

# TÍTULO DEL TRABAJO FIN DE GRADO

FERNANDO JOSÉ GARCÍA PADILLA

Trabajo fin de Grado

Supervisado por Dr. Pablo Trinidad Martín-Arroyo



Universidad de Sevilla

marzo 2017

Publicado en marzo 2017 por  
Fernando José García Padilla  
Copyright © MMXVII

[fergarpad@alum.us.es](mailto:fergarpad@alum.us.es)

Pon aquí cuestiones acerca del copyright

Yo, D. Fernando José García Padilla con NIF número 47214882E,

**DECLARO**

mi autoría del trabajo que se presenta en la memoria de este trabajo fin de grado que tiene por título:

*Título del Trabajo Fin de grado*

Lo cual firmo,

Fdo. D. Fernando José García Padilla  
en la Universidad de Sevilla  
30/03/2017



*Dedicado a todos los que me han apoyado  
Y en especial a...*





---

# AGRADECIMIENTOS

No olvides añadir una nota de agradecimiento a quienes hayan contribuido emocionalmente al proyecto fin de Grado.





---

## RESUMEN

El objetivo de este proyecto es la realización, haciendo uso de la ingeniería del software, de un videojuego independiente que sea exigente con sus mecánicas y se aventure combinando varios de los géneros existentes en el mercado con el propósito de llegar a un público específico dentro del mismo. Para su ejecución se ha escogido hacer uso del motor gráfico Unreal Engine, un potente framework que nos ofrece las herramientas que vamos a necesitar durante el transcurso del desarrollo, cuyo uso es gratuito y además está muy extendido en la actualidad.



# ÍNDICE GENERAL

<b>I</b>	<b>Introducción</b>	<b>1</b>
<b>1.</b>	<b>Contexto</b>	<b>3</b>
1.1.	El mundo del videojuego . . . . .	4
1.1.1.	El boom de los e-sports y los juegos para móvil . . . . .	5
1.1.2.	El auge del videojuego independiente . . . . .	6
1.2.	Estado del arte . . . . .	7
<b>2.</b>	<b>Objetivos del proyecto</b>	<b>9</b>
2.1.	Motivación . . . . .	10
2.2.	Listado de objetivos . . . . .	11
2.2.1.	Objetivos del producto . . . . .	11
2.2.2.	Objetivos individuales . . . . .	12
<b>II</b>	<b>Organización del proyecto</b>	<b>15</b>
<b>3.</b>	<b>Metodología</b>	<b>17</b>
3.1.	Estructura organizacional del proyecto . . . . .	18
3.2.	Metodología de desarrollo . . . . .	18
3.2.1.	Presentación de la metodología . . . . .	18
3.2.2.	Resumen de la metodología . . . . .	18

3.2.3. Adaptación de la metodología . . . . .	20
<b>4. Planificación</b>	<b>23</b>
4.1. Resumen temporal del proyecto . . . . .	24
4.2. Planificación inicial . . . . .	24
4.3. Informe de tiempos del proyecto . . . . .	25
<b>5. Costes</b>	<b>29</b>
5.1. Resumen de costes del proyecto . . . . .	30
5.2. Costes de personal . . . . .	30
5.3. Costes materiales . . . . .	33
5.4. Costes indirectos . . . . .	35
<b>III Desarrollo del proyecto</b>	<b>37</b>
<b>6. Arranque</b>	<b>39</b>
6.1. Lista de características . . . . .	40
6.2. Diseño arquitectónico . . . . .	40
<b>7. Primera iteración</b>	<b>41</b>
7.1. Características a desarrollar . . . . .	42
7.2. Diseño . . . . .	42
7.3. Implementación . . . . .	42
7.4. Pruebas . . . . .	44
7.5. Despliegue . . . . .	44

<b>IV Cierre del proyecto</b>	<b>45</b>
<b>8. Manual de usuario</b>	<b>47</b>
8.1. Sección libre . . . . .	48
<b>9. Conclusiones</b>	<b>49</b>
9.1. Informe post-mortem . . . . .	50
9.1.1. Lo que ha ido bien . . . . .	50
9.1.2. Lo que ha ido mal . . . . .	50
9.1.3. Discusión . . . . .	50
9.2. Trabajos futuros . . . . .	50
<b>V Appendices</b>	<b>51</b>
<b>Referencias bibliográficas</b>	<b>54</b>



---

# ÍNDICE DE FIGURAS

7.1. Diagrama UML de diseño para la iteración 1 . . . . .	42
---	----





# ÍNDICE DE CUADROS

4.1. Tabla resumen de tiempos y planificación . . . . .	24
4.2. Planificación temporal de fases e iteraciones . . . . .	24
4.3. Planificación temporal detallada de iteraciones . . . . .	26
4.4. Tabla resumen de horas de trabajo por rol . . . . .	26
4.5. Planificación temporal de iteraciones . . . . .	27
5.1. Tabla resumen de costes . . . . .	30
5.2. Tabla salarial BOE 2009 (simplificada) . . . . .	31
5.3. Tabla salarial del proyecto por hora . . . . .	32
5.4. Tabla sueldos netos . . . . .	33
5.5. Tabla de amortización de materiales . . . . .	34
5.6. Tabla de costes indirectos . . . . .	35
6.1. Tabla de características del dispositivo sobremesa . . . . .	40
6.2. Tabla de características del dispositivo portátil . . . . .	40
7.1. Memorando técnico 0001 . . . . .	43



---

# TODO LIST

Figure: Aquí el modelo de diseño en formato vectorial preferentemente (pdf) . . . 42

---

# PARTE I

---

## INTRODUCCIÓN

---



## CONTEXTO

*The good parts of a book may be only something a writer is lucky enough to overhear or it may be the wreck of his whole damn life – and one is as good as the other.*

*Ernest Hemingway (1899–1961),  
Novelist*

***E**n este primer capítulo abordaremos el entorno que rodea al producto software que pretendemos crear, la industria del videojuego, con el objetivo de situarnos en el mismo para conocer las oportunidades que ofrece el mercado y los riesgos, relacionados con el mismo, que debemos tener presentes.*

## 1.1 EL MUNDO DEL VIDEOJUEGO

La industria del videojuego vive, desde hace años, una etapa de popularización y expansión. Según datos de la asociación española de videojuegos (AEVI) y del informe anual de Newzoo (2016) el pasado año el sector creció un 8.5%, lo que representa pasar de 91.800 millones de dólares en 2015 a 99.600 millones de dólares en 2016. Además, esta tendencia es global y aumenta en cada uno de los cinco continentes, siendo la región geográfica que representan África y Oriente medio la que más prosperó en el transcurso del año.

Teniendo siempre presente que estamos en un mercado global y que lanzar un producto desarrollado en nuestro país, si se satisfacen necesidades como la localización del producto, puede repercutir en ventas en todo el mundo, España se sitúa en la octava posición del ranking mundial, gracias a la nada despreciable cifra de 1.812 millones de dólares de beneficios generados en 2016, pero lejos de los dos principales países, China y Estados Unidos, que sobrepasan con holgura los 20.000 millones de beneficios el pasado año.

