



**FACULTAD DE INGENIERÍA CARRERA DE INGENIERÍA  
INFORMÁTICA FACULTAD COMUNITARIA DE  
CAACUPÉ**

## **OPTATIVA I**

**Tema: Desarrollo de un juego con Python**

**Alumno:**

- Edgar Nahuel Alfonso Amarilla

**Profesor:** Ing. Ricardo Maidana.

**Curso:**

- Quinto año, Noveno semestre

**Turno:**

- Noche

**Caacupé – Paraguay Año:2024**

## **Introducción**

El desarrollo de juegos es una disciplina que combina creatividad, programación y diseño para crear experiencias interactivas que entretienen y desafían a los jugadores. Pygame, una biblioteca de Python diseñada específicamente para el desarrollo de juegos, ofrece una plataforma accesible y flexible para principiantes y desarrolladores experimentados. En este proyecto, crearemos un juego básico de Buscaminas utilizando Pygame.

El objetivo de este proyecto es desarrollar un juego en el que el jugador pueda descubrir todas las celdas sin minas en un tablero. Aunque es un proyecto sencillo, proporciona una excelente introducción a los conceptos fundamentales del desarrollo de juegos.

## **1. Inicialización**

El código comienza importando las bibliotecas necesarias: pygame, random, y sys. Luego, se inicializa Pygame y se establece un reloj para controlar la velocidad de actualización de la pantalla.

## **2. Configuración de la Pantalla**

Se define el tamaño de la pantalla con un ancho de 1280 píxeles y una altura de 600 píxeles. A continuación, se crea la superficie de visualización utilizando estos parámetros.

## **3. Carga de Imágenes**

Se cargan varias imágenes que se utilizarán en el juego, incluyendo fondos de pantalla, botones (piedra, papel, tijera), y otros elementos gráficos como la línea divisoria y el contador de rondas.

## **4. Inicialización de la Fuente**

Se inicializa una fuente personalizada utilizando el archivo de fuente "Roboto.ttf" para renderizar texto en la pantalla.

## **5. Variables del Juego**

Se definen varias variables que mantendrán el estado del juego:

- tirada: Guarda la elección de la computadora (piedra, papel o tijera).
- jugado: Un booleano que indica si el jugador ha hecho una jugada.
- estado: Indica el estado del juego (-1 para juego en progreso, 0 para empate, 1 para victoria del jugador, 2 para derrota del jugador, 3 para fin del juego).
- jugada: Guarda la elección del jugador.
- ronda: Contador de rondas restantes.
- score\_jugador y score\_ia: Puntajes del jugador y la computadora respectivamente.

## **6. Bucle Principal del Juego**

El juego corre dentro de un bucle infinito que se encarga de:

- Dibujar las imágenes de fondo y otros elementos gráficos en la pantalla.
- Renderizar el contador de rondas.
- Mostrar mensajes de resultado según el estado del juego.
- Manejar los eventos de entrada del usuario, como clics del ratón y cierre de la ventana.
- Determinar el resultado de cada ronda basándose en las elecciones del jugador y la computadora.
- Actualizar los puntajes y el estado del juego según el resultado de cada jugada.
- Controlar la animación de elementos gráficos.

## **7. Dibujar Elementos**

El juego dibuja los elementos gráficos en la pantalla, incluyendo los botones de piedra, papel y tijera, la línea divisoria, el contador de rondas, y los mensajes de resultado según el estado actual del juego.

## **8. Manejo de Eventos**

El código maneja diferentes tipos de eventos:

Cierre de la ventana del juego.

Clics del ratón para registrar las elecciones del jugador (piedra, papel, tijera).

Reinicio del juego cuando el estado es "Game Over" y se hace clic en cualquier parte de la pantalla.

## **9. Lógica del Juego**

La lógica del juego se encarga de:

Generar una elección aleatoria para la computadora.

Comparar la elección del jugador con la elección de la computadora para determinar el resultado de la ronda.

Actualizar el estado del juego y los puntajes según el resultado.

## **10. Actualización de la Pantalla**

El bucle principal se asegura de que la pantalla se actualice con el nuevo estado del juego en cada iteración, controlando la tasa de fotogramas por segundo para una animación suave.

## **Conclusión**

Este código implementa un sencillo juego de "Piedra, Papel o Tijera" utilizando Pygame. El jugador compite contra la computadora a través de varias rondas, con los resultados mostrados en la pantalla. La estructura del código asegura que el juego sea interactivo y que se manejen adecuadamente los eventos y estados del juego.

Anexo

