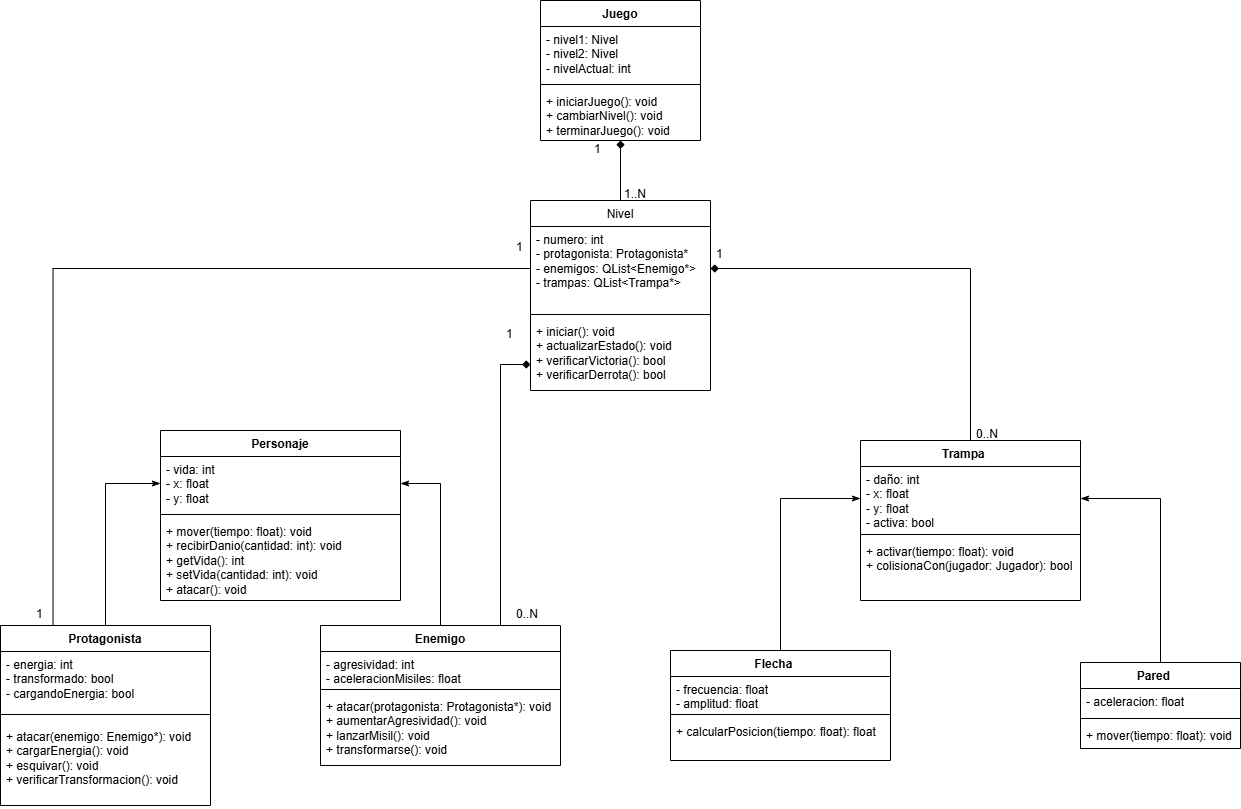
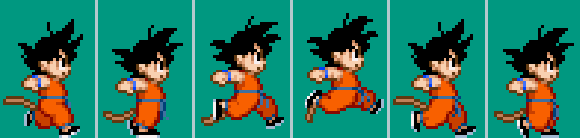
En este momento no se realizaron cambios grandes con respecto a la propuesta presentada en el Momento I. Se mantiene la estructura de dos niveles, sus dinámicas, físicas y objetivos, tal como fueron descritos anteriormente.

**Diagrama de clases de la capa lógica:**

El siguiente diagrama representa la capa lógica del videojuego. Se incluye herencia propia a través de las clases *Personaje* (base de Jugador y Pilaf) y *Trampa* (base de Flecha y Pared). La clase Nivel organiza los elementos de cada etapa, y Juego coordina el flujo general. Todas las relaciones de composición y herencia están representadas.

**Sprites a usar:**

**Nivel 1**



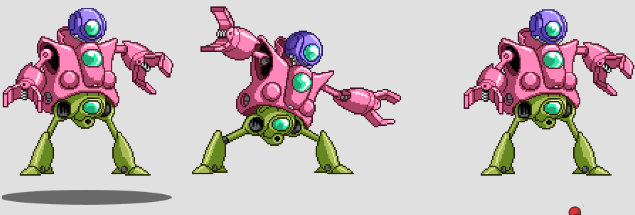




C:\Users\iclic\AppData\Local\Temp\af04421c-b614-4cff-ac7d-cb5049f36cb2_Traps.zip.cb2\Push_Trap_Right.png

