

Trabajo de fin de grado en Ingeniería Informática

Curso académico 2019-2020

APLICACIÓN PARA GESTIÓN DE DOCUMENTACIÓN DIRIGIDA AL ALUMNO EN LA ASIGNATURA DE INGENIERÍA DEL SOFTWARE

Documentation management
application for students of Software
Engineering course

Autor/Author: Almudena López Ortega

Director: Antonio Sarasa Cabezuelo

Facultad de Informática

Universidad Complutense de Madrid

*“Cuando comas el trabajo de tus manos, dichoso
serás y te irá bien.*

Salmo 128:2”

DEDICATORIA

Porque te enorgulleces de mis éxitos más que si fueran tuyos, porque me apoyas sin condiciones cuando aún no lo he logrado, porque me acompañas en la duda y me brindas consuelo, porque nunca te cansas ni tienes palabras de desaliento para mí, porque me das ejemplo, porque sí, Miguel.

Porque el camino fácil no es el que más luce, porque sacas de mí siempre un poco más, porque te confortas en mis progresos y sufres con mis desvelos, porque sin ti no habría sabido escribir este documento, Mamá.

Porque me inspiras, porque me enseñas, porque me precedes, por tus esfuerzos para sacar la familia adelante, por enseñarme que madrugar es la clave del éxito, por arreglar lo que se rompe, por transmitirme tu alegría, por tus líneas rectas a mano alzada y tus diseños imposibles, Papá.

Por tu compañía, porque nos reímos, porque no me juzgas, porque siempre estas y porque en las peores te quedas, porque eres mi amiga y eso basta, Estefanía.

Porque me cuidáis como a una hija, y me consentís como a una nieta, porque antes de convertirse en necesidad ya lo habéis resuelto, por vuestro cariño y por compartirlo todo conmigo, por abrirme las puertas desde el primer día, Pilar y Jose Miguel.

Porque cuando no estabas me hacías falta, porque no tienes que hacer nada y así te quiero, Carmen.

AGRADECIMIENTOS

Me gustaría agradecer a mi director Antonio Sarasa Cabezuelo su confianza, su paciencia, su apoyo y su buena voluntad.

A Ferrer y a Escolano, agradecer su tiempo y que me hayan suministrado el conocimiento necesario para salvar el abismo que me separaba de del éxito.

A mi ultreya, por ayudarme a mantener siempre el objetivo, y acompañarme en el silencio que envuelve mi tecleo nocturno.

A ti Señor, que me lo has dado todo.

RESUMEN

La asignatura Ingeniería del Software, común a todos los grados en Ingeniería ofertados desde la facultad de Informática, inculca al alumno, entre otros conocimientos, la capacidad de planificar, concebir, desplegar y dirigir proyectos fomentando las habilidades de comunicación en todos los ámbitos de desarrollo del software mediante metodologías y ciclos de vida aplicados al mismo.

Para reflejar, tanto el proceso de aprendizaje, como el trabajo invertido por el alumno durante el transcurso de la evaluación, los profesores se sirven de tres entregas documentales a lo largo del curso académico.

Estas entregas, que responden a unos requisitos bien definidos, acompañan a un proyecto de desarrollo, realizado durante el curso por grupos de entre cinco y ocho alumnos.

Actualmente la redacción de esta documentación se hace de forma manual.

El proyecto aquí descrito pretende facilitar esta labor, generando de manera automatizada, y uniforme a todos los grupos de trabajo los archivos entregables que solicitarán los profesores.

Lo cual facilitará no solo la labor del alumno, sino también la revisión y posterior corrección, por parte del docente, del trabajo realizado al tratarse de un esquema común a todos los grupos de trabajo.

En este documento se analizará desde la necesidad, pasando por los requisitos funcionales y no funcionales, las decisiones de diseño y las pruebas de funcionamiento hasta completar el desarrollo de la herramienta.

PALABRAS CLAVE

Ingeniería Software, Especificación de requisitos, Plan de proyecto, Memoria final, aplicación web, alumno, IS, documentación, generación automática, PDF.

ABSTRACT

The Software Engineering course, which is a common subject in all 4-Year Bachelor Program of the Faculty of Computer Science, teaches each student, among others, to be able to plan, create, deploy and manage projects, focused on communication skills for software development using applied life cycle and methodologies.

Teachers evaluate not only the learning process but also the effort invested in the course, using three deliverables during the academic year.

These documents, which must comply with a well-defined requirement, in addition to the software development project, is made during the course by groups of five to eight students.

Nowadays, this documentation is made manually. This final paper expects to ease this task, generating these deliverables, required by teachers, automatically and homogeneously among all student working groups.

The aim of this project is to ease not only the student tasks but also the reviewing and evaluation made by the teacher, as all the groups will have the same schema in their deliverables

In this final report, the necessity, functional and not functional requirements, the decisions made for the design and the functional testing are explained.

KEYWORDS

Software Engineering, Requirement Analysis, Project Plan, Final Report, Web Application, Student, SE, Documentation, Automatic Generation of Reports, PDF.

ÍNDICE

RESUMEN	5
PALABRAS CLAVE	5
ABSTRACT	6
KEYWORDS	6
ÍNDICE DE FIGURAS	10
GLOSARIO	11
1 INTRODUCCIÓN	12
1.1 ANTECEDENTES	12
1.2 MOTIVACIÓN	12
1.3 OBJETIVO	13
2 INTRODUCTION	14
2.1 BACKGROUND	14
2.2 MOTIVATION	14
2.3 OBJECTIVES	15
3 ANÁLISIS DE REQUISITOS	16
3.1 REQUISITOS FUNCIONALES	16
3.2 REQUISITOS NO FUNCIONALES	17
4 PLAN DE PROYECTO	17
4.1 DIAGRAMA DE GANTT	18
4.2 ETAPAS DEL PROYECTO	19
5 TECNOLOGÍAS UTILIZADAS	20
5.1 PHP	20
5.2 MySQL	20
5.3 CSS3	20
5.4 BOOTSTRAP	20
5.5 HTML5	21
5.6 SUBLIME TEXT	21
5.7 APACHE	21
5.8 XAMPP	21
5.9 Google Drive	22
6 CASOS DE USO	22
6.1 ADMINISTRADOR	22
6.2 ALUMNO	23
7 MODELADO DE DATOS	28

7.1 MODELO ENTIDAD-RELACIÓN	28
7.2 Usuario	29
7.3 Proyecto	29
7.4 SRC	29
7.4 PlanDeProyecto	30
7.4 MemoriaFinal	30
8 ARQUITECTURA DE LA APLICACIÓN	31
8.1 PATRONES DE DISEÑO	31
8.1.1 Modelo Lineal Secuencial	31
8.1.2 Patrón Modelo Vista Controlador	32
8.2 MODELO FUNCIONAL	33
8.3 VISTA	34
8.4 CONTROLADOR	34
8.5 SERVIDOR	35
9 DISEÑO DE LA APLICACIÓN	36
9.1 El usuario	36
9.2 La apariencia	36
9.2.1 Colores y tipografía	36
9.2 Interfaz	36
9.2.1 Estático	37
9.2.2 Dinámico	39
10 TEST FUNCIONALES	43
11 CONCLUSIONES	44
11.1 LÍNEAS FUTURAS	44
12 CONCLUSIONS	46
12.1 FUTURE WORK	46
BIBLIOGRAFÍA/ BIBLIOGRAPHY	48
ANEXO	49
ANEXO I: GUÍA DE USO	49
1 Acceso a la aplicación	49
2 Registro	49
3 Cerrar sesión	50
4 Página principal	50
5 Editar perfil	51
6 Cumplimentar Especificación de requisitos	51
7 Mensajes de ayuda	51
8 Cumplimentar Plan de proyecto	52

9 Complimentar Memoria final	53
9 Incluir imágenes	53
10 Guardar borrador	53
11 Hacer entrega	54

ÍNDICE DE FIGURAS

FIGURA 1: PLAN DE PROYECTO, DIAGRAMA DE GANTT	18
FIGURA 2: CASO DE USO, EL USUARIO ADMINISTRADOR	22
FIGURA 3: TABLA DE CASO DE USO ACTOR, ADMINISTRADOR	22
FIGURA 4: TABLA DE CASO DE USO, BORRAR BASE DE DATOS	23
FIGURA 5: CASO DE USO PARA EL USUARIO ALUMNO	23
FIGURA 6: TABLA DE CASO DE USO ACTOR, ALUMNO	24
FIGURA 7: TABLA DE CASO DE USO, REGISTRARSE	24
FIGURA 8: TABLA DE CASO DE USO, ACCEDER	24
FIGURA 9: TABLA DE CASO DE USO, RECUPERAR CONTRASEÑA	24
FIGURA 10: TABLA DE CASO DE USO, EDITAR PERFIL	25
FIGURA 11: TABLA DE CASO DE USO, EDITAR DOCUMENTO	25
FIGURA 12: TABLA DE CASO DE USO, SUBIR IMAGEN	25
FIGURA 13: TABLA DE CASO DE USO, GUARDAR CAMBIOS	26
FIGURA 14: TABLA DE CASO DE USO, GENERAR DOCUMENTO	26
FIGURA 15: TABLA DE CASO DE USO, CERRAR SESIÓN	26
FIGURA 16: TABLA DE CASO DE USO, SUBIR IMAGEN	27
FIGURA 17: DIAGRAMA DE ENTIDAD RELACIÓN	28
FIGURA 18: MODELO LINEAL SECUENCIAL, ARQUITECTURA SOFTWARE	32
FIGURA 19: PATRÓN MVC, ARQUITECTURA SOFTWARE	33
FIGURA 20: CABECERA Y BARRA DE NAVEGACIÓN.	37
FIGURA 21: MENÚ DESPLEGABLE, ESPECIFICACIÓN DE REQUISITOS.	37
FIGURA 22: MENÚ DESPLEGABLE, PLAN DE PROYECTO.	37
FIGURA 23: MENÚ DESPLEGABLE MEMORIA FINAL.	38
FIGURA 24: PIE DE PÁGINA, ESPECIFICACIÓN DE REQUISITOS	38
FIGURA 25: PIE DE PÁGINA, PLAN DE PROYECTO	38
FIGURA 26: PIE DE PÁGINA, MEMORIA FINAL	38
FIGURA 27: PIE DE PÁGINA, ENTREGA	38
FIGURA 28: PIE DE PÁGINA, INICIO	38
FIGURA 29: REDIMENSIÓN DE INTERFAZ	39
FIGURA 30: BARRA DE NAVEGACIÓN COMPRIMIDA	39
FIGURA 31: DIAGRAMA DE RELACIONES ENTRE LAS PÁGINAS	40
FIGURA 32: INTERFAZ, ACCESO A LA APLICACIÓN	41
FIGURA 33: INTERFAZ, RECORDAR CONTRASEÑA	41
FIGURA 34: INTERFAZ, FORMULARIO DE REGISTRO	41
FIGURA 35: INTERFAZ, EDITAR PERFIL	42
FIGURA 36: INTERFAZ, CREAR GRUPO DE TRABAJO	42
FIGURA 37: GUÍA DE USO, INICIO DE SESIÓN	49
FIGURA 38: GUÍA DE USO, FORMULARIO DE REGISTRO	49
FIGURA 39: GUÍA DE USO, RECORDAD CONTRASEÑA	50
FIGURA 40: BARRA DE NAVEGACIÓN, CERRAR SESIÓN	50
FIGURA 41: PÁGINA PRINCIPAL	50
FIGURA 42: GUÍA DE USO, EDITAR PERFIL	51
FIGURA 43: BARRA DE NAVEGACIÓN, MENÚ DESPLEGABLE SRC	51
FIGURA 44: MENSAJE DE CONTENIDO	52
FIGURA 45: BARRA DE NAVEGACIÓN, MENÚ DESPLEGABLE SRC	52
FIGURA 46: BARRA DE NAVEGACIÓN, MENÚ DESPLEGABLE MEMORIA FINAL	53
FIGURA 47: AÑADIR IMAGEN	53
FIGURA 48: GUARDAR BORRADOR	53
FIGURA 49: HACER ENTREGA, GENERAR PDF	54

GLOSARIO

En lo sucesivo las siguientes siglas corresponden a:

SRC	Especificación de requisitos
FDI	Facultad de informática
SW	Software
IS	Ingeniería del Software
MVC	Modelo Vista Controlador
BD	Base de Datos
UCM	Universidad Complutense de Madrid
PDF	Portable Document Format
TFG	Trabajo Fin de Grado
CSS	Cascading Style Sheets
PHP	Hypertext Preprocessor
HTML	HyperText Markup Language

1 INTRODUCCIÓN

La presente memoria describe el desarrollo de la aplicación web para gestión de documentación realizada para la asignatura de Ingeniería de Software como trabajo de fin de grado.

En ella, se analiza el carácter técnico y funcional de la aplicación desarrollada. Presentando la necesidad real que ha propiciado dicho desarrollo, además del análisis de los requisitos técnicos realizado, las decisiones de diseño que se tomaron, la arquitectura de la aplicación y las pruebas de funcionamiento necesarias para finalizar el desarrollo completo de la herramienta.

Se puede encontrar también, una guía de uso destinada a los alumnos, aunque, como se describe en las conclusiones, tras las pruebas realizadas con usuarios potenciales, se observó que no necesitaron hacer uso del manual, al encontrar la aplicación intuitiva y fácil de usar.

