**10:10 Games Programming Test Dev Diary - Oscar Gilley**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | | Progress Made | My Thoughts | Time Taken |
| 10/07/2023 | Setting up initial UE5 project | | So much to setup! Managed to get a project setup with Visual Studio 2022 and the latest UE5 version. | 1 hour |
| 11/07/2023 | | Made the character able to shoot a projectile that applies force to objects and destroys itself on collision with blueprints. | This was nice! I was happy with the little bullet I got to make. I wanted to do this before tackling any of the C++ side, since I’m not as familiar with how code and blueprints interact. | 2 hours |
| 11/07/2023 | | Started work on making the projectile firing work within C++ code, spawning the bullet within a C++ function. | This was a bit of a struggle but I’m getting there. I’ve gotten to the point where my code looks right but I’m unable to set a projectile type for my bullet. I want to eventually do homing through code rather than the blueprint version, which is why I’m doing it this way! | 2 hours |
| 12/07/2023 | | Projectile is fired from a combination of blueprints and C++ code | This was solved easily, I just needed a break! | 10 minutes |
| 12/07/2023 | | Projectile homes in on a set target determined via raycast. | The raycast can currently target other projectiles so it looks a bit dodgy, going to stop that and also set up some parameters to play with for the projectile! | 1 hour |
| 12/07/2023 | | Speed and turning speed of projectile determinable via blueprint, but declared in code. | This is kind of unnecessary for a homing projectile since these are available in Unreal anyway, but I wanted to get used to writing code with designers in mind. For a much more complicated system, having values exposed in blueprints for designers will be really handy for polishing! | 1 hour |
| 13/07/2023 | | Feature partially implemented where holding right click saves targets to fire projectiles at. Once the button is released, a projectile spawns per selected target | Currently having a bug with this where all my projectiles spawn on top of each other, and collide with themselves to destroy each other. Need to figure out a way of doing a delay while I loop through all the objects in my array! C++ will be needed. | 2 hours |
| 14/07/2023 | | Fully implemented right click target scanning feature. | Quite happy with this! I think this lets me use Blueprints and C++ to their strengths. | 1 hour |
| 14/07/2023 | | Moved some Blueprint logic into macros for clarity. | Just general code cleanup! My blueprint looks a lot nicer now. | 30 minutes |
| 14/07/2023 | | Modelled the rocket and added a small fire particle effect. | This took so long because I couldn’t find any suitable assets for the current version of Unreal! I hope my creation is a good enough substitute… | 2 hours |
| 15/07/2023 | | Added explosion effect when rocket hits its target. | The rockets are fully featured now! | 1 hour |
| 15/07/2023 | | Added tagging system for use during raycasts. | This is a way of validating the targets, we don’t want to be firing at certain wall and floor objects. (1 hour) | 1 hour |
| 15/07/2023 | | Macro added to allow for a For Each loop with an extra delay | This is for the right click mode! Now the missiles don’t spawn in a clump. | 1 hour |
| 15/07/2023 | | Added highlights to selected targets, and disabled targeting on map geometry/non ‘enemy’ targets | This was the final major feature I wanted to implement! It lets you see who you’re targeting with the right click mode before firing, quite helpful! | 1 hour |
| 16/07/2023 | | Added sound when rocket is fired | Just an extra wav file but it makes the implementation feel more complete! | 30 minutes |
| 16/07/2023 | | Cleaned up code and blueprints for clarity. | Naming conventions were solidified here (before, the projectile was flipping between being called a ‘rocket’ and a ‘missile’) and bits of unneeded test code were removed! | 1 hour |
| 16/07/2023 | | Added more elements to the test world to play around with. | There’s a Jenga tower now! And more structures to run around and shoot rockets at. | 30 minutes |
| 16/07/2023 | | Changed how rockets explode, allowing for explosion upon destruction rather than collision. | This means that the rockets explode when they time out as well as when they collide with something! They don’t output any force when they explode through their timer expiring though. | 50 minutes |
| 16/07/2023 | | Rockets fired fly up at a slightly randomised angle | It looks a lot nicer for the right click target painting mode. | 10 minutes |

**Total time spent: 20 hours**

**Final Thoughts**

I’m overall quite happy with how this turned out! The rockets, while only using base effects and models, look and sound quite good. I’m also happy that the speed/homing capabilities of the rocket are editable in Blueprints via UProperties. It could have been done easier by using the homing function within a Blueprint, but doing it this way let me get familiar with exposing variables to Blueprints for other team members. This will be helpful knowledge for when I work on more advanced features!

**Future Work**

If I were to continue working on this, these are the features I would prioritise implementing next:

* Some animations/particle effects for when the rocket is spawned from both the character and the area around the rocket’s spawn location.
* More advanced homing for the rocket so it can path around terrain.
* Custom models for the rocket and the player character.