**Software Implementation and Testing Document**

**For**

**Group 8**

Version 2.0

**Authors**:

Zak K.

Taj B.

Ash C.

Taylor R.

John M.

# Programming Languages (5 points)

We decided to use GDScript for all aspects of this project simply because it is the best way to interact with the Godot gaming engine.

# Platforms, APIs, Databases, and other technologies used (5 points)

We use the Godot gaming engine for every aspect of this project.

**3. Execution-based Functional Testing (10 points)**  
We tested the player’s interaction with the ‘door’ node by using debugging statements to verify that the player was correctly identified when passing through the door. Additionally, debugging statements were used to ensure that the game transitioned to the correct scene after the interaction, confirming that the door functionality and scene-switching behavior were working as intended. Verifying that the “Continue” button on the pause menu successfully unpauses the game, removes the menu, and resumes gameplay smoothly required this kind of testing.

**4. Execution-based Non-Functional Testing (10 points**)  
There was no exact testing on non-functional requirements, as scene transitions were put on a timer. In the future, when we add more to each level, testing how long scene transitions take might be necessary.

**5. Non-Execution-based Testing (10 points)**  
No non-execution-based testing was done at this point, since all testing involved running the application to ensure features were implemented properly.