1) Users can create an account.

* Users’ accounts are recorded in a database.
* Database in MySQL.

2) Users can log in and log out of the game through their account.

* Users’ accounts are recorded in a database.
* Database in MySQL.

3) Users can play the game by interacting with its GUI.

* Frontend written in HTML, JS, CSS.
* Backend written in Java.
* The game’s story tells an engaging interactive story for the user.

4) Users can save their status on the game and load it.

* The game is saved through a string approach database.

The game is user-friendly.

The game has at least two endings.

The application can allow two games.

GUI Approach.

|  |  |  |
| --- | --- | --- |
| **Essential** | **Desirable** | **Optional** |
| Users can create an account. | The game has a least two endings | The player can choose from more than two options during the game. |
| Users’ accounts are recorded in a database. | The game’s story tells an engaging interactive story for the user. | The application can allow more than one game to be played |
| Users can log in and log out of the game through their account. | The game is saved through a string approach database. | The player can choose their avatar. |
| Users can save their status on the game and load it. | Front end (HTML/CSS/Javascript). Backend (Java). | Multiple save Files |
| Users can play the game through the UI | Database in MySQL |  |