**Individual Contribution – Daniel Masterson**

Throughout the development of the project, I have primarily focussed on engine development while also working on some of the more complex gameplay systems that would involve engine work. Using this engine, the other programmers then easily implemented the game and various menus to match the given specification.

I started the engine by creating a prototype based off of my original concept before the first team meeting. Coincidentally, the concept we finally settled on used a node capturing system that – while different to the prototype – was similar enough that we could easily repurpose existing engine code, while swapping out a majority of the gameplay code. In doing so, we had a rough prototype of the final concept within a week.

The engine uses a pseudo component architecture, with entities being made out of simpler components like images, buttons and text labels. Components are actually a subclass of entity, meaning that components can contain other components, to make a component chain (This can be seen in the button entity, which creates its own label entity for drawing the button text). The entities themselves modify the components as appropriate, while the components themselves handles things like drawing, handling mouse events and positioning (in the case of labels). While this did make it easy to add otherwise complex code, the need for resolution independence (particularly with UI elements) meant that there was a significant amount of boilerplate code when creating the components. If I were to make this engine again, I would have a separate UI subsystem based around HTML or another design markup language to help reduce the amount of boilerplate and to help make designing the UI significantly simpler.

As time went on, I moved onto handling and helping with the more complex gameplay programming tasks. This including rewriting the scene system - A method of containing large numbers of entities separate from each other - and improving key parts of the AI that proved challenging to solve. I also assisted the other programmers in their tasks at times, and helped modify assets so that they were in a game ready state.

Nearing the end of the project, my work shifted towards correcting bugs and completing any half-implemented features, as well as continuing to help the other programmers with their remaining work. I also completed the audio system and found various audio assets to play during various UI events.