

Instituto Tecnológico de Costa Rica

Sede Limón

Proyecto Programado 3

Taller de Programación (GR60)

Código: IC-1803

Profesor:

Jean Carlos Miranda Fajardo

Estudiantes:

Sebastián Sánchez y Hackney Aguilar

I Semestre

2025

Tabla de contenido

Manual de Usuario.....	1
1. Introducción.....	1
2. Requisitos del Sistema.....	1
3. Uso del Programa	1
Descripción del problema.....	11
Diseño de software.....	11
Librerías utilizadas	16
Análisis de resultados	17
Conclusiones	19

Manual de Usuario

Proyecto Programado 3: El gran Torneo

Por: Sebastián Sanches y Hackney Aguilar

1. Introducción

Bienvenidos al manual de usuario del III proyecto programado “El gran Torneo”, el cual tiene como objetivo brindar al usuario una guía clara del funcionamiento de las funciones principales del programa, se mostrarán las instrucciones de compilación, ejecución y uso de este. Este programa está basado en un juego de luchas, el ganador de cada lucha se elige mediante las estadísticas elegidas, se comparan y el que tenga mas puntos gana el round. El ganador del torneo es el que tenga más luchas ganadas.

Se tomo en cuenta lo que es el mundo de los comics en el cual existen mucha variedad de héroes y villanos, además de los antihéroes, por razones que se desconocen sucedió una gran controversia, entre los defensores de la justicia y los de la famosa frase “el fin justifica los medios”, por lo tanto, surgió un gran torneo y así saber de una vez por todas quiénes eran los mejores.

2. Requisitos del Sistema

- Tener Python instalado sea la versión 3.10 en adelante
- Tener Sistema Operativo Windows, Linux o MacOS

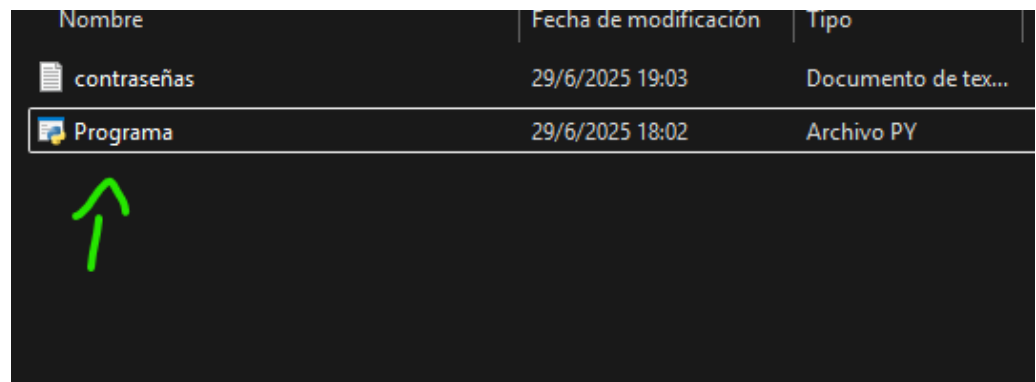
3. Uso del Programa

A continuación, se mostrará como utilizar el programa, las imágenes utilizadas como referencias son de un equipo con el sistema operativo Windows 11 Pro.

