

Documentación del Trabajo de Fin de Grado del Curso 2023-2024 en Desarrollo de Aplicaciones Multiplataforma SimplePlatformer_TFG

Alumno: Álvaro Márquez Cazorla.

Centro escolar: IES Al-Ándalus, Almería.

Índice

Descripción e Información.....	3
Motivación.....	3
Objetivos del Proyecto.....	3
Módulos utilizados.....	4

Descripción e Información

SimplePlatformer_TFG es un proyecto de investigación y desarrollo con el que busco averiguar y aprender a hacer diversas pero comunes mecánicas de videojuegos.

Es un juego simple, con saltos, obstáculos y objetos que romper o mover a nuestro antojo en el que debemos llegar a la meta haciendo uso de las distintas habilidades a nuestra disposición.

Como tal no es un juego completo, pero sí uno en el que probé a implementar algunas de las mecánicas más frecuentes en esta clase de productos.

Motivación

La motivación de este proyecto es algo con muchas fuentes, tanto profesional como personal.

Siempre he querido saber cómo desarrollar videojuegos, ya que les veo un fuerte potencial como medio de entretenimiento y como posible herramienta de contar historias (tanto simples como complejas).

Todo esto con el fin de tener una base con la que seguir investigando y mejorando mis habilidades de diseño y desarrollo orientados específicamente en la creación de más videojuegos.

Por algo se tiene que empezar.

Objetivos del Proyecto

El objetivo es bastante simple: Tener una versión de prueba de un videojuego en el que probar distintas mecánicas y donde intentar “tirar de distintas palancas” para conseguir un movimiento satisfactorio.

Obviamente todo debe ser funcional, y debe mantener la forma a que alguien pueda hacer algo “mal”, el juego debe estar a la altura y no romperse.

Módulos utilizados

Primer año:

- Sistemas Informáticos:
 - Instalación y configuración de Windows 11 Pro
 - Instalación y configuración de Unity Engine, versión 2022.3.24f1
 - Instalación y configuración de Visual Studio
 - Instalación y configuración de GitHub Desktop
- Programación:
 - Programado en lenguaje C-Sharp (también nos enseñaron a adaptarnos a terreno desconocido)
- Entornos de Desarrollo:
 - Documentación del desarrollo, tanto en forma de diario como de resumen de objetivos
 - Documentación interna del código, comentarios varios indicativos del funcionamiento de distintas partes
- Formación y Orientación Laboral:
 - Investigación de cómo me afectaría como desarrollador usar ciertas herramientas como desarrollador particular










Segundo año:

- Desarrollo de Interfaces:
 - Análisis de diseño y distribución de elementos gráficos interactivables
 - Elección de color y tamaño de diversos elementos para ser fácilmente legibles
- Programación Multimedia y Dispositivos Móviles:
 - Empleo de archivos multimedia (imágenes, sonidos, música) adaptado al videojuego (música de fondo, efectos de sonido, animaciones...)
- Programación de Servicios y Procesos:
 - Uso de Corrutinas (hasta cierto punto, compatible con PMDM) y scripts interconectados
- Empresa e Iniciativa Emprendedora:
 - Estudio de la viabilidad de los videojuegos en el mercado
- Acceso a Datos:
 - Consulta de datos entre scripts e inserción de datos hardcodeados y otorgados externamente en el editor.

Investigación

Al ser este un proyecto de investigación, he hecho un fuerte uso de foros, tutoriales y similares para la creación de mi concepto de juego.

En general, puedo atribuir la ayuda que recibí a las siguientes páginas:

