

# Diseñando “The Lord of Trolls”

Autor: Jose Vicente Anton Coy

Tutor: Jordi Duch Gavaldà

Profesor: Joan Arnedo Moreno

Máster en diseño y programación de videojuegos

Diseño de experiencias de Juego

# Créditos/Copyright



Esta obra está sujeta a una licencia de Reconocimiento - No Comercial – Sin Obra Derivada

[4.0 España de Creative Commons.](https://creativecommons.org/licenses/by-nc-nd/4.0/es/)

## FICHA DEL TRABAJO FINAL

<b>Título del trabajo:</b>	<i>Diseñando "The Lord of Trolls"</i>
<b>Nombre del autor:</b>	<i>Jose Vicente Anton Coy</i>
<b>Nombre del colaborador/a docente:</b>	<i>Jordi Duch Gavalda</i>
<b>Nombre del PRA:</b>	<i>Jordi Duch Gavalda</i>
<b>Fecha de entrega (mm/aaaa):</b>	<i>01/2024</i>
<b>Titulación o programa:</b>	<i>Máster en diseño y programación de videojuegos</i>
<b>Área del Trabajo Final:</b>	<i>Diseño de experiencias de Juego</i>
<b>Idioma del trabajo:</b>	<i>Español</i>
<b>Palabras clave</b>	<i>Juego, Fantasía, Diseño, Estrategia</i>
<b>Resumen del Trabajo (máximo 250 palabras):</b> <i>Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo</i>	
<p>El proyecto de "The Lord of Trolls" consiste en un juego de estrategia con toques de fantasía. La historia relata la perspectiva de los "Trolls" que son masacrados por los humanos y otras razas, en esta problemática despertara un líder que será el jugador y deberá comandar a las diferentes razas aliadas para combatir a los enemigos. La metodología de trabajo a emplear será la de <b>Kanban</b> limitando las tareas por hitos.</p> <p>Los resultados han sido satisfactorios porque ha cumplido las metas propuestas, aun así, se llega a la conclusión de que hace falta aplicar más trabajo sobre el mismo para alcanzar un producto óptimo, para poner a disposición del usuario final.</p>	

**Abstract (in English, 250 words or less):**

The project for 'The Lord of Trolls' consists of a strategy game with elements of fantasy. The storyline presents the perspective of the Trolls, who are being massacred by humans and other races. Amidst this conflict, a leader will emerge, who is the player. The player will need to command various allied races to combat their enemies. The working methodology to be employed is Kanban, with tasks limited by milestones.

The results have been satisfactory because it has met the proposed goals, yet, it is concluded that more work needs to be applied to it to achieve an optimal product, to make it available to the end user.

**Enlaces al repositorio**

Repositorio del proyecto: [https://gitlab.com/jantoncoy/tfm\\_videojuegos.git](https://gitlab.com/jantoncoy/tfm_videojuegos.git)

Release Hito 2: [https://gitlab.com/jantoncoy/tfm\\_videojuegos/-/releases/Hito-2](https://gitlab.com/jantoncoy/tfm_videojuegos/-/releases/Hito-2)

Release Hito 3: [https://gitlab.com/jantoncoy/tfm\\_videojuegos/-/releases/Hito-3](https://gitlab.com/jantoncoy/tfm_videojuegos/-/releases/Hito-3)

Release Hito 4: [https://gitlab.com/jantoncoy/tfm\\_videojuegos/-/releases/Hito-4](https://gitlab.com/jantoncoy/tfm_videojuegos/-/releases/Hito-4)

**Enlaces a videos**

Video Hito 2: <https://youtu.be/-GKsPWzcM-E>

Video Hito 3: <https://youtu.be/LgJqvluE48E>

Video Hito 4: [https://youtu.be/A9q\\_\\_s8ILY0](https://youtu.be/A9q__s8ILY0)

Video Trailer: <https://youtu.be/l8tIUNDy-gc>

## Cita

En mi búsqueda de conocimientos como desarrollador de videojuegos encontré una cita que a mi modo de ver definía muy bien la vida de un desarrollador o jugador de videojuegos, decía así:

*"No importa cuántas veces fracasas, debes seguir adelante.  
Esa es la actitud de un verdadero gamer. Masaya Nakamura"*

Creo que es lo que define el tiempo y esfuerzo que llevamos a cabo las personas en el sector por seguir adelante con lo que nos apasiona.

# Agradecimientos

A mi querida familia y amigos,

Quiero expresar mi sincero agradecimiento por su apoyo inquebrantable a lo largo de este proyecto. Su comprensión y ánimos en el tiempo han sido fundamentales en mi camino hacia la finalización de este TFM.

Este logro es el resultado de nuestro esfuerzo conjunto, y quiero dedicárselo con toda mi gratitud.

## Resumen

El proyecto de "The Lord of Trolls" consiste en un juego de estrategia con toques de fantasía. La historia relata la perspectiva de los "Trolls" que son masacrados por los humanos y otras razas, en esta problemática despertara un líder que será el jugador y deberá comandar a las diferentes razas aliadas para combatir a los enemigos.

La metodología de trabajo a emplear será la de Kanban, limitando las tareas por hitos.

### Palabras clave

Memoria, Trabajo de Final de Máster, Juego, Unity, Fantasía, Estrategia, Diseño

# Índice

<b>1. Introducción.....</b>	<b>11</b>
<b>1.1. Introducción.....</b>	<b>11</b>
<b>1.2. Definición.....</b>	<b>12</b>
1.2.1. efinición de primera zona.....	13
<b>1.3. Objetivos generales.....</b>	<b>14</b>
1.3.1. Objetivos principales.....	14
1.3.2. Objetivos secundarios.....	14
<b>1.4. Metodología y proceso de trabajo.....</b>	<b>15</b>
<b>1.5. Planificación.....</b>	<b>16</b>
<b>1.6. Presupuesto.....</b>	<b>18</b>
<b>2. Análisis de mercado.....</b>	<b>19</b>
<b>2.1. Introducción al mercado de los videojuegos.....</b>	<b>19</b>
<b>2.2. Público objetivo.....</b>	<b>19</b>
<b>2.3. Análisis de la competencia.....</b>	<b>19</b>
<b>2.4. Análisis DAFO.....</b>	<b>23</b>
<b>3. Propuesta.....</b>	<b>25</b>
<b>3.1. Definición de objetivos.....</b>	<b>25</b>
<b>3.2. Modelo de negocio.....</b>	<b>26</b>
<b>3.3. Estrategia de marketing.....</b>	<b>27</b>
<b>4. Diseño.....</b>	<b>28</b>
<b>4.1. Justificación tecnológica.....</b>	<b>28</b>
<b>4.2. Arquitectura del proyecto.....</b>	<b>29</b>
<b>4.3. Escenas del juego.....</b>	<b>30</b>
<b>4.4. Diseño de interfaces.....</b>	<b>31</b>
<b>4.5. Diseño de personaje.....</b>	<b>32</b>
<b>4.6. Diseño de nivel.....</b>	<b>33</b>
<b>4.7. Mecánicas del juego.....</b>	<b>34</b>
<b>4.8. Narrativa.....</b>	<b>35</b>



4.9. Efectos de sonido.....	35
4.10. Controles.....	35
4.11. Diseño de la IA.....	36
<b>5. Implementación.....</b>	<b>39</b>
5.1. Requisitos de instalación.....	39
5.2. Estructura del proyecto unity.....	39
5.3. Nivelado de características.....	40
5.4. Implementación de la IA.....	41
5.5. Traducciones.....	43
5.6. Calculo de daño.....	43
5.7. Iluminación.....	44
5.8. Sistema de misiones.....	44
5.9. Controles.....	44
5.10. Guardado de partida.....	45
5.11. Ayuda visual.....	45
5.12. Minimapa.....	46
5.13. Selector multiple.....	46
<b>6. Demostración.....</b>	<b>47</b>
6.1. Instrucciones de uso.....	47
6.2. Pruebas.....	47
6.3. Guía de usuario.....	48
<b>7. Conclusiones y líneas de futuro.....</b>	<b>50</b>
7.1. Conclusiones.....	50
7.2. Líneas de futuro.....	51
<b>Bibliografía.....</b>	<b>52</b>
<b>Anexos.....</b>	<b>55</b>

# Figuras y tablas

## Índice de figuras

Figura 1: <i>Goblin</i> generado con <i>DALL·E 3</i> con pícaro aspecto.....	11
Figura 2: <i>Troll</i> generado con <i>DALL·E 3</i> con un aspecto amigable.....	12
Figura 3: Ejemplo de <i>Kanban</i> .....	15
Figura 4: Diagrama de componentes en unity.....	29
Figura 5: Diagrama de escenas.....	30
Figura 6: Interfaz del menú principal.....	31
Figura 7: Interfaz de las opciones de hub en el juego.....	31
Figura 8: Diseño del héroe principal.....	32
Figura 9: Guerrero humano.....	32
Figura 10: Representación del nivel 1.....	33
Figura 11: Estados disponibles para las unidades del jugador.....	36
Figura 12: Árbol de comportamiento del agente.....	36
Figura 13: Árbol de comportamiento del aldeano.....	37
Figura 14: Árbol de comportamiento del explorador.....	37
Figura 15: Árbol de comportamiento de los guerreros y arqueros.....	38
Figura 16: Ejemplo de iluminación variable.....	44
Figura 17: Elemento cambiando de color para facilitar la detección por parte del jugador.....	45
Figura 18: Se muestra una selección múltiple de unidades.....	46
Figura 19: Menú de construcción o artesanías.....	48
Figura 20: Menú de unidades al pulsar sobre herrería.....	48
Figura 21: Recursos.....	49
Figura 22: Mapa de objetivos señalados con una estrella.....	49

## Índice de tablas

Mesa 1: Coste por hora.....	18
Tabla 2: Tabla de estadísticas de las diferentes unidades.....	40

## Índice de fragmentos de código

Código 1: Plantilla básica de perfil para IA.....	42
Código 2: Plantilla básica de traducción.....	43

# 1. Introducción

## 1.1. Introducción

Actualmente, en el mercado existen multitud de obras y videojuegos donde se hace uso de los *trolls*, *goblins*, orcos y otros seres relacionados del folclore de todo el mundo, en casi todas estas obras se representan como seres **diabólicos**, cosa que hace empatizar al jugador con la parte contraria para defenderse de ellos. Este proyecto viene en consecuencia por el **interés creciente** del público en estos seres y la incógnita de que haría el jugador si estuviera en el lado contrario.



