

# Alvheim

**Lorena Arriaga Mateo**

Máster de creación y desarrollo de videojuegos

**Javier Luis Cánovas Izquierdo**

**Jordi Duch Gavalrà**

**Helio Tejedor Navarro**

16/06/2019



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-

SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

## **B) GNU Free Documentation License (GNU FDL)**

Copyright © 2019 LORENA

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

## **C) Copyright**

© Lorena

Reservados todos los derechos. Está prohibido la reproducción total o parcial de esta obra por cualquier medio o procedimiento, comprendidos la impresión, la reprografía, el microfilme, el tratamiento informático o cualquier otro sistema, así como la distribución de ejemplares mediante alquiler y préstamo, sin la autorización escrita del autor o de los límites que autorice la Ley de Propiedad Intelectual.

## FICHA DEL TRABAJO FINAL

<b>Título del trabajo:</b>	<i>Alvheim</i>
<b>Nombre del autor:</b>	<i>Lorena Arriaga Mateo</i>
<b>Nombre del consultor/a:</b>	<i>Helio Tejedor Navarro</i> <i>Jordi Duch Gavaldà</i>
<b>Nombre del PRA:</b>	<i>Helio Tejedor Navarro</i>
<b>Fecha de entrega (mm/aaaa):</b>	06/2019
<b>Titulación:</b>	<i>Máster de diseño y desarrollo de videojuegos</i>
<b>Área del Trabajo Final:</b>	<i>Trabajo Final de Máster</i>
<b>Idioma del trabajo:</b>	<i>Castellano</i>
<b>Palabras clave</b>	<i>tfm, 2d-rogue, uoc, unity</i>
<p><b>Resumen del Trabajo (máximo 250 palabras):</b> <i>Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.</i></p>	
<p>Se ha creado un videojuego como trabajo final de máster de creación y desarrollo de videojuegos para aplicar y hacer referencia a todos los conocimientos aprendidos durante éste. Es un videojuego <i>RPG</i> creado con el motor Unity en 2D haciendo uso de <i>Assets</i> públicos, algunos de ellos adaptados al juego. Tiene referencias a juegos muy populares en su época, incluso de culto, ayudando a apoyar el auge de los videojuegos y popularizarlos, hasta crear fans de sagas y siendo juegos recordados hoy en día.</p> <p>Se utiliza una metodología de tipo cascada, donde se crea la idea, se analizan sus requisitos y se desglosan la idea hasta tener diferentes componentes implementables. Se sigue diseñando la parte gráfica, implementando el código y, por último, la corrección de errores que pueda haber.</p>	

Los resultados han sido la creación de un videojuego propio desde cero con los objetivos básicos establecidos e implementados que cumplen con las reglas de diseño de niveles y haciendo uso de distintos conocimientos aprendidos (analizar los elementos más importantes que deben ser implementados, desglosar una idea en términos de desarrollo de videojuegos, implementación de inteligencia artificial en los enemigos con máquinas de estados, aplicación de reglas de diseño y usabilidad, etc.).

Las conclusiones extraídas han sido la importancia en tener unos objetivos iniciales claros para poder desglosar correctamente una idea en videojuegos y ser capaz de implementarla. Además, la planificación es una parte fundamental en el desarrollo de un proyecto, en el cual se debe tener claro qué se quiere y, sobretodo, qué se puede hacer durante un determinado periodo de tiempo para priorizar e implementar lo estrictamente necesario para, en un futuro, desarrollar las líneas de futuro.

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

A videogame has been created as a master's degree in videogame creation and development in order to apply and refer to all the knowledge learnt during this one. It's an RPG video game created with the Unity engine using public Assets, some of them adapted to the game. It has references to very popular games in its time, even cult games, helping to support the rise of video games and popularize them, to create fans of sagas and being games remembered today.

A cascade type methodology is used, where the idea is created, its requirements are analyzed and the idea is broken down into different implementable components. We continue designing the graphic part, implementing the code and, finally, correcting any errors.

The results have been the creation of an own videogame from scratch with the established and implemented basic goals that comply with the design rules of levels and making use of different learned knowledge (to analyze the most important elements that must be implemented, to break down an idea in terms of videogame development, implementation of artificial intelligence in enemies with state machines, application of design and usability rules, etc. ).

The conclusions drawn have been the importance of having clear initial

objectives in order to be able to correctly break down an idea into videogames and be able to implement it. In addition, planning is a fundamental part in the development of a project, in which it must be clear what is wanted and, above all, what can be done during a certain period of time to prioritize and implement what is strictly necessary for, in the future, implement future lines.

# Índice

## Contenido

<b>Alvheim</b> .....	1
Índice.....	iv
1. Introducción.....	1
1.1 Contexto y justificación del Trabajo.....	1
1.2 Objetivos del Trabajo.....	1
1.3 Enfoque y método seguido.....	2
1.4 Planificación del Trabajo .....	2
1.5 Breve resumen de productos obtenidos .....	3
1.6 Breve descripción de los otros capítulos de la memoria.....	3
2. Estado del arte .....	5
3. Planificación .....	8
3.1. Diagrama de Grant escueto .....	9
3.2. Diagrama de Grant extendido .....	10
3.3. Hoja de recursos .....	11
3.4. Coste del proyecto .....	12
4. Definición del juego y Game Design .....	13
4.1. Idea principal y género del juego.....	13
4.2. Conceptualización .....	14
4.3. Personajes .....	14
4.4. Interacciones entre actores .....	15
4.5. Objetivos del jugador.....	15
4.6. Referencias .....	15
4.7. Target.....	18
4.8. Sensaciones a transmitir .....	18
4.9. Niveles de dificultad.....	18
4.10. Recompensas .....	18
5. Diseño técnico.....	19
5.1. Unity3D.....	19
5.2. Unreal Engine.....	19

5.3. Decisión.....	20
5.4 Assets y recursos del juego.....	20
5.5. Componentes del juego.....	21
5.7. Funcionamiento IA de los enemigos .....	22
6. Diseño de niveles (Level Design) .....	23
6.1. Mapa: primera aproximación .....	23
6.2. Mapa final implementado .....	24
7. Usabilidad.....	29
8. Manual de usuario .....	33
8.1. Requerimientos .....	33
8.2. Instrucciones del juego.....	33
9. Conclusiones.....	34
10. Glosario .....	36
11. Bibliografía .....	37

## Lista de figuras

Ilustración 1: Esbozo del mapa	23
Ilustración 2: Granja	24
Ilustración 3: Hogar	25
Ilustración 4: Ciudad	25
Ilustración 5: Nivel de bosque	26
Ilustración 6: Nivel de montaña	26
Ilustración 7: Nivel del castillo	27
Ilustración 8: Mapa completo	27
Ilustración 9: esbozo primera interfaz	30
Ilustración 10: interfaz con el inventario	31
Ilustración 11: interfaz con el inventario de compraventa	31

# 1. Introducción

## 1.1 Contexto y justificación del Trabajo

Este proyecto abarca la creación de una idea para un videojuego pasando por el proceso de *Game Design*, donde desarrollaremos la idea principal, obtendremos un guion a seguir especificando el *target*, mecánicas y dinámicas (en relación a la inteligencia artificial), especificando los personajes y qué se pretende transmitir al jugador. En adición a este proceso también está el de diseñar una buena interfaz de usuario basada en las reglas de usabilidad y accesibilidad creadas por *Jakob Nielsen*, analizando las diferentes opciones y escogiendo las más idóneas y crear un nivel acorde.

En este proyecto también se incluye un estudio del estado del mercado y de la historia del género RPG, analizando las tecnologías y herramientas utilizadas en el proyecto y la organización que se ha seguido.

Por último, se incluye las conclusiones extraídas de este proyecto, junto a una lista de líneas futuras con las cuales continuar este proyecto.

## 1.2 Objetivos del Trabajo

Generales:

- Planificar y estructurar un proyecto de creación de videojuegos
- Analizar el mercado y la historia del género RPG
- Desarrollar la conceptualización para un videojuego
- Conocer las diferentes técnicas de usabilidad y diseño de videojuegos
- Analizar los elementos, tanto en herramientas como en organización, más importantes a la hora de crear un videojuego

Específicos:

- Aplicar y ampliar los conocimientos adquiridos durante el máster de creación y diseño de videojuegos
- Poner en práctica los principios de un buen diseño de nivel y de juego

- Crear un videojuego completo de principio a fin
- Aplicar al menos tres mecánicas de juego
- Utilizar por lo menos tres patrones distintos de inteligencia artificial para los enemigos
- Analizar el resultado y establecer las posibles líneas de futuro que puede adquirir el proyecto

### **1.3 Enfoque y método seguido**

Para crear el videojuego se ha optado por adaptar un producto ya existente en el cual hacer referencia, para así poder demostrar el desarrollo de las diferentes mecánicas básicas y extras del juego. Con ello, es más fácil implementar nuevas mecánicas y nuevos objetivos al juego.