La transformación del sector no tiene sólo que ver con cifras económicas y no me gustaría pasar por alto algo que bajo mi punto de vista es bastante más interesante y que, aunque bien daría para un trabajo aparte, describe perfectamente el giro tan drástico al que se está viendo sometida la industria en estos momentos: hace no tanto tiempo jugar a videojuegos era un territorio un tanto desconocido, lo que provocaba que a veces fuese visto con cierto recelo socialmente por una parte de la población. Esta tendencia se ha revertido vertiginosamente en un breve lapso de tiempo e incluso se podría decir que los videojuegos, en gran medida, han pasado a formar parte de la cultura popular: las grandes empresas textiles hacen camisetas con referencias a videojuegos o videoconsolas reconocidas; se realizan grandes eventos mediáticos en torno a personas relacionadas con los videojuegos, a los que se les tratan como a auténticas estrellas del rock; se crean canales de televisión en grandes plataformas audiovisuales dirigidos únicamente a los videojuegos y toda esta vorágine alcanza su sùmmum cuando en un evento de tanta proyección internacional como es el acto de clausura de los juegos olímpicos, el primer ministro del país que recibe el testigo para organizarlos en cuatro años, en este caso Japón, aparece disfrazado de un reconocido personaje de videojuegos: Mario. Aún queda un largo camino por recorrer, es un problema que persigue al sector desde su creación, pero en los últimos años hemos vivido una apertura de los videojuegos a la sociedad de tal magnitud que una parte relevante de la

misma lo ha pasado a ver de una forma distinta, dejando de lado ese citado recelo o cualquier connotación negativa, lo cual sin duda no puede ser más que favorable para la industria y su crecimiento.

Estableciendo un símil, podríamos comparar en algunos aspectos la fase en la que se encuentra la industria del videojuego con aquella en la que se podría encontrar la industria cinematográfica hace ya unas décadas: el sector está en etapa de expansión, cada vez más universidades empiezan a formar en ámbitos relacionados directamente con los videojuegos, se crean cada vez más puestos de trabajo bajo su cobijo e incluso están creciendo verdaderas celebridades en torno a la industria, ya que es habitual que las personas que son consumidores de la industria conozcan directores de juegos, compositores o jugadores profesionales, por citar unos ejemplos. Igualmente, no todos los aspectos el sector cinematográfico está más avanzado, ya que no podemos comparar las industrias del videojuego y del cine pasando por alto que la primera, pese a su más que notable juventud, ya consigue doblar en beneficios a la segunda. En conclusión, es cuestión de tiempo que a los videojuegos se le otorgue socialmente el mismo estatus que al cine: ser calificados de cultura o arte, y remarco socialmente porque en países como el nuestro ya se les consideran cultura, pero ese mensaje aún está por calar en muchos estratos de la sociedad.

### 1.1.1 El boom de los e-sports y los juegos para móvil

Aunque ninguno de los dos sectores estén estrictamente relacionados con lo que nos ocupa en este proyecto, son dos importantes partes de la industria que debemos analizar para contextualizarnos y entender plenamente la magnitud del mercado de los videojuegos actualmente.

Los deportes electrónicos, también denominados e-sports, son prueba del auge que está viviendo la industria, de los puestos de trabajo que genera en diversos sectores y de las posibilidades de negocio que hay alrededor del éxito que están viviendo actualmente.

Como se describía en la sección anterior, cadenas de ámbito nacional ya dedican algunos de sus canales a tiempo completo a los videojuegos y, realmente, casi la totalidad de estos suelen estar ligados a los e-sports, algo que actualmente es tendencia en los estados de nuestro entorno y algo normal desde hace años en algunos países concretos como Corea del Sur, pero el poderío de los e-sports va más allá: los deportes electrónicos mueven millones de personas en todo el mundo, consiguen llenar esta-



dios, reparten millones en premios y, quizá algo que es mucho más importante, generan un fenómeno fan que, aunque pueda impresionar si no se conoce a fondo este sector, está en vías de convertirse en uno tan poderoso como el que podría ser a día de hoy el fútbol: los espectadores de este tipo de entretenimiento no paran de crecer, conocen los equipos, sus jugadores y, de manera similar como ocurre en los deportes tradicionales, muchos de ellos marcan en el calendario el siguiente torneo de su juego preferido o el próximo partido de su equipo favorito. El fenómeno es tal que incluso se realizan realities de cómo estos equipos de deportes electrónicos viven o entrenan.

Por otra parte, los dispositivos móviles están ayudando a que la industria llegue a un número mucho mayor de personas y, lo que es mucho más importante, de una manera más indirecta: ya no es necesario que el usuario sienta esa necesidad, deseo o impulso específico de jugar a videojuegos y realice un desembolso para adquirir una consola, algo que sin duda puede hacer que muchos usuarios potenciales se pierdan en el proceso, sino que cualquiera que disponga de un teléfono móvil puede gozar de un gran catálogo de juegos sea cual sea su plataforma y de una manera muy accesible, por lo que tarde o temprano la mayoría probará lanzar algún videojuego en su dispositivo.

Este cambio en la accesibilidad amplía no sólo el espectro de jugadores que llegarán a poseer una consola o un ordenador gaming para tener más títulos a su disposición, generando beneficios para la industria en el proceso, sino que llega a sectores de la población a los que antes era difícil acceder, como el de las personas de mediana edad, y provocan cambios en las tendencias tales como que las mujeres superen a los hombres, en un sector tradicionalmente de los últimos, en el número total de jugadores.

### 1.1.2 El auge del videojuego independiente

Centrándonos más en el proyecto que nos ocupa, cabe destacar el fenómeno de los videojuegos indie: juegos desarrollados con poco presupuesto, por un equipo de desarrollo pequeño y que, generalmente, se atreven a arriesgarse un poco más que la media puesto que, tradicionalmente, los juegos desarrollados por grandes empresas y que cuentan con un gran presupuesto detrás tienden a asegurar más, a ser un producto similar a lo que ya está triunfando en ese momento en el mercado. En definitiva, no suelen estar dirigidos al gran público sino que suelen ser un producto de nicho dirigido a un sector muy específico dentro de la comunidad de jugadores.

Si bien se podría decir que nacieron en la clandestinidad, desde hace años están teniendo una enorme aceptación y prueba de ello es que cada vez tienen mayor visibi-

lidad: Steam, la principal plataforma de distribución digital en PC y que es propiedad de VALVE, apoya los videojuegos independientes gracias a su «Steam Greenlight», permitiendo a los usuarios de su plataforma valorar juegos desconocidos y, si la aceptación es buena, incluirlos en su catálogo. Sony con su videoconsola PlayStation y Microsoft con su homónima Xbox incorporan desde hace ya algunos años indies en su catálogo, y si bien en un principio eran bastante escasos y sujetos a una serie de restricciones (por ejemplo, en Xbox se apoyaban los juegos realizados con su propio creador de juegos) cada vez son más, y recientemente ha sido Nintendo quien no ha podido dejar pasar el carro de los videojuegos indies incorporando estos a su catálogo de lanzamiento de Nintendo Switch.

En conclusión, estamos en un momento en el que existe un público dentro de la comunidad que valora desde hace años el soplo de aire fresco que supone el contenido independiente en la industria, que permanecen atentos al panorama indie y no sólo a los lanzamientos triple A (o AAA), y en muchos casos valoran este tipo de contenido por encima de una superproducción que va dirigida a un público mucho menos específico.

