This document comprises of the design for the third prototype. With the majority of the mechanics and ‘game’ already implemented via the previous prototypes, the focus of prototype three is tweaking and refinement. I plan in this prototype to implement the final method of spawning, the end goal for the game and tweak the user interface.

**User Interface Design – Visual Changes**The main task with changing the user interface is to remove it away from developer/stock text and buttons and provide it with a theme – this isn’t a major part of the project as a whole and rather will be there to accompany it and not have it be presented as something barebones. These changes consist of the following:

* Providing a name and an accompanying title to the project and thus the main menu
* Re-designed buttons for the main menu
* Selection of new general font for the user-interface

I was initially considering to include a heads-up-display (HUD) into the active game but I have decided not to. There is not much active during gameplay and so therefore isn’t much to measure. During the focus group, a person mentioned including a crosshair into the HUD, either a cross or a dot to indicate direction; I have also decided against this as there isn’t anything to click or collect within the game and so having a crosshair would instead just be something obstructing view.

**Unity Environments – Tweaks & Additions**

* **End Goal:** Within the procedurally generated map, I am going to include and exit point. This exit point will be built in the same way the I built the player spawning hub and instead move the player spawning hub to one corner of the map and have the end goal hub and the opposite corner on the map.
* **Teleporter:** Inside the end goal space, will be a teleporter to end the game and return the player to the main menu. This object will be a simple rotating object – a simple indicator of something important in comparison to the rest of the environment around the player.
* **Textures:** Up until this point, I have been using stock textures for development of the procedurally generated map. These were set primarily as placeholders, just to define what was a wall and what was the floor in the generated map before I replace them with something else. These will be changed but not into something complex; I would like to keep it minimalistic, simple and based upon geometry.
  + Floor: The floor of the procedurally generated map will be simple black and white tiles for textures – reminiscent of a chess game board.
  + Walls: I’m planning to have the textures of the walls kept as a simple matt black so that it doesn’t clash in colour association with the floor.

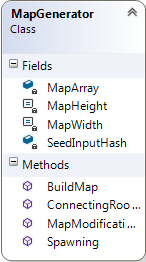
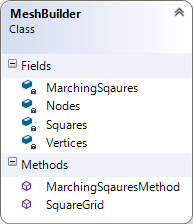
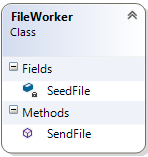
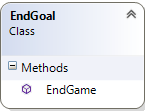
**Identifying Classes – How Will it be Built?**

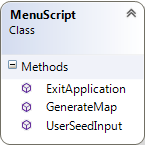
Class Additions

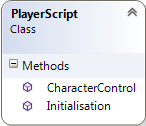
* **End Goal:** This new script will be managing the end goal Unity object. While this will be a small script with only one function, it is needed to be in its own script in order to interact with the Unity object.
  + Method – End Game: The purpose of this singular method in the script is to end the game and open the main menu when the player collides with the end goal object. This method also works with a Unity object component that works with collision detection.

Class Tweaks

* **Map Generator:** As mentioned in the Unity Environments additions, an additional spawning hub in the procedurally generated map is required; these hubs also need to be moved around in the position of the map.

Class Diagrams





Class Communication  
I have created a final communications diagram consisting of all of the scripts that will be in this final prototype. Minus the addition of the End Goal script, there isn’t all that much difference in comparison to the communications diagram seen the 2nd prototype design section.

