



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 — 25 de mayo  
de 2021

Tutor: nombre tutor



---

# Índice general

---

<b>Índice general</b>	<b>i</b>
<b>Índice de figuras</b>	<b>iii</b>
<b>Índice de tablas</b>	<b>iv</b>
<b>Apéndice A Plan de Proyecto Software</b>	<b>1</b>
A.1. Introducción . . . . .	1
A.2. Planificación temporal . . . . .	1
A.3. Estudio de viabilidad . . . . .	1
<b>Apéndice B Especificación de Requisitos</b>	<b>3</b>
B.1. Introducción . . . . .	3
B.2. Objetivos generales . . . . .	3
B.3. Catalogo de requisitos . . . . .	3
B.4. Especificación de requisitos . . . . .	5
<b>Apéndice C Especificación de diseño</b>	<b>7</b>
C.1. Introducción . . . . .	7
C.2. Diseño de datos . . . . .	7
C.3. Diseño procedimental . . . . .	7
C.4. Diseño arquitectónico . . . . .	7
<b>Apéndice D Documentación técnica de programación</b>	<b>9</b>
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
<b>Apéndice E Documentación de usuario</b>	<b>11</b>
E.1. Introducción . . . . .	11
E.2. Requisitos de usuarios . . . . .	11
E.3. Instalación . . . . .	11
E.4. Manual del usuario . . . . .	11

---

## Índice de figuras

---

---

## Índice de tablas

---

## *Apéndice A*

---

# **Plan de Proyecto Software**

---

**A.1. Introducción**

**A.2. Planificación temporal**

**A.3. Estudio de viabilidad**

Viabilidad económica

Viabilidad legal





## *Apéndice B*

---

# Especificación de Requisitos

---

### B.1. Introducción

### B.2. Objetivos generales

El desarrollo del proyecto busca lograr los siguientes objetivos:

- Desarrollar un videojuego que implemente las mecánicas definidas en el fichero de diseño del juego.
- Ofrecer versiones del producto final (el videojuego) para Windows, Linux y WebGL.
- Ofrecer un juego controlable tanto con teclado y ratón como con mando.

### B.3. Catalogo de requisitos

#### Requisitos funcionales

Los requisitos funcionales especificarán cual es el funcionamiento que se espera del producto. Se concretarán a continuación.

- **RF-1 Gestión de menús:** El jugador debe poder navegar por los menús.
  - **RF-1.1 Navegación entre pantallas:** Las pantallas deben de permitir navegar a otras pantallas (de manera directa o indirecta).

- **RF-2 Gestión del menú principal:** Deberá haber un menú principal que sea la primera escena que se muestre en el videojuego.
  - **RF-2.1 Selección de nivel:** Debe de poderse acceder a cualquier nivel desde el menú principal.
  - **RF-2.2 Cierre de la aplicación:** Se debe de poder cerrar la aplicación desde el menú principal.
  - **RF-2.3 Viaje al menú de opciones:** Se puede viajar desde el menú principal al menú de opciones.
- **RF-3 Gestión del menú de opciones:** Deberá haber un menú de opciones que permita modificar aspectos generales del juego. Desde el menú de opciones se debe poder volver al menú principal.
- **RF-4 Gestión de niveles:** Los niveles deben de contener todos los elementos necesarios y obligatorios de un nivel.
  - **RF-4.1 Controlar un avatar:** Debe de haber un avatar controlable por el jugador que realice las operaciones que le indique el jugador.
  - **RF-4.2 Zonas de muerte:** Debe haber zonas que maten al avatar del jugador cuando entre en contacto con ellas y devuelvan el nivel a un estado inicial.
  - **RF-4.3 Zonas de victoria (meta):** Debe haber zonas en las que, al entrar, se considere el nivel terminado y se vuelva al menú principal.
  - **RF-4.4 Gestión del menú de pausa:** Se debe poder acceder al menú de pausa, que parará la ejecución del nivel.
    - **RF-4.4.1 Abrir el menú de pausa:** Se deberá de poder abrir el menú de pausa, lo que pausará la ejecución del nivel.
    - **RF-4.4.2 Operaciones del menú de pausa:** Se deben de poder realizar las mismas operaciones que en el menú de opciones.
    - **RF-4.4.3 Cerrar el menú de pausa:** Se deberá poder cerrar el menú de pausa, lo que reanudará la ejecución del nivel.

- **RF-4.5 Manipulación de la gravedad:** Debe de ser posible manipular la gravedad del nivel.
  - **RF-4.6 Manipulación del tiempo:** Debe de ser posible modificar la escala de tiempo que afecta al nivel.
  - **RF-4.7 Aplicación de impulsos:** Se debe poder aplicar impulsos que varíen la trayectoria que lleva un objeto.
  - **RF-4.8 Obstáculos:** Puede haber obstáculos que maten al avatar jugable cuando entre en contacto con ellos.
  - **RF-4.9 Portales:** Puede haber portales que teletransporten al avatar jugable desde el punto en el que se encuentra a otro que corresponderá con otro portal.
- **RF-5 Gestión del avatar del jugador:** El jugador debe contar con un avatar controlable por el este. El avatar debe ser capaz de: saltar, moverse, realizar un acelerón y activar el tiempo bala.

## Requisitos no funcionales

Al ser el producto a entregar un videojuego (un software de ocio), resulta clave especificar que requisitos no funcionales se van a tener en cuenta.

- **Facilidad de uso:** Un videojuego cuyo uso no sea sencillo y cómodo puede perder muchos jugadores por esa única razón. Es clave que los jugadores tengan una experiencia agradable cuando interactúen con el juego.
- **Soporte:** Los cambios y actualizaciones del videojuego tienen que ser transparentes al usuario, pues no tiene necesidad de conocer como funciona internamente el juego, sino solo abrir el ejecutable y disfrutar del juego.
- **Apariencia o interfaz externa (*look and feel*):** No se ha especificado un diseño de interfaz, sin embargo en el mundo de los videojuegos hay un modelo general muy establecido. Es lógico adoptarlo para ofrecerle al jugador una experiencia que le resulte familiar.
- **Escalabilidad:** En un videojuego se van a ir añadiendo continuamente funcionalidades sobre el código para poder implementar todas las mecánicas, tanto definidas como que puedan surgir en el futuro. Por ello el código debe ser fácil de mantener y extender.

## **B.4. Especificación de requisitos**

## *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