



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería
Informática**

**título del TFG
Documentación Técnica**



Presentado por nombre alumno
en Universidad de Burgos — 3 de junio de 2024
Tutor: nombre tutor

Índice general

Índice general	i
Índice de figuras	iii
Índice de tablas	iv
Apéndice A Plan de Proyecto Software	1
A.1. Introducción	1
A.2. Planificación temporal	1
A.3. Estudio de viabilidad	3
Apéndice B Especificación de Requisitos	5
B.1. Introducción	5
B.2. Objetivos generales	5
B.3. Catálogo de requisitos	5
B.4. Especificación de requisitos	5
Apéndice C Especificación de diseño	7
C.1. Introducción	7
C.2. Diseño de datos	7
C.3. Diseño procedimental	7
C.4. Diseño arquitectónico	7
Apéndice D Documentación técnica de programación	9
D.1. Introducción	9
D.2. Estructura de directorios	9
D.3. Manual del programador	9

D.4. Compilación, instalación y ejecución del proyecto	9
D.5. Pruebas del sistema	9
Apéndice E Documentación de usuario	11
E.1. Introducción	11
E.2. Requisitos de usuarios	11
E.3. Instalación	11
E.4. Manual del usuario	11
Apéndice F Anexo de sostenibilización curricular	13
F.1. Introducción	13
Bibliografía	15

Índice de figuras

Índice de tablas

B.1. CU-1 Nombre del caso de uso.	6
---	---

Apéndice A

Plan de Proyecto Software

A.1. Introducción

En un proyecto con un equipo completo de personas trabajando, podríamos definir la planificación de un proyecto software como una etapa que implica definir los objetivos y el alcance del proyecto, identificar los requisitos y recursos necesarios, establecer un cronograma con hitos clave, asignar tareas a los miembros del equipo y prever los posibles riesgos.

Dentro del contexto de este trabajo académico, se va a detallar en qué ha consistido la planificación temporal y se tratará de hacer un estudio de viabilidad realista.

A.2. Planificación temporal

En la memoria se ha indicado que se ha utilizado una metodología ágil de gestión de proyectos, con clara fundamentación en Scrum. Algunos de los aspectos más relevantes que han cubierto esta filosofía han sido:

- Desarrollo incremental a través de iteraciones llamadas *sprints*.
- Utilización de repositorio *Git* para solicitud de mejoras y para realizar un seguimiento de la evolución, con acceso para los tutores desde las fases iniciales.
- Control temporal a través de los *sprints*, valorando al inicio de cada uno cuál sería la duración adecuada basándonos en las tareas que se iban a realizar y el producto que se esperaba al final de esa iteración.
- Seguimiento de tareas en un panel *Kanban*.

Además de los *sprints* se diseñaron hitos relevantes denominados *Milestones* en *GitHub* que sirven como referencia de la producción realizada.

Sprints

A continuación se muestra un resumen de los *sprints* que han tenido lugar en las diferentes fases de este trabajo:

Sprint 0. Presentación inicial a tutores del TFG

- **Duración:** 19/10/2023 a 23/10/2023
- **Objetivo:** Presentar una idea de TFG a los posibles tutores.
- **Contexto:** Inicio del último curso del Grado de Ingeniería Informática. Se trata de contactar con los posibles tutores que controlen el desarrollo de un TFG. Este TFG estará basado en la idea de desarrollar una herramienta, escrita en Python, para realizar análisis financiero.
- **Tareas:**
 - Contactar con los posibles tutores por email.
 - Contactar por Teams con los tutores que muestren interés en el TFG e intercambiar primeras impresiones.
 - Determinar las herramientas necesarias para desarrollar un TFG.
 - Preguntar por consejos y opiniones que puedan ser relevantes.
 - Concertar cita para una futura reunión cuando se haya avanzado con los primeros pasos del TFG.

Sprint 1. Empezar TFG con selección de herramientas adecuadas

- **Duración:** 24/10/2023 a 08/11/2023
- **Objetivo:** Empezar el desarrollo del TFG y su correspondiente documentación. Hacerlo utilizando las herramientas adecuadas sugeridas por los tutores en el trascurso del sprint 0.
- **Contexto:** Tras haber mantenido una primera reunión con los tutores, se detectan ciertas necesidades en cuanto a la utilización de herramientas adecuadas para el desarrollo de un TFG.
- **Tareas:**

- Lectura completa de documentación de TFGs disponible en la plataforma UBUVirtual.
- Rellenar el formulario de "Oferta de TFG Grado de Informática Online". Buscar referencias bibliográficas teóricas más adecuadas.
- Utilización de Zube para gestionar un proyecto con metodología ágil: <https://zube.io/>. Para ello, hay que buscar la documentación adecuada que permita entender su utilización.
- Crear un repositorio en GitHub que permita llevar un seguimiento de las acciones realizadas.
- En el repositorio, añadir a los tutores e integrar con Zube.
- Determinar la mejor herramienta para la documentación del TFG. Elegir entre LaTeX y Word. Justificar la elección y documentar.
- Buscar un gestor de referencias bibliográficas: Zotero vs Mendeley.
- Concertar cita para una futura reunión cuando se haya avanzado con los primeros pasos del TFG.

