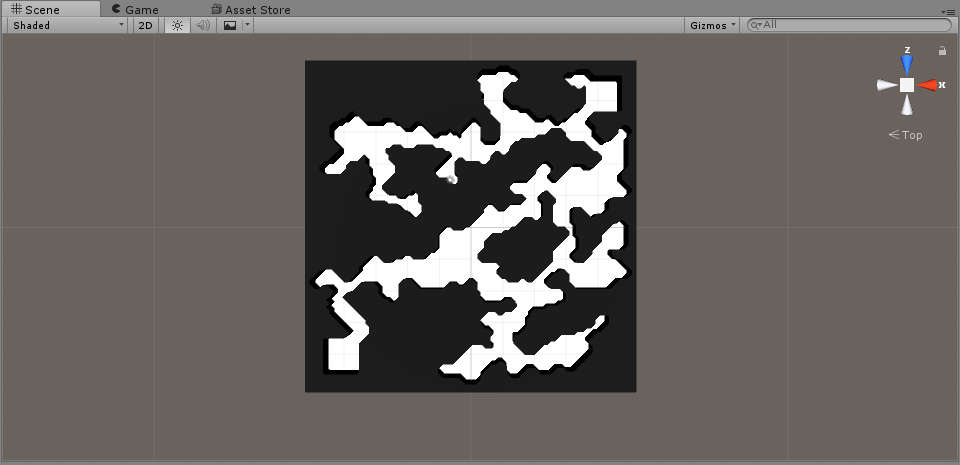
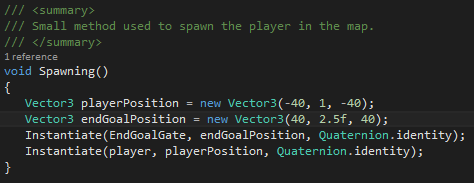
This document contains the development process for the third prototype. This will include the iterative development of the code as well as notations of any errors/corrections and testing done during the development of the prototype.

**Setting Up Unity – Current Status**The folders and arrangements of current scripts and objects are essentially in their final positions. With the addition of only one other script during this development process – nothing currently needs major changed or modification. At the end of this development however, unused assets and objects (such as placeholder textures and text) will be removed before building the Unity project.

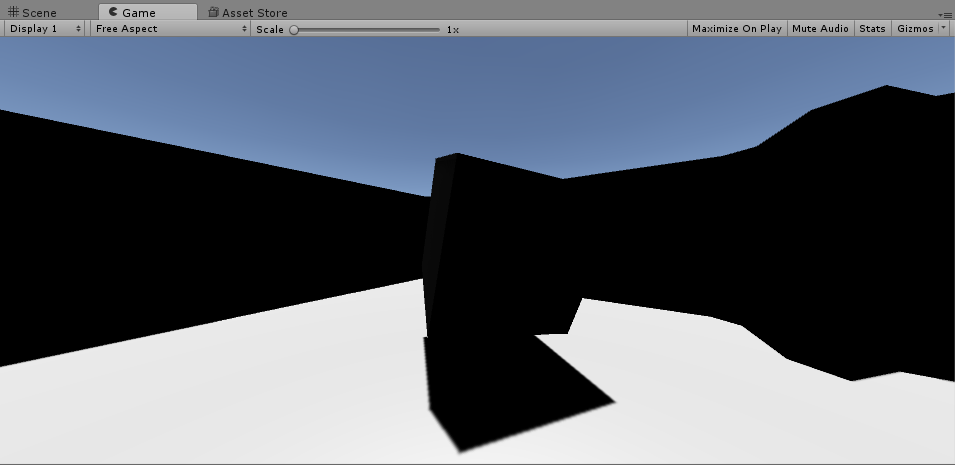
**Project Development – Modified Player Spawning & End Goal**The main objective of the game which this project comprises of is to escape the procedurally generated map – this is done via reaching an exit, what I have dubbed as the ‘end goal’. My intital idea for this project was to have the ‘end goal’ in a random location as well as the player spawning being random. This idea has since been dissolved and I have decided on the method of spawning the two on opposite ends of the map.

