



Florida

Universitatària

**Grado en Diseño y Desarrollo de
Videojuegos y Experiencias Interactivas**

Trabajo de Fin de Grado

Desarrollo completo de un videojuego

Autor: Jorge Robledo Durendez

DNI: 24474772-N

Tutor: Juan Peralta Donate

Fecha: 09/09/2024

Curso: 2023-2024

Titulació n adscrita a



UNIVERSITAT
POLITÀCNICA
DE VALÈNCIA

Resumen

Este Trabajo de Fin de Grado tiene como objetivo el desarrollo de una versión Alpha de un videojuego indie en el género First Person Shooter (FPS). El proyecto abarca todas las fases del desarrollo, desde la planificación inicial y la conceptualización, hasta la producción, optimización y publicación. Utilizando el motor Unity y herramientas como Blender para el modelado 3D y Photoshop para el diseño gráfico, se ha creado una experiencia jugable que combina el uso de armas y magia elemental. Durante el proceso de desarrollo se ha implementado un sistema de clasificación global (ranking) y se ha optimizado el rendimiento a través de técnicas de postprocesado. La publicación del videojuego se ha realizado en la plataforma Itch.io, dirigida a un público global, con la página personalizada en inglés para atraer a un mayor número de jugadores.

Palabras clave: Videojuegos, Desarrollo indie, Unity, First Person Shooter, Magia, Mecánicas de juego, Optimización, Postprocesado, Publicación, Itch.io, Blender, Scriptable Objects, Clasificación global.

Abstract

This Final Degree Project aims to develop an Alpha version of an indie video game in the First Person Shooter (FPS) genre. The project covers all stages of development, from initial planning and conceptualization to production, optimization, and publication. Using the Unity engine, along with tools such as Blender for 3D modeling and Photoshop for graphic design, a gameplay experience combining firearms and elemental magic has been created. During the development process, a global ranking system has been implemented, and performance was optimized through post-processing techniques. The game has been published on the Itch.io platform, targeting a global audience, with the page customized in English to reach a larger number of players.

Keywords: Video games, Indie development, Unity, First Person Shooter, Magic, Game mechanics, Optimization, Post-processing, Publishing, Itch.io, Blender, Scriptable Objects, Global ranking.

Índice

1	Índice de Ilustraciones y Tablas	4
2	Introducción.....	5
3	Estado del Arte	6
3.1	Historia de los Videojuegos	6
3.2	Evolución de los Géneros: FPS	8
3.3	Motores de Videojuegos: Unity	10
3.3.1	Historia de Unity	10
4	Objetivos y Metodología	12
4.1	Objetivos	12
4.2	Metodología	12
5	Contenido	13
5.1	Planificación	13
5.1.1	Conceptualización de la Idea	14
5.1.2	Análisis DAFO	16
5.1.3	Gantt	19
5.2	Preproducción	19
5.2.1	Desarrollo de Ten Page	19
5.2.2	Prototipos y Pruebas de Concepto	20
5.2.3	Elección de Herramientas	21
5.3	Producción	22
5.3.1	Programación y Desarrollo de Mecánicas.....	23
5.3.2	Diseño y Construcción de Niveles	27
5.3.3	Creación de Arte y Animaciones	29
5.4	Periodo de Prueba	30
5.4.1	Pruebas Internas (Alpha Testing) y Pruebas Externas (Beta Testing)	31
5.5	Prelanzamiento y lanzamiento	32
5.5.1	Preparación Final y Optimización	32
5.6	Publicación	33
6	Conclusiones	34
7	Bibliografía	37
8	Anexos	38
8.1	Ten Page	38

8.1.1	Proyecto	38
8.1.2	Resumen	38
8.1.3	Contexto	39
8.1.4	Mecánicas	39
8.1.5	Progresión	40
8.1.6	Diseño de niveles	41
8.1.7	Arte y Estilo Visual	41
8.1.8	Animaciones	42
8.1.9	Sonido y Música	42
8.1.10	Publicación y Monetización	43
8.2	Enlaces	43

1 Índice de Ilustraciones y Tablas

Ilustración 1	Análisis de los géneros de videojuegos en las diferentes plataformas	14
Ilustración 2	Ejemplo de estructura de un análisis DAFO	16
Ilustración 3	Gantt realizado sobre la estimación de las tareas	19
Ilustración 4	Concepto de la idea de los brazos, la pistola y los VFX, imagen generada por IA	21
Ilustración 5	Los scripts encargados de la gestión de inputs y de las mecánicas del personaje	23
Ilustración 6	Enumeradores de las Magias disponibles para el jugador y de los estados	24
Ilustración 7	Ejemplo de un Scriptable Object que almacena uno de los diálogos	26
Ilustración 8	Funciones <u>utilizaras</u> para cargar e insertar las entradas de la Leaderboard	27

2 Introducción

Los videojuegos son una de las principales formas de entretenimiento en el mundo, y su desarrollo ha evolucionado significativamente en las últimas décadas. Con el avance de la tecnología, ha sido posible crear experiencias más inmersivas, complejas y accesibles. Este Trabajo de Fin de Grado (TFG) se enfoca en la creación de un videojuego del género First Person Shooter (FPS), combinando mecánicas tradicionales de disparo con elementos de magia, lo que ofrece una jugabilidad única y estratégica.

El objetivo principal del proyecto es desarrollar una versión Alpha de un videojuego indie, siguiendo un enfoque metodológico estructurado que abarca todas las fases del desarrollo: planificación, preproducción, producción, pruebas y optimización. Durante el desarrollo han sido utilizadas herramientas como Unity, Blender y Photoshop para crear los elementos visuales y de programación necesarios para llevar a cabo el proyecto. Asimismo, ha sido implementado un sistema de clasificación global y una publicación en la plataforma Itch.io para hacer accesible el juego a una audiencia internacional.

A lo largo de este documento, se detallará el proceso de desarrollo del videojuego, destacando los desafíos encontrados y las soluciones implementadas para cumplir con los objetivos planteados.

3 Estado del Arte

3.1 Historia de los Videojuegos

El inicio de los videojuegos es relativamente reciente, data de la década de los 1950 poco después de la aparición de los ordenadores. De forma experimental surgieron programas como Nimrod (1951) y OXO (1952) versiones de juegos tradicionales trasladados a lo electrónico. Los primeros videojuegos pioneros fueron Tennis for Two (1958) y Spacewar! (1962). A pesar de considerarse ya videojuegos, eran juegos muy simples y su uso estaba limitado a universidades o centros de investigación.

Junto a los walkmans y los televisores llegaron las máquinas recreativas y con ellas los primeros videojuegos dirigidos al público. Computer Space (1971) y Pong (1972) fueron los que inauguraron las primeras máquinas recreativas. El auge y la popularidad hizo que las recreativas fueran comunes en bares y salones recreativos. Clásicos como Space Invaders (1978), Asteroids (1978) o Pac-Man (1980) marcaron la primera época de los videojuegos, creando un nuevo fenómeno sociocultural.

Hasta la década de los 1980, el mundo de los videojuegos estaba monopolizado por Atari y sus recreativas, pero desde Japón aparecieron dos nuevos competidores: Nintendo y SEGA. Estas dos empresas se enfocaron en los videojuegos para los hogares con las consolas NES (1983) y la Master System (1983). Paralelo al auge de estas dos aparecieron los primeros ordenadores domésticos como el Spectrum (1982) o el Commodore 64 (1982). A partir de esto los videojuegos se convirtieron en una gran industria. A finales de década aparecieron las primeras consolas portátiles como la Game Boy (1989) que implicó un gran avance tecnológico.

En los 1990 hubo un gran salto de tecnología con la tecnología de 16-bit en consolas como la SNES (1990) y la Mega Drive (1988). Esta mejora en la tecnología trajo un avance gráfico significativo y un nuevo género de moda: los shooters en 3D. Con este gran cambio, empiezan a aparecer consolas nuevas de empresas que ya estaban en el sector como la Nintendo 64 (1996) o la Sega Saturn (1994), pero también entraron al sector nuevos competidores como Sony con la PlayStation (1994) o Microsoft con la Xbox (2001). Los videojuegos se volvieron más complejos, con narrativas más desarrolladas y gráficos más realistas.