1.1 ANTECEDENTES

En la actualidad, no existe ninguna herramienta que automatice la entrega de documentación por parte de los alumnos para su evaluación. La redacción de los distintos entregables que se requieren durante la asignatura, se realiza de forma manual.

Los propios alumnos, en función de una serie de pautas dadas por el profesorado, realizan las diferentes memorias de manera no uniforme, detectándose por parte de los profesores la falta de algunos puntos de interés para la evaluación o trabajos mal estructurados.

Al comienzo de la asignatura, se establecen las fechas límite para las distintas entregas, por lo que los alumnos conocen previamente el calendario, pero no disponen de una herramienta que les ayude a documentar todo el proyecto a medida que se va realizando, siendo frecuente las entregas fuera de plazo en las que se aprecia un software bien desarrollado, pero mal documentado.

1.2 MOTIVACIÓN

La necesidad de este proyecto surge al comprobar los distintos problemas que se encuentran durante la asignatura de Ingeniería del Software, tanto los alumnos, a la hora de documentar sus proyectos de software, como los profesores, en el momento de evaluar esas entregas. En concreto, algunos de los problemas detectados son:

- Ineficiencia a la hora de evaluar por parte del profesor debido a la poca homogeneidad de las documentaciones entregadas, en las que frecuentemente faltan apartados o siguen un orden inconexo.
- Ineficiencia a la hora de documentar al no realizarse de manera secuencial según va avanzando el proyecto software. Se tiende a documentar todas las

fases al final, por lo que se requiere de un trabajo adicional para recordar los desarrollos.

- Falta de planificación de las distintas fases que requiere un proyecto completo de Ingeniería del Software.
- Falta de control de versiones en la documentación.
- Falta de colaboración entre los diferentes alumnos de un grupo, que tienden a documentar las distintas fases del proyecto de manera individual y en local.

Por lo tanto, queda demostrada la necesidad de una aplicación web que facilite, no solo la labor del alumno, sino también la revisión y posterior corrección por parte del docente.

1.3 OBJETIVO

El objetivo de este Trabajo Fin de Grado es el de desarrollar una aplicación web, que permita a los alumnos de la asignatura de Ingeniería del Software generar la documentación necesaria para presentar durante el transcurso de la asignatura en formato PDF, de forma automática y siguiendo un formato estándar. Los principales objetivos a cubrir por este TFG son:

- Permitir a los alumnos registrarse en la aplicación y crear un grupo de trabajo seleccionando a los compañeros de una lista.
- Definir un esquema claro de las tareas a realizar, dividiendo la documentación en las diferentes entregas: Especificación de Requisitos (SRC), Plan de Proyecto y Memoria final.
- Facilitar el cumplimiento de la información durante el cuatrimestre, mediante mensajes de ayuda con las instrucciones para rellenar los apartados.
- Guardar de forma automática la información para que pueda ser editada o consultada en los siguientes accesos.
- Gestionar las imágenes y diagramas que se deben insertar en los documentos.
- Mantener la sesión iniciada mientras se insertan contenidos en la aplicación.
- Imprimir en formato PDF el documento requerido por el profesor en cada una de las entregas.

2 INTRODUCTION

This final report describes the development of the web application used for the documentation management in the Software Engineering course.

In this report, the technical and functional characteristics are analyzed. Besides, the necessity, which has motivated this development, and the requirement analysis are shown. The design decisions are explained as well as the architecture of the application. Finally, the functional testing and their results are described.

It has been decided to include an annex with a user guide for the students. Although, just as described in the conclusions, the functional testing demonstrates that the application is intuitive and user-friendly. So, there is not a real necessity to use this guide.

2.1 BACKGROUND

Nowadays, the students have not any tool that automates the deliverables which must be made for their evaluation. This documentation, required by the course, must be written manually.

The students make all the reports according to some guidelines provided by the teachers, but these reports are not homogeneous. Therefore, the teachers have detected some mistakes in them such as the lack of some interesting points for the evaluation or a bad structure of the deliverables.

At the beginning of the course, the different deadlines are defined for the deliverables so the students know the complete schedule. However, they do not have any tool to help them to document the project whenever it is being done. The deliverables are often sent after the deadlines and although the software deployment has a good quality, it is poorly documented.

2.2 MOTIVATION

The motivation of this project starts by checking the different problems that not only the students have to document their software projects but also the teachers have to evaluate the deliverables during the course of Software Engineering. Basically, some of these detected problems are:

- The teachers lack efficiency to evaluate the deliverables because all the documentation is not homogeneous, and usually misses some sections and has an incorrect order.
- The students are inefficient to document the project during the progress of the course. tending to write all the deliverables at the end so an extra effort is needed to remember the whole software deployment.
- There is not a good planification of the whole project just as a project of software engineering requires.
- There is not a proper control version for the documentation.

- There is not a good partnership among all the members of the working group, tending to document all the phases of the project individually.

Therefore, it has been demonstrated the necessity of a web application which eases the students' work as well as the evaluation made by the teachers.

2.3 OBJECTIVES

The aim of this final project is to deploy a web application that allows students of Software Engineering course to automatically generate, in a PDF format, the reports required during the course. Also, these reports must be homogeneous. The main objectives to cover in this project are:

- To allow the students to register in the application and create working groups, selecting their partners from a list.
- To define a simple schema of the required tasks, dividing the documentation in different reports, such as, Requirement Analysis, Project Plan or Final Report.
- To ease the filling of the information during the course, using help messages with the instructions to complete the different sections.
- To save automatically the information with the objective of completing it in the future.
- To manage the images and graphics which must be included in the deliverables.
- To keep the session in course while the documents are completed in the application.
- To generate in a PDF format the final report required by the teachers in each deliverable.

3 ANÁLISIS DE REQUISITOS

En este apartado se van a describir los diferentes requisitos recogidos durante la fase de análisis del proyecto. Se ha decidido dividirlos en dos grupos:

- **Requisitos funcionales:** Son aquellos que describen los servicios que la aplicación debe proporcionar, es decir, aquellas funcionalidades que la aplicación debe permitir realizar a los usuarios, como, por ejemplo, generar un documento PDF.
- **Requisitos no funcionales:** Son aquellos que no se refieren directamente a las funciones que tiene que realizar la aplicación sino a como tiene que hacer la aplicación las diferentes funciones. Así, por ejemplo, un requisito no funcional sería el de usar lenguaje de programación PHP.

Todos estos requisitos serán la base del diseño de los casos de uso que permitirán diseñar la aplicación para cumplir con todos los requisitos expuestos.

3.1 REQUISITOS FUNCIONALES

El listado de requisitos funcionales detectados durante la fase de análisis es:

- **Registro de nuevos usuarios:** Un alumno de la asignatura de Ingeniería del Software tiene que poder registrarse en la aplicación mediante la utilización de un correo electrónico y una contraseña.
- **Recuperación de contraseña:** Al ser un acceso mediante usuario y contraseña, el usuario debe poder recuperar la contraseña en caso de olvido.
- **Modificación del perfil del usuario:** Un usuario en cualquier momento puede cambiar alguno de los datos de su usuario como puede ser el nombre, correo electrónico o contraseña.
- **Grupos de trabajo:** Los usuarios podrán formar grupos de trabajo, permitiendo a los usuarios compartir la información con los miembros de su grupo.
- **Editar documentos:** Los usuarios podrán rellenar los distintos apartados de la documentación que deben presentar para su evaluación. Además, la información deberá estar perfectamente estructurada en secciones, indicando en cada una de ellas, mediante mensajes de ayuda, qué debe ser escrito en dicho apartado.
- **Subir imágenes:** Ciertos apartados de la documentación consistirán en diagramas o imágenes que los usuarios podrán subir a la aplicación.
- **Información accesible y segura:** Los usuarios podrán guardar toda la información escrita y podrán recuperarla en cualquier momento. Para ello, la información debe guardarse en una base de datos que permita las copias de seguridad.
- **Generación de documento PDF:** una vez los usuarios hayan completado toda la información de los diferentes apartados, se podrán generar un documento PDF que siga siempre la misma plantilla. Ese documento se podrá descargar automáticamente en el ordenador del usuario.

3.2 REQUISITOS NO FUNCIONALES

El listado de requisitos funcionales detectados durante la fase de análisis es:

- Lenguaje de programación PHP.
- Instalación de la aplicación en un servidor web Apache.
- Almacenamiento de la información en una base de datos MySQL.
- Almacenamiento de imágenes en un repositorio remoto.
- Aplicación de las metodologías de desarrollo de proyectos propias de la Ingeniería del Software.
- Securitización de la aplicación mediante el uso de un usuario y contraseña.
- Gestión de usuarios y grupos.
- Diseño fácil e intuitivo mediante el uso de hojas de estilo CSS.
- Uso de software libre en todo el proyecto.
- Compatibilidad con los navegadores web estándar como Mozilla Firefox, Google Chrome e Internet Explorer.
- Descarga de documentos en formato PDF.

4 PLAN DE PROYECTO

A continuación, se detallan las etapas de trabajo que ha implicado el desarrollo de este proyecto.

4.1 DIAGRAMA DE GANTT

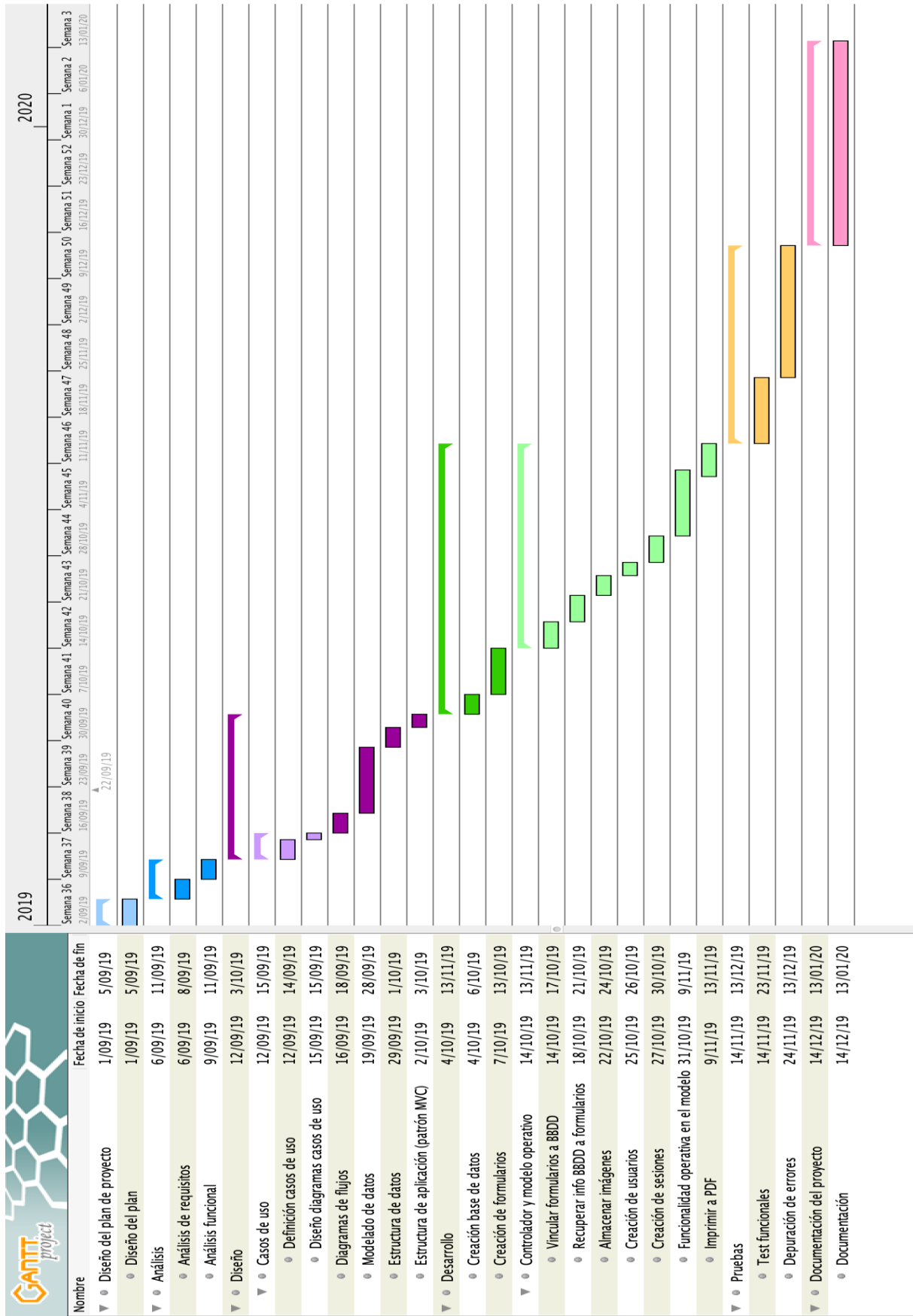


Figura 1: Plan de proyecto, diagrama de Gantt

4.2 ETAPAS DEL PROYECTO

Como vemos en la Figura 1, el diagrama de Gantt exhibe la lista de tareas realizadas y su planificación temporal. Se han contemplado también los fines de semana como trabajados, ya que en este desarrollo han supuesto los días de más actividad. Veamos el desglose:

