Software Implementation and Testing Document

For

Group <14>

Version 1.0

Authors:

Sergio Pantoja
Tyler Starling
Joseph Schmitt
Nathan Zhao
Talal Mahmoudi

1. Programming Languages (5 points)

For our Unity project we are using the C# programming language. The reason for our choice is because Unity by default uses C# for its scripting. We use these scripts and attach them to game objects for gameplay functionality. Most of our members are familiar with C# and the learning curve wasn't as difficult as some other languages.

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

Our group used the Unity game engine for game design and Microsoft Visual Studio for programming the C# scripts

3. Execution-based Functional Testing (10 points)

We have tested the function for the scoring/streak counter for when a spawner is dropped toward the key to correctly increment scoring, while implementing a missed-key sound for when a key is incorrectly hit. The main menu was tested for hitting play and exiting the game using the 'escape' button at the top left. We have additional tested the settings within the main menu for the sound of the keys being hit.

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

We were not able to perform any non-functional testing in this increment of our project. We are not able to determine lag at this point in development and have yet to determine security of the project

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

We looked over the scripts in Unity as a group during team meetings and made sure to edit the source code and debug issues as a group as needed.