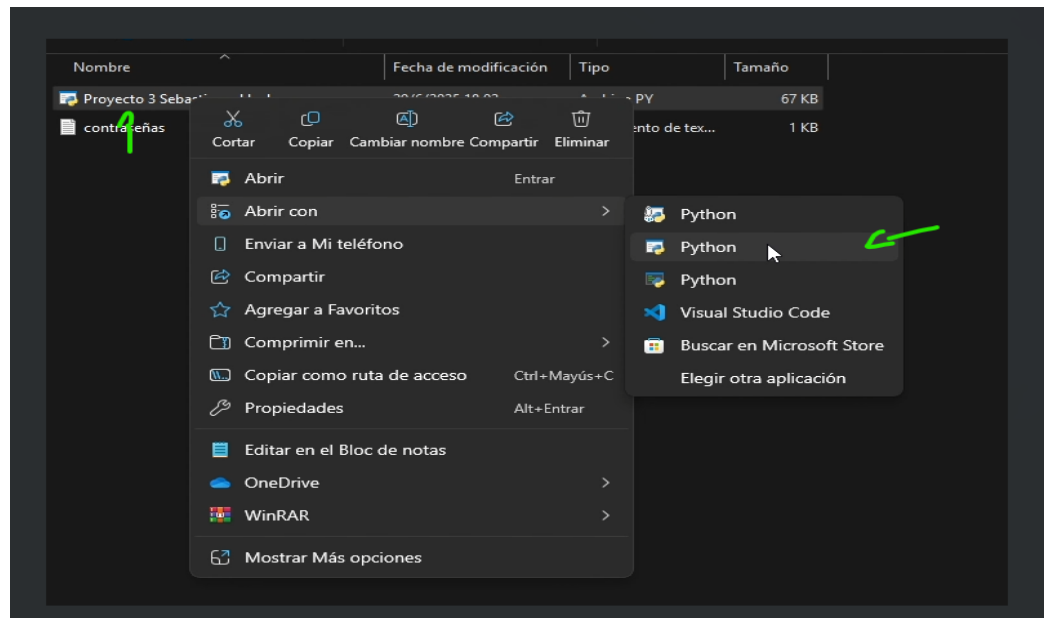
1. **Primer paso descomprimir archivo y su contenido:** Descomprima el archivo llamado “P3_Sebastian_Hackney.zip”, al hacerlo, se mostrará una carpeta con el mismo nombre, en esta carpeta, se encontrarán dos carpetas, la primera se llama “**Documentación**” “contiene la documentación del proyecto y la segunda el “**Programa**” que contiene el programa con sus respectivos archivos

Además, de mostrarse un archivo .txt, que se llama info.txt
2. **Segundo paso ingreso a la documentación:** Al ingresar a la carpeta documentación se encontrará un archivo de Word(Documentación), dentro de este se encontrar la documentación correspondiente,
3. **Tercer paso ingreso al programa:** Al ingresar a la carpeta del programa, esta contiene el archivo(.py) llamado “Proyecto” con el programa.

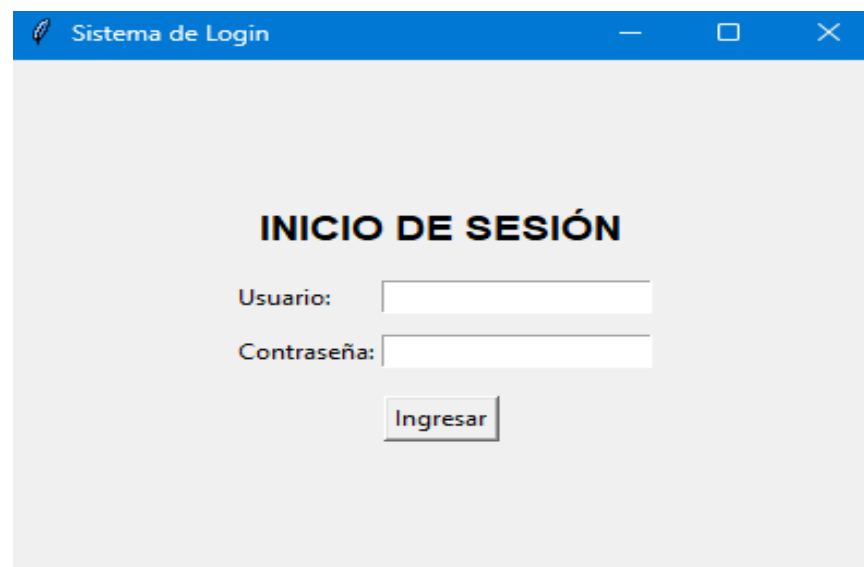
- **Imagen 1**, ingresar a la carpeta programa se muestra el archivo .py, llamado “programa”.



