

Kor'Dul: Tales Of An Unknown Hero

Gorka Magra Tolosa

Máster en Diseño y desarrollo de Videojuegos.
Diseño y Desarrollo de un videojuego.

**Jordi Duch Gavaldà, Heliodoro Tejedor Navarro
Joan Arnedo Moreno**

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/)

FICHA DEL TRABAJO FINAL

| | |
|---|--|
| Título del trabajo: | <i>Kor'Dul: Tales Of An Unknown Hero</i> |
| Nombre del autor: | <i>Gorka Magra Tolosa</i> |
| Nombre del consultor/a: | <i>Jordi Duch Gavalda, Heliodoro Tejedor Navarro</i> |
| Nombre del PRA: | <i>Joan Arnedo Moreno</i> |
| Fecha de entrega (mm/aaaa): | 06/2019 |
| Titulación:: | <i>Máster en Diseño y desarrollo de Videojuegos.</i> |
| Área del Trabajo Final: | <i>Diseño y Desarrollo de un videojuego.</i> |
| Idioma del trabajo: | <i>Español</i> |
| Palabras clave | |
| <p>Resumen del Trabajo (máximo 250 palabras): <i>La finalidad de este proyecto es la creación de un trabajo propio y original desde cero, desde la planificación del mismo, el diseño de niveles y gráficos hasta el desarrollo de la lógica de sistemas y jugabilidad. De esta forma, se obtiene un proyecto en el que poder seguir trabajando a futuro y la experiencia de haber trabajado en un juego desde todas sus fases.</i></p> <p><i>El videojuego será un RPG por turnos de dos dimensiones, basado en un mundo de fantasía donde el jugador encarna un humano con una aventura por delante.</i></p> <p><i>Dado que el proyecto se realizó únicamente por un alumno, la organización del mismo fue simple. Se realizó un seguimiento de los objetivos a realizar por medio de una tabla de Trello y se subieron los cambios realizados para las entregas en repositorio de Git. En cuanto a las fases de diseño y desarrollo, se utilizaron las guías presentadas por el profesorado a lo largo del curso, siguiendo meticulosamente las diferentes fases de diseño y respetando las reglas de usabilidad y diseño. Por último, se utilizó el motor de juego Unity para el desarrollo del mismo, así como la herramienta Piskel para la creación de los diseños en PixelArt.</i></p> <p><i>Por último, los resultados de este trabajo son dos; el producto en si, que servirá para poder seguir desarrollando en el futuro y añadir al portfolio del estudiante, y la experiencia obtenida de un desarrollo real de un proyecto en todas sus etapas.</i></p> | |

Abstract (in English, 250 words or less): *The main purpose of this project is the development of a own and original videogame from scratch, all the way from the planification, going through the level and art design and the development of the logic systems and gameplay. Doing this, we would achieve a project in which it could keep working on the future and the experience of working in a videogame at all it's stages.*

The title will be a 2D turn based RPG based in a fantasy world where the players would incarnate a human on its personal quest.

Given that the project was accomplished by one student, the management was simple. A trello table was used to get a simple but reliable view of the amount of objectives to do, and the changes made were uploaded to a Git repository upon the upload deadlines approached. The design and development of the game was made following the resources given by the teachers along the course of the year, following meticulously the usability and design guidelines. Finally, Unity game engine was used during the game development and the software called Piskel gave me the tools I needed in order to make the game's pixelart designs.

Lastly, this project's results were two. Firstly, the product obtained as a videogame, which could be continued developing in a future or added in the student's portfolio as a sample of what he could achieve, secondly, the experience obtained during this time, realizing a real videogame during all of its phases and development.

Índice

| | |
|--|----|
| 1.-Introducción..... | 1 |
| 1.1.-Contexto y justificación del Trabajo..... | 1 |
| 1.2.-Objetivos del Trabajo | 1 |
| 1.4.-Planificación del Trabajo | 2 |
| 1.5.-Breve resumen de productos obtenidos | 4 |
| 1.6.-Breve descripción de los otros capítulos de la memoria | 4 |
| 2.-Objetivos y alcance | 5 |
| 2.1.-Objetivo general | 5 |
| 2.2.-Objetivos Específicos | 6 |
| 3.-Justificación del proyecto | 8 |
| 4.-Metodología | 9 |
| 5.-Estado del arte | 10 |
| 5.1.- Genero de videojuego:..... | 11 |
| 5.2.-Historia del Genero | 14 |
| 5.2.1.-Inicios del genero..... | 14 |
| Pedit5 | 14 |
| Rogue | 15 |
| Ultima | 16 |
| 5.2.2.-Época dorada | 17 |
| The legend of Zelda..... | 17 |
| Dragon Quest | 18 |
| Final Fantasy | 19 |
| 5.2.3.-Declive y resurgir | 20 |
| Fallout..... | 20 |
| Baldur's Gate | 21 |
| Dragon Age: Origins | 22 |
| 5.3.-Revisión de la tecnología utilizada | 23 |
| 5.3.1.-Cry Engine | 24 |
| 5.3.2.-Unreal Engine | 25 |
| 5.3.3.-Unity..... | 26 |
| 6.-Definición del juego..... | 28 |
| 6.1.-Idea del juego..... | 28 |
| 6.1.1.-Breve descripción del juego..... | 28 |
| 6.1.2.-Subgenero y referencias a videojuegos existentes..... | 28 |
| 6.1.3.-Tipo de interacción juego-jugador..... | 29 |
| 6.1.4.-Plataforma de destino | 29 |
| 6.2.-Conceptualización | 30 |
| 6.2.1.-Historia, ambientación y/o trama | 30 |
| 6.2.2.-Definición de los personajes/elementos e interacción entre los mismos | 30 |
| 6.2.3.-Objetivos planteados al jugador..... | 31 |
| 6.2.4.-Concept Art: Screenshots... .. | 31 |
| 6.3.-Desarrollo y roadmap | 33 |
| 6.3.1.-Planificación de Objetivos..... | 33 |
| 6.3.2.-Cuantificación de tiempo y recursos por objetivo..... | 34 |

| | |
|---|----|
| 7.-Diseño técnico..... | 35 |
| 7.1.-Entorno de desarrollo..... | 35 |
| 7.2.-Herramientas empleadas..... | 37 |
| 7.2.1.-Piskel..... | 37 |
| 7.3.-Recursos utilizados..... | 39 |
| 7.3.1.-Diseño de los Sprites..... | 39 |
| 7.3.2.-Diseño de la interfaz..... | 41 |
| 7.4.-Esquema de arquitectura del juego..... | 42 |
| 8.-Diseño de niveles..... | 44 |
| 8.1.-Escena del pueblo..... | 44 |
| 8.2.-Escena del bosque..... | 45 |
| 9.-Manual de usuario..... | 46 |
| 9.1.-Requisitos hardware mínimos..... | 46 |
| 9.2.-Manual de usuario..... | 46 |
| 10.-Conclusiones..... | 47 |
| 10.1.-Conclusiones del trabajo realizado..... | 47 |
| 10.2.- Líneas de trabajo futuro..... | 48 |
| 11.-Glosario..... | 49 |
| 12.-Bibliografía..... | 50 |

Lista de figuras

| | |
|--|----|
| Figura 1: Vista general de las tareas realizadas en el diagrama de Gantt. | 3 |
| Figura 2: Vista general de las tareas realizadas en el diagrama de Gantt. | 3 |
| Figura 3: Ingresos del mercado en 2018 según plataformas. | 5 |
| Figura 4: Imagen de la escena "Pueblo" del videojuego. | 6 |
| Figura 5: Fragmento de la clase "EnemyBehaviour". | 7 |
| Figura 6: Ranking de unidades vendidas en base a género en estados unidos durante 2018. | 8 |
| Figura 7: Ejemplo de flujo de trabajo en metodología agil. | 9 |
| Figura 8: Margen de beneficios en miles de millones del cine, música y videojuegos. | 10 |
| Figura 9: Captura de DeusEx, un ARPG de un mundo futurista. | 12 |
| Figura 10: Captura de Divinity: Original Sin2, un TRPG sin grid de casillas. | 12 |
| Figura 11: Captura de world of warcraft, un icono de los MMORPG a lo largo de los años. | 13 |
| Figura 12: Pantallazo del videojuego "Pedit5" | 14 |
| Figura 13: Pantalla de "rogue" en su versión de ascii | 15 |
| Figura 14: Ultima1, durante la pantalla de mundo | 16 |
| Figura 15: Legend of Zelda. | 17 |
| Figura 16: Dragon Warrior. (Dragon Quest 1) | 18 |
| Figura 17: Final Fantasy 1. | 19 |
| Figura 18: Fallout: A post nuclear role playing game. | 20 |
| Figura 19: Baldur's Gate. | 21 |
| Figura 20: Dragon Age: Origins. | 22 |
| Figura 21: Ventana de ejemplo de motor Unity3D. | 23 |
| Figura 22: CryEngine | 24 |
| Figura 23: UnrealEngine | 25 |
| Figura 24: Unity | 26 |
| Figura 25: Captura del videojuego "Golden Sun". | 28 |
| Figura 26: Mapa de territorio de Warhammer Fantasy. | 32 |
| Figura 27: Concept art de pueblo medieval europeo. | 32 |
| Figura 28: Imagen del IDE "Visual Studio". | 35 |
| Figura 29: Visual Studio | 36 |
| Figura 30: Imagen de un sprite en Piskel. | 37 |
| Figura 31: SmartGit. | 38 |
| Figura 32: Bocetos de diseño. | 39 |
| Figura 33: Diseño de un edificio en Piskel | 40 |
| Figura 34: Imagen de la escena "Town". | 40 |
| Figura 35: Boceto en Invision de la Interfaz de usuario. | 41 |
| Figura 36: Imagen de la interfaz dentro del juego. | 41 |
| Figura 37: Estructura del proyecto en Unity. | 42 |
| Figura 38: Diagrama de clases del proyecto. | 43 |
| Figura 39: Esquema de la escena "Town". | 44 |
| Figura 40: Esquema de la escena "Forest". | 45 |

1.-Introducción

1.1.-Contexto y justificación del Trabajo

A la hora de realizar este proyecto se tuvieron en cuenta dos necesidades a cubrir, la primera desde un punto de vista comercial, y la segunda personal.

La necesidad comercial nace en base a los títulos que nos encontramos hoy en día en el mercado. Tal y como veremos más adelante en el apartado correspondiente, la mayoría de videojuegos que salen al mercado y que tienen un mínimo impacto en el mismo son títulos de ARPG o JRPG. Títulos que se separan del clásico RPG por turnos y que podría seguir teniendo una buena base de jugadores afines. Con este producto se intenta llegar a ese segmento de público que no dispone de títulos de este género tan frecuentemente.

Por otro lado, la necesidad personal radica en la obtención de experiencia en proyectos completos, que hayan pasado por todas las fases de desarrollo y que sirvan el día de mañana a la hora de desarrollar proyectos independientes o como portfolio al entrar en una empresa que así lo requiera.

1.2.-Objetivos del Trabajo

- Diseño y desarrollo de los gráficos PixelArt del videojuego.
- Diseño e implementación de interfaz básica del videojuego.
- Diseño y creación de dos escenas de juego básicas para el funcionamiento del mismo durante la exposición.
- Implementación de un sistema de estadísticas RPG.
- Implementación de un sistema de combate por turnos.
- Implementación de un sistema de itemización.
- Diseño y creación de una misión de prueba para mostrar durante la exposición.
- Balanceo de estadísticas de jugador y enemigos para hacer un videojuego, desafiante y escalable.

1.3.-Enfoque y método seguido

A la hora de cumplir los objetivos establecidos para este proyecto, se barajaron diferentes opciones, cada una de las cuales requeriría un tiempo de realización diferente y unas características específicas para poder llegar a cumplirlo. De esta forma, al inicio del proyecto se estudiaron las siguientes vías de poder realizar el mismo:

- **Creación de todo el contenido del videojuego:** El producto mostrado al final del periodo del trabajo de fin de master sería completamente propio: Diseños, interfaz, sonidos, animaciones... Todo habría sido realizado por el estudiante y requeriría por tanto un esfuerzo extra para el estudio y aprendizaje a la hora de realizar esto.
- **Creación del apartado gráficos y lógico del videojuego:** Al contrario de la primera opción, ni los sonidos ni los efectos se realizarían de manera propia por el estudiante, sin embargo, se realizaría un estudio y posterior aprendizaje sobre el diseño gráfico de la interfaz y aspecto del juego.
- **Creación del apartado lógico / codificación del videojuego:** La opción más sencilla, únicamente se desarrollaría la lógica y sistemas del videojuego,

Dados las dos variables fundamentales a la hora de desarrollar este proyecto (Esto es, el tiempo requerido y el aprendizaje obtenido), se optó por la segunda opción, ya que es la que permitirá desarrollar en profundidad el título sin excederse ni abarcar demasiado poniendo en riesgo el tiempo disponible.

