

Universidad José Antonio Páez

Facultad de Ingeniería

Departamento de Ingeniería en Computación

**Documentación del Código para**

**Simulación de el Juego de dardos**

PROFESOR: ALUMNOS:

María García Acosta Gregory 29.947.116

Daniel Racero 28.465.991

Daniel Almarza 30.195.507

Daniel Rivero 30.334.420

Manuel Peralta 28.022.934

**Código para Simulación de Juego de dardos**

Este código es un juego de dardos para dos jugadores con una interfaz gráfica de usuario (GUI) creada con la biblioteca Tkinter de Python. El juego consiste en un tablero de dardos dividido en radianes de diferentes colores, cada uno con una puntuación asignada. Los jugadores se turnan para lanzar dardos haciendo clic en el botón “Lanzar dardo”. Los dardos caen en lugares aleatorios dentro de los radianes en el tablero y se muestra un círculo del mismo color que el radian donde cayó el dardo. Cuanto más cerca del centro del tablero caiga el dardo, más puntos ganará el jugador. Cuando un jugador alcanza los **250 puntos**, se muestra una ventana indicando que ese jugador ha ganado. El botón “Volver a jugar” se puede hacer clic para reiniciar el juego y comenzar de nuevo.

El código está organizado en una clase llamada DartGame que contiene varios métodos para manejar diferentes aspectos del juego. El método \_\_init\_\_ inicializa la ventana y los elementos de la interfaz gráfica, como el tablero de dardos, las etiquetas de puntuación y los botones para lanzar dardos y reiniciar el juego. El método throw\_dart se llama cuando un jugador hace clic en el botón “Lanzar dardo” y maneja el lanzamiento del dardo, calculando una posición aleatoria para el dardo, verificando si ya hay un dardo en esa posición, calculando la puntuación del lanzamiento y actualizando la puntuación del jugador correspondiente. El método end\_game se llama cuando un jugador alcanza los 250 puntos y muestra una ventana indicando que ese jugador ha ganado. El método reset\_game se llama cuando se hace clic en el botón “Volver a jugar ” y reinicia el juego borrando todos los dardos del tablero y restableciendo las puntuaciones y turnos de los jugadores. Finalmente, el método draw\_board dibuja el tablero de dardos en la ventana.

**Funciones:**

Este código contiene una clase llamada DartGame que tiene varios métodos para manejar diferentes aspectos del juego. Los métodos son los siguientes:

* **\_\_init\_\_(self, master**)**:** Este método es el constructor de la clase DartGame. Inicializa la ventana y los elementos de la interfaz gráfica, como el tablero de dardos, las etiquetas de puntuación y los botones para lanzar dardos y reiniciar el juego.
* **throw\_dart(self):** Este método se llama cuando un jugador hace clic en el botón “Lanzar dardo” y maneja el lanzamiento del dardo. Calcula una posición aleatoria para el dardo, verifica si ya hay un dardo en esa posición, calcula la puntuación del lanzamiento y actualiza la puntuación del jugador correspondiente.
* **end\_game(self, winner):** Este método se llama cuando un jugador alcanza los 250 puntos y muestra una ventana indicando que ese jugador ha ganado.
* **reset\_game(self):** Este método se llama cuando se hace clic en el botón “Volver a jugar” y reinicia el juego borrando todos los dardos del tablero y restableciendo las puntuaciones y turnos de los jugadores.
* **draw\_board(self):** Este método dibuja el tablero de dardos en la ventana.

**Uso a través de la Interfaz Gráfica:** El programa ofrece una experiencia interactiva a través de la interfaz. Los siguientes pasos describen cómo usar el programa:

1. Ejecute el código en un intérprete de Python con la biblioteca Tkinter instalada para iniciar el juego.
2. Haga clic en el botón “Lanzar dardo” para lanzar un dardo.
3. Observe dónde cayó el dardo y cuántos puntos ganó.
4. Espere su turno mientras el otro jugador lanza su dardo.
5. Repita los pasos 2-4 hasta que uno de los jugadores alcance los 250 puntos.
6. Haga clic en el botón “volver a jugar” para reiniciar el juego y comenzar de nuevo.