## 1.2 ESTADO DEL ARTE

La industria de los videojuegos, aunque es difícil determinar su origen exacto (los primeros videojuegos surgieron tras la Segunda Guerra Mundial, pero a un ritmo lento y no son más que los albores del sector que conocemos hoy en día), es una industria joven. A pesar de esta juventud, y poniéndonos en antecedentes, es un sector que ya ha pasado por una grave crisis a principios de los ochenta y que lo llevó al borde de su extinción. Esta crisis fue motivada, principalmente, por el descontrol que originó la coexistencia en el mercado de demasiadas consolas y, sobre todo, el lanzamiento al mercado de un gran número de videojuegos de baja calidad, sin ningún tipo de control, mermando la confianza del usuario y haciendo descender las cifras de ventas.

La crisis se extendió hasta mediado de los ochenta y, desde entonces, la industria goza de salud y ha ido en crecimiento sin atravesar de nuevo ningún valle, por el contrario, es un sector que desde aquel momento ha ido en una clara tendencia positiva y que actualmente sigue en pleno auge.

Centrándonos en lo que supone en nuestro país, cada vez las universidades y academias ofertan más titulaciones o cursos relacionados con los videojuegos, y esto ayu-

da a que poco a poco el tejido empresarial relacionado con la industria del videojuego vaya creciendo. La existencia de poco tejido empresarial no ha impedido que en el pasado, durante la corta historia de la industria, en nuestro país hayan aparecido títulos de primer orden a nivel internacional: el caso más conocido es el de la saga «Commandos», que hizo su aparición a finales de los noventa, y más recientemente, un título que podríamos catalogar de súper producción, «Castlevania: Lord of the Shadows» (2010).

Aunque hay excepciones, la notable la mayoría de los juegos actuales son realizados a partir de un motor gráfico comercial ya definido. Estos motores suelen ser propiedad de la compañía, o bien de terceros bajo acuerdo, pero en los últimos años han surgido varios motores de uso gratuito que han ganado enorme popularidad y que hacen más accesible el desarrollo de videojuegos, entre los que encontramos Unreal Engine, CryEngine o Unity.

Tomando como ejemplo Unreal Engine, que es el motor que se usará para llevar a cabo este proyecto, es una potente herramienta que más allá de satisfacer las necesidades básicas del desarrollador (renderizado 2D/3D, detección de colisiones, texturizado de objetos, sistema de luces, etc.) incorpora importantes herramientas en su haber tales como «Persona», que permite al desarrollador crear o realizar cambios en animaciones o «UMG», que permite crear una interfaz de usuario de una manera sencilla dentro del motor. Además, permite utilizar tanto C++ como programación gráfica (a lo que se suele hacer referencia como «Blueprints») siendo la segunda forma unas 10 veces peor en rendimiento que la primera, según la propia Epic Games, desarrolladores de Unreal Engine.

## OBJETIVOS DEL PROYECTO

*The secret of happiness is not in doing what one likes, but in liking what one does.*

*Sir James Matthew Barrie (1860–1937),*

*Novelist*

**L**a siguiente sección presenta más detalladamente el proyecto, la razón de ser del mismo y un listado de objetivos, tantos del producto como individuales, que se pretenden alcanzar en el transcurso de su realización.

## 2.1 MOTIVACIÓN

Dada la saturación del mercado con productos similares o clónicos se pretende crear un videojuego que, siguiendo la filosofía indie que se describía previamente, consiga ser innovador mezclando géneros, que se salga del esquema predominante y rete al jugador siendo desafiante con sus mecánicas y haciéndole pensar, haga uso de la principalmente primera persona para tener la posibilidad de aprovechar el auge de la tecnología de realidad virtual y que, premeditadamente, deje bastante de lado la historia, personajes o desarrollo de los mismos y, por tanto, se centre plenamente en la jugabilidad. El jugador deberá ir haciendo uso de todas las mecánicas de las que dispone para sortear obstáculos, esquivar trampas, solucionar puzzles y hallar el camino correcto para ir avanzando en el juego.

El gameplay se basaría en la repetición y memorización de niveles, esquite y timing, así como en la gestión de adrenalina y, en menor medida, en la resolución de pequeños puzzles y exploración:

**Repetición y memorización de niveles:** Los peligros del entorno (balas, flechas, láseres, etc. y, por supuesto, caídas al vacío) matarían instantáneamente al jugador, lo que le obligaría a empezar el nivel desde el último punto de control o, en el caso de no disponer de ninguno, desde cero.

**Esquite y timing:** No se le daría la posibilidad al jugador de defenderse o eliminar peligros, por lo que el jugador deberá ir avanzando por el escenario haciendo uso de las mecánicas para sortear peligros en el momento justo sin morir en el intento.

**Gestión de adrenalina:** Cada acción que realice el jugador tendrá un peso asociado, por tanto, en todo momento el jugador deberá vigilar la barra de adrenalina y, así pues, pensar antes de realizar una acción.

**Resolución de puzzles:** A menudo el jugador deberá encontrar un botón / palanca / llave para avanzar o bien tendrá que encontrar un orbe dorado para acceder a una determinada zona que antes era inaccesible.

**Exploración:** No sólo deberá encontrar el camino correcto, o la forma de desbloquear o pasar por un camino, sino que por el mundo habrá atajos que interconecten el mapa con antiguos puntos de control, así como coleccionables repartidos por el mismo.

Podríamos concluir que, principalmente, mezcla los géneros de primera persona, aventura / exploración, plataformas, bullet hell y puzzle.

Teniendo en cuenta lo descrito anteriormente sería un producto que no va dirigido al gran público, sino que el público objetivo sería más bien un público de nicho, que busque algo diferente e intente poner a prueba sus capacidades.

## 2.2 LISTADO DE OBJETIVOS

He decidido dividir esta sección en dos partes, los objetivos que pretendo alcanzar individualmente y los objetivos que pretendo que alcance el producto:

### 2.2.1 Objetivos del producto

**Sistema de plataforma:** Realización de un sistema de mecánicas básicas de plataforma como saltar, agacharse, correr, manejar al personaje en el aire, etc. teniendo en cuenta que, sobre todo ésta última, sean amigables con el usuario y permitan llevarlas al límite sin frustración por parte del usuario.

**Sistema de escalada:** Elaboración de un sistema de escalada que maximice la interacción del personaje con el entorno y permita acciones tales como escalar paredes, desplazarse verticalmente o dejarse caer entre cornisas, moverse por ellas, saltar de una a otra, etc.

**Sistema de poderes:** Producción de un sistema de poderes coherente, en el que el personaje pueda ganar poderes tales como el de la teletransportación sin que entre en conflicto con las mecánicas, más básicas, de plataforma y escalada, es decir, que sólo los pueda usar durante un área determinada para poder avanzar en el juego pero que al abandonarla vuelva a tener que hacer uso de las mecánicas de plataforma y escalada.

**Elaboración del tutorial:** Elaborar un nivel de tutorial en el que se le presenten al jugador las mecánicas básicas.

**Elaboración de niveles prototipo:** Realización uno o una serie de niveles prototipo (dependiendo de la longitud de los mismos) de lo que sería un escenario del juego real.