1.4.-Planificación del Trabajo

Los recursos utilizados en este proyecto se listarán en su propio capítulo de la memoria, pero el software a utilizar será brevemente nombrado a continuación:

- **Unity 2018.3.0f2:** El conocido motor de juego nos dará soporte para poder desarrollar nuestro videojuego 2D de forma eficaz.
- **Piskel 0.14.0:** Este software nos permitirá diseñar e importar los diferentes sprites que utilizaremos en el videojuego, escalándolos en caso de que fuese necesario o editándolos.
- **SmartGit:** El cliente de Git servirá para poder actualizar nuestro repositorio y tener un control de versiones del proyecto.
- **TeamGantt:** Permitirá realizar los diagramas de Gantt necesarios para la siguiente sección de esta introducción.
- **Trello:** Nos dará el soporte para la creación y seguimiento de objetivos a realizar.

Por otro lado, la planificación se estimó en base a lo explicado en la primera entrega realizada, se hizo un seguimiento en base a las PEC de la asignatura y a los requisitos mostrados por el profesorado.

Kor'Dul

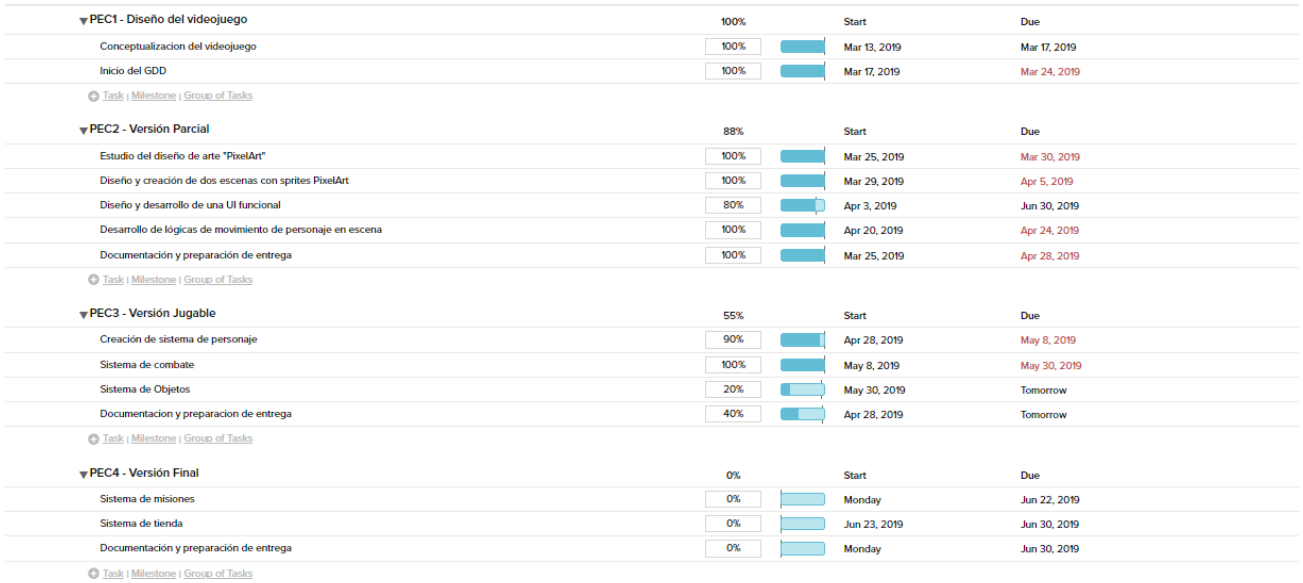


Figura 1: Vista general de las tareas realizadas en el diagrama de Gantt.

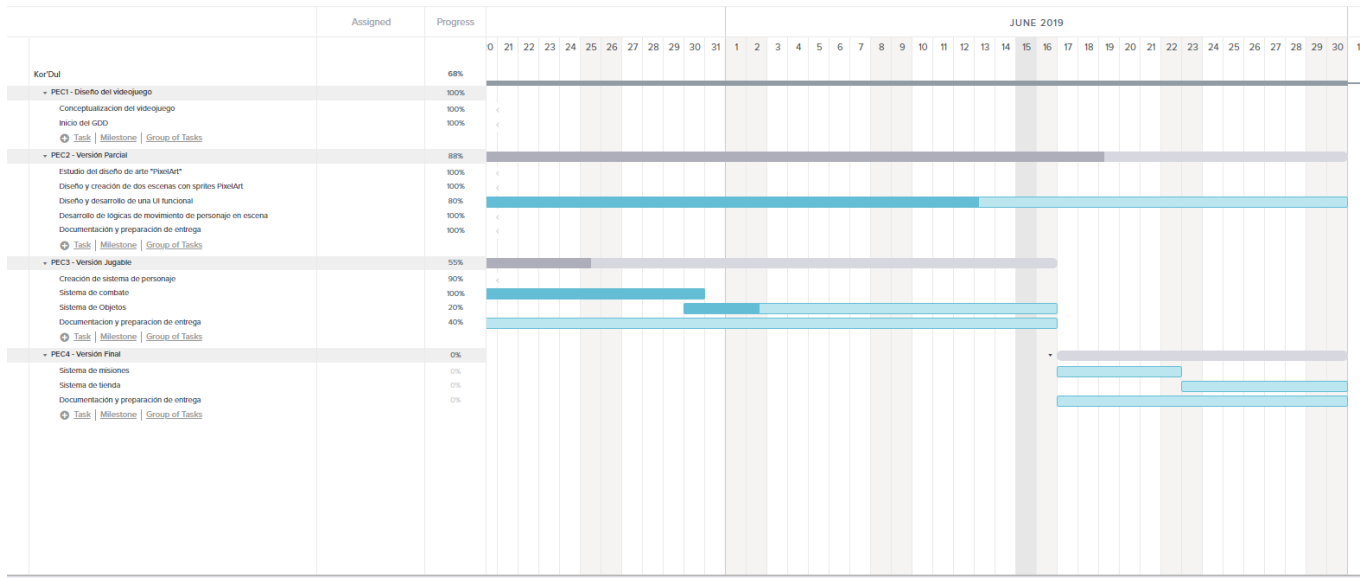


Figura 2: Vista general de las tareas realizadas en el diagrama de Gantt.

1.5.-Breve resumen de productos obtenidos

El producto obtenido al finalizar el proyecto es el videojuego “Kor’Dul: Tales Of An Unknown Hero”, un videojuego del genero Aventura, RPG y acción, que transportara a los jugadores a una historia en la que un humano que sobrevivió a un ataque de demonios intenta buscar una solución a la inevitable victoria de estos frente a la humanidad.

1.6.-Breve descripción de los otros capítulos de la memoria

En los siguientes puntos del documento se realizará un análisis de la toma de decisiones realizada en el proyecto y los criterios utilizados para ellos, a continuación, se mostrará un breve resumen del contenido de cada capítulo.

- **Objetivos y alcance:** Los objetivos que se plantean en el videojuego y el alcance del mismo. Este capítulo servirá para definir qué es lo que el producto hará y que no.
- **Justificación del proyecto:** Aquí se definirá por qué se hace este videojuego, las razones que han llevado al alumno a desarrollar este proyecto en concreto.
- **Metodología:** Este capítulo servirá para definir qué criterio se ha llevado a la hora de realizar las entregas y avanzar en el mismo.
- **Estado Del Arte:** El estado del arte nos permitirá echar un vistazo al mercado, ver de qué otros títulos bebe nuestro videojuego, como se ha visto influenciado y que particularidades nuevas añade. A su vez, nos servirá para hacer una comparativa entre las tecnologías disponibles y ver los criterios que se han utilizado para elegirlos.
- **Definición del juego:** Una versión resumida del GDD, incluyendo información básica sobre los diferentes aspectos del juego, el diseño del mismo y un resumen de la planificación realizada.
- **Diseño técnico:** Un listado de las tecnologías utilizadas para la creación del título y las decisiones tomadas.
- **Diseño de niveles:** Este capítulo mostrará los diferentes niveles y un resumen de las decisiones de diseño utilizadas y que se podrá encontrar el jugador en cada una.
- **Manual de usuario:** Un resumen de las especificaciones necesarias y atajos de teclado útiles para poder utilizar el juego correctamente.

2.-Objetivos y alcance

2.1.-Objetivo general

El objetivo principal de este proyecto es la planificación, diseño y desarrollo de un videojuego de Rol, en el que el usuario deberá, emprendiendo su aventura en el mundo de Kor'Dul, salvar al imperio de la humanidad de la destrucción por parte de los demonios. Para poder cumplir esto, deberá subir de nivel y avanzar en su propia historia mejorando en su viaje, obteniendo mejores objetos, ayudando a las aldeas por las que viaje y aprendiendo nuevas habilidades.

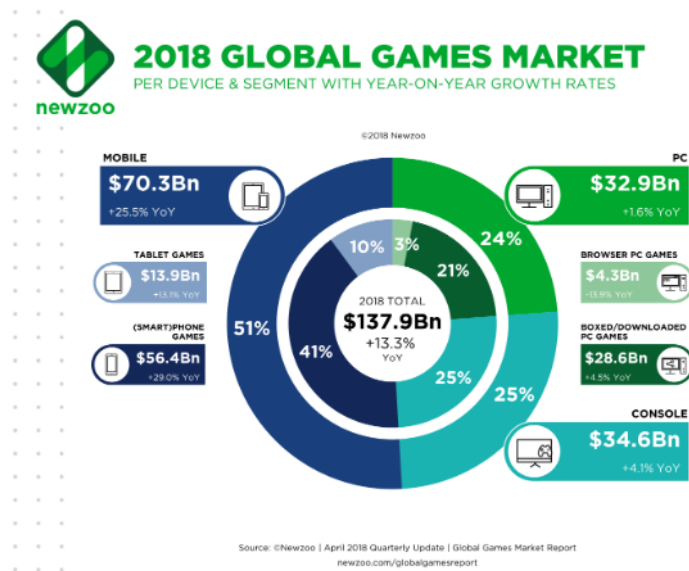


Figura 3: Ingresos del mercado en 2018 según plataformas.

La plataforma elegida para publicar este videojuego será PC, ya que, si bien el género de juegos de rol es común tanto en consolas portátiles como en sobremesa, el videojuego estará diseñado para salir en PC en primer lugar, y en base a la aceptación del mismo, llevar este videojuego a las diferentes consolas posteriormente. La interfaz de usuario estará dirigida principalmente para punteros, elecciones de personaje, inventario del mismo con mouseover y los diferentes menús harán que el jugador de ordenador se sienta más cómodo con él, si bien, en un futuro podría amoldarse para ofrecer una versión de GamePad más sencilla. Por otro lado, será mucho más sencillo organizar una comunidad en torno a un juego de PC desde la plataforma Steam o Epic Games, fomentando a estos jugadores desde foros y páginas web.

Por último, la opción de publicar el videojuego en dispositivos móviles también sería atractiva, ya que, como vemos en la imagen superior, el mercado móvil crece exponencialmente y ocupa la mayoría de los ingresos de la industria, además, el traspaso de la interfaz de puntero podría hacerse fácilmente a versiones de pantallas táctiles.

2.2.-Objetivos Específicos

Para el desarrollo final del videojuego se realizara el diseño de tres escenas, un pueblo en el que el jugador empezara y actuara como zona segura (Con vendedores, personajes no jugadores que den misiones al jugador, etc...), un bosque, que hará las veces de zona de peligro para cuando el jugador viaje por ahí en busca de completar misiones o avanzar en la historia, y una escena de combate en el bosque, completando de esta manera el mínimo necesario para que se pueda mostrar la funcionalidad del videojuego.

A su vez, todos los assets utilizados en el videojuego serán de estilo propio, tal y como se ha ido mostrando a lo largo de las entregas, esos diseños han requerido pasar por diversas fases de creación hasta implementarlas dentro del juego. Para hacer estos sprites se ha decidido utilizar la técnica llamada "PixelArt" que consiste en hacer dibujos complejos basándose en los propios pixeles. De esta forma, el juego obtiene un toque más clásico, propio de los antiguos géneros de Rol por turnos.

