Se reflejan los cambios en la base de datos.
Secuencia normal	<ol style="list-style-type: none"> 1. Selecciona la opción "Gestionar cuentas". 2. En la nueva ventana aparece un listado paginado de las cuentas de profesores gestionadas por el sistema

	3. El administrador selecciona las cuentas de los profesores que desea activar. Así mismo selecciona las cuentas de profesores que desea eliminar.
Secuencia alternativa	El administrador decide cancelar el proceso pulsando sobre cualquier otra opción disponible.

3.5 Casos de uso comunes.

Se describen las diferentes actividades que deben realizar los diferentes roles para llevar a cabo sus procesos.

Caso de uso	P-1: Iniciar sesión.
Actores	Profesor, estudiante o administrador.
Objetivo	Iniciar sesión de un usuario.
Entrada	Usuario y contraseña.
Salida	Tiene acceso a la aplicación.
Secuencia normal	<ol style="list-style-type: none"> 1. El usuario pulsa en el botón “Iniciar sesión”. 2. El usuario rellena los campos obligatorios necesarios para iniciar sesión. 3. Se comprueba si los datos rellenos coinciden con la base de datos para verificar si son correctos. 4. Se muestra la página principal para cada usuario.
Secuencia alternativa	<p>A) El contenido de algún campo no es correcto, se le notifica que no puede finalizar el proceso hasta que corrija los errores pudiendo volver a introducir los datos nuevamente.</p> <p>B) El usuario cancela la operación.</p>

Caso de uso	P-2: Gestionar perfil de usuario.
Actores	Profesor, estudiante o administrador.
Objetivo	Gestionar el perfil de un usuario.
Entrada	El usuario debe haber iniciado sesión y selecciona la opción “Perfil”.

Salida	Modificación del perfil o eliminación de la cuenta de usuario. En todos los casos, los cambios realizados se reflejan en la base de datos.
Secuencia normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción “Perfil”. 2. Se visualiza el perfil del usuario registrado. 3. El usuario puede modificar la información existente en su perfil, eliminar la cuenta de usuario o bien salir sin realizar ninguna modificación. 4. Para finalizar pulsa en el botón “Aceptar”.
Secuencia alternativa	<p>A) El usuario ha modificado incorrectamente algún campo de su perfil, se le notifica con un mensaje de error y se le permite volver a modificar los campos.</p> <p>B) El usuario cancela la operación y vuelve a la pantalla principal de su cuenta.</p>

Caso de uso	P-3: Buscar por palabras clave.
Actores	Profesor, estudiante o administrador.
Objetivo	Usar el buscador para acceder directamente al contenido deseado.
Entrada	Introducir en el buscador una serie de palabras clave que identifican la información a recuperar.
Salida	Todo el contenido dentro del sistema que tenga asociadas alguna de las palabras clave introducidas en el buscador.
Secuencia normal	<ol style="list-style-type: none"> 1. El usuario escribe un conjunto de palabras clave que identifican el contenido a buscar. 2. Pulsar sobre el botón “Buscar”. 3. Se visualización una lista de los contenidos recuperados.
Secuencia alternativa	Se muestra un mensaje indicando que no se ha encontrado nada para la búsqueda realizada.

Caso de uso	P-4: Ver recursos del sistema.
Actores	Profesor, estudiante o administrador.
Objetivo	Visualizar los objetos de aprendizaje existentes en un recurso del sistema.

Entrada	El usuario ha iniciado sesión y pulsa sobre el recurso deseado.
Salida	El usuario accede a todos los objetos de aprendizaje del recurso.
Secuencia normal	<ol style="list-style-type: none"> 1. El usuario selecciona el recurso que desea visualizar. 2. El usuario accede a todos los objetos de aprendizaje del recurso. 3. El usuario visualiza el contenido del objeto de aprendizaje que desee visualizar. 4. El usuario puede interactuar con el objeto de aprendizaje, visualizando todo su contenido.
Secuencia alternativa	El usuario pulsa sobre pulsa cualquier otra opción disponible.

Caso de uso	P-5: Gestionar recursos del sistema.
Actores	Profesor y administrador.
Objetivo	Añadir o eliminar recursos al repositorio que tiene asociado cada profesor dentro del sistema.
Entrada	Recursos que se desean añadir o eliminar en dicho repositorio.
Salida	Se reflejan los cambios en la base de datos.
Secuencia normal	<ol style="list-style-type: none"> 1. El profesor pulsa sobre el recurso que desea gestionar. 2. Se abre una nueva ventana donde aparece el recurso seleccionado anteriormente, con dos botones disponibles “Editar” o “Eliminar”. 3. En el caso de editar un recurso, deberá completar o modificar una serie de campos: título y descripción de dicho recurso, archivos o links. 4. Pulsar en “Crear recurso”.
Secuencia alternativa	<p>A) El profesor decide cancelar el proceso pulsando cualquier otra opción disponible, el proceso no será guardado.</p> <p>B) El profesor no completa alguno de los campos requeridos para un determinado recurso, el sistema le notifica con un mensaje que no puede finalizar la acción hasta completar todos los campos obligatorios.</p>

Caso de uso	P-6: Gestionar objeto de aprendizaje.
Actores	Profesor, estudiante o administrador.
Objetivo	Interaccionar con un objeto de aprendizaje.
Entrada	El usuario debe haber iniciado sesión con su cuenta y estar visualizando un objeto de aprendizaje.
Salida	Los resultados de la interacción según la acción realizada se reflejan en la base de datos y en la web.
Secuencia normal	<ol style="list-style-type: none"> 1. El usuario se encuentra con una ventana donde puede visualizar cualquier archivo o links existente en el recurso. 2. En el caso de los archivos pueden descargarlos.
Secuencia alternativa	El usuario decide cancelar el proceso pulsando sobre pulsando sobre cualquier otra opción disponible..

4. Tecnologías aplicadas.

En este capítulo se explica individualmente las tecnologías utilizadas para este proyecto, distinguiendo entre las tecnologías usadas en la parte del cliente o del servidor, tecnologías semánticas o las usadas para bases de datos.

4.1 Tecnologías de la parte del cliente.

Se analizan las diferentes tecnologías utilizadas en la parte de cliente, estas tecnologías son JavaScript, HTML, CSS y Bootstrap.

JavaScript

Es un lenguaje de programación interpretado, orientado a objetos, basado en prototipos porque utiliza funciones como constructores de objetos, imperativo y estructurado porque es compatible con gran parte de la estructura de programación de C, débilmente tipado y dinámico.

Implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. A continuación, se proporciona un enlace para complementar la información de esta herramienta, <https://www.w3schools.com/js/> [16].

HTML

Es un lenguaje de marcado para la elaboración de páginas web, define una estructura básica y un código. Los elementos son la estructura básica y tienen dos propiedades, atributos y contenido, los atributos en su mayoría son pares nombre-valor.

Este enlace <https://www.w3schools.com/html/> [29], proporciona más información sobre HTML, el cual es un estándar que sirve de referencia del software conectando la elaboración de páginas web en sus diferentes versiones.

CSS

Es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en lenguaje de marcado. En el siguiente enlace aparece más información sobre esta tecnología, <https://www.w3schools.com/css/> [31].

Sirve para separar la presentación y la escritura, mejorando la accesibilidad del documento. Permite que varios documentos HTML compartan el mismo estilo y reduce la complejidad y la repetición de código.

Bootstrap

Es un *framework* [8] web cuyo objetivo es facilitar el diseño de sitios y aplicaciones web *responsive* [9].

Se utiliza como diseño de sitios y aplicaciones web, además, contiene plantillas de diseño con tipografía, formularios, botones, cuadros y otros elementos de diseño basado en HTML y CSS.

Posee una estructura modular, componentes reutilizables, *plugins* [34] de JavaScript y comprende las hojas de estilo CSS. En el siguiente enlace aparece más información sobre esta herramienta, <https://getbootstrap.com/> [7].

4.2 Tecnologías de la parte del servidor.

Se analiza la tecnología PHP usada en la parte del servidor.

PHP

Es un lenguaje de programación de propósito general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico, orientado al desarrollo de aplicaciones web y al cliente, posee la capacidad de conexión con la mayoría de los motores de base de datos.

El código es interpretado por un servidor web con un módulo de procesador de PHP, <https://php.net/manual/es/intro-what-is.php> [19], que genera HTML resultante. Dicho código fuente escrito es invisible al navegador web.

4.3 Tecnologías semánticas.

Se analiza la tecnología de SPARQL.

SPARQL

Se trata de un lenguaje estandarizado para la consulta de grafos RDF, siendo una tecnología clave en el desarrollo de la web semántica.

En un principio SPARQL únicamente incorpora funciones para la recuperación de sentencias RDF. Sin embargo, algunas funciones también incluyen operaciones para el mantenimiento de datos.

Las características principales de esta tecnología son: utilizar un lenguaje de consulta para RDF, está basado en características SQL y los resultados de una consulta pueden ser

conjuntos de resultados XML o grafos RDF. Estas y otras características se encuentran en el siguiente enlace <http://skos.um.es/TR/rdf-sparql-query/> [4].

4.4 Bases de datos.

Se analizan las diferentes tecnologías utilizadas en la parte de cliente, estas tecnologías son MySQL y phpMyAdmin.

MySQL

Es un sistema de gestión de base de datos relacional considerada como la base de datos de código abierto más popular del mundo, <https://www.mysql.com/> [13].

Archiva datos en tablas separadas, en vez de colocar todos los datos en un gran archivo, esta disponible en gran cantidad de plataformas y sistemas, posibilidad de selección de mecanismos de almacenamiento.

Gracias a una conectividad segura permite las transacciones, claves foráneas y la búsqueda e indexación de campos de texto.

PhpMyAdmin

Es una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web, utilizando un navegador web.

Puede crear y eliminar bases de datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en campos, administrar privilegios y exportar datos en varios formatos.

Este es el enlace <https://www.phpmyadmin.net/> [35], una herramienta capaz de administrar múltiples servidores, crear gráficos PDF del diseño de la base de datos y crear consultas complejas entre otras características.

4.5 Otras tecnologías.

Se analizan las diferentes tecnologías utilizadas en la parte de cliente, estas tecnologías son Google Drive, Sublime text, Xampp, Infinity Free, Modelio Open Source y Formularios de Google.

Google drive

Es un servicio de alojamiento de archivos, cuyo enlace es https://www.google.com/intl/es_ALL/drive/ [36]. Tiene la capacidad para sincronizar

archivos y documentos con el ordenador, proporcionando una visualización mejorada de documentos de Google fuera de línea.

Entre otras características destaca por editar archivos en el ordenador y tenerlos disponibles en la nube, cuenta con un respaldo automático, un control de versiones y permite subidas o bajadas masivas.

Sublime Text

Es un editor de texto y editor de código fuente que permite la edición de código en diferentes lenguajes de programación.

Posee una previsualización de la estructura del código, selección múltiple de un término por diferentes partes del código, acceso rápido a línea o archivo, coloreado y envoltura de sintaxis y se pueden abrir varios documentos y organizarlos en pestañas. Estas y otras características se encuentran en este enlace <https://www.sublimetext.com/> [11].

Xampp

Xamp <https://www.apachefriends.org/es/index.html> [12], Es un paquete de software libre. Consiste principalmente en el sistema de gestión de bases de datos MySQL, al servidor web Apache y los intérpretes para lenguajes script PHP.

Destaca por ser segura, fácil de usar, capaz de interpretar páginas dinámicas y estar disponible para varios sistemas operativos.

Infinity free

Es un *hosting* gratuito, <https://infinityfree.net/> [17]. Permite subir cualquier trabajo sin necesidad de salir del servidor.

Destacado por ofrecer un servicio gratuito e ilimitado, 99,9% de tiempo de actividad y no contener anuncios.

Modelio Open Source

Es una herramienta UML de código abierto compatible con otros estándares. Sirve tanto para el modelado de servicios de entornos distribuidos como para el modelador de procesos de negocios.

También admite los perfiles UML2 , interoperabilidad y módulos de la comunidad que agregan soporte para el modelado. Este es su enlace <https://www.modelio.org/> [18] .

Formularios de Google

Es una herramienta de Google Docs, cuyo enlace es <https://www.google.es/intl/es/forms/about/> [37]. Permite crear un formulario desde Google Drive o a partir de una hoja de cálculo existente.

Respuestas inmediatas, encuestas con estilo, personalización de preguntas, diseño adaptativo, estructura organizada y análisis de respuestas son algunas características principales de esta tecnología.

5. Modelado de datos

En este capítulo se describe el modelo entidad-relación utilizado y la relación entre este modelo y la base de datos implementada.

5.1. Modelo Entidad-Relación.

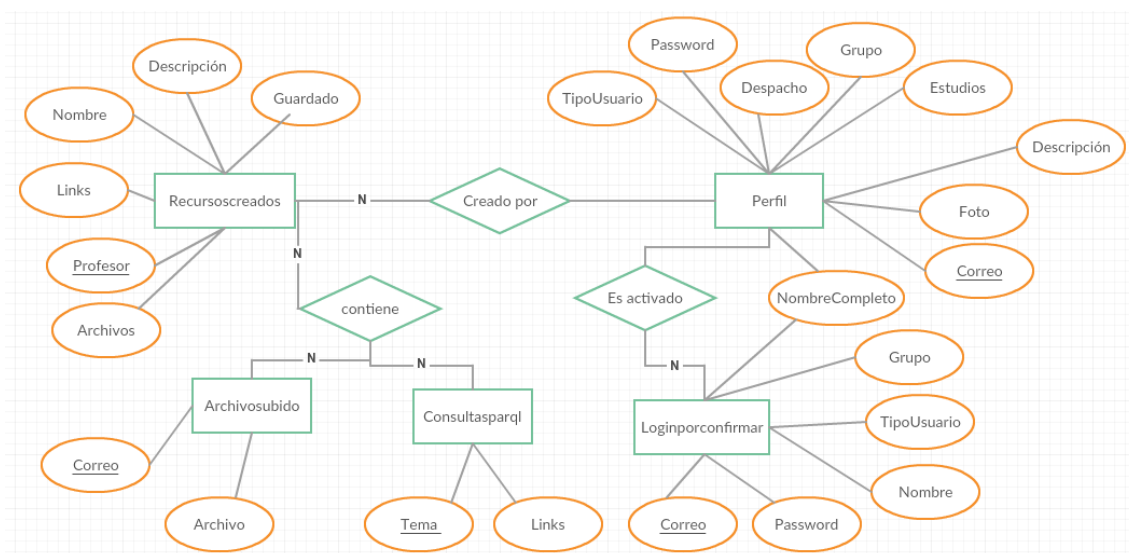


Figura 4: Diagrama entidad – relación.

Se analiza el diagrama entidad-relación de la Figura 4, las cuentas pendientes de activación deben ser activadas por una única persona, esto provoca la relación (1 : N) entre las entidades “Perfil” y “Loginporconfirmar” respectivamente. El profesor puede generar N recursos, pero estos solo pueden tener un único creador, lo cual permite una relación total con la entidad “Recursoscreados”.

Los recursos contienen información, esa información se extrae de las entidades “Archivosubido” y “Consultassparql”, como los recursos pueden contener cualquier información se produce una relación (N : N) con la entidad “Recursoscreados”.

5.2. Base de datos.

Se implementa una base de datos relacional a partir del diagrama entidad-relación del apartado anterior, almacenando información de los tres tipos de usuarios que se gestionan en la aplicación: administrador, profesor y alumno.

Para implementar el diagrama entidad-relación se han creado cinco tablas en la base de datos, dichas tablas son: “Perfil”, “Loginporconfirmar”, “Archivosubido”, “Recursoscreados” y “Consultassparql”. Utilizando los atributos subrayados en la Figura 4 como claves primarias y creando las conexiones necesarias entre tablas se consigue esta funcionalidad.

Para gestionar los archivos y links se almacena esta información en los campos “Archivos” y “Links” de la tabla “Recursoscreados”, los títulos correspondientes a las tablas “Archivosubido” y “Consultassparql” respectivamente se guardan separados por comas, dichos títulos son proporcionados por los campos “Archivos” y “links”.

Se implementan estas dos tablas similares “Loginporconfirmar” y “Perfil”, para distinguir entre usuarios existentes en el sistema y los que están pendientes de activación.

A continuación, se define la funcionalidad y los campos que componen las tablas representadas en la Figura 5.

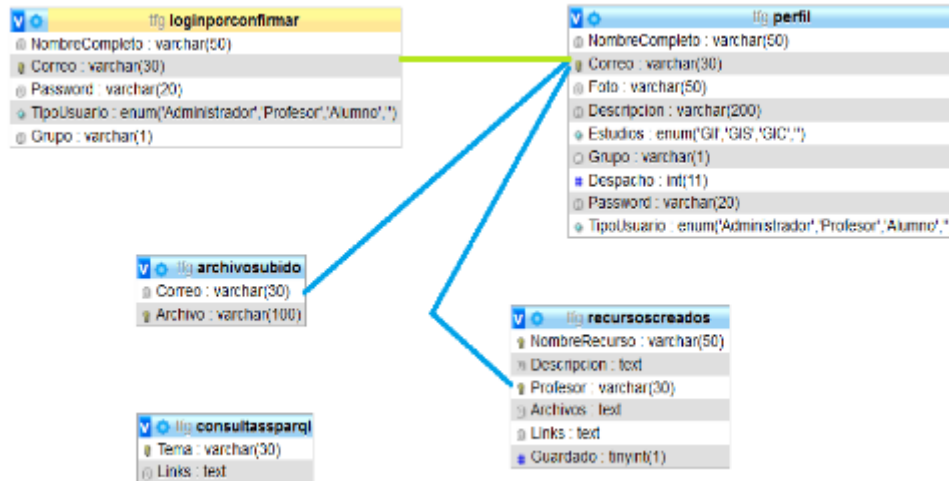


Figura 5: Diagrama de la base de datos.

Tabla Perfil

Esta tabla contiene información relacionada con los tres tipos de usuarios existentes en la herramienta, almacenando los datos correspondientes para la identificación del usuario. Compuesta por los siguientes campos.

- NombreCompleto: Contiene el nombre y apellidos del usuario.
- Correo: Es la clave primaria, se utiliza como el identificador del usuario y para iniciar sesión.
- Foto: Contiene la ruta al directorio donde se encuentra la foto correspondiente al usuario.
- Descripción: Campo donde el usuario puede describir brevemente algo sobre su persona o citar alguna frase que le haga sentirse identificado.
- Estudios: Enumerado que se utiliza solamente para el tipo de usuario alumno, teniendo las posibilidades de elegir GII (Grado en Ingeniería Informática), GIS (Grado en Ingeniería de Software) o GIC (Grado en Ingeniería de Computadores).
- Grupo: Campo válido solamente para alumno y profesores, establece la relación de grupo entre ambos usuarios.
- Despacho: Campo que facilita la información sobre el despacho del profesor.

- Password: Este campo contiene el hash de la contraseña, el cual es utilizado para validar el acceso a las cuentas de usuario junto con el correo.
- TipoUsuario: Enumerado de los tres tipos de usuarios existentes en la herramienta, administrador, profesor y alumno.

Tabla LoginPorConfirmar

Esta tabla contiene información relacionada con las cuentas pendientes de activación, sean de un profesor o de un alumno. Compuesta por los siguientes campos.

- NombreCompleto: Contiene el nombre y apellidos del usuario.
- Correo: Es la clave primaria, se utiliza como el identificador del usuario y para iniciar sesión, una vez confirmada la cuenta pasa a ser la clave primaria de la tabla “Perfil”.
- Password: Este campo contiene el hash de la contraseña, el cual es utilizado para validar el acceso a las cuentas de usuario junto con el correo.
- TipoUsuario: Enumerado de los tres tipos de usuarios existentes en la herramienta, administrador, profesor y alumno. Contiene el valor elegido por el usuario en el momento de registrarse.
- Grupo: Contiene el grupo seleccionado por el usuario en el momento de registrar la cuenta.

Tabla Archivosubido

Esta tabla contiene información relacionada con un archivo subido y esta compuesta por los siguientes campos.

- Correo: Identifica al usuario que ha subido el archivo.
- Archivo: Clave primaria, además contiene la ruta al directorio donde se encuentra almacenado el archivo.

Tabla Recursoscreados

Esta tabla contiene información relacionada con un recurso, puede ser un recurso en estado guardado o ya creado y publicado en la web, sus campos son los siguientes.

- NombreRecurso: Clave primaria, almacena el título relacionado con el recurso creado.
- Descripción: Campo que contiene una breve descripción de la información almacenada en este recurso.
- Profesor: Campo que contiene el correo electrónico que identifica al usuario que ha creado este recurso.
- Archivos: Campo de tipo texto que almacena todos los nombres de los archivos utilizados en este recurso separados por comas.

- **Links:** Campo de tipo texto que almacena todos los nombres de los links utilizados en este recurso separados por comas.
- **Guardado:** Campo booleano que identifica si un recurso está guardado o ha sido creado.

Tabla consultassparql

Esta tabla contiene información relacionada con los resultados obtenidos por las consultas SPARQL y esta compuesta por los siguientes campos.

- **Tema:** Clave primaria, campo que identifica el tema sobre el que se han recogido los resultados obtenidos en la consulta.
- **Links:** Campo de tipo texto que almacena todos los links obtenidos en la consulta de SPARQL realizada para este tema en concreto.

6. Arquitectura de la aplicación.

La arquitectura utilizada para la aplicación es el modelo cliente/servidor. Este modelo se caracteriza por el reparto de tareas entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes.

Separando los datos y la lógica de su representación y su módulo para gestionar comunicaciones se cumple con el patrón modelo-vista-controlador. Actuando la base de datos como modelo, el código como servidor y el cliente para la representación visual separando así conceptos. El cliente interactúa con la herramienta generando acciones sobre el controlador, dependiendo de dichas interacciones provoca modificaciones en la base de datos produciendo así cambios en la vista.

Se ha utilizado como entorno de desarrollo XAMPP, el cual integra en una sola herramienta PHP [19], MySQL y Apache [20]. El cliente solicita una acción al servidor web, Apache en este caso, el cual recibe el mensaje, comprueba que se trata de una petición válida, y al comprobar que es una extensión “.php” solicita al intérprete de PHP. Este intérprete a su vez ejecuta el contenido del archivo y eventualmente se comunica con el gestor de base de datos, mediante MySQL. Si se produce la comunicación, este gestor es capaz de devolver la información solicitada contenida en la base de datos.

Finalizado la ejecución del código contenido en el archivo y recibida toda la información necesaria, se envían los resultados al servidor web. El servidor envía la página al cliente solicitante y el navegador muestra por pantalla los resultados.

El funcionamiento descrito anteriormente se representa de manera visual en la Figura 6, donde se indica quien desempeña cada papel dentro de esta aplicación.

