



**Escuela Superior  
de Ingeniería y Tecnología**  
Universidad de La Laguna

# Trabajo de Fin de Grado

Grado en Ingeniería Informática

## Desarrollo de videojuego RPG en La Laguna, ambientado en el siglo XVI

*Development of a RPG video game in La Laguna, set in the 16th  
century*

Gabriel Melián Hernández

La Laguna, 14 de julio de 2023

D. **Isabel Sánchez Berriel**, con N.I.F. 42.885.838-S, profesora Contratada Doctora adscrita al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutora.

D. **Fernando Andrés Pérez Nava**, con N.I.F. 42.091.420-V profesor Titular de Universidad adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como cotutor.

## **C E R T I F I C A ( N )**

Que la presente memoria titulada:

*“Desarrollo de videojuego RPG en La Laguna, ambientado en el siglo XVI”*

ha sido realizada bajo su dirección por D. **Gabriel Melián Hernández**, con N.I.F. 42.234.738-E.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 14 de julio de 2023

# Agradecimientos

A mis padres,

por siempre estar detrás de mí ayudando  
a levantarme después de cada caída  
y mostrando todo su apoyo y amor  
en todas las decisiones que he tomado.

A Cecilia,

por dar voz a este proyecto,  
por haber sido tan importante en mi camino y  
por haber querido siempre lo mejor para mí,  
sobran las palabras.

A Noelia,

por ser una increíble artista y amiga,  
por su aportación a este trabajo  
y por su apoyo siempre que lo necesité.

A Isabel,

Por su inestimable ayuda en este proyecto y  
por ser quien me enseñó  
el precioso mundo de la programación.

A Fernando,

Por su inestimable ayuda en este proyecto,  
por su tiempo y por ser un excelente profesor.

Al resto del profesorado,

Por su trato y todas las enseñanzas  
durante mi camino universitario.

# Licencia



© Esta obra está bajo una licencia de Creative Commons Reconocimiento 4.0 Internacional.

## Resumen

El conocimiento de nuestro pasado y la lucha por preservarlo para que pueda seguir siendo estudiado y admirado por las generaciones venideras, ha sido desde hace siglos un tema de gran importancia en la sociedad. El hecho de conocer nuestras raíces, como vivían nuestros antepasados y lo mucho que se ha avanzado hasta la actualidad, constituyen uno de los campos de investigación más amplios que existen.

Por otra parte, desde que somos pequeños se le da una gran importancia a la imaginación. Esa cualidad que nos permite en nuestra mente viajar a lugares que nunca hemos visitado, recrear épocas e historias que nunca hemos vivido e incluso en algunos casos, inventar aquello que aún no ha sido siquiera pensado.

En el marco que se crea con la unión de estos elementos, por un lado la imaginación y por otro la preservación de nuestra historia y el afán de conocimiento por nuestro pasado, nace la motivación y la idea de este proyecto: La creación de un videojuego de rol (RPG) que permita conocer desde otra perspectiva más amigable para el usuario nuestra historia.

San Cristóbal de La Laguna, es una de esas ciudades patrimonio de la humanidad que merece la pena ser estudiada en su pasado y preservada para su futuro. Se ha desarrollado una reconstrucción de esta ciudad ambientada en el Siglo XVI que recoge gran parte de las características que la hicieron pionera y referente en su época. Y no hay mejor manera de conocerla que pudiendo recorrer sus calles con total libertad y dejando volar un poco la imaginación para conocer cómo podría haber sido la vida en el día a día de uno de sus habitantes.

En este punto es en el que centra la gran parte de trabajo de este proyecto. La creación de un videojuego que te permita vivir de primera mano esas experiencias cotidianas, conocer la ciudad, conocer a sus habitantes e incluso ayudarlos en varias tareas y ganarte su gratitud.

También es conveniente destacar que este tipo de proyectos en la sociedad actual se rigen por el mercado y por sus consumidores, por lo que se ha realizado una investigación sobre las diferentes técnicas de financiación y distribución presentes que permitan que un proyecto de esta índole, se pueda llevar a cabo.

**Palabras clave:** San Cristóbal de La Laguna, videojuego, RPG, recreación, comercialización, financiación, mercado.

## Abstract

The knowledge of our past and the struggle to preserve it so that it can continue to be studied and admired by future generations has been a topic of great importance in society for centuries. The fact of knowing our roots, how our ancestors lived and how much progress has been made up to the present day, constitutes one of the most extensive fields of research that exist.

On the other hand, since we are children, great importance is given to imagination. That quality that allows us in our mind to travel to places we have never visited, to recreate times and stories we have never lived and even in some cases, to invent that which has not even been thought of yet.

In the framework that is created with the union of these elements, on the one hand the imagination and on the other hand the preservation of our history and the desire of knowledge of our past, the motivation and the idea of this project was born: The creation of an RPG video game that allows us to know our history from another more user-friendly perspective.

San Cristóbal de La Laguna is one of those world heritage cities that deserves to be studied in its past and preserved for its future. A reconstruction of this city set in the sixteenth century has been developed that includes many of the characteristics that made it a pioneer and a reference in its time. And there is no better way to know it than being able to walk through its streets with total freedom and letting your imagination fly a little to know how life could have been in the daily life of one of its inhabitants.

At this point is the focus of much of the work of this project. The creation of a video game that allows you to live firsthand those daily experiences, know the city, meet its inhabitants and even help them in various tasks and earn their gratitude.

It is also worth noting that this type of projects in today's society are governed by the market and its consumers, so an investigation has been conducted on the different financing and distribution techniques present that allow a project of this nature to be carried out.

**Keywords:** San Cristóbal de La Laguna, video game, RPG, recreation, marketing, financing, market.

# Índice general

<b>Capítulo 1 Introducción</b>	<b>1</b>
1.1 Antecedentes	2
1.2 Objetivos	3
1.3 Alcance y motivación.	4
<b>Capítulo 2 Herramientas Utilizadas.</b>	<b>5</b>
2.1 Unity (versión 2019.4.9f1)	5
2.2 Blender.	6
2.3 Microsoft Visual Studio 2019 y C#.	7
2.4 HTML, CSS y JavaScript.	8
<b>Capítulo 3 Desarrollo del Proyecto</b>	<b>9</b>
3.1 Dinámicas de trabajo.	10
3.2 Movimiento.	12
3.3 Interacción con los NPC	14
3.3.1 Triggers	14
3.3.2 Colliders y Triggers en los NPC	14
3.3.3 GameControllers y Eventos.	16
3.3.4 NPCs.	17
3.4 Misiones	18
3.5 Sistema de Reputación.	21
3.6 Interacción con el entorno.	21
3.7 Extras.	22
3.7.1 Minimapa.	22
3.7.2 Menú.	23
3.7.3 Sonido.	25
3.8 Sitio Web.	26
3.9 Pruebas y Rendimiento.	28
<b>Capítulo 4 Comercialización.</b>	<b>31</b>
4.1 Financiación.	31

4.2 Distribución mediante Play Store.	33
4.3 Distribución mediante Steam.	35
4.4 Distribución mediante Epic Games.	37
<b>Capítulo 5 Conclusiones y líneas futuras</b>	<b>40</b>
<b>Capítulo 6 Summary and Conclusions</b>	<b>41</b>
<b>Capítulo 7 Presupuesto</b>	<b>42</b>
7.1 Tabla de Presupuesto.	42
<b>Bibliografía.</b>	<b>43</b>



# Índice de figuras

Figura 1.- Entorno final, representación de La Laguna.	1
Figura 2.- Baldur's Gate 3, aclamado videojuego RPG (imagen accesible desde aquí).	2
Figura 3.- Plano de San Cristóbal de La Laguna, realizado por Leonardo Torriani [4].	2
Figura 4.- Pantalla del editor de Unity correspondiente a la versión utilizada.	5
Figura 5.- Pantalla de la vista del editor de Blender.	7
Figura 6.- Vista del entorno final utilizado.	9
Figura 7.- Entorno de pruebas utilizado durante el desarrollo.	11
Figura 8.- NavMesh utilizada en el proyecto. (región azul)	13
Figura 9.- Estructura de Triggers utilizada. Trigger de Aviso (exterior) y Trigger de Diálogo (interior)	15
Figura 10.- Sistema de interacciones con collider y trigger de diálogo para un NPC.	16
Figura 11.- Estructura y contenidos de un GameController.	17
Figura 12.- Construcciones que forman parte de la misión.	19
Figura 13.- Elementos recolectables.	19
Figura 14.- NPC Secundario.	20
Figura 15.- Alcalde y vista del Ayuntamiento.	21
Figura 16.- Triggers para los objetos recolectables.	22
Figura 17.- Vista del minimapa desarrollado.	23
Figura 18.- Vista del menú principal.	24
Figura 19.- Vista del menú de opciones.	25
Figura 20.- Vista de un AudioSource utilizado en el proyecto.	26
Figura 21.- Portada del prototipo de sitio web.	27
Figura 22.- Diseño responsivo del sitio web.	28
Figura 23.- Cuadro de rendimiento del proyecto.	29
Figura 24.- Captura de ejecución del videojuego, resultados finales.	30
Figura 25.- Captura de ejecución del videojuego, resultados finales.	30
Figura 26.- Fragmento de la página principal de IndieGogo.	31
Figura 27.- Página principal de la Steam Store.	35
Figura 28.- Página principal de la Epic Games Store.	38

# Índice de tablas.

Tabla 3.9: Resultados y Rendimiento.

28

Tabla 7.1: Resumen de tipos

42

# Capítulo 1 Introducción

En el mundo de los videojuegos, la posibilidad de sumergirnos en mundos virtuales ha captado la imaginación tanto de jugadores de todas las edades como de todo tipo de desarrolladores. Desde explorar vastos reinos de fantasía llenos de magia hasta participar en épicas batallas, los videojuegos RPG [1] han sido una ventana a experiencias interactivas inigualables. En este contexto, se presenta este proyecto de Trabajo de fin de grado (TFG), que se adentra en la creación de un videojuego RPG ambientado en la ciudad de San Cristóbal de La Laguna durante el siglo XVI.

San Cristóbal de La Laguna, una ciudad histórica situada en la isla de Tenerife, se erige como un escenario fascinante para explorar y vivir diversas aventuras virtuales. Su rica historia, arquitectura y cultura se mezclan en este videojuego que busca ofrecer una experiencia cultural al usuario y permitirle conocer un poco más en profundidad el espacio en el que se desarrolla el proyecto.

El objetivo principal, es permitir a los jugadores sumergirse en un entorno abierto donde podrán moverse con total libertad por el trazado de la ciudad. Mediante una cuidada recreación del paisaje urbano, los jugadores podrán explorar sus calles empedradas, admirar los edificios más emblemáticos de la ciudad en la época y sumergirse en una atmósfera histórica detallada.

No obstante, la experiencia va más allá de la mera exploración de la ciudad. En este RPG, los jugadores tendrán la posibilidad de entablar relaciones con los diversos personajes que habitan la ciudad. Cada uno de ellos poseerá su propia historia, motivación y demandas y será tarea del jugador interactuar con ellos y ayudarles a cumplir sus objetivos. A medida que se progresa, el jugador contribuirá al desarrollo y crecimiento de la ciudad, ganando estatus y reconocimiento en el progreso.

En resumen, el presente TFG tiene como objetivo llevar a los jugadores a una época fascinante y sumergirlos en la ciudad de San Cristóbal de La Laguna en el siglo XVI (Figura 1). Permitiendo explorar la ciudad con total libertad, establecer relaciones con los personajes locales y contribuir en el progreso de la ciudad. De esta manera, se busca fusionar la diversión y la inmersión virtual con el contexto histórico, brindando a los jugadores una experiencia de juego en una ciudad llena de historia y encanto.



*Figura 1.- Entorno final, representación de La Laguna.*

## 1.1 Antecedentes

La creación de videojuegos es un campo en constante evolución que combina la tecnología, la narrativa y la creatividad para brindar experiencias interactivas únicas a los usuarios. Más en concreto, los videojuegos de rol (RPG) han sido especialmente populares, ya que permiten a los jugadores sumergirse en mundos imaginarios y desempeñar roles protagonistas en aventuras épicas.

En el ámbito de los RPG [2], la libertad de movimiento y la capacidad para interactuar con personajes y entornos han sido elementos clave para ofrecer experiencias más inmersivas. Los juegos de mundo abierto han ganado popularidad, permitiendo a los jugadores explorar vastos entornos virtuales sin restricciones, desbloqueando nuevas áreas y completando misiones a su propio ritmo (Figura 2).



Figura 2.- Baldur's Gate 3, aclamado videojuego RPG (imagen accesible desde [aquí](#)).

En este contexto, se han desarrollado diversos videojuegos que recrean ciudades históricas, permitiendo a los jugadores sumergirse en el pasado y explorar su riqueza cultural. Estos juegos han demostrado ser una forma atractiva de educación y entretenimiento, fusionando la narrativa histórica con la jugabilidad interactiva, como por ejemplo la exitosa saga de videojuegos Assassin's Creed desarrollada por Ubisoft.

Sin embargo, hasta la fecha de corte de este trabajo, no se ha desarrollado ningún videojuego RPG ambientado en la ciudad de San Cristóbal de La Laguna durante el siglo XVI. Esta ciudad, declarada Patrimonio de la Humanidad por la UNESCO [3], presenta una rica historia y una arquitectura colonial única que ofrece un escenario perfecto para un videojuego de estas características (Figura 3).



Figura 3.- Plano de San Cristóbal de La Laguna, realizado por Leonardo Torriani [4].

El presente proyecto busca llenar ese vacío, brindando a los jugadores una oportunidad de explorar y experimentar la ciudad de San Cristóbal de La Laguna en una época fascinante. El enfoque en el siglo XVI, permitirá capturar la atmósfera de la época, con sus calles empedradas, plazas animadas y edificios históricos que cobrarán vida en el juego.

Además, el proyecto busca ofrecer una experiencia de juego más inmersiva al permitir a los usuarios establecer relaciones con los personajes de la ciudad. Estas interacciones añadirán una capa adicional de profundidad y realismo al juego, ya que los jugadores deberán cumplir las demandas y objetivos de los personajes no jugables para poder avanzar en la trama y contribuir al desarrollo de la ciudad.

Con el avance de la tecnología y el creciente interés en los videojuegos, este proyecto busca aprovechar las capacidades actuales para crear una experiencia de juego que combine la diversión y la inmersión histórica.

## 1.2 Objetivos

El objetivo principal de este proyecto sobre el que se centrará la base de todo el trabajo es el desarrollo de un videojuego de rol (RPG) ambientado en la ciudad de San Cristóbal de La Laguna en el siglo XVI, que brinde a los jugadores una experiencia inmersiva y envolvente. Se tratará de crear una recreación visualmente cautivadora y detallada de la ciudad, que capture la arquitectura, el entorno y la atmósfera histórica de la época, utilizando tecnologías y herramientas de diseño de vanguardia.

Además, como característica esencial de este tipo de videojuego, se presenta el objetivo de proporcionar a los jugadores una libertad de movimiento y exploración completa dentro del entorno virtual de la ciudad, permitiéndoles recorrer sus calles, plazas y lugares emblemáticos de manera realista y sin restricciones. Sin olvidar otro de los puntos esenciales de los videojuegos RPG, el diseño y desarrollo de una variedad de personajes no jugables (NPC [5]) con los que los jugadores podrán interactuar a lo largo del juego.

También será necesario implementar un sistema de misiones y objetivos para los NPC, que permita a los jugadores cumplir con las demandas y solicitudes de los personajes para progresar en la trama y contribuir al desarrollo y crecimiento de la ciudad. Todo el desarrollo debe ser testeado y refinado, con el objetivo de optimizar la jugabilidad, corregir errores y mejorar la experiencia general del juego, para en definitiva, lograr un producto final que combine de manera equilibrada la diversión y el entretenimiento de un videojuego contemporáneo con la autenticidad histórica y educativa, brindando a los jugadores una experiencia única y memorable en el contexto de la ciudad de San Cristóbal de La Laguna en el siglo XVI.

## 1.3 Alcance y motivación.

La motivación principal para llevar a cabo este proyecto es combinar la pasión por los videojuegos con el interés en la historia y la cultura de la ciudad de San Cristóbal de La Laguna. La creación de un videojuego de rol ambientado en esta ciudad durante el siglo XVI brinda la oportunidad de explorar y dar vida a su rica herencia histórica de una manera interactiva y atractiva.

