



UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería  
Informática**

**título del TFG**



Presentado por Nombre del alumno  
en Universidad de Burgos — 22 de enero  
de 2020

Tutor: nombre tutor







UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



D. nombre tutor, profesor del departamento de nombre departamento, área de nombre área.

Expone:

Que el alumno D. Nombre del alumno, con DNI dni, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado título de TFG.

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 enero de 2020

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

D. nombre tutor

D. nombre co-tutor





## 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   |
| Introducción                                    | 1   |
| Objetivos del proyecto                          | 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   |
| Técnicas y herramientas                         | 9   |
| 4.1. Técnicas . . . . .                         | 9   |
| 4.2. Herramientas . . . . .                     | 11  |
| Aspectos relevantes del desarrollo del proyecto | 13  |
| Trabajos relacionados                           | 15  |
| Conclusiones y Líneas de trabajo futuras        | 17  |
| Bibliografía                                    | 19  |

---

# Índice de figuras

---

|  |   |
|--|---|
| 3.1. Autómata para una expresión vacía . . . . . | 6 |
|--|---|

---

# Índice de tablas

---

|   |   |
|---|---|
| 3.1. Herramientas y tecnologías utilizadas en cada parte del proyecto | 8 |
|---|---|



---

# Introducción

---

Descripción del contenido del trabajo y del estructura de la memoria y del resto de materiales entregados.



---

# 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.

## Objetivos Principales

Son aquellos marcados por los requisitos que debe tener el software final. Este proyecto trata de:

- Permitir a los usuarios seleccionar en el mapa distintos puntos:
  - Mediante la búsqueda de la dirección deseada.
  - Mediante la elección sobre el mapa de un punto cualquiera.
- Visualizar los puntos seleccionados: información acerca de la ubicación, marcador en el mapa.
- Poder añadir puntos al recorrido cuando el usuario lo desee.
- Calcular el recorrido mas corto para el usuario de manera que pase por todos los puntos seleccionados.
- La aplicación final debe ser útil, y eficaz para cualquier usuario.

## Objetivos Técnicos

## Objetivos a nivel personal





---

# 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 L<sup>A</sup>T<sub>E</sub>X<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` [2]. Para citar webs, artículos o libros [1].

---

<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.

▪

### 3.5. Tablas

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

| 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

---

# Técnicas y herramientas

---

Esta parte de la memoria tiene como objetivo presentar las técnicas metodológicas y las herramientas de desarrollo que se han utilizado para llevar a cabo el proyecto. Si se han estudiado diferentes alternativas de metodologías, herramientas, bibliotecas se puede hacer un resumen de los aspectos más destacados de cada alternativa, incluyendo comparativas entre las distintas opciones y una justificación de las elecciones realizadas. No se pretende que este apartado se convierta en un capítulo de un libro dedicado a cada una de las alternativas, sino comentar los aspectos más destacados de cada opción, con un repaso somero a los fundamentos esenciales y referencias bibliográficas para que el lector pueda ampliar su conocimiento sobre el tema.

## 4.1. Técnicas

### Scrum

Scrum es una metodología de desarrollo ágil la cual proporciona un marco de trabajo y desarrollo de productos. No es un solo proceso, si no que en esta metodología se aplican un conjunto de buenas practicas y procesos para que el producto final sea de la mejor calidad posible. El principal elemento del Scrum consiste en los llamados Sprints, que son ciclos de trabajo de una semana de duración, período el cual sirve para producir un desarrollo o mejora del producto final. Estos sprints están marcados por dos reuniones:

- Planificación: en ella se presentan los requisitos o avances que tiene que cumplir el proyecto., a la vez que se estiman los tiempos y se realiza la planificación.

- Reunión de revisión: entrega de los requisitos acordados en la reunión de planificación y el equipo analiza el sprint.

El uso de esta metodología, junto con las diversas reuniones que se realizan, permite que el producto final sea de mejor calidad ya que en todo momento se conoce el feedback del cliente y se pueden realizar distintos cambios incrementales a medida que avanza el proyecto. Es una metodología pensada para el trabajo en equipo, por lo que en este proyecto se han mantenido las bases pero se ha adaptado la forma de trabajar, de manera que las reuniones han sido entre los tutores y el alumno y la fecha de la reunión de planificación del Sprint coincide con la fecha de revisión del sprint anterior.

