

Dreambound: Odyssey of Illusions

Autor: Marco Antonio Gómez Romo

Tutor: Ruth Contreras Espinosa

Profesor: Joan Arnedo Moreno

Grado de ingeniería informática

Videojuegos

Créditos/Copyright



Esta obra está sujeta a una licencia de Atribución-NoComercial-SinDerivadas [4.0 Internacional de Creative Commons.](https://creativecommons.org/licenses/by-nc-nd/4.0/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Dreambound: Odyssey of Illusions</i>
Nombre del autor:	<i>Marco Antonio Gómez Romo</i>
Nombre del colaborador/a docente:	<i>Ruth Contreras Espinosa</i>
Nombre del PRA:	<i>Joan Arnedo Moreno</i>
Fecha de entrega (mm/aaaa):	<i>01/2024</i>
Titulación o programa:	<i>Grado de ingeniería informática</i>
Área del Trabajo Final:	<i>Videojuegos</i>
Idioma del trabajo:	<i>Español</i>
Palabras clave	<i>Videojuego de puzles, 2D, Unity</i>
Resumen del Trabajo:	
<p>Este proyecto se enfoca en la creación de un videojuego que abarca diversas etapas, desde la concepción inicial, planificación y diseño, hasta el análisis, implementación y demostración. Asimismo, se definirán líneas para posibles expansiones del videojuego en el futuro.</p> <p>En cuanto a la idea central, se centra en la realización de un videojuego para ordenador en 2D con perspectiva desde arriba (Top-Down), dirigido a un público de entre 6 y 10 años. Dado que, en este rango de edad, los padres suelen desempeñar un papel fundamental en la introducción a los videojuegos, se ha optado por el género de aventura de puzles. Donde el jugador se enfrentará a juegos como <i>Light Out</i> que se centra en el razonamiento, <i>Simon</i> que estimula la memoria y la atención, y <i>Sokoban</i> que promueve la planificación estratégica, además, de tener que resolver series y preguntas lógicas que ayudan a estimular el pensamiento analítico del jugador.</p> <p>Para finalizar la conceptualización de esta idea, el juego será desarrollado utilizando el motor de videojuegos Unity. Este componente desempeñará un papel crucial en el proceso de aprendizaje del proyecto, brindando habilidades para aprovechar al máximo sus diversas funcionalidades. Esto incluirá desde la implementación de las físicas del personaje y las colisiones de los distintos elementos, hasta la lógica de los puzles y la interacción del protagonista con los mismos.</p> <p>Repositorio: https://github.com/mgomro/Dreambound</p>	

Abstract (in English, 250 words or less):

This project focuses on the creation of a video game that encompasses various phases, from initial conception, planning and design, to analysis, implementation and demonstration. Additionally, guidelines will be defined for potential expansions of the video game in the future.

Regarding the main concept, it focuses on the development of a 2D computer game with a Top-Down perspective, aimed at an audience between 6 and 10 years old. Given that, in this age range, parents often play a crucial role in introducing video games, the puzzle-adventure genre has been chosen. In this game, the player will face challenges like Light Out, which focuses on reasoning, Simon, which stimulates memory and attention, and Sokoban, which promotes strategic planning. Additionally, the player will have to solve series and logical questions that help stimulate the player's analytical thinking.

To conclude the conceptualization of this idea, the game will be developed using the Unity game engine. This component will play a pivotal role in the learning process of the project, providing skills to make the most out of its diverse functionalities. This will encompass everything from implementing the character's physics and handling collisions of various elements, to the logic of the puzzles and the interaction of the protagonist with them.

Dedicatoria

A mis padres, por el inmenso tiempo y esfuerzo que han invertido en mi educación y por los valiosos valores que me han inculcado, forjando la persona que soy hoy. A mi esposa, Elena, le dedico un agradecimiento especial por su apoyo incondicional a lo largo de estos años. Asimismo, a mi hija Elena, quiero expresarle mi gratitud por su contribución invaluable en el desarrollo de este proyecto. Sus ingeniosas ideas, su entusiasmo al probar el videojuego y sus opiniones constructivas han sido esenciales para su mejora constante. ¡Gracias por ser mi mayor fuente de inspiración y apoyo!

Resumen

Este proyecto se enfoca en la creación de un videojuego que abarca diversas etapas, desde la concepción inicial, planificación y diseño, hasta el análisis, implementación y demostración. Asimismo, se definirán líneas para posibles expansiones del videojuego en el futuro.

En cuanto a la idea central, se centra en la realización de un videojuego para ordenador en 2D con perspectiva desde arriba (Top-Down), dirigido a un público de entre 6 y 10 años. Dado que, en este rango de edad, los padres suelen desempeñar un papel fundamental en la introducción a los videojuegos, se ha optado por el género de aventura de puzzles. Donde el jugador se enfrentará a juegos como *Light Out* que se centra en el razonamiento, *Simon* que estimula la memoria y la atención, y *Sokoban* que promueve la planificación estratégica, además, de tener que resolver series y preguntas lógicas que ayudan a estimular el pensamiento analítico del jugador.

Para finalizar la conceptualización de esta idea, el juego será desarrollado utilizando el motor de videojuegos Unity. Este componente desempeñará un papel crucial en el proceso de aprendizaje del proyecto, brindando habilidades para aprovechar al máximo sus diversas funcionalidades. Esto incluirá desde la implementación de las físicas del personaje y las colisiones de los distintos elementos, hasta la lógica de los puzzles y la interacción del protagonista con los mismos.

Palabras clave

Videojuego de puzzles, 2D, Unity

Abstract

This project focuses on the creation of a video game that encompasses various phases, from initial conception, planning and design, to analysis, implementation and demonstration. Additionally, guidelines will be defined for potential expansions of the video game in the future.

Regarding the main concept, it focuses on the development of a 2D computer game with a Top-Down perspective, aimed at an audience between 6 and 10 years old. Given that, in this age range, parents often play a crucial role in introducing video games, the puzzle-adventure genre has been chosen. In this game, the player will face challenges like Light Out, which focuses on reasoning, Simon, which stimulates memory and attention, and Sokoban, which promotes strategic planning. Additionally, the player will have to solve series and logical questions that help stimulate the player's analytical thinking.

To conclude the conceptualization of this idea, the game will be developed using the Unity game engine. This component will play a pivotal role in the learning process of the project, providing skills to make the most out of its diverse functionalities. This will encompass everything from implementing the character's physics and handling collisions of various elements, to the logic of the puzzles and the interaction of the protagonist with them.

Keywords

Puzzle video game, 2D, Unity

Índice

1. Introducción.....	12
1.1. Introducción/Prefacio	12
1.2. Descripción/Definición.....	14
1.3. Objetivos generales.....	16
1.3.1. Objetivos principales.....	16
1.3.2. Objetivos secundarios	17
1.4. Metodología y proceso de trabajo	18
1.5. Planificación	19
1.6. Presupuesto	22
1.7. Estructura del resto del documento.....	24
2. Análisis de mercado	25
2.1. Público objetivo (i.e. <i>target audience</i>) y perfiles de usuario.....	25
2.2. Estado del arte	28
2.2.1. Historia y evolución del género	28
2.2.2. Referencias del género	30
3. Propuesta.....	34
3.1. Definición de objetivos/especificaciones del producto.....	36
3.2. Modelo de negocio	38
4. Diseño.....	40
4.1. Entorno de desarrollo elegido.....	40
4.1.1. Requisitos técnicos del entorno de desarrollo	43
4.2. Inventario de las herramientas utilizadas.....	44
4.3. Inventario de recursos utilizados	47
4.4. Arquitectura general de la aplicación	56
4.5. Arquitectura de clases del sistema de juego	58
4.6. Diagramas de navegación	64
4.7. Diseño de niveles.....	68
5. Implementación.....	71
5.1. Requisitos de instalación	71

5.2. Instrucciones de instalación	71
6. Demostración	72
6.1. Guía de usuario.....	72
7. Conclusiones y líneas de futuro	75
7.1. Conclusiones	75
7.2. Líneas de futuro	78
Bibliografía	79
Anexos.....	80

Figuras y tablas

Índice de figuras

Figura 1: Izq. Juego <i>Light Out</i> . Fuente: minijuegos.com Der. Juego <i>Sokoban</i> Fuente: wikipedia.org.....	14
Figura 2: Izq. Juego <i>Machinarium</i> . Fuente: steampowered.com Der. Juego <i>BabyBus</i> . Fuente: <i>portalprogramas.com</i>	15
Figura 3: Estilo artístico Primera sala.....	15
Figura 4: Mapa mental investigación preliminar.....	18
Figura 5: Diagrama de Gantt PEC 1.....	19
Figura 6: Diagrama de Gantt PEC 2.....	19
Figura 7: Diagrama de Gantt PEC 3.....	20
Figura 8: Diagrama de Gantt PEC 4.....	20
Figura 9: Diagrama de Gantt PEC 5.....	20
Figura 10: Diagrama de flujo.....	21
Figura 11: <i>Hunt the Wumpus</i> . Fuente: https://techland.time.com/	28
Figura 12: <i>Lemmings</i> . Fuente: https://en.wikipedia.org/	29
Figura 13: Portada juego <i>Myst</i> . Fuente: portal.33bits.net.....	30
Figura 14: <i>Machinarium</i> , arte <i>steampunk</i> . Fuente: playingclub.org.....	31
Figura 15: Escenario <i>Baba Is You</i> . Fuente: store.steampowered.com.....	32
Figura 16: Clío en su casa.....	34
Figura 17: Der. Morfeo – Sala <i>Light Out</i> . Izq. Clío- Sala series lógicas.....	35
Figura 18: Controles.....	35
Figura 19: Estadística obtenida del portal itch.io, consultado 10/12/2023.....	41
Figura 20: <i>Tileset</i> interior casa Escena 2.....	47
Figura 21: Izq. Clío, Der. Morfeo.....	48
Figura 22: <i>Tileset</i> suelo y parades Escena 3.....	48
Figura 23: <i>Tileset</i> elementos decorativos Escena 3.....	48
Figura 24: NPC cinemática final.....	49
Figura 25: Casilla inventario.....	49
Figura 26: Puntero del ratón.....	49
Figura 27: Botones y bobinas Teslas modificadas.....	50
Figura 28: Panel de código.....	50
Figura 29: Pizarras – Reto series lógicas.....	50
Figura 30: Paneles matraces.....	51
Figura 31: Suelo laboratorio modificado.....	51
Figura 32: Portapapeles y batería.....	51
Figura 33: <i>Sprites</i> robot.....	52
Figura 34: <i>Sprites</i> diseñados propios.....	52
Figura 35: Mapa conceptual elementos de la escena.....	56
Figura 36: Diagrama de clases del sistema de juego.....	58
Figura 37: Flujo de conversación con DialogueEditor.....	63

Figura 38: Cambio realizado a la interfaz de DialogueEditor	63
Figura 39: Arquitectura general escenas	64
Figura 40: Diagrama de flujo Sala 0-4.....	66
Figura 41: Diagrama de flujo Sala 5-9.....	66
Figura 42: Mapa completo nivel 1	68
Figura 43: Diagrama de flujo: patrón resolución de retos.....	73
Figura 44: Objetos de inventario	74

Índice de tablas

Tabla 1: Presupuesto hardware y mobiliario	22
Tabla 2: Presupuesto equipo humano	23
Tabla 3: Presupuesto software.....	23
Tabla 4: Presupuesto total	23
Tabla 5: Prototipo usuario 1	26
Tabla 6: Prototipo usuario 2	27
Tabla 7: Necesidades, preferencias y limitaciones usuarios.....	27
Tabla 8: Resumen características principales juegos: <i>Myst</i> , <i>Machinarium</i> y <i>Baba Is You</i>	33
Tabla 9: Aspectos de referencia aplicados a <i>Dreambound</i>	37
Tabla 10: Comparación entre Unity, Godot y GameMaker Studio.	41
Tabla 11: Requisitos técnicos mínimos Unity ver. 2022.3 (LTS)	43
Tabla 12: Efectos de sonidos del juego	55

1.Introducción

1.1. Introducción/Prefacio

Durante décadas, los videojuegos han llevado consigo un estigma que los vincula con una influencia negativa en niños y adolescentes, relacionándolos con el incremento de comportamientos agresivos, exposición a la violencia y la desconexión de la realidad. Y aunque es cierto que demasiadas horas sin supervisión con algún juego cuyo contenido sea violento puede llevar a estas consecuencias, es esencial subrayar que el impacto de los videojuegos está fuertemente ligado al diseño y propósito de cada título (García-Bullé, 2019).

Y a diferencia de la creencia generalizada, el uso moderado de los videojuegos no solo carece de consecuencias perjudiciales para los jóvenes, sino que también aporta una serie de beneficios considerables a su desarrollo. Entre los que se destacan el fomento del trabajo en equipo, la mejora de la memoria y otras funciones cognitivas, así como la facilitación del aprendizaje de idiomas y diversas materias. Además, los videojuegos también se presentan como herramientas eficaces para estimular las diversas inteligencias múltiples y la creatividad de los jóvenes, ofreciendo un amplio abanico de oportunidades para el crecimiento integral (García, 2020).

No obstante, es primordial reconocer que la mayoría de los niños y adolescentes tienden a jugar juegos que contienen algún tipo de violencia, como se recoge en un estudio realizado en el portal web qustodio.com (27 de agosto, 2020).

7 de los juegos más prohibidos por los padres coinciden con los más utilizados por sus hijos en el rango de edad de 4 a 9 años, y este porcentaje aumenta al 80% entre los mayores de 10 años. Entre los juegos más restringidos, la mayoría presenta contenidos violentos, lo que resalta la concienciación de las familias españolas sobre la presencia de violencia en los videojuegos (Equipo Qustodio, 2020).

Dada la relevancia de la influencia de los videojuegos en los niños y adolescentes, se busca desarrollar un videojuego completamente libre de violencia, que no solo brinde entretenimiento, sino que también tenga un enfoque educativo. Es por eso por lo que se ha optado por crear un juego del género de puzzles, diseñado específicamente para un público de 6 a 10 años.

Para ello, la historia de este juego gira en torno a un protagonista que se caracteriza por ser un soñador intrépido, cuyas aventuras se desenvuelven en un universo onírico repleto de desafiantes acertijos. Donde deberá explorar minuciosamente las distintas salas recolectando las pistas esenciales para resolver los enigmas que se le presenten. Este enfoque responde a la necesidad de proporcionar alternativas de entretenimiento y aprendizaje sin recurrir a contenidos violentos, en consonancia con la concienciación de las familias sobre la importancia de la elección de videojuegos para sus hijos.

Por último, la conceptualización de este juego se inspira en la dinámica de las mazmorras presentes en la saga *The Legend of Zelda*, donde la resolución de acertijos es esencial para enfrentarse al jefe final, en este caso, finalizar el nivel/sueño. Además, se adopta la idea de que el jugador debe explorar cada rincón de las distintas salas, influencia tomada del juego *Machinarium*. Por ejemplo, se implementa un laberinto con paredes invisibles en el que, para trazar el camino correcto, el jugador deberá recolectar previamente unas notas que detallan los pasos a seguir. Por otro lado, la narrativa de este primer nivel se cuenta a través de los elementos que el jugador encuentre en cada sala, un aspecto que se tiene en cuenta como característica destacable del juego *Myst*.

1.2. Descripción/Definición

Una vez adquirido los conocimientos de las distintas materias del grado y la oportunidad que brinda la realización de este trabajo, se parte con la idea de desarrollar un videojuego del género de puzzles para un público entre 6 y 10 años. Con el cual se pretende alcanzar dos pilares, que el jugador se sumerja en una historia y hacer que la resolución de los diversos rompecabezas sea entretenida y educativa.

Dado el público objetivo del juego, es esencial considerar la etapa de las *Operaciones concretas de Piaget*¹ (Cano, 25), que abarca precisamente este rango de edades, importante en el desarrollo cognitivo de los niños. En esta etapa, los pequeños comienzan a pensar de manera más lógica y concreta sobre situaciones específicas. Es un momento en el que la capacidad de resolución de problemas y el pensamiento lógico están en pleno auge por lo que es fundamental que los puzzles seleccionados sean apropiados y desafiantes para este grupo de edad.

En línea con estos principios, cada uno de los juegos elegidos ha sido meticulosamente seleccionado. Por ejemplo, alguno de los juegos que se enfrentará el jugador serán, *Light Out* que se centra en el razonamiento, *Simon* que estimula la memoria y la atención, y *Sokoban* que promueve la planificación estratégica, alineándose de manera precisa con las áreas cognitivas que se pretenden potenciar.



Figura 1: Izq. Juego *Light Out*. Fuente: minijuegos.com Der. Juego *Sokoban* Fuente: wikipedia.org

Además, la inclusión de enigmas que requieren descubrir elementos ocultos en el escenario agrega un nivel adicional de desafío, estimulando tanto la capacidad de observación como el pensamiento analítico del jugador, así como su destreza en la resolución de series y

¹ Jean Piaget, fue un psicólogo, epistemólogo y biólogo suizo, reconocido por sus aportes al estudio de la infancia y por su teoría cognitiva constructivista del desarrollo de la inteligencia.
Fuente: https://es.wikipedia.org/wiki/Jean_Piaget

preguntas lógicas que también se presentarán. Esta selección de rompecabezas, además de brindar entretenimiento y aspectos educativos, contribuye también al estímulo del sistema cognitivo.

En cuanto a la relevancia del tema elegido se debe a que actualmente existen multitud de juegos de este género que tienen muy buena acogida, como puede ser *Machinarium*, *Monument Valley* o *BabyBus*, este último con un enfoque para un público infantil con una gran colección de minijuegos de lógica. Y todos estos juegos comparten ciertas características que han sido decisiva para la elección del género, ya que ofrecen muchas horas de entretenimiento, suelen tener una curva de aprendizaje optimizada, sirven como entrenamiento para la mente y tienen una agradable inmersión visual.



