

BIG MONEY

Autor: Carlos Diéguez Alonso

Tutor: Raúl Montoliu Borrás

Profesor: Joan Arnedo Moreno

Ingeniería Informática

Computación

Fecha de entrega 14/01/2024



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>Big Money</i>
Nombre del autor:	<i>Carlos Diéguez Alonso</i>
Nombre del colaborador docente :	<i>Raúl Montoliu Borrás</i>
Nombre del PRA:	<i>Joan Arnedo Moreno</i>
Fecha de entrega (mm/aaaa):	<i>01/2024</i>
Titulación o programa:	<i>Ingeniería Informática</i>
Área del Trabajo Final:	<i>Videojuegos</i>
Idioma del trabajo:	<i>Castellano</i>
Palabras clave	
Resumen del Trabajo (máximo 250 palabras): <i>Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo</i>	
<p>El objetivo de este proyecto es crear un videojuego, generando todo el contenido y código necesario para lograr un producto funcional. El tiempo de producción del proyecto está limitado a dos meses, por lo que se planea continuar con una segunda iteración del juego final diseñado para lograr un producto completo. El género del juego es First Person Shooter y Boomer Shooter como subgénero. Se lanzará para PC y se recomienda jugarlo con ratón y teclado, o también es posible con un mando. En el documento y su anexo, se explican en detalle todas las fases del diseño y construcción de los juegos. El proyecto está planeado como la primera iteración de un modelo Scrum. La producción final, el arte, el código y el documento de diseño del juego se mejorarán en las próximas iteraciones. El Documento de Diseño del Juego está incluido como anexo.</p> <p>El producto final se puede jugar descargando los archivos o viendo un video del gameplay.</p>	
Abstract (in English, 250 words or less):	
<p>This project goal is to create a video game, creating all content and code included to achieve a working product. The production time of the project is limited to two months so, it is planned to be continued on a second iteration of the final game designed to get a complete product.</p> <p>The game genre is First Person Shooter and Boomer Shooter as subgenre. It is released for PC and it</p>	

is recommended to be played with mouse and keyboard, or also possible with a pad controller.

In the document and its annex, all phase of the design and construction of the games are explained in detail.

The project is planned to be the first iteration of a scrum model. The final production, art, code and the game design document will be enhanced in next iterations. The Game Design Document is included as an annex.

The final product can be played downloading the files or watching a video of the gameplay.

Enlace al repositorio Git:

<https://github.com/CDieg/BigMoney>

"It's time to kick ass and chew bubblegum and I'm all outta gum" - Duke Nukem (Duke Nukem 3D video game)

Agradecimientos

A mi familia, a la que les debo lograr llegar hasta aquí.

Abstract

This project goal is to create a video game, creating all content and code included to achieve a working product. The production time of the project is limited to two months so, it is planned to be continued on a second iteration of the final game designed to get a complete product.

The game genre is First Person Shooter and Boomer Shooter as subgenre. It is released for PC and it is recommended to be played with mouse and keyboard, or also possible with a pad controller.

In the document and its annex, all phase of the design and construction of the games are explained in detail.

The project is planned to be the first iteration of a scrum model. The final production, art, code and the game design document will be enhanced in next iterations. The Game Design Document is included as an annex.

The final product can be played downloading the files or watching a video of the gameplay.

Índice

Figuras y tablas.....	10
1. Introducción	11
1.1. Introducción/Prefacio	11
1.2. Descripción/Definición	12
1.2.1 Concepto	12
1.2.2 Referencias.....	15
1.3 Objetivos generales.....	19
1.3.1 Objetivos principales.....	19
1.3.2 Objetivos secundarios.....	19
1.4 Metodología y proceso de trabajo	20
1.5 Planificación.....	21
1.6 Presupuesto	23
1.7 Estructura del resto del documento	24
2. Análisis de mercado	25
2.1. Género del juego.....	25
2.1.1 <i>Boomer shooter</i>	26
2.2. Público objetivo y perfiles de usuario.....	26
2.3. Competencia	28
2.4. Análisis DAFO.....	31
3. Propuesta	33
3.1. Definición de objetivos/especificaciones del producto	33
3.2. Modelo de negocio	33
3.3. Estrategia de marketing	35
4. Diseño.....	37
4.1. Entorno.....	37
4.2. Requisitos técnicos del entorno de desarrollo.....	38
4.3. Inventario de herramientas utilizadas.	40
4.4. Inventario de assets y recursos.....	41
4.5. Arquitectura del juego.....	43
4.6. Funcionamiento de la IA.....	45
4.7. Diseño de niveles.....	46
5. Implementación.....	48
6. Demostración.....	49

7. Conclusiones y líneas de futuro	50
Bibliografía	51
Anexos.....	52
Glosario.....	52
Entregables.....	52
Capturas de pantalla	53

Figuras y tablas

Índice de figuras

Ilustración 1: Gráfico facturación de videojuegos de Pwc.....	11
Ilustración 2: Imagen conceptual generada con IA.....	12
Ilustración 3: Doom de ID Software.....	13
Ilustración 4: Sistema de niveles.....	13
Ilustración 5: Concepto de diseño de niveles.....	14
Ilustración 6: Pantalla de subida de nivel de Superior	14
Ilustración 7: Smash TV Arcade game, Williams 1990.....	15
Ilustración 8: Fotograma de The Running Man (1987).....	16
Ilustración 9: Mirro's Edge, DICE 2008.....	17
Ilustración 10: Gantz movie, 2011.....	18
Ilustración 11: Imagotipo oficial de Unity.....	20
Ilustración 12: Diagrama de Gantt.....	22
Ilustración 13: Captura del juego Painkiller.....	28
Ilustración 14: Captura del juego Doom Eternal.....	29
Ilustración 15: Captura del juego Dusk.....	30
Ilustración 16: Captura del juego Ultrakill.....	30
Ilustración 17: Discos <i>shareware</i> de Doom.....	33
Ilustración 18: Promoción de PUBG con FFC.....	34
Ilustración 19: Efectos de sonido del proyecto en Unity.....	41
Ilustración 20: Música de Karl Casey utilizada en Unity.....	41
Ilustración 21: Vista aérea del nivel 01.....	46
Ilustración 22: Sala interior del nivel 01.....	47
Ilustración 23: Vista desde el suelo de la sala interior del nivel 01.....	47
Ilustración 24: Controles del teclado.....	49

Índice de tablas

Tabla 1: Presupuesto.....	24
---------------------------	----

1.Introducción

1.1. Introducción/Prefacio

Los videojuegos nacen en la década de los cincuenta. En sus inicios, la tecnología que los soportaba se encontraba aún en una era temprana, sin madurar, a la vez que sus seguidores, que principalmente se movían en círculos universitarios y eran minoritarios.

Con el tiempo y la evolución, la industria de los videojuegos ha conseguido alzarse como la más en forma del sector del entretenimiento, no solo en números totales de facturación, como en porcentaje de usuarios. Con su continuo crecimiento se estima que en 2027 tenga unos ingresos de 312.000 dólares (Pwc, *Global Entertainment and Media Outlook 2022-26*). La ilustración 1 muestra el continuo crecimiento en ingresos globales de los videojuegos.

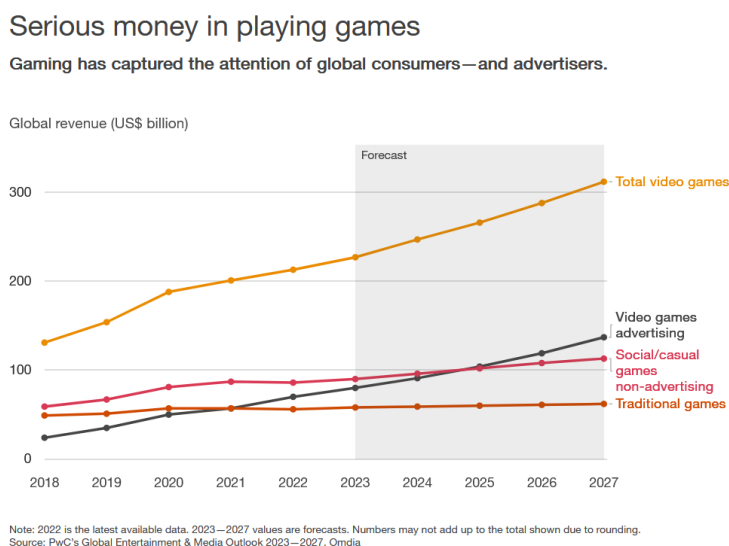


Ilustración 1: Gráfico facturación de videojuegos de Pwc
<https://www.pwc.com/gx/en/industries/tmt/media/outlook/insights-and-perspectives.html>

Pero la influencia de los videojuegos no es únicamente económica, sino que también es muy influyente desde el punto de vista artístico. Multitud de juegos están traspasando la barrera jugable de las consolas/ordenadores, y podemos encontrarlos en comics, películas, series o incluso en parques de atracciones.

El desarrollo de un videojuego es una tarea compleja, cubre todas las fases de desarrollo de software tradicional y además engloba múltiples roles de diferentes sectores (artistas, audio, video, marketing, producción, publicidad, redes sociales, infraestructura IT...).

Por esto, la realización de un videojuego como proyecto final de grado por una sola persona, se plantea como una tarea muy compleja, a la vez que interesante por el aprendizaje obtenido durante el proceso de creación.

1.2. Descripción/Definición

El proyecto parte de cero y engloba todas las fases de producción de un videojuego, aunque limita a un solo ciclo su ejecución y perfeccionamiento. El producto final aportado con este trabajo es un videojuego jugable con un set de niveles completo.

