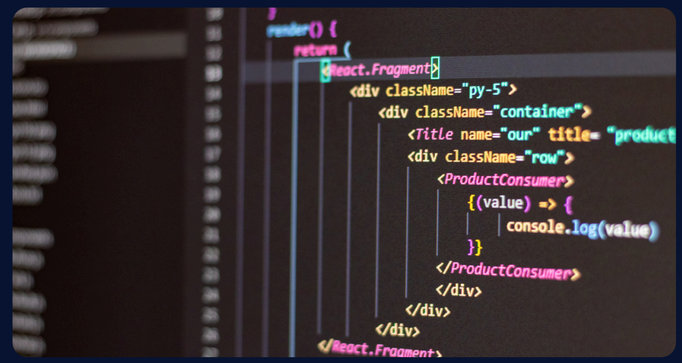


Proyecto de Python: Ping-Pong

Presentado por:

Iker Díaz-Maroto
Roberto Carrascoso
David García

Índice

Configuración del Entorno de Desarrollo

Para comenzar con el proyecto, lo primero que hicimos, fue instalar Python y VScode en nuestros equipos. Después, creamos un directorio para el proyecto.

Una vez iniciado el proyecto, lo primero que hicimos fue instalar la biblioteca de pygame con el comando:
pip install -U pygame
Comprobamos la instalacion ejecutando un ejemplo en pygame:
python -m pygame.examples.aliens

Estructura de Archivos y Recursos

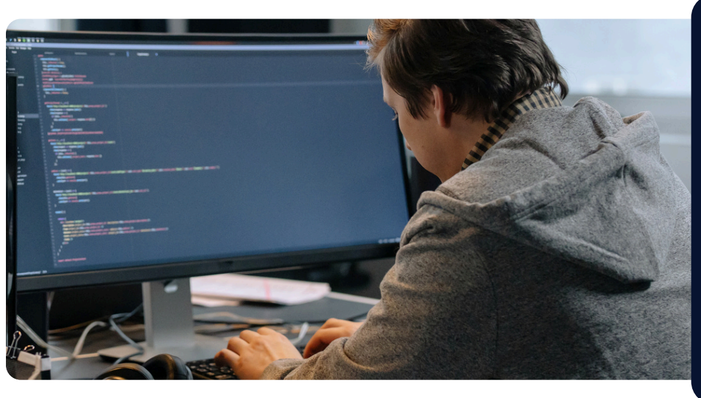
Hemos utilizado como lenguaje de programación principal Python, en especifíco en la biblioteca Pygame. Nos hemos enfocado en una programación orientada a los objetos.

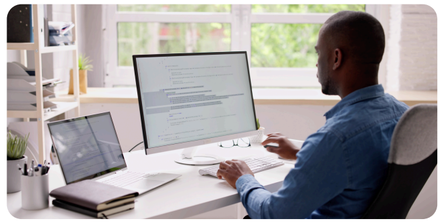
Para el juego, nos basamos en esta estructura de archivos:

```
.
├── pingpong.py          # Archivo principal del juego
├── README.md            # Documentación del proyecto
└── assets/              # Carpeta con recursos multimedia
    ├── error.mp3        # Sonido cuando un jugador falla
    ├── fondo.png        # Imagen de fondo principal
    ├── fondo2.png       # Imagen de fondo alternativa
    ├── fondo3.png       # Imagen de fondo alternativa
    ├── musica.mp3       # Música de fondo
    ├── pelota.png       # Imagen de la pelota
    ├── raqueta.png      # Imagen de las raquetas
    └── rebote.mp3       # Sonido de rebote de la pelota
```

Controles del Juego

Para un juego tan simple para el usuario como lo es el Pong, los controles deberían de ser igual de simples, por lo que fuimos a lo facil; El jugador 1 (raqueta izquierda) usara las teclas W y S para moverse hacia arriba o hacia abajo, respectivamente. El jugador 2 (raqueta derecha) se movera con las flechas de “arriba” o “abajo”, en el sentido de la flecha.





Caracteísticas Implementadas

Nuestro juego incluye gráficos en 2D con imágenes personalizadas, un sistema de físicas básico para el movimiento y rebote de la pelota, y un aumento progresivo de la velocidad para hacer el juego más difícil.

Tiene detección de colisiones entre la pelota y las raquetas, un sistema de puntuación visual, efectos de sonido para rebotes y puntos, música de fondo y un mensaje final que muestra al ganador.

Desafíos y Soluciones

La programación se trata de resolver problemas. Los programadores debemos pensar de manera creativa y lógica para descubrir cómo hacer que sus programas hagan lo que desean.

Gracias a este proyecto, hemos aprendido a resolver problemas, como el manejo eficaz de rutas de archivos; que fue solucionado usando funciones de *os.path*, o la deteccion de las colisiones, que solucionamos implmentando un sistema basado en coordenadas para que el sistema sepa donde esta la pelota y las palas en todo momento.

Conclusiones y Aprendizajes

Este proyecto nos permitió aplicar:

- Una programación orientada a los objetos
- El manejo de eventos en el codigo
- Procesamiento de contenidos multimedia
- Implementacion de la lógica de los juegos

Instalación desde otro equipo

Para poder instalar el juego, debemos de tener en cuenta que nuestro equipo debe cumplir ciertos requisitos:

- Tener instalado Python 3.13
- Tener instalada la libreria Pygame

Una vez comprobados los requisitos, clonaremos el repositorio con el comando;
git clonehttps://github.com/robertocarrascoso/PingPong_RDI.git
Y ejecutaremos el juego con el comando;
python pingpong.py.

Reglas del juego

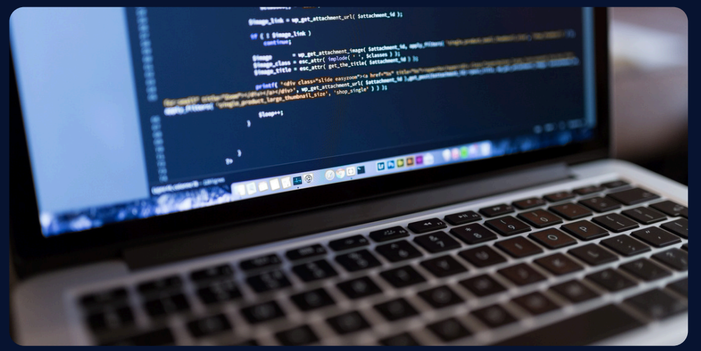
El juego se juega entre dos jugadores, cada uno controlando una raqueta. La pelota rebota en las paredes superior e inferior y en las raquetas. Si un jugador no golpea la pelota, el oponente gana un punto.

El primero en llegar a 5 puntos gana la partida, y la velocidad de la pelota y las raquetas aumenta progresivamente durante el juego.

Recursos utilizados para crear el juego

Para poder llevar a cabo este proyecto de una manera correcta y fluida, hemos utilizado un tutorial de Pygame en Youtube como guia para completar nuestro proyecto con un codigo simple y legible a primera vista, enfocandonos en la elegancia del mismo.

Videotutorial sobre Python y Pygame en VScode



```
if presentacion_termina:
    print("¡Gracias a
todos por la atención!")
```