Modified Player Spawning  
Currently the spawning space for the player is a square 10x10 in the centre of the map with the player then being able to spawn in the middle of the map. This square and thus the player object will be moved to the top left corner of the map. I have decided to keep the square small, as to prevent the map being easily opened up to the player – even with the small space, the procedural generation algorithm is still able to connect it to the rest of the map with narrow corridors which I prefer.

End Goal Spawning Space  
In the same fashion as the square space for the player object, the spawning space for the end goal/exit will be positioned in the opposite corner to the player spawning space. The reason for this particular space choosing is that within the procedurally generated map it will be the furthest point away from the player – combined with the numerous dead ends within the map and the game will be a decent challenge in navigation.

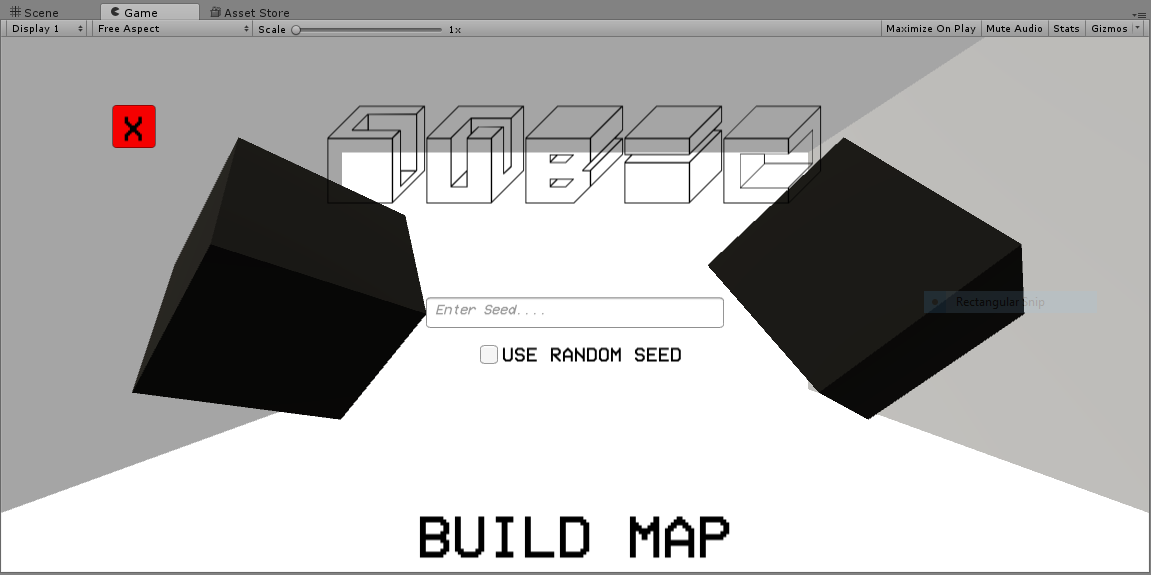


Creating the End Goal  
The exit of the procedurally generated map I have built as a simple cube. This cube has an accompanying script which only has two active functions to it. The first is Unity’s update function which is use to constantly rotate the cube; the second being an *OnTriggerEnter* function which is used to allow the player to exit the game and return to the main menu when in contact with the cube.  
This is all setup with the previously made spawning method which will now spawn both the player and the cube into the procedurally generated map.