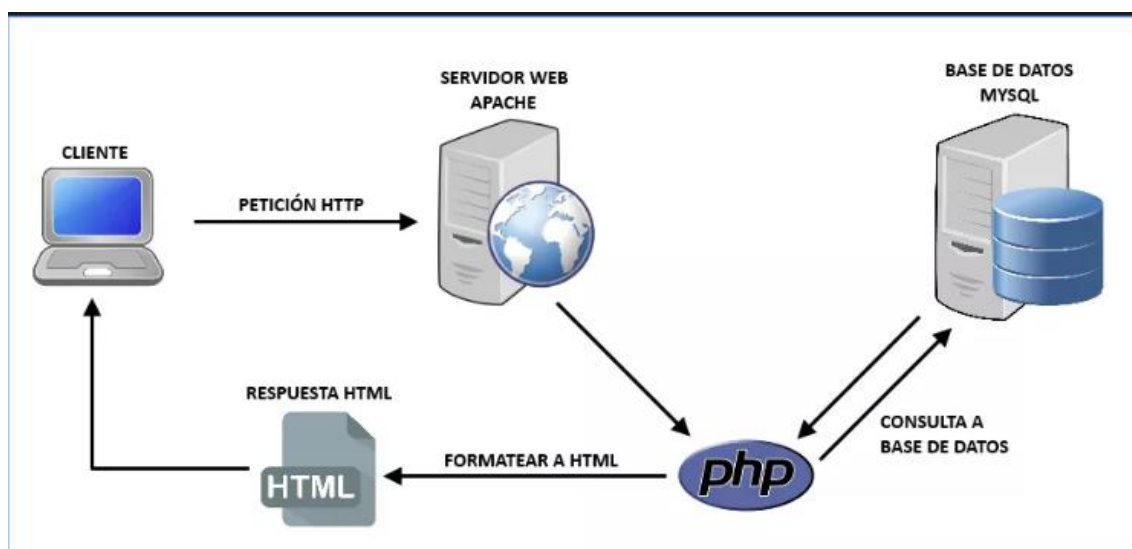


Figura 6: Funcionamiento de PHP, MySQL y apache con cliente.

7. Diseño e implementación.

Este capítulo se introduce con los principios utilizados para el desarrollo de la herramienta. Después se explica con detalle la funcionalidad implementada en la aplicación.

7.1 Principios de usabilidad.

Como la mayoría de los diseñadores web, en este proyecto se ha procurado cumplir con los principios de usabilidad web de Nielsen [21], estos principios son la base de cualquier web para que sea *user friendly* o lo que es lo mismo, es un tipo de diseño centrado en el usuario para conseguir mejorar la experiencia de este.

Algunos de los principios que se tratan en este proyecto:

- Relación entre el sistema y el mundo real: se utilizan imágenes claras buscando que la interacción con el usuario sea lo más natural posible.
- Control y libertad del usuario: se proporciona al usuario la posibilidad de subsanar el error, por ello en la web se incluye la posibilidad de editar el perfil o los recursos.
- Consistencia y estándares: utilizar los convenios establecidos para ciertos iconos o botones, por ello se establece un menú de navegación idéntico en todas las interfaces, incluyendo botones con colores representativos acordes a la acción que desempeñan.
- Diseño estético y minimalista: se hace uso de un estilo minimalista, acompañado de una gama de colores claros que favorece la usabilidad de la herramienta. La imagen utilizada como fondo simula un entorno de trabajo familiar, buscando la sensación de paz y tranquilidad del usuario, lo cual se acompaña de un diseño no sobrecargado generando una herramienta sencilla e intuitiva.
- Ayudar a los usuarios a reconocer: diagnosticar y corregir los errores, se implementa un lenguaje entendible por todos, indicando a los usuarios que es lo que pasa en cada momento con mensajes explicativos de los errores.

7.2 Funcionalidad.

Se describe la estructura de la aplicación mostrando las clases del sistema, sus atributos, operaciones (o métodos), y las relaciones entre los objetos, Figura 7.

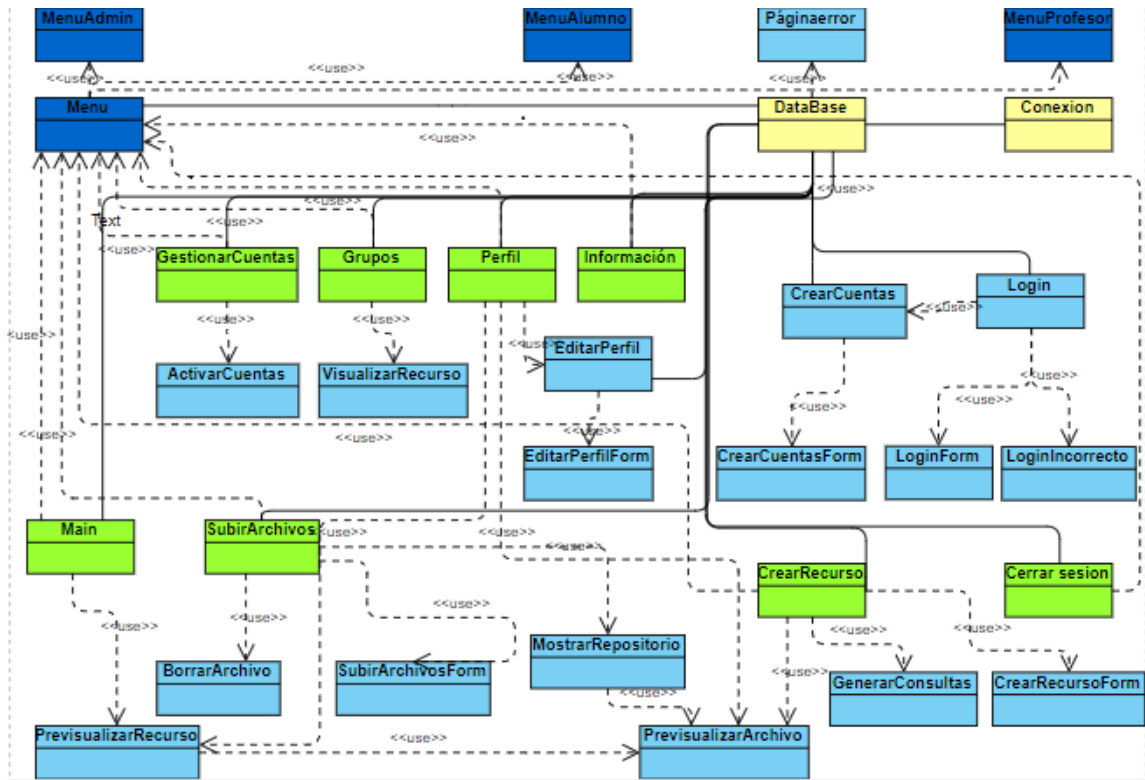


Figura 7: Diagrama de clases.

A continuación, se explica detalladamente cada una de las funcionalidades implementadas.

a) Inicio de sesión con contraseña encriptada.

En primer lugar, cuando el usuario accede a la herramienta debe identificarse con su cuenta de usuario. Se le requerirá su correo electrónico y contraseña, como se ve en la figura 8.

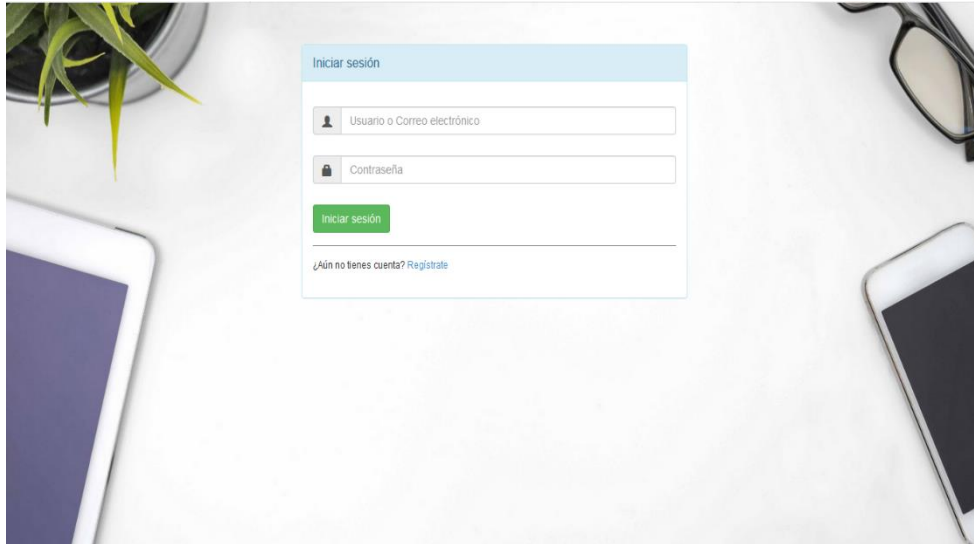


Figura 8: Inicio de sesión.

Una vez registrado y activado el usuario podrá iniciar sesión, para ello se ha implementado un formulario donde se recoge el correo electrónico y la contraseña encriptada introducidas, realizando una consulta sobre la tabla “Perfil” de la base de datos para su comprobación.

El método utilizado para encriptar las contraseñas, el cual también está implementado en la clase “crearCuentaForm.php”, se realiza con una función propia de PHP para calcular el hash de un *string* [23], el método es `md5(string)`, utilizando el resultado arrojado por el método en la función `crypt(string, salt)`, se consigue un *string encriptado*. Este proceso se representa en la Figura 9.

```
<?php
if((include 'database.php')==FALSE) {
    header('Location: paginaerror.php');
}
$usuario = htmlspecialchars(trim(strip_tags($_POST['usuario'])));
$contr = htmlspecialchars(trim(strip_tags($_POST['password'])));

$salt = md5($contr);
$password_encriptado = crypt($contr, $salt);

$consulta = loginUsuario($usuario, $password_encriptado);
if ($consulta) {
    $fila=mysqli_fetch_assoc($consulta);
    if($fila['Correo'] == $usuario) {
        session_start();
        $_SESSION['Correo']=$usuario;
        header('Location: main.php');
    }
    else {
        header('Location: loginIncorrecto.php');
    }
}
?>
```

Figura 9: Confirmar inicio de sesión en código.

b) Registro de usuario.

Si el usuario no posee una cuenta en la web, puede registrarse rellorando los distintos campos que se le exigen. Estos campos son: nombre y apellidos, correo electrónico con el que desea ser identificado, contraseña y el tipo de usuario acompañado del grupo al que se perteneces, Figura 10.

```
<div style="margin-bottom: 25px" class="input-group">
  <span class="input-group-addon"><i class="glyphicon glyphicon-user"></i></span>
  <input id="login-Nombre" type="text" class="form-control" name="nombre" value="" placeholder="Nombre y
  Apellidos" required>
</div>

<div style="margin-bottom: 25px" class="input-group">
  <span class="input-group-addon"><i class="glyphicon glyphicon-user"></i></span>
  <input id="login-username" type="text" class="form-control" name="usuario" value="" placeholder="Usuario o
  Correo electrónico" required>
</div>

<div style="margin-bottom: 25px" class="input-group">
  <span class="input-group-addon"><i class="glyphicon glyphicon-lock"></i></span>
  <input id="login-password" type="password" class="form-control" name="password" placeholder="Contraseña"
  required>
</div>

<div style="margin-bottom: 25px" class="input-group col-md-12">
  <div class="col-md-6">
    <input type="radio" name="estado" value="3" required> Estudiante
    <select name="Estudiante">
      <while($fila=mysqli_fetch_array($1stGrupos)){
        <echo '
          <option value="" . $fila['1'] . '>Grupo: ' . $fila['1'] . ' - ' . $fila['0'] . '</
          option>';
        }
      <echo ' </select>
    </div>
    <div class="col-md-6">
      <input type="radio" name="estado" value="2" required> Profesor
      <input id="login-username" type="text" class="form-control" name="Profesor" value="" placeholder="Escriba el
      grupo deseado">
    </div>
  </div>
```

Figura 10: Registro de usuario.

La implementación de esta funcionalidad es muy similar al realizada en el inicio de sesión. A través de un formulario se recogen los datos introducidos por el usuario y mediante consultas a la base de datos se insertan los resultados en la tabla “Loginporconfirmar”.

c) Página principal.

La Figura 11 representa la página principal para los usuarios, dependiendo del tipo que seas la cabecera contiene diferentes opciones. Dichas opciones se muestran más adelante.

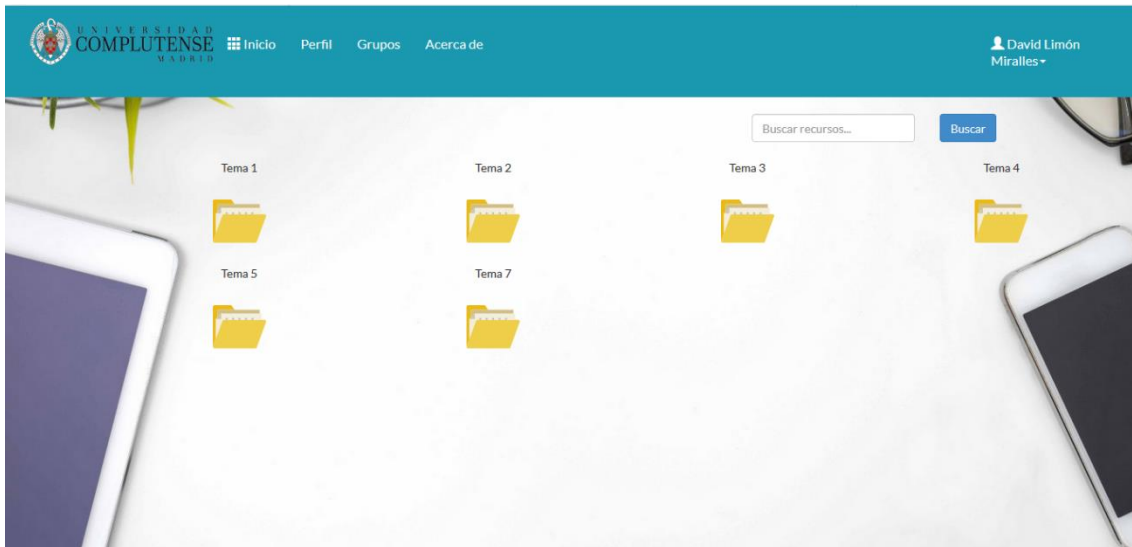


Figura 11: Página principal para el tipo de usuario alumno.

Para mostrar los recursos que corresponden al usuario registrado se realizan varias consultas. La primera de ellas devuelve un resultado con los campos correspondientes, utilizando el valor del campo “TipoUsuario” para determinar la visual que le corresponde.

Después se ejecuta una consulta sobre la tabla “Recursoscreados” utilizando el correo del usuario y el booleano “Guardado” para identificar que no es un recurso creado, como se puede apreciar en la Figura 12, retornando así los datos correspondientes a mostrar. Con los resultados arrojados por la consulta y dependiendo de si se ha realizado una búsqueda en la página principal se muestran de una manera u otra.

```

if($tipo['TipoUsuario'] == "Profesor" ){
    $nombreRecursos = buscarRecursosCreados($usuario, 0);
    if(mysqli_num_rows($nombreRecursos)==0) $nofilas = true;
    else $nofilas = false;
    echo '<div class="bloque">';

    if($nofilas){
        echo '<div class="informacion col-md-12">
        <p> No hay recursos creados. Puedes crear uno pulsando <a href="crearRecurso.php" >aqui.</a> </p>
        </div>';
    }else{
        if($recursoBuscado){
        }else{
            echo '<div class="row col-md-11 col-md-offset-1">';
            while($fila=mysqli_fetch_array($nombreRecursos)){
                echo '<div class="col-sm-3">
                <div class="tituloRecurso">
                    . $fila[0] . '
                </div>
                <a href="previsualizarRecurso.php?titulo=' . $fila[0] . '" >
                <div class="bloc fa-6">
                    
                </div>
                </a>
                </div>';
            }
            echo '<div>';
        }
    }
}

```

Figura 12: Mostrar los recursos de la página principal.

La interfaz para el alumno contiene los recursos generados por el profesor de su grupo, además cuenta con la posibilidad de acceder a los demás grupos seleccionando la opción del menú “Grupos”, esto se ve representado en la Figura 13. Para implementarlo simplemente se realizan consultas a la tabla “Perfil” de la base de datos, Figura 14, obteniendo los campos necesarios para identificar el tipo de usuario y el grupo al que pertenece.



Figura 13: Cabecera de la página principal para el tipo de usuario profesor.

```
function perfilUsuario($usuario) {
    if(!require 'conexion.php') {
        header('Location: paginaerror.php');
    }
    $sql= "SELECT NombreCompleto, Correo, Foto, Descripcion, Estudios, Grupo, Despacho FROM perfil WHERE perfil.Correo like '$usuario' ";
    $result = mysqli_query($db, $sql);
    mysqli_close($db);
    return $result;
}
```

Figura 14: Consulta a la tabla “Perfil” de la base de datos.

El profesor tiene la misma interfaz que el administrador, con la diferencia de que los recursos que le aparecen son los suyos propios y en la cabecera él tiene la opción “Gestionar alumnos”, la cual se explicará posteriormente.

Por otro lado, el desplegable del nombre contiene diferentes opciones para cada usuario, en el caso del administrador y el alumno solo tienen la posibilidad de cerrar sesión, mientras que el profesor también cuenta con la posibilidad de “Subir un archivo” y “Crear un recurso”.

La interfaz del administrador sigue el mismo patrón, pero con la diferencia de que contiene todos los recursos creados por todos los profesores, además en su cabecera tiene la opción Gestionar cuentas. Esto se puede apreciar en la Figura 15.

```

}else if($tipo['TipoUsuario'] == "Administrador" ){
    $nombreRecursos = buscarTodosRecursos(0);
    if(mysqli_num_rows($nombreRecursos)==0) $nofilas = true;
    else $nofilas = false;
    echo '<div class="bloque">';
    if($nofilas){
        echo '<div class="informacion col-md-12">
            <p> No hay recursos creados. </p>
        </div>';
    }else{
        if($recursoBuscado){
            echo '<div class="row col-md-11 col-md-offset-1">';
            $contador = 0;
            while($fila=mysqli_fetch_array($consulta)){
                $contador += 1;
                echo '<div class="col-sm-3">
                    <div class="tituloRecurso">
                        . $fila[0] . <br>
                        . $fila[2] .
                    </div>
                    <a href="previsualizarRecurso.php?titulo=' . $fila[0] . '&correo=' . $fila[2] . '" >
                        <div class="bloc fa-6">
                            
                        </div>
                    </a>
                </div>';
            }
        }
    }
}

```

Figura 15: Página principal para el tipo de usuario administrador.

d) Buscador de recursos.

La página principal posee un buscador, el cual facilita la búsqueda de los recursos al usuario.

Para implementar la funcionalidad del buscador, Figura 16, se utiliza un formulario con el role *search*, pasando los datos recogidos por el método POST, utilizando estos resultados en la consulta a la base de datos sobre la tabla “Recursoscreados”. Para lanzar la búsqueda basta con pulsar la tecla *enter* o seleccionar el botón proporcionado “Buscar”.

```

if(!empty($_POST["Buscar_todos"])){
    $recursoBuscado = true;
    $filtrar = htmlspecialchars(trim(strip_tags($_POST['q'])));
    if($tipo['TipoUsuario'] == "Alumno" ){
        $consulta = buscarLosRecursos($filtrar, $profesor[0], 0);
    }else if($tipo['TipoUsuario'] == "Profesor" ){
        $consulta = buscarLosRecursos($filtrar, $usuario, 0);
    }else{
        $consulta = buscarTodosRecursosBuscador($filtrar, 0);
    }
}

echo '
<div class="buscador col-md-10 col-md-offset-1">
    <div class="col-md-4">
        <form action="main.php" method="POST" role="search">
            <div class="col-md-8">
                <input class="form-control" type="search" id="miBusqueda" name="q" placeholder="Buscar recursos..." aria-label="Buscar en el contenido del sitio">
            </div>
            <div class="col-md-4">
                <input class="btn btn-primary" type="submit" value="Buscar" id="Buscar_todos" name="Buscar_todos" />
            </div>
        </form>
    </div>
</div>';

function buscarLosRecursos($palabra, $profesor, $guardado){
    if(!require 'conexion.php') {
        header('Location: paginaerror.php');
    }
    $sql = "SELECT NombreRecurso FROM recursoscreados where recursoscreados.NombreRecurso LIKE '%$palabra%' AND recursoscreados.Profesor LIKE '$profesor' AND recursoscreados.Guardado LIKE '$guardado'";
    $result = mysqli_query($db, $sql);
    mysqli_close($db);
    return $result;
}

```

Figura 16: Buscador de la página principal.

e) Perfil.

Todo usuario tiene la posibilidad de editar los datos de su perfil. Dependiendo del usuario que seas se mostrará una interfaz u otra, todas contienen una serie de campos idénticos para cualquier rol. En la Figura 17 se aprecia el editar perfil de un alumno.

```
<div>
<div class="col-md-4 col-sm-6 col-xs-12">
  <div class="text-center">
    

    <input type="file" class="text-center center-block well well-sm" name="foto" type="file" value="'. $perfil['Foto'] . '" maxlength="50">
  </div>
</div>
<!-- edit form column -->
<div class="col-md-8 col-sm-6 col-xs-12 personal-info">

  <div class="form-group">
    <label class="col-lg-3 control-label">Nombre:</label>
    <div class="col-lg-8">
      <input class="form-control" name="nombre" value="'. $perfil['NombreCompleto'] . '" type="text">
    </div>
  </div>
  <div class="form-group">
    <label class="col-lg-3 control-label">Descripción:</label>
    <div class="col-lg-8">
      <input class="form-control" id="descripcion" name="descripcion" value="'. $perfil['Descripcion'] . '" type="text">
    </div>
  </div>
  <div class="form-group">
    <label class="col-lg-3 control-label">Estudios:</label>
    <div class="col-lg-8">
      <input placeholder = "GII, GIS o GIC" title="Grado al que perteneces: GII, GIC o GIS" class="form-control" id="estudios" name="estudios" value="'. $perfil['Estudios'] . '" type="text">
    </div>
  </div>
  <div class="form-group">
    <label class="col-md-3 control-label"></label>
    <div class="col-md-8">
      <input class="btn btn-primary" value="Guardar cambios" type="submit">
      <a href="perfil.php" class="btn btn-default">Cancelar</a>
    </div>
  </div>
</div>
```

Figura 17: Página del perfil para un alumno.

El perfil para el Administrador tiene la misma interfaz que el mostrado en la imagen anterior para el alumno, cambiando algunos campos no relevantes para el administrador.

La interfaz para el profesor, Figura 18, es diferente a los otros usuarios, contiene un par de pestañas, “Mis Recursos” y “Mis alumnos”, donde se muestra la información correspondiente en cada caso.

```

<div class="aux col-md-8">
  <div class="tab-content">
    <div id="home" class="tab-pane fade in active">;
      while($fila=mysqli_fetch_array($listaRekursos)){
        echo '
          <div class="Recurso col-md-8">
            <a href="previsualizarRecurso.php?titulo=' . $fila['NombreRecurso'] . '&correo=' . $usuario . '" >
              <div class="fila">
                <label>' . $fila['NombreRecurso'] . '</label>
                <p>' . $fila['Descripcion'] . '</p>
              </div>
            </a>
          </div>
        ';
      }
      echo '
    </div>
    <div id="menu1" class="tab-pane fade">;
      while($fila=mysqli_fetch_array($listaAlumnos)){
        echo '
          <div class="Recurso col-md-8">
            <a href="previsualizarAlumno.php?nombre=' . $fila['NombreCompleto'] . '" >
              <div class="fila ">
                <label>' . $fila['NombreCompleto'] . '</label>
                <p>' . $fila['Correo'] . '</p>
              </div>
            </a>
          </div>
        ';
      }
      echo '
    </div>
  </div>

```

Figura 18: Página del perfil para un profesor.