**Nivel 2:**





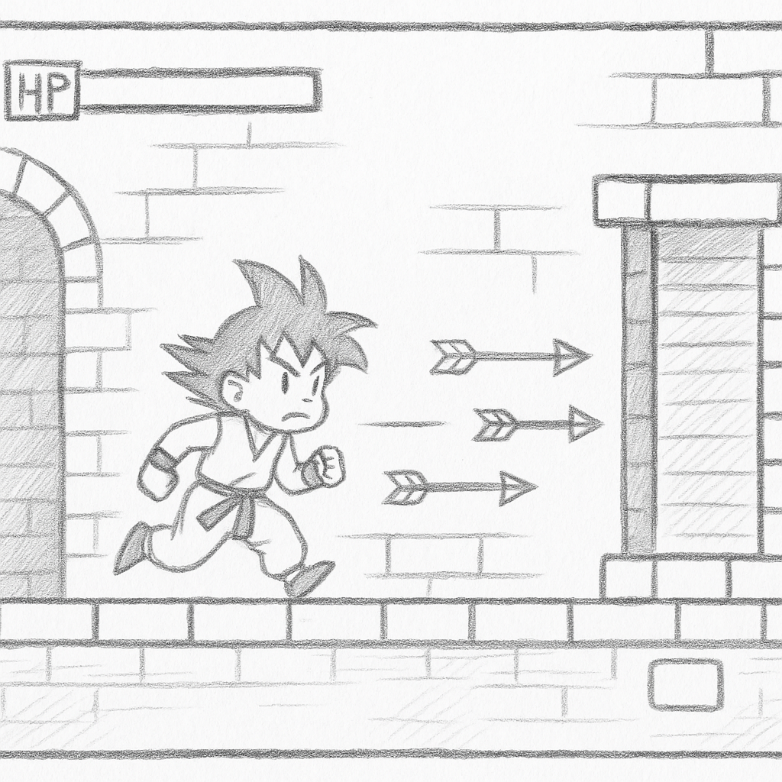
C:\Users\iclic\AppData\Local\Temp\68788e68-c07b-497b-97c6-6153656ff83e_missile.zip.83e\spr_missile.png

**Vistas del juego**

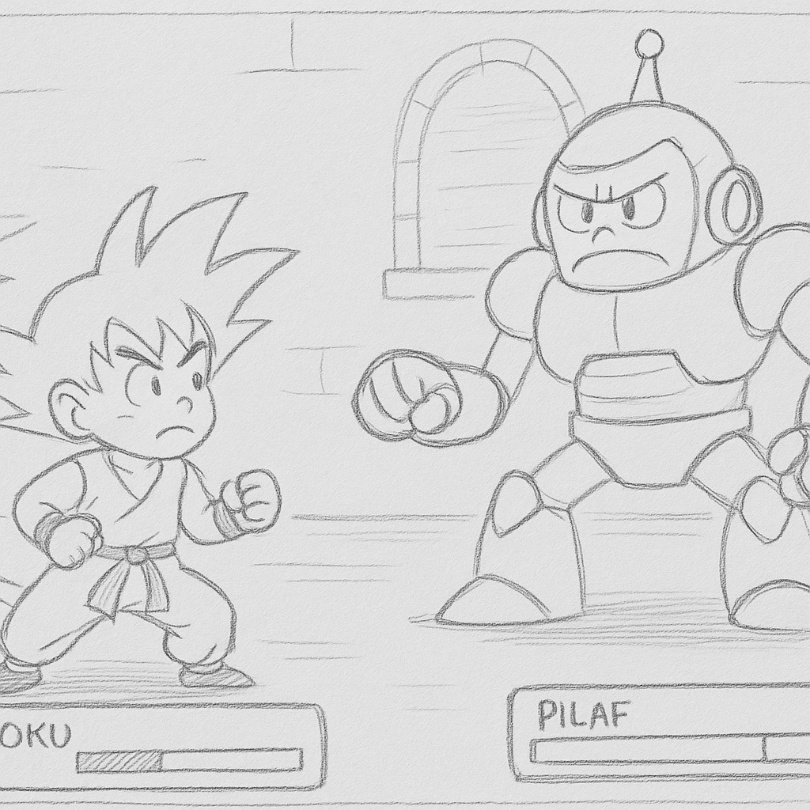
- Nivel 1: Vista lateral 2D con desplazamiento.

- Nivel 2: Vista frontal tipo combate por turnos.

**Boceto nivel 1: exploración lateral:**



**Boceto nivel 2: combate frontal:**



**Físicas a implementar**

***1 nivel:***  
- *Movimiento oscilatorio (flechas):*  x(t) = A \* sin(ω \* t)  
- *Movimiento parabólico (salto de Goku):* y(t) = -0.5 \* g \* t² + v₀ \* t  
- *Movimiento con aceleración (pared):* x(t) = 0.5 \* a \* t²

***2 nivel:***  
**-***Movimiento parabólico (misiles de Pilaf):* y(t) = -0.5 \* g \* t² + v₀ \* t  
-*Movimiento con aceleración (Kamehameha cargado de Goku):* x(t) = 0.5 \* a \* t²