Figura 1: *Goblin* generado con DALL·E 3 con pícaro aspecto.

Actualmente, mayoritariamente **Japón** está empleando estos seres del folclore con multitud de obras como en *Goblin Slayer* [1] del famoso autor Kumo Kagyu u *Overlord* [2] de Kugane Maruyama. No hay que irse muy lejos para ver obras más reconocidas en **occidente** como puede ser *El señor de los anillos* de J. R. R. Tolkien [3] o *World of Warcraft* de Blizzard Entertainment [4] famosa desarrolladora de videojuegos. El gran éxito de estas obras denota el interés de los usuarios en estos seres, haciéndolos un recurso apropiado para el desarrollo de obras basadas en ellos.

Existen juegos que hacen más o menos cosas parecidas como *Overlord* de la desarrolladora *Codemasters* [5], en este juego controlabas a un señor de las tinieblas que invocaba a *goblins* con su poder y arrasaba con sus enemigos. Por otra parte, podemos ver también grandes juegos como *Sombras de Mordor* [6] donde te enfrentas a hordas de orcos, escalando en poder en el caso de que el usuario pierda.

## 1.2. Definición

El proyecto consistirá en un juego donde se **controlara** seres del folclore relacionados entre sí como son los *trolls*, orcos y *goblins* para defenderse de forma **estratégica** de otras razas como son los humanos.

De esta forma supliremos un género que es muy llamativo en los videojuegos que será ser el malo o el monstruo de la obra. Al ser un género que es muchas veces difícil de empatizar con el usuario final, no se generan muchos juegos dedicados a ello de forma exclusiva, por lo que existe un **nicho de mercado** que se puede suplir con relativa facilidad.

Al final el resultado que se quieren obtener es que el usuario genere una **empatía** con las razas anteriormente mencionadas de seres folclóricos, para ello se les dará una visualización amigable y se armará la trama para que el usuario se sienta forzado a sentir un sentimiento de ayuda hacia ellos.



Figura 2: Troll generado con DALL·E 3 con un aspecto amigable.

El producto final será un juego de estrategia en **tiempo real (RTS)** donde se irán **desbloqueando** zonas conforme se vayan completando retos del juego, la idea es que el juego no tengan ningún **final**, simplemente te defiendas durante generaciones de las otras razas que amenazan la existencia de los dominios del jugador. Esto no quiere decir que no exista la derrota, se podrán perder zonas si los contraataques destruyen edificaciones críticas del jugador, haciendo que el jugador tenga que volver a hacer un esfuerzo para conquistar las zonas.

En el juego se podrá construir edificaciones como torres de defensa, bases o casas para aumentar la defensa y ampliar el dominio existente, de esta forma poco a poco se ampliara las **fronteras** del jugador o de la IA conforme avance el juego. Se podrá generar diferentes unidades como son los aldeanos, guerreros o arqueros.

El héroe se podrá controlar como una unidad más de batalla, teniendo características **aumentadas** conforme se suba de nivel en el juego, de esta forma el jugador notara una **progresión** del héroe, lo que causara un sentimiento de agrado en el mismo. El héroe, aunque pueda **debilitarse** durante la batalla, se volverá a recuperar pasado un tiempo de castigo para evitar que se abuse en exceso de él, esto también quitara bonificaciones por su presencia entre las tropas.

### 1.2.1. efinición de primera zona

La primera zona del juego que será la parte que se desarrollara en este proyecto, esta zona será **abierta** para el jugador teniendo **misiones guiadas** para que aprenda las mecánicas básicas. Una vez que el jugador haya superado las misiones guiadas, se deberá enfrentar a los enemigos, en este caso la raza humana.

El mapa será de tamaño mediano con varias ciudades y aldeas humanas, utilizarán varias estrategias y unidades.

Puntos importantes:

- La idea es que los **exploradores** tengan un uso muy importante, serán quienes darán **información** de la posición de los enemigos y sus estructuras, basándonos en esto se podrán lanzar varios ataques y la IA podrá actuar de una forma realista sin ser omnipotente.
- Se emplearán estructuras para **mejorar** la extracción y recolección de recursos, si no se emplean tendrán un desempeño base.
- Tanto la obtención de nuevas unidades o edificios y sus mejoras se obtendrán con un **árbol de habilidades**, que serán las habilidades principales del protagonista.
- Habrá **vista de pájaro** o de mapa y en **tercera persona** desde nuestro héroe.

### 1.3. Objetivos generales

En este punto se detallan los objetivos del proyecto de forma resumida.

#### 1.3.1. Objetivos principales

Objetivos de la aplicación/producto/servicio:

- Llegar al **máximo** público posible.

Objetivos para el cliente/usuario:

- Generar **empatía** en el jugador por seres folclóricos.
- **Divertir** al jugador y sobre todo generar un reto en él.

Objetivos personales del autor del proyecto:

- Desarrollar un juego con un buen diseño.
- Pasar de la idea al producto.

#### 1.3.2. Objetivos secundarios

Objetivos adicionales que enriquecen el proyecto.

- Generar proyecto **estable** a largo plazo.
- Proporcionar al usuario final una alternativa a los juegos actuales.

## 1.4. Metodología y proceso de trabajo

Se va a realizar el proyecto con la metodología de trabajo **Kanban**. Cada hito tendrá sus tareas definidas, cuando se comience la tarea se pasarán a "En curso" evitando tener más de dos o tres tareas simultáneas, luego pasarán a **testing** en este punto se podrán probar en conjunto con otras o en individual y finalmente cuando pase las pruebas pasará ha finalizado. Para poder pasar al siguiente hito se deberá completar las tareas del anterior hito hasta el final del proyecto.

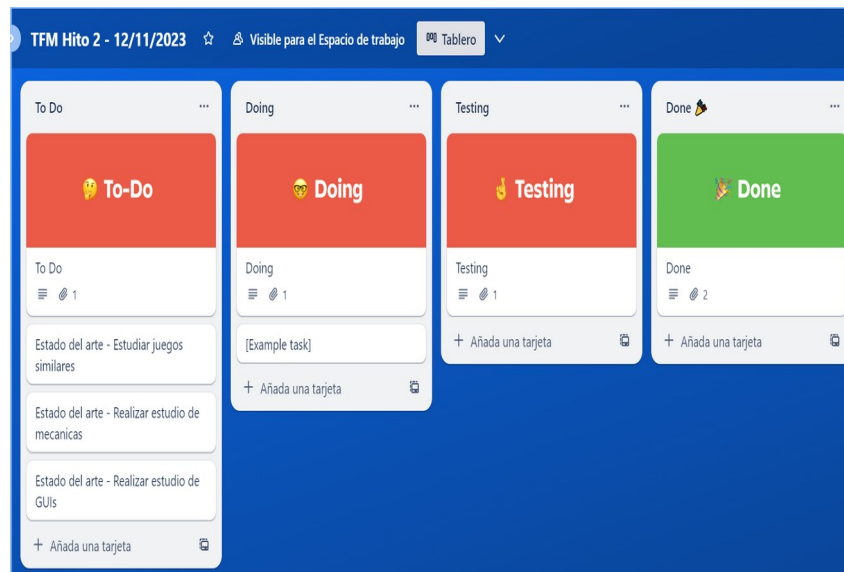


Figura 3: Ejemplo de Kanban

Para cada hito donde exista desarrollo se harán **pruebas integradas** de todo el desarrollo y se corregirá los errores, si fueran errores críticos pasarían a ser parte de las tareas del siguiente hito.

## 1.5. Planificación

Las tareas comentadas en cada hito pasarán a formar las tareas de los tablonos de *Kanban*.

### Hito 1 - 08/10/2023: Definición en memoria

En esta primera entrega se definirá el alcance del proyecto junto al presupuesto.

#### Tareas realizadas:

- Definir la descripción básica y presupuesto.
- Detallar tareas a realizar en siguientes hitos.
- Creación de tablonos *Kanban*.

### Hito 2 - 12/11/2023: Primera versión del juego

Se entregará la memoria con el estado del arte y con una versión mínima de las mecánicas.

#### Tareas realizadas:

- Estado del arte.
- Creación de GIT y proyecto inicial.
- Bocetos iniciales de la IA.
- Bocetos del nivel 1.
- Bocetos del GUI.
- Implementación básica en *whitebox*.



### **Hito 3 - 17/12/2023: Versión jugable**

En este punto se habrán incorporado elementos visuales y sonoros, además de modelos (**Assets**) al primer nivel.

#### **Tareas realizadas:**

- Modelado de unidades básicas.
  - Unidades
  - Edificios
- Incorporación de sonido.
- Incorporación de texturas.
- Incorporar IA del juego basado en los bocetos.

### **Hito 4 - 14/01/2024: Entrega final**

Cuando se llegue a este punto se habrá pulido la entrega lo máximo posible al igual que la memoria.

#### **Tareas realizadas:**

- Agregar mejoras generales en el juego.
- Solucionar errores de las primeras entregas.
- Terminar memoria.

### **Hito 5 - 31/01/2024: Defensa de la memoria**

Se efectuará la defensa del proyecto frente al comité evaluador.

#### **Tareas realizadas:**

- Preparación para la defensa.

## 1.6. Presupuesto

El juego está pensado para ser un juego **indie** incremental en el tiempo de bajo presupuesto, para ello se necesita los siguientes perfiles:

Perfil	Precio / Hora
Programador en <b>Unity</b>	105,00 € Brutos [7]
Artista Gráfico	194,00 € Brutos [7]
Artista de Sonido	127,00 € Brutos [7]
Asesor Legal	175,00 € Brutos [7]

Mesa 1: Coste por hora

Se calcula que habrá un programador 4 horas diarias, el desarrollo dura 3 meses, siendo las últimas semanas para **pulir** el juego existente.

