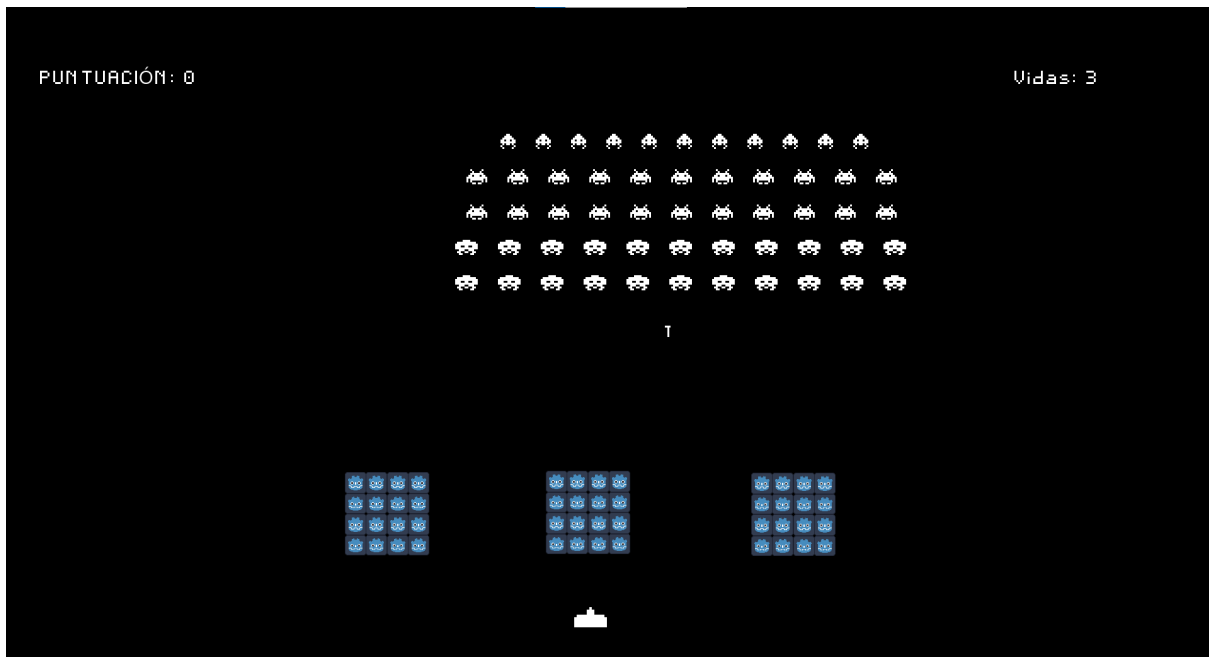
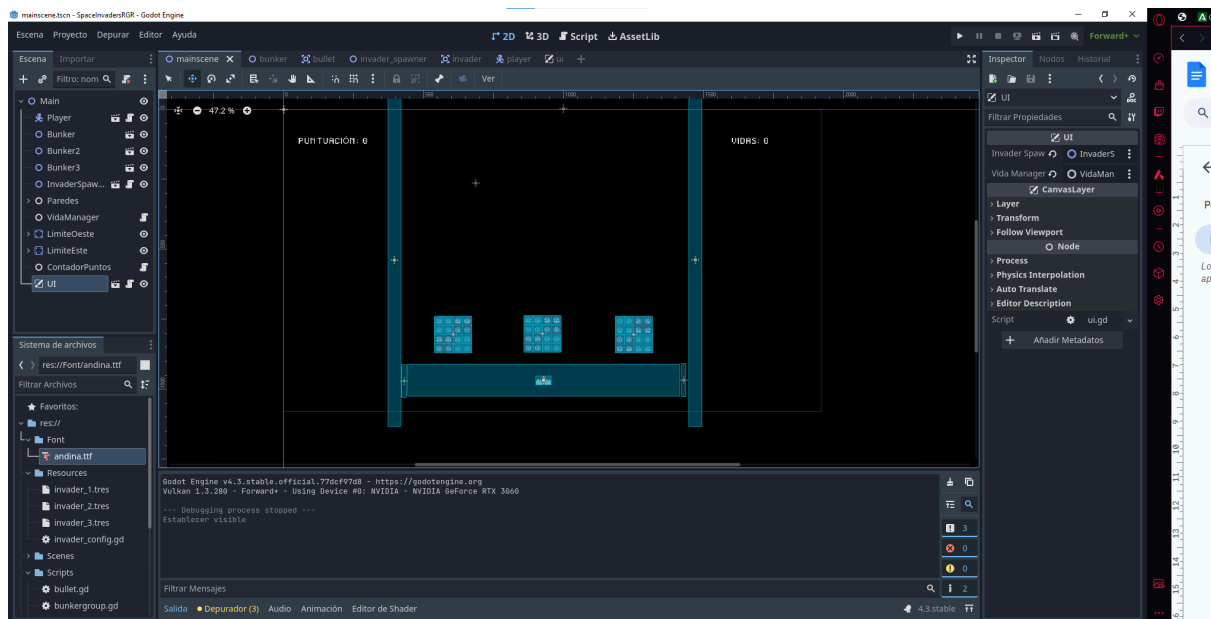


