

# Documentación Proyecto II

Montserrat Monge Téllez, Luis David Richmond Soto.

## I. INTRODUCCIÓN

El presente documento se desarrollará la documentación del programa que pertenece al segundo proyecto del curso Taller de Programación. Ahora bien, en qué consiste dicho proyecto mencionado, consiste tratando una explicación general en la realización de una matriz 9x5 donde cada casilla será un espacio por avatar o rook, los avatar que ingresen al tablero de juego se generarán de manera aleatoria y el jugador colocará los rooks que pueda comprar con el dinero recogido para defenderse de los avatar. Los personajes tendrán un movimiento horizontal desde el límite derecho hasta el izquierdo. Este proyecto se deriva de una simulación del juego Plantas vs. Zombies, mas en este caso se llamará Avatars vs. Rooks. El lenguaje de programación que se utilizó fue Python (3.8.2) y la biblioteca principal que se usó fue pygame(1.9.6) para la parte programada se utilizaron los métodos de programación de iteración y recursividad.

## II. MARCO TEÓRICO

**1. pygame=** Biblioteca que permite la creación de videojuegos en distintas dimensiones de manera sencilla permitiendo el uso de objetos.

**2.sys=** Dentro de este módulo se encuentra la capacidad de acceder a variables usadas por el intérprete.

**3. time=** Se encarga de las mediciones del tiempo en si ya sea en tiempo real o para contar tiempos específicos y medir tiempos.

**4.thread=** Son los conocidos hilos que se encargan de ejecutarse como un subproceso del programa independiente.

## III. PROPUESTAS DE DISEÑO DE SOLUCIÓN

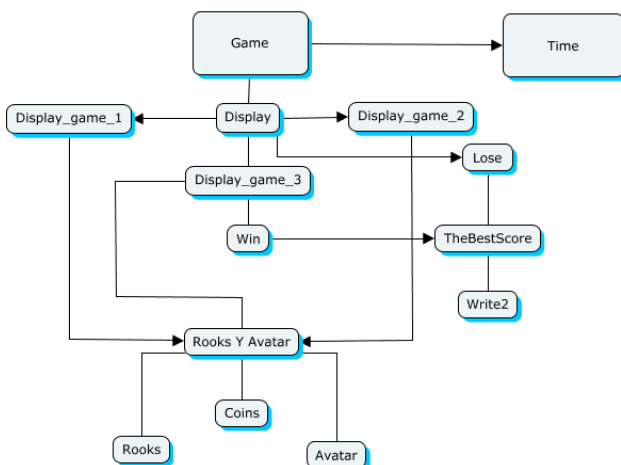


Fig. 1. Propuesta 1

### • Propuesta Game

Esta propuesta se encuentra "Game" que es la función principal de donde sale el hilo llamado "Time" y los tres Display que constan en las acciones de ganar o perder y así llegar a los mejores si cumple la condición de ganar. Seguidamente creará los rooks y avatars y luego las monedas, hasta que de esta manera se logre coponer el Juego completo.