### **1.4 Planificación del Trabajo**

Para poder realizar el trabajo se han utilizado diferentes herramientas:

- Unity3D, motor sobre el que se desarrolla el trabajo
- *VisualStudio* 2010, herramienta donde se programan las mecánicas y comportamientos en C#
- GitHub, repositorio online donde crear un *backup* de los avances y que sea accesible al público

Los hitos que se han seguido para las distintas PEC son:

- PEC1: Diseño del videojuego. Los objetivos son establecer las bases del videojuego (historia, tipo, plataforma de destino, personajes, referencias, nombre, idea principal del juego y objetivos)
- PEC2: Versión parcial. El objetivo principal para esta versión parcial es plasmar la estética del juego y establecer animaciones y mecánicas básicas del protagonista y del primer enemigo. También se crean distintos mapas jugables que, más tarde, serán jugables.

- PEC3: Versión jugable. Es necesario implementar una versión jugable, casi definitiva del juego, con todas las mecánicas que se quieren mostrar y casi sin fallos, pero apta para poder seguir siendo modificable.
- PEC4: Versión final. Se presenta el juego sin errores, finalizado al 100% y la memoria acorde.

### **1.5 Breve resumen de productos obtenidos**

El producto obtenido ha sido un videojuego creado desde 0 a partir de los distintos *Assets* libres y gratuitos de la página *OpenGameArt.org*, implementando distintas mecánicas por parte del personaje principal como en los enemigos (detalladas en precisión en los siguientes capítulos) que son funcionales; además, con las diferentes guías de usabilidad, diseño y creación de niveles, se ha creado una interfaz lo más sencilla y jugable posible para el jugador.

Gracias a la creación del producto se ha podido, además, añadirlo como *portfolio* y al repositorio propio y personal de *UnityConnect*.

### **1.6 Breve descripción de los otros capítulos de la memoria**

Los siguientes capítulos que componen la memoria son:

- Capítulo 2: Estado del arte. Explicación de la situación actual y contexto sobre el género RPG del cual se basa el videojuego creado, analizando las plataformas de desarrollo y características básicas de éstas.
- Capítulo 3: Planificación. Explicación a través del diagrama de Grant de la planificación y organización del proyecto, además de una hoja de recursos y costes del proyecto.
- Capítulo 4: Definición del juego. Desarrollo de las características básicas del videojuego, con las distintas ideas y concepciones de éste.
- Capítulo 5: Diseño técnico. En este capítulo se entrará en detalle del entorno utilizado para el desarrollo del juego y los requerimientos del propio, qué herramientas se han empleado, qué *assets* se han utilizado

y su origen, explicación básica de la inteligencia artificial (IA) de los enemigos y, finalmente, la arquitectura del juego.

- Capítulo 6: Diseño de niveles. Muestra del mapa creado, qué objetivos tienen cada uno de los niveles que se han diseñado y qué criterios se han utilizado para la elección de ese diseño.
- Capítulo 7: Usabilidad. Análisis de la interfaz creada con diferentes usuarios, sacando conclusiones y propuestas para mejorarla y crear la interfaz final.
- Capítulo 8: Manual de usuario. Explicación de los requerimientos básicos (en términos de *hardware*) para jugar e instrucciones del juego.
- Capítulo 9: Conclusiones del juego.

## 2. Estado del arte

Los juegos de género RPG empezaron a mediados de los años 70 inspirado en juegos de rol de mesa como el famoso *Dungeon & Dragons*. La época dorada de este género se produjo entre mediados y finales de los años 80 hasta mediados de los 90, donde sufrió un declive a causa de los incrementos en los costes de desarrollo y el continuo cambio de hardware, con los cuales no podían ir a la par. A partir del año 2000 tuvo lugar un resurgimiento gracias a los nuevos motores de videojuegos en 3D y las mejoras en la calidad de gráficos.

Algunos videojuegos de este género famosos cronológicamente por sus aportaciones e innovaciones son *Ultima* (sobre todo *Ultima III: Exodus* (1983)) donde se mezclaban diferentes elementos fantasía y ciencia ficción con gráficos divididos. La serie *Might and Magic* (1986) proporcionaba un gran mundo en el cual jugar, con muchos tipos de armas y hechizos e introdujo la particularidad de que una raza y género importe para la jugabilidad, adoptando diferentes particularidades (más vida, habilidades especiales, etc.). Otro juego que influyó al género RPG es *Dungeon Master* (1987), el cual introdujo diferentes innovaciones a nivel de interfaz de usuario, como el tiempo real, la vista en primera persona o la manipulación directa de los objetos y entorno utilizando el *mouse*. Además, añadió un sistema de runas para hechizos, las cuales podían ser combinadas entre sí para crear diferentes hechizos mágicos (el jugador debía basarse en la prueba y error para descubrir las combinaciones). Por último, gracias a aportar un juego de mundo abierto y jugabilidad no lineal -del cual la serie *The Elder Scrolls* citó como influencia- fue *Legends of Valour* (1992).

En 1996 salió a la luz la mítica serie de *Diablo*, justo en medio del estancamiento del género en la época. Tiene un enfoque de juego de mazmorras y avanza por niveles, además de ser en tiempo real, aportar unos gráficos de alta calidad y la posibilidad de jugarlo de manera colaborativa con una LAN (red de área local). A partir de la época de resurgimiento, muchos juegos han marcado un antes y un después en la vida de la gente que jugaba a videojuegos, introduciendo y reutilizando diferentes mecánicas, añadiendo nuevos sistemas, desarrollando

diferentes opciones que hacen del personaje principal no solamente como un simple personaje, sino como el protagonista del cual el jugador se llega a sentir identificado gracias a las múltiples opciones de personalización (habilidades, personalizar al personaje, elecciones de respuestas a diálogos cambiando así el rumbo de la historia, etc.). Todo ello produce una sensación y una experiencia única que se adapta a cada jugador.

Algunos de los videojuegos más icónicos a partir de los años 2000 son *Dragon Quest*, *Final Fantasy*, *Diablo*, *Mass Effect*, *The Elder Scrolls*, *Dragon Age*, *Fallout*, *World of Warcraft*, *The Witcher*, *Harvest Moon*, *Moonlighter* y muchos más; además, otros géneros tienen toques o partes las cuales se basan también en este género, como *The Legend Of Zelda*.

Como se puede observar, este género no está en decadencia a pesar del número de años que lleva existiendo, sigue siendo muy popular y uno de los géneros que más agrada a sus jugadores. Su evolución se ha basado en introducir distintos elementos personalizables para el jugador y combinaciones con otros géneros o subgéneros, creando un árbol con muchísimas ramas de entre las cuales se basan en esta raíz, pero cabe destacar dos de ellas:

- JRPG: juegos de rol creados en Japón, con gráficos más brillantes y parecidos al anime con personajes relativamente jóvenes con historiales lineales. En muchas ocasiones están dirigidos a un público joven y adolescente en la que se utilizan las batallas por turnos.
- WRPG: juegos de rol creados en Europa con gráficos más oscuros y personajes mayores. Se centran en el realismo y en dar libertad al jugador con mecánicas que permitan al jugador tomar decisiones sobre el rumbo de su personaje. El *target* de estos juegos es un público adolescente y adulto, mayormente hombres.

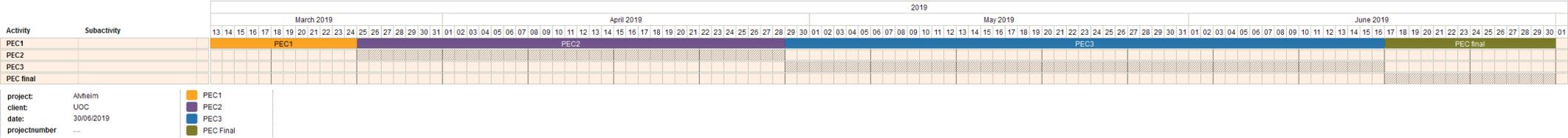
Las plataformas de desarrollo que se suelen usar para hacer estos juegos suelen ser motores propios de las compañías que los crean y, para los indies, Unity o UnrealEngine. Ambos motores son muy potentes, pero Unity se utiliza más para la creación de juegos 2D y con UnrealEngine para 3D, pues, como se analiza más adelante en el capítulo 4, cada uno de ellos se especializa en un ámbito.

Cada motor tiene sus pros y sus contras que se deben analizar antes de desarrollar el proyecto (por ejemplo, *Unity* es gratuito hasta que se llegan a 3000€ de ganancias, tiene una gran comunidad y tutoriales oficiales, pero *UnrealEngine* es gratuito, pero no posee la misma comunidad y tutoriales como *Unity*).