1.2.1 Concepto

Respecto a la trama, se centra en un programa de televisión distópico en el año 2080, en el que un concursante debe abrirse camino a tiros hacia la fortuna como única forma de escapar de la pobreza; el programa se emite por internet a todo el mundo. El objetivo del concurso es ganar o morir, el concursante debe pasar por distintos retos en los que debe sobrevivir y eliminar a todos sus enemigos para seguir jugando. El juego termina una vez es eliminado el jugador o consigue superar todos los retos. La ilustración muestra un concepto inicial del diseño del concursante creado con Midjourney.



Ilustración 2: Imagen conceptual generada con IA

Mecánicas de juego

El género de este videojuego es **Acción-FPS** con mecánicas de **Rogue-lite** y construido en un entorno 3D.

FPS son juegos en primera persona de disparos. Destacan por su alta inmersibilidad al usar una cámara que proporciona una perspectiva en primera persona de la vista del jugador en un entorno 3D. Es un género que nació en los noventa con el gran éxito de masas *Doom* y su predecesor *Wolfenstein 3D*, que crearon los arquetipos del género. En este artículo del medio CBR habla sobre ello: <https://www.cbr.com/wolfenstein-first-person-shooter/>

En la ilustración 3 podemos ver una imagen del juego Doom.



Ilustración 3: Captura del juego Doom
<https://www.britannica.com/topic/Doom>

Rogue-lite es un subgénero de videojuego que se puede considerar una evolución de los **roguelike**. Aunque no tiene características propias con la que sea más fácil detectar su género, posee varias de los *roguelike* como la muerte permanente, niveles aleatorios y evolución del personaje. Se hizo tan popular el uso de estos sistemas que llegaron a tener su propio género creando el *rogue-lite*. El género *roguelike* es un subgénero de los juegos de rol caracterizado por sus niveles de mazmorras generadas proceduralmente y juego por turnos.

En este proyecto, el jugador se enfrenta a diferentes oleadas de enemigos por niveles, que debe eliminar en el menor tiempo posible para obtener la mayor puntuación. Una vez superado un nivel pasará al siguiente donde de nuevo comenzará otra oleada, continuando así hasta llegar al nivel final donde el jugador puede superar el concurso como ganador.

La ilustración 4 nos muestra un concepto inicial del diseño de niveles para el juego.

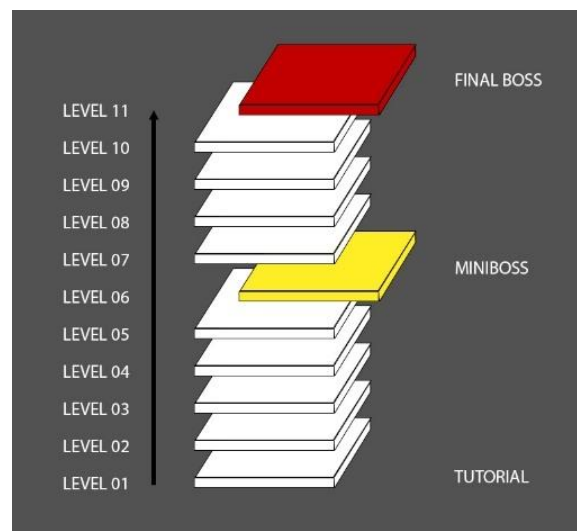


Ilustración 4: Diseño del sistema de niveles.

Para simplificar el desarrollo, los distintos niveles se basan en una misma estructura al igual que en un edificio, en las que solo varían los elementos estructurales que componen el interior. En la ilustración 5 podemos ver un ejemplo del diseño de nivel.

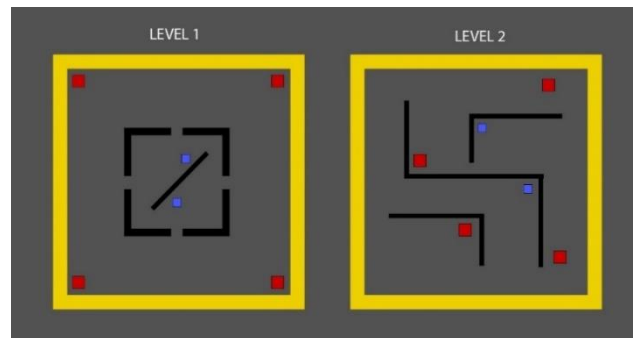


Ilustración 5: Concepto de diseño de niveles.

Al finalizar los distintos niveles el jugador puede elegir entre varias opciones para mejorar las características del personaje, como por ejemplo la potencia de disparo o su velocidad. Dependiendo de las elecciones aleatorias que aparezcan, permite al jugador adaptar el *gameplay* a su preferencia, o por el contrario tener que ajustarse a otro modo de jugar al no conseguir la evolución del personaje buscada. Esto ayuda a crear una sensación de juego diferente al comenzar cada partida. La ilustración 6 nos muestra un ejemplo del juego 'Superior' de una pantalla de mejora.

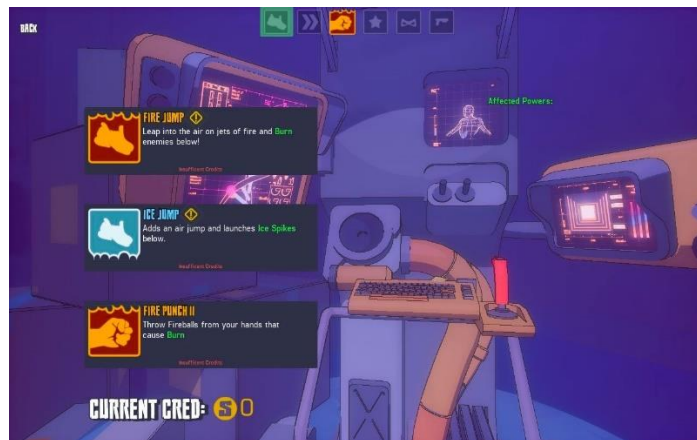


Ilustración 6: Captura del juego Superior que muestra la pantalla de subida de nivel.
<https://www.playtoearn.online/2022/03/11/superior-preview-of-gala-games-upcoming-roguelite-developed-by-drifter/>

Por otro lado, la dificultad del juego aumenta progresivamente al superar niveles, forzando al jugador a aprender distintas estrategias para poder superarlos, y equilibrar el juego a la subida de niveles del personaje.

1.2.2 Referencias

Historia

En videojuegos, **Smash TV** es la referencia principal de este proyecto. Se basa en un programa de televisión con público en directo, que emite el concurso donde uno o varios participantes luchan por sobrevivir entre mazmorras.

La ilustración 7 nos muestra una captura del juego.



Ilustración 7: Captura de Smash TV Arcade game, Williams 1990.
<https://www.classic-games.net/genesis/super-smash-tv/>

En libros, en 1982 se publica **The Running Man** de **Stephen King**. Trata sobre un concurso televisivo en el que el público consigue dinero si es capaz de matar a los concursantes. Está localizado en un distópico 2025, emitido desde Estados Unidos.

En 1987 se presentó la película **The Running Man** basada en el libro homónimo. Dirigida por **Paul Michael Glaser** y protagonizada por **Arnold Schwarzenegger**, nos muestra el mismo mundo distópico en el que programas televisivos violentos son emitidos para pacificar a los ciudadanos. Los concursantes son criminales que luchan por su vida huyendo de mercenarios armados. El ganador del concurso consigue el perdón del gobierno. En la ilustración 8 podemos ver un fotograma de la película.



Ilustración 8: Fotograma de The Running Man (1987).
<https://www.doblu.com/2022/11/10/the-running-man-4k-uhd-review/>

Gameplay

Muchos juegos *FPS* o *Third-Person Shooter* inspiran y construyen el *gameplay* de este proyecto. De cada uno toma alguna característica o es inspirado por ellas.

- **Smash TV:** concepto de niveles. El juego consiste en una sucesión de niveles por separado. Es necesario acabar con todos los enemigos de un nivel para pasar al siguiente.
- **Returnal:** contraste de los enemigos y peligros. Este juego destaca por la facilidad de distinguir los enemigos, sus disparos y diferentes peligros. Al usar colores sobresaturados que contrastan con el ambiente, el jugador tiene más fácil distinguirlos para poder esquivarlos en la mínima fracción de tiempo.
- **Doom:** respuesta de los enemigos, diseño de niveles. Dependiendo del tipo de arma que utilice el jugador, los enemigos reaccionan de distinta forma pudiendo, por ejemplo, activar una animación que deja al enemigo inmovilizado durante una fracción de tiempo. El diseño de niveles busca la sorpresa del jugador en dónde y cómo se va a encontrar a los distintos enemigos.
- **Devil Daggers:** simplicidad de escenario. La simplicidad del escenario, en este caso un círculo vacío, centra al jugador solo y únicamente en los enemigos, eliminando cualquier distracción y enfocando el juego en la habilidad de esquivar y atacar.

- **Tarkov:** diseño de entrada y salida de los niveles. Este juego ha hecho famoso el estilo de juego llamada '*extraction shooter*' (juego de disparos y extracción), en el que el jugador entra al nivel y tiene que conseguir salir después de haber terminado con su misión. El hecho de salir del nivel se convierte así en otra prueba.
- **Assassins Creed Valhalla:** modo *roguelite*. Este juego ha modernizado el concepto inicial de avanzar por niveles, añadiendo una progresión de las habilidades del jugador, las cuales pueden ser elegidas respecto a varias aleatorias ofrecidas. Así el jugador puede ir adaptando su estilo de juego según sean estas habilidades.
- **Resident Evil 4:** finalización de enemigos. Un concepto que cambia la jugabilidad del juego es el finalizar algunos enemigos una vez su vida ha bajado al mínimo. Esto fuerza al jugador a acercarse y no jugar en modo francotirador desde la distancia.

Arte

Distintos juegos y películas han servido como inspiración para dotar al juego de sus características especiales.

- **Juegos:**

Mirror's Edge.

Los escenarios se basan en polígonos muy simplificados (*low-poly*) y utiliza colores llamativos para orientar al jugador y facilitar su comprensión del nivel. En la ilustración 9 podemos ver un fotograma del juego.