La implementación consiste en recoger los datos introducidos por el usuario a través de un formulario usando el método POST. Una vez almacenada la foto en nuestro servidor, realizamos una consulta sobre la tabla “Perfil”, actualizando los campos completados por el usuario.

Para cada tipo de usuario se realiza su propia consulta, puesto que tienen algunos campos que son diferentes. También se realiza una comprobación para el campo foto evitando que se recojan valores *null*, véase en la Figura 19.

```

$nombre = htmlspecialchars(trim(strip_tags($_POST['nombre'])));
$correo = $usuario;
$descripcion = $_POST['descripcion'];
$grupo = $_POST['grupo'];
$despacho = $_POST['despacho'];

$target_dir = $_FILES["foto"]["tmp_name"]; // guardamos el nombre temporal
$name = $_FILES["foto"]["name"]; // guardamos el nombre del archivo subido
if(!empty($name)){
  $foto = "imagenes/" . $name; // creamos el nombre de archivo que se quiere almacenar
  move_uploaded_file($target_dir, $foto); // almacenamos en la carpeta la nueva imagen

  $consulta = editarPerfil($nombre, $correo, $foto, $descripcion, NULL, $grupo, $despacho);
}else{
  $consulta = editarPerfilSinFoto($nombre, $correo, $descripcion, NULL, $grupo, $despacho);
}
if ($consulta) {
  header('Location: perfil.php');
}else{
  header('Location: perfil.php?error=ok');
}

```

Figura 19: Editar perfil.

f) Gestionar cuentas.

Esta interfaz solo es posible para el tipo de usuario administrador y profesor, Figura 20, se muestra en forma de lista las cuentas pendientes de activación, en el caso del administrador, tendrá la posibilidad de activar o rechazar las cuentas de los profesores y el profesor es quien dará de alta las cuentas de los alumnos relacionados con su grupo.

```
while($fila=mysqli_fetch_array($consultaDelAdmin)){  
  
    echo '<form action="activarCuentas.php?contr='.$fila['2'].'" method="POST" >  
        <tr class="filaTabla">  
            <td><input id="nombre" name="nombre" value= '' . $fila['0'] . '' readonly="readonly" type="text" ></td>  
            <td><input id="correo" name="correo" value= '' . $fila['1'] . '' readonly="readonly" type="text" ></td>  
            <td><input id="tipo" name="tipo" value= '' . $fila['3'] . '' readonly="readonly" type="text" ></td>  
            <td><input id="grupo" name="grupo" value= '' . $fila['4'] . '' readonly="readonly" type="text" ></td>  
            <td><input type="submit" value="Activar" id="Activar" name="Activar" class="btn btn-success"/></td>  
            <td><input type="submit" value="Rechazar" id="Rechazar" name="Rechazar" class="btn btn-danger" /></td>  
        </tr>  
    </form>  
    ;  
}  
echo'</table>  
</div>;  
}  
else{  
    echo '<div class="rowPerfil col-md-10 col-md-offset-2">  
  
    <table class="tablaProfesores" WIDTH="75%" style="text-align:center;">  
  
    <tr> <th>Nombre</th> <th>Correo</th> <th>Tipo</th> <th>Grupo</th> <th></th> <th></th> </tr>;  
    while($fila=mysqli_fetch_array($consultaDelProfesor)){  
  
        echo '  
        <form action="activarCuentas.php?contr='.$fila['2'].'" method="POST" >  
            <tr class="filaTabla">  
                <td><input id="nombre" name="nombre" value= '' . $fila['0'] . '' readonly="readonly" type="text" ></td>  
                <td><input id="correo" name="correo" value= '' . $fila['1'] . '' readonly="readonly" type="text" ></td>  
                <td><input id="tipo" name="tipo" value= '' . $fila['3'] . '' readonly="readonly" type="text" ></td>  
                <td><input id="grupo" name="grupo" value= '' . $fila['4'] . '' readonly="readonly" type="text" ></td>  
                <td><input type="submit" value="Activar" id="Activar" name="Activar" class="btn btn-success"/></td>  
                <td><input type="submit" value="Rechazar" id="Rechazar" name="Rechazar" class="btn btn-danger"/></td>  
            </tr>  
        </form>  
    </div>
```

Figura 20: Gestionar cuentas para un profesor.

Para gestionar el control de activación de cuentas se utilizan consultas a la base de datos, mostrando los datos de los usuarios pendientes de activación de la tabla Loginporconfirmar. A través de un formulario y dependiendo del botón seleccionado por el usuario se determina la consulta a realizar, Figura 21.

Ejecutando un *insert* sobre la tabla “Perfil” con el nuevo usuario y borrando el propio usuario creado en la tabla “Loginporconfirmar” en caso de haberse seleccionado la opción “Activar”, y realizando un *delete* sobre dicha tabla en caso de haberse escogido “Rechazar”.

```

if((include 'database.php')==FALSE) {
    header('Location: paginaerror.php');
}
$nombre = htmlspecialchars(trim(strip_tags($_POST['nombre'])));
$usuario = htmlspecialchars(trim(strip_tags($_POST['correo'])));
$contr = htmlspecialchars(trim(strip_tags($_GET['contr'])));
$tipo = htmlspecialchars(trim(strip_tags($_POST['tipo'])));
$grupo = htmlspecialchars(trim(strip_tags($_POST['grupo'])));

if(!empty($_POST["Activar"])){
    $consulta1 =crearCuentaPerfil($nombre, $usuario, $contr, $tipo, $grupo);
    $consulta2 =borrarCuentaPerfil($usuario);
    if ($consulta1 && $consulta2) {
        header('Location: gestionarCuentas.php?activada=ok');
    }else{
        header('Location: gestionarCuentas.php?error=ok');
    }
}else if(!empty($_POST["Rechazar"])){
    $consulta2 =borrarCuentaPerfil($usuario);
    if ($consulta2) {
        header('Location: gestionarCuentas.php?rechazada=ok');
    }else{
        header('Location: gestionarCuentas.php?error=ok');
    }
}
}

```

Figura 21: Activar cuentas.

g) Subir archivos.

La interfaz de la Figura 22 está habilitada solamente para el rol de profesor, el profesor puede subir cualquier archivo pudiendo ser utilizado por el mismo o por algún compañero en la creación de un recurso, se le muestran todos los archivos subidos por el profesor y se le brinda un buscador para facilitarle la búsqueda de algún archivo en concreto.

```

if($tipo['TipoUsuario'] == "Profesor" ){
    echo '
    <form action="subirArchivosForm.php" method="post" enctype="multipart/form-data">
        <div class="tituloArchivo">
            Seleccione el archivo:
        </div>
        <input type="file" class="text-center center-block well well-sm" name="archivo"
            accept="file_extension|audio/*|video/*|image/*|media_type" maxlength="50">
        <div class="prueba">
            <input type="submit" value="Subir archivo" class="btn btn-primary">
        </div>
    </form> ';
    include 'mostrarRepositorio.php';
}

```

Figura 22: Subir archivos por un profesor.

Para conseguir implementar la subida de objetos de aprendizaje a la web y su posterior uso en la creación de recursos se utilizan formularios con el atributo `enctype="multipart/form-data"`.

Almacenando en el sistema los archivos mediante dos carpetas. Una de ellas contiene todos los archivos subidos por todos los profesores y la otra es la carpeta propia del profesor, habrá una carpeta por cada profesor con sus respectivos archivos subidos. Guardando la ruta de los archivos en la tabla "Archivosubido" de la base de datos. El código que representa lo aquí explicado se proporciona en la Figura 23.

El mecanismo de almacenamiento de ficheros evita la sobrecarga de las bases de datos con archivos, almacenando solo la ruta al objeto de aprendizaje y no el objeto en sí.

```
if(!file_exists ("archivos/todos")){
    mkdir("archivos/todos/", 0777);
}
if(!file_exists ("archivos/" . $user)){
    mkdir("archivos/" . $user, 0777);
}
$target_dir = $_FILES["archivo"]["tmp_name"]; // guardamos el nombre temporal
$name = $_FILES["archivo"]["name"]; // guardamos el nombre del archivo subido
$archivo = "archivos/todos/" . $name; // creamos el nombre de archivo que se quiere almacenar
move_uploaded_file($target_dir, $archivo); // almacenamos en la carpeta el nuevo archivo
```

Figura 23: Subir archivos.

h) Crear recursos.

Esta página que solamente se encuentra disponible para el tipo de usuario profesor, le permite crear un recurso.

Compuesta por una cabecera donde se visualizan tres botones, el primero de ellos está acompañado de un *combox* [22] con el nombre del recurso que se desee cargar, el segundo botón es un generador de consultas que refresca las consultas realizadas en SPARQL y por último aparece un buscador, el cual remarca los archivos que contengan la búsqueda deseada.

Los siguientes campos destacados son el título y la descripción del recurso, acompañado de dos grandes cajas donde aparecen los archivos subidos por el profesor que está creando el recurso y todos los archivos subidos por todos los profesores. Debajo de las dos grandes cajas el usuario se encontrará con los links generados por las consultas.

Por último, existen tres botones: previsualizar, guardar y crear recurso respectivamente.

Todo esto se aprecia visualmente en la Figura 24.

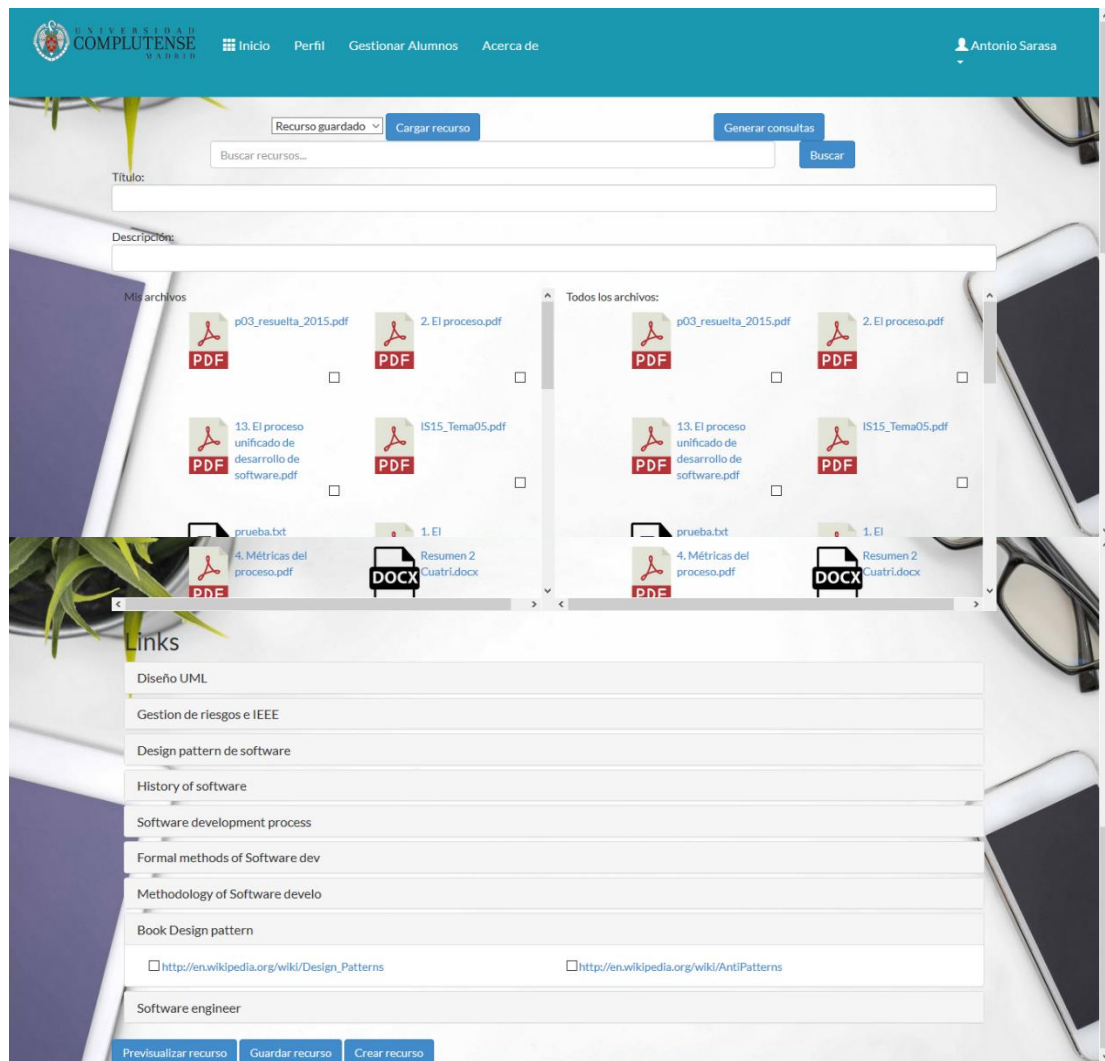


Figura 24: Crear recurso por un profesor.

La implementación sobre la creación de un recurso es compleja, contiene diferentes consultas a la base de datos y distintas funcionalidades proporcionando una gran usabilidad a la creación de un recurso.

En esta clase se distingue entre el tipo de usuario profesor, que es quien posee mayor funcionalidad y administrador. Por esto, se realiza una consulta obteniendo los datos necesarios de la tabla “Perfil”, en otra consulta se obtienen los archivos subidos por el usuario y todos los archivos existentes, utilizando la funcionalidad de la Figura 26 y Figura 27 se muestran los resultados. Para mostrar los links generados por las consultas SPARQL, se realiza una consulta sobre la tabla “Consultassparql”.

Para facilitar la creación de un recurso se proporciona un buscador, este buscador marca los objetos de aprendizaje que concuerden con la búsqueda realizada y tiene una implementación similar al descrito en la Figura 16.

Recogiendo los datos seleccionados a través de un formulario, se almacenan todos los objetos de aprendizaje y links en un mismo campo separándolos por comas, como se

muestra en la Figura 25 realizando un *insert* a la tabla “Recursoscreados” de la base de datos.

```
}else if(!empty($_POST["Crear_recurso"])){
    $titulo = $_POST["titulo"];
    $descripcion = $_POST["descripcion"];

    foreach($_POST['archivos'] as $selected){
        if($archivos == ""){
            $archivos = $selected ;
        }else{
            $archivos = $archivos . " ," . $selected ;
        }
    }
    foreach($_POST['links'] as $selected){
        if($links == ""){
            $links = $selected ;
        }else{
            if(strpos($links, $selected) == false){
                $links = $links . " ," . $selected ;
            }
        }
    }
}
```

Figura 25: Crear recurso.

i) Visualizar archivos.

La visualización de archivos se realiza mediante la etiqueta *Iframe* [24], Figura 26, esta etiqueta sirve para crear un espacio donde se puede insertar otra web. Con esta herramienta se consigue implementar varias funcionalidades, entre ellas visualizar y descargar el archivo. Para acceder a esta vista basta con seleccionar cualquier archivo existente en el sistema.

```
echo '<div class="imgArch col-md-offset-3 col-md-9">
    <iframe src="archivos/todos/' . $titulo. '" width="750" height="750" frameborder="2"><img class="archivoPrevisualizar rowImg"
        style="width:70px; height:70px;"
    </iframe> <br>
</div>';
```

Figura 26: Visualizar archivos.

j) Recorrer el directorio donde se almacenan los archivos.

Para recorrer el directorio donde se almacenan los objetos de aprendizaje, se implementa una función donde se utilizan métodos como *is_dir()*, *opendir()*, *readdir()* o *closedir()*, Figura 27. Los métodos nombrados anteriormente permiten el manejo de ficheros, mostrando los archivos dependiendo de su extensión con el icono correspondiente. El formato del archivo se determina mediante un método propio del manejo de cadenas, comparando si el nombre del archivo contiene la extensión en concreto.

```

function obtener_estructura_directorios($ruta,$archivosBusqueda, $recursoBuscado, $user){
// Se comprueba que realmente sea la ruta de un directorio
echo '<div class= "row col-md-11 col-md-offset-1"';
if (is_dir($ruta)){// Abre un gestor de directorios para la ruta indicada
$gestor = opendir($ruta);
echo "<ul>";// Recorre todos los elementos del directorio
while (($archivo = readdir($gestor)) != false) {
$ruta_completa = $ruta . "/" . $archivo; // Se muestran todos los archivos y carpetas excepto "." y ".."
if ($archivo != "." && $archivo != "..") {// Si es un directorio se recorre recursivamente
if (is_dir($ruta_completa)) {

```

Figura 27: Recorrer directorio.

k) Guardar y cargar un recurso.

La funcionalidad implementada para guardar un recurso es idéntica a lo explicado en el apartado h), donde se explica la creación de un recurso. Se recogen los datos del formulario y se guardan en la tabla “Recursoscreados” con la diferencia de que se marca el campo “Guardado” a true y no es un recurso publicado y visible para los demás usuarios.

Estos recursos se pueden cargar, permitiendo continuar con la creación del recurso hasta la posterior publicación de este. Para cargar dicho recurso se facilita un *combobox* y el botón “Cargar recurso”, una vez pulsado el botón se utiliza la opción seleccionada para realizar la consulta sobre la base de datos en la tabla “Perfil”.

l) Previsualizar recurso.

Esta funcionalidad permite al usuario visualizar el recurso antes de publicarlo. Recogiendo los valores del formulario y pasándoselos a la clase “previsualizarRecurso.php” a través de la URL [25], dicha clase muestra el recurso previsualizado, Figura 28, utilizando funcionalidades explicadas en apartados anteriores, por ejemplo, la obtención de los archivos a través de una ruta proporcionada, Figura 27, o almacenar todos los nombres de los archivos en una variable separados por comas, Figura 25.

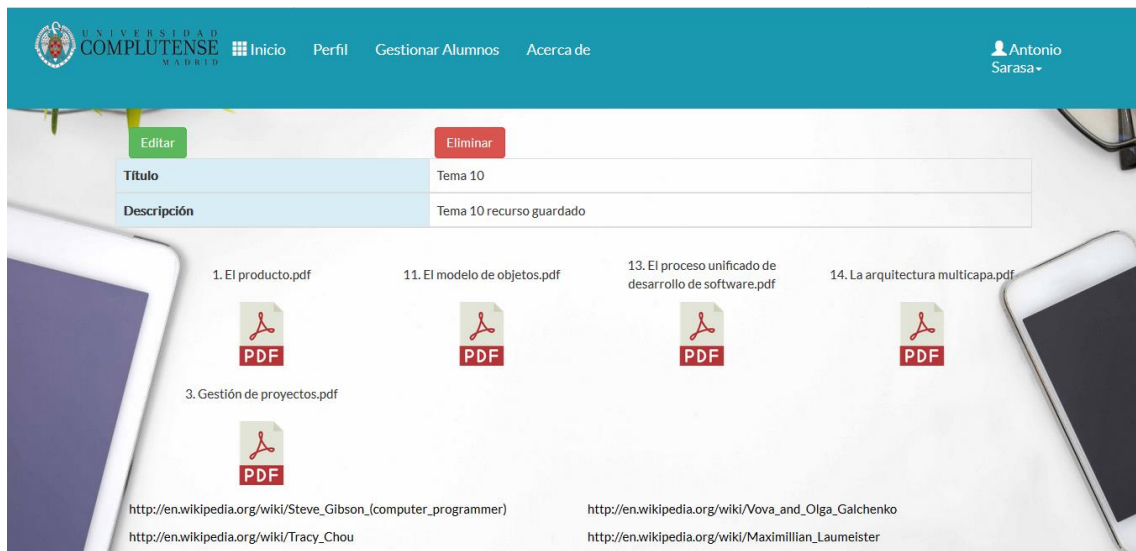


Figura 28: Previsualización de un recurso.

m) Generar consultas.

Este botón lanza las consultas predefinidas, mediante el lenguaje SPARQL, obteniendo resultados sobre la web semántica. A continuación, se muestra una de las consultas realizadas, Figura 29, donde se ve cómo se almacenan todas estas consultas en un array.

```
$query = array(
  "Diseño_UML" => "PREFIX dct:<http://purl.org/dc/terms/>
  PREFIX dbo:<http://dbpedia.org/ontology/>
  PREFIX foaf:<http://xmlns.com/foaf/0.1/>
  PREFIX dbr:<http://dbpedia.org/resource/>
  PREFIX dbc:<http://dbpedia.org/resource/Category:>

  SELECT ?link WHERE{
    ?engineer dct:subject dbc:Unified_Modeling_Language.
    ?engineer foaf:isPrimaryTopicOf ?link
  }",
  "Gestion_de_riesgos_e_IEEE" => "PREFIX dct:<http://purl.org/dc/terms/>
  PREFIX dbo:<http://dbpedia.org/ontology/>
```

Figura 29: Consulta SPARQL.

El componente ARC2 permite el funcionamiento de SPARQL, dicho componente permite tratar los *endpoints* [26] que se utilizarán en las consultas. Además, se analizan y almacenan individualmente los resultados mediante un *foreach* [27] sobre el array que contiene todas las consultas.

Los resultados se almacenan en la base de datos en la tabla “Consultassparql”, la cual se vacía antes de volver a rellenar con los nuevos resultados. Esto se puede apreciar en la Figura 30.

```
$dbpconfig = array(
  "remote_store_endpoint" => "http://dbpedia.org/sparql",
);
$store = ARC2::getRemoteStore($dbpconfig);
if ($errs = $store->getErrors()) {
  echo "<h1>getRemoteSotre error<h1>" ;
}
/* execute the query */
eliminarConsultas();
foreach ($query as $clave => $valor) {
  $links = '';
  $rows = $store->query($valor, 'rows');
  if ($errs = $store->getErrors()) {
    echo "Query errors" ;
    print_r($errs);
  }
  foreach( $rows as $row ) {
    if($links == ""){
      $links = $row['link'] ;
    }else{
      $links = $links . " , " . $row['link'] ;
    }
  }
  $consulta = guardarConsulta($clave, $links);
  if (!$consulta) {
    break;
  }
}
```

Figura 30: Generar consulta.