Además, uno de los requisitos que se mostraran en la entrega final será la Interfaz de Usuario diseñada para el título. Para poder hacer esto, se seguirán las reglas de usabilidad y diseño provistas por el profesorado a lo largo del curso, haciendo un estudio meticuloso de las mismas e intentando aplicarlas en la medida de lo posible.



Figura 4:Imagen de la escena "Pueblo" del videojuego.

Pasando a los objetivos de la logica de sistemas del juego, este contara con un sistema de estadísticas propio de cualquier juego de rol, con estadísticas principales como Fuerza, destreza o magia, y secundarias como suerte o agilidad. Sera tarea del alumno estudiar como equilibrarlas para ofrecer la mejor experiencia de juego haciendo uso de ellas.

Ademas, el jugador dispondra de un sistema de inventario en el cual podra equiparse distintos objetos o consumibles para poder hacer uso de ellos, el daño realizado sera en base al arma que este utilizando el jugador, la estadística vinculada a ese arma y el tipo de ataque o habilidad que se este

utilizando. Además, cada personaje podrá llevar un máximo de tres consumibles para utilizar en combate.

Otro de los aspectos relevantes del juego serán los enemigos. Cada zona del juego dispondrá de diferentes enemigos, que se generaran con un rango de estadísticas de forma que todos tengan siempre distintas características.

```
if (this.gameObject.tag.Equals("Slime"))
{
    life = 50 + (Random.Range(-10, 10));
    remainingLife = life;
    damage = 10 + (Random.Range(-2, 2));
    droppedMoney = 5 + (Random.Range(-1, 1));
    initiative = 5;
    loot = null;
}
```

Figura 5: Fragmento de la clase "EnemyBehaviour".

Tal y como se puede ver en la figura superior, los enemigos no tendrán un "nivel" predefinido, de forma que lo único que cambiara de ellos será ese rango en el cual un mismo monstruo podría ser más fuerte o más débil, de esta manera, el jugador podría encontrar que un combate contra un mismo monstruo es relativamente más difícil que contra otro de su mismo tipo.

El sistema mediante el cual se le irá presentando la historia principal al jugador será mediante el sistema de misiones, una serie de tareas que guiarán al usuario presentándole las diferentes funcionalidades al principio de juego a modo de tutorial y mostrándole las opciones y decisiones que tendrá a lo largo del mismo cuando vaya avanzando.

Por último, el sistema de combate será por turnos como ya se ha mencionado y incluirá en esta primera versión combates de un personaje contra un enemigo. Sin embargo, la idea para un futuro será hacer que estos combates puedan aparecer un grupo de enemigos y que el jugador a su vez, encuentre diversos aliados por el camino que puedan acompañarle en sus aventuras, mejorando así la funcionalidad de trabajo en grupo en los diferentes equipos.

3.-Justificacion del proyecto

Tal y como se mencionó en la propia introducción del proyecto, el mismo está justificado en dos grandes razones, las cuales se desglosarán a continuación:

En primer lugar, en la industria de los videojuegos los juegos de rol (RPG o Role Playing Games) siempre han gozado de una posición privilegiada en cuanto a comunidad de jugadores, sin ir más lejos, tal y como podemos ver en la figura inferior, en 2018 fue el cuarto género que más copias vendió en estados unidos.

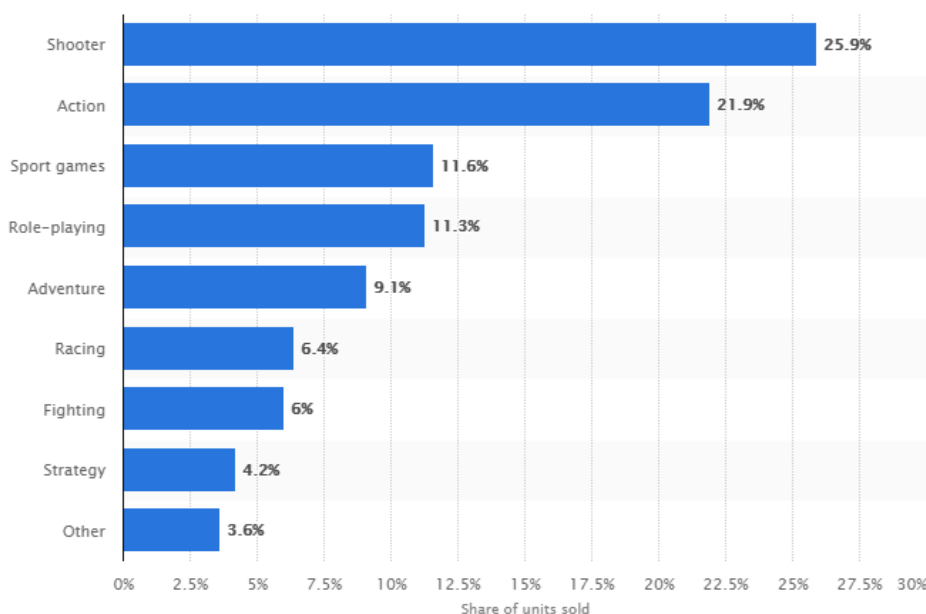


Figura 6: Ranking de unidades vendidas en base a género en estados unidos durante 2018.

Sin embargo, en la lista de juegos de rol más vendidos en los últimos años, tenemos que irnos a la décima posición (después de juegos tan vendidos como Skyrim o The Witcher 3) para encontrar al primer RPG por turnos del género, Divinity: Original Sin 2. Juego el cual, fue lanzado y financiado en Kickstarter y apoyado por una inmensa cantidad de jugadores.

Por otro lado, la decisión de crear un proyecto de estas dimensiones viene de la necesidad de aprender a desarrollar el mismo durante todas sus fases de producción. De esta forma se obtienen dos resultados, en primer lugar, obtener la experiencia necesaria en el área de diseño, ya que mi perfil a lo largo de los años ha sido de programación. Por otro lado, el producto resultante podría resultar o bien para formar parte de un portfolio a la hora de entrar en una empresa, que pueda servir como ejemplo de los realizado, o para seguir trabajando en el mismo y publicarlo en un futuro.

4.-Metodología

La metodología por la que se ha optado a seguir en el proyecto a lo largo del mismo está basada en las metodologías ágiles. Dado que el grupo de desarrolladores de este proyecto está limitado a una persona, la repartición de tareas no ha sido necesaria y por lo tanto se procederá a obviarla.

Las metodologías ágiles se basan en iteraciones incrementales sobre un software en base a los requisitos que un cliente va dando al proyecto a lo largo del mismo, estas iteraciones suelen llamarse “Sprints” y como ejemplos más conocidos de este tipo de metodologías tenemos a Scrum o Kanban.

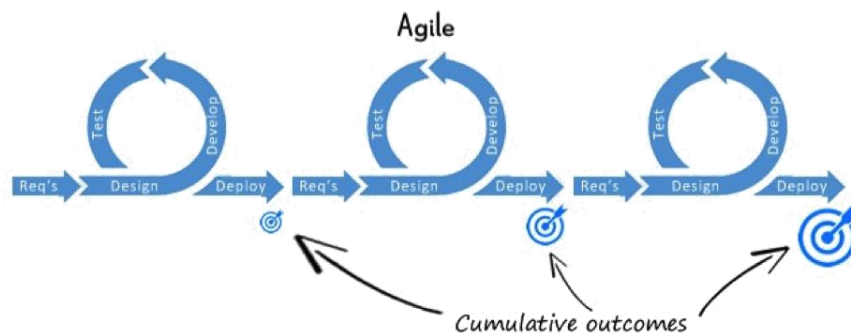


Figura 7: Ejemplo de flujo de trabajo en metodología ágil.

Tal y como podemos ver en la imagen superior, las metodologías ágiles se definen en base a reuniones con el cliente donde este da su visto bueno al proyecto y da directrices a los desarrolladores para poder continuar con el mismo. Estos puntos de iteración tan importantes serán los puntos de entrega de las PEC del proyecto, y los sprints, el tiempo entre los mismos que el alumno tendrá para desarrollar lo establecido y mejorar lo puntualizado por los profesores encargados. De esta forma, y al igual que en la metodología ágil, podremos encontrar el resultado óptimo para el mismo, mejorando nuestro producto y con el mínimo de retroceso posible en caso de que un cambio no sea el deseado.

5.-Estado del arte

La industria de los videojuegos está en constante crecimiento y es que, para un sector que prácticamente está naciendo si lo comparamos con las diferentes áreas del entretenimiento, la cantidad de dinero invertido y ganado es inmensamente superior al de su competencia directa, el cine y la música.

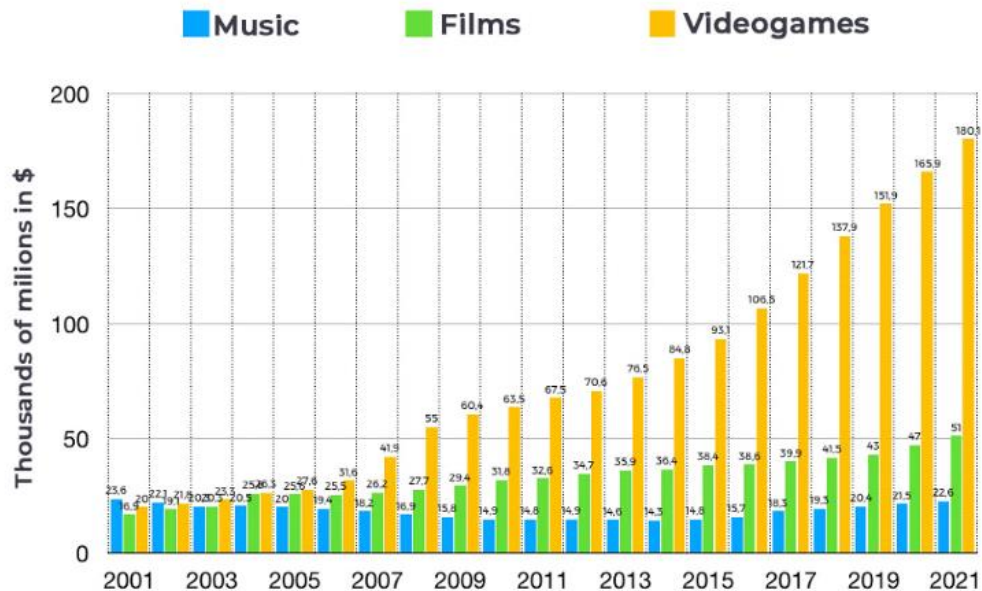


Figura 8: Margen de beneficios en miles de millones del cine, música y videojuegos.

A esto se le suma la relativa facilidad con la que puedes obtener un sitio en esta industria, esto lo hemos podido ver con la aparición de plataformas globales relacionadas con los mismos como Twitch o Discord. O la repercusión de los eventos gaming sobre la sociedad, como la E3. El mundo de los videojuegos cada vez tiene más influencia, y por lo tanto, requiere de un flujo de títulos de calidad que nutra esa demanda requerida.

Si entramos a datos más específicos dentro del mundo de los videojuegos, tal y como veíamos en la figura 3, que el incremento del uso del móvil no hará, sino que favorecer estas estadísticas, todo el mundo podrá consumir videojuegos desde cualquier parte gracias a ello.

5.1.- Genero de videojuego:

El género de los videojuegos de rol (RPG) es un género donde el jugador asume el control y las acciones de un personaje o grupo de personajes en un mundo en concreto, ya sea de fantasía o real. Este género tiene su origen en los juegos de rol de mesa (Tabletop Roleplay games) y la mayoría de estos beben de parte de sus mismas mecánicas. (Niveles, atributos, experiencia, objetos, clases...) A su vez, uno de los añadidos más importantes de este tipo de juegos es la rica historia que hay detrás, ya que el juego como tal tendrá que superponer a la antigua figura del maestro de juego a la hora de inventar el trasfondo y el matiz de los personajes en el juego.

A lo largo de los años la mayoría de los videojuegos dentro de este género han ido distanciándose más y más de sus inicios, llegando a los modernos ARPG que estamos acostumbrados en un intento de llamar a mas jugadores como público objetivo.

Las características principales de un videojuego de rol serían las siguientes:

- Sistema de personajes con estadísticas y niveles.
- Sistema de itemizacion de objetos equipables.
- Combates como forma principal de mejorar a nuestro personaje.
- Una historia rica como trasfondo para la aventura.
- Juegos generalmente de un jugador.

