



Región de Murcia
Consejería de Educación, Formación y Empleo

 Colegio
Miralmonte
CENTRO CONCERTADO

AVENTURAS VERTIGINOSAS



DESARROLLO DE
APLICACIONES MULTIPLATAFORMA

JOSÉ MARÍA GARCÍA CELDRÁN
PEDRO MANUEL MÉNDEZ SAN LEANDRO
MÓNICA PIERNAS CARRILLO



ÍNDICE

ÍNDICE	2
1. Datos del proyecto y resumen	3
2. Introducción	3
3. Objetivos	4
4. Entorno del proyecto	5
4.1 Contexto	5
4.2 Justificación	5
4.3 Stakeholders	5
5. Alcance	6
5.1 Situación actual	6
5.2 Alcance y posibles obstáculos	6
6. Planificación temporal	7
6.1 Fases del proyecto	7
6.2 Planificación inicial	7
6.3 Punto de control	8
6.4 Planificación final	8
7. Leyes y normativas	9
8. Análisis	10
8.1 Especificación de los requisitos	10
8.1.2 No funcionales	11
9. Diseño	11
9.1 Arquitectura	11
9.2 Interfaz	12
9.3 Tecnología	12
10. Desarrollo	13
10.1 Estrategia de desarrollo	13
11. Pruebas	14
11.1 Pruebas de la aplicación	14
11.2 Usabilidad	15
12. Conclusiones	16
13. Bibliografía	16
14. Anexos	17
Anexo 1: Bocetos de Diseño y Conceptos Iniciales	17
JUGADOR	17
ENEMIGOS	17
ENTORNO	17



1. DATOS DEL PROYECTO Y RESUMEN

DATOS:

- **TÍTULO:** AVENTURAS VERTIGINOSAS.
- **AUTORES:** Pedro Manuel Méndez, Jose María García, Mónica Piernas.
- **WEB:** <https://aventuras-vertiginosas.vercel.app/>

RESUMEN:

"Aventuras Vertiginosas" es un juego en 2D tipo Endless Runner desarrollado en el marco del programa de Desarrollo de Aplicaciones Multiplataforma (DAM). Este juego, creado con gráficos tipo pixel art, invita a los jugadores a controlar una ardilla ágil que se desplaza a través de un bosque, esquivando obstáculos y avanzando sin cesar. El objetivo es lograr la mayor distancia posible mientras el terreno se vuelve cada vez más desafiante.

2. INTRODUCCIÓN

En un mundo donde la velocidad y la destreza son fundamentales, surge "Aventuras Vertiginosas", un emocionante juego en 2D que invita a los jugadores a sumergirse en una experiencia trepidante llena de desafíos y diversión. En este juego, los jugadores asumen el papel de una ardilla intrépida que se aventura a través de un bosque, saltando entre los árboles y esquivando obstáculos mientras avanza sin cesar. Con un estilo visual cautivador de pixel art y una jugabilidad adictiva, promete transportar a los jugadores a un mundo lleno de emoción y adrenalina.

Este proyecto representa un desafío emocionante para poner en práctica nuestras habilidades técnicas y creativas mientras colaboramos en el diseño, desarrollo y pruebas de un producto interactivo. A lo largo de esta documentación, se detalla el proceso de creación de "Aventuras Vertiginosas", desde la planificación inicial hasta la implementación final, destacando los logros alcanzados y las lecciones aprendidas en el camino.



3. OBJETIVOS

El objetivo principal de nuestro proyecto es desarrollar un juego en 2D utilizando Unity. A continuación, se detallan los objetivos específicos que se deben alcanzar durante la realización del proyecto:

Juego:

- Implementar un sistema de juego que permita a los jugadores controlar una ardilla que se desplaza a través de un bosque, saltando entre árboles y evitando obstáculos.
- Diseñar niveles progresivamente más desafiantes a medida que avanza el juego.
- Incorporar gráficos estilo pixel art para crear una estética visual atractiva y nostálgica.
- Integrar un contador de metros que registre la distancia recorrida por el jugador.

