Kevin Huang

I have spoken with Hosni and I have learned that the issue with my bullets was that I do not destroy the object when it fails to cast to the damaging character. It means that if the object touches another object that it is not meant to damage, it will do “X”. After fixing the bullet, I have gone to work on the player health, damage and enemy health, damage, etc system. I created a widget that makes damage show up when you deal damage to the enemy and created an enemy patrol system that simulates dodging. I created the bullet spread for the bullet blueprints and animated the enemy widgets(Damage taken numbers increase in size and move up overtime before disappearing).

I created the menu systems and the blueprint code to navigate between them; The main menu, options menu, pause menu, and death menu. Options menu took 5 hours alone due to the lack of knowledge, lookup and the massive amount of logic needed to create the options menu. I set it up so that it saves the information across the games for the options you choose. The main menu, pause menu, and death menu were much more straightforward and took around 30 minutes-2 hours each(After understanding the binding systems and how to create the menus). For the options menu, I needed to duplicate the options menu and modify a separate options menu for the pause menu because the regular options menu will create the main menu inside the in-game level/scene.

I have set up the enemy weapons and attached them to the skeletal mesh joint of the enemies. The score system was set up which counts up the score based on the enemies you defeat and I have set it up so the enemies are removed when they have their health clamped/set to 0. The player receives a death menu when this occurs. The high score system currently does not work and needs polishing and we need to add it in if we had time as it does not save the high score.

We have decided that we need a new map that is separate from our demo map. Hosni has told us that the Demo map that I have created is way too big and there are several things in my code that need to be fixed before the end of the semester. This included the bullets going through things which I later found out that it is because we need to create code for after something fails to cast to a specific game actor and instead casts onto another game actor.

The pause and death menu utilizes pause states within the engine. The options menu utilizes code that sets a series of resolution and graphics to itself in order to change the options. It also saves the information in an in-built save and load system for options specifically in the options menu. This allows us to remember the options information in the menu, after the game quits, and the in-game or menu options widget.

I needed to work on the bullets and reload system and took a session learning what Suraj has done with the weapon component to know how to edit upon the code. We needed to swap out the ray tracing to actual bullets and I found out that his ray tracing was a separate damage system from the bullet system that we were using so we lost time working on that. I have created several materials for our bullets and blueprints to spawn them.

In the end, in order to rush out the project with the limited amount of time me and Suraj had together, we decided that he should finalize the environment as I complete the remaining mechanics. The issue was finally fixed with some assistance from Hosni and some research. The things I have learned assisted me in learning how to problem solve in visual coding. In conclusion, the experience I learned from using Unreal is that you have to utilize the Unreal engine documentation in order to learn what is in Unreal Engine and look for anything that assists you in your current task or problem.