Figura 2: Izq. Juego *Machinarium*. Fuente: steampowered.com Der. Juego *BabyBus*. Fuente: portalprogramas.com

Finalmente, la idea se concretará mediante la creación de un juego 2D con una perspectiva *Top-Down*, utilizando la plataforma *Unity*. El objetivo es lograr un producto final completo, que se compone de un primer nivel con diez salas con sus respectivos desafíos. Además, se planea la posibilidad de agregar nuevos escenarios con puzles adicionales en futuras actualizaciones, lo que permitirá expandir la narrativa de la historia.



Figura 3: Estilo artístico Primera sala

1.3. Objetivos generales

Listado y descripción de los objetivos del TF, ordenados por relevancia.

1.3.1. Objetivos principales

Objetivos de la aplicación/producto/servicio:

- Desarrollar un videojuego para todos los públicos, libre de contenido violento y lenguaje inapropiado.
- Garantizar que sea fácil de entender y que las acciones sean intuitivas, permitiendo la inclusión de un amplio público de jugadores, independientemente de su nivel de experiencia o habilidades previas.
- Elaborar un nivel de juego integral que incluya todos los puzles.
- Concebir una narrativa envolvente y ambientación atractiva.
- Desarrollar una experiencia que sea a la vez sencilla y atractiva.

Objetivos para el cliente/usuario:

- Proporcionar una experiencia satisfactoria al resolver los puzles.
- Estimular habilidades cognitivas como la memoria, el razonamiento, la percepción o la inteligencia.
- Fomentar la inmersión en la trama y el mundo del personaje.
- Fomentar la colaboración entre padres e hijos al enfrentar desafíos con mayor grado de dificultad durante el juego.

Objetivos personales del autor del TF:

- Reconocer y comprender los distintos elementos que conforman un videojuego.
- Desarrollar un producto desde un punto inicial hasta el final, tomando individualmente las acciones a seguir (Dirigir un proyecto propio).
- Aprender cómo usar un motor de desarrollo de juegos.
- Desarrollar la competencia necesaria para aplicar aspectos destacados en la creación de un videojuego dirigido a un público específico, adquiriendo el conocimiento clave para su implementación efectiva.

1.3.2. Objetivos secundarios

Objetivos adicionales que enriquecen el TF.

- Implementar animaciones de transición para resaltar zonas específicas.
- Enriquecer la representación de cada sala con elementos temáticos que describan su función o propósito.
- Adquirir conocimientos sobre estrategias de marketing efectivas para el lanzamiento de un videojuego.

1.4. Metodología y proceso de trabajo

El desarrollo del proyecto comienza con una fase inicial de investigación que se divide en dos vertientes. Primero, se elige un género adecuado para el perfil del usuario final seleccionado, se determina la historia, se conceptualiza el estilo artístico y se define qué tipos de puzzles se incluirán en el juego. Este proceso inicia con la creación de un primer mockup que encapsula estas ideas, seguido por la elaboración de un documento de diseño de videojuegos inicial.



Figura 4: Mapa mental investigación preliminar

Una vez definida la idea inicial, el siguiente paso consiste en recopilar información sobre los componentes técnicos que integran un videojuego. Por otro lado, se evalúa qué motor de videojuego se ajusta de manera óptima al estilo de juego seleccionado, y se inicia la búsqueda de *assets* iniciales para utilizar en las primeras etapas del desarrollo del proyecto. Cabe destacar que la búsqueda de recursos continuará a lo largo del proyecto, ya que la idea principal puede evolucionar para implementar mejoras o cambios necesarios en el producto final.

En la selección de la metodología para este proyecto, se contempla la aplicación de un enfoque ágil como Scrum. Esta elección se sustenta en la atención a dos aspectos: el tiempo ajustado para la entrega final y la necesidad de producir entregables periódicos a lo largo del desarrollo. Optar por Scrum garantiza la entrega de valor al concluir el proyecto, alineándose con uno de los principios fundamentales de esta metodología.

Como se detalla en la planificación, cada elemento del videojuego se divide en tareas de desarrollo con una duración de una a dos semanas, conocidas como *sprints*. Esta estructura de iteraciones cortas posibilita la entrega de versiones jugables en un periodo breve de tiempo. En cada iteración, se evalúan y retroalimentan las funcionalidades desarrolladas hasta el momento, facilitando la adaptación del producto según las necesidades cambiantes.

1.5. Planificación

Para alcanzar los hitos clave en la entrega de las PEC y completar las diversas fases de desarrollo del videojuego, se lleva a cabo una planificación meticulosa del proyecto. Esto incluye la consideración de los aspectos a abordar en cada PEC, así como la elaboración de una tabla de prioridades para los elementos más significativos del videojuego. Con esta estrategia se garantiza la entrega de un producto final de calidad. Y es por ello por lo que se asignan estimaciones de tiempo más holgadas a las tareas iniciales de las fases de desarrollo del videojuego.

Por otro lado, para crear un diagrama de Gantt más claro y conciso, se optó por definir tareas más generales en el desarrollo del videojuego. Dentro de estas tareas, se llevarán a cabo revisiones diarias del producto implementado y la posibilidad de agregar nuevos elementos que inicialmente no estén contemplados. Además, se establecerá un proceso continuo de investigación para explorar la mejor manera de implementar diversas funcionalidades en Unity a lo largo de todo el proyecto.

Diagrama de Gantt

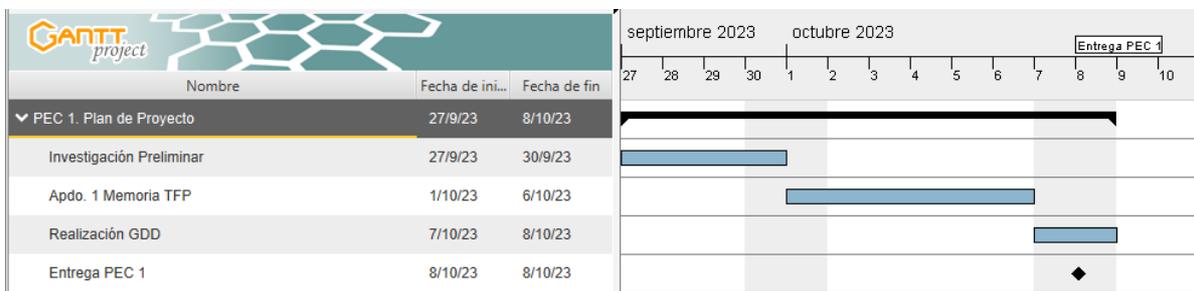


Figura 5: Diagrama de Gantt PEC 1

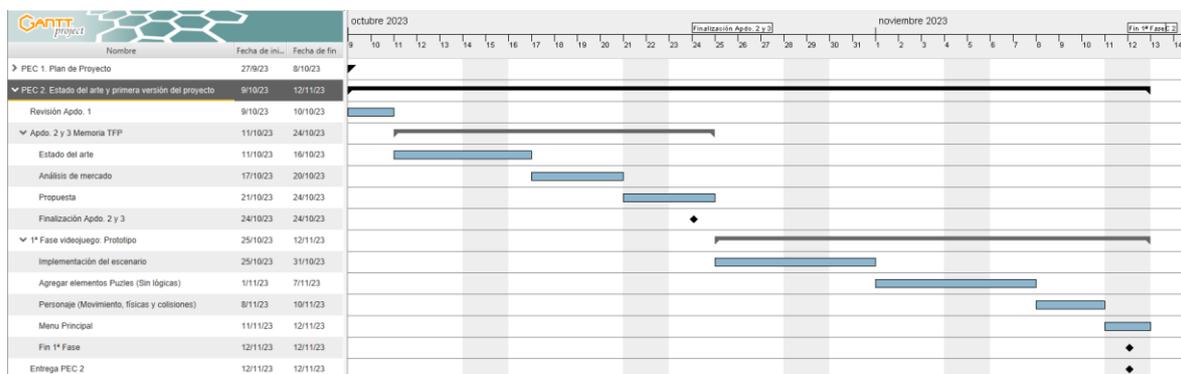


Figura 6: Diagrama de Gantt PEC 2

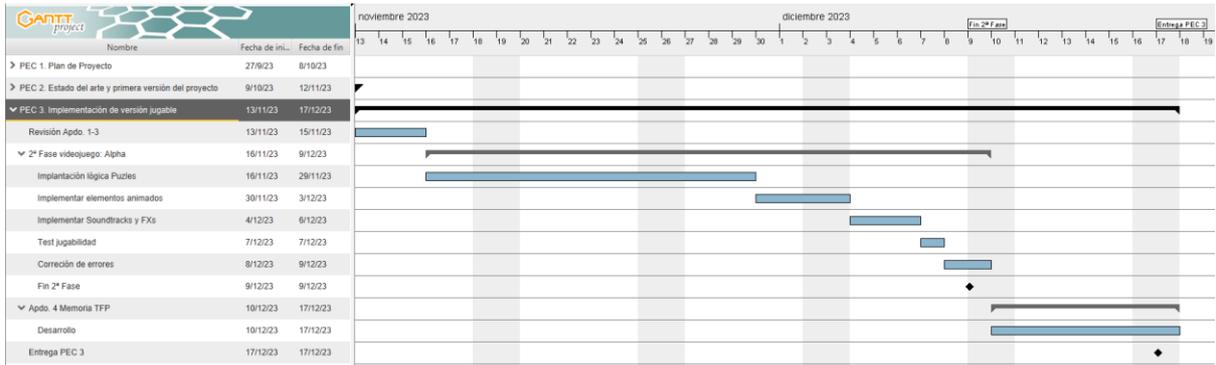


Figura 7: Diagrama de Gantt PEC 3

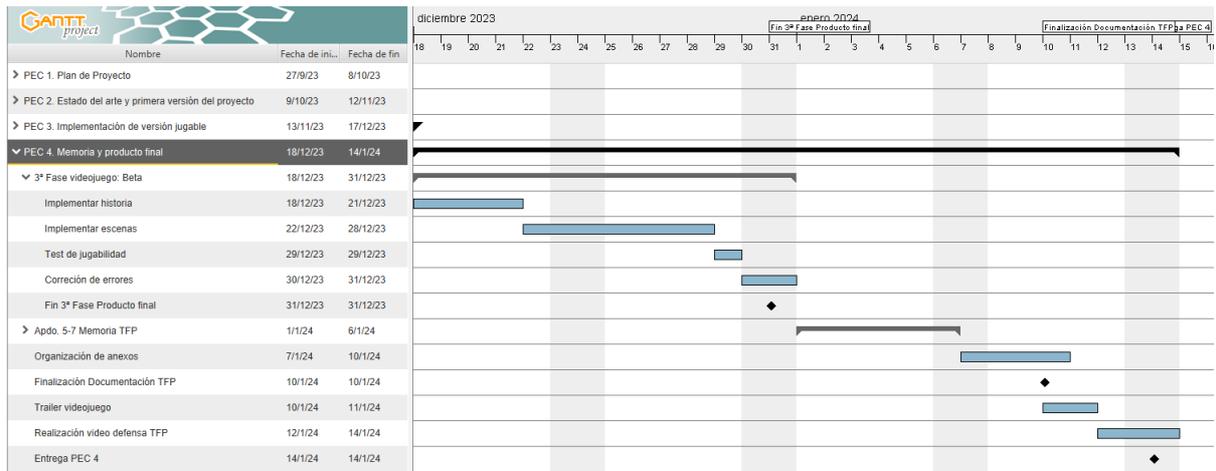


Figura 8: Diagrama de Gantt PEC 4

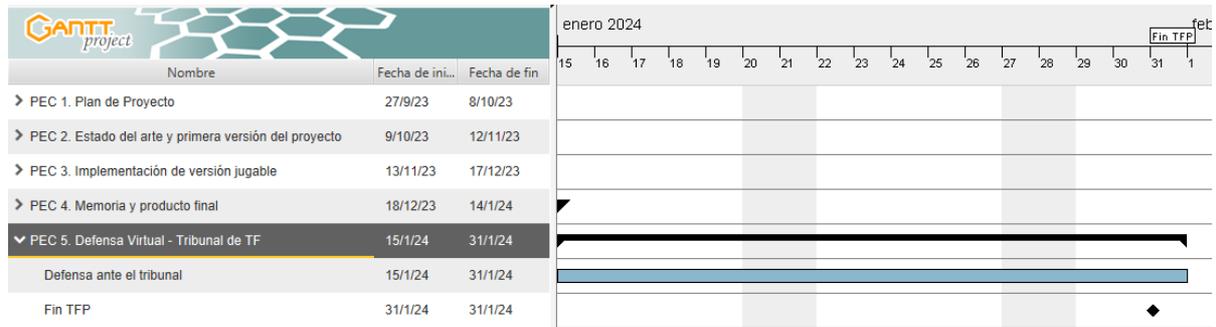


Figura 9: Diagrama de Gantt PEC 5

Diagrama de flujo

Dado que el diagrama de Gantt presenta el proceso de manera lineal y se está utilizando un enfoque ágil como Scrum, se ha creado el siguiente diagrama de flujo. En este, se destacan de manera más notable las iteraciones que ocurren entre los distintos procesos durante el desarrollo del videojuego, además de las revisiones periódicas a la documentación del trabajo.

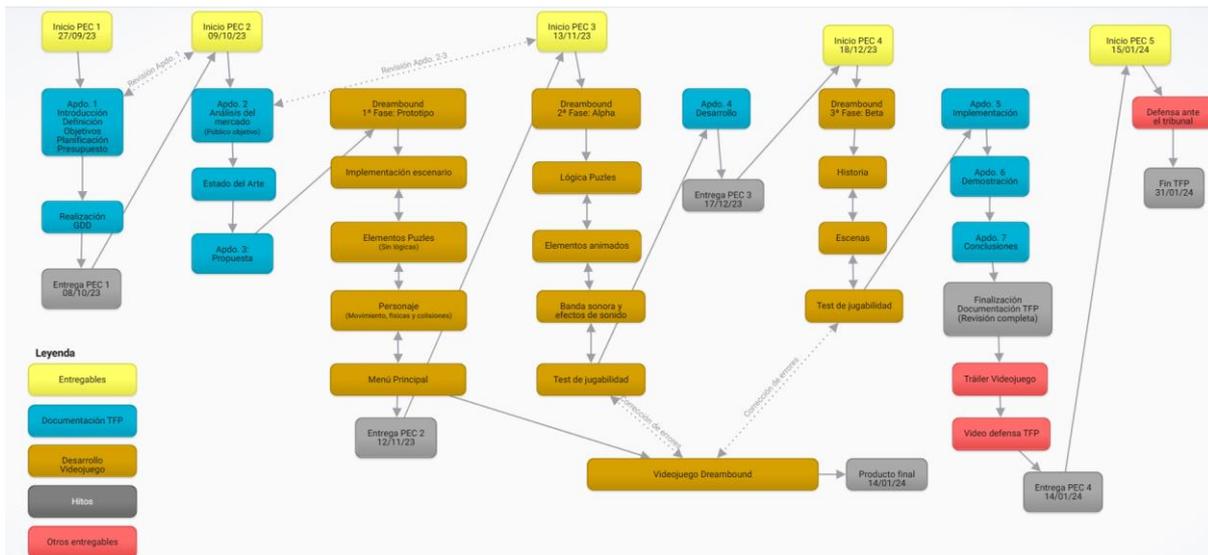


Figura 10: Diagrama de flujo

Actualización planificación inicial

Durante el desarrollo del proyecto, se ha decidido implementar nuevos objetivos que eran necesarios para poder ofrecer un videojuego más completo. Las actualizaciones a la planificación inicial han sido las siguientes:

- PEC 2
 - Selección de personajes 09/10/23 - 10/10/23
- PEC3
 - Sistema inventario 27/11/23 – 29/11/23
- PEC 4
 - Sistema de guardado 27/12/23 - 28/12/23

1.6. Presupuesto

Dada la entidad del proyecto de naturaleza académica, los presupuestos que se realizan son una estimación del coste total con los recursos usados tanto de equipamiento como software y se basan en la suposición de que el proyecto hubiese sido ejecutado por un programador de nivel junior de manera individual.

Equipamiento técnico		
Hardware	Descripción	Coste (€)
CPU	i7-12700K 32GB RAM RTX 3060	1500
Monitor 1	Asus VA24DQLB	150
Monitor 2	AOC G2490VXA	200
Teclado	Corsair K55	65
Ratón	Logitech MX Master 3	100
Auriculares	Logitech G733	100
Mobiliario		
Mesa		200
Silla		200
Total		2.515

Tabla 1: Presupuesto hardware y mobiliario

Para calcular el presupuesto del equipo humano, inicialmente se han determinado las horas necesarias para cada etapa del desarrollo del videojuego, considerando una jornada laboral de 8 horas y excluyendo los días no laborables del cómputo total. Además, se ha evaluado el costo por hora del trabajo de un programador junior estimado en 25 € (¿Cuánto Cobra un Desarrollador de Software?, s.f.),. El presupuesto total para el equipo humano se detalla en la siguiente tabla.

Equipo humano			
Documentación	Horas	Coste €/h	Gasto (€)
Planificación	64	25	1600
Diseño y análisis	216	25	5400
Estudio de mercado	32	25	800
Implementación			
Diseño gráfico	80	25	2000
Programación	272	25	6800
Test de jugabilidad / Corrección de errores	32	25	800
Total			17.400

Tabla 2: Presupuesto equipo humano

Software			
Descripción	Producto	Licencia	Coste (€)
Motor de videojuegos	Unity	Personal - Gratuita	0
Control de versiones	GitHub	Gratuita	0
Editor	Aseprite	Licencia adquirida vía <i>Steam</i>	16,79
Editor de texto	MS Office 365	Proporcionada por la UOC	0
Gestión de proyecto	GanttProject	Gratuita	0
Mapa Mental	Bubbl	Gratuita	0
Assets	<i>Tileset</i> ²	itch.io	3,62
	<i>Sprites</i>	Unity Store - Gratuito	0
	Audio	Unity Store - Gratuito	0
Total			20,41 €

Tabla 3: Presupuesto software

Una vez determinado el coste parcial de los diversos elementos necesarios para la ejecución del proyecto, se incorpora la sección de "otros gastos", que comprende los desembolsos asociados a la electricidad y la conexión a internet, estimados en 160 € durante los cuatro meses de duración del proyecto.