Enlace Github: <https://github.com/Zetille03/SpaceInvadersRGR>

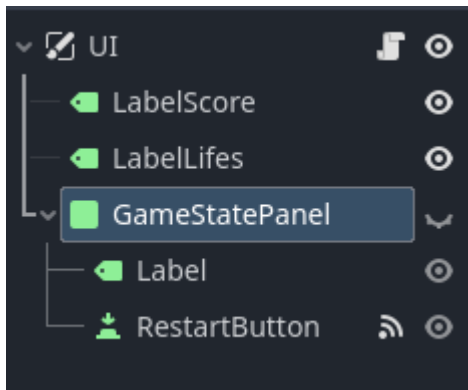
- **Pantalla de Juego:**



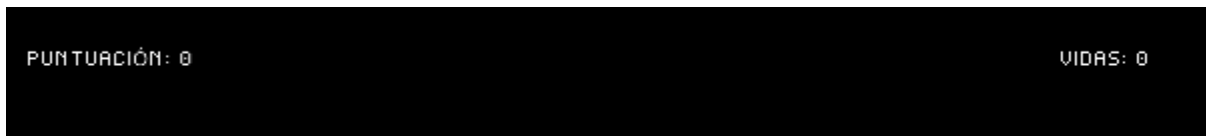
- **Espacio Jugable:** delimitados por las dos nodos de tipo Area2D orientadas verticalmente.



- UI: Bastante Simplona



Primero se maneja la puntuación (***LabelScore***) y las vidas (***LabelLives***).

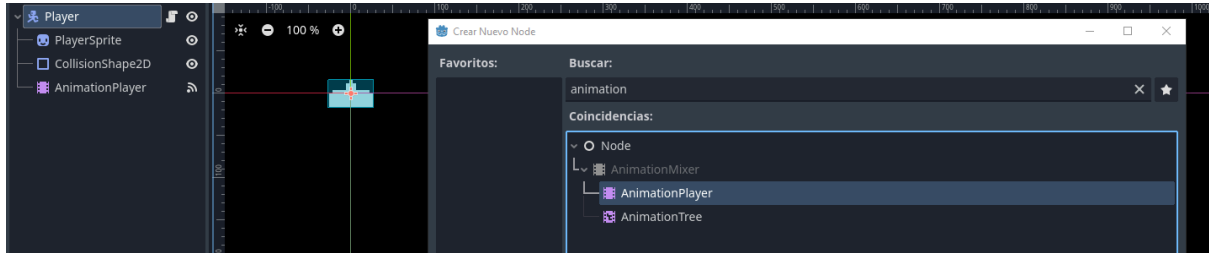


Luego uso un panel para mostrar si ha ganado o lo reutilizo para mostrar si ha perdido y añadir un botón para reiniciar la escena:



- **Animaciones:** tienen el player y los invaders

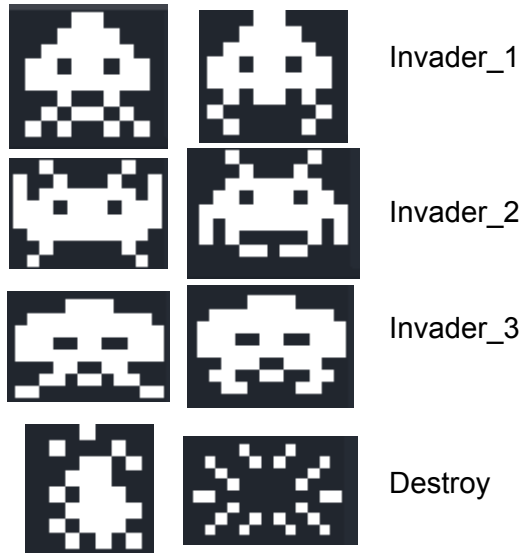
Para esto, debemos añadir un elemento al personaje deseado que se llama AnimationPlayer.



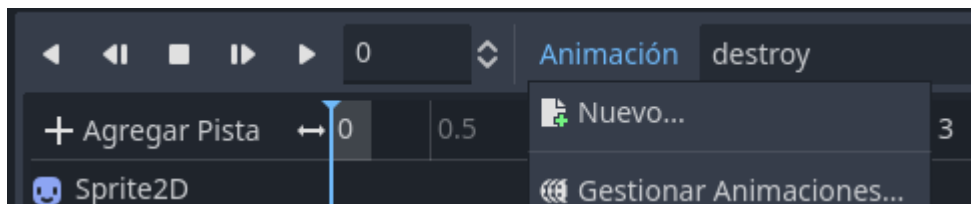
- **Player:** tiene solo una animación que repite 3 veces 3 sprites cada 0.25 veces y dura en total 2.25 segundos (no se repite).