- Tarea 1. Diseñar plan de proyecto: Tiene 5 días de duración, ha consistido en definir las tareas a realizar y trazar el plan de trabajo que se iba a seguir hasta culminar el proyecto.
- Tarea 2. Análisis:
 - a. Análisis de requisitos: Examinar los objetivos y trazar los requisitos mínimos que debe cubrir el desarrollo de la aplicación. Tiene 3 días de duración.
 - b. Análisis funcional: Dados los requisitos anteriores, establecer que funciones deberá tener la aplicación para cubrirlos. Tiene 3 días de duración.
- Tarea 3. Diseño:
 - a. Casos de uso.
 - i. Definición de casos de uso: Establecer actores y operaciones que van a realizar. Tiene 3 días de duración.
 - ii. Diseño diagramas casos de uso: Describir de forma esquemática los casos de uso definidos. Tiene 1 día de duración.
 - b. Diagramas de flujo: Establecer la secuencia que va a seguir la aplicación y las conexiones entre las distintas funcionalidades. Tiene 3 días de duración.
 - c. Modelado de datos diagramas: Dibujar diagramas UML. Tiene 10 días de duración.
 - d. Estructura de datos: Tiene 3 días de duración.
 - e. Estructura de aplicación, integrando patrón MVC Tiene 2 días de duración.
- Tarea 4. Desarrollo:
 - a. Creación base de datos Tiene 3 días de duración.
 - b. Creación de formularios para definir la interfaz gráfica de la aplicación. Tiene 1 semana de duración.
 - c. Controlador y modelo operativo:
 - i. Vincular formularios a base de datos. Tiene 4 días de duración.
 - ii. Recuperar información de base de datos a formularios. Tiene 4 días de duración.
 - iii. Almacenar imágenes, en el repositorio. Tiene 4 días de duración.
 - iv. Creación de usuarios. Tiene 2 día de duración.
 - v. Creación de sesiones. Tiene 4 días de duración.
 - vi. Implementar funcionalidad operativa en el modelo. Tiene 10 días de duración.
 - vii. imprimir a PDF el documento generado por la aplicación de forma automatizada. Tiene 5 días de duración.
- Tarea 5. Pruebas:
 - a. Test funcionales: Probar toda la aplicación y verificar que no genera errores. Tiene 10 días de duración.

- b. Depuración de errores: Junto a los test funcionales, generan un ciclo iterativo de prueba y error. Tiene 20 días de duración.
- Tarea 6. Documentación: Redactar la memoria adjunta al proyecto. Tiene 31 días de duración.

5 TECNOLOGÍAS UTILIZADAS

Para el desarrollo de la aplicación se han usado las siguientes tecnologías:

5.1 PHP

Es un lenguaje de programación de propósito general de código del lado del servidor. Se ha usado para implementar el modelo lógico de la aplicación, y para enlazar los formularios HTML con la base de datos MySQL a través de un controlador que describiremos más adelante.

5.2 MySQL

Es un sistema de gestión de base de datos relacional. Se ha usado en la aplicación para implementar las tablas de almacenamiento de información necesarias.

5.3 CSS3

Se trata de un lenguaje de diseño gráfico, concretamente la evolución más reciente de las hojas de estilo en cascada. Se usa para modificar la apariencia de documentos estructurados escritos en un lenguaje de marcado.

Se ha usado para dar formato a la Interfaz de los documentos HTML que forman la vista del proyecto.

5.4 BOOTSTRAP

Bootstrap es un frameworks que sirve para desarrollo front-end en exclusiva. Puede encontrarse como uno de los proyectos más destacados en GitHub. Se usa como una biblioteca multiplataforma de código abierto para diseñar aplicaciones de tipo web.

En esta aplicación se ha usado para complementar CSS3 antes descrito y mejorar la apariencia de la interfaz.

5.5 HTML5

Hablamos de un lenguaje de marcado necesario para elaborar páginas web. Se forma añadiendo texto y etiquetas de tipo HTML a un documento. Finalmente se encuentra con la extensión “.html” o “.htm” dentro del directorio de la aplicación web.

En esta herramienta se ha utilizado para definir la estructura y describir el contenido de la aplicación que cumplimenta a la funcionalidad realizada en PHP.

5.6 SUBLIME TEXT

Se trata de un editor de texto hecho en C++ con plugins en Python. Tiene una apariencia clásica y es muy funcional para usarse en desarrollo. Al no tratarse de código libre requiere de licencia para su uso. Aunque también dispone de un modo de prueba con la funcionalidad limitada, bastante eficiente para desarrollo de aplicaciones web.

Utilizado en esta herramienta para editar código fuente en PHP, HTML5, CSS3 y Bootstrap.

5.7 APACHE

Apache es un servidor web multiplataforma con protocolo HTTP de código abierto, aplicable en páginas web estáticas y dinámicas. Su desarrollo y mantenimiento está supervisado por la Apache Software Foundation aunque toda la comunidad de usuarios del proyecto HTTP Server (httpd) puede hacer correcciones y aportes para su extensión. Su código es sencillo de conseguir dada su popularidad global.

En este trabajo se ha usado como servidor de acceso local, en la creación de un entorno virtualizado para el correcto funcionamiento de la herramienta desarrollada.

5.8 XAMPP

Es una plataforma que sirve como servidor independiente y se obtiene en un paquete de SW libre. Tiene entre otras utilidades: gestión de base de datos MySQL, interprete de PHP, la conexión entre ambas a través de la herramienta PHPmyAdmin, servidor web Apache y Open SSL. Sirve tanto para usarse dentro de la propia LAN como para crear un servidor que exponga aplicaciones de forma globalizada en toda la red.

Se ha usado como plataforma de soporte para conectar el servidor Apache, con la base de datos MySQL y el módulo funcional y estructural de la aplicación, descritos en PHP y HTML5 respectivamente.

5.9 Google Drive

Se trata de un espacio de almacenamiento gratuito de hasta 15 GB vinculado a una cuenta de usuario administrada por Google. Puede usarse como repositorio ya que dispone de control de versiones y recuperación de versiones pasadas.

Usado como repositorio para el control de versiones durante todo el desarrollo, ha supuesto la principal medida de contingencia en la recuperación de pérdida de datos.

6 CASOS DE USO

En este apartado, se detallan los diferentes casos de uso tenidos en cuenta para el desarrollo de la aplicación. Se ha decidido dividir el apartado en función del actor reflejado en el caso de uso.

Téngase en cuenta que la aplicación es una herramienta fundamentalmente para los alumnos, no será necesario que los profesores o administradores realicen ninguna tarea sobre la misma, salvo la de acceder para borrar la base de datos al finalizar la asignatura, y dejar así la aplicación lista para el siguiente curso académico.

Por lo tanto, la mayoría de los casos de usos estarán referidos a los alumnos, usuarios de la aplicación.

Se dispone de dos actores:

- **Administrador:** Se tratará de los profesores que puedan acceder al servidor para dejar la aplicación a punto para un nuevo curso académico.
- **Alumno:** Cualquiera de los alumnos de la asignatura de Ingeniería del Software que automatice las entregas de la documentación usando la aplicación.

6.1 ADMINISTRADOR

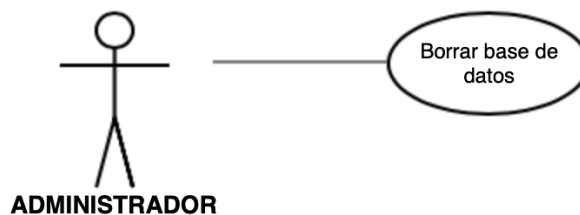


Figura 2: Caso de Uso, el usuario Administrador

Actor	Administrador
Descripción	Se tratará de los profesores de la asignatura de Ingeniería del Software
Responsabilidad	Se encargarán de dejar la aplicación preparada para cada nuevo curso académico
Comentario	

Figura 3: Tabla de caso de uso actor, Administrador

Para este rol se define un único caso de uso:

Caso de uso	Borrar base de datos
Actor	Administrador
Descripción	Borrar el contenido completo de la base de datos para dejar la aplicación lista para el siguiente curso
Precondición	Tener acceso al servidor en donde está desplegada la aplicación y la base de datos MySQL
Secuencia	<ol style="list-style-type: none"> 1. Acceso mediante usuario y contraseña al servidor 2. Acceso mediante usuario y contraseña a la base de datos. 3. Borrado de la base de datos
Postcondición	La base de datos queda limpia y lista para el siguiente curso académico
Excepciones	<p>Paso 1: Usuario o contraseña incorrecto</p> <p>Paso 2: Usuario o contraseña incorrecto</p>
Comentarios	En caso de no disponer de la contraseña, se podrá siguiendo los pasos definidos por el servidor

Figura 4: Tabla de Caso de uso, Borrar base de datos

6.2 ALUMNO

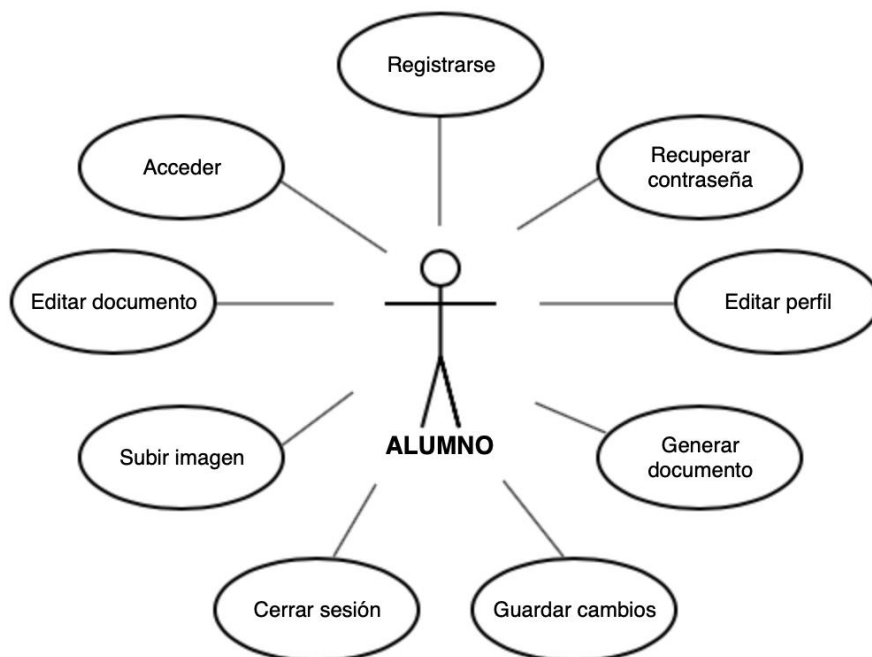


Figura 5: Caso de Uso para el usuario Alumno

Actor	Alumno
Descripción	Cualquiera de los alumnos de la asignatura de Ingeniería del Software
Responsabilidad	Serán los usuarios potenciales de la aplicación y podrán realizar todas las tareas necesarias sobre ella con el fin de automatizar la entrega de la documentación necesarias para la asignatura
Comentario	

Figura 6: Tabla de Caso de Uso actor, Alumno

Para este rol se definen los siguientes casos de uso:

Caso de uso	Regístrate
Actor	Alumno
Descripción	Un alumno nuevo se puede registrar en la aplicación
Precondición	Ser alumno de este curso de la asignatura Ingeniería del Software
Secuencia	<ol style="list-style-type: none"> 1. Acceso desde la página de inicio al botón de registro 2. Complimentar el formulario. 3. Finalizar el registro
Postcondición	El usuario ya podrá acceder a la aplicación
Excepciones	Paso 3: El usuario ya está registrado
Comentarios	

Figura 7: Tabla de Caso de Uso, Registrarse

Caso de uso	Acceder
Actor	Alumno
Descripción	Acceso a la aplicación
Precondición	Estar registrado como usuario de la aplicación
Secuencia	<ol style="list-style-type: none"> 1. Acceso a la página web de inicio de la aplicación 2. Introducir usuario y contraseña 3. Acceder
Postcondición	El usuario ya podrá acceder a la aplicación
Excepciones	Paso 3: Usuario o contraseña incorrecto
Comentarios	

Figura 8: Tabla de Caso de Uso, Acceder

Caso de uso	Recuperar contraseña
Actor	Alumno
Descripción	Un alumno no recuerda su contraseña de acceso
Precondición	Estar registrado como usuario de la aplicación
Secuencia	<ol style="list-style-type: none"> 1. Acceso desde la página de inicio al botón de recuperar la contraseña 2. Complimentar el usuario 3. Pulsar recuperar contraseña
Postcondición	El usuario recibe su contraseña
Excepciones	Paso 3: El usuario no está registrado en la aplicación
Comentarios	

Figura 9: Tabla de Caso de Uso, Recuperar contraseña

Caso de uso	Editar perfil
Actor	Alumno
Descripción	Un alumno podrá cambiar los datos de su registro, excepto el nombre de usuario
Precondición	Estar registrado como usuario de la aplicación
Secuencia	<ol style="list-style-type: none"> 1. Acceder a la aplicación con usuario y contraseña 2. Pulsar el botón de editar perfil 3. Modificar los datos 4. Guardar
Postcondición	El usuario modifica sus datos registrados
Excepciones	Paso 1: El usuario no está registrado en la aplicación Paso 4: El usuario intenta modificar un dato no permitido
Comentarios	

Figura 10: Tabla de Caso de Uso, Editar perfil

Caso de uso	Editar documento
Actor	Alumno
Descripción	Un alumno rellenará alguno de los campos de la documentación
Precondición	Estar registrado como usuario de la aplicación
Secuencia	<ol style="list-style-type: none"> 1. Acceso a la aplicación con usuario y contraseña 2. Navegación a la sección que desea rellenar desde la barra de la aplicación, 3. Completar el apartado correspondiente
Postcondición	El usuario cumplimenta uno de los apartados de la documentación
Excepciones	Paso 1: El usuario no está registrado en la aplicación
Comentarios	Téngase en cuenta que antes de salir de la aplicación, deberá guardar los datos.

