Informe sobre el Diseño y Estructura del Juego de Rol

1. Diseño del Juego: El juego diseñado es un juego de rol (RPG) basado en texto que incorpora elementos clásicos del género, incluyendo diferentes clases de personajes, combates, mazmorras, desafíos y un sistema de progresión. Los principales componentes del juego son:

* Personajes jugables: Guerrero, Mago y Arquero
* Enemigos y jefes
* Mazmorras con desafíos
* Sistema de combate por turnos
* Ítems y sistema de mejora de personajes

1. Estructura del Código: El código está organizado en clases que representan los diferentes elementos del juego:

a) Clases de Personajes:

* Character: Clase base para todos los personajes
* Warrior, Mage, Archer: Clases específicas que heredan de Character

b) Clase Enemy: Representa a los enemigos del juego

c) Clase Challenge: Representa los desafíos en las mazmorras

d) Clase Item: Representa los objetos que pueden usar los personajes

e) Clase Mazmorra: Representa las mazmorras que los personajes pueden explorar

f) Funciones auxiliares:

* generate\_enemy: Para crear enemigos aleatorios
* combat: Maneja la lógica de combate
* game\_menu: Gestiona el menú principal del juego
* explore\_dungeons: Maneja la exploración de mazmorras

1. Decisiones Técnicas:

a) Encapsulación: Se utilizó encapsulación en todas las clases, haciendo los atributos privados y proporcionando métodos getter y setter para acceder y modificar estos atributos. Esto mejora la seguridad y el control sobre los datos de los objetos.

b) Herencia: Se implementó herencia para las clases de personajes (Warrior, Mage, Arche),Enemy, que heredan de la clase base Character. Esto permite reutilizar código y facilita la extensión del juego con nuevas clases de personajes.

c) Polimorfismo: Se utilizó polimorfismo en métodos como damage() y to\_attack(), que se sobrescriben en las clases hijas para proporcionar comportamientos específicos para cada tipo de personaje.

d) Modularidad: El código está organizado en clases y funciones separadas, lo que mejora la legibilidad y facilita el mantenimiento y la expansión del juego.

e) Aleatoriedad: Se incorporó aleatoriedad en la generación de enemigos y en la resolución de desafíos para aumentar la rejugabilidad y la imprevisibilidad del juego.

f) Sistema de Menú: Se implementó un sistema de menú interactivo que permite al jugador elegir entre diferentes acciones y personajes, mejorando la experiencia de usuario.

g) Flexibilidad: El diseño permite fácilmente añadir nuevos tipos de personajes, enemigos, ítems y mazmorras sin necesidad de modificar significativamente el código existente.

En resumen, el juego presenta una estructura sólida basada en principios de programación orientada a objetos, con un diseño modular y extensible que facilita futuras ampliaciones y modificaciones.

Estructura

