**Term Project Proposal**

By Andrew Tran & Kate LaFrance

CST 223 Winter 2020

Version 4.0

**Language**

Our chosen language for this project is Lua. Lua has several strengths, include its ability to write the same application with less lines of code than in C or C++, but it’s also fast, portable, and embeddable. This makes it a useful tool in designing embedded gaming devices. While C is known for its speed, in Lua the [1] “game logic is easily prototyped, written, and tested with Lua, which requires no recompilation” giving C a run for its money.

This language can also support other languages by extensibility, and it is mainly used as an extensive support for other languages. The programming paradigms that Lua has is object oriented and scripting. The framework we are using with Lua is love2d, a framework a lot of people use to make games with.

I believe using this language and framework allows anyone to easily create a simple game that may take more time and effort on any other language and framework. With that reason, it is a good choice for this project. The resources that we may need for this development environment is the love2d framework and Lua, which we can get both from their respective websites.

**Project Features and Flow**

Our goal is to develop both a version of Space Invaders classic game and a Maze game in Lua. They have a shared menu to choose which game from (This is a stretch goal).

On game Load, the menu screen appears:

**Option 1: Maze Game**

1. A preset maze/path loads to screen.
2. Player starts at maze entrance
3. Player uses arrow keys to traverse the maze.
4. Once Player navigates to “Finish” point, game is over.

**Option 2: Lua Space Invaders**

1. A screen Loads with the player and a background.
2. Enemies spawn and begin coming down from space
3. Player uses left and right arrow keys to move side-to-side.
4. Player uses the spacebar to shoot the enemies.
5. If the player destroys all enemies, a “Game Won” screen appears.
6. If the enemies reach earth, the game is lost and a “Game Over” screen.

**References**

1. <https://www.activestate.com/blog/lua-not-your-average-scripting-language/>