Network Programming – Morgan Ohlström

Overview of game design:

I made a racing game with checkpoints/waypoints you have to go through to not cheat the game and to finish the default two lap race when crossing the finishline which just is a waypoint to. You can chat through writing in the inputfield and press the button send.

Move right, left, up and down with a,s,d,w keys.

Activate permanent speedbost through pressing the "space" button.

Write in the input field what you want to say to the other player(s) and press send when you want to send it.

Implementation details of the network features:

First I made a platform and added the networkmanager code to make the game able to be hosted and players join

Then I added a networkmanagerUI that added buttons for host and join and also server that I probably didn't need. In the end of the development I added the chat inputfield, sendbutton and the textfield where the chatmessages are shown.

After the networkmanagerUI script I added the playernetwork script. It have all movement, boosts, waypoints and boundary detection. It should have been divided into a few separate scripts. I also made prefabs along the way of player, canvas, networkmanager and waypoint.

Then I added the chat script and the rest of the UI components mentioned above. And also implemented the chat logic there.

Challenges faced and solutions implemented:

My first big challenge was the movement synchronization. I pretty fast could make both players move but the client was moving slower than the host and at some part it was lagging behind to. It took a lot of time to fix, I thought I was doing something wrong in the server authority code and did a lot of debug.logs. But after a few days I saw that others code on the internet hade the synchronization on fixed update and realized that my server was overworked with requests so I moved around stuff so that it would be more in the fixed update and less in the regular update and it made a lot of difference. But also I changed from network transform to client network transform and added so the client moved immediately when it wanted and the server checked if it was right and then changed the movement if needed.

Another problem I had was that I debug loged if the host was host and it was. But when the client joined it seemed that the host was not host anymore. Probably I did something wrong but I saved and ended the day without an answear and the next day I did not think about it. I had big problems with understanding the debug logs in the beginning but then I found the client logs in appdata and it helped a lot to see both host and client debugs.

Another thing I did not understand in the beginning that made it hard was about networkvariables. I thought since the variable is updated bu the server that it was only one networkvariable but later learned that the server had its network variable value and the client/clients had their own values but the server updated them (in server authoritive coding).

I also implemented a boundary checking script that I first got working good but somewhere it started to behave strange and you could not anymore move across the open space on thi inside boundary . Then later when you moved on the inside boundary you got some default movement so when you did not move with the asdw keys input your player still moved anyway. But I did not fix that in the end