8. Evaluación.

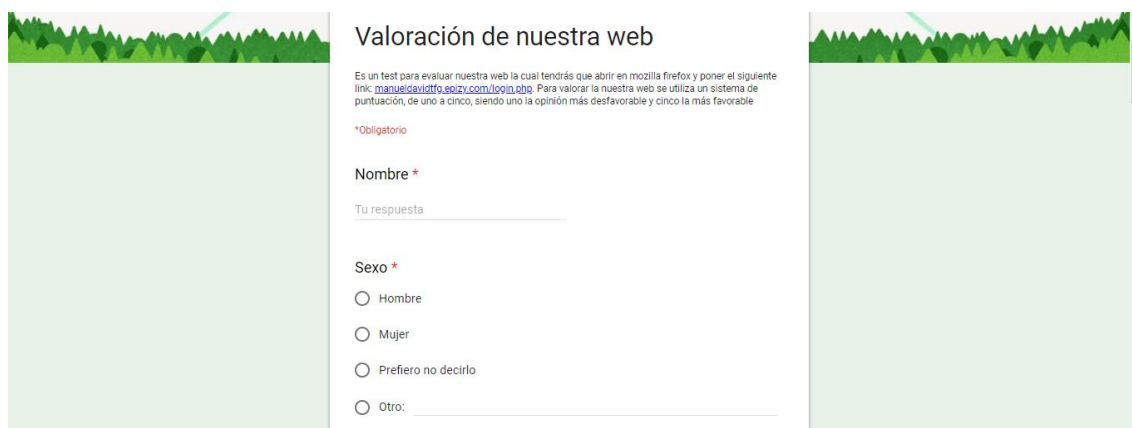
En este capítulo se detalla la encuesta realizada a diferentes miembros de la comunidad educativa, participando en ella tanto estudiantes como profesores. La primera parte aborda una descripción de la encuesta realizada para la evaluación del proyecto, seguido de una segunda parte, donde se analizan los resultados obtenidos en dicha encuesta.

8.1. Guion de pruebas.

Se ha realizado, mediante la ayuda de los formularios de Google, una encuesta para evaluar la usabilidad del proyecto.

Para ello, en la encuesta, se realizan catorce preguntas donde el usuario debe realizar unas determinadas tareas, posteriormente, evaluará del 1 al 5, dichas preguntas, siendo 1 la opción más desfavorable y la 5 la más favorable, o bien indicando cuanto esfuerzo ha supuesto realizar la acción, señalando mucho, sin más o poco es intuitivo, véase en la Figura 31.

Una vez finalizada la encuesta, el usuario procede a enviar los resultados.



The image shows a Google Form titled "Valoración de nuestra web". The form includes the following elements:

- Title:** Valoración de nuestra web
- Introduction:** Es un test para evaluar nuestra web la cual tendrás que abrir en mozilla firefox y poner el siguiente link: manueldavittro.es/zy.com/loqjg.ubq. Para valorar la nuestra web se utiliza un sistema de puntuación, de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable.
- Required Field:** *Obligatorio
- Name:** Nombre *
Tu respuesta
- Sex:** Sexo *
 - Hombre
 - Mujer
 - Prefiero no decirlo
 - Otro: _____

Descárgate algún elemento del Recurso test 3. ¿Cuanto te ha costado?

- Mucho
- Sin más
- Poco, es intuitivo

Cierra sesión e inicia con el rol del profesor. Para ello utiliza el usuario profesortest@ucm.es con contraseña profesortest.

Navega por la web. Valora de uno a cinco la estética utilizada, siendo uno la opinión más desfavorable y cinco la más favorable.

- 1
- 2
- 3
- 4
- 5

Crea una cuenta con el rol de alumno y asóciate al grupo del profesortest. Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable, como de intuitivo ha sido realizar la acción anteriormente mencionada.

- 1
- 2
- 3
- 4
- 5

A continuación te activarás la cuenta desde el profesortest. Para ello utiliza la cuenta con el usuario profesortest@ucm.es con contraseña profesortest. Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable.

- 1
- 2
- 3
- 4
- 5

A continuación cerraremos la sesión de profesortest. ¿Cuanto te ha costado?

- Mucho
- Sin más
- Poco, es intuitivo

Inicia sesión con el usuario que has creado anteriormente y que acabas de activar con la cuenta de profesortest@ucm.es

Visualiza los recursos existentes. Valora de uno a cinco el diseño de estos, siendo uno la opinión más desfavorable y cinco la más favorable

- 1
- 2
- 3
- 4
- 5

Modifica tu perfil (Nombre completo, descripción...). Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable

- 1
- 2
- 3
- 4
- 5

Investiga como poder ver los recursos en función del profesor que lo haya creado. Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable

- 1
- 2
- 3
- 4
- 5

Sube un archivo y posteriormente crea un recurso con el archivo subido previamente. Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable.

- 1
- 2
- 3
- 4
- 5

Crea otro recurso, pero en vez de publicarlo, guárdalo. Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable.

- 1
- 2
- 3
- 4
- 5

Carga el recurso anteriormente guardado y añádele algún link y publícalo. ¿Ha sido intuitivo? Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable.

- 1
- 2
- 3
- 4
- 5

Modifica tu perfil y examina tus alumnos. Elimina el alumno que has creado al principio y examina tus recursos, ahora aparecen tus recursos creados anteriormente. Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable.

- 1
- 2
- 3
- 4
- 5

Notificar uso inadecuado - [Condiciones del servicio](#)' and the 'Google Formularios' logo."/>

Utiliza el buscador para encontrar tus recursos creados anteriormente y elimínalos. Valora cuan intuitivo ha sido.

1

2

3

4

5

ENVIAR

Nunca envíes contraseñas a través de Formularios de Google.

Este formulario se creó en Universidad Complutense de Madrid. [Notificar uso inadecuado](#) - [Condiciones del servicio](#)

Google Formularios

Figura 31: Encuesta.

8.2. Resultados de la evaluación.

Se somete a evaluación el proyecto de fin de grado para obtener un *feedback* [28] y comparar los resultados con los objetivos marcados.

Con esta evaluación se obtuvo, sobre todo, retroalimentación acerca de cuan fácil e intuitiva es la interfaz para el usuario y como de difícil es la consecución de las actividades que se pueden realizar en el proyecto.

Para ello se realiza una evaluación con ayuda de los formularios de Google.

En primer lugar, se seleccionó a ocho personas para que realizasen un guion de pruebas en el que se les iba pidiendo que fuesen ejecutando todas las acciones que se pueden efectuar en el proyecto web.

De estas ocho personas, una tenía el puesto de profesor en la universidad complutense de Madrid, y las otras siete personas restantes eran usuarios normales, que estaban un poco familiarizados con la tecnología.

De esas siete personas, dos eran mujeres y las cinco personas restantes eran hombres. En este guion de pruebas se le preguntaba primero su nombre y su género y posteriormente ya se comenzaban con las cuestiones sobre el proyecto.

- a) La primera pregunta se corresponde con el sexo del usuario que va a realizar el formulario. Por medio del gráfico, se observan que el 75% son hombres y el 25% mujeres. Como se muestra en la Figura 32.

Sexo

8 respuestas

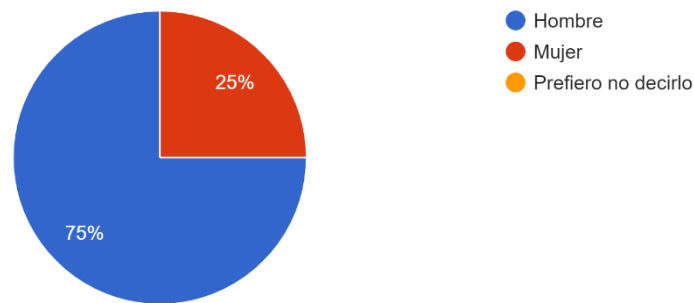


Figura 32: Resultado de la encuesta, gráfico de sexo.

- b) La segunda pregunta corresponde con la creación de una nueva cuenta con el rol de alumno. Como se puede apreciar en el gráfico el 75% ha opinado que ha sido muy intuitivo realizar la acción anteriormente mencionada y un 25% que ha sido bastante intuitivo, Figura 33.

Crea una cuenta con el rol de alumno y asóciate al grupo del profesortest. Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable, como de intuitivo ha sido realizar la acción anteriormente mencionada.



8 respuestas

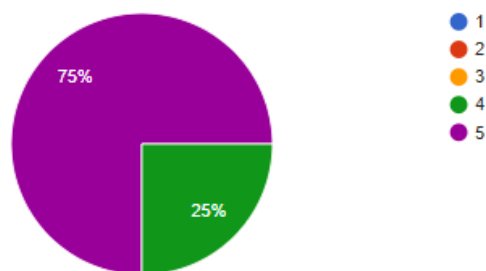


Figura 33: Resultado de la encuesta, gráfico de crear cuenta.

- c) En la tercera pregunta, Figura 34, el usuario deberá activar la cuenta anteriormente creada gracias a una cuenta con el rol de profesor que se facilita. Como pasaba en el punto anterior, se obtienen los mismos resultados, un 75% de respuestas han sido muy intuitivas y el 25% han sido bastante intuitivas.

A continuación te activarás la cuenta desde el profesortest. Para ello utiliza la cuenta con el usuario profesortest@ucm.es con contraseña profesortest. Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable.

8 respuestas

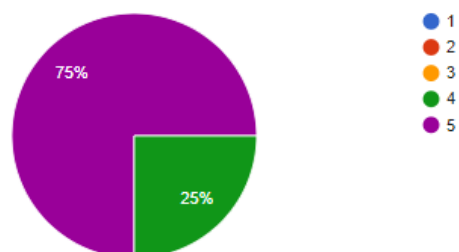


Figura 34 Resultado de la encuesta, gráfico de activar cuenta.

d) La siguiente pregunta hace referencia al cierre de sesión.

El 87.5 de los usuarios ha respondido que es intuitivo y que les ha costado poco y el 12.5% restante responde la opción de sin más, véase en la Figura 35.

A continuación cerraremos la sesión de profesortest. ¿Cuanto te ha costado?

8 respuestas

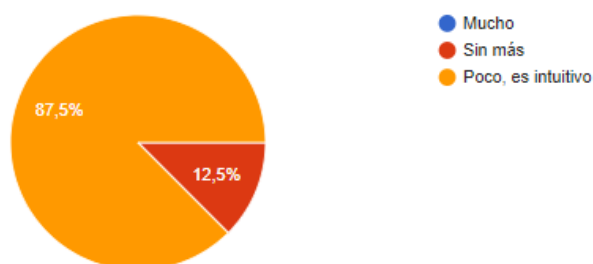


Figura 35: Resultado de la encuesta, gráfico de cierre de sesión.

e) En la quinta pregunta se pide al usuario que inicie sesión con el usuario creado y activado anteriormente para que visualice los recursos que están creado y de su opinión.

El 25% opina que el diseño es muy bueno, el 62.5% opina que es bastante bueno y el 12.5% opina, sin más, estos resultados aparecen en la Figura 36.

Visualiza los recursos existentes. Valora de uno a cinco el diseño de estos, siendo uno la opinión más desfavorable y cinco la más favorable

8 respuestas

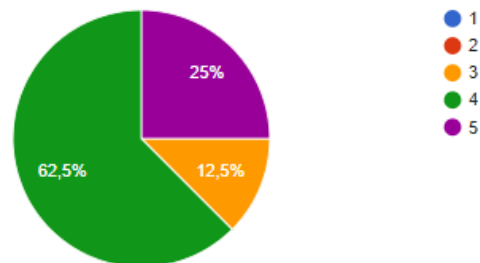


Figura 36: Resultado de la encuesta, gráfico de estilo recurso.

- f) La siguiente pregunta pide al usuario que modifique su perfil y puntúe que le ha parecido, Figura 37.

Donde el 37.5% opina que el diseño es muy bueno, el 37.5% opina que es bastante bueno y el 25% opina que sin más.

Modifica tu perfil (Nombre completo, descripción...). Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable

8 respuestas

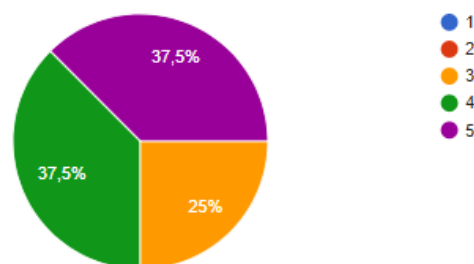


Figura 37: Resultado de la encuesta, gráfico de perfil.

- g) Para la séptima pregunta se pide al usuario que interactúe en la web para encontrar donde poder ver los recursos ordenados por profesores.

El 50% opina que no les ha costado nada, el 25% opina que casi nada y el 25% opina que sin más. Se muestran los resultados en la Figura 38.

Investiga como poder ver los recursos en funcion del profesor que lo haya creado. Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable

8 respuestas

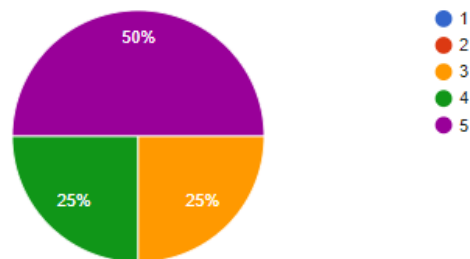


Figura 38: Resultado de la encuesta, gráfico de recursos.

h) En la siguiente pregunta, Figura 39, se pide al usuario que intente descargar un archivo de uno de los recursos en concreto.

El 75% opina que les ha costado poco, que es intuitivo y el 25% opina que sin más.

Descárgate algún elemento del Recurso test 3. ¿Cuanto te ha costado?

8 respuestas

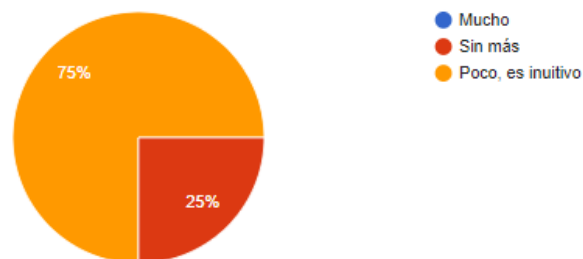


Figura 39: Resultado de la encuesta, gráfico de descarga.

i) Para la novena pregunta, se le demanda al usuario que investigue por la web y se vaya fijando en la estética de esta.

En la Figura 40, se muestran los siguientes resultados. Donde el 50% opina que tiene un muy buen diseño y el otro 50% opina que es buen diseño.

Navega por la web. Valora de uno a cinco la estética utilizada, siendo uno la opinión más desfavorable y cinco la más favorable.

8 respuestas

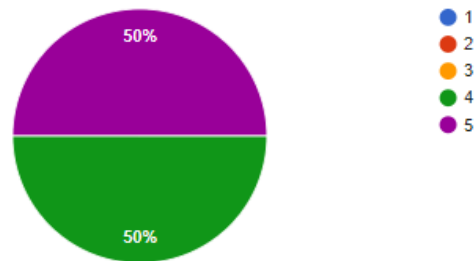


Figura 40: Resultado de la encuesta, gráfico web.

- j) En la décima pregunta, el usuario debe subir un archivo a la web y posteriormente crear un recurso con el archivo subido anteriormente.

El 62.5% opina que no les ha costado nada y el otro 37.5% opina que no les ha costado casi nada. Véase en la Figura 41.

Sube un archivo y posteriormente crea un recurso con el archivo subido previamente. Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable.

8 respuestas

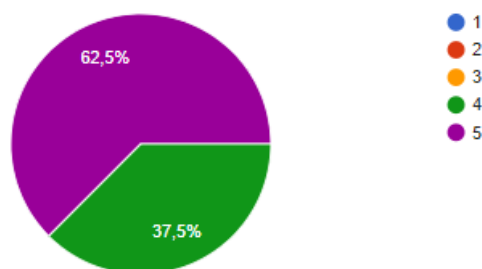


Figura 41: Resultado de la encuesta, gráfico de crear recurso.

- k) En la siguiente pregunta se pide al usuario que cree otro recurso, pero en vez de publicarlo, lo guarde.

La Figura 42 arroja los siguientes resultados, el 37.5% opina que no les ha costado nada, el 25% opina que no les ha costado casi nada, el 25% opina que sin más y el 12.5% que les ha costado un poco.

Crea otro recurso, pero en vez de publicarlo, guárdalo. Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable.

8 respuestas

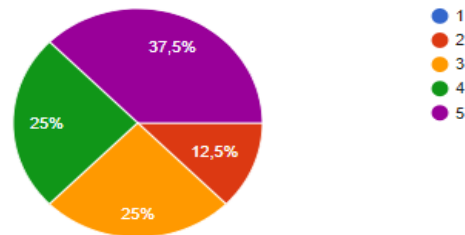


Figura 42: Resultado de la encuesta, gráfico de guardar recurso.

- 1) A continuación, se pide al usuario que cargue el recurso anteriormente guardado, y lo modifique añadiendo algún link para hacerlo más completo.

El 25% opina que no les ha costado nada, el 50% opina que no les ha costado casi nada, el 12.5% opina que sin más y el 12.5% restante que les ha costado un poco. Estos resultados se encuentran en la Figura 43.

Carga el recurso anteriormente guardado y añádele algún link y publícalo. ¿Ha sido intuitivo? Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable.

8 respuestas

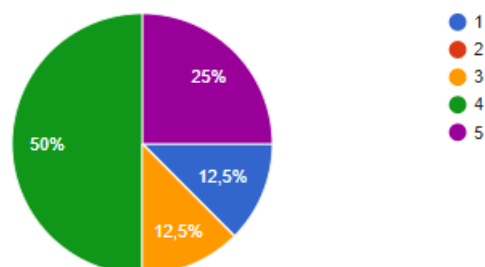


Figura 43: Resultado de la encuesta, gráfico de modificar recurso guardado.

m) En la siguiente pregunta, se pide al usuario que examine su perfil para ver que los recursos que ha creado anteriormente aparecen en la pestaña “Mis recursos”, y el alumno que ha creado antes, ahora sale en la pestaña “Mis alumnos”.

Después de realizar esa comprobación, se le dice que borre al alumno que se ha creado al principio del formulario.

El 62.5% opina que no les ha costado nada y el 37.5 opina que no les ha costado casi nada, los resultados explicados corresponden con la Figura 44.

Modifica tu perfil y examina tus alumnos. Elimina el alumno que has creado al principio y examina tus recursos, ahora aparecen tus recursos creados anteriormente. Valora de uno a cinco, siendo uno la opinión más desfavorable y cinco la más favorable.

8 respuestas

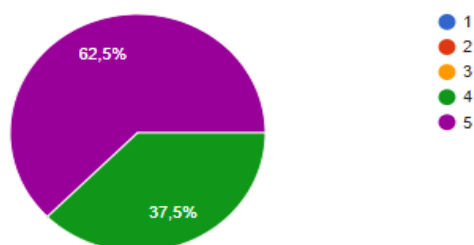


Figura 44: Resultado de la encuesta, gráfico borrar alumno desde perfil.

n) Por último, Figura 45, el usuario debe utilizar el buscador para encontrar los recursos creados anteriormente y eliminarlos.

El 50% opina que no les ha costado nada, el 25% opina que no les ha costado casi nada y el 25% restante opina que sin más.

Utiliza el buscador para encontrar tus recursos creados anteriormente y elimínalos. Valora cuan intuitivo ha sido.

8 respuestas

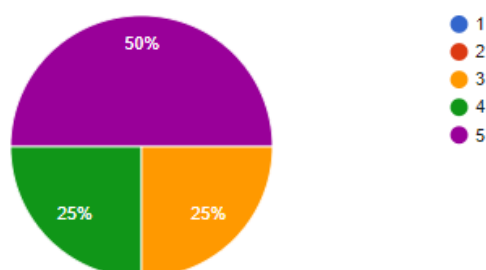


Figura 45: Resultado de la encuesta, gráfico eliminar recursos.

9. Conclusiones y trabajo futuro.

En este capítulo se van a presentar las conclusiones que se han obtenido tras la realización del proyecto de fin de grado sobre desarrollo de recursos educativos para ingeniería del software utilizando repositorios de Open Data y Linked Data.

9.1 Conclusiones del trabajo.

Se ha realizado una herramienta para la gestión y explotación de material relacionado con la asignatura de ingeniería del software.

Desarrollando un directorio de recursos, que varía en función del rol del usuario, siendo el administrador capaz de eliminar cualquier recurso, el profesor capaz de borrar o editar sus propios recursos y, por último, el alumno que simplemente puede visualizarlos. Además, se ha implementado que todos los roles sean capaces de descargar los archivos del recurso.

Realizando un perfil, que puede ser modificado, para cada usuario, con un perfil de profesor algo más completo para que pueda visualizar sus propios alumnos y recursos.

Se ha realizado una pestaña para que el profesor pueda subir sus propios archivos a la hora de crear un recurso.

Otra pestaña que permite la creación de los recursos, pudiendo añadirle cualquier archivo subido a la web, además de una serie de links que se recuperan mediante consultas SPARQL.

Para el inicio y registro de sesión, se realiza una encriptación de las contraseñas para respetar la privacidad del usuario.

Se crea una herramienta capaz de unificar todo el contenido de la asignatura de Ingeniería del Software, limitada en las consultas SPARQL recuperando solamente enlaces de Wikipedia.

9.2 Trabajos de futuro.

Las principales líneas de trabajo futuro para continuar con este proyecto son:

- Extender esta aplicación a otras asignatura: enriqueciendo los contenidos ofrecidos.
- Añadir algún modelo de valoración sobre los recursos: proporcionando un *feedback* o retroalimentación interna, con el que el alumno pueda ver que archivo

es el mejor o más completo, o en el caso del profesor, para saber qué tipo de material es el que le suele gustar al alumno. Y de cara a la visualización de contenido se podría mostrar el contenido en función de los recursos y asignaturas más visitadas o mejor valoradas por un alumno.

- Añadir mejoras en la interfaz: tales como la elección del idioma en el que se muestra el contenido o la posibilidad de mostrar contenido por orden alfabético o por fecha de creación.
- Hacer de la web una aplicación móvil.
- Crear una validación de correo: permitiendo solamente el registro de usuarios de la Universidad Complutense de Madrid.
- Crear un foro interno donde los usuarios y profesores puedan preguntar y responder dudas fácilmente y diseñar un sistema de comentarios para cada recurso creado.

9.3 Aportaciones de David Limón Miralles.

- Realización de bocetos diseñados a papel.
- Diseño y creación de la especificación de los casos de uso.
- Posteriormente, se desarrollaron sus diagramas para favorecer la comprensión de estos de manera visual.
- Diseño y estructura de la base de datos donde se iba a almacenar toda la información de la herramienta.
- Se desarrolla el diseño de la interfaz, donde se creó la estructura de la web, implementando una funcionalidad básica .
- Creación de un mecanismo de registro y de inicio de sesión en la web, desarrollando la conexión con la base de datos.
- Realización de consultas para crear la lógica de gestión de cuentas con la activación por parte del administrador y profesores respectivamente.
- Se realizó la creación de una pestaña para la visualización de las cuentas pendientes de activación, implementando las consultas determinadas para su funcionamiento.

- Desarrollo de la lógica de la creación de recursos.
- Desarrollo e implementación del diseño y estructura de gestión de archivos pertenecientes a los recursos.
- Implementación de una función capaz de recorrer la estructura de directorios. Esta función es capaz de recorrer cualquier ruta pasada por parámetro y comparar la extensión del archivo del directorio que le corresponde para determinar la acción a realizar.
- Desarrollo e implementación de la funcionalidad y estructura de la creación, previsualización o el guardado de recursos.
- Investigación y documentación sobre consultas SPARQL.
- Implementación de consultas SPARQL.
- Diseño y estructura de la memoria.
- Completar la memoria.