Además, la falta de un videojuego de estas características centrado en la ciudad de San Cristóbal de La Laguna durante el siglo XVI crea un espacio para la innovación y la creatividad.

La creación de un entorno virtual detallado, permitirá a los jugadores explorar y aprender sobre la historia y la arquitectura de la ciudad. La combinación de elementos históricos auténticos con la narrativa y la jugabilidad de un videojuego de rol brinda la oportunidad de crear una experiencia de juego inmersiva y educativa que puede atraer a un público amplio, tanto jugadores ávidos como entusiastas de la historia.

Dentro de la complejidad presente en la realización de un proyecto de estas características, el objetivo del trabajo se enfocará en la realización de una base jugable sobre la que poder expandir el proyecto. Esto conlleva el desarrollo de todas las funcionalidades necesarias para el correcto funcionamiento e interacción de los diferentes personajes que formen parte del videojuego, así como las interacciones con los elementos del entorno.

Además de llevar a cabo, este tipo de implementaciones para el videojuego, se aportará una investigación sobre las diferentes opciones de financiación y distribución de un proyecto de este tipo de modo que se amplíen los conocimientos en el área de mercado la cual debe ser el objetivo final del desarrollo de un videojuego, así como también se realizará el desarrollo de un sitio web propio del proyecto donde un cliente potencial pueda acceder a toda la información necesaria sobre el funcionamiento y desarrollo del proyecto.

# Capítulo 2 Herramientas Utilizadas.

Para el desarrollo de los trabajos realizados en este proyecto, se ha hecho uso de diversas herramientas de software sobre las que se han realizado las tareas necesarias para llevar a cabo el desarrollo de este proyecto.

## 2.1 Unity (versión 2019.4.9f1)

Unity [6] es un motor de desarrollo de videojuegos multiplataforma ampliamente utilizado en la industria del entretenimiento interactivo. Fue creado por Unity Technologies y lanzado inicialmente en 2005, desde entonces y gracias al apoyo mostrado para el desarrollo de videojuegos independientes, se ha convertido en una de las herramientas más populares para desarrollar videojuegos, tanto para principiantes como para estudios de desarrollo de renombre.

Unity destaca por su flexibilidad y versatilidad, permitiendo a los desarrolladores crear juegos tanto en 2D como en 3D para una amplia variedad de plataformas, incluyendo PC, consolas, dispositivos móviles y realidad virtual. Ofrece un entorno de desarrollo integrado (IDE, por sus siglas en Inglés) que facilita la creación y edición de contenido, así como una amplia gama de funcionalidades y herramientas (Figura 4).

Una de las principales y más notables características de Unity, es su motor de físicas [7] en tiempo real, que permite simular interacciones realistas entre objetos y personajes en el juego, un aspecto de vital importancia en proyectos de este tipo donde el personaje principal del videojuego, está en constante interacción con el entorno que lo rodea. También ofrece un sistema de renderizado avanzado, soporte para animaciones, detección de colisiones, scripting mediante el uso de lenguajes como C# y JavaScript, indispensable para escribir y añadir nuestras propias funcionalidades a través de la codificación de scripts, gestión de assets (recursos del juego) y herramientas de exportación e importación entre otros.



Figura 4.- Pantalla del editor de Unity correspondiente a la versión utilizada.

Además, Unity cuenta con una comunidad activa y una amplia y detallada documentación que sirve de ayuda a los desarrolladores para aprender y resolver problemas que puedan encontrar en el desarrollo de sus proyectos. Existe también, una tienda de recursos (assets) en línea donde los usuarios pueden adquirir recursos adicionales, como modelos 2D o 3D, efectos visuales, sonidos y scripts, tanto de pago como gratuitos, que permiten al desarrollador facilitar la creación de videojuegos y mejorar su calidad sin la necesidad de desarrollar todo desde cero.

Por otra parte, la versión específica de Unity utilizada es la, 2019.4.9f1, se trata de una versión lanzada en el año 2019. Unity utiliza un sistema de numeración para sus versiones, en el cual el primer número representa la versión principal del motor, el segundo número indica actualizaciones importantes y el tercer número incluye correcciones y mejoras menores.

La versión 2019.4.9f1 es parte de la rama “Long-Term Support”(LTS) de Unity 2019, lo que significa que está diseñada para ofrecer estabilidad y soporte a largo plazo en lugar de introducir nuevas características. Esta rama es ideal para proyectos que están en una etapa avanzada de desarrollo y requieren de una base sólida y confiable sobre la que sustentar su estabilidad, tal y como es el caso de este proyecto.

La “f1” al final del número de versión indica que es una revisión específica dentro de la serie 2019.4 y se utiliza para diferenciar las actualizaciones menores y las correcciones de errores. Estas revisiones se lanzan regularmente para abordar problemas identificados y mejorar la estabilidad general del motor.

La elección de Unity para la realización del proyecto frente a otros motores, como puede ser Unreal Engine, se debe a que el proyecto se encuentra iniciado y desarrollado en esta herramienta, estando la representación de la ciudad de San Cristóbal de La Laguna en el siglo XVI, lugar en el que se va a desarrollar el videojuego, ya creada y adaptada para esta herramienta.

Dentro de todas las características que ofrece Unity como entorno de desarrollo, se han utilizado diversas herramientas propias del motor, como el uso de assets, tanto desarrollados para Unity por terceros o propios de Unity Technologies haciendo uso de la “Unity Asset Store”, como extraídos de páginas externas. Algunos de estos assets han sido utilizados para generar elementos con los que el usuario puede interactuar en algunas misiones, como troncos o piedras y por otra parte también para recrear la estructura de un mercado de la época.

También se han utilizado herramientas propias e integradas en el motor, como el visor de estadísticas en tiempo real o el Analizador de rendimiento, imprescindibles para las tareas de optimización. El gestor de navegación que ha permitido ajustar los espacios transitables dentro del entorno. La iluminación, donde se han ajustado los efectos de luces, niebla y mapas de luces (Baking [8]) y reducir la sobrecarga de cálculos de luces en tiempo real. La eliminación selectiva (Occlusion culling [9]), que se ocupa de reducir vértices y desactivar temporalmente los elementos fuera de nuestro ángulo de visión.

## **2.2 Blender.**

Blender [10] es un software de creación y animación 3D de código abierto. Fue desarrollado por Blender Foundation en 1994 y desde entonces, se ha convertido en una herramienta completa y potente utilizada en diversas industrias, como, los videojuegos, la animación, el diseño e impresión en 3D o incluso el cine.

Blender ofrece una amplia gama de herramientas y funcionalidades para creación de



elementos en 3D, incluyendo modelado, escultura, texturizado, animación y simulación de físicas entre otras. El modelado en Blender permite a los usuarios crear y modificar todo tipo de geometrías 3D y ofrece diversas técnicas de modelado, como el poligonal, por subdivisión o por curvas, lo cual aporta flexibilidad y versatilidad a la hora de la creación de elementos en 3D. También, Blender ofrece herramientas para aplicar texturas, pintar directamente sobre modelos 3D, crear materiales realistas y simular superficies complejas. La animación en Blender se basa en la técnica de animación por keyframes, donde los usuarios establecen posiciones clave en el tiempo para los objetos y personajes en la escena. Blender también admite la animación basada en físicas y la animación de personajes mediante el uso de esqueletos y rigging (Figura 5).

Además, Blender incluye herramientas para realizar simulaciones de físicas, como la simulación de partículas, fluidos, cabello y telas. Estas características permiten crear efectos realistas y dinámicos en las escenas.

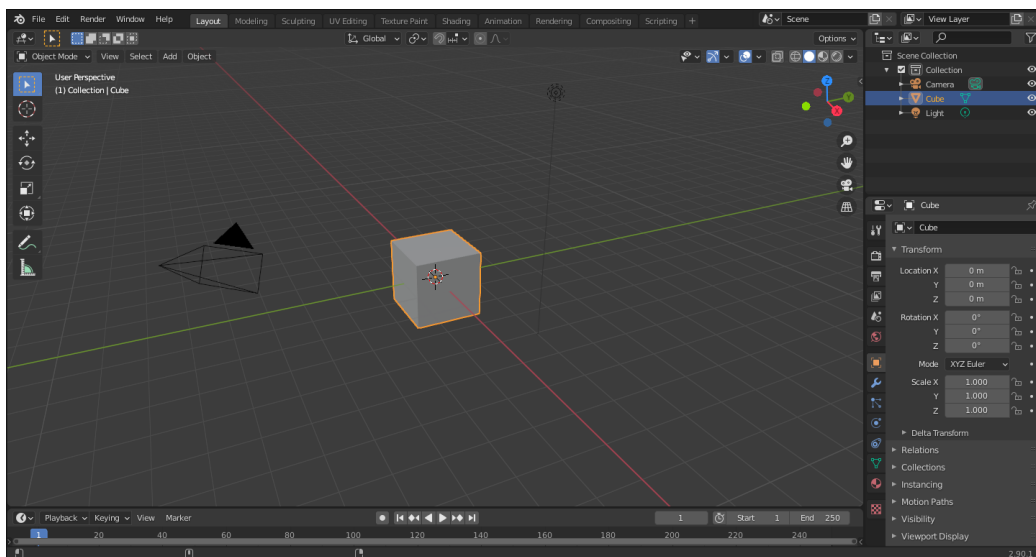


Figura 5.- Pantalla de la vista del editor de Blender.

## 2.3 Microsoft Visual Studio 2019 y C#.

Estas dos herramientas se han utilizado en este proyecto, para llevar a cabo en su totalidad, el trabajo de scripting necesario para la creación de funcionalidad propia y a medida para el proyecto. Este tipo de funcionalidad abarca: el comportamiento de personajes, tanto el principal como los diversos NPCs, la activación o desactivación de elementos del entorno y cualquier tipo de interacción posible del personaje principal con los demás NPC presentes o con el propio entorno en el que se desarrolla el videojuego. Se encarga además de la creación de controladores que se encargarán de que el videojuego siga correctamente su flujo establecido, controlando el comportamiento del mundo ante los diversos eventos que puedan suceder y se hayan implementado.

Microsoft Visual Studio, es un entorno de desarrollo integrado (IDE), que destaca por ser compatible con una gran cantidad de lenguajes de programación diferentes como C++, C#, Java, Python, etc. Su utilización e integración con Unity para la creación de scripts es muy sencilla y funcional y permite crear y mantener archivos de proyectos

automáticamente. Otra de las ventajas que posee Microsoft Visual Studio es la existencia de una versión Community gratuita que permite la integración completa con Unity.

Por su parte, C# es el lenguaje de programación principal utilizado por Unity, derivando en la mayoría de escenarios de la clase MonoBehaviour, que contiene la mayoría de propiedades y métodos específicos para el desarrollo en Unity. C#, es un lenguaje de programación multiparadigma (estructurado, imperativo, orientado a objetos, dirigido por eventos, funcional, genérico y reflexivo).

## **2.4 HTML, CSS y JavaScript.**

Uno de los puntos importantes de este proyecto, es no solo el desarrollo del mismo, sino ir un paso más allá e indagar en el aspecto comercial de un proyecto de este tipo. Para ello además de las diferentes estrategias de financiación o comercialización, es conveniente disponer de un espacio en el que se proporcione información relevante sobre el proyecto a todos aquellos usuarios que estén interesados, ya sea en fase de desarrollo o posterior al lanzamiento del proyecto.

Por este motivo, se aborda la creación de un sitio web propio para el proyecto en el que se almacene todo tipo de información relevante acerca del mismo y que se convierta en el punto de encuentro de todos los potenciales jugadores del videojuego desarrollado.

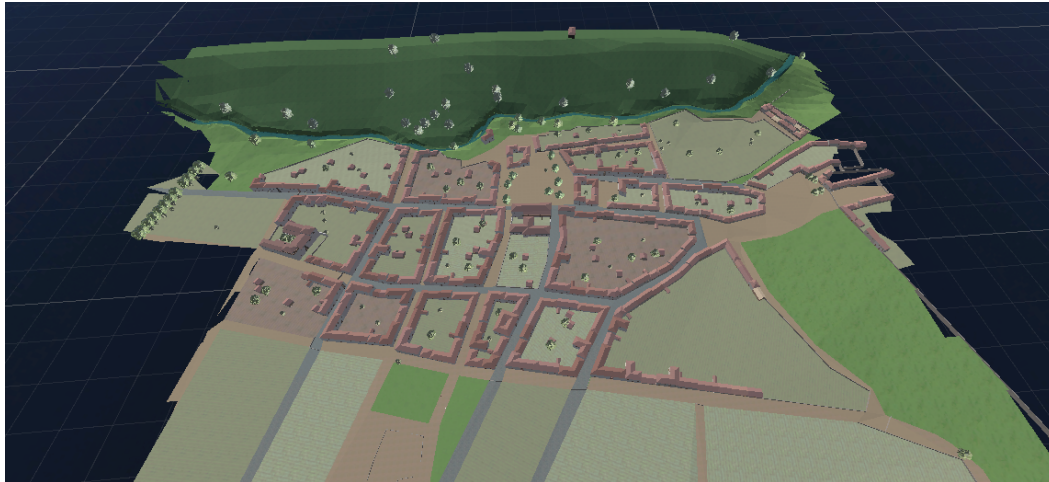
Este sitio web se ha desarrollado desde cero a medida para el proyecto y como la mayoría de los sitios webs actuales está desarrollado haciendo uso de HTML para definir la estructura del contenido web, así como los comportamientos y enlaces que conectan las páginas web entre sí, ya sea dentro de un único sitio web o entre varios.

HTML, es un lenguaje de marcado que nos permite indicar la estructura de nuestro documento mediante etiquetas y referencias a las diferentes ubicaciones de elementos, de modo que la página web solo contiene texto y el navegador web es el encargado de unir todos los contenidos en una página web final.

Como es habitual en las páginas web actuales, además de HTML, se utilizan diversos lenguajes de programación para añadir estilo o funcionalidad extra. En el caso de este proyecto se ha utilizado el lenguaje de estilos CSS, para definir de forma personalizada y propia ,el estilo que contendrá el sitio web que se ha creado. Por otra parte también se ha hecho uso de JavaScript como lenguaje de apoyo para incluir funcionalidades extra en diversos comportamientos del sitio web.

# Capítulo 3 Desarrollo del Proyecto

El contenido principal de trabajo realizado para el desarrollo de este proyecto, ha sido la implementación de la funcionalidad necesaria para el correcto funcionamiento de un videojuego de tipo RPG ambientado en la ciudad de San Cristóbal de La Laguna, haciendo uso del entorno proporcionado (Figura 6) por el proyecto “Reconstrucción Histórica Virtual de San Cristóbal de La Laguna” [11].



*Figura 6.- Vista del entorno final utilizado.*

A su vez, se ha realizado la implementación de una página web que contendrá información sobre el proyecto a modo de ejemplo para simular los contenidos necesarios en una página web de este tipo de videojuegos.

Por último se ha realizado una investigación de los posibles acercamientos al mercado para este tipo de proyectos, desde la perspectiva del desarrollador independiente.

Cabe mencionar en cuanto al nivel de conocimientos previos antes de comenzar con el desarrollo del proyecto, que se parte de los conocimientos adquiridos durante la realización del grado de Ingeniería Informática. En especial, aquellos obtenidos en la asignatura de Interfaces Inteligentes, donde se impartieron los primeros pasos y conocimientos sobre el motor Unity que se utiliza en este proyecto, así como el uso del lenguaje de programación C#, para llevar a cabo todo el proceso de scripting necesario para aportar funcionalidad propia a los proyectos desarrollados.

Otros conocimientos de Unity acerca de funcionalidades extra o más específicas para este tipo de proyectos de videojuegos se han desarrollado por mero interés personal en la materia y a través de la documentación oficial de Unity disponible en internet y diversos videos y tutoriales de otros proyectos, con explicaciones muy interesantes y educativas sobre las técnicas y herramientas utilizadas en ellos.

Además durante el período de prácticas externas realizadas en la empresa colaboradora Bakata Solutions S.L se han recibido sesiones tutorizadas sobre Unity y se han ampliado conocimientos y ganado experiencia al participar en proyectos para la empresa colaboradora en que se llevaron a cabo dichas prácticas.