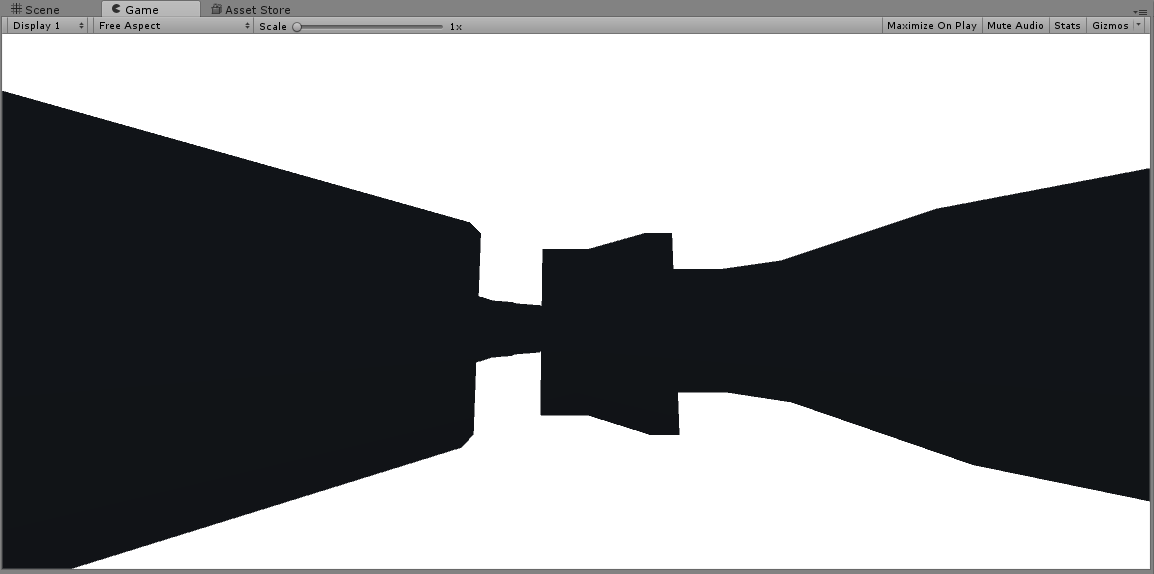


**Project Development – Theming & Final Tweaks**At this point, the main task set for this project has finished development. The procedurally generated map has been completed and the gameplay within the project has also been completed. The final process of development comes with theming and the final tweaks to the project.

New Title Screen & New Name  
The first part comes for designing a new title screen as well as providing the game created with a name. To match the end goal being a cube object, I decided to go with the name ‘Cubic’ for the game. With this I could start designing a title screen. I decided to collect some new fonts for the main menu to replace the placeholder fonts as well as create a physical background within the Unity environment; the physical background comprises 2 of the end goal cubes rotating within a white space.



Generated Map Tweaks  
One of the comments made in the previous prototype was a concern about the open top within the procedurally generated map; I have included a roof to close off the blank sandbox around the map and give the procedurally generated map a more ‘cave’ like appearance.



**Final In-Development Testing**With the final development modifications put into place, I can build and test-run what would now be a completed game. This would be in the form of a pseudo black-box test – I am not concerned about how the code is running and whether it is operating at maximum efficiency, rather I want to be able to see the game itself operating successfully and in accordance to my success criteria. I will be conducting a total of 3 tests in this in-development assessment which will be the following:

* Menu Resolution Test: During development, I turned on a feature that allowed the main menu to scale with the selected screen resolution when the game is being run; this test will allow me to see if visual bugs occur with multiple monitors.
* Gameplay Test: This bulky test will be assessing the maps generated by the procedural algorithm and see whether the game is actually playable from start to completion.
* Breaking Test: This test won’t be performed by me but by people from my focus group; their main objective during this test is to try and break the game by whatever means. This will be useful for finding bugs before using it for main testing and evaluation.

Gameplay Test  
Testing whether the game is functional from start to finish is relatively difficult. The reason being due to the fact that the main aspect of the game is procedural – even if there are 100 successful generations, there can still be an unsuccessful generation due to a dormant problem. I have up until this point found no issues with playing and the completing the game.

Breaking Test  
The participant for the breaking test was given the option to do whatever he wanted inside the game. He was allowed to press any keys, go anywhere and click on anything in order to try and break the game. The breaking test came out quite successful, the participant for the test found no methods of breaking the game or causing something wrong to happen. No crashes occurred during the test either.

Menu Resolution Test  
The method to perform this test was relatively simple; run the game multiple times to at least the main menu and change the resolution each time. To count this test as a success, the user interface of the menu will have to be able to scale properly to the size of the screen. After multiple resolution tests, I can say that the menu was scaling properly to different resolutions.