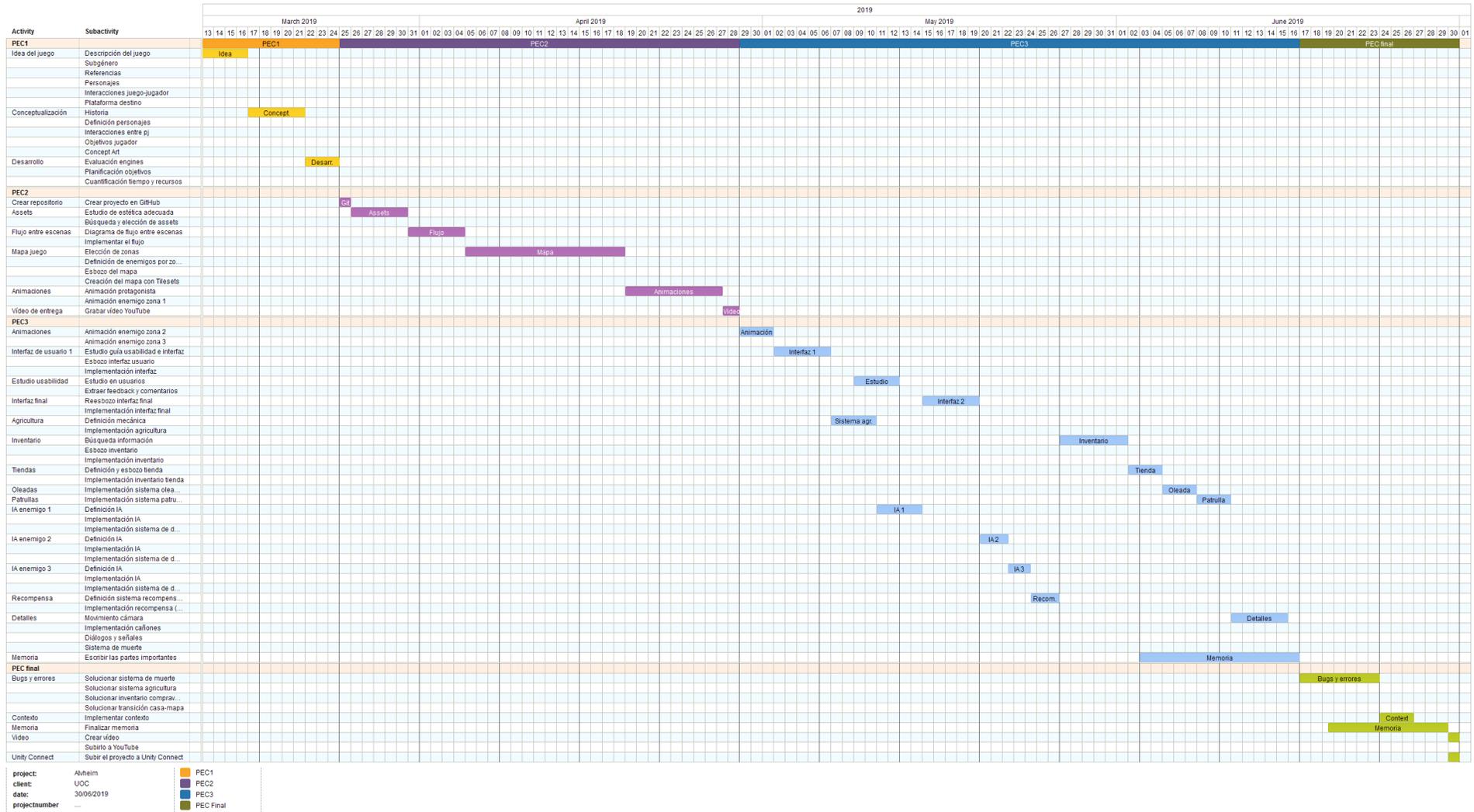
### 3. Planificación

Para la planificación de este proyecto se ha usado la herramienta Diagrama de Grant, en el cual se pueden establecer grupos de trabajo, las tareas y subtareas de cada grupo además del tiempo establecido por cada tarea. También incluye un calendario en el cual se puede ver la organización completa por fecha.

### 3.1. Diagrama de Grant escueto



### 3.2. Diagrama de Grant extendido



### 3.3. Hoja de recursos

Nombre del recurso	Tipo	Tasa estándar	Carga de trabajo	Horas	Coste
Ordenador	Material	250 €		-	250 €
Estudios	Trabajo	15,00€ / hora	20%	140	2.100 €
Planificación proyecto	Trabajo	10,00€ / hora	15%	100	1.000 €
Implementación proyecto	Trabajo	30,00€ / hora	65%	250	7.500 €
					10.850 €

### **3.4. Coste del proyecto**

Para calcular el coste del proyecto he aplicado los parámetros como si fuese un proyecto real.

El gasto principal está en la implementación del proyecto, ya que habría diferentes trabajadores los cuales un salario proporcionado sería el de 30€/h, ya que recibiría unos 15€/h en neto y el resto serían para impuestos. También se contabiliza la planificación hecha al principio del proyecto, donde se establecen los límites de tiempo y las tareas a realizar, la cual un precio justo sería de 10€/h. Finalmente, los estudios y el aprendizaje para aplicar al proyecto he utilizado el salario de 15€/h.

El material también es un recurso a tener en cuenta, en este caso, el ordenador. El coste de éste es de 1500€ aproximadamente, lo cual si decidimos amortizarlo en 3 años nos deja una amortización de 500€/año, y como este proyecto tiene una cabida de medio año, la amortización final del material es de 250€.

En cuanto al software en este caso se ha utilizado el programa Unity3D versión personal, la cual es una versión gratuita de Unity donde la financiación de la empresa no excede los 100.000\$ al año. Como no se ha tenido que usar ninguna herramienta de Unity Pro y la financiación no supera el límite establecido por Unity, se ha optado por esta versión del programa.

El conjunto de costes nos da un total de 10.850€; puede ser considerado un presupuesto elevado, pero conlleva muchas horas de aprendizaje, estudio e implementación, además de solución de errores. En proyectos futuros no se contabilizarán tantas horas en aprendizaje, así que el coste del proyecto y la cantidad de horas invertidas disminuye considerablemente.

## 4. Definición del juego y Game Design

### **4.1. Idea principal y género del juego**

El género del juego es de acción 2D tipo RPG de mundo abierto a tiempo real en el cual el jugador tratará de avanzar en su aventura con tal de derrotar a todos los enemigos que acechan al pueblo con tal de devolver a la villa la vida que tenía antes de que aparecieran los enemigos; El personaje principal tiene una cantidad de vida y energía limitada, los cuales irán menguando, respectivamente, al recibir ataques y efectuar un ataque. El jugador será provisto de un inventario inicial con distintos elementos que le ayudarán a progresar en el juego, como alimentos para recuperar energía, semillas de hortalizas para cultivarlas y recoger más alimentos y pociones de vida.

La seguridad del pueblo irá incrementando a medida que el jugador vaya superando los diferentes retos que se le plantean y los negocios volverán a instalarse en el pueblo. Con la llegada de estos negocios, el jugador podrá proveerse de ítems que serán de gran ayuda para su avance, ya sea comprando provisiones para recuperar energía como salud.

El jugador va creando su propio rol, tomando las decisiones que cree oportunas para poder seguir avanzando en el juego y derrotar a los enemigos para que todo vuelva a la normalidad.

Las mecánicas de los combates contra los enemigos están basadas en dos tipos:

- **Oleadas:** en cuanto el jugador pise la “trampa” preparada antes de pasar a la siguiente zona, se activan oleadas donde los enemigos de esa zona van apareciendo y acumulándose. El jugador no podrá volver atrás ni seguir avanzando hasta que todos los enemigos de la zona hayan desaparecido.
- **Patrullas:** los enemigos tienen como patrón un camino a seguir y, si detectan que el jugador está dentro de su rango de patrulla, persiguen al jugador hasta llegar al rango mínimo de ataque.

## **4.2. Conceptualización**

La historia consiste en Astrid, un chico que vive en la pacífica villa de Alvheim. La villa vive del cultivo agrícola e intercambio de bienes entre los distintos aldeanos. Aun siendo una villa pequeña, está dotada de bastantes comercios muy diferentes lo cual hace que no se eche de menos ningún establecimiento propio de las grandes ciudades. Astrid tiene su propia tiendecita, donde vende sus verduras y hortalizas que cuida, año tras año, con esmero y paciencia.

