

# PLATFORM JUMPING



DESARROLLO DE UN VIDEOJUEGO 2D EN GODOT ENGINE 4

JOSEP BARRERA & MICHAEL CASTILLO  
CFGM SMX2A | INSTITUT PUIG CASTELLAR

# ÍNDICE

1. Introducción
2. Objetivos
3. Tecnologías propuestas
4. Tecnologías utilizadas
5. Metodología
6. Arquitectura del proyecto
7. Conclusión



# INTRODUCCIÓN

## SECTOR DEL VIDEOJUEGO

- Industria enfocada al desarrollo 3D
- Juegos cada vez más exigentes técnicamente

## GÉNERO

- Videojuego de plataformas y acción 2D
- Inspiración en juegos clásicos arcade

## RETO

- Pasar de usuarios a desarrolladores
- Aprender Godot Engine y GDScript

## FILOSOFÍA

- Crear un juego ligero y accesible
- Experiencia fluida y rápida

# OBJETIVOS

## OBJETIVO GENERAL

- Desarrollar un videojuego de plataformas funcional
- Implementar movimiento, salto y ataque
- Gestionar el sistema de salud
- Crear enemigos con comportamiento autónomo
- Garantizar estabilidad y jugabilidad

## OBJETIVOS ESPECÍFICOS

### Técnicos

- Movimiento fluido y combate
- HUD y GUI

### Diseño

- Niveles progresivos
- Estilo pixel art

### Mecánicas

- Colisiones precisas
- Movimiento de enemigos

### Gestión y web

- Git/GitHub
- Desarrollo con HTML y CSS
- Botón para jugar al videojuego

### Aprendizaje

- Dominio de GitHub
- Resolución de conflictos
- Desarrollo en equipo

# TECNOLOGÍAS PROPUESTAS



## UNITY

- Motor muy utilizado en la industria
- Gran cantidad de documentación y recursos
- Menos intuitivo para proyectos 2D



## UNREAL ENGINE

- Alto rendimiento gráfico
- Enfocado principalmente al 3D
- Problemas de fluidez y rendimiento



## ROBLOX STUDIO

- Muy fácil de usar
- Compatible con muchos dispositivos
- Entorno cerrado



## GODOT ENGINE

- Motor gratuito y de código abierto
- Muy ligero
- Sistema basado en nodos y escenas

# HERRAMIENTAS UTILIZADAS

## Motor gráfico

- Godot Engine

## Editor de código

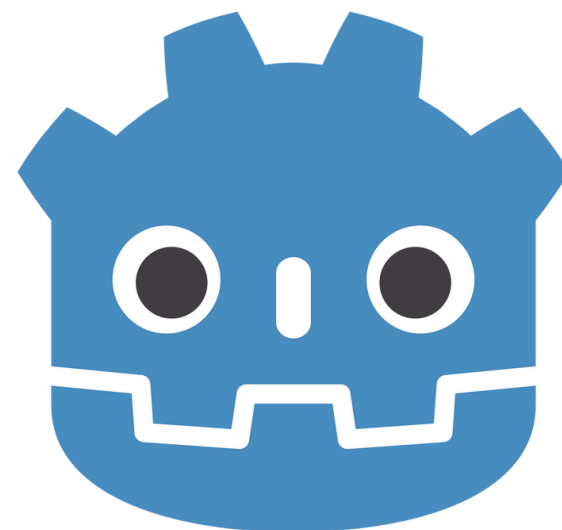
- Visual Studio Code

## Lenguaje de programación

- HTML/CSS
- GDScript

## Control de versiones

- Git i Github

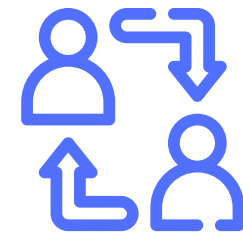


# METODOLOGÍA



## PLANIFICACIÓN Y ORGANIZACIÓN

- Forma de trabajo flexible y organizada
- Planificación y revisión constante
- Reparto de tareas
- Colaboración continua entre los miembros



## DESARROLLO Y APRENDIZAJE

- Uso de GitHub para gestionar el proyecto
- Investigación
- Aprendizaje mediante tutoriales
- Ayuda de la IA

# ARQUITECTURA DEL PROYECTO



## JUGADOR

- Movimiento y salto
- Combate
- Sistema de vida
- Animaciones
- Sonidos



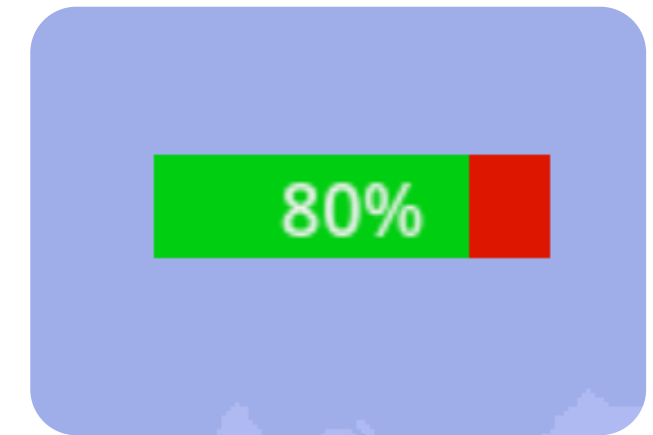
## MAPA / NIVELES

- Diseño de niveles mediante TileMaps con distintas capas
- Checkpoints
- Partículas
- Cambio dinámico de niveles



## ENEMIGOS

- Patrullaje por el mapa
- Detección de bordes
- Ataque con hitbox



## INTERFAZ

- Barra de vida tanto del jugador como del enemigo
- Menú de pausa
- Sistema de derrota

# VIDEO DE MUESTRA DEL PROYECTO



# CONCLUSIÓN

## RESULTADOS OBTENIDOS

- Juego funcional y estable
- Núcleo principal completado
- Sistema de combate y animaciones implementado
- Plataforma web promocional desarrollada

## APRENDIZAJE

- Godot 4 y GDScript
- GitHub y control de versiones
- Resolución de conflictos y merges
- Trabajo colaborativo y comunicación

## FUTURO

- Más niveles y escenarios
- Música y efectos de sonido
- Diferentes personajes y enemigos
- Interacción con el entorno
- Distintos ataques y habilidades
- Sistema de loot e inventario
- Jefes finales
- Compatibilidad con móviles

# MUCHAS GRACIAS

Josep Felipe Barrera Callupe  
&  
Michael Alejandro Castillo Girón

[https://josepb80.github.io/platfrom\\_jumping/Web/index.html](https://josepb80.github.io/platfrom_jumping/Web/index.html)