Figura 11: Tabla de Caso de Uso, Editar documento

Caso de uso	Subir imagen
Actor	Alumno
Descripción	Un alumno adjunta alguno de las imágenes necesarias para la documentación
Precondición	Estar registrado como usuario de la aplicación
Secuencia	<ol style="list-style-type: none"> 1. Acceso a la aplicación con usuario y contraseña 2. Navegación a la sección donde se desea subir la imagen 3. Pulsar botón examinar 4. Seleccionar imagen del directorio local 5. Guardar borrador
Postcondición	La imagen se almacena en un repositorio
Excepciones	Paso 2: Que la sección no permita adjuntar imágenes Paso 5: Que la imagen exceda el tamaño permitido.
Comentarios	

Figura 12: Tabla de Caso de Uso, Subir imagen

Caso de uso	Guardar cambios
Actor	Alumno
Descripción	Un alumno desea guardar todos los apartados rellenos
Precondición	Estar registrado como usuario de la aplicación y haber cumplimentado alguno de los apartados
Secuencia	1. Pulsar el botón guardar borrador de cada uno de los formularios
Postcondición	La información se almacena en la base de datos
Excepciones	Paso 1: No se ha relleno ninguno de los apartados
Comentarios	

Figura 13: Tabla de Caso de Uso, Guardar cambios

Caso de uso	Generar Documento
Actor	Alumno
Descripción	Un alumno genera la memoria final en formato PDF
Precondición	Estar registrado como usuario de la aplicación y haber cumplimentado los apartados de la documentación
Secuencia	1. Acceso a la sección de vista previa y generar documento. 2. Pulsar el botón de generar documento y seleccionar en que directorio se guardará 3. Descarga del documento
Postcondición	El usuario descarga en su ordenador el documento PDF con toda la información de la entrega
Excepciones	Paso 2: No se ha relleno ninguno de los apartados
Comentarios	

Figura 14: Tabla de Caso de Uso, Generar Documento

Caso de uso	Cerrar sesión
Actor	Alumno
Descripción	Un alumno cierra sesión en la aplicación
Precondición	Estar registrado como usuario de la aplicación
Secuencia	1. Pulsar el botón de cerrar sesión
Postcondición	Se vuelve a la página de inicio
Excepciones	
Comentarios	Es necesario guardar la información antes de cerrar sesión

Figura 15: Tabla de Caso de Uso, Cerrar sesión

Caso de uso	Subir imagen
Actor	Alumno
Descripción	
Precondición	Estar registrado
Secuencia	<ul style="list-style-type: none"> • Pulsar botón examinar. • Seleccionar imagen del directorio local. • Guardar borrador.
Postcondición	La imagen se almacena en un repositorio.
Excepciones	<ul style="list-style-type: none"> • Que la imagen exceda el tamaño permitido. • Que no se almacene debidamente.
Comentarios	

Figura 16: Tabla de Caso de Uso, Subir imagen

7 MODELADO DE DATOS

En este apartado se va a detallar el modelo de datos que se ha seguido para guardar toda la información aportada por los alumnos en la base de datos MySQL.

7.1 MODELO ENTIDAD-RELACIÓN

En la figura 17 se muestra el diagrama entidad-relación orientativo utilizado para el modelado en la base de datos. En él se presentan 5 entidades básicas, siendo cada una de ellas una tabla diferente en la base de datos: Usuario, Proyecto, SRC, PlanDeProyecto y MemoriaFinal.

Con estas entidades se definen las siguientes relaciones:

- Usuario – Proyecto: siempre un proyecto estará formado por varios usuarios, y además, un usuario solo podrá pertenecer a un proyecto.
- Proyecto – SRC: Un proyecto podrá contener una única sección de SRC, y a su vez una sección de SRC pertenecerá a un único proyecto.
- Proyecto – PlanDeProyecto: Un proyecto podrá contener una única sección de PlanDeProyecto, y a su vez una sección de PlanDeProyecto pertenecerá a un único proyecto.
- Proyecto – MemoriaFinal: Un proyecto podrá contener una única sección de MemoriaFinal, y a su vez una sección de MemoriaFinal pertenecerá a un único proyecto.

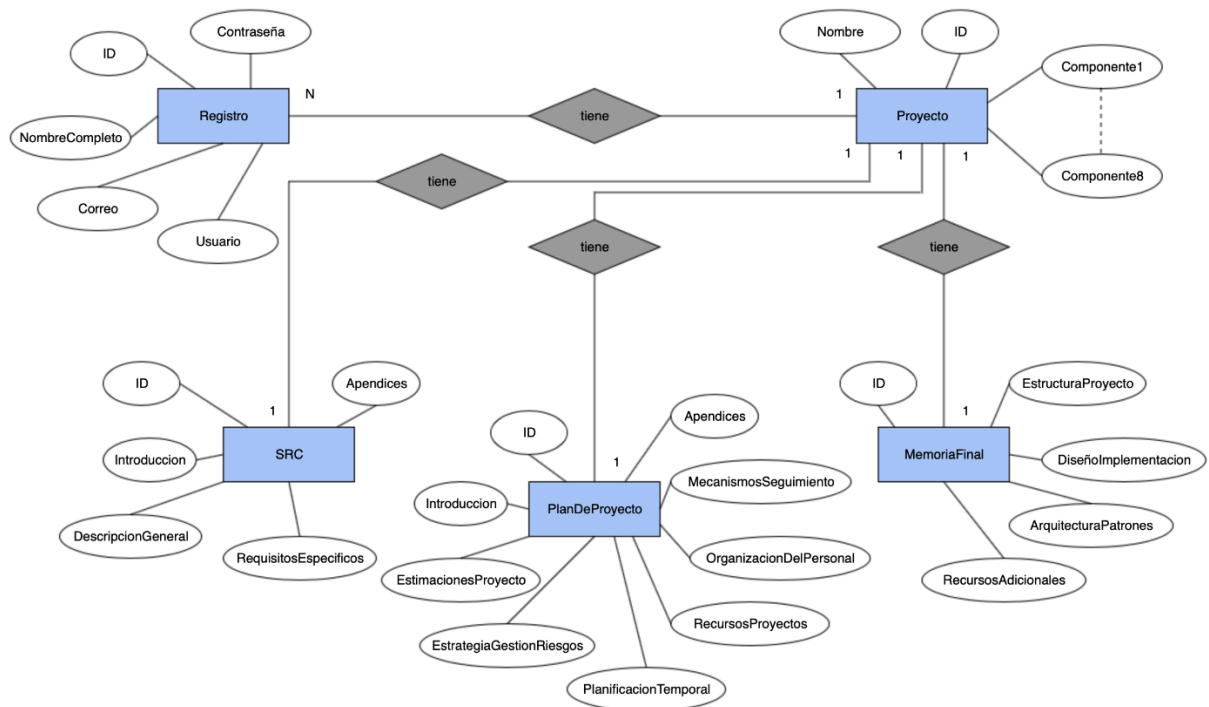


Figura 17: Diagrama de entidad relación

7.2 Usuario

En esta tabla se guardará el registro del alumno que utiliza la aplicación. En concreto cada sujeto usuario quedará definido por los siguientes atributos:

- ID: identificador autoincremental que define unívocamente a cada usuario.
- NombreCompleto: Será el nombre con el que se ha matriculado el alumno en la asignatura
- Email: Correo electrónico del alumno.
- Usuario: Será usado para acceder a la aplicación web.
- Contraseña: Será usada para verificar las credenciales del usuario.

7.3 Proyecto

En esta tabla se guardará el registro del proyecto que se está documentando con la aplicación. Cada uno de los proyectos quedará definido por los siguientes atributos:

- ID: identificador auto-incremental que define unívocamente a cada proyecto.
- Nombre: nombre con el que se identifica el proyecto.
- Componentes: será un array con los identificadores de los usuarios que componen el grupo de trabajo para este proyecto.

7.4 SRC

En esta tabla se guardará toda la información relativa a la entrega que tendrán que hacer los alumnos en la fase de análisis de requisitos. Esta tabla tendrá los siguientes conjuntos de atributos, la mayoría de ellos serán las diferentes secciones o campos que tendrán que ir rellenando los alumnos. Aquí se ha tenido en cuenta que estos atributos podrían ser textos o imágenes:

- ID: identificador que corresponde con el ID del proyecto al que pertenece.
- Introduccion: En este conjunto de atributos se recogerá toda la información relativa a la introducción, como, por ejemplo, el propósito o el alcance del proyecto.
- DescripcionGeneral: En este conjunto de atributos se recogerá toda la información relativa a la descripción general de la especificación de requisitos, como, por ejemplo, las características del usuario o los requisitos futuros.
- RequisitosEspecificos: En este conjunto de atributos se recogerá toda la información específica de los requisitos, como, por ejemplo, los requisitos funcionales o las restricciones de diseño.
- Apendices: En este conjunto de atributos se recogerá los posibles anexos que quieran adjuntar los alumnos.

7.4 PlanDeProyecto

En esta tabla se guardará toda la información relativa a la entrega que tendrán que hacer los alumnos en la fase de planificación del proyecto. Esta tabla tendrá los siguientes conjuntos de atributos, la mayoría de ellos serán las diferentes secciones o campos que tendrán que ir rellenando los alumnos. De igual forma, aquí se ha tenido en cuenta que estos atributos podrían ser textos o imágenes:

- ID: identificador que corresponde con el ID del proyecto al que pertenece.
- Introduccion: En este conjunto de atributos se recogerá toda la información relativa a la introducción, como, por ejemplo, el propósito del plan o los aspectos de rendimiento.
- EstimacionesProyecto: En este conjunto de atributos se recogerá toda la información relativa a las estimaciones de tiempo, coste o esfuerzo, entre otros campos.
- EstrategiaGestionRiesgo: En este conjunto de atributos se recogerá toda la información específica de la gestión del riesgo, como, por ejemplo, el plan de gestión del riesgo o el estudio de los riesgos.
- PlanificacionTemporal: En este conjunto de atributos se recogerá los diagramas específicos de la planificación temporal del proyecto, como, por ejemplo, el diagrama de Gantt o las tablas de uso de recursos.
- RecursosProyecto: En este conjunto de atributos se recogerá el listado de recursos necesarios para llevar a cabo el proyecto: personal, hardware, software y otros recursos.
- OrganizacionDelPersonal: En este conjunto de atributos se recogerá la información relativa a la gestión del personal: la estructura del equipo y los informes de gestión.
- MecanismosSeguimiento: En este conjunto de atributos se recogerá la información relativa a los mecanismos de seguimiento y control del proyecto: el plan de calidad y la gestión del control de cambios.
- Apendices: En este atributo se recogerá, en diferentes campos, los posibles anexos que quieran adjuntar los alumnos.

7.4 MemoriaFinal

En esta tabla se guardará toda la información relativa a la entrega que tendrán que hacer los alumnos en la fase final del proyecto. Esta tabla tendrá los siguientes conjuntos de atributos, la mayoría de ellos serán las diferentes secciones o campos que tendrán que ir rellenando los alumnos. De igual forma, aquí se ha tenido en cuenta que estos atributos podrían ser textos o imágenes:

- ID: identificador que corresponde con el ID del proyecto al que pertenece.
- EstructuraProyecto: En este atributo se recogerá toda la información relativa a la estructura del proyecto.
- DiseñoImplementacion: En este conjunto de atributos se recogerán los diagramas de secuencia y otros que los alumnos quieran adjuntar.
- ArquitecturaPatrones: En este conjunto de atributos se recogerá toda la información específica de la arquitectura o los patrones utilizados durante el desarrollo del proyecto, como, por ejemplo, el patrón MVC o el patrón Singleton.
- RecursosAdicionales: En este atributo se recogerán los recursos que hayan sido utilizados por los alumnos y que no hayan sido descritos en ninguna otra sección.

8 ARQUITECTURA DE LA APLICACIÓN

A continuación, se describe la arquitectura empleada para implementar la herramienta que se está analizando en el presente documento.

8.1 PATRONES DE DISEÑO

8.1.1 Modelo Lineal Secuencial

Durante todo el proceso de diseño, se ha seguido el modelo lineal secuencial o en cascada, el cual ha facilitado la gestión del desarrollo. Se decidió elegir este modelo, en vez de modelos más complejos como el iterativo, al tener desde el inicio unos requisitos bien definidos, los cuales se estimaba que no iban a variar sustancialmente durante el desarrollo del proyecto.

Al tratarse de un modelo lineal, cada fase quedó perfectamente definida en el tiempo, pudiéndose organizar el trabajo mejor, y teniendo en cuenta, que para poder pasar a la siguiente fase tenía que darse por finalizada la anterior.