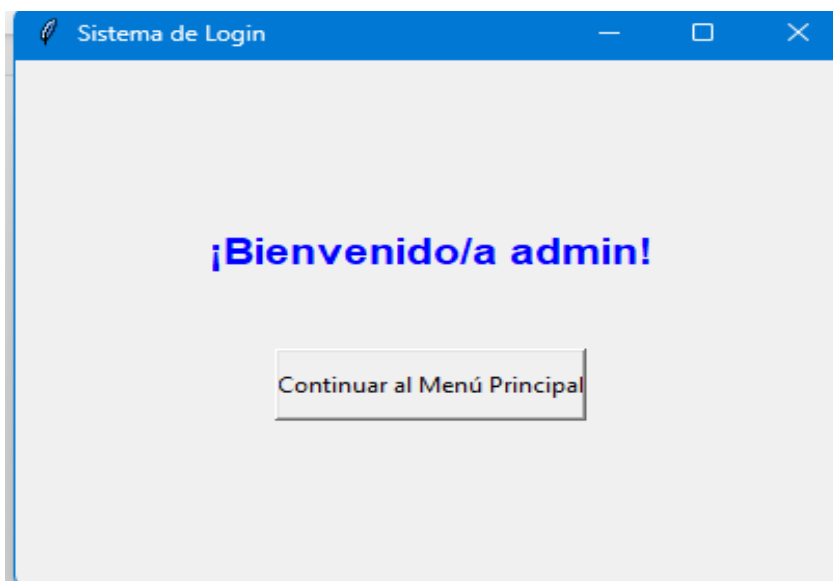
- **Imagen 2**, localizar el programa. Se le da clic derecho al archivo .py, luego se selecciona la opción "abrir con" y se le da clic al icono con el nombre Python, como se muestra en la pantalla.



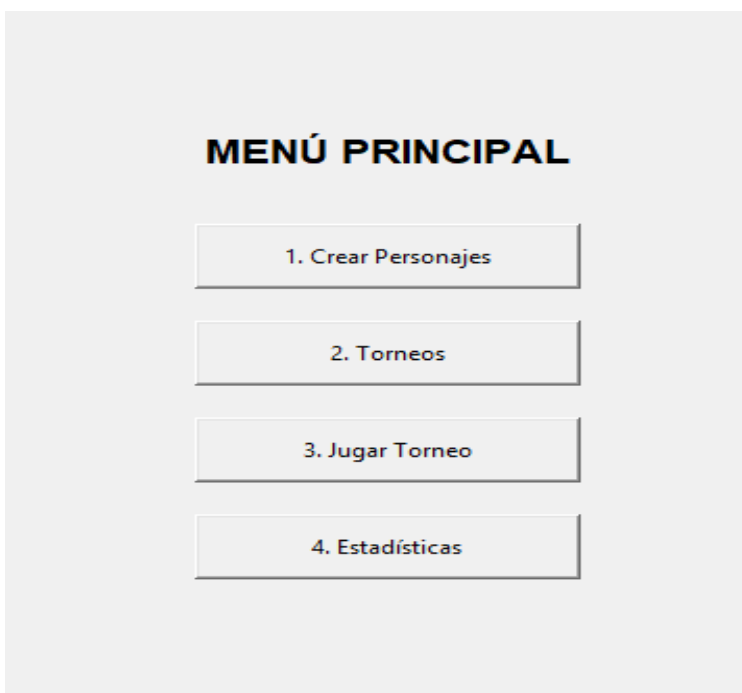
- **Imagen 3**, Una vez abierto el programa, se muestra la ventana del juego, en la cual debemos de ingresar los datos que se solicitan, estos se encuentran en el archivo "contraseñas", y luego damos clic a ingresar.



- **Imagen 4**, Después de darle clic a ingresar se muestra un mensaje con la bienvenida al juego y damos clic al botón “Continuar al menú principal”.