La década de los 2000 vio la expansión masiva de los videojuegos en línea, con juegos como World of Warcraft (2004) y Call of Duty (2003). Las consolas también evolucionaron, con la PlayStation 2 (2000), Xbox 360 (2005), y Wii (2006) como líderes del mercado. La alta definición (HD) se convirtió en el estándar de la tecnología, llevando los gráficos y las experiencias de juego a nuevos niveles de realismo.

En la década de 2010, los videojuegos se diversificaron aún más con la llegada de los dispositivos móviles y la consolidación de las plataformas de juego como Steam (2003). Juegos como Angry Birds (2009) y Fortnite (2017) consolidaron los videojuegos móviles y los videojuegos multijugador en línea. La realidad virtual y aumentada también comenzaron a ganar tracción con dispositivos como Oculus Rift () y juegos como Pokemon Go (2016). Además, el mercado de los eSports creció exponencialmente, convirtiéndose en una industria multimillonaria con una audiencia global.

(Facultat d'Informàtica de Barcelona (Universitat Politècnica de Catalunya), s.f.)

3.2 Evolución de los Géneros: FPS

El origen de los First Person Shooter (FPS) puede rastrearse hasta los años 70 y 80, con juegos como "Maze War" (1973) y "Spasim" (1974), que son considerados los primeros ejemplos del género. Estos juegos, aunque primitivos en comparación con los estándares modernos, introdujeron la idea de la perspectiva en primera persona y el movimiento a través de un espacio tridimensional.

Sin embargo, el verdadero precursor del género fue "Wolfenstein 3D" (1992) desarrollado por id Software. Este juego estableció muchas de las convenciones que se convertirían en estándar para los FPS, como la exploración de laberintos, el combate contra enemigos y la recolección de objetos. "Wolfenstein 3D" fue un éxito comercial y crítico, y su motor de juego sirvió como base para futuros desarrollos.

La década de los 90 se considera la era dorada de los FPS, marcada por la innovación y la popularización del género. En 1993, id Software lanzó "Doom", un juego que revolucionó la industria con sus gráficos avanzados, jugabilidad rápida y soporte para mods creados por la comunidad. "Doom" no solo fue un hito técnico, sino que también estableció una base sólida para la cultura de los FPS.

Otro título influyente de esta época fue "Quake" (1996), también de id Software, que introdujo gráficos en 3D completamente renderizados y un enfoque en el multijugador en línea. "Quake" y su motor de juego dieron lugar a una serie de secuelas y derivados, y establecieron un estándar para los FPS multijugador.

Con la llegada del nuevo milenio, los FPS comenzaron a diversificarse en términos de ambientación, narrativa y mecánicas de juego. Juegos como "Half-Life" (1998) de Valve, que ofrecía una narrativa profunda y una experiencia inmersiva, demostraron que los FPS podían ser mucho más que simples juegos de disparos.

El lanzamiento de "Halo: Combat Evolved" (2001) para Xbox fue otro hito importante, popularizando el género en las consolas y estableciendo una franquicia duradera. "Halo" introdujo mecánicas innovadoras como el regenerador de salud y el uso de vehículos en el combate.

Durante esta década, los FPS multijugador también ganaron popularidad. Juegos como "Counter-Strike" (1999) y "Battlefield 1942" (2002) se centraron en la experiencia competitiva en línea, mientras que "Call of Duty" (2003) comenzó una de las franquicias más exitosas y prolíficas en la historia de los videojuegos.

La década de 2010 vio un enfoque creciente en el realismo y la narrativa dentro de los FPS. Juegos como "Call of Duty: Modern Warfare" (2007) y sus secuelas ofrecieron experiencias cinematográficas y gráficos realistas, mientras que títulos como "Bioshock" (2007) combinaron elementos de FPS con narrativas profundas y ambientes detallados.

En años más recientes, el género ha continuado evolucionando con la introducción de juegos de servicio en vivo como "Destiny" (2014) y "Overwatch" (2016), que combinan elementos de RPG y juegos de disparos en un entorno multijugador continuo. Además, los juegos Battle Royale como "Fortnite" (2017) y "Apex Legends" (2019) han redefinido el género con su enfoque en la supervivencia y la competencia masiva en línea.

(Jensen, 2022)

3.3 Motores de Videojuegos: Unity

Unity es un motor de desarrollo de videojuegos multiplataforma, reconocido por su capacidad para crear experiencias 2D y 3D interactivas. Es ampliamente utilizado en la industria de los videojuegos, así como en otros campos como la realidad aumentada (AR), la realidad virtual (VR) y la simulación. Unity es conocido por su interfaz amigable y sus potentes herramientas que permiten a los desarrolladores diseñar, probar y publicar juegos en diversas plataformas.

3.3.1 Historia de Unity

Unity Technologies fue fundada en 2004 por David Helgason, Nicholas Francis y Joachim Ante. El primer lanzamiento, Unity 1.0, en 2005, se enfocó en desarrolladores para Mac OS. Con el tiempo, Unity se expandió a Windows y otras plataformas, convirtiéndose en un motor de referencia para el desarrollo multiplataforma.

El lanzamiento de Unity 2.0 en 2007 y la adición de soporte para iOS en 2008 ampliaron su base de usuarios. En 2010, Unity añadió soporte para Android, consolidando su posición en el desarrollo de juegos móviles. Unity 5, lanzado en 2015, trajo mejoras significativas en calidad visual y herramientas de desarrollo.

En 2023, Unity enfrentó controversia debido a cambios en su estructura de precios y modelo de licencias, lo que provocó una reacción negativa de la comunidad de desarrolladores. Las nuevas tarifas y condiciones generaron preocupaciones sobre la sostenibilidad y accesibilidad del motor para estudios pequeños e independientes. Unity Technologies tuvo que ajustar su estrategia y comunicarse activamente con los desarrolladores para abordar sus preocupaciones y mantener la confianza en la plataforma.

Unity ha evolucionado desde sus inicios como una herramienta para Mac OS a ser uno de los motores de desarrollo de videojuegos más populares y versátiles. A pesar de las recientes controversias, sigue siendo una opción preferida por millones de desarrolladores gracias a su capacidad para soportar múltiples plataformas y sus potentes herramientas de desarrollo.



4 Objetivos y Metodología

4.1 Objetivos

Con este trabajo de fin de grado se pretende desarrollar una versión Alpha de un videojuego indie, abarcando todas las etapas del proceso de desarrollo, con un enfoque principal en la programación y diseño, y en menor medida, el apartado artístico. Este proyecto tiene como finalidad ampliar y aplicar los conocimientos adquiridos durante el Grado en Diseño y Desarrollo de Videojuegos y Experiencias Interactivas.

Para ello se han planteado el siguiente objetivo:

1. Desarrollar y publicar una versión Alpha de un videojuego indie.
 - a. Aplicar los conocimientos teóricos adquiridos para diseñar y programar las mecánicas básicas del videojuego.
 - b. Crear e integrar los elementos artísticos necesarios para la versión Alpha del videojuego y asegurar una cohesión estética y funcional.
 - c. Preparar el videojuego para su publicación, desarrollar una estrategia de lanzamiento y material promocional adecuado.

4.2 Metodología

Para cumplir el objetivo mencionado en el apartado anterior se aplicará la siguiente metodología:

1. Se realizará una fase de planificación del proyecto con el objetivo de tener un buen flujo de trabajo.
2. Se realizará un análisis de mercado para investigar los géneros, la competencia en el mercado de los videojuegos y de esta forma determinar el género.
3. Realizar el diseño y programación de un nivel en el motor de videojuegos Unity.