Las fases en las que ha consistido el proyecto han sido, en este orden:

- 1 **Investigación:** La cual ha permitido identificar las necesidades a cubrir con el desarrollo de la aplicación.
 - 2 **Análisis de requisitos:** Encontrar el tándem perfecto entre las ideas que iban surgiendo y las necesidades a cubrir. En esta fase, se fijan los objetivos para su posterior implementación.
 - 3 **Diseño funcional:** Definir la forma concreta en la que se quiere dar solución a los requisitos fijados en la etapa anterior.
 - 4 **Desarrollo:** Hablamos ahora de implementar la estructura que requiere cubrir estas necesidades funcionales, creando los distintos módulos funcionales que ejecutarían la parte operativa.
 - 5 **Pruebas:** Realizar las pruebas pertinentes para verificar que el desarrollo realizado cumple con las funcionalidades buscadas y que a su vez da solución a los requisitos marcados en la fase 3 del modelo.
- 5.1 **Reparación o Mantenimientos:** Esta es una de las fases que más tiempo requiere, ya que en caso de que las pruebas fallen, antes de dar por concluido el proyecto, es necesario hacer ajustes en el código de la aplicación hasta obtener el resultado esperado. Cabe destacar que en este punto sí se han realizado iteraciones dentro de la misma fase, ya que una vez se realizaba la reparación, se volvían a realizar las pruebas para comprobar el funcionamiento de la aplicación.

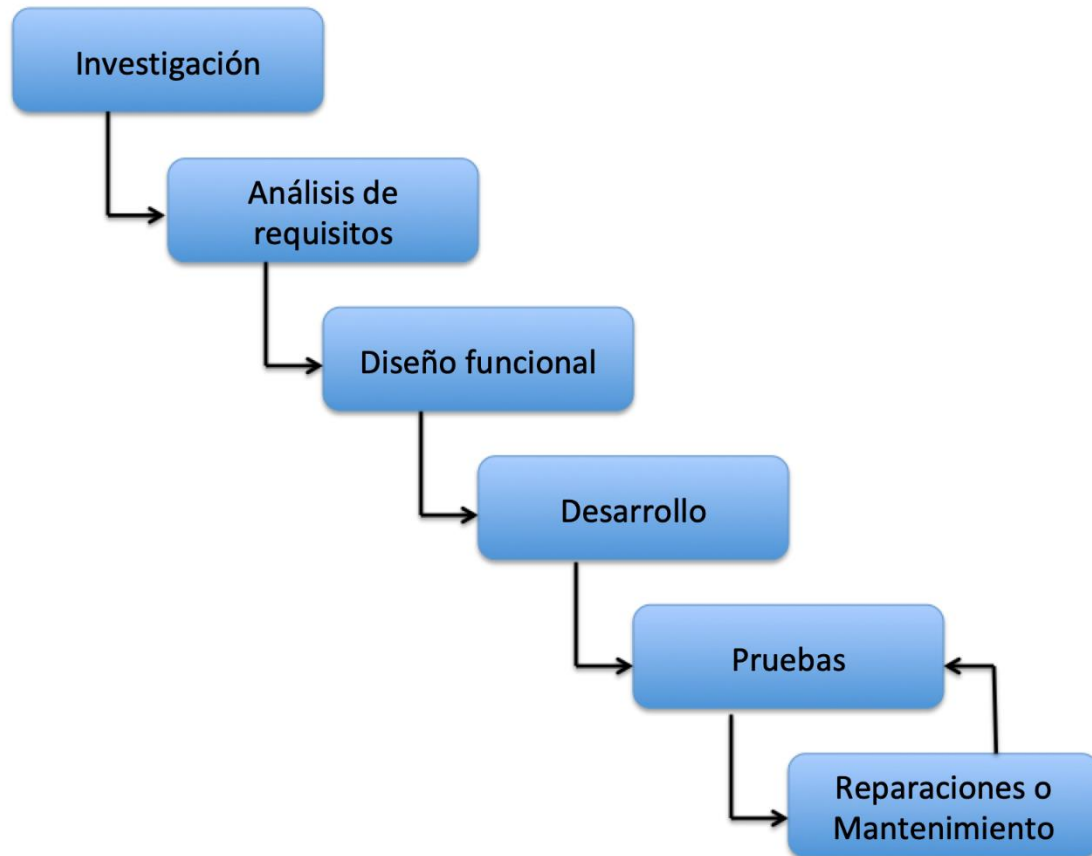


Figura 18: Modelo lineal secuencial, Arquitectura Software

8.1.2 Patrón Modelo Vista Controlador

Es un patrón de arquitectura software que consiste en separar en tres módulos los datos, la lógica y la gestión de eventos junto a la comunicación entre ellos.

Nos situamos pues en la estructura principal de la aplicación, ya que se trata del patrón usado para el desarrollo de la herramienta. Abarca los puntos cuatro y cinco del diseño lineal secuencial que se ha seguido para realizar este proyecto.

Se ha implementado separando estos módulos en la siguiente estructura:

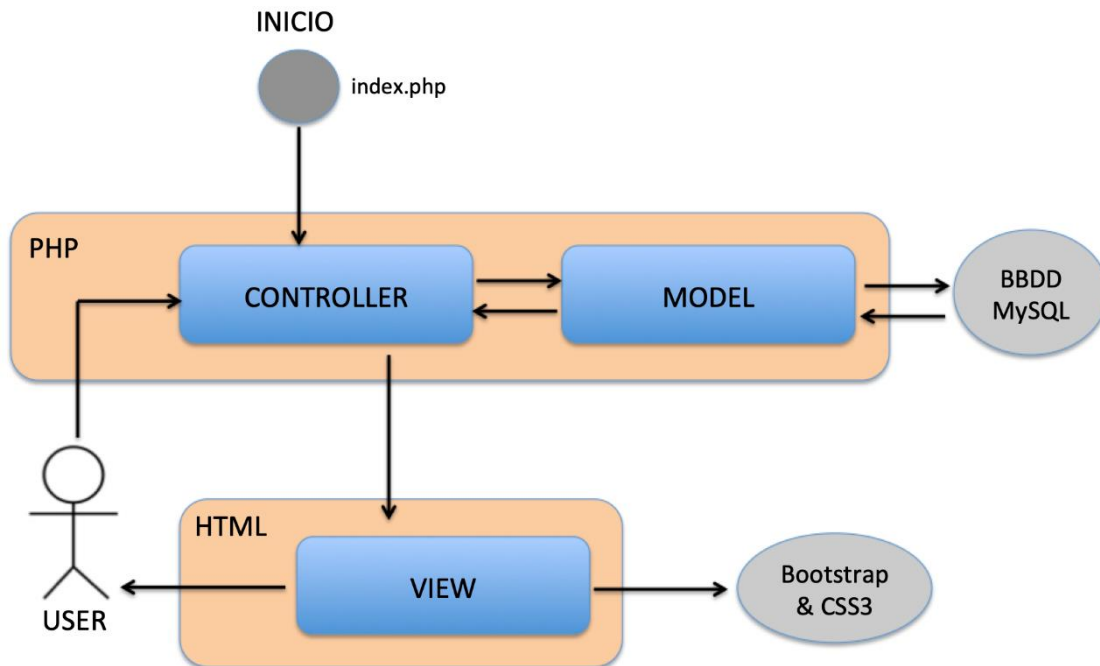


Figura 19: Patrón MVC, Arquitectura Software

Como puede observarse en la Figura 19, el controlador (Controller) es el encargado de hacer las comunicaciones entre el usuario (User) y el modelo operativo. A su vez, es quien indica a la estructura de la aplicación cuándo y qué vista (View) debe mostrar.

Estos tres módulos principales se sirven a su vez de las tecnologías antes mencionadas para desplegar su funcionalidad. El modelo (Model) es quien consulta y almacena la información en la base de datos a través de consultas SQL inyectadas mediante código PHP.

La vista se encarga de mostrar la estructura de datos, y se sirve de Bootstrap y CSS3 para mejorar la apariencia de la aplicación.

Destacar que estos tres módulos han sido diseñados al mismo nivel, aislando los contenidos de cada uno de ellos a los otros dos, lo cual mejora el desarrollo y mantenimiento en paralelo de dichos componentes, ya que cumplen ciclos de vida muy distintos entre sí.

Analicemos ahora, cada uno de los módulos observables en la Figura 19.

8.2 MODELO FUNCIONAL

Es uno de los componentes del patrón MVC antes descrito, y engloba el conjunto de clases y funciones que contienen la información con la cual el sistema opera. Gestiona tanto los accesos a dicha información, como actualizaciones y consultas a la base de datos.

En este módulo se han implementado los privilegios de acceso, de manera que cada usuario pueda interactuar únicamente con la información que tiene asociada.

Contiene la operativa de la aplicación, y en este módulo se ejecutan las acciones que llegan a través del controlador, que es quien las solicita. El resultado de esta ejecución es enviado a la vista para ser expuesta al usuario.

Uno de los aspectos más destacables dentro de este patrón (MVC), es que la comunicación con la Base de datos se hace exclusivamente a través del modelo, lo cual permite aislar tanto el componente gráfico (Vista) como el controlador del acceso a la base de datos.

8.3 VISTA

El segundo componente descrito en el patrón MVC.

Es el encargado de proporcionar al usuario la interfaz gráfica que presenta la información y los eventos del modelo en un formato adecuado para facilitar la interacción del usuario.

Las acciones del usuario en esta interfaz, son recogidas por el controlador y trasladadas al modelo para su posterior ejecución. La respuesta es reconducida nuevamente hasta la vista a través del controlador.

La vista puede cambiar la apariencia mostrada siempre y cuando la interacción del usuario no requiera activar ningún evento que precise de ejecución. Por ejemplo: si se encuentra en la página de Login, ir a cambiar de contraseña.

8.4 CONTROLADOR

Hablamos ahora del tercer componente del patrón MVC que estamos desglosando. Es el intermediario encargado de las comunicaciones entre Vista y Modelo.

En este módulo se reciben los eventos accionados por el usuario y se dirige una petición al modelo (anteriormente descrito). También se gestionan internamente las peticiones de ejecución necesarias para resolver dichos eventos.

Cuando el evento en ejecución requiere mostrar al usuario un cambio en la interfaz de la aplicación, el controlador envía una petición a la vista proporcionando la información necesaria para efectuarlo.

8.5 SERVIDOR

La aplicación desarrollada es de tipo web, por lo que requiere ser alojada en un servidor.

Durante el desarrollo de la herramienta, se ha optado por utilizar un servidor local tipo Apache instalado en el equipo de referencia para el desarrollo, utilizando el paquete XAMPP (ambos descritos en tecnologías utilizadas).

Los elementos que componen la aplicación, están instalados dentro de este servidor, que puede encontrarse accediendo desde la máquina a la red local en la dirección <http://127.0.0.1> o en <http://localhost/>.

Se ha optado por trabajar desde un servidor local por las ventajas que éste aporta:

- No ha sido necesario contratar un dominio ya que <http://127.0.0.1> actúa como hosting.
- Se puede trabajar con los archivos offline.
- Permite guardar copias de seguridad en el propio servidor, y ejecutarlas simultáneamente.
- Tiene una instalación sencilla mediante el software XAMPP elegido, el cual permite la navegación y la comunicación entre las distintas tecnologías empleadas.
- Facilita realizar todo tipo de pruebas sobre el sitio web sin temor a estropearlo, este supone uno de los principales beneficios ya que ha permitido optimizar, cambiar y reparar errores en la aplicación de forma rápida.
- Se puede acceder remotamente de forma segura sin ninguna vulnerabilidad.
- Se puede ampliar la capacidad del servidor en caso de necesitarlo.

9 DISEÑO DE LA APLICACIÓN

9.1 El usuario

La aplicación está dirigida a los alumnos de las carreras de ingeniería de la facultad de informática UCM, lo cual traza un perfil de usuario en su mayoría joven y con facilidad para las tecnologías.

Esto nos dirige, en esencia, a un público diestro en el manejo de aplicaciones web.

El usuario interesado en usar esta herramienta, estará matriculado y cursando la asignatura de IS, por lo que usará la aplicación todo el curso para desarrollar la documentación para la que ha sido implementada.

Una vez en la aplicación, el usuario podrá registrarse y formar equipo de trabajo con un grupo de compañeros, a continuación, puede empezar a cumplimentar los diferentes campos que se facilitan en la aplicación hasta completarlos y finalmente generar el documento entregable.

9.2 La apariencia

Se buscaba conseguir una apariencia sencilla, sin demasiado ruido en la imagen, para generar una experiencia de uso cómoda e intuitiva.

9.2.1 Colores y tipografía

Se ha elegido un esquema de colores fríos (blanco, azul y gris) para generar la navegación

Para la fuente se ha optado por la familia “Helvetica”, la cual no resulta estridente a la vista y puede leerse con facilidad.

9.2 Interfaz

La interfaz se ha construido diseñando una estructura estática que permanece constante durante la navegación y una serie de formularios que se van mostrando sobre esa estructura fija en la medida que el usuario va interactuando con la herramienta.

9.2.1 Estático

Esta estructura fija, cuenta con una cabecera con el nombre de la asignatura, una barra de navegación, el perfil del usuario activo el cual se puede editar haciendo clic encima y un botón para cerrar la sesión activa.