- Sistema de animaciones:** Creación de una máquina de estados que permita pasar de una animación a otra sin bugs y de una manera sutil, sin saltos entre animaciones. 1 2
- Bullet hell en 3D:** Conseguir adaptar el género del bullet hell a un entorno 3D de manera adecuada, ya que este género es típico de entornos 2D donde, al eliminar toda una dimensión, es mucho más sencillo de implementar porque se reducen drásticamente los puntos en los que los peligros y el jugador pueden coincidir. En otras palabras, llenar el entorno de peligros en un escenario más amplio es una tarea mucho más difícil de diseñar. 3 4 5 6 7 8
- Realidad virtual:** Diseñar el juego de manera que fuese amigable con la innovadora tecnología de realidad virtual, es decir, enfocarlo hacia ella y que pudiese ser utilizado con ésta sin problemas. 9 10 11
- Optimización:** Optimizar lo máximo posible realizando todo el contenido posible en C++ ya que como explicaba anteriormente (tomando la propia Epic Games como fuente, desarrolladora de Unreal Engine), los tiempos de ejecución de ese código se reducen en torno a 10 veces usando el lenguaje de programación en C++ sobre la implementación en programación gráfica. 12 13 14 15 16

## 2.2.2 Objetivos individuales 17

- Uno de los objetivos principales que me he marcado es simplemente aprender y disfrutar, conocer tanto cómo es el desarrollo de videojuegos en sí (aspectos organizativos) como descubrir de qué manera están hechos los videojuegos desde el punto de vista de un desarrollador (aspectos, por tanto, de desarrollo) que es algo que siempre me ha parecido muy interesante. Es un objetivo muy fácil de satisfacer, y no podría decir que cumpliendo sólo este ya estuviese satisfecho, pero sin duda me es indispensable. 18 19 20 21 22 23 24
- Aprender a utilizar de una forma fluida el entorno y el conjunto de herramientas que conforman Unreal Engine. 25 26
- Ampliar mis conocimientos en el lenguaje de programación C++, aprovechando que es el lenguaje en el cual se programa Unreal Engine. 27 28
- Adquirir conocimientos en programación visual, ya que es un aspecto que en la titulación se ha abordado de manera muy escueta y habiendo elegido Unreal Engine para la ejecución del proyecto es algo que voy a necesitar en algún momento del desarrollo. 29 30 31 32

- 1 Formarme en diversas aptitudes que no entran dentro del ámbito de la titulación de
- 2 Ingeniería del Software, como pueden ser la creación de entornos 3D, la textu-
- 3 rización e iluminación de los mismos, creación de landscapes, la animación de
- 4 personajes, la creación de interfaces, diseño de niveles, etc.





---

## PARTE II

# ORGANIZACIÓN DEL PROYECTO

---



## METODOLOGÍA

*The good parts of a book may be only something a writer is lucky enough to overhear or it may be the wreck of his whole damn life – and one is as good as the other.*

*Ernest Hemingway (1899–1961),  
Novelist*

A continuación se presenta la estructura organizacional del proyecto, la metodología que se ha elegido seguir para el desarrollo de este software, un breve resumen de la misma y la forma en la que se va a adaptar a nuestro problema específico.

### 3.1 ESTRUCTURA ORGANIZACIONAL DEL PROYECTO

Dado que el proyecto no se realiza en grupo no se podría decir que existe una estructura organizacional propiamente dicha, o al menos carecería de sentido establecer una estructura organizacional para una sola persona, ya todas las responsabilidades y roles posibles recaerían sobre una única persona, salvando la figura del tutor.

A raíz de lo descrito surge una consecuencia directa, y es que algunos roles o actividades desaparecerían del proyecto, todos los que tengan que ver con la interacción entre dos o más personas, al carecer de sentido en un proyecto con un único miembro en él. Ejemplo de esto podrían ser todas aquellas actividades que tengan que ver con la sincronización / puesta en común con el resto de miembros del equipo o cualquier rol relacionado con la resolución de conflictos dentro del equipo del proyecto.

### 3.2 METODOLOGÍA DE DESARROLLO

#### 3.2.1 Presentación de la metodología

Para la realización de este proyecto se hará uso de la metodología «Feature-Driven Development» (FDD), denominada «Desarrollo basado en funcionalidades» en español.

Fue creada por Jeff De Luca y Peter Coad a finales de siglo para salvar un importante proyecto que había sido declarado como irrealizable y tiene su razón de ser en la calidad y el monitoreo constante del proyecto.

#### 3.2.2 Resumen de la metodología

FDD es una metodología englobada en el grupo de las denominadas ágiles, constituye un proceso iterativo e incremental de desarrollo software y consta de 5 partes:

1. **Desarrollo del modelo global:** La realización de un proyecto siguiendo el esquema dictado por FDD empieza por la realización de un modelo global de alto nivel que tiene en cuenta el contexto y el alcance del sistema. El modelo se subdivide en partes más pequeñas que se van completando y se confecciona un diagrama de clases por cada una. Finalmente cada una de las partes en las que se subdividió se van agrupando para formar el modelo global final.

2. **Elaboración de la lista de funcionalidades:** Seguidamente, utilizando el conocimiento obtenido durante el desarrollo del modelo global, se confecciona una lista de funcionalidades que posteriormente se dividirán en funcionalidades más específicas.
3. **Planificación por funcionalidad:** Nuevamente partiendo del punto anterior, tomamos la lista de funcionalidades y esta vez las ordenamos según su prioridad y teniendo en cuenta su dependencia.
4. **Diseño por funcionalidad:** Se elige un conjunto de funcionalidades de la lista y se procede a diseñarlas y desarrollarlas mediante un proceso iterativo.
5. **Construcción por funcionalidad:** Después de una fase de diseño satisfactoria se procede a la construcción total del proyecto.

Además, la metodología FDD consta de 6 **roles clave**, los cuales se describen a continuación:

1. **Jefe de proyecto:** Es el responsable de gestionar el presupuesto, el tiempo, el espacio y los recursos del proyecto. También es el responsable de transmitir el progreso del proyecto a los altos cargos de la empresa.
2. **Arquitecto jefe:** Es el responsable del diseño global del sistema. Tiene la última palabra en todas las cuestiones de diseño.
3. **Jefe de desarrollo:** Es el responsable del desarrollo en el día a día, de evitar situaciones de bloqueo y de solucionar conflictos en el proyecto.
4. **Programadores jefes:** Son desarrolladores experimentados que se encargan de diseñar los requisitos a alto nivel y trabajan junto a otros programadores jefes para resolver las dificultades a las que se enfrenta el proyecto día a día.
5. **Encargados de clases:** Desarrolladores que junto a su pequeño equipo de trabajo y bajo la supervisión de un programador jefe diseñan, programan, prueban y documentan las funcionalidades que se implementan en el sistema.
6. **Expertos de dominio:** Son los encargados, haciendo uso de su conocimiento del negocio, de detallarle minuciosamente a los desarrolladores las características que debe tener el producto.

