

– Introducción:

Battle Elves es un proyecto de un videojuego desarrollado en Python cuyo objetivo es defender una fuente mágica del ataque constante de duendes. A lo largo del desarrollo, se utilizó Python con librerías, así como conceptos teóricos aplicados en la lógica del juego y su desarrollo. La interfaz fue realizada con la librería *tkinter*, aprovechando su método *canvas* para crear los elementos visuales. El proyecto integra también almacenamiento y encriptación de datos, lo que permitió guardar los resultados del jugador de forma segura. Para lograr completarlo hay que superar los dos niveles en un tiempo de 4:15 minutos, primero 1:15 y segundo 3:00

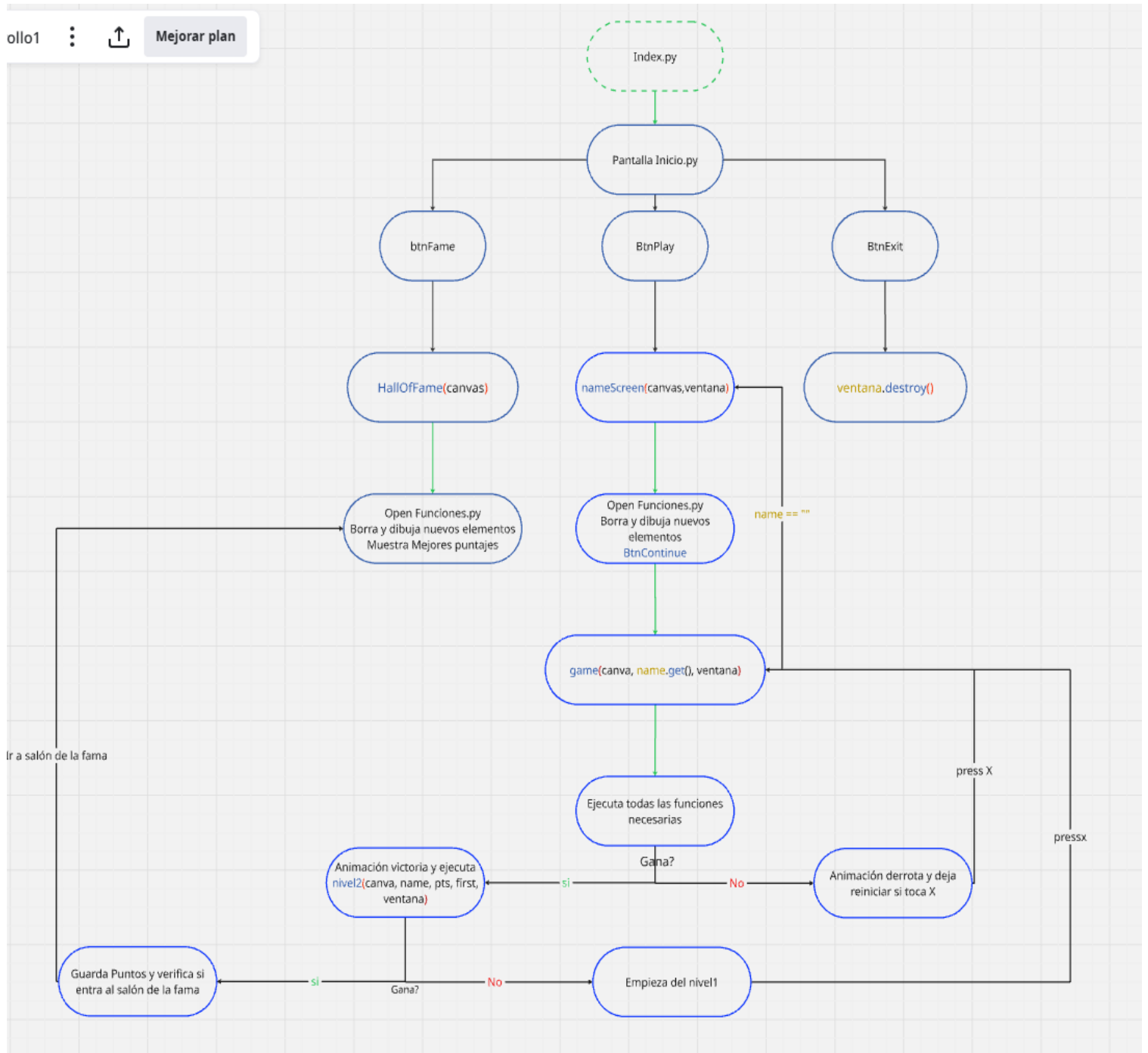
– Marco teórico:

Para desarrollar este proyecto fue necesario aplicar distintas bases teóricas como la recursividad para gestionar ciertos comportamientos como animaciones, disparos, movimientos y otros más, matemáticas básicas (especialmente el cálculo de distancias entre puntos para el manejo de algunas colisiones necesarias), e iteraciones para algunos ciclos del juego y además para la generación de terreno y decoraciones. Además, se utilizó manejo de archivos para guardar la puntuación y nombre del jugador, y la librería *pickle* para la encriptación de resultados. También se hizo uso de la librería *random* para generar eventos aleatorios como la aparición de enemigos y su movimiento, se usó *Poo* para el desarrollo de dos clases principales, *player* y *enemy*, las cuales tienen todos los atributos necesarios y métodos para hacer más práctico el manejo de eventos y datos del jugador y enemigos. La teoría fue acompañada con documentación de Python y *Tkinter*, conocimientos aprendidos en clase y fuentes como *2025 El Libro De Python*, una web muy completa sobre python.