9.4 Aportaciones de Manuel Martín Canora.

- Realización de casos de uso.
- Creación de mockups de los bocetos a papel.
- Diseño de las tablas de la base de datos.
- Decisión del uso de servidor Xampp.
- Creación de una primera interfaz utilizable.
- Investigación, documentación y posterior creación de consultas SPARQL.
- Implementación de la interfaz definitiva.
- Creación de la encuesta a través de Formularios de Google.

- Subida del proyecto a un host facilitando la interacción con las encuestas realizadas.
- Examinar el proyecto a fondo solucionando errores de funcionamiento.
- Distribución de la encuesta realizada.
- Análisis de los resultados arrojados en la encuesta.
- Completar la memoria.

9. Conclusions and future work

In this chapter we will present the conclusions that have been obtained after the completion of the end of grade project on development of educational resources for software engineering using Open Data and Linked Data repositories.

9.1 Conclusions of the work.

A tool has been created for the management and exploitation of material related to the software engineering subject.

Developing a directory of resources, which varies depending on the role of the user, being the administrator capable of eliminating any resource, the teacher capable of deleting or editing their own resources and, finally, the student who can simply visualize them. In addition, it has been implemented that all the roles are capable of downloading the resource files.

Making a profile, which can be modified, for each user, with a somewhat more complete teacher profile so you can visualize your own students and resources.

A tab has been made so that the teacher can upload their own files when creating a resource.

Another tab that allows the creation of resources, being able to add any file uploaded to the web, as well as a series of links that are retrieved through SPARQL queries.

For the initiation and registration of the session, an encryption of the passwords is carried out to respect the privacy of the user.

A tool is created that can unify all the content of the Software Engineering subject, limited in the SPARQL queries, retrieving only Wikipedia links.

9.2 Future Jobs.

The main lines of future work to continue with this project are:

- Extend this application to other subject: so that the contents offered are enriched.
- Add some valuation model about the resources: providing feedback or internal feedback, with which the student can see which file is the best or most complete, or in the case of the teacher, to know what type of material is the one that usually like the student.

And in the face of the visualization of content the content could be shown according to the resources and subjects most visited or best valued by a student.

- Add improvements in the interface: such as the choice of the language in which the content is displayed or the possibility of displaying content in alphabetical order or by date of creation.
- Make the web a mobile application.
- Create a mail validation: allowing only the registration of users of the Complutense University of Madrid.
- Create an internal forum where users and teachers can easily ask and answer questions and design a comment system for each resource created.

9.3 Contributions of David Limón Miralles.

- Making sketches designed to paper.
- Design and creation of the specification of use cases.
- Subsequently, their diagrams were developed to favor the understanding of these visually.
- Design and structure of the database where all the information of the tool was to be stored.
- The design of the interface is developed, where the structure of the web was created, implementing a basic functionality.
- Creation of a registration and login mechanism on the web, developing the connection with the database.
- Conducting queries to create the logic of account management with activation by the administrator and teachers respectively.
- Creation of a tab for the display of pending activation accounts, implementing the queries determined for its operation.
- Development of the logic of the creation of resources.
- Development and implementation of the design and management structure of files belonging to the resources.
- Implementation of a function capable of traversing the directory structure. This function is capable of traversing any route passed by parameter and comparing

the extension of the file of the corresponding directory to determine the action to be performed.

- Development and implementation of the functionality and structure of the creation, preview or storage of resources.
- Research and documentation on SPARQL queries.
- Implementation of SPARQL queries.
- Design and structure of memory.
- Complete the memory.

9.4 Contributions of Manuel Martin Canora.

- Realization of use cases.
- Creation of mockups of paper sketches.
- Design of the tables of the database.
- Decision to use Xampp server.
- Creation of a first usable interface.
- Research, documentation and subsequent creation of SPARQL queries.
- Implementation of the definitive interface.
- Creation of the survey through Google Forms.
- Upload of the project to a host facilitating the interaction with the surveys carried out.
- Examine the project thoroughly, solving operational errors.
- Distribution of the survey carried out.
- Analysis of the results of the survey.
- Complete the memory.

BIBLIOGRAFÍA

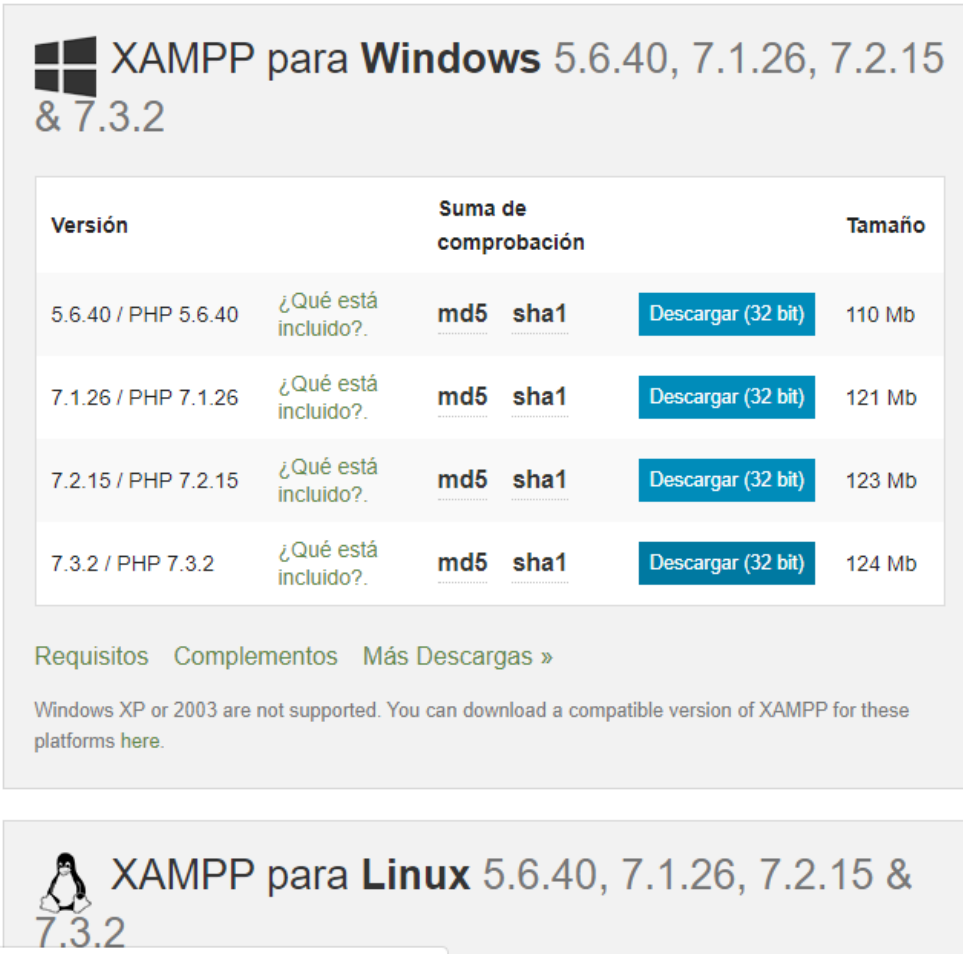
- [1] <<Campus virtual>>. [En línea]. Disponible en: <https://www.ucm.es/campusvirtual>
- [2] <<Open data>>. [En línea]. Disponible en: https://es.wikipedia.org/wiki/Datos_abiertos
- [3] <<Linked data>>. [En línea]. Disponible en: https://es.wikipedia.org/wiki/Datos_enlazados
- [4] << SPARQL >>. [En línea]. Disponible en: <https://www.w3.org/TR/rdf-sparql-query/>
- [5] <<Moodle>>. [En línea]. Disponible en: <https://moodle.org/?lang=es>
- [6] << Google classroom >>. [En línea]. Disponible en: <https://classroom.google.com/u/0/h?hl=es>
- [7] << Bootstrap >>. [En línea]. Disponible en: <https://getbootstrap.com/>
- [8] <<Framework>>. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/Framework>
- [9] <<Blackboard>>. [En línea]. Disponible en: <https://ush.blackboard.com/>
- [10] << Emodo >>. [En línea]. Disponible en: <https://www.edmodo.com/?language=es>
- [11] << Sublime text >>. [En línea]. Disponible en: <https://www.sublimetext.com/>
- [12] <<Xampp>>. [En línea]. Disponible en: <https://www.apachefriends.org/es/index.html>
- [13] << MySQL >>. [En línea]. Disponible en: <https://www.mysql.com/>
- [14] << ARC2 >>. [En línea]. Disponible en: <https://github.com/semsol/arc2/blob/master/doc/SPARQL-support.md>
- [15] << RDF >>. [En línea]. Disponible en: <https://www.w3.org/RDF/>
- [16] <<JavaScript>>. [En línea]. Disponible en: <https://www.w3schools.com/js/>
- [17] << Infinity Free >>. [En línea]. Disponible en: <https://infinityfree.net/>
- [18] << Modelio Open Source >>. [En línea]. Disponible en: <https://www.modelio.org/>
- [19] <<PHP>>. [En línea]. Disponible en: <https://php.net/manual/es/intro-what-is.php>
- [20] << Apache >>. [En línea]. Disponible en: <https://httpd.apache.org/>
- [21] <<Principios de usabilidad de Nielsen>>. [En línea]. Disponible en: https://es.wikipedia.org/wiki/Heur%C3%ADsticas_de_Nielsen
- [22] <<Combobox>>. [En línea]. Disponible en: https://es.wikipedia.org/wiki/Cuadro_combinado
- [23] <<String>>. [En línea]. Disponible en: <https://www.php.net/manual/es/book.strings.php>
- [24] <<Iframe>>. [En línea]. Disponible en: https://www.w3schools.com/tags/tag_iframe.asp
- [25] <<URL>>. [En línea]. Disponible en: https://es.wikipedia.org/wiki/Localizador_de_recursos_uniforme
- [26] <<Endpoints>>. [En línea]. Disponible en: <https://en.wikipedia.org/wiki/Endpoint>
- [27] <<Foreach>>. [En línea]. Disponible en: <https://php.net/manual/es/control-structures.foreach.php>

- [28] <<Feedback>>. [En línea]. Disponible en: <https://definicion.de/feedback/>
- [29] <<HTML>>. [En línea]. Disponible en: <https://www.w3schools.com/html/>
- [30] << Sistema Operativo >>. [En línea]. Disponible en: https://es.wikipedia.org/wiki/Sistema_operativo
- [31] <<Windows>>. [En línea]. Disponible en: <https://www.microsoft.com/es-es/windows>
- [31] <<CSS>>. [En línea]. Disponible en: https://es.wikipedia.org/wiki/Hoja_de_estilos_en_cascada
- [32] <<iOS>>. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/iOS>
- [33] <<Hosting>>. [En línea]. Disponible en: https://es.wikipedia.org/wiki/Alojamiento_web
- [34] <<Plugins>>. [En línea]. Disponible en: [https://es.wikipedia.org/wiki/Complemento_\(inform%C3%A1tica\)](https://es.wikipedia.org/wiki/Complemento_(inform%C3%A1tica))
- [35] <<PhpMyAdmin>>. [En línea]. Disponible en: <https://www.phpmyadmin.net/>
- [36] <<Google drive>>. [En línea]. Disponible en: https://www.google.com/intl/es_ALL/drive/
- [37] <<Formularios google>>. [En línea]. Disponible en: <https://www.google.es/intl/es/forms/about/>
- [38] <<Licencia GNU>>. [En línea]. Disponible en: https://es.wikipedia.org/wiki/GNU_General_Public_License

ANEXOS

ANEXO I. Guía de instalación de la aplicación.

1. Primero se debe descargar la última versión disponible de Xampp, la cual se encuentra en esta dirección: <https://www.apachefriends.org/es/download.html>
En esa dirección, se debe descargar la versión de Xampp correspondiente con cada Sistema Operativo [30], para este proyecto, la versión 7.3.2 para Windows [31], véase en la Figura 46.



XAMPP para Windows 5.6.40, 7.1.26, 7.2.15 & 7.3.2

Versión	Suma de comprobación	Tamaño
5.6.40 / PHP 5.6.40 ¿Qué está incluido?.	md5 sha1	Descargar (32 bit) 110 Mb
7.1.26 / PHP 7.1.26 ¿Qué está incluido?.	md5 sha1	Descargar (32 bit) 121 Mb
7.2.15 / PHP 7.2.15 ¿Qué está incluido?.	md5 sha1	Descargar (32 bit) 123 Mb
7.3.2 / PHP 7.3.2 ¿Qué está incluido?.	md5 sha1	Descargar (32 bit) 124 Mb

[Requisitos](#) [Complementos](#) [Más Descargas](#) »

Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms [here](#).

XAMPP para Linux 5.6.40, 7.1.26, 7.2.15 & 7.3.2

Figura 46: Descarga de Xampp.

2. Tras el primer paso, se procederá a descargar un ejecutable (.exe), el cual se instalará como se muestra en la Figura 47. A continuación, se debe dar a siguiente hasta que le pida la dirección en la que se quiere instalar (Figura 48). Se seleccionará la dirección para instalar y se le dará a siguiente hasta terminar la instalación.

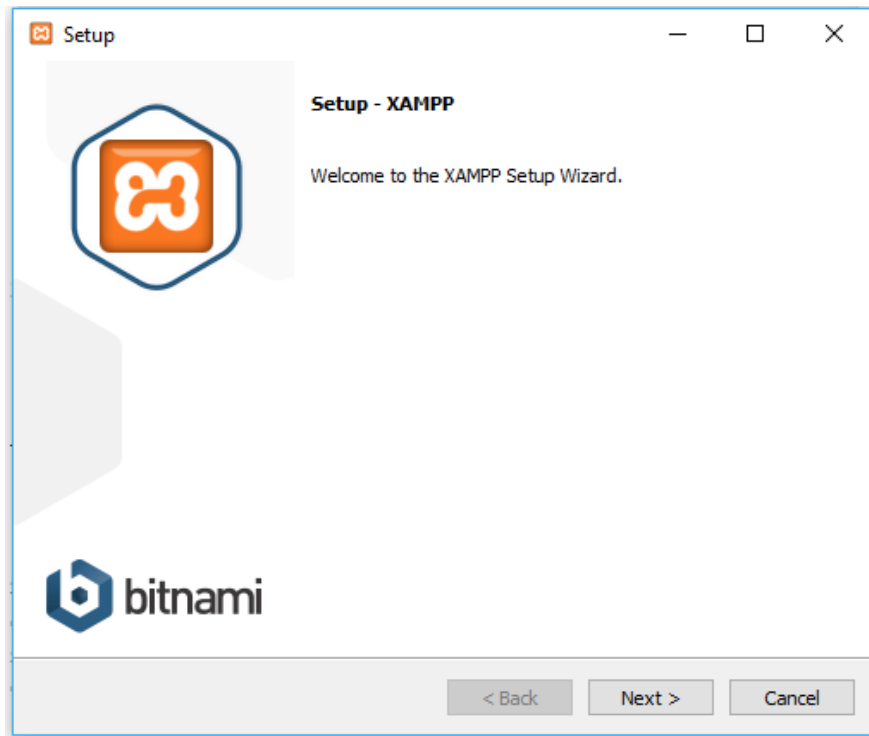


Figura 47: Inicio de instalación de Xampp.

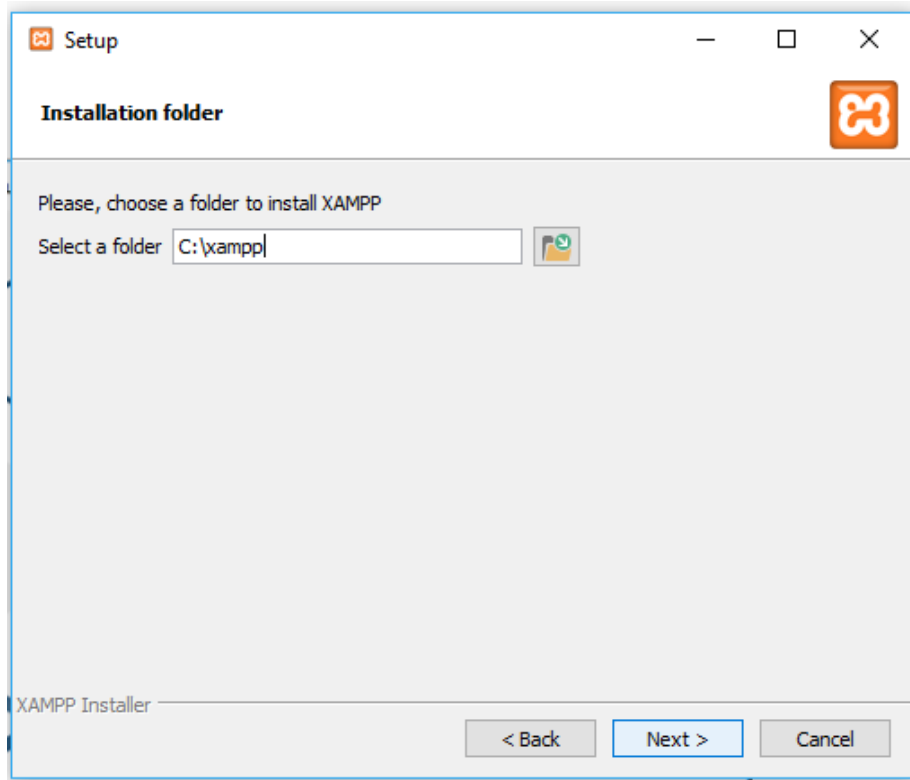


Figura 48: Instalación de Xampp.

3. Se crea una carpeta llamada Xampp en la dirección de instalación que se haya elegido, Figura 48. Dentro de este, tendrá otra carpeta denominada htdocs, la cual será de suma importancia, puesto que, dentro de esta, se tendrá que meter el proyecto TFG. Para

ello, dentro de htdocs se creará una carpeta llamada “TFG” y dentro de esta se introducirá el proyecto (es decir todos los CSS, PHP y las carpetas que se tengan como por ejemplo la carpeta imágenes).

- Después, habrá que iniciar Xampp para poder trabajar en localhost y poder importar la base de datos. Dentro de la carpeta Xampp, se encontrará un ejecutable llamado “xampp-control.exe” como se indicia en la Figura 49. Se hará doble clic en este para iniciarlo y deberá aparecer un panel de control como se muestra en la Figura 50. Una vez se haya iniciado el panel de control, simplemente se debe pulsar *start* en las opciones, Apache y MySQL (las primeras veces pide acceso y habrá que autorizarle) y ya tendrá iniciado el servicio Xampp. Después de iniciarlo, el panel de control debería quedar de la siguiente manera, Figura 51.

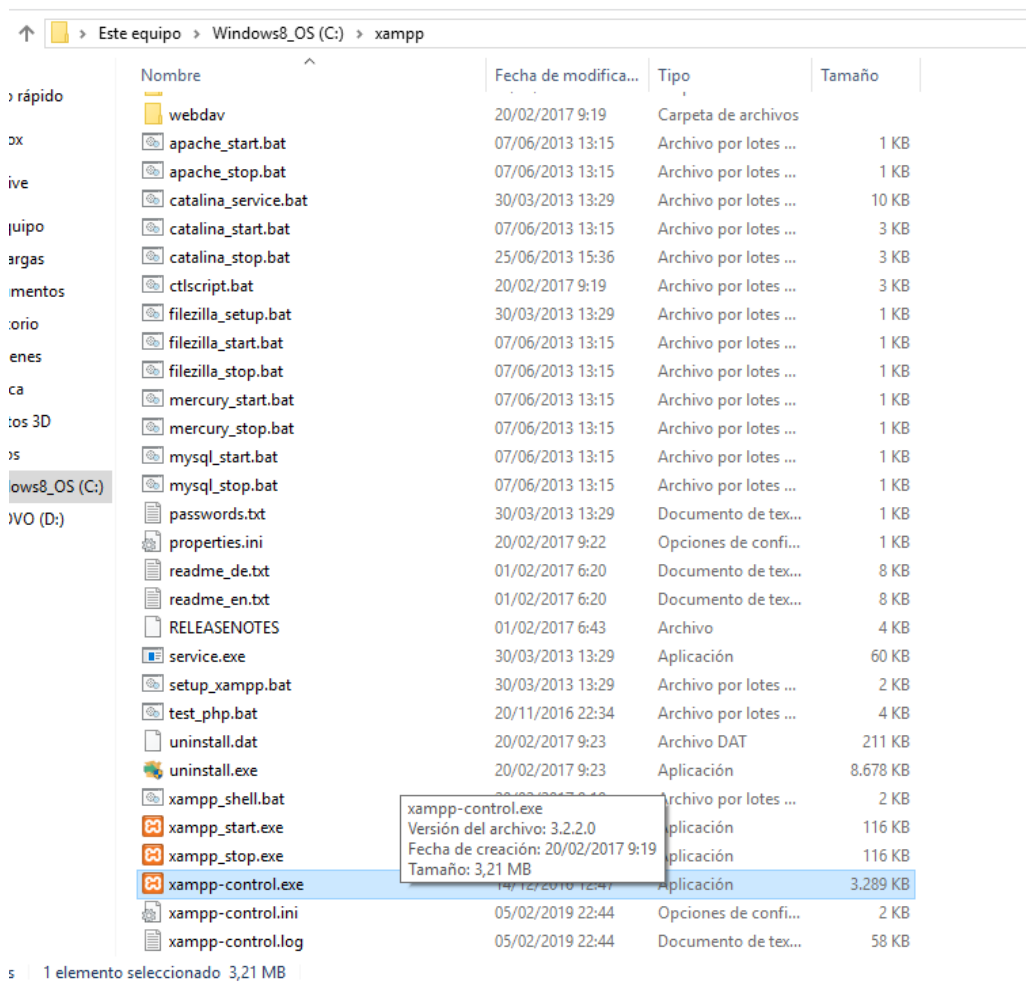


Figura 49: Ejecutable de Xampp.

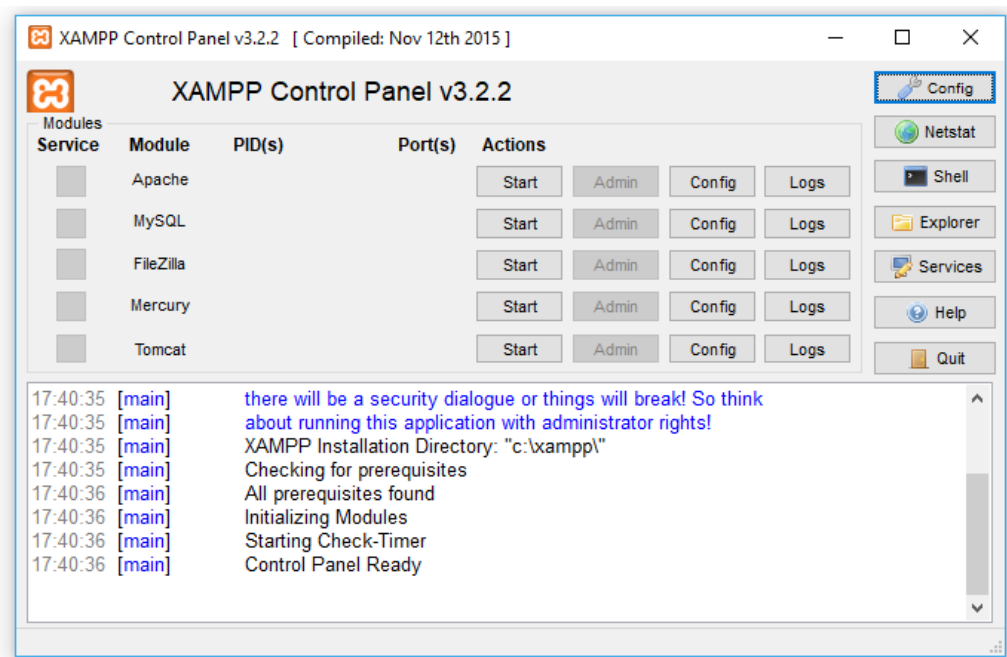


Figura 50: Panel de control de Xampp.

