Instituto Tecnológico de Costa Rica

Escuela de Ingeniería en Computadores

II Semestre 2024

Joshua Morales Guzmán

Carnet : 2023800086

**Introducción:**

En este proyecto cuyo objetivó era la creación de un video juego, basándose en el clásico de Super Pang de loa años 1990 además de eso uno de los retos fue el utilizar la librería de interfaces graficas con tkinter, así poder simular las físicas del juego con métodos recursivos.

Este código contiene una implementación detallada del juego "PANG" con el personaje Goku. Cuenta con múltiples funciones para manejar las mecánicas de movimiento, colisiones y manejo de objetos del juego, así como la inicialización y actualización de la interfaz gráfica. Cada componente y función tiene un rol específico que contribuye a la dinámica del juego. Haciendo este juego eficaz, eficiente y dinámico.

**Objetivos**

1. El primer objetivó de este proyecto es: poder utilizar las técnicas de recursividad para la creación de eventos y dinámicas del video juego. Así como funciones para una mejor funcionalidad de nuestro juego. Lo cual es todo un reto debido a que se nos limita un poco el poder ahorran líneas de código con el uso de whiles o for, pero como en este caso usaremos recursividad tendremos que ser más explícitos a la hora de programar.
2. El segundo objetivo es: utilizar las interfaces gráficas, para crear estas mismas de una forma intuitivas, dinámicas y entretenidas para los jugadores. Además de agregarle un toque personalizable a nuestro juego. Este objetivó es uno de los más importantes ya que en los video juegos es muy importante su presentación y su dinámica con los usuarios.
3. El tercer objetivo es: El poder simular físicas en el video juego como la gravedad y el rebote de esferas. Es una de las mayores dificultades para crearse, ya que se implementan funciones con cierta realidad en un video juego esto haciendo que las esferas se vean de una manera más naturales, como si de un balón de fútbol se tratase.
4. El cuarto objetivo es: La creación de una documentación del proceso de desarrollo y los resultados que poco a poco se van obteniendo del proyecto. Aquí podemos ver como es el proceso de crear un juego y como poder crear funciones o atributos de un video juego usando tkinter de Python en implementaciones de físicas de vida real a un juego.

**Descripción del problema.**

**Problema**

**Problema** = En el primer problema que tuve en mi video juego fue la función de mis esferas ya que estas debían de tener muchas funciones entre sí. Además de esto tuve muchos problemas en la división de las esferas y sus comportamientos naturales como la gravedad de las esferas y su movimiento sin perder energía a la hora de colisionar con la ventana del juego.

**Elemento clave** = Poder desarrollar un control a las esferas con todas sus características, es decir hacer un código donde mis esferas se puedan crear con un mínimo de radio donde este se dividirá en una esfera más pequeña y así consecutivamente, en adicional le agregamos una función llamada “def move (self )”. Con esta función podremos agregar nuestros movimientos aleatorios manteniendo su energía y controlando sus colisiones con la ventana y objetos además estos serían sus atributos y funciones.

* min\_sphere\_size: Tamaño mínimo de las esferas.
* sphere\_velocities: Velocidades de las esferas.
* spheres: Lista de esferas presentes en el juego.
* create\_spheres(): Crea las esferas del juego.

**Solución propuesta**

* Creamos la función de las esferas
* Creamos el movimiento de las esferas, dentro de la ventana así podemos ver las colisiones para que estas no se salgan de la ventana del juego.
* Definimos la gravedad establecida para que puedan rebotar sin perder energía, estas esferas además de su colisión con la misma ventana en su totalidad, colisiones con el proyectil, sus colisiones con el personaje y su colisión con los obstáculos.
* Definir correctamente el código de las esferas, para que estas aparezcan y aportar su gravedad, además de eso una función que haga el movimiento y que no se caigan al suelo, sino que mantengan su rebote y su energía.
* Se necesita esta solución de este problema ya que no podríamos cumplir con el Objetivo Principal del Juego: Destruir todas las esferas en pantalla para avanzar al siguiente nivel. Y sin esferas no podemos hacer que tengan las otras características ya que a estas mismas se le aplica estas funciones.

