🕹️ League of Warriors

📖 Project Overview

League of Warriors is an adventure game implemented in Java, featuring two versions:

1. **Command-Line Interface (CLI):** A text-based version for terminal interaction.

2. **Graphical User Interface (GUI):** A more user-friendly version using Java Swing.

The game revolves around character-driven exploration within a grid-based map, where players encounter enemies, collect rewards, and progress through levels.

🎮 Game Features

- **Character Selection:** Choose from unique classes like Warrior, Mage, and Rogue, each with specific abilities and attributes.

- **Dynamic Map:** The game board is a matrix (n x m) where each cell can hold enemies, treasures, sanctuaries, or portals.

- **Combat System:** Engage in turn-based battles with enemies, using either standard attacks or special abilities.

- **Level Progression:** Earn experience points (XP), level up, and enhance character stats (strength, dexterity, charisma).

- **Event-Driven Cells:** Encounter different cell types:

  - **Enemy:** Fight and gain XP or face defeat.

  - **Sanctuary:** Restore health and mana.

  - **Portal:** Advance to the next level with increased difficulty.

  - **Empty:** Neutral cells for movement.

🚀 How It Works

1. **Login:** Users authenticate and select an existing character.

2. **Game Loop:**

   - Move through the grid (North, South, East, West).

   - Engage with cells based on their type.

   - Battle enemies with turn-based logic and ability management.

   - Level up upon reaching experience milestones.

3. **Game Over:** The game ends when the character's health reaches zero.

🖥️ CLI vs GUI

- **CLI Version:**

    o Run directly from the terminal for fast interaction.

    o Command-based navigation and battles.

- **GUI Version:**

    o Built with Java Swing for an intuitive interface.

    o Visual representation of the map and actions.

🛠️ Design Patterns Used

- **Singleton:** Manages game state to ensure a single instance.

- **Builder:** Constructs complex objects like character profiles.

- **Factory:** Creates characters based on user selection.

- **Visitor:** Applies abilities with different effects depending on entities.

📁 Project Structure

- **/src:** Java source files for both CLI and GUI versions.

- **/resources:** Configuration files and assets.

- **/docs:** Game documentation and user guide.

- **/bin:** Compiled class files for easy execution.

🤖 How to Play

1. Run the CLI or GUI version via the provided JAR files or IDE.

2. Log in, choose a character, and start exploring the world.

3. Fight enemies, collect rewards, and progress through levels.