Roles:

- **Rol de jugador:**
 - Permitir a los jugadores controlar el movimiento de la ardilla mediante controles de teclado.
 - Proporcionar una experiencia de juego emocionante y desafiante.
 - Incentivar a los jugadores a competir por alcanzar distancias más largas y obtener puntuaciones más altas.
 - Implementar un sistema de registro de puntuaciones para que los jugadores puedan comparar sus logros con otros usuarios.
- **Rol de desarrollador:**
 - Gestionar el desarrollo del juego, incluyendo diseño, programación, arte y sonido.
 - Colaborar en equipo para garantizar un progreso eficiente y una calidad consistente del juego.
 - Realizar pruebas exhaustivas para identificar y corregir errores antes del lanzamiento del juego.



4. ENTORNO DEL PROYECTO

4.1 CONTEXTO

El proyecto se desarrolla como parte del programa de grado superior de Desarrollo de Aplicaciones Multiplataforma (DAM).

4.2 JUSTIFICACIÓN

Se busca aplicar los conocimientos adquiridos en el programa en un proyecto práctico y desafiante.

4.3 STAKEHOLDERS

Los stakeholders incluyen a los estudiantes, profesores evaluadores y potenciales usuarios del juego.



5. ALCANCE

5.1 SITUACIÓN ACTUAL

En el desarrollo, hemos avanzado en el diseño del fondo y del personaje utilizando la plataforma Piskel. Estos diseños están en proceso y formarán la base visual del juego.

La necesidad principal que aborda este proyecto es proporcionar a los jugadores una experiencia de juego emocionante y divertida que puedan disfrutar en su tiempo libre.

5.2 ALCANCE Y POSIBLES OBSTÁCULOS

En cuanto al alcance del proyecto, nos enfrentamos a varios desafíos y obstáculos que debemos superar para llevar el juego a su realización completa:

- **Integración de diseños de Piskel:** Aunque hemos realizado los diseños del fondo y del personaje en Piskel, la integración de estos diseños en Unity puede presentar desafíos técnicos que requerirán tiempo y esfuerzo para resolver.
- **Ajustes y refinamientos:** Es probable que necesitemos realizar ajustes y refinamientos en los diseños y mecánicas del juego a medida que avancemos en el desarrollo. Esto puede implicar cambios en el diseño de niveles, la jugabilidad y otros aspectos del juego.
- **Optimización del rendimiento:** A medida que agreguemos más elementos al juego, como animaciones y efectos visuales, es posible que necesitemos optimizar el rendimiento del juego para garantizar una experiencia fluida.

A pesar de estos posibles obstáculos, estamos comprometidos a superarlos y llevar "Aventuras Vertiginosas" a su máximo potencial, ofreciendo a los jugadores una experiencia de juego emocionante y satisfactoria.



6. PLANIFICACIÓN TEMPORAL

6.1 FASES DEL PROYECTO

1. **Inicio:** Nos dedicaremos a definir el alcance y los objetivos. Además, estableceremos un equipo de desarrollo, asignaremos roles y responsabilidades.
2. **Planificación:** Elaboramos un conjunto detallado de planes para guiar el desarrollo del juego. Esto incluirá la elaboración de un plan de proyecto que establezca las fechas de entrega y los recursos necesarios.
3. **Ejecución:** Nos centraremos en llevar a cabo el plan de trabajo establecido en la fase de planificación. Esto implica el desarrollo activo del juego, incluyendo la implementación de la mecánica de juego, la integración de los diseños de Piskel y la realización de pruebas periódicas para garantizar la calidad del juego en desarrollo.
4. **Seguimiento y control:** Nos aseguraremos de realizar un seguimiento continuo del progreso del proyecto y de controlar cualquier desviación con respecto al plan establecido. Esto incluirá la revisión regular de hitos y la realización de ajustes según sea necesario para mantener el proyecto en curso. También nos aseguraremos de gestionar eficazmente los cambios, los riesgos y los recursos durante todo el proceso de desarrollo.
5. **Cierre:** Esta última fase comprende todos procesos orientados a completar formalmente el proyecto y las obligaciones contractuales inherentes.