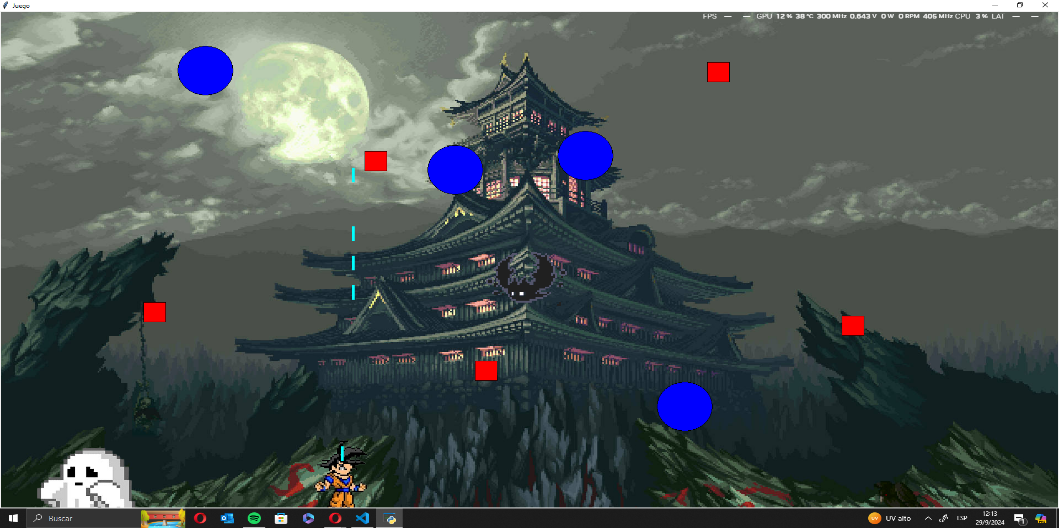
**Análisis de resultados**

1. En primer lugar, podemos ver como nuestras esferas que hemos creado se logran ver en nuestra pantalla de juego, ademas podemos ver como estas misma so se salen de la pantalla es decir que sus colisiones están bien definidas.
2. En segundo lugar, tenemos que las esferas en ningún momento están en el suelo esto nos da a entender que su gravedad está haciendo aplicada correctamente y que su energía no se pierde
3. En el cuarto lugar, tenemos el resultado de cómo se dividen la esfera en esferas con un tamaño de radio menor y está dándonos un ítem que nos da vida en una imagen de un corazón para simular la vida, ademas estas esferas más pequeñas mantienen la gravedad, energía y movimiento en nuestra pantalla de juego.