Los conocimientos sobre Blender, han sido adquiridos de forma autodidacta por interés personal además de haber contado con varias sesiones de tutorización por profesionales del sector durante el período de prácticas externas.

Por último pero no menos importante, los conocimientos adquiridos en el uso de HTML, CSS y JavaScript para el desarrollo de páginas web han sido obtenidos, en parte durante la realización del grado en la asignatura de Sistemas de Interacción Persona-Computador y la asignatura de Procesadores de Lenguajes y por otra parte mediante el uso de tutoriales y documentación sobre el uso de dichos lenguajes.

## **3.1 Dinámicas de trabajo.**

Para la realización de este proyecto una vez conocido el tema sobre el que trabajar y el alcance que iba a tener, se decidió llevar a cabo sesiones semanales de seguimiento del trabajo realizado, para poder mantener una correcta comunicación con el profesorado encargado de tutorizar el proyecto y poder exponer y solucionar de forma rápida y efectiva las posibles dudas que pudieran surgir durante el transcurso de la semana y poder marcar objetivos e hitos a corto plazo para avanzar de forma organizada en el desarrollo del proyecto siempre con un punto de destino en común.

En estas reuniones semanales se presentaba y probaba el correcto funcionamiento de todo el trabajo desarrollado durante la semana, corrigiendo aquellos errores que pudieran haberse cometido y se marcaba una hoja de ruta para las siguientes sesiones de seguimientos, que favorecen por otra parte la resolución de dudas previas al comienzo de desarrollo de un punto en el proyecto.

Independientemente de la realización de estas reuniones semanales siempre ha existido comunicación directa con el profesorado que tutoriza el proyecto para cualquier duda o sugerencia adicional a aquellas presentadas durante las sesiones de seguimiento.

Por otra parte y en cuanto al desarrollo del proyecto, en primera instancia se decidió abordar en su totalidad y con principal prioridad el desarrollo e implementación de una versión base y funcional del videojuego a partir de la cual a posteriori se podría añadir cualquier extra que fuera necesario o se considerase oportuno añadir, además de realizar las tareas de implementación de la página web y la investigación acerca de los diferentes modelos de comercialización.

Para ello en un primer momento se realizó una lluvia de ideas sobre las posibles características que debería tener el proyecto de forma que fuera más sencillo abordar desde un primer momento aquellas que tuvieran mayor peso en un proyecto de este tipo, siendo por ejemplo, la libertad de movimiento o la interacción con los diferentes personajes esparcidos por el entorno, dos de los pilares fundamentales en los videojuegos de tipo RPG.

Una vez definidas estas tareas esenciales, a la hora de implementarlas y comenzar su desarrollo se decidió actuar de forma que:

1. Se analiza la característica a implementar, para conocer su contexto y su finalidad.
2. Identificar limitaciones y problemática que pueda presentar dicha característica.

3. Elaborar diferentes aproximaciones para la implementación.
4. Implementación de aquella opción que se considere más óptima.
5. Evaluación y prueba de los resultados obtenidos.

Este esquema de trabajo se utilizó para llevar a cabo la implementación de cada una de las funcionalidades que se iban a añadir al proyecto. Comenzando siempre en un entorno de prueba (Figura 7) donde agilizar la carga y ejecución de los test y que no hubiera riesgo de perder o provocar errores en el entorno final proporcionado por el proyecto de “Reconstrucción Histórica Virtual de San Cristóbal de La Laguna”.

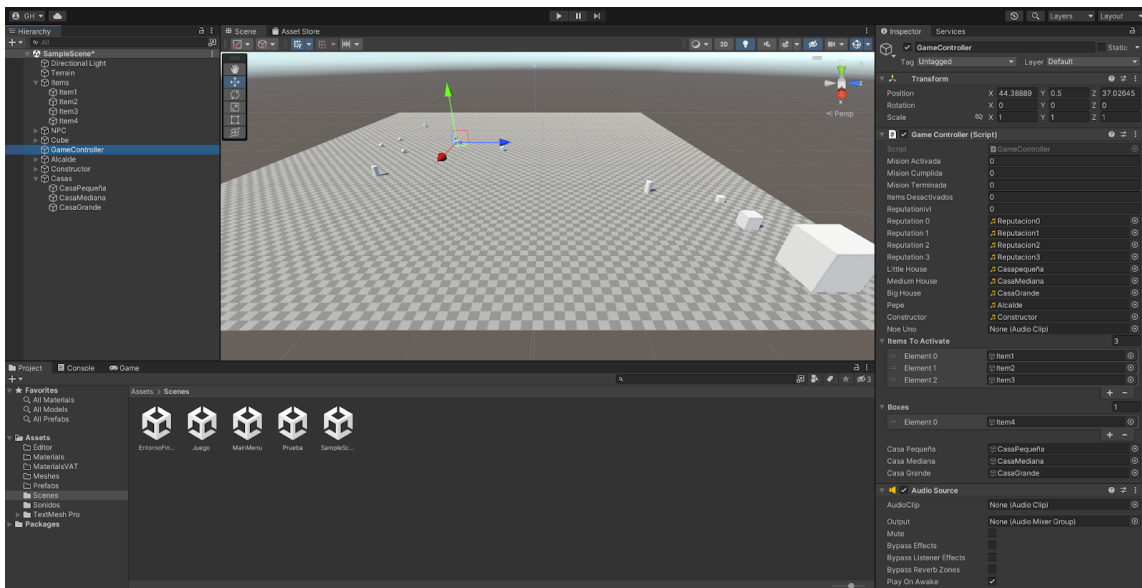


Figura 7.- Entorno de pruebas utilizado durante el desarrollo.

Una vez que se completaba el desarrollo e implementación de una nueva funcionalidad en dicho entorno de pruebas, se añadía al entorno al final donde se comprobaba que efectivamente no se había trasladado ningún error o fallo de programación que pudiese comprometer el desarrollo del proyecto y diera lugar a necesitar rehacer dicha funcionalidad.

Las diferentes tareas que se consideraron esenciales para el desarrollo del proyecto y en las que se ha enfocado el principal esfuerzo de trabajo fueron las siguientes:

- **Movimiento Libre:** Los videojuegos de tipo RPG se caracterizan por dar la libertad al usuario de avanzar con total libertad en el escenario presentado, pudiendo avanzar en la historia de la manera que considere oportuna.
- **Interacción con los NPC:** Otra característica indispensable en un RPG, es la posibilidad de interactuar con los diferentes NPC que formen parte del entorno, ya sea con diálogos que aporten información a la historia o por ejemplo mercaderes con los que poder comerciar mediante el intercambio de elementos en el juego.

- **Misiones:** Una de las formas de interacción más usuales de este tipo de títulos son las misiones. Diversas tareas que los NPC encomiendan al usuario y que normalmente presentan una recompensa en el momento de su finalización.
- **Sistema de Reputación:** En proyectos profesionales de este tipo, el usuario siempre tiene un objetivo final y necesita cumplir diversos pequeños hitos para poder avanzar y desbloquear nuevo contenido. En el caso de este proyecto se ha optado por implementar un nivel de reputación que se incrementa con la finalización de misiones y desbloquee nuevas interacción de forma que sea visible un progreso en el juego.
- **Interacción con el entorno:** Normalmente no solo se interactúa con los personajes disponibles en el videojuego, sino que además gracias a ellos se habilita la interacción con diferentes objetos presentes en el entorno que servirán de apoyo para el progreso en el videojuego.
- **Extras:** Otros elementos a tener en cuenta y que están presentes en este tipo de videojuegos pueden ser un minimapa o una estructura de menú donde puedas configurar alguna opción como el volumen de los diálogos o la música ambiental presente durante el transcurso del videojuego.

## 3.2 Movimiento.

Al llevar a cabo el movimiento libre característico de los RPG se llegó a una serie de problemas que obligaron a modificar el enfoque inicial que se había definido.

En un primer momento se decidió llevar a cabo una versión de movimiento aplicando físicas, haciendo uso del rigidbody, una característica presente en Unity, que permite a los GameObjects actuar bajo el control de las físicas. Un rigidbody [12] puede recibir una fuerza y aceleración para conseguir que el objeto que lo utiliza se mueva de una manera más realista. Además, cualquier objeto que contenga un rigidbody puede ser influenciado por la gravedad y por las fuerzas que defina el desarrollador vía scripting.

La implementación de este sistema en el entorno de pruebas se consiguió realizar sin mayores dificultades, aunque por el contrario originó ciertos problemas al transferir dicha funcionalidad al entorno final.

Cabe recordar que el entorno final utilizado forma parte del proyecto, “Reconstrucción Histórica Virtual de San Cristóbal de La Laguna”, que se encuentra en una fase de desarrollo avanzado y al cual es necesario adaptar toda la funcionalidad ya implementada. En este caso y en lo referente al movimiento, en el entorno final del proyecto para asegurar un mejor rendimiento y eliminar en gran medida el número de colliders que se cargan, se optó por eliminar las colisiones con todos los elementos que forman parte del entorno. Para asegurar que el tránsito se realiza de forma correcta, se delimitó el espacio transitable haciendo uso de la herramienta de navegación de Unity y creando una “Nav Mesh” [13] para todo el entorno de forma que no existiera nunca la posibilidad de tocar aquellos elementos cuyos colliders habían sido eliminados (Figura 8).

Esta técnica funciona perfectamente para conseguir un mejor rendimiento y además permitir el movimiento de personajes controlados por IA haciendo uso de la herramienta “NavMesh Agent” [14], la cual permite al personaje moverse por la escena utilizando la malla de navegación, pero sin generar una colisión física con la misma.



Figura 8.- NavMesh utilizada en el proyecto. (región azul)

Esto generó el problema de que al transferir el personaje principal con la funcionalidad propia desarrollada para el uso del rigidbody, al no existir ningún tipo de collider, el personaje se viese afectado por la gravedad y cayese hacía el infinito atravesando cualquier elemento del entorno.

La solución que se llevó a cabo para poder realizar la implementación del movimiento en el videojuego, fue transformar al personaje principal en un “navmesh agent”, al igual que los NPC. De esta forma la malla de navegación creada actúa como freno para la fuerza de la gravedad y se puede mantener el modelo de movimiento físico sin necesidad de revertir los cambios en los colliders e incrementar la carga en ejecución, trayendo consigo un peor rendimiento.

Otro de los elementos que se tuvo que modificar para que esta nueva aproximación funcionase de manera correcta fue el control del “drag”, que es la resistencia del aire al moverse el objeto como consecuencia de una fuerza aplicada al mismo. En este caso, sufrió un incremento para asegurar que la velocidad de movimiento del personaje fuera siempre la misma, impidiendo que se acelerase de manera infinita al no tener contacto directo con el suelo.

Unity utiliza un motor de físicas que simula cómo los diferentes materiales interactúan entre sí. En un caso normal, para aquellos elementos que no tendrán movimiento en el entorno, se coloca en ellos un “static collider” que simulará sus físicas ante colisiones pero no se moverán en respuesta a ellas. Por otra parte, a aquellos elementos que se moverán por el entorno, se les aplica un rigidbody y sus colliders son conocidos como “dynamic colliders”. Estos dos tipos de colliders pueden interactuar entre sí sin problemas y cuando lo hacen sus superficies necesitan simular las propiedades del material al que representan, por ejemplo un trozo de hielo y un trozo de asfalto tendrán diferentes propiedades y mientras que el primero será resbaladizo el segundo no debería serlo.

Este tipo de comportamiento se pierde en este proyecto al no poseer colliders que puedan generar esa interacción entre el rigidbody del personaje y el collider del suelo, de forma que es necesario crear una falsa ilusión al usuario de que si está en contacto con el suelo. En este caso, se optó por el incremento en el drag, limitando la velocidad y aceleración que puede conseguir el personaje y eliminado el efecto de deslizamiento producido por no estar en contacto con una superficie de alta fricción, pero sustituyéndolo por una mayor resistencia al aire que frene al personaje.

## **3.3 Interacción con los NPC**

Como se mencionó con anterioridad, la interacción del personaje principal con los NPC presentes en el entorno, es uno de los puntos más importantes en este tipo de videojuegos. Para realizar estas interacciones y la funcionalidad ligada a ellas, se optó por un modelo de interacción basado en “Triggers” [15] y en “Eventos” controlados por un “controller” independiente para cada uno de los personajes presentes con los que el usuario principal pudiera interactuar.

Cabe destacar que al igual que pasaba con el movimiento anteriormente, toda la funcionalidad desarrollada referente a la interacción con los personajes del entorno, fue previamente desarrollada y comprobada en el entorno de pruebas antes de su transferencia al entorno final, donde a diferencia del movimiento libre, en este caso no se detectó ningún problema o error al introducir toda esta nueva funcionalidad en el entorno final.

### **3.3.1 Triggers**

Haciendo uso del sistema de scripting y del motor de colisiones, se pueden detectar colisiones entre dos objetos con la función “OnCollisionEnter”. Pero por otra parte también es posible simplemente usar el motor de físicas para detectar cuando un collider entra al espacio de otro sin necesidad de crear una colisión física. Esto se consigue configurando como trigger a un collider convencional (activando la propiedad “is Trigger”) de manera que este no se comporte como un objeto sólido sino que permita que otros objetos puedan pasar a través de él sin generar una colisión. De esta manera mediante scripting, gracias a la función “OnTriggerEnter” podemos conocer cuando un collider entra en el espacio del trigger y lanzar la acción correspondiente.

### **3.3.2 Colliders y Triggers en los NPC**

En el caso de este proyecto, tal y como se ha comentado anteriormente, los NPC presentes en el entorno van a ser “navmesh agents”, pero esta condición no inhabilita la capacidad de los mismos para poseer colliders. Cada uno de estos personajes va a presentar varios colliders y triggers que serán los encargados de que el NPC reaccione de la forma adecuada frente a las acciones del personaje principal (Figura 9).



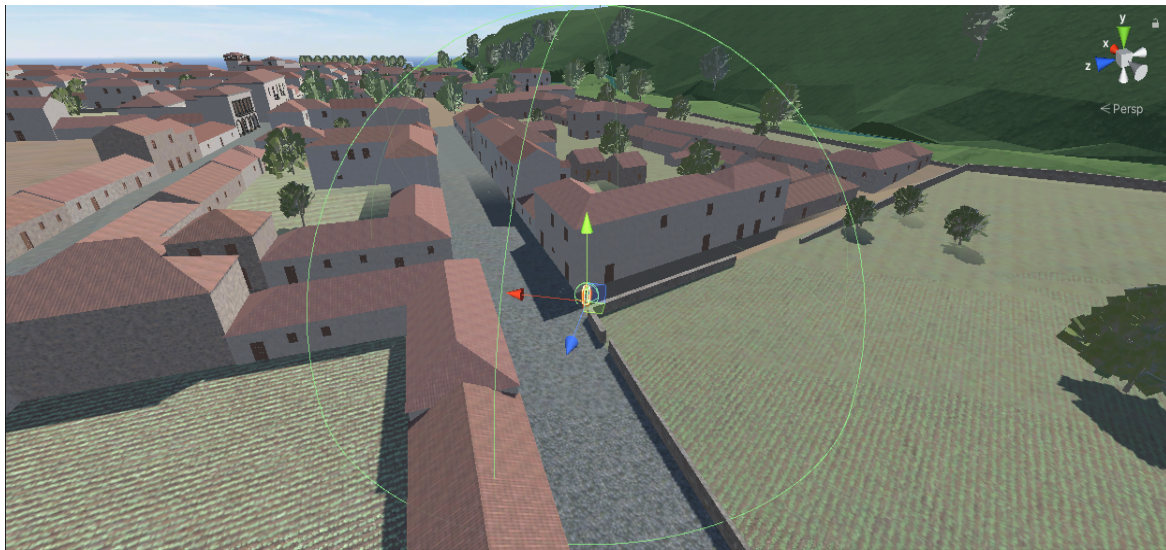


Figura 9.- Estructura de Triggers utilizada. Trigger de Aviso (exterior) y Trigger de Diálogo (interior)

- **Collider Físico:** Aunque tanto el personaje principal como los NPC tienen la condición de navmesh agent, ambos estarán dentro de la zona transitable de la malla, por lo que se vuelve necesario poseer algún mecanismo que impida que se puedan atravesar entre sí y que se genere la colisión correspondiente entre ambos. Para conseguirlo, tanto el personaje principal como todos los demás NPC, tendrán un collider básico que permita este tipo de colisiones entre ellos sin que se produzca ningún tipo de acción adicional.
- **Trigger de aviso:** Se llegó a la conclusión, de que existía la necesidad de que algunos de los personajes tuviesen la capacidad de llamar la atención del usuario debido a que la interacción con ellos puede ser importante para el correcto avance y desarrollo de la experiencia de juego. Aunque haya total libertad de movimiento, es conveniente establecer una pequeña guía que ayude al usuario a conocer los puntos importantes a visitar para avanzar. Para este primer tipo de interacción entre el usuario principal y los NPC, es de gran ayuda el uso de los triggers. Más en concreto, se ha utilizado en la implementación un collider de esfera que ha sido ampliado en tamaño hasta una distancia que simula el momento en el que el NPC establecería contacto visual con el personaje principal y captaría su atención. De modo que cuando el usuario entre en el espacio del trigger active la funcionalidad asociada y se produzca la reacción deseada.
- **Trigger de Diálogo:** Otro de los triggers que presentarán los diferentes NPC disponibles en el entorno, activará la ejecución de los diálogos disponibles para cada uno de ellos. En este caso se ha utilizado al igual que en el caso anterior un collider de esfera pero de un tamaño mucho menor con la propiedad de trigger activada, para que se ejecute en el momento en el que el personaje principal se encuentre cerca y en disposición de mantener una conversación con el NPC que corresponda. Siempre que se entre en uno de estos triggers, se activará el diálogo correspondiente al estado de la interacción actual (Figura 10).