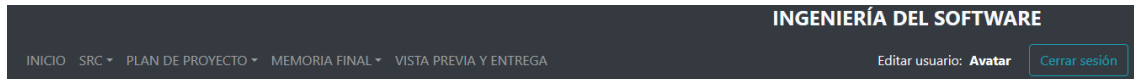


Figura20: Cabecera y barra de navegación.

La barra de navegación a su vez, tiene tres menús desplegables en cada uno de los tres documentos editables. Las opciones que muestran corresponden a cada una de las secciones en que pueden dividirse estos documentos. Se ha hecho esta división, para facilitar el reparto de tareas entre los componentes del equipo.

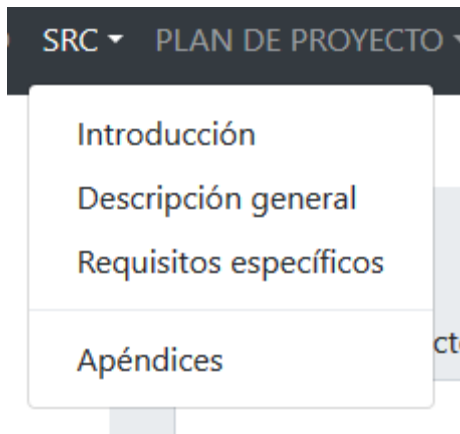


Figura 21: Menú desplegable, Especificación de Requisitos.

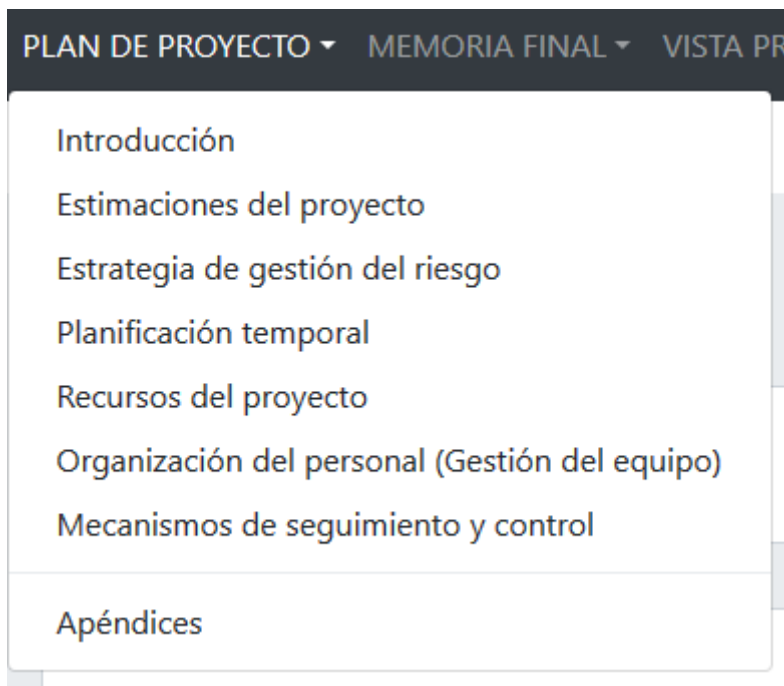


Figura 22: Menú desplegable, Plan de proyecto.

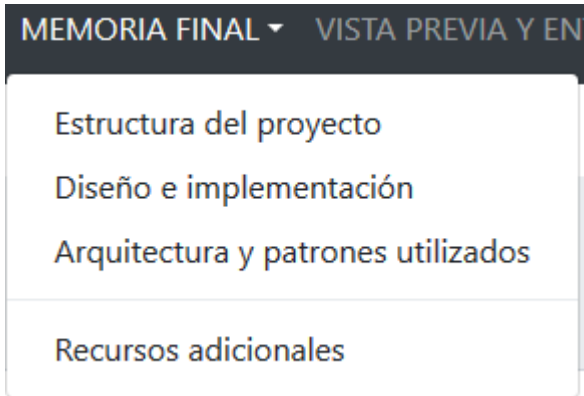


Figura 23: Menú desplegable Memoria final.

En la parte inferior, el pie de página, muestra un mensaje que identifica a la facultad de informática a la que pertenece la herramienta, y un segundo mensaje el cual, aunque se mantiene en el mismo lugar, irá cambiando para indicar dentro de que sección de la aplicación nos encontramos.

SRC:

ESPECIFICACION DE REQUISITOS
Facultad de informática - Universidad Complutense de Madrid

Figura 24: Pie de página, Especificación de Requisitos

Plan de proyecto:

PLAN DE PROYECTO
Facultad de informática - Universidad Complutense de Madrid

Figura 25: Pie de página, Plan de Proyecto

Memoria

final:

MEMORIA FINAL
Facultad de informática - Universidad Complutense de Madrid

Figura 26: Pie de página, Memoria final

Entrega:

ENTREGA
Facultad de informática - Universidad Complutense de Madrid

Figura 27: Pie de página, Entrega

Inicio:

INICIO
Facultad de informática - Universidad Complutense de Madrid

Figura 28: Pie de página, Inicio

Por último, indicar que la interfaz está capacitada para ser redimensionada en distintos dispositivos y pantallas.



Figura 29: Redimensión de interfaz



Figura 30: Barra de navegación comprimida

La barra de navegación en este caso se comprime en un “hamburguer button” y al accionarlo desvela las opciones en un desplegable que se ajusta al tamaño de la pantalla, véase figura 29.

9.2.2 Dinámico

A continuación, se muestra un diagrama con las relaciones que generan la navegación entre las páginas.

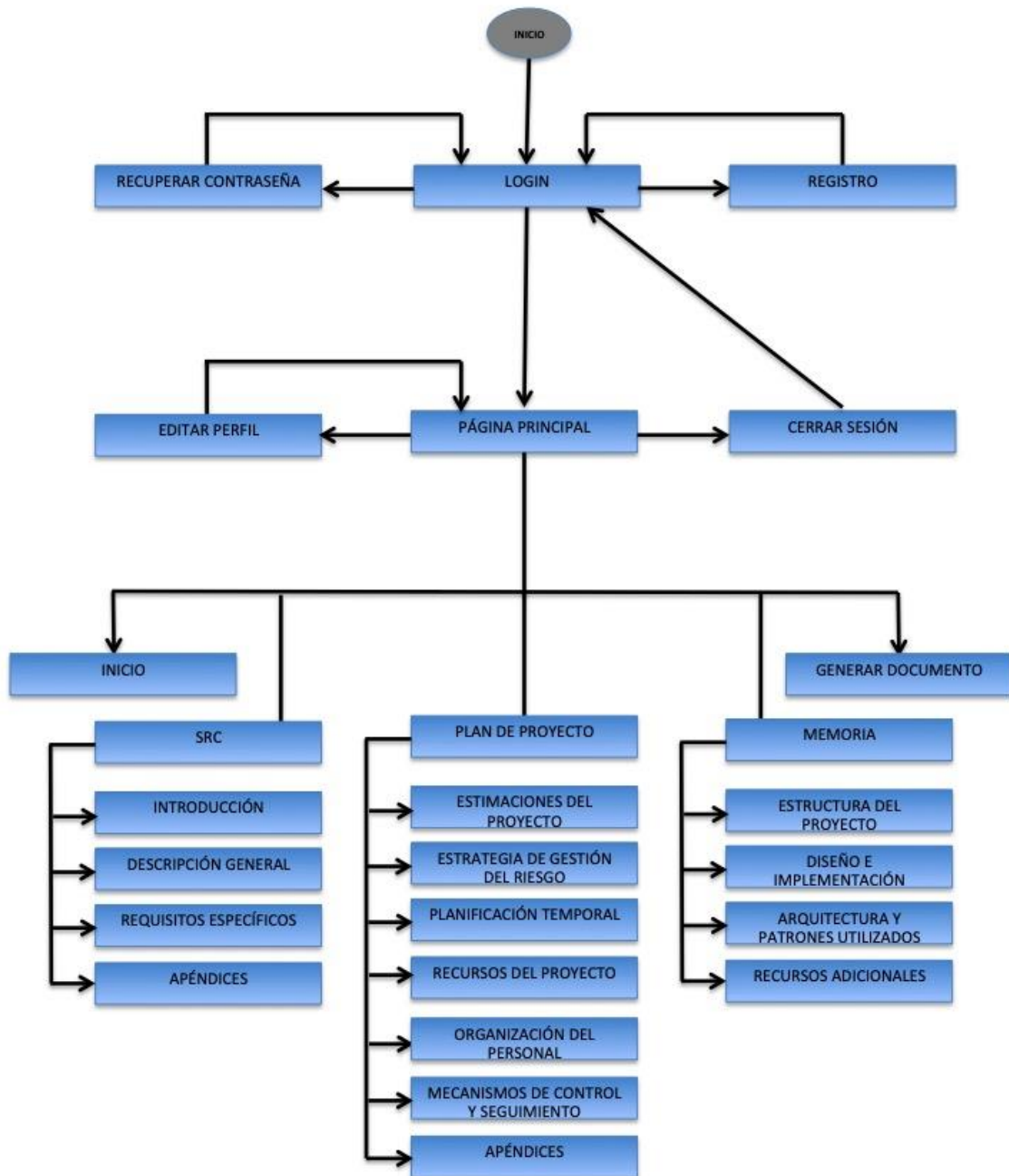


Figura 31: Diagrama de relaciones entre las páginas

Puede observarse dos niveles de uso:

1. Antes de acceder a la aplicación, se puede hacer el registro, recuperar contraseña o acceder al menú principal introduciendo usuario y contraseña.

INICIO DE SESIÓN

Usuario

Contraseña

Recordar sesión

[He olvidado la contraseña](#)

Entrar

o

Registrarse

Figura 32: Interfaz, Acceso a la aplicación

Recordar contraseña

Inserte usuario

Recordar

Volver

Figura 33: Interfaz, Recordar contraseña

REGISTRO

Correo de uso grupal:

Correo

Usuario de acceso:

Usuario

Contraseña:

Contraseña

Verifica contraseña introducida:

repetirContraseña

Registrarse

Volver

Figura 34: Interfaz, Formulario de registro

2. Una vez se haya efectuado el login, el usuario puede generar el equipo de trabajo, editar su perfil, cumplimentar los diferentes formularios de cada uno de los tres documentos y finalmente generar el documento para entregarlo a profesor.

PERFIL DE USUARIO

Cambiar correo de uso grupal:

Correo

Cambiar usuario de acceso:

Usuario

Cambiar contraseña:

Contraseña

Verifica contraseña introducida:

repetirContraseña

Guardar

Volver

Figura 35: Interfaz, Editar Perfil EQUIPO Y PROYECTO

Nombre del proyecto:

Miembros del grupo:

GUARDAR BORRADOR

Figura 36: Interfaz, Crear grupo de trabajo

Consúltense anexo I: Manual de uso, para visualizar todos los formularios accesibles desde este espacio.

10 TEST FUNCIONALES

Para probar el correcto funcionamiento de la herramienta desarrollada se han definido una serie de pruebas unitarias:

- Creación de nuevo usuario.
- Dar de alta un nuevo grupo de trabajo.
- Recuperación de contraseña.
- Edición del perfil de usuario.
- Cumplimentación de los apartados de la documentación.
- Subida de imágenes.
- Generación de documento final en formato PDF.
- Cerrar sesión.

Además de estas pruebas unitarias, se han realizado las siguientes pruebas con usuarios potenciales:

- Pruebas con usuarios externos a la asignatura de Ingeniería del Software, a los cuales se les ha facilitado una memoria de un curso pasado y se les ha pedido que generaran ese mismo documento desde la aplicación.
- Pruebas con varios alumnos actuales de la asignatura IS. Se les ha facilitado el manual de usuario para que intentarán generar el documento final completo en formato PDF.

En ambos casos, los usuarios han finalizado correctamente las pruebas, consiguiendo los objetivos que se habían marcado.

11 CONCLUSIONES

Con este Trabajo Fin de Grado se ha conseguido cumplir los objetivos que se proponían al comienzo del mismo. Se ha desarrollado una aplicación web, la cual permite a los alumnos de la asignatura de Ingeniería del Software, documentar los distintos entregables durante el transcurso de la materia.

Se ha conseguido establecer un formato homogéneo en las documentaciones que ahorra tiempo tanto a los alumnos como a los profesores, ya que:

- Los alumnos pueden reducir esfuerzos a la hora de documentar sus proyectos. Usando la aplicación tendrán una guía paso a paso con toda la información que deben facilitar al profesor para su evaluación, no siendo necesario consultar los apuntes de la asignatura. Lo que les permite centrarse exclusivamente en los contenidos sin preocuparse por la apariencia de sus documentos, ahorrando tiempo.
- Los profesores, al tener documentos con el mismo formato, tienen más facilidad a la hora de corregirlos.

Además, con un diseño sencillo, se ha conseguido que la aplicación pueda usarse intuitivamente, guardando la información de manera segura en una base de datos y pudiendo recuperarla en el momento que se desee.

El poder haber realizado pruebas con alumnos de la asignatura ha permitido confirmar que la herramienta cumple con las funcionalidades que se esperaban.

Se han puesto en práctica los conceptos y metodologías propios de la ingeniería del software a la hora de planificar y diseñar el proyecto, facilitando su implementación.

