



UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



TFG del Grado en Ingeniería  
Informática

Aplicación de gestión del PDI  
de un área de la UBU



Presentado por Ignacio Dávila García  
en Universidad de Burgos — 22 de abril  
de 2023

Tutores: Álvaro Arnaiz González y Carlos  
Pardo Aguilar







UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



D. Álvar Arnaiz González, profesor del departamento de Ingeniería Informática, área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. Ignacio Dávila García, con DNI 71755022J, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado *Aplicación de gestión del PDI de un área de la UBU*.

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

En Burgos, 22 de abril de 2023

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

D. Álvar Arnaiz González

D. Carlos Pardo Aguilar





## Resumen

En este primer apartado se hace una **breve** presentación del tema que se aborda en el proyecto.

## Descriptores

Palabras separadas por comas que identifiquen el contenido del proyecto Ej: servidor web, buscador de vuelos, android ...

## **Abstract**

A **brief** presentation of the topic addressed in the project.

## **Keywords**

keywords separated by commas.



---

# Índice general

---

Índice general	iii
Índice de figuras	iv
Índice de tablas	v
1. Introducción	1
2. Objetivos del proyecto	3
3. Conceptos teóricos	5
3.1. Secciones . . . . .	5
3.2. Referencias . . . . .	5
3.3. Imágenes . . . . .	6
3.4. Listas de ítems . . . . .	6
3.5. Tablas . . . . .	7
4. Técnicas y herramientas	9
4.1. Técnicas . . . . .	9
4.2. Herramientas . . . . .	12
5. Aspectos relevantes del desarrollo del proyecto	17
6. Trabajos relacionados	19
7. Conclusiones y Líneas de trabajo futuras	21
Bibliografía	23

---

# Índice de figuras

---

3.1. Autómata para una expresión vacía . . . . .	6
4.1. <i>Proceso seguido en Scrum [2]</i> . . . . .	11
4.2. <i>Explicación de los elementos del proceso [2]</i> . . . . .	11

---

# Índice de tablas

---

3.1. Herramientas y tecnologías utilizadas en cada parte del proyecto	7
---	---



---

# 1. Introducción

---

El presente trabajo de fin de grado tiene como objetivo la creación de una aplicación web para la gestión del personal docente e investigador de la Universidad de Burgos utilizando el lenguaje de programación Python y el *framework* Flask. La gestión del personal docente e investigador es una tarea crítica y compleja para cualquier institución de educación superior, ya que implica el manejo de información confidencial y la coordinación de múltiples actividades administrativas. Por lo tanto, contar con una herramienta informática eficiente y segura para gestionar esta tarea es esencial.

En este contexto, se ha desarrollado una aplicación web que permite la gestión de información de los docentes e investigadores, así como el acceso a diferentes funcionalidades de forma sencilla y eficiente. Para lograr este objetivo, se ha utilizado el *framework* Flask, una herramienta de código abierto que permite crear aplicaciones web de manera rápida y eficiente, utilizando el lenguaje de programación Python.

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



---

## **2. Objetivos del proyecto**

---

Este apartado explica de forma precisa y concisa cuales son los objetivos que se persiguen con la realización del proyecto. Se puede distinguir entre los objetivos marcados por los requisitos del software a construir y los objetivos de carácter técnico que plantea a la hora de llevar a la práctica el proyecto.





---

## 3. Conceptos teóricos

---

En aquellos proyectos que necesiten para su comprensión y desarrollo de unos conceptos teóricos de una determinada materia o de un determinado dominio de conocimiento, debe existir un apartado que sintetice dichos conceptos.

Algunos conceptos teóricos de  $\text{\LaTeX}$ <sup>1</sup>.

### 3.1. Secciones

Las secciones se incluyen con el comando `section`.

#### Subsecciones

Además de secciones tenemos subsecciones.

#### Subsubsecciones

Y subsecciones.

### 3.2. Referencias

Las referencias se incluyen en el texto usando `cite` [6]. Para citar webs, artículos o libros [3].

---

<sup>1</sup>Créditos a los proyectos de Álvaro López Cantero: Configurador de Presupuestos y Roberto Izquierdo Amo: PLQuiz

### 3.3. Imágenes

Se pueden incluir imágenes con los comandos standard de  $\text{\LaTeX}$ , pero esta plantilla dispone de comandos propios como por ejemplo el siguiente:



Figura 3.1: Autómata para una expresión vacía

### 3.4. Listas de items

Existen tres posibilidades:

- primer item.
- segundo item.

1. primer item.
2. segundo item.