## **Control de versiones**

El control de versiones hace referencia a un sistema que permite gestionar y registrar los cambios realizados en un proyecto a lo largo del desarrollo del mismo.

### **GitHub**

Para el control de versiones de este proyecto he utilizado GitHub, que es un repositorio en línea que emplea Git. De esta manera tenemos acceso en línea a los diferentes cambios de nuestro proyecto. Git maneja los diferentes archivos del proyecto como un conjunto de copias instantáneas.

### **Tablero**

El tablero, o mejor dicho la herramienta de ZenHub nos permite dividir el desarrollo total del sprint en pequeñas issues o tareas. Estas tareas podemos ponerlas en un tablero, en función de las que sean prioritarias, se estén desarrollando, quien las hace.... u otras clasificaciones que queramos darle (normalmente en función de la prioridad) Normalmente, el uso del tablero es más útil cuando el desarrollo se realiza en equipo, ya que se pueden asignar a distintas personas. Individualmente también resulta útil ya que ayuda a mantener un control y un orden.

## 4.2. Herramientas

### Flask

#### Entorno de desarrollo

Para el desarrollo del proyecto, se valoraron inicialmente dos editores:

- Visual Studio Code
- PyCharm

Finalmente se escogió PyCharm, ya que es un editor pensado para el desarrollo de código escrito en Python y permite gestionar las librerías y los directorios de manera más fácil.

### Google API

En este proyecto, para la selección de los distintos puntos a recorrer por parte del usuario he empleado los mapas de Google. Google proporciona una plataforma para los desarrolladores en la que se puede encontrar una gran cantidad de documentación<sup>2</sup>. Para poder integrar en la aplicación web tanto los mapas como las diferentes funcionalidades que ofrecen debemos adquirir lo que llama <sup>3</sup>, la cual se trata de una clave "privada" para tener acceso a los servicios de su API. Para su obtención es necesario incluir tus datos bancarios, ya que durante el primer año el uso de los servicios es gratis y luego comienza a pagarse a partir de un determinado número de peticiones. Una vez obtenida la clave, puede restringirse su uso para ciertas direcciones o dominios, de modo que puedes mantener el control de quien la usa. Además, no vale con conseguir una clave y ya esta, si no que para usar los diferentes servicios que proporciona Google hay que activar diferentes APIs. Las APIs que se usan en este proyecto son:

- **Maps JavaScript API:** Se utiliza en el cliente, de manera que se muestra el mapa al cargar la página y permite realizar diferentes acciones en el, como buscar, seleccionar puntos o moverte a través de él. Algunas de estas acciones implican el uso de algunas de funcionalidades que proporcionan las APIs explicadas a continuación.

---

<sup>2</sup><https://cloud.google.com/maps-platform/>

<sup>3</sup><https://developers-dot-devsite-v2-prod.appspot.com/maps/documentation/geocoding/get-api-key>

- **Geocoding API:** este API consta de dos elementos:
  - Geocodificación: Consiste en convertir direcciones en coordenadas.
  - Geocodificación inversa: Consiste en convertir coordenadas en una dirección legible.
- **Places API:** este servicio devuelve como resultado de la petición toda la información acerca de un lugar.
- **Distance Matrix API:** este API proporciona tanto la distancia como el tiempo de viaje que hay entre una lista de orígenes y una de destinos. En otras palabras, como resultado devuelve la distancia y tiempo que hay entre cada origen y cada destino.
- **Directions API:** como respuesta nos devuelve las indicaciones a seguir para llegar desde el punto de inicio hasta el punto de destino. Además, puede configurarse para diferentes modos de transporte, diferentes momentos de salida o llegada.

## JavaScript para dibujar redes

networkx comparativa distintas herramientas dibujo javascript

## Documentación



---

## 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.



---

## 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.



---

## **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] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.
- [2] Wikipedia. Latex — wikipedia, la enciclopedia libre, 2015. [Internet; descargado 30-septiembre-2015].