

## Objetos del juego [ [LINK](#) ]

**Jugador:** Es el personaje del juego controlado por el usuario. Tiene que acumular tantos puntos como sea posible destruyendo los objetos del juego **Enemigo** mientras evita su contacto. Puede moverse hacia la derecha, hacia la izquierda, saltar y disparar objetos del juego **Balas** sobre los enemigos. Los puntos y las vidas se almacenan en propiedades personalizadas.



**Enemigo:** El personaje antagonista que cae en escena y elige aleatoriamente una dirección de movimiento. Rebota al chocar con los límites horizontales de la escena. Puede dañar al jugador y puede destruirse con objetos del juego **Balas**.



**Terreno:** Objeto estático que configura las zonas transitables del jugador y del enemigo.

**Plataformas:** Objeto estático que configura las zonas donde el jugador y el enemigo se pueden parar.

**Bala:** generada por el jugador para destruir enemigos y por el enemigo para destruir al jugador. Se mueve en la dirección de quien la lanza hasta que alcanza un objetivo o los límites de la escena, en ambos casos es destruida.

**Vidas:** Inicia con tres vidas y pierde una de estas con cada bala recibida o con cada colisión con una trampa. La recuperación de la salud del jugador solo es posible si encuentra corazones a lo largo del nivel.



**Puntuación:** Se debe proveer visualización de puntuación, Dicha puntuación se incrementa por cada enemigo destruido y por cada elemento **Fruta** recolectado.



**Cronómetro:** Controlador de tiempo para los sesenta segundos disponibles para cada nivel. Cuando llega a cero, el juego termina.



**Frutas:** objeto auxiliar que forma parte de cada nivel a efectos de ser recolectado por el Jugador



**Trampas:** objeto auxiliar que forma parte de cada nivel a efectos de dificultar el libre movimiento del Jugador, al entrar en contacto con este le resta una vida.



**Generador de enemigos:** objeto auxiliar que suelta aleatoriamente objetos enemigos desde la parte superior de la pantalla.

**Niveles:** Objeto auxiliar encargado de agrupar a todos los elementos antes descritos a efectos de conformar un nivel. Dicha información deberá estar soportada en un archivo JSON.

## Reglas asociadas a los movimientos del Jugador y del Enemigo.

El **Jugador** está restringido a cambios de izquierda a derecha y a saltos. En este caso, los saltos están limitados por una condición de contacto con el suelo o plataforma.

Los movimientos del **Enemigo** son autónomos y están determinados por dos reglas de comportamiento que dependen de los eventos de colisión. El primero ocurre con su primer contacto con el suelo, cuando seleccionará aleatoriamente una dirección de movimiento (izquierda o derecha). El segundo se activa si alcanza los límites horizontales de la escena (ya sea por colisionar con un elemento o por alcanzar los límites de la pantalla), momento en el que cambiará de dirección.

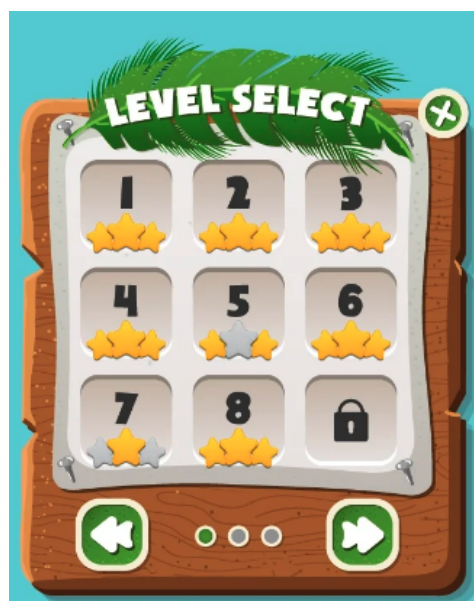
## Audio

**Efectos de sonido:** el juego deberá contar con efectos de sonidos que acompañen las acciones del **Jugador** y del **Enemigo** (por ejemplo al saltar, colisionar con las balas, recolectar una fruta, etc.)

