

# Documentación Técnica - Proyecto GamePython

## ■ Propósito general

El paquete `game` contiene la lógica interna y los sistemas base del motor del juego. Su objetivo es separar la infraestructura del juego (motor, cámara, lógica del mundo) de la parte jugable (personaje, enemigos, escenarios).

## ■ Descripción de cada módulo

### **engine.py**

Núcleo del motor del juego. Maneja el ciclo principal (main loop): eventos, actualización, renderizado y control de FPS. Coordina los subsistemas como cámara, entidades y niveles.

### **camera.py**

Maneja la cámara y el desplazamiento del mundo. Sigue al jugador y mantiene el enfoque dentro de los límites del escenario.

### **core.py**

Define las clases base: GameObject, Entity, Component. Contiene lógica común como posición, sprite, eventos de actualización y dibujo.

### **enemies.py**

Define las clases de enemigos, sus patrones de IA, movimiento, colisiones y comportamiento general.

### **main.py**

Punto de entrada principal. Inicializa el motor y configura los parámetros globales (resolución, FPS, título).

## ■ Arquitectura general

main.py → engine.py → (camera.py, core.py, enemies.py) → Pygame API El módulo principal inicia el motor, el motor coordina subsistemas, y éstos se comunican con Pygame para manejar eventos, lógica y renderizado.

## ■ Roadmap de evolución del juego

### **Fase 1 — Base técnica (Nivel Básico)**

- Crear estructura base de carpetas (assets/, sprites/, levels/).
- Configurar ventana con Pygame.
- Implementar personaje con movimiento lateral y salto.
- Agregar cámara que siga al jugador.
- Renderizar fondo y escenario.

### **Fase 2 — Mecánicas y enemigos (Nivel Medio)**

- Colisiones con el terreno.
- IA básica de enemigos.
- Ataques y proyectiles.
- Sistema de vidas y daño.
- Sonidos, música y HUD.

### **Fase 3 — Escenarios y pulido (Nivel Avanzado)**

- Sistema de niveles (Tiled, JSON, TMX).

- Transiciones y pantallas de menú.
- Guardado de progreso.
- Optimización y empaquetado ejecutable.

## ■ Herramientas recomendadas

- Pygame CE (Community Edition) — versión moderna de Pygame.
- Tiled Map Editor — para diseño visual de niveles.
- PyInstaller — para empaquetar el juego como ejecutable.