

Development Diary

Jack Hopkins

I started my development by writing down what was needed to create a guided missile system. This initially included:

- The ability to control a character.
- The ability to shoot a projectile from said character.
- Create a moving object.
- The ability to aim and lock onto one target out of many to choose from.

With these broad aims in mind, I then broke down what I would need to do to get them working. After looking at the sample projects offered, I elected for the first-person shooter, as it already had two of my broad goals implemented – The ability to control and shoot a projectile from a character.

Since I was more familiar with Unreal Engine's blueprint tool, I spent some time reading the sample code and documentation to begin to understand the framework of Unreal Engine's C++ libraries.

Moving Object

I started by implementing an object that moves along a planned path. This would allow me to display the guided projectile's ability to follow an object past its initial location. While it may have made more sense to do this last and focus on the bullet first, I felt implementing a completely new, yet simple, object would help we understand the relationship between C++ and blueprints better.

It was, relative to the rest of the project, easy to implement. It used a spline and timeline component to guide the object around.

Homing Projectile

Initially, I planned to implement the physics of the homing projectile myself - adjusting the acceleration and velocity of the projectile every tick. However, when looking into the *UProjectileMovementComponent*, I found this functionality already implemented. To test this function, I would assign the moving object as the default *HomingTargetComponent*.

However, this is where I would hit my major roadblock of the project. The *TickComponent()* for my projectile movement component was not being called. To try help myself debug the issue, I added a Tick function and called *TickComponent()* manually within the projectile class. However, even then I was not able to even step into the function. I found this perplexing. I started Googling and looking through the documentation, trying to understand what was going on. I checked the dependencies. The game still ran perfectly fine, it just wouldn't call this function.

So, after being stumped on it for around 3 or so hours, I decided to message a more experienced Unreal developer, my friend PJ. He came over and had a look at my issue. We quickly found out the UE5 project was not building for whatever reason. We checked the dependencies again and spent an evening trouble shooting.

Eventually, I elected to download Unreal Engine again, but this time not through the Epic Games Launcher. Instead, I installed it from the official GitHub repo. This came with multiple additional files

and once built, the *TickComponent()* work as it intended. After this, implementing the bullet went swimmingly.

TP_WeaponComponent

To create an aiming function, I wanted to have a raycast be projected from the camera and once hitting an actor it would write to the bullet's *HomingTargetComponent*. An issue came when trying to implement in C++, because I need to called the *TickComponent()* but the actor holding the component was a blueprint. This confused me at first, but I choose to implement this function within that actor's blueprint. However, the bool controlling this function was controlled by the component in C++.

Conclusion

Overall, I felt like the programming side went well and I learnt a lot. But the bulk of my time was spent wrestling with the Unreal Engine building properly and errors outside my own code. Nevertheless, this is all part of the game dev experience and so I'm glad to learnt it here to hopefully help me solve these problems more efficiently and stressfully in the future.