Ilustración 91: Captura de Mirror's Edge, DICE 2008.
<https://wall.alphacoders.com>

Marathon.

El juego aún no ha salido a la venta, pero *trailer* de anunciamiento deja ver el arte. Mezcla entornos neutros con distintos colores sobresaturados, que le dan un aspecto único y facilita la visión de los distintos elementos en el juego.

Overkill.

Es la referencia de juego *Boomer Shooter* que muestra la base de realizar un juego de aspecto retro. Los modelos 3D y el arte agresivo simulan los realizados en la época de los 90.

- Audiovisual:

La inspiración artística de los proyectos audiovisuales se basa en tomar distintos elementos como son la composición, los colores, el diseño o el ritmo. La ilustración 11 muestra un fotograma de una de las películas de Gantz.

Akira

Blame!

Attack on Titans

Kavinsky

Gantz



Ilustración 2: Captura de Gantz movie, 2011.
<https://comicsalliance.com/gantz-movie-review/>

1.3 Objetivos generales

1.3.1 Objetivos principales

Objetivos de la aplicación/producto/servicio:

- El objetivo principal de este proyecto es realizar un videojuego funcional.
- Planificación de todas las fases de creación.
- Publicación de un juego.

Objetivos para el cliente/usuario:

- Diseñar una experiencia que sea entretenida.
- Las mecánicas son accesibles y satisfactorias.

Objetivos personales del autor del TF:

- Experiencia de aprendizaje de realizar todas las fases de desarrollo de un videojuego.
- Aprendizaje de Unity y C#.
- Oportunidad de usar creatividad en un proyecto personal.

1.3.2 Objetivos secundarios

Objetivos adicionales que enriquecen el TF.

- Establecer una ruta de actualizaciones y mejoras para la continuación del proyecto una vez presentado.

1.4 Metodología y proceso de trabajo

La estrategia para llevar a cabo el proyecto es la realización de un producto nuevo. Adaptar un producto existente requeriría de un tiempo extra de investigación del proyecto elegido del que no disponemos.

Respecto al apartado artístico, se ha simplificado teniendo en cuenta el tiempo y los recursos de los que se disponen para la realización del proyecto.

El motor gráfico elegido para este proyecto es **Unity**. El principal motivo para su elección es la simplicidad de la curva de aprendizaje respecto a otros, desde el punto de vista de los desarrolladores. Es una aplicación multiplataformas que permite un rápido ciclo de desarrollo.



Ilustración 3: Imagotipo de Unity.
<https://unity.com/>

Otro de los motivos fundamentales en su elección es la comunidad en internet. Es sencillo y rápido encontrar multitud de documentación de diferentes características del software, o incluso respuestas en foros similares a la pregunta buscada.

Otra característica importante es que para este tipo de proyectos *Unity* es gratuito. Su modelo permite utilizarlo mientras el juego genere menos de 200.000 dólares en ingresos.

Como otra opción importante en el sector tenemos el motor gráfico de *Unreal* que, aunque tiene herramientas y está orientado a proyectos 3D con una calidad de acabado excepcional, necesita más recursos y tiempo de desarrollo del proyecto, que no se ajustan al disponible. Este motor fue originalmente desarrollado para juegos FPS por la compañía *Epic Games*.

Como IDE (*Integrated Development Environment*) se utiliza *Visual Studio*, gratuito y compatible con C#.

Respecto a programas para edición de 3D, *Blender* es una herramienta que con los años está ganando su parte en el sector profesional, siendo una aplicación de código abierto. Ser de código abierto y estar a la altura de sus competidores de pago es el principal motivo de su elección. Cuenta con todas las funciones necesarias para generar los elementos 3D que necesitamos en el proyecto.

Para la edición de gráficos e imágenes, *GIMP* es un programa *open-source* gratuito. Profesionalmente no es habitual encontrarlo, pero para este proyecto, en el que no se plantean muchas texturas, es adecuado.

Para la edición del audio se buscan librerías gratuitas y se editan con el software *Audacity*, gratuito y de código abierto.

Para el control de versiones se utiliza *Sourcetree* que es un software gratuito basado en *Git*.

1.5 Planificación

Para la planificación de este proyecto nos centramos en los 5 grandes hitos, que marcan las fechas claves de entrega de evaluación continua. En general, la primera PEC nos da la conceptualización del proyecto. La segunda y tercera contienen la realización del juego en sí. La cuarta se basa en la documentación y preparación de material para la presentación, así como de la publicación del juego.

PEC1:

Fecha de entrega: 08/10/2023

Contenido: Determinar la temática y forma concisa del trabajo. Explicar la motivación y justificación del trabajo escogido.

PEC2:

Fecha de entrega: 12/11/2023

Contenido: Documentar el arte y diseño de juego. Aprendizaje de herramientas. Realizar un prototipo del juego.

PEC3:

Fecha de entrega: 17/12/2023

Contenido: Realizar el proyecto y adquirir experiencia. Documentar el trabajo realizado.

PEC4:

Fecha de entrega: 14/01/2023

Contenido: Realizar video de síntesis del proyecto. Redactar documentos técnicos y preparar presentación.

PEC 5:

Fecha de entrega: del 15/01 al 31/01 de 2023

Contenido: Defensa del proyecto.

Durante la PEC2 se desarrolla el prototipo de juego, donde podemos ver en funcionamiento las mecánicas jugables. Los escenarios se prueban en modo de bloques para ir viendo la dirección más adecuada respecto a su diseño.

En la PEC3 creamos los elementos 3D definitivos. Esto engloba tanto al escenario, como a los enemigos, las armas o el carácter principal al que solo le vemos los brazos al utilizar una cámara en primera persona. Una vez todo este montado se pasa a la fase de pruebas y corrección de errores.

En la PEC4 trabajaremos la presentación del proyecto y su publicación y en la PEC5 se defiende el proyecto.

Lo vemos en un diagrama de Gantt que contiene los hitos y contenido principal del proyecto:

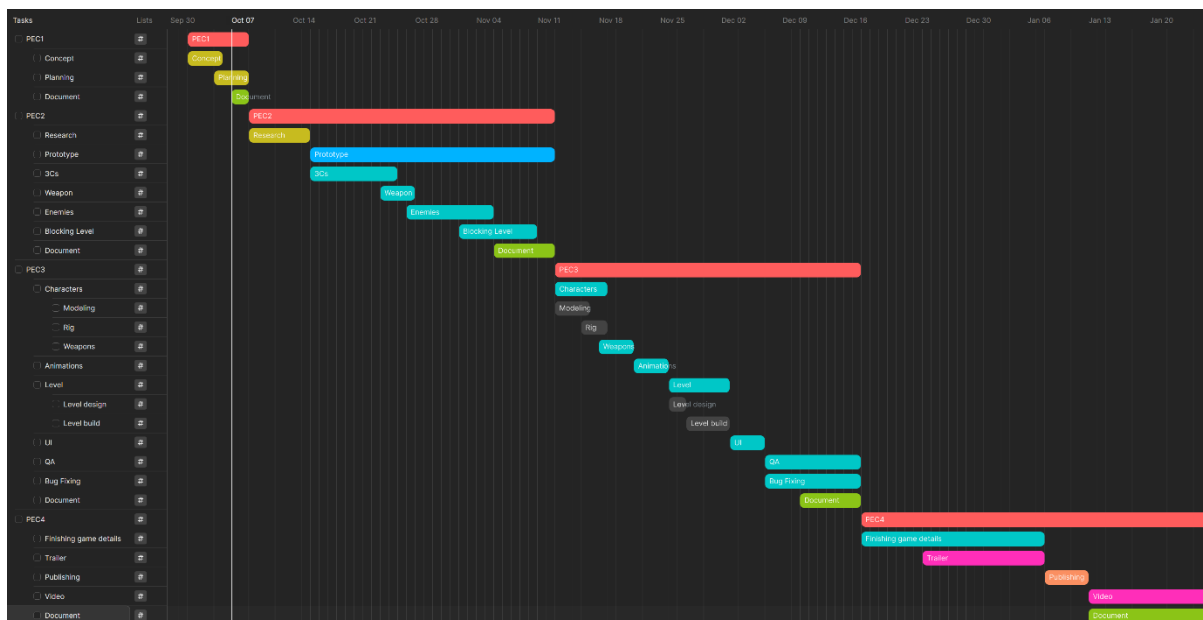


Ilustración 4: Diagrama de Gantt del proyecto.
Realizado con <https://height.app/>

Este diagrama no incluye detalles como la realización de apartados más pequeños, como audio o efectos visuales. Tampoco incluye el aprendizaje de herramientas que se realizará conforme son necesarias en el desarrollo del proyecto.

1.6 Presupuesto

Para realizar el presupuesto, dado que es un proyecto final de carrera que no pretende comercializarse, se reduce ya que usamos principalmente software *open-source*. Para proyectos de mayor envergadura, o que busquen rentabilidad económica, es más adecuado usar software de pago en general, y también *plugins* o contenido de librería que encarecerían el proyecto, pero mejorarían el resultado final.

La tabla 1 muestra el presupuesto necesario para la realización y entrega de el proyecto final de grado en el caso de que el autor cobrara por realizarlo.

Software		
Unity	Motor gráfico	0 €
Blender	Herramienta gráficos 3D	0 €
Visual Studio	IDE	0 €
Gimp	Editor de gráficos	0 €
Audacity	Editor de audio	0 €
Sourcetree	Control de versiones	0 €
Midjourney	Creación de conceptos visuales	33 €
Hardware		
PC		2500 €
Assets		
Librería Audio	Gratuita en Unity	0 €
Música	Libre de derechos	0 €
Recursos Humanos		
1 Desarrollador	A tiempo completo durante 2 meses	6000 €
TOTAL		
TOTAL		8533 €

1.7 Estructura del resto del documento

En las siguientes secciones se mostrarán un análisis de mercado que nos indicará el estado actual de la industria en general, y del género seleccionado en específico. Con él veremos el público objetivo al que se apunta.