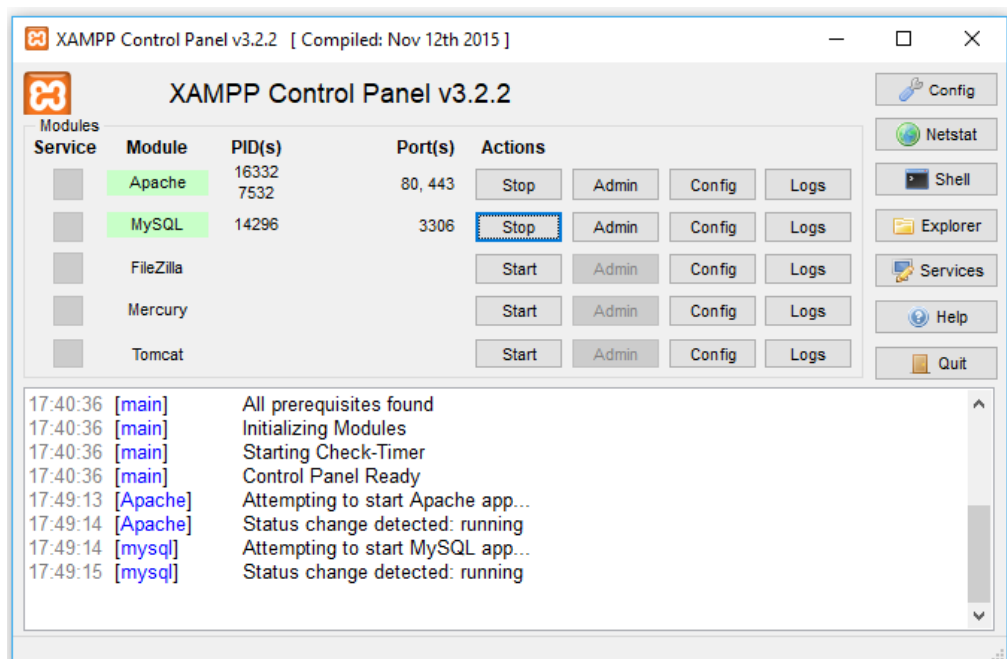


Figura 51: Panel de control de Xampp (Activado).

- Una vez se haya iniciado Xampp, se abrirá el navegador y se deberá buscar localhost, esto se muestra en la Figura 52, y pulsar la tecla *enter* para que se realice la búsqueda. Esto le llevará a la página home de Xampp, en la que se tendrá que dar a phpMyAdmin, situado en la esquina superior derecha como se indica en la Figura 53. Esto le llevará a donde está situada la base de datos, la cual tendremos que importar.

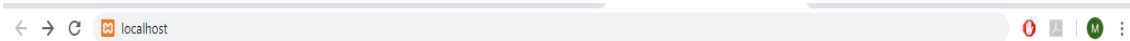


Figura 52: Buscador con localhost.

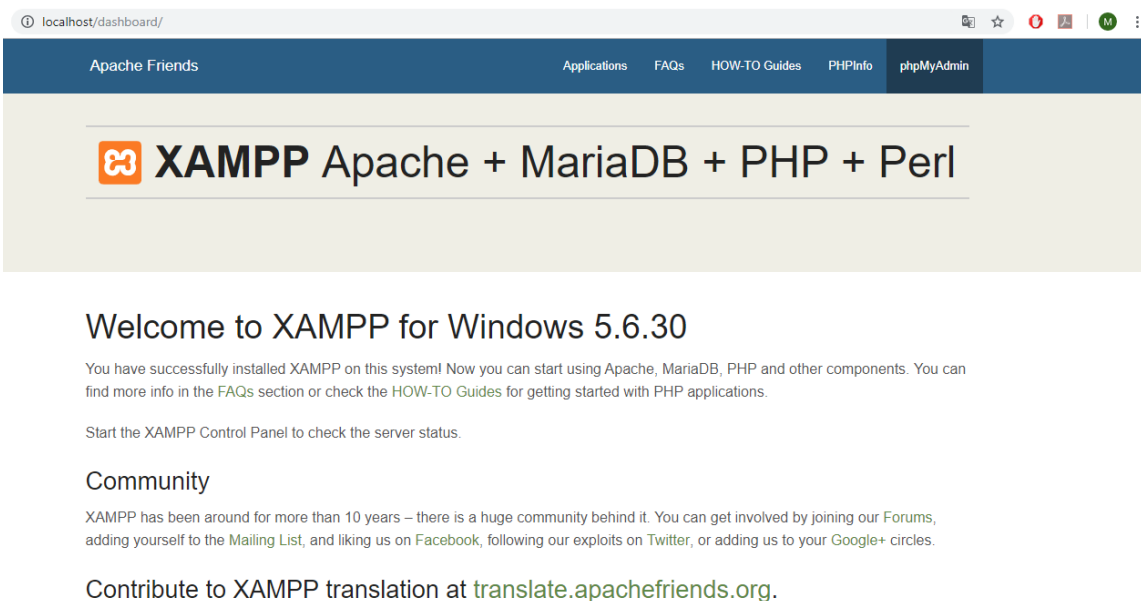


Figura 53: Página de bienvenida de Xampp.

6. Para importar la base de datos se tendrá que crear una base de datos llamada “tfg” dándole arriba a la izquierda, donde pone “Nueva”, Figura 54. Le dará el nombre de “tfg”, se escogerá “utf8_spanish_ci”, Figura 55, y se pulsa en “Crear”. Una vez creada, aparecerá dicho nombre en el menú de la izquierda. Se selecciona y se pulsa importar, que estará situado en el menú del centro arriba. Una vez se haya dado a importar, le dirá que se examine el archivo a subir, que será un .sql.zip, como se muestra en la Figura 56. Una vez se haya seleccionado el archivo, abajo del todo se pulsará continuar. Si todo ha salido de forma exitosa deberían salir las tablas en la base de datos “tfg” y algunos mensajes de éxito como se muestran en la Figura 57.

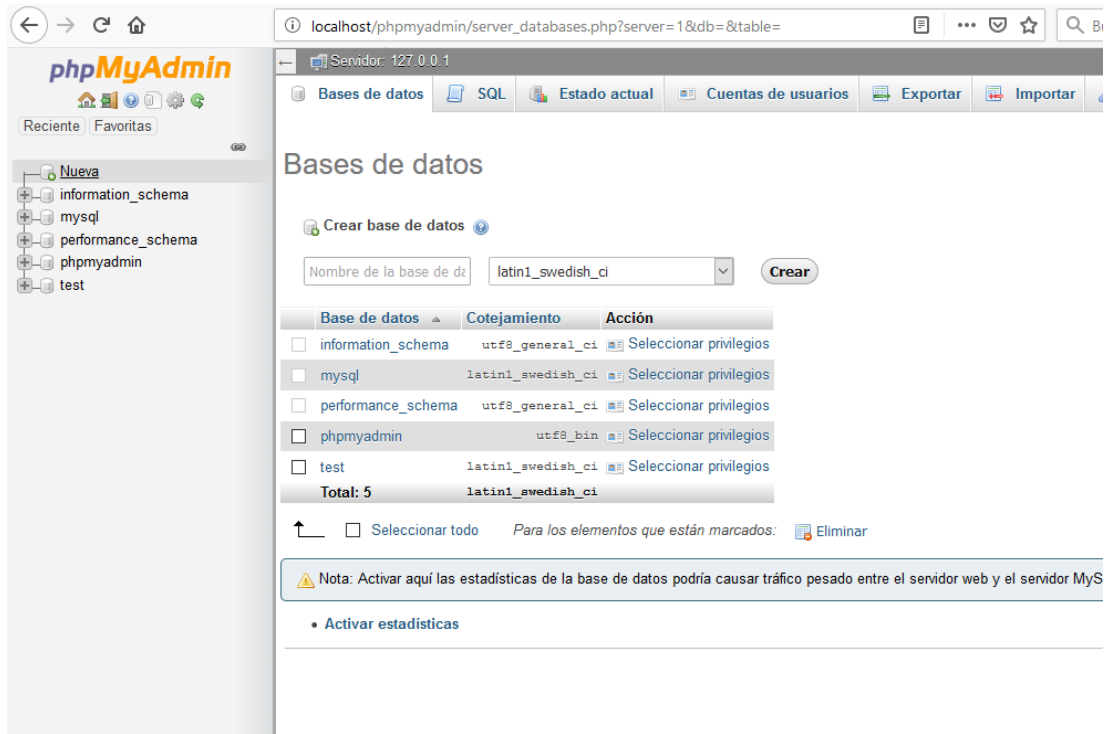


Figura 54: Inicio de Base de datos PHPMyAdmin.



Figura 55: Creación de una Base de datos.

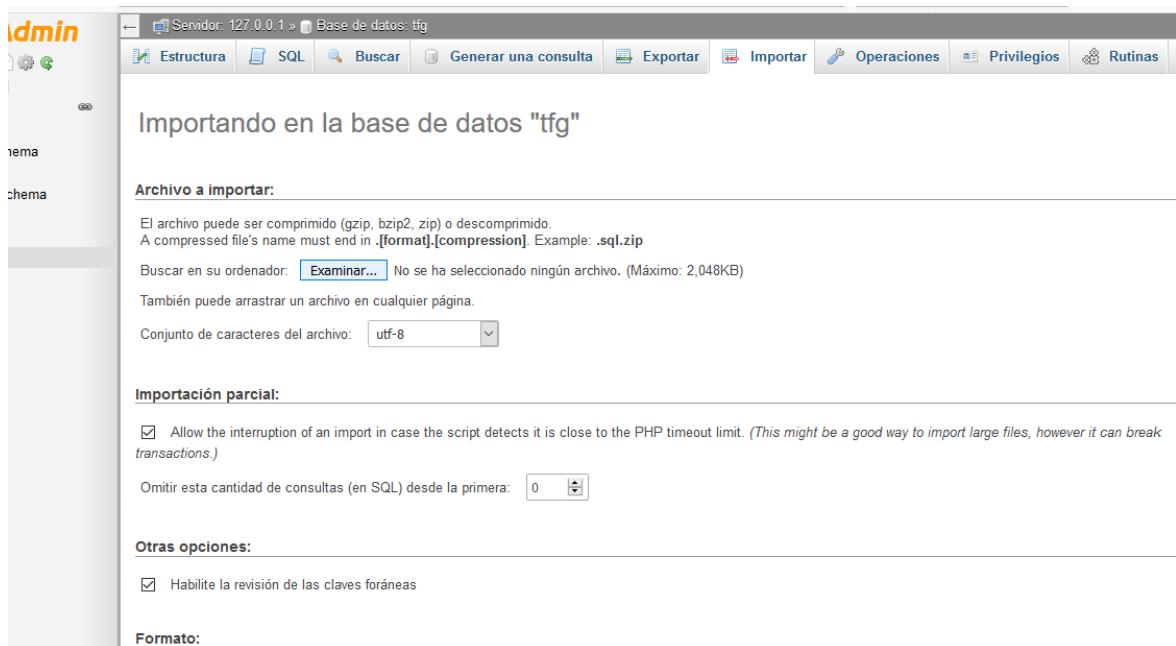


Figura 56: Importar una base de datos.

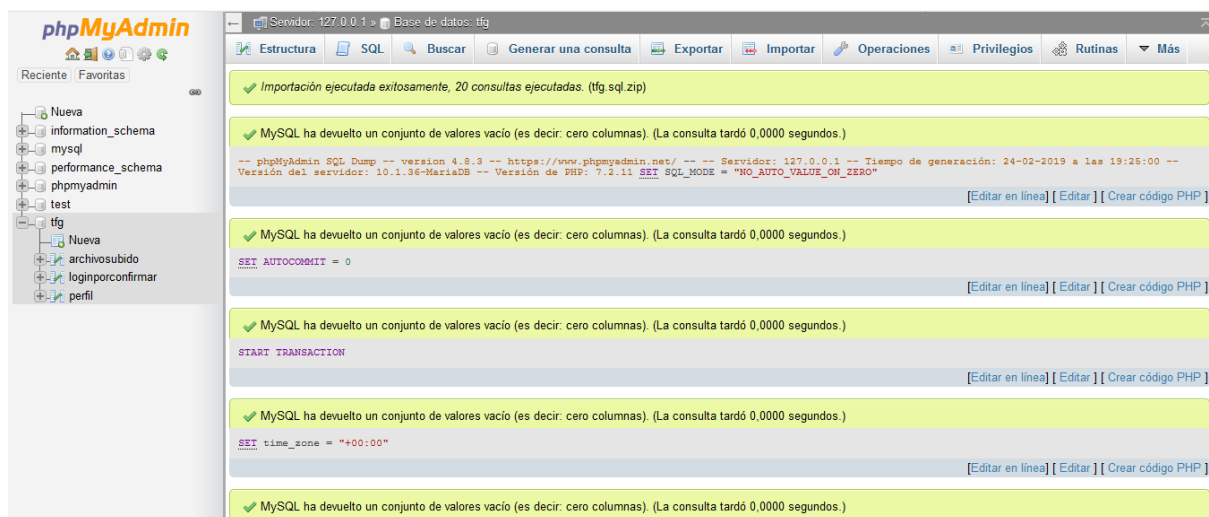


Figura 57: Resultado de la importación.

- Por último, una vez se tenga la base de datos importada y Xampp iniciado, lo único que se debe hacer es abrir el navegador y poner en el buscador superior <http://localhost/TFG/login.php>, esta dirección le llevará al inicio de sesión de la web.

II. Guía de uso de la aplicación.

1. Profesor

Paso 1: Inicio sesión.

Se inicia su propia sesión desde la página de inicio de sesión, véase en la Figura 58, o en caso de que no se disponga una cuenta, se realiza un registro de cuenta, Figura 59, para que posteriormente sea activada por el administrador.

Para realizar el registro de cuenta se necesita un nombre de usuario, un correo con una contraseña, y por último se selecciona el rol como profesor y se indica que grupo quiere.

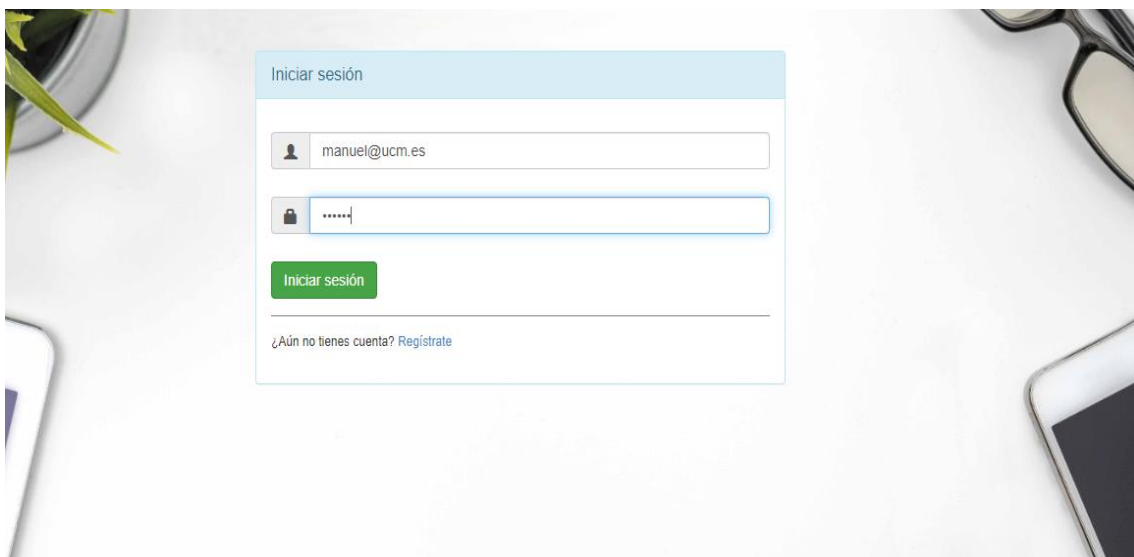


Figura 58: Inicio de sesión desde profesor.

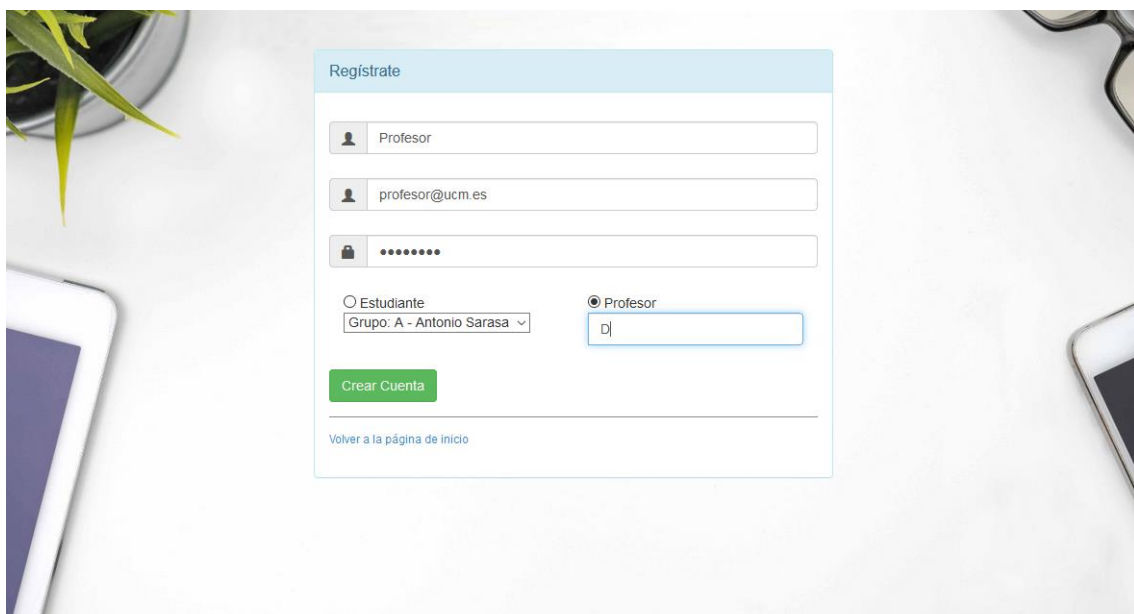


Figura 59: Registro de cuenta desde profesor.

Paso 2: Activar alumnos.

El profesor puede dar de alta o activar las cuentas de sus propios alumnos, es decir, los alumnos que están asignados a su grupo. Para ello el profesor debe ir al menú superior y seleccionar “Gestionar alumnos”. Una vez se ha hecho esto, aparecerán las cuentas pendientes de activación, a las que podrá activar pulsando en el botón con el mismo nombre o, en caso de que no se quiera activar, se pulsará en rechazar, como se aprecia en la Figura 60.



Figura 60: Activar Alumno desde profesor.

Paso 3: Perfil.

Para modificar o ver el perfil del profesor se deberá ir al menú superior y seleccionar “Perfil”, llegando a una ventana idéntica a la de la Figura 61.

Se podrá ver el perfil del profesor y se le dará a editar para modificarlo. Una vez hecho, se modificarán los campos que se quieran cambiar y se seleccionará en “Guardar” cambios, Figura 62.

Los campos que el profesor podrá modificar son el nombre, la descripción, la foto y el despacho.

Se podrá acceder a los recursos del profesor desde la pestaña “Mis Recursos”, Figura 63 y también a sus alumnos en la pestaña “Mis alumnos”.

En “Mis Alumnos”, Figura 64, se podrá hacer clic en ellos para poder ver sus perfiles y eliminar su cuenta si se considera necesario.

Desde la pestaña “Mis Recursos”, será capaz de ir a un recurso haciendo clic en este, permitiéndose la eliminación de recursos como se indica en la Figura 65.

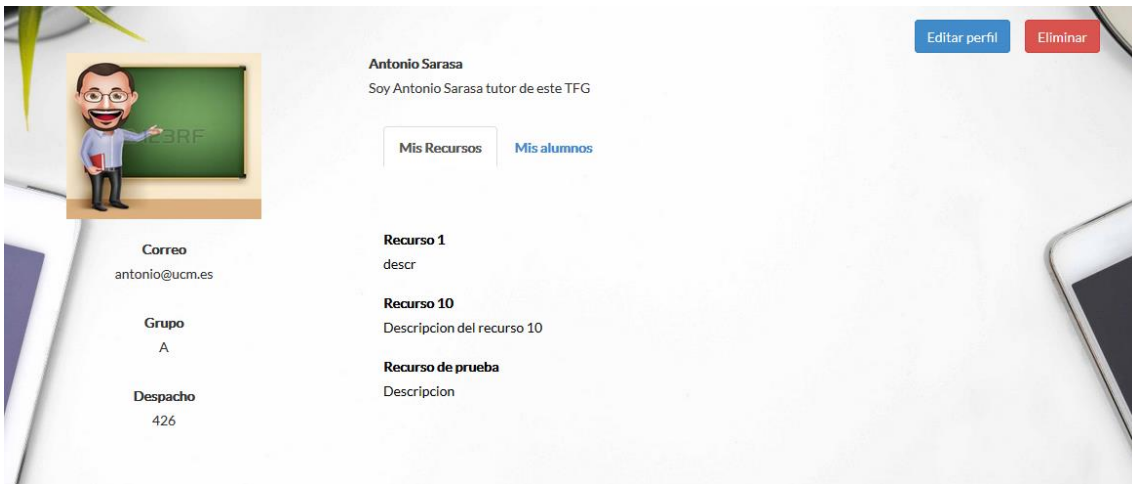


Figura 61: Perfil desde profesor.



Figura 62: Modificar perfil desde profesor.