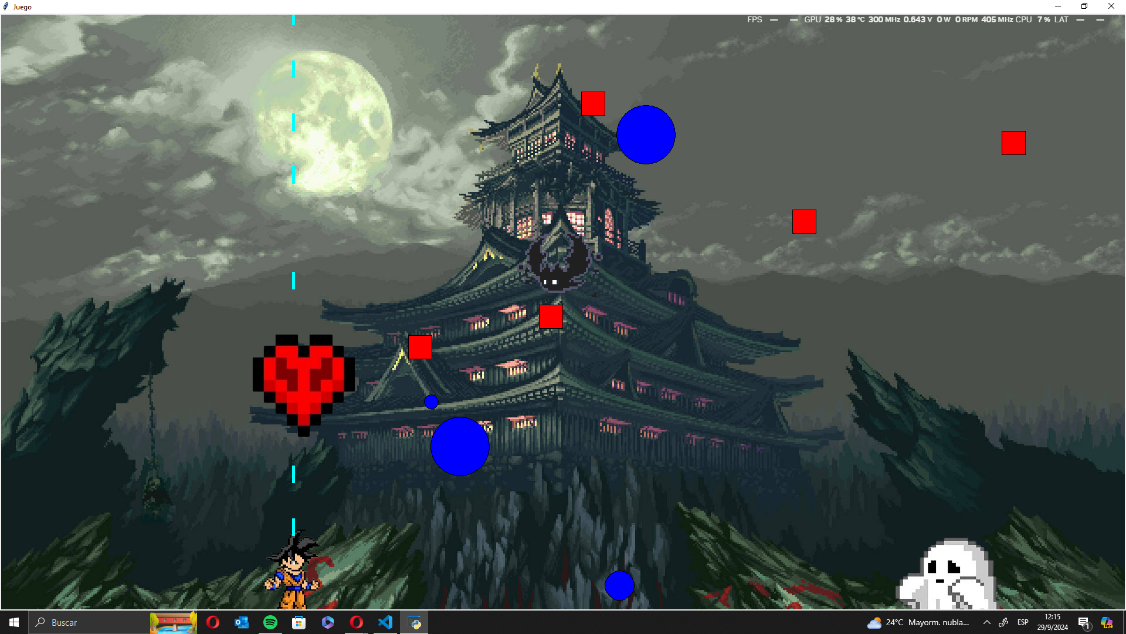
***Imagen del punto 1.***



***Imagen del punto 2.***



***Imagen del punto 3.***



**Dificultades encontradas.**

**Desafío 1**

La primera dificultad fue mi función de mis esferas ya que estas debían de tener muchas funciones entre sí. Además de esto tuve muchos problemas en la división de las esferas y sus comportamientos naturales como la gravedad de las esferas y su movimiento sin perder energía a la hora de colisionar con la ventana del juego.

**Solución y estrategias**

Poder desarrollar un control a las esferas con todas sus características, es decir hacer un código donde mis esferas se puedan crear con un mínimo de radio donde este se dividirá en una esfera más pequeña y así consecutivamente, en adicional le agregamos una función llamada “def move (self )”. Con esta función podremos agregar nuestros movimientos aleatorios manteniendo su energía y controlando sus colisiones con la ventana y objetos ademas estos serían sus atributos y funciones.

* min\_sphere\_size: Tamaño mínimo de las esferas.
* sphere\_velocities: Velocidades de las esferas.
* spheres: Lista de esferas presentes en el juego.
* create\_spheres(): Crea las esferas del juego.

**Solución propuesta**

* Creamos la función de las esferas
* Creamos el movimiento de las esferas, dentro de la ventana así podemos ver las colisiones para que estas no se salgan de la ventana del juego.
* Definimos la gravedad establecida para que puedan rebotar sin perder energía, estas esferas además de su colisión con la misma ventana en su totalidad, colisiones con el proyectil, sus colisiones con el personaje y su colisión con los obstáculos.

Desafió 2

La creación de enemigos en imágenes png me costó mucho ya que en un inicio quería hacerlos con gifs así estos tienen un movimiento, pero no logre poder hacerlos que queden de una forma óptima ya que había que cargar sus frames y la verdad se me dificulto demasiado.

**Solución y estrategias**

Mi estrategia fue simple no utilizar los gifs y usar imágenes a ver si era el codigo mi error o era por los frames que no se lograba cargar.

Solución Propuesta

Para los enemigos no utilizar gifs ya que se me pegaba todo el juego en completo y para no perder mucho tiempo, cambié el código a imágenes png, más fáciles de importar en tkinter.

Bitácora de actividades.

Inicio del Proyecto: 13/09/2024

Fecha de finalización del Proyecto: 29/09/2024

13 de Septiembre de 2024

**Actividad Realizada:**  Importamos todas las librerías a nuestro código, ademas de crear la carpeta de todos los archivos necesarios para la creación del juego usando gifs, imágenes y archivos de mp3.

**Tipo de Actividad:**  Importación de todas las librerías y archivos necesarios para la creación del juego.

14 de Septiembre de 2024

**Actividad Realizada:** Creación de la funciones, start\_game, initialize\_game\_window, class sphere , estas teniendo sus características, funcionalidad, lógica y sus respectivos códigos. Estos para la creación de la ventana principal de nuestro juego, la pantalla de juego y sus esferas.

