**VIRTUAL ESCAPE ROOM**

**Deliverable 1: Requirements Specification**

**Group B**

|  |  |  |  |
| --- | --- | --- | --- |
| *Alvi Jawad*  *101148341*  *alvi.jawad@carleton.ca* | *Isaac Wiebe*  *101207943*  *isaac.wiebe@carleton.ca* | *John Jegede*  *100984967*  *john.jegede@carleton.ca* | *Felix Elugbaju*  *101211228*  *felix.elugbaju@carleton.ca* |

**Problem Statement:**

The software will allow users to take the role of an imprisoned player and dive into an exciting and challenging adventure in a text-based interactive environment.

The user plays as a person locked in a room filled with interesting objects (tables, letters, clocks, graffiti, etc.) that he/she can freely interact with. Each object and its distinct features conceal puzzles, riddles, and hidden messages in the room. The user will have to examine objects, gather hints, and piece the clues together in order to gain access to a key or code that will open the lock on the door. The key objective of the game is to overcome diverse challenges in order to leave the room through an unlocked door.

Opening one door will lead to a different room with a distinctively different set of objects, interactions, and challenges. Challenges in each room are more difficult than the previously cleared rooms. The game will end when the user solves all the mysteries in all rooms and leaves through the grand door of the final room.

**Note:** From this point onwards, we will refer to the “user” and the “locked rooms” as the “player” and the “stages”, respectively.

**Software Requirements:**

This section lists the different software requirements separated into two functional releases. Requirements are classified into three broad categories:

* Critical: Requirements that are vital for the proper functional behavior of the software and included with **release 1** of the software.
* High: Requirements with a slightly lower priority than Critical. The software will be able to function without these requirements, but these are considered valuable additions.
* Low: These are general improvements to the software and is considered the lowest priority requirements. The completion of these requirements doesn’t affect the playability of the game as much, and the software is fully functional even without these requirements. These are mostly part of **release 2** of the software.

**Release 1:**

Release 1 will emphasize designing the layout of a single stage with three different challenges. In this release, the game will end when the player successfully solves all three challenges and opens the lock on the door.

1. The game shall be playable by a single player. [Critical]
2. The game shall feature cross-platform capabilities. [Critical]
3. Every stage shall contain a fixed number (3 to 7) of objects. [Critical]
4. The player shall be able to interact with objects using the command-line interface (CLI). [Critical]
5. The software shall be able to get, parse, and execute any command from the player in the form of <command> <arg>. [Critical]
6. The command "ask help" shall display a list of common commands. [High]
   * Some example commands may be:
     + “examine <objects>”
     + “listen <sounds>”
     + “get <item>”
     + “use <key/code>”
7. The command "ask hint" will provide a hint towards the solution. [High]
8. Examining an object will give further information that may lead to unique commands. [High]
9. Common commands, when applicable, shall return suitable results. In contrast, unique commands are situation specific and shall only work once certain conditions are satisfied. [Critical]
10. The parsing of unsuited commands will be associated with a suitable message indicating that the player is doing something wrong. [Low]
11. Deciphering all secret messages and solving puzzles and/or riddles in a stage shall give access to a key or a code. [Critical]
12. Using the correct key or code on the lock shall open the corresponding door. [Critical]

**Release 2:**

The second release will focus on creating multiple stages in the game and general improvements to the software.

1. Opening the door of one stage will lead to the next stage of the game. [High]
2. The game will have three stages in total. [High]
3. Each stage will be more difficult than the previous one. [High]
4. The total number of hints will be less than the total number of stages to encourage players to use them sparingly. [Low]
5. Some doors might contain multiple locks, and as such, will require the player to acquire the corresponding number of keys and/or codes. [Low]