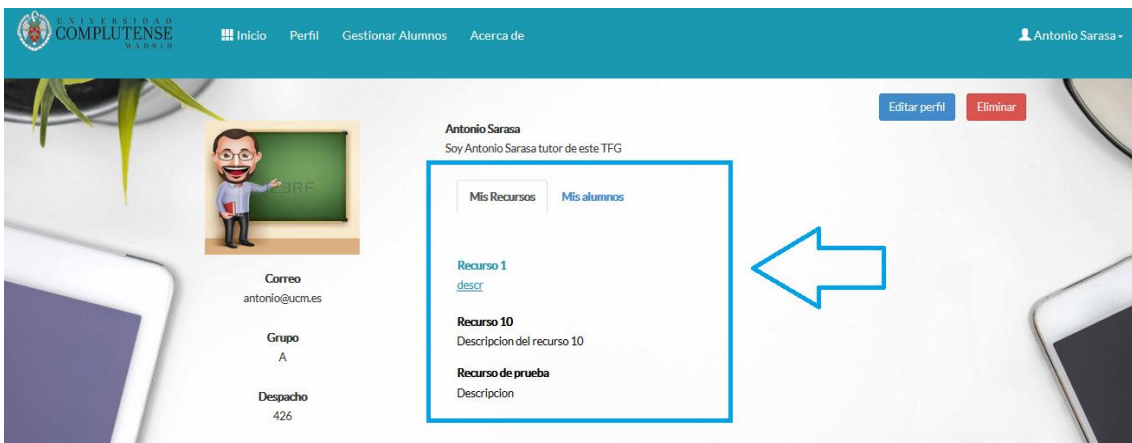


Figura 63: Mis recursos desde profesor.

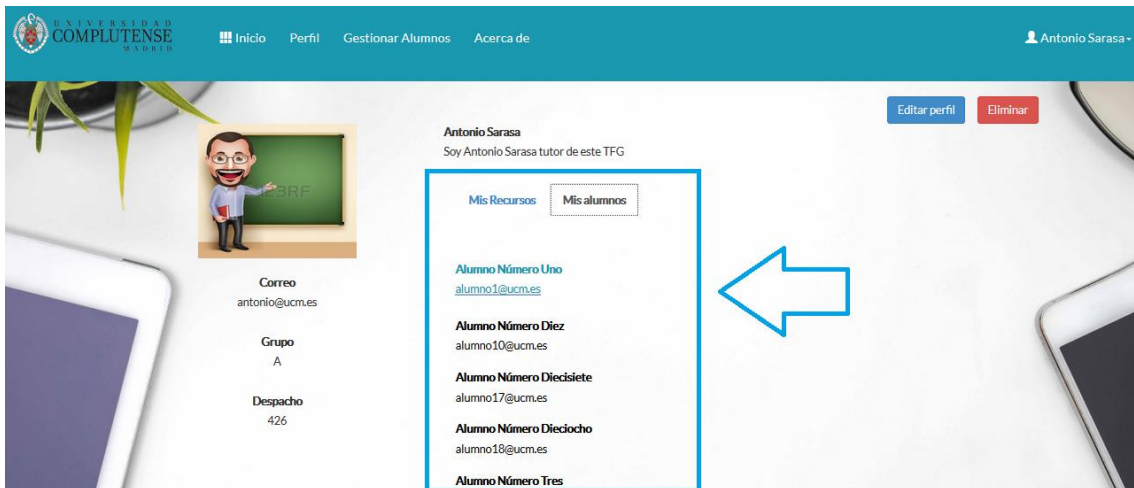


Figura 64: Mis alumnos desde profesor.



Figura 65: Gestionar alumno desde perfil desde profesor.

Paso 4: Visualizar Recursos

Para ver los recursos que se han creado, Figura 66, se deberá ir al menú superior y seleccionar en inicio. En la página de inicio estarán todos los recursos creados por el profesor.

Desde ahí, se podrán visualizar los recursos haciendo clic sobre estos. De esta forma se podrá ver los archivos y los links que contiene, Figura 67 y se tendrá la opción de modificar el recurso haciendo clic en “Editar” o, en caso de que se quiera eliminar, pulsando en “Eliminar”, ambos botones situados en la parte superior derecha del recurso.



Figura 66: Visualizar Recursos, desde profesor.

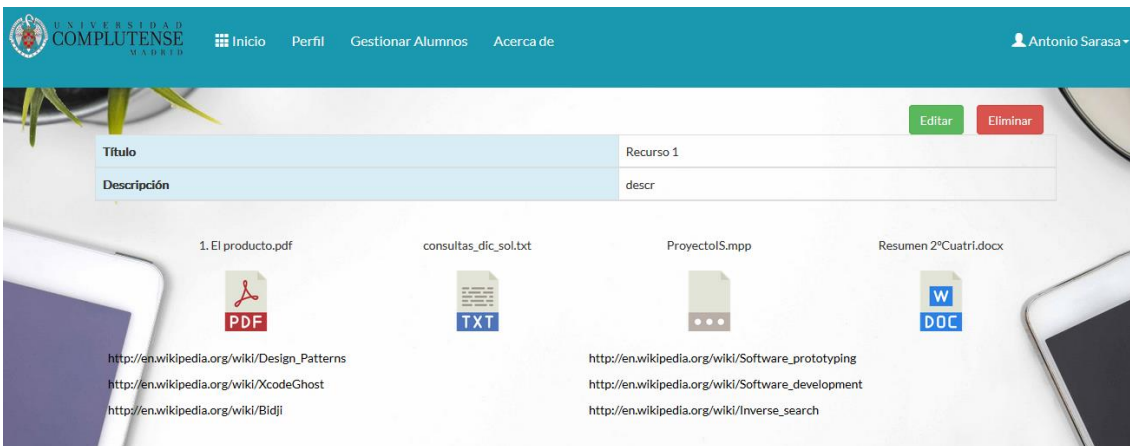


Figura 67: Recurso desde profesor.

Paso 5: Descargar archivo.

Dentro del recurso, se seleccionará el elemento que se quiera descargar, haciendo clic en el archivo y, posteriormente, dándole a la opción de guardar archivo, o en su defecto, si es un PDF, dándole al símbolo de descarga situado en la esquina superior derecha. En la Figura 68 y 69 se muestran dos ejemplos de descargas de archivos.

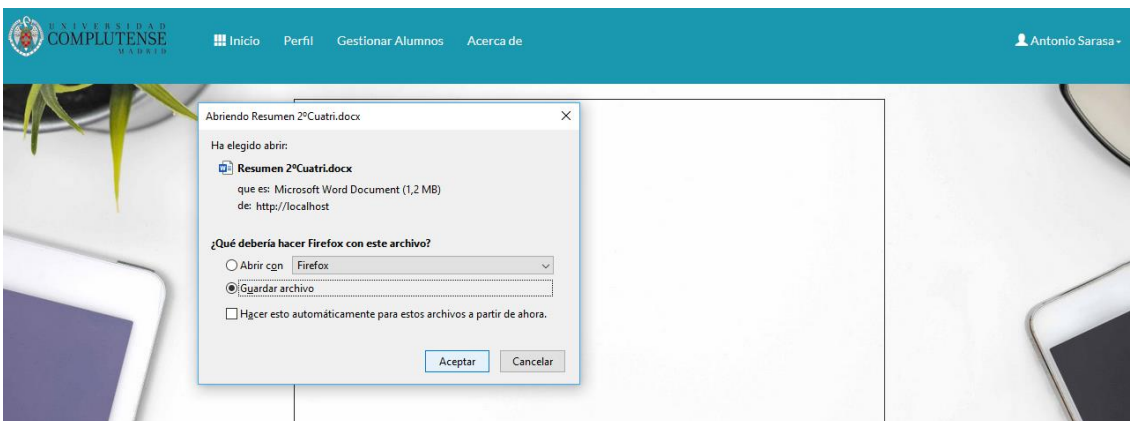


Figura 68: Descarga de archivo desde profesor.



Figura 69: Descarga de archivo PDF desde profesor.

Paso 6: Creación y eliminación de archivos.

Para la creación de archivos, se irá al menú desplegable que está situado en la esquina superior derecha con su nombre y le dará a subir archivos, Figura 70. Después se hará clic en examinar, se escogerá el archivo que se quiera subir y, por último, se pulsará a subir archivo, Figura 71.

Si se quiere eliminar un archivo subido, el profesor tendrá que hacer clic en el símbolo de la papelera que aparece al lado del archivo, Figura 72, y aparecerá un recuadro de confirmación para saber si de verdad se quiere borrar, para prevenir que se elimine un archivo sin desearlo, en la Figura 73 y 74 aparecen ambos mensajes. Este no dejará borrarlo si se usa en otro recurso existente. Si de verdad se desea borrar, habrá que borrar primero los recursos que lo contengan.

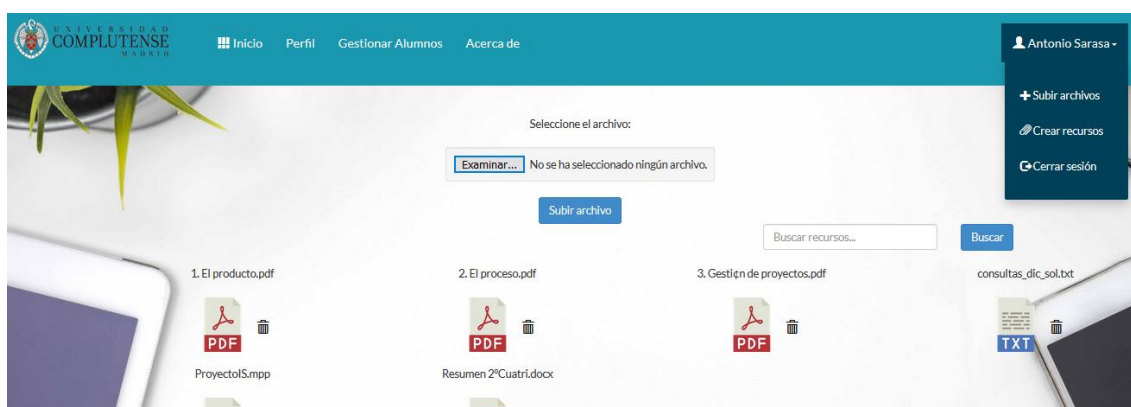


Figura 70: Menú desplegable desde profesor.



Figura 71: Selección de archivo desde profesor.

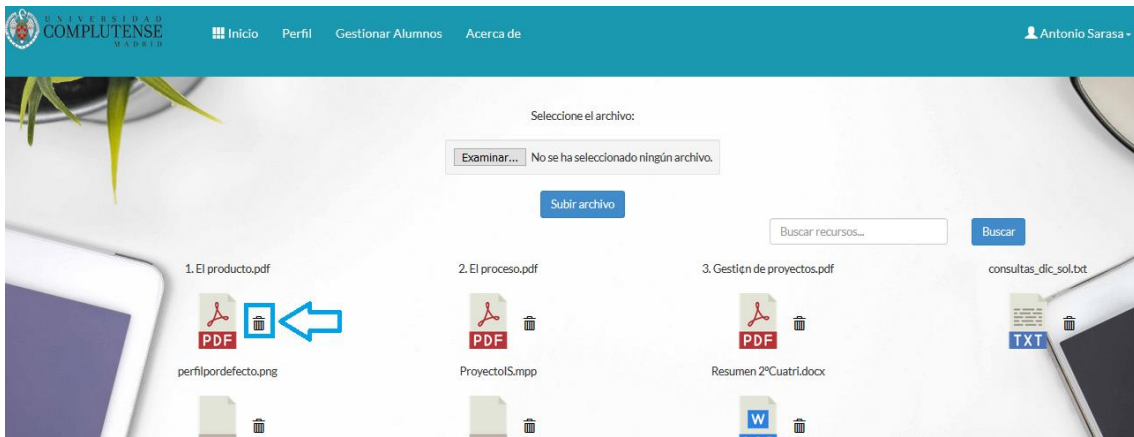


Figura 72: Borrar archivo desde profesor.

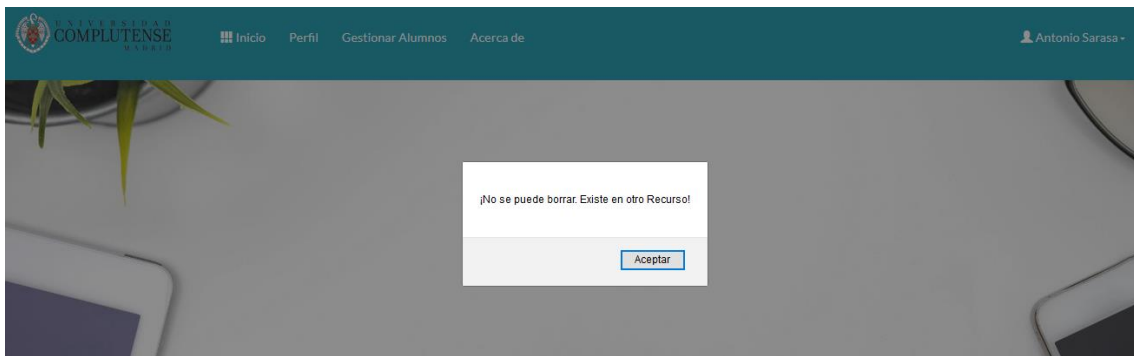


Figura 73: No se puede borrar desde profesor.

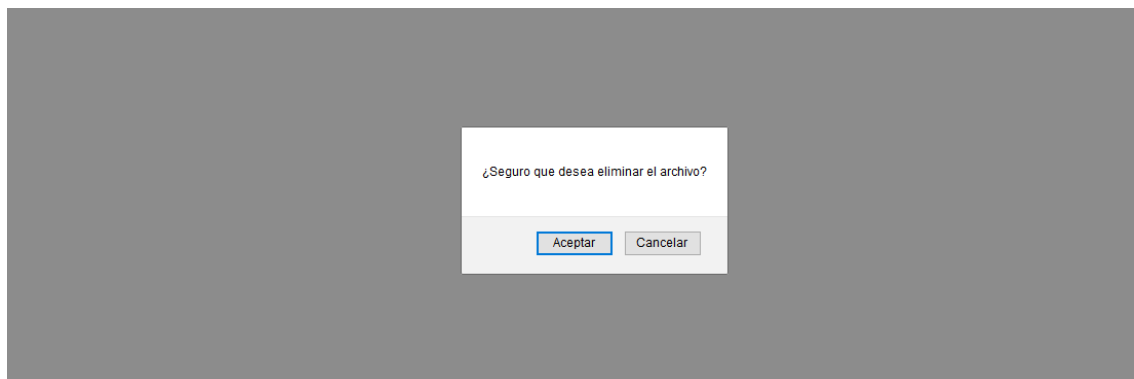


Figura 74: Mensaje de confirmación desde profesor.

Paso 7: Crear recursos.

Para ello se debe ir al menú superior derecha que hay en el nombre de usuario del profesor y se pulsará en “Crear Recursos”. Una vez se haya hecho esto, se podrá empezar a crear el recurso del profesor.

Se le podrá dar un título y una descripción, Figura 75, y posteriormente, se elegirá los archivos y los links que se quieran introducir dentro del recurso.

Los archivos se podrán escoger de dos secciones, como se muestra en la Figura 76. En la parte de la izquierda, estarán los archivos que ha subido el profesor y en la derecha, todos los archivos que se han subido a la web, por si se quiere utilizar un archivo que sea de otro profesor.

Para seleccionarlos solo habrá que hacer clic en el *checkbox* que está situado al lado del archivo que se quiera introducir en el recurso.

Más abajo se encuentran los links, Figura 77, que se seleccionarán de la misma forma, haciendo clic en su *checkbox*.

Por último, se tendrá las opciones de previsualizar el recurso, para poder verlo antes de publicarlo, la opción de guardar, para continuarlo en otro momento, y para terminar, la opción de crear recurso, el cual lo creará y publicará, véase en la Figura 78.

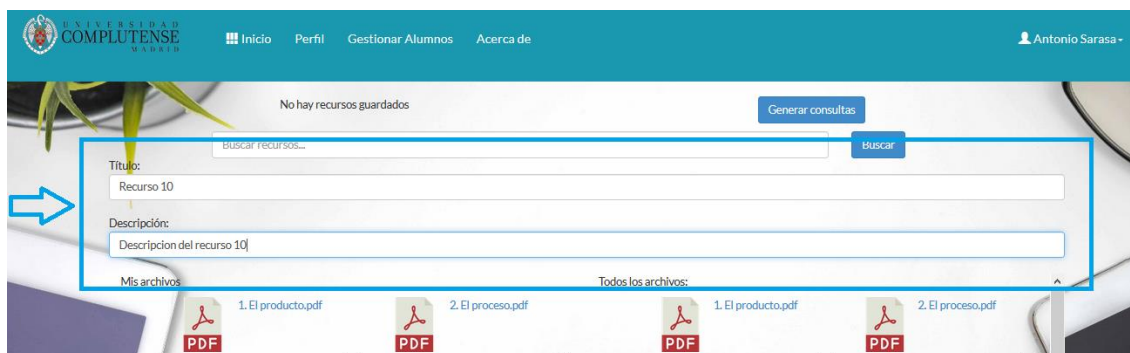


Figura 75: Título y descripción desde profesor.

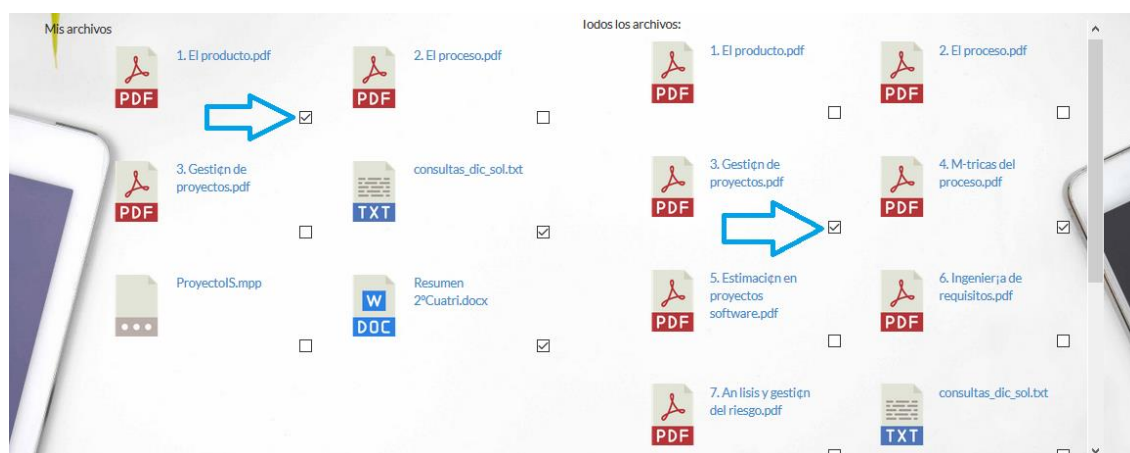


Figura 76: Elección de archivos desde profesor.

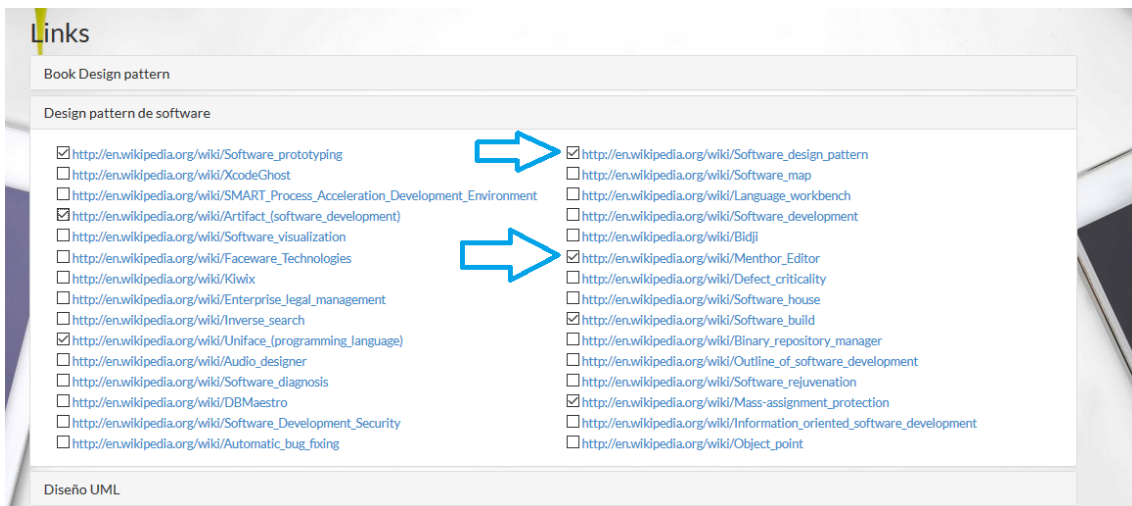


Figura 77: Selección de links desde profesor.



Figura 78: Selección de opción desde profesor.

Paso 8: Cerrar sesión.

Por último, cuando se haya terminado la interacción con la web, para cerrar sesión el profesor deberá pulsar en el menú que está situado en la esquina superior derecha en su nombre de usuario y hacer clic en “Cerrar sesión”, Figura 79.

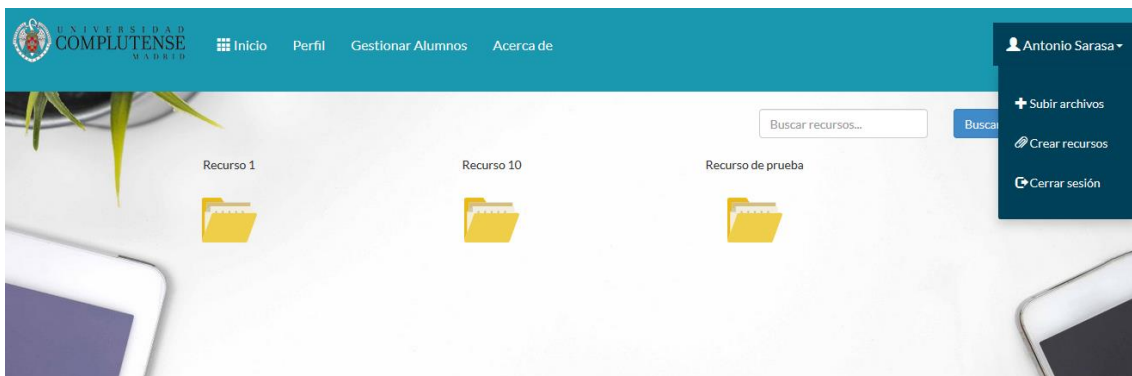



Figura 79: Cerrar sesión desde profesor.

2. Alumno

Paso 1: Inicio sesión, Figura 81.

Se inicia su propia sesión desde la página de inicio de sesión, o en caso de que no se disponga una cuenta, se realiza un registro de cuenta, para que posteriormente sea activada por el profesor que se asocie.