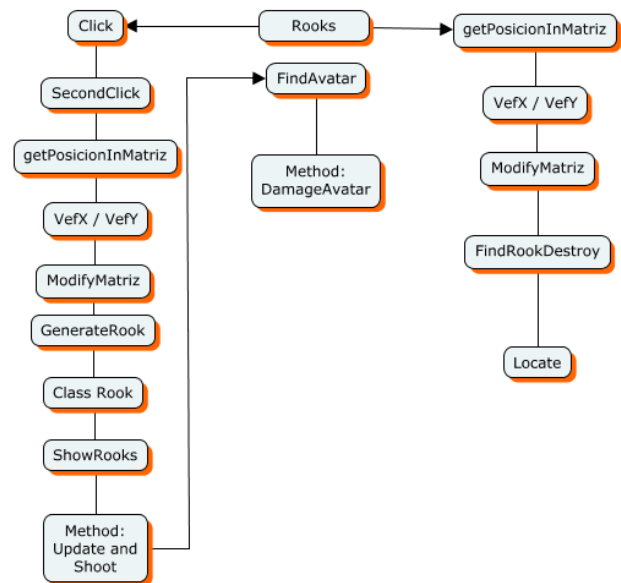


Fig. 2. Propuesta 2

### • Propuesta Rooks

Dentro de este apartado se encuentra el click que su función es captar cuando se quiere crear dicho rook específico y además el segundo click que lo posicionaría y muestra en pantalla (función de selección), esto llamaría a la posición de la matriz donde se va a mostrar y la llena(Tablero).Seguidamente el rook ya posicionado en el tablero lanzará sus disparos en cuanto detecte un avatar y este al tener contacto con el disparo será dañado y futuramente destruido.

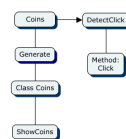
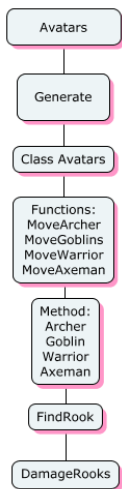


Fig. 3. Propuesta 3

### Propuesta Coins

Este pequeño diagrama explica la función de detectar un click y de esta manera realizar los cambios respectivos, ya sea crear

el objeto seleccionado o posicionarlo en la matriz que será el tablero visible.



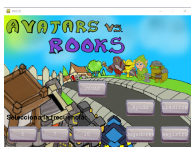
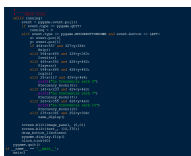
### • Propuesta Avatars

Este diagrama explica la creación, como los genera y mueve de manera aleatoria en las posiciones respectivas de la matriz, de manera que el movimiento sea fluido y ordenado. Cada uno tiene una clase diferente ya que cada uno ejerce un daño distinto. Al chocar con un rook este lo detectará y lo dañará para que así pueda avanzar y hacer que el jugador sea derrotado.

## IV. EJEMPLO DE FUNCIONAMIENTO

### 1. Pantalla de Inicio:

Dentro de esta pantalla se podrá encontrar una bienvenida amena con fondo musical de fondo, posee siete botones cada uno con su función propia, se encuentra el botón de ayuda que lleva a su respectiva ventana, el de créditos, jugadores y registro; además se encuentran tres botones de selección de frecuencia entre el cinco y el diez y finalmente el botón que iniciará la ventana del juego. Cabe destacar que si el usuario no ingresa un valor de frecuencia predeterminado de cinco.



Un pequeño ejemplo: Aquí se encuentra la validación de la posición de los botones y las respectivas llamadas a las frecuencias y la ventana del juego.

**2. Pantalla de Ayuda:** En esta pantalla se podrán encontrar las instrucciones y reglas correspondientes del juego. Se le muestra al usuario que debe de hacer para ganar el juego y además se le retrata la manera en la que funciona el juego.



Un pequeño ejemplo: Aquí se muestra cómo se ingresaron los textos, cómo fueron creados los botones del programa, como se realizó un bucle sencillo de una ventana y como se mostraron o dibujaron dichos aspectos anteriores.



**3. Pantalla de Créditos:** Dentro de esta pantalla se podrá encontrar datos de importancia para el programa, como sus creadores, profesor de curso, institución, entre otros. Es decir, se encuentran los respectivos reconocimientos.



Fig. 4. Propuesta 4

**4. Pantalla de Jugadores:** En esta pantalla se muestran los primeros tres lugares en orden de puntaje/tiempo.



Fig. 5. Propuesta 4

**5. Pantalla de registro:** [1]Dentro de esta pantalla se encuentra el registro del jugador, donde debe iniciar con el nombre que desee y seguidamente guardarlo. Una vez el usuario Guarda su nombre y la interfaz mostrará un mensaje que informe si ya está registrado, o bien si no lo ha hecho [2]Un pequeño ejemplo: Muestra como lee y almacena los datos que entraran en el cuadro de texto al presionar el botón de guardar y de esta manera hacer que el usuario quede registrado(mediante otra función que llama a esta mencionada).