Se profundiza en el análisis de la competencia existente, así como un DAFO centrados en las características fuertes y débiles del producto.

También es analizado el modelo de negocio y la estrategia de *marketing* seguida, y en la sección 5 el producto y sus objetivos principales.

Una vez analizado el entorno profundizamos en el documento de diseño del juego donde se muestran las características del *gameplay* así como el contenido.

En la última sección se explica qué conclusiones se extraen del desarrollo del juego y su proyección futura.

2. Análisis de mercado

2.1. Género del juego

El surgimiento del género *First Person Shooter* (FPS) encuentra su génesis en títulos emblemáticos como *Wolfenstein 3D* y su predecesor *Doom*, desarrollados por *ID Software* (<https://www.idsoftware.com/>), los cuales catapultaron el género a la cúspide de la industria de los videojuegos. Desde entonces, los *shooters* en primera persona han mantenido una presencia significativa como referentes destacados en la esfera de los videojuegos. La clave de su éxito radica en una mecánica de juego elemental: disparar indiscriminadamente a objetivos en movimiento. Esta fórmula, acompañada por un distintivo estilo visual, diversión efervescente y explosiones espectaculares, define su identidad sin depender excesivamente de elementos como la trama, la narrativa o el desarrollo profundo de personajes.

Es pertinente señalar que la mecánica de disparo, aunque no constituyó una innovación revolucionaria, remonta su origen a uno de los recursos más antiguos en la historia de los videojuegos. Incluso en los albores de los arcades, juegos exitosos como *Asteroids* o *Space Invaders* se basaron en mecánicas de disparos para cautivar a los jugadores.

La evolución del género ha transitado diversas fases en las cuales nuevas mecánicas y tendencias han refrescado el género, generando éxitos globales. La introducción del modo multijugador fue una de las primeras innovaciones significativas, evidenciada por títulos como *Quake*, seguido posteriormente por sagas icónicas como *Call of Duty* o *Battlefield*, que lograron cautivar a una amplia audiencia. Más recientemente, la corriente del género *Battle Royale*, donde varios participantes compiten para ser el último sobreviviente en un entorno de batalla en constante reducción de tamaño, ha dado lugar a fenómenos de la industria como *PUBG* o *Fortnite*. Por otro lado, podemos destacar también el juego *Tarkov* por sus mecánicas de incursiones, innovadoras como aportación al género.

En consonancia con las corrientes contemporáneas que modelan el panorama de los videojuegos, este proyecto se fundamenta en una tendencia emergente conocida como *Boomer Shooters*. Los reinicios de la franquicia *Doom* son quizás los ejemplos más notorios de este fenómeno, si bien existen numerosos casos adicionales que reflejan la esencia de este género, o una evolución más actual que proyecta su influencia, como es el caso de *Witchfire*.

2.1.1 *Boomer shooter*

El origen del término *boomer shooter* carece de una procedencia clara. El significado asociado a la palabra *boomer* no alude específicamente a individuos pertenecientes a la generación *baby boomer*, sino más bien a cualquier persona mayor que es reconocida como cerrada de mente y está desconectada de las tendencias contemporáneas.

La definición precisa del género *boomer shooter* tampoco goza de consenso unánime entre la comunidad de jugadores. En la década de 1990, los *shooters* en primera persona se caracterizaban por su rapidez, intensidad y contenido violento. Estos títulos se apoyaban en armamento poderoso, presentaban una amplia variedad de enemigos y exigían la destreza del jugador para avanzar, no siendo particularmente accesibles. A medida que el género evolucionó, se observó una ralentización en los controles, un realismo creciente en las armas, una disminución en su potencia y una reducción en la diversidad de arsenal.

La reciente ola de *boomer shooters* se inspira en los FPS originales de los años 90, fusionando la acción y la violencia características de los *shooters* modernos. Es imperativo destacar que los *boomer shooters* no se limitan simplemente a recrear juegos *shooter* de la década de 1990 en un formato compatible con los dispositivos contemporáneos. Aunque es cierto que muchos de estos juegos incorporan gráficos propios de la época, como *sprites* 2D en un entorno 3D o modelos y texturas *low-poly*, es esencial diferenciarlos de los juegos retro, ya que incorporan mecánicas o elementos propios de los juegos modernos.

2.2. Público objetivo y perfiles de usuario

En el contexto de los videojuegos, los juegos del género *shooter* destacan por su ritmo vertiginoso y complejas mecánicas, exigiendo habilidad y determinación por parte de los jugadores para dominarlos. Estas características atraen a una audiencia apasionada, compuesta no solo por individuos altamente comprometidos en perfeccionar sus habilidades, sino también por aquellos que buscan una experiencia divertida y colaborativa.

A lo largo de la evolución de este género, se ha observado una variación significativa en el público objetivo. En 2010, Sony Computer Entertainment America identificó que el segmento demográfico clave para los juegos de disparos comprendía a individuos de edades comprendidas entre los 13 y los 34 años, según información proporcionada por la agencia Reuters [1].

Sin embargo, el fenómeno de masas representado por el juego "Fortnite" de Epic Games en 2017 demostró un cambio en esta dinámica. Este título logró un éxito abrumador al dirigirse a una audiencia más joven, con el 60% de sus jugadores situados en el rango de edades de 18 a 24 años. Además, se destacó una marcada predominancia masculina, representando el 90% de la población de jugadores [2].

En lo que respecta a la formación educacional de los jugadores de este género, los resultados de un estudio llevado a cabo por Jeroen Janz y Martin Tanis en 2007, titulado 'Appeal of Playing Online First Person Shooter Games', revelaron una distribución variada. Según este estudio, el 37,5% de los jugadores tenía estudios de nivel medio, el 34,5% había recibido formación universitaria, y el 21,0% poseía formación profesional [3].

Estos datos proporcionan una visión holística de la diversidad en el perfil de los jugadores de juegos de disparos, abordando aspectos demográficos y educativos que subrayan la amplia atracción de este género en el ámbito de los videojuegos.

Tipo de jugadores

En función de las motivaciones subyacentes de los jugadores, se puede llevar a cabo una clasificación de los mismos en diversas categorías, adoptando la taxonomía de jugadores propuesta por Richard Bartle [4]. Las categorías principales identificadas son los buscadores de emociones, los competitivos y los maestros de habilidades.

Los buscadores de emociones, también conocidos como *Thrill Seekers*, encuentran especial atracción en los juegos de acción. Este grupo de jugadores disfruta de experiencias que demandan rapidez y agilidad en los reflejos. Adicionalmente, se sienten atraídos por elementos como los gráficos en 3D, los sistemas de logros y los sistemas de misiones, los cuales añaden capas de complejidad y desafío a su experiencia de juego.

Por otro lado, los maestros de habilidades, o *Skill Masters*, se inclinan hacia juegos que destacan la destreza del jugador. Caracterizados por su nivel de dificultad y desafío, estos juegos son apreciados por los Maestros de Habilidades debido a las pronunciadas curvas de aprendizaje que presentan. Este grupo de jugadores valora la ausencia de aleatoriedad en los juegos y prefiere la introducción progresiva de nuevas mecánicas a medida que avanza el juego. Asimismo, son atraídos por sistemas de progresión de niveles que reconocen y recompensan el desarrollo de sus habilidades.

Por último, los competitivos, también conocidos como *King of the Hill* o Rey de la Colina, están centrados en la competición. Su motivación principal radica en alcanzar el primer lugar en las clasificaciones y tablas de puntuación, ya sea compitiendo contra otros jugadores o, en algunos casos, desafiándose a sí mismos. Esta categoría de jugadores encuentra su motivación en la superación constante y la búsqueda de la excelencia competitiva.

2.3. Competencia

Dentro del subgénero *Boomer Shooter* encontramos en la actualidad numerosos juegos con distintas características que los hacen únicos.

Painkiller (2004)

Desarrollado por *People Can Fly*, emerge como uno de los primeros exponentes catalogados bajo la denominación de *Boomer Shooters*. Este título se distingue por la incorporación de disparos alternativos innovadores para sus variadas armas, introduciendo elementos como una escopeta capaz de lanzar pernos de hielo, así como una pistola equipada con "shurikens" y rayos. Estas características contribuyen a la singularidad del juego, otorgando a los jugadores una experiencia de juego rica en posibilidades estratégicas. *Painkiller* incorpora, además, elementos de juego de rol (RPG) que se manifiestan a través de cartas equipables, agregando una capa adicional de complejidad y personalización a la jugabilidad. La inclusión de una transformación que proporciona invulnerabilidad temporal amplía la dimensión estratégica del juego. Este título, caracterizado por su elevado nivel de dificultad, se sitúa en la categoría de juegos desafiantes. La ilustración 13 nos muestra una captura del juego.



Ilustración 5: Captura del juego Painkiller
https://store.steampowered.com/app/3270/Painkiller_Overdose/

Doom (2016) y **Doom Eternal** (2020)

Desarrollados por *ID Software*, representan una evolución significativa en el *gameplay* del título original *Doom*. Estas entregas no solo preservan la esencia característica de la serie, sino que también incorporan ajustes sustanciales para adaptarse a las innovaciones contemporáneas en mecánicas de juego.

Un aspecto destacado de estas entregas es el ingenioso sistema de recuperación de vida y munición, el cual se basa en la eliminación de enemigos mediante distintas mecánicas. Esta novedosa implementación impulsa al jugador a mantener un constante estado de ataque para recuperar vitalidad y munición, desviándose así de las estrategias convencionales de resguardo para esperar una recuperación automática de vida. Este enfoque contribuye a instaurar un ritmo frenético en los combates, intensificando la experiencia del jugador al exigir un compromiso activo y constante en el campo de batalla. El rediseño de esta mecánica de juego no solo enriquece la dinámica de combate, sino que también redefine la estrategia tradicional asociada a los juegos de esta índole. La ilustración 14 nos muestra una captura del juego *Doom Eternal*.