Presupuesto Total	
Presupuesto	Coste Parcial (€)
Equipamiento técnico	2.515
Equipo humano	17.400
Software	20,41
Otros gastos	160
Total	20095,41

Tabla 4: Presupuesto total

² *Tileset* utilizado para la escena introductoria del nivel 1. Ver apdo. 4.3 Inventario de recursos utilizados.

1.7. Estructura del resto del documento

Los siguientes apartados conformarán el resto del documento.

Análisis de mercado: se seleccionará un público objetivo para el cual se enfoca el desarrollo del videojuego, teniendo en cuenta los aspectos más relevantes de los prototipos de usuarios definidos.

Estado del arte: se realizará un estudio exhaustivo de los videojuegos más relevantes en la categoría de género elegido, seleccionando los aspectos más significativos de estos.

Propuesta: este apartado tiene como finalidad ofrecer de forma concisa la esencia del trabajo final, enfocándose especialmente en las características distintivas del videojuego que lo hacen atractivo para el público objetivo. Asimismo, se establecerán los objetivos y especificaciones que serán llevados a cabo.

Diseño: se detallarán los aspectos clave del desarrollo, abordando la estructura global de la aplicación, el funcionamiento minucioso de sus diferentes componentes y scripts, así como los recursos visuales empleados en el juego.

Implementación: en este capítulo se detallarán los requisitos y componentes necesarios para descargar, instalar y ejecutar el producto.

Demostración: se proporcionarán instrucciones de uso y una guía para el usuario.

Conclusiones y líneas de futuro: finalmente, en esta sección se abordarán las conclusiones personales sobre el proyecto, el proceso de trabajo y los resultados finales. Además, se presentarán sugerencias para posibles expansiones del proyecto en el futuro.

2. Análisis de mercado

2.1. Público objetivo (i.e. *target audience*) y perfiles de usuario

El desarrollo del proyecto se basa en la premisa que está dirigido a un público de entre 6 y 10 años. En este rango de edad los niños se encuentran en la etapa de educación primaria obligatoria en España³ y marca un periodo en el desarrollo de los niños, donde experimentan un rápido crecimiento físico, intelectual y emocional.

En consonancia con este apartado, a partir de las recomendaciones proporcionadas en este artículo (Bonet de Luna & Prochazcova, 2019), que abordan aspectos para esta fase del desarrollo infantil, se han identificado puntos clave a considerar para definir los perfiles de usuarios objetivo.

Desarrollo

Los niños se enfrentan a nuevas experiencias de aprendizaje, explorando la lectura, la escritura y conceptos más complejos. Además, la autonomía también cobra protagonismo, manifestándose en la capacidad de realizar tareas cotidianas por sí mismos.

Actitudes y habilidades parentales

Como padres, es fundamental fomentar actitudes positivas hacia el desarrollo de los niños. Reconocer y estar orgullosos de sus logros, así como proporcionar un ambiente donde puedan expresar sus ideas. Además, desempeñar roles parentales significativos y brindar apoyo en la organización del tiempo y tareas según sea necesario, respetando la autonomía cuando puedan hacerlo por sí mismos.

Gestión de tecnologías

Es necesario ser consciente de que las tecnologías, a pesar de ser beneficiosas, pueden volverse adictivas y peligrosas. Se debe evitar que consuman excesivo tiempo, afectando actividades importantes como el ejercicio, la lectura, los estudios y las relaciones familiares. Por lo que se debe prestar especial atención en la elección de programas de TV, juegos y sitios web, además de aplicar restricciones en los dispositivos tecnológicos.

³ La Educación Primaria tiene carácter obligatorio y gratuito. Comprende seis cursos académicos que conforman una progresión paulatina en el proceso enseñanza-aprendizaje, y que ordinariamente se realizan entre los seis y los doce años. Fuente: <https://www.educacionyfp.gob.es/>

En cuanto a los perfiles de usuarios, se han definidos dos perfiles. El primero hace referencia al usuario potencial que es quien jugará al videojuego y el segundo perfil corresponde a la figura parental, que juega un papel crucial como primera barrera de entrada ya que determina la idoneidad del juego para sus hijos.

Prototipo usuario 1	
Nombre	Nerea
Edad	6
Ocupación	1º Educación primaria obligatoria
Residencia	Burgos
Acerca del usuario	Nerea ha comenzado su educación primaria, lo que implica que ahora se enfrenta a desafíos más complejos que en su etapa preescolar. Por lo que ha comenzado a llevar deberes a casa, además, participa en actividades extraescolares. Pero, también tiene tiempo para disfrutar jugando en el parque con sus amigos. Y en casa le encanta jugar con sus juguetes, ver videos en el móvil o jugar a videojuegos.
Debilidades del usuario	Aunque está buscando más independencia, todavía necesita apoyo y guía de sus padres, especialmente cuando se enfrenta a situaciones nuevas o desafiantes. Y a veces puede tener dificultades para poner límites por sí misma, especialmente cuando se trata de determinar cuándo parar de jugar.
Solución del usuario	Principalmente, son los padres quienes se encargan de establecer límites en cuanto al tiempo de juego, ya sea con sus juguetes, videojuegos o el uso del dispositivo móvil. Sin embargo, a los padres tampoco les agrada interrumpir la diversión a mitad de una partida. Por ello, se considera que una sesión de juego tenga una duración recomendada de entre 45 y 60 minutos.

Tabla 5: Prototipo usuario 1

Prototipo usuario 2	
Nombre	Luis
Edad	38
Ocupación	Operario de fábrica
Residencia	Burgos
Acerca del usuario	Luis es el padre de Nerea, trabaja en una fábrica a jornada completa, algunas semanas de mañana y otras de tarde. En cuanto al tiempo libre que tiene es para compartirlo con su familia y realizar sus hobbies.
Debilidades del usuario	Debido al trabajo no dispone de todo el tiempo que quisiera para poder compartir con su familia, pero si tiene oportunidad pasa un rato con ellos ya sea yendo al parque, al cine o jugando en casa.
Solución del usuario	Para Luis, es primordial que tanto los contenidos que su hija consume como los juegos que juega sean libres de violencia, atractivos y educativos. Sin embargo, también es consciente de que su hija aún no ha aprendido a establecer límites para dejar de jugar. Por esta razón, prefiere que las sesiones de juego sean de corta duración.

Tabla 6: Prototipo usuario 2

A partir de los prototipos definidos, se obtienen conclusiones generales que facilitan la identificación de las necesidades, preferencias y limitaciones de los usuarios clave. La información recopilada en esta sección se establece como objetivo para la definición detallada de las especificaciones que el juego debe cumplir en relación con el público objetivo.

Prototipo usuario 1	
Necesidades	Apoyo y guía de sus padres, especialmente en situaciones nuevas o desafiantes. También puede beneficiarse de límites claros en el tiempo de juego para evitar excesos.
Preferencias	Participar en actividades extraescolares, jugar en el parque con amigos y dedicar tiempo a juegos en el ordenador y al uso de dispositivos móviles
Limitaciones	Dificultad para establecer límites por sí misma.
Prototipo usuario 2	
Necesidades	Pasar tiempo de calidad con su familia.
Preferencias	Concienciación acerca del contenido digital que su hija consume y los juegos que juega, sean educativos, atractivos y libres de violencia.
Limitaciones	No disponer de todo el tiempo que quisiera para compartir con su familia debido a sus responsabilidades laborales.

Tabla 7: Necesidades, preferencias y limitaciones usuarios

2.2. Estado del arte

2.2.1. Historia y evolución del género

El género de los juegos de puzzles ha evolucionado de manera notable desde sus comienzos en las décadas de 1970 y 1980 hasta convertirse en un pilar fundamental de la industria del videojuego actual. Esta transformación no solo ha sido impulsada por la innovación tecnológica, sino también por la influencia decisiva del desarrollo *indie*⁴. Muchos de los títulos más emblemáticos y revolucionarios de este género han sido creados por desarrolladores individuales o equipos reducidos (León, 2020).

Orígenes y primeros juegos (Años 70 y 80)

La popularidad de los videojuegos de puzzles empezó a crecer en las décadas de 1970 y 1980. *Hunt the Wumpus* (1972) es un ejemplo notable que, a pesar de su simplicidad, estableció las bases del pensamiento lógico y la resolución de problemas en los videojuegos. Sin embargo, fue en 1984 cuando el género alcanzó un nivel de popularidad significativo con el lanzamiento de Tetris, creado por Alexey Pajitnov. Este juego, con su mecánica sencilla pero profundamente adictiva, no solo se convirtió en un fenómeno global, sino que también redefinió el concepto de los juegos de puzzles.

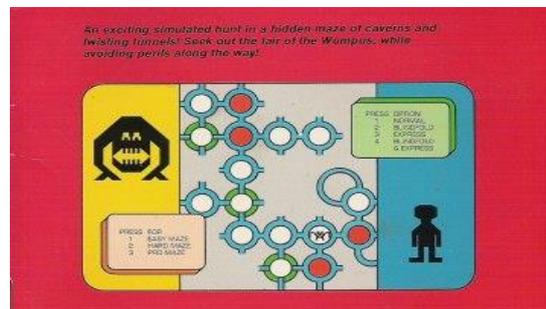


Figura 11: *Hunt the Wumpus*. Fuente: <https://techland.time.com/>

Expansión y diversificación (Años 90)

En la década de 1990, el género de puzzles vivió una expansión y diversificación significativas. Títulos como *Lemmings* (1991) y *Myst* (1993) presentaron nuevos tipos de desafíos de rompecabezas y narrativas envolventes. El auge de las consolas de videojuegos y los ordenadores personales facilitó que una audiencia más amplia disfrutara de una variada selección de juegos de puzzles.

⁴ El desarrollo de videojuegos independiente es el proceso de crear videojuegos sin el apoyo financiero de una distribuidora de videojuegos, generalmente son diseñados por un individuo o un pequeño equipo. Fuente: https://es.wikipedia.org/wiki/Desarrollo_de_videojuegos_independiente

Primeros juegos móviles (Años 2000)

La llegada de la tecnología móvil y el internet abrió un nuevo capítulo para los juegos de puzles. Juegos como *Bejeweled* (2001) y posteriormente *Candy Crush Saga* (2012) alcanzaron un éxito masivo en dispositivos móviles, aprovechando las interfaces táctiles y ofreciendo una jugabilidad rápida y accesible, perfecta para sesiones de juego breves.



Figura 12: *Lemmings*. Fuente: <https://en.wikipedia.org/>

Integración de Historia y Personajes (Años 2010 en adelante)

En los últimos años, los juegos de puzles han comenzado a incorporar elementos de narrativas más profundas y personajes bien desarrollados. Ejemplos de ello son *Portal* y su secuela, *Portal 2*, que combinan rompecabezas innovadores con una trama fascinante y un humor ingenioso. Otros títulos, como *The Witness* (2016), han explorado cómo integrar rompecabezas en entornos abiertos y narrativas misteriosas, enriqueciendo así la experiencia del jugador y llevando el género a nuevas alturas.

2.2.2. Referencias del género

En la actualidad, el desarrollo de videojuegos se ha consolidado como un campo en constante evolución, donde la creatividad y la innovación son pilares esenciales. No obstante, la mera intención de crear un juego basado en estos principios no garantiza el éxito. Para ello, es necesario comprender y aprender de las contribuciones significativas de aquellos que han dejado huella en la historia del desarrollo de videojuegos.

Por ello en la fase de conceptualización, se ha buscado inspiración en otros juegos del género de aventura gráfica, donde la resolución de puzzles es el núcleo de la experiencia. De estos juegos, se han extraído elementos fundamentales para integrar en el proyecto y de este modo evitar iniciar el desarrollo desde cero.

Myst

Un juego de aventura gráfica de apuntar y clicar (*point-and-click*) diseñado por los hermanos Robyn y Rand Miller bajo el sello propio Cyan, Inc., lanzado para Macintosh en 1993. Donde la narrativa se inicia con el jugador siendo transportado a un misterioso universo a través de un libro especial, desencadenando así una odisea de descubrimiento y resolución de enigmas. A medida que los jugadores desentrañan los secretos de *Myst*, se embarcan en un viaje por cuatro impresionantes mundos, conocidos como Eras, donde la trama se despliega para revelar las historias de fondo que entrelazan a los personajes (Miguel, 2020).



Figura 13: Portada juego *Myst*. Fuente: portal.33bits.net

Este juego es ampliamente reconocido como un hito en la historia de los videojuegos por varias razones:

Narrativa intrigante y mundo inmersivo

- *Myst* presenta una narrativa rica y envolvente que gira en torno a la exploración de mundos misteriosos, cada uno con su propia historia y desafíos.
- Los jugadores se sienten inmersos en el universo de este juego desde el principio, gracias a la atmósfera única y la sensación de descubrimiento constante.

Gráficos revolucionarios

- En 1993, los gráficos de *Myst* eran impresionantes y representaron un gran salto en cuanto a realismo y detalle en comparación con otros juegos de la época.
- La capacidad de explorar entornos 3D estáticos y prerrenderizados era innovadora.

Puzles desafiantes e ingeniosos

- Los puzles eran el núcleo del juego y ofrecían un nivel de desafío que atraía a los jugadores que buscaban una experiencia más mental.
- Cada mundo tenía su propio conjunto de puzles, lo que agregaba variedad y profundidad al juego.

Banda sonora

- Contribuye significativamente a la inmersión del jugador en el entorno del juego y ayuda a establecer el tono y la atmósfera en cada una de las islas explorables.

Machinarium

Juego de aventura gráfica y puzles, desarrollado por Amanita Design. Fue lanzado al mercado en octubre del 2009 para Microsoft Windows, Mac OS X y Linux. En *Machinarium*, los jugadores se ponen en la piel de Josef, un pequeño robot con la misión de salvar a su novia secuestrada de los peligros de una ciudad robótica corrupta. La narrativa se desarrolla en un mundo de estética *steampunk*⁵, repleto de detalles intrincados que hacen de cada nivel una obra de arte (Machinarium Review, s.f.).



Figura 14: *Machinarium*, arte *steampunk*. Fuente: playingclub.org

⁵ Subgénero literario nacido dentro de la ciencia ficción especulativa que surgió durante la década de 1980. El subgénero incorporaba tecnologías y estéticas retro futuristas inspiradas la maquinaria industrial de vapor del siglo XIX. Fuente: <https://es.wikipedia.org/wiki/Steampunk>.

En cuanto a los aspectos destables de este videojuego:

Sorprendente diseño visual

- El juego presenta un estilo artístico único y detallado que sumerge a los jugadores en un universo de objetos metálicos, engranajes y paisajes industriales, creando una atmósfera que es a la vez nostálgica y futurista.

Narrativa sin diálogos escritos u orales

- La historia se narra a través de ingeniosas animaciones y escenas, lo que permite a los jugadores interpretar la trama y los objetivos sin la necesidad de palabras.

Desafiantes rompecabezas y acertijos

- *Machinarium* presenta una variedad de desafíos lógicos y mecánicos que requieren ingenio y creatividad para resolver, además de estar perfectamente integrados en el entorno.

Banda sonora

- La música complementa el ambiente industrial y aporta una capa adicional de inmersión al juego.

Baba Is You

Innovador juego de lógica creado por el desarrollador independiente Arvi Teikari, lanzado en marzo de 2019 inicialmente para Windows y Nintendo Switch. En este título, los jugadores desafían las convenciones tradicionales de los juegos al alterar las reglas de cada nivel. En cada escenario se presenta una variedad de elementos y personajes, además de palabras dispuestas en casillas que definen las reglas del rompecabezas. Resolver cada nivel implica la habilidad de reorganizar estas reglas, creando nuevas combinaciones que desbloquean el camino hacia la solución (Wikipedia, 2023).



Figura 15: Escenario *Baba Is You*. Fuente: store.steampowered.com

Características destacadas:

Manipulación de los puzles

- Los jugadores se convierten en los arquitectos de los rompecabezas al tener la capacidad de alterar los elementos en el escenario para llegar a la solución del nivel.

Creatividad y pensamiento lógico

- El juego estimula la creatividad, el pensamiento lógico y la exploración de conceptos abstractos sobre cómo las reglas de un sistema pueden ser manipuladas para lograr un objetivo.

Estética minimalista

- La estética del juego es sencilla pero efectiva, con gráficos limpios y una paleta de colores atractiva.

Banda sonora

- Acompaña a la estética minimalista del juego utilizando tonos y melodías simples pero cautivadoras, además de cambiar a medida que el jugador avanza en el juego, añade un elemento adicional de inmersión.

Una vez definidos los aspectos más reseñables de cada videojuego se recogen en la siguiente tabla:

	<i>Myst</i>	<i>Machinarium</i>	<i>Baba Is You</i>
Narrativa inmersiva	Si a través de elementos que se encuentran en el entorno.	Si, además de innovadora, ya que emplea ingeniosas animaciones y escenas, sin dialogo escrito u oral.	No destaca debido a la ausencia de una trama narrativa.
Mundo inmersivo	El escenario, historia, puzles y banda sonora conforman el mundo inmersivo.	El estilo artístico, cada pista para resolver los puzles y banda sonora conforman el mundo inmersivo.	La resolución innovadora de puzles es el pilar principal de inmersión en el juego.
Estilo artístico	Gráficos prerrenderizados. Gran innovación para su época	Diseño de un mundo <i>steampunk</i> , parte importante en la inmersión del jugador.	Sencillo pero atractivo.
Banda sonora	Añade una capa de inmersión al juego.	Añade una capa de inmersión al juego.	Añade una capa de inmersión al juego.
Puzles	Desafiantes e ingeniosos. ofrecían un nivel de desafío intelectual Cada mundo con sus propios puzles.	Desafiantes e ingeniosos. Integrados con el entorno y requiere de este para su solución.	Fomenta la creatividad y el pensamiento lógico al permitir al jugador ser el arquitecto de los rompecabezas de forma innovadora

Tabla 8: Resumen características principales juegos: *Myst*, *Machinarium* y *Baba Is You*.