6.2 PLANIFICACIÓN INICIAL

Durante la fase de planificación inicial, nos centraremos en establecer las bases del proyecto y definir un plan de trabajo detallado. Las principales actividades incluirán:

1. **Definición del alcance:** Detallaremos las características clave del juego, incluyendo la mecánica de juego, los elementos visuales y sonoros, y los requisitos técnicos.
2. **Creación de un equipo de desarrollo:** Asignaremos roles y responsabilidades a los miembros del equipo, incluyendo desarrolladores, diseñadores gráficos y sonoros.
3. **Elaboración de un plan de proyecto:** Estableceremos fechas de entrega, hitos importantes y recursos necesarios para el desarrollo del juego.
4. **Identificación de riesgos:** Identificamos posibles riesgos y estableceremos estrategias para mitigarlos.



6.3 PUNTO DE CONTROL

Durante el punto de control del proyecto, evaluaremos el progreso realizado hasta el momento y realizaremos ajustes según sea necesario para mantener el proyecto en curso. Las principales actividades incluirán:

1. **Revisión del progreso:** Analizaremos el avance del desarrollo del juego en comparación con el plan establecido en la fase de planificación inicial.
2. **Identificación de desviaciones:** Identificamos cualquier desviación con respecto al plan y determinaremos las causas subyacentes.
3. **Ajuste de la planificación:** Realizaremos ajustes en el plan de trabajo según sea necesario para abordar cualquier desviación identificada y mantener el proyecto en curso.
4. **Gestión de riesgos:** Revisaremos y actualizaremos la evaluación de riesgos, y tomaremos medidas adicionales si es necesario para mitigar cualquier riesgo emergente.

6.4 PLANIFICACIÓN FINAL

Después de haber desarrollado el juego es necesario pasar una serie de pruebas para que sea estable y garantizar el correcto funcionamiento. Por lo tanto, esta etapa está destinada a la realización de plan de pruebas unitarias. Corrigiendo de esta forma los bugs que se pueden encontrar o mejoras que se puedan realizar para un mejor funcionamiento. Esta etapa es fundamental para garantizar que sea estable y que funcione correctamente.



7. LEYES Y NORMATIVAS

Es crucial cumplir con las leyes y normativas aplicables en el ámbito del desarrollo de videojuegos y tecnología en general. Algunas de las leyes y regulaciones relevantes a considerar incluyen:

- **Derechos de autor:** Respetar los derechos de autor y las licencias de cualquier contenido utilizado en el juego.
- **Protección de datos:** Garantizar la privacidad y seguridad de los datos personales de los usuarios del juego.
- **Seguridad en línea:** Implementar medidas de seguridad en línea para proteger la integridad del juego y la privacidad de los usuarios.
- **Contenido adecuado para la edad:** Asegurarnos de que no incluya contenido inapropiado para ciertas edades.
- **Publicidad y monetización:** Cumplir con las regulaciones de publicidad y monetización si se incluyen en el juego, asegurando transparencia y honestidad.
- **Accesibilidad:** Es importante diseñar el juego de manera que sea accesible para personas con discapacidades, cumpliendo con las regulaciones de accesibilidad y siguiendo las pautas de diseño inclusivo.

Al cumplir con estas leyes y normativas, podemos asegurar que sea un producto legal, ético y seguro para todos sus usuarios.



8. ANÁLISIS

8.1 ESPECIFICACIÓN DE LOS REQUISITOS

8.1.1 FUNCIONALES

Los requisitos funcionales describen las características y funciones que el juego debe tener para cumplir con sus objetivos. Estos incluyen:

1. Mecánica de juego:

- La ardilla debe poder moverse hacia adelante automáticamente y saltar de árbol en árbol.
- Los jugadores deben poder controlar los saltos de la ardilla presionando la tecla espaciadora.
- El juego debe contar con un contador de distancia que mide los metros recorridos.
- La dificultad del juego debe aumentar gradualmente a medida que el jugador avanza.

2. Interfaz de usuario:

- La pantalla principal debe mostrar el contador de metros y el nivel de dificultad actual.
- El juego debe incluir un menú principal con opciones para iniciar el juego, ver las instrucciones y acceder a la configuración.
- Debe haber una pantalla de final de juego que muestre la distancia total recorrida y permita reiniciar el juego o volver al menú principal.

