Juego FlappyBird

Desarrollo de un Juego Interactivo

Presentation - 2025

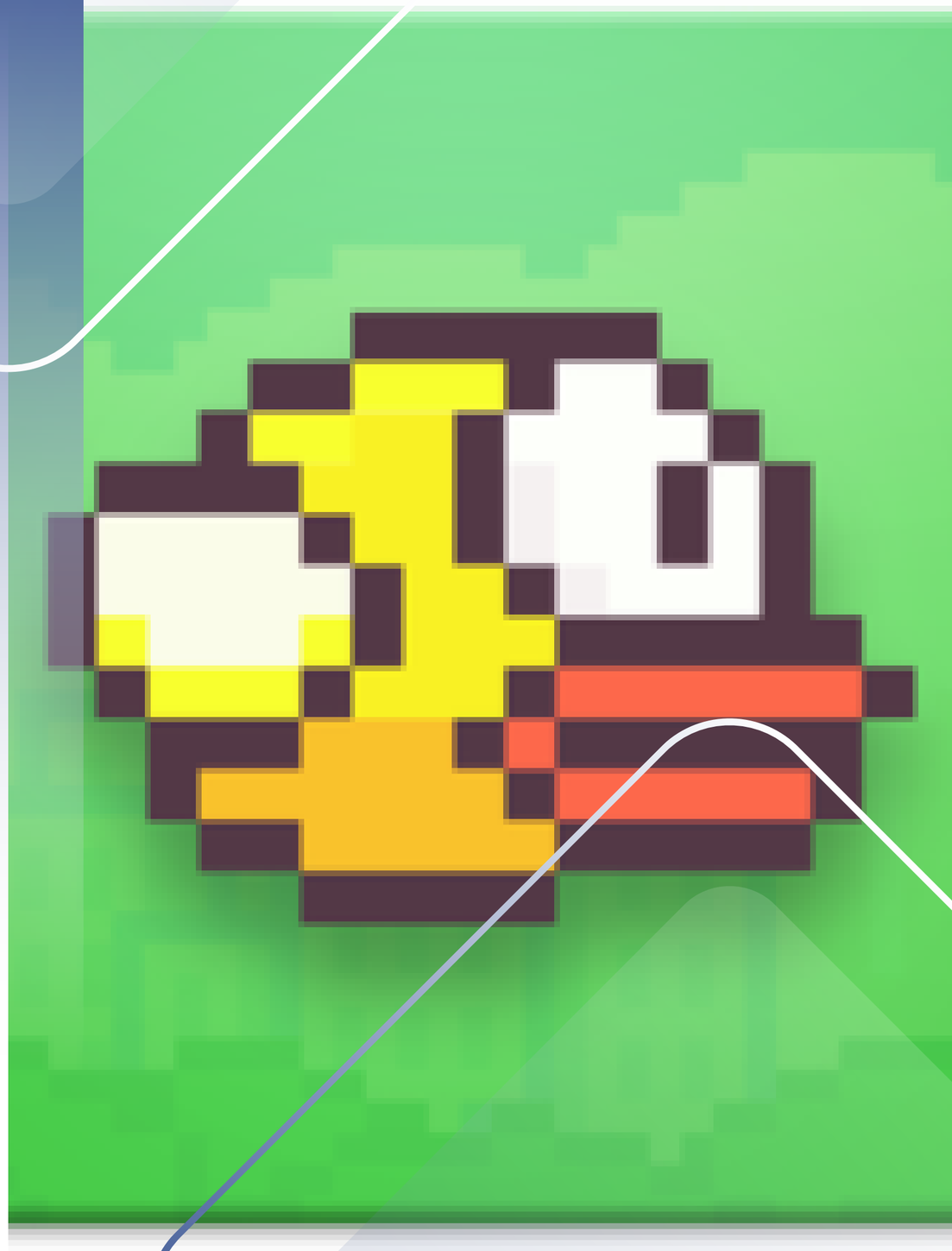
Ricki Maik Surco Aquise
Manuel Lazarte Infa
Enrique Mamani

Qué es Flappy Bird

Es un juego interactivo donde controlas un pájaro que debe pasar entre obstáculos, sumando puntos con cada éxito. Inspirado en el original, pero adaptado y mejorado.

Objetivo

Aprender desarrollo de juegos simples con programación y crear una experiencia divertida



Tecnologías utilizadas

Corona SDK / Solar2D

Es una plataforma de desarrollo de código abierto ideal para crear aplicaciones móviles y juegos 2D. Permite usar código en Lua para manejar gráficos, física y animaciones

Lenguaje: Lua.

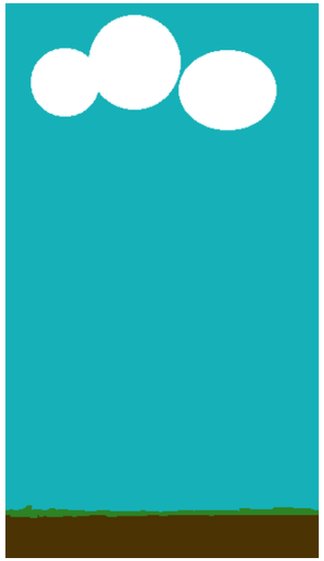
Un lenguaje sencillo y eficiente, perfecto para proyectos ligeros como juegos móviles.

Recursos adicionales

Me guié de guías y tutoriales en línea para complementar mi aprendizaje y adaptar el código a las necesidades del proyecto.