4. Realizar un “dressing” del escenario, personajes y VFX utilizando recursos de la Asset Store del motor de videojuegos Unity y realizando el modelado y efectos de lo necesario para la creación del videojuego.
5. Realizar las animaciones del personaje principal, enemigos que aparezcan en el nivel y de ser necesario de elementos de este.
6. Se realizará una fase de pulido y finalización en la que se añadirá iluminación, se añadirán sonidos y música, se realizará un postprocesado visual, se pulirá la interfaz de usuario y se realizarán pruebas de experiencia de usuario, búsqueda de errores y optimización y para finalizar se realizarán unas pruebas con usuarios para obtener feedback.
7. Se analizarán y realizará el marketing y publicación de un videojuego indie.

5 Contenido

5.1 Planificación

La fase de planificación es la etapa inicial en el desarrollo de un videojuego, donde se establecen las bases del proyecto. Durante esta fase, se definen los objetivos, se identifican los recursos necesarios, se desarrollan estrategias y se crean planes detallados que guiarán el desarrollo del videojuego. Esta fase es crucial para asegurar que todas las partes involucradas entiendan el alcance y los requerimientos del proyecto, lo que ayudará a prevenir problemas y asegurar un desarrollo más fluido.

5.1.1 Conceptualización de la Idea

Durante el proceso de conceptualización de la idea, se tomaron decisiones clave sobre la selección del género, la plataforma objetivo y el motor de desarrollo para el videojuego de este Trabajo de Fin de Grado. Estas decisiones se llevaron a cabo considerando tanto las tendencias del mercado como la experiencia técnica adquirida durante el grado.

Según el informe de Newzoo de 2023 (Wijman, 2023), los géneros de videojuegos más rentables en PC incluyen shooters, aventuras y juegos de rol. Entre estos, el género First Person Shooter (FPS) ha demostrado ser el más lucrativo y popular en la plataforma PC, lo que motivó su elección para este proyecto.



Ilustración 1 Análisis de los géneros de videojuegos en las diferentes plataformas

La plataforma PC fue seleccionada como objetivo principal debido a su flexibilidad y amplia compatibilidad tanto con herramientas de desarrollo, como motores y software de diseño, así como con diferentes sistemas operativos, lo que facilita la portabilidad del juego. Desarrollar para PC permite un control más exhaustivo sobre la optimización y el rendimiento del juego, además de contar con una amplia base de jugadores que prefieren este tipo de títulos. La facilidad de desarrollo en PC, utilizando herramientas como Unity, facilita el

proceso de pruebas y depuración, asegurando que el juego cumpla con los estándares esperados.

El motor de desarrollo elegido es Unity, principalmente debido a la experiencia y conocimiento adquiridos durante el Grado en Diseño y Desarrollo de Videojuegos y Experiencias Interactivas. Unity ofrece una gran versatilidad y capacidad para manejar gráficos avanzados y físicas complejas, aspectos cruciales para un juego FPS. Además, su amplia comunidad de desarrolladores y abundante documentación lo convierten en una opción ideal para este proyecto, permitiendo resolver problemas y aprender nuevas técnicas de manera eficiente.

En conclusión, la decisión de desarrollar un juego del género First Person Shooter (FPS) para la plataforma PC utilizando el motor Unity está respaldada por un análisis detallado del mercado y las habilidades técnicas adquiridas. Esta elección asegura una dirección sólida y viable para el desarrollo del videojuego, alineándose con las tendencias actuales y maximizando el potencial de éxito del proyecto.

Una vez decidido el género, la plataforma y el motor, se inició un proceso de búsqueda de ideas mediante una sesión de lluvia de ideas, de la cual surgió el concepto de fusionar un First Person Shooter (FPS) con un sistema de magia elemental, permitiendo al jugador realizar diferentes combinaciones para resolver puzles o derrotar enemigos con mayor facilidad.

A la vez que se definían los conceptos base, también se planteaba el contexto y la historia del juego. Esta narrativa surgió en paralelo con la idea de fusionar elementos de magia en un juego de disparos en primera persona. La historia se centra en un alumno de una escuela de magia, que nunca ha destacado por sus habilidades mágicas. Sin embargo, cuando la escuela es invadida por malvados monstruos, el protagonista es el único que

puede salvarla. Dado que no posee la fuerza mágica suficiente para enfrentarlos, decide combinar su magia elemental con el uso de una pistola, utilizando ambas habilidades para derrotar a los enemigos y restaurar la paz en la escuela.

5.1.2 Análisis DAFO

El Análisis DAFO (también conocido como SWOT por sus siglas en inglés: Strengths, Weaknesses, Opportunities, Threats) es una herramienta estratégica utilizada para identificar y analizar las Fortalezas, Debilidades, Oportunidades y Amenazas de un proyecto, organización, producto o individuo. Este análisis ayuda a comprender los factores internos y externos que pueden influir en el éxito o fracaso del proyecto.

El objetivo del Análisis DAFO es proporcionar una visión clara y estructurada de la situación actual de un proyecto u organización, facilitando la toma de decisiones estratégicas informadas. Al identificar las fortalezas y debilidades internas, así como las oportunidades y amenazas externas, los equipos de trabajo pueden desarrollar estrategias efectivas para maximizar el potencial y minimizar los riesgos.

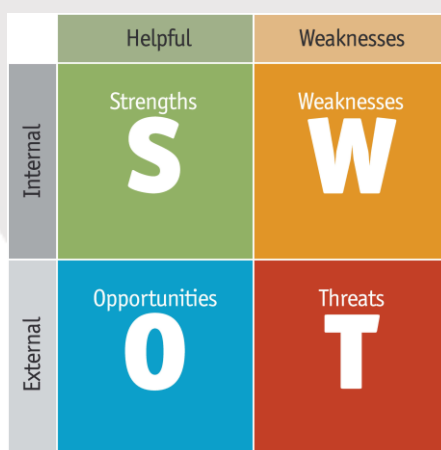


Ilustración 2 Ejemplo de estructura de un análisis DAFO

Fortalezas:

- Innovación en el género FPS: La mezcla de disparos con magias elementales ofrece una experiencia fresca dentro de un género muy explorado, aportando originalidad y profundidad estratégica.
- Diversidad de mecánicas: Combinar armas y magias permite ofrecer una jugabilidad más variada, donde los jugadores pueden experimentar diferentes estilos de combate y estrategias.
- Atracción de varios públicos: Este tipo de juegos puede atraer tanto a los fanáticos de los FPS clásicos como a aquellos interesados en juegos de rol o fantasía debido al uso de magias y poderes.
- Rejugabilidad: La posibilidad de experimentar con distintas combinaciones de magias incrementa la rejugabilidad, ya que cada partida puede ser diferente.
- Elementos visuales atractivos: Las magias ofrecen oportunidades para crear efectos visuales impactantes, que pueden mejorar la inmersión y la estética del juego.

Debilidades:

- Curva de aprendizaje: Al mezclar mecánicas de disparo con magias, los jugadores pueden encontrar el juego más complicado que los FPS tradicionales, lo que puede desanimar a aquellos que prefieren una experiencia más directa.
- Mayor demanda de recursos: Implementar magias, con sus diferentes efectos visuales y mecánicas, puede requerir más recursos de desarrollo y optimización, lo que podría impactar el rendimiento del juego.

- Posibles comparaciones con otros títulos: Juegos que han mezclado armas y magia, como BioShock o Destiny, podrían generar expectativas altas y comparaciones difíciles de superar.

Oportunidades:

- Mercado en expansión: Los juegos que mezclan géneros, como los FPS con elementos de rol o fantasía, tienen un atractivo creciente.
- Nuevas tecnologías y plataformas: El auge de tecnologías como la realidad virtual y la realidad aumentada podría ofrecer nuevas maneras de interactuar con el sistema de magia, ampliando las posibilidades de inmersión.