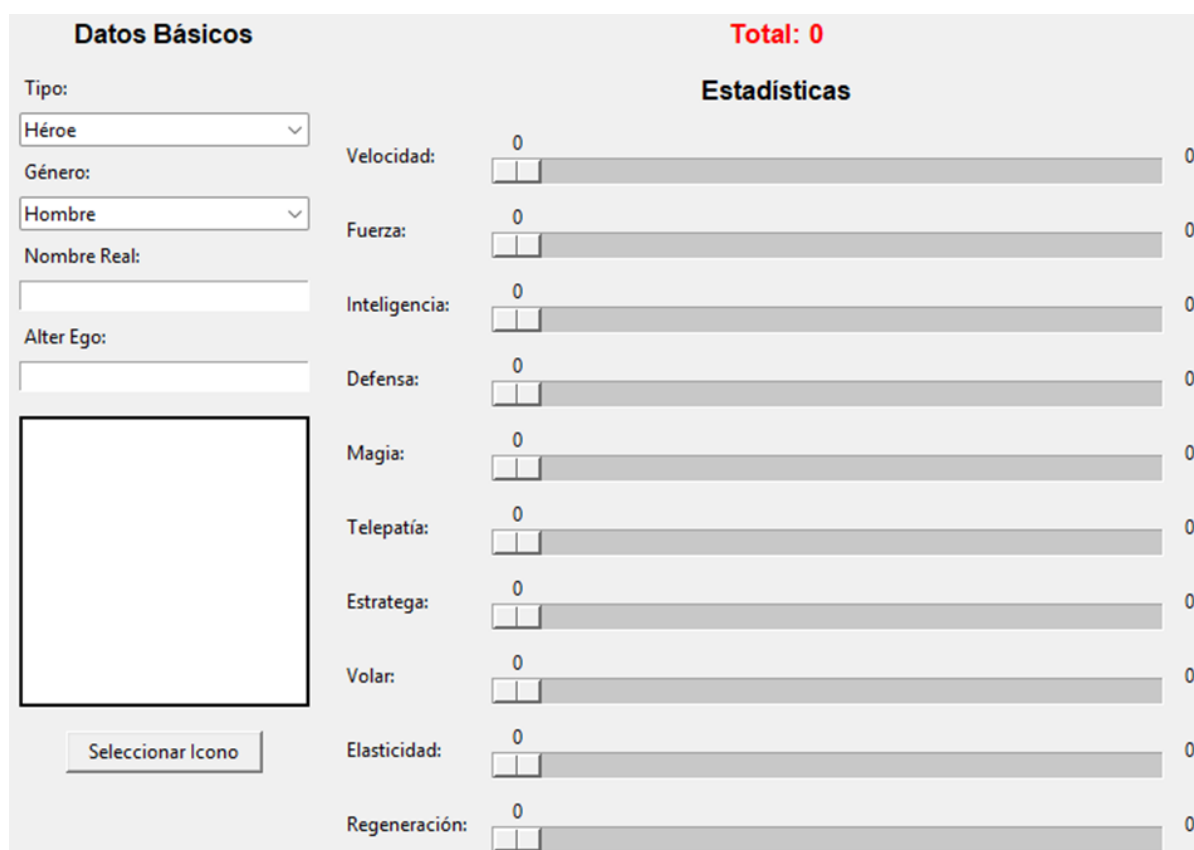
- **Imagen 5**, Se muestra el menú principal el cual contiene 4 opciones, Crear personaje, Torneos, Jugar torneo, Estadísticas.



- **Imagen 6**, En la primera opción crear personaje, se muestra las siguiente opciones mostradas en la imagen.



- **Imagen 7**, En la opción crear nuevo personaje: se nos muestra todas las características que se le debe dar al personaje, la suma de todas las estadísticas no debe ser mayor a 100.



Datos Básicos

Tipo: Héroe

Género: Hombre

Nombre Real:

Alter Ego:

Seleccionar Icono

Total: 0

Estadísticas

Velocidad:	0	0
Fuerza:	0	0
Inteligencia:	0	0
Defensa:	0	0
Magia:	0	0
Telepatía:	0	0
Estratega:	0	0
Volar:	0	0
Elasticidad:	0	0
Regeneración:	0	0

- **Imagen 8,** Opción lista de personajes se nos muestra todos los personajes creados.



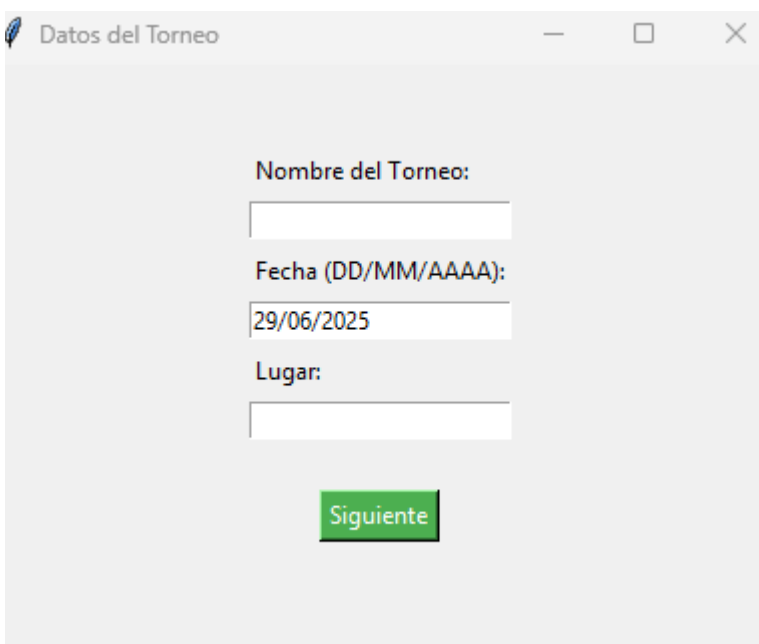
- **Imagen 9,** La opción eliminar personaje se elige el personaje que se desea eliminar.