### 3.2.3 Adaptación de la metodología

Teniendo en consideración que FDD es una metodología pensada para grupos de personas de un tamaño considerable, grupos en los que incluso otras metodologías ágiles como SCRUM no tendrían cabida al ser imposible realizar una autogestión, y enfocada al desarrollo de un software más convencional, donde se podría diseñar un diagrama de clases con facilidad (como podrían ser las aplicaciones de escritorio, móvil, web, etc.).

Puesto que este proyecto será llevado a cabo por una única persona y el producto generado en su desarrollo será un videojuego, es necesario hacer algunas modificaciones en nuestra metodología:

1. **Fase de diseño del videojuego:** La principal diferencia radica aquí puesto que, dada la naturaleza del proyecto, al no poder organizar un diagrama de clases tenemos que buscar alternativas. En este caso se ha optado por elaborar una documentación con las decisiones de diseño del videojuego, esto vendría a coincidir con el conjunto de las mecánicas que va a contener, las características que incluiría el videojuego y las distintas singularidades que conformarían el mismo.
2. **Elaboración de la lista de funcionalidades:** Al igual que en la metodología FDD tradicional, se elabora una lista de funcionalidades a partir del paso anterior y éstas se desglosan a su vez en funcionalidades más específicas.
3. **Planificación por funcionalidad:** Nuevamente como en la metodología FDD tradicional, ordenamos las funcionalidades resultantes según su prioridad y teniendo en cuenta sus dependencias.
4. **Diseño por funcionalidad:** Otra gran diferencia sería que, puesto que el grupo de desarrollo está formado por una sola persona, se irían eligiendo funcionalidades de la lista de una en una y se diseñaría e implementaría.
5. **Construcción por funcionalidad:** Igual que en la metodología FDD se procedería a la construcción total del proyecto.

La **documentación** se generaría de forma constante durante todo el transcurso del proyecto y lo haría de forma iterativa tanto cada vez que se empezara a diseñar una funcionalidad como cuando se terminase el diseño de la misma.

1 Para finalizar, los **roles** también necesitan adaptarse al proyecto, ya que muchos de  
2 ellos dejan de tener sentido por las mismas razones que se citaban anteriormente. En  
3 este caso es necesario una simplificación, que llevaría a que los roles de programador  
4 se aglutinaran en uno solo y lo mismo pasaría con los roles de diseño, lo que daría  
5 lugar a:

- 6 ■ **Jefe de proyecto:** Es el responsable de gestionar el presupuesto, el tiempo, el  
7 espacio y los recursos del proyecto. En el caso que nos ocupa sería el encargado  
8 de la elaboración de la memoria en términos generales.
- 9 ■ **Diseñador:** Es el responsable del diseño global del sistema y de la elicitación de  
10 requisitos tanto a alto nivel como de posteriormente detallarlos. En resumidas  
11 cuentas, sería un analista pero esta vez centrado en el mundo del videojuego.
- 12 ■ **Programador:** Diseña, programa, prueba y documenta las funcionalidades que  
13 se implementan en el sistema.





## PLANIFICACIÓN

*The good parts of a book may be only something a writer is lucky enough to overhear or it may be the wreck of his whole damn life – and one is as good as the other.*

*Ernest Hemingway (1899–1961),  
Novelist*

*En la siguiente sección se expone la planificación del proyecto y el informe de tiempos generado durante el transcurso del mismo que, comparado con lo anterior, nos permitirá ver la desviación en cuanto a tiempo del proyecto.*

## 4.1 RESUMEN TEMPORAL DEL PROYECTO

1

Resumen del proyecto	
Fecha de inicio	TODO/TODO/2017
Fecha de fin	TODO/TODO/2017
Periodicidad de las revisiones (promedio)	TODO semanas
Carga de trabajo semanal (promedio)	TODO horas
Horas totales previstas	TODO horas
Horas finales	TODO horas

Cuadro 4.1: Tabla resumen de tiempos y planificación

## 4.2 PLANIFICACIÓN INICIAL

2

La siguiente tabla muestra un resumen de las iteraciones junto con su fecha de comienzo y fin:

3

4

Resumen de fases e iteraciones	
Fase de estudio	20/02/17 a 12/03/17
Fase de inicio del proyecto	13/03/17 a 02/04/17
Iteración 1	03/04/17 a 16/04/17
Iteración 2	17/04/17 a 30/04/17
Iteración 3	01/05/17 a 14/05/17
Iteración 4	15/05/17 a 28/05/17
Iteración 5	29/05/17 a 11/06/17
Iteración 6	12/06/17 a 25/06/17
Fase de cierre del proyecto	26/06/17 a 30/06/17

Cuadro 4.2: Planificación temporal de fases e iteraciones

Observando la tabla, podemos apreciar que podríamos establecer una clara diferenciación entre las iteraciones:

5

6

- **Fase de estudio:** En esta primera etapa interviene principalmente el jefe de proyecto, que será el encargado de estudiar las diferentes tecnologías que tiene a su

7

8

disposición y de poner los cimientos más básicos de la ruta que seguirá el proyecto.

- **Fase de inicio del proyecto o iteración 0:** Es la etapa en la que se realizan las primeras fases de la metodología y buena parte de la elaboración de la memoria, salvando los apartados finales (por ejemplo, el manual o las conclusiones) y las secciones centradas en el desarrollo, principalmente. Teniendo en cuenta lo descrito, en esta fase intervienen el jefe de proyecto y el diseñador.

- **Iteraciones 1-6:** Son las etapas más puramente de desarrollo. Intervienen tanto el analista, que será el encargado de preparar las iteraciones, como el programador, que será quien codifique y genere documentación de las iteraciones (como ya se explicó en la sección §3.2.3).

- **Fase de cierre del proyecto:** En esta iteración intervienen el diseñador, encargado de elaborar el manual de la aplicación, y el jefe de proyecto, que será quien finalice los últimos apartados de la memoria, haga las últimas modificaciones sobre la misma y realice la presentación mediante la cuál se expondrán los resultados del proyecto.

Nota: En prácticamente todas las fases del proyecto intervendrían los tres roles, pero dado el escenario en el que se enmarca el proyecto (con una única persona para tres roles), en cada fase debemos focalizar más en uno u otro.

TODO: Explicar cómo se han decidido las fechas, interacción con fechas importantes y situaciones personales.

A continuación se muestra una tabla detallada de las iteraciones, mostrando qué rol interviene, el número de horas de trabajo que participa y el número de horas totales de cada iteración:

Y, por último, de la anterior tabla podemos extraer el número total de horas de trabajo de cada rol, lo cuál nos será útil en el próximo capítulo cuando elaboremos la documentación relativa a los costes de personal (ver sección §5.2):

## 4.3 INFORME DE TIEMPOS DEL PROYECTO

TODO: Lo mismo que el anterior pero con datos reales.