3. Gráficos y sonido:

- El juego debe utilizar gráficos en pixel art para el fondo y el personaje de la ardilla.
- Debe haber música de fondo que se reproduzca durante el juego.



8.1.2 NO FUNCIONALES

Los requisitos no funcionales describen los atributos de calidad y las restricciones del sistema. Estos incluyen:

1. Rendimiento:

- El juego debe tener tiempos de carga mínimos para ofrecer una experiencia fluida.
- Debe mantener un framerate constante de al menos 30 FPS.

2. Usabilidad:

- La interfaz debe ser intuitiva y fácil de usar para jugadores de todas las edades.
- Las instrucciones del juego deben ser claras y accesibles desde el menú principal.

3. Escalabilidad:

- El juego debe ser fácilmente ampliable para agregar nuevas características, niveles y mejoras en el futuro.

4. Seguridad:

- El juego debe estar protegido contra trampas y manipulaciones que puedan alterar el contador de distancia o la dificultad.

5. Mantenibilidad:

- El código del juego debe ser claro, bien documentado y modular para facilitar futuras modificaciones y actualizaciones.

9. DISEÑO

9.1 ARQUITECTURA

La arquitectura del juego se basa en el patrón Modelo-Vista-Controlador (MVC) para separar las preocupaciones y facilitar la gestión del desarrollo. A continuación se describe cada componente:

- **Modelo:** Contiene la lógica del juego y los datos, como la distancia recorrida, el nivel de dificultad y el estado de la ardilla.
- **Vista:** Se encarga de la representación visual del juego, incluyendo los gráficos en pixel art y la interfaz de usuario.
- **Controlador:** Gestiona la interacción del jugador, como el control de los saltos y la actualización de la vista según el estado del modelo.

9.2 INTERFAZ

La interfaz de usuario de "Aventuras Vertiginosas" está diseñada para ser intuitiva y accesible:

- **Menú principal:** Incluye opciones para iniciar el juego, ver instrucciones y acceder a la configuración.