Así mismo, dentro del género de RPG podemos encontrar diferentes subgéneros:

ARPG (Action Role Playing Game): Los llamados RPG de acción suelen estar centrados en controlar a un único personaje con una trama única y relaciones centradas en el mismo y con tanto combates como acciones en tiempo real. Generalmente estos juegos ofrecen sistemas parecidos a los "Hack And Slash" o los "Shooters". Estos juegos tienen el combate como foco principal de entretenimiento y generalmente tienen un trasfondo e historia de peor calidad que el resto de los RPG. Los ejemplos más sonados de este subgénero son: "Mass Effect", "The legend of Zelda" o "Borderlands".



Figura 9: Captura de DeusEx, un ARPG de un mundo futurista.

TRPG (Tactical Role Playing Game): También llamados SRPG (Strategy Role Playing Game) es un género con elementos de los RPG y los juegos de estrategia. Este género basa su combate en un sistema muy parecido a los juegos de rol de tablero. Y es que sus movimientos, habilidades y distancias en combate están completamente supeditadas a las casillas de los mismos. A su misma manera, estos videojuegos suelen hacer uso de mecánicas de coberturas, terrenos elevados, terrenos difíciles o reglas de combate específicas. Estos videojuegos pueden ser difíciles de entender y generalmente obligan al jugador a hacer uso de todas las habilidades y características del grupo del personaje. Algunos de los ejemplos más famosos son “Final Fantasy Tactics” o “X-COM”.



Figura 10: Captura de Divinity: Original Sin2, un TRPG sin grid de casillas.

MMORPG (Masive Multiplayer Online Role Playing Games): Los juegos de rol multijugador masivo se caracterizan, como bien dice su nombre, por la posibilidad de poder jugar con cientos de jugadores en el mismo mundo. Estos juegos poseen generalmente una historia y trasfondo de mundo algo mas pobre que los RPG clasicos y ofrecen un contenido mas casual y menos centrado en numeros, estrategias y estadisticas. Por el contrario, profundizan mas en las mecanicas sociales y en el aspecto multijugador del mismo, como por ejemplo en las hermandades o clanes que tienen estos juegos, las mazmorras conjuntas o los combates masivos entre los jugadores. Estos juegos van desde los primeros juegos con capacidad para varios jugadores como Ultima Online, hasta iconos de la industria como World of Warcraft.



Figura 11: Captura de world of warcraft, un icono de los MMORPG a lo largo de los años.

5.2.-Historia del Genero

La historia de los juegos de RPG se ramifico desde sus inicios, y es que desde los primeros videojuegos hubo dos confluencias entre los videojuegos de rol japoneses (JRPG) y los videojuegos de rol del oeste (WRPG). A continuación, se mostrará una lista de los videojuegos más relevantes o que más influencia tuvieron a lo largo de los años, destacando aquellos que pudiesen haber influenciado el proyecto a presentar.

5.2.1.-Inicios del genero

Pedit5

También llamado “The Dungeon”, pedit5 es probablemente el primer RPG creado del que se tiene constancia, el creador, Rusty Rutherford lo lanzo al mercado en invierno de 1975. En él, el jugador avanzaba a lo largo de un sinfín de habitaciones de mazmorra matando monstruos y consiguiendo su tesoro.



Figura 12: Pantallazo del videojuego “Pedit5”

Si bien estamos hablando del primer juego de RPG, las mecánicas típicas de los mismos ya estaban bien definidas en él. Los tesoros y los monstruos se generaban aleatoriamente y en combate el personaje tenía varias opciones y hechizos para utilizar. Al morir, el jugador tendría que empezar de nuevo, siendo así también el primero del género “RogueLike”.

Rogue

Para descubrir el siguiente gran título tendremos que viajar hasta 1980, cuando A.I. Design publico el juego de Rogue: Exploring the dungeons of Doom.

En este videojuego se representa ya una historia que hace que el videojuego cobre sentido. El principal objetivo del jugador será encontrar el amuleto de Yendor en la cámara más profunda de las mazmorras. Para ello, el jugador deberá avanzar por las salas derrotando a todos los monstruos disponibles y equipándose con los tesoros que estos dejen caer.



Además de equipamiento, este nuevo sistema incluirá la utilización de consumibles para poder ayudar al jugador en combate, pociones, pergaminos y vendas se volverían los recursos indispensables en este juego.

A su vez, rogué fue uno de los primeros juegos en implementar el combate en turnos, utilizando cada punto del tablero como casilla para realizar un movimiento, los jugadores deberían planificar bien sus turnos para poder salir victoriosos. Además, y tal y como podemos ver en la figura inferior, se publicó con unos gráficos basados en lenguaje ASCII. Pasando a tener unos gráficos propios en la versión de Atari.

Rogue se convirtió rápidamente en un juego de culto entre los estudiantes de las universidades en 1980, viendo claras influencias de el en los siguientes juegos WRPB durante los siguientes años.



Figura 13: Pantalla de "rogue" en su versión de ascii

Ultima

Ultima o Ultima I: The first Age of Darkness es el primer juego de la serie de ultima creado por Origin Systems y publicado en 1981. En ella, el jugador deberá embarcarse en la misión de destruir la gema de la inmortalidad, que está siendo utilizada por el mago Mondain.

En este título, al jugador se le presenta como “El extranjero”, un héroe invocado a ese mundo con el fin de terminar con el reinado de Mondain. Como podemos ver, el background de este juego RPG es con gran diferencia, mucho más profundo que los ya vistos hasta ahora en el listado, el juego, además, ofrece diferentes puntos de vista para el jugador durante su gameplay; durante la exploración, el extranjero se moverá a lo largo del mundo desde una cámara superior. Mientras que en interior de mazmorras la perspectiva será de primera persona.

Durante la partida, el jugador también podría ir a tiendas con el fin de comprar objetos y consumibles con los tesoros y dinero ganados al derrotar enemigos. O comida, para evitar que el personaje muriese de inanición.

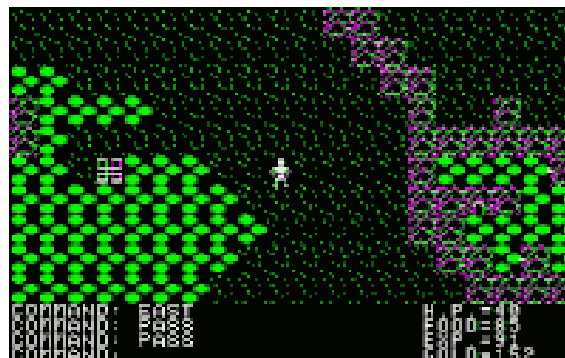


Figura 14: Ultima1, durante la pantalla de mundo

La serie de ultima ha sido de gran transcendencia para el género, desde el inicio con el primer título, hasta el último desarrollado en 2013. Muchos concuerdan en que el desarrollo del mundo y el trasfondo de personajes de este juego dieron el pistoletazo de salida a la época dorada de los RPG.

5.2.2.-Época dorada

The legend of Zelda

En febrero de 1986 Nintendo desarrollo y publico el primer juego de las aventuras de Link. "The Legend Of Zelda". Un juego de acción y aventura RPG que llevaría al jugador a la tierra de Hyrule encarnando a Link en busca de la trifuerza para poder salvar a Zelda de Ganon.

Este título, el primer gran éxito de los JRPG, constituía un juego basado completamente en la cámara superior mientras nuestro personaje se movía por el escenario. Los cambios de escena se realizaban en los extremos de la pantalla y los enemigos se movían libremente por la misma, haciendo que los combates fuesen en tiempo real.

Lo novedoso del juego, además de la propia historia en la que se nos introducía, era las características armas que se nos presentaban a lo largo del mismo. La espada, el bumerán o la ocarina nos permitirían manejarnos por las escenas de diferentes maneras y usando múltiples estrategias, así mismo, cada cierto tiempo se nos presentaba un monstruo mayor con el que deberíamos de actuar de una manera en concreto con el fin de derrotarlo.

La saga de Legend of Zelda ha sido y sigue siendo uno de los grandes gigantes del RPG, desde su primer título en 1986 hasta los títulos que siguen apareciendo hoy en día.



Figura 15:Legend of Zelda.

Dragon Quest

El primer título de dragon quest (llamado Dragon Warrior cuando salió en Estados Unidos) se desarrolló por Chunsoft en 1986, y fue el predecesor de una saga que durara hasta el día de hoy, en el que siguen apareciendo títulos.

En el primer dragon warrior el jugador se le presenta con la historia principal nada más empezar y se le deja a su libre albedrío por el mundo. De esta forma, se obtiene una sensación de libertad en el mundo abierto y se le da la opción al jugador de proceder como le apetezca.

Para que el jugador pueda salir victorioso en su aventura, el juego le dispone de varios pequeños escenarios o "misiones secundarias" en los que podrá ayudarse para progresar en su nivel de experiencia, haciendo de esta forma que el camino hacia la aventura sea menos monótono. De la misma manera, si el jugador así lo quisiera, no tendría ninguna penalización en avanzar directamente en el juego hacia zonas de más nivel, topándose con enemigos más difíciles sin ningún tipo de advertencia.



Figura 16: Dragon Warrior. (Dragon Quest 1)

Esta capacidad para dar la libertad al jugador y dejarlo a merced del mundo y su historia, le hizo ganar a la saga una gran cantidad de jugadores, que vieron como el género ganaba cada vez más adeptos.

Final Fantasy

En 1987 Square (Que todavía no se había fusionado con Enix) desarrollo y publico el primero de sus Final Fantasy's. La saga, que como se ha ido viendo a lo largo de los años ha tendido hacia los ARPG, empezó como un RPG por turnos de combate sencillo.

En el título, el jugador encarnaba a los cuatro héroes que llevaban los orbes elementales, oscurecidos por los cuatro enemigos elementales. Juntos, deberían aventurarse en su misión de volver a iluminarlos para salvar su mundo.



Figura 17:Final Fantasy 1.

Tal y como se puede presuponer del título, Final Fantasy dotaba al jugador de un grupo de 4 héroes a su disposición, pudiendo hacer uso de ellos para cualquier estrategia que quisiera, añadiendo de esta forma complejidad a la hora de decidir qué es lo que hacer en cada turno de combate y que con que personaje hacer que acción.

La interfaz era bastante sencilla, una vista global para mover al jugador por el mundo cuando necesitase explorar, y una vista de combate basada en pequeños menús cuando este se encontrase a un enemigo. Con esta sencilla formula, la desarrolladora japonesa consiguió ganar la comunidad de RPG y continuar sacando títulos hasta la época actual.

5.2.3.-Declive y resurgir

Fallout

Desde finales de los años 80 viajamos hasta finales de los 90 para ver el resurgir de los RPG de manos de Interplay Productions, el primer juego de Fallout, llamado Fallout: A post nuclear role playing game. El juego nos lleva a un mundo alternativo post apocalíptico del siglo 22, en el que el protagonista, un habitante de un refugio, intenta sobrevivir a los peligros de un mundo radioactivo para salvar su hogar.



Figura 18: Fallout: A post nuclear role playing game.

Una de las características que hizo tan llamativo a este título en su momento es la interacción con los diferentes NPC's del mundo. Cada uno de ellos tiene su propia historia y en base a las decisiones del jugador, tendrá un resultado u otro. En base a estas acciones, el jugador obtendría puntos de experiencia y karma, dándole al jugador diferentes salidas para poder tener varios resultados.

Además, la interfaz que desarrollo Interplay productions favorecía la inmersión del jugador en el universo. El uso del pipboy para la mayoría de los menús, las estadísticas SPECIAL del jugador y el uso de los puntos de acción para el combate denotaban una gran mejora en cuanto a interfaces de usuario de la época.

Baldur's Gate

El primer videojuego de la serie de Baldur's Gate vino de la mano de BioWare y Black Isle Studios en 1998. En el videojuego, el jugador crea su propio personaje para viajar a través de la Sword Coast con un grupo de mercenarios para desentrañar el misterio de una crisis que afecta a la zona.

El juego causo furor debido a que implementaba las reglas de D&D de segunda edición, haciendo que los jugadores acérrimos de este juego de rol de mesa se unieran para probar la contrapartida digital. Fue uno de los títulos señalados por la comunidad de revitalizar y resurgir el género de RPG.