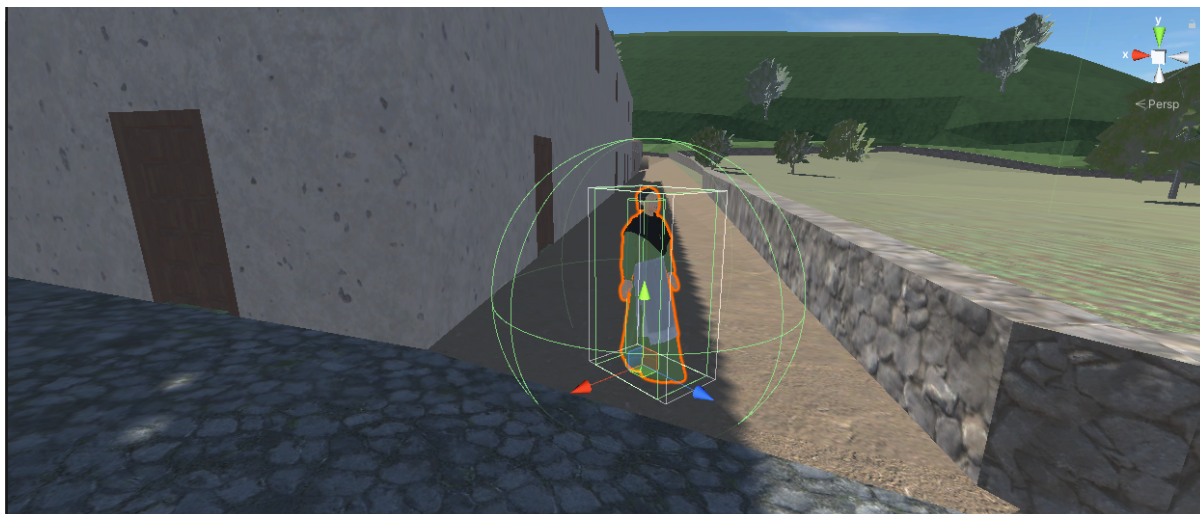


Figura 10.- Sistema de interacciones con collider y trigger de diálogo para un NPC.

### 3.3.3 GameControllers y Eventos.

Una de las técnicas más usadas en desarrollo de videojuegos es el uso de los eventos. En el instante en el que ocurre una determinada acción, se ejecuta una determinada consecuencia en forma de función desarrollada mediante scripting y esta forma de trabajar es la que se ha elegido para llevar a cabo las interacciones en este proyecto.

Para tener un mejor control de lo que sucede durante el transcurso del videojuego, además de conseguir mayor claridad y organización en el código, se ha optado por crear para cada uno de los personajes un controlador independiente, en el que se almacenen e implementen todos los eventos, funciones e interacciones que va a presentar cada uno de los personajes, de forma que conseguiremos un código mucho más modular y correcto (Figura 11).

Por otra parte, para llamar a estos eventos cuando se produzcan, se hará uso de las funciones Delegate [16]. Un Delegate es una referencia a un método. Nos permite tratar ese método como una variable y pasarlo como variable para un callback. Cuando se invoca, notifica a todos los métodos que hacen referencia al Delegate. El funcionamiento es el del patrón Observador [17], donde se crea una dependencia uno a muchos, de manera que cuando uno cambia su estado, notifica este cambio a todos los dependientes. Un ejemplo práctico del mundo real, podría ser un periódico. Una persona puede suscribirse a dicho periódico y cada vez que salga una nueva entrega todas las personas suscritas la recibirán. Los eventos [18] añaden una capa de abstracción y protección al Delegate. Esto previene al cliente del Delegate de resetearlo y eliminar la lista de suscriptores.

En este proyecto, los diferentes triggers creados en el punto anterior tendrán asociado un script cada uno con la creación de un determinado evento y una determinada función delegate. A su vez el controlador asociado a cada personaje se suscribirá a cada uno de los eventos que tenga asignados el personaje correspondiente e implementará la funcionalidad necesaria para que se ejecute correctamente la acción correspondiente al evento activado.

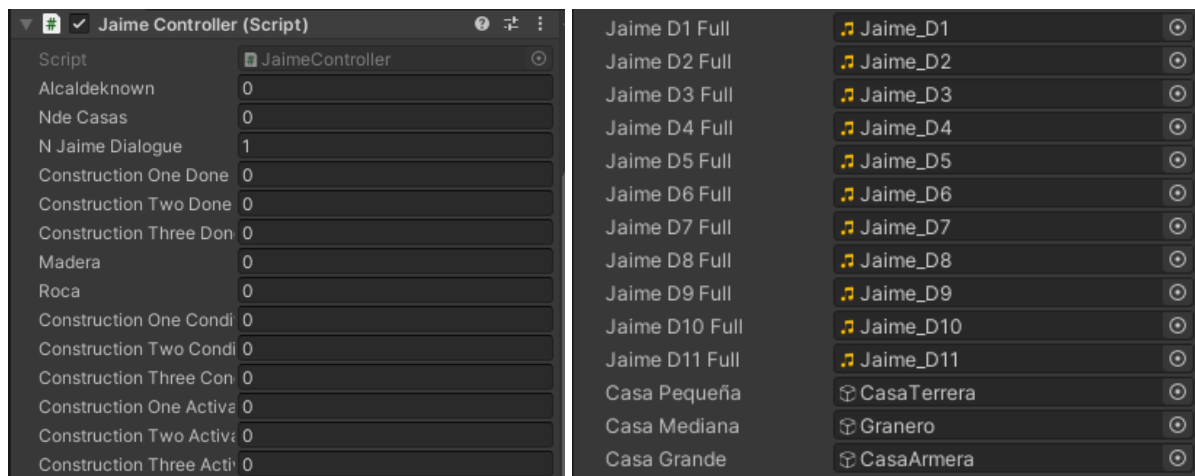


Figura 11.- Estructura y contenidos de un GameController.

En el caso de la interacción con los personajes el controlador de cada personaje estará suscrito a los eventos de diálogos del mismo. Estos diálogos se ejecutarán cada vez que el personaje principal, controlado por el usuario, entre en uno de los triggers correspondientes a ese personaje y ejecutarán en primer lugar el diálogo introductorio del personaje y posteriormente el diálogo correspondiente dependiendo del estado actual de progreso en el videojuego. Existen determinados diálogos que no se ejecutarán hasta que se den condiciones concretas u otros que se ejecutarán en bucle hasta que se cumpla con algún objetivo de modo que sea necesario progresar en el desarrollo del juego para conocer todos los detalles disponibles para cada uno de los personajes.

Además de estos eventos necesarios para activar la funcionalidad de los personajes, dentro de cada uno de los controladores también se encuentran todas aquellas flags y variables necesarias para llevar el control del avance del videojuego y que serán las encargadas de mantener interacciones específicas bloqueadas hasta que se cumpla una determinada condición, al igual que también se incluye en estos scripts el código correspondiente a la implementación de las diferentes misiones de las que hablaremos en el próximo apartado.

### 3.3.4 NPCs.

A lo largo de la ejecución del videojuego, el usuario se encontrará con diversos personajes no jugables, con los que podrá interactuar y que serán los encargados de conseguir la progresión necesaria para el correcto desarrollo de este tipo de proyectos. En este caso el usuario podrá interactuar con los siguientes personajes:

- **Campesina Introductoria:** Este personaje será el primero que encuentre el usuario en el momento en el que comience la ejecución del videojuego. Su función principal es la de introducir al usuario al contexto en el que se sitúa el entorno y guiarlo hacia el siguiente personaje importante.
- **Alcalde:** Situado frente a las puertas del ayuntamiento, el alcalde es el personaje sobre el que gira el sistema de reputación y progreso del videojuego. Es el encargado de guiar al usuario hacia alguna de las misiones y será quién, una vez

completadas, entregará al usuario la pertinente recompensa por sus labores, creando de esta manera un punto de encuentro para el usuario siempre que realice correctamente una misión y quiera seguir progresando.

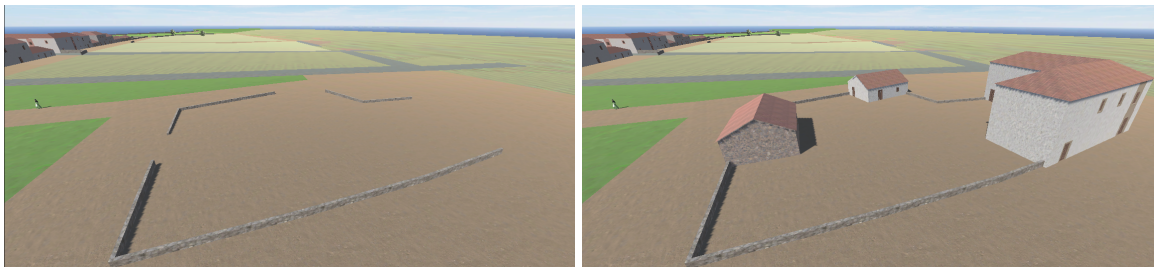
- **Campesina en apuros:** Situada en una de las casas de la plaza del Adelantado, este personaje se encargará de ofrecer al usuario una serie de diálogos para conocer su contexto, además de ser la encargada de activar una de las misiones existentes en el videojuego. A medida que se avanza en el mismo y cuando se completen todas las misiones existentes, se desbloquearán nuevas interacciones con este personaje.
- **Cura:** Constituye otro de los personajes que activan una misión para el usuario y de él dependen las monjas, sus ayudantes para completar la misión. Se encuentra situado en la iglesia de la plaza del Adelantado.
- **Monjas:** Son los únicos NPCs implementados que no presentan líneas de diálogo con el usuario y forman parte de la misión del cura, siendo los diferentes puntos de destino a los que tiene que acudir el usuario. Están esparcidos por diferentes casas de la ciudad y son cinco personajes en total.
- **Constructor:** Situado en los exteriores de La Laguna, junto a una parcela vacía, será el encargado de activar para el usuario otra de las misiones existentes. Para dicha misión el usuario tendrá que entregar diversos materiales y este personaje tendrá también la función de ser el punto de entrega de esos elementos.
- **Mercader:** Desarrollado como personaje extra para la misión que activa el constructor. El mercader actúa como punto de recogida opcional de materiales para la parte final de la misión del constructor. Está situado cerca del mercado exterior de La Laguna.

## 3.4 Misiones

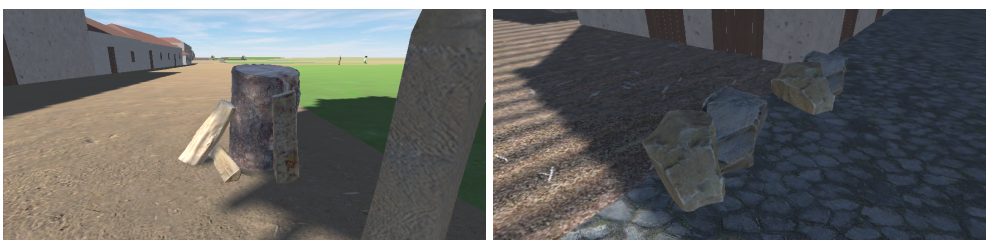
Este punto es, junto con los dos anteriores la tríada que conforma el comportamiento base de los videojuegos RPG. Las misiones están compuestas por un conjunto de objetivos que pueden ser de mayor o menor dificultad y que su finalización otorga al usuario algún tipo de recompensa y desbloqueo de contenido posterior que normalmente estaría bloqueado hasta la conclusión de dicha misión.

Para este proyecto y a modo de introducción para poder poner a prueba este funcionamiento y poder recrear a pequeña escala el funcionamiento de un videojuego real se ha elaborado una serie de misiones sencillas que también aportan un componente de exploración, de modo que el usuario tenga la necesidad de avanzar por el entorno para completarlas y así poder intensificar el componente histórico y de patrimonio del videojuego.

- **Misión Introdutoria:** La primera misión disponible y gracias a la cual se desbloqueará el resto de misiones, es la de conocer al personaje del Alcalde, la máxima autoridad de la ciudad y persona encargada de proporcionar al usuario las recompensas por cumplimentar las diferentes misiones que reciba. En el instante en que comienza el juego, el primer personaje que se encuentra el usuario, tras escuchar su diálogo lo envía a conocer al alcalde y una vez lo hayamos conocido y escuchado sus diálogos introductorios, activará uno de los eventos descritos en el punto anterior, al que estarán suscritos los demás NPC que otorgan misiones y activará la flag de conocimiento del alcalde, requisito necesario para poder comenzar a realizar las misiones de estos otros personajes y que sus diálogos e interacciones se activen.
- **Misión del Constructor:** La misión más compleja de las actuales incorporadas al videojuego, es necesario haber completado la misión introductorio para desbloquearla y se inicia tras los diálogos introductorios con el personaje del constructor. Se trata de una misión que activa varios elementos en el mapa (Figura 12), pues es necesario que el usuario avance por el entorno en busca de materiales (Figura 13) necesarios para la construcción de una serie de viviendas típicas de la época. Estos elementos que en principio están desactivados, sólo se mostrarán visibles mientras la misión esté activa y desaparecerán en cuanto sean recogidos. Para no reutilizar en exceso el bucle de dicha misión, ya que se trata de la construcción de tres viviendas que cada vez necesitan de la obtención de más materiales, en última instancia se da la posibilidad al usuario tanto de ir a buscar los materiales como hasta el momento o de hablar con otro personaje a modo de Mercader, que proporcionará instantáneamente los materiales necesarios para la construcción de la última vivienda. De esta manera se da al usuario la posibilidad de completar una misión de más de una forma consiguiendo así un mayor número de interacciones y un bucle de juego menos repetitivo. Cabe destacar que aunque sea un personaje secundario, la forma de interacción con el mercader se basa en los mismos aspectos comentados con anterioridad para todos los personajes NPC principales. Una vez se haya completado la misión se pedirá al usuario que interactúe de nuevo con el alcalde para conseguir la recompensa.



*Figura 12.- Construcciones que forman parte de la misión.*



*Figura 13.- Elementos recolectables.*

- **Misión del cura:** En esta misión se presenta un nuevo tipo de personaje secundario. Al igual que la misión anterior, es necesario haber completado la misión introductoria para poder acceder a ésta y consiste en buscar por todo el entorno a varias monjas ayudantes (Figura 14) del alcalde para interactuar con ellas. Estos personajes actúan de forma diferente a los demás NPC, siendo los únicos que no tienen secuencia de diálogo sino que su única interacción disponible es durante el desarrollo la misión e incrementan el número de ayudantes encontrados, permitiendo que se complete la misión en el momento en que el usuario localiza a todos los ayudantes. A su vez, estos personajes presentan un trigger que inicialmente está desactivado y se activa en el momento en que se cumplen las condiciones necesarias para comenzar el desarrollo de la misión, una vez el personaje principal entre en el espacio del trigger, este activará el evento correspondiente e incrementará el número de ayudantes encontrados, desactivando de nuevo el trigger para no poder utilizar el mismo ayudante repetidas veces. Esto se consigue mediante el uso de una lista de GameObjects que se rellena en el momento en que se inicia la misión y se va vaciando a medida que se localizan a los diferentes personajes u objetos que la componen. Al finalizar la misión estará disponible una nueva interacción con el alcalde para conseguir una recompensa.