Amenazas:

- Saturación del mercado FPS: El género de los FPS está muy saturado, con títulos como Call of Duty y Fortnite dominando el mercado. Esto puede hacer que un juego con magias tenga dificultades para destacar.
- Dificultad para atraer a jugadores tradicionales de FPS: Los jugadores más puristas de FPS podrían rechazar la mezcla de elementos de fantasía, prefiriendo experiencias más realistas y directas.
- Competencia con otros juegos híbridos: Títulos que combinan disparos con elementos mágicos o fantásticos, como Destiny, Warframe o Paladins, podrían representar una competencia directa.
- Problemas técnicos: Implementar efectos mágicos avanzados y mecánicas complejas podría llevar a problemas de rendimiento, errores, o dificultades en la optimización del juego para diferentes plataformas.

5.1.3 Gantt

Para la organización de las tareas a realizar en este proyecto se decidió crear un Gantt donde se realizó una estimación del número y tiempo de tareas para este proyecto.

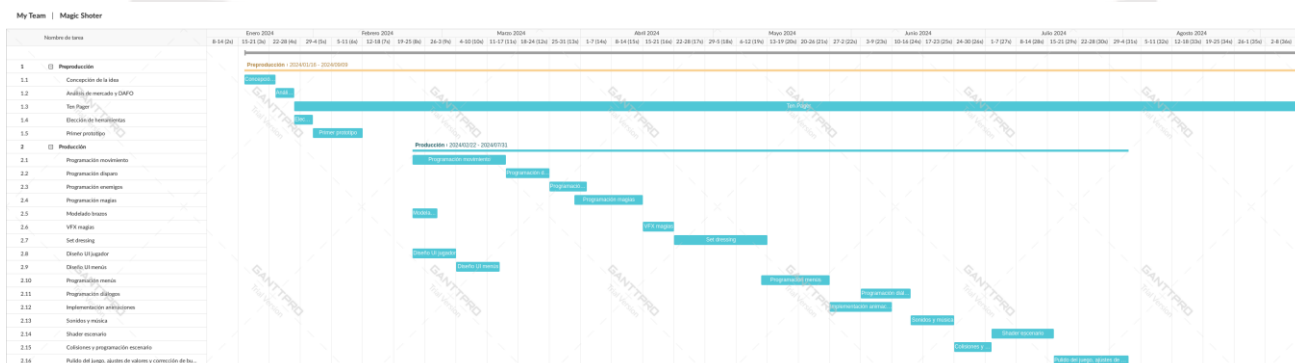


Ilustración 3 Gantt realizado sobre la estimación de las tareas

5.2 Preproducción

A continuación, se procedió a la realización de la preproducción del videojuego.

La fase de preproducción es la etapa en la que se detallan los conceptos y planes establecidos durante la planificación y se preparan los recursos necesarios para la producción. En esta fase, se crean los documentos de diseño, se desarrollan prototipos, se establecen los flujos de trabajo y se realizan las primeras pruebas de concepto. La preproducción es crucial para identificar posibles problemas y refinar las ideas antes de invertir tiempo y recursos significativos en el desarrollo completo del juego.

5.2.1 Desarrollo de Ten Page

Para estructurar el diseño del videojuego se procedió a la creación de un documento “Ten Page”, el cual es una versión condensada y enfocada del Game Design Document

(GDD). Se utiliza para presentar de manera clara y concisa los conceptos y características principales del videojuego, abarcando todos los aspectos fundamentales en solo diez páginas. Este documento es especialmente útil en las primeras etapas del desarrollo para comunicar la visión del juego a todos los miembros del equipo y a posibles inversores o colaboradores.

[Ten Page](#)

5.2.2 Prototipos y Pruebas de Concepto

Una vez todo fue documentado, se procedió a crear un prototipo simple de la idea para evaluar la viabilidad y el funcionamiento de las mecánicas básicas. Durante el proceso de prototipado, se consideraron varios complementos para enriquecer la idea, como la posibilidad de que las diferentes magias también fueran representadas por distintos tipos de armas. Por ejemplo, se propuso que, al cambiar a la magia de agua, el arma se transformara en una escopeta, mientras que la magia de fuego estuviera vinculada a una pistola, y así sucesivamente con los diferentes elementos. Sin embargo, tras madurar la idea, se concluyó que sería necesario limitar el alcance del proyecto para asegurar un resultado pulido y accesible para el jugador.

Desde el principio, la idea de las reacciones entre las distintas magias se planteó inspirada en el videojuego “Genshin Impact”. En dicho juego, existen cinco elementos que reaccionan entre sí al aplicarse sobre un enemigo. Por ejemplo, si un enemigo está afectado por el elemento agua y luego se le aplica una habilidad de fuego, se genera una reacción llamada 'vaporizado', que otorga una bonificación de daño.

Tomando este sistema como referencia, se desarrolló un modelo de reacciones elementales para el videojuego en cuestión. Este sistema está basado en tres elementos:

agua, fuego y rayo, que generan los estados 'mojado', 'en llamas' y 'con carga' respectivamente. El jugador tendría una representación visual del elemento activo en su mano izquierda, mientras que en la mano derecha sostendría un arma. Dependiendo de la magia equipada, los disparos aplicarían dicho elemento al objetivo, alterando su estado.



Ilustración 4 Concepto de la idea de los brazos, la pistola y los VFX, imagen generada por IA

Si el objetivo está bajo un estado específico y recibe otro elemento diferente, se produciría una alteración del estado o una reacción, como por ejemplo: si el objetivo está mojado y recibe una magia de rayo, sería electrocutado, aplicándole un efecto adicional.

5.2.3 Elección de Herramientas

Para el desarrollo de este proyecto, se ha elegido un conjunto de herramientas ampliamente utilizadas en la industria de los videojuegos, garantizando un flujo de trabajo eficiente y compatible con los requerimientos del proyecto. Estas herramientas fueron seleccionadas en función de su capacidad técnica, versatilidad, y de la experiencia adquirida durante el Grado en Diseño y Desarrollo de Videojuegos y Experiencias Interactivas.

El motor de desarrollo Unity ha sido la herramienta principal utilizada para la creación del videojuego. Unity destaca por su flexibilidad y capacidad de manejar gráficos avanzados

y físicas complejas. Además, su amplia documentación y la comunidad de desarrolladores lo convierten en un motor accesible y bien soportado, facilitando la resolución de problemas y el aprendizaje continuo.

Para el control de versiones, se ha utilizado GitHub, una plataforma que permite gestionar y sincronizar el código fuente de manera eficiente. GitHub ha sido fundamental para asegurar la integridad del proyecto a lo largo de su desarrollo, el seguimiento de cambios y la recuperación de versiones anteriores en caso de necesidad. Esto ha permitido gestionar el progreso de manera estructurada y evitar posibles pérdidas de trabajo.

En el apartado artístico, se han utilizado dos herramientas clave. Blender, un software de modelado 3D. Blender ofrece una amplia gama de funcionalidades para modelar, texturizar y animar en un entorno 3D. Para la creación y edición de texturas y elementos 2D, se ha utilizado Photoshop, una herramienta versátil para el trabajo de diseño gráfico. Photoshop ha facilitado el desarrollo de texturas, interfaces y otros elementos visuales con un control preciso sobre los detalles gráficos.

En conjunto, estas herramientas han permitido un desarrollo coordinado y eficiente, integrando aspectos técnicos, artísticos y de gestión de versiones, lo que ha contribuido al éxito del proyecto.

5.3 Producción

Con todo ya diseñado y planteado, se inicia el proceso de producción. La fase de producción es la etapa en la que se crea la mayor parte del contenido del videojuego y se implementan las mecánicas definidas durante la preproducción. En esta fase, el equipo de desarrollo se enfoca en la programación, el diseño de niveles, la creación de arte y

animaciones, así como la integración de sonido y música. Es durante la producción cuando el juego empieza a tomar forma de manera tangible, acercándose a su versión final.

5.3.1 Programación y Desarrollo de Mecánicas

El desarrollo comenzó con la programación del Character Controller del personaje. Para ello, se decidió estructurarlo en tres scripts: InputHandle, encargado de recibir los inputs del dispositivo de control (teclado, ratón o mando); FP_Controller, responsable de convertir los inputs de movimiento, salto, agacharse, correr y mover la cámara; y FPS_Controller, encargado de gestionar los inputs de disparo, cambio de magias y recarga del arma.

