



Galleto's Adventures

Enrique Ferrer Agius

Grado Ingeniería informática

Videojuegos

Nwdd Garcia Romero - Consultor/a

Joan Arnedo Moreno - Profesor responsable de la asignatura

Junio 2021



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada a [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Descripción del trabajo</i>
Nombre del autor:	Enrique
Nombre del consultor/a:	Nwdd Garcia Romero
Nombre del PRA:	Joan Arnedo Moreno
Fecha de entrega (mm/aaaa):	06/2021
Titulación:	Grado de Ingeniería Informático
Área del Trabajo Final:	Videojuegos
Idioma del trabajo:	Español
Palabras clave	Unity 2D, Videojuego, plataformas
Resumen del Trabajo (máximo 250 palabras):	
<p>El objetivo de este Trabajo de Fin de Grado (TFG) ha sido el desarrollo de un videojuego en todas sus etapas en el que se puedan aplicar todos los conocimientos adquiridos en el grado de ingeniería informática.</p> <p>Galleto's Adventures es un juego de plataformas 2D donde el personaje principal deberá de superar cada uno de los desafíos que se vaya encontrando por el camino para poder lograr su objetivo, poder salir del mundo en el que se encuentra. A lo largo del recorrido irá adquiriendo nuevas habilidades y conocimientos que le facilitarán el proceso para poder escapar de dicho mundo.</p> <p>Se han utilizado herramientas como ClickUp para la planificación del proyecto, Unity como motor gráfico para la creación del videojuego, Visual Studio como entorno de desarrollo para los scripts, Github para alojar el proyecto utilizando el sistema de control de versiones Git y otras como Premiere para la elaboración de un "tráiler" para promocionar Galleto's Adventures.</p> <p>Como amante de los videojuegos uno de los retos personales era desarrollar un videojuego y gracias a Galleto's Adventures se ha podido cumplir. Además del reto personal, la satisfacción ha sido positiva. El aprendizaje ha sido notable en todas las áreas del proyecto y los objetivos más importantes establecidos en la planificación se han logrado todos sin ningún tipo de contratiempo. En un futuro se espera pulir con más detalle Galleto's Adventures además de ampliar las plataformas en las que pueda ser jugado.</p>	

Abstract (in English, 250 words or less):

The goal of this final degree project has been the development of a video game at all stages in which all the knowledge acquired in the computer engineering degree can be applied.

Galletto's Adventures is a 2D platform game where the main character must overcome each of the challenges that he meets along the way to achieve his goal, and be able to escape the world he is in. Along the way, Galletto will acquire new skills and knowledge that will facilitate the process to return to home.

Tools such as ClickUp have been used for project planning, Unity as the graphics engine for the creation of the video game, Visual Studio as the development environment for the scripts, Github to host the project using the Git version control system and others such as Premiere to create a trailer to promote Galletto's Adventures.

As a video game lover, one of the personal challenges was to develop a video game and thanks to Galletto's Adventures it has been possible to fulfill it. In addition to the personal challenge, the satisfaction has been positive. Learning has been remarkable in all areas of the project and the objectives more important set out in the planning have all been achieved without any setback. In the future, Galletto's Adventures is expected to be polished in more detail, as well as expanding the platforms on which it can be played.

AGRADECIMIENTOS

Agradecer a toda mi familia y amigos por el apoyo durante todos estos años de carrera y concretamente a mi amigo Mariano (@mariano_morell_alou en Instagram para ver sus dibujos) por aportar su granito de arena en el proyecto con el dibujo de la portada.

Índice

1. Introducción	5
1.1 Contexto y justificación del Trabajo	5
1.2 Objetivos del Trabajo	6
1.3 Enfoque y método seguido	6
1.4 Planificación del Trabajo	7
1.5 Breve resumen de productos obtenidos	11
1.6 Breve descripción de los otros capítulos de la memoria	11
2. Estado del arte	11
2.1 Género plataformas	11
2.2 Herramientas disponibles	15
2.2.1 Unity	15
2.2.2 GameMaker studio 2	16
2.2.3 Construct3	17
3. Definición del juego	17
3.1 Idea del juego	17
3.1.1 Breve descripción del juego	17
3.1.2 Subgénero y referencias a videojuegos existentes	18
3.1.3 Tipo de interacción juego-jugador	18
3.1.4 Plataforma de destino	19
3.2 Conceptualización	19
3.2.1 Historia, ambientación y/o trama	19
3.2.2 Definición de los personajes / elementos	20
3.2.2.1 Player: Galleto	20
3.2.2.2 Enemigos y trampas	20
3.2.2.3 Elementos con los que puede interactuar	22
3.2.2.4 Elementos que forman parte de la escena	23
3.2.3 Objetivos planteados al jugador	25
3.2.4 Concept Art	25
3.3 Desarrollo y Roadmap	25
3.3.1 Por qué Unity 3D?	25
3.3.2 Planificación de objetivos y cuantificación del tiempo y recursos	26
4. Diseño técnico	27
4.1 Requerimientos técnicos de desarrollo	27
4.2 Herramientas utilizadas	27
4.3 Assets y recursos utilizados	27
4.4 Arquitectura del juego	29
4.5 Inteligencia artificial	30

5. Diseño de niveles	30
5.1 Nivel Introductorio	30
5.2 Nivel 1	31
5.3 Nivel 2	32
5.4 Nivel 3	32
5.5 Nivel 4	33
5.6 Nivel 5	33
5.7 Nivel 6	34
5.8 Nivel 7	35
5.9 Mapa Speed Run	35
7. Manual de usuarios	36
7.1 Requerimientos técnicos para jugar	36
7.2 Instrucciones del juego	37
7.2.1 Formas de iniciar el juego	37
7.2.2 Modos de juego	37
7.2.3 Controles	38
7.2.3 Menús, escenas e interfaces	39
9. Conclusiones	42
10. Glosario	42
11. Bibliografía	43
12. Anexos	44

Lista de figuras

Ilustración 1: Diagrama de Gantt	1
Ilustración 2: The first true platformer. © NINTENDO	12
Ilustración 3: Super Mario Bros. defined the genre. © NINTENDO	13
Ilustración 4: Super Metroid is still a top-tier platform game. © NINTENDO	13
Ilustración 5: Super Mario 64 was a revolutionary game. © NINTENDO	14
Ilustración 6: 2017's Hollow Knight is lovably old-school. © TEAM CHERRY	14
Ilustración 7: Logo de Unity	16
Ilustración 8: Logo de GameMaker Studio 2	17
Ilustración 9: Logo de CONSTRUCT3	17
Ilustración 10: Captura del juego Celeste. © Matt Makes Games	18
Ilustración 11: Captura juego Galleto's Adventures	20
Ilustración 12: Personaje principal del juego, Galleto.	20
Ilustración 13: Pinchos	21
Ilustración 14: Laser	21
Ilustración 15: Enemigo volador	21
Ilustración 16: Enemigo de tierra	22
Ilustración 17: Palanca interactiva	22
Ilustración 18: Plataforma de salto	22
Ilustración 19: Plataforma móvil tipo 1	23
Ilustración 20: Plataforma móvil tipo 2	23
Ilustración 21: Plataforma fija	23
Ilustración 22: Checkpoint	24
Ilustración 23: Llama azul para pasar al siguiente nivel	24
Ilustración 24: Ejemplo de elementos decorativos	24
Ilustración 25: Ejemplo de un diseño de nivel detallado 1.	25
Ilustración 26: Ejemplo de un diseño de nivel detallado 2.	25
Ilustración 27: Captura de la herramienta de Unity	26
Ilustración 28: Logo de ClickUp	27
Ilustración 29: Tileset empleado para el diseño gráfico del juego	29
Ilustración 30: Diagrama de estados del juego Galleto's Adventures	29
Ilustración 31: Código de la función del movimiento de los enemigos	30
Ilustración 32: Configuración del script de la IA de los enemigos	30
Ilustración 33: Ejemplo gráfico del funcionamiento del script	31
Ilustración 34: Nivel introductorio de Galleto's Adventures	32
Ilustración 35: Nivel 1 de Galleto's Adventures	32
Ilustración 36: Nivel 2 de Galleto's Adventures	33
Ilustración 37: Nivel 3 de Galleto's Adventures	34
Ilustración 38: Nivel 4 de Galleto's Adventures	34

Ilustración 39: Nivel 5 de Galleto's Adventures	35
Ilustración 40: Nivel 6 de Galleto's Adventures	35
Ilustración 41: Nivel 7 de Galleto's Adventures	36
Ilustración 42: Nivel del modo Speed Run de Galleto's Adventures	37
Ilustración 43: Logotipo itch.io	38
Ilustración 44: Captura del modo aventura	39
Ilustración 45: Captura del modo Speed Run	39
Ilustración 46: Captura del menú principal	41
Ilustración 47: Captura del menú de selección de nivel	41
Ilustración 48: Captura de la pantalla de pausa	42
Ilustración 49: Captura de la interfaz de inicio de nivel	42
Ilustración 50: Captura de la interfaz de muerte	43

1. Introducción

1.1 Contexto y justificación del Trabajo

La principal motivación tras el desarrollo del proyecto, al igual que cursar el grado de Ingeniería Informática, ha sido poder desarrollar de manera profesional un videojuego y conocer con más detalle la industria de este sector que se está convirtiendo a pasos agigantados en un fenómeno social y cultural. Más aún en estos tiempos de pandemia que está sufriendo el mundo hoy en día.

Según Palou (2020) el panorama que está dejando la pandemia en la mayoría de sectores es desolador, excepto el sector de los videojuegos que tras un año de pandemia ha visto un incremento en el consumo respecto a otros sectores del entretenimiento como podría ser la industria cinematográfica. Esto es gracias en gran parte, además del entretenimiento, a los beneficios sociales que producen los videojuegos.

Gran parte de este incremento exponencial es gracias al éxito que están teniendo los títulos “*Indie*” respecto a otros videojuegos de grandes superproducciones. Según Pável Ramirez (2021) expertos en videojuegos como Jeremy Signor asegura que esto se debe a que gran parte de los jugadores:

“Sienten mayores emociones con cosas tan elementales como la explosión de un barril en un momento de acción que con gráficos pulidos o mecánicas de juego profundas (pero a la vez más complejas).”

La popularidad de este género es debido mayormente a la producción en masa que hay gracias a la facilidad de desarrollo que otorgan herramientas como Unity, Godot y otros motores gráficos, junto con plataformas de publicación digital como Steam, Google Play Store y otras plataformas de distribución digital como Itch.io que hacen que pequeños estudios independientes sin el apoyo de una distribuidora de videojuegos sean capaces de hacer llegar sus proyectos al público de una manera más sencilla.

Este proyecto pretende mostrar la metodología de desarrollo de un videojuego en cada una de sus etapas e incentivar a que grupos reducidos de personas o nuevas pequeñas empresas se animen a desarrollar videojuegos para que de

esta manera el sector de los videojuegos esté en constante evolución e innovación.