*Figura 14.- NPC Secundario.*

- **Misión de la campesina:** Al igual que las demás misiones presentes en el proyecto, es necesario haber completado la misión introductoria para desbloquear su acceso. Consiste en una misión de búsqueda en la que el usuario debe conseguir diferentes objetos esparcidos por el entorno que en un inicio están desactivados y en el momento de inicio de la misión se activan haciendo uso al igual que en el caso anterior de una lista de GameObjects. A medida que se van encontrando los objetos, el NPC que inicia la misión tendrá disponibles diferentes líneas de diálogo que se podrán conocer si el usuario vuelve a interactuar con el personaje después de recoger cada uno de los objetos. En caso contrario, si se vuelve a interactuar con el personaje cuando se hayan conseguido todos los objetos y por tanto se haya completado la misión, estos diálogos intermedios no podrán ser reproducidos, añadiendo al proyecto otra de las características clásica de los RPG, donde dependiendo de las acciones que lleve a cabo de usuario podrá ser testigo o no de algunas interacciones. Por último, tras completar la misión se abrirá una nueva interacción con el alcalde para conseguir una nueva recompensa.

## 3.5 Sistema de Reputación.

Es esencial establecer un sistema de progreso en todos los títulos de este tipo, para generar en el usuario la necesidad o el objetivo de progresar para conocer todo lo que el videojuego puede aportar. En este caso, para simular este sistema de progreso al ser un proyecto en pequeña escala se ha optado por la implementación de un sistema de reputación con los habitantes del entorno que se incrementa siempre que se complete una misión.

De este modo, en el estado inicial del juego, no será posible para el usuario interactuar con ninguno de los personajes y en el momento que se conozca al alcalde (Figura 15) se consigue una reputación inicial que permite al usuario comenzar con el desarrollo de las diferentes misiones disponibles en el juego.

En el momento en que se completa cualquiera de las misiones disponibles y se interactúa de nuevo con el alcalde para recibir la recompensa, se incrementa en una unidad este nivel de reputación con los ciudadanos. De esta manera se desbloquean, si están disponibles, nuevas interacciones con los personajes que sin dicho nivel de reputación no serían posibles.

Este método de progreso se implementa de forma bastante sencilla utilizando las herramientas de scripting ya que simplemente actúa como una flag que se activa al cumplir las condiciones designadas y permite la ejecución de eventos e interacciones que sin estar activa no se podrían ejecutar.



*Figura 15.- Alcalde y vista del Ayuntamiento.*

## 3.6 Interacción con el entorno.

Tal y como se ha comentado anteriormente, además de la interacción con los diferentes personajes, otro punto importante dentro de los videojuegos de tipo RPG es la interacción con el entorno y con los diferentes objetos que están presentes en él, ya sea recogéndolos u obteniendo información acerca de ellos.

En el caso de este proyecto, el tipo de interacción con los objetos del entorno que se ha establecido es de tipo "pick up". Siempre precedido de su activación en una misión determinada. Los objetos del entorno con los que se pueda interactuar tendrán asociado a ellos un trigger y el evento correspondiente que se activará cuando el usuario entre en su espacio. Esta forma de interacción es similar a la presente en los NPC, con la diferencia de que en los objetos se trata de una interacción de un solo uso (Figura 16).

Son objetos estáticos, que no poseen collider físico puesto que no es necesario realizar una colisión física con ellos, sino que contienen un trigger esférico en el que al usuario penetrar su espacio automáticamente el objeto se recogerá, se desactivará y dependiendo de la misión que se esté realizando aportará un valor concreto en el controller correspondiente.



Figura 16.- Triggers para los objetos recolectables.

## 3.7 Extras.

Además de los puntos comentados anteriormente, que representan las características base que conforman un videojuego RPG, se han implementado una serie de extras, que aunque no forman parte de todos los videojuegos de este tipo si son características importantes de los videojuegos en general, por lo que cabe destacar su implementación.

### 3.7.1 Minimapa.

El minimapa constituye una de las fuentes de navegación más importantes y utilizadas a lo largo de la historia de los videojuegos, es una herramienta simple pero muy útil para el usuario que le permite conocer con antelación los alrededores del entorno en el que se encuentra pudiendo resultar de gran ayuda para diversas situaciones.

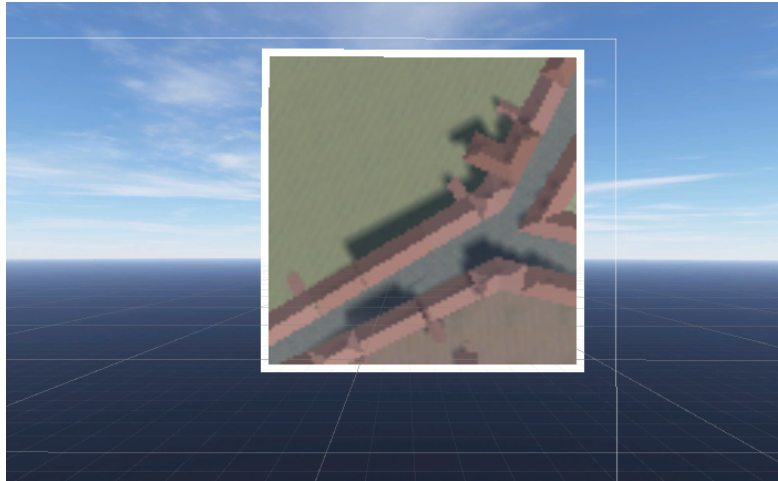
Para llevar a cabo la implementación de un minimapa para este proyecto se utilizó una cámara con proyección ortogonal [19], de modo que estuviese suspendida en el aire capturando el entorno de manera perpendicular a él y ofreciera una proyección cúbica del mismo, con la que se presenta de manera precisa cada parte que lo compone, eliminando así la distorsión que aporta la vista perspectiva al acercar un objeto a la cámara.

Una vez configurada correctamente la cámara que contendrá el minimapa, el siguiente paso sería superponerla a la cámara principal que enfoca al protagonista y será la encargada de llevar el peso de la ejecución del videojuego. Para ello, se ha hecho uso de



la herramienta canvas [20] de Unity para crear la interfaz que se superponga a la cámara principal. Un canvas es un gameobject con componentes canvas en él y constituye el área donde deben estar todos los elementos de UI (User Interface) (Figura 17).

En este canvas, es donde irá colocado el resultado que proyecta la cámara ortogonal, ya que la cámara como tal no puede ser añadida directamente y es necesario un paso intermedio en el que la imagen proyectada por la cámara se almacene en una “render texture” [21].



*Figura 17.- Vista del minimapa desarrollado.*

Un render texture, es un tipo de textura que crea Unity y que mantiene actualizada en tiempo de ejecución. Para su uso en este proyecto se ha modificado su tamaño por defecto para ajustarlo a las dimensiones estándar de un minimapa y se ha eliminado la propiedad de buffer de profundidad al no ser necesaria para este tipo de proyección. Una vez configurada esta textura es necesario asignarla como “Target Texture” en los componentes de la cámara ortogonal que se ha creado, de modo que el resultado de la proyección sea almacenado en dicha textura, que finalmente, puede ser añadida como elemento “Raw Image” al canvas creado con anterioridad y ajustada su posición a la deseada para el minimapa.

Por último, para su correcto funcionamiento es necesario la creación de un script en C# que será añadido como componente a la cámara ortogonal, donde se actualice la posición de la cámara en función de la posición en donde se encuentre el personaje principal, de modo que imite su movimiento y mantenga siempre actualizada de forma correcta la información mostrada en el minimapa de forma relativa a la posición en la que se encuentra el usuario.

### **3.7.2 Menú.**

Se ha llevado a cabo la implementación de un menú de inicio, haciendo uso de una segunda escena, en la que el usuario podrá iniciar el juego, acceder a un menú de opciones donde ajustar el volumen del videojuego o cerrar la aplicación.

Para ello, al igual que en el caso anterior durante la implementación del minimapa, se va a hacer uso de un canvas que contendrá todos los componentes de dicho menú que formarán parte de la UI de la escena.

Es importante mencionar que todos los elementos que formen parte del canvas se mostrarán en orden jerárquico inverso, es decir, aquellos elementos que estén posicionados más arriba en la lista de jerarquía, tendrán prioridad y serán los primeros en ser dibujados en la UI, mientras que a medida que se baja en la lista de jerarquía los nuevos elementos se irán superponiendo a los ya añadidos. Esta es una característica a tener en cuenta para realizar una correcta organización de los diferentes elementos que conformen el menú que se ha implementado.

En primer lugar se ha añadido la imagen de background, que actuará como fondo para la escena del menú. En el caso concreto de este proyecto se ha realizado una ilustración a medida correspondiente a una posible representación de la ciudad de San Cristóbal de La Laguna en la época en la que se desarrolla el videojuego. Este trabajo de diseño ha sido llevado a cabo por Noelia Negrín Casanova, alumna del Grado de Historia del Arte en la Universidad de La Laguna (Figura 18).



*Figura 18.- Vista del menú principal.*

El siguiente elemento en la jerarquía corresponde al menú principal, el cual está compuesto por tres botones con texto y funcionalidad personalizados, mediante los cuales se podrá acceder al juego, cargar el menú de opciones o cerrar la aplicación.

Para estos botones se ha creado una base de color de cuatro gradientes en composición con la ilustración utilizada para el background, apoyándose en el uso del asset TextMeshPro [22] presente en Unity. Un asset ideal para este tipo de tareas debido a las técnicas de renderizado de texto avanzadas que utiliza que junto a un conjunto de shaders personalizados permiten conseguir una mejora sustancial en la calidad visual de los textos a la vez que otorga flexibilidad en lo referente a estilo y texturizado del mismo.

La funcionalidad de estos botones viene generada mediante scripting, donde se actualiza la escena que se presenta al usuario o se procede a cerrar la aplicación, pero también mediante las propias herramientas de Unity que aporta por defecto para los botones de UI, la cual permite en tiempo de ejecución desactivar el menú de inicio y activar el menú de opciones al hacer click en el botón correspondiente, consiguiendo de esta manera alternar entre diferentes menús y su funcionalidad sin necesidad de crear el proceso de scripting necesario para realizar dichas acciones.

El último elemento de la jerarquía, corresponde al menú de opciones (Figura 19) y presenta un botón al igual que el menú principal, pero añade el uso de un slider para llevar a cabo las tareas de ajuste de volumen del videojuego. Este slider, comprenderá valores entre 0 y 1, para representar el nivel de volumen seleccionado y el valor elegido por el usuario se almacenará mediante un script y quedará guardado para ser establecido por defecto al iniciar la aplicación en ocasiones posteriores. Este comportamiento se

consigue gracias al uso de la clase “PlayerPrefs” que almacena entre sesiones de juego las diferentes preferencias del usuario en registros que pueden ser de tipo string, float o integer.



Figura 19.- Vista del menú de opciones.

### 3.7.3 Sonido.

Para llevar a cabo la reproducción tanto de la música de fondo como los diferentes diálogos que presentarán los NPC, se han utilizado diversos componentes “AudioSources” [23] en los diferentes controllers que presenta cada NPC (Figura 20). Esto presenta la ventaja de que varios sonidos se pueden reproducir al mismo tiempo al estar utilizando fuentes diferentes, dando lugar a que no exista una interrupción en la música ambiental mientras se esté ejecutando un diálogo de un personaje.

Como formato de audio se ha utilizado “.ogg”, en vez del estándar “.mp3” debido a que presenta una serie de características que lo sitúan por encima en términos de calidad respecto al estándar. Permite almacenar 255 canales de audio mientras que el estándar solo permite dos, por lo que se consigue un sonido de mayor calidad, que el que se puede conseguir con el estándar. Por otra parte, el formato de compresión Vorbis permite obtener calidades de audio superiores en un tamaño de archivo menor. También cabe mencionar que tanto el contenedor .ogg como el códec Vorbis son completamente abiertos, libres y sin patentes por lo que la comunidad puede encargarse de seguir mejorándolo en un futuro ininterrumpidamente.

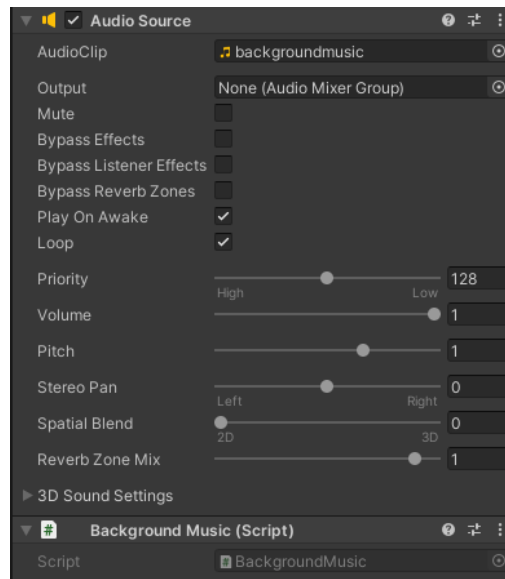


Figura 20.- Vista de un AudioSource utilizado en el proyecto.

## 3.8 Sitio Web.

Un proyecto profesional de este tipo debe contemplar además de las tareas de desarrollo, los aspectos relacionados con el marketing, la financiación y la comercialización. Siendo el objetivo final de los proyectos de desarrollo de videojuegos hacerse con un hueco en un mercado muy exigente donde una inmensa cantidad de productos no consiguen viabilidad y terminan siendo cancelados.

En este marco, resulta interesante para este proyecto llevar a cabo además del desarrollo del videojuego RPG, la implementación de un sitio web que simule y contenga la información necesaria para el potencial usuario final de un proyecto de este tipo.

Se ha llevado a cabo un proceso de desarrollo, tomando como base sitios webs profesionales de proyectos del sector, para recrear en la medida de lo posible un sitio web propio que pueda ofrecer a un potencial usuario o cliente, información atractiva y de utilidad que pueda impulsar el proyecto, por lo que se ha considerado esencial una estructura del siguiente tipo:

- **Portada:** La página principal que todo proyecto de este tipo debe poseer y que contendrá los principales y más importantes elementos que puedan captar la atención de los usuarios, como puede ser una pequeña descripción de los contenidos que se podrán encontrar en el producto, capturas de pantalla o vídeos demostrativos y los enlaces a las diferentes secciones de interés, que contendrán toda la información disponible de manera mucho más detallada. Para esta pestaña se ha utilizado la ilustración utilizada en el menú del videojuego como imagen de fondo, para aportar mayor identidad al proyecto (Figura 21).

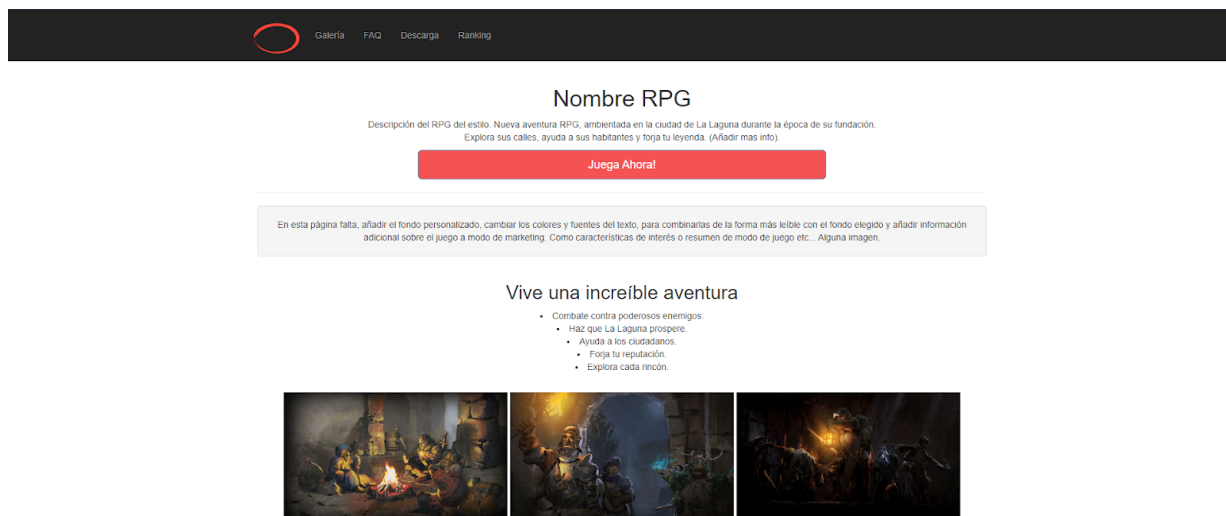


Figura 21.- Portada del prototipo de sitio web.