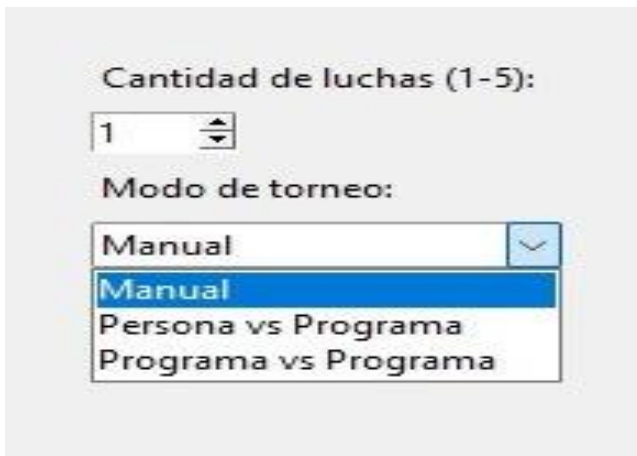
- **Imagen 10**, En el menú principal(imagen 5), la opción Torneos nos muestra tres opciones como se ven en la siguiente imagen.



- **Imagen 11**, La opción crear torneo, se deben llenar los datos solicitados:

Una ventana de aplicación titulada "Datos del Torneo" con botones de control de ventana (minimizar, maximizar, cerrar). El formulario contiene tres campos de entrada: "Nombre del Torneo:" con un campo vacío, "Fecha (DD/MM/AAAA):" con el valor "29/06/2025", y "Lugar:" con un campo vacío. Debajo de los campos hay un botón verde con el texto "Siguiente".

- **Imagen 12,** Una vez llenado le damos siguiente y se nos muestra lo siguiente, lo cual debemos de seleccionar la cantidad de luchas, que van de 1 a 5 y los modos de torneo que son tres, luego realizado lo solicitado le damos siguiente para crear el torneo.



Cantidad de luchas (1-5):

1

Modo de torneo:

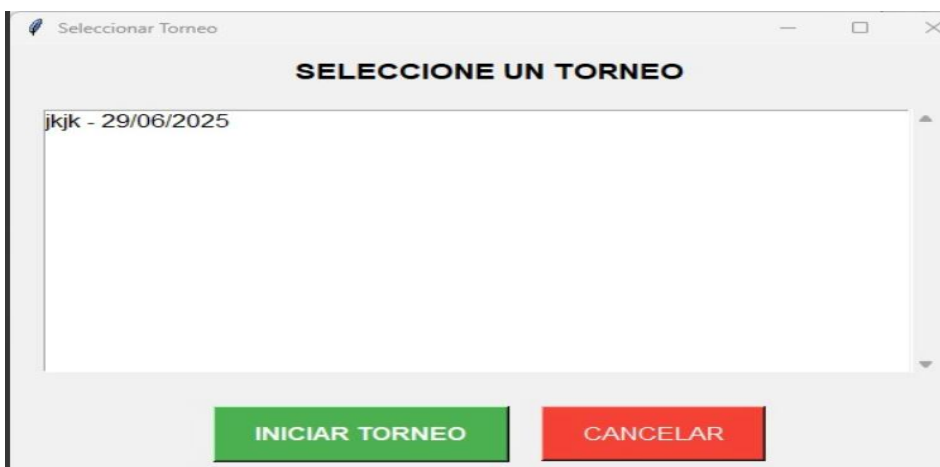
Manual

Manual

Persona vs Programa

Programa vs Programa

- **Imagen 13,** En el menú principal(imagen 5) si elegimos la opción jugar torneo, para esto deberíamos de tener al menos dos personajes creados, debemos de elegir un torneo y luego darle clic a iniciar torneo.