- StackOverflow: [Stack Overflow](#)
- Foros de Unity: [Unity Forum](#)
- Discusiones de Unity: [Unity Discussions](#)
- Tienda de Assets de Unity: [Unity Asset Store](#)
- Youtube: <https://youtube.com/>
 - Y en este caso, especialmente a:
 -  2D Movement [Rigidbody vs Transform] Mastery Tutorial Unity (...)
 -  Unity 2D Platformer for Complete Beginners - #1 PLAYER MOV...
(cierto es que tiene una lista de reproducción con bastante ayuda, pero solo hice uso de unos pocos vídeos de la misma, quise crear mecánicas por mi cuenta para aplicar lo aprendido)
 -  Ultimate 2D Platformer Controller in Unity (source code provided)
 -  6 Minute PAUSE MENU Unity Tutorial
 -  How to add multiple sound effects in Unity (3 MINUTES) - Unity ...
aunque este vídeo fue una completa pérdida de tiempo, no me sirvió en absoluto.
 -  Create great GAME OVER screen in Unity UI - Unity tutorial
 -  How To Add Sound Effects the RIGHT Way | Unity Tutorial aunque solo usara una de las técnicas que enseña, ya que para mis intereses con eso me bastaba.
 -  5 tips for better platformer controls
 -  Fix Rigidbody Jitter in One Click - Unity Quick Tips este fue especialmente útil para hacer fluido al más importante elemento del juego: el jugador.
 - Y canales como
 - [Masahiro Sakurai on Creating Games - YouTube](#)
 - [Guinxu - YouTube](#)
 - [Alva Majo - YouTube](#)
 - [Game Maker's Toolkit - YouTube](#)
 - [Leyendas & Videojuegos - YouTube](#)

Y mucha, mucha, MUCHA prueba y error.

Análisis

El **público potencial** de mi juego realmente podría ser cualquier persona, pero con lo que llevo hecho podrían ser niños de 7 años en adelante.

Especifico una fecha tan concreta por el reglamento europeo de calificación por edades “Pegi”, omnipresente desde comienzos de los 2000s en Europa junto a sus equivalentes en otras partes del mundo.

Se encargan de calificar adecuadamente cada videojuego que sale al mercado.

Para más información: [PEGI.info](https://www.pegi.info)

Los **requisitos** son simples: Necesita tener un sistema operativo reciente (como Windows 10 o Windows 11) en esencialmente cualquier máquina que los soporte.

No he hecho un programa exigente, pero sí que lo he construido solo con Windows en mente, la plataforma estrella de los juegos en ordenador.

No descarto en un futuro adaptar el juego a más plataformas.

Planificación

Primero hay que establecer unas bases:

Plataformas con colisiones y un personaje móvil.

Según se vaya terminando todo eso, se pueden ir añadiendo más mecánicas según se terminen las anteriores.

Cálculo de Horas de Trabajo

Incluso fuera de las horas activas de desarrollo, hay una planificación previa a cada sesión. Calculo que la suma de todo me habrá llevado unas **80 horas**.

Tendré que planificar bien las horas entre la formación en el centro de trabajo, mis deberes personales y este mismo proyecto.

Entre unas cosas y otras, he calculado esas 80 horas como las involucradas directa o indirectamente al proyecto.

Desarrollo del Código Fuente

Este apartado puede verse mejor reflejado en el Diario de Desarrollo, “hermanado” a este documento, donde puede verse día a día cómo fui elaborando cada parte, paso a paso, con capturas de cada avance y decisión.

Control de Errores

Los errores encontrados en la elaboración de este proyecto fueron creciendo según me alejaba de los tutoriales y procuraba hacer las cosas por mi cuenta.

Destacables (y todos arreglados):

- Jugador cayéndose al suelo por volcarse al no tener su eje de rotación fijado como estático.
- Projectiles detonando según eran disparados.
- Projectiles quedándose en el sitio una vez disparados.
- Enemigos no moviéndose a pesar de aparentemente estar bien programados.
- Colisiones no funcionando como se esperaba.
- Objetos no comportándose como debían (no desaparecen, no dan el efecto deseado, etc).
- Segundas puertas lógicas de un código haciendo lo mismo por despiste.
- Archivos de audio no funcionando como debían por mala implementación de los componentes en cuestión.
- Errores no marcados por el editor de código, pero sí por el de Unity.
- Unity marcando error en cosas que ni existen.
- Triggers debiendo usar un código ineficaz para funcionar debido a la inexperiencia del desarrollador.
- Cambios de escena disparándose demasiado pronto antes de completarse un especificado tiempo de detenimiento del código. (Solventado refactorizando esa parte del código).
- Interfaces no deseadas permanentemente activadas (solventado al espabilarse el desarrollador)
- Y muchos más que probablemente no recuerdo.

Normas de Usabilidad aplicadas

Este proyecto, lógicamente, hace uso de las interfaces.