– Desarrollo:

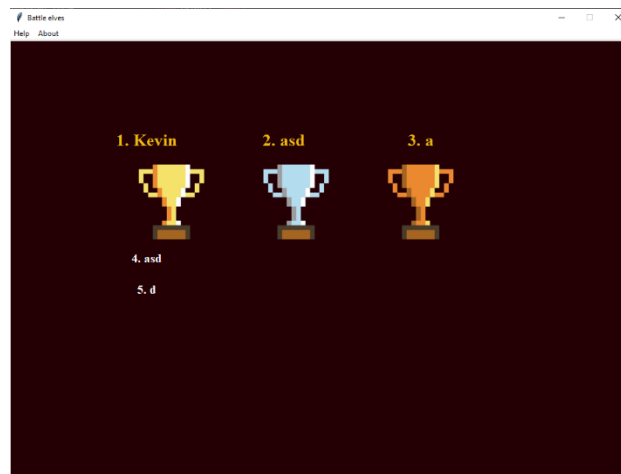
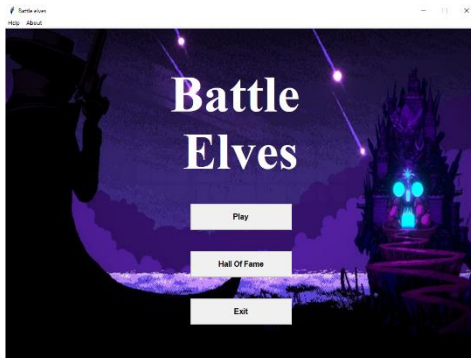
El proyecto fue hecho con programación modular por lo que hay varios archivos .py, el principal es *index.py* donde crea la ventana principal los botones y menú, para hacer esto llama a la función *pantallaInicio* la cual coloca todo en el canvas, si se da click en inicio se llama la función *screenName* la cual está en funciones y es importada, dicha función limpia el canvas y muestra un entry, botón de continuar y texto, si el jugador ingresa su nombre y le da a al botón ejecuta la función *game()*, la cual es el nivel 1, esta función se encarga de cargar todas las imágenes necesarias las agrega a una lista para que no se pierdan y continua generando la decoración y mapa así como todos los elementos, como tiempo, llamando a *update_timer* la cual va actualizando el reloj con un contador en segundos, el personaje, creando una instancia de la clase *Player* y llamando todos sus métodos necesarios para su creación y también la generación de enemigos que para ello llama *enemyGenerate*, esta se encarga de generar a los enemigos cada 3s, solo si no están, luego si el jugador logra aguantar 1:15min entonces gana el nivel 1 y le salta la animación de victoria, esto está dentro de *update_timer*, ya que una vez es igual a cero cumple otra condición, luego de 4s ejecuta *nivel2()* esta función es igual a *game* pero cambian varias cosas, el tiempo, que ahora son 3min, la velocidad de los enemigos, la velocidad de disparo, la apariencia, la cantidad de vidas de los enemigos (ahora dos hits) e igual que la otra al ganar ejecuta la animación de victoria en *update_timer*, solo que esta vez verifica si a entrado a el salón de la fama llamando a la función *cargarDatos* y *guardarDatos*, para ver si

el puntaje obtenido es superior a alguna que ya este o si no hay ninguno o si hay menos de 5.



– Resultados:

Resultados del proyecto.



– Conclusiones:

A través del desarrollo de *Battle Elves*, fue posible aplicar de forma práctica distintos conocimientos adquiridos en clase y diferentes investigación realizadas por mi parte, desde la implementación de interfaces hasta la lógica matemática para detectar colisiones con paredes, balas y movimientos. Además, se logró integrar conceptos como la encriptación y el almacenamiento de datos en un entorno de entretenimiento, mostrando la gran cantidad de cosas que ofrece Python. El uso de *tkinter* demostró ser una herramienta eficaz para crear una experiencia gráfica interactiva y agradable sin necesidad de herramientas externas complejas.

– Bibliografía:

- Documentación oficial de Python (<https://docs.python.org/>)
- 2025 El Libro De Python (<https://ellibrodepython.com>)
- Apuntes y referencias del profesor en clase