1.2 Objetivos del Trabajo

El objetivo principal del proyecto es diseñar y desarrollar un videojuego de plataformas 2D con una versión de demostración para PC y WebGL tratando de poner en práctica todos los conocimientos adquiridos durante el grado de ingeniería informática sumado al aprendizaje y adaptación de nuevas herramientas para lograr el objetivo final. A continuación se detalla un listado de los objetivos del trabajo:

- **Objetivo principal:** Desarrollar un videojuego de plataformas 2D teniendo en cuenta todas sus fases.

- **Objetivos secundarios:**
 - Gestión del proyecto mediante el uso de herramientas. En este caso ClickUp.
 - Aprendizaje del motor Unity.
 - Aprendizaje del lenguaje C.
 - Desarrollo gráfico y técnico del juego.
 - *Testing* del videojuego.
 - Aprendizaje del uso de Adobe Premiere Pro para la edición de video y elaboración de un tráiler.
 - Investigación de plataformas de distribución digital de cara a un futuro.

- **Objetivos a largo plazo:** Continuar trabajando en el proyecto en un futuro añadiendo contenido y ampliando el número de plataformas en las que pueda ser jugado.

1.3 Enfoque y método seguido

Como bien se ha comentado en el contexto y justificación del trabajo, uno de los objetivos principales del trabajo es ponerse en la piel de los desarrolladores independientes y poder mostrar la metodología de desarrollo en cada una de sus etapas. Para ello se ha decantado por el desarrollo de un videojuego nuevo.

La creación de un videojuego suele ser un trabajo de equipo donde durante su desarrollo se aplican conocimientos y habilidades de diferentes áreas. Áreas

enfocadas al diseño gráfico como podrían ser el el dominio del *concept art* para plasmar un diseño, idea o concepto referente al juego o el modelado digital para convertir digitalmente los personajes. Por otro lado, intervienen también áreas enfocadas más al *game design* para tener una idea teórica de las mecánicas y situaciones que conforman el juego o la programación para darle vida.

Al tratarse de un proyecto unipersonal y disponer de un tiempo limitado, combinado además con una serie de tareas adicionales, ha sido necesario utilizar motores de desarrollo de terceros y además, recursos gratuitos de uso libre para suplir todo el diseño gráfico y también, para la incorporación de música y efectos de sonido.

De esta manera se dejará de lado parte del diseño gráfico para centrarse sobretodo en el *game design* y la programación del mismo. En los próximos capítulos se explicarán con más detalle que motor y recursos gratuitos se han utilizado para el desarrollo de Galleto's Adventures.

1.4 Planificación del Trabajo

Para el desarrollo del proyecto se ha establecido una planificación en función de cada una de las entregas establecidas inicialmente:

- Versión Parcial
- Versión Jugable
- Versión Final

Donde cada una de ellas conforma un “*sprint*” que tiene como objetivo alcanzar determinados hitos para asegurarse que el proyecto final pueda estar disponible en el plazo adecuado y que de esta manera no sufra ningún retraso. Algunos de los hitos establecidos en la planificación inicial no se llevaron a cabo o se modificaron debido a contratiempos que fueron surgiendo durante su desarrollo.

A continuación se presentarán cada una de las iteraciones (*sprints*) y las tareas más importantes asociadas a cada una de ellas con todas las modificaciones respecto a la planificación inicial. Además, al final se incluirá un diagrama de Gantt para ilustrar de manera gráfica la planificación.

Sprint Versión Parcial (5 semanas)

Pese a que el entregable consistía en tener una versión parcial del juego, se planteó como objetivo principal de este *sprint* obtener un primer prototipo

jugable del modo principal con todos los movimientos básicos del personaje principal. Las tareas de esta iteración son:

- **Preparar el entorno de trabajo.** Descargar todas las herramientas necesarias para el desarrollo de juego y leer el Unity User Manual[3].
- **Diseñar el menú del videojuego.**
- **Diseñar el menú de selección de nivel.**
- **Diseñar el nivel introductorio del juego.** Crear el diseño gráfico empleando los tilemaps del motor de Unity.
- **Diseñar el personaje.** Añadir las animaciones del personaje y desarrollar un script para gestionar todos los movimientos principales del jugador.
- **Implementar el movimiento de la cámara,** que debe seguir al personaje principal.
- **Diseño enemigo y plataforma móvil.** Implementar el primer enemigo estático y la plataforma móvil junto al desarrollo de los scripts para gestionar su comportamiento.
- **Diseño de los tres primeros niveles del juego** empleando el componente Tilemap.
- **Test para comprobar el correcto funcionamiento.**

Sprint Versión Jugable (5 semanas)

El objetivo principal de este *sprint* es obtener una versión del videojuego lo más parecido a la versión final a falta de algunos detalles que facilitan y mejoran la experiencia de juego. Estas son las tareas asociadas a este *sprint*:

- **Diseño de los cuatro últimos niveles del juego** empleando los tilemaps.
- **Diseño de nuevos enemigos y nuevas plataformas.** Añadir un nuevo enemigo estático, un enemigo de tierra y un enemigo volador con todas sus animaciones y un script para gestionar el comportamiento.
- **Modificar script de control de personaje** para añadir nuevas habilidades con sus animaciones.
- **Script para implementar la funcionalidad del menú principal y el menú de selección de nivel.**
- **Test para comprobar el correcto funcionamiento.**

Sprint Versión Final (4 semanas)

El objetivo principal de esta última iteración es dar por finalizado el modo principal del juego pulido con todos los detalles que mejoran la experiencia de juego, además de diseñar, desarrollar y finalizar un modo secundario de juego

(*Speed Run*) para complementar el modo principal. Las tareas a complementar de esta iteración son:

- **Añadir mejoras visuales**, como nueva decoración, diálogos para informar al jugador, etc.
- **Añadir efectos de sonido y música** para los niveles y el personaje principal.
- **Implementar de manera definitiva el modo secundario de juego**, haciendo uso de los tilemaps para el diseño gráfico y desarrollando un script para gestionar su funcionalidad.
- **Elaborar un trailer del videojuego**, empleando un software de edición de vídeo.
- **Conjunto definitivo de pruebas**, para analizar el correcto funcionamiento del juego.

Tras la finalización de cada *sprint* se contactó con jugadores para que jugaran a cada una de las versiones del juego y de esta manera obtener un *feedback* y poder implementar las sugerencias de cara a la versión final del juego.

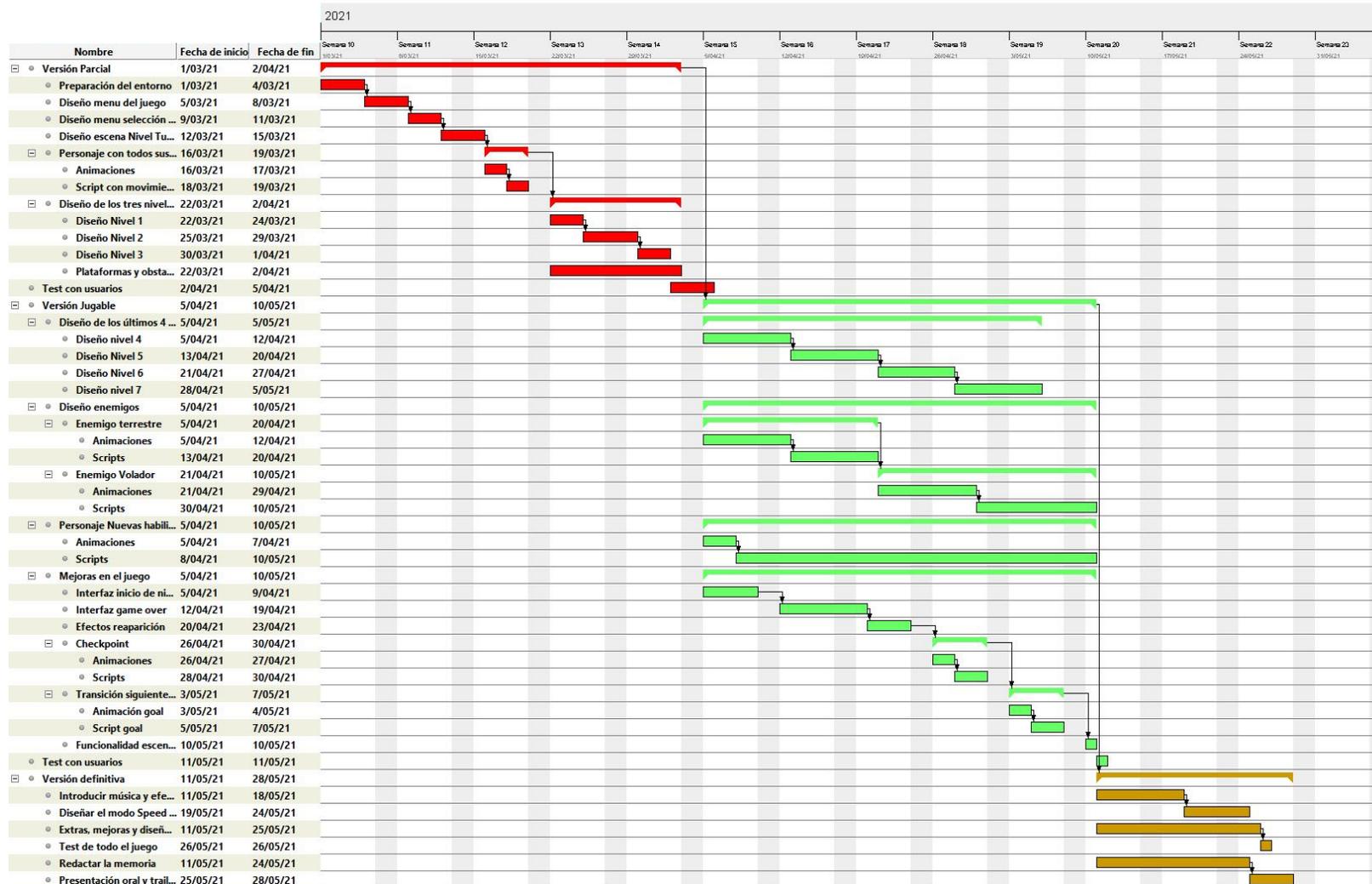


Ilustración 1: Diagrama de Gantt

1.5 Breve resumen de productos obtenidos

- *Release* de la versión definitiva del juego.
- Memoria final del proyecto.
- Proyecto completo de Unity.
- Video explicativo del proceso de creación del proyecto.
- Tráiler del videojuego.

1.6 Breve descripción de los otros capítulos de la memoria

En los próximos capítulos se revisará el arte del juego, haciendo hincapié en el género y la tecnología utilizada para desarrollar videojuegos del mismo estilo.

Además, se explicarán de manera detallada aspectos generales del juego como la historia, los personajes y otros elementos secundarios. También se entrará en más detalle sobre aspectos más técnicos como el diseño y la implementación del juego.

Y por último se describirán todos los niveles que conforman el modo principal y el modo secundario, se incluirá un manual de usuario con los requerimientos técnicos para jugar a *Galleto's Adventures* y se finalizará con las conclusiones finales del proyecto.