**6. Pantalla de Juego:** [1]Esta es la pantalla de mayor importancia dentro de este proyecto, ya que contiene toda la lógica de la función principal, consta de tres niveles, cada uno con música y fondo diferentes. Dentro de esta pantalla deberás jugar y ganar para obtener el respectivo reconocimiento de victoria o ya sea de derrota. Un pequeño ejemplo: [2]Se muestra el click que posiciona el rook en la pantalla después del click de selección.



Fig. 6. Pantalla Registro[1]

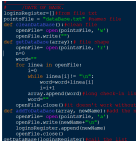


Fig. 7. Pantalla Registro[2]

## V. CONCLUSIÓN

Finalmente, tanto como en la sección del proyecto programado como en la en esta sección escrita se adquirió el conocimiento para lograr alcanzar la total realización de la lógica que posee el juego, se conocieron desde los datos mas básicos para poder iniciar hasta los datos mas complejos para la realización del juego, como lo es por ejemplo la importación de Pygame(mediante códigos) y su uso que adicionandolo fue una experiencia de aprendizaje desde cero hasta lo que se logró realizar, la creación de ventanas, botones, posicionamiento de textos, cambios de colores, ingresar imágenes de fondo o personajes, musica, objetos, entre otros. Hasta llegar a leer archivos por aparte del archivo de python como lo es un archivo txt para leer los mejores jugadores, o tambien la importacion de archivos externos que podrían ayudar al programa, la creación de funciones recursivas e iteraticas que realizan funciones como ingresar cambiar la matriz y sus posiciones, ingresar avatars y rooks, cronometro, botones, movimientos, animaciones, etc. A este resultado se ha llegado gracias a semanas de investigaciones y a pesar de los errores que se pudieron presentar durante la realización, se lograron resolver bien y concluir con el programa de manera exitosa. Este programa ha sido principalmente un reto personal para ambos programadores en el que la motivación y la constancia han jugado un gran papel, sin embargo se desea incitar al lector a profundizar en el conocimiento del tema, a interesarse sobre los métodos utilizados más allá de las bibliotecas e incluso los métodos de cada biblioteca que no fueron utilizados en este proyecto ya que brindan amplias capacidades al programa para ser realizado de la manera más eficiente y cómoda para el programador y el usuario.

## VI. RECOMENDACIONES

- Se utilizó programación orientada a objetos para la programación de Monedas, Avatar y Rooks ya que esto nos dió bastante flexividad para los atributos, por ejemplo: vida, daño que pueden realizar, posición en el tablero y posición en la matriz, imagenes, las animaciones, entre otras.
- Todos los objetos del juego a excepción de las monedas (Rooks,Avatars y proyectiles) interactúan con la matriz costantemente lo cual nos dio la facilidad para realizar los ataques de los Rooks como de los Avatars, también para saber las a cual Rook o Avatar se está dañando.
- Para realizar las animaciones "lose and



Fig. 8. Pantalla Juego[1]



Fig. 9. Pantalla Juego[2]

win" se usaron los conocidos sprites para lograr una fluidez mas visible y además por ahorro de código. -Se consideró la importación de otros archivos de python para mayor conicidad y flexibilidad con la interfaz. -Se recomienda el uso de botones por imagen ya que le brindan a la interfaz estética y le brinda al programador una forma consiza de crear botones. [4] [1] [?] [3] [2]

## REFERENCES

- [1] Vern Ceder and Nathan Yergler. Teaching programming with python and pygame. *Apresentado na PyCon*, 2003.
- [2] Arturo Fernández. *Python 3 al descubierto*. Alfaomega Grupo Editor, 2013.
- [3] L López. Programación orientada a objetos, 2006.
- [4] Albert Sweigart. *Making Games with Python & Pygame*. Independent, 2012.