Durante la historia que está dividida en varios capítulos (como una campaña del juego de mesa) los jugadores podrán personalizar su personaje tal y como se pudiese hacer en Dungeons & Dragons, dotando así de una personalización nunca antes vista antes. Los jugadores podrían aprender habilidades, mejorar sus estadísticas, cambiar la formación e incluso llegar a hacer multi-clase.

Este arsenal de sistemas de juego completamente nuevos le dio una clara victoria en el mercado y el resultado fue una continua saga del título hasta la época actual.



Figura 19: Baldur's Gate.

Dragon Age: Origins

El siguiente título en la lista no llegaría hasta 2009, cuando BioWare lanzaría al mercado su primer Dragon Age. En este primer juego, un RPG por turnos sin grid, encarnaríamos el papel de un guardián gris, una orden de caballeros de elite encargados de proteger al mundo del archidemonio cuando sea necesario.



Figura 20: Dragon Age: Origins.

La historia que nos presentó Bioware era lineal, sin embargo, donde realmente era excepcional fue en las decisiones del jugador, ya que estas podrían afectar al desenlace de la historia.

Además, el grupo de NPC's que acompañarían al jugador sería de un total de 6 personajes, pudiendo alternar entre ellos, mantener conversaciones y relacionarse con ellos. A su vez, la relación con los mismos podría empeorar o mejorar en base a las decisiones tomadas durante la historia. Los personajes que te acompañaban podrían ser controlados a su vez por la inteligencia artificial mediante una serie de reglas que podía introducir el jugador en su configuración, haciendo que este solo se centrara en el héroe principal.

5.3.-Revisión de la tecnología utilizada

Tal y como se ha visto a lo largo del curso, el motor de juego es la herramienta principal de un desarrollador. En el no solo se puede modificar las escenas, el diseño y localización de los assets, sino que también ejerce de puente para el desarrollador entre el propio videojuego y el entorno de desarrollo, herramienta de animaciones y mucho más...

Es por esto que la elección de motor de juego influye en el producto final que se está realizando, ya que este se debe acoplar a las necesidades del juego, llegando incluso a desarrollar un propio motor de juego si un producto así lo necesitase. Obviamente, debido a los requisitos del proyecto y el tiempo disponible para el mismo, la opción de realizar un propio motor de juego se descartó al inicio, de forma que alumno tendría que elegir uno de los disponibles del mercado.

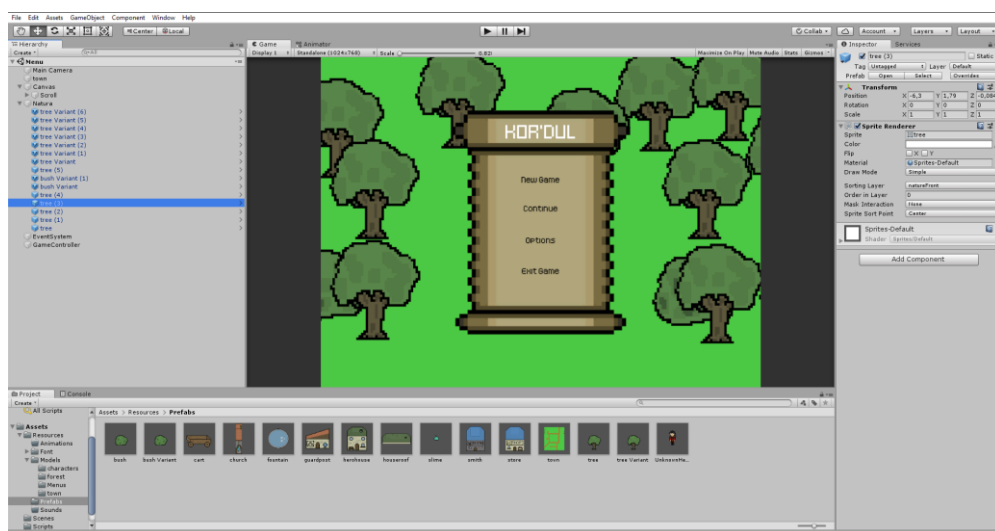


Figura 21: Ventana de ejemplo de motor Unity3D

En las siguientes páginas se mostrará los principales motores de juego disponibles, enumerando sus ventajas, sus desventajas y las características que podrían ser útiles para este proyecto en concreto. Al final se mostrará una conclusión con el motor de juego elegido y la serie de ventajas que aportaría al proyecto.

5.3.1.-Cry Engine

CryEngine es un motor de juego creado por la empresa Crytek, fue implementado por primera vez en el videojuego Far Cry en 2004 y posteriormente en Crysis en 2009. Este motor se puede programar en C++, Lua y C# y es compatible con todos los SO de consolas disponibles.

El fuerte de CryEngine es su potencia gráfica, sobresale tanto en iluminación, en físicas realistas y en técnicas de animación, sin duda alguna es el motor que actualmente saca más provecho a animaciones y gráficos de calidad alta y es completamente necesario en juegos que quieran deslumbrar en este apartado. Al ser un motor tan potente y centrado en la calidad de animaciones y gráficos controlar el rendimiento y la optimización del mismo es una tarea complicada, por lo tanto, no es recomendable para desarrolladores que tienen poca experiencia, ya que el motor en si no es user-friendly. Además, los desarrolladores podrían tener problemas al tratar físicas no-realistas.

Como conclusión, podemos decir que CryEngine es una bestia capaz de mover cualquier tipo de gráficos que creas posible, su atractivo se basa en las animaciones y los elementos gráficos del juego, pero para saber utilizar bien el motor y poder aprovecharlo en condiciones sin que ello afecte al rendimiento del juego.



Figura 22:CryEngine

5.3.2.-Unreal Engine

Tal y como se comentó anteriormente, Unreal Engine es un motor creado por la compañía Epic Games, comenzó siendo utilizado en 1998 por la misma compañía para la creación de sus primeros videojuegos, Unreal Tournament, Turok o BioShock. Este motor está programado en C++ y es compatible tanto con OpenGL como con DirectX 11 y 12. El motor es de licencia gratuita para el uso y desarrollo de proyectos, salvo en caso de que el mismo se comercialice, en cuyo caso se deberá pagar un 5% de royalty a la empresa. Además, es multiplataforma y permite la implementación en la mayoría de las consolas.

En 2015 la empresa publicó la cuarta generación de su motor de juego, esta versión soporta y ha servido para muchos juegos AAA que han salido en el mercado, proporcionando herramientas útiles para los nuevos proyectos, además unreal posee una de las mejores comunidades, apoyando con la documentación que publica Epic Games y ayudando a los mismos a mejorar el motor, ya que unreal engine es open source.

Por otro lado, al ser un motor tan potente requiere de un equipo de última generación para poder trabajar de forma óptima, los usuarios recomiendan un procesador i7 y 8GB de RAM para poder utilizarlo en buenas condiciones. Además, el motor de Epic Games es uno de los motores con peor curva de aprendizaje, ya que de la forma que está desarrollado, su estructura se limita a hacer los procedimientos de una forma en particular, de forma que podría suponer más de un quebradero de cabeza.

En resumen, unreal engine es un motor muy potente apto para muchos juegos con requisitos altos, pero salirse de las directrices principales para las que el motor fue creado (mundos pequeños de primera persona) podría suponer un problema para los desarrolladores.



Figura 23:UnrealEngine

5.3.3.-Unity

Unity es un motor creado por Unity Technologies, lanzado a mediados de 2005 e inicialmente diseñado únicamente para funcionar en Mac, pero que tras su éxito inicial consiguió obtener soporte para una amplia gama de plataformas, incluyendo el PC y las consolas más conocidas. Se puede programar en C, C# y JavaScript. El motor de juego es de licencia gratuita y basada en Royalties al llegar a cierta cantidad de ganancia por comercialización.

Unity ha sido el lenguaje más atractivo para los nuevos desarrolladores de videojuegos, principalmente porque pese a ser un motor de juego, tiene una curva de dificultad más sencilla que el resto del mercado y las opciones e interfaces son user-friendly. Además, la comunidad de unity ofrece muchas herramientas, ya sean guías o post de ayuda a usuarios nuevos como assets gratuitos que poder utilizar en tu proyecto.

Como contrapartida, unity nunca ha destacado por ser un buen motor gráfico, si es cierto que han salido juegos gráficamente avanzados como “The Dwarves” o “The Forest”, pero nada comparado con los motores mencionados anteriormente. Además, dada la estructura del propio motor de juego, la creación de juegos de gran escala, triple A, se desaconseja dado los inconvenientes de rendimiento.

Para finalizar, Unity es un motor de juego ideal para estudios pequeños o medianos que están empezando en el mundo de los videojuegos o que vayan a desarrollar un videojuego sin requerimientos muy grandes. Es por ello que lo hace el motor de juego idóneo para nuestro proyecto y para el grupo de desarrolladores que lo conformamos.



Figura 24:Unity

5.3.4.-Conclusión

Como bien se hizo saber a lo largo de las entregas realizadas, el proyecto se realizó en el motor de juego Unity. La versión de la misma que se utilizara será 2018.3.0f2 y no se utilizara ningún paquete de importación ni assets de la tienda de Unity, la razones por las que se eligió este motor de juego fueron las siguientes:

- La creación de **proyecto en 2D se ajustaba a este motor de juego**, y es que Unity fue y sigue siendo uno de los mejores motores para diseñar y renderizar escenas en 2D, por esto mismo, se ajusta a las necesidades del proyecto.
- **El conocimiento del motor** suponía una ventaja a la hora de elegirlo, ya sea por las horas de aprendizaje durante el postgrado como antes del mismo, ese tiempo que de otra manera se tendría que haber utilizado en utilizar el motor de juego, se invirtió en otras tareas de más importancia.
- **La comunidad que existe detrás del motor** es un punto a favor del mismo, si bien estos últimos años Unreal ha ganado más fuerza entre los desarrolladores, Unity dispone de una gran cantidad de apoyo para aquellos que deseen desarrollar haciendo uso de la plataforma y una gran comunidad en la que apoyarse.
- Como ya se comentó durante las entregas, **el título podría pasar a ser multiplataforma** en un futuro cercano, dicho esto, Unity facilita al desarrollador esta tarea, haciendo que las builds en diferentes plataformas sean más sencillas de ajustar y compilar.
- Por último, las características de **planes económicos que provee la plataforma es muy atractiva para los nuevos desarrolladores**. Se podría mantener la versión gratuita hasta la fecha de publicación, donde, se pasaría a un plan de pago de Unity para seguir con su desarrollo.

6.-Definicion del juego

6.1.-Idea del juego

6.1.1-Breve descripción del juego

El juego a realizar se titulará “Kor’dul: Tales of an unknown hero”, y en el juego el jugador encarnará al personaje principal de la historia, el héroe desconocido, recorriendo el mundo de Kor’dul en busca de completar su cruzada personal.

6.1.2.-Subgenero y referencias a videojuegos existentes

El juego entrara en el género de RPG clásico, con mecánicas del mismo como combate contra enemigos, misiones que aparecerán a lo largo de la trama y la subida de niveles a lo largo de la historia, que permitirán al jugador controlar a su personaje conforme vaya avanzando en el juego y obtener un grado extra de profundidad a la hora de caracterizar a este individuo.

En cuanto a subgéneros dentro del mismo, el videojuego estará dentro de los clásicos “RPG¹ por turnos”, ya que a diferencia del auge de los ARPG² que se ha visto a lo largo de los años, este título estará más enfocado a el rol clásico de juego de mesa, con estadísticas principales, habilidades y combate por turnos. Además, el videojuego presentara una historia linear con posibles alternativas y diferentes personajes. La trama estará centrada en el viaje que realizará un aventurero y las diferentes subtramas irán ocurriendo conforme el jugador avance en la historia, dicho esto, otro de los subgéneros del título seria “Aventura”.



Figura 25: Captura del videojuego “Golden Sun”.

¹ RPG: del inglés Role-Playing Game, juego en el que los jugadores toman control de un personaje en una historia.

² ARPG: del inglés Action Role-Playing Game, juego RPG con combates fluidos enfocados más a la acción del mismo.

El juego será 2D y tendrá una cámara top-down, de forma que el jugador pueda ver su entorno e interactuar con él. La cámara se asemejaría a la que podríamos ver en el videojuego “Golden Sun”. En cuanto a los combates, las mecánicas acordes con este tipo de juego serían las del combate por turnos, los enemigos y los jugadores se disputan un combate en el que cada cual dispone de un turno para elegir las acciones que desea realizar, una vez elegida, será el turno del enemigo hasta que el combate se dé por finalizado. Por último, los menús del juego se realizarán de forma que asemeje lo máximo posible a un sistema de RPG de mesa.