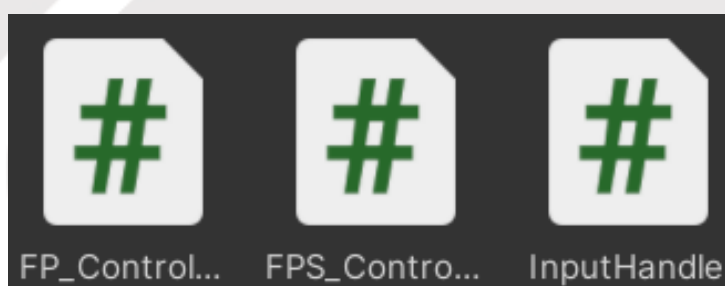


Ilustración 5 Los scripts encargados de la gestión de inputs y de las mecánicas del personaje

Como base para la lectura de inputs, se utilizó el Input System de Unity, el cual permite gestionar de manera sencilla los diferentes tipos de dispositivos de control que se desean implementar en el juego, en este caso: teclado, ratón y cualquier mando de videoconsola.

El movimiento del personaje se implementó mediante el componente Character Controller de Unity, diseñado específicamente para el control de personajes en entornos 3D. Este componente facilita el manejo de movimientos como caminar, correr y saltar sin la complejidad de una simulación física completa, proporcionando una forma sencilla de gestionar colisiones y terrenos.

El script FP_Controller contiene las funciones y parámetros relacionados con las mecánicas de movimiento base en un videojuego en primera persona, como el movimiento

del personaje, el control de la cámara, la vida, el salto, correr y el agacharse. También gestiona sistemas complementarios como el headbob, utilizado para mejorar la sensación de inmersión del jugador, y un sistema de sonidos adaptado a los diferentes tipos de superficies. Inicialmente, se desarrolló un sistema de interacción con el entorno, inicialmente pensado para permitir que el jugador pudiera abrir y cerrar puertas pero finalmente se utilizó para interactuar con un NPC debido al cambio de assets y escenario.

Por otro lado, el script FPS_Controller contiene las funciones y parámetros relacionados con las mecánicas asociadas al disparo y la selección de magias. Este script gestiona la programación de los disparos, los sonidos y los efectos visuales (VFX), como el fogonazo en el cañón del arma al disparar, el brillo en el punto de impacto de la bala, y la creación de agujeros en función de la superficie impactada.

Las magias en el juego están gestionadas mediante dos scripts principales: Magias y Magias Enemigos. El primero contiene dos enumeradores que incluyen los nombres de las magias disponibles para el jugador y los estados asociados, los cuales cambian en función de la magia aplicada. Este script no se aplica directamente a ningún objeto, ya que su propósito es permitir el acceso a los enumeradores.

```
public enum MagiasDisponibles
{
    Null,
    Fuego,
    Rayo,
    Agua
}

public enum EstadosDisponibles
{
    Null,
    EnLlamas,
    Evaporacion,
    ConCarga,
    Electrocutado,
    Mojado
}
```

Ilustración 6 Enumeradores de las Magias disponibles para el jugador y de los estados

Por otro lado, el script Magias Enemigos está asignado a los enemigos del juego. Este se encarga de gestionar los estados y sus efectos. Cada enemigo almacena dos magias: la correspondiente al último disparo recibido y la del disparo anterior. Además, cada enemigo

tiene un estado que cambia en función de la última magia recibida. Cada magia tiene un estado asignado: el agua aplica el estado mojado, el fuego en llamas y el rayo con carga. Dependiendo del estado almacenado y la magia recibida, se generan reacciones específicas. Por ejemplo, Evaporización, que provoca que los enemigos pierdan de vista al jugador, o Electrocutado, que reduce la vida del enemigo y crea un área de efecto que, al impactar a otros enemigos en estado mojado, también les aplica el estado electrocutado.

Inicialmente, se planeó implementar un sistema complejo basado en una State Machine (Máquina de Estados) para los enemigos. Sin embargo, este enfoque fue descartado, ya que solo se diseñó un tipo de enemigo cuerpo a cuerpo, que no requería mecánicas avanzadas. En su lugar, los enemigos se gestionan mediante el NavMesh de Unity, que ofrece un sistema de pathfinding (algoritmo de navegación), facilitando el movimiento de los enemigos por el entorno de manera eficiente. Además del NavMesh, los enemigos cuentan con un script que utiliza un sistema de estados para gestionar su comportamiento. Este sistema define los estados en los que se encuentra cada enemigo, como patrullando, atacando o persiguiendo al jugador, lo que permite controlar cómo reaccionan ante el entorno y el jugador.

También se añadieron varias funciones complementarias para mejorar la experiencia del jugador, como una que permite que los enemigos roten suavemente hacia el objetivo, y otra que evita que el jugador se coloque encima de los enemigos, contribuyendo a una jugabilidad más fluida y realista. Además, se implementó un sistema de ragdoll para que los enemigos adopten una física realista al ser derrotados, ofreciendo una animación más natural en su caída, lo que contribuye a una experiencia visual más satisfactoria.

En cuanto a la interacción en el juego, se implementaron dos tipos de objetos con los que puedes interactuar: las luces y los fuegos. Las luces de la ciudad deben ser encendidas y los fuegos apagados, y estas tareas se realizan mediante disparos. Dependiendo de la

magia activa en el momento del disparo, la tarea será cumplida o no. Ambos objetos están señalizados visualmente en el entorno para que el jugador pueda identificarlos fácilmente y cumplir con los objetivos asignados.

El diseño del juego incluye un sistema cíclico en el que los enemigos reaparecen y las tareas se reinician tras cierto tiempo, manteniendo un flujo continuo de acción. Además, completar tareas otorga al jugador tiempo adicional, fomentando la interacción constante con el entorno. Este sistema contribuye a la sensación de ritmo rápido y repetitivo, característica del estilo arcade. El objetivo principal es eliminar el mayor número posible de enemigos en el tiempo que dure la partida.

Por otro lado, se creó un sistema de diálogos mediante Scriptable Objects (objetos de Unity que permiten almacenar datos de forma eficiente y reutilizable), lo que facilita la creación y edición de diálogos de manera sencilla y accesible. Este sistema permite escribir y almacenar los diálogos en un formato organizado, optimizando el flujo de trabajo durante el desarrollo.

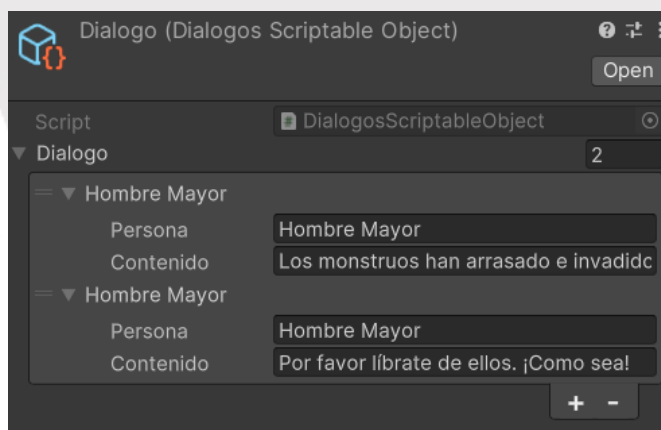


Ilustración 7 Ejemplo de un Scriptable Object que almacena uno de los diálogos

Con el objetivo de fomentar la competitividad entre los jugadores, se implementó un sistema de posicionamiento global (ranking). Este sistema fue aplicado mediante una herramienta llamada Leaderboard Creator (Jumagaliyev, s.f.), desarrollada por Danial

Jumagaliyev, un desarrollador de videojuegos y creador de contenido online. La herramienta permite la creación y gestión de tablas de clasificación en la plataforma de distribución digital Itch.io. Mediante una conexión entre el juego y dicha plataforma, es posible crear, borrar o modificar entradas en la base de datos, lo que facilita el manejo dinámico del ranking de los jugadores.

```
3 referencias
public void LoadEntries()
{
    Leaderboards.MagicShooterLeaderboard.GetEntries(entries =>{...});
}

1 referencia
public void SetEntry(string username, int score)
{
    Leaderboards.MagicShooterLeaderboard.UploadNewEntry(username, score, isSuccessful =>{...});
}
```

Ilustración 8 Funciones utilizadas para cargar e insertar las entradas de la Leaderboard (base de datos)

Finalmente, el menú de opciones del juego incluye los ajustes básicos, tales como la posibilidad de activar o desactivar la pantalla completa, cambiar la resolución, ajustar los niveles de volumen del juego y modificar la sensibilidad de la cámara. Estos ajustes permiten al jugador personalizar su experiencia de juego de acuerdo con sus preferencias.