Sin embargo, el mundo ha sufrido el ataque de los *Thall*, unos monstruos del subsuelo que se dedican a destruir, robar y matar a la gente para quedarse con sus ciudades con tal de convertir a los humanos en sus esclavos. La gente, atemorizada, huye de sus casas con tal de encontrar un refugio o algún lugar donde poder sentirse seguros. Lamentablemente, muchos de ellos no consiguen encontrar un sitio seguro y vagan por bosques, intentando buscar algo de comida a lo que llevarse a la boca para poder sobrevivir. La llegada de los *Thall* atrae a más tipos de monstruos, como los *Korhien* (árboles podridos que atacan a cualquiera) y los *Qhetzall* (esqueletos de enemigos pasados que han revivido). Astrid, cansada de tanto huir, decide armarse de valor y aprender a luchar para recuperar la villa en la que pasó toda su vida. Para lograrlo, volvió a su casa de su querida villa con la intención de reconstruirla, ahuyentar a los monstruos y lograr que la gente volviera a habitar su hogar.

La primera vez que se vuelve a la villa, las tiendas ya no existen, solo queda su granja. A medida que el jugador vaya avanzando y derrotando a los primeros enemigos, restaurará las primeras tiendas y el nivel de seguridad de la ciudad aumentará. Gracias al carpintero, se reconstruirán los lugares y negocios en el orden que el jugador decida y, mientras el carpintero trabaja y el jugador se adentra en las minas para ir limpiándolas de los monstruos, más gente llegará.

## **4.3. Personajes**

**Astrid**, un joven agricultor, aprende el noble arte de la lucha con tal de defender la villa de *Alfheim* y devolver la paz en tiempos oscuros. Es un héroe que no destaca con ninguna habilidad especial, empático y movido por su lucha y su objetivo de recuperar lo que antaño tuvo.

#### **4.4. Interacciones entre actores**

Las interacciones que se producen en esta fase del videojuego son:

- Interacción protagonista – enemigos: no hay ningún diálogo entre el protagonista o los enemigos, pero su interacción se basa en la lucha de ambos bandos por defender lo que consideran correcto.
- Interacción protagonista – jugador: existen diferentes diálogos que componen una relación entre el protagonista y el jugador; el protagonista explica al jugador su historia y cuál es su objetivo principal, además de proporcionarle un lugar en el cual obtener información extra para continuar la aventura. El protagonista busca la empatía del jugador para que éste tome la iniciativa y le ayude, mediante la toma de decisiones, a conseguir lo que el protagonista anhela.

#### **4.5. Objetivos del jugador**

Los objetivos que se plantean al jugador son los siguientes:

1. Conseguir que la gente vuelva al pueblo hasta llegar al 100% de los habitantes que tuvieron que huir
2. Destruir y erradicar a los *Thall*, *Korhien* y *Qhetzall*, que acechan a la aldea para recuperar la paz
3. Restaurar la aldea con los negocios que existían con anterioridad
4. Obtener dinero para mejorar la ciudad, la granja y el hogar para obtener más recursos

#### **4.6. Referencias**

Buscando en el mercado diferentes juegos de los cuales se puedan obtener referencias similares a la idea principal de nuestro juego para poder analizarlos, se obtienen tres juegos, dos de ellos del género JRPG y uno WRPG: *Harvest Moon: Friends of Mineral Town* y *The Legend of Zelda: The Minish Cap*, y *Moonlighter*;

*Harvest Moon: Friends Of Mineral Town* fue el primer videojuego de la saga *Harvest Moon* lanzado para GameBoy Advance, donde el jugador toma el rol de un joven que hereda una granja de su viejo amigo que ha muerto, la cual debemos sacar adelante.

Los objetivos consisten en llevar la granja con éxito gracias al cultivo y a la cría de ganado, a la vez que relacionarse con los diferentes habitantes de la ciudad y casarse con una pretendiente (si el personaje principal es hombre) antes que los rivales.

La jugabilidad de este videojuego simulador RPG de una granja en un mundo abierto da mucha flexibilidad a la hora de que el jugador pueda tomar decisiones: se puede centrar en el objetivo principal o puedo centrarse en misiones secundarias que no son relevantes para el objetivo principal.

El *target* al cual apela se podría decir, por su estilo *Cartoon*, que sería un público joven, pero realmente atrajo mucho más a un público más adulto gracias a las diferentes estrategias y tareas secundarias, junto a la melodía relajada con los cuales ayudan a evadir un poco de la vida a la que el jugador está acostumbrado a vivir hacia una vida más tranquila y relajada como la que proponen.

*The Legend Of Zelda: The Minish Cap* fue el duodécimo videojuego de la saga para la plataforma GameBoy Advance, donde el protagonista Link debe recorrer varias mazmorras venciendo a los jefes de los santuarios junto a su "gorro" Ezero para vencer a Vaati, el cual ha petrificado a la princesa Zelda.

La jugabilidad es muy fluida y cómoda, apoyada en las bases de la entrega, pero con una fórmula que le ayuda a no encasillarlo como otro juego más de la saga. Es un mundo abierto que se puede explorar, con algunos elementos o puzles que se deben ir resolviendo a medida que se avanza en el juego además de las mazmorras y santuarios. Un elemento que lo destaca del resto es la capacidad, junto a Ezero, de cambiar de tamaño para poder combinar los dos tamaños posibles para poder explorar Hyrule a fondo y superar los obstáculos, además de conocer a la raza *minish*. También aporta enemigos con mecánicas muy variadas y una interfaz de juego muy bien diseñada. En definitiva, el sistema de este juego da un aire fresco a la jugabilidad y funciona muy bien.

El *target* es un público joven y adolescente, con un estilo muy *Cartoon* y estética *Pixel Art* que acompaña muy bien al estilo de juego.

Finalmente, *Moonlighter* fue un videojuego muy reciente, lanzado en 2018 publicado para las plataformas PS4, Xbox One, PC y Nintendo Switch; Fue un *Kickstarter* cuyo argumento trata de rolear a Will, un tendero que ha heredado la tienda Moonlighter donde se venden objetos que se hayan encontrado en las mazmorras tras vencer a los enemigos. El objetivo principal es mejorar la tienda al nivel máximo posible y lograr vencer a los jefes finales de las cuatro mazmorras ampliando el pueblo con nuevos habitantes.

La jugabilidad consiste en llevar una tienda vendiendo objetos que se obtienen de derrotar a los enemigos de las mazmorras; como se puede observar, tiene una clara idea y referencia al videojuego *Harvest Moon* además de una influencia a *The Binding Of Isaac* en el sistema de mazmorras. Incluye diferentes armas a utilizar y qué elementos vender o no, pero su recorrido sigue un estilo muy lineal sin posibilidad de dar mucho abanico a la hora de ofrecer personalización del personaje.

Los elementos principales de los tres juegos que gustan mucho han sido:

- La importancia de una buena jugabilidad con controles sencillos
- Batallas a tiempo real con un sistema de combate ágil
- Las diferentes opciones de personalización del protagonista, ofreciendo al jugador tomar sus propias decisiones
- Añadir historias secundarias complementarias a la historia original para dar mayor variedad al juego
- Enemigos con diferentes puntos débiles y fuertes, lo cual obliga al jugador a buscar una estrategia para cada tipo de enemigo
- Estilo y estética acorde al *target* y tipo de juego

Para la realización de las mecánicas de este juego se ha cogido de referencia estos elementos principales extraídos del análisis.

#### **4.7. Target**

El target principal para este proyecto es el de un público joven y adolescente al cual le gustan los videojuegos del género RPG que le gusten las aventuras, la exploración y elaborar estrategias y recursos a la par de afán por superar retos para lograr los objetivos del juego.

#### **4.8. Sensaciones a transmitir**

El juego busca la identificación del jugador mediante la empatía con la historia del protagonista, además de querer apelar a la nostalgia del jugador adolescente por la estética y tipo de juego característico de los años 90.

#### **4.9. Niveles de dificultad**

La dificultad se regula mediante las zonas, donde la primera zona es relativamente más sencilla y sirve para que el jugador aprenda el alcance de su ataque, a formular estrategias delante de hordas de enemigos y, sobretodo, a prepararse para el combate de manera previa.

Los enemigos aumentan de dificultad al ir avanzando en el juego, los cuales tendrán diferentes mecánicas de ataque o patrulla y serán más rápidos, precisos y agresivos. Los jefes finales de cada raza de enemigo supondrán un reto para el jugador, ya que o bien tienen mucha vida o sus ataques son muy potentes.

#### **4.10. Recompensas**

Las recompensas irán ligadas al nivel de dificultad del enemigo; Los enemigos más sencillos *lootean* ítems que pueden ser recogidos por el jugador, los cuales le ayudan en su aventura (para recuperar energía o vida).

Los jefes finales ofrecen recompensas necesarias para pasar al siguiente nivel: más capacidad de vida para el jugador y, como *loot*, el jugador podrá recoger las llaves necesarias para abrir las puertas del castillo y de la montaña.