6.1.3.-Tipo de interacción juego-jugador

Las mecánicas con las que contara el jugador a la hora de interactuar con el juego serán las básicas de cualquier RPG, esto es: Combate contra enemigos por turnos, movimiento por la escena e interacción con la misma, sistema de inventario (utilización de consumibles en combate o fuera de él, equipamiento de armas o armaduras), sistema de personalización de personaje (sistema de puntos de características o habilidades) y creación de un entramado de misiones principales y secundarias que el jugador vaya avanzando a lo largo de la historia.

6.1.4.-Plataforma de destino

Por último, la plataforma de destino elegida para este videojuego será el PC, esto es, desktop. Se ha elegido esta plataforma puesto que concuerda más con el tipo de interacciones y controles que tienen en este género, el uso de ratón y teclado permitirá un control fácil sobre las diferentes pantallas que aparezcan a lo largo del juego, y el teclado podría hacerse uso para los diferentes “shortcuts” que podría tener el juego. Además, el público objetivo de este tipo de juegos está concentrado en PC y valora esta plataforma, permitiéndoles jugar desde su ordenador y dándoles más opciones para posibles añadidos (como modificaciones por parte de la comunidad).

Tal y como se mencionó a lo largo de las entregas, la opción de hacer el título para móviles no estará descartada de cara a futuro, pero la versión final de este proyecto únicamente estará para sobremesa.

6.2.-Conceptualización

6.2.1.-Historia, ambientación y/o trama

La historia se centra en el mundo de "Kor'Dul", un mundo de fantasía en el que conviven humanos, demonios, goblins y todo tipo de criaturas fantásticas. Es un mundo en el que la civilización humana ha quedado recluida a pequeños asentamientos autosuficientes en una sociedad feudal, esto se debe principalmente a que los reinos humanos de Kor'Dul reniegan de casi todo uso de la magia (salvo las runas mágicas), dándoles una desventaja frente a las hordas de demonios que amenazan con acabar con ellos.

La trama que concierne al jugador se centrará en la vida de un joven superviviente de un pueblo asolado por la legión de demonios. Una vez a salvo, este personaje, emprenderá una cruzada personal para evitar de cualquier forma que lo que le ocurrió en el pasado vuelva a repetirse de. Por el camino tendrá que realizar decisiones que podrían afectar a la trama por completo y descubrirá una invasión en ciernes de la horda de demonios por un general demoníaco.

6.2.2.-Definición de los personajes/elementos e interacción entre los mismos

El héroe desconocido

Este personaje será el que encarne el jugador a lo largo de la historia. Una persona callada (En las interacciones se podrá ver como suele dar respuestas cortas y concisas), disciplinada y con un odio extremo hacia su enemigo acérrimo. Este personaje, superviviente de una tragedia en la que perdió todo (Su hogar, su familia e incluso su nombre), que tratará de evitar una catástrofe similar a la que le sucedió en su pasado, quebrando si fuese necesario los tabús establecidos por la sociedad humana en el intento. Su única persona cercana en el entorno será la persona que le acogió cuando más lo necesitaba, el caballero Lucius. Con la idea de terminar con los demonios, partirá por el mundo de Kor'Dul en busca de volverse más fuerte e investigar los secretos del mundo para cumplir sus fines.

Lucius LawStroke

Antiguo caballero del reino, Lucius LawStroke será a la vez el único acompañante del personaje principal y a la vez el narrador de la historia. Acogió a este cuando lo encontró entre las ruinas de su poblado y una vez rescatado le enseñó el arte de la espada. Tendrá un papel fundamental dando una visión experimentada a el héroe, contribuyendo de esta manera con su propia visión del mundo a las decisiones morales que afecten al personaje.

La legión demoniaca

Si bien hasta hace relativamente poco los demonios únicamente aparecían esporádicamente en el mundo de Kor'Dul, su actividad reciente hace sospechar que podría estar ocurriendo un evento especial. Los demonios en este mundo de fantasía se representan como la encarnación del mal y el desorden, afectando a la fauna salvaje, las personas o incluso el clima. Estos poderosos entes harán las veces de enemigo en el juego, apareciendo en numerosas ocasiones en combates contra el jugador, y dotando de varios "capitanes" en su ejército que se enfrentarán al jugador como Jefes durante el transcurso de la historia.

6.2.3.-Objetivos planteados al jugador

El jugador tendrá un objetivo claro desde el inicio del juego: Derrotar a la legión demoniaca para poder consolidar la paz en el territorio humano y evitar la extinción del mismo. Si bien podría suponer un objetivo claro y sencillo, a lo largo de la trama se irán descubriendo multitud de tareas e investigaciones, ya que la corrupción demoniaca podría haber contagiado hasta los mismos cimientos de la humanidad. Tal y como se ha comentado anteriormente, el personaje tendrá una evolución de personaje, empezando como un aventurero con poca experiencia hasta terminar como un héroe como tal.

A su vez, durante la historia aparecerán diferentes decisiones que afectan a la propia trama. ¿Debería el héroe aprender a utilizar la magia contra sus enemigos? ¿O debería respetar las creencias humanas?, ¿Podría detener la invasión demoniaca en ese pueblo en el linde del territorio humano? Estas decisiones irán tomando forma conforme el jugador avance.

6.2.4.-Concept Art: Screenshots...

Tal y como se ha explicado, el mundo de Kor'Dul será un mundo de fantasía con asentamientos humanos extendidos por todo el territorio, rodeando a estos, zonas boscosas y montañas escabrosas llenas de monstruos. Un buen ejemplo sería el mundo de "Warhammer: Fantasy".

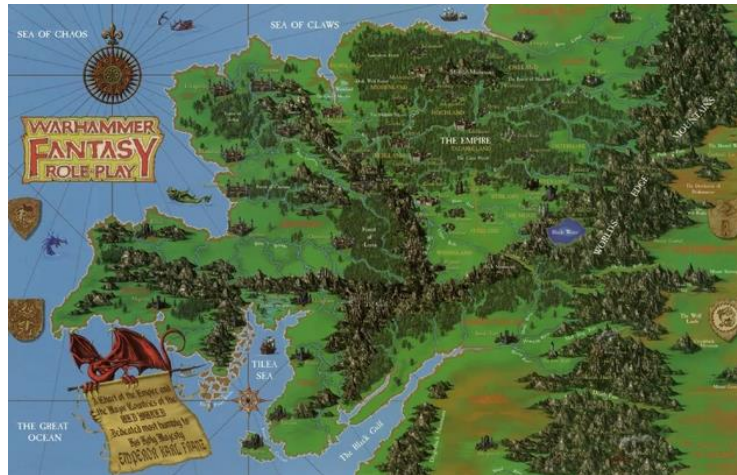


Figura 26: Mapa de territorio de Warhammer Fantasy.

Por otro lado, los pueblos humanos serian de pequeño tamaño y temática medieval europea. Coronando el imperio humano, Daturia, la capital del imperio, sería la sede de los guardias imperiales y la única gran ciudad de tamaño considerable.



Figura 27: Concept art de pueblo medieval europeo.

6.3.-Desarrollo y roadmap

6.3.1.-Planificación de Objetivos

Dado que un juego como el expuesto requeriría de un tiempo de desarrollo mayor al disponible para esta asignatura, a continuación, se procederá a señalar que apartados se realizaran y como se realizaran para su posterior planificación en el último punto del documento.

Los objetivos a realizar a lo largo de la asignatura serán:

- **Creación de los gráficos esenciales del videojuego:** Se crearán los gráficos esenciales para definir el entorno, los personajes, los enemigos y los objetos básicos del juego. Para ello, se ha decidido utilizar “PixelArt”, de forma que concuerde con el estilo y dinámica de juego.
- **Creación de dos niveles del juego:** La aventura que se presentará al finalizar la asignatura constará de dos niveles del juego y estará situada en un punto de la trama ligeramente posterior al inicio de la misma. Se diseñarán dos escenarios, un pueblo y un escenario de bosque.
 - El escenario del bosque tendrá la probabilidad de hacer aparecer enemigos conforme el jugador avance por el mismo, haciendo así un escenario de combate.
 - El escenario de ciudad permitirá al jugador acceder a misiones y a la tienda. Las calles tendrán pueblerinos con los que poder hablar.
- **Diseño de interfaz de juego y personaje:** Como todo juego este debe tener una interfaz sencilla y usable. La interfaz debería darnos acceso a las funciones básicas del juego y mostrar a su vez un resumen de las características del personaje principal.
- **Creación de una misión secundaria:** Las misiones son una de las principales mecánicas de los RPG. Durante la demo mostrada el jugador tendrá que ir al bosque, cazar unos monstruos, y volver con la prueba de que los ha vencido al pueblo.
- **Sistema de combate:** Para poder derrotar a los monstruos y obtener sus objetos, se implementará un sistema de combate en el que el jugador luche por turnos contra los enemigos hasta que uno de los dos muera.
 - Tal y como se mencionó durante el apartado de mecánicas, la complejidad del combate se realizará como “Golden Sun”.
- **Sistema de objetos:** Por último, se implementará un sistema de objetos y una tienda en el pueblo. Al terminar la misión el jugador podrá ir a la tienda a vender los objetos restantes o poder comprar algún objeto extra.

6.3.2.-Cuantificación de tiempo y recursos por objetivo

Al ser únicamente una persona y haber elegido un motor de juego gratuito sin más gastos de recursos que el tiempo que emplee en hacerlo, el único recurso que se utilizará en este apartado será el tiempo utilizado en el proceso. Dado que la propia asignatura estará dividida en dos grandes entregas, se dividirá entonces la carga de trabajo a realizar en esta misma división basándose en las exigencias de los mismos.

Para la primera entrega, tal y como se expone en el enunciado de la misma, se realizará el diseño de los dos niveles de juego (Con los gráficos PixelArt, tal y como se mencionó en el apartado de “objetivos”) y la interfaz básica del juego. Concretamente, el diseño de las escenas con el apartado gráfico se estima que se tardara las dos primeras semanas, y el diseño de la interfaz y su implementación la semana y media siguiente. Por último, se programará un “personaje” que actué como de jugador para poder viajar entre las dos escenas para la entrega de esta parte del proyecto en la media semana restante.

Al alcanzar la segunda entrega, la entrega jugable, estarán ya definidas las mecánicas de combate, el sistema de personaje con los puntos de características y el inicio del sistema de objetos. De esta forma, la escena del bosque estará completa, pudiendo demostrar la jugabilidad principal del juego en la misma. La primera semana se dedicará en la creación del sistema de personaje, los puntos de características y el balanceo de los mismos. Una vez creado este sistema, las próximas dos semanas serán las elegidas para realizar el sistema de objetos, armas, armaduras o pociones para el combate deberán de utilizarse para poder prosperar en el juego. Por último, en las últimas tres semanas antes de la entrega, se terminará el sistema de combate, acoplándose a las mecánicas ya creadas y dotando al juego de esta funcionalidad.

Para finalizar, en la última entrega se realizará el sistema de misiones y la tienda, terminando así con la funcionalidad esperada del videojuego en los 13 días restantes. Al terminar el tiempo, se realizará la entrega del proyecto y documentación, esta última, se ira realizando a lo largo del tiempo conforme se vaya complementando el proyecto.

7.-Diseño técnico

7.1.-Entorno de desarrollo

A la hora de desarrollar nuestro proyecto, además del motor de juego necesitaremos un entorno de desarrollo o IDE para poder codificar la parte lógica del juego.