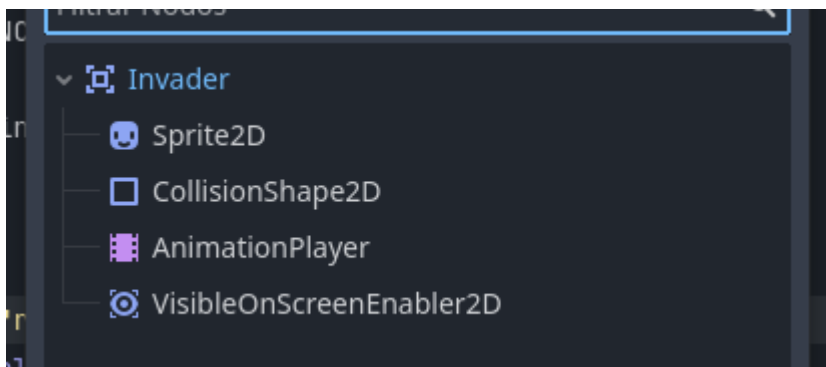
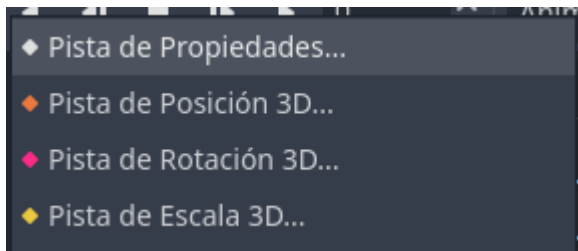
- **Invader:** tiene 4 animaciones, una por cada tipo de invaders para cuando estos se mueven y una cuando lo destruyen.



Para añadir la animación, cuando clicamos sobre el node de tipo **AnimationPlayer**, abajo del editor, de las 5 opciones (Salida, Depurador, etc) tenemos la de **Animación**. Ahí clicamos en **Animación>Nuevo** y le ponemos un nombre:



Le damos a **Agregar Pista** y en la ventana que nos sale seleccionamos Pista de **Propiedades>Sprite2D>Texture**.



Luego desde código dependiendo de las condiciones que se vayan cumpliendo se activa de la siguiente manera:

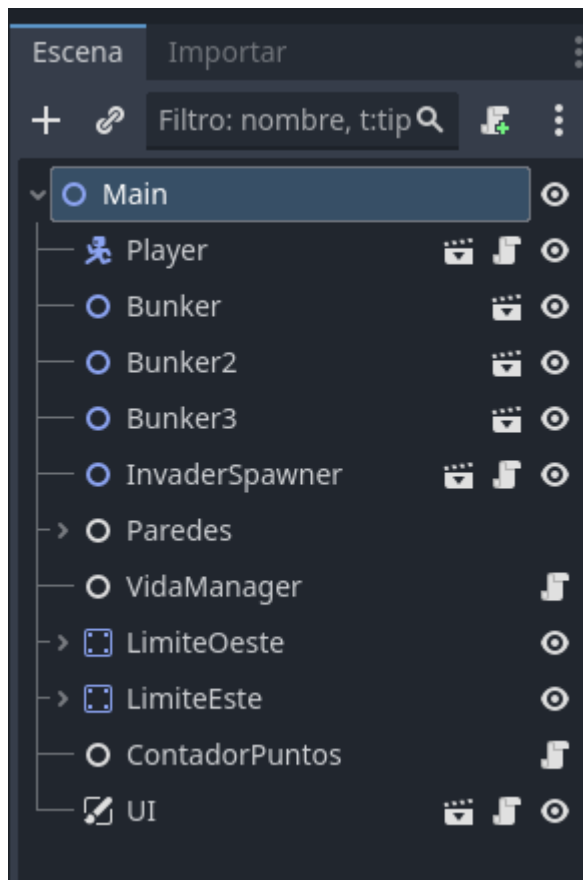
Registramos la variable:

```
@onready var animation_player = $AnimationPlayer
```

Y ejecutamos la animación (teniendo en cuenta el nombre que se estableció):

```
func _on_area_entered(area: Area2D) -> void:
    if area is Bullet:
        animation_player.play("destroy")
```

La escena principal queda así:



- **InvaderSpawner:** encargado de spawnear todos los invaders nada más se empieza la ejecución y de que se muevan y disparen. También maneja si se gana la partida al matar todos los invaders o si se pierde al llegar abajo.

- **VidaManager:** como su nombre indica, maneja las vidas, que se reflejen en el UI y que si llega a 0 salga la pantalla de derrota.

- **Límites:** ambos de tipo StaticBody2D para no tener que manejar por código los límites de movilidad del jugador.

- **ContadorPuntos:** como su nombre indica, maneja los puntos y que se reflejen en el UI (cada invaders suma diferentes puntos).

Lo demás es bastante obvio