230 horas \* 1 programador = 24.150 €

Luego, en la parte artística se calcula que sea aproximadamente 40 horas para crear escenarios en personajes y diferentes interfaces.

40 horas \* 1 Artista Gráfico = 7.760 €

En el aspecto sonoro se predice que tendrá unas 30 horas en la creación de los sonidos: menús, interacciones de los personajes con el entorno, acciones de los personajes sobre otros personajes y ambientales.

30 horas \* 1 Artista de Sonido = 3.810 €

Por último, se necesitaría un asesor legal para realizar las patentes del juego y **acciones legales** que fueran necesarias, para ello se calcula que serían 20 horas en total.

20 horas \* 1 Asesor Legal = 3.500 €

Para las siguientes fases solo se tendrían en cuenta el programador de *Unity* y el diseñador gráfico para los nuevos elementos.

Precio total **primera** fase: 39.220 €

Precio de fases **siguientes**: 31.910 €

## 2. Análisis de mercado

### 2.1. Introducción al mercado de los videojuegos

El mercado de los videojuegos en pleno 2023 es uno de los negocios que más **beneficios** genera en la actualidad, en concreto en Estados Unidos [8] es la segunda industria que genera más beneficios. Por otra parte, en Europa es un mercado que **crece** sin precedentes [9] siendo una de las industrias mejor posicionadas del mercado por sus beneficios.

### 2.2. Público objetivo

El mercado muestra un claro objetivo por **aumentar** el número de juegos disponibles en plataformas **móviles** como Android o iOS [9], por lo que un principal público objetivo son estas plataformas donde existe un público más amplio, incluso desde una perspectiva crítica se podría decir que es el público más amplio actual por la sencilla razón de que la mayoría de personas tienen un dispositivo móvil para su uso **cotidiano** a diferencia de ordenadores o consolas.

Otros públicos objetivos serían las plataformas de ordenador y consolas, la diferencia más amplia es que el mercado de consolas suele ser un mercado **excluyente** por sus fabricantes y sus diferentes fases de filtrado, lo que hace que el ordenador se convierta en la plataforma inicial de llegada de un producto junto a los dispositivos móviles.

### 2.3. Análisis de la competencia

Para el análisis se han revisado varios juegos y se ha escogido los que se ha creído que son **modelo** en la industria como es *Warcraft III: Reign of Chaos* y *The Frozen Throne* [10] y *Total War: Warhammer II* [11].

Además, por la temática de juegos de estrategia a continuación también se ha llevado a cabo el análisis de *Age of Mythology* [12], *Age of Empires* [13], *SpellForce* [14] y *Trópico* [15].

En el análisis nos hemos fijado tres puntos, como son la **jugabilidad** que expresa las mecánicas principales y la flexibilidad del mismo, el **estilo** del juego que indica como han querido mostrar al jugador el producto para resaltar ciertas características. Finalmente, la **narrativa** que explica a grandes rasgos cómo el guion del juego lleva al jugador a realizar ciertas situaciones y de que forma.

## Warcraft III

### Jugabilidad

- Presenta batallas de **corta** duración con mucha acción.
- Gran importancia en la utilización de los héroes.
- Batallas de escala media (menos de 100 unidades)

### Estilo

- Gráficos con texturas **planas** y simples.
- Es menos costoso a nivel computacional.

### Narrativa

- Ambientado en un mundo de fantasía con seres mágicos.

## Warhammer II

### Jugabilidad

- Las batallas son más **largas**, incluso pueden llevar varias fases.
- Batallas a **gran escala** (miles de unidades).
- Se enfoca más en la estrategia.

### Estilo

- Gráficos más **realistas** y con mucho más detalles.
- Coste de computación más **alto**.

### Narrativa

- Ambientado en el folclore de *Warhammer*, enfocado en la construcción de un imperio y su gestión.

## Age of Mythology

### Jugabilidad

- Presenta batallas agrupadas en varias **fases** hasta su finalización.
- Utilizan dioses y criaturas míticas para la batalla.
- Batallas de escala media (menos de 100 unidades)

### Estilo

- Mezcla estilos míticos con realistas.
- No muy costoso a nivel computacional.

### Narrativa

- Basada en varias culturas míticas uniéndolas en una historia común.

## Age of Empires

### Jugabilidad

- Permite el desarrollo de una civilización a lo largo de diferentes **épocas**.
- Estrategia en tiempo real (**RTS**).
- Suele ser de acción más directa.

### Estilo

- Presenta gran detalle en las estructuras empleadas en el juego.
- Coste computacional medio.

### Narrativa

- Cada civilización presenta su historia propia.

## SpellForce

### Jugabilidad

- Mezcla de juego en tiempo real y **RPG**.
- Gran importancia en la utilización de los héroes.
- Grandes batallas entre diferentes grupos de forma simultánea con diferentes **alturas** en el terreno.

### Estilo

- Estilo principalmente basado en la fantasía.
- En su tiempo tenía que ser jugador con potencia gráfica **alta**.

### Narrativa

- Ambientado en un mundo de fantasía, tiene una campaña principal y misiones secundarias.

## Trópico

### Jugabilidad

- Batallas a una escala reducida, incluso con unidades en el agua.
- Se enfoca mucho en la **negociación** política, gobiernas como dictador en el juego.

### Estilo

- Estilo gráfico muy colorido detallando ambientes cálidos.
- Coste de computación **medio**.

### Narrativa

- Inspirado en el Caribe durante la Guerra Fría.

## 2.4. Análisis DAFO

En este punto se lleva a cabo un análisis de los puntos fuertes y débiles del proyecto propuesto.

### Debilidades

- **Reconocimiento en el mercado:** al ser un producto nuevo puede ser un desafío darlo a conocer y sacarle un rendimiento apropiado.
- **Juego de bajo presupuesto:** al ser un juego creado por un equipo pequeño, **limita** mucho la experiencia para el usuario final.

### Amenazas

- **Gran competencia:** la industria de los videojuegos como previamente se ha nombrado, es una de las más grandes del mundo, por lo que salen títulos nuevos que pueden **opacar** a un producto.
- **Opiniones negativas:** la industria pasa por una fase donde los usuarios critican de forma activa por redes sociales aspectos muy concretos de los juegos, por lo que puede hacer que el usuario final pierda las ganas de jugar antes de probar el producto.

### Fortalezas

- **Propuesta original:** al centrarnos en la perspectiva de los *trolls* el juego toma un aspecto más interesante para el jugador.
- **Sector de jugadores:** muchos jugadores consumen juegos de estrategia habitualmente, por lo que si lo ven como una propuesta válida lo jugaran.
- **Empatía con los jugadores:** el juego intentará empatizar con los jugadores, por lo que causara sentimientos positivos al jugarlo.

## Oportunidades

- **Mercado en alza:** el mercado de los videojuegos es un mercado en constante **crecimiento**, por lo que es más probable encontrar una oportunidad de éxito.
- **Múltiples plataformas:** el juego inicialmente saldrá en pocas plataformas, pero se puede **expandir** a nuevas por su motor de juego Unity que permite una flexibilidad muy amplia.
- **Actualizaciones constantes:** el juego traerá nuevas zonas, por lo que será un juego que recibirá expansiones durante su tiempo de vida, lo que hará que el jugador sienta ganas de quedarse jugando por el contenido nuevo.



## 3.Propuesta

### 3.1. Definición de objetivos

#### Experiencia de los jugadores

El jugador tiene que tener una posición de **liderazgo** en el mundo, por ello será el pilar principal y el que dominará las tropas, por otra parte, tendrá un **crecimiento constante**. También se quiere lograr que los usuarios tengan una experiencia similar entre plataformas diferentes, por ello los menús, opciones e interfaces deben ser ampliamente **compatibles** entre ellas.

#### Mecánicas del juego clave

- **Construcción:** se debe de poder construir estructuras con los recursos adquiridos que sean de utilidad para el usuario, tanto defensivas como de expansión. Además, las estructuras tendrían que tener un tiempo de construcción para que tengan que tener un planteamiento previo a la acción, de esta forma si las unidades enemigas llegaran antes de que esté totalmente operativa sería fácilmente destruibles.
- **Desarrollo del personaje:** el personaje debe ir creciendo por niveles, la experiencia para subir de nivel se iría incrementando con la construcción de estructuras y por el cumplimiento de objetivos o misiones. Los niveles aportarán un incremento de las características del héroe en el juego.
- **Mejora de unidades y construcciones:** se podrán utilizar los puntos adquiridos en la subida de nivel para desbloquear nuevas unidades y aumentar las características del resto de unidades.
- **Manejo de recursos:** se debe manejar los recursos de forma planeada, tanto los materiales como las construcciones, de esta forma tendrán la máxima eficacia.
- **Gestión de aldeas y zonas:** el juego tendrá varias zonas, por lo que se tiene que poder cambiar entre ellas y gestionar las zonas de forma dinámica.
- **Decisiones estratégicas:** la IA debe de poder emplear estrategias que sean simples, pero efectivas contra el jugador, como la estrategia en pinza, ataque directo o distensión y concentración.

## Idiomas

El juego estará disponible en los idiomas inglés y español con la posibilidad de expandir de forma **sencilla** los textos a otros idiomas.

## Plataformas

Las plataformas para las que estará disponible inicialmente serán para *Windows* y *MacOs* en la página principal de desarrollo y finalmente en la *Play Store* para sistemas Android. Posteriormente, si el juego ganara público y recibiera ingresos, se podría **promover** a otras plataformas como *Steam* [16] o *Epic Games* [17].