- **Galería:** Otro de los apartados más importantes a la hora de promocionar un proyecto de este estilo, es ofrecer al usuario un entorno en el que pueda comprobar de primera mano, la estética, dinámica o funcionamiento del producto. Para ello se ha implementado una galería de imágenes haciendo uso de lightbox, que contendrá todas aquellas capturas de pantalla que muestren las características del videojuego.
- **FAQ:** Una de las características que debe presentar este tipo de sitios web, es un apartado en el que el usuario pueda conocer la respuesta a la dudas principales que pueda llevar consigo un proyecto de este tipo, como pueden ser, los requisitos mínimos y recomendados para su correcto funcionamiento, el sistema de pago del producto, ya sea suscripción, pago único o modelo free-to-play o cualquier otro dato de interés sobre el videojuego como controles, configuración, tipo de videojuego, etc.
- **Descarga:** Siendo que muchos proyectos independientes, en sus fases preliminares aún no están alojados en plataformas de distribución o poseen directamente su propio modelo de distribución, es común que estos sitios web, contengan un enlace de descarga del proyecto, para que normalmente, tras realizar la compra del producto o el registro en la web, los usuarios puedan descargar el videojuego directamente sin intermediarios y comenzar a jugarlo desde un primer momento.

Toda la implementación de este sitio web, además de contener la información previamente comentada, ha sido realizada responsivamente de modo que pueda ser correctamente visualizada en cualquier dispositivo (Figura 22).

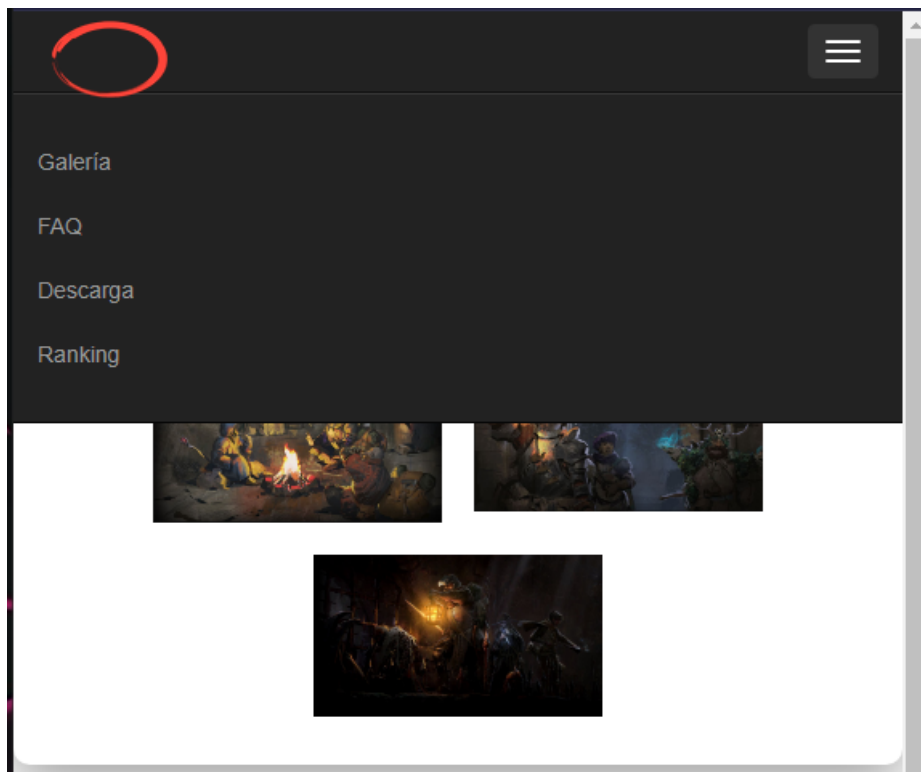


Figura 22.- Diseño responsivo del sitio web.

### 3.9 Pruebas y Rendimiento.

Para llevar a cabo las diferentes pruebas de rendimiento del proyecto se han utilizado diversos ordenadores de diferentes prestaciones, con el objetivo de poder establecer cuáles serían las características mínimas o recomendables para el correcto funcionamiento del proyecto.

Tipo de Ordenador	Media de FPS Logrados
High end	220
Medium End	135
Low End	50

Tabla 3.9: Resultados y Rendimiento.

Se han conseguido resultados satisfactorios en cuanto al rendimiento del proyecto, consiguiendo en ordenadores de altas especificaciones picos de más de trescientos frames por segundo y una media de doscientos veinte, durante una ejecución completa del videojuego. Además para ordenadores de bajas prestaciones se han conseguido medias que rozan el estándar de sesenta frames por segundo, consiguiendo de esta forma, que el videojuego desarrollado, pueda ser reproducido en un amplio rango de ordenadores y sea accesible para cualquier usuario potencial.

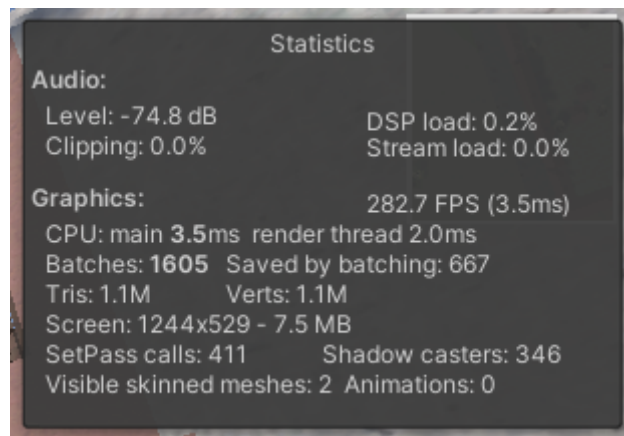


Figura 23.- Cuadro de rendimiento del proyecto.

Estas pruebas de rendimiento se han realizado una vez finalizado el proyecto y se han conseguido lograr los objetivos marcados en el inicio, desarrollando una versión funcional del videojuego propuesto.

Se ha llevado a cabo la correcta implementación de las funcionalidades básicas y características de un videojuego RPG, tales como:

El movimiento libre, que aunque limitado por la malla de navegación, permite al usuario moverse con total libertad por las calles del entorno y visitar cada rincón del mismo. Usando además el movimiento físico adaptándolo a este modelo de entorno sin colisiones.

La interacción entre el usuario y los NPC presentes en el videojuego, ya sea mediante la reproducción de diálogos o la asignación y realización de las diferentes misiones implementadas. Al igual que la interacción con los diferentes objetos que forman parte de las misiones como pueden ser los materiales necesarios para llevar a cabo las construcciones de las viviendas.

Tras llevar a cabo, el trabajo de desarrollo e implementación del proyecto, se ha conseguido una versión final, que aunque es ampliable en contenido en numerosos aspectos, presenta una duración de aproximadamente treinta minutos para completarlo, siempre dependiendo de la rapidez con la que el usuario complete las diferentes tareas y el tiempo extra que pueda añadir el visitar cada rincón del entorno (Figuras 24 y 25).



*Figura 24.- Captura de ejecución del videojuego, resultados finales.*



*Figura 25.- Captura de ejecución del videojuego, resultados finales.*

Todos los contenidos del proyecto, como capturas de pantalla o videos de presentación, además de una versión demo final empaquetada y preparada para su ejecución pueden ser encontrados haciendo uso del repositorio de Github asociado al proyecto [24].



# Capítulo 4 Comercialización.

## 4.1 Financiación.

Tras haber llevado a cabo el desarrollo de un proyecto y haber llegado a su finalización o en el caso de haber conseguido realizar una versión estable en la que en un futuro se siga trabajando pero se necesite financiación para poder continuar con el desarrollo, es necesario conocer algunas de las diferentes aproximaciones al mercado existentes, tanto si el proyecto que hemos desarrollado es para plataformas móviles o para ordenador. En este apartado se van a tratar las principales plataformas de crowdfunding y los pasos necesarios para llevar a cabo la publicación de un proyecto en las principales distribuidoras.

Si el proyecto se encuentra en una fase de desarrollo temprana, a mitad de desarrollo o incluso si no se ha comenzado su desarrollo, las plataformas de crowdfunding constituyen una de las principales opciones de financiación para proyectos independientes o indies.

Existen muchas plataformas de crowdfunding diferentes, como por ejemplo, Kickstarter [25] o Indiegogo [26], que son dos de las plataformas más conocidas (Figura 23). Antes de elegir una de ellas, es importante investigar cuál es la más adecuada para el proyecto.

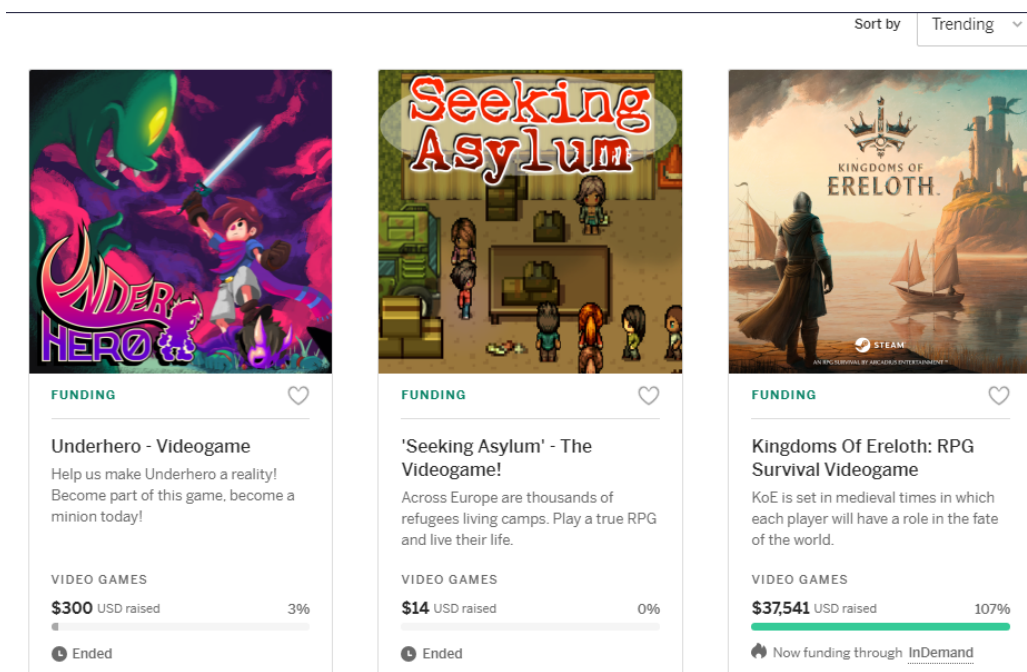


Figura 26.- Fragmento de la página principal de IndieGogo.

Una vez se haya elegido una plataforma, se debe crear una campaña para recaudar fondos para el proyecto. Esta campaña tiene que incluir información sobre el proyecto, una descripción de los objetivos del proyecto y una lista de recompensas para los patrocinadores.

Para que la campaña tenga más posibilidades de ser exitosa, es importante promocionarla tanto como sea posible. Esto puede incluir la creación de contenido promocional para las redes sociales, la participación en grupos de discusión relevantes y la colaboración con otros desarrolladores.

Durante la campaña, es importante mantener a los patrocinadores informados y responder a cualquier pregunta o inquietud que puedan tener. También es necesario asegurarse de cumplir con las fechas de entrega de las recompensas prometidas.

Si la campaña es exitosa, se deben recoger los fondos y utilizarlos para desarrollar el juego. Si no se alcanza el objetivo de financiación fijado y por tanto la campaña no sea exitosa, es posible que se deba buscar otras opciones de financiamiento o cancelar el proyecto.

Aunque se puede profundizar más al respecto, estos son los aspectos básicos que hay que tener en cuenta a la hora de llevar a cabo una campaña de crowdfunding. Siendo Kickstarter e Indiegogo dos de las plataformas más conocidas, se va a tratar el proceso necesario para crear y llevar a cabo una campaña en una de estas dos plataformas, las cuales funcionan de una forma muy similar.

Lo primero que se debe hacer es crear una cuenta en una de estas plataformas. Para hacerlo, debemos ir al sitio web de la plataforma elegida y hacer clic en el botón "Registrarse" e introducir todos los datos necesarios para la creación de una cuenta de usuario en la plataforma que nos permita el siguiente paso, la creación de un proyecto.

Una vez que se haya creado la cuenta e iniciado sesión, es necesario hacer clic en el botón "Crear un proyecto" en la página de inicio para luego poder seguir las instrucciones mostradas en pantalla para poder crear una página de proyecto.

La página de proyecto debe incluir una descripción detallada de la idea, imágenes y vídeos que muestren el proyecto, información sobre el equipo de desarrollo, antecedentes del mismo y una lista de las recompensas que serán ofrecidas a los patrocinadores.

También es necesario establecer una meta de financiamiento para nuestro proyecto. Esta es la cantidad total de dinero que es necesaria recaudar para poder dar como exitosa la campaña y comenzar o continuar el desarrollo del proyecto. Es importante establecer una meta realista y factible para el proyecto que se está realizando.

Una vez introducida toda la información necesaria sobre el proyecto, se debe establecer una duración para la campaña, que como máximo será de 60 días y es el tiempo límite para lograr recaudar la cantidad de dinero establecida en la meta de financiamiento.

En lo que corresponde a las recompensas para patrocinadores, se debe establecer una serie de ellas dependiendo la cantidad de dinero con la que contribuye el patrocinador. Por ejemplo, se puede comenzar con pequeñas recompensas simbólicas e ir subiendo la calidad a medida que la aportación es más sustanciosa. Cabe destacar que las recompensas, aunque pueden serlo, no tienen porqué ser artículos materiales, sino también por ejemplo, pueden ser artículos dentro del juego como cosméticos o acceso anticipado al mismo.

Una vez que se haya terminado de crear la página del proyecto, esta se debe enviar para su revisión por parte del equipo de la plataforma, que serán los encargados de verificar que toda la información incluida en el proyecto es correcta y válida. Si el proyecto es aprobado por el equipo de la plataforma, puede ser publicado y se puede comenzar a promocionar para conseguir la financiación necesaria.

Este es un punto de gran importancia, debido a que un proyecto que cuente con una promoción a la altura tiene muchas más posibilidades de llegar a la meta de financiación y poder comenzar su desarrollo. Es recomendable llevar a cabo esta promoción en todas

las redes sociales y medios posibles. Además de involucrar a la mayor red de contactos posibles que difundan y generen un mayor impacto y alcance.

Si la campaña es un éxito y se alcanza la meta de financiación, se recibirán los fondos recaudados, menos una tarifa que aunque varía, oscila en el 5% de la cantidad total que se guarda la plataforma a modo de comisión.

Una vez se hayan recibido los fondos, se podrá comenzar con el desarrollo del proyecto y trabajar en las recompensas para los patrocinadores.

## **4.2 Distribución mediante Play Store.**

Tras realizar el desarrollo de un proyecto y haberlo finalizado o al menos, haber llegado a un punto del desarrollo donde exista una versión estable y comercial del mismo, es momento de llevar a cabo su distribución y comercialización y que el proyecto pase a ser un producto que genere ingresos.

Para ello, si el proyecto se trata de una versión de aplicación para dispositivos móviles, se explicará los diferentes pasos a seguir para poder llevar a cabo la publicación del proyecto en la plataforma Play Store de Google [27], que conforma la mayor distribuidora del mercado.