2. Estado del arte

2.1 Género plataformas

Durante toda la historia de los videojuegos cada generación ha estado determinada e influenciada por varias consolas y videojuegos. Desde la primera videoconsola, *Magnavox Odyssey*, hasta las consolas de la última generación como la *PlayStation 5*, se ha podido apreciar de manera notable una gran evolución en los juegos y en los géneros donde cada vez existe una mayor competencia entre ellos.

A pesar de la gran cantidad de géneros que hay hoy en día y el renombre que tienen alguno de ellos, los videojuegos de plataformas han sido de los juegos más populares y apreciados en todo el mundo a lo largo de estos años.

Según Dan Maher, presentador del programa británico sobre videojuegos Video Game Nation, afirma sobre los juegos de plataformas que:

“El concepto se ha mantenido más o menos igual” (Powell, 2015).

“Saltar de plataforma en plataforma y llegar al último nivel, coleccionando baratijas por el camino” (Powell, 2015).

Se puede ver en juegos como Mario en el pasado o Celeste en esta última generación. Son juegos de plataformas que se las han ingeniado para seguir siendo relevantes en la industria de los videojuegos. Gran parte esto es gracias a una gran reinención e innovación a lo largo de todos estos años.

Según Gaming, N. (2020) los videojuegos de plataformas han marcado épocas desde sus inicios hasta en la actualidad. A continuación repasamos las más importantes:

- **Single Screen:** las plataformas en los inicios estaban limitadas a una sola pantalla. No había desplazamientos, ni saltos, pero los jugadores podían caer y subir mediante escaleras. Un ejemplo de esta era fue *Space Panic* ya que fue el primer videojuego que involucró este tipo de mecánicas. Pese a que no tuvo mucho éxito, inspiró a una gran cantidad de títulos de la época como *Donkey Kong*.



Ilustración 2: *The first true platformer.* © NINTENDO

- **Side-Scrolling:** esta época fue una gran innovación respecto a la anterior porque la partida se desarrollaba en un solo plano lateral. El jugador podía moverse de izquierda a derecha y además saltar. Jump Bug fue el primer videojuego que introdujo esta nueva tecnología, pero no brilló tanto como Super Mario Bros que trajo niveles de desplazamiento lateral, enemigos, jefes, caminos secretos, etc.

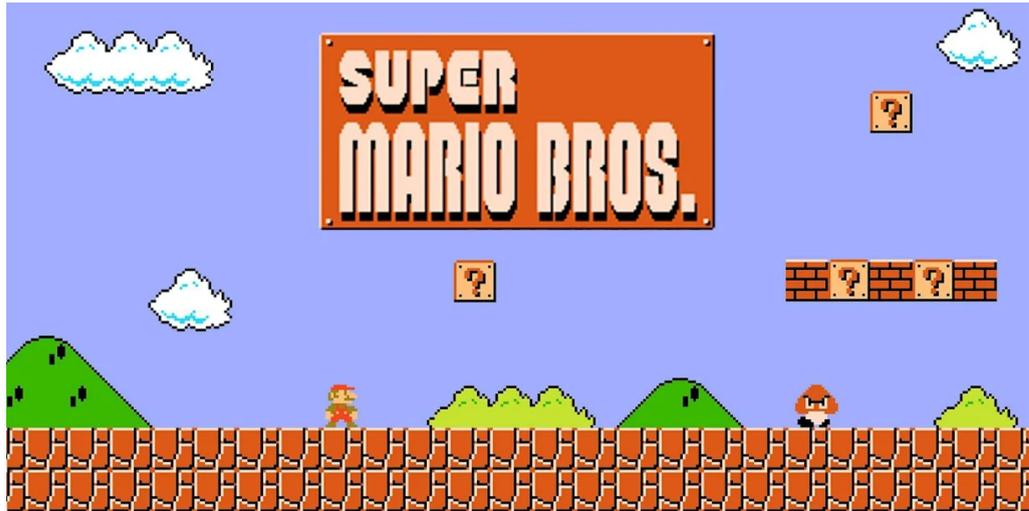


Ilustración 3: *Super Mario Bros. defined the genre.* © NINTENDO

- **The 16-bit:** el nombre de la época fue caracterizado por los microprocesadores de las consolas de la cuarta generación, donde normalmente eran de 16-bit. Gracias a las mejoras de las consolas, los juegos de esta era se caracterizaban por tener múltiples controles, nuevos efectos como el efecto *parallax* y ambientes de múltiples estratos, nueva gama de colores en pantalla y una mejora en la música. La introducción de todas estas nuevas características y que los desarrolladores empezaban a dar un enfoque en 2.5D dio a pie a exitosos títulos de videojuegos como Super Metroid o Rayman entre otros muchos.



Ilustración 4: *Super Metroid is still a top-tier platform game.* © NINTENDO

- **3D Platform:** Alpha Waves fue en 1990 el primer juego de plataformas en 3D que incorporó una cámara, movimientos y gráficos 3D, pero no fue hasta el año 1996 con Super Mario 64 para la Nintendo 64 que se estableció como el estándar y referente para todos los juegos en plataformas en 3D. Introdujo niveles de final abierto, dando la posibilidad de explorar y elegir el propio camino. Incluyó también mejoras en la cámara y controles respecto a otros videojuegos 3D de la época.



Ilustración 5: *Super Mario 64 was a revolutionary game.* © NINTENDO

- **The retro Revival:** la aparición de numerosos desarrolladores *indies*, hizo que gran parte de ellos debutaron con juegos plataformas 2D lo más parecido a la vieja escuela. Mejorando notablemente la calidad tanto gráfica como mecánica para que de esta manera se asemeja a los videojuegos de otros géneros de la época. Hollow Knight y Cuphead son juegos plataformas con mecánicas y gráficos sólidos de los últimos años.



Ilustración 6: *2017's Hollow Knight is lovably old-school.* © TEAM CHERRY

Aunque los juegos de plataformas han crecido exponencialmente a lo largo de todos estos últimos años, no se sabe qué dirección tomarán en un futuro. Algunos desarrolladores como Rex Crowle, creador del juego de plataformas *Tearaway*, afirman que:

“Seguirán existiendo, simplemente de distintas formas” Powell (2015).

2.2 Herramientas disponibles

En la actualidad existen multitud de *engines* y motores que pueden satisfacer en gran parte las necesidades de los desarrolladores independientes ya que la mayoría de estos disponen de licencia gratuita y precios asequibles donde te permiten automatizar gran parte del desarrollo del videojuego haciendo uso de sus bases.

A continuación, se realizará una recopilación de los mejores *engines* y motores que están disponibles en el mercado para el desarrollo de un juego de plataformas con la posibilidad de desplegarse en múltiples plataformas y los servicios que ofrecen cada uno de ellos.

2.2.1 *Unity*

Se trata de uno de los mejores motores 2D y 3D que existen actualmente para el desarrollo *indie* debido a la gran comunidad de usuarios que tiene. Dispone de su propio manual de usuario (*Unity User Manual*), y además al ser uno de los más conocidos, hay a la disposición de cualquier usuario multitud de documentación, foros y comunidades. Convirtiéndose de esta manera, en uno de los motores favoritos para aquellos que quieren iniciarse en la industria de los videojuegos.

Las principales funcionalidades que proporciona el motor de Unity entre otras son:

- Motor gráfico para renderizar tanto los gráficos 2D como 3D.
- Motor físico que simula las leyes de la física.
- Control completo de las animaciones.
- Audio que incluye sonido completo, mezcla, mastering...
- *Scripting*.
- etc.

Además dispone de una interfaz bastante sencilla, por lo que resulta fácil interactuar con ella y combinado con la gran variedad de documentación y el uso del lenguaje de programación C#, hace que la curva de aprendizaje sea bastante fácil. También tiene la capacidad de exportar a un multitud de plataformas tanto PC, móviles, consolas y dispositivos virtuales.

Por otro lado, destacar la gran cantidad de servicios que Unity ofrece como:

- Monetizar el juego.
- Realizar analíticas.
- Realizar construcciones remotas en la nube.
- Distribuir los juegos a múltiples tiendas de aplicaciones a través de una consola.
- etc.



Ilustración 7: Logo de Unity

2.2.2 *GameMaker studio 2*

Es un motor de desarrollo tanto en 2D como en 3D pero está más enfocado al desarrollo en dos dimensiones debido a un sistema sencillo de *Drag&Drop* (arrastrar y soltar) con objetos, eventos y acciones. Esto sumado a una interfaz muy sencilla permite hacer prototipos de juegos a los pocos minutos.

Además tiene su propio lenguaje llamado *GML*, con una sintaxis bastante sencilla, que permite realizar proyectos más avanzados, ya que nos permite utilizar físicas o poder hacer uso de inteligencia artificial entre otras muchas funcionalidades.

Permite exportar a gran cantidad de plataformas desde dispositivos móviles, hasta ordenadores y consolas, y al igual que unity ofrece otros servicios como monetización y análisis de estadísticas.

A pesar de la sencillez que puede resultar utilizar este motor para desarrollar los videojuegos, no se trata de la primera opción por parte de los desarrolladores independientes debido a que se trata de un motor de pago con diferentes precios que vendrán dados en función de las licencias. Por ejemplo, la licencia para desarrollar en consola puede llegar a costar 799 dólares .



Ilustración 8: Logo de GameMaker Studio 2

2.2.3 Construct3

Construct es uno de los motores más adecuados para los principiantes que quieran desarrollar juegos en 2D por las facilidades que ofrece gracias a su interfaz y al sistema de *scripting* visual similar al *Drag&Drop* de *GameMaker*.

Del mismo modo que los otros motores anteriores, Construct3 utiliza el lenguaje JavaScript para el desarrollo y poder trabajar de esta manera con elementos más técnicos que no serían posible con el sistema de arrastrar y soltar.

Permite exportar en una gran variedad de plataformas al igual que *GameMaker*, aún así, Construct3 se centra especialmente en los navegadores, haciendo énfasis en HTML5, donde el propio editor es ejecutado como una aplicación web.

Pese a la facilidad, *Construct3* no es gratuito. La licencia es anual con un coste aproximado de 99 dólares al año con opción de exportar a todas las plataformas. El precio para ser usado hace que no sea la opción de muchos desarrolladores *indie*.



Ilustración 9: Logo de CONSTRUCT3

3. Definición del juego

3.1 Idea del juego

3.1.1 Breve descripción del juego

Galleto's Adventures es un juego en dos dimensiones donde el personaje principal, Galleto, deberá de superar cada uno de los desafíos que se vaya encontrando por el camino para poder lograr su objetivo, poder salir del mundo en el que se encuentra. A lo largo del recorrido deberá hacer uso de sus habilidades y conocimientos para poder afrontar todos y cada uno de los obstáculos.

3.1.2 Subgénero y referencias a videojuegos existentes

Galleto's Adventures pertenece al género de plataformas, que en algunas clasificaciones de videojuegos se considera un subgénero de Acción. Se trata de un género que se caracteriza por tener que caminar, correr, saltar o escalar sobre una serie de plataformas y acantilados, con enemigos y obstáculos, hasta llegar a una meta.

