

TPI - Juego (Mini Mario Bros simplificado)

Integrantes:

- [Marcelo Carabajal]
- [Gisela Lanzillotta]
- [Gastón Carlos Carabajal]
- [Federico Negreira]

Tipo de proyecto: Juego

Tema: Mini Mario Bros (versión reducida, un nivel)

Descripción:

Se desarrollará un juego de plataformas 2D en C++ utilizando **SFML** como librería gráfica. El jugador controlará a un personaje que puede moverse y saltar en un escenario con plataformas, enemigos básicos y una meta final. El objetivo es completar el nivel evitando enemigos y obstáculos, acumulando puntaje y guardando los récords en un archivo binario utilizando la estructura `FILE` de C.

Archivos a utilizar

Puntajes.dat

- `idJugador` : int
- `nombre` : char[20]
- `puntaje` : int
- `tiempo` : float

(Opcional - ver despues si nos conviene agregarlo)

Configuracion.dat (para guardar preferencias básicas)

- `volumen` : int
 - `dificultad` : int
 - `pantallaCompleta` : bool
-

Clases principales

1. Entidad (abstracta)

- `posX` : float

- `posY : float`
- `velX : float`
- `velY : float`
- `ancho : int`
- `alto : int`
- Métodos virtuales: `update()` , `draw()`

2. Jugador : public Entidad

- `vidas : int`
- `score : int`
- Métodos: `mover()` , `saltar()` , `colisionar()` .

3. Enemigo : public Entidad

- `tipo : int` (ej: 1 = básico, 2 = rápido)
- Métodos: `mover()` , `colisionarJugador()` .

4. Plataforma : public Entidad

- `solida : bool`
- Métodos: `colisionarJugador()` .

5. Meta : public Entidad

- `activada : bool`
- Método: `ganar()` .

6. Juego

- `vector<Entidad*> entidades`
- `Jugador player`
- Métodos: `loopPrincipal()` , `detectarColisiones()` , `guardarPuntaje()` .