## 5. Diseño técnico

Para desarrollar el videojuego se han barajado diversas opciones:

### 5.1. Unity3D

Es un motor gráfico es con el que más se ha trabajado y profundizado a lo largo del desarrollo del máster, por lo tanto, con el que más familiarizada me siento. Se han desarrollado distintos tipos de juegos, tanto en 2D como en 3D y los resultados que me ha aportado han sido muy satisfactorios.

Uno de los grandes puntos fuertes de este motor es la gran cantidad de manuales, tutoriales y assets públicos en la página oficial, donde cualquier duda es fácil de ser resuelta.

Además, resulta sencillo compaginar el proyecto con un repositorio online en el cual tener copias de seguridad y poder cambiar entre los diferentes *commits* para cambiar el proyecto.

Los requerimientos para utilizar Unity son:

- Windows 7 o superior versiones de 64 bits / macOS 10.12+
- Tarjeta gráfica con capacidad para DX10
- Soporte para instrucciones SSE2

Estos requerimientos son fáciles de encontrar, lo cual facilita el desarrollo del proyecto.

### 5.2. Unreal Engine

Este motor gráfico, creado por la compañía *Epic Games*, es muy potente y uno de los más utilizados recientemente por varias compañías de videojuegos triple A. Se obtienen unos resultados muy realistas y gráficamente impecables en 3D, y gracias a su variedad de plataformas, se pueden desarrollar los proyectos en distintas plataformas y crear ports fácilmente.

Los requerimientos básicos son:

- Windows 7 o superior versión 64 bits

- Procesador de 2.0 o más GHz
- 2 GB de memoria RAM
- Compatibilidad con DirectX10
- Disco duro de 15 GB de espacio libre

Los requisitos para este motor son más altos, y además la comunidad de desarrolladores en UnrealEngine no es tan grande como la que posee Unity.

### **5.3. Decisión**

Por la familiaridad, el desarrollo de proyectos, la cantidad de información y ayuda disponible junto a la manera ágil y flexible de trabajar que proporciona se ha decantado por usar Unity3D. Ambas herramientas son muy potentes, pero con Unity la experiencia obtenida es mayor.

### **5.4 Assets y recursos del juego**

Los *assets* y recursos se han obtenido todos de la página web *opengameart.org*, donde diferentes usuarios suben sus creaciones con el fin de hacerlos públicos, con tal de que se les rinda créditos si se han utilizado en los proyectos. Hay todo tipo de recursos, desde música, efectos de sonido, pasando por *tilesets* en 2D y modelos en 3D.

Los recursos obtenidos son todos en 2D, pero en algunos casos por cuestiones técnicas del juego se han adaptado; Por ejemplo, con los enemigos *Thall* (goblins), los enemigos finales son dos de ellos de diferentes colores, así que con la herramienta *Photoshop* se han modificado los diferentes colores, tamaños o formas para poder adaptar los recursos al proyecto.

## 5.5. Componentes del juego

El juego se compone de distintos objetos jerárquicamente clasificados:

1. **Player:** jugador principal, compuesto por distintos *Scripts* que controlan el movimiento, la vida y la energía del jugador. Contiene distintas animaciones, tanto de combate como de movimiento o muerte.
2. **MainCamera:** cámara limitada por las esquinas de los mapas y sigue al jugador.
3. **GameManager:** componente que controla la visibilidad de los inventarios (tanto de las tiendas como el del jugador) y controla la mecánica del conreo.
4. **InventoryCanvas:** componente de la UI que contiene los inventarios del jugador y de las tiendas y el menú de pausa. Se encarga de controlar cada uno de los elementos que forman parte de los inventarios como el uso de éstos.
5. **Canvas:** interfaz gráfica global que muestra la vida, energía y dinero del jugador, además de las teclas de acceso rápido que aparecen en la parte inferior de la pantalla. Son un total de 6 teclas, 4 de ellas personalizables. Se pueden clicar pulsando el número del elemento. Las dos primeras no se pueden modificar, sirven para arar y regar los conreos, la tercera solamente es para indicar qué semillas se tienen para conrear y las últimas son personalizables por el jugador.
6. **Map:** mapa del juego entero con los distintos niveles, donde cada uno de los niveles se compone de un tamaño de 47 x 23 cuadrados.
7. Los componentes **SpawnPoints**, **Fences**, **Oleadas**, **Canons**, **Patrols**, **MountainEnemies** y **CastleEnemies** contienen los distintos *Scripts* y posiciones para controlar la aparición de los monstruos (dependiendo de si son en oleada o patrullan) y su movimiento. Mientras en aquel nivel existan enemigos, no se podrá volver atrás ni avanzar en el mapa.
8. **Shops:** tiendas con las cuales se podrá comerciar, tanto comprar como vender objetos dependiendo del dinero que se posea.
9. **Signs:** carteles esparcidos por el mapa que muestran en la interfaz un mensaje con información importante.

## **5.7. Funcionamiento IA de los enemigos**

Como se ha mencionado, hay tres tipos de enemigos, donde cada uno de ellos representa un comportamiento diferente:

**Thall (goblins):** enemigos de ataque cuerpo a cuerpo. Irán patrullando por la zona y, en cuanto detecten que el jugador está cerca, le perseguirá hasta acercarse lo suficiente como para poder atacar con un cuchillo.

**Korhien (árboles):** enemigos de ataque cuerpo a cuerpo, pero sin arma. Su ataque consiste en perseguir al jugador cuando éste entre en su radio de ataque y empujarle, lo cual genera daño. Su mecánica es muy simple y son los enemigos del principio del juego, los cuales resultan más molestos cuando se encuentran en masas. Además, los hay de distintos tamaños, donde los más pequeños son más rápidos, pero hacen menos daño y, cuanto más grandes, más lentos y más daño.

**Qhetzall (esqueletos):** enemigos de ataque a distancia, a base de proyectiles. Son arqueros, los cuales patrullan por su zona con un área de detección muy alta. En cuanto detectan al jugador, disparan sus flechas en dirección al enemigo, las cuales son muy rápidas y hay una alta probabilidad de que causen daño. No son especialmente resistentes, pero cuando se encuentran en grandes cantidades resulta muy molesto tener que esquivar todas las flechas, acercarse a ellos y atacarles.

## 6. Diseño de niveles (Level Design)

### 6.1. Mapa: primera aproximación

Como el mapa se divide en tres zonas, se hizo un esbozo como primera aproximación en el cual se muestran las conexiones entre estas, qué tipos de enemigos hay en cada zona y donde se encuentran los jefes finales.

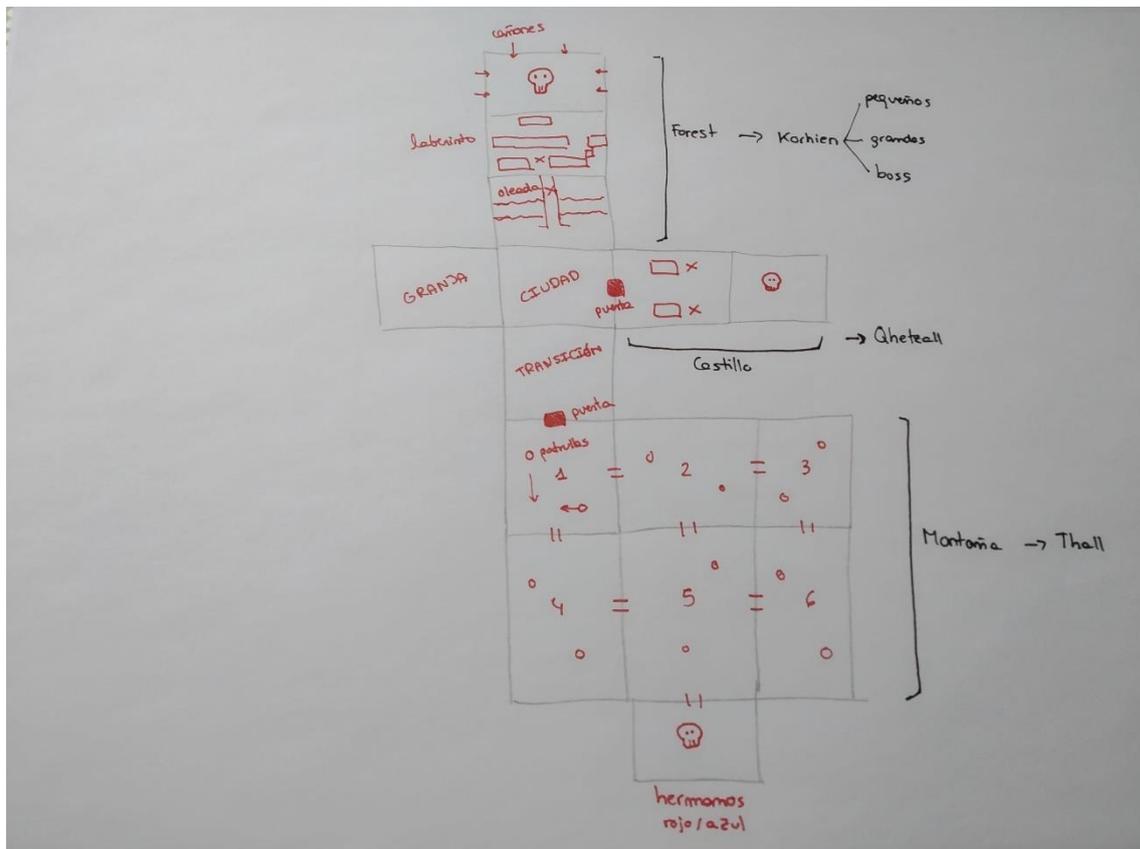


Ilustración 1: Esbozo del mapa

El mapa final ha variado un poco del mapa que se puede ver en la ilustración, por comodidad y por conexiones entre los mapas, para hacerlo finalmente más claro y ágil de desarrollar.