Un videojuego reciente con una mecánica y estética similar podría ser Celeste. La protagonista Madeline deberá sobrevivir en su viaje hasta la cima de la montaña.



Ilustración 10: Captura del juego Celeste. © Matt Makes Games

Conforme Madeline inicia su travesía hasta la cima de la montaña se irá topando con numerosos escenarios peligrosos fruto de la diversidad de terrenos que se pueden encontrar en una montaña. Cada uno de ellos tendrá un papel importante en la trama del videojuego.

Me hubiera gustado que Galleto's Adventures tuviera una historia, ambientación y trama parecida a la que ha conseguido el videojuego Celeste. Lamentablemente no se ha podido conseguir debido a los sprites de terceros utilizados para el desarrollo del videojuego, que han sido un limitador para innovar en este aspecto.

3.1.3 Tipo de interacción juego-jugador

La interacción entre el jugador y el juego es bastante sencilla. Básicamente, el jugador deberá hacer uso de las teclas del teclado W, A, S, D y Space para interactuar con el actor principal del videojuego, Galleto. El conjunto de todas estas teclas conforman el movimiento básico del personaje.

Al igual que en el juego Celeste, el jugador deberá hacer uso de ellas para poder desplazarse entre plataformas y superar cada uno de los desafíos.

3.1.4 Plataforma de destino

La plataforma principal donde Galleto's Adventures estará disponible será en ordenador, tanto PC como Mac, como en formato WebGL. Se trata de una de las mejores plataformas ya que en ella no encontraremos apenas restricciones y limitaciones para el diseño y la elaboración del videojuego. Sin embargo se espera en un futuro poder hacer llegar el videojuego a otras plataformas donde el alcance de la gente sea mucho mayor. Dando prioridad a los dispositivos móviles donde a día de hoy es una de las plataformas con mayor número de usuarios.

3.2 Conceptualización

3.2.1 Historia, ambientación y/o trama

Galleto no sabe ni como y porque se encuentra encerrado en un mundo tan oscuro y tenebroso. Al darse cuenta de que no es un sueño, hará todo lo posible y más para poder salir de allí. Durante el largo camino hará uso de sus habilidades para afrontar los obstáculos y sus miedos.

Para la ambientación se ha elegido un tileset que combinan a la perfección con la historia de Galleto's Adventures. En la siguiente captura se muestra un ejemplo de escenario que se pueden crear con el tileset que se utilizará para la parte gráfica del videojuego.



Ilustración 11: Captura juego Galleto's Adventures

Además, para completar y mejorar la ambientación y crear una buena inmersión al jugador, se hará uso de música y efectos sonoros.

3.2.2 Definición de los personajes / elementos

3.2.2.1 *Player*: Galleto

Galleto es el personaje principal de nuestra aventura y el jugador deberá de guiarle en todo momento para que logre su objetivo, escapar del mundo en el que se encuentra. Galleto puede interactuar y colisionar con elementos que se mencionan en los apartados siguientes.

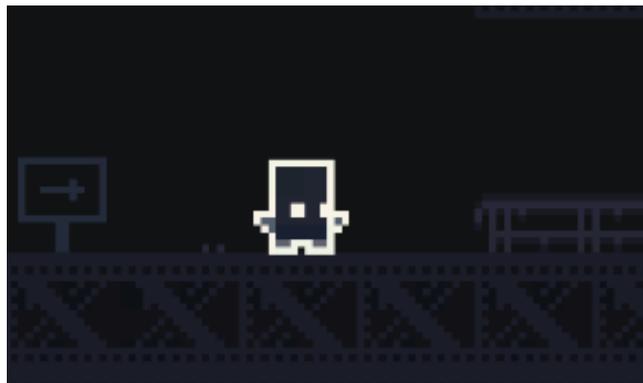


Ilustración 12: Personaje principal del juego, Galleto.

3.2.2.2 Enemigos y trampas

Los obstáculos que deberá hacer frente Galleto serán trampas y enemigos, que pueden ser voladores o terrestres.

Las trampas que Galleto deberá superar son:

- **Pinchos:** trampa estática. El mínimo roce de Galleto con un pincho significa la muerte.



Ilustración 13: Pinchos

- **Láser:** al igual que los pinchos, se trata de una trampa estática que se activa y se desactiva durante un determinado tiempo. También el mínimo contacto supondrá la muerte.

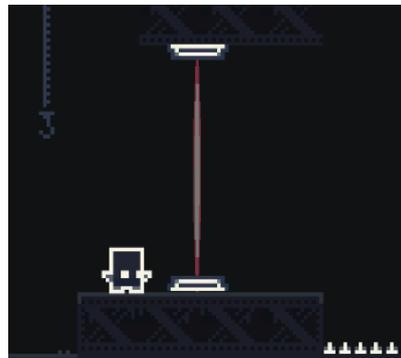


Ilustración 14: Láser

Además de las trampas, Galleto deberá enfrentarse a dos tipos de enemigos que deberá superar si quiere continuar por el camino:

- **Enemigo volador:** cualquier contacto de Galleto con el enemigo volador supondrá la muerte. Si Galleto desea eliminarlo, deberá saltar un total de dos veces encima de él.

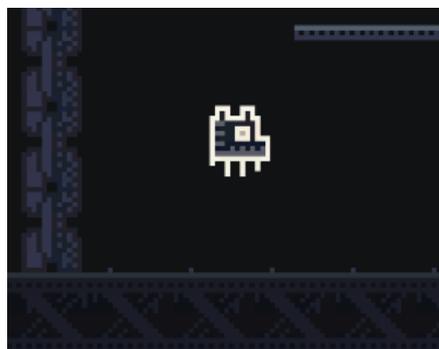


Ilustración 15: *Enemigo volador*

- **Enemigo terrestre:** misma mecánica que el enemigo volador, solo que en este caso estará situado sobre las plataformas.



Ilustración 16: Enemigo de tierra

3.2.2.3 Elementos con los que puede interactuar

El personaje principal deberá de interactuar con una serie de elementos para poder superar cada uno de los niveles. Entre ellos se encuentran:

- **Palancas:** puede activarlas en cualquier momento para habilitar nuevos caminos.



Ilustración 17: Palanca interactiva

- **Plataformas de salto:** puede impulsarse sobre ellas para acceder a nuevos lugares.



Ilustración 18: Plataforma de salto

- **Plataformas móviles:** Galleto podrá hacer uso de ellas para desplazarse entre plataformas donde no pueda llegar mediante su salto.



Ilustración 19: Plataforma móvil tipo 1

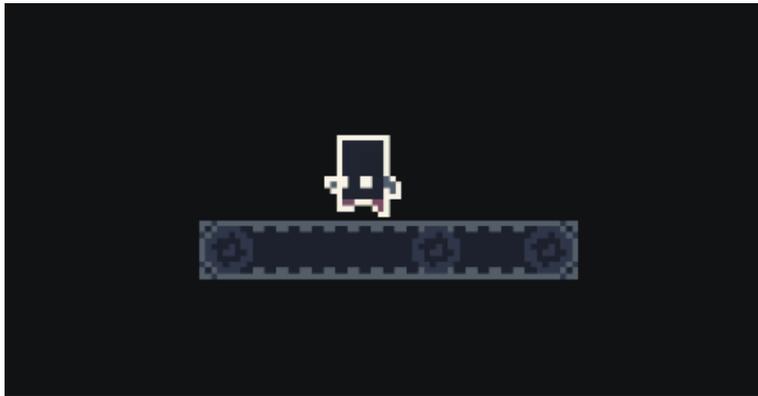


Ilustración 20: Plataforma móvil tipo 2

3.2.2.4 Elementos que forman parte de la escena

- **Plataformas fijas:** conjunto de cubos que conforman el suelo y los muros de la escena.



Ilustración 21: Plataforma fija

- **Checkpoints:** punto de control que guarda la partida. Si Galleto muere y ha pasado por un punto de control, reaparecerá en él en lugar del *spawn* del inicio del nivel.

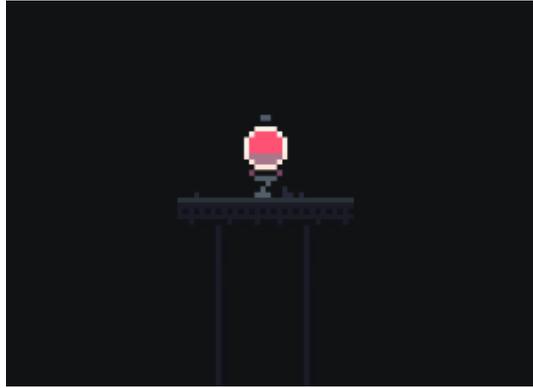


Ilustración 22: *Checkpoint*

- **Goal:** llama azul que deberá conseguir Galleto para pasar de nivel.



Ilustración 23: Llama azul para pasar al siguiente nivel

- **Elementos decorativos:** tuberías, cadenas, bidones, señales y muchos otros más que ayudan a la ambientación de la escena.

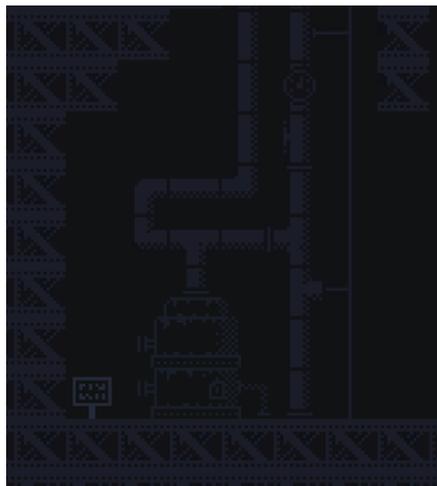


Ilustración 24: Ejemplo de elementos decorativos

3.2.3 Objetivos planteados al jugador

El jugador debe comprender a la perfección las mecánicas del personaje principal para superar todo tipo de obstáculos.

Entre estos obstáculos están los enemigos, que si quiere hacer frente a ellos debe saltar encima de ellos un total de dos veces para matarlos.

Una vez consiga la llama azul finalizará el nivel y se cargará automáticamente el siguiente nivel. Conforme avanzan los niveles, la dificultad aumenta.

3.2.4 Concept Art

En cuanto al concepto del arte, nos hemos basado en los siguientes dibujos de otros autores que transmiten una idea visual y estilo que Galleto's Adventures quiere conseguir para el diseño de los mapas de cada nivel.