**Tipo de Actividad:**  Creamos las primeras funciones es agregando sus características y sus requisitos utilizando recursividad.

15 de Septiembre de 2024

**Actividad Realizada:**  Creamos todas las funciones con relación a nuestras esferas, donde se necesitaba el agregar su gravedad, movimiento, colisiones de pantalla, colisiones con nuestro personaje, colisiones con el proyectil, generación del ítem de vida cuando una esfera se rompa, función de división de esferas, función de movimiento de esferas aleatoriamente y las físicas de las esferas.

**Tipo de Actividad:**  Creamos todas las funciones con base a nuestras esferas y sus objetivos en nuestro juego ademas de sus características.

18 de Septiembre de 2024

**Actividad Realizada:**

**Tipo de Actividad:**  Creación completa de las esferas, personajes, fondos de pantalla, fondo de pantalla de niveles ademas de sus características de aumentar su dificultad cada vez más a la medida que se va avanzando.

19 de Septiembre de 2024

**Actividad Realizada:** Configuraciones de las ventanas de cada nivel ademas de la creación de los botones de cada pantalla, es decir: de la pantalla de inicio, pantalla de niveles, pantalla de juego, pantalla final, la pantalla de game over y la pantalla de mejores puntuaciones.

Agregamos los dos enemigos. El primero es el aéreo el cual consiste en que hace una caída de picada a intentar golpear el personaje, para eso tuvimos que hacer funciones de movimiento de caída y de regreso colisiones de pantallas, colisiones de nuestro personaje con nuestro enemigo aéreo ademas de crear una llamada recursiva a PIL para la implementación de la imagen de nuestro enemigo en el juego. También debimos hacer que el enemigo se elimine cuando un proyectil le dé.

Ahora lo que hice con el enemigo terrestre fue más simple, solo que se mueva de un lado a otro dentro de la ventana del juego, y que este se pueda eliminar con un proyectil solo que cuando eso pase este nos de munición así poder obtener más municiones. Sin olvidar que le haga daño a nuestro personaje cuando este lo golpe

**Tipo de Actividad:**  Configuraciones de todas las pantallas, configuraciones e creaciones de enemigos.

23 de septiembre de 2024

**Actividad Realizada:** Descripción del código, creaciones de prueba en esta sección le pedí a mis familiares que probaran el juego para una retroalimentación, de que podría mejorarse.

Cambios que se hicieron

* Las esferas de un tamaño más grandes
* Creación de gifs una mejora más intuitiva a la hora de jugar
* Agregar archivos png a mis enemigos
* Configurando el game over.
* Correcciones Sintaxis
* Descripción del código