**Música:** también se deberá agregar música ambiental que se reproduzca durante el gameplay

## Pantallas del juego

**Selección de nivel:** El juego deberá contar con una pantalla que permita seleccionar por lo menos tres niveles distintos, cada nivel deberá proponer un nuevo mapa con distintos enemigos.



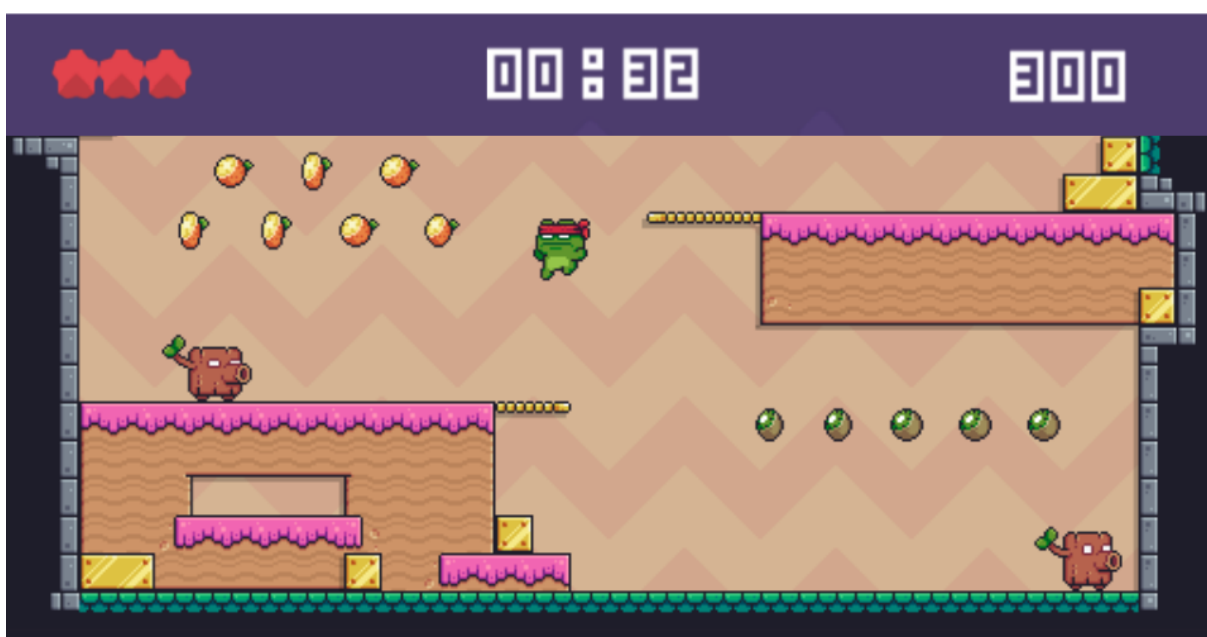
**Seteos del Juego:** En todo momento el juego deberá permitir encender y apagar tanto los efectos de sonido como la música de fondo.



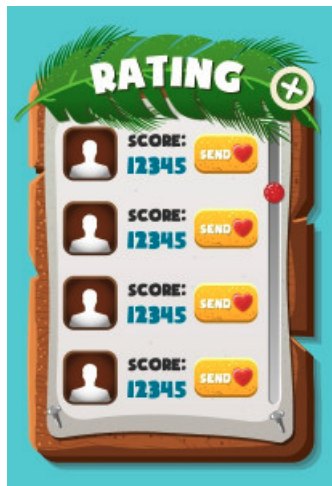
**Pausar Juego:** En todo momento el juego deberá permitir la posibilidad de ser pausado, para lo cual deberá contar con una pantalla que represente dicho estado.