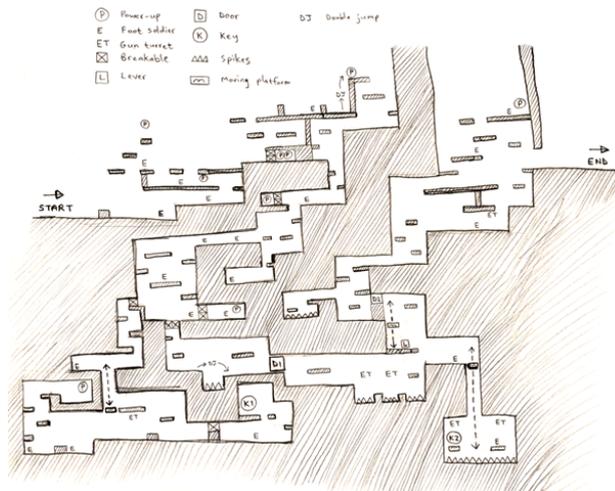


Ilustración 25: Ejemplo de un diseño de nivel detallado 1. Realizado por Dev.Mag

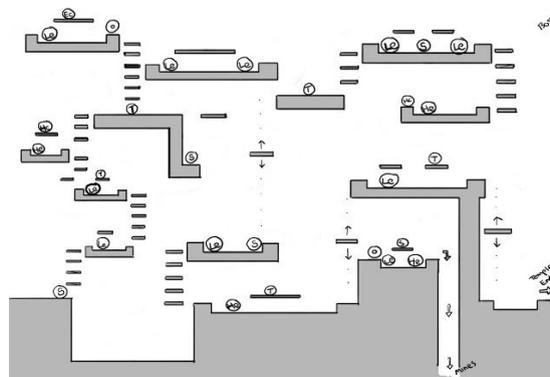


Ilustración 26: Ejemplo de un diseño de nivel detallado 2. Realizado por Rachelb5

3.3 Desarrollo y Roadmap

3.3.1 Por qué Unity 3D?

Existen una variedad extensa de herramientas que facilitan el desarrollo de un videojuego. Para el desarrollo de Galleto's Adventures se ha decantado por el uso de un motor de juego que nos proporcione una serie de funcionalidades avanzadas que nos faciliten el desarrollo del videojuego. Entre estas se encuentran: un motor gráfico para renderizar gráficos 2D, un motor físico que simula todas las leyes de la física, motor de audio, diseño y gestión de animaciones, y muchas otras más. Existen una extensa diversidad de engines al alcance de un desarrollador de videojuegos. La elección de uno u otro dependerá de muchos factores como la facilidad de aprendizaje del entorno de desarrollo o si se conoce o no el lenguaje con el que se trabaja. En este proyecto se ha decantado por utilizar Unity3D.

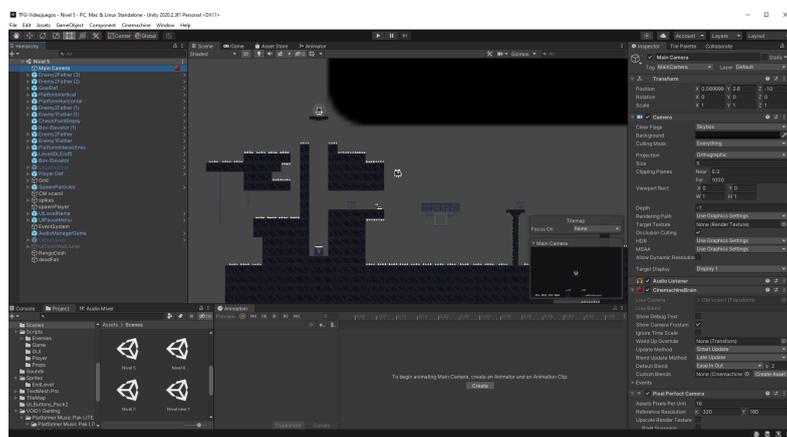


Ilustración 27: Captura de la herramienta de Unity

El factor más importante de esta decisión ha sido porque en lo personal ya se había trabajado anteriormente con este engine y a mi parecer se trata de uno de los engines más completos y con la interfaz más amigable que hay a la disposición de cualquier usuario que quiera desarrollar un videojuego. Además, Unity ofrece una gran serie de servicios que nos pueden ayudar en un futuro a monetizar nuestro juego, realizar analíticas, realizar construcciones remotas en la nube similar a un sistema de control de versiones... También con Unity tendremos a nuestra disposición una documentación bastante extensa y foros que nos facilitará la curva de aprendizaje.

3.3.2 Planificación de objetivos y cuantificación del tiempo y recursos

Para la planificación y organización del desarrollo del proyecto se hará uso de la herramienta ClickUp, un software de gestión de proyectos. Con esta herramienta se ha creado una lista por cada una de las versiones del juego con todas las tareas y objetivos que se deben cumplir en los plazos determinados.

Además la herramienta ofrece una vista con un diagrama de Gantt donde nos facilita una visión más general del proyecto, así como un seguimiento del mismo.



Ilustración 28: Logo de ClickUp

Para conocer en más detalle las tareas asociadas a cada una de las versiones visitar el apartado 1.4 Planificación de trabajo, donde además, se adjunta un diagrama de Gantt para visualizar de manera gráfica el tiempo de dedicación previsto para cada tarea de cada una de las versiones del juego. Gracias a ClickUp se ha podido medir el progreso y detectar cualquier contratiempo que haya podido afectar a los plazos de los objetivos del proyecto.

4. Diseño técnico

4.1 Requerimientos técnicos de desarrollo

Para el desarrollo del juego se ha utilizado un PC personalizado con las siguientes características:

- **Sistema Operativo:** Windows 10.
- **CPU:** AMD Ryzen 5 2600x.
- **Tarjeta gráfica:** Gigabyte GeForce GTX 1660 SUPER OC 6GB GDDR6
- **Memoria RAM:** 6 GB (2 x 8 GB) DDR4 3200 MHz

4.2 Herramientas utilizadas

Además de el motor de Unity para el desarrollo de Galleto's Adventures y ClickUp para la planificación de los objetivos, se han utilizado las siguientes herramientas:

- **Visual Studio Code:** IDE multiplataforma utilizado para el desarrollo de los *scripts*.

- **GitHub:** Plataforma de alojamiento de código para el control de versiones del proyecto. Se ha utilizado la aplicación GitHub Desktop para interactuar con GitHub de una manera más sencilla y de esta manera sustituir la mayoría de comandos de Git.
- **OBS Studio:** Programa de grabación de la pantalla utilizado de cara a las explicaciones del trabajo y para la elaboración del tráiler.
- **Adobe Premiere Pro:** Software de edición de video utilizado para la edición del trailer de Galleto's Adventures.

4.3 Assets y recursos utilizados

Como bien se ha comentado anteriormente, por motivos de tiempo y falta de conocimientos ha sido necesario utilizar varios *assets* de terceros para cubrir la parte gráfica del videojuego como la música y efectos de sonido del mismo. A continuación se detallarán en una tabla todos los assets utilizados junto el enlace web donde han sido adquiridos.

Nombre	Autor	Licencia	Enlace
16x16 Industrial Tileset	0x72	CC0 1.0	https://0x72.itch.io/16x16-industrial-tileset
Pixel Art Effect - FX065	NYKNCK	CC0 1.0	https://kvsr.itch.io/fx065
Pixel Art Effect - FX065	NYKNCK	CC0 1.0	https://kvsr.itch.io/fx065
Sound FX Pack 1	edwardcufaude	CC0 1.0	https://edwardcufaude.itch.io/soundfxpack1
Jump Sound	SMStudios	CC0 1.0	https://simon-develop.itch.io/jump-sound-100-free
Platformer Music Pack LITE	VOiD1 Gaming	CC0 1.0	https://assetstore.unity.com/packages/audio/ambient/platformer-music-pack-lite-178337#content
Close your Eyes	Infraction	CC-3.0	https://inaudio.org/license-terms/
True 2D™ Font Family	Galdino Otten	CC-0 1.0	https://www.1001fonts.com/true-2d-font.html

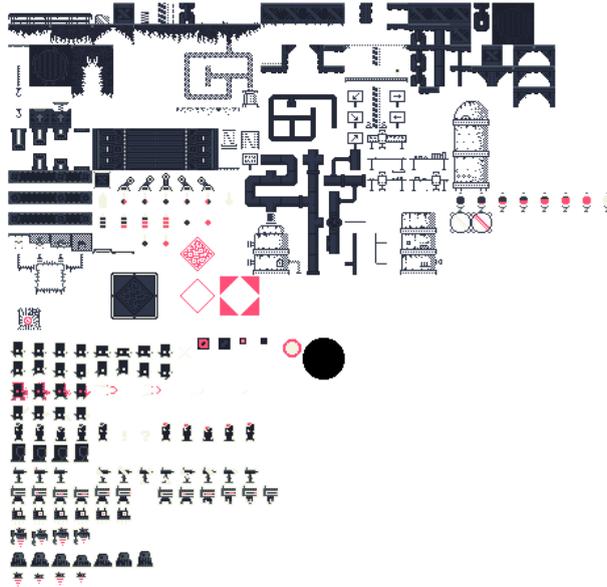


Ilustración 29: Tileset empleado para el diseño gráfico del juego

4.4 Arquitectura del juego

Mediante el siguiente diagrama de estados, se puede observar el comportamiento que tendrá el videojuego en función de las interacciones que vaya teniendo el jugador.

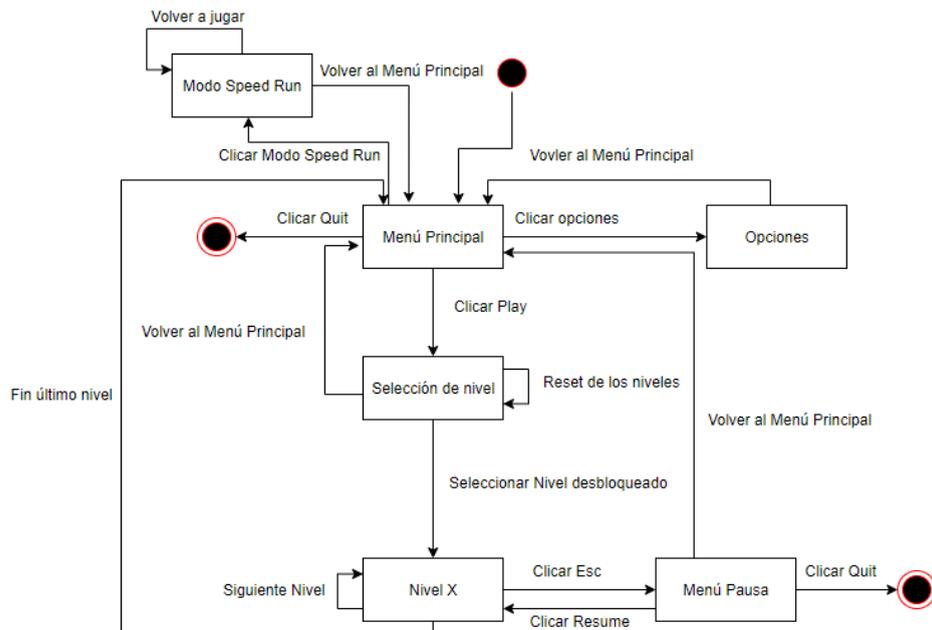


Ilustración 30: Diagrama de estados del juego Galleto's Adventures

4.5 Inteligencia artificial

Como bien se ha mencionado anteriormente en Galleto's Adventures se deberá hacer frente a dos tipos de enemigos, enemigos de tierra y enemigos voladores.