Por último, cabe destacar que se han tenido en cuenta todos los requisitos no funcionales que se plantearon al inicio, usando aquellas tecnologías de código abierto que se creyeron más convenientes durante la fase de análisis.

11.1 LÍNEAS FUTURAS

Se han cubierto los requisitos iniciales, tanto funcionales como no funcionales. La aplicación funciona con fluidez, y resulta útil para el fin que fue diseñada. No obstante, se ha detectado que la aplicación podría mejorar en una nueva versión incluyendo alguna de estas propuestas:

- Integración con el Campus virtual, para tener acceso a ella desde esta plataforma usada diariamente por los alumnos. Además, al estar integrada en el Campus virtual, las entregas se podrían realizar de manera automática desde el Campus. Esto facilitaría aún más a los alumnos la entrega de la documentación y el correcto seguimiento de la asignatura. Por ejemplo, se podrían implementar alertas ante entregas inminentes o ir informando al alumno del progreso de la asignatura.
- Añadir un modo profesor, para que pueda acceder y tener constancia de cómo están trabajando los alumnos, hacer precorrecciones u orientar la

tarea. Así en todo momento, tendría un control exhaustivo de la marcha de la asignatura.

- Integración con herramientas de Ingeniería del Software que permitan elaborar diagramas en la propia aplicación, en vez de realizar dichos diagramas en aplicaciones externas para luego subirlos.

12 CONCLUSIONS

This final project has reached the objectives proposed at the beginning. A web application has been developed, which enables students of Software Engineering subject to document the different deliverables during the course.

It has been set an homogenous format for the deliverables. So, students and teachers can save a lot of time. The reasons for this time saving are:

- The students can decrease their efforts for documenting their projects. Using the web application, they have a complete guideline with all information that have been delivered to the teachers for their evaluation. So, it is not necessary to check the notes of the course constantly. The students can focus just on the contents and, so, they can be unconcerned with the format and style of the documents.
- The teachers have more facility to evaluate the deliverables because all the documents have the same format.

Besides, the web application, based on a simple design, can be used intuitively. All the information is saved in a safe way in the database and can be recovered anytime.

The functional testing made with real students of Software Engineering confirmed that the web application achieves the desire functionalities.

In addition, the concepts and methodologies of Software Engineering have been implemented during the project, especially in the planning and design phases. These theories have eased the deployment of the application.

Finally, it should be pointed out that all non-functional requirements have been considered, using the most proper open-source technologies.

12.1 FUTURE WORK

As it noticed in the conclusions, the functional and non-functional requirements have been covered. The web application works fluently, and it is useful for the students. However, the application can be improved in a new version with these proposals:

- The integration with Virtual Campus should permit access to the web application from the campus which is used daily by the students. Besides, thanks to this integration, the deliverables may be sent automatically from the Campus. This would ease even more the delivery of the reports by the students and the correct follow-up of the subject. For example, some warnings could be implemented for upcoming deliverables or some information about the progress of the subject could be provided.
- The addition of a teacher mode. This improvement should permit the teacher access to the application and be aware of the students' progress. So, in every moment, teachers could have a detailed control of the progress of the course.

- The integration with other Software Engineering tools. So, for example, the own application should permit to elaborate some diagrams instead of doing these diagrams using external applications to attach them to the web application later.

BIBLIOGRAFÍA/ BIBLIOGRAPHY

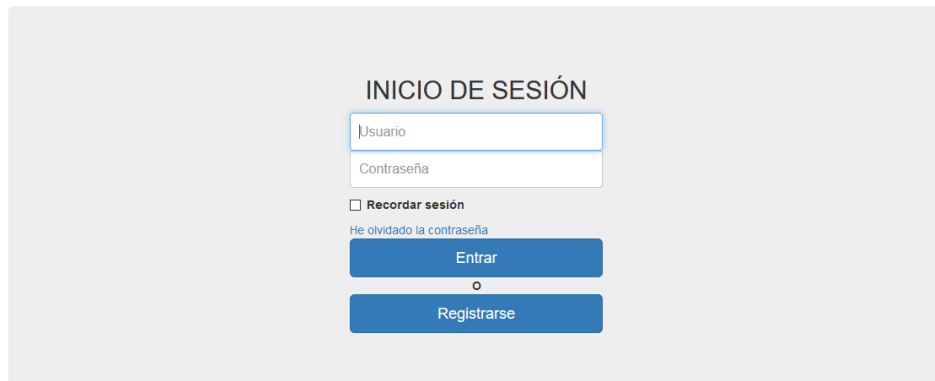
- Plan docente de la asignatura Ingeniería del sw descrito por su coordinador Héctor Gómez Gauchía disponible en la web FDI.
- Documentación, esquemas de trabajo y apuntes de la asignatura Ingeniería del SW facilitados por Antonio Sarasa Cabezuelo.
- Libro de texto Creación de un sitio web con PHP y MySQL 5ª edición de Jacobo Pavón Puertas y Ezequiel Llarena Borges. Obtenido en préstamo de la biblioteca FDI.
- Wikipedia para formación y descripciones técnicas <https://es.wikipedia.org/>
- PHP consultado en:
 - Manual PHP: <https://www.php.net/manual/es/index.php>
 - J&G Proyectos Web de Gian Carlos Molina consultado en youtube.com
- HTML
 - <https://codigofacilito.com/> de Javier Flores Herrera.
 - <https://desarrolloweb.com/> de Miguel Ángel Álvarez.
 - Páginas web HTML y hojas de estilo CSS de Bartolomé Sintés Marco, consultado online en <http://www.mclibre.org>
- CSS y BOOTSTRAP
 - Manual oficial de Bootstrap consultado online en <https://getbootstrap.com>
 - <https://uniwebsidad.com/libros/bootstrap-3>
- MySQL
 - <https://www.w3schools.com/html/>
- Apache
 - <http://www.apache.org/>
- XAMPP
 - <https://www.apachefriends.org/es/index.html>

ANEXO

ANEXO I: GUÍA DE USO

1 Acceso a la aplicación

Al abrir la aplicación se muestra el inicio de sesión,



INICIO DE SESIÓN

Usuario

Contraseña

Recordar sesión

[He olvidado la contraseña](#)

Entrar

o

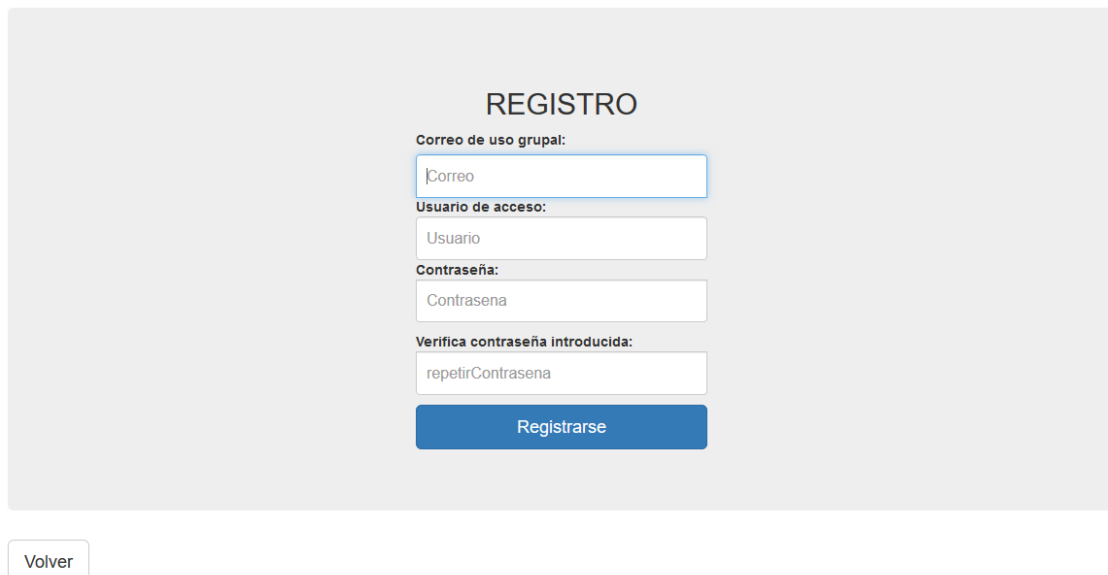
Registrarse

Figura 37: Guía de uso, inicio de sesión

introducir credenciales y hacer clic en Entrar para acceder.

2 Registro

En caso de no tener usuario y contraseña, hacer clic en Registrarse para darse de alta en la aplicación.



REGISTRO

Correo de uso grupal:

Correo

Usuario de acceso:

Usuario

Contraseña:

Contraseña

Verifica contraseña introducida:

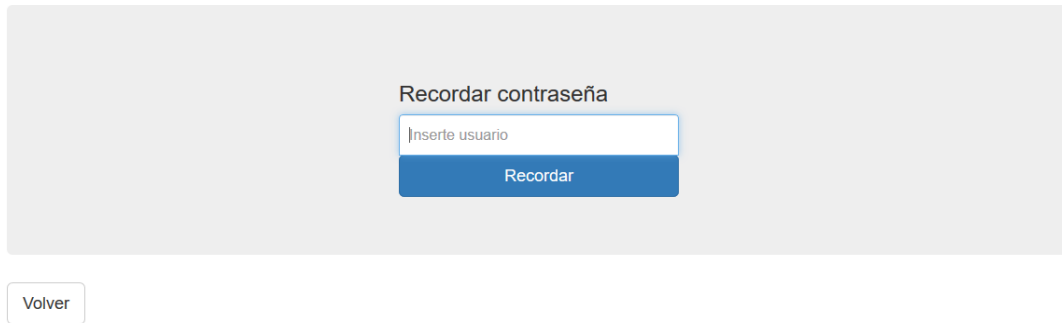
repetirContraseña

Registrarse

Volver

Figura 38: Guía de uso, Formulario de registro

o en recordar contraseña si se ha olvidado



Recordar contraseña

Inserte usuario

Recordar

Volver

Figura 39: Guía de uso, recordad contraseña

3 Cerrar sesión

Una vez dentro de la aplicación en la parte superior izquierda puede hallarse un botón azul para efectuar el cierre de sesión como puede verse en la siguiente figura 40.

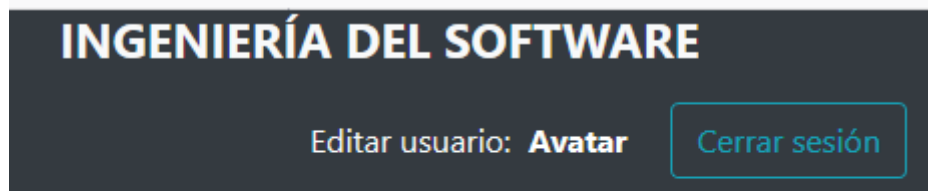


Figura 40: Barra de navegación, cerrar sesión

4 Página principal

Tras el acceso, se pueden configurar los datos del proyecto o editar el perfil haciendo clic en editar usuario visible en la barra de navegación de forma estática hasta cierre de sesión.



INGENIERIA DEL SOFTWARE

INICIO SRC PLAN DE PROYECTO MEMORIA FINAL VISTA PREVIA Y ENTREGA

Editar usuario: Avatar Cerrar sesión

EQUIPO Y PROYECTO

Nombre del proyecto:

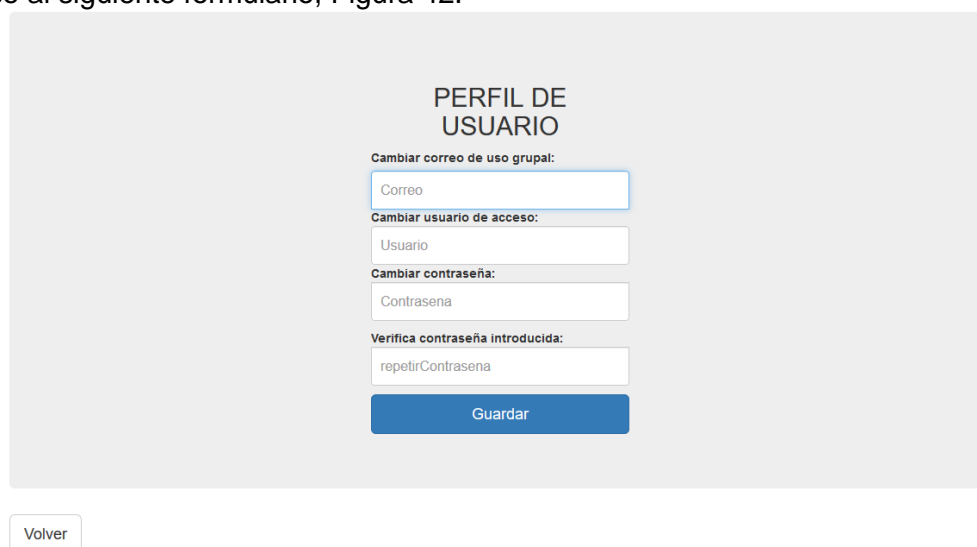
Miembros del grupo:

GUARDAR BORRADOR

Figura 41: Página principal

5 Editar perfil

Para editar el perfil dirigirse a Editar perfil en la esquina superior derecha, da acceso al siguiente formulario, Figura 42.