## 6.2. Mapa final implementado

El mapa general es un mundo abierto que se compone de distintas zonas con una estética *Cartoon* y *Pixel Art*, las cuales se describen y se muestran a continuación:

- Granja / Hogar: granja en la que está nuestro hogar; la zona amarilla es la zona en la que se pueden conrear hortalizas para recuperar energía o venderlas. Es la zona principal del juego y donde se regenera el jugador al morir.



Ilustración 2: Granja



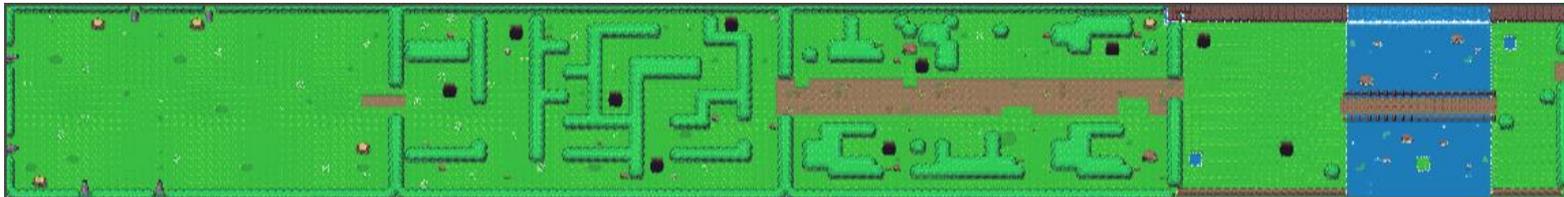
*Ilustración 3: Hogar*

- Ciudad: zona intermedia donde se conectan los niveles del castillo, bosque y montaña. Se muestran las tres tiendas que se pueden desbloquear a medida que se avance en el juego, donde se puede comerciar y vender o comprar objetos. Cada tienda se especializa en sus ítems, donde por ejemplo la tienda de frutas solamente vende frutas. Las tiendas son importantes de desbloquear, ya que el consumo de hortalizas aumenta la energía disponible (necesaria para atacar) y las pociones sirven para recuperar vida.



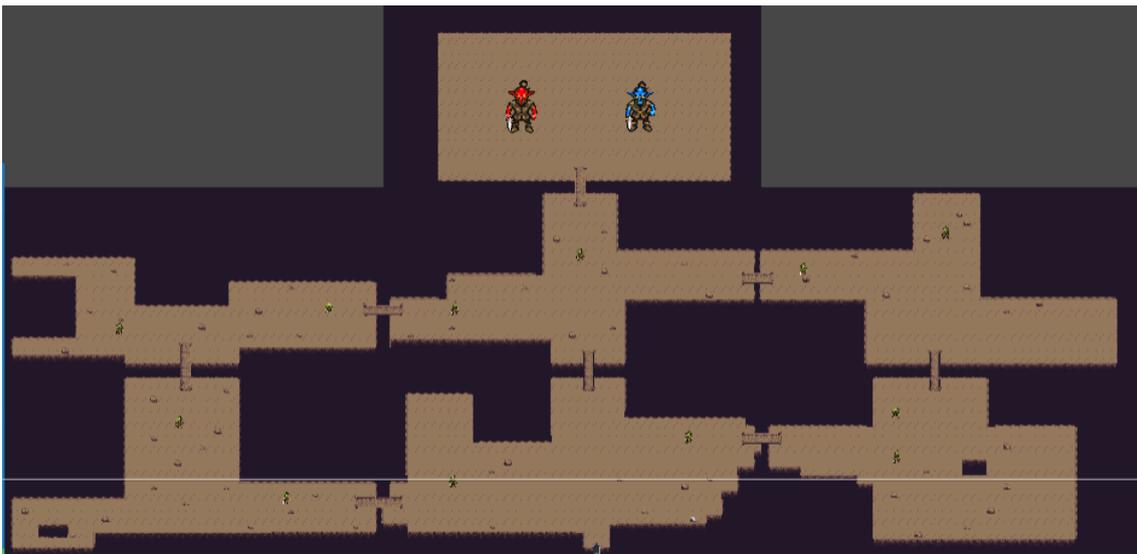
*Ilustración 4: Ciudad*

- Nivel del bosque: primera zona disponible en el juego. Se compone de generación de distintas oleadas de enemigos *Korhien* de distintos tamaños, los cuales atacan empujando al jugador. La última zona del nivel está compuesta por cañones que ayudarán al jugador a vencer al *Boss* de la zona, el cual proporcionará una llave con la que acceder al nivel de la montaña.



*Ilustración 5: Nivel de bosque*

- Nivel de montaña: segunda zona disponible con la llave de la montaña. Se compone de siete pequeñas zonas interconectadas a través de puentes y llenas de *Thalls*. Solamente se podrá acceder al *Boss* si se derrotan a todos los *Thall* de las montañas y, una vez derrotado el enemigo final, se obtendrá la llave con la que se accede al último nivel, el del castillo.



*Ilustración 6: Nivel de montaña*

- Nivel del castillo: tercera y última zona disponible con la llave del castillo, compuesto por dos zonas. Los enemigos de este nivel son los *Qhetzall*; una vez derrotados todos los enemigos se ha finalizado el juego.



*Ilustración 7: Nivel del castillo*

El mapa es obviamente ampliable en un futuro, añadiendo nuevos niveles y nuevas zonas que explorar. Finalmente, el mapa final completo quedaría como se muestra en la siguiente imagen.



*Ilustración 8: Mapa completo*

Los criterios de diseño que se han utilizado para la creación de los distintos niveles del mapa son:

- Nivel de dificultad aumenta progresivamente. El primer nivel y las primeras zonas tiene muchos sitios en los cuales esconderse y los enemigos son fáciles de derrotar. Así el jugador se acostumbra al alcance de sus ataques, cuántos puede hacer y cuánta energía recupera a medida que pasa el tiempo. Además, el jugador aprende las mecánicas básicas y se le “obliga” a hacer uso del conreo para mejorar la energía.
- Coherencia entre niveles. Entre los niveles existen las mismas mecánicas de juego, se añaden nuevas, pero sin abandonar las anteriores. El jugador entiende que cuando un enemigo se acerca o dispara algo es porque está atacando, ya que puede extrapolarlo con la realidad.
- Equilibrio. Para mantener la atención del jugador y evitar la monotonía de juego, se hacen pequeños cambios dentro de los distintos enemigos (tamaño, rapidez y potencia de ataque) y mecánicas que tienen cada uno de ellos. Además, la dificultad del juego no supera los límites de diversión, sino que depende de la estrategia del jugador y de qué tan bien equipado está (suficientes pociones de salud, suficiente comida para recuperar hortalizas, etc.).
- Navegabilidad. La navegabilidad entre las distintas zonas y niveles del juego se hace de manera suave y coherente, facilitando la memorización de los distintos caminos por los cuales el jugador ha pasado y llega a familiarizarse con ellos.
- Desafíos. Los niveles deben ofrecer distintos tipos de desafíos para mantener enganchada la atención del jugador. No deben superar un nivel en el cual resulta desagradable jugar ni tampoco caer en niveles donde resulte demasiado fácil. Por ello, los niveles aumentan progresivamente y ofrecen diferentes tipos de desafíos con los cuales el jugador usará sus habilidades y estrategias para avanzar.
- Recompensas. Las recompensas son importantes para animar al jugador a seguir adelante y seguir con la narrativa de la historia. Por ello, los enemigos finales ofrecen recompensas que permiten avanzar en los niveles además de mejorar la capacidad de vida del personaje y los enemigos normales ofrecen pociones para recuperar vida.