Lo primero que es necesario llevar a cabo para poder publicar el proyecto desarrollado en la Play Store es crear una cuenta de desarrollador. Para ello, se deben seguir los pasos indicados en la documentación oficial de Google.

En primer lugar se debe acceder al sitio web de Google Play Console e iniciar sesión con una cuenta de Google válida. Posteriormente, se presentarán los términos y condiciones del servicio que han de ser leídos detenidamente y en caso de estar de acuerdo, marcar la casilla correspondiente para aceptarlos y poder seguir adelante con el proceso.

Una vez aceptados los términos y condiciones, el siguiente paso es completar el perfil de desarrollador, proporcionando información sobre el usuario o sobre la empresa a la que se representa si corresponde. Algunos de los datos solicitados son el nombre, la dirección y otros datos solicitados. Cabe destacar que algunos de estos son obligatorios para poder avanzar.

Una vez rellenados los datos solicitados, el siguiente paso poder crear la cuenta de desarrollador es abonar la tarifa de registro que actualmente es de 25\$USD, pero puede variar en el futuro. Este es un pago único de tarifa de registro necesario para poder publicar aplicaciones en la Play Store.

Después de realizar el pago, es posible que se solicite una verificación de identidad. Lo cual puede implicar el envío de ciertos documentos o información adicional, dependiendo de la ubicación y la forma en que se esté registrando la cuenta de desarrollador.

Seguidamente, para poder vender aplicaciones o realizar compras en la aplicación, se deberá configurar una cuenta de pago. Es de vital importancia en este caso, proporcionar los detalles necesarios para recibir pagos, como la información bancaria, y asegurarse de que esté correctamente configurada.

Una vez que se hayan completado todos los pasos anteriores, se tendrá acceso completo a la consola de Google Play, desde donde se podrá administrar las aplicaciones, realizar actualizaciones, definir precios y realizar un seguimiento del rendimiento de los proyectos publicados en la Play Store.

Cabe destacar que el proceso de creación de una cuenta de desarrollador de Google puede variar en el tiempo y los requisitos específicos pueden cambiar. Es necesario asegurarse de leer y seguir las instrucciones proporcionadas en el sitio web oficial de Google Play Console para obtener la información más actualizada y precisa sobre el proceso de registro.

Una vez creada la cuenta de desarrollador, antes de publicar un juego, es importante que se revisen las políticas de la Play Store para estar seguros de que el proyecto cumple con todos los requisitos. Si el proyecto viola alguna de estas políticas, es probable que sea rechazado.

La Play Store de Google tiene políticas muy estrictas que los desarrolladores deben seguir para publicar su juego en la plataforma. Como por ejemplo, Google prohíbe el contenido que sea considerado como inapropiado o ilegal, como puede ser la pornografía, la violencia extrema o el acoso. Por otro lado, los desarrolladores no pueden publicar información engañosa o falsa sobre su proyecto en la Play Store. Además, también está prohibido el uso de Spam o Malware. Se debe ser transparente sobre las compras dentro del juego. Se deben cumplir con todas las leyes aplicables a su país y en la Play Store y se deben de respetar los derechos de autor y no se puede publicar contenido que infrinja estos derechos. Estas son algunas de las políticas de la Play Store, pero existen muchas otras que deben ser leídas y conocidas completamente antes de intentar publicar cualquier tipo de proyecto o contenido en la plataforma.

Una vez que se ha creado y aceptado la cuenta de desarrollador y se ha verificado que el proyecto cumple con todas las políticas de la Play Store, el siguiente paso en la publicación de un proyecto es la creación de una ficha de producto para el mismo. Esta ficha incluye información como el nombre del proyecto, una descripción del mismo, capturas de pantalla o un vídeo promocional. En esta ficha de producto también se puede configurar el precio del mismo y las opciones de monetización.

Aunque existen varias opciones de monetización diferentes dentro de la Play Store, las más comunes incluyen la compra única del producto, donde el usuario compra el producto una única vez y puede acceder a todo su contenido.

Las opciones de compra dentro del producto, donde el usuario puede pagar un precio simbólico por la compra del producto y además dentro del mismo puede comprar elementos virtuales como cosméticos o elementos especiales.

Otra de las opciones de monetización disponibles, es la publicidad, donde los desarrolladores incluyen anuncios dentro del proyecto para generar ingresos.

Por otra parte, existen también las opciones de suscripción donde los usuarios pueden suscribirse al producto para obtener acceso a contenido exclusivo, o la opción "Freemium" donde el producto se puede descargar gratuitamente, pero los usuarios pueden comprar elementos virtuales o acceso a contenido exclusivo dentro del juego.

Aunque estas son las opciones de monetización más comunes, hay muchas otras disponibles que pueden ser adecuadas para un proyecto en específico.

El último paso tras elegir la opción de monetización que mejor se ajuste al proyecto, es la propia publicación en la Play Store. Hay que tener en cuenta que este proceso puede llevar tiempo, ya que es necesario que sea aprobado por el equipo de la Play Store antes de que el proyecto esté disponible para los usuarios.

### 4.3 Distribución mediante Steam.

Antes de poder publicar un juego en Steam, es necesario asegurarse de cumplir con algunos requisitos. Estos requisitos son, tener una cuenta de desarrollador de Steam activa, tener un juego completamente funcional que cumpla con las pautas de contenido de Steam [28], tener un acuerdo de distribución de Steam activo y aceptado y tener un documento de plan de negocios donde se detalle la planificación de la comercialización y la venta del videojuego (Figura 24).

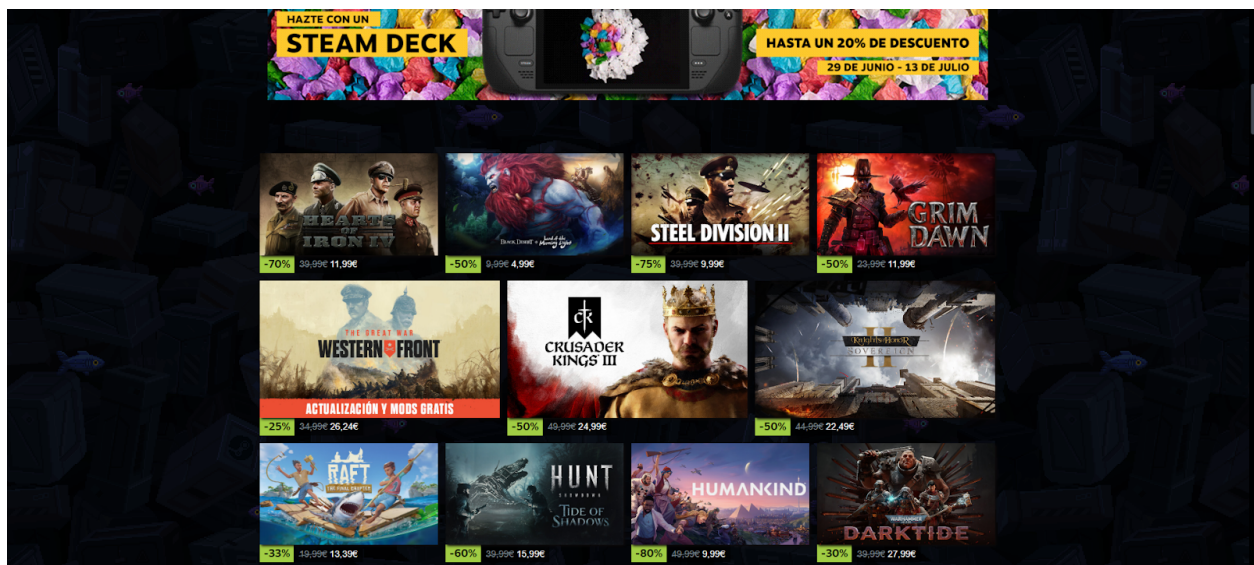


Figura 27.- Página principal de la Steam Store.

Una vez se hayan cumplido los requisitos previos, se puede comenzar el proceso de publicación de un videojuego en Steam.

La primera tarea que se debe realizar es la de crear una página para el proyecto de videojuego en Steamworks. Esto permitirá que los usuarios de Steam vean la información sobre el videojuego, como la descripción del mismo, las capturas de pantalla y el tráiler o videos promocionales.

Para crear la página del proyecto, se debe iniciar sesión en Steamworks, hacer clic en "Mi contenido" y luego en "Agregar un nuevo juego". Se deberá ingresar información detallada sobre el videojuego, incluyendo el título, la descripción, el género, las capturas de pantalla y el tráiler.

Después de crear la página del videojuego, se deberá configurar los precios y la disponibilidad en las diferentes regiones del mundo. Steam ofrece diferentes precios para diferentes regiones, y se puede establecer un precio propio para cada una de ellas.

Aunque existen diversas formas de monetización para un producto distribuido en Steam, las más comunes son, la venta directa, donde los usuarios pagan por el producto y luego lo descargan desde su biblioteca de Steam. Las ventas en oferta, son períodos donde se puede ofrecer un producto con descuento en diferentes épocas del año como verano o invierno. Por otra parte, se puede monetizar en base al uso de microtransacciones, donde si el producto ofrece contenido adicional, como elementos cosméticos, estos pueden ser comprados a posteriori por el usuario mediante microtransacciones.

Es posible también, ofrecer contenido adicional descargable mediante pago a modo de expansiones de contenido para un producto ya publicado o permitir la descarga del producto de manera gratuita pero ofrecer compras integradas dentro del producto para generar ingresos.

Aunque existen otras muchas formas de monetización que pueden ser más beneficiosas que otras dependiendo del producto en específico, éstas son las más usuales. Cabe destacar que Steam cobra un 30% de tarifa sobre todas las ventas realizadas en la plataforma.

Tras llevar a cabo la elección del método de monetización para el producto se deberá especificar si el videojuego ofrece contenido descargable o logros, que deberán ser configurados en Steamworks. Esto permitirá que los usuarios de Steam puedan ver qué contenido descargable está disponible y qué logros pueden desbloquear.

También es necesario, configurar las características del videojuego, como los controles del mismo, los requisitos del sistema, los idiomas admitidos y los ajustes de la interfaz de usuario.

El siguiente paso antes de la publicación de un producto en Steam es la realización de una serie de pruebas de calidad para asegurarse de que el proyecto esté completamente funcional y no tenga errores.

Por ejemplo se han de llevar a cabo pruebas de jugabilidad, donde se debe jugar a través del videojuego varias veces para estar totalmente seguros de que todo funciona correctamente y no hay problemas de equilibrio en el videojuego.

También se han de llevar a cabo pruebas de compatibilidad, donde se debe probar el videojuego en diferentes sistemas operativos y hardware para asegurarse de que funcione correctamente en todos los casos. Al igual que también se deben realizar pruebas de estabilidad donde el videojuego se ejecuta durante largos períodos de tiempo para asegurarse de que no haya problemas de estabilidad o errores frecuentes, como fugas de memoria.

Por otra parte, las pruebas de rendimiento se llevan a cabo ejecutando el producto en diferentes resoluciones y configuraciones de hardware para comprobar que funciona correctamente en dichas especificaciones. Por último se realizan pruebas de localización para asegurarse de que el videojuego está disponible en diferentes idiomas, que la traducción es correcta y que no existen problemas de diseño relacionados con el idioma.

Una vez que se hayan completado todos los pasos anteriores, se debe enviar el videojuego para revisión. Steam revisará el proyecto para asegurarse de que cumpla con sus pautas de contenido y funcione correctamente.

Si el proyecto pasa la revisión de Steam, podrá ser publicado. Una vez que esté disponible en Steam, se podrá promocionar y comercializar a través de la plataforma.

Además de la realización de este proceso hay que tener en cuenta diversas pautas, como la propiedad intelectual. Aspecto en el que el acuerdo de distribución de Steam establece que el desarrollador del videojuego mantiene todos los derechos de propiedad intelectual sobre el mismo. Sin embargo, Steam necesita ciertos derechos para poder publicar y vender el videojuego en su plataforma. Por lo tanto, el acuerdo otorga a Steam una licencia no exclusiva para utilizar, reproducir, distribuir, transmitir y mostrar el videojuego en su plataforma.

También es importante estar al tanto de las políticas acerca de los pagos y precios dónde, Steam cobra una tarifa del 30% sobre todas las ventas realizadas en su plataforma. Lo que significa que el desarrollador del videojuego recibirá el 70% de todas las ventas. Además, el acuerdo de distribución establece que el desarrollador del videojuego puede establecer su propio precio para el mismo y puede cambiar ese precio

en cualquier momento. Sin embargo, es importante tener en cuenta que Steam se reserva el derecho de ofrecer descuentos en el videojuego y el desarrollador no tiene control sobre cuándo o cuán cuantiosos serán esos descuentos.

Sobre el control de contenido y las actualizaciones al videojuego, el acuerdo de distribución establece que Steam se reserva el derecho de revisar y aprobar el contenido del videojuego antes de su publicación en la plataforma. Steam también puede retirar el producto de la plataforma si se determina que viola los términos del acuerdo de distribución.

Además, el acuerdo establece que el desarrollador del videojuego es responsable de proporcionar actualizaciones para el mismo y de asegurarse de que el videojuego siga siendo compatible con Steam y sus políticas de contenido.

En cuanto al soporte al cliente, el acuerdo de distribución establece que el desarrollador del juego es responsable de proporcionar soporte al cliente para el juego. Esto incluye responder a preguntas y solucionar problemas de los usuarios de Steam.

Y por último, acerca de la duración del acuerdo de distribución, Steam no aplica una duración específica y se puede rescindir en cualquier momento por cualquiera de las partes. Sin embargo, el acuerdo establece que los términos y condiciones seguirán siendo aplicables después de que el acuerdo sea rescindido en lo que respecta a las ventas realizadas en Steam antes de la rescisión.

Es importante leer cuidadosamente el acuerdo de distribución de Steam antes de aceptarlo, ya que establece los términos y condiciones para la publicación y venta del juego en la plataforma.

## **4.4 Distribución mediante Epic Games.**

Epic Games [29] no nace como competencia directa de la plataforma de Steam, sino como un estudio desarrollador de videojuegos y de Unreal Engine [30], que a día de hoy se ha convertido en uno de los motores de juegos más utilizados en la industria, tanto por desarrolladores independientes como por otros estudios de desarrollo de renombre.

Sin embargo, en los últimos años, Epic Games ha ampliado su alcance y ha lanzado la Epic Games Store, una plataforma de distribución digital de videojuegos. La Epic Games Store se ha posicionado como una alternativa a Steam, que es una de las plataformas de distribución digital más grandes y establecidas en la industria de los videojuegos.

La Epic Games Store (Figura 25) ha ganado atención y ha buscado competir con Steam al ofrecer condiciones más favorables para los desarrolladores, como un mayor porcentaje de regalías y acuerdos exclusivos con juegos populares. Epic Games también ha estado implementando una estrategia agresiva de adquisición de exclusivas temporales, asegurando que ciertos juegos solo están disponibles en su plataforma durante un período determinado antes de llegar a otras tiendas.

Esta competencia entre Epic Games Store y Steam ha llevado a una mayor diversificación y opciones para los desarrolladores y jugadores, y ha generado debates y discusiones en la industria de los videojuegos sobre los beneficios y desventajas de cada plataforma.

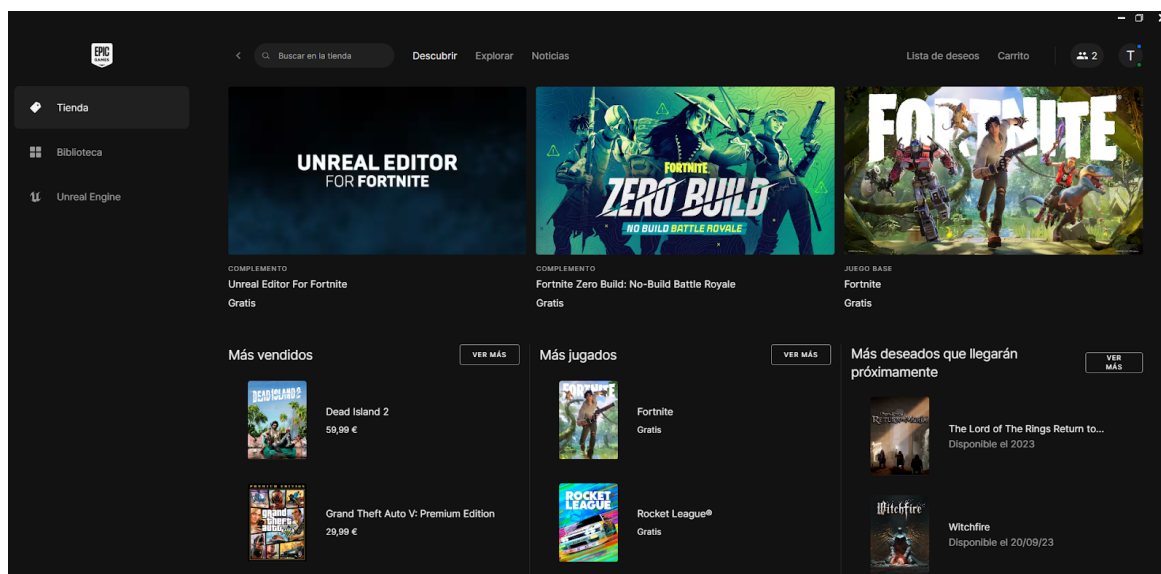


Figura 28.- Página principal de la Epic Games Store.