5.3.2 Diseño y Construcción de Niveles

El diseño inicial de los niveles se planteó con la idea de una escuela de magia, consistiendo en un conjunto de salas y pasillos en los que el jugador debía derrotar a todos los enemigos para salvar a los alumnos. Se propuso un sistema de progresión basado en niveles separados. El primer nivel estaba diseñado para cumplir la función de tutorial, en el cual se enseñaban las mecánicas básicas de movimiento, disparo y el uso de una única magia. Para reforzar estos conceptos, se diseñó un nivel simple con pocos enemigos. A medida que el jugador avanzaba, los niveles introducirían nuevas magias, puzzles y obstáculos que fomentaban el uso de las reacciones entre las magias.

Debido a limitaciones de recursos, esta idea se descartó y se optó por un diseño basado en un único nivel, permitiendo también darle un estilo más arcade. Se diseñó un escenario de tamaño intermedio con diferentes tareas que el jugador debe completar, como derrotar a los enemigos, encender las luces y apagar los fuegos en la ciudad.

Para la construcción del escenario, se utilizó un entorno preconstruido del pack de assets Low Poly Fantasy Medieval Village Vol.2 de JustCreate (JustCreate, s.f.), disponible en la Unity Asset Store. Este escenario se modificó para adaptarse a las mecánicas del juego, añadiendo los objetos con los que el jugador puede interactuar como las luces y los fuegos. Estos objetos se distribuyeron de manera estratégica en el nivel, fomentando que el jugador explore el entorno para cumplir con las tareas. Además, se señalaron visualmente para facilitar su identificación y mejorar la accesibilidad a las mecánicas del juego.

A pesar de que el diseño de un solo nivel simplificaba la progresión, se mantuvo el ritmo arcade a través de un ciclo continuo en el que los enemigos reaparecen y las tareas se reinician en intervalos de tiempo determinados. Completar tareas como encender luces o apagar fuegos otorga tiempo adicional, incentivando al jugador a centrarse en estas acciones. Sin embargo, al derrotar a los enemigos no se añade tiempo, lo que mantiene el enfoque en la gestión de las tareas para prolongar la partida. Este ciclo continuo refuerza el estilo arcade del juego, manteniendo la acción constante y la rejugabilidad alta.

El diseño de este nivel, aunque más simple en comparación con la idea inicial, está enfocado en maximizar la acción y la competitividad, permitiendo al jugador mejorar su rendimiento en cada partida mediante la repetición de tareas y la eliminación de enemigos.

5.3.3 Creación de Arte y Animaciones

Desde el comienzo del proyecto, se decidió utilizar un pack de assets de la Unity Asset Store para la creación del arte del escenario. Esta decisión se tomó considerando que el enfoque principal del proyecto era la programación y el diseño del videojuego, y que no se disponía de habilidades artísticas avanzadas. Tras una búsqueda de conjuntos de assets que combinaran una temática escolar con un estilo medieval, se determinó que no era posible encontrar un pack que cumpliera ambos requisitos. Finalmente, se optó por el pack Low Poly Fantasy Medieval Village Vol.2 de JustCreate (JustCreate, s.f.), el cual no tiene temática escolar, pero proporcionaba el estilo visual medieval que se requería para el entorno del juego. Este pack fue modificado para adaptarse a las mecánicas y requerimientos específicos del proyecto.

En cuanto al diseño del personaje principal, se realizó el modelado y texturizado de los brazos del protagonista en estilo low poly, siguiendo la estética del paquete de assets seleccionado. El proceso de modelado y texturizado se llevó a cabo en Blender, utilizando un texturizado simple con colores planos, con el objetivo de lograr un estilo cartoon que se ajustara al resto de los elementos del juego.

Para el diseño del arma, se utilizó un modelo de pistola proporcionado por Single Sapling Games (Single Sapling Games, s.f.), un creador de contenido y desarrollador de videojuegos. Aunque el modelo del arma fue tomado de este creador, la textura se realizó de forma personalizada en Blender, manteniendo la coherencia con el estilo visual general del proyecto.

La animación de los brazos se desarrolló en conjunto con la pistola, ya que el jugador la lleva equipada todo el tiempo. El árbol de animaciones consta de tres animaciones principales: una animación idle, en la cual los brazos y el arma se mueven de arriba a abajo

en bucle; una animación de recarga simple, donde el arma se agita ligeramente; y una animación de disparo, que acompaña a la acción del jugador al atacar. Estas animaciones fueron diseñadas para mantener la coherencia con el estilo visual y la jugabilidad del juego, proporcionando una experiencia fluida.

En el juego se incluye un personaje que simula a un ciudadano de la ciudad, quien solicita ayuda al jugador. Este personaje es un modelo obtenido de Sketchfab, creado por François Espagnet (Espagnet, s.f.). Las animaciones y el rigging fueron realizados mediante Mixamo, una plataforma web que facilita la creación de animaciones y rigging para modelos 3D.

El enemigo del videojuego es un modelo 3D de un zombie en estilo low poly, también obtenido de Sketchfab, creado por EliveltonX (EliveltonX, s.f.). Al igual que el personaje ciudadano, las animaciones y el rigging del enemigo se realizaron utilizando Mixamo.

Para los efectos visuales del juego, se seleccionó un pack de assets con estilo cartoon. El pack utilizado fue Cartoon FX Remaster Free (Moreno, s.f.). De este pack se emplearon los efectos visuales para representar elementos como el fuego, el rayo, la electrocución y la evaporación, que también se aplicaron a las tareas del juego. Además, se crearon dos efectos visuales personalizados: un shader de agua con un estilo estilizado y varios efectos gráficos relacionados con la pistola y el disparo, como el fogonazo del cañón, el brillo del impacto y el golpe del impacto.

5.4 Periodo de Prueba

Aquí inicia uno de los procesos críticos del desarrollo de un videojuego, el periodo de pruebas.

La fase de período de prueba, también conocida como fase de testing, es una etapa crítica en el desarrollo de un videojuego donde se realiza una evaluación exhaustiva del juego para identificar y corregir errores, optimizar el rendimiento y mejorar la experiencia del usuario. Esta fase asegura que el juego esté listo para el lanzamiento al público, verificando que todas las mecánicas funcionen correctamente, que no haya bugs críticos y que el juego sea divertido y accesible para los jugadores.

5.4.1 Pruebas Internas (Alpha Testing) y Pruebas Externas (Beta Testing)

Durante el proceso de producción, se realizaron diversas pruebas tanto internas como externas con el objetivo de identificar y corregir errores, así como de ajustar las mecánicas del juego para optimizar la experiencia del usuario. Estas pruebas se llevaron a cabo mediante la creación de varias versiones preliminares (builds) del juego.

Las pruebas internas (Alpha Testing) se realizaron de manera continua a lo largo del desarrollo. Se probó cada nueva versión internamente, revisando aspectos como la estabilidad del juego, el rendimiento en diferentes dispositivos, la correcta implementación de las mecánicas y la coherencia visual de los elementos gráficos. Los errores y problemas detectados durante esta fase fueron documentados y corregidos en versiones posteriores.