Sprint 2. Mostrar primeras tablas y gráficos en servidor web

- **Duración:** 08/11/2023 a 29/11/2023
- **Objetivo:** Comenzar a desarrollar una aplicación web, mostrando información de precios cotizados y de gráficos. Almacenar información en una base de datos SQLite3, no en DataFrames ni en archivos .csv.
- **Contexto:** Una de las primeras recomendaciones de los tutores fue no utilizar archivos .csv para almacenar la información, por lo tanto, se va a utilizar un sistema de almacenamiento en base de datos. La idea general es que dichos datos se descargan con una API, se almacenan y, posteriormente, se procesan para mostrar la información relevante al usuario en archivos HTML.
- **Tareas:**
 - Preparar entorno local para utilizar una base de datos SQLite3. Instalar una GUI para la BD (no relevante para el usuario final).
 - Preparar clases (Descargador, Vista, Ticker y/o similares) que permitan automatizar la tarea de almacenar información en una base de datos.
 - Mostrar un primer producto - no necesariamente visualmente atractivo - en un servidor web con tablas extraídas de la BD.
 - Documentar, en el capítulo 4 de la memoria, las herramientas utilizadas.

- Concertar cita para una futura reunión con los tutores.

**Sprint 3. Mejorar lógica de web y proteger enlaces con registro.
Mostrar gráficos y dar primeros estilos.**

- **Duración:** 29/11/2023 a 13/12/2023
- **Objetivo:** Mejorar la lógica de acceso a la web. Proteger los enlaces no públicos mediante registro de usuarios. Mostrar gráficos y no sólo tablas en un estilo visualmente atractivo.
- **Contexto:** Una vez llevado el proyecto a "producción"(subido a un servidor web) hay que mejorar y asegurar los enlaces para que no sean accesibles en un mal uso de los mismos. Además, es necesario hacer un registro de usuarios que permita mostrar información relevante dependiendo del rol (invitado / registrado).
- **Tareas:**
 - Crear lógica de control de usuarios registrados.
 - Asignar permiso de acceso a enlaces dependiendo del tipo de usuario.
 - Crear portada y dar primeros estilos visuales.
 - Preparar un índice o zona de breadcrumbs de la página para mejorar la navegación.
 - Diseñar una vista que permita mostrar gráficos según el stock escogido.

A.3. Estudio de viabilidad

Viabilidad económica

Viabilidad legal

Apéndice B

Especificación de Requisitos

B.1. Introducción

Una muestra de cómo podría ser una tabla de casos de uso:

B.2. Objetivos generales

B.3. Catálogo de requisitos

B.4. Especificación de requisitos

CU-1	Ejemplo de caso de uso
Versión	1.0
Autor	Alumno
Requisitos asociados	RF-xx, RF-xx
Descripción	La descripción del CU
Precondición	Precondiciones (podría haber más de una)
Acciones	<ol style="list-style-type: none"> 1. Pasos del CU 2. Pasos del CU (añadir tantos como sean necesarios)
Postcondición	Postcondiciones (podría haber más de una)
Excepciones	Excepciones
Importancia	Alta o Media o Baja...

Tabla B.1: CU-1 Nombre del caso de uso.

Apéndice C

Especificación de diseño

- C.1. Introducción
- C.2. Diseño de datos
- C.3. Diseño procedimental
- C.4. Diseño arquitectónico

Apéndice D

Documentación técnica de programación

- D.1. Introducción
- D.2. Estructura de directorios
- D.3. Manual del programador
- D.4. Compilación, instalación y ejecución del proyecto
- D.5. Pruebas del sistema

Apéndice E

Documentación de usuario

- E.1. Introducción
- E.2. Requisitos de usuarios
- E.3. Instalación
- E.4. Manual del usuario

Apéndice F

Anexo de sostenibilización curricular

F.1. Introducción

Este anexo incluirá una reflexión personal del alumnado sobre los aspectos de la sostenibilidad que se abordan en el trabajo. Se pueden incluir tantas subsecciones como sean necesarias con la intención de explicar las competencias de sostenibilidad adquiridas durante el alumnado y aplicadas al Trabajo de Fin de Grado.

Más información en el documento de la CRUE https://www.crue.org/wp-content/uploads/2020/02/Directrices_Sostenibilidad_Crue2012.pdf.

Este anexo tendrá una extensión comprendida entre 600 y 800 palabras.

Bibliografía