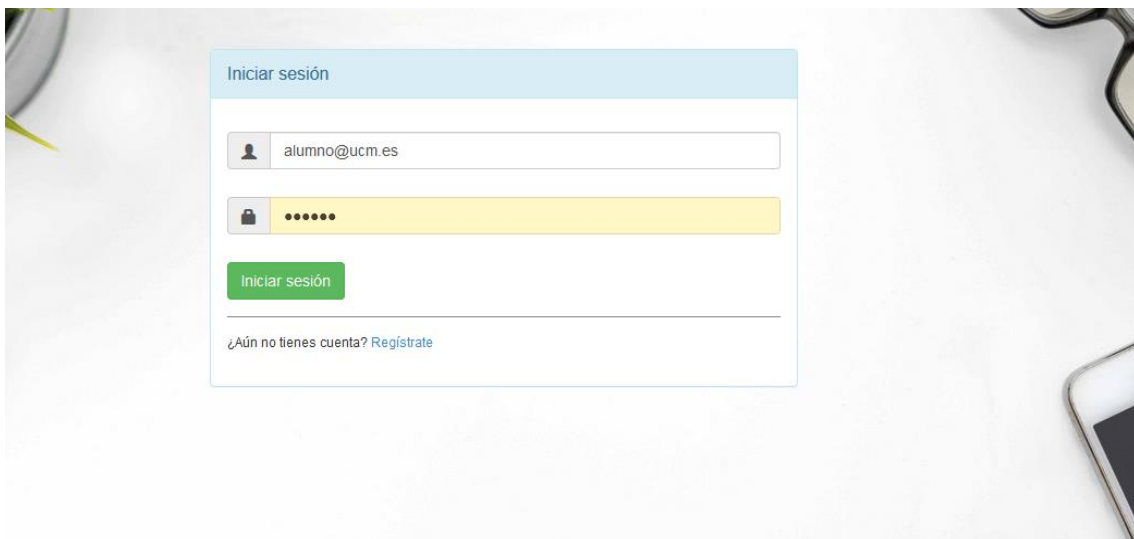
Para realizar el registro de cuenta se necesita un nombre de usuario, un correo con una contraseña, y por último, se selecciona el rol como estudiante y se indica que grupo quiere, esto se aprecia en la Figura 80.



The image shows a web form titled "Regístrate" (Sign Up) for a student. The form is set against a light blue header. It contains the following fields and options:

- A text input field for the username, containing the text "alumno".
- A text input field for the email address, containing "alumno@ucm.es".
- A password input field with a yellow background and masked characters ".....".
- Two radio buttons for role selection: "Estudiante" (selected) and "Profesor".
- A dropdown menu for the student group, currently showing "Grupo: A - Antonio Sarasa".
- A text input field for the professor's group, with the placeholder text "Escriba el grupo deseado".
- A green button labeled "Crear Cuenta" (Create Account).
- A link at the bottom that says "Volver a la página de inicio" (Return to the home page).

Figura 80: Registro de cuenta del alumno.



The image shows a web form titled "Iniciar sesión" (Log In) for a student. The form is set against a light blue header. It contains the following fields and options:

- A text input field for the email address, containing "alumno@ucm.es".
- A password input field with a yellow background and masked characters ".....".
- A green button labeled "Iniciar sesión" (Log In).
- A link at the bottom that says "¿Aún no tienes cuenta? Regístrate" (Don't you have an account? Sign up).

Figura 81: Inicio de sesión del alumno.

Paso 2: Perfil.

Para modificar o ver el perfil del alumno se deberá ir al menú superior y seleccionar “Perfil”.

Se podrá ver el perfil del alumno, Figura 82 y se le dará a editar para modificarlo. Una vez hecho, se modificarán los campos que se quieran cambiar y se seleccionará en “Guardar cambios”.

Los campos que el alumno podrá modificar son el nombre, la descripción, la foto y los estudios (GII, GIS y GIC).



Figura 82: Modificar perfil del alumno.

Paso 3: Visualizar recursos.

Para poder visualizar todo el contenido del cual dispone el alumno, tendrá dos opciones. La primera se encuentra en la pestaña de inicio, la cual es la principal. En ella aparecerán todos los recursos creados por el profesor asociado, esto se muestra en la Figura 83. Para poder ver uno de estos recursos, el alumno simplemente tendrá que pulsar sobre él, y se le mostrará el contenido de este.

La segunda opción es ir al menú superior y hacer clic en “Grupos”. Ahí aparecerán todos los grupos disponibles que haya, siendo cada grupo un lugar donde se podrán ver los recursos creados por el profesor asociado a ese grupo, véase en la Figura 84.



Figura 83: Página principal del alumno.

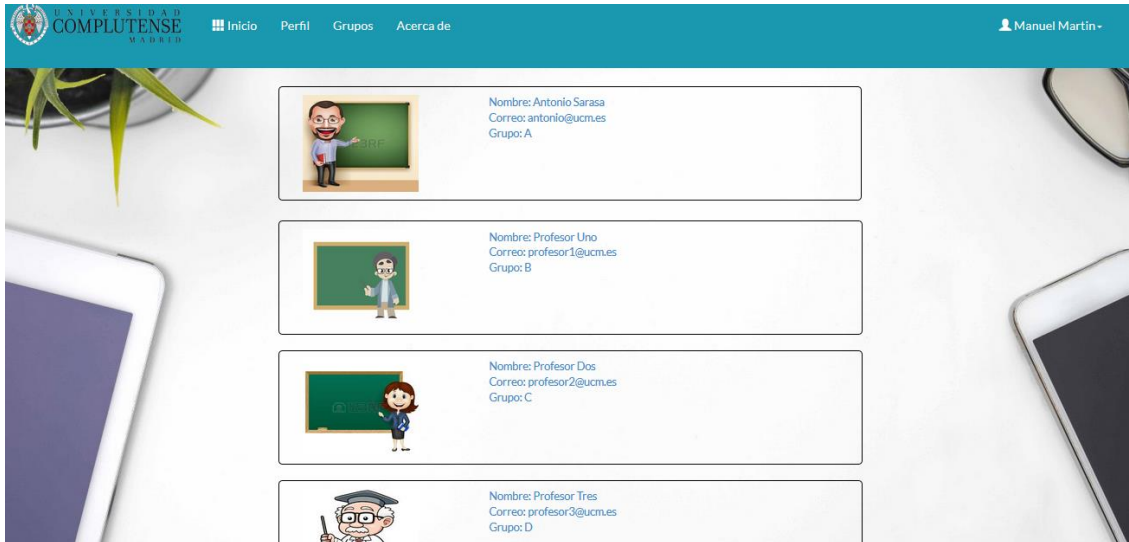


Figura 84: Visualizar grupos desde alumno.

Paso 4: Descargar archivos.

Dentro del recurso, se seleccionará el elemento que se quiera descargar, haciendo clic en el archivo y, posteriormente, dándole a la opción de guardar archivo, o en su defecto, si es un PDF, dándole al símbolo de descarga situado en la esquina superior derecha. En las Figuras 85 y 86 se indica la descarga de varios archivos.

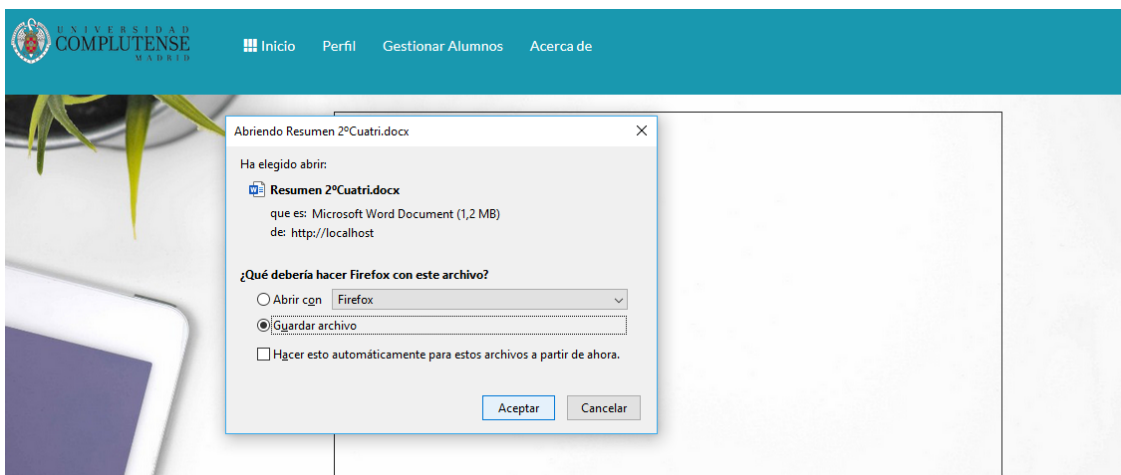


Figura 85: Descarga del archivo desde alumno.



Figura 86: Descarga del archivo PDF desde alumno.

Paso 5: Cerrar sesión.

Por último, cuando se haya terminado la interacción con la web, para cerrar sesión el alumno deberá pulsar en el menú que está situado en la esquina superior derecha en su nombre de usuario y hacer clic en “Cerrar sesión”, Figura 87.

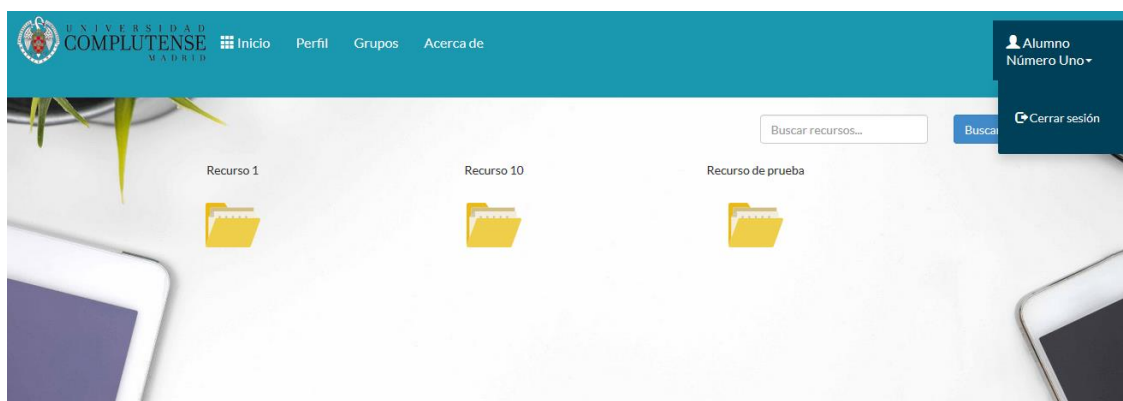


Figura 87: Cerrar sesión desde alumno.

3. Administrador

Paso 1: Inicio sesión.

Se inicia su propia sesión desde la página de inicio de sesión, Figura 88.

Al ser un rol muy importante, será una cuenta que se proporcionará a alguien de confianza para que se encargue de gestionar las cuentas de los profesores.

Para iniciar sesión, el usuario simplemente tendrá que introducir el nombre de usuario y su contraseña.

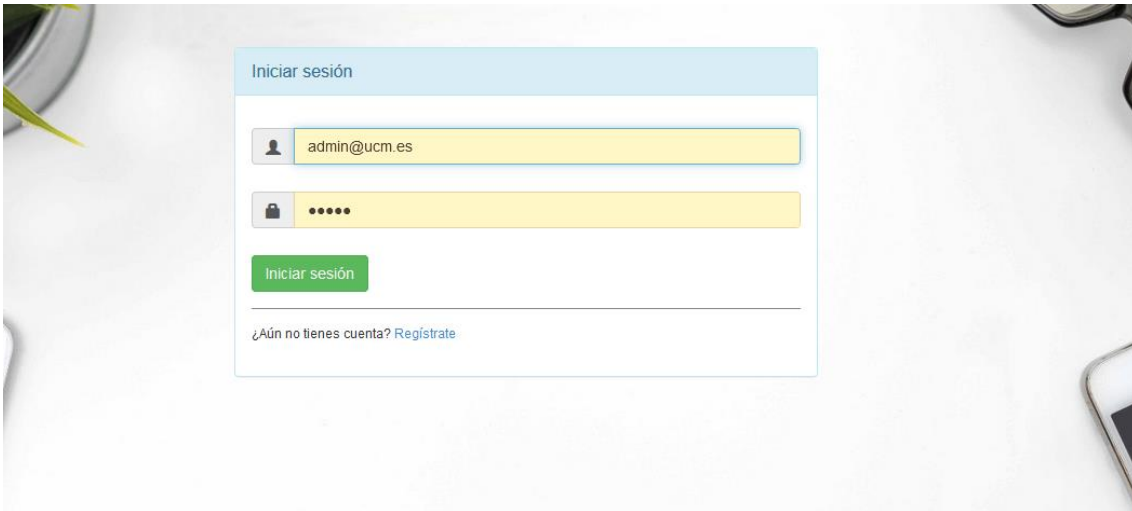


Figura 88: Inicio de sesión del administrador.

Paso 2: Perfil.

Para modificar o ver el perfil del administrador se deberá ir al menú superior y seleccionar "Perfil".

Se podrá ver el perfil del administrador, Figura 89, y se dará a editar para modificarlo. Una vez hecho, se modificarán los campos que se quieran cambiar y se seleccionará en "Guardar cambios".

Los campos que el administrador podrá modificar son el nombre, la foto y la descripción.



Figura 89: Modificar perfil del administrador.

Paso 3: Activar profesores.

El administrador puede dar de alta o activar las cuentas de los profesores. Para ello el administrador debe ir al menú superior y seleccionar “Gestionar cuentas”. Una vez se ha hecho esto, aparecerán las cuentas pendientes de activación, a las que podrá activar pulsando en el botón con el mismo nombre o, en caso de que no se quiera activar, se pulsará en rechazar, esto se muestra en la Figura 90.



Figura 90: Activar cuentas de profesores.

Paso 4: Visualizar Recursos.

Para ver los recursos que se han creado, Figura 91, se debe ir al menú superior y seleccionar en inicio. En la página de inicio estarán todos los recursos creados por los profesores, con su título y su correspondiente autor.

Desde ahí, se podrán visualizar los recursos haciendo clic sobre estos. De esta forma se podrá ver los archivos y los links que contiene y se tendrá la opción de eliminar el recurso haciendo clic en “Eliminar”, botón situado en la parte superior derecha del recurso, véase en la Figura 92.

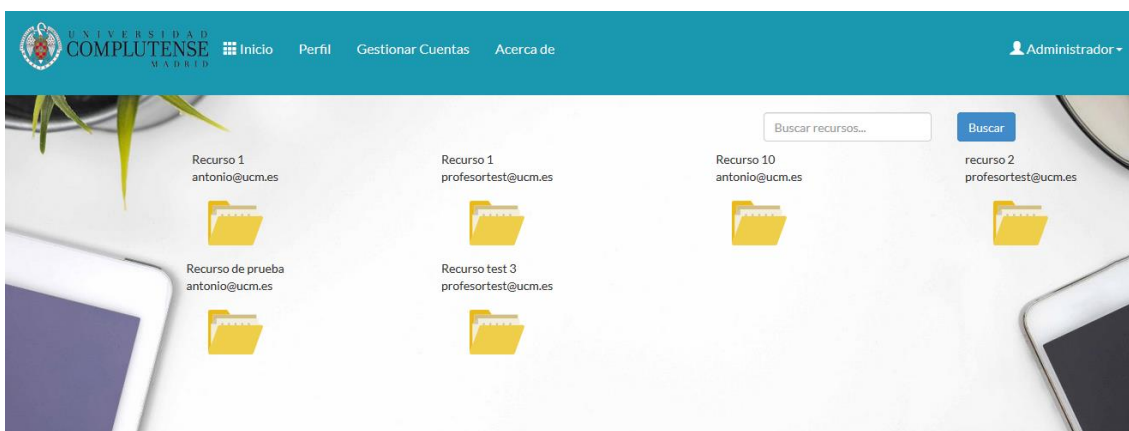


Figura 91: Visualizar Recursos de los profesores desde administrador.

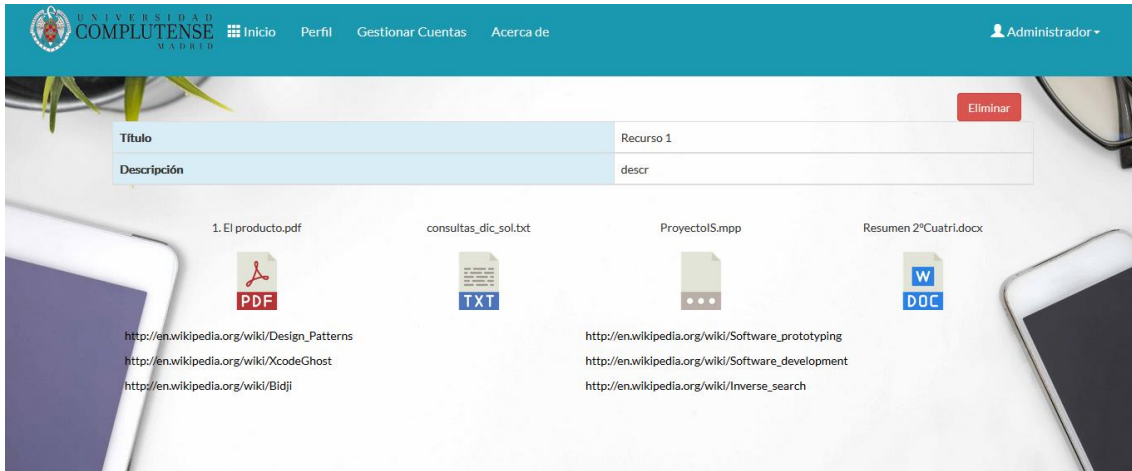


Figura 92: Recurso visto desde administrador.

Paso 5: Descargar archivo.

Dentro del recurso, se seleccionará el elemento que se quiera descargar, haciendo clic en el archivo y, posteriormente, dándole a la opción de guardar archivo, o en su defecto, si es un PDF, dándole al símbolo de descarga situado en la esquina superior derecha. En las Figuras 93 y 94 se aprecia la descarga de dos archivos diferentes.

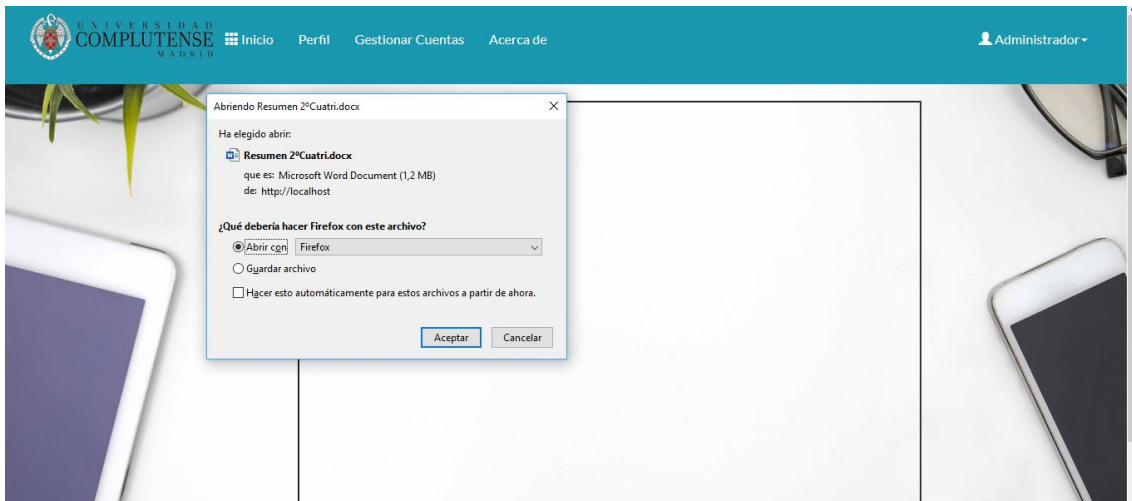


Figura 93: Descarga de archivo desde administrador.

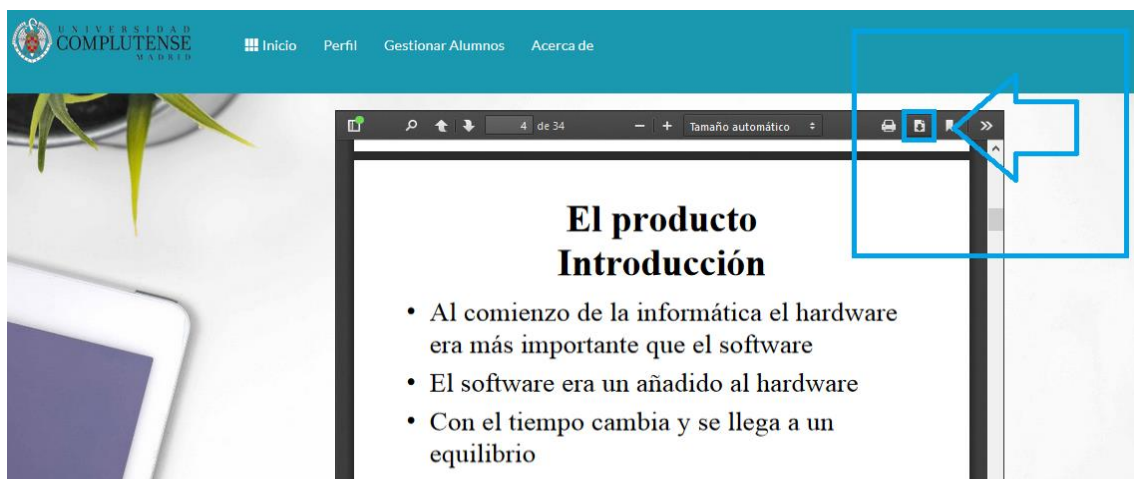


Figura 94: Descarga de archivo PDF desde administrador.

Paso 6: Cerrar sesión.

Por último, cuando se haya terminado la interacción con la web, para cerrar sesión el administrador deberá pulsar en el menú que está situado en la esquina superior derecha en su nombre de usuario y hacer clic en “Cerrar sesión”, Figura 95.

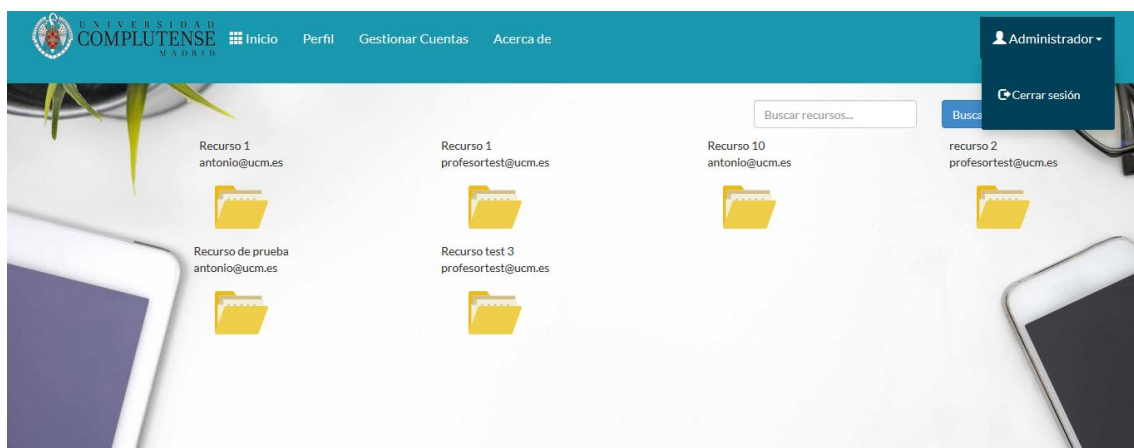


Figura 95: Cerrar sesión desde administrador.