

Introducción:

Para este proyecto se nos solicitó desarrollar un simulador de gimnasio Pokémon, con batallas por turnos entre entrenadores. El propósito principal fue aplicar los conceptos de programación orientada a objetos (POO), especialmente la composición, el encapsulamiento y la separación de responsabilidades.

Decisiones del diseño:

Durante la etapa de diseño se discutieron diferentes formas de implementar los gimnasios. Finalmente, se decidió incluir tres gimnasios en total. Esta elección se tomó con el objetivo de mantener un equilibrio entre la complejidad del juego y la carga de trabajo, facilitando la asignación de Pokémon a los entrenadores y la organización de los combates. De esta manera, el proyecto conserva una estructura manejable sin perder la esencia del desafío que representa avanzar en un gimnasio Pokémon. Para la batalla se decidió porque el usuario pudiera escoger tres pokemones de una lista de 7 pokemones aleatorios de los que veinte que disponíamos en total

Estructuras de paquetes:

1- Paquete modelo

Contiene las clases que representan las entidades principales del programa. Estas clases se limitan a definir los atributos y los métodos *getter* y *setter* de cada objeto. Entre ellas se incluyen:

- Pokemon
- Gimnasio
- Lideres
- Ataque
- Jugador

Este paquete define la **estructura de datos** utilizada en el resto del programa.

2- Paquete objetos

En este paquete se crean los **objetos concretos** del juego. Aquí se inicializan y agrupan las variables que almacenan la información de cada entrenador, líder y Pokémon, permitiendo así gestionar sus características y relaciones.

3- **Paquete lógica**

Contiene las clases que implementan la **lógica del juego**. En este paquete se encuentran las reglas de funcionamiento, los cálculos de daño, y la gestión de los turnos durante las batallas.

Incluye clases como:

- Combate
- EntrenadorCombate

Estas clases determinan el comportamiento interno del sistema.

4- **Paquete menú**

Este paquete contiene la clase encargada de **iniciar la ejecución del programa** y de gestionar la interacción con el usuario.

A través del menú principal, se controlan las opciones disponibles y se da inicio al flujo completo del juego.

Limitaciones:

Actualmente, el simulador presenta algunas limitaciones. En primer lugar, la cantidad de Pokémon disponibles es reducida, lo que limita la variedad de equipos tanto para el jugador como para los entrenadores del gimnasio. Además, en cada batalla se permite utilizar un número pequeño de Pokémon activos, lo cual simplifica el flujo del combate, pero restringe la estrategia del jugador.

Por otro lado, el programa se ejecuta completamente en consola, por lo que aún no cuenta con una interfaz gráfica que permita visualizar las imágenes de los Pokémon o los efectos de los ataques.

Conversación con la IA:

Acá se dejarán los enlaces de las conversaciones con la IA:

En este chat se pidieron ideas de nombres para los pokemones, nombres inspirados un poco en apodos un poco “chatas” como forma de hacer un poco más cómico

- 1- <https://chatgpt.com/share/68e570e9-86a4-8001-b5d8-d4654a6c8c67>

Acá se preguntó como hacer un cronometro en java para utilizarlo en diferentes métodos

- 2- <https://chatgpt.com/share/68e58a94-af4c-8007-ae3c-c4863ec3d8af>

En este se pidieron opiniones sobre ideas del código

- 3- <https://chatgpt.com/share/68e591ed-b238-800c-9e2f-24bde2278aa5>

Para futuro:

Para futuro se pretende agregar una interfaz gráfica para poder ponerle un diseño a los personajes, esto haría para el usuario una mejor experiencia en cada batalla; también hacer que el jugador pueda curar a los pokemones en medio de la batalla para así poder hacer la batallas un poco más interesantes y duraderas; y por último optimizar el código para que se vea mas estético.