Y se han usado conceptos de usabilidad tales como:

- Relación con la realidad:
 - Lenguaje claro, esperable para las funciones indicadas.
- Visibilidad:
 - Uso de colores y tamaños para discernir la importancia y funcionamiento de cada componente.
 - Distinción de elementos detrás.
- Reconocimiento:
 - Los elementos que esperarías de una forma, hechos así (Vida del jugador mostrada como lista de corazones).
- Consistencia y estándares:
 - La distribución de los componentes similares es consistente.

Demostración de las interfaces:

Menú Principal



Juego en curso



Nivel completado



Pantalla de “Trabajo en Curso y en Camino”:



Pruebas

Las pruebas de funcionalidad del proyecto se han hecho de una manera más bien rudimentaria, probando valores antes de ejecutar el juego en el editor de Unity, otras cambiándolo sobre la marcha para distinguir qué me convencía más (valores de velocidad, peso del jugador, gravedad, volumen de los efectos de sonido y la música...).

Al final hay que tener en cuenta que los videojuegos son algo que se deben sentir por uno mismo, no es lo mismo ver a alguien jugar que jugarlo tú.

Es por eso que es tan importante el concepto de kinestética, cuyos resultados de su buena realización se muestran difíciles de reflejar en texto o incluso vídeo y audio.

Ahora bien, por supuesto que han habido pruebas de picar código, buscar errores, montar y desmontar cosas hasta dar con la clave...

Lo normal en cualquier desarrollo de programación.

Pruebas de Usuarios

Lamentablemente, no he tenido oportunidad de que otra persona pruebe en sus propias manos el juego, pero he recibido opiniones y sugerencias de forma verbal.

Conclusiones Finales

Atreverme a explorar cómo es el desarrollo en un campo tan extremadamente inmenso y complejo como es el de los videojuegos ha demostrado ser difícil y muy duro. Lo hacía de antes por escuchar historias de periodistas y gente de la industria, pero admiro incluso más ahora todo lo que conlleva hacer un videojuego funcional

Ya ni digo bueno, funcional.

Todos los distintos elementos con sus variables, sus interacciones, sus maneras de romperse y tratarse adecuadamente.

Esto es duro, pero siendo honestos: Me ha resultado gratificante cada vez que completaba un pequeño reto.

Y creo que esto me ha dado más ganas de seguir adelante en el Curso de Especialización en Videojuegos.

Seguiré formándome, oficialmente y en mis ratos libres.

Diría que he cumplido con mis objetivos principales, que eran tener un personaje interesante de controlar con unas mecánicas básicas.

Pero esto está muy lejos de ser mi producto ideal, aún tengo muchísimo que aprender y hay cosas que me habría gustado hacer mejor, como:

- Las animaciones del personaje
- El sistema de sonido
- Los distintos menús
- Detalles concretos del movimiento del personaje
- La funcionalidad interna de esencialmente todos los objetos, para darme más flexibilidad al retocar cosas
- Cómo se comporta la música de fondo al morir
- Me habría gustado poder implementar un sistema de checkpoints
- Me habría gustado poder dar la opción de cambiar el idioma del juego
- Dar opción de poder jugar en ventana, con y sin bordes
- Tener la opción de cambiar el volumen de los distintos elementos
- Y muchas más cosas




Por lo general, me habría gustado hacer un juego más completo, pero las responsabilidades y el tiempo han dado para lo que se ha conseguido.

Lo bueno es que siempre tengo la oportunidad de seguir aprendiendo y mejorar mis habilidades.

Siempre avanzando.

Bibliografía

Todo lo indicado en el apartado de [Investigación](#) y:

- [cobalt.tools](#) como extracción de melodías para el sonido de fondo de cada pantalla, tales como:
 -  Jeremy Blake - Powerup! 🎵 NO COPYRIGHT 8-bit Music
 -  A Lonely Cherry Tree 🌸
 -  Begin Your Journey 🗺️

Agradecimientos Especiales

Luna Gris Adán Ruiz

Pablo Ramos Blánquez

Anaís Márquez Meléndez

Alejandro José Rodríguez Martín

María García González

A mi tía

A mis padres

Lolo el Remolino

Naia la Revoltosa

Y a ustedes, por formarme.

Y ya está.