Ilustración 6: Captura de *Doom Eternal*
<https://simoncoenen.com/blog/programming/graphics/DoomEternalStudy>

Dusk (2018)

Es una obra que evoca una poderosa sensación nostálgica para los entusiastas de los juegos de disparos en primera persona de un jugador, rememora la era dorada de los videojuegos en la década de los años 90. Este juego se erige sobre la premisa de velocidad extrema, impregnada de efectos visuales sangrientos, y presenta un variado arsenal que abarca desde hoces hasta rifles, escopetas e incluso lanzacohetes.

La trama de *Dusk* se desenvuelve en la aparentemente apacible localidad homónima, donde los jugadores se ven inmersos en inquietantes ruinas imbuidas de un estilo *lovecraftiano*. El relato toma un giro siniestro cuando los científicos enviados a explorar este lugar se ven poseídos por entidades demoníacas. En este contexto, el protagonista asume la responsabilidad de erradicar a estas criaturas infernales. Los jugadores se ven desafiados por una diversidad de enemigos que incluyen cultistas, animales de granja poseídos e incluso espantapájaros resucitados. La ilustración 15 nos muestra una captura del juego *Dusk*, y la ilustración 16 del juego *Ultrakill*.

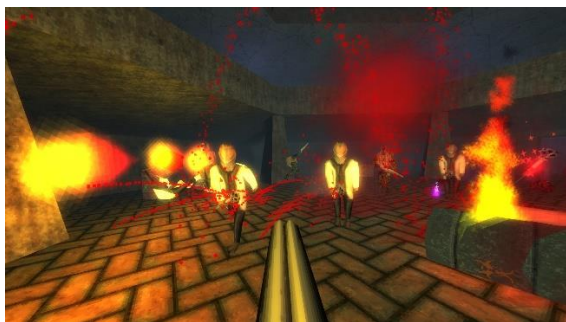


Ilustración 15: Captura de *Dusk*
<https://store.steampowered.com/app/519860/DUSK/>



Ilustración 16: Captura de *Ultrakill*
<https://store.steampowered.com/app/1229490/ULTRAKILL/?l=spanish>

Ultrakill (2022)

El juego, creado por Arsi "Hakita" Patala y publicado por *New Blood Interactive*, se erige como un testimonio de la fusión entre la elegancia estética y una jugabilidad cautivadora. En él, los jugadores asumen el papel de V1, un resistente superviviente robótico que emerge de las secuelas de una guerra en la que los robots de combate alimentados con sangre exterminaron a la humanidad. V1 recurre a un reservorio inesperado: los demonios y las almas condenadas en el mismísimo Infierno.

La esencia de *Ultrakill* radica en su homenaje a los *boomer shooters*, canalizando un estilo de juego que otorga máxima importancia al movimiento rápido y constante para esquivar ataques, y la agresión desenfrenada dentro de niveles diseñados en formato de arena. La supervivencia de V1 depende de un mecanismo único: la restauración de la salud se logra exclusivamente a través de la inmersión en la sangre de los enemigos vencidos, lo que exige tácticas de combate a corta distancia. A medida que V1 avanza por el paisaje infernal, eliminar enemigos de maneras diversas y estilizadas no solo garantiza la supervivencia, sino que también acumula puntos. Estos puntos, a su vez, pueden ser gastados de manera prudente por los jugadores para mejorar su arsenal mediante mejoras de armas, añadiendo una capa de profundidad estratégica a la dinámica del juego.

2.4. Análisis DAFO

Se realiza un análisis DAFO en el que observamos primero los factores internos y a continuación los externos.

Factores Internos.

Debilidades:

1. **Limitaciones en los gráficos:** La calidad media del arte se refleja en el bajo presupuesto, lo que puede afectar la experiencia visual y atraer menos a jugadores que valoran los aspectos estéticos.
2. **Recursos limitados:** Tener un presupuesto limitado puede impedir el desarrollo de contenido adicional, la corrección de errores o la mejora de la calidad general del juego.
3. **Competencia en el mercado:** Incluso en subgéneros más específicos como el *boomer shooter*, la saturación del género shooter puede hacer que sea difícil destacar entre otros títulos más pulidos y con mayores recursos.
4. **Dependencia de expansiones:** La gratuidad del juego y los precios bajos de las expansiones pueden hacer que seas demasiado dependiente de ingresos adicionales a largo plazo.

Fortalezas:

1. **Concepto innovador:** La idea de un concurso de televisión como base del juego es innovadora y puede despertar el interés de los jugadores que buscan experiencias novedosas.
2. **Precio atractivo:** Para aquellos jugadores que buscan experiencias de calidad sin gastar mucho dinero, la gratuidad del juego inicial y los bajos precios de las expansiones pueden ser un atractivo.
3. **Enfoque en el subgénero *boomer shooter*:** este subgénero es popular en la actualidad y puede atraer a un grupo de personas que aprecian la jugabilidad más tradicional y desafiante.
4. **Potencial de comunidad:** La gratuidad puede ayudar a formar comunidades activas y comprometidas, lo que puede generar boca a boca positiva y mantener el juego por mucho tiempo.

Factores externos.

Oportunidades:

1. **Actualizaciones constantes:** Con el tiempo, la calidad del juego puede mejorar si la comunidad proporciona comentarios y actualizaciones regulares.
2. **Marketing creativo:** Crear estrategias de marketing innovadoras que resalten la originalidad del juego utilizando la idea original del concurso de televisión.
3. **Colaboraciones con streamers:** como el juego es gratuito, colaborar con *streamers* y creadores de contenido puede aumentar la visibilidad y atraer a un público más amplio.

Amenazas:

1. **Recepción negativa Inicial:** Antes de que se implementen actualizaciones, la calidad gráfica limitada puede generar críticas negativas durante el lanzamiento del juego, lo que podría afectar la percepción del juego.
2. **Competencia constante:** La industria del juego es altamente competitiva, y puede ser difícil destacar debido a la saturación del mercado de *shooters*.
3. **Dependencia de ingresos adicionales:** El juego podría tener problemas financieros si la venta de expansiones a precios bajos no es suficiente para cubrir los gastos y generar ganancias.
4. **Cambios en las preferencias del jugador:** Las preferencias del jugador pueden cambiar rápidamente, y si el juego no se adapta a estas tendencias, el juego podría perder relevancia.

En resumen, el proyecto Big Money tiene el potencial de destacar debido a su concepto innovador y precios atractivos, pero enfrenta desafíos debido a limitaciones gráficas y competencia en el mercado. Para mantener la relevancia y la sostenibilidad a largo plazo, es fundamental aprovechar las fortalezas y oportunidades mientras se abordan las debilidades y amenazas.

3. Propuesta

3.1. Definición de objetivos/especificaciones del producto

La meta fundamental de este proyecto consiste en desarrollar un programa completamente funcional que incluirá los primeros niveles del juego, presentados en el formato de venta shareware.

Para contextualizar el proyecto, se anexa un archivo PDF que ofrece un avance del desarrollo del documento de diseño del juego (GDD), en formato *ten-page* o de diez páginas. Este documento detalla las características esenciales del juego que se están desarrollando.

3.2. Modelo de negocio

En el modelo de negocio propuesto para este proyecto se busca volver a la estrategia de venta de Shareware de la década de 1990, con el objetivo de crear una comunidad activa. A pesar de las restricciones presupuestarias, el objetivo principal es maximizar los ingresos y garantizar la sostenibilidad a largo plazo del proyecto. La ilustración 17 nos muestra las cajas del juego Doom en su formato *shareware*.



Ilustración 17: Discos *shareware* de Doom
<https://archive.org/>

Esta estrategia es la misma que adoptaron los juegos que iniciaron el género *First-Person Shooter* que basa la generación de beneficios en la venta de contenido, pero alejándose de

la venta *freemium* donde el tiempo de juego está limitado o el contenido de venta es únicamente estético.

La gratuidad del juego constituye un enfoque esencial, permitiendo que los jugadores accedan al juego base sin costos iniciales. Esta estrategia tiene la intención de atraer a una audiencia amplia, eliminando barreras económicas al ingreso. La venta de expansiones a precios bajos complementa la gratuidad del juego. Al ofrecer contenido adicional a precios asequibles, se pretende incentivar las compras, fidelizar a los jugadores y generar ingresos adicionales de manera continua con una constante actualización.

Estrategias de monetización creativas, como eventos temáticos y ofertas especiales, se implementarán para estimular compras en momentos específicos. Asimismo, la introducción de paquetes combinados y descuentos temporales buscará motivar la participación activa de los jugadores en las compras. La ilustración 18 nos muestra una imagen de la promoción de KFC en el juego PUBG.



Ilustración 18: Promoción de PUBG con KFC
<https://impulsogeek.com/videojuegos/kfc-pubg-battlegrounds-para-que-los-saboreen-la-victoria/>

Se fomentará la creación de contenido gratuito generado por usuarios, como *skins* y modelos 3D, pero dejando la creación de niveles y su venta a los desarrolladores. Además de enriquecer la experiencia del usuario, este modelo permitirá a los creadores de contenido obtener ingresos compartidos a través de las compras realizadas por la comunidad. La participación activa de la comunidad será un elemento esencial. Incentivos como eventos exclusivos, recompensas y reconocimientos dentro del juego fortalecerán el sentido de pertenencia y contribuirán a la retroalimentación positiva, así como al boca a boca.

En cuanto a la publicidad, se explorarán acuerdos de patrocinio y publicidad integrada con cautela. Estas estrategias se utilizarán de manera selectiva para generar ingresos adicionales sin comprometer la inmersión en el juego.