3.Propuesta

Título: *Dreambound: Odyssey of Illusions*

N.º jugadores: 1

Género: Aventura de puzles

Plataforma: Microsoft Windows

Público objetivo: 6-10 años

Duración: 20-40 minutos cada nivel, dependiendo de la destreza del jugador.

Características claves

- Sin contenido violento y lenguaje inapropiado, apto para todos los públicos.
- Desafíos diseñados para ofrecer entretenimiento mientras promueven la actividad cerebral.
- Escenarios únicos para cada aventura que presentan una narrativa diferente en cada nivel y versiones adaptadas y distintivas de puzles conocidos.
- Gráficos *pixel art* de estilo atractivo, con gran cantidad de elemento en cada escenario que favorecen la inmersión del jugador en la historia de su personaje.

Narrativa

En *Dreambound*, el jugador podrá elegir entre ser Morfeo o Clío. Y junto a su personaje, se adentrará en una sucesión de sueños, explorando cada escenario y enfrentándose a desafíos enigmáticos en su afán por despertar. No obstante, como sugiere el título, nuestro protagonista está ligado a los sueños, dejando en suspenso la certeza de si ha conseguido despertar o tal vez se halla inmerso en otro nuevo sueño.

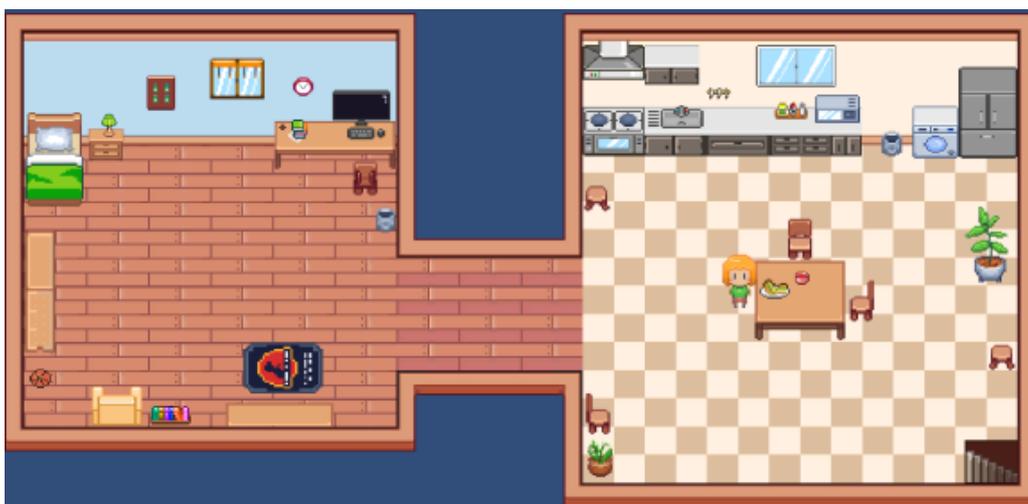


Figura 16: Clío en su casa

El juego

El juego comienza en un laboratorio donde se llevan a cabo diferentes experimentos que forman parte de los distintos retos que tendrá que resolver el jugador. Este escenario inicial consta de diez salas, cada una con su propia ambientación. El jugador debe resolver distintos desafíos para avanzar entre las salas como, *Light Out*, descifrar un código secreto a través de un acertijo, responder una serie de preguntas a un robot (tras encontrar una batería para activarlo), enfrentarse a juegos como *Simon*, atravesar un laberinto de paredes invisibles con la ayuda de notas previas, resolver series lógicas, ordenar cajas en su sitio correspondiente (*Sokoban*) y colocar objetos en su lugar correspondiente tras descubrirlos en las diferentes salas.



Figura 17: Der. Morfeo – Sala Light Out. Izq. Clío- Sala series lógicas.

Gráficos

Se utilizan gráficos en 2D con un estilo artístico de *pixel art* y una perspectiva *Top-Down*. Estos gráficos han sido desarrollados para una resolución FULL HD (1920x1080). La paleta de colores seleccionada es atractiva y vibrante, con tonalidades vivas.

Controles

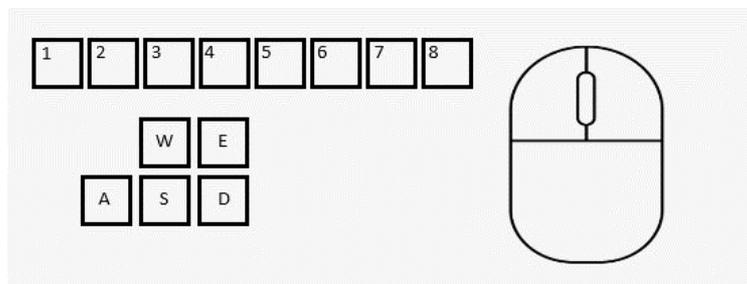


Figura 18: Controles

- Teclas W, A, S y D: control de movimiento del jugador.
- Tecla E: interactuar con los objetos del inventario.
- Teclas numéricas 1-8 elección de los objetos del teclado.
- Clic derecho ratón: interactuar con elementos del escenario.
- Clic izquierdo ratón: interactuar con elementos del dialogo, arrastrar elementos del inventario para ordenar.

3.1. Definición de objetivos/especificaciones del producto

Objetivos principales del producto

- Inicialmente el producto no será comercializado, el desarrollo del videojuego en conjunto con el desarrollo de este trabajo es con fines educativos.
- Juego apto para todos los públicos, enfocado en un público objetivo entre 6 y 10 años, libre de contenido violento y lenguaje inapropiado.
- El producto final contendrá las siguientes partes: menú principal, selección de personaje, introducción hacia el primer sueño, diez salas con sus respectivos desafíos y cinemática final del primer sueño.

Especificaciones del producto

Luego de abordar los apartados de público objetivo y análisis del estado del arte, se identifican una serie de especificaciones que desempeñarán un papel fundamental en el desarrollo del videojuego.

Características claves para el público objetivo:

- Proporcionar una experiencia de juego intuitiva y fácil de entender.
- Integrar elementos educativos para enriquecer el aprendizaje de manera lúdica.
- Garantizar la completa ausencia de contenido violento o lenguaje inapropiado.
- Ofrecer desafíos con niveles de dificultad apropiados, fomentando la interacción y participación de los padres para compartir tiempo de calidad juntos.
- Establecer una duración estimada de 20 a 40 minutos por nivel, permitiendo una experiencia de juego óptima y adaptada a las necesidades de los usuarios.

Aspectos destacados seleccionados de los juegos analizados en el estado de arte y como se abordarán en el desarrollo de *Dreambound*.

<i>Dreambound: Odyssey of Illusions</i>	
Narrativa inmersiva	A través de elementos que se encuentran en el entorno, que sirven tanto para contar la trama narrativa como ayuda a resolver los distintos desafíos.
Mundo inmersivo	La trama narrativa en conjunto con la resolución de los puzles, además del uso de una banda sonora y efectos de sonidos acorde a cada situación.
Estilo artístico	Gráficos 2D con perspectiva Top-Down, sencillos, pero con colores atractivos.
Banda sonora	Cada nivel contará con su propia banda sonora acorde a la temática del escenario, además de aplicar efectos de sonidos a los elementos del escenario.
Puzles	Puzles que estimulen la memoria, percepción, planificación de la estrategia para su resolución, etc. Principalmente son puzles ya conocidos pero adaptados al escenario.

Tabla 9: Aspectos de referencia aplicados a *Dreambound*.

3.2. Modelo de negocio

Este proyecto se inicia con una visión clara pero flexible: en sus etapas iniciales, no se busca la comercialización inmediata del producto. Sin embargo, el desarrollo de esta primera fase se contempla que el producto finalizado no es solo un fin en sí mismo, sino también un punto de partida trascendental para una futura comercialización. En esta sección se detalla el modelo de negocio para el videojuego con el objetivo que el producto inicial genere interés y reconocimiento en el mercado, y que, con el tiempo crezca y se adapte en respuesta a las opiniones de los usuarios.

Desarrollo y branding

En la etapa inicial de este proyecto, el juego no se ofrecerá para la venta, pero sí estará accesible al público. Esta decisión estratégica posiciona la versión inicial como una fase beta, que incluirá únicamente un nivel. Se opta por un enfoque meditado y no precipitado hacia la comercialización, eligiendo publicar el juego gratuitamente en plataformas como itch.io y Steam. Con el propósito de generar interés y reconocimiento entre la comunidad de jugadores, además de obtener el *feedback* valioso del público.

La retroalimentación obtenida en esta fase se considera un recurso esencial, a través del cual se espera captar las preferencias y expectativas de los usuarios, con el objetivo de ajustar y mejorar el juego en base a estas. De esta manera, la versión beta no solo actúa como una herramienta promocional, sino también como un medio integral para refinar y enriquecer continuamente la experiencia de juego, asegurando así que el producto final responda de manera óptima a las necesidades y deseos del público.

Plan de promoción

El lanzamiento gratuito del juego tiene como objetivo principal maximizar su exposición y accesibilidad. Pero, parte esencial de este plan incluye la promoción activa a través de canales de redes sociales, orientados específicamente a un público entre 6 y 10 años. Estos canales, que se especializan en contenido de videojuegos, son fundamentales para establecer una conexión directa con los usuarios potenciales del juego.

La inversión en promoción digital, aunque implica un costo significativo para el proyecto, se considera crucial para llegar a un mayor número de potenciales usuarios. Con esta táctica no

solo se busca aumentar la visibilidad del juego, sino también fomentar una comunidad activa y comprometida desde sus inicios.

Política de precios y estrategia de venta

A medida que el juego se desarrolle y expanda, especialmente después de alcanzar un hito de 10 niveles que ofrecerán entre 4 a 6 horas de juego, se planea lanzar el juego con un precio reducido y asequible. Con el objetivo de seguir llegando a una amplia audiencia mientras se comienza a generar ingresos. Además, es fundamental adoptar una política de precios flexible, una práctica común en juegos lanzados en plataformas como itch.io y Steam, considerando descuentos por tiempo limitado y precios especiales para los primeros compradores, adaptándose siempre a las respuestas y necesidades del mercado.

Desarrollo continuo y expansión a largo plazo

Dada la naturaleza no lineal del juego, con historias y escenarios variados en cada nivel, existe un potencial significativo para expandir el juego mediante la incorporación de expansiones. Estas expansiones, pensadas para agregar nuevos niveles al juego base, se desarrollarán con la premisa de mantener un precio accesible. Este enfoque garantiza que la expansión no solo mantenga la visibilidad del juego, sino que también continúe atrayendo a la audiencia establecida y a nuevos jugadores.

Más allá de aumentar el volumen de contenido, las futuras expansiones podrán introducir también elementos como la personalización de personajes y desafíos basados en tiempo, los cuales no solo ofrecerán una experiencia de juego más profunda y personalizada, sino que también permitirán a los jugadores obtener recompensas exclusivas de personalización. Además, se pueden incluir logros integrados con plataformas de lanzamiento como Steam, fomentando así la participación y el reconocimiento entre la comunidad de jugadores.

La implementación de estas expansiones es un componente clave para asegurar la sostenibilidad del modelo de negocio a largo plazo. Al proporcionar contenido nuevo y atractivo de manera regular, el juego se mantiene relevante y atractivo para los jugadores, fortaleciendo así su posición en un mercado competitivo y dinámico.

4. Diseño

4.1. Entorno de desarrollo elegido

Para la selección del motor de desarrollo del videojuego, se han considerado diversos aspectos claves que influyen en el proceso de desarrollo. En primer lugar, se priorizó un motor con un enfoque especialmente robusto en entornos 2D, buscando maximizar la eficacia en la creación de juegos con este tipo de estética.

Además, se valoró la importancia de la popularidad y la actividad de la comunidad asociada al motor, con el objetivo de asegurar un respaldo sólido y de recursos disponibles para superar los posibles problemas durante el desarrollo.

La accesibilidad a recursos gratuitos o de bajo coste se convirtió en otro criterio esencial. Esto incluye no solo la disponibilidad de activos y *plugins*, sino también la existencia de guías detalladas y material en formato de video que faciliten el aprendizaje y la resolución de problemas específicos relacionados con el desarrollo del videojuego.

Considerando los tiempos límites para el desarrollo y entrega del proyecto, se priorizó un motor que permitiera un rápido prototipado. Ya que la capacidad de establecer una base sólida en un corto período de tiempo es fundamental para cumplir con los plazos de desarrollo establecidos y garantizar una progresión eficiente en las etapas iniciales del proyecto. Finalmente, se consideró cuidadosamente la cuestión del costo, con el objetivo de minimizar gastos o, idealmente, optar por una solución de desarrollo gratuita.

Estos factores claves se tuvieron en cuenta en primera instancia durante la selección de los motores que se ajustaran a las características deseadas. En consecuencia, se llevó a cabo un estudio de las capacidades ofrecidas por los motores de desarrollo Unity, Godot y GameMaker Studio (Do Unity and GameMaker Studio Do the Same Thing?, s.f.), (García, 2019), (GDB Team, 2023). Tras una minuciosa evaluación de las ventajas y desventajas de estos motores, se organizó la información en una tabla detallada. En esta tabla también se abarcan aspectos secundarios relevantes que influyen en la decisión final sobre qué motor de videojuegos utilizar.

Características	Unity	Godot	GameMaker Studio
Lengua programación	C#	GScript, C#	Propio lenguaje de scripting (GML)
Herramientas visuales / Editor	Herramientas visuales y editor	Herramientas visuales y editor	Herramientas visuales y editor
Curva de aprendizaje	Moderada a alta	Moderada	Baja a moderada
Soporte 2D	Sí	Sí	Sí (Enfocado en 2D, pero admite 3D)
Soporte 3D	Sí	Sí	Sí (Limitado en comparación con Unity)
Escalabilidad	Muy escalable	Escalable	Adecuado para proyectos de tamaño mediano
Comunidad	Grande y activa	En crecimiento	Activa, especialmente para juegos 2D
Asset Store/MarketPlace	Sí (Asset Store)	No (Depende más de recursos externos)	Sí (Marketplace)
Desarrollo multiplataforma	Sí	Sí	Sí
Exportación a otras plataformas	Amplia gama, incluyendo móviles y consolas	Amplia gama, incluyendo móviles y consolas	Amplia gama, incluyendo móviles y consolas ⁶
Tipo de licencia	Propietario, con versión gratuita	MIT (código abierto)	Propietario, con versión gratuita
Código abierto	No	Sí	No
Precio	Gratuito (con opciones de pago)	Gratuito (donaciones aceptadas)	Gratuito (con opciones de pago)

Tabla 10: Comparación entre Unity, Godot y GameMaker Studio.

Tras identificar los aspectos claves para la selección del motor y respaldados por consideraciones secundarias relevantes, la elección definitiva para el desarrollo del videojuego recae en Unity.

Figura 19: Estadística obtenida del portal itch.io, consultado 10/12/2023

⁶ Exportación para consola, requiere licencia de pago y para acceder a las capacidades de exportación para consolas necesitas estar registrado como un desarrollador en la plataforma elegida.

Unity se destaca por su extraordinaria popularidad en el ámbito del desarrollo de videojuegos, respaldada por una amplia y activa comunidad de usuarios. Este punto se considera crucial, ya que, al iniciar el proyecto desde cero, es fundamental contar con el respaldo de una comunidad amplia y muy activa. Esto facilita la obtención de información y soluciones para abordar diversos problemas que puedan surgir durante el desarrollo del videojuego.

Dos aspectos secundarios fundamentales que inclinaron la balanza hacia Unity fueron el lenguaje de programación utilizado y la escalabilidad del motor. Unity emplea C# para el desarrollo de scripts, un lenguaje que, a pesar de no haber sido tratado previamente, comparte similitudes con Java, un lenguaje estudiado durante el desarrollo académico.

En cuanto a la escalabilidad, el proyecto comienza con dimensiones modestas para un uso académico y Unity ofrece una notoria capacidad de escalado. Esto permite que la idea inicial evolucione y crezca de manera significativa en el futuro, brindando flexibilidad y adaptabilidad a medida que se desarrolla el juego.

Considerando el factor coste de la licencia, Godot cuenta como una opción atractiva al ser de código abierto y completamente gratuito. Sin embargo, es importante señalar que tanto Unity como GameMaker Studio ofrecen opciones de licencia personal/educativa que permiten el desarrollo completo del videojuego sin incurrir en costos considerables.

En resumen, la elección de Unity se basa en su popularidad, robusta comunidad de usuarios, empleo de C# como lenguaje de programación, escalabilidad destacada y opciones de licencia que se ajustan a nuestras necesidades de desarrollo.

4.1.1. Requisitos técnicos del entorno de desarrollo

La versión de Unity que se está utilizando es la versión 2022.3 (LTS), y los requisitos técnicos del entorno de desarrollo se han obtenido de la documentación específica para esta versión, que está disponible en su portal web (Unity Technologies, 2023).

Requisitos para el sistema del Editor de Unity

En esta tabla se recogen los requisitos mínimos para ejecutar el Editor de Unity en los distintos sistemas operativos. Aunque se especifica en el mismo portal que el rendimiento y la calidad de representación puede variar dependiendo de la complejidad del proyecto.

Requisitos mínimos	Windows	macOS	Linux
Versión del sistema operativo	Windows 7 (SP1+), Windows 10 y Windows 11, versiones de 64 bits solamente.	Mojave 10.14+ (Editor Intel) Monterey 12.0 (Editor Apple silicon)	Ubuntu 20.04 y Ubuntu 18.04.
CPU	Arquitectura X64 con soporte de conjunto de instrucciones SSE2	Arquitectura X64 con soporte de conjunto de instrucciones SSE2 (procesadores Intel) o Apple M1 o superior (basados en Apple silicon)	Arquitectura X64 con soporte de conjunto de instrucciones SSE2
API de gráficos	GPUs compatibles con DX10, DX11 y DX12.	Metal-capable Intel y AMD GPUs (procesadores Intel) Metal 3.0 o superior (Apple silicon-based processors)	OpenGL 3.2+ o Vulkan-capable, Nvidia y AMD GPUs.
Requisitos adicionales	Controladores oficialmente soportados por el proveedor de hardware.	Controladores oficialmente soportados por el proveedor de hardware (Intel processor) Rosetta 2 es requerido para dispositivos con Apple silicon corriendo en las versiones Apple silicon o Intel del Editor Unity.	Entorno de escritorio Gnome sobre el sistema de ventanas X11, Controlador gráfico Nvidia o AMD Mesa. Otra configuración y entorno de usuario proporcionados con la distribución compatible (Kernel, Compositor, etc.)