Seleccionar Torneo

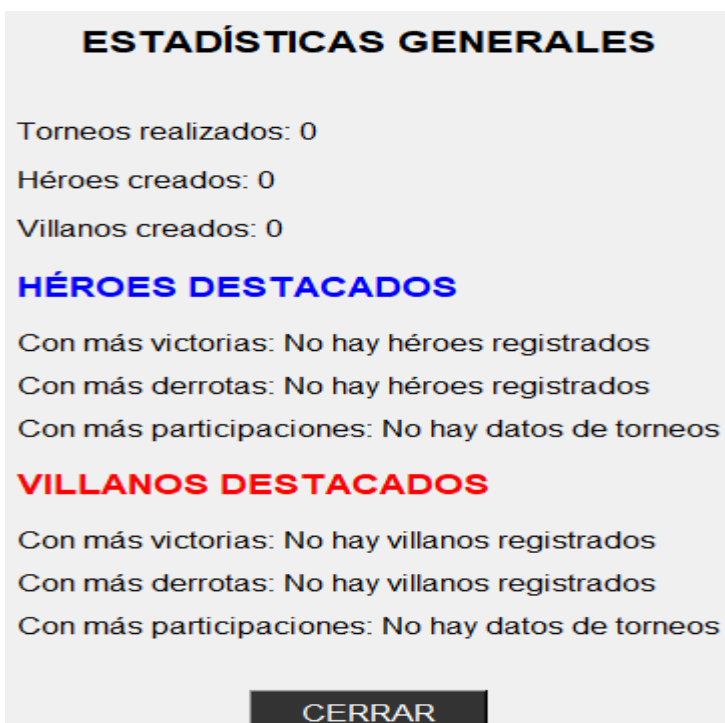
SELECCIONE UN TORNEO

jkjk - 29/06/2025

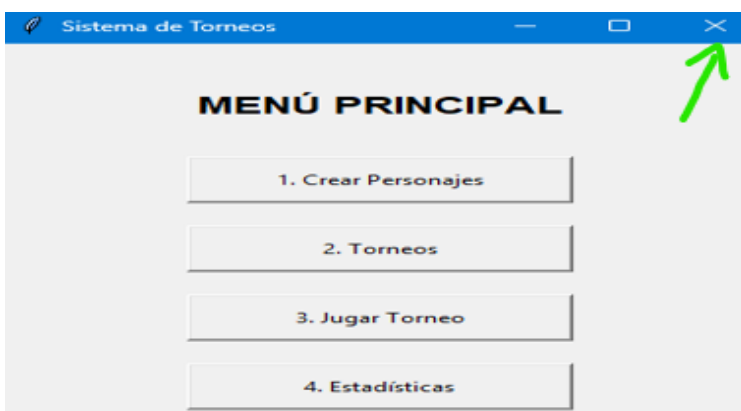
INICIAR TORNEO

CANCELAR

- **Imagen 17**, En el menú principal(imagen 5). La opción estadísticas, se muestran todas las estadísticas de los torneos y luchas.



- **Imagen 18**, para salir del juego simplemente se le da clic a la “x” de la esquina superior derecha de la ventana.



Este es el manual de usuario para el programa, se espera que lo mostrado sea de ayuda al momento de utilizarlo.

Descripción del problema

El presente proyecto consiste en desarrollar un juego en Python en parejas que simule un juego de luchas llamado “El Gran Torneo”, que permite enfrentar a personajes ficticios, definidos por una serie de atributos, para esto se debe de crear diferentes clases, métodos y funciones, para poder llevarlo a cabo. En este proyecto se debe tomar en cuenta lo que es el trabajo en equipo, para llevar a cabo una buena organización al momento de implementar el código del proyecto.

Entre las clases y funciones que se deben de tomar en cuenta son las de crear personajes, control de acceso, lucha, torneo, se debe tomar en cuenta que existen 3 métodos para jugar, el manual(donde el usuario elige los personajes), persona vs programa(la persona elige los luchadores de su bando y el programa el del otro bando), y programa vs programa(el programa elige los luchadores de ambos bandos).

Por otra parte, para definir un ganador se deben tomar en cuenta las estadísticas seleccionadas aleatoriamente, el luchador que tenga más puntos de estadísticas(se compara estadística por estadística de cada luchador), gana el round, los cuales son 3.