## Publico clave

Los públicos ideales para este proyecto son varios que pueden llegar a ser complementarios entre sí:

- **Jugador consumidor:** este público sentiría gran pasión por el título, tendrán un **vínculo sólido** por él. Buscarán expandir su experiencia de juego mediante la **adquisición** de personalizaciones visuales, serán propensos a explorar el contenido de pago, por lo que le darán sostenibilidad al juego.
- **Jugador casual:** serán un público esporádico que disfrutaran del juego de forma más **relajada** sin necesidad de explorar las opciones de pago, **participaran** en eventos temporales dándole publicidad al título y generando marketing de forma indirecta.

## 3.2. Modelo de negocio

El juego presenta **anuncios** en el menú principal del juego a modo de modal que posteriormente se puede cerrar. Además, el juego ofrece la posibilidad de comprar paquetes de personalizaciones visuales para los personajes. Estas adquisiciones son puramente cosméticas y agregan variedad al estilo visual del juego, pero no confieren una ventaja *in game* alguna, manteniendo el equilibrio del juego.

En conclusión, este modelo **mixto** asegura ingresos fijos y variables, de esta forma los usuarios que quieran seguir mejorando visualmente el juego producirán de forma **activa** ingresos y, por otra parte, al ser el juego **gratuito** llegará a un público mayor.

## Ejemplos en el mercado

Existen muchos juegos que utilizan este tipo de modelo con gran éxito, en este apartado se van a nombrar los que se creen más influyentes.

- **Fortnite [18]:** saca promociones regulares de diferentes franquicias con grandes ventas, de esta forma aporta un incremento del interés del jugador y la posibilidad de adquirir el contenido.
- **League of Legends (LOL) [19]:** dentro de su folclore saca nuevos estilos a sus personajes.
- **Paladins [20] y Overwatch [21]:** dos juegos muy parecidos entre sí que generan dentro de su universo nuevas personalizaciones que se pueden adquirir mediante el juego normal o la adquisición de paquetes.
- **Valorant [22]:** cada ciertos meses sacan temáticas nuevas para sus personajes y armas, sobre todo sacan diseños novedosos y frescos para el público más joven.

### 3.3. Estrategia de marketing

La estrategia de marketing será principalmente **mostrar anuncios** de bajo coste en aplicaciones de **terceros** que sean breves atractivos para los posibles jugadores, además como complemento durante periodos específicos (Navidad, Pascua) se regalaran paquetes de personalización visual por promover el juego en redes sociales (esto se controlaría mediante un sistema básico de clic).

## Ejemplos en el mercado

- **Candy Crush Saga [23]:** suelen mostrar en juegos de su mismo **ámbito** anuncios de sus juegos para promover el interés de los jugadores.
- **Hearthstone [24]:** mucha publicidad por redes sociales y servicios de streaming.
- **Clash of Clans [25]:** combinación tanto de publicidad de juegos de terceros como de redes sociales.

## 4. Diseño

### 4.1. Justificación tecnológica

Para este proyecto se va a utilizar como motor gráfico *Unity* [26] por su **facilidad** al desarrollar contenido interactivo, *Unity* se divide en **componentes** altamente personalizables, lo que hace que con pequeños scripts se tenga una **versatilidad** muy alta. Por otro lado, la comunidad de desarrolladores es una de las más grandes del mundo en el sector de los videojuegos, por lo que es muy fácil encontrar documentación o soluciones a posibles problemas durante el desarrollo.

Por motivos de tiempo estimado de desarrollo del proyecto se van a emplear muchos recursos externos, estos recursos serán de **libre uso** o de **licencia adquirida**.

#### Software utilizado

- **Unity**: en la versión *2022.3.10f1 LTS* motor gráfico que será quien contenga el juego nombrado en este proyecto.
- **Gimp** [27]: para el retoque de imágenes y texturas.
- **Blender** [28]: para la modificación de animaciones y arreglo del mallado.
- **Mixamo** [29]: nos servirá para la generación de las **animaciones** de forma sencilla, posteriormente serán editadas en *Unity* y *Blender*.

#### Recursos utilizados

- **Personajes y decoración del juego**: los personajes del juego se utilizarán de la página distribuidora *Synty* [30], página dedicada a la generación de assets con **polígonos reducidos** o *low poly*.
- **Texturas del nivel**: las texturas se sacarán de *OpenGameArt* [31].
- **Armas**: las armas serán adquiridas de *Leartes Studios* [32], página especializada en venta de assets.
- **Sonido ambiente y de acciones**: se emplean sonidos de *Ovni Sound* [33].
- **Hub**: creados manualmente con el software *gimp*.

## 4.2. Arquitectura del proyecto

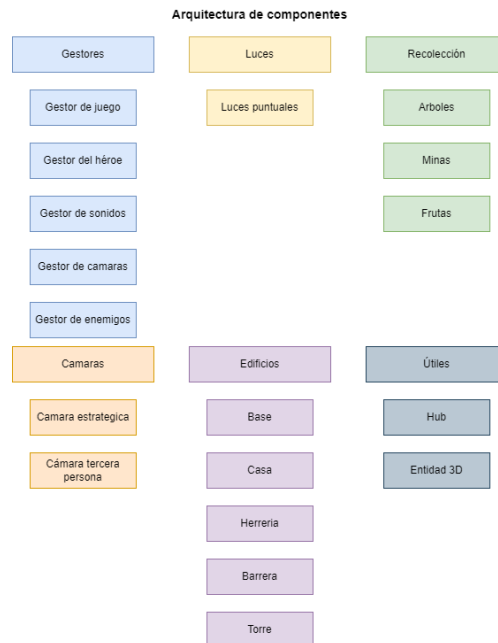


Figura 4: Diagrama de componentes en unity

Se van a generar los componentes vistos en la figura 4, estos componentes serán los **pilares** principales que sostendrán el juego de forma funcional. A continuación se detallan las funcionalidades de los componentes más importantes:

- **Gestor de juego:** este componente controlará el sentido del juego, determinará cuando debe acabar la fase y cuando debe comenzar otra.
- **Gestor del héroe:** será el gestor del jugador, se encargará de mapear las acciones del jugador a reacciones dentro del juego, desde controlar al héroe, mandar a atacar ciertas unidades o indicar a los aldeanos que deben conseguir algún recurso, entre otras posibilidades.
- **Gestor de cámaras:** su función principal es saber en todo momento que cámara está activa y en el caso de que fuera necesario realizar las **configuraciones** en ellas para su uso. Este componente controlará los componentes de la cámara estratégica y la de tercera persona.
- **Gestor de enemigos:** su principal objetivos es controlar los gestores de las ciudades y determinar comunicaciones entre ellos.

El resto de componentes se dedica a la gestión **propia** de sus funciones.

### 4.3. Escenas del juego

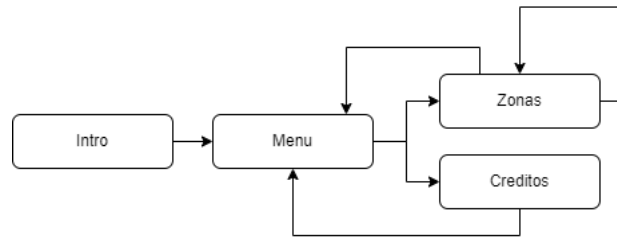


Figura 5: Diagrama de escenas

El juego contendrá principalmente cuatro escenas en su fase **inicial**, la primera escena será la introducción (**Intro**) que no se podrá **evadir** y que presentara el estudio o los estudios implicados en el desarrollo, esta escena dará paso al menú.

El menú contendrá las opciones de iniciar el juego, cargar una partida (limitado a 3), ver los créditos y finalmente salir del mismo. Al pulsar en ver los créditos se empezará la cinemática visualizando los créditos a los diferentes autores de las obras incluidas en el juego, esta parte tendrá un botón de **omitir**.

Cuando se pulse sobre el botón de comenzar partida o cargar partida se cargará la escena de la zona inicial o la zona donde estuviera por última vez el jugador, desde esta escena se podrá volver al menú o **saltar** a otras escenas de zonas jugables.

#### 4.4. Diseño de interfaces

Las interfaces serán construidas para que sean lo más simple y **funcionales** entre diferentes plataformas **táctiles** o no, la idea artística principal es que haya **homogeneidad** entre ellas y así no realizar un impacto visual en el jugador que pueda llegar a ser desagradable.



Figura 6: Interfaz del menú principal

En la siguiente figura se puede apreciar como se mantiene la armonía de los colores jugando con el color verde relacionado con el folclore de los *trolls* y razas asociadas.



Figura 7: Interfaz de las opciones de hub en el juego

## 4.5. Diseño de personaje

Para el diseño de los personajes se ha optado por personajes con bajo nivel de polígonos, también llamados *low poly*, esto nos permite mantener el **rendimiento** del juego mientras existen un alto nivel de entidades dentro de él.



Figura 8: Diseño del héroe principal

Como se puede apreciar en los personajes del bando *troll* se han escogido personajes muy humanizados dando colores verdes para dar su color característico a la raza, además se han estilizado con rasgos vikingos dado su temperamento y **ferocidad** en la batalla.



Figura 9: Guerrero humano

En el guerrero humano se ha escogido colores claros y azules para representar la nobleza y la **confianza** en sus fuerzas, además llevan equipadas armas más **sofisticadas** como espadas largas que las tropas *trolls* que llevan hachas.



## 4.6. Diseño de nivel



Figura 10: Representación del nivel 1

En la figura anterior se puede visualizar las zonas más importantes numeradas según el orden en el que jugador interactuara con ellas.

1. Será la zona donde el jugador **comenzara** el nivel, tendrá su base inicial construida y donde se desarrollara el punto más relevante de **gestión** del jugador.
2. Zona de comida donde el jugador podrá **recolectar** este recurso para incrementar sus posibilidades en la batalla.
3. Zona de metales, esta zona está más alejada, pero será una zona de **conflicto** donde la IA enemiga podrá luchar por su uso.
4. Primer poblado humano que será de un tamaño **moderado**, para que el usuario pueda enfrentarse a ellos fácilmente.
5. Segunda zona de comida para que el usuario pueda **recolectar** más recursos y aumentar sus tropas.
6. Segundo poblado del nivel que será el verdadero **reto** para el usuario, este poblado contará con más unidades enemigas y defensivas.
7. Segunda zona de metales donde también será de difícil acceso por su cercanía al poblado humano.

