Software Implementation and Testing Document

For

Group 25

Version 2.0

Authors:
Bryce Hart,
Nicholas Tidwell,
Keith Van Dyke

1. Programming Languages

We are using C# scripting language to implement the majority of this project. We chose C# because there is more community support for it in Unity then Java(or any other language). It is used for all scripts in the project.

2. Platforms, APIs, Databases, and other technologies used

All of our work is done in Unity and we are using a few of the Unity APIs:
- UnityEngine
- TMPro

Unity also provides various tools such as Animator and Physics engine that we are working with.

3. Execution-based Functional Testing

Game has been tested by each dev by play testing their work.
Main Menu has been tested for working functional features by observing resulting visuals when played and clicking through buttons

4. Execution-based Non-Functional Testing

Non functional testing has been minimal so far in our development.
We have made sure we can run the game with 2 people smoothly with no lag or delays.

5. Non-Execution-based Testing

Each developer is required to check their own code before pushing to repository and each developer has the opportunity to critique others' code before it makes it to master branch.