## 7. Usabilidad

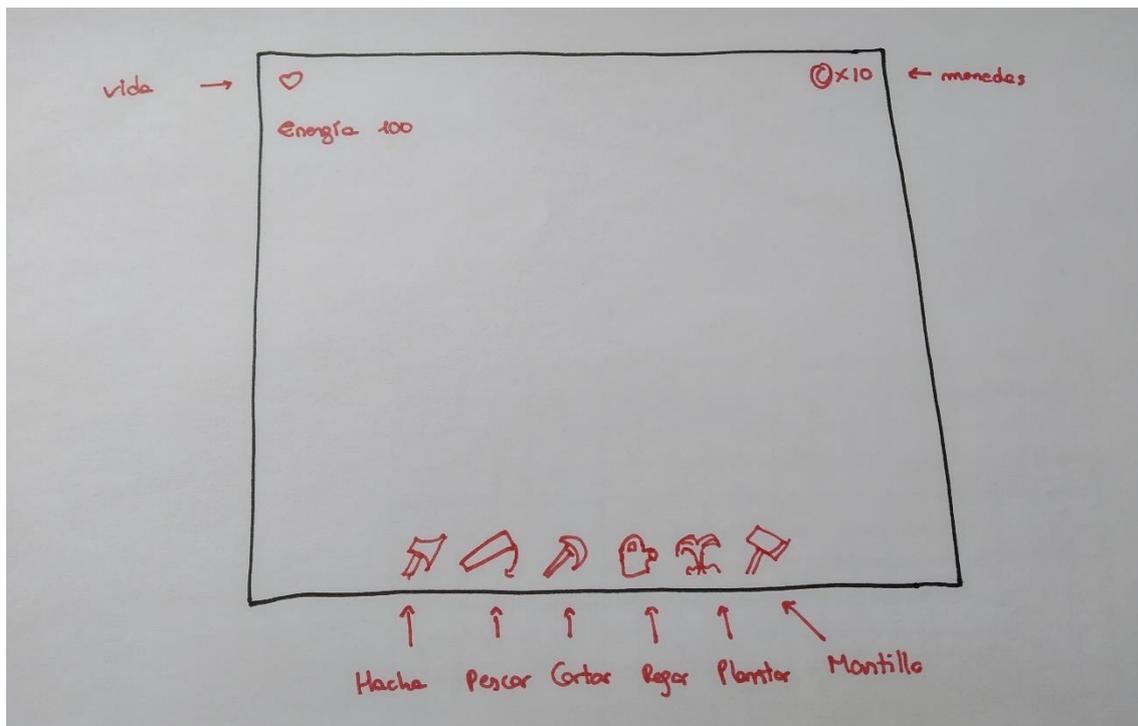
Una vez concluido el juego, se han realizado diferentes pruebas con usuarios para valorar la usabilidad y la interfaz del juego. Gracias a estas pruebas se genera un *feedback* del usuario al desarrollador con el cual se obtiene:

- Una puntuación sobre cuán agradable y sencilla es la interfaz gráfica para el jugador
- Cuantificación de tiempo de cuánto tarda el jugador en aprender a usar los controles del juego
- Detección de posibles errores o comportamientos erróneos del juego
- Retroalimentación de si las instrucciones del juego se entienden o si se necesita añadir información extra
- Propuestas para la mejora de la interfaz

Con la primera propuesta de interfaz gráfica, los resultados que se obtuvieron fueron que ésta no era la más adecuada para el tipo de juego, pues había elementos que no eran necesarios ya que aún no estaban implementados y que, además, para realizar una acción, el camino para realizarla no era sencillo: era repetitivo e ineficiente. Por ejemplo, para plantar una semilla, se debía escoger cada una de ellas a través del inventario, plantarlo, volver a abrir el inventario y repetir el proceso.

Otra propuesta fue el cambio de la energía, en vez de ser un número que fuera un *slider*, ya que es mucho más gráfico y si disminuye o aumenta es mucho más perceptible para el usuario.

Por último, los controles no eran del todo adecuados pues era fácil confundirse entre ellos, pero las instrucciones sí que fueron claras.



*Ilustración 9: esbozo primera interfaz*

Con ello se introdujo la propuesta de crear un inventario con todos los ítems del protagonista y, además, la creación de un inventario rápido como en el juego *Minecraft*, donde el jugador puede asignar a través del inventario qué ítems quiere tener más a mano para hacer la experiencia más amena y divertida sin tener que perder el tiempo buscando.

Con los cambios propuestos, se implementó la nueva interfaz de usuario como se muestra en la siguiente imagen, de la cual se volvió a realizar más pruebas con los mismos usuarios y se obtuvieron los siguientes resultados:

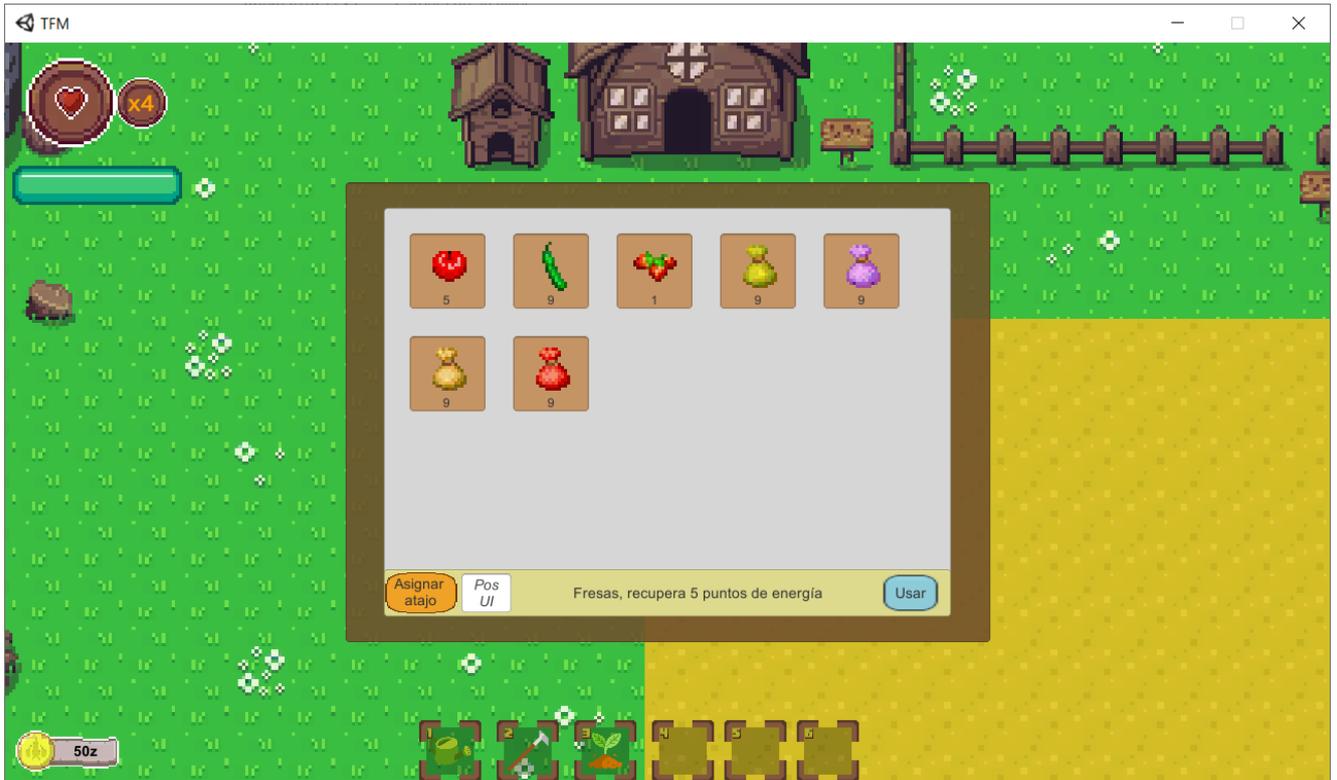


Ilustración 10: interfaz con el inventario

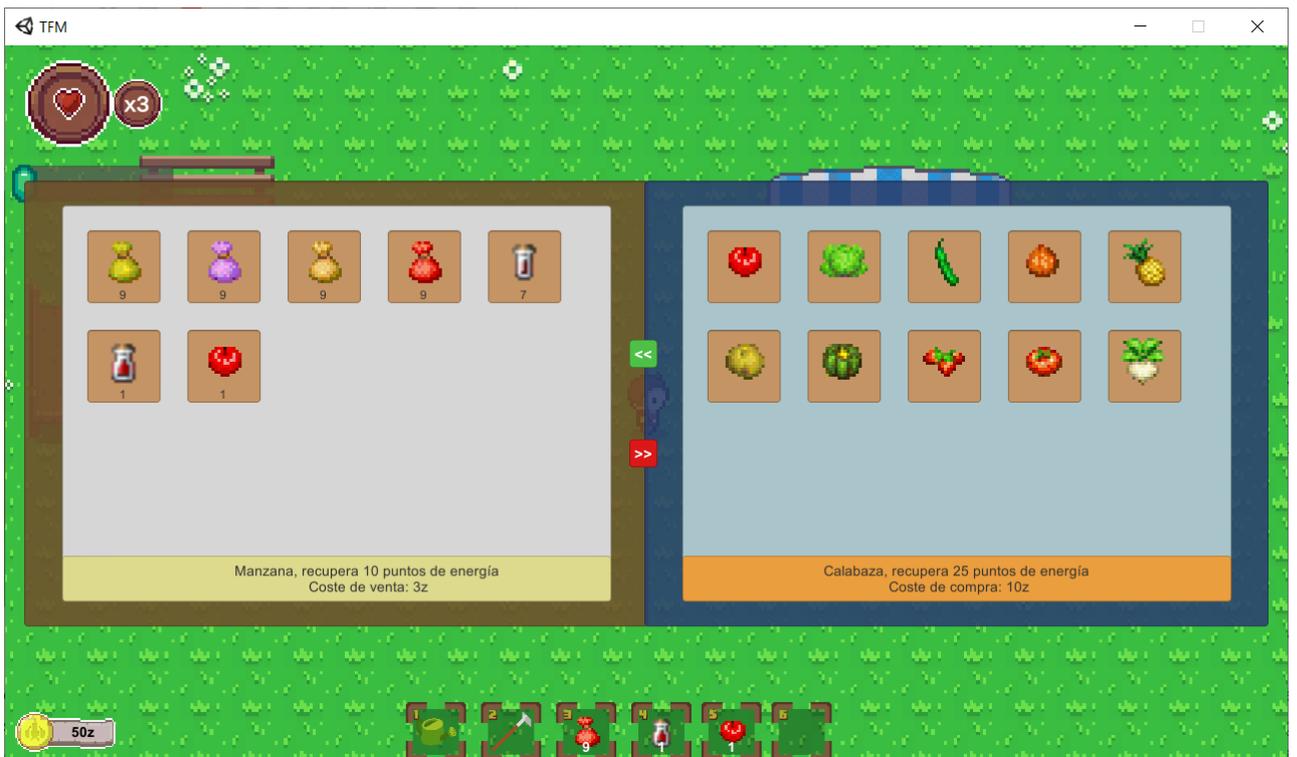


Ilustración 11: interfaz con el inventario de compraventa

Ahora los usuarios encuentran que la interfaz proporciona los elementos y la información necesarios para jugar, además de que encuentran muy útil la inclusión del inventario rápido. Consideran, también, que los controles son fáciles y sencillos de memorizar. Se detectaron múltiples bugs y errores del juego en los sistemas de inventario, muerte y cambio entre escenas que ya han sido resueltos. En general, coinciden en que la experiencia y la sensación al jugar al juego es positiva.

## 8. Manual de usuario

### 8.1. Requerimientos

Los requerimientos para jugar al juego son:

- Teclado
- Ratón

Se ha desarrollado en Windows 10, pero a partir de Windows 7 y con una tarjeta gráfica normal se puede jugar.

**IMPORTANTE:** para que se pueda ver correctamente todo el diseño de nivel, ejecutar en tamaño 1280x720

### 8.2. Instrucciones del juego

Para jugar el videojuego se utiliza:

- Ratón para seleccionar elementos gráficos (elementos de inventario, clicar en las tiendas o para la agricultura)
- Teclas de teclado del 1 al 6: acceso a las herramientas de acceso rápido visibles en la interfaz. Pulsando el botón 1 se equipa la regadera, con el 2 la azada, y los cuatro restantes son personalizables a partir del inventario.
- Tecla I: ataque del jugador
- Tecla Escape: mostrar y quitar inventario
- Tecla P: mostrar y quitar menú de pausa
- Teclas WASD: movimiento del personaje

## 9. Conclusiones

Algunas de las conclusiones que puedo extraer de la elaboración de este trabajo serían la importancia de la organización y tener claros los objetivos y mecánicas que se quieren incluir. Al principio la idea y los objetivos que quería implementar eran demasiado ambiciosas para el tiempo del que disponía, por lo tanto, tuve que reconducir los objetivos y las mecánicas que podía y era capaz de implementar en el tiempo disponible. Gracias a eso, se han logrado adquirir los objetivos básicos del juego e implementar distintas mecánicas para hacer uso de los conocimientos adquiridos durante el máster, como es el diseño de los niveles, misiones u objetivos y hechos para tener en cuenta a la hora de diseñar el videojuego.

No todos los objetivos se han logrado, ya que se tuvieron que reconducir por el poco tiempo útil disponible y éstos eran demasiado ambiciosos. Sin embargo, los objetivos básicos que consistían en crear un videojuego con principio y fin (al menos de un nivel) con la implementación de tres mecánicas distintas y utilizar nuevas técnicas para el desarrollo del proyecto, sí se han logrado hacer.

La planificación a groso modo (distribuida entre las cuatro entregas) sí se ha cumplido y se ha hecho un seguimiento mediante la planificación del diagrama de Grant. Por falta de tiempo (trabajo y otros estudios) no se ha seguido a rajatabla, sino que se ha desarrollado cuando se ha podido. Se han tenido que introducir diferentes cambios para garantizar que el proyecto siguiera adelante, como el replanteamiento de objetivos y minimizar las expectativas del resultado final.

Algunos objetivos que se quedan como líneas de futuro son los siguientes:

- Inclusión de NPC y otros actores con los cuales conversar y mostrar el repoblamiento de la ciudad a medida que se avanza en la narrativa del juego.
- Crear distintos negocios (como el herrero o el carpintero) y agrandar la ciudad
- El uso de distintos tipos de arma (arco, lanza, escudo)

- Crear más niveles de juego con más enemigos (cuyas mecánicas sean distintas)
- Crear mazmorras de generación aleatoria con la inclusión de un minimapa.

## 10. Glosario

*RPG*: role-playing game (juego de rol)

*Loot*: botín

*Asset*: recurso que representa un ítem. Puede ser un archivo de audio, una imagen, un modelo 3D, etc.

*NPC*: non-playable character, personaje que no representa al jugador, sino un personaje más incluido en la historia

*Backup*: copia de seguridad

*IA*: inteligencia artificial, comportamiento del objeto según determinados eventos

*Objetivo*: meta que se quiere alcanzar

*Mecánica*: describe los componentes jugables o no que forman el videojuego (por ejemplo, saltar, disparar, etc.)

*Script*: archivo código que maneja y ejecuta unas órdenes

*Componente*: elemento que forma parte de la jerarquía y arquitectura del videojuego, pudiendo ser compuesto por uno o más componentes dentro de él mismo

## 11. Bibliografía

*Unity3D Productos* (s.f.):

<https://unity3d.com/es/unity>

*UnrealEngine cómo funciona* (s.f.):

<https://www.unrealengine.com/en-US/what-is-unreal-engine-4>

*Unity3D Tilemap manual* (sin fecha):

<https://docs.unity3d.com/Manual/class-Tilemap.html>

*Harvest Moon* (s.f.):

[https://harvestmoon.fandom.com/es/wiki/Harvest\\_Moon:\\_Friends\\_of\\_Mineral\\_Town](https://harvestmoon.fandom.com/es/wiki/Harvest_Moon:_Friends_of_Mineral_Town)

*Wikipedia The Leyend of Zelda: The Minish Cap* (s.f.):

[https://es.wikipedia.org/wiki/The\\_Legend\\_of\\_Zelda:\\_The\\_Minish\\_Cap](https://es.wikipedia.org/wiki/The_Legend_of_Zelda:_The_Minish_Cap)

*Wikipedia Moonlighter* (s.f.):

<https://es.wikipedia.org/wiki/Moonlighter>

*Wikipedia Cronología de los juegos de rol* (s.f.):

[http://es.wikipedia.org/wiki/Anexo:Cronolog%C3%ADa\\_de\\_los\\_juegos\\_de\\_rol](http://es.wikipedia.org/wiki/Anexo:Cronolog%C3%ADa_de_los_juegos_de_rol)

*Diagrama de grant* (s.f.):

<https://plan.tomsplanner.es/>

*3dJuegos diferencia JRPG WRPG* (s.f.):

<https://www.3djuegos.com/foros/tema/26054469/0/rpg-y-jrpg-analisis/>

*Wikipedia – Historia del rol* (s.f.):

[https://es.wikipedia.org/wiki/Historia\\_de\\_Videojuegos\\_de\\_Rol\\_Occidentales#Resurgimiento\\_\(2000s%E2%80%93presente\)](https://es.wikipedia.org/wiki/Historia_de_Videojuegos_de_Rol_Occidentales#Resurgimiento_(2000s%E2%80%93presente))

*Jakob Nielsen, 10 Usability Heuristics (s.f.):*

<https://tfa.stanford.edu/download/TenUsabilityHeuristics.pdf>