## 4.7. Mecánicas del juego

- **Recolección de recursos:** para aumentar el número de unidades y edificios se **necesitará** materiales, para ello los aldeanos se deben de poder mandar a los recursos, en el caso de que se **agote** el recurso deberán buscar otro cercano.
- **Construcción de bases:** las bases servirán como almacenes para los aldeanos para **guardar** los diferentes recursos, por lo que se deben poder construir cerca de ellos.
- **Producción de unidades:** mediante la construcción de una herrería se podrán **producir** las unidades desbloqueadas hasta el momento.
- **Exploración con niebla de guerra:** el mapa inicialmente estará **oculto** para el jugador, por lo que deberá utilizar las diferentes unidades para explorar lo máximo posible.
- **Control de puntos estratégicos:** en los sitios de mayores recursos habrá más **disputa** con la IA por su conquista, por lo que se debe mantener en todo momento una posición defensiva.
- **Subida de nivel:** conforme se superen las misiones y se construyan más edificios, se ganara experiencia y se podrá **aumentar** el nivel del héroe, aumentando sus características base (ataque, defensa y vida).
- **Personalización de unidades:** se podrá adquirir opciones de personalización mediante una tienda integrada o instaladores de la personalización.
- **Tecnología y mejoras:** cuando se suba de nivel se adquiere un punto de habilidad que podrá ser gastado en el árbol de mejoras y tecnologías.
- **Realización de misiones:** se podrá recibir misiones durante el juego, al cumplir los objetivos descritos se recibirá una recompensa (normalmente experiencia).

## 4.8. Narrativa

La narrativa tratará como punto principal la **salvación** de la raza de *trolls* y aliados en el continente ficticio de *Kumil*. Todo el poblado existente pondrá las **esperanzas** en el Héroe controlado por el jugador, que deberá mostrar su valía y vencer a la raza invasora humana.

De esta manera se otorga el liderazgo de las tropas al jugador y empezará su aventura, se pondrá especial **énfasis** en conseguir una perspectiva en él se pueda llegar a sentir **empatía** por la raza. Para ello se humanizarán las formas de los *trolls*, dándole aspecto tosco como la cultura *vikinga* en historias míticas.

## 4.9. Efectos de sonido

Los efectos de sonidos empleados en el juego serán los siguientes:

**Sonido ambiente:** cuando se cambie entre menús, zonas o cuando cambie la situación sonarán un sonido ambiente diferente.

**Pulsación sobre la unidad u objeto:** se oirá un sonido breve que represente esa entidad, sea un enemigo, un objeto de recolección o una estructura.

**Mandar órdenes a las entidades:** al mandar una entidad a realizar una acción se oirá la acción a realizar: atacar, recolectar, reparar, moverse a una posición o incluso producir unidades en la herrería.

## 4.10. Controles

Para este punto se ha decidido la **sencillez** en los controles y que todo se pueda realizar con pocos clics de ratón o toques táctiles, de esta forma el usuario podrá familiarizarse con los controles de forma sencilla e **intuitiva** sin necesidad de explicaciones.

Las acciones contempladas por los inputs serían las siguientes:

- **Pulsación sobre botón:** clic simple o toque simple.
- **Selección de diversas unidades:** clic sostenido de punto A al punto B o pulsación sostenida efectuando una zona de selección rectangular.
- **Desplazamiento de direcciones:** arrastre táctil o pulsación de botones de dirección.

#### 4.11. Diseño de la IA

En el apartado de la IA de las unidades **controladas por el jugador** se ha escogido utilizar una **máquina de estados**, de esta forma al ser **órdenes directas** se pueden iterar fácilmente sobre el mismo estado hasta que deje de ser válido o el usuario le mande otra orden.

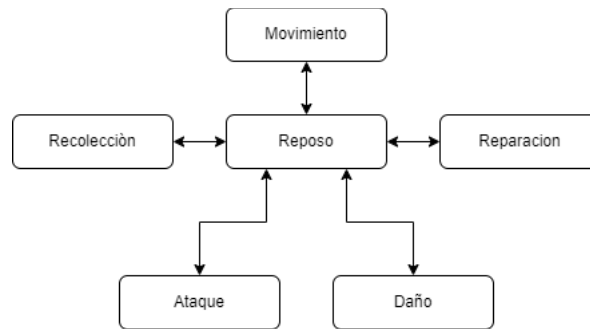


Figura 11: Estados disponibles para las unidades del jugador

Por otra parte, para la **IA enemiga** se han diseñado **árboles de comportamiento** para que sean **autosuficientes** y no dependan de acciones de otras entidades para actuar. Además, también existirán **agentes especializados** en la gestión del poblado, estos agentes se encargarán de gestionar las acciones **críticas** como generar recursos, mandar a recolectar a los aldeanos o atacar cuando se encuentre a los *trolls* en el juego.

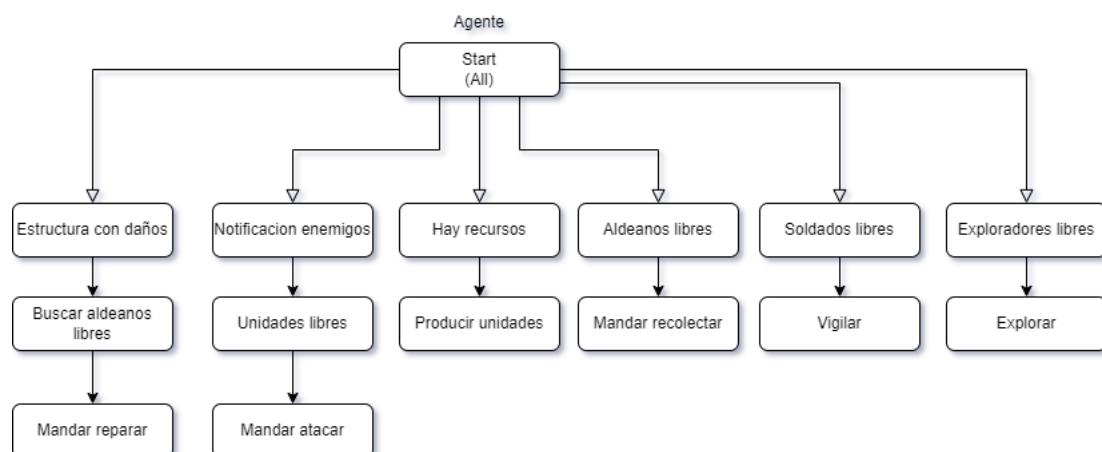


Figura 12: Árbol de comportamiento del agente

Los agentes se dedican a revisar el estado general de los poblados, por ello comprueban siempre si existen daños en sus estructuras para solicitar que las reparen los aldeanos. También controla todas las unidades, si están libres les asigna una función y si existe recursos genera nuevas unidades.

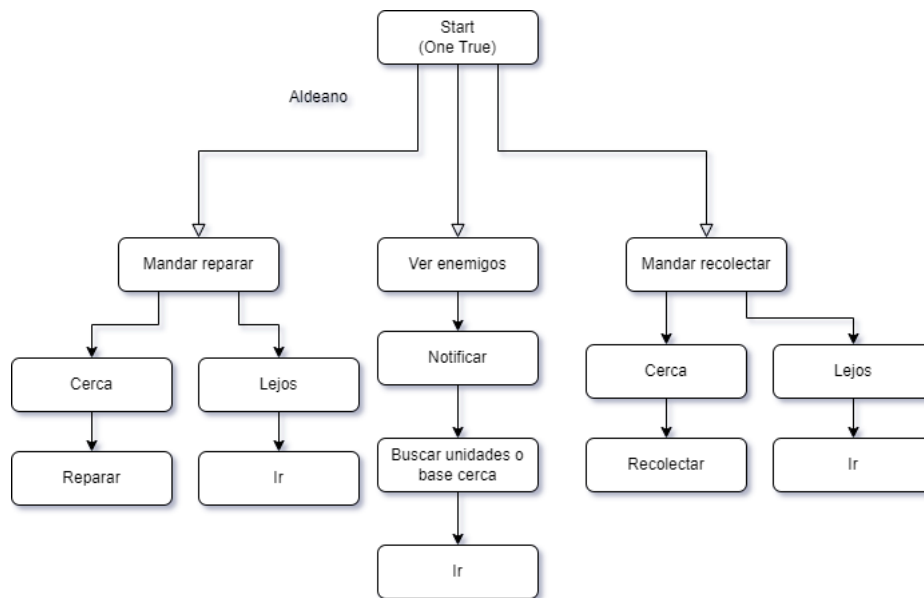


Figura 13: Árbol de comportamiento del aldeano

En los aldeanos vemos que existe un comportamiento peculiar que es notificar la presencia de enemigos y **huir** a la base humana más cercana, priorizando su vida a la recolección de recursos, pero dejando la posibilidad de que el gestor le mande reparar los edificios cercanos.

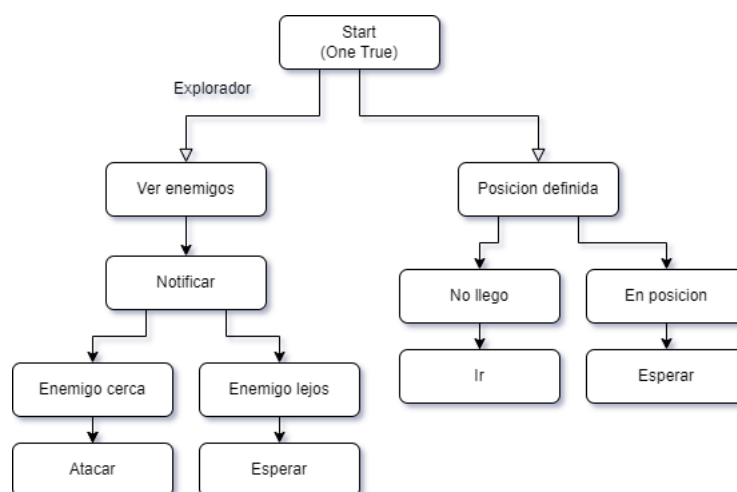


Figura 14: Árbol de comportamiento del explorador

En el caso de los exploradores, **priorizan** la aportación de información a la batalla, pero si el enemigo estuviera demasiado cerca, atacarían. A su vez, los exploradores van cambiando de posición, cuando pasa un tiempo determinado notifican al gestor de que su orden ha terminado y el gestor les asigna una nueva posición.

