***Programming Game Specialism Reflective Report by Rahul C***

First off I would like to state that I’ve never programmed before let alone used the Unity engine. I chose this specialism to test these things out and to get an understanding of this part of game development (I also like to challenge myself and acquire various skills across all boards). The three briefs I chose to complete all come from the beginner tab. I made this decision as I am not a skilled coder whatsoever and I didn’t find it appealing at the beginning.

*FPS COUNTER*

This was the first brief I set out to do. I completed this brief during a seminar class where I had watched online tutorials to figure out how to use the TEXT UI functions in Unity. This was where I first retained the knowledge on how I can add components-such as scripts-to things in the game world. This helped me in my other briefs as I got more comfortable with the engine and understood how to assign the code I wrote to the object that I wish for it to take effect.

First I created a ‘Text Mesh Pro’ object in the hierarchy. I have the text set to ‘---’ which is a placeholder. This acts in a way with the code that when I press play on the scene, for a second before the FPS is calculated and displayed, the text isn’t just empty or ‘0’, instead it is ‘---’ which then will turn into the FPS itself. For e.g. it will display ‘---’ then half a second later ‘405’ then another quarter second later or so it will display the next frame count. This achieves real time frame updates while conveying the illusion that it is ‘calculating’ the FPS at the beginning when really the script just hasn’t loaded.

When beginning the code, I remove the things I don’t need from the base layout. This includes the start functions and the ‘Using system collections’ parameters. I import the Text Mesh Pro library. I then added all the variables, such as polling time (how frequently the FPS is updated onscreen). The rest of the code is written under the ‘Update’ section as this updates every frame the game is running which is perfect for the one frame updates that the FPS counter utilises.

*ROLLING ROAD*

This was the second brief I set out to do. This wasn’t the biggest struggle I had however it was the first thing that gave me trouble during this module. This was my introduction to importing externally sourced assets and trying to make sense of them on my own. I downloaded visual assets (linked in doc. file) to use to demonstrate a basic rolling road level. I first attempted to include a character however I was unable to get past a hurdle regarding the player controls and gave it up so I could focus on getting the level generator to work. My main struggle during this brief was getting the level to continuously load while moving forward. When I first tried and pressed play, the camera would travel to the near end of the level and then reset back to the starting position and replay. To fix this I had to fix some compiler errors and only focus on the level generator script. I then completed the rolling road, so it continuously loads between two different tiles of level as the camera moved forward.

When creating the code, I had to first reference the tile prefabs that I imported. I used private float index and random variables determine what tile will be randomly generated in succession as the game runs. At the start of the level, tiles did not spawn so I had to add in code which creates StartPlanes/Tiles without any obstacles to make it seamlessly join with the randomly generated tiles with obstacles.

*SPEEDOMETER*

The last brief I completed and the fourth one I attempted. I first tried the radar brief however had too many troubles, so I chose last minute to change to the speedometer. I downloaded the Unity template kart game to work in. I then imported some images from Google to use as the speedometer display in the canvas. I then created the script for it. I use the Rigidbody component in this script.

To start, I create a Raw Image component in the hierarchy. I set this to use the speedometer visual. I then create a subset Text component which will display ‘0 mph’. I create a subset Raw Image which will be the arrow icon for the speedometer. I re-size and adjust everything accordingly. I ran into a brick wall here. I couldn’t figure out how to change the pivot to rotate the arrow from its end rather than its centre. After some researching and troubleshooting I was able to figure out what to do. I create an Empty component in the hierarchy and position this to where I want the arrow to pivot. I then copy its position make this the top priority in the arrow subsets, so the indicator and arrow image is a subset of the Empty component (the pivot). I then was able to set the script in the speedometer and assign the car as the body, the indicator as the image and the ‘0mph’ text as the display. When running the script, the vehicles’ speed is measured and displayed accurately.

I’ve learned a lot this semester. I understand what goes into programming. The bulk of it is lines of code however there is also a lot of knowledge of the game engine required. I was able to navigate the engine to a degree, having used Unreal Engine a plenty, however Unity was far different, so it was a challenge. It takes a lot to make even the littlest things work and one minor error can disrupt the workflow entirely.