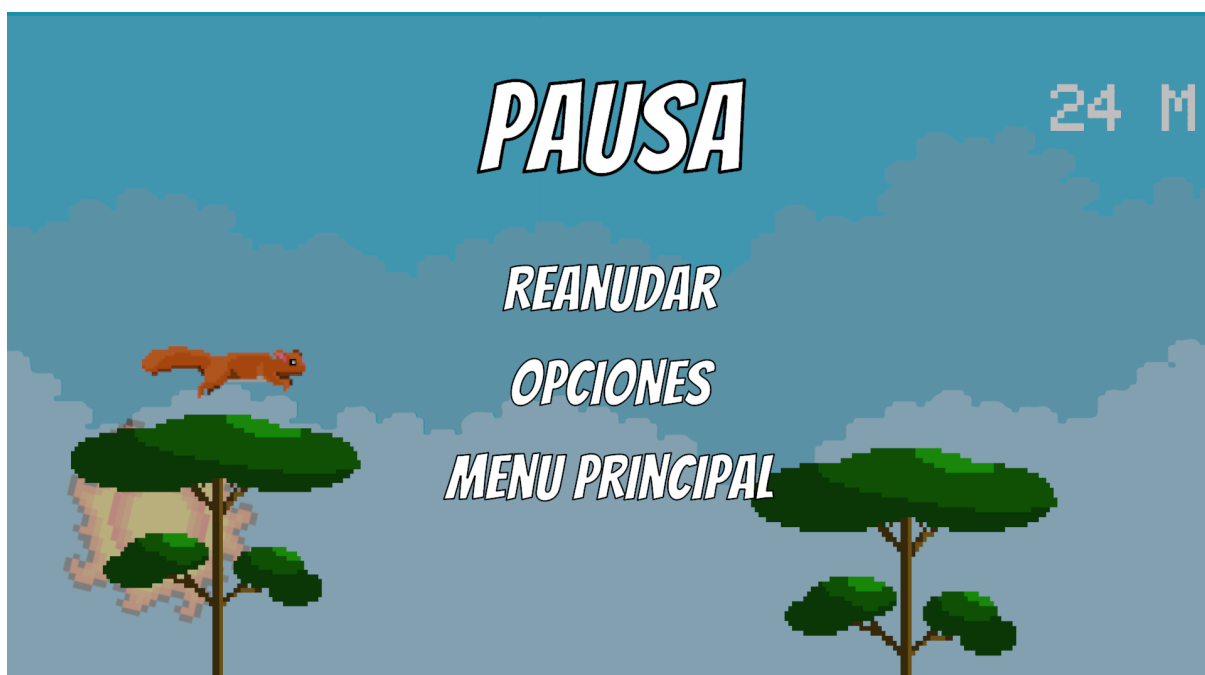


- **Pantalla de juego:** Muestra el contador de metros, además de los gráficos del juego.



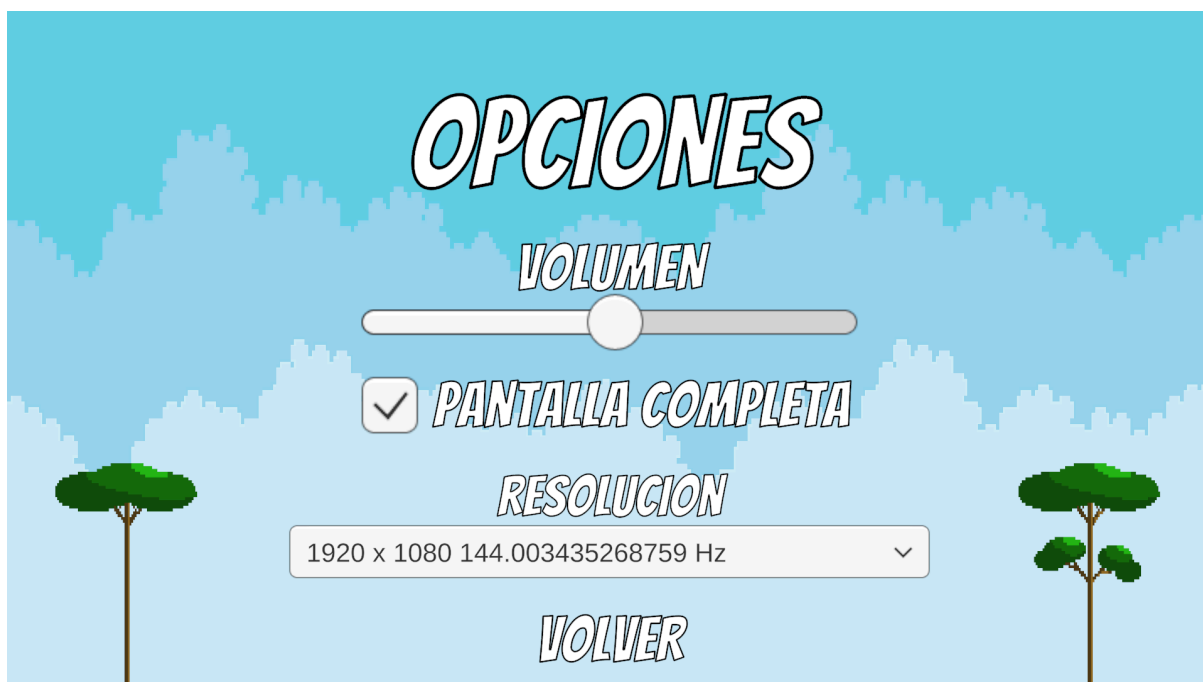


- **Pantalla de fin de juego y modo pausa:** Muestra la distancia total recorrida y opciones para reiniciar el juego o volver al menú principal.





- **Configuración:** Permite ajustar el volumen de la música y los efectos de sonido, y seleccionar el nivel de dificultad inicial.





9.3 TECNOLOGÍA

El desarrollo de "Aventuras Vertiginosas" utiliza las siguientes tecnologías:

- **Motor de juego:** Unity, por su capacidad para desarrollar juegos multiplataforma y sus herramientas de desarrollo robustas.



- **Lenguaje de programación:** C#, utilizado en Unity para escribir la lógica del juego.



- **Diseño de gráficos:** Piskel, para crear gráficos en pixel art.





- **Herramientas de sonido:** Reaper, para la creación y edición de efectos de sonido y música.