Tabla 11: Requisitos técnicos mínimos Unity ver. 2022.3 (LTS)

4.2. Inventario de las herramientas utilizadas

En esta sección se describen de manera breve las distintas herramientas que se han empleado en el desarrollo del videojuego y la documentación de este trabajo.

Entorno de desarrollo integrado

Visual Studio 2022

Como un entorno de desarrollo integrado (IDE) completo, ofrece capacidades para escribir, editar, depurar, compilar y desplegar aplicaciones. Además de estas funcionalidades, Visual Studio incluye una serie de características avanzadas como compiladores integrados, herramientas avanzadas de finalización de código, gestión de control de código fuente y una variedad de extensiones para optimizar cada aspecto del desarrollo de software.

Para el desarrollo del proyecto se ha utilizado con su licencia gratuita de Visual Studio Community y se ha instalado la extensión HLSL para Visual Studio, que ha sido fundamental en la programación y el desarrollo de scripts, ya que proporciona una visualización clara de las clases, variables y métodos, mejorando significativamente la organización y comprensión del código. Además, la función de autocompletado en Visual Studio facilita enormemente el proceso de desarrollo y evita los errores de sintaxis.

Editor de gráficos rasterizados

Aseprite

Una aplicación de gráficos rasterizados enfocada en el arte de píxeles, muy popular entre los desarrolladores de videojuegos, es por lo que se decidió a obtener la licencia de pago, además de a ver sido muy intuitiva para la modificación y creación de los diferentes *assets* que se han utilizado en el desarrollo del videojuego.

Respecto a las funciones que ofrece encontramos herramientas de dibujo en píxeles, paletas de colores personalizables, capas, y opciones de animación que facilitan la creación de *sprites* y gráficos en movimiento.

Editor de audio

CapCut

Es una aplicación versátil de edición de audio y video, diseñada principalmente para la edición de video destinado a las redes sociales. Esta aplicación ofrece a los usuarios la capacidad de crear y modificar videos de manera sencilla y eficiente. Ofrece una amplia gama

de herramientas y características que incluyen recorte, corte, fusión de clips, inserción de efectos visuales y de sonido, integración de música, superposición de texto y creación de transiciones fluidas entre segmentos de video.

En un contexto del desarrollo de este proyecto se ha utilizado principalmente para ajustar la duración de los efectos de sonido, realizar transiciones suaves reduciendo el volumen inicial en las pistas de audio ambiente, y agrupar pistas de sonido similares para crear un efecto de sonido distinto al original.

Editor de texto

Word

Microsoft Word, integrado en la suite Microsoft Office, es una aplicación de procesamiento de texto ampliamente reconocida en contextos académicos y profesionales. En el marco de la licencia proporcionada por la Universidad Oberta de Catalunya (UOC), se ha tenido acceso gratuito a esta herramienta a lo largo del desarrollo del Trabajo de Fin de Grado.

Esta herramienta ha desempeñado un papel fundamental en la creación de toda la documentación del proyecto. Gracias a su uso, se ha editado y maquetado todo el contenido, lo que incluye la redacción de texto, la inserción de imágenes, tablas y gráficos, así como la gestión de citas y referencias bibliográficas.

Planificación

GanttProject

Es una aplicación de software de código abierto diseñada para la gestión de proyectos y la creación de diagramas de Gantt. Los diagramas de Gantt son herramientas visuales que permiten planificar y programar tareas en un proyecto a lo largo del tiempo. Con GanttProject, los usuarios pueden crear y mantener un calendario de proyectos, asignar recursos, establecer dependencias entre tareas y realizar un seguimiento del progreso del proyecto. Además, al ser de código abierto, es gratuito y ofrece una amplia comunidad de usuarios y desarrolladores que contribuyen a su mejora constante.

Fundamentalmente esta aplicación se ha utilizado para diseñar toda la planificación con sus respectivas fechas de entrega de cada PEC, ha permitido visualizar de una manera clara el tiempo que duraba cada sprint y cuando se debe realizar cada entrega.

Mapa Conceptual

[Bubbl](#)

Es una aplicación en línea que se utiliza para crear mapas mentales y diagramas de forma sencilla y colaborativa. Los mapas mentales son herramientas visuales que ayudan a organizar ideas, conceptos y relaciones entre ellos de manera gráfica. Con Bubbl.us, los usuarios pueden crear, editar y compartir mapas mentales de manera intuitiva. Es una herramienta útil para la planificación, la toma de decisiones, la resolución de problemas y la visualización de información en diversos contextos, como la educación, los negocios y la creatividad.

Esta aplicación se ha usada de manera totalmente gratuita y principalmente en la fase inicial del proyecto para organizar las ideas que tenían que ser implementadas en el desarrollo del videojuego y los pasos a seguir.

Diagrama de clases y flujo

[LucidChart](#)

Es una herramienta en línea que se utiliza para crear diagramas y visualizaciones de manera colaborativa. Ofrece una amplia variedad de plantillas y herramientas que permiten a los usuarios crear diagramas de flujo, organigramas, mapas conceptuales, diagramas de red, diagramas de proceso y muchos otros tipos de diagramas.

En su versión gratuita, Lucidchart tiene algunas limitaciones, como el almacenamiento de hasta máximo tres diagramas y un límite de 60 elementos por diagrama. A pesar de estas restricciones, esta versión gratuita ha resultado suficiente para definir y elaborar los diferentes tipos de diagramas disponibles en este proyecto. Además, esta aplicación se distingue por su interfaz intuitiva y su capacidad para crear diagramas de forma rápida y eficiente.

4.3. Inventario de recursos utilizados

Durante el proceso de desarrollo del videojuego, se ha utilizado una amplia variedad de recursos, todos ellos adquiridos en el portal itch.io. En general, todos los *assets*, ya sean visuales, de sonido o efectos de sonido, han sido obtenidos de forma gratuita. En lo que respecta al aspecto visual del proyecto, se ha optado por utilizar *assets* de 32x32 píxeles como base, tanto para los elementos existentes como para la creación de nuevos *sprites*.

A continuación, se presentan los recursos gráficos que se han utilizado en el desarrollo del videojuego y no han sido modificados:

Escena menú principal / Selección de personaje y nivel

En esta escena se incluyen los siguientes elementos que han sido diseñado con el *asset*: <https://bit-life.itch.io/free-parallax-clouds-background>.

- Background
- Nubes (efecto Parallax)

Además, se ha aprovechado el diseño de las nubes para utilizarlo como icono del ejecutable del videojuego.

Para los botones de jugar, opciones y salir del Juego se ha utilizado el siguiente *asset*: <https://bdragon1727.itch.io/border-and-panels-menu-part-2>.

Escena Introductoria

En se ha utilizado en su totalidad el siguiente *asset*, tanto para el diseño del suelo, paredes y elementos decorativos: <https://livingtheindie.itch.io/pixel-house-interior>.



Figura 20: *Tileset* interior casa Escena 2

Además de este paquete recurso se han obtenido los personajes principales del juego:



Figura 21: Izq. Clio, Der. Morfeo

Escena nivel 1 – Laboratorio

El paquete de recursos principal para diseñar este nivel se compone de animaciones para elementos que se presentan en la escena, *tileset* del suelo, pared y elementos decorativos.

<https://marceles.itch.io/land-of-pixels-laboratory-tileset-pixel-art>

Suelo y pared

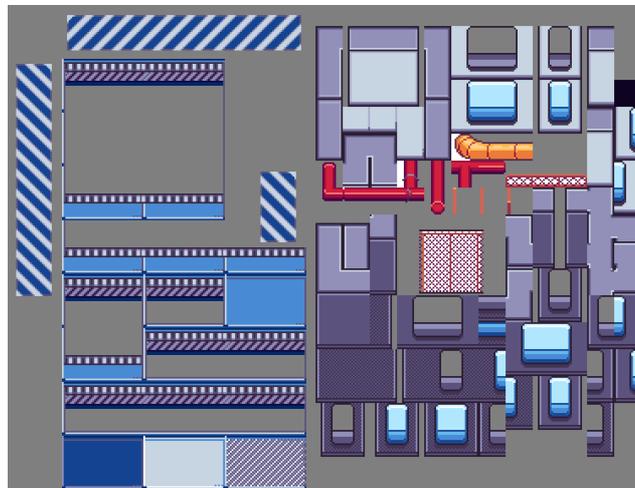


Figura 22: *Tileset* suelo y paredes Escena 3

Elementos decorativos



Figura 23: *Tileset* elementos decorativos Escena 3

Además de este recurso se ha utilizado un personaje para crear el NPC de la cinemática final.



Figura 24: NPC cinemática final

Sistema de inventario

Para la implementación de las casillas del inventario, se ha empleado el siguiente paquete de recursos que incluye una amplia variedad de objetos para definir interfaces:

<https://crusenho.itch.io/complete-gui-essential-pack>

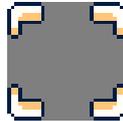


Figura 25: Casilla inventario

Cursor ratón

En el diseño del juego el puntero del ratón cambia de imagen al pasar por encima de un objeto con el que puede interactuar el jugador, para ello se ha utilizado:

<https://megabyte-games.itch.io/mouse-cursor-pack>



Figura 26: Puntero del ratón

Assets modificados

Botones juego Simon y bobina de Tesla

En la imagen que se presenta a continuación, los botones de color verde y rojo provienen del *tileset* original de elementos decorativos del laboratorio. Sin embargo, se ha duplicado estos botones y creado versiones adicionales con otros dos colores para que se ajusten mejor al estilo del juego Simon.



Figura 27: Botones y bobinas Teslas modificadas

En cuanto a la bobina de Tesla, la versión original es de color azul. En este caso, se ha aplicado el mismo proceso que para los botones. Además, se ha ajustado los colores de su animación correspondiente para que coincidan con la animación específica de cada bobina de Tesla en el juego.

Panel de código

La modificación realizada en el *sprite* original consiste en la eliminación de los dos tonos de color en la pantalla, con el propósito de mejorar la legibilidad del código. También se efectuó un rediseño de la botonera, asegurando que la miniatura del objeto que aparece en la escena sea coherente en estilo con los botones del panel de código maximizado.

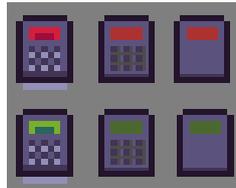


Figura 28: Panel de código

Pizarras

La modificación de la pizarra original consiste en que el jugador pueda visualizar inicialmente el tipo de serie lógica que tendrá que responder.

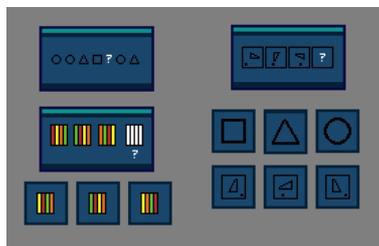


Figura 29: Pizarras – Reto series lógicas

Paneles de objetos

La principal modificación consiste en que se ha añadido la imagen del tipo de matraz que se deberá depositar en el mechero de laboratorio en la escena.

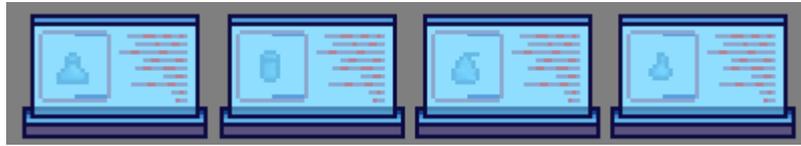


Figura 30: Paneles matraces

Suelo del reto laberinto

De los tipos de suelo que se encuentra en el *tileset* del laboratorio, se ha añadido una cuadrícula, que permite poder realizar el laberinto teniendo en cuenta la pista que puede encontrar el jugador.

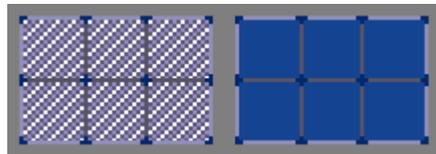


Figura 31: Suelo laboratorio modificado

Assets creados partir de una imagen

Para crear los siguientes assets, se han empleado imágenes disponibles en Pixabay bajo licencia de uso gratuito. Estas imágenes se transformaron en imágenes de píxeles utilizando la funcionalidad de la aplicación Aseprite, que permite la conversión de imágenes vectorizadas en representaciones pixeladas.

- Portapapeles: <https://pixabay.com/es/vectors/icono-portapapeles>
- Batería: <https://pixabay.com/es/vectors/bater>



Figura 32: Portapapeles y batería

Robot

En este caso, se ha utilizado una imagen del portal [Freepik](https://www.freepik.com/free-vector/futuristic-robot-constructor) con licencia de uso gratuito. Para lograr el resultado deseado, se ha aplicado la misma funcionalidad de Aseprite mencionada anteriormente. Sin embargo, a diferencia de la imagen original, se ha tenido que ensamblar el robot y crear un conjunto de sprites para llevar a cabo la animación de su movimiento.

- Robot: <https://www.freepik.com/free-vector/futuristic-robot-constructor>

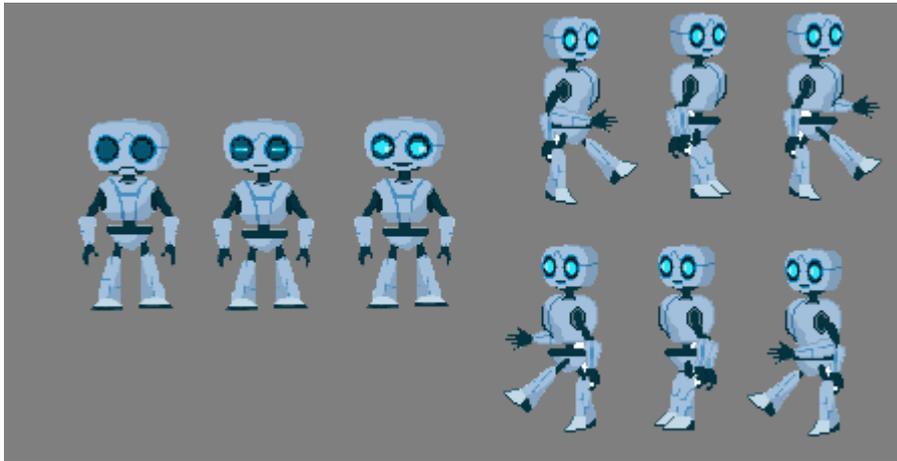


Figura 33: Sprites robot

Assets diseñados desde cero

En la siguiente imagen se muestran los assets creados en su totalidad para los elementos narrativos que se encuentran en la escena.

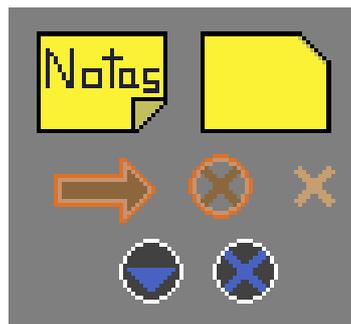


Figura 34: Sprites diseñados propios

Sonido ambiental y efectos de sonido

Todas las pistas musicales y efectos de sonido utilizados en el juego se han obtenido de forma gratuita y bajo licencia de uso gratuito.