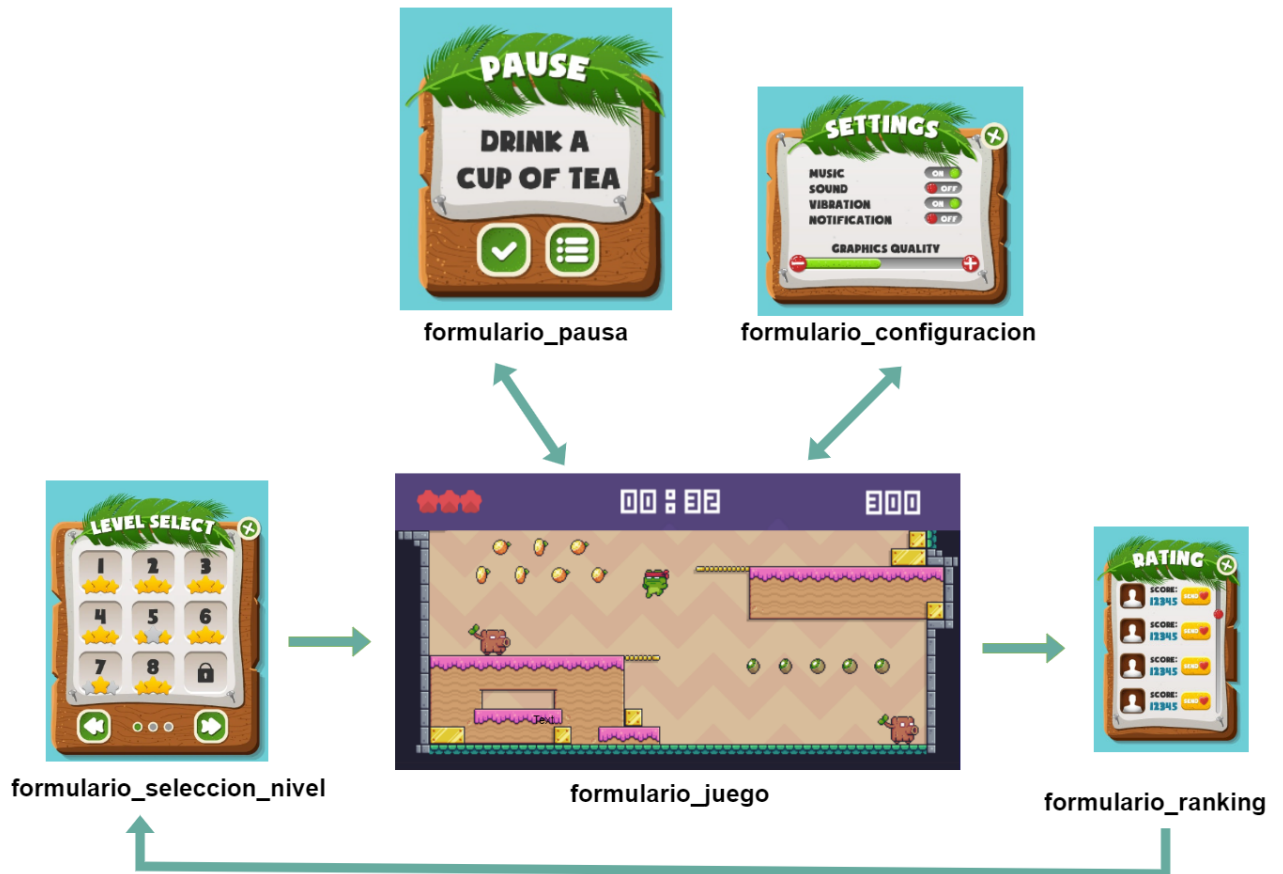
**Pantalla Principal:** La pantalla deberá contener no solo el escenario, los enemigos, las recompensas, las trampas y al jugador, sino también la información de vida, tiempo y puntaje del juego.



**Pantalla de ranking:** Al finalizar cada juego si el puntaje realizado por el jugador así lo amerita se deberá registrar a este en el ranking. Dicha información deberá estar soportada en una base de datos Sqlite.



El flujo de navegación entre los formularios debería ser de la siguiente manera:



**Posibles dinámicas de juego:** A continuación se presentan posibles dinámicas que podrá tener el juego con los recursos disponibles.