Por último, la entrega regular de actualizaciones con nuevas expansiones de niveles será clave para retener a la comunidad y ofrecer oportunidades continuas para la monetización a través de expansiones y elementos premium.

En resumen, este modelo de negocio busca equilibrar la gratuidad inicial del juego con estrategias de monetización creativas y una participación activa de la comunidad. La adaptabilidad a las tendencias cambiantes y la continua entrega de valor a los jugadores son esenciales para el éxito sostenido de *Big Money* dentro de las limitaciones presupuestarias.

3.3. Estrategia de marketing

La estrategia de marketing se centra en identificar y comprometer al público objetivo, destacando las fortalezas distintivas del juego.

1. Identificación del público objetivo: se dirigirá específicamente a jugadores que tengan interés en el género boomer shooter y también tengan afinidades con programas de concursos televisivos. El público objetivo busca experiencias innovadoras y desafiantes.

2. Creación de contenido atractivo: Se creará contenido promocional que resalte la originalidad del concepto del concurso de televisión y la emocionante jugabilidad del juego. Se diseñarán *trailers*, *teasers* y contenido audiovisual para atraer al público objetivo.

3. Colaboraciones con *streamers* y *youtubers*: Se establecerán asociaciones con creadores de contenido prominentes que tengan audiencias afines al género *boomer shooter*. La distribución de versiones de prueba a *streamers* relevantes aumentará la visibilidad del juego en plataformas clave como *Twitch* y *Youtube*.

4. Estrategia de gratuidad y expansión: El formato de venta es el mismo que el habitual de los primeros *First-Person Shooters*. Se destacará la ventaja de la gratuidad del juego, así como la disponibilidad de expansiones a precios razonables. Esta táctica tiene como objetivo motivar la adquisición de contenido adicional para generar ingresos sostenibles.

5. Participación en eventos y convenciones: Para aumentar la visibilidad y obtener retroalimentación directa, el juego se presentará en eventos de la industria y convenciones de videojuegos. La organización de eventos como concursos y torneos ayudará a atraer a nuevos jugadores.

6. Estrategia en redes sociales: Las plataformas de redes sociales serán los canales principales para compartir actualizaciones regulares, contenido multimedia y una visión detrás de cámaras del desarrollo del juego. Se utilizarán varias actividades, como concursos y encuestas, para promover la participación de la comunidad.

7. Enfoque en Comunidad: la creación de foros, grupos y canales de Discord ayudará a los jugadores a interactuar más y mejorar su sensación de pertenencia a la comunidad. Se reconocerá y destacará el gran trabajo que han hecho los miembros de la comunidad para fomentar la participación activa.

8. Actualizaciones Frecuentes: Las mejoras gráficas, las nuevas características y el contenido adicional se destacarán mediante una comunicación clara y regular sobre las actualizaciones del juego. Para mantener el interés a largo plazo, se presentará este compromiso con la mejora continua.

9. Publicidad Online Segmentada: Se implementará publicidad en línea dirigida a jugadores que tienen intereses particulares en *shooters* y concursos. Para aumentar la visibilidad del

juego en las búsquedas relacionadas, se utilizarán técnicas SEO de optimización de motores de búsqueda.

10. Testimonios y Reseñas: Se recopilarán testimonios de jugadores y reseñas positivas para respaldar las campañas de marketing. El énfasis en la experiencia positiva de la comunidad se utilizará como un punto clave para resaltar la calidad y atractivo del juego.

Esta estrategia integral tiene como objetivo maximizar la visibilidad del juego, atraer a la audiencia objetivo y construir una comunidad sólida, elementos cruciales para el éxito sostenido del juego a pesar de las limitaciones presupuestarias.

4. Diseño

4.1. Entorno

Para la elección del entorno, al ser un juego basado en 3D, dos opciones gratuitas se alzan como las más adecuadas.

Unreal Engine.

Creado por la desarrolladora Epic Games, Unreal Engine (<https://www.unrealengine.com/>) se destaca como el motor de juegos insignia que redefinió la calidad y la funcionalidad del desarrollo interactivo en 3D. Se basa en un motor de renderizado avanzado que crea nuevos paradigmas estéticos al combinar gráficos fotorrealistas con efectos visuales de última generación. Su arquitectura escalable y la capacidad de producir contenido para múltiples plataformas lo convierten en el preferido de desarrolladores y estudios de renombre.

Unreal Engine se destaca en el campo de la simulación al incorporar herramientas de física y dinámica de partículas que permiten crear entornos inmersivos y realistas. Utiliza el lenguaje C++. Respecto a la calidad gráfica, es el líder en la creación de experiencias interactivas 3D. Además, también es usado en la industria audiovisual en cine y publicidad al contar con características que permiten crear escenarios virtuales en tiempo real.

En los últimos años, Epic Games ha realizado una gran inversión en el motor y en el contenido para formación gratuito.

Unity.

El proceso creativo se ve facilitado por su interfaz fácil de entender, soporte para múltiples plataformas y una gran biblioteca de *assets*. Consolidándose como una herramienta clave en el sector, Unity ha sido la fuerza impulsora detrás de nuevos juegos de móvil.

Se erige como el motor que democratiza la creación digital, liderando la industria de desarrollo de videojuegos. Se ha convertido en el estándar de elección para desarrolladores en todo el mundo debido a su versatilidad y robustez. Este entorno versátil creado por Unity

Technologies (<https://unity.com/>) permite a los diseñadores y programadores convertir sus ideas en experiencias interactivas increíbles. Unity se destaca por su potente motor de renderizado y su habilidad para administrar activos multimedia complejos, lo que permite la creación de juegos visualmente impactantes en diversas plataformas.

Este motor se destaca por su arquitectura basada en componentes y su lenguaje de programación C#, que ofrece flexibilidad y modularidad en el desarrollo. Los recursos de aprendizaje extensos y la comunidad activa refuerzan su posición como un entorno de desarrollo accesible y colaborativo.

Aprovechando su flexibilidad y eficacia en el despliegue multiplataforma, Unity se ha consolidado como el favorito para los juegos móviles. Su posición como líder en el desarrollo de juegos en dispositivos móviles se consolida con su integración con servicios en la nube y herramientas analíticas, lo que facilita la optimización y monetización de aplicaciones.

Ambas opciones se encuentran plenamente respaldadas y exhiben similitudes notables. La determinación final de optar por Unity se ha fundamentado en las preferencias personales del autor, quien considera que este motor representa un desafío valioso desde el punto de vista del aprendizaje, dado su desconocimiento previo por parte del mismo, no así con el motor Unreal. Este enfoque se alinea con la voluntad de expandir el conocimiento técnico y abordar nuevos retos, aspectos que se consideran fundamentales para el desarrollo integral del autor en el ámbito del diseño y desarrollo de videojuegos.

4.2. Requisitos técnicos del entorno de desarrollo.

Para la realización de este proyecto se ha tomado la decisión de no depender de *assets* externos, en la medida de lo posible, para el desarrollo del mismo. Como resultado, se requieren habilidades especializadas en una amplia gama de disciplinas:

Programación en C#. Para elaborar el código de juego dentro del entorno del motor Unity, es esencial tener un conocimiento profundo del lenguaje de programación C#.

Creación del contenido 3D: para crear elementos visuales tridimensionales, es esencial tener habilidades en el manejo de software especializado en la creación de contenido tridimensional.

Procesamiento de imágenes 2D: Para manipular, crear texturas y optimizar los recursos visuales, es necesario tener habilidades en el uso de software destinado al procesamiento de imágenes bidimensionales.

Estudio de los métodos de renderizado del motor Unity: es crucial para decidir el aspecto visual del juego y sus limitaciones técnicas, como el Built-in Render Pipeline de renderizado incorporado, URP (Universal Render Pipeline) y HDRP (High Definición Render Pipeline).

Técnicas de iluminación: comprender y utilizar técnicas de iluminación avanzadas de los distintos diseños de niveles.

Funcionamiento de Shaders y Materiales: Aprender a usar y manipular *shaders* y materiales para lograr efectos visuales específicos.

Diseño de juegos: Desarrollar habilidades en el diseño integral de videojuegos que involucre la conceptualización, la mecánica del juego y la experiencia del usuario.

Técnicas de optimización: Aprenda estrategias y técnicas de optimización para garantizar que el juego desarrollado funcione de manera eficiente.

La ejecución exitosa del proyecto dependerá de la combinación de estos conocimientos técnicos, lo que indica la necesidad de un enfoque multidisciplinario y una sólida base técnica en diversas áreas del desarrollo de videojuegos.

4.3. Inventario de herramientas utilizadas.

Para la realización de este proyecto se han utilizado distintas herramientas. El sistema operativo del PC es Windows 10.

Unity 2022: es un poderoso motor de desarrollo de videojuegos que ofrece herramientas versátiles para la creación de experiencias interactivas en diversas plataformas. Destaca por su capacidad de desarrollo multiplataforma y su interfaz amigable, respaldada por una amplia comunidad de usuarios.

Microsoft Visual Studio 2022: es un entorno de desarrollo integrado (IDE) líder que proporciona potentes herramientas para la creación de aplicaciones, incluido el desarrollo de software en C# para proyectos en Unity. Ofrece características avanzadas de depuración, edición de código y colaboración, mejorando la eficiencia del desarrollo.

GitHub: es una plataforma de desarrollo colaborativo que utiliza el sistema de control de versiones Git. Facilita la gestión y colaboración en proyectos de software, permitiendo a los equipos trabajar de manera eficiente, realizar seguimiento de cambios y gestionar el código fuente de manera distribuida.

SourceTree: es una interfaz gráfica para Git que simplifica la gestión de repositorios. Ofrece una visualización clara de las ramas y cambios, facilitando la interacción con Git. Es una herramienta útil para aquellos que prefieren una interfaz gráfica para gestionar su control de versiones.