**Tipo de Actividad: Descripciones del código y prueba de juego además de opiniones de mejora.**

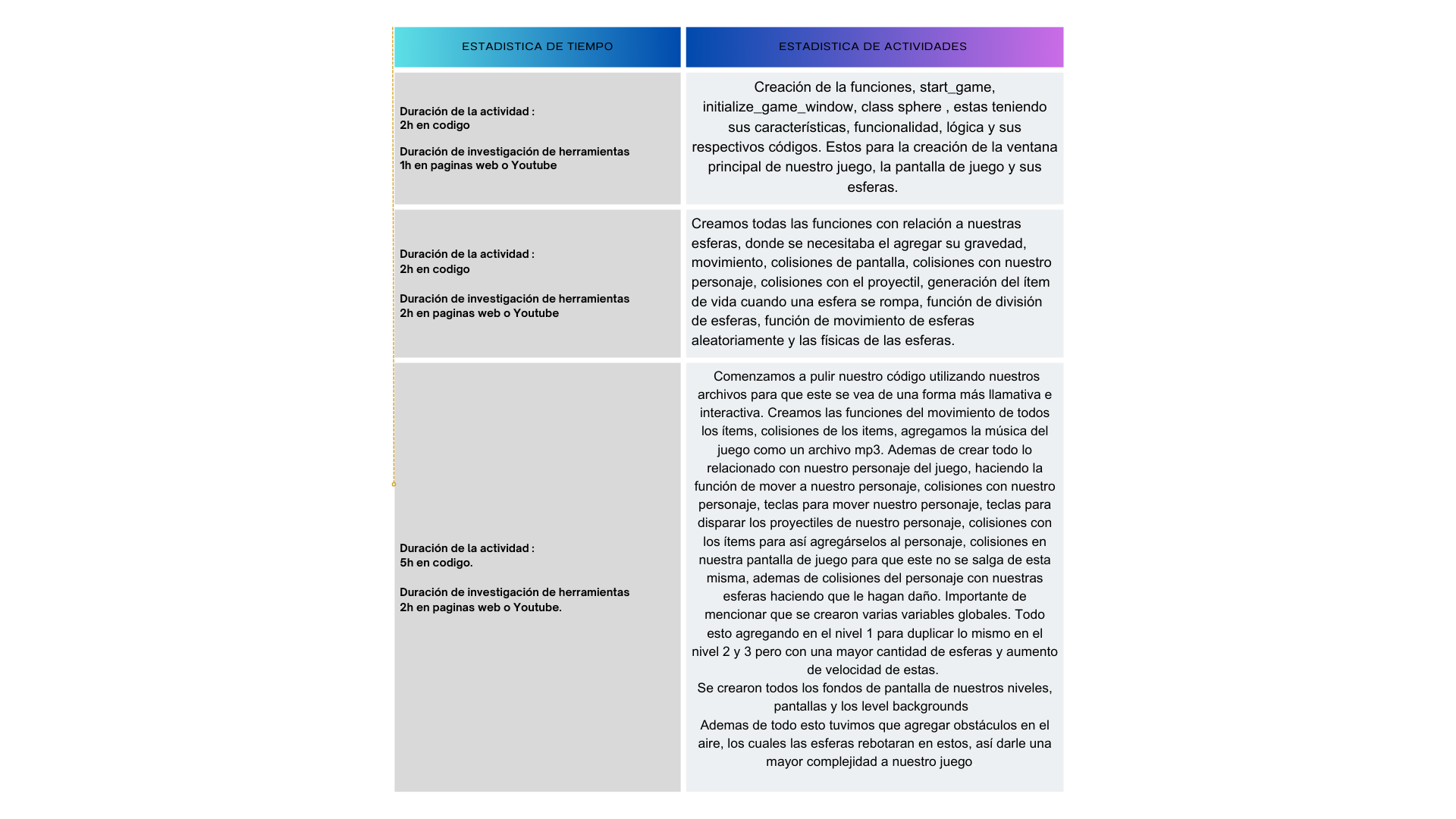
27 de Septiembre de 2024

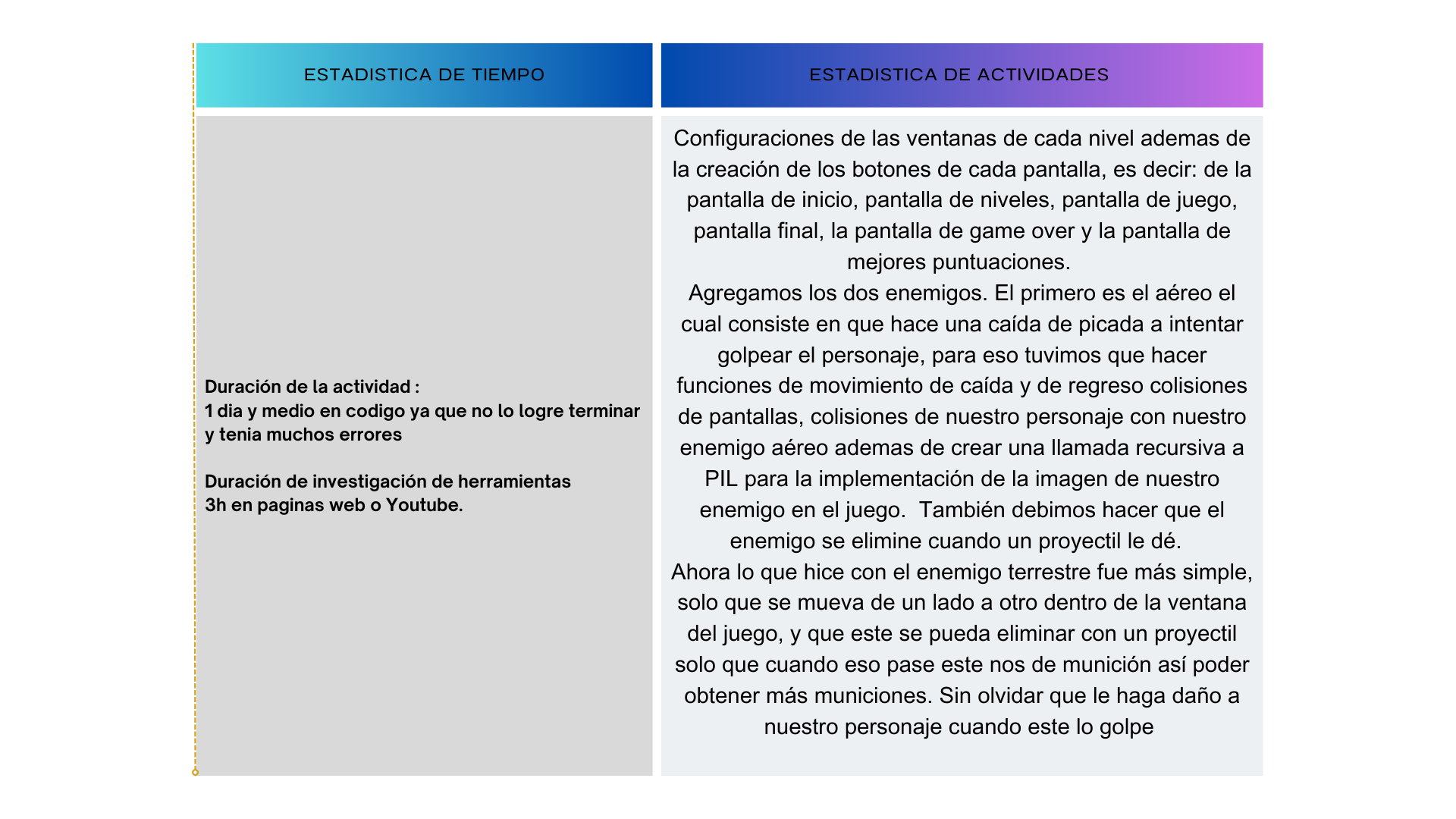
**Actividad Realizada:**  Creación de mi documentación la cual necesitaba tener los siguientes puntos

* Introducción.
* Descripción del problema.
* Análisis de resultados.
* Dificultades encontradas.
* Bitácora de actividades.
* Estadística de tiempos.
* Conclusión
* Literatura o fuentes consultadas.

**Tipo de Actividad:**  Creamos nuestra documentación ademas de nuestro video explicativo.

***Estadística de tiempos.***





Conclusión. En esta sección se resumen los logros alcanzados, lecciones aprendidas y posibles áreas de mejora, dando una perspectiva general del proyecto.

Logros alcanzados

1. El primer logro que se pudo completar fue el objetivo principal del juego: Destruir todas las esferas en pantalla para avanzar al siguiente nivel.
2. Movimiento del personaje se logró completar y hacer intuitivo con las clásicas teclas WASD
3. Físicas de las esferas ya que estas tienen su propia energía, gravedad y su rebote.
4. Fragmentación de las esferas, todas y cada una se dividen cuatro veces
5. Podes hacer que la vida del jugador se pierda o se gane es decir que si se logró la vida del jugador.
6. Los enemigos nos dan vida y municiones lo cual cumplimos con el objetivo de obtener municiones.