Ambos tipos de enemigos presentan un mecanismo de IA bastante básico. El script en cuestión hará que los enemigos se muevan de un punto a otro, para finalmente acabar en la posición inicial y volver a iniciar esta secuencia. A continuación se mostrará una captura de dicho script con la función del movimiento.

```
void movimientoEnemigo() {  
    StartCoroutine(ComprobarMovimientoEnemigo());  
    transform.position = Vector2.MoveTowards(transform.position, spotsMovimiento[i].transform.position, velocidad * Time.deltaTime);  
    if(Vector2.Distance(transform.position, spotsMovimiento[i].transform.position) < 0.1f){  
        if(tiempoEspera <= 0){  
            if(spotsMovimiento[i] != spotsMovimiento[spotsMovimiento.Length-1]){  
                i++;  
            }  
            else{  
                i = 0;  
            }  
            tiempoEspera = inicioTiempoEspera;  
        }  
        else{  
            tiempoEspera -= Time.deltaTime;  
        }  
    }  
}
```

Ilustración 31: Código de la función del movimiento de los enemigos

El vector de spots de movimiento no permitirá fijar spots en el mapa donde queramos que el enemigo se vaya moviendo. En la siguiente imagen se puede apreciar con más detalle:

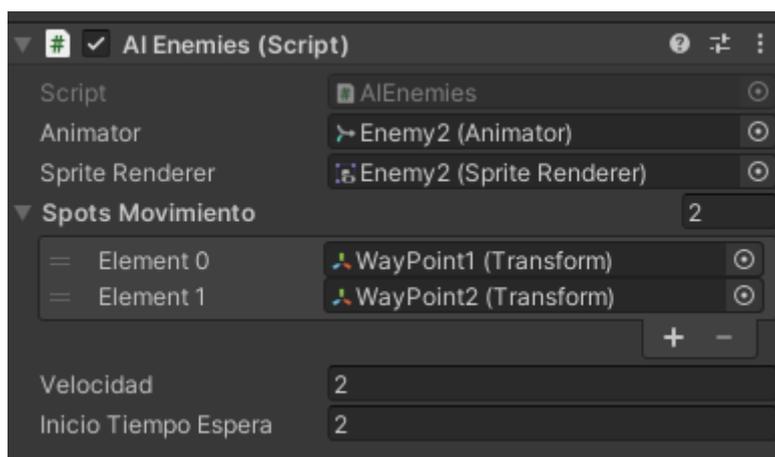


Ilustración 32: Configuración del script de la IA de los enemigos

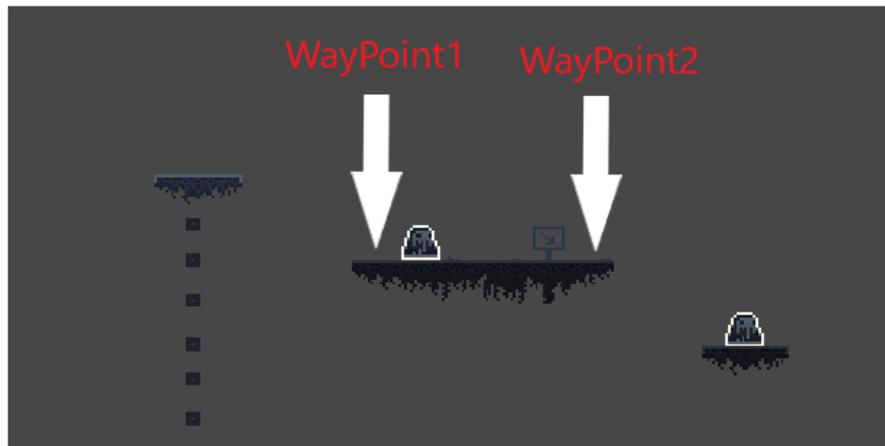


Ilustración 33: Ejemplo gráfico del funcionamiento del script

5. Diseño de niveles

A continuación se detallan los criterios de diseño de cada uno de los niveles que se pueden encontrar en Galleto's Adventures.

5.1 Nivel Introductorio

El nivel introductorio tiene como objetivo, como bien dice su nombre, servir como una especie de introducción donde se explicarán los controles y mecánicas principales del personaje para que se familiarice de cara a futuros niveles más complejos.

Como se puede apreciar en la imagen se trata de un mapa pequeño donde apenas encontrará dificultades. El jugador se hará una idea de los peligros y elementos básicos con los que podrá interactuar en los próximos niveles.



Ilustración 34: Nivel introductorio de Galleto's Adventures

5.2 Nivel 1

El primer nivel se puede observar a primera vista que se trata de un nivel más extenso donde el jugador deberá poner en práctica todas las habilidades y mecánicas aprendidas en el nivel introductorio.

Para poder llegar hasta la meta el personaje deberá superar dos zonas peligrosas. En la primera zona podrá ver la muerte por caer sobre pinchos afilados y en la segunda zona por caer al vacío. Además en esta segunda zona, deberá hacer un buen uso de dos tipos de plataformas móviles que nunca antes había visto si no quiere caer al vacío y ver la muerte.

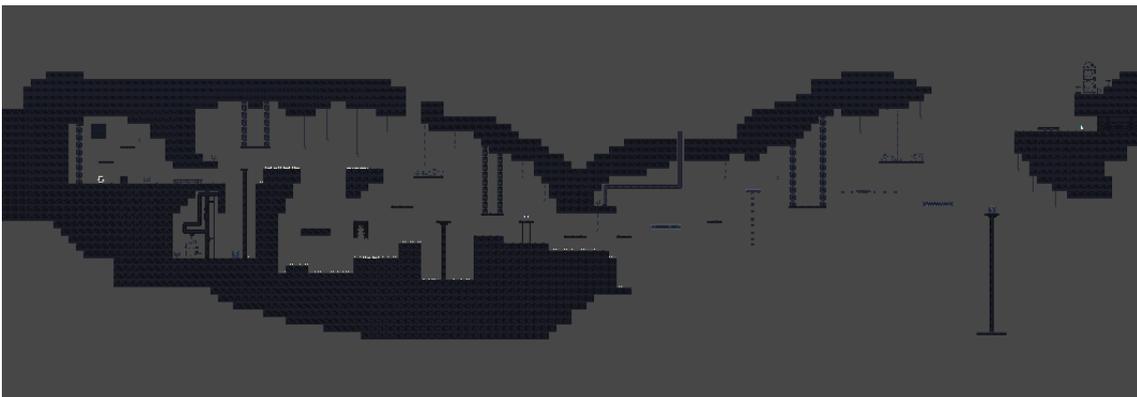


Ilustración 35: Nivel 1 de Galleto's Adventures

5.3 Nivel 2

El nivel dos sigue la misma línea que el primer nivel. Se trata de un nivel con una extensión y peligros similares al nivel anterior. Con este nivel se pretende incorporar elementos vistos anteriormente, pero en una mayor cantidad para que el jugador los conozca en mayor profundidad y que de esta manera pueda aprender con más detalle las habilidades y mecánicas principales del juego con más detalle.

Prácticamente presenta la misma dificultad que el primer nivel exceptuando la zona justo después del checkpoint, donde el jugador deberá enlazar cuatro plataformas de salto consecutivas para poder llegar hasta una plataforma móvil para poder continuar el camino hasta llegar a la meta.



Ilustración 36: Nivel 2 de Galleto's Adventures

5.4 Nivel 3

El nivel tres presenta una extensión bastante más corta que los dos primeros niveles del juego y esto es debido a la introducción de una nueva mecánica del personaje principal. A partir de este nivel, el personaje será capaz de realizar en el aire un *dash* para poder desplazarse a una mayor longitud respecto al salto normal.

Se trata de un nivel con una menor extensión debido a que en la mayoría de plataformas solo se puede llegar mediante el uso de esta nueva habilidad. El objetivo de este nivel, es que el jugador sea capaz de familiarizarse con esta nueva mecánica. El jugador deberá conocer y saber a la perfección cuando usar el *dash* si quiere superar todos los desafíos que se le presenten en los siguientes niveles.

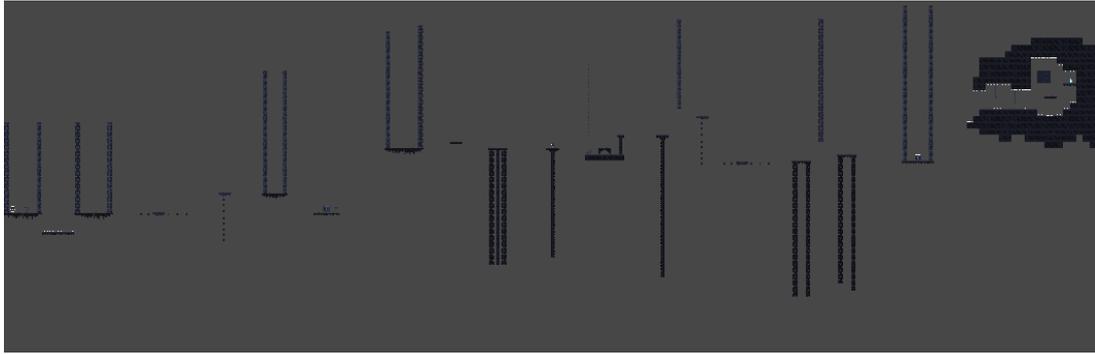


Ilustración 37: Nivel 3 de Galleto's Adventures

5.5 Nivel 4

En el nivel cuatro aparecerán no sólo elementos y peligros vistos en anteriores niveles, sino que también, será el primero donde aparezcan enemigos y nuevas interacciones con plataformas.

El jugador deberá ser capaz de buscar las debilidades a enemigos terrestres como voladores y de interactuar con plataformas que deberán ser activadas mediante el uso de una palanca. Se trata de un nivel corto pero lleno de acción.



Ilustración 38: Nivel 4 de Galleto's Adventures

5.6 Nivel 5

El nivel cinco es muy similar al anterior. Presenta los mismos elementos y peligros, pero con la diferencia que se deberán poner en prácticas habilidades y mecánicas del personaje principal, deslizarse y hacer *wall jump* por las paredes.

Se tratan de habilidades que podían ser usadas en niveles anteriores, pero en este nivel será necesario recurrir a ellas para poder superar un tramo repleto de pinchos afilados y otro tramo donde será necesario escalar una pared enorme.