Menú principal

- Pista musical: On the Shore of Secrets
- Enlace: <https://assetstore.unity.com/packages/audio/music/fantasy-ambience-free>
- Licencia: [Standard Unity Asset Store EULA](#)

Sonido ambiental – Introducción (Escena 2)

- Pista musical: Garden
- Enlace: <https://www.chosic.com/download-audio/53611/>
- Licencia: [Creative Commons CC BY 3.0](#)
- Créditos
 - Garden by Sakura Girl | https://soundcloud.com/sakuragirl_official
 - Music promoted by <https://www.chosic.com/free-music/all/>

Sonido ambiental – Principal laboratorio (Escena 3)

- Pista musical: Foreboding
 - Enlace: <https://www.chosic.com/download-audio/57909/>
 - Licencia: [Creative Commons CC BY 4.0](#)
 - Créditos
 - Foreboding by PeriTune | <https://peritune.com/>
 - Music promoted by <https://www.chosic.com/free-music/all/>
-

Sonido ambiental – Reto pizarras (Escena 3)

- Pista musical: Sneaky Adventure
- Enlace: <https://www.chosic.com/download-audio/45415/>
- Licencia: [Creative Commons CC BY 3.0](https://creativecommons.org/licenses/by/3.0/)
- Créditos
 - Sneaky Adventure by Kevin MacLeod | <https://incompetech.com/>
 - Music promoted by <https://www.chosic.com/free-music/all/>

Sonido ambiental – Reto laberinto (Escena 3)

- Pista musical: Sneaky Snitch
- Enlace: <https://www.chosic.com/download-audio/39325/>
- Licencia: [Creative Commons CC BY 3.0](https://creativecommons.org/licenses/by/3.0/)
- Créditos
 - Sneaky Snitch by Kevin MacLeod | <https://incompetech.com/>
 - Music promoted by <https://www.chosic.com/free-music/all/>

Sonido ambiental – Puzle Light Out (Escena 3)

- Pista musical: Foam Rubber
 - Enlace: <https://www.chosic.com/download-audio/28657/>
 - Licencia: [Creative Commons CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)
 - Créditos
 - Foam Rubber by Alexander Nakarada | <https://creatorchords.com>
 - Music promoted by <https://www.chosic.com/free-music/all/>
-

Sonido ambiental – Puzle Sokoban (Escena 3)

- Pista musical: Hidden Agenda
- Enlace: <https://www.chosic.com/download-audio/45412/>
- Licencia: [Creative Commons CC BY 3.0](https://creativecommons.org/licenses/by/3.0/)
- Créditos
 - Hidden Agenda by Kevin MacLeod | <https://incompetech.com/>
 - Music promoted by <https://www.chosic.com/free-music/all/>

Efectos de sonidos

Recurso	Enlace
ActivateLightOut	https://www.zapsplat.com/music/light-switch-or-mains-power-socket-switch-flick-on-or-off-1/
ActivateRobot	https://www.zapsplat.com/music/cyberpunk-electronic-equipment-beep-power-1/
ActivateTesla	https://www.zapsplat.com/music/electric-buzz-tone-001/
ActiveCodePanel	https://www.zapsplat.com/music/bicycle-gear-lever-click-change-through-several-gears-fast-1/
ActivateNote	https://www.zapsplat.com/music/adhesive-sticky-tape-rip-2/
Alien	https://www.zapsplat.com/music/alien-chatter-over-radio-1/
AnswerBoard	https://www.zapsplat.com/music/marker-pen-draw-squares-on-whiteboard/
ButtonLightOut	https://www.zapsplat.com/music/light-switch-or-mains-power-socket-switch-flick-on-or-off-1/
ButtonPanel	https://www.zapsplat.com/music/modern-plastic-telephone-single-press-on-the-hang-up-button/
ClickMenu	https://www.zapsplat.com/music/apple-computer-mouse-click-1-2/
CorrectCode	https://www.zapsplat.com/music/3-tone-high-pitched-short-digital-beep/
CountDown	https://www.zapsplat.com/music/game-sound-short-and-fast-high-pitched-beep/
DisableCapsule	https://www.zapsplat.com/music/cartoon-glass-tap-hit-1/
eatingSandwich	https://www.zapsplat.com/music/person-eats-bite-into-a-crispy-crunchy-bread-roll-sandwich-1/
FlaskInLighter	https://www.zapsplat.com/music/chemical-reaction-dry-ice-dissolve-in-jelly-and-water-bubbling-version-18/
GetFlask	https://www.zapsplat.com/music/glass-bottle-in-paper-bag-pick-up-1/
GetObject	https://www.zapsplat.com/music/bed-mattress-with-sheets-and-covers-on-impact-object-or-person-land-on-bed-1/
Incorrect	https://www.zapsplat.com/music/gameshow-quiz-incorrect-buzzer-buzz-for-wrong-answer-2/
lift	https://www.zapsplat.com/music/bell-ring-small-could-be-reception-elevator-or-bicycle-bell-etc-1/ https://www.zapsplat.com/music/elevator-hotel-doors-open-close-lift-operates-then-stops-with-floor-tone/
OpenDoor	https://www.zapsplat.com/music/old-metal-screen-door-close-7/
TakeClipboard	https://www.zapsplat.com/music/card-document-wallet-paper-sheet-take-out-remove-2/
TurnPage	https://www.zapsplat.com/music/a4-paper-notepad-thin-page-turn-2/
SimonDo	https://pixabay.com/es/sound-effects/do-80236/
SimonMi	https://pixabay.com/es/sound-effects/mi-80239/
SimonSi	https://pixabay.com/es/sound-effects/si-80238/
SimonSol	https://pixabay.com/es/sound-effects/sol-101774/

Tabla 12: Efectos de sonidos del juego

Los efectos de sonidos se han obtenido del portal [ZapSplat](https://zapsplat.com/) y [Pixabay](https://pixabay.com/), son de uso gratuito bajo [Licencia Standard](https://creativecommons.org/licenses/by/3.0/) y [Licencia de contenido](https://pixabay.com/es/terminos-y-condiciones/) de Pixabay respectivamente.

4.4. Arquitectura general de la aplicación



Figura 35: Mapa conceptual elementos de la escena

En la fase inicial de este proyecto, y ante la falta de experiencia previa en la creación de videojuegos, se procedió a elaborar un mapa conceptual que delineaba los componentes fundamentales de un videojuego. Este proceso de planificación se estructuró de manera que permitiera el desarrollo individualizado de cada elemento, garantizando su integración cohesiva en la escena principal, que encapsula la mayor parte de la jugabilidad del juego.

Este mapa conceptual, establecido como la estructura general de los componentes de cada nivel, ha sido la guía esencial a lo largo del desarrollo del videojuego, definiendo los elementos clave que el jugador encontrará en el juego:

- **Jugador:** permite moverse libremente por la escena e interactuar con otros elementos, incluyendo puzles, elementos narrativos y objetos del inventario.
- **Salas:** cada nivel consta de varias salas, que contendrán un elemento narrativo principal, y opcionalmente un elemento narrativo secundario, que ayuda al jugador a resolver desafíos o acceder a nuevas salas. Además, cada nivel también incluye un puzle o reto que el jugador debe resolver para avanzar.

- Sistema de Inventario: almacena diversos objetos encontrados en las salas, que pueden ayudar al jugador, activar o resolver desafíos.
- Sonido Ambiental: mejora la inmersión durante la narrativa o la resolución de puzles.
- Efectos de Sonido: añaden realismo al interactuar con los elementos del juego.
- Elementos Animados: aportan dinamismo a los elementos decorativos y puzles.
- Cinemáticas: definidas principalmente para introducir y concluir cada nivel.

4.5. Arquitectura de clases del sistema de juego

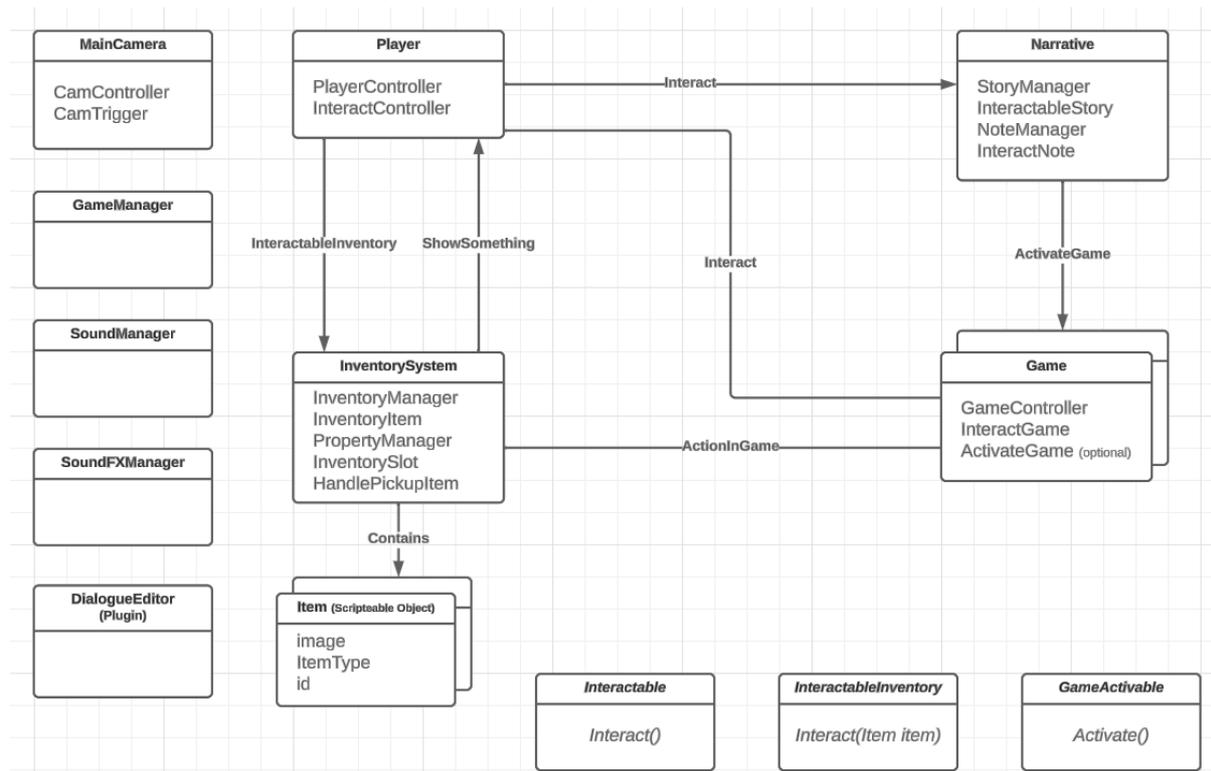


Figura 36: Diagrama de clases del sistema de juego

En el siguiente diagrama se encuentran cuatro componentes principales que han sido definidos con el siguiente cometido:

Player

Es el componente jugador el cual tiene asociado el script *PlayerController*, donde se han definido las funcionalidades para que el personaje se desplace en ocho direcciones en la escena. Por otro lado, tiene asociado el script *InteractController*, el cual desempeña un papel fundamental en la gestión de las interacciones que el jugador puede llevar a cabo en el juego, abarcando acciones tales como la manipulación de objetos en el inventario, la interacción con componentes de los puzles y la activación de los elementos narrativos del juego.

InventorySystem

El sistema de inventario constituye el componente encargado de gestionar el almacenamiento de los diversos objetos recolectados por el jugador en la escena del juego. Para llevar a cabo esta tarea, se emplea un objeto *scriptable* denominado *Item*, el cual posee una serie de parámetros predefinidos y se encuentra presente en la escena del juego. Cuando el jugador recoge un objeto, este último es eliminado de la escena y posteriormente

agregado al inventario en calidad de *InventoryItem*. Esta operación es llevada a cabo mediante la ejecución de la función proporcionada por el script denominado *InventoryManager*.

Adicionalmente, dado que se han establecido diversos tipos de objetos, cada uno de los cuales dispone de sus propios scripts y parámetros específicos, estos elementos son referenciados y gestionados mediante el uso del script *PropertyManager*.

Por último, el sistema diseñado para el inventario también incorpora la funcionalidad de organizar los objetos contenidos en él mediante la acción de arrastrar y soltar entre las casillas disponibles. Esta funcionalidad está definida en el script denominado *InventorySlot* y permite además resaltar la casilla en la que se encuentra seleccionado un objeto mediante un cambio de color distintivo.

Narrative

Dentro de este componente principal, se han establecido dos tipos de elementos claves. En primer lugar, se encuentra el elemento destinado a contener la narrativa correspondiente al nivel actual del juego. Estos elementos están distribuidos a lo largo de las distintas salas del nivel y presentan tres componentes: un identificador que permite mostrar la parte de la historia correspondiente al lugar en el que se encuentra el jugador, la posibilidad de mostrar instrucciones de manera opcional si en la sala existe un puzle, por último, un parámetro que determina si el juego que se encuentra en la sala debe activarse después de que el jugador haya leído la trama correspondiente. Cada una de estas funciones es controlada a través del script *StoryManager*.

Por otro lado, se define un elemento narrativo adicional diseñado para proporcionar asistencia al jugador. Este componente narrativo de ayuda es gestionado por el script *NoteManager*. Estos elementos pueden ser recolectados y utilizados en cualquier momento durante el juego, ofreciendo al jugador orientación para superar desafíos específicos o para acceder a otras salas dentro del juego.

Game

Dada la diversidad de tipos de puzles definidos en el nivel, en el diagrama se representa como una clase abstracta que engloba a estos. Cada uno de los componentes asociados a esta clase abstracta se encuentra representado y configurado de manera específica en los diferentes juegos del nivel.

El *GameManager* desempeña un papel fundamental en la coordinación de las diversas acciones que pueden llevar a cabo los elementos que conforman el juego. Por ejemplo, en el juego *Light Out*, el *GameManager* se encarga de gestionar el encendido y apagado de las luces correspondientes al pulsar un botón específico. En el caso del juego *Simon*, registra y almacena los valores de los botones pulsados por el jugador en una secuencia o en el juego *Sokoban* controla el comportamiento de las cajas al chocar con otras. Además de estas funciones, el *GameManager* se encarga de verificar si se ha superado el juego y, en caso afirmativo, de proceder a la apertura de la siguiente sala, entre otras tareas.

En el contexto del script *InteractGame*, este se asocia a los diversos elementos con los que el jugador interactúa durante el juego. Este script transmite los parámetros necesarios al *GameController* para garantizar una interacción fluida y coherente.

Finalmente, aunque es una característica opcional, por lo general, los puzles no serán interactivo hasta que el jugador haya leído la trama narrativa correspondiente. Para gestionar esta funcionalidad, se ha definido el script *ActivateGame*.

Por otro lado, en el diagrama de clases del juego, también se destacan elementos cruciales para el desarrollo y funcionamiento del juego:

Interacción Jugador – Componentes

La interacción entre el jugador y los componentes del juego se rige mediante la definición de tres clases abstractas distintas:

Interactable: esta clase posee el método *Interact()*, el cual es implementado en los elementos que contienen un script con comportamiento basado en esta clase. En el script *InteractController* del jugador, cada vez que se pulsa el botón derecho del ratón sobre un objeto, se verifica si dicho objeto se encuentra dentro del rango de acción predefinido para interactuar con él. Si se cumple esta condición, se invoca el método *Interact()* del objeto, el cual ejecuta la acción específica definida para ese elemento en particular. Por ejemplo, puede consistir en activar un componente visual en la escena, como un portapapeles que contiene la

narrativa, o realizar una acción que involucra la transferencia de información desde un elemento de un puzle hacia su *GameController*.

InteractableInventory: ha sido diseñada con el propósito de evitar posibles conflictos de interacción entre los objetos presentes en la escena y los objetos almacenados en el inventario del jugador. Para lograr esta separación de funciones de manera eficiente, se ha asignado exclusivamente la tecla "E" del teclado para interactuar con los elementos dentro del inventario del jugador.

La pulsación de la tecla "E" es capturada y gestionada a través del script *InteractController* del jugador. De esta manera, cada vez que el jugador presiona esta tecla mientras tiene un objeto seleccionado en su inventario, se activará la acción específica previamente definida para ese objeto en particular. Estas acciones están asociadas a un script que implementa el comportamiento correspondiente a esta clase abstracta. Por ejemplo, esta interacción podría implicar la visualización del contenido de una nota o la colocación de un objeto en una posición específica dentro de la escena.

GameActivable: se ha implementado este script con la intención de asegurar que, antes de que un puzle pueda ser interactivo, el jugador debe interactuar previamente con un objeto narrativo específico. Una vez que el jugador completa esta interacción narrativa, se activa automáticamente el juego.

MainCamera

La cámara se encuentra configurada en una posición estática que abarca la totalidad del tamaño de una sala en el nivel. En relación con los scripts asociados, *CamTrigger* cumple la función de detectar cuando un jugador ingresa a la sala contigua y, posteriormente, actualiza la posición objetivo de la cámara en el segundo script, *CamController*. Este último guía la transición de manera fluida hacia la nueva área, brindando una experiencia de movimiento de cámara característica de juegos como *The Binding of Isaac* o la saga *The Legend of Zelda* en de las consolas NES o GameBoy.

GameManager

Este componente se encarga de garantizar la existencia de una única instancia a lo largo de las distintas escenas. Su principal objetivo es mantener una lista pública de personajes disponibles para selección, permitiendo que un personaje seleccionado mediante el uso de *PlayerPrefs* pueda ser instanciado y persista sin ser destruido durante la transición entre escenas. Esto asegura la continuidad y la retención de la elección del personaje a lo largo de la experiencia de juego.

SoundManager

Este componente gestiona la transición de la música ambiental entre diferentes escenas y ajusta la música de fondo según la situación del juego, como cuando se inicia un puzzle, cambiando la música ambiental para que se adapte a la atmósfera del juego. Además, brinda la capacidad de controlar el volumen del sonido según las preferencias del jugador y ajustarlo de acuerdo con las necesidades del juego, ya sea disminuyendo el volumen para resolver un rompecabezas o restaurándolo a su valor por defecto.

SoundFXManager

Este componente se decide implementarlo como controlador independiente en lugar de integrarlo con el controlador de música ambiental debido a la gran cantidad de efectos de sonido que se manejarán. Desde el inicio del juego, este controlador realiza una carga asincrónica de los recursos de efectos de sonido ubicados en una carpeta específica del proyecto, lo que permite una forma eficiente y rápida de agregar nuevos sonidos al juego. Además, este controlador ofrece al jugador la flexibilidad de ajustar el volumen de los efectos de sonido a su preferencia.

DialogueEditor

DialogueEditor, desarrollado por [Grasshop Dev](#), es un plugin gratuito disponible en la [Unity Asset Store](#), diseñado para facilitar la creación de diálogos en juegos a través de un sistema de nodos. Esta herramienta se destaca por su capacidad para configurar nodos de dos tipos principales: *speech*, utilizado para presentar diálogos, y *option*, que ofrece al jugador la posibilidad de elegir entre diferentes opciones de respuesta. Cada nodo puede personalizarse con características adicionales como sonido, icono y nombre del personaje que habla y tipo de fuente, ofreciendo una gran flexibilidad en la construcción de diálogos.

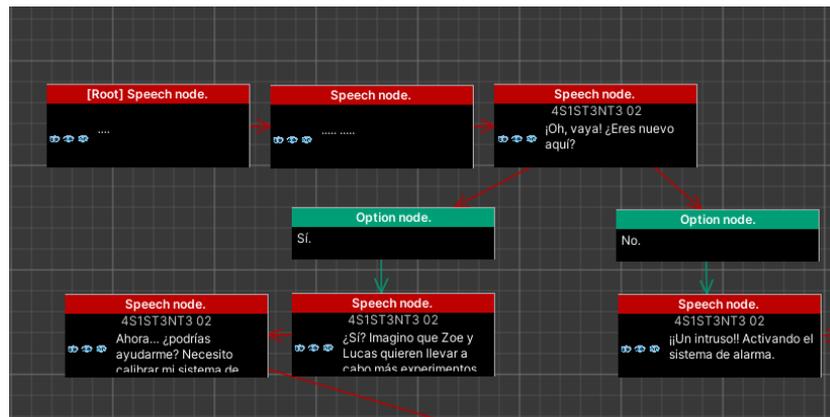


Figura 37: Flujo de conversación con DialogueEditor

Por otro lado, aunque *DialogueEditor* ofrece una función de transición automática de diálogos, basada en un tiempo de espera predefinido, se ha tomado la decisión de desactivar esta opción. En su lugar, se ha preferido que sea el jugador quien controle el ritmo de la conversación, avanzando a través de los cuadros de diálogo a su propio ritmo.