**Primer item** más información sobre el primer item.

**Segundo item** más información sobre el segundo item.

■

Herramientas	App	AngularJS	API REST	BD	Memoria
HTML5		X			
CSS3		X			
BOOTSTRAP		X			
JavaScript		X			
AngularJS		X			
Bower		X			
PHP			X		
Karma + Jasmine		X			
Slim framework			X		
Idiorm			X		
Composer			X		
JSON		X	X		
PhpStorm		X	X		
MySQL				X	
PhpMyAdmin				X	
Git + BitBucket		X	X	X	X
MikTeX					X
TeXMaker					X
Astah					X
Balsamiq Mockups		X			
VersionOne		X	X	X	X

Tabla 3.1: Herramientas y tecnologías utilizadas en cada parte del proyecto

### 3.5. Tablas

Igualmente se pueden usar los comandos específicos de  $\text{\LaTeX}$  o bien usar alguno de los comandos de la plantilla.



---

## 4. Técnicas y herramientas

---

### 4.1. Técnicas

#### Scrum

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

#### Elementos

- **Pila del producto o *product backlog*:** Es una lista de requisitos desde el punto de vista del cliente que comienza con una visión inicial del producto, pero que se irá incrementando a lo largo del desarrollo.
- **Pila del sprint o *sprint backlog*:** Conjunto de requisitos o tareas desde el punto de vista del desarrollador que el equipo de desarrollo del proyecto espera realizar durante el *sprint*.
- **Incremento:** El incremento es el resultado del desarrollo de un *sprint*, es una parte terminada y probada.
- **Gráfico de avance o *burn-down chart*:** Sirve para representar tanto el trabajo pendiente como el trabajo realizado durante un *sprint* para ver la velocidad a la que se están completando las tareas. De este modo, se puede saber si se va dentro de la planificación o no.
- **Gráfico de producto o *burn-up chart*:** Gráfico utilizado para ver el trabajo realizado respecto al total.

- **Gráfico de velocidad o *velocity chart*:** Muestra la velocidad de trabajo durante el *sprint*. Es útil para ir ajustando la previsión de tiempo en futuros *sprints*.

## Roles

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

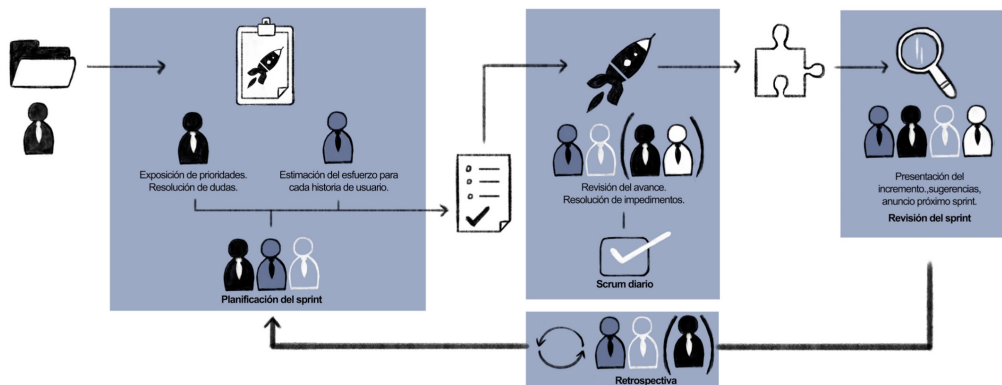
- ***Scrum master*:** Persona con conocimientos avanzados en la metodología Scrum que se encarga de que se cumpla correctamente el procedimiento. También es el encargado de dar formación y asesorar al resto de personal sobre Scrum.
- **Propietario del producto o *product owner*:** Es la persona que hace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

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

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

Con la pila del producto creada, todo el personal se reúne para planificar el *sprint*. Estas reuniones son las que gestionan la pila del *sprint*. Después de esta reunión, el equipo de desarrollo comienza a trabajar en las tareas que se ha comprometido para completarlas en el tiempo acordado. Cada día de trabajo se realiza una pequeña reunión de aproximadamente 15 minutos donde se expone el trabajo realizado, el trabajo que se va a hacer en el día, las dudas, etc., esta reunión sirve para ir actualizando la pila del *sprint*.

Al final del *sprint*, todo el personal se vuelve a reunir para presentar el incremento realizado y ver posibles cambios. Además, se fija el anuncio del

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

próximo *sprint* que comenzará de nuevo con una reunión de planificación.

Durante este proyecto se ha seguido una metodología Scrum con algunos cambios ya que no se hacían reuniones diarias ni los roles estaban tan marcados, pero la forma de trabajar era la misma con planificaciones y revisiones de *sprints*.