Iteraciones detalladas			
Iteración	Roles implicados	Horas est. / rol	Horas estimadas
Fase de estudio	Jefe de proyecto	30 horas	30 horas
Fase inicio proyecto	Jefe de proyecto	30 horas	40 horas
	Diseñador	10 horas	
Iteración 1	Diseñador	4 horas	24 horas
	Programador	20 horas	
Iteración 2	Diseñador	4 horas	24 horas
	Programador	20 horas	
Iteración 3	Diseñador	4 horas	24 horas
	Programador	20 horas	
Iteración 4	Diseñador	4 horas	24 horas
	Programador	20 horas	
Iteración 5	Diseñador	4 horas	24 horas
	Programador	20 horas	
Iteración 6	Diseñador	4 horas	24 horas
	Programador	20 horas	
Fase cierre proyecto	Diseñador	0 horas	0 horas
	Jefe de proyecto	0 horas	
TOTAL			X horas

Cuadro 4.3: Planificación temporal detallada de iteraciones

Resumen de horas de trabajo por rol	
Jefe de proyecto	X horas
Diseñador	X horas
Programador	X horas

Cuadro 4.4: Tabla resumen de horas de trabajo por rol

TODO: Justificar los retrasos de forma detallada aquí para cada una de las iteraciones. Explicar las razones.

1  
2

Resumen de iteraciones	
<b>Iteración 1</b>	10/10/14 a 21/10/14
<b>Iteración 2</b>	21/10/14 a 15/11/14
...	dd/mm/aa a dd/mm/aa

Cuadro 4.5: Planificación temporal de iteraciones



## COSTES

1

2 *The good parts of a book may be only something a writer is lucky enough to overhear or it may be the wreck*  
3 *of his whole damn life – and one is as good as the other.*

4

*Ernest Hemingway (1899–1961),*

5

*Novelist*

6

7 *E* n este capítulo abordaremos una de las partes más importantes en la planificación de  
8 un proyecto: los costes, y para ello los dividiremos en costes de personal, de material e  
9 indirectos.



## 5.1 RESUMEN DE COSTES DEL PROYECTO

Uno de los aspectos más importantes para que un proyecto finalice con éxito es que esté bien presupuestado desde su inicio. No obstante, es un objetivo difícil de alcanzar, por tanto lo que se pretende con la realización de este capítulo es, primeramente, intentar que se ajuste lo máximo a la realidad y, en caso de que se desvíe demasiado de la misma, aprender de los errores cometidos para futuros proyectos.

La tabla que se muestra a continuación anticipa los costes a los que tendremos que hacer frente durante la ejecución del proyecto:

Resumen del proyecto	
<b>Costes de personal</b>	TODO €
Sueldo neto	TODO €
Impuestos	TODO €
Costes sociales	TODO €
<b>Costes materiales</b>	TODO €
<b>Costes indirectos</b>	TODO €
<b>TOTAL</b>	TODO €

Cuadro 5.1: Tabla resumen de costes

En las siguientes secciones veremos en detalle cómo están calculados cada uno de los resultados mostrados.

## 5.2 COSTES DE PERSONAL

Los costes de personal son, con diferencia, los más importantes en un presupuesto de un proyecto del sector TIC: por regla general, en un proyecto de este sector, aproximadamente el 90% de los costes se puede catalogar como costes de personal, y dentro de ese porcentaje alrededor de un 5% son impuestos.

En este caso, el proyecto tendría un único trabajador pero debemos tener en cuenta que desarrollaría diversas tareas dentro del mismo, por tanto el salario no puede ser idéntico para una labor u otra. Tal y como se ha descrito previamente se han establecido 3 roles: jefe de proyecto, diseñador y programador (para más detalles, revisar sección §3.2.3).

Con el fin de calcular el coste que supondrían los honorarios nos apoyamos en el Boletín Oficial del Estado (BOE), en este caso el boletín número 82 del sábado 4 de abril de 2009, que especifica el salario en el sector de tecnologías de la información y la comunicación, en el que como se citaba se enmarca el proyecto. Para hallar los datos que buscamos debemos fijarnos en la tabla salarial de 2009 de dicho boletín, a continuación se muestran los roles de nuestro proyecto junto con los roles que nos interesan de los descritos en el BOE:

Tabla salarial BOE 2009 (simplificada)			
Categoría	Categoría BOE	Salario mes (x14)	Salario año
<b>Jefe de proyecto</b>	Jefe/a de Operación	1.103,04 €	15.442,56 €
<b>Analista</b>	Analista y A. de sistemas	1.569,25 €	21.969,50 €
<b>Programador</b>	Programador/a senior	1.103,04 €	15.442,56 €

Cuadro 5.2: Tabla salarial BOE 2009 (simplificada)

Nótese que nomenclatura de los roles de nuestro proyecto no coinciden plenamente con la que utiliza el BOE, sino que en la tabla anterior jefe de proyecto se corresponde con «jefe/a de operación», analista con «analista y analista de sistemas» y programador con «programador/a senior». A continuación se justifica, para cada uno de ellos, la relación de correspondencia, por tanto citando al BOE:

- **Jefe/a de Operación:** Es el trabajador que, con personal a su cargo, tiene bajo su responsabilidad: la planificación del trabajo a realizar en cada uno de los ordenadores y turnos; la asignación de recursos humanos en cada puesto de trabajo y en cada momento; la coordinación con los integrantes de las distintas áreas del proyecto; el conocimiento, en todo momento, del sistema operativo y del trabajo para tomar decisiones en las dudas que se presenten en operación; y, por último, la conservación de los manuales de operación.
- **Analista de Sistemas:** Le corresponde el diseño, puesta a punto y mantenimiento de los sistemas operativos a utilizar en los procesos de mecanización; formarse e informarse de todo lo concerniente al proceso de datos; y, finalmente, asesorar y coordinar con todo el personal de la empresa sobre las posibilidades de proceso de datos.
- **Programador/a Senior:** Es el trabajador que debe tener un conocimiento profundo de las técnicas y recursos que maneja, enfocado principalmente a los lenguajes

de programación existentes en el ordenador que utiliza así como de las facilidades y ayuda que le presta al software para la puesta a punto de programas, correspondiéndole estudiar los problemas complejos definidos por los analistas, confeccionando organigramas detallados de tratamiento. Le corresponde redactar programas en el lenguaje de programación que le sea indicado. Asimismo, confecciona juegos de ensayo, pone a punto los programas y completa los expedientes técnicos de los mismos.

Como podemos apreciar en el texto extraído del BOE (que se ha presentado de forma resumida) nuestros roles coinciden en un amplio porcentaje con los roles descritos en él (de nuevo, para más detalles sobre los roles de este proyecto, revisar sección §3.2.3).

A partir de la tabla anterior, debemos calcular los honorarios por hora de cada uno de los roles. Para realizar esto debemos dividir el salario anual entre la jornada anual:

$$\frac{\text{Salario}_{\text{Anual}}}{\text{Jornada Laboral Anual}_{\text{Horas}}} = \text{Salario}_{\text{Hora}}$$

La jornada máxima anual del sector TIC está fijada en 1.800 horas, por tanto, una vez aplicado esto a cada rol, obtendríamos los siguientes resultados:

Tabla salarial del proyecto por hora	
Categoría	Salario por hora
Jefe de proyecto	8,58 €
Diseñador	12,20 €
Programador	8,58 €

Cuadro 5.3: Tabla salarial del proyecto por hora

Dentro de los costes de personal debemos diferenciar entre el sueldo neto del trabajador, los impuestos y los costes sociales.

