Introduction

In this lab we learned how to use a network manager to establish a server and client connection between games. With that you’re able to sync variables and data to the server and then that will send back to the client.

Methods

Adding the UNET Network manager to your scene will allow you to choose your scene that you want to use to play over the network with. You must first put that scene in your build settings to be able to test. The offline scene option allows your client to load the game without trying to connect immediately. Using Network Identity anything you need to control with your client will need that to send data to the server. If you connect with two players you might run into controlling to players at once. You should fix this by disabling its controller, and then when you play re-enable if (isLocalPlayer) so that you can only control your local character.

The next issue you will see is the jagged delay between packets. You can fix this with lerp, by grabbing the players position currently on the client view and then sending your position to the server and that will lerp between the old and new position. Using fixed update and [SyncVar] for sync positions will always be synced between servers and clients at a consistent rate.

Conclusion

Networking in general will vastly increase our playability for games. Also UNET makes it very easy to control player and server commands. Also the lerping between packets helps smooth out our movement to make it look nice and smooth. I learned how to send and receive data to server and client and keeping the variables synced and up to date.