Además, se debe tomar en cuenta que la creación de este juego se utiliza el interfaz gráfica, con la librería Tkinter además de la implementación de imágenes con el PIL, por lo tanto, se debe desarrollar tanto las clases y funciones del juego, y como implementar esta con la interfaz gráfica.

Diseño de software

En este proyecto, se tuvo que realizar la implementación de un sistema de gestión de personajes y torneos utilizando clases y la librería Tkinter en el apartado de la interfaz gráfica, por lo tanto, a continuación, se mostraran la descripción de las clases utilizadas:

- **Clase Luchador:** Esta clase representa un personaje (héroe o villano), con atributos y estadísticas:
 - **Atributos:**
 - Tipo: tipo de personaje héroe o villano.
 - Género: genero del personaje.
 - Nombre_real: nombre real del personaje.
 - Alter_ego: Nombre alternativo del personaje.
 - Stats: Diccionario con las estadísticas de los personajes.
 - Foto: ruta de la imagen asociada al personaje.
 - Victorias: Numero de victorias del personaje.
 - Derrotas: Numero de derrotas del personaje.
 - Estadísticas: recibe las estadísticas de los personajes
 - **Metodos:**
 - Calcular el poder total()
 - Luchar_contra(oponente)
- **Clase CreadorPersonajes:** Permite crear el personaje mediante la interfaz grafica
 - **Atributos:**
 - root: ventana principal de la interfaz.
 - callback_guardar: funcion que se ejecuta al guardar el personaje.
 - tipo_var, genero_var, nombre_real_var, alter_ego_var: variables para los datos de los personajes.
 - Stats_vars: diccionario con las variables de estadísticas
 - Icono_path: ruta del icono seleccionado

- **Metodos:**
- crear_interfaz()
- verificar_nombre()
- actualizar_total()
- seleccionar_icono()
- seleccionar_icono_callback (path, top)
- guardar_personaje()
- nombre_existe(alter_ego)
- **Clase SistemaTorneos:** Gestiona el sistema de torneos y personajes.
 - **Atributos:**
 - Root: ventana principal de la interfaz
 - Personajes: lista de personajes cargados desde un archivo
 - Torneos: lista de torneos creados
 - Mostrar_menu_principal: es para volver al menú principal
 - **Métodos:**
 - cargar_personajes()
 - actualizar_personajes()
 - actualizar_archivo_torneos()
 - mostrar_menu_principal()
 - menu_personajes()
 - abrir_creador_personajes()
 - listar_personajes()
 - mostrar_detalle_personaje()

- mostrar_pantalla_borrado()
 - confirmar_borrado()
 - opciones_torneo()
 - mostrar_pantalla_borrado_torneos()
 - pantalla_crear_torneo_datos()
 - pantalla_crear_torneo_cantidad(nombre, fecha, lugar)
 - pantalla_seleccion_bando(nombre, fecha, lugar, cantidad, modo)
 - pantalla_seleccion_iconos(nombre, fecha, lugar, cantidad, modo, bando, manual)
 - pantalla_seleccion_iconos_otro_bando(nombre, fecha, lugar, cantidad, modo, seleccionados_primeros, candidatos)
 - crear_torneo_final(nombre, fecha, lugar, cantidad, modo, heroes_sel, villanos_sel)
 - jugar_torneo()
 - _jugar_torneo_interactivo(torneo)
 - _jugar_siguiente_lucha()
 - _mostrar_round()
 - _mostrar_resumen_torneo()
 - mostrar_estadisticas()
- **Clase Inicio_sesion:** Se utiliza para manejar el sistema de inicio de sesión y registro de usuario.
 - **Atributos:**
 - Root: ventana principal de la interfaz
 - **Metodos:**
 - crear_archivo_por_defecto()

- crear_widgets_login()
- validar_login()
- mostrar_bienvenida()
- abrir_sistema_torneos()
- jugar_torneo()
- **Clase lucha:** Simula la lucha entre dos personajes
 - **Atributos:**
 - primer_luchador, segundo_luchador: nombres de los alter ego de los personajes.
 - nombre_torneo: Nombre del torneo.
 - ganador_round_1, ganador_round_2, ganador_round_3: ganadores de cada round.
 - Ganadador_lucha: ganador final de la lucha.
 - **Atributos:**
 - registrar_ganador(round_num, ganador)
 - realizar_lucha()
 - determinar_ganador()