Midjourney: es una herramienta de creación de imágenes por inteligencia artificial. Se utiliza en el proyecto para crear todos los diseños conceptuales iniciales. Para su uso es necesario el programa **Discord**, que es una plataforma de comunicación en línea que combina voz, video y chat, diseñada para comunidades y grupos.

Blender: es un software de modelado y animación 3D de código abierto. Destaca por su amplio conjunto de herramientas que abarcan desde modelado hasta renderizado, siendo utilizado tanto por artistas independientes como por profesionales de la animación y los efectos visuales.

Gimp: es un software de edición de imágenes de código abierto que ofrece una amplia gama de funciones similares a las de programas comerciales. Es una herramienta versátil para la manipulación de imágenes, diseño gráfico y retoque fotográfico.

YouTube: es la plataforma líder de alojamiento y compartición de videos en línea. Permite a usuarios cargar, ver y compartir contenido audiovisual, abarcando una variedad de temas que van desde entretenimiento hasta educación, convirtiéndose en una fuente omnipresente de información y entretenimiento en línea. En esta plataforma se han subido los videos del desarrollo del proyecto.

4.4. Inventario de assets y recursos.

La realización de este proyecto se ha enfocado en realizar tanto el código como los diferentes recursos manualmente, evitando en la medida de lo posible el uso de librerías.

Sin embargo, debido a la dificultad y el tiempo necesario para aprender a realizar los efectos de sonido, se ha utilizado la librería de **Shapeforms Audio Free Sound Effects** para los distintos efectos utilizados en el juego.

<https://assetstore.unity.com/packages/audio/sound-fx/shapeforms-audio-free-sound-effects-183649>

En concreto, la ilustración 19 nos muestra los utilizados:

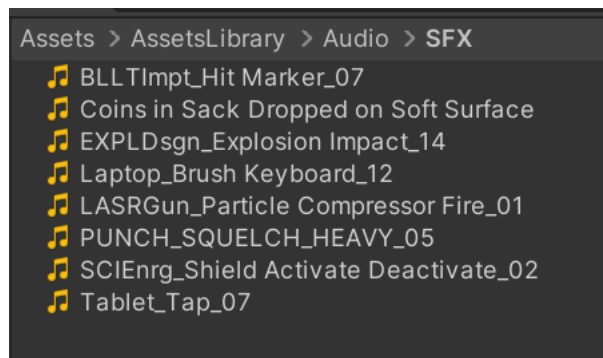


Ilustración 19: Efectos de sonido del proyecto en Unity

A su vez, la música utilizada es libre de derechos del autor **Karl Casey** (estudio White Bat Audio). En la ilustración 20 podemos observar los temas musicales utilizados en el proyecto.

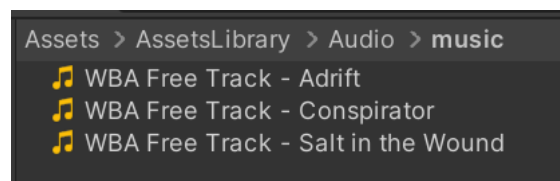


Ilustración 20: Música de Karl Casey utilizada en Unity

La fuente utilizada en el juego **ArchivoBlack-Regular** es libre de derechos descargada de Google Fonts.

Respecto a los assets generados por el autor del proyecto:

3D.

player_arms.fbx: modelo 3D de los brazos del jugador. Incluye las animaciones de estado idle, caminando, disparo y recarga.

animations_laserRifle.fbx: modelo 3D del arma. Incluye las animaciones de estado idle, caminando, disparo y recarga.

chaser_01.fbx: modelo 3D del enemigo Chaser.

Projectile_v01.fbx: modelo 3D del proyectil disparado por el rifle laser.

Wall_external01.fbx: modelo 3D de los muros externos.

Wall_externalelevator.fbx: modelo 3D del muro del ascensor.

Weapon_laserRifle.fbx: modelo 3D del arma rifle laser.

Texturas.

Arms.png: textura de los brazos del personaje.

Floor_mosaic_yellow.png: textura del suelo del nivel 01.

Floor_mosaic_red.png: textura del suelo del nivel 02.

Floor_mosaic_purple.png: textura del suelo del nivel 03.

Logo_Wall.png: logotipo del juego.

Logo_Wall_alphaBG.png: logotipo del juego con fondo transparente.

logoBM.png: logotipo del juego en formato grande.

Player.png: imagen del jugador para el menú de controles.

Projectile.png: textura para el proyectil.

Icon1024.png: icono del ejecutable del juego.

UI.

BMlogo.png: logotipo del juego.

Damage.png: efecto de daño.

Health_armleft.png, Health_armright.png, health_legright.png, health_leg_left.png, health_head.png, health_body.png: sistema visual de vida.

4.5. Arquitectura del juego.

Los componentes principales de la estructura de juego son:

GameManager.

Desde aquí se controlan los estados del juego, si nos encontramos en un menú, jugando un nivel o alguna pantalla de menú como por ejemplo el principal de inicio. También controla las acciones del fin del juego, ya sea partida perdida o ganada y su reinicio.

Respecto a los menús, controla el bloqueo y visión del ratón dependiendo de si estamos en partida o en un menú.

También realiza las inicializaciones y comprobaciones del manager de audio y del manager de puntuación.

Player.

La cámara principal del juego se encuentra dentro del objeto Player ya que al ser primera persona debe moverse con el jugador, simulando su visión. También el modelo del personaje (los brazos), y el arma utilizada.

Dentro del jugador podemos encontrar podemos encontrar varios scripts con distintas funciones importantes.

InputManager controla la entrada de comandos ya sea por teclado o por *gamepad*, ejecutando las funciones que les corresponden.

PlayerMotor controla las acciones de movimiento del personaje, así como el tiempo de recarga del movimiento especial de esquivar *dash*.

PlayerLook actualiza hacia donde mira el personaje según el movimiento del ratón o el *stick* del *gamepad*.

PlayerInteract, esta clase es usada para la interacción con objetos, como por ejemplo abrir una puerta.

PlayerUI es donde tenemos los distintos elementos de UI del jugador, es decir, el casco con la puntuación, el efecto de daño o la vida.

PlayerHealth, es donde controlamos el daño recibido por ataques al jugador, así como su muerte, y un corto tiempo de invencibilidad una vez es dañado.

Enemy

Respecto a los enemigos tenemos un manager (**EnemyManager**) que controla la vida, envía una señal cuando uno muere y además sus efectos y sonidos. Aparte, varios sistemas de control de movimiento y ataque según su tipo.

Los enemigos que realizan patrulla (**EnemyPatrol**) tienen una máquina de estados **StateMachine** que controla los distintos estados. Estos son: patrulla, ataque y búsqueda del enemigo una vez lo pierde de vista.

GameAssets

Utiliza una lista para tener un fácil acceso a los assets usados en el proyecto y tenerlos localizados en un mismo sitio.

ScoreManager

Controla y actualiza la puntuación del jugador.

SoundManager

Realiza las reproducciones de los distintos audios.

MainMenu

Controla la navegación del menú principal, así como el control de sus distintos elementos.

Weapon

Para controlar las características del arma tenemos un **WeaponManager** el cual gestiona el disparo único, el automático o el de ráfagas. Las características de cada arma se guardarán en el objeto **Weapon** en el momento en que se implementen otras. Las físicas de las balas se encuentran en el objeto **ProjectileMove**.

La estructura de escenas consta de los distintos menús de selección o finalización del juego y los niveles en sí. Siendo **MainMenu** la escena principal donde encontramos el menú inicial. Una vez pulsamos en el botón de jugar cargará la escena **Level01** y continuaremos con otros niveles. Si el jugador muere cargará la escena **GameOver** y si termina el juego **Win**.

4.6. Funcionamiento de la IA

En el proyecto se usan dos tipos de IA para los enemigos: los persecutores y los patrulleros.

EnemyChaser.

Este enemigo siempre persigue al jugador desde que aparece en el nivel. Se basa en formato muy sencillo, un Nav Mesh Agent que le indica por donde se puede desplazar y se le indica por medio de un script la posición del jugador que actualiza una vez por *frame*.

EnemyPatrol.

El enemigo patrullero usa un sistema más complejo de diferentes estados, pasando de uno a otro según la situación y su estado actual.

En el estado patrulla **PatrolState**, el enemigo tiene asignado unos puntos de patrulla que generan un camino por donde se desplaza. Continúa en este ciclo hasta descubrir al jugador. Para ello usa un método de *raycast* con unos valores de apertura que constantemente buscan el objeto del jugador.

Una vez descubre al jugador pasa al estado de ataque **Attack_State**. En este estado fija la posición del jugador con un **LookAt** y dispara su arma cada cierto tiempo.

Si el enemigo pierde de vista al jugador, es decir, se aleja del alcance o es bloqueado por un objeto, el enemigo pasa al estado de búsqueda **SearchState**. En este estado el enemigo se desplaza hacia la última posición conocida del jugador, luego a un par de puntos más a cierta distancia de ese último punto. Si consigue encontrar al jugador volverá al estado de ataque **Attack_State**, si no es capaz de encontrarlo entonces pasará al estado de patrulla **PatrolState**, volviendo a realizar su camino inicial en el punto en el que lo dejó.

4.7. Diseño de niveles.

Todos los niveles utilizan la misma base, ya que corresponde a cada planta del rascacielos. A medida que el jugador supera niveles sube plantas ficticiamente.

La variación de los elementos internos del nivel es lo que los hace únicos, siendo distintos dependiendo del tipo de enemigo al que nos enfrentamos o de la colocación de los puntos de generación de enemigos.

Como muestra la ilustración 21, en la parte inferior podemos ver el ascensor y el pasillo de entrada al nivel.