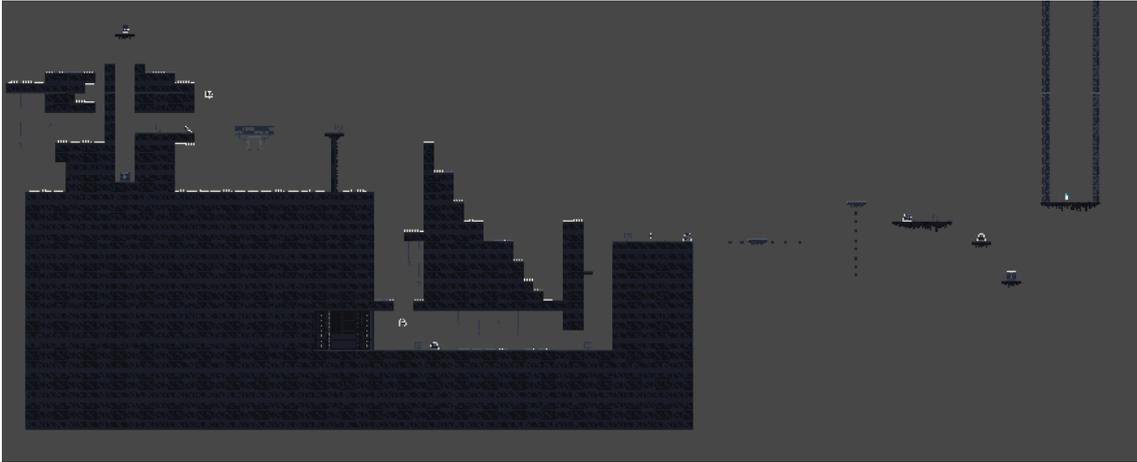


Ilustración 39: Nivel 5 de Galleto's Adventures

5.7 Nivel 6

El nivel seis incorpora un nuevo peligro, un rayo láser. El jugador deberá conocer a la perfección cuando se activa y se desactiva para poder pasar por él en el momento adecuado. En caso de que toque el láser el personaje morirá al instante.

En este nivel es importante mencionar dos tramos que contienen un grado de dificultad añadida:

El primero de ellos, justo después del *checkpoint*, se deberá hacer uso del deslizamiento por las paredes y el *wall jump* para poder superar peligros como el láser y los pinchos afilados.

El segundo tramo será necesario impulsarse con la plataforma de salto para caer encima sobre un enemigo volador y así, poder llegar a la plataforma fija para poder continuar por el camino. Parece complicado, pero solo será necesario elegir el momento adecuado ya que el enemigo volador estará en movimiento.

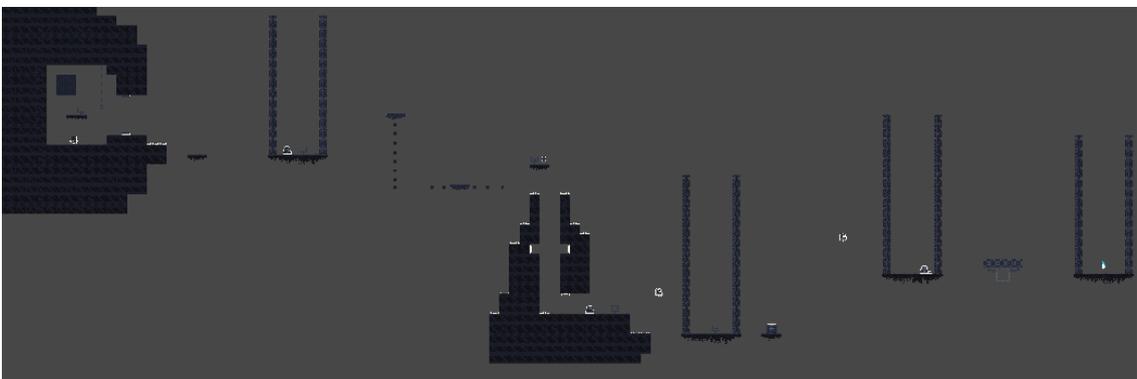


Ilustración 40: Nivel 6 de Galleto's Adventures

5.8 Nivel 7

En el último nivel, pese a no contener ningún elemento ni peligro nuevo, se podrán en práctica todas habilidades y mecánicas del personaje principal. De esta manera, nos se comprobará que el jugador haya comprendido y sepa utilizar de manera adecuada cada una de las habilidades y mecánicas del jugador entrenadas durante el transcurso de los niveles anteriores.

Es importante destacar el tramo de las plataformas de cinta seguido de un wall jump. Este tramo puede resultar bastante estresante a causa de la dificultad que presenta, ya que se debe realizar de manera seguida. Esto puede provocar que el jugador se quede atascado y que no pueda continuar el nivel. Solo se necesita paciencia para superarlo y llegar hasta el *checkpoint*. Una vez pasado por el *checkpoint* el resto del nivel no debería causar ningún tipo problema.

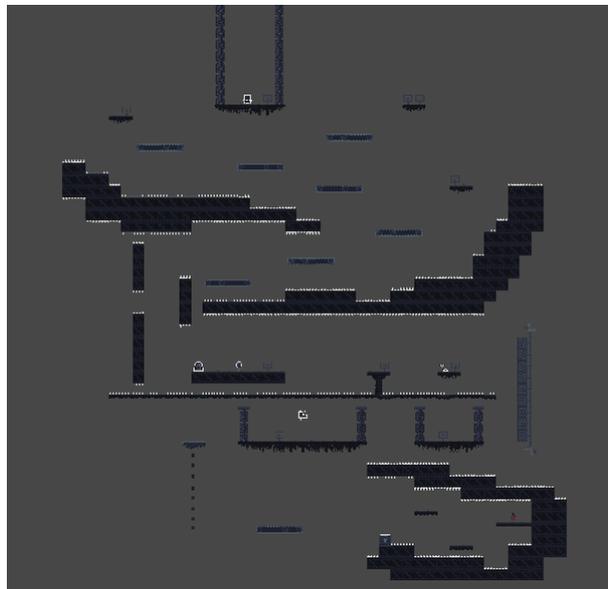


Ilustración 41: Nivel 7 de Galleto's Adventures

5.9 Mapa *Speed Run*

El mapa *Speed Run* forma parte de un modo de juego adicional, el modo *Speed Run*. Como se puede apreciar en la figura, se trata de un mapa no muy largo pero intenso donde el jugador deberá completar una especie de circuito en el mejor tiempo posible.

No habrá obstáculos que supongan la muerte pero podrá hacer uso de todas sus habilidades y mecánicas para poder lograr un mejor tiempo. Un mal uso de ellas podrá verse reflejado en el tiempo final.

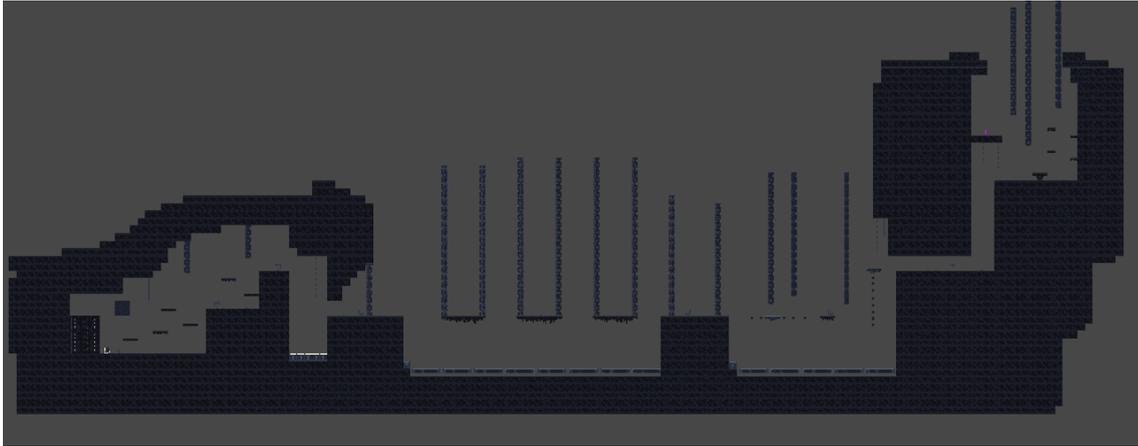


Ilustración 42: Nivel del modo *Speed Run* de Galleto's Adventures

7. Manual de usuarios

7.1 Requerimientos técnicos para jugar

Teniendo en cuenta que Galleto's Adventures se trata de un juego de plataformas y está destinado para plataformas tipo PC, el videojuego debería ejecutarse en la mayoría de ordenadores con un mínimo de prestaciones.

Durante el test de usuarios, la mayoría de pruebas se realizaron en ordenadores de alta gama con el sistema operativo de Windows con la versión 10 por lo que no se sabe a ciencia cierta cómo rinde el videojuego en otros ordenadores de diferentes gamas y sistemas operativos. Ante esta duda se indicarán en una tabla los requerimientos mínimos que se necesitan para ejecutar el editor de Unity para que de esta manera, los usuarios lo puedan comparar con las características de su ordenador. Según el Editor de Technologies, U (2021) el siguiente gráfico representaría los requisitos mínimos para ejecutar el editor de Unity.

Requerimientos mínimos	Windows	macOS	Linux (Support in Preview)
Versión sistema operativo	Windows 7 (SP1 +) y Windows 10, solo versiones de 64 bits.	High Sierra 10.13+	Ubuntu 20.04, Ubuntu 18.04 y CentOS 7
CPU	Arquitectura X64 con soporte de conjunto de instrucciones SSE2	Arquitectura X64 con soporte de conjunto de instrucciones	Arquitectura X64 con soporte de conjunto de instrucciones SSE2

		SSE2	
API de gráficos	GPU compatibles con DX10, DX11 y DX12	GPU Intel y AMD con capacidad de metal	GPU Nvidia y AMD compatibles con OpenGL 3.2+ o Vulkan.

7.2 Instrucciones del juego

7.2.1 Formas de iniciar el juego

Galleto's adventures ha sido destinado tanto para PC como para navegadores web. Por lo tanto habrá dos maneras de iniciar el videojuego.

La primera de ella será ejecutando el archivo ejecutable del juego (fichero .exe en el caso de Windows). Una vez se ejecute el ejecutable, el juego se iniciará y mostrará el menú principal.

La segunda forma de iniciar el juego será para aquellos usuarios que quieran jugarlo mediante el navegador web. Para ello se ha subido el videojuego a itch.io, una plataforma de distribución digital donde los desarrolladores independientes publican sus videojuegos.

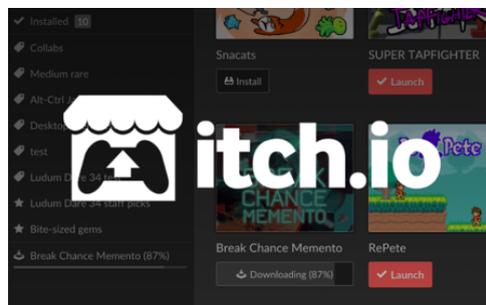


Ilustración 43: Logotipo itch.io

En el siguiente enlace se podrá jugar a Galleto's Adventures mediante el navegador: <https://panyu1512.itch.io/galletos-adventures>

7.2.2 Modos de juego

Galleto's Adventures ofrece dos tipos de modo de juego para que el jugador pueda disfrutar de dos modalidades de juego.

- **Modo Aventura:** modo que estará compuesto por un total de ocho niveles de menor a mayor dificultad. El jugador deberá de comprender y dominar todas las mecánicas para poder superar los niveles.