El proceso para publicar y distribuir un título propio en Epic Games, es similar al de la plataforma de Steam. Antes de comenzar este proceso de publicación es importante asegurarse de que el videojuego esté totalmente desarrollado y listo para su lanzamiento. Esto implica haber probado y depurado el juego para garantizar el cumplimiento de las pruebas de calidad y estabilidad exigidas por la plataforma. Además también es necesario disponer de todos los recursos del videojuego como arte, música o sonidos listos para ser incluidos en el paquete final.

El siguiente paso consiste en la creación de una cuenta de desarrollador en Epic Games. Para ello, es necesario visitar el sitio web de Epic Games y en la sección “Desarrolladores”, se encontrará la opción para registrarse. Será necesario cumplimentar un formulario de registro y proporcionar toda la información requerida, además de que existe la posibilidad de pagar una tarifa única para poder acceder al programa de desarrolladores.

A continuación, si como es el caso, el producto está desarrollado en Unity u otro motor gráfico que no sea Unreal Engine, será necesario exportar el proyecto en un formato compatible con Epic Games. Unity presenta varias opciones de exportación como para creación de un paquete de juego (game package) o la compilación del videojuego para una plataforma específica. Si por el contrario se usa Unreal Engine para el desarrollo del producto se deberá obtener una licencia para poder distribuirlo. Esto se puede realizar desde el sitio web de Unreal Engine, donde será posible encontrar información sobre las distintas opciones de licencia y sus respectivos precios.

Además también es necesario proporcionar los archivos y recursos del videojuego en un paquete adecuado y la creación de una descripción detallada del juego, capturas de pantalla y videos promocionales. Una vez se hayan completado dichos pasos de preparación, el producto deberá ser enviado a Epic Games para su revisión. Esto se realiza desde el portal de desarrolladores de Epic Games y se ha de asegurar de proporcionar toda la información requerida, incluyendo descripción del videojuego, capturas de pantalla, requisitos del sistema y cualquier otra documentación relevante y requerida.

Una vez se haya enviado el producto a Epic Games, el equipo de revisión lo evaluará para asegurarse de que cumpla con sus estándares de calidad y políticas de contenido. Dicha acción puede demorarse, dependiendo del volumen de videojuegos enviados al equipo y de su velocidad de respuesta. Durante este período es donde se realizan las



pruebas de calidad. Estas pruebas pueden variar en alcance y requisitos según las políticas internas de la plataforma aunque algunos de los aspectos esenciales que se tratan son:

- **Funcionalidad:** Una serie de pruebas en las que se verifica que el videojuego proporcionado funciona correctamente sin errores graves, fallos o bloqueos de rendimiento.
- **Rendimiento:** En estas pruebas se evalúa que el rendimiento del videojuego se ajuste a los requisitos del sistema especificados con anterioridad y se evalúa en términos de velocidad de fotogramas, fluidez, tiempos de carga, etc.
- **Contenido inapropiado:** Se verifica que el videojuego no contenga contenido ofensivo, ilegal o inapropiado según las pautas y políticas de la plataforma.
- **Compatibilidad:** El equipo de revisión de Epic Games, se asegura en este caso de que el producto entregado sea compatible con la plataforma de distribución y cumpla con los requisitos técnicos establecidos.

Una vez que el videojuego haya sido aprobado, se podrá comenzar a planificar su comercialización y lanzamiento. Esto implica establecer una estrategia de precios, promociones y marketing para dar a conocer el producto a la comunidad de jugadores. Además, se establecen los términos y condiciones de la venta del videojuego como los porcentajes de pago y los acuerdos de distribución con la plataforma.

El contrato de distribución es un acuerdo legal entre el desarrollador del proyecto y la plataforma de distribución, en este caso Epic Games y suelen incluir cláusulas relacionadas por ejemplo con los derechos de distribución donde se establecen los derechos que se otorgan a Epic Games para distribuir y vender el juego en su plataforma.

También se establecen las regalías y pagos y se acuerda la estructura de las mismas que el desarrollador recibirá por las ventas del juego. Al igual que los términos y condiciones de la distribución donde se incluyen aspectos como la exclusividad, la duración del contrato y las responsabilidades legales de ambas partes.

Por otra parte, este tipo de contratos también incluyen cláusulas adicionales para la propiedad intelectual, en las que se establece la propiedad de los recursos del videojuego y cómo se tratarán sus derechos. Se especifican también, las obligaciones del desarrollador en cuanto a proporcionar soporte y lanzar actualizaciones para el juego y por último también se discute la colaboración entre el desarrollador y Epic Games en términos de promoción y marketing del videojuego en la plataforma.

Es importante señalar que los detalles específicos del contrato de distribución y las pruebas de calidad que se realizan a la hora de la publicación de un producto en la Epic Games Store, pueden cambiar con el tiempo, por lo que es recomendable imprescindible consultar la documentación oficial de Epic Games y comunicarse directamente con su equipo de soporte para conocer la información más actualizada y precisa al respecto.

Una vez que todo el proceso esté completado, Epic Games se encargará de publicar el videojuego en su plataforma donde los usuarios podrán comprar y descargar el videojuego y será de nuevo Epic Games quien se encargará de gestionar las transacciones y pagos, proporcionando a los desarrolladores, informes y pagos de regalías de acuerdo con los términos acordados.

# Capítulo 5 Conclusiones y líneas futuras

El trabajo realizado ha cumplido con las expectativas propuestas en el momento de iniciar el proyecto, aunque el alcance del mismo en un futuro puede ser mucho mayor se ha conseguido llevar a cabo la implementación correcta y funcional de un videojuego base sobre el que montar los cimientos que, con suficiente trabajo y recursos pueda en algún momento, convertirse en un producto comercial de cara al extenso mercado de los videojuegos que existe hoy en día.

Por otra parte la implementación del sitio web para el proyecto y el proceso de investigación acerca de las diferentes opciones de financiación disponibles para proyectos independientes y las diversas opciones de distribución para los productos finales, abren un abanico de posibilidades, quizás no tan conocidas, pero de gran importancia para aquellos que busquen una oportunidad en un sector tan competitivo como lo es, el de los videojuegos.

Lejos de estar en un estado final, este proyecto presenta una amplia cantidad de aspectos en los que es posible mejorar y avanzar en un futuro. Ya sea por ejemplo el traslado del mismo a versiones posteriores de Unity en la que poder beneficiarse de las nuevas tecnologías incorporadas para conseguir un mayor rendimiento y calidad del mismo.

Así mismo, es interesante indagar en los diferentes modelos de interacción entre personajes para añadir dinamismo al proyecto y expandir el abanico de posibles tipos de misiones y retos para el usuario, así como la introducción del uso de bases de datos para poder llevar a cabo un registro de puntuaciones entre los usuarios dependiendo de la rapidez o habilidad que presenten para superar los desafíos propuestos en un futuro.

Aún con todos los posibles frentes abiertos para realizar mejoras al proyecto, se han conseguido alcanzar los objetivos de jugabilidad previstos y realizar una investigación completa sobre el acercamiento al mercado, lo cual, ha supuesto una experiencia tremendamente enriquecedora de cara a aumentar los conocimientos en el área de desarrollo de videojuegos, así como una ganancia de experiencia sustancial en las dinámicas de trabajo necesarias para llevar a cabo un proyecto de esta envergadura.

# Capítulo 6 Summary and Conclusions

The work done has fulfilled the expectations proposed at the time of starting the project, although the scope of the same in the future may be much greater, it has been possible to carry out the correct and functional implementation of a base video game on which to build the foundations that, with sufficient work and resources may at some point, become a commercial product for the extensive market of video games that exists today.

On the other hand, the implementation of the website for the project and the research process about the different financing options available for independent projects and the various distribution options for the final products, open a range of possibilities, perhaps not so well known, but of great importance for those seeking an opportunity in a sector as competitive as it is, that of video games.

Far from being in a final state, this project presents a large number of aspects in which it is possible to improve and advance in the future. Whether for example the transfer of the same later versions of Unity in which to benefit from the new technologies incorporated to achieve higher performance and quality of it.

It is also interesting to investigate the different models of interaction between characters to add dynamism to the project and expand the range of possible types of missions and challenges for the user, as well as the introduction of the use of databases to carry out a record of scores between users depending on the speed or skill they present to overcome the proposed challenges in the future.

Even with all the possible open fronts to make improvements to the project, we have managed to achieve the planned gameplay objectives and conduct a thorough investigation on the approach to the market, which has been a tremendously enriching experience in order to increase knowledge in the area of game development, as well as a substantial gain of experience in the dynamics of work necessary to carry out a project of this magnitude.

# Capítulo 7 Presupuesto

## 7.1 Tabla de Presupuesto.

<b>Tipos</b>	<b>Descripción</b>	<b>Total de Horas</b>	<b>Precio €/h</b>	<b>Precio Total (€)</b>
<b>Desarrollo</b>	Desarrollo de funcionalidad en el entorno de pruebas	110	20	2.200
<b>Implementación</b>	Implementación de la funcionalidad desarrollada en el entorno final	80	20	1.600
<b>Diseño</b>	Añadido de assets y elementos estéticos al entorno	30	20	600
<b>Corrección de Errores</b>	Corrección de errores en la implementación final	20	20	400
<b>Desarrollo Web</b>	Desarrollo y diseño del sitio web del proyecto	50	20	1.000
<b>Ilustración</b>	Diseño de la ilustración a medida para el proyecto	15	10	150
<b>Pruebas</b>	Período de pruebas de rendimiento	10	20	200
<b>Servidor Web</b>	Servidor en el que se lanzará el sitio web	720	0,24	175€/mes

**Tabla 7.1: Resumen de tipos**

El presupuesto necesario para la realización de este proyecto ascendería hasta los 6.325€ inicialmente y aumentaría con las tareas de mantenimiento y hosting de servidor.

# Bibliografía.

- [1] "Role-Playing video game" <https://www.britannica.com/topic/role-playing-video-game> (consultado el 14 de junio de 2023).
- [2] "¿Qué son los juegos RPG?" <https://www.winp.es/que-son-los-juegos-rpg/> (consultado el 14 de junio de 2023).
- [3] "Declaración de Patrimonio de la humanidad, de San Cristóbal de La Laguna" <https://www.aytolalaguna.es/ayuntamiento/el-municipio/declaracion-de-patrimonio-de-la-humanidad/#:~:text=El%20conjunto%20Hist%C3%B3rico%20de%20San,28%20de%20Diciembre%20de%201985.> (consultado el 14 de junio de 2023).
- [4] "Torriani, L. Alla Maesta Del Re Catolico, Descrittione et Historia Del Regno de l'isole Canarie Gia Dette Le Fortvnate Con Il Parere Delle Loro Fortificationi; Biblioteca Geral da Universidade de Coimbra: Coimbra, Portugal, 1590.)" [https://digitalis-dsp.uc.pt/bg6/UCBG-MS-314/UCBG-MS-314\\_item2/UCBG-MS-314\\_PDF/UCBG-MS-314\\_PDF\\_24-C-R0120/UCBG-MS-314\\_0000\\_Obra\\_Completa\\_t24-C-R0120.pdf#page=1](https://digitalis-dsp.uc.pt/bg6/UCBG-MS-314/UCBG-MS-314_item2/UCBG-MS-314_PDF/UCBG-MS-314_PDF_24-C-R0120/UCBG-MS-314_0000_Obra_Completa_t24-C-R0120.pdf#page=1)
- [5] "The incredible Evolution of NPC's In Video Games" <https://www.small-screen.co.uk/the-incredible-evolution-of-npcs-in-video-games/> (consultado el 15 de junio de 2023).
- [6] "Learn game development w/ Unity courses and tutorials in game design" <https://learn.unity.com/> (consultado el 16 de febrero de 2023).
- [7] "Physics - Unity Manual" <https://docs.unity3d.com/es/2018.4/Manual/PhysicsSection.html> (consultado el 16 de febrero de 2023).
- [8] "Lightning - Unity Manual" <https://docs.unity3d.com/Manual/LightingOverview.html> (consultado el 25 de Febrero de 2023).
- [9] "Occlusion Culling - Unity Manual" <https://docs.unity3d.com/Manual/OcclusionCulling.html> (consultado el 25 de Febrero de 2023).
- [10] "Blender Manual" <https://docs.blender.org/manual/es/latest/> (consultado el 18 de Junio de 2023)
- [11] "Patrimonio Virtual La Laguna" <http://torriani.iaas.uill.es/> (consultado el 16 de febrero de 2023).
- [12] "Rigidbody - Unity Manual" <https://docs.unity3d.com/es/2018.4/Manual/class-Rigidbody.html> (consultado el 15 de abril de 2023).
- [13] "Navigation - Unity Manual" <https://docs.unity3d.com/es/2019.4/Manual/Navigation.html> (consultado el 20 de abril de 2023).
- [14] "NavMesh Agent - Unity Manual" <https://docs.unity3d.com/es/530/Manual/class-NavMeshAgent.html> (consultado el 20 de abril de 2023).
- [15] "Triggers - Unity Manual"

<https://docs.unity3d.com/ScriptReference/Collider.OnTriggerEnter.html> (consultado el 15 de abril de 2023).

[16] “Delegates - Unity Learn” <https://learn.unity.com/tutorial/delegados> (consultado el 23 de abril de 2023).

[17] “Observer” <https://refactoring.guru/es/design-patterns/observer> (consultado el 2 de mayo de 2023).

[18] “UnityEvents - Unity Manual” <https://docs.unity3d.com/es/530/Manual/UnityEvents.html> (consultado el 20 de abril de 2023).

[19] “Scene View Ortographic - Unity Manual” <https://docs.unity3d.com/es/530/Manual/SceneViewNavigation.html> (consultado el 12 de mayo de 2023).

[20] “Canvas - Unity Manual” <https://docs.unity3d.com/es/2019.4/Manual/UICanvas.html> (consultado el 12 de mayo de 2023).

[21] “Render Texture - Unity Manual” <https://docs.unity3d.com/Manual/class-RenderTexture.html> (consultado el 12 de Mayo de 2023).

[22] “Quick start to TextMeshPro” <https://learn.unity.com/tutorial/working-with-textmesh-pro#5f86410eedbc2a00249a4928> (consultado el 19 de mayo de 2023).

[23] “Audio Sources - Unity Manual” <https://docs.unity3d.com/ScriptReference/AudioSource.html> (consultado el 22 de mayo de 2023).

[24] “Repositorio Github Asociado al proyecto” <https://github.com/alu0100819786/TFG-Videojuego-RPG>

[25] “KickStarter” <https://www.kickstarter.com/?ref=nav> (consultado el 1 de junio de 2023).

[26] “IndieGogo” <https://www.indiegogo.com/> (consultado el 1 de junio de 2023).

[27] “Google Play Console” <https://developer.android.com/distribute?hl=es-419> (consultado el 1 de junio de 2023).

[28] “Steamworks” <https://partner.steamgames.com/steamdirect?l=latam> (consultado el 1 de junio de 2023).

[29] “Epic Games distribution” <https://store.epicgames.com/es-ES/distribution> (consultado el 1 de junio de 2023).

[30] “Unreal Engine 5.2 Documentation” <https://docs.unrealengine.com/5.2/en-US/> (consultado el 1 de junio de 2023).