Para calcular el sueldo neto del trabajador, debemos tomar la planificación del proyecto que hemos visto previamente (sección §4.1) y multiplicar el total de horas de cada rol por su coste a la hora que acabamos de calcular, así pues:

$$\text{Sueldo}_{\text{Neto}} = \text{Salario}_{\text{Hora}} * \text{Horas}_{\text{Proyecto}}$$

Y, a su vez, podemos concluir que:

$$TotalSueldos_{Neto} = \sum_{i=1}^{n^{\circ}empleados} SueldoNeto_i = TODO$$

Sueldos netos			
Rol	Total de horas	Sueldo neto por hora	Sueldo neto
Jefe de proyecto	TODO horas	8,58 €	TODO €
Costes materiales	TODO horas	12,20 €	TODO €
Costes indirectos	TODO horas	8,58 €	TODO €
<b>TOTAL</b>			<b>TODO €</b>

Cuadro 5.4: Tabla sueldos netos

Por tanto, el coste del proyecto en cuanto a sueldos netos asciende a **TODO**.

TODO: Impuestos y costes sociales.

## 5.3 COSTES MATERIALES

Los costes materiales son costes independientes de un proyecto en concreto, por lo que el coste que supone para este proyecto no se corresponde directamente su valor de adquisición, sino que es necesario realizar una amortización teniendo en cuenta la duración de este proyecto respecto a la vida útil de cada elemento.

La amortización se realizaría de la siguiente manera:

$$Coste_{Mes} = \frac{CosteCompra}{VidaUtil_{Meses}}$$

Para la ejecución del proyecto se cuenta con:

- Dos equipos, un sobremesa y un portátil (para más detalles sobre los equipos, ver sección §6.2) ambos con una vida útil estimada de 3 años ya que el sobremesa, aún siendo más antiguo, ha sido actualizado recientemente.
- Una mesa de trabajo, con una vida útil de 4 años.

- Una pequeña mesa de reuniones que se utilizaría en caso de tener que mantener una reunión con una persona interesada en el proyecto y que contaría con una vida útil de 6 años.
- Cinco sillas: una de escritorio utilizada en la mesa de trabajo y cuatro, más simples, utilizadas en la mesa de reuniones. La utilizada en la mesa de trabajo costaría con una vida útil de 4 años y el resto contarían con una vida útil de 6 años.
- Dos repisas, ambas con una vida útil de 6 años.
- Una estantería, con una vida útil de 6 años.
- Dos papeleras, con una vida útil de 10 años.

La tabla que se muestra a continuación muestra los materiales que van a cumplir una utilidad en la realización del proyecto, su vida útil, su valor y el coste real al mes que supone en el proyecto:

Amortización de materiales			
Material	Valor	Vida útil	Coste real mes
Sobremesa	1100 €	3 años (36 meses)	30,55 €
Portátil	700 €	3 años (36 meses)	19,44 €
Mesa trabajo	200 €	4 años (48 meses)	4,16 €
Mesa reuniones	150 €	6 años (72 meses)	2,08 €
Silla trabajo	60 €	4 años (48 meses)	1,25 €
Silla reuniones x 4	40 €	6 años (72 meses)	0,55 * 4 = 2,22 €
Repisa x 2	30 €	6 años (72 meses)	0,41 * 2 = 0,83 €
Estantería	80 €	6 años (72 meses)	1,11 €
Papelera x 2	5 €	10 años (120 meses)	0,04 * 2 = 0,08 €
TOTAL (mes)			59,63 €

Cuadro 5.5: Tabla de amortización de materiales

Una vez realizado esto, sólo faltaría multiplicar el coste material al mes por la duración del proyecto en meses, de esta forma:

$$\text{CosteMaterial}_{\text{Proyecto}} = \text{CosteAmortizado} * \text{MesesProyecto}$$

Teniendo en cuenta el número de meses del proyecto, en este caso TODO meses, el coste material total es de **TODO**.

## 5.4 COSTES INDIRECTOS

Los costes indirectos son, sin lugar a dudas, los más complicados de calcular a la hora de elaborar un presupuesto.

Siguiendo su definición, un coste indirecto es aquel que afecta al proceso productivo en general de uno o más productos, en este caso proyectos, y que por tanto es complicado asignar una parte de ese coste a un proyecto en concreto sin elaborar algún criterio, sistemático y estable en el tiempo, que nos permita hacerlo.

Existen gran variedad de costes indirectos, como podrían ser licencias, seguros, bancos, contratación de servicios profesionales (como podrían ser abogados para posibles temas legales o gestoras para, por ejemplo, llevar a cabo la contabilidad o las declaraciones a hacienda), alquileres, etc.

En el caso de este proyecto, al mes, son los siguientes:

Costes indirectos	
Motivo del coste	Coste mes
Licencias	0 €*
Alquiler	400 €
Internet (fibra óptica)	40 €
Luz	60 €
Agua	20 €
Material fungible	20 €
Servicios de limpieza	150 €
<b>TOTAL (mes)</b>	<b>TODO €</b>

Cuadro 5.6: Tabla de costes indirectos

[\*]: En cuanto a licencias debemos tener en presente que Unreal Engine es un motor de uso gratuito pero, sin embargo, la compañía propietaria del motor, en este caso Epic Games, establece en su EULA una cláusula por la que habrá que abonar cierta cantidad, un loyalty, si se superan los 3.000\$ de beneficio por cuatrimestre. Esta cantidad en gira en torno a un 5% de los beneficios que pasen de 3.000\$: por ejemplo, si se generan 5000\$ de beneficio habrá que remitirle a Epic Games aproximadamente 100\$ (el 5% de 2000\$, resultado de la resta de 5.000\$ - 3.000\$).

En este caso, el coste indirecto al mes no sería más que el sumatorio de todos los

costes indirectos al mes:

1

$$CosteMaterial_{Mes} = \sum_{i=1}^{n^o \text{ costes}} CosteIndirecto_i$$

Por consiguiente, el coste indirecto total del proyecto sería el coste indirecto por mes multiplicado por el número de meses del proyecto:

2

3

$$CosteMaterial_{Proyecto} = CosteAmortizado * MesesProyecto$$

Por último, teniendo presente el número de meses del proyecto, en el caso que nos ocupa TODO meses, el coste indirecto total del proyecto asciende a **TODO**.

4

5

---

## PARTE III

---

# DESARROLLO DEL PROYECTO

---





## ARRANQUE

*The good parts of a book may be only something a writer is lucky enough to overhear or it may be the wreck of his whole damn life – and one is as good as the other.*

*Ernest Hemingway (1899–1961),  
Novelist*

*D*urante este capítulo de la memoria se presentará la lista de características que conforman el proyecto así como el diseño arquitectónico, que describe los sistemas de producción, preproducción y pruebas.

6.1 LISTA DE CARACTERÍSTICAS

1

TODO: Aplicar aquí la primera iteración de Feature Driven Development.

2

6.2 DISEÑO ARQUITECTÓNICO

3

Para la realización del proyecto se dispone de dos dispositivos, un ordenador sobremesa y un ordenador portátil (a partir de ahora sobremesa y portátil) en el que se realizarán las labores de producción, preproducción y pruebas.