Ilustración 44: Captura del modo aventura

- **Modo SpeedRun:** modo donde el jugador deberá pasarse un nivel en el menor tiempo posible. El nivel siempre será el mismo y la mejor marca será guardada. Siempre que el jugador lo desee podrá entrar para mejorar y batir su tiempo.



Ilustración 45: Captura del modo *Speed Run*

7.2.3 Controles

- **Desplazamiento:** el personaje es capaz de desplazarse de izquierda a derecha. Será necesario hacer uso del desplazamiento para superar cada uno de los desafíos.
- **Salto:** el personaje es capaz de saltar estando quieto, en una pared o en movimiento.
- **Dash:** habilidad que permite desplazarse horizontalmente con una mayor velocidad para poder alcanzar o evitar obstáculos.

- **Deslizamiento:** capacidad del personaje para deslizarse por las paredes para sortear obstáculos o para poder escalar paredes con una mayor facilidad.
- **Interactuar con objetos:** el personaje principal será capaz de interactuar con palancas para habilitar plataformas de paso.
- **Menú de pausa:** el jugador puede pausar el juego cuando lo desee. Desde dicho menú se podrá reanudar la partida, acceder al menú principal del juego o salir por completo del juego.

A continuación se muestra una tabla con los controles y sus respectivas teclas:

Control	Teclado
Desplazamiento	Flechas de dirección izquierda y derecha o A y D
Salto	Espacio
Dash	E
Deslizamiento	Flechas de dirección izquierda y derecha o A y D sobre una pared.
Interactuar con objetos	F
Menú de pausa	Escape

7.2.3 Menús, escenas e interfaces

- **Menú principal:** acceso a la selección de niveles del modo principal, jugar al modo secundario (*speedRun*), controles, configuración y salir del juego.

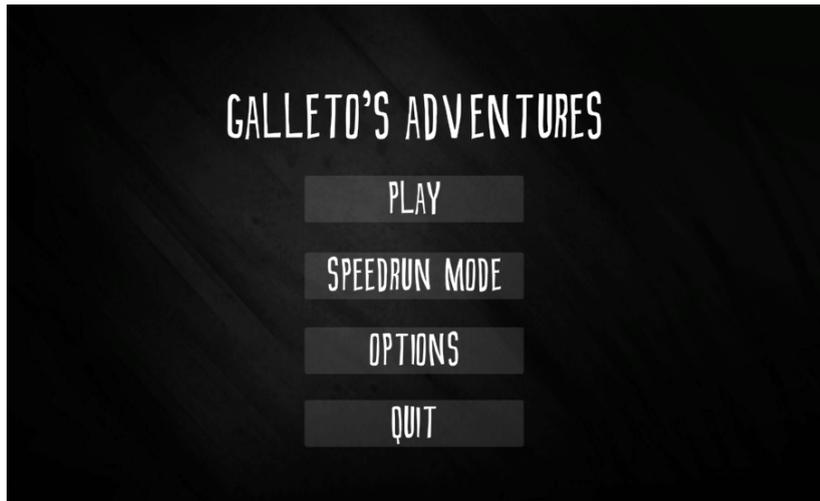


Ilustración 46: Captura del menú principal

- **Menú selección de nivel:** da acceso a todos los niveles del modo principal. En un principio están todos bloqueados menos el nivel introductorio. Se deberá superar cada nivel para desbloquear el siguiente. Una vez desbloqueados todos los niveles se podrá reiniciar mediante el botón de “RESET”. También se podrá acceder al menú principal



Ilustración 47: Captura del menú de selección de nivel

- **Menú de pausa:** el jugador cuando pausa el juego se le habilita este menú. En él podrá reanudar la partida, volver al menú principal del juego o salir del juego. Destacar que si el jugador vuelve al menú o sale de la partida no se guardará el estado de la partida, pero si los niveles desbloqueados.



Ilustración 48: Captura de la pantalla de pausa

- **Interfaz Inicio de nivel:** al inicio de cada nivel se mostrará un texto con el nivel en el que se encuentra.

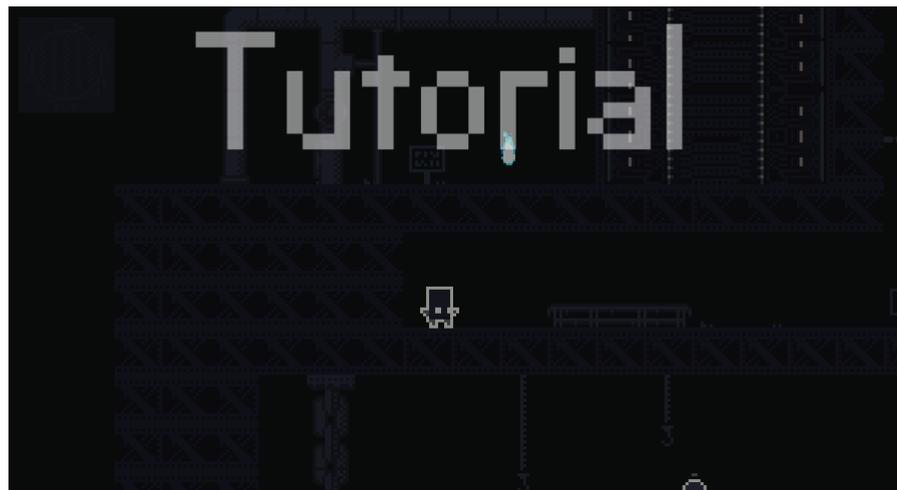


Ilustración 49: Captura de la interfaz de inicio de nivel

- **Interfaz Game Over:** cada vez que el personaje muera aparecerá el texto *GAME OVER*. Automáticamente después se reiniciará el nivel.



Ilustración 50: Captura de la interfaz de muerte

9. Conclusiones

Desde que tengo uso de razón me han gustado los videojuegos y uno de mis sueños era poder realizar algún día un videojuego. Elegí estudiar el grado de ingeniería informática en gran parte por este sueño, ya que me proporcionaba los conocimientos suficientes para poder desarrollar un juego.

A mediados del grado, cuando ya tenía conocimientos suficientes de programación, decidí empezar a desarrollar lo que sería mi primer juego. Desgraciadamente tuve que abandonarlo porque no encontraba tiempo para enfocarme en él. Al ver las temáticas del TFG y ver que una de ellas consistía en desarrollar un videojuego, no dudé ni un segundo en cogerla para poder finalmente conseguir mi sueño.

Pese a que en un principio tenía miedos ya que no tenía una experiencia con las herramientas de desarrollo, he de decir que me siento satisfecho con el resultado ya que los principales objetivos establecidos en la planificación inicial se han establecido sin ningún tipo de dificultad.

Aún estar satisfecho, no todo ha salido a la perfección. La planificación inicial fue bastante ambiciosa y fue necesaria adaptarla debido a los contratiempos que surgieron durante el desarrollo del proyecto. La mayoría de ellos por falta de tiempo.

Pese a los obstáculos encontrados durante el desarrollo del proyecto se ha podido conseguir un producto acabado y no solo eso, sino también, una experiencia personal que marcará un antes y un después en mi vida personal.

Le tengo mucho aprecio a Galleto's Adventures por lo que se espera que en un futuro se puedan añadir más implementaciones y mejoras al juego y además, poder desplegarlo en todas las plataformas posibles.

10. Glosario

Indie: abreviación del inglés del término "*independent*". Cuando se habla de desarrollo de videojuegos *indie* se refiere a aquellos proyectos desarrollados sin el apoyo financiero de una gran distribuidora. Normalmente los juegos *indies* están desarrollados por pequeños estudios o grupos de personas.

Diagrama de gantt: cronograma de un proyecto para representar el tiempo de dedicación para diferentes tareas o actividades a lo largo de un periodo de tiempo determinado.

Script: secuencia de código.

Steam: plataforma de distribución de videojuegos para PC.

Google Play Store: plataforma de distribución digital para dispositivos móviles. Generalmente para el sistema operativo Android.

itch.io: plataforma de distribución digital para juegos independientes.

2D: referencia aquellos videojuegos que solo utilizan dos dimensiones de las tres dimensiones posibles.

2.5D: referencia a aquellos videojuegos con proyecciones gráficas en 2D y técnicas que hacen que ciertas escenas parezcan ser de tres dimensiones cuando realmente no lo son.

3D: referencia aquellos videojuegos que solo utilizan tres dimensiones. Lo que permite que los juegos tengan libertad de movimiento en el espacio y no en un plano.

Efecto *parallax*: efecto donde las imágenes del fondo se mueven a una velocidad diferente que los objetos del mismo plano del protagonista

Engine: herramienta para el desarrollo de videojuegos. Facilita el proceso de creación.

C#: lenguaje de programación.

tilemaps: forma sencilla de hacer niveles para juegos 2D. Se utilizan *tiles* (“baldosas”) que son puestos en una grilla tantas veces como uno desee.

Release: conjunto de estados del progreso del proyecto.

Asset: conjunto de ítems que pueden ser utilizados en un proyecto o videojuego.

Asset store: tienda de Unity en la que se pueden encontrar elementos tanto gratuitos como de pago para el desarrollo del videojuego.

Git: sistema de control de versiones.

11. Bibliografía

Palou, N. (2020). *La crisis impulsa el sector del videojuego, pero frena la producción*. Economía Digital. Economía Digital. [Página Web] Disponible en: https://www.economiadigital.es/tecnologia/la-crisis-impulsa-la-industria-del-videojuego-pero-frena-la-produccion_20056101_102.html

Ramírez, P. (2021). *Ni FIFA ni Cyberpunk: los juegos «indie» triunfan más que las superproducciones*. La Información. [Página Web] Disponible en: <https://www.lainformacion.com/management/videojuegos-exito-indie-fall-guys-a-mong-us-rust/2826808/>

Gaming, N. (2020). *The evolution of platform games in 9 steps*. Red Bull. [Página Web] Disponible en: <https://www.redbull.com/in-en/evolution-of-platformers>

Powell, S. (2015). *Por qué aún nos gustan Mario Bros y otros videojuegos de plataformas*. BBC News Mundo. [Página Web] Disponible en: https://www.bbc.com/mundo/noticias/2015/08/150828_videojuegos_plataformas_tecnologia_ac

Editor Technologies, U. (2021). *Unity - Manual: System requirements for Unity 2021.2(alpha)*. Unity3d. [Página Web] Disponible en:

<https://docs.unity3d.com/2021.2/Documentation/Manual/system-requirements.html>

12. Anexos

12.1 Repositorio de GitHub

Enlace al repositorio del juego:

- <https://github.com/kike1512/game>

12.2 Release de la versión final del juego

Enlace al release de la versión final:

- <https://github.com/kike1512/game/releases/tag/1.2>

12.3 Enlace del juego en itch.io

Enlace del juego en itch.io:

- <https://panyu1512.itch.io/galletos-adventures>

12.3 Trailer

Enlace al trailer de Galleto's Adventures:

- <https://www.youtube.com/watch?v=AbPkmSJJX9o>