Lecciones Aprendidas

El programar te da un análisis muy diferente de ver las cosas, ademas de por primera vez pude crear mi propio juego, ya que nunca había programado así.

Una de las lecciones que más me enseño este proyectó es el ser más organizado con las cosas para así poder cumplirlas ademas de que me enseñó a como crear interfaces graficas utilizando tkinter y sin usar iteración.

Posibles áreas a mejora

* La creación de enemigos
* Aumentar la práctica de programar
* El crear gifs en interfaces graficas
* El usar canvas en tkinter.
* Organización de codigo

Perspectiva General

En este Proyecto logramos crear un juego el cual jamás me imagine recrear y ademas de eso personalizarlo a mi gusto fue lo más divertido ya que podia crear cosas a mis gustos. También el gran aprendizaje que me deja este proyecto al programar me resulta más fácil que al inicio del curso, ya que me llevo mucho tiempo de investigación y logre saber que herramientas me sirven para aprender y practicar la programación.

***Literatura o fuentes consultadas.***

**1.Parte videos o paginas web**

Guia de tkinder <https://guia-tkinter.readthedocs.io/es/develop/>

Addin Images <https://www.youtube.com/watch?v=d2iMDXB27fI>

Moving Images <https://www.youtube.com/watch?v=u7zfpCPZ7bM>

Collision Detection <https://www.youtube.com/watch?v=_flFVAtmkoQ>

Keeping Score <https://www.youtube.com/watch?v=TosEDC0Y7E4>

Continuous Motion <https://www.youtube.com/watch?v=e3Mx6UGX13s&t=2s>

Controlled Start <https://www.youtube.com/watch?v=XmAn2y_iFAs>

Start Screens <https://www.youtube.com/watch?v=8vGzGgU6JH8>

Threading and Sound <https://www.youtube.com/watch?v=1IwjUiMCs0s>

Animation <https://www.youtube.com/watch?v=dF0OtdYVi_c>

Tutorias de guias de tkinter <https://www.tutorialspoint.com/index.htm>

Learn Python multiple animations <https://youtu.be/qK8Pfll5ha8?si=jb3LFa3ogO1GV98S>

Todo python tkinter [https://www.youtube.com/playlistlist=PLZPZq0r\_RZOOeQBaP5SeMjl2nwDcJaV0T](https://www.youtube.com/playlist?list=PLZPZq0r_RZOOeQBaP5SeMjl2nwDcJaV0T)

INTERFACES GRÁFICAS <https://youtu.be/MpkTYMzhV0A?si=NSIE30IrkVqB1jv6>

Tutorias de Intro a la Programacion de Rodrigo <https://github.com/lVoidi/Tutorias-ce1101/blob/main/Ejemplos/tkinter-barra.py>

Informacion de todo un poquito/ videos explicativos y codigo <https://www.geeksforgeeks.org/minsize-method-in-tkinter-python/>

***Parte 2 ,sacado de páginas web***

Python tkinter

Informacion sacada de ----------- https://www.tutorialspoint.com/index.htm

Cursores para tkinter

para aplicar en = playground.config (cursor= "arrow")

"arrow"

"circle"

"cross"

"dotbox"

"exchange"

"fleur"

"heart"

"heart"

"man"

"mouse"

"pirate"

"plus"

"shuttle"

"sizing"

"spider"

"spraycan"

"star"

"target"

"tcross"

"trek"

"watch”

Fuentes de texto en tkinter

Informacion sacada de-------https://docs.python.org/es/3.9/library/tkinter.font.html

Helvetica

Arial"

Times

Courier

Comic Sans MS

Georgia

Impact

Cambio de bordes de tkinter

Informacion sacada de-------https://guiatkinter.readthedocs.io/es/develop/chapters/7-options/7.1-Intro.html sunken

raised

groove

ridge

flat

state