Figura 38: Cambio realizado a la interfaz de DialogueEditor

Con el objetivo de simplificar la interacción manual y enriquecer la experiencia del usuario, se ha realizado ajustes en la interfaz del plugin. Se han añadido dos botones adicionales al script que permiten avanzar en la conversación de manera más sencilla y directa, lo que, además de mejorar la accesibilidad y usabilidad del sistema de diálogo, soluciona un problema estético. Ya que los botones predeterminados del plugin compartían características de tamaño y fuente con los botones de opciones, lo que generaba una discrepancia visual. Con esta modificación, se ha logrado que el diseño sea más coherente y compacto, mejorando la apariencia general de la interfaz.

En resumen, estos componentes y scripts definidos en el diagrama de clase se han diseñados para crear una base sólida y compacta que permita reutilizar el código desarrollado y facilitar la creación de futuros niveles. Esto permitirá centrarse en la lógica de los diversos rompecabezas y retos que se puedan implementar en los nuevos niveles y la narrativa, agilizando así el proceso de desarrollo y garantizando una experiencia de juego coherente.

4.6. Diagramas de navegación

Escenas

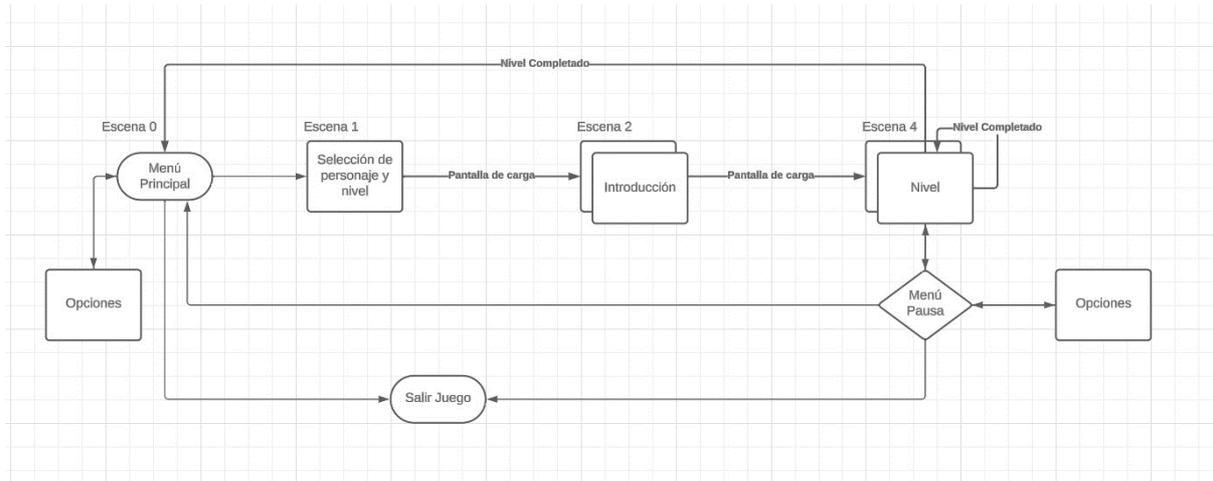


Figura 39: Arquitectura general escenas

El flujo del juego ha sido diseñado de forma lineal, comenzando en el menú principal, donde el jugador se encuentra con tres opciones: jugar, opciones o salir. Al elegir la opción de jugar, el jugador debe seleccionar el nivel que desea completar y el personaje con el que desea jugar. Esto desencadenará la carga de una escena introductoria vinculada a la escena que contiene el nivel seleccionado.

En lo que respecta a la escena introductoria del nivel seleccionado, se ha tomado la decisión de separarla del nivel mismo con el propósito de ofrecer un contexto independiente a la narrativa introductoria. Esto se fundamenta en la concepción principal de la narrativa, que plantea que el jugador permanece de manera constante en un estado de ensueño. En consecuencia, la escena introductoria transcurre en un entorno diferenciado, a pesar de encontrarse igualmente inmerso en un contexto onírico, distinto al propio entorno del sueño correspondiente al nivel seleccionado.

En relación con la estructura del juego, no se ha incorporado la opción de guardar o cargar partidas. Al acceder a la pantalla de selección de niveles, el jugador se encontrará con todos los niveles bloqueados, con la excepción del primer nivel. Para desbloquear los siguientes niveles, el jugador deberá completar satisfactoriamente el nivel actual. Una vez que haya logrado este objetivo, tendrá la opción de avanzar al siguiente nivel o regresar al menú principal, y en la selección de niveles también encontrará el siguiente nivel desbloqueado y listo para jugar.

Esta decisión se ha tomado con la consideración de que cada nivel debe mantener una duración relativamente breve, en línea con las pautas de tiempo de juego establecidas en las necesidades del público objetivo definido en el apartado 2.1. De esta manera, se busca garantizar que la experiencia del jugador no se vea afectada negativamente ya que la duración de cada nivel es corta y su narrativa y ambientación es única en cada nivel.

En lo que respecta al diagrama de flujo del juego, también se encuentran los siguientes elementos:

Pantalla de carga: esta pantalla cumple una doble función, carga asíncrona de recursos para la siguiente escena, lo que permite tiempos de carga reducidos y mantiene la fluidez del juego, además de proporcionar una transición musical gradual para mantener o cambiar la ambientación de manera suave y coherente con la narrativa del juego. La duración de estas pantallas es breve para evitar interrupciones prolongadas en la experiencia del jugador.

Opciones: tanto desde el menú principal como desde el menú de pausa, el jugador puede ajustar la configuración del juego, incluyendo el volumen de sonido, el volumen de efectos de sonido y la velocidad de los diálogos.

Salir del juego: esta opción está disponible tanto en el menú principal como en el menú de pausa y permite al jugador salir del juego y regresar al sistema operativo.

Diagrama del flujo de la jugabilidad

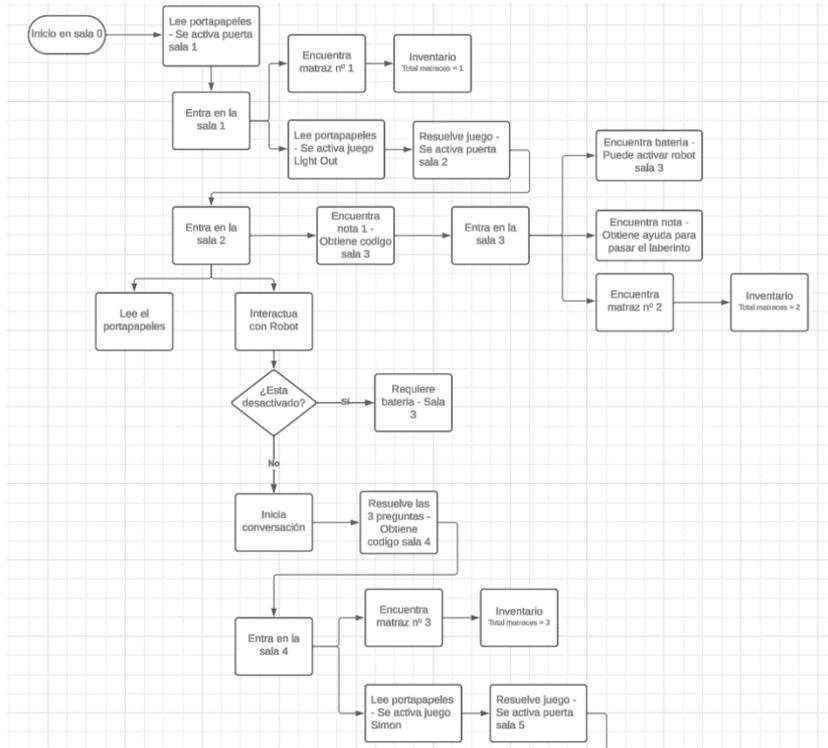


Figura 40: Diagrama de flujo Sala 0-4

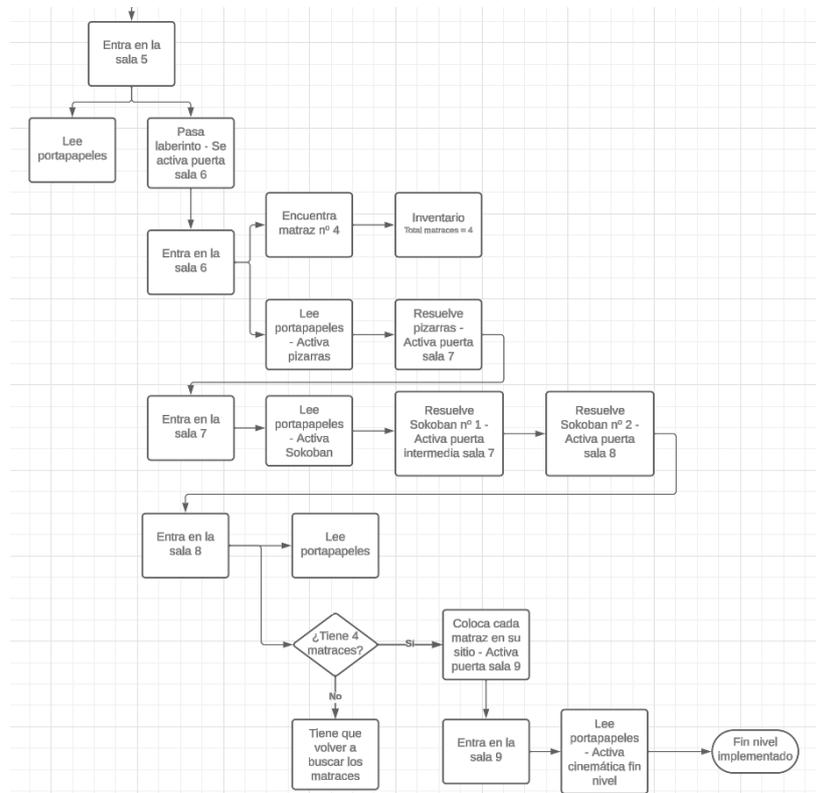


Figura 41: Diagrama de flujo Sala 5-9

Dado que el juego está orientado a una audiencia de 6 a 10 años, se ha diseñado con un enfoque de jugabilidad lineal para asegurar una experiencia gratificante para el jugador. Y son los puzzles y desafíos los que se han creado específicamente para motivar al jugador a pensar y resolver los retos que se les presentan.

Como se muestra en el diagrama de flujo, la acción se desarrolla al ingresar a una sala. En este punto, resulta esencial que el jugador lea primero la narrativa y las instrucciones de cada prueba antes de intentar resolver un desafío. Una vez que el desafío se resuelve con éxito, se le otorga acceso a la siguiente sala, con este enfoque se garantiza que el flujo del juego progrese de manera unidireccional.

Sin embargo, para evitar simplificar demasiado el flujo del juego, se alienta al jugador a explorar minuciosamente cada sala, ya que pueden encontrarse objetos que permitan el acceso a otras salas. Por ejemplo, la nota encontrada en la sala 2 es necesaria para obtener la batería en la sala 3 y activar el robot para resolver sus preguntas y acceder a la sala 4.

Por otro lado, es importante destacar que el único punto en el que el jugador se dará cuenta de la importancia de una exploración detallada es en la recolección de objetos ocultos, que en este nivel son los matraces que se encuentran repartidos por las diferentes salas. Y es que cuando el jugador llegue a la penúltima sala, comprenderá la importancia de la exploración ya que de no tener los cuatro objetos deberá volver a explorar las salas previas, para poder completar ese reto y acceder a la sala final.

4.7. Diseño de niveles

En esta sección, se muestra el mapa completo del primer nivel diseñado, resaltando los objetos narrativos tanto principales como secundarios que ayudarán al jugador, así como los objetos que deben recolectarse para superar diversos desafíos. A continuación, se ofrece una breve descripción de los obstáculos y desafíos que el jugador encontrará en cada sala.



Figura 42: Mapa completo nivel 1

Sala 0

El jugador comienza en esta sala, donde se encuentra la puerta que conduce a la siguiente sala, la cual está cerrada. Para avanzar, el jugador deberá interactuar con el portapapeles y leer la narrativa asociada. Una vez que haya concluido la lectura, la puerta se abrirá automáticamente.

Sala 1

Al entrar en esta sala, el jugador debe interactuar con el portapapeles. Primero, leerá la narrativa relacionada con la sala y luego recibirá las instrucciones para superar el juego *Light Out*. Una vez que el jugador haya cerrado el portapapeles, se activará el juego. La puerta que da acceso a la sala 2 se abrirá después de que el jugador haya apagado todas las luces. Si el jugador explora la sala, también encontrará el primer matraz.

Sala 2

En esta sala, el jugador puede leer el portapapeles, pero para activar al robot, debe explorar la sala y encontrar una nota que contiene el código de acceso necesario para ingresar a la sala 3. Una vez que tenga la batería en su inventario, podrá activar el robot y mantener una conversación. Y cuando haya superado las diferentes preguntas que le realiza el robot, se abrirá la puerta que lleva a la sala 4.

Sala 3

Esta sala debe ser explorada cuidadosamente, ya que el jugador encontrará otra nota que proporciona información para superar el laberinto, indicando los pasos a seguir mediante flechas. Aquí también se encuentra el segundo matraz y, finalmente, la batería necesaria para activar el robot en la sala 2.

Sala 4

El jugador debe leer el portapapeles y luego las instrucciones. Una vez que haya concluido la lectura, comenzará una cuenta regresiva visual y sonará que indica el inicio del juego *Simon*. Después de completar con éxito un total de 6 secuencias, se abrirá la puerta que conduce a la sala 5. Antes de avanzar a la siguiente sala, es importante que el jugador explore esta sala para encontrar el tercer matraz.

Sala 5

Al ingresar a esta sala, la habitación se oscurecerá por completo, excepto por un halo circular alrededor del jugador que proporciona algo de visión. En esta sala, el jugador puede leer el portapapeles y, si tiene la nota con las instrucciones para el laberinto, podrá superarlo con facilidad. Una vez que llegue al final del laberinto, la sala se iluminará por completo, el laberinto desaparecerá y se abrirá la puerta hacia la sala 6.

Sala 6

El jugador debe leer el portapapeles para poder interactuar con la pizarra central. Una vez que comience la interacción, tendrá que responder correctamente a las preguntas que aparecen en las distintas pizarras. En caso de responder incorrectamente, se escuchará un sonido que indica el error y las preguntas se mostrarán nuevamente en cada pizarra. Si el jugador responde correctamente, se abrirá la puerta hacia la sala 7. Antes de salir de la sala, es importante explorarla para encontrar el cuarto matraz.

Sala 7

Para iniciar el juego *Sokoban*, el jugador debe leer el portapapeles y completar el primer desafío. En caso de cometer un error, hay un botón en las consolas de la sala que permite restablecer la posición de las cajas. Una vez que todas las cajas se hayan colocado correctamente, se abrirá la puerta intermedia de la sala. Luego, el jugador puede abordar el segundo *Sokoban* y una vez que se complete, se abrirá la puerta hacia la sala 8.

Sala 8

Si el jugador ha explorado detenidamente las salas anteriores, habrá recolectado los cuatro matraces. Deberá colocarlos en los paneles del mechero de laboratorio correspondiente, siguiendo las indicaciones adecuadas. Una vez que los cuatro matraces estén en su lugar, la puerta hacia la sala final se abrirá. En esta sala, también se encuentra un portapapeles que proporciona parte de la narrativa y las instrucciones para resolver el último desafío.

Sala 9 - Sala final

En esta sala, el jugador solo puede interactuar con el portapapeles, que marca el final de la sala. Al cerrar la historia, el jugador quedará inmóvil, y se mostrará una cinemática en la que un NPC baja por el ascensor y expulsa al personaje del nivel. De esta manera, se concluye el primer nivel, y el jugador aparecerá en la escena de introducción preguntándose si lo que vivió fue real o un sueño.

5. Implementación

5.1. Requisitos de instalación

El videojuego se ha diseñado y compilado específicamente para Windows y no requiere de disponer de un equipo con especificaciones de alta gama para su ejecución. Esta ventaja permite su ejecución fluida en una amplia gama de ordenadores, incluyendo aquellos de especificaciones más básicas. Como resultado, la jugabilidad se mantiene fluida y agradable, incluso en equipos de gama baja, proporcionando una experiencia de juego consistente y accesible para todos los usuarios.

Requisitos mínimos para Windows:

- Sistema Operativo: Windows 7 SP1 (64-bit), Windows 8 (64-bit) o superior.
- Procesador: Intel Core i3 o AMD equivalente.
- Memoria RAM: 2 GB de RAM.
- Gráficos: Tarjeta gráfica con soporte DirectX 10 (como NVIDIA GeForce GTX 650, AMD Radeon HD 7750).
- DirectX: Versión 10.
- Espacio en Disco Duro: 500 MB de espacio disponible.

5.2. Instrucciones de instalación

El videojuego no requiere de ninguna instalación, únicamente descargar el archivo *.zip* proporcionado (ver Anexo B). Tras descomprimir, tendrá acceso a la carpeta *Dreambound_1.0b*, donde se encuentra el ejecutable *Dreambound.exe*. Debido a que esta aplicación no está firmada digitalmente por una entidad reconocida que verifique la autenticidad del videojuego, es probable que la función de seguridad Smart Screen de Windows genere una alerta en forma de ventana emergente, impidiendo la ejecución inmediata del juego. Para proceder a la ejecución del juego, el usuario deberá hacer clic en *Más Información* en la ventana emergente y, a continuación, seleccionar *Ejecutar de todas formas*. Este paso permitirá que el juego se inicie normalmente y evitará futuras interrupciones por parte de Smart Screen en ejecuciones posteriores.

6. Demostración

6.1. Guía de usuario

Controles

- Teclas W, A, S, D: permiten el movimiento del personaje en todas direcciones, facilitando la exploración libre del escenario.
- Tecla E (acción de objetos del inventario):
 - Notas: al presionar la E, se despliega el contenido de la nota en el centro de la pantalla. Se cierra al hacer clic en el icono X de la imagen.
 - Objetos usables: de los objetos que pueden ser recogido por el jugador, existen objetos para activar un reto o para completarlo, es necesario que el jugador se coloque enfrente del elemento que se quiere activar o donde se quiere colocar el objeto del inventario y pulsar la tecla E.
- Clic derecho del ratón: para activar elementos del escenario, como botones, paneles de código, portapapeles o para recoger los objetos.
- Clic izquierdo del Ratón: sirve para interactuar con los botones de avance y cerrar el cuadro de dialogo, botones de configuración, menú principal y de pausa, y para organizar el inventario arrastrando objetos a casillas vacías.