Elementos principales

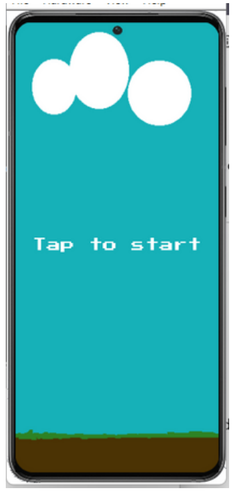
- Fondo: Escenario sencillo que da la sensación de estar volando.
- Personaje principal: Un pájaro que el jugador controla.
- Obstáculos: Tuberías que aparecen de manera aleatoria.
- Interfaz: Contadores de puntajes y botones básicos.



Extras

El diseño se centró en ser minimalista pero visualmente atractivo, asegurando que la experiencia de juego sea fluida y entretenida.

Estructura del juego



Inicio

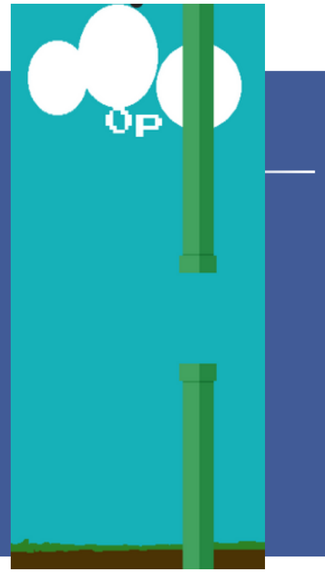
El juego comienza automáticamente al ingresar, con el pájaro listo para ser controlado.

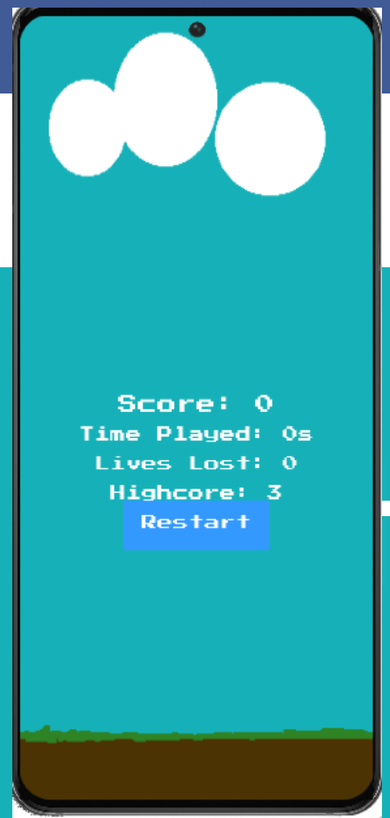
Mecánica

El jugador da toques táctiles en la pantalla para que el pájaro vuele y esquive los obstáculos. Cada tubería pasada suma puntos.

Extras

- Contador de segundos (tiempo).
- Registro de puntajes (ranking).
- Botón para reiniciar el juego.





Layout

W, _H, _CX, _CY: Valores obtenidos de relayout que representan el ancho, alto, y las coordenadas centrales de la pantalla.

Scene

Se crea una nueva escena con composer.newScene(), que gestionará el ciclo de vida del juego.

Grupos

grpMain, grpWorld, grpHud: Grupos visuales que organizan los elementos de la escena.

- grpMain: Grupo principal que contiene todos los demás.
- grpWorld: Contiene elementos del juego, como el pájaro y las tuberías.
- grpHud: Contiene elementos de la interfaz de usuario, como el marcador.

Variables

- pipes: Tabla que almacena las tuberías en pantalla.
- backgrounds: Tabla que contiene los fondos animados.
- canAddPipe: Controla cuándo se debe generar una nueva tubería.
- hasStarted: Indica si el juego ya comenzó.
- score: Puntaje actual.
- bird: Objeto que representa al pájaro en el juego.
- lblScore: Texto que muestra el puntaje en pantalla.
- restartButton: Botón para reiniciar el juego

```lua
-- -----------------------------------------------------------------------------------
-- Establecer variables
-- -----------------------------------------------------------------------------------

-- Layout
local _W, _H, _CX, _CY = relayout._W, relayout._H, relayout._CX, relayout._CY

-- Scene
local scene = composer.newScene()

-- Grupos
local grpMain
local grpWorld
local grpHud

-- Sonidos

local sndFlap = audio.loadStream("flap.wav")
local sndCrash = audio.loadStream("crash.wav")
local sndScore = audio.loadStream("score.mp3")

-- Variables

local pipes = {}
local backgrounds = {}
local canAddPipe = 0
local hasStarted = false
local score = 0
local bird
local lblScore
local restartButton


local score = 0
local timePlayed = 0
local livesLost = 0
```

```lua
-- ------------------------------------------------------------------
-- Establecer variables principales
-- ------------------------------------------------------------------

-- Variables de juego
local pipes = {}           -- Almacena las tuberías en pantalla
local canAddPipe = 0       -- Controla cuándo generar nuevas tuberías
local hasStarted = false   -- Verifica si el juego ha comenzado
local score = 0            -- Lleva el puntaje actual del jugador

-- Sonidos
local sndFlap = audio.loadStream("flap.wav") -- Sonido al aletear
local sndCrash = audio.loadStream("crash.wav") -- Sonido al colisionar
local sndScore = audio.loadStream("score.mp3") -- Sonido al ganar puntos
```

Muchas Gracias