Librerías utilizadas

A continuación, se mostrarán las librerías que se utilizaron para la elaboración de este proyecto:

1. **Tkinter:** Para proporcionar la interfaz grafica de usuario, sean la ventana, los botones, cuadros de diálogos.
2. **Pil(Pillow):** carga y redimensiona y muestra imágenes en la interfaz Tkinter.
3. **Random:** Selecciona de forma aleatoria las estadísticas y los personajes en los modos de juego, persona vs programa y programa vs programa.
4. **Datetime:** Se utiliza para la agregar la fecha del torneo.
5. **Os:** Esta librería se usó para lo que es el manejo de archivos.

Análisis de resultados

Objetivos Alcanzados

1. **Aplicar programación orientada a objetos:** Se crearon las clases con los atributos y métodos necesarios, para tener un código bien estructurado.
2. **Crear una interfaz gráfica funcional:** Se utilizó la librería Tkinter para lograr la creación de la interfaz gráfica para el programa.
3. **Integrar imágenes con el PIL:** Se implementó con éxito la carga y visualización de las imágenes para representar los personajes.
4. **Implementar diferentes modos de juego:** Se logró programar los tres modos de juego.
5. **Diseñar y gestionar luchas y torneos:** Se crearon mecanismos para registrar las luchas, definir ganadores y guardar los resultados.
6. **Control de acceso por usuario y contraseña:** Se programó un inicio de sesión seguro que almacena los datos en un archivo.
7. **Guardar la información:** Se utilizó la compresión de archivos para guardar los personajes, información de las luchas, los torneos y las estadísticas.
8. **Trabajar en equipo y organización:** Se logró tener una buena coordinación con el grupo cumpliendo con los avances y la planificación.
9. **Validaciones e interacciones:** Se realizaron las validaciones necesarias para evitar errores de usuario.

Objetivos no Alcanzado

Realizar las funciones extras del proyecto: no se logró realizar las funciones extras para el proyecto, por falta de tiempo.

Entregar a tiempo del proyecto: se entregó el proyecto a la segunda hora de entrega, por falta de validaciones y clases que no se habían realizado.

Decoración del juego: No se le agrego imágenes en los menús al programa, por falta de tiempo

Guardar Torneo: Un error que no permite guardar los torneos, se desconoce el porque no se guarda el torneo además por la falta de tiempo no se pudo corregir de manera adecuada.

Conclusiones

En el presente proyecto, se trabajó en forma colaborativa, se logró comprender como es realizar este tipo de proyectos de programación junto a otra persona, recalcando lo que es la comunicación y retroalimentación entre sí, para poder tener una buena planificación y orden al momento de implementar el código, con los diversos enfoques de cada persona.

Además, a lo largo de este proyecto se logró entender más lo que son los conceptos básicos de clases y la programación orientada a objetos, de esta manera, siendo esto una experiencia que contribuyo significativamente en una comprensión y aplicación de estos conceptos que son fundamentales en programación.

Por otra parte, se investigó la librería Tkinter, al momento de implementar lo que es la interfaz gráfica del código, gracias a los conocimientos anteriores de los proyectos realizados, se pudo llevar a cabo un mejor uso de esta librería con mucho más facilidad, además que en este proyecto se tuvo investigar la librería PIL de Python, para poder realizar el proyecto, se eligió esta librería porque su uso facilito lo que es la implementación de imágenes al momento de realizar la luchas.

También, el desarrollo de este proyecto ha permitido consolidar conocimientos claves en lo que es la programación orientada a objetos, en la comprensión archivos de texto para tener persistencia en los en la información, validaciones, control de acceso, tomar aleatoriamente los elementos necesarios para las luchas, y lo que fue el análisis de estadísticas de los resultados, de esta manera, permitiendo integrar diversos conocimientos que se han visto a lo larga del curso.

Para finalizar, este proyecto nos ayudó a comprender y a mejorar como debe ser el trabajo en equipo en programación, aprendiendo lo importante organizarnos, investigar por nuestra cuenta y encontrar soluciones a los problemas que iban surgiendo. Por lo tanto, este proyecto a sido una experiencia enriquecedora para tomar en cuenta más adelante como futuros ingenieros.