Ejecución del juego

Una vez se ejecuta el juego, el jugador se encuentra en la pantalla principal, donde podrá elegir comenzar a jugar, configurar las opciones o salir del juego. Al pulsar el botón de *Jugar* se carga la pantalla de selección de personaje y de nivel, El jugador inicialmente puede elegir cualquiera de los dos personajes disponibles en el juego, pero solamente podrá acceder al primer nivel, hasta que no se pase el primer nivel el jugador no desbloqueará el siguiente, en la fase actual del juego el segundo nivel son los créditos.

Cinemat ica de introducci3n

En la pantalla de selecci3n de personaje al pulsar el bot3n jugar, se inicia una pantalla de carga y tras unos segundos se inicia una cinemat ica donde el jugador no podr  realizar ninguna acci3n, una vez termine la cinemat ica dar  paso a la siguiente escena donde se carga el escenario del primer nivel.

Flujo resoluci3n de retos

Al iniciar el nivel, el jugador se encuentra en una sala cuya puerta principal hacia la siguiente habitaci3n se encuentra cerrada. Para ello el jugador debe interactuar con el portapapeles de la mesa, una vez lea la narrativa inicial del nivel y cierre la imagen del portapapeles, se activa la apertura de la puerta que da acceso a la siguiente sala.

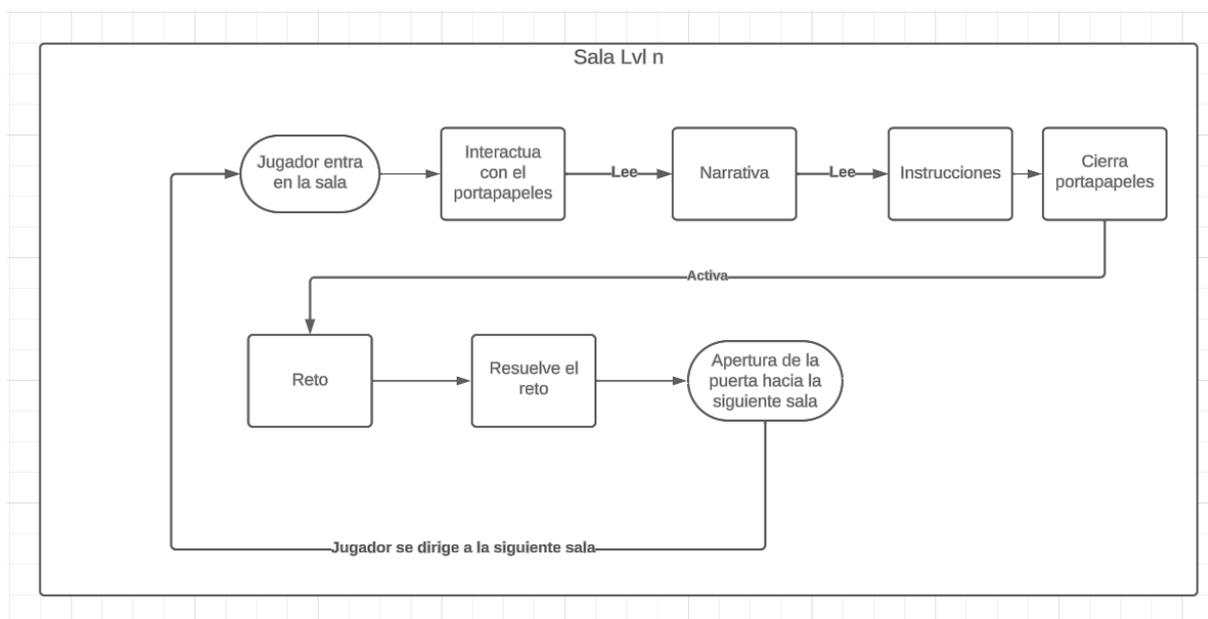


Figura 43: Diagrama de flujo: patr3n resoluci3n de retos

En general las acciones que debe realizar el jugador en las distintas salas, es un patr3n que se repite en las diferentes salas, lectura de la narrativa, opcionalmente puede existir las instrucciones que aportan una ayuda a resolver el reto y una vez se cierra el portapapeles se activa el reto. Cuando el jugador soluciona el puzle se activa la apertura de la puerta que da acceso a la siguiente sala.

Objetos de inventario

El jugador no solo tendrá que realizar el conjunto de acciones que se han mostrado en el diagrama de flujo anterior, también debe explorar cada sala detenidamente ya que debe de obtener los siguientes objetos, necesarios para la ayuda y superación de algunos retos.



Figura 44: Objetos de inventario

Notas: ayudan al jugador a acceder a otras salas, o superando un reto con las pistas que se encuentren en estas.

Objetos usables: batería para activar el robot, que permite activar la conversación con este e iniciar el reto de las preguntas lógicas. Por otro lado, se encuentran los matraces que se deben colocar en los mecheros de laboratorio correspondiente que se encuentra en una de las salas para resolver el reto y poder acceder a la siguiente habitación.

7. Conclusiones y líneas de futuro

7.1. Conclusiones

Lecciones aprendidas

Tras la finalización del proyecto, se han extraído valiosas lecciones que enriquecen el bagaje personal, destacando la importancia de la gestión eficiente del tiempo en las diversas fases y objetivos del proyecto. Esta lección adquiere una relevancia especial con la implementación de la metodología Scrum, cuya planificación y seguimiento meticuloso han sido fundamentales. Su aplicación práctica, enfocada y adaptada a las necesidades específicas del proyecto, superó las expectativas previas, las cuales estaban basadas principalmente en conocimientos teóricos obtenidos durante la formación académica.

En el transcurso del proyecto realizado de forma autónoma, se ha obtenido una perspectiva más amplia y profunda sobre la importancia y complejidad de cada elemento que compone un videojuego. Se ha reconocido que cada uno de los participantes, desde diseñadores y programadores hasta artistas y *testers*, juegan un papel indispensable en el éxito del proyecto. Inicialmente se percibía la programación como el componente más crítico, sin embargo, la experiencia del desarrollo ha revelado que no hay facetas más importantes que otras. Se ha comprendido que mantener un equilibrio entre todos los elementos es esencial para la creación de un producto final de alta calidad.

En resumen, se considera que la experiencia adquirida en el desarrollo de este proyecto representa un activo valioso tanto en el ámbito personal como profesional. Se han identificado fortalezas clave y áreas de mejora, lo que proporciona una base sólida para la búsqueda de asesoramiento y asociación estratégica en proyectos futuros.

Objetivos alcanzados

En general, se ha logrado un alto grado de satisfacción con el proyecto, alcanzando los objetivos personales establecidos. Se ha entendido la importancia de cada elemento y participante en el desarrollo de un videojuego. El motor de desarrollo Unity se utilizó en su forma básica, aprovechando eficazmente sus herramientas sin depender excesivamente de recursos externos. Este enfoque ha permitido alcanzar una competencia considerable para proyectos futuros, enfocándose en la investigación del público objetivo y en la inspiración de otros juegos para mejorar el propio proyecto.

Con respecto a los objetivos del producto, se considera que se ha obtenido un resultado completo y bien ejecutado. Se ha prestado especial atención a la usabilidad, accesibilidad, animación, coherencia en el diseño del escenario y la integración adecuada de los desafíos en sus respectivas áreas. Además, se han incorporado cinemáticas para introducir y concluir los niveles, así como efectos de sonido realistas y pistas de audio variadas para enriquecer diferentes partes de la historia.

Finalmente, en cuanto a los objetivos destinados a la experiencia del usuario, se ha creado un juego apto para todos los públicos, libre de violencia y con un diseño artístico atractivo. Sin embargo, se reconoce que no se ha alcanzado un alto grado de satisfacción en la integración de la narrativa, especialmente en lo que respecta al arte gráfico y al sonido. Se considera que trabajar con recursos gráficos de un único diseñador y música de un solo compositor habría mejorado la inmersión del jugador en la narrativa y el ambiente del juego. La transición suave entre las pistas musicales y la asignación de mayor relevancia a la música en momentos clave son aspectos que se identifican como fundamentales para aumentar la inmersión del jugador.

Planificación y metodología

La metodología Scrum fue la elegida para la gestión y ejecución del proyecto. La adherencia a las fechas de entrega de cada etapa del proyecto fue un aspecto significativo y estuvo bien coordinada con la metodología. A lo largo del desarrollo, surgieron numerosas ideas que podrían haber sido candidatas para su implementación. Sin embargo, se mantuvo una disciplina estricta en este aspecto. A pesar del objetivo de desarrollar un producto completo, se reconocía que no se podía aspirar a un nivel profesional avanzado desde el principio, dado que el proyecto partía de una base inicial sin experiencia previa.

En este contexto, la prioridad fue cumplir con cada hito definido, evitando desviarse en nuevas ideas. Se concentró el esfuerzo en mejorar y optimizar el producto en desarrollo. No obstante, se realizaron tres implementaciones adicionales que no estaban planificadas inicialmente pero que se consideraron esenciales: la selección de personajes, el sistema de inventario y el sistema de desbloqueo de niveles. Estas inclusiones se hicieron para enriquecer el proyecto, manteniendo siempre el enfoque en los objetivos clave del desarrollo.

7.2. Líneas de futuro

En consideración a la estructura del juego, que no vincula una historia concreta entre los diferentes niveles, se percibe un amplio potencial de expansión. Además, conforme al modelo de negocio establecido, este primer producto se visualiza como una herramienta de marketing eficaz para ganar reconocimiento. En cuanto a las futuras mejoras, se plantea un enfoque en el rediseño del producto, manteniendo la idea central y asociándose con un diseñador gráfico y un compositor. Estos dos elementos son fundamentales para atraer y sumergir al público objetivo en el juego. Mejorar los gráficos, respetando el estilo actual, y enriquecer la banda sonora son dos aspectos prioritarios que desarrollar.

En lo que respecta a los objetivos a largo plazo, se contemplan los siguientes:

- Lanzar una versión completa con 10 niveles, que ofrezca una amplia variedad de desafíos y narrativas cortas y encantadoras, con la idea de expandir el juego mediante futuras extensiones que incluyan igual o mayor número de niveles.
- Incluir selección de dificultad, que dificulten en mayor medida la superación de los distintos puzzles o retos para otorgar un mayor grado de desafío al jugador.
- Proporcionar al jugador la posibilidad de crear y personalizar su personaje a su gusto.
- Incorporar logros que otorguen recompensas, como objetos para la personalización del personaje.
- Dar mayor importancia a la exploración en el juego, incluyendo la búsqueda de elementos en cada nivel que desbloqueen cinemáticas o revelen aspectos de la vida del personaje fuera de sus sueños.

En resumen, al igual que ocurrió durante el desarrollo, la generación de nuevas ideas para implementar continúa creciendo. Es importante evaluar estas ideas en términos de tiempo y viabilidad para determinar si son esenciales para el juego y si aportan innovación y frescura, con el fin de atraer a un mayor número de jugadores

Bibliografía

- ¿Cuánto Cobra un Desarrollador de Software? (s.f.). Recuperado el 06 de octubre de 2023, de YeePLY: <https://www.yeePLY.com/blog/precio-hora-programador-cuanto-cobra-desarrollador-de-software/>
- Bonet de Luna, C., & Prochazcova, M. (16 de mayo de 2019). *Niños de primaria: el niño de 6 a 11 años*. Obtenido de Enfamilia: <https://enfamilia.aeped.es/edades-etapas/ninos-primaria>
- Cano, C. V. (2023 de agosto de 25). *Piaget y las cuatro etapas del desarrollo cognitivo*. Recuperado el 22 de octubre de 2023, de Actualidadenpsicología: <https://www.actualidadenpsicologia.com/piaget-cuatro-etapas-desarrollo-cognitivo/>
- Do Unity and GameMaker Studio Do the Same Thing? (s.f.). Recuperado el 12 de diciembre de 2023, de Ilogos: <https://ilogos.biz/unity-vs-gamemaker-what-to-choose-for-game-development/>
- Equipo Qustodio. (27 de agosto de 2020). *Aumenta en un 45% el tiempo que pasan los menores españoles en apps de videojuegos este verano*. Obtenido de Qustodio: <https://www.qustodio.com/es/research/menores-tiempo-en-apps-videojuegos/>
- García, C. (14 de diciembre de 2020). *5 beneficios de los videojuegos en niños y adolescentes*. Obtenido de Codelearn: <https://codelearn.es/blog/5-beneficios-de-los-videojuegos-en-ninos-y-adolescentes/>
- García, D. E. (12 de junio de 2019). *Ventajas y diferencias entre Unity, Unreal Engine y Godot*. Obtenido de Openwebinars: <https://openwebinars.net/blog/ventajas-diferencias-unity-unreal-engine-godot/>
- García-Bullé, S. (25 de enero de 2019). *Videojuegos: Una herramienta educativa en potencia*. Obtenido de Observatorio del Instituto para el Futuro de la Educación: <https://observatorio.tec.mx/edu-news/juegos-y-educacion/>
- GDB Team. (22 de septiembre de 2023). *Godot vs Unity (for making your first game)*. Recuperado el 14 de diciembre de 2023, de Gamedevbeginner: <https://gamedevbeginner.com/godot-vs-unity-for-making-your-first-game/>
- León, J. (28 de diciembre de 2020). *El género de los puzzles no habría llegado hasta donde está sin los indies*. Obtenido de Vidaextra: <https://www.vidaextra.com/otros-generos/genero-puzles-no-habria-llegado-donde-esta-indies>
- Machinarium Review*. (s.f.). Obtenido de Playingclub: <https://playingclub.org/machinarium>
- Miguel, P. D. (26 de junio de 2020). *MYST: EL SURREALISMO HECHO AVENTURA*. Obtenido de Portal 33bit: <https://portal.33bits.net/myst-el-surrealismo-hecho-aventura/>
- Unity Technologies. (28 de abril de 2023). *Unity User Manual 2020.3 (LTS)*. Obtenido de Docs.unity3d: <https://docs.unity3d.com/2020.3/Documentation/Manual/UnityManual.html>
- Wikipedia. (08 de octubre de 2023). *Baba Is You*. Recuperado el 14 de octubre de 2023, de Wikipedia: https://es.wikipedia.org/wiki/Baba_Is_You

Anexos

Anexo A: Glosario

Áreas cognitivas: Regiones específicas del cerebro que se asocian con funciones cognitivas particulares, como la memoria o el procesamiento visual.

Asset: Cualquier elemento o recurso utilizado en el desarrollo de software o juegos, como imágenes, sonidos o código.

Background: El fondo de una escena o imagen, a menudo utilizado en juegos y aplicaciones para proporcionar contexto visual.

C#: Un lenguaje de programación desarrollado por Microsoft, ampliamente utilizado en el desarrollo de aplicaciones y juegos.

Efecto Parallax: Técnica visual que crea la ilusión de profundidad al mover objetos en diferentes velocidades en una escena.

Gráficos prerrenderizados: Gráficos que se crean previamente y se almacenan en lugar de generarse en tiempo real.

Hardware: Los componentes físicos de una computadora o dispositivo electrónico, como la CPU, la memoria, la tarjeta gráfica, etc.

IDE (Entorno de Desarrollo Integrado): Un software que proporciona herramientas para escribir, depurar y desarrollar aplicaciones de software.

Java: Un lenguaje de programación ampliamente utilizado en el desarrollo de aplicaciones y juegos multiplataforma.

Licencia MIT: Una licencia de software de código abierto que permite el uso y la redistribución con pocos requisitos.

Light Out: Un juego en el que el objetivo es apagar todas las luces de un tablero, cambiando el estado de las luces circundantes al hacer clic en ellas.

Lenguaje GML: GameMaker Language, un lenguaje de programación utilizado en la plataforma GameMaker para el desarrollo de videojuegos.

Matraz: Un recipiente de laboratorio utilizado para contener líquidos.

Mockup: Una representación visual de un diseño o concepto, a menudo utilizada en el diseño de interfaces de usuario.

NPC (Personaje No Jugador): Un personaje controlado por la inteligencia artificial en un juego, que no es el jugador principal.

Onírico: Relativo a los sueños o lo surrealista.

Pixel: La unidad más pequeña de una imagen digital, que representa un punto de color.

Pixel Art: Un estilo de arte digital que utiliza píxeles individuales para crear imágenes.

Plugins: Extensiones de software que añaden funcionalidad adicional a una aplicación o programa.

Point and Click: Un estilo de juego de aventuras donde el jugador interactúa con el entorno haciendo clic en objetos.

Programador junior: Un programador en sus primeras etapas de carrera, generalmente con menos experiencia que un programador senior.

Scrum: Un marco de trabajo ágil utilizado en el desarrollo de software para gestionar proyectos y equipos de manera eficiente.

Simon: Un juego en el que se presenta una secuencia de colores y sonidos, y el jugador debe repetir la secuencia correctamente para avanzar en el juego. El juego se vuelve más difícil a medida que se añaden más colores a la secuencia.

Sokoban: Un juego de rompecabezas en el que el jugador mueve cajas para colocarlas en lugares específicos.

Software: Programas de computadora y datos que proporcionan instrucciones a la máquina para realizar tareas específicas.

Tester: Persona encargada de probar y verificar la calidad de un producto de software en busca de errores antes de su lanzamiento.

Tileset: Un conjunto de gráficos utilizados para crear mapas o escenarios en juegos.

Top-Down: Un estilo de juego o vista en el que la acción se presenta desde una perspectiva aérea, donde el jugador ve la escena desde arriba.

Anexo B: Entregables del proyecto

Enlace repositorio Github

<https://github.com/mgomro/Dreambound>

Enlace descarga Dreambound

https://github.com/mgomro/Dreambound/releases/download/Dreambound_v1.0b/Dreambound_V1.0b.zip