- **Control de versiones:** Google Drive, para facilitar la colaboración entre los miembros del equipo.



Este diseño asegura que sea un juego bien estructurado, seguro y agradable para los jugadores, facilitando su desarrollo.



10. DESARROLLO

10.1 ESTRATEGIA DE DESARROLLO

La estrategia de desarrollo se ha planteado para asegurar una creación eficiente y organizada del juego, asegurando la calidad y la coherencia en todas las etapas del desarrollo. A continuación se detallan los aspectos clave de esta estrategia:

1. Metodología Ágil:

- Las reuniones diarias se llevarán a cabo para revisar el progreso y abordar cualquier problema o impedimento.

2. Desglose de tareas:

- El proyecto se dividirá en tareas más pequeñas y manejables, asignadas a los miembros del equipo según sus habilidades.
- Las tareas se priorizarán en función de su impacto en la jugabilidad.

3. Desarrollo iterativo:

- Se adoptará un enfoque iterativo, permitiendo la implementación y prueba de características básicas primero, seguidas de mejoras y adiciones más complejas.
- Cada iteración incluirá etapas de desarrollo, prueba y revisión para asegurar la calidad y la funcionalidad.

4. Control de versiones:

- Se utilizará Google Drive para el control de versiones, asegurando que cualquier cambio pueda ser revertido si es necesario.

5. Pruebas continuas:

- Las pruebas serán una parte integral del desarrollo, con pruebas unitarias y de integración ejecutadas regularmente para identificar y corregir errores de manera temprana.
- Se realizan pruebas de jugabilidad para asegurar que la experiencia del usuario sea satisfactoria y que las mecánicas del juego funcionen como se espera.

6. Revisión y retroalimentación:

- Se planificaron sesiones de revisión al final de cada sprint para evaluar el progreso y recibir retroalimentación.
- La retroalimentación se utilizará para ajustar el plan de desarrollo y mejorar la calidad del juego.

7. Gestión de riesgos:

- Se identificarán y evaluarán los posibles riesgos y se desarrollarán estrategias para mitigarlos.
- Los riesgos se monitorizan continuamente y se tomarán acciones correctivas cuando sea necesario.



11. PRUEBAS

11.1 PRUEBAS DE LA APLICACIÓN

Para asegurar que "Aventuras Vertiginosas" funcione correctamente y ofrezca una experiencia de juego satisfactoria, se realizan diversas pruebas a lo largo del desarrollo. Estas pruebas incluyen:

1. Pruebas básicas de funcionalidad:

- Verificar que el juego se puede iniciar y jugar sin errores críticos.
- Probar que la ardilla salta correctamente y que el contador de metros funciona adecuadamente.

2. Pruebas de integración simples:

- Asegurarse de que los elementos del juego, como la ardilla, los árboles y el fondo, se muestren correctamente juntos.

3. Pruebas de rendimiento básicas:

- Jugar al juego en diferentes dispositivos para asegurar que no haya caídas significativas en el rendimiento.
- Verificar que el juego se ejecuta sin problemas en los dispositivos más comunes utilizados por el equipo.

4. Pruebas de jugabilidad:

- Jugar repetidamente para identificar errores y mejorar la mecánica del juego.
- Pedir a amigos o familiares que jueguen y den su opinión sobre la experiencia general.



11.2 USABILIDAD

La usabilidad es crucial para asegurar que el juego sea accesible y disfrutable para todos los jugadores. Las pruebas de usabilidad incluirán:

1. Pruebas con amigos y familiares:

- Invitar a amigos y familiares a jugar y observar cómo interactúan con el juego.
- Pedirles que comenten sobre cualquier dificultad que encuentren o aspectos que les resulten confusos.

2. Retroalimentación informal:

- Realizar preguntas simples después de que los jugadores prueben el juego, como "¿Fue fácil entender cómo jugar?" o "¿Qué partes encontraste más difíciles?"
- Usar esta retroalimentación para realizar mejoras en la interfaz y las mecánicas del juego.

3. Iteración básica de diseño:

- Basado en la retroalimentación recibida, hacer pequeños ajustes en el diseño y la funcionalidad del juego.
- Probar las nuevas versiones del juego para asegurar que las mejoras solucionen los problemas identificados.