4

5

6

Características sobremesa	
Sistema/s operativo/s	Windows 10
Procesador	Intel Core i7 930 2.80Ghz
Placa base	Asus P6X58D-E
Memoria RAM	12GB DDR3 1600Mhz PC3-12800 CL6 (2x4GB + 2x2GB)
Disp. de almacenamiento	SSD 120GB + Disco duro 3TB SATA3 7200rpm
Tarjeta gráfica	Sapphire Radeon HD 7950 OC 3GB GDDR5

Cuadro 6.1: Tabla de características del dispositivo sobremesa

Características portátil	
Sistema/s operativo/s	Windows 10
Procesador	Intel Core i7 4712MQ 2.3 GHz
Placa base	(Dato no proporcionado)
Memoria RAM	8GB DDR3 SODIMM (1x8GB)
Disp. de almacenamiento	Disco duro 1TB SATA 5400rpm
Tarjeta gráfica	Nvidia GeForce GT820M 2GB GDDR3

Cuadro 6.2: Tabla de características del dispositivo portátil

## PRIMERA ITERACIÓN

1

2 *The good parts of a book may be only something a writer is lucky enough to overhear or it may be the wreck*  
3 *of his whole damn life – and one is as good as the other.*

4

*Ernest Hemingway (1899–1961),*

5

*Novelist*

6

7

8

*R* esumen de lo que va a ocurrir en el capítulo. ¿Cuál es el objetivo que tenemos con este capítulo?

7.1 CARACTERÍSTICAS A DESARROLLAR1

1. Funcionalidad A.2
2. Funcionalidad B.3

7.2 DISEÑO4

Aquí una discusión de cómo va a afectar todo al diseño5

Debe insertarse un diagrama UML de diseño con los cambios y hacer referencia en el texto así Fig. §7.1.67

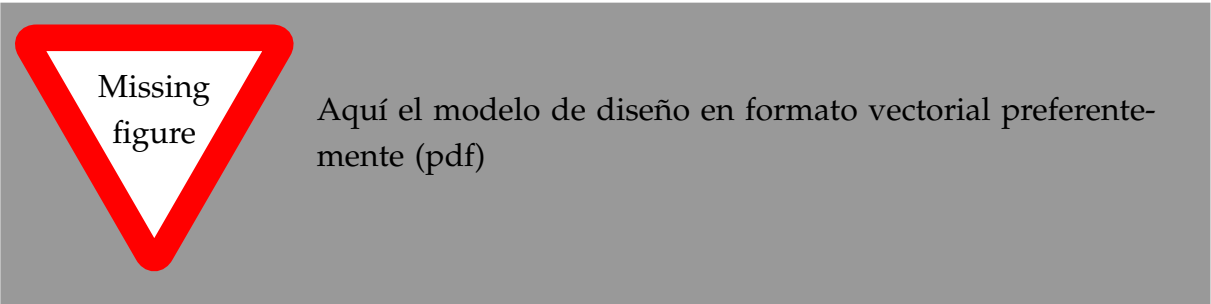


Figura 7.1: Diagrama UML de diseño para la iteración 1

Un memorando técnico por cada decisión de diseño.8

7.3 IMPLEMENTACIÓN9

Un memorando técnico por cada decisión de implementación y refactorización que afecte al diseño.1011

## Memorando técnico 0001

<b>Asunto</b>	¿Cuál es el problema?
<b>Resumen</b>	¿Cuál es la solución propuesta?
<b>Factores causantes</b>	Descripción pormenorizada del problema
<b>Solución</b>	Descripción pormenorizada de la solución propuesta
<b>Motivación</b>	¿Por qué propone esta solución?
<b>Cuestiones abiertas</b>	Factores a tener en cuenta en la solución cuya dimensión se reconoce.
<b>Alternativas</b>	Otras soluciones consideradas y la razón por la que se excluyeron.

Cuadro 7.1: Memorando técnico 0001

Identificador	Descripción de la acción de alto nivel			
0001	Prueba			
Métodos de alto nivel				
[return.type] method_name1 (param1:type1, ...)				
Pasos (Usar Pseudocódigo o similar)				
1. Paso 1.				
2. Paso 2.				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Técn.	IU
1	ClassName	[return.type] method_name1 (param1:type1, ...)	001	SI
Diagrama de Colaboración				

Identificador	Descripción de la acción de alto nivel			
alvotermar02	Grubber			
Métodos de alto nivel				
[return_type] grubber (param1:type1, ...)				
Pasos (Usar Pseudocódigo o similar)				
1. Lanzar 2 dados				
2. Compara resultado de los dados con kicking del open-side				
2.1. Si valor dados es menor o igual a kicking, avanza 10m				
3.1. Si no hay defensa y el golpeo es exitoso, el pateador retiene la posesión del balón				
3.2. Si hay defensa y el golpe es exitoso, el atacante tira un dado y suma su valor al de speed y strength y el defensor lanza 2 dados y lo suma al valor de speed y strength de su jugador, el vencedor será aquel que tenga más puntos, si es igual, la posesión es del defensor				
4.1. Si no es exitoso y hay defensa el balón pasa a posesión del defensor				
4.2. Si no es exitoso y no hay defensa de lanza un line-out				
Métodos de bajo nivel necesarios				
Paso	Clase	Método	Mem. Técn.	IU
1	Dice	[Integer] throwDice ()	001	SI
2	ClassName	[Int] compareKickingToDice (kicking:Integer, dice: Integer)	001	SI
2.1	ClassName	[Integer] setLine (line:Integer)	001	SI
4.2	ClassName	[Integer] lineOut ()	001	SI

## 7.4 PRUEBAS

3

Descripción de las pruebas realizadas al software

4

## 7.5 DESPLIEGUE

5

Breve resumen de cómo se han desplegado los cambios en el sistema de producción.

6

7

---

## PARTE IV

---

### CIERRE DEL PROYECTO

---





# MANUAL DE USUARIO

1

2 *The good parts of a book may be only something a writer is lucky enough to overhear or it may be the wreck*  
3 *of his whole damn life – and one is as good as the other.*

4

*Ernest Hemingway (1899–1961),*

5

*Novelist*

6

7

8

*R* esumen de lo que va a ocurrir en el capítulo. ¿Cuál es el objetivo que tenemos con este capítulo?

## 8.1 SECCIÓN LIBRE

1

Estructurar en función del proyecto.

2

## CONCLUSIONES

*The good parts of a book may be only something a writer is lucky enough to overhear or it may be the wreck of his whole damn life – and one is as good as the other.*

*Ernest Hemingway (1899–1961),  
Novelist*

*R*esumen de lo que va a ocurrir en el capítulo. ¿Cuál es el objetivo que tenemos con este capítulo?

## 9.1 INFORME POST-MORTEM 1

Qué es un informe post-mortem 2

### 9.1.1 Lo que ha ido bien 3

- Argumento a favor 1. 4
- Argumento a favor 2. 5
- Argumento a favor 3. 6

### 9.1.2 Lo que ha ido mal 7

- Argumento en contra 1. 8
- Argumento en contra 2. 9
- Argumento en contra 3. 10

### 9.1.3 Discusión 11

En función de lo anterior, qué cambiaría si empezara hoy el proyecto de nuevo. 12

## 9.2 TRABAJOS FUTUROS 13

Enumera los puntos abiertos y que no se han resuelto. Indica si darían lugar a otro proyecto y de qué forma se podría acotar. 14  
15

---

## PARTE V

### APPENDICES

---









---

## BIBLIOGRAFÍA