## Web

La web (*World Wide Web*) es un sistema que permite la transmisión y acceso a datos y documentos a través de internet utilizando el protocolo HTTP para la comunicación y el lenguaje HTML para la representación de las páginas web. [7]

La tecnología web es una de las que más desarrollo ha sufrido en los últimos años. Esto ha provocado un cambio completo en nuestra forma de vida,

afectando a aspectos como la búsqueda de información por Internet, realizar compras online, hacer trámites sin necesidad de acudir a una oficina, etc.,

## 4.2. Herramientas

### GitHub

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

### ZenHub

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

### Python

Python es un popular lenguaje de programación que puede ser utilizado en diferentes ámbitos, entre ellos el desarrollo web. Algunas de las razones para utilizar este lenguaje es su facilidad de uso, la posibilidad de orientación a objetos y la amplia variedad de bibliotecas y frameworks, como puede ser Flask.[4]

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

### Flask

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



## JavaScript

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

## HTML

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

## CSS

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

## Bibliotecas / *plugins*

- **SQLAlchemy**

Biblioteca de SQL para Python que incluye una versión para el *framework* Flask que permite su completa integración. SQLAlchemy contiene un conjunto de herramientas SQL que permiten el mapeo objeto-relacional, *Object Relational Mapper (ORM)*. Esto permite que con un conjunto de expresiones requeridas por la biblioteca, se creen clases de objetos en Python que serán transformadas a tablas de una forma prácticamente abstracta para el desarrollador. Además, al estar los objetos vinculados a registros de una tabla SQL, cuenta con métodos para realizar distintas operaciones SQL como búsquedas, inserciones, eliminaciones, etc., sin tener que escribir sentencias SQL.[5]

- **Grid.js**

Grid.js es un *plugin* de JavaScript que ayuda con el tratamiento de

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

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

Algunas de las razones son la siguientes:

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

## Software

### ■ Pincel Project

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

### ■ Draw.io

Aplicación web que permite la creación de diagramas utilizando diferentes figuras geométricas. Además cuenta con objetos ya creados para crear diagramas utilizando UML, diagramas entidad-relación, diagramas relacionales, etc. Esta aplicación permite una personalización avanzada por lo que es muy útil para crear cualquier tipo de

diagrama. Además, al ser una aplicación web no requiere de ningún tipo de instalación aunque si que cuenta con una versión de escritorio para Windows, Linux, macOS y Chrome OS.

- **PyCharm**

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

PyCharm es uno de los IDE más completos y avanzados para el desarrollo con Python, ya que cuenta con una gran cantidad de características que facilitan el trabajo y ayudan a ahorrar tiempo. Además, a la hora de crear proyectos se puede indicar el tipo de *framework* a utilizar, lo que hará que cargue de forma automática todos los paquetes necesarios y creará una versión simple del proyecto desde la que empezar a trabajar. Esto ahorra mucho tiempo al no tener que ir buscando e instalando los paquetes necesarios desde distintos repositorios.



---

## 5. Aspectos relevantes del desarrollo del proyecto

---

Este apartado pretende recoger los aspectos más interesantes del desarrollo del proyecto, comentados por los autores del mismo. Debe incluir desde la exposición del ciclo de vida utilizado, hasta los detalles de mayor relevancia de las fases de análisis, diseño e implementación. Se busca que no sea una mera operación de copiar y pegar diagramas y extractos del código fuente, sino que realmente se justifiquen los caminos de solución que se han tomado, especialmente aquellos que no sean triviales. Puede ser el lugar más adecuado para documentar los aspectos más interesantes del diseño y de la implementación, con un mayor hincapié en aspectos tales como el tipo de arquitectura elegido, los índices de las tablas de la base de datos, normalización y desnormalización, distribución en ficheros<sup>3</sup>, reglas de negocio dentro de las bases de datos (EDVHV GH GDWRV DFWLYDV), aspectos de desarrollo relacionados con el WWW... Este apartado, debe convertirse en el resumen de la experiencia práctica del proyecto, y por sí mismo justifica que la memoria se convierta en un documento útil, fuente de referencia para los autores, los tutores y futuros alumnos.



---

## 6. Trabajos relacionados

---

Este apartado sería parecido a un estado del arte de una tesis o tesina. En un trabajo final grado no parece obligada su presencia, aunque se puede dejar a juicio del tutor el incluir un pequeño resumen comentado de los trabajos y proyectos ya realizados en el campo del proyecto en curso.





---

## **7. Conclusiones y Líneas de trabajo futuras**

---

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



---

## Bibliografía

---

- [1] Amazon. ¿qué es javascript?, 2023. <https://aws.amazon.com/es/what-is/javascript/>.
- [2] Claudia Ruata Juan Palacio. *Scrum Manager. Gestión de Proyectos*. 2011.
- [3] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.
- [4] Phyton. Documentación de python, 2023. <https://docs.python.org/3/tutorial/index.html>.
- [5] SQLAlchemy. Documentación de sqlalchemy, 2023. <https://docs.sqlalchemy.org/en/20/intro.html>.
- [6] Wikipedia. Latex — wikipedia, la enciclopedia libre, 2015. [Internet; descargado 30-septiembre-2015].
- [7] Wikipedia. World wide web — wikipedia, la enciclopedia libre, 2023. [Internet; descargado 3-abril-2023].
- [8] Wikipedia contributors. Github — Wikipedia, the free encyclopedia, 2023. [Online; accessed 2-April-2023; <https://en.wikipedia.org/w/index.php?title=GitHub&oldid=1147528749>].