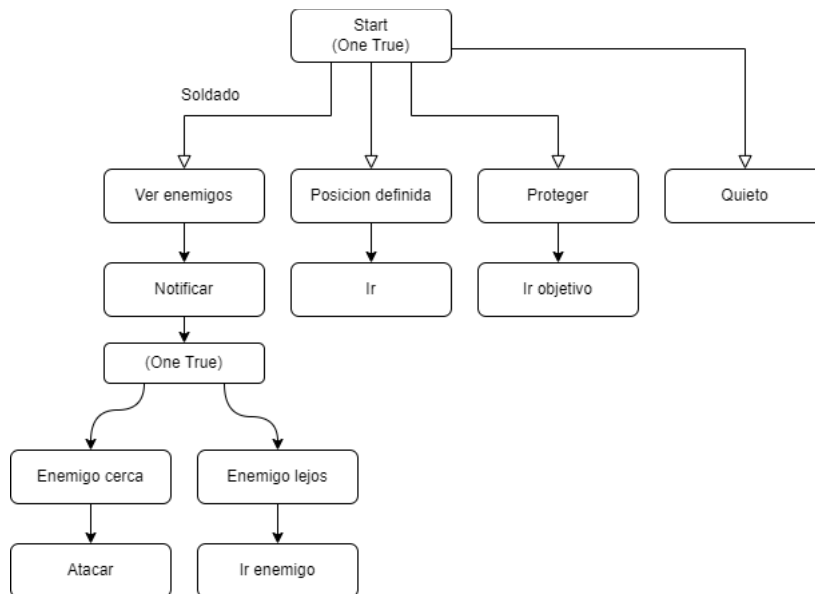


Figura 15: Árbol de comportamiento de los guerreros y arqueros

Por último, los guerreros y arqueros tienen la función de **defender** posiciones determinadas y revisar posibles notificaciones de enemigos, en el caso de que se les mande **proteger** una posición y pase un tiempo determinado notificarían al gestor e irían a proteger otra posición. Los guerreros no abandonarán a una batalla si no lo manda el gestor.

## 5. Implementación

### 5.1. Requisitos de instalación

Para la instalación en ordenadores de **forma directa**, desde la web se deberá descargar el archivo y situar en la zona de aplicaciones de cada sistema operativo compatible el juego, posteriormente es recomendado crear un acceso directo para su acceso rápido.

A futuro se pueden generar **asistentes de instalación** que faciliten la misma, dando opciones de donde realizar la instalación y donde situar el acceso directo. En el caso de que se instale desde alguna plataforma de terceros, se instalara de forma asistida **facilitando** esta tarea al usuario final.

### 5.2. Estructura del proyecto unity

La jerarquía de carpetas desde la carpeta **assets**.

#### Assets

- **Scenes**: contiene las escenas y *NavMesh* generados del terreno.
- **Resources**: almacena todos los elementos del juego.
- **Fonts**: fuentes que se utilizan en el juego.
- **Icons, Images, Sounds y Videos**: elementos multimedia que se incorporan al juego.
- **Misions**: archivos que definen las misiones.
- **Models**: modelos, animaciones y árboles de animaciones.
- **Prefabs**: contiene las instancias predefinidas del juego.
- **Profiles**: perfiles de máquinas de estados y árboles de decisión.
- **Terrain**: archivos de los terrenos generados con *Unity*.
- **Texture**: texturas y materiales para los terrenos y modelos.
- **Translations**: archivos que contiene las traducciones para el juego.
- **Code**: recopila todos los scripts del juego.
- **Managers**: componentes gestores del juego (sonido, nivel, héroe).
- **Scripts**: scripts genéricos para funcionalidades de componentes.
- **IAJson**: librería generada para dar toda la lógica de IA.
- **TranslateJson**: librería creada para la traducción de los textos en el juego.
- **TextMesh Pro**: librería para aumentar las posibilidades de los campos de texto del juego.
- **InputSystem**: guarda la asignación de controles.

### 5.3. Nivelado de características

En un juego de estrategia es muy importante que las características de los diferentes elementos tengan un **equilibrio** para que no haya una desventaja inicial o una superioridad, por ello se ha tenido en cuenta en los edificios que tengan los mismos puntos de vida en ambos bandos, además las unidades a diferencia del héroe también tienen la misma cantidad de vida.

Elemento	Vida	Ataque	Defensa
Héroe	800	15	30
Guerrero	100	9	20
Arquero	50	5	10
Aldeano	100	0	10
Explorador	50	7	10
Base	1500	0	0
Casa	250	0	0
Herrería	400	0	0
Torre	200	20	0
Barrera	500	0	0

Tabla 2: Tabla de estadísticas de las diferentes unidades.

Como se aprecia en la tabla, se ha escogido valores superiores para el héroe para que tenga más protagonismo y el usuario lo utilice como una **dinámica** de juego. También la torre supera en daño inicialmente al héroe, puesto que deben ser los primeros objetivos, por lo que generamos otra **dinámica** de atacar primero a las torres de defensa.



## 5.4. Implementación de la IA

Para este proyecto se ha generado una librería **IAJson** pensada para emplear en otros proyectos y expandirla de forma sencilla si fuera necesario. La librería contiene un controlador que es el componente que se agrega en la entidad a controlar, este componente se le puede indicar el **perfil** y el tipo de IA que será, así como el número de ejecuciones por segundo.

### Máquinas de estado

Las unidades controlables por el jugador se mueven por máquinas de estados, se ha implementado para que cada entidad pueda tener un estado en **ejecución** y otro en **espera** para cuando se cumpla la condición del estado principal.

De esta forma, como la mayoría de acciones suelen tener dos propósitos, es más fácil hacer que la unidad cumpla con ello. Por ejemplo, cuando se manda a atacar un edificio o unidad se le está mandando ir hacia el objetivo y atacarlo. Además, se agrega un **listado** de posibles estados y también se indica cuantas **ejecuciones por segundo** se deberá ejecutar dicha acción hasta que se complete, gracias a esto ahorramos recursos y nos permite cambiar de forma sencilla el **consumo** de la IA.

### Árboles de decisión

Este tipo de IA también lo hemos implementado con su control de ejecuciones, a diferencia del anterior se implementa como un **listado jerárquico** donde cada nodo puede cambiar el tipo de ejecución de sus nodos siguientes en el caso de que los ejecute.

Al ser de tipo jerárquico, cada nodo puede cambiar el tipo de ejecución de sus nodos siguientes, desde ejecutar todos, solo el primero que se cumpla o hasta que uno no se cumpla. Las posibilidades son ilimitadas por ello se ha escogido para la **IA enemiga**.

## Perfiles

Los perfiles en la librería cumplen un papel muy importante, determinan como funcionara la IA de la entidad y qué posibilidades tendrá, se configuran mediante **archivos .json** esto facilita mucho la personalización al poder reordenar de forma sencilla en el caso de los árboles como se ejecuta el mismo.

```
{
  "name": "Nombre del perfil",
  "description": "Descripcion",
  "type": "Tipo (arbol o estados)",
  "nodes": [
    {
      "id": "ID unico",
      "name": "Nombre del nodo",
      "nodes": [
        {
          "id": "ID unico",
          "name": "Nombre del nodo",
          "nodes": null
        }
      ]
    }
  ]
}
```

*Código 1: Plantilla básica de perfil para IA*

El formato anterior de código define de forma **sencilla** como definir los diferentes tipos de IA que tenemos en el juego.

## 5.5. Traducciones

Para las traducciones se ha creado una funcionalidad que nos permite traducir **fácilmente** los elementos de texto mediante archivos *JSON*, para ello se adjunta el script a los objetos de *Unity* que contengan un componente de texto.

```
{
  "texts": [
    {
      "id": "bMenu",
      "text": "Menú"
    },
    {
      "id": "bJugar",
      "text": "Jugar"
    }
  ]
}
```

*Código 2: Plantilla básica de traducción*

El contenido del texto actúa como un identificador que luego es buscado en el diccionario, se traduce el texto cuando se inicia o se llama a los métodos específicos del componente que produce el script, contiene propiedades **estáticas** que definen el **idioma** del que se obtienen los textos.

## 5.6. Calculo de daño

Para el cálculo de daño utilizamos principalmente dos estadísticas que es la **defensa** y el **ataque** el daño que se realiza se calcula por la siguiente fórmula:

$$DAÑO = ATAQUE - (DEFENSA / 10)$$

Esta sencilla fórmula nos permite **anular** el daño que no supere el cálculo de la defensa, en el caso de que no la supere la defensa **restara** algo de daño al ataque realizado. Por ejemplo, una unidad con 90 de defensa anularía totalmente el daño del guerrero con 9 puntos de daño, pero no anularía al héroe o torre de defensa.

## 5.7. Iluminación

Para la iluminación se han utilizado **puntos de luz (spot)** de **Unity**, ya que nos permitía iluminar zonas muy concretas. Unido a estos puntos de luz tenemos adjunto un script que nos permite saber si entra en su área, si es así aumenta **progresivamente** su iluminación hasta su máximo, si la zona se queda sin unidades aliadas se apaga de forma continua.

En este punto conviene usar las unidades exploradoras, ya que son las unidades más **veloces**. El siguiente minimapa se puede ver un ejemplo de cómo afecta la luz al entorno.

