**Software Implementation and Testing Document**

**For**

**Group 13**

Version 1.0

**Authors**:

Thomas Cano

Tyler Flaherty

Jack Hayes

Thomas Marconi

# Programming Languages (5 points)

We are using C# to write the scripts and object oriented components of the game (like classes for instance) due to its relatively low time to learn and its compatibility in Unity. We are using Unity as our game engine because we believe Unity offers the most versatility and utility for developers which grants you greater freedom to fully design and implement your vision for your game.

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

In terms of creating the game, we will just be using unity and C# for the reasons previously stated. We do not plan on adding any additional softwares as of right now but that may change if it is necessary. We are also using github to collaborate on the project and easily share code.

# Execution-based Functional Testing (10 points)

We have not performed any execution-based testing yet.

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

We have not performed any execution-based testing yet. Because we are making a game, most of this testing will have to be manual checking for features.

# Non-Execution-based Testing (10 points)

We have not coded enough yet to do this yet. Our group is using github which will make it easy to track changes in code. That way if there are bugs it will be easier to see where they may have happened between versions. We will perform code inspections when we have enough code where it is necessary.