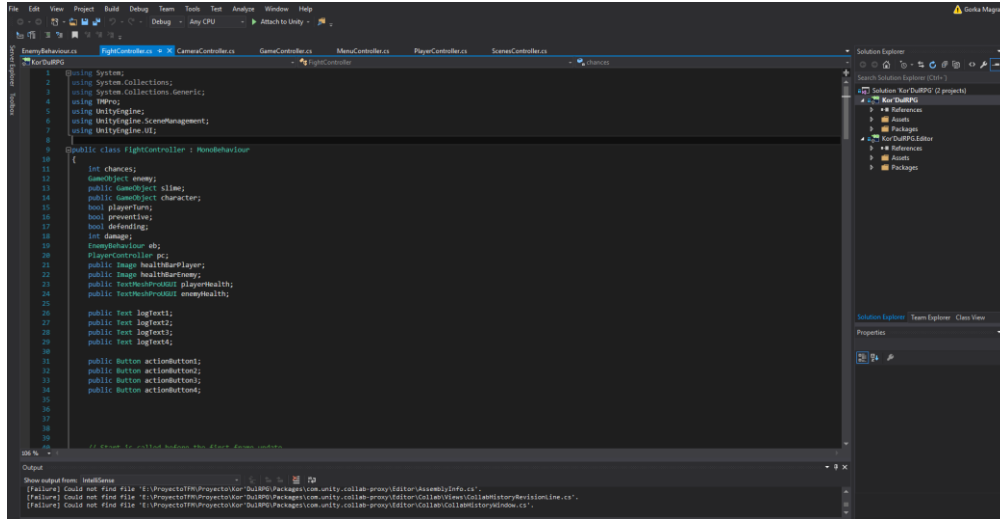


Figura 28: Imagen del IDE "Visual Studio".

Un IDE no solamente nos permite codificar en un determinado lenguaje de programación, también da ayuda al desarrollador con los paquetes, la importación, las herramientas de autocompletado y muchos más. La elección de un buen IDE por tanto puede suponer una gran ventaja a la hora de desarrollar un proyecto de grandes dimensiones, dando soporte al programador en aspectos que él no tendría que preocuparse.

Para este proyecto se ha elegido el entorno de desarrollo "Microsoft Visual Studio", la herramienta creada por Microsoft en 1997 con el fin de dar soporte a sus productos software. Las razones por las que finalmente se eligió este IDE fueron las siguientes:

- Es **compatible con el lenguaje de programación a utilizar**, ya que este entorno de desarrollo es compatible con múltiples lenguajes de programación, como C#, C++ o Java, entre otros.
- Es el **entorno de desarrollo por defecto para Unity**. Esto no solo significa que este IDE sea el predeterminado cuando instalas un sistema operativo Windows/Mac, si no que gran parte de la documentación oficial de la plataforma y la comunidad harán referencia a este entorno y, por tanto, será más sencillo sacarlo provecho.

- **La licencia es gratuita**, medida que afecta al apartado económico del videojuego y por lo tanto un plus extra a la hora de elegirlo como nuestra plataforma de uso.
- Gracias a las herramientas de la plataforma, como, por ejemplo, “IntelliCode”, **se facilita mucho la creación de código**, gracias a la IA que dispone para dar soporte a los desarrolladores.



Figura 29: Visual Studio

De la misma manera, Unity nos permite programar la lógica de nuestro videojuego en dos lenguajes de programación, C# o JavaScript. En este caso se eligió la primera opción por las siguientes razones:

- En primer lugar, y quizás la razón más importante sería el conocimiento del mismo, ya que al **haber realizado las practicas del postgrado íntegramente en este lenguaje**, supondría un esfuerzo y tiempo menor realizarlo en el mismo, pudiendo utilizar esos recursos en otros aspectos del proyecto.
- Por último, C# es con diferencia el **lenguaje más utilizado por la comunidad de unity**, utilizándose como código de referencia en la mayoría de los artículos y foros del mismo. De esta forma, se podría encontrar recursos con más facilidad en este lenguaje.

7.2.-Herramientas empleadas

El motor de juego y el entorno de desarrollo son herramientas fundamentales para poder realizar el proyecto, sin embargo, por las características del mismo y los requisitos especificados por el profesorado de la asignatura, no serán las únicas, a continuación, se procederá a explicar en detalle la función de las demás herramientas y por qué han sido necesarias para el proyecto.

7.2.1.-Piskel

Piskel es la herramienta utilizada para generar los diferentes diseños del videojuego en "PixelArt". Este software, que en principio puede utilizarse desde su versión de navegador, pero para el proyecto se utilizó la aplicación, nos dará el soporte necesario para poder realizar tanto sprites como animaciones e importarlos a Unity. Las características por las que se eligió esta herramienta fueron las siguientes:

- Tal y como y como se ha comentado, los sprites y animaciones realizadas en esta plataforma se pueden importar a GIF, PNG etc. Esto permite una fácil incorporación de los mismos a el proyecto.
- La curva de aprendizaje de la herramienta es asequible, ya que utiliza una interfaz sencilla que permite ver al diseñador todos los elementos de forma sencilla y utilizarlos cómodamente.
- La aplicación dispone de una licencia gratuita que nos permitirá sacar el máximo potencial a la misma sin interferir con los requisitos del proyecto ni los recursos económicos del mismo.
- El editor posee una vista en vivo, un añadido muy útil que nos permitirá ver cómo quedará nuestro sprite una vez este importado a Unity.



Figura 30:Imagen de un sprite en Piskel.

7.2.2.-SmartGit

Dados los requisitos del proyecto, la plataforma donde el profesorado de la asignatura nos señaló de subir el mismo fue en GitHub, de esta forma, el repositorio de archivos serviría para que los consultores tuviesen en todo momento un lugar desde el cual seguir el proceso del proyecto cuando quisieran, y como control de versiones del mismo, facilitando así el progreso del videojuego.

El software elegido para dar soporte a este requisito del proyecto fue SmartGit, una herramienta de Git gráfica que permitirá al usuario acceder a los repositorios desde la aplicación del cliente. Habiendo ya varias en el mercado (incluyendo SourceTree, la recomendada por el profesorado) se eligió SmartGit por las siguientes características:

- En primer lugar, como la mayoría de clientes Git, este abstrae al usuario de la complejidad propia de Git, con él, se ahorrarán los comandos por consola y se obtendrá una interfaz limpia y sencilla con el que poder organizar los repositorios.
- SmartGit funciona de la misma manera en cualquier plataforma, ya sea Linux, Windows o Mac, la interfaz será la misma y evitara cualquier tipo de fallo específico por plataforma, haciéndolo así mucho más versátil.
- Este software tiene integrado los servicios más comunes, como por ejemplo GitHub o BitBucket.



Figura 31:SmartGit

- SmartGit a su vez, provee de herramientas “Out-Of-The-Box” para los desarrolladores, como, por ejemplo, Git-Flow, SSH o resolución de conflictos.
- La licencia del software es gratuita para su uso no-comercial, reduciendo costes del proyecto en su apartado económico.

7.3.-Recursos utilizados

Una de las grandes premisas de este proyecto era realizarlo íntegramente utilizando assets generados por el propio alumno, por lo tanto, cabe recalcar de que todos los assets utilizados para este proyecto son propios y no se han aprovechado ninguno de terceros.

El proyecto utilizara un arte categorizado como "PixelArt" y su diseño y desarrollo se realizaron en distintas fases que se procederán a explicar a continuación. En cuanto al resto de media (Animaciones/sonidos) se decidió no profundizar en ellos, ya que integrarlos de manera propia aumentaría de forma exponencial el tiempo requerido para el proyecto.

Cabe decir que a las fases que se comentaran en el siguiente apartado se debería de añadir el estudio del arte realizado antes de comenzar a realizarlas, ya que el propio "PixelArt" dispone de diferentes técnicas de diseño, y el estudio y elección de una de ellas formo parte del aprendizaje de este apartado.

7.3.1.-Diseño de los Sprites

A la hora de diseñar los Sprites se definieron diferentes fases, mediante las cuales se obtendría un producto con el cual trabajar en la siguiente, refinando de esta manera el sprite final.

El primer paso sería la conceptualización de los sprites, tal y como se explicó a lo largo del curso, los primeros bocetos a lápiz serian el material sobre el cual los diseñadores más adelante darían forma a nuestros assets, indicando de manera sencilla la idea principal de los mismos.



Figura 32: Bocetos de diseño

Tras realizar los diseños se empezó a crear los primeros sprites en la herramienta piskel. La idea era la creación de los diferentes objetos del escenario siguiendo un estilo y formato, de forma que el jugador encontrase, por ejemplo, patrones en los tipos diferentes de edificios basados en el color de su tejado, o los diferentes caminos disponibles en base a el posicionamiento de los mismos.



Figura 33:Diseño de un edificio en Piskel

Por último, los diseños se introdujeron en el propio mapeado de niveles, cambiando las estructuras de los mismos en los casos que fuesen necesarios. Tras organizarlos y añadir adornos al mapeado, el nivel quedaría de la siguiente manera.



Figura 34:Imagen de la escena "Town".

7.3.2.-Diseño de la interfaz

Para el apartado de la interfaz se utilizó un proceso de diseño similar al ya comentado anteriormente. Tal y como se vio a lo largo del curso, se decidió utilizar el software “Invision” para hacer un esquema sobre los posibles detalles y el flujo de ventanas en la interfaz.



Figura 35: Boceto en Invision de la Interfaz de usuario.

Tras establecer un orden de ventanas y organizar las mismas para que siguieran los principios de usabilidad, se empezaron a realizar los diseños de interfaz en piskel. Dado que el juego sería en forma pixelart, se decidió realizar también los menús siguiendo este estilo artístico, aplicando formas sencillas y reconocibles para los diferentes iconos.



Figura 36: Imagen de la interfaz dentro del juego.

7.4.-Esquema de arquitectura del juego

Como cualquier software, el videojuego tendrá su propia estructuración, la forma en la que los componentes interactúan entre sí para dar una experiencia de juego agradable al jugador, la forma en la que está dispuesta la actividad del jugador cuando este juega a nuestro juego o la forma en la que interacciona el propio jugador con el programa.

En primer lugar, cabe decir que en la versión actual el videojuego no dispone de ninguna manera de almacenar los datos, ni tiene ninguna relación con ningún elemento externo aparte del propio jugador, de forma que un diagrama estructural de la relación entre estos elementos sería demasiado simple.

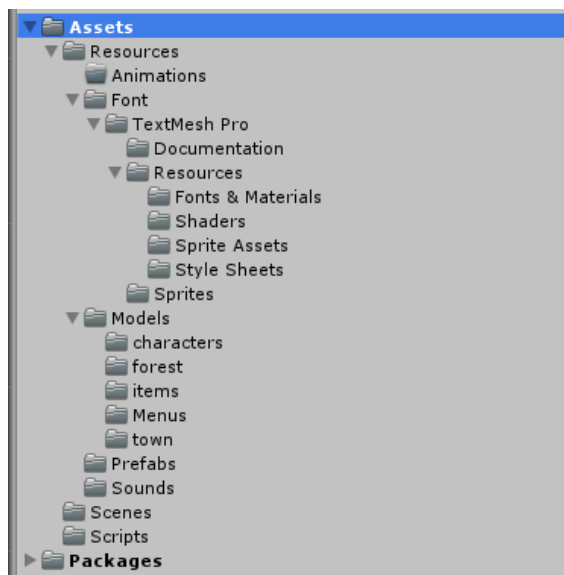


Figura 37: Estructura del proyecto en Unity.

La estructura del propio proyecto, por otro lado, se ha ido cuidando conforme se desarrollaba el mismo. Las tres grandes carpetas del archivos (Recursos, Escenas y scripts) lograran simplificar al maximo las tareas de actualizacion del mismo, ademas, dentro de la carpeta de recursos se podran localizar facilmente los diferentes assets generados, ya sean sonidos, animaciones prefabs o modelos.

