Sabrina Ramos

Culmination Reflection

Culmination Section: Spring 2020/Professor Ryoya

While working on 9/20 (An Area 51 Raid) for Culmination, there were various changes that took place which ultimately altered the end goal for my group’s Unity game. Initially, we had planned to include various aspects like cutscenes, dialogue, and other storytelling concepts to really help the players engage more with the actual story behind the main character Soldier Briggs and the actual basis of the Area 51 Raid events that began to spiral on Facebook. However, we ended up scrapping the cutscenes and dialogues to ensure most of the time spent on this project was focusing on the functionalities in game; this was to make sure that by the end of the semester we would have very limited amount of bugs or even none. While this project was worked on with three members in total, I tackled the animations, controls, and mechanics of the game for the player (main character) and enemies within the play space.

Animations handled a variety of different play styles within the actual game, which included the main weapons like a pistol, knife, and shotgun. However, there were ideas to add more weapons like a rocket launcher, submachine gun, rifle, and even consumables like grenades that the player would have in addition to their inventory. In a future addition, this can be added as most of the animation sprite sheets were already within the files of the game; just executing them through C# and constantly testing for bugs would be needed- which we didn’t initially have much time for. While the animations were mapped correctly, directional movement was something that caused various issues within mapping the animations to the player’s controls and ensuring they moved correctly. However, one of the biggest challenges faced when dealing with animations is properly mapping them to key presses and making sure the transitions are seamless, so the player doesn’t notice. Only working with three total animations for this version further expanded on the idea of variety and covered the basic survival layout that is seen in most video games that share the same genre.

Mechanics and controls are were heavily revamped for the player to move easier and shoot without having to press the traditional spacebar; which was implemented in the earlier versions of creating 9/20. While I believe the mechanics were smooth, I did want to add more to the concept of the player to have a running phase where the speed would be increased for a short amount of time to potentially ward off stronger enemies while the waves progressed. The biggest challenges with implementing mechanics and controls was trying to culminate these two functionalities into a single script, where it would be mapped to the player (main character’s game object) similarly like a “Game Manager”; which is used for a variety of compiled functions that run through in one set location. However, this couldn’t work because animations and movements are called quite differently in Unity when working through scripts and there were times that these two functions of code would override each other and cause various bugs. Bullet mapping was heavily imposed within this process and there were times that I had to rescale bullets, ensure the sprites locked properly, and making sure enemy feedback would trigger once they were hit with a bullet. Without these basic concepts, the game’s main grasp towards the player wouldn’t have been as enjoyable- as this could’ve meant many concepts between the player and enemy would have been broken. Bugs were very critical to keep an eye out for, because with one thing working, there was always a high chance of something else being altered with new integrations of code. Also, one big goal for revamping this concept for 9/20 is to condense the amount of test scripts within the build, as this can be confusing for programmers to understand as each script of code that performs similar can be altered slightly and create clashes in how the build works.

This project was worked in a three-person group where my other group members covered a variety of different functionalities like environment design, sounds/music, UI/UX (user interface), and collectively providing proper feedback during each wave of the game. As ambitious as we were, speaking to the technical advisor and ultimately scaling down made this process a lot easier to manage and understand; especially during moments with learning how to compile scripts with completely new functions that I was aware of. Overall, if there was anything I would change about the project; it would be the steps we took towards getting to our condensed goal. Taking that leap at a quicker pace, we initially could’ve achieved more with the game and created a better feel for the player with different effects to truly interact with users and perform in a standard arcade styled format; which would’ve been more fitting for this.

Overall, this project has helped me grow as a programmer and showed me an appreciation towards the craft of coding. I couldn’t have learned this new concept without my team and advisor assisting with different branches we wanted to include. Taking a simple idea and fabricating that idea into a digital space was an enchanting experience; being able to bring your ideas to life and showcase them with different emotions was what made this project truly exciting. Moving forward, I would love to continue working on this with my team and continue to expand my programming skills to create a new experience for players, one that is traditional in its roots, but vastly different in story.