PERFIL DE USUARIO

Cambiar correo de uso grupal:

Cambiar usuario de acceso:

Cambiar contraseña:

Verifica contraseña introducida:

Guardar

Volver

Figura 42: Guía de uso, Editar Perfil

6 Complimentar Especificación de requisitos

Para empezar a rellenar la información relativa a Especificación de requisitos se puede acceder haciendo clic sobre SRC en la barra de tareas y seleccionando cualquiera de los formularios visibles en su menú desplegable, véase Figura 43.

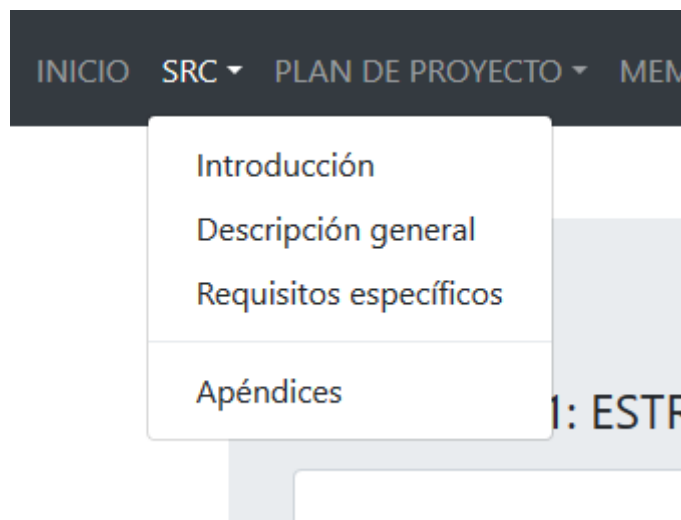


Figura 43: Barra de navegación, menú desplegable SRC

A continuación, empezar a cumplimentar los campos poniendo el cursor encima.

7 Mensajes de ayuda

Una vez dentro de cualquier formulario, al acercar el cursor tanto al área de texto como al título de la sección. aparecerá un mensaje que indica que tipo de contenido debe rellenarse en cada apartado, véase Figura 44.

1.2 Alcance

En este apartado se debe indicar la siguiente información:

- Identificar el producto software que se va a realizar por un nombre.
- Explicar a alto nivel qué es lo que va a hacer la aplicación y qué es lo que no va a hacer.
- Objetivos que se plantean cubrir.
- Describir la utilidad que tendrá el producto desarrollado, cómo va a beneficiar a la organización, qué ventajas tendrá,...

Figura 44: Mensaje de contenido

8 Complimentar Plan de proyecto

Al igual que ocurre en el SRC, encontraremos un menú desplegable desde la barra de tareas que nos lleva a cada una de las secciones de este documento.

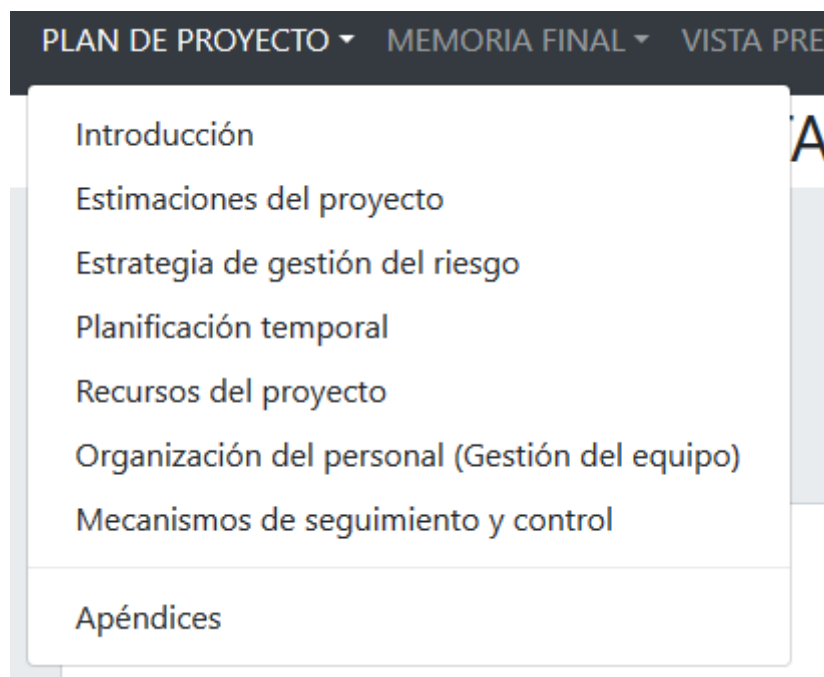


Figura 45: Barra de navegación, menú desplegable SRC

9 Cumplimentar Memoria final

Mostramos ahora el desplegable de memoria final, véase Figura 46.

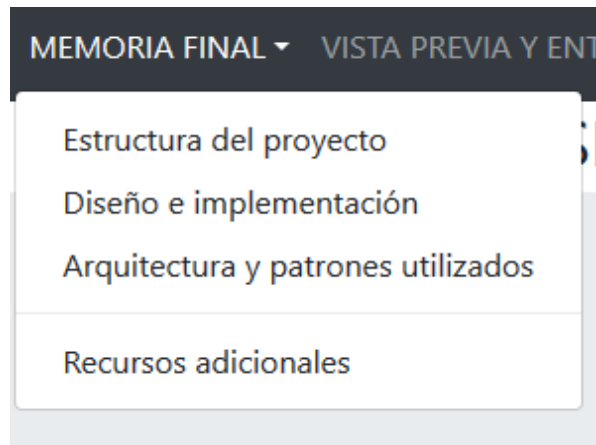


Figura 46: Barra de navegación, menú desplegable Memoria Final

9 Incluir imágenes

Para añadir diagramas u otras imágenes, se han añadido selectores de imágenes allí donde son necesarias. Para adjuntarlas, hacer clic sobre botón examinar y seleccionarla de su equipo. Véase Figura 47.

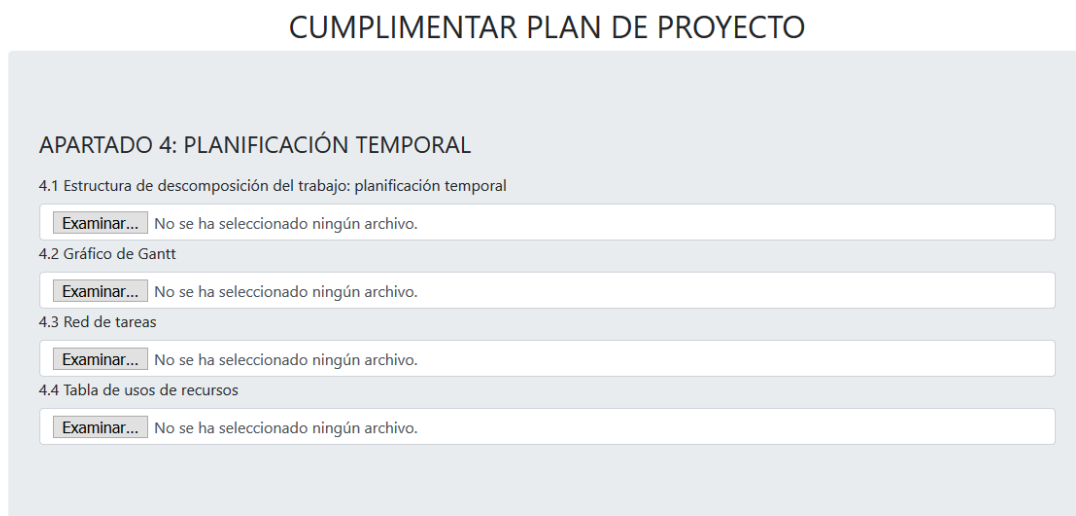


Figura 47: Añadir imagen

10 Guardar borrador

Cuando acabe de cumplimentar una sección, antes de pasar a la siguiente o de cerrar sesión, se deberá siempre salvar el trabajo realizado para conservarlo. Para guardar el borrador, pulsar el botón transversal Guardar borrador presente en la parte inferior en todas las secciones. Véase Figura 48.

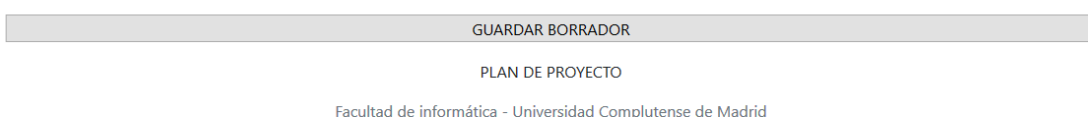


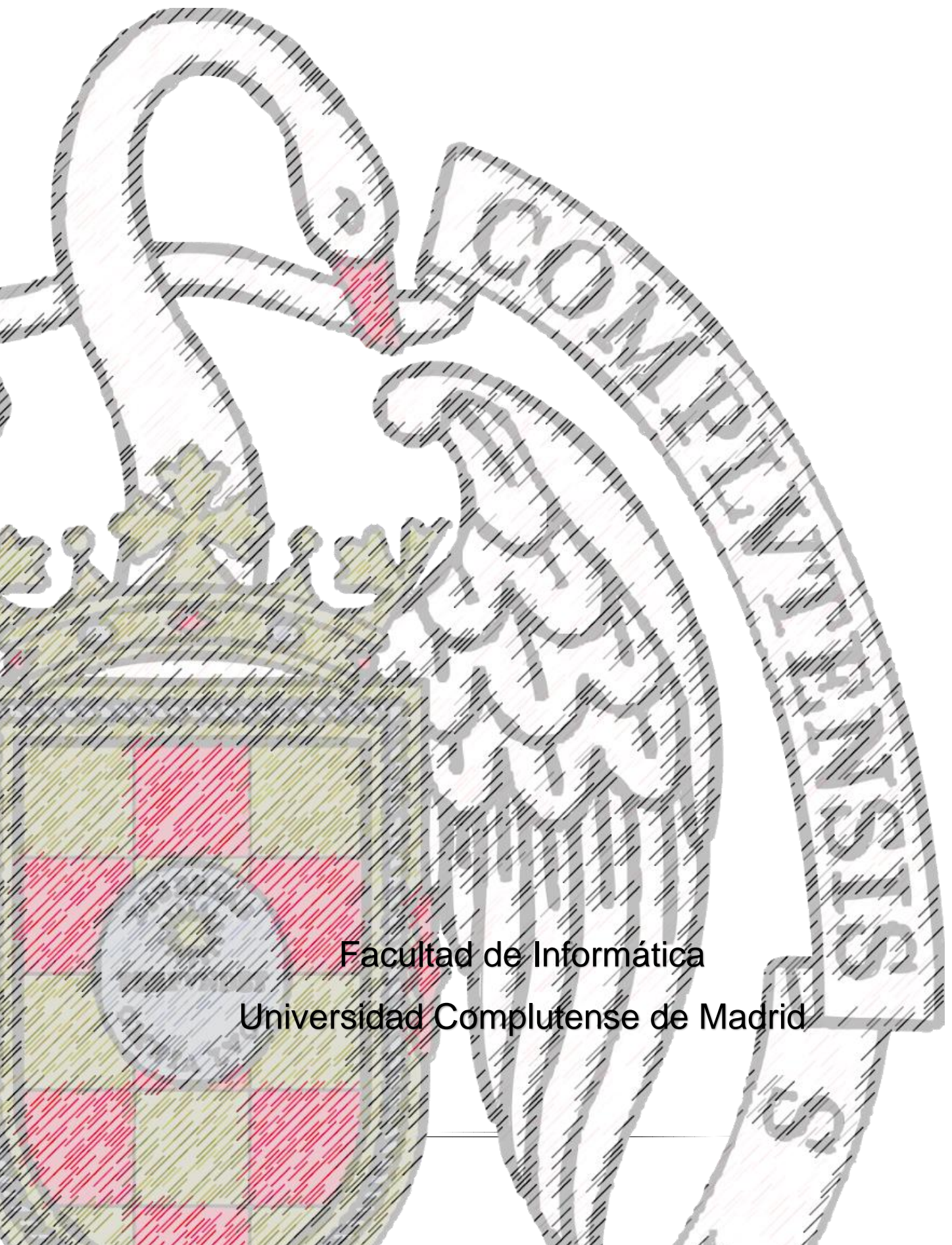
Figura 48: Guardar borrador

11 Hacer entrega

Para hacer la entrega del documento diríjase a Entrega en la barra de navegación, una vez ahí selecciones que documento desea generar, se obtendrá la versión PDF del documento generado.

The screenshot shows a web interface for generating a PDF document. At the top, there is a dark navigation bar with the text 'INGENIERÍA DEL SOFTWARE' on the right. Below the navigation bar, a breadcrumb trail reads 'INICIO > SRC > PLAN DE PROYECTO > MEMORIA FINAL > VISTA PREVIA Y ENTREGA'. On the right side of this bar, there are links for 'Editar usuario: Avatar' and 'Cerrar sesión'. The main content area is titled 'VISTA PREVIA Y GENERAR DOCUMENTO PDF'. Below the title, it says 'Seleccione documento que desea generar:'. There are three horizontal buttons stacked vertically, labeled 'SRC', 'PLAN DE PROYECTO', and 'MEMORIA FINAL'. At the bottom of the page, there is a footer with the text 'VISTA PREVIA Y GENERAR DOCUMENTO PDF' and 'Facultad de informática - Universidad Complutense de Madrid'.

Figura 49: Hacer entrega, generar PDF



Facultad de Informática
Universidad Complutense de Madrid