

Alexander Hubble

EGP-410-101

Assignment 1 writeup

Over all architecture

The overall architecture of the project was fairly straightforward. When adding any new steering or messaging functions, just add the appropriate type to the enum value and include the class in the master class. This way it would be accessible easily through those classes.

Challenges faced

Full disclaimer, I'm not great at 3D math. So attempting to understand the math presented in the sudo code when not having a super great foundation in that field of math was really tricky. This came especially into play when trying to do the Face steering, with help however I was eventually able to understand the problem and come to a solution.

Possible improvements

First I would probably swap into an event system over an input system for the sake of this project. I find that it is easier to stay decoupled when using an event system, this is due to the functionality of the events / messages being held in different locations. With events the functionality is within the classes that use them where with messages it seems like the message themselves hold the functionality, this can lead to code being coupled together and it's not something I prefer.

The second is that I feel like a couple of my steering methods are off slightly, they all get the job done in the end but sometimes acts a little weird. For example my face steering will sometimes take a super long time to face the target, I believe this is cause

by the rotation being at 0 or 2π . When at these values it wouldn't register that it would loop by going backwards and take what it thinks is the fastest route. Also my facing on wander seems to be a little off, no idea what's causing that one.

Third, I had to change some of the `getSteering` functions to public to get `ArriveAndFace` and `ChaseAndWander` to work. It feels like that setting those to public over protected is not a great idea, however I wasn't able to get it to work setting them as friend classes of one another so this was the only solution I could think off to get it working.