En cuanto a las pruebas externas (Beta Testing), se enviaron varias builds a compañeros del grado y a personas de confianza ajenas al equipo de desarrollo. El objetivo de estas pruebas era obtener retroalimentación de usuarios con una perspectiva fresca, quienes no estaban familiarizados con el proceso de desarrollo. Las pruebas externas se centraron en la jugabilidad, la usabilidad de la interfaz, la curva de aprendizaje de las mecánicas, y la percepción general del jugador. Los comentarios y sugerencias obtenidos fueron evaluados y, cuando fue posible, se implementaron mejoras o ajustes en el juego.

Ambas fases de prueba fueron fundamentales para detectar errores y refinar las mecánicas del juego, contribuyendo al resultado final del proyecto.

5.5 Prelanzamiento y lanzamiento

Para finalizar el desarrollo se iniciará una fase de prelanzamiento y lanzamiento.

La fase de prelanzamiento y lanzamiento es la etapa final del desarrollo de un videojuego, donde se realizan las últimas preparaciones antes de poner el juego a disposición del público y se lleva a cabo el lanzamiento oficial. Durante esta fase, se finalizan los ajustes y optimizaciones, se desarrollan y ejecutan las estrategias de marketing, y se asegura que todos los aspectos del juego y su distribución estén listos para una presentación exitosa al mercado.

5.5.1 Preparación Final y Optimización

La etapa de preparación final y optimización se realizó tomando en cuenta los resultados de las últimas pruebas internas y externas. Estas pruebas revelaron varios aspectos que requerían ajustes para mejorar tanto la jugabilidad como el rendimiento general del juego.

Uno de los principales comentarios recibidos durante las pruebas externas estuvo relacionado con la velocidad de movimiento del personaje y los enemigos, que se percibió como excesivamente rápida. Para solucionar esta cuestión, se ajustaron los valores de movimiento tanto del personaje como de los enemigos, y se recortaron algunas de las animaciones de los enemigos para lograr una mayor coherencia en el ritmo del juego.

También se detectó un problema de bajo rendimiento debido a la baja tasa de FPS (frames por segundo). Inicialmente, se consideró que el origen del problema podría estar

relacionado con los enemigos, ya que parecían consumir una cantidad considerable de recursos. Como solución, se implementó un sistema de actualización personalizada para los enemigos, en el cual el update de sus acciones no se ejecutaba de manera constante, sino en intervalos de tiempo específicos. Sin embargo, esta solución no mejoró el rendimiento y afectó negativamente el comportamiento de los enemigos, por lo que finalmente se descartó.

Después de más análisis, se concluyó que el problema estaba vinculado a un elevado número de draw calls en pantalla, lo que sobrecargaba la capacidad de procesamiento gráfico. Durante esta fase de optimización, se añadió un efecto de pixelado con fines estéticos, que incidentalmente contribuyó a mejorar el rendimiento del juego al reducir la carga gráfica y aumentar la tasa de FPS de manera significativa. Este efecto permitió mejorar la fluidez del juego sin sacrificar la calidad visual.

5.6 Publicación

Para la publicación de la versión Alpha del juego, se ha creado una página en la plataforma Itch.io, la cual ha sido personalizada mediante HTML con el objetivo de lograr un diseño visual acorde al estilo pixelado y retro del juego. El diseño de la página refleja la estética del propio juego, utilizando elementos que evocan la sensación de un título clásico de acción y magia.

En la página de Itch.io, se han incluido descripciones detalladas del juego, capturas de pantalla y toda la información pertinente para que los jugadores comprendan las mecánicas básicas y el contexto general del proyecto. Se ha prestado especial atención a la presentación visual y a la estructura de la información para garantizar que sea atractiva y fácil de navegar.

Con la intención de alcanzar un público más amplio, toda la página ha sido redactada en inglés, facilitando así el acceso a jugadores de diversas regiones y comunidades internacionales. Esta decisión permite expandir el alcance del juego más allá de una audiencia local, incrementando las posibilidades de interacción y feedback de una comunidad global de jugadores.

6 Conclusiones

Este Trabajo de Fin de Grado tenía como objetivo el desarrollo y publicación de una versión Alpha de un videojuego indie, cubriendo todas las etapas del proceso de desarrollo, con un enfoque principal en la programación y el diseño, y en menor medida en el apartado artístico. Se puede afirmar que los objetivos planteados han sido en gran medida alcanzados, permitiendo aplicar de manera efectiva los conocimientos adquiridos durante el Grado en Diseño y Desarrollo de Videojuegos y Experiencias Interactivas.

El proyecto ha logrado crear una versión Alpha funcional, en la que se implementaron y optimizaron las mecánicas básicas de un videojuego tipo First Person Shooter (FPS). La programación y el diseño del nivel se realizaron utilizando el motor Unity, tal como se había planeado.

En cuanto al apartado artístico, aunque no era el enfoque principal del proyecto, se logró una integración estética coherente mediante la utilización de recursos disponibles en la Unity Asset Store y el modelado y texturizado de elementos personalizados, como los brazos del protagonista y la pistola. Las animaciones fueron generadas utilizando Mixamo, y se integraron de manera funcional en el juego. Aunque la producción de arte fue limitada, se logró cumplir con la cohesión estética y funcional propuesta para la versión Alpha.

El proceso de optimización también fue una parte fundamental del desarrollo. Inicialmente, surgieron problemas de bajo rendimiento debido a una tasa de frames reducida,

que se identificaron como resultado de un elevado número de draw calls en pantalla. Aunque se intentó implementar un sistema de update personalizada para los enemigos, el problema se resolvió finalmente al aplicar un efecto de pixelado en el postprocesado del proyecto. Esta solución redujo de manera considerable los draw calls y mejoró significativamente el rendimiento.

En términos de publicación, se ha desarrollado y ejecutado un plan para la publicación de la versión Alpha en plataformas digitales como Itch.io. Además, se implementó un sistema de ranking global, diseñado para fomentar la competitividad entre los jugadores, utilizando la herramienta Leaderboard Creator (Jumagaliyev, s.f.).

No obstante, a pesar de los logros alcanzados, existen áreas que podrían mejorarse en futuras iteraciones. La inteligencia artificial de los enemigos podría ampliarse para añadir comportamientos más complejos y adaptativos, mejorando así la experiencia de juego. Además, se podría considerar la expansión del contenido del juego con nuevos niveles, enemigos y magias.

En cuanto al aprendizaje obtenido, este proyecto permitió aplicar los conocimientos adquiridos en diversas disciplinas como la programación, diseño de niveles, optimización de rendimiento y la integración de elementos artísticos. Se ha adquirido un mayor dominio del motor Unity y herramientas relacionadas, como Blender para el modelado 3D, Mixamo para las animaciones y Scriptable Objects para la gestión de diálogos y datos. Además, se ha alcanzado un conocimiento actualizado sobre el proceso de desarrollo de videojuegos en la industria indie.

En conclusión, el proyecto ha cumplido con los objetivos principales, proporcionando una versión Alpha funcional del videojuego con una experiencia de juego sólida y optimizada. Existen oportunidades de mejora y expansión en el futuro, pero los conocimientos y

habilidades adquiridos durante el proceso han sido valiosos para el desarrollo del proyecto y para futuros desafíos en el ámbito del desarrollo de videojuegos.