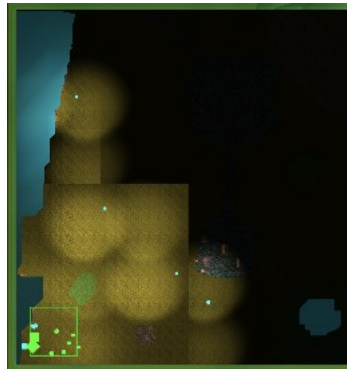


Figura 16: Ejemplo de iluminación variable

## 5.8. Sistema de misiones

Para las misiones se ha generado un componente que actúa de instancia **única** cuando se está en escena, este componente se ocupa de ir agregando las misiones **activas** mediante una propiedad de tipo **Action** que va apilando las misiones y reflejando sus cambios.

Por otra parte, para **empezar** una misión y **avanzar** en su progreso existen dos propiedades **Action** que se ocupan de esta finalidad, las misiones tienen varias condiciones en su definición que permite saber cuándo se debe avanzar al siguiente punto de la misión. Al **finalizar** la misión se puede recompensar con experiencia.

## 5.9. Controles

Para los controles se ha escogido implementar **Input System** de **Unity**, para ello se han definido dos grupos de operaciones, los genéricos que se efectúan durante todo el juego y los que aplican únicamente a la vista superior.

En los genéricos tenemos las pulsaciones en pantalla, el cambio de cámara, la rotación de construcciones y finalmente la obtención de la posición donde se sitúa la pulsación.

Por otro lado, la vista superior contiene las pulsaciones tanto **mantenidas** como simplemente la pulsación rápida, también se recogen las pulsaciones a los botones y teclas para mover la cámara en ambos ejes.

### 5.10. Guardado de partida

Para el guardado se ha creado un script que sirve para identificar los elementos a guardar y a la vez realizar el guardado. Al ser agregado en un elemento se guarda su posición y si existen determinados scripts guarda sus propiedades de forma **dinámica**, de esta forma nos sirve para todo tipo de elementos.

### 5.11. Ayuda visual

Se han implementado varios elementos visuales para **ayudar** al usuario a guiarse en el juego, entre algunas de ellas está en cambiar las interfaces a un color amarillento y los elementos del juego.



*Figura 17: Elemento cambiando de color para facilitar la detección por parte del jugador*

## 5.12. Minimapa

En el minimapa se ha jugado con las **alturas** de las cámaras y con materiales en los que no **afectaban** la iluminación, de esta forma podemos representar las unidades aliadas de forma sencilla en el minimapa.

Se generó un *render material* para la salida de la cámara y así representarlo en una imagen en la interfaz gráfica del usuario, este material es actualizado constantemente cuando la imagen realiza una render, así el mapa está constantemente **actualizado**.

## 5.13. Selector múltiple

Se ha implementado mediante *line renderer* y las pulsaciones del usuario, un sistema para seleccionar múltiples unidades en el juego. Para ello se detecta la primera pulsación sostenida del jugador y cuando suelta la pulsación se recoge la última posición, con estas dos posiciones se dibuja un recuadro entre ellas y finalmente se genera una colisión rectangular calculando el punto central.



Figura 18: Se muestra una selección múltiple de unidades

## 6. Demostración

### 6.1. Instrucciones de uso

El juego tiene un flujo bastante **simple**, cuando lo inicias lo primero que visualizas es la intro que pasara automáticamente al menú principal, en él podrás empezar a jugar, cargar una partida o ver los créditos. Los créditos disponen de un botón para volver al menú y al finalizar saltan automáticamente al menú.

#### Comenzar juego

Al comenzar el juego se visualiza la historia principal que le dan un **motivo** a la acción del juego. Al finalizar la historia se desbloquea la interfaz y empieza un pequeño **tutorial** para aprender a utilizar todos los elementos del juego, una vez finalizado se deja al usuario libre acción para que juegue esta primera fase.

### 6.2. Pruebas

Se ha llevado a cabo varias pruebas durante el desarrollo, sobre todo **integradas**, de todo el conjunto. En ellas se han detectado muchos errores, se van a detallar algunos de los más importantes:

- Fallos en el **flujo** de los árboles de decisión.
- Iluminación **insuficiente** en el escenario.
- Puntos en el escenario **inalcanzables** por los *NPCs*.

Por otra parte, también se han efectuado pruebas en varios **dispositivos** como tablets, móviles y ordenadores tanto portátiles como de escritorio con varias configuraciones gráficas, para ver que todos los elementos cumplen su función.

En última instancia se ha pasado algunas versiones del juego a varios familiares y amigos para efectuar unos pruebas y recibir mejoras, este punto quizás haya sido de los más **importantes** porque han sacado muchas faltas que desde la perspectiva del diseñador o desarrollador es difícil de prever, entre ellas se han podido implementar:

- Sonido **característico** de cada tipo de personaje cuando lo seleccionas.
- Mejorar la **selección** de personajes.
- Arreglo de colisiones e **interacciones** mientras las unidades estaban sin puntos de vida.

### 6.3. Guía de usuario

En este punto vamos a detallar el objetivo **principal** del juego y como lograrlo, también se describe que propósito tiene cada unidad y las opciones que pueden tener. Lo primero que se debe hacer es ampliar las construcciones y las unidades, por lo que se aconseja construir más viviendas para aumentar el número máximo.



Figura 19: Menú de construcción o artesanías

Para **generar** unidades se debe pulsar sobre las herrerías y se desplegará las disponibles. Las que están bloqueadas se pueden desbloquear desde el menú de habilidades **gastando** puntos de subida de nivel disponibles.

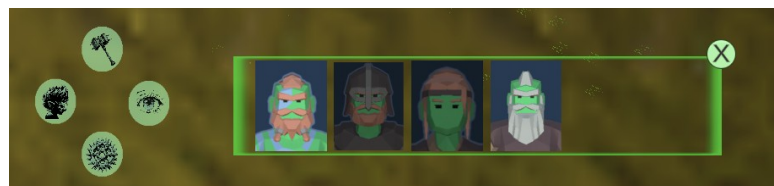


Figura 20: Menú de unidades al pulsar sobre herrería

Se aconseja **producir** aldeanos para empezar la recolección de recursos y aumentar la velocidad de producción de unidades de batalla y estructuras defensivas, ya que pasado un tiempo los exploradores empezaran a explorar cerca del punto inicial. Los recursos de recolección en esta primera versión son tres: maderas, minerales y comida.





Figura 21: Recursos

Una vez que se tengan suficientes unidades, se debe enviar las unidades a los diferentes poblados para vencer al enemigo.

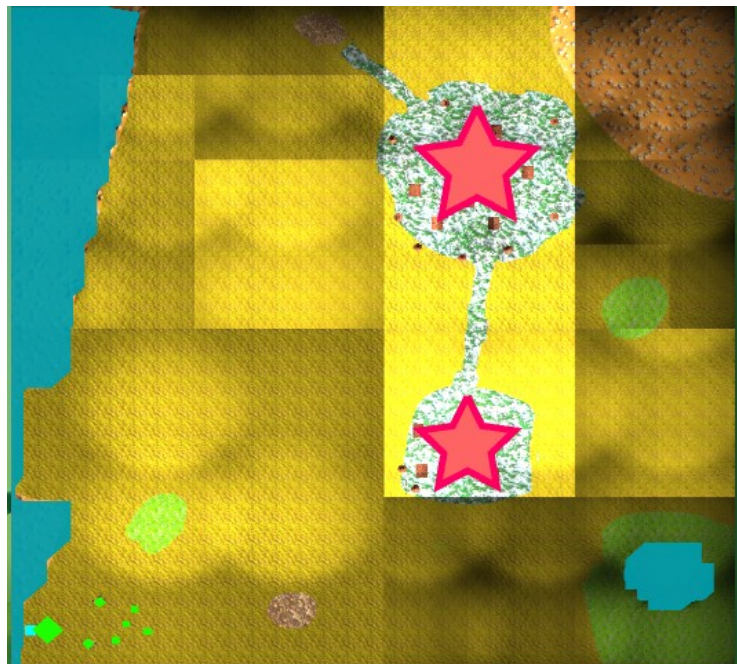


Figura 22: Mapa de objetivos señalados con una estrella

El juego **no tiene un punto final**, se irá **ampliando** por lo que se puede seguir jugando generando más recursos y ampliando el poblado lo máximo posible para así vencer con más facilidad la siguiente zona.

## 7. Conclusiones y líneas de futuro

### 7.1. Conclusiones

La primera conclusión sacada del proyecto es que al ser un juego de estrategia se debe agregar muchos elementos que alimenten el mundo del juego, puesto que al ser una vista superior, las zonas **vacías** son fácilmente apreciables.

Por otra parte, el proyecto al ser un juego en 3D, las implicaciones a la hora de **cómputo** y cálculo tienen un consumo muchísimo más alto que si fuera un juego 2D, por lo que hay que **cuidar** mucho cuantos cálculos se realizan, por ejemplo con las colisiones o con las luces dentro del juego.

Se han conseguido todos los objetivos propuestos, aunque al ser un proyecto tan grande, cada zona necesitaría mucho más tiempo de trabajo y más mano de obra en la realización del mismo para llegar a un punto óptimo. Además, han quedado partes a mejorar como el guardado de partida, también faltaría añadir más sistemas de partículas para darle un aspecto más completo.

La planificación se ha seguido en su **totalidad**, aunque se han incorporado objetivos nuevos conformes el tutor daba feedback y conforme se probaba y se analizaba el juego durante la experiencia.

Entre ella se han llevado a cabo cambios como dar más funcionalidad al minimapa, incluir opciones de **usabilidad** como botones de cierre en las interfaces o incluir elementos descriptivos en los menús con varias opciones.

## 7.2. Líneas de futuro

Para futuras mejoras las dividimos en dos tipos, extender el contenido o experiencia del usuario y, por otra parte, la mejora del contenido existente.