4. Simplicidad en el diseño:

- Mantener la interfaz de usuario simple y clara para que sea fácil de entender para todos los jugadores.
- Asegurarse de que los controles sean intuitivos y respondan adecuadamente.



12. CONCLUSIONES

El desarrollo de "Aventuras Vertiginosas" ha sido una experiencia valiosa y enriquecedora para nosotros. A lo largo del proyecto, hemos enfrentado y superado diversos desafíos, lo que nos ha permitido adquirir nuevas habilidades y profundizar en nuestro conocimiento sobre el desarrollo de videojuegos. A continuación, se presentan las principales conclusiones de este proyecto:

1. Aprendizaje y crecimiento:

- Hemos adquirido un conocimiento profundo en el uso de Unity para la creación de juegos 2D, incluyendo la implementación de mecánicas de juego, diseño de niveles y optimización del rendimiento.

2. Colaboración y trabajo en equipo:

- La colaboración constante y la comunicación efectiva dentro del equipo fueron fundamentales para el éxito del proyecto. Aprendimos a trabajar juntos, a dividir tareas según las habilidades individuales y a resolver problemas de manera colaborativa.
- Las revisiones periódicas y la retroalimentación continua nos permitieron mejorar el juego iterativamente, asegurando una calidad óptima.

En resumen, no solo es el resultado tangible de nuestro esfuerzo y dedicación, sino también una representación de nuestro crecimiento como desarrolladores y profesionales. Este proyecto ha sentado una base sólida para futuros desarrollos y nos ha inspirado a seguir explorando el apasionante mundo del desarrollo de videojuegos.

13. BIBLIOGRAFÍA

1. Tutoriales y guías sobre Unity:

- BravePixelG: <https://www.youtube.com/BravePixelG>
 - [Cómo crear enemigos patrullen en una plataforma detectando objetos en Unity.](#)
 - [Cómo crear un juego infinito en Unity. \(Generar automáticamente un nivel en Unity\)](#)
 - [Cómo crear un fondo con animación en Unity \(Efecto Parallax en Unity\)](#)
 - [Cómo crear un salto regulable en Unity](#)
 - [Cómo crear un menú de opciones en Unity](#)

- Don Pachi: <https://www.youtube.com/@DonPachi>
 - [Unity - Como crear un menu de opciones | Parte 1](#)
 - [Unity - Como crear un menu de opciones | Volumen](#)
 - [Unity - Como crear un menu de opciones | Pantalla Com...](#)
 - [Unity - Como cambiar la resolucion](#)
- SuperJustin5000: <https://www.youtube.com/@superjustin5000>
 - [Speed and Parallax BG | MAKE AN INFINITE RUNNER game like CANABALT #2 | Unity How to Tutorial - YouTube](#)

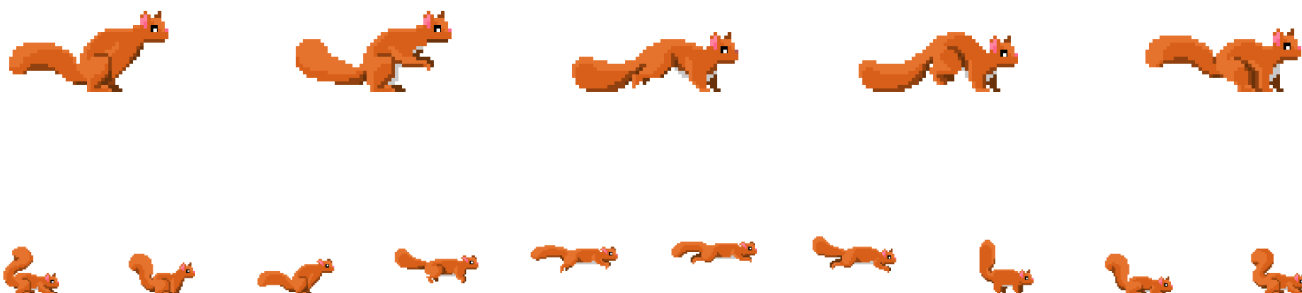
2. Piskel:

- Piskel. (n.d.). Free online sprite editor. Recuperado de <https://www.piskelapp.com/>

