Software Implementation and Testing Document
For

Group 11

Version 1.0

Authors:
Carly Sweeney
Maddy Burns
Miranda Arnold

1. **Programming Languages (5 points)**

   We are using C# because it is the default language used in Unity. C# is similar to C++ which we are all proficient in, and one of us has taken a C# class.

2. **Platforms, APIs, Databases, and other technologies used (5 points)**

   We are using Blender to 3D model the wizards in the game and we are using Unity for all other game aspects.

3. **Execution-based Functional Testing (10 points)**

   We tested the functional requirements by doing a run-through of the game. We currently satisfy the following requirements:

   1. The player must be able to move in four directions (up, down, left, right) using keyboard arrow keys or WASD
   2. The player can jump using space bar and sprint using shift
   3. The player must be able to explore the game world
   4. The player must be able to press a key and interact with NPCs (HIGH)
   5. The NPCs must have dialogue boxes
   6. The player must be able to start a puzzle
   7. The player must be able to interact with the puzzle and complete it
   8. The player must be able to win the entire game
   9. The user must be given a start menu to begin the game
   10. The user must be shown a win game menu to exit the game
   11. The user must be able to pause the game
   12. The user must have access to control menu to view controls

4. **Execution-based Non-Functional Testing (10 points)**

   While doing a walkthrough of the game we were able to verify that we meet the following non-functional requirements

   1. The game must have minimal lag for the player
   2. The game has a consistent art style throughout
   3. The player is not able to access anything that affects the way the game runs
   4. The code is easily adaptable for future changes
   5. The game is playable on multiple platforms (macOS, Windows, etc.)
   6. The game does not crash

5. **Non-Execution-based Testing (10 points)**

   After finishing this increment we went through and did a code review, refactoring any code that needed changing without modifying functionality.