

# NEON HAVOC

## Juego de Disparos Top-Down

Antonio Morales C.

[TILV22] Desarrollo de Videojuegos

Profesor: Sebastián Alejandro Pizarro Álvarez

## Vampire Survivors



Figura: Bullet heaven 2D

## Megabonk



Figura: Bullet heaven 3D

*Combinación de supervivencia arcade y mecánicas automáticas*

## Proyecto Educativo de Videojuegos

- Objetivo: Crear un shooter arcade top-down funcional
- Motor de desarrollo: Godot Engine 4.5
- Lenguaje: GDScript (scripting nativo de Godot)
- Arquitectura: Basada en escenas y nodos

## Concepto del Juego

- El jugador controla a Mateo, un héroe que debe sobrevivir oleadas infinitas de drones enemigos

## Mecánicas Principales:

- Sistema de disparo automático con auto-aim
- proyectiles auto-dirigidos (homing missiles)
- Enemigos con IA de persecución
- Sistema de vida para jugador (100 HP) y enemigos (3 HP)
- Contador de kills y feedback visual

## Escenas Principales (.tscn)

- nivel.tscn - Escena principal del juego
- mateo.tscn - Jugador controlable
- drone\_paco.tscn - Enemigo con IA
- bala.tscn - Proyectil auto-dirigido

## Scripts Asociados (.gd)

- nivel.gd — mateo.gd — drone\_paco.gd — bala.gd

# Node2D - Nodo Nivel

**Herencia:** Node2D (Nodo 2D base)

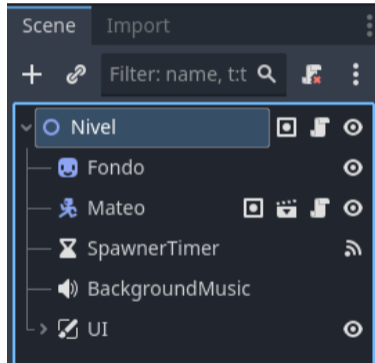
## Descripción:

- Contenedor principal del juego, gestiona el estado global

## Responsabilidades:

- Spawn procedural de enemigos cada 2 segundos
- Sistema de contador de kills
- Gestión de la interfaz de usuario (UI)
- Reproducción de música de fondo en loop

**Componentes:** Sprite2D (fondo), AudioStreamPlayer, CanvasLayer (UI)



# CharacterBody2D - Jugador (Mateo)

**Herencia:** CharacterBody2D (Cuerpo físico controlable)

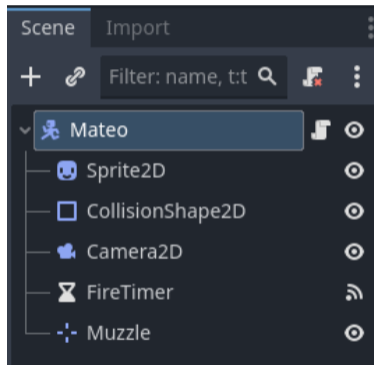
- Personaje controlado por el jugador con física integrada

**Propiedades:**

- Velocidad: 250 px/s — Vida: 100 HP
- Collision Layer: 1 — Collision Mask: 2

**Funcionalidades:**

- Movimiento con flechas (move\_and\_slide)
- Disparo automático cada 2 seg hacia enemigo más cercano
- Sistema de vida con feedback visual (flash rojo)
- Límites de pantalla (clamp 0-1280, 0-720)



# CharacterBody2D - Enemigo (DronePaco)

**Herencia:** CharacterBody2D (Cuerpo físico con IA)

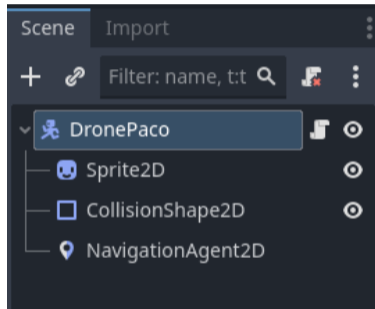
- Enemigo con inteligencia artificial de persecución

**Propiedades:**

- Velocidad: 100 px/s — Vida: 3 HP — Daño: 10 HP
- Collision Layer: 2 — Collision Mask: 1

**Comportamiento IA:**

- Busca al jugador mediante grupo 'player'
- Persigue constantemente al jugador (dirección normalizada)
- Sprite flip horizontal según dirección (mira de frente)
- Inflige daño por contacto (get\_slide\_collision\_count)



# Area2D - Proyectoil (Bala)

**Herencia:** Area2D (Área de detección sin física)

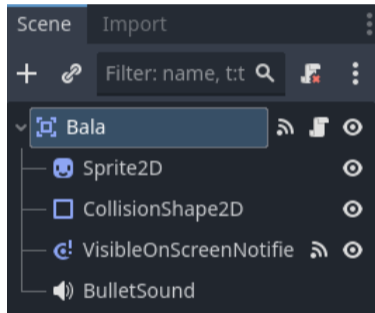
- Proyectoil auto-dirigido tipo misil teledirigido

**Propiedades:**

- Velocidad: 600 px/s — Daño: 1 HP
- Collision Layer: 4 — Collision Mask: 2

**Sistema Homing (Auto-dirección):**

- Busca enemigo más cercano mediante grupo 'enemy'
- Interpolación suave con lerp() (homing\_strength: 3.0)
- Rotación del sprite según dirección (angle())



- Timer (SpawnerTimer, FireTimer) - Eventos periódicos
- AudioStreamPlayer - Música de fondo y efectos de sonido
- CanvasLayer + Label - Interfaz de usuario (Kills, HP)
- VisibleOnScreenNotifier2D - Detección de salida de pantalla

**¡Prueba el juego ahora! Versión Prototipo Disponible:**

**Enlace principal:**

`https://sauk1346.github.io/neon-havoc/`

**Enlace alternativo:**

`https://www.shorturl.at/UKsxa`