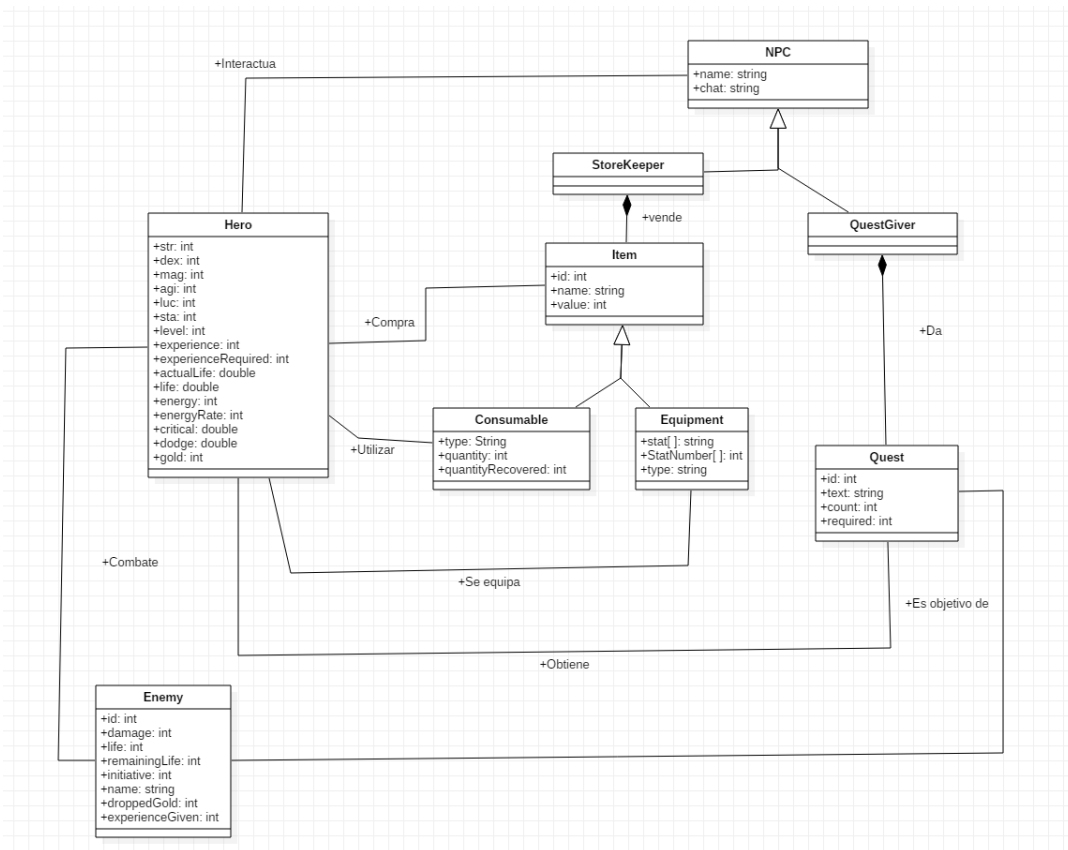


Figura 38:Diagrama de clases del proyecto.

Como vemos, la estructura de los elementos del videojuego y su relacion entre los mismos sera la presentada en la figura superior. Tal y como vemos en el mismo, en la version actual del proyecto el jugador podra interactuar con los NPC's vendedores para comprar objetos, con los NPC que dan misiones para recibirlas, y combatir enemigos ya sea para obtener botin o completar misiones.

Por ultimo, la Inteligencia artificial presentada en esta version del videojuego no sera de gran relevancia, ya que los enemigos unicamente atacaran al jugador. Una vez se desarrollen las versiones posteriores, los enemigos tendran probabilides de hacer diferentes tipos de habilidades o atacar a otros enemigos, llegando incluso a tomar ciertas decisiones en base al nivel de dificultad.

8.-Diseño de niveles

8.1.-Escena del pueblo

En la escena del pueblo se congregará gran parte de las funcionalidades del juego (Salvo las del combate, por ser una zona segura), y por ello, tendrá que tener un diseño que permita aprovechar al máximo su espacio y exponer las diferentes posibilidades al jugador.



Figura 39:Esquema de la escena "Town".

Como podemos ver, el diseño de nivel del pueblo está realizado de forma que el usuario pueda ver los diferentes edificios y sus funcionalidades, destacando el color de los tejados para ello.

Dicho esto, las misiones, la principal funcionalidad de las zonas seguras, podrán venir de edificios con tejados rojos, como la caseta del guardia (1) o la Iglesia (3), o de edificios con el tejado verde, como la casa de Lucius (2) o la posada (4).

Por otro lado, las zonas seguras también contarán de vendedores, generalmente marcados con techos azules, como la herrería y la tienda general (7), estos ocasionalmente podrían dar alguna misión, pero su principal funcionalidad será de actuar como vendedores para el jugador.

Por último, las zonas de ciudad, al igual que el resto de las escenas, tendrán conexiones con el resto de las escenas en los bordes de las mismas, ya sean caminos abiertos (disponibles), como por ejemplo el camino a los bosques cercanos (5), como caminos que aún no están disponibles y requerirán avanzar en la historia para desbloquear, como por ejemplo el carro mercante varado en el camino (6).

8.2.-Escena del bosque

El otro tipo de escenas de las que dispondrá el videojuego serán las escenas de “combate” o peligrosas, en ellas, el jugador podrá enfrentarse a los diferentes enemigos del juego, resolver pequeños puzles y hacer diferentes misiones con el fin de avanzar en la historia principal o moverse de ciudad en ciudad.

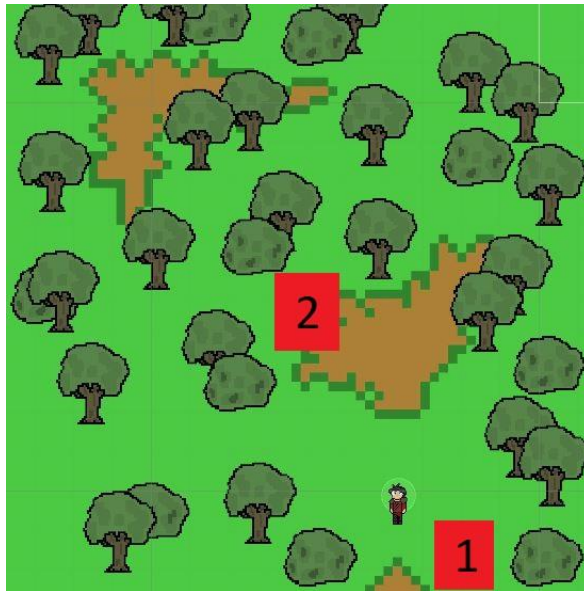


Figura 40:Esquema de la escena "Forest".

El funcionamiento de estas escenas será mucho más sencillo. Para poner de ejemplo tendremos la escena de bosque, que para la demo únicamente servirá como “zona de combate”, en ella, podremos encontrar enemigos por todo (2) el mapa, con una probabilidad de triggerear el evento mientras caminemos a través de él.

Además, y como ya se comentó en la anterior escena, todas las pantallas de este videojuego estarán conectadas por sus extremos, iniciando un cambio de escena si el jugador, por ejemplo, decidiese acercarse al límite con el pueblo (1).

9.-Manual de usuario

9.1.-Requisitos hardware mínimos

Dada la simplicidad de los gráficos y assets utilizados (Recordemos que es un juego 2D top-down, sin físicas y con sprites en PixelArt) y la relativa poca carga lógica que necesita, las especificaciones mínimas serán las siguientes.

Procesador: 2GHz.

Memoria: 2 GB de RAM

Gráficos: 256 MB memoria de video.

Almacenamiento: 110MB

9.2.-Manual de usuario

El juego, al tener como plataforma objetivo el PC, dispondrá de controles para teclado y ratón, de forma que la mayoría de ventanas y menús se puedan interactuar mediante el uso de un puntero (tanto fuera como dentro de combate). En cuanto al uso de teclado las teclas por defecto serían las siguientes:

- W / tecla “arriba”: Mover el personaje hacia arriba.
- A / tecla “izquierda”: Mover el personaje a la izquierda.
- S / tecla “abajo”: Mover el personaje hacia abajo.
- D / tecla “derecha”: Mover el personaje a la derecha.
- E: Interactuar para hablar con un NPC.
- Q: Interactuar para abrir la tienda de un NPC.

Dado que el videojuego estará en inglés, la forma de empezar la partida será hacer click en “New Game” o “Exit Game” para salir. Una vez dentro del juego, hacer click en los diferentes menús nos permitirá navegar por las diferentes funcionalidades, principalmente concentradas en la parte inferior derecha de la pantalla (Tanto dentro de combate como fuera)

10.-Conclusiones

10.1.-Conclusiones del trabajo realizado

Una vez terminado el proyecto y echando la vista hacia atrás se pueden sacar reflexiones sobre lo ocurrido a lo largo de las entregas y los objetivos que se han ido cumpliendo durante los plazos establecidos, así como una conclusión final sobre lo aprendido y conseguido.

En primer lugar, los objetivos necesarios para el proyecto se cumplieron en su mayoría, de forma que únicamente no se crearon para la entrega final las animaciones personalizadas del personaje. La razón detrás de la falta de estas animaciones es el coste exponencial de crear los sprites para las mismas. Ya que la creación de los gráficos ya supuso de por si una demora en el resto de objetivos del proyecto, realizar el resto de sprites para animar a los personajes habría dado como resultado la falta de alguna de las demás funcionalidades de jugabilidad del videojuego. Por el contrario, el resto de los objetivos se completaron satisfactoriamente según lo especificado en la primera entrega.

La planificación se ha seguido acorde a lo especificado en la primera entrega, si bien el sistema de combate llevo más de lo esperado, y por tanto, el sistema de inventario y consumibles se retrasó hasta la última entrega. Para garantizar el éxito del trabajo se tuvo que finalizar con la creación de sprites tras la segunda entrega, tras recibir un aviso por parte del profesorado en el informe de autoevaluación, ya que si se hubiera seguido trabajando en el apartado grafico tal y como se había previsto, el juego tendría una falta de funcionalidades en su versión final.

Como conclusiones y lecciones aprendidas se puede sacar la experiencia obtenida durante todo el proceso de creación del producto. La dinámica de creación del apartado grafico e inclusión en el videojuego, así como el diseño de niveles con los sprites realizados. También la gestión de varios apartados del juego simultáneamente y la programación escalar de cara a mejorar el producto a futuros. Por último, también ha supuesto una grata experiencia el aprendizaje realizado respecto a la creación de PixelArt, el estudio del mismo y los diseños realizados supusieron una mejora a la hora de crear videojuegos, ya que, al tener un perfil de programador, no se había desarrollado anteriormente en esa faceta.

10.2.- Líneas de trabajo futuro

Si bien el producto final para la entrega ya se realizó, el desarrollo del videojuego seguirá en un futuro, teniendo unos objetivos claros a realizar en corto-medio plazo, los cuales serán los siguientes:

- Creación de un sistema de almacenamiento de datos, haciendo posible que el jugador guarde la partida, la cargue y mantenga sus datos en el mismo.
- Sistema de habilidades mejorado, ya que en el sistema actual únicamente se podría atacar y defenderse, aunque la base del sistema de habilidades, las estadísticas, este ya implementado dentro del juego.
- Sistema de equipamiento, añadiendo huecos para introducir armaduras, armas y accesorios con el que poder complementar el sistema de estadísticas, pudiendo así personalizar el personaje completamente.
- Mejorar el apartado de diseño del juego. La idea principal del juego era incluir también sonido y animaciones propias, pero el tiempo disponible no permitió este apartado.
- Soporte para múltiples personajes / enemigos. El sistema actual se nos presenta con una dinámica de un jugador contra un enemigo, pero el diseño de la narrativa del videojuego precisara que nos tengamos que enfrentar a varios enemigos, y, por ende, el héroe desconocido necesite ayuda de otros aventureros.
- Por último, la creación de la historia y las subsecuentes escenas de la misma, construyendo una historia rica y atractiva para los jugadores, además de una cadena de misiones que mantenga el interés de los usuarios e introduzca la historia a los mismos.

11.-Glosario

- **IDE:** Del inglés, Integrated Development Environment. Son las siglas de entorno de desarrollo integrado, la aplicación informática que dan soporte al desarrollador.
- **RPG:** Role Playing Game, juego de rol, en el que los personajes encarnan a un héroe o héroes en una aventura y el desarrollo de la misma.
- **ARPG:** Action Role Playing Game, un juego de rol de acción, en el que la dinámica de combate es fundamental y que normalmente se detalla en tiempo real.
- **Sprite:** Se trata de un diseño realizado para videojuegos basado en un mapa de bits.
- **Asset:** Recurso utilizado para la creación de un videojuego.
- **NPC:** Non-Player Character, personaje no jugador que aparecerá a lo largo del juego, pero que, por razones de diseño, no podrán ser utilizados por el usuario.

12.-Bibliografía

Tom Wijman. (2018). Mobile Revenues Account for More Than 50% of the Global Games Market as It Reaches \$137.9 Billion in 2018. 10/06/019, de Newzoo Sitio web: <https://newzoo.com/insights/articles/global-games-market-reaches-137-9-billion-in-2018-mobile-games-take-half/>

Statista. (2019). Genre breakdown of video game sales in the United States in 2017. 10/06/2019, de Statista Sitio web: <https://www.statista.com/statistics/189592/breakdown-of-us-video-game-sales-2009-by-genre/>

Glen Canessa. (2018). Una metodología ágil para las mejoras SAP. 10/06/2019, de Novis Sitio web: <https://www.novis.cl/noticias-novis/servicios-sap-2/una-metodologia-agil-para-las-mejoras-sap/>