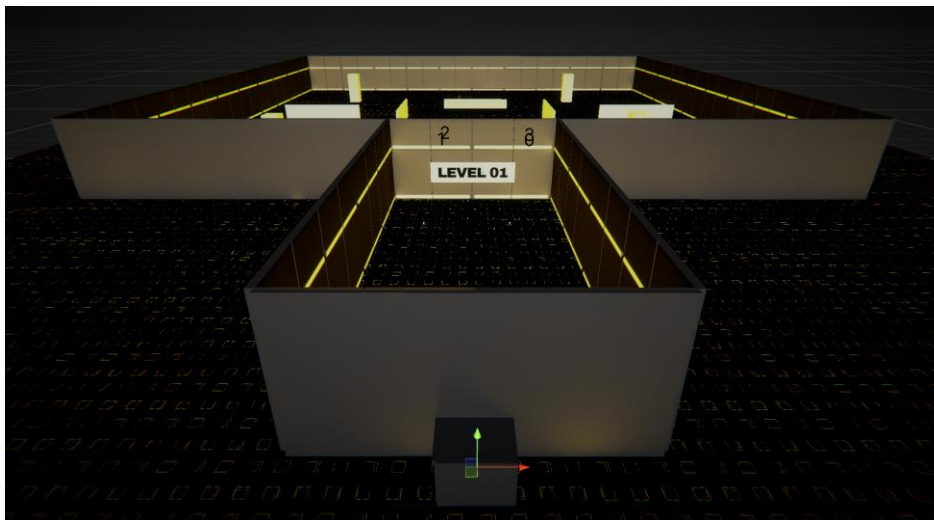


Ilustración 7: Vista aérea del nivel 01

Esta estructura de entrada es común en todos los niveles y lo que cambia es la colocación de los elementos de la sala interior.

Las salas interiores no están sobresaturadas de elementos ya que el juego busca la confrontación directa a los enemigos y que la habilidad del jugador sea la que decida el ganador, no dando muchas opciones de usar 'trucos' en los niveles.

En la ilustración 22 y 23 podemos observar la sala interior del nivel 01.

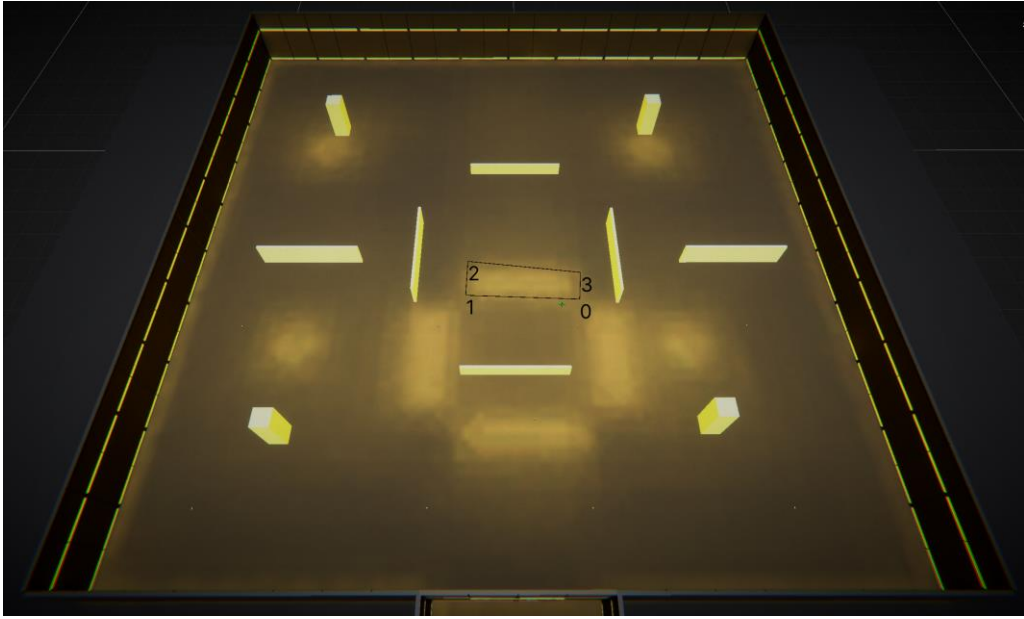


Ilustración 8: Sala interior del nivel 01

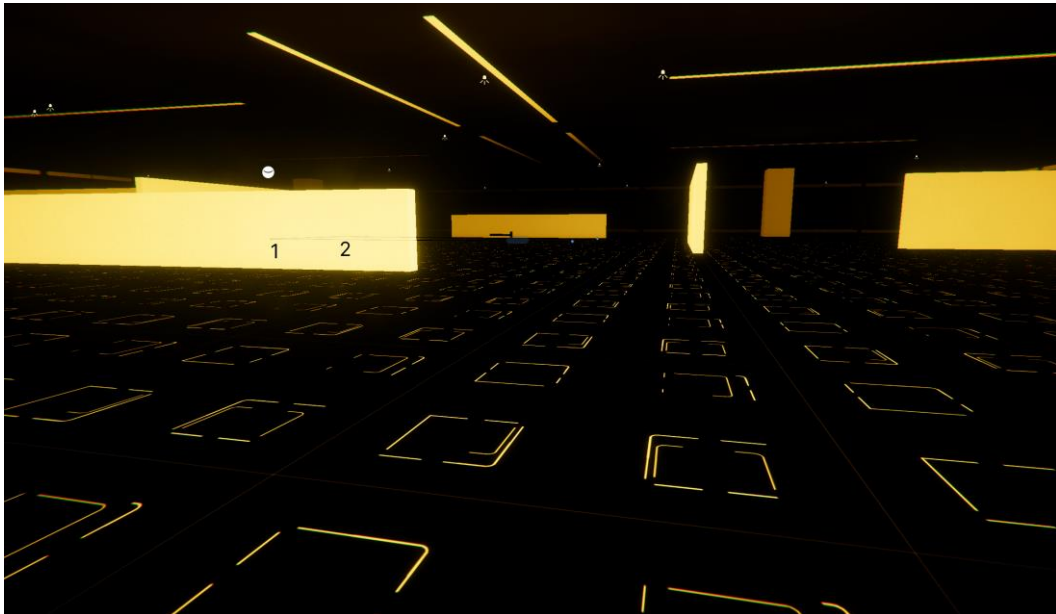


Ilustración 9: Vista desde el suelo de la sala interior del nivel 01

5. Implementación

Para la reproducción del juego no es necesaria la instalación. Abrir el archivo '**BigMoney.exe**' ejecutará el juego a pantalla completa.

El ejecutable del juego está realizado para sistemas Windows con procesadores de 64.

6. Demostración

Los controles para jugar son mostrados antes del inicio de cada partida nueva. Estos son:



Ilustración 10: Controles del teclado

7. Conclusiones y líneas de futuro

Lograr terminar un proyecto jugable en el tiempo establecido es un logro del que me siento muy orgulloso.

La planificación, aunque con ligeras modificaciones durante la producción, se ha mantenido a tiempo. Sin embargo, aunque predecible, el alcance se fue reduciendo desde las primeras ideas iniciales hasta el resultado final. Es normal ya que, sin experiencia, calcular los costes y tiempo de un proyecto tan amplio es una tarea complicada. Por el camino algunos bugs complicaron el desarrollo, pero su resolución o incluso el cambio completo de partes del código resultaron en un aprendizaje gratificante.

El proyecto me ha servido como lanzadera a introducirme más profundamente en el motor gráfico Unity, y a descubrir una producción completa de un videojuego. También que para la programación con un código más limpio es necesario un estudio más profundo de los sistemas de juegos.

La creación de un videojuego es un proyecto muy interesante, ya que engloba muchas áreas distintas y requiere de diversas habilidades para poder culminarlo con éxito. Sin duda, es un reto y una pasión que enfrentar en mi trayectoria profesional.

Por último, comentar que una terminado el proyecto, siendo este la finalización de la primera iteración del juego completo, comienza la planificación de la segunda. La organización ahora dispone de un calendario más amplio en el que sin duda se podrá poner más amor a pulir el detalle y el perfeccionamiento. Esta segunda iteración incluirá:

- Nuevo contenido: varias armas, enemigos y niveles.
- Mejoras del arte, la iluminación, efectos visuales y de sonido.
- Pulido del gameplay, gunplay y ajuste de la dificultad.
- Creación de cinemáticas que descubran la historia del juego.

Aquí comenzó el que seguramente sea un largo camino hasta completar el que es mi primer proyecto personal publicado.

Bibliografía

- [1] Reuters. The Sydney Morning Herald (2010).
<<https://www.smh.com.au/technology/sony-targets-adult-gamers-with-multiplayer-shooter-game-mag-20100122-mp9t.html>> [Consulta: 15 Octubre 2023]
- [2] Mansoor Iqbal (2023). "Fortnite usage and revenue statistics." En *Business of apps*.
<<https://www.businessofapps.com/data/fortnite-statistics/>> [Consulta: 15 Octubre 2023]
- [3] Jeroen Janz, Martin Tanis (2007). *Appeal of playing online first person shooter games*. Cyber Psychology & Behavior. Volume 10, number 1 (2007).
<https://www.researchgate.net/publication/6501010_Appeal_of_Playing_Online_First_Person_Shooter_Games> [Consulta: 18 Octubre 2023]
- [4] Richard Bartle (1996). *Hearts, clubs, diamonds, spades: players who suit muds*.
<<https://mud.co.uk/richard/hcds.htm>> [Consulta: 18 Octubre 2023]

Anexos

Glosario

FPS: First Person Shooter, juego de disparos en primera persona.

Asset: archivos externos utilizados en la creación de un juego. Por ejemplo, una textura o un modelo 3D.

Shooter: Género de videojuegos de acción, en el que la mecánica principal del personaje son los disparos.

Gameplay: acciones que realiza un jugador para interactuar con el juego.

Gunplay: mecánicas relacionadas con las armas.

Entregables

Adjunto a este encontramos un archivo pdf con el Documento de Diseño del Juego.

Capturas de pantalla

