**Software Implementation and Testing Document**

**For**

**Group 4**

Version 1.0

**Authors**:

Cristhian Prado

Caleb Rachel

Lucas Compton

Mason Day

# Programming Languages (5 points)

*Unity (C#)*

* Used in Scene Management as well as general button functionality
* Will be used in Single Player functionality primarily
* C# was chosen as the primary language of choice due to its built-in functionality with the Unity Engine. An alternative was C++, but that would have required more overhead that we would have to accrue to get that set up as well.

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

*Platform: Desktop / Laptop*

*Unity Engine: Used as the Game Engine to run scenes as well as house the game itself*

*Virtual Studio Code: Works in tandem with Unity Engine when opening the main scripts for the game*

*GitHub Desktop: Chosen as opposed to Unity VCS given the teams familiarity with GitHub as well as the issue tracker as a point of usage for project contribution paired with easy-access commit history to review*

# Execution-based Functional Testing (10 points)

*Describe how/if you performed functional testing for your project (i.e., tested for the* ***functional requirements*** *listed in your RD).*

* *Testing of scene switching from MainMenu to SinglePlayer scene done on two machines to make sure button permanence / no resizing of button*
* *Testing of scene switching from MainMenu to Tutorial*
  + *Testing of the Scroll View of the Tutorial Cards done; ensured you cannot endlessly scroll and dragging further invokes debounce*
* *Testing of scene switching from MainMenu to Settings*
* *Testing of Quit Button done through Debug.Log() as Application.Quit() will only quit out when used in a .exe file*

# Execution-based Non-Functional Testing (10 points)

*N/A – No execution-based testing has been done; we are focusing on functionality first*

# Non-Execution-based Testing (10 points)

*General inspection of the project’s transferability across our varying machines to make sure current functionality worked the same on all platforms.*