14. ANEXOS

Anexo 1: Bocetos de Diseño y Conceptos Iniciales

JUGADOR



ENEMIGOS



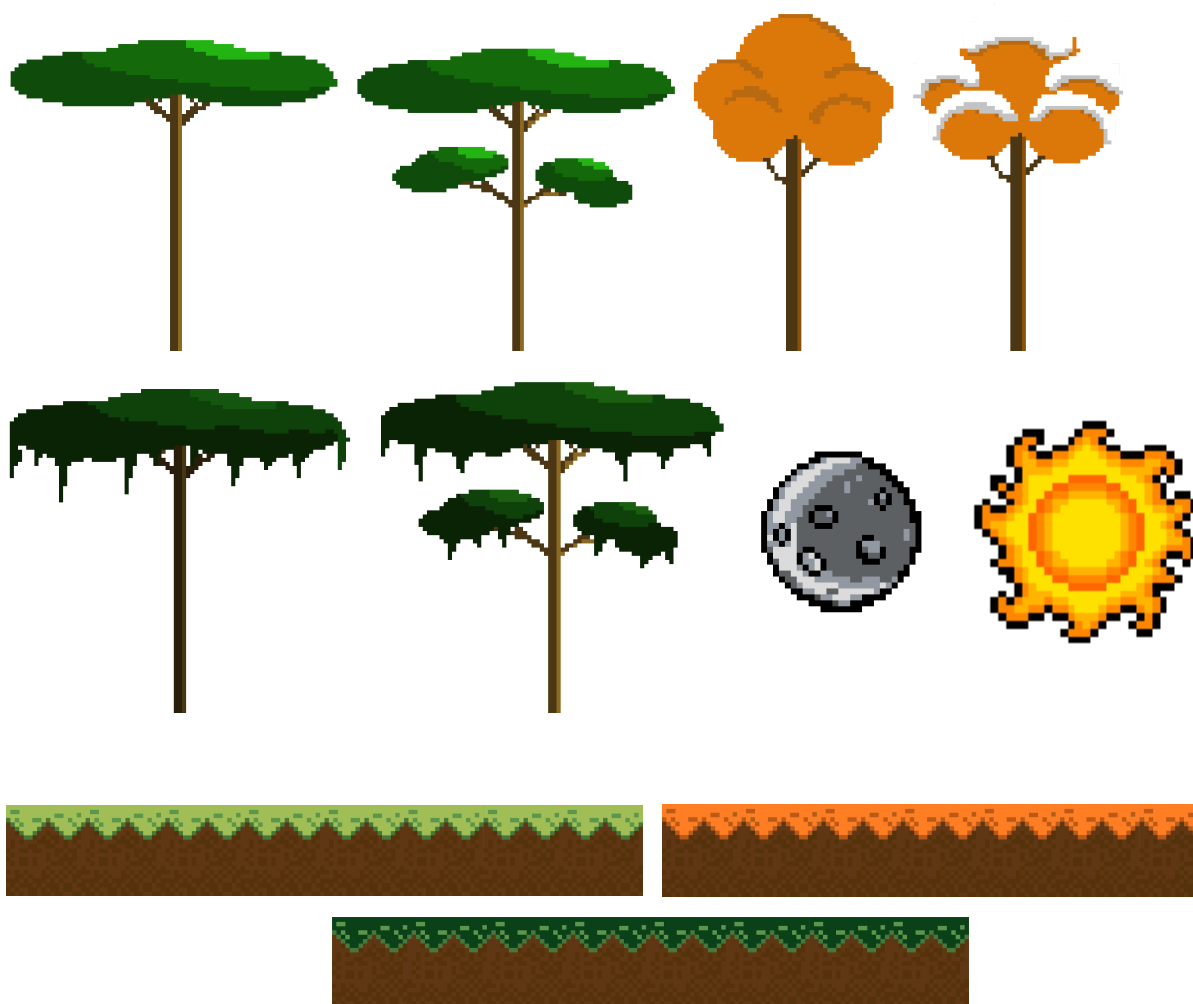


Región de Murcia
Consejería de Educación, Formación y Empleo





ENTORNO



FONDO