### Mejoras de contenido

- **Mejoras en los personajes y animaciones:** se puede incluir mejoras en los tiempos de animación y en los rasgos de los personajes.
- **Incluir mas complejidad en la IA enemiga:** en los perfiles se pueden agregar más **estados**, que agreguen mejoras en los **comportamientos** por ejemplo que varios aldeanos si se encuentran se pongan a hablar entre ellos de forma breve.
- **Añadir sonidos que tengan mejor armonia:** algunos sonidos les falta rasgos para que quede totalmente inmersivo.
- **Perfeccionar iluminación general:** la iluminación se puede mejorar abarcando zonas más concretas para que queden con una iluminación acorde.
- **Optimizaciones generales de código:** todo código se puede optimizar, bien porque se realicen muchas ejecuciones o porque tengan comportamientos poco eficientes.
- **Mejorar guardado de partida:** el guardado al ser el último punto le falta pequeñas mejoras para ser totalmente efectivo.

### Extender el contenido

- **Mas zonas jugables:** extender las zonas de juego para poder interactuar y gestionar varias zonas de forma simultánea y así aumentar la complejidad de forma escalada.
- **Incluir la adquisición de personalizaciones alternativas:** incluir tienda en la interfaz del juego con opciones de pago.
- **Agregar más efectos y sistemas de partículas:**
- **Añadir sistemas de trofeos:** muchos juegos modernos tiene una serie de trofeos que estén diseñados para acompañar la **experiencia** del mismo.
- **Agregar pausa y velocidades de juego:** la gran mayoría de juegos modernos de estrategia contienen opciones para modificar la velocidad del mismo o incluso pausarlo, sería una buena mecánica que generaría **dinámicas** en puntos estratégicos para pensar los siguientes movimientos.

## Bibliografía

1. Colaboradores de Wikipedia. Goblin Slayer [Internet]. Wikipedia, La enciclopedia libre. 2023 [citado el 7 de octubre de 2023]. Disponible en: [https://es.wikipedia.org/wiki/Goblin\\_Slayer](https://es.wikipedia.org/wiki/Goblin_Slayer)
2. Colaboradores de Wikipedia. Overlord (novela ligera) [Internet]. Wikipedia, La enciclopedia libre. 2023 [citado el 7 de octubre de 2023]. Disponible en: [https://es.wikipedia.org/wiki/Overlord\\_\(novela\\_ligera\)](https://es.wikipedia.org/wiki/Overlord_(novela_ligera))
3. Tolkien JRR. El Señor de los Anillos. Barcelona: Ediciones Minotauro; 1978.
4. Blizzard Entertainment. World of Warcraft [videojuego]. Irvine, CA: Blizzard Entertainment; 2004.
5. Codemasters. Overlord [videojuego]. Southam, Reino Unido: Codemasters; 2007.
6. Monolith Productions. La Tierra Media: Sombras de Mordor [videojuego]. Kirkland, WA: Warner Bros. Interactive Entertainment; 2014.
7. Glassdoor.es, Buscador de empleo y información laboral [Web]; [citado el 7 de octubre de 2023]. Disponible en: <http://www.glassdoor.es>
8. Statista. Industria estadounidense del videojuego [Web]. Statista. [citado el 15 de octubre de 2023]. Disponible en: <https://es.statista.com/temas/9665/industria-estadounidense-del-videojuego/>
9. García J. El estado de los videojuegos en Europa: un mercado de 21.600 millones de euros liderado por las consolas y el móvil, según ISFE [Web]. Xataka. 2020 [citado el 18 de octubre de 2023]. Disponible en: <https://www.xataka.com/videojuegos/estado-videojuegos-europa-mercado-21-600-millones-euros-liderado-consolas-movil-isfe>
10. Cabello M. Análisis de Warcraft 3 (PC) [Web]. Vandal. 2002 [actualizado el 17 agosto 2020; citado el 19 de octubre de 2023]. Disponible en: <https://vandal.elespanol.com/analisis/pc/warcraft-3/960#p-13>
11. Amechazurra Falagán M. Análisis de Total War Warhammer II, el juego de estrategia para PC [Web]. Hobby Consolas. 2017 [citado el 23 de octubre de 2023]. Disponible en: <https://www.hobbyconsolas.com/reviews/analisis-total-war-warhammer-ii-juego-estrategia-pc-166530>

12. Ensemble Studios. Age of Mythology [videojuego]. Redmond, WA: Microsoft Game Studios; 2002.
13. Ensemble Studios. Age of Empires [videojuego]. Redmond, WA: Microsoft Game Studios; 1997.
14. Phenomic Game Development. SpellForce: The Order of Dawn [videojuego]. Alemania: JoWood Productions; 2003.
15. PopTop Software. Trópico [videojuego]. Baltimore, MD: Gathering of Developers; 2001.
16. Steam. Valve Corporation [plataforma de distribución digital]. Bellevue, WA: Valve Corporation. Disponible en: <http://store.steampowered.com/>
17. Epic Games Store. Epic Games, Inc. [plataforma de distribución digital]. Cary, NC: Epic Games, Inc. Disponible en: <https://www.epicgames.com/>
18. Epic Games. Fortnite [videojuego]. Cary, NC: Epic Games; 2017.
19. Riot Games. League of Legends [videojuego]. Los Ángeles, CA: Riot Games; 2009.
20. Hi-Rez Studios. Paladins [videojuego]. Alpharetta, GA: Hi-Rez Studios; 2018.
21. Blizzard Entertainment. Overwatch [videojuego]. Irvine, CA: Blizzard Entertainment; 2016.
22. Riot Games. Valorant [videojuego]. Los Ángeles, CA: Riot Games; 2020.
23. King. Candy Crush Saga [videojuego]. Estocolmo, Suecia: King; 2012.
24. Blizzard Entertainment. Hearthstone [videojuego]. Irvine, CA: Blizzard Entertainment; 2014.
25. Supercell. Clash of Clans [videojuego]. Helsinki, Finlandia: Supercell; 2012.
26. Unity Technologies. Unity [software de motor gráfico]. San Francisco, CA: Unity Technologies; 2005.
27. Mixamo. Mixamo [servicio web de animaciones]. San Francisco, CA: Mixamo; 2010.
28. Blender Foundation. Blender [software de modelado 3D]. Ámsterdam, Países Bajos: Blender Foundation; 1998.
29. Spencer Kimball, Peter Mattis. GIMP (GNU Image Manipulation Program) [software de edición de imágenes]. Berkeley, CA: University of California; 1996.
30. Synty Store. Synty Store [tienda online de activos para juegos]. Disponible en: <https://www.syntystore.com/>
31. OpenGameArt. OpenGameArt [plataforma de arte de código abierto]. Disponible en: <https://opengameart.org/>
32. Leartes Studios. Leartes Studios [tienda online en Gumroad]. Disponible en: <https://leartesstudios.gumroad.com/>

33. Ovni Sound. Ovni Sound [plataforma de efectos de sonido]. Disponible en:  
<https://ovnisound.com/>

# Anexos

## Anexo A: Glosario

- **RTS (Real-Time Strategy):** Un género de videojuegos donde la estrategia se juega en tiempo real, sin turnos.
- **Unity:** Motor gráfico utilizado en el desarrollo de videojuegos.
- **Assets:** Recursos gráficos, sonoros y otros elementos usados en la creación de videojuegos.
- **IA (Inteligencia Artificial):** Tecnología empleada para simular comportamientos reales en videojuegos.
- **Git:** Sistema de control de versiones empleado para la gestión del código fuente.
- **GUI (Graphical User Interface):** Interfaz gráfica del usuario en el juego.
- **JSON (JavaScript Object Notation):** Formato de intercambio de datos y de fácil lectura.
- **Blender:** Herramienta de modelado 3D utilizada en el desarrollo del juego.
- **Mixamo:** Plataforma para la generación de animaciones de forma fácil.
- **Low Poly:** Estilo gráfico que usa un número reducido de polígonos.
- **Spotlighting:** Técnica de iluminación empleada en el juego, donde se ilumina una zona determinada.
- **NavMesh:** Herramienta usada para el pathfinding y la navegación de IA en entornos 3D y 2D.
- **Whitebox:** Fase de desarrollo temprano donde se prueban conceptos básicos del juego.
- **API (Application Programming Interface):** Conjunto de protocolos y operaciones para construir aplicaciones.
- **Debugging:** Proceso de identificar y corregir errores en el código programado.
- **Gameplay:** Manera en que los jugadores interactúan con un juego.
- **Level Design:** El proceso de crear niveles, misiones y desafíos en un juego.
- **NPC (Non-Player Character):** Personajes controlados por la IA en un videojuego.
- **Rendering:** Proceso de generar una imagen a partir de un modelo 3D o 2D.

## Anexo B: Entregables del proyecto

1. **Memoria del trabajo:** documento que detalla todo el ciclo de vida del trabajo final de master.
2. **Video gameplay (gameplay.mp4):** se muestra una partida guiada, donde se muestran todas las mecánicas implementadas y el nivel.
3. **Video promocional (trailer.mp4):** pequeño video con forma de trailer, que realiza una primera puesta en escena para atraer posibles jugadores.
4. **Ejecutables:** Se adjuntan tres ejecutables en distintos archivos comprimidos de diferentes plataformas, los sistemas son Android (se necesitan permisos de desarrollador), Windows (64 bits) y MacOs (tanto procesadores Intel como Silicon).
5. **Documento de cesión de derechos:** documento donde se acepta la retransmisión y promoción del trabajo en cualquier medio.

## Anexo D: Currículum Vitae

Soy un apasionado de la informática con énfasis en la programación y diseño de videojuegos. Actualmente, soy Programador Backend con Java en The Cocktail, destacando en mi trayectoria profesional por mi trabajo en NTT Data.

Cuento con una sólida formación académica, incluyendo una Ingeniería Multimedia con especialización en videojuegos en la Universidad de Alicante y habiendo cursado el Máster en Diseño y Programación de Videojuegos en la Universitat Oberta de Catalunya.

Mi **entusiasmo** por los videojuegos se refleja en mis proyectos personales, visibles en la página web **[www.czycats.com](http://www.czycats.com)**.