7 Bibliografía

- EliveltonX. (s.f.). *Sketchfab*. Obtenido de <https://sketchfab.com/3d-models/zombie-for-game-unity-1df959e0e0b34c04b5c894ad1b415580>
- Espagnet, F. (s.f.). *Sketchfab*. Obtenido de <https://sketchfab.com/3d-models/the-tavern-old-man-f9932a88070940edb0d1bac736e48aaa>
- Facultat d'Informàtica de Barcelona (Universitat Politècnica de Catalunya). (s.f.). *Retroinformàtica. El pasado del futuro*. Obtenido de <https://www.fib.upc.edu/retro-informatica/credits.html>
- Jensen, K. T. (14 de Septiembre de 2022). *PCMag*. Obtenido de <https://www.pcmag.com/news/the-complete-history-of-first-person-shooters>
- Jumagaliyev, D. (s.f.). *Itch.io*. Obtenido de <https://danqzq.itch.io/leaderboard-creator>
- JustCreate. (s.f.). *Unity Asset Store*. Obtenido de <https://assetstore.unity.com/packages/3d/environments/fantasy/low-poly-fantasy-medieval-village-vol-2-240465>
- Moreno, J. (s.f.). *Unity Asset Store*. Obtenido de <https://assetstore.unity.com/packages/vfx/particles/cartoon-fx-remaster-free-109565>
- Single Sapling Games. (s.f.). *Patreon*. Obtenido de <https://www.patreon.com/posts/54756498>
- Wijman, T. (7 de Septiembre de 2023). *Newzoo International B.V.* Obtenido de <https://newzoo.com/resources/blog/top-game-genres-2023>

8 Anexos

8.1 Ten Page

8.1.1 Proyecto



Título: Magic Shooter

Género: FPS (First Person Shooter) con elementos de fantasía

Plataforma objetivo: PC (publicado en Itch.io)

8.1.2 Resumen

Descripción: Magic Shooter es un FPS arcade en el que los jugadores utilizan una pistola junto con habilidades mágicas para derrotar enemigos en un entorno abierto. El juego presenta un único escenario en el que las tareas y enemigos se regeneran continuamente, creando un flujo constante de acción. Los jugadores deben combinar disparos con el uso estratégico de magia elemental para completar objetivos y derrotar oleadas de enemigos.

Objetivo del jugador: El jugador debe eliminar el mayor número posible de enemigos y completar tareas dentro del tiempo disponible, utilizando una combinación de disparos y magias elementales para aumentar la efectividad en combate.



8.1.3 Contexto

Historia: El juego no tiene una narrativa profunda, ya que su enfoque está en la acción arcade. El jugador se encuentra en un escenario devastado por criaturas hostiles, y debe usar una combinación de armas y magia para sobrevivir. A medida que el tiempo avanza, los enemigos reaparecen y las tareas del escenario se reinician, desafiando al jugador a mantenerse activo y estratégico.

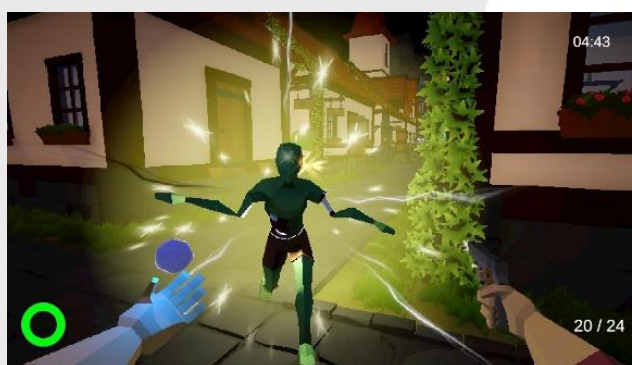
8.1.4 Mecánicas

Controles del Juego:

- Movimiento: WASD o Stick izquierdo.
- Disparo: Botón izquierdo del ratón o Gatillo derecho.
- Cambio de Magia: Teclas Q y E, o Gatillos superiores.

Mecánicas Principales:

- Combate: El jugador utiliza una pistola junto con magias elementales (agua, fuego y rayo) para derrotar enemigos. Las magias pueden ser combinadas para generar efectos más poderosos, como electrocutar enemigos mojados o crear un efecto de vapor para despistar a los enemigos en llamas.
- Ciclo de tareas: A lo largo del escenario, el jugador debe completar tareas como encender luces y apagar fuegos mediante el uso de los disparos. Estas tareas se reinician periódicamente, lo que añade un componente estratégico al juego.
- Sistema de puntuación: La puntuación final del jugador será la cantidad de enemigos derrotados



8.1.5 Progresión

Progresión: El juego está diseñado como una experiencia arcade sin niveles de progresión clásicos. El objetivo es perfeccionar las habilidades del jugador en cada partida, intentando superar la puntuación anterior. El jugador mejora su control de las magias y las armas a través de la práctica, aprendiendo a combinar diferentes elementos de forma efectiva.



8.1.6 Diseño de niveles

Nivel Principal: El juego tiene un único escenario de tamaño intermedio, basado en una ciudad mágica medieval, donde el jugador debe completar tareas y enfrentarse a oleadas de enemigos. El escenario tiene varias áreas con puntos de interacción (luces y fuegos), que deben ser activados o desactivados mediante el arma, esto recompensará al jugador con más tiempo en la partida. Los enemigos reaparecen en intervalos de tiempo regulares, lo que mantiene la acción constante.

8.1.7 Arte y Estilo Visual

Estilo Visual: El juego tiene un estilo low poly con gráficos retro y pixelados. Esto ayuda a crear una experiencia visual atractiva que complementa el estilo arcade del juego. El ambiente está basado en una ciudad mágica medieval.

Personaje: Los brazos del jugador y los enemigos han sido diseñados en low poly para mantener la coherencia visual con el entorno.



8.1.8 Animaciones

Animaciones: Las animaciones del personaje, específicamente los movimientos de los brazos al disparar y recargar el arma fueron desarrollados en conjunto con la pistola. Se utilizaron animaciones simples para mantener el ritmo arcade del juego. Los enemigos tienen animaciones básicas para atacar y para cuando son derrotados tienen un sistema de ragdoll.



8.1.9 Sonido y Música

Sonido: El juego cuenta con efectos de sonido para las principales acciones, como los disparos y los pasos del personaje, los cuales tienen un estilo más realista, en contraste

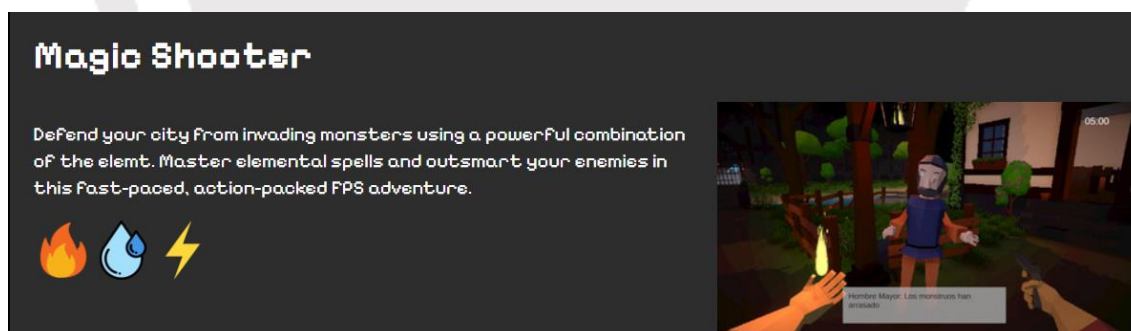
con la estética retro visual del juego. Estos efectos son clave para aumentar la inmersión del jugador y aportar dinamismo a la jugabilidad.

Música: La banda sonora del juego utiliza música retro rock, lo que añade una sensación de energía y ritmo constante a la partida. Este estilo musical acompaña perfectamente la acción rápida del combate, elevando la experiencia arcade sin ser invasiva, y motivando al jugador a mantenerse en movimiento.

8.1.10 Publicación y Monetización

Publicación: El juego ha sido publicado en Itch.io como una versión Alpha gratuita, con la página personalizada en HTML para reflejar el estilo visual del juego. El objetivo de la publicación en esta plataforma es alcanzar a un público global, permitiendo a los jugadores disfrutar de la experiencia y ofrecer retroalimentación.

Monetización: El juego se ha lanzado de manera gratuita.



8.2 Enlaces

Proyecto Unity: [ENLACE](#), el proyecto esta almacenado en GitHub y es público.

Enlace Itch.io: [ENLACE](#)