Assignment 15.3

Amogh Mukherjee

Since my game is a twin stick shooter, it needs the following channels for input in order to meet the bare minimum of the genre.

- One directional input to move the character around the screen
- One directional input to rotate the character around the screen
- A button that, when pressed, will shoot out a projectile

Without these three actions (let's call them Move(), Rotate(), and Fire() respectively), it isn't possible to create a twin stick shooter at all. There are other actions we might want to implement as the game grows more complicated (for instance, a pause button, a super attack, or a secondary fire option), but for our purposes of getting a prototype up and running, we only need the three listed above.

If you look at the github project this is attached to, I've actually already implemented these inputs using the input system! Here's all the inputs It used:

- WASD/Left stick of the gamepad
    - This moves the character around the screen.
- Mouse position/Right stick of the gamepad
    - This rotates the player to face a different direction. I used separate functions to deal with rotating the player via a mouse and rotating the player with the stick, since it made for cleaner, more digestible code.
    - I considered also adding the arrow keys for this input, but decided against it since it made for a physically uncomfortable control scheme. That being said, it would be very easy to add arrow keys due to the elegant design of the input system
- Left mouse button/Right trigger on the gamepad
    - This causes the player to shoot out a projectile. In the project, it's just a small red ball that shoots off into the distance forever. The code in the project isn't very pretty or elegant. I just wanted to implement these features as quickly as possible, and it was very easy to just have the player spawn a projectile in when they press a particular button.

If I could polish the prototype some more, the highest priority would be adding in a pause button and a screen that lets you close the game. Those are the bare minimum things you need to make a minimum viable product of a twin stick shooter, but they weren't high enough priority to actually implement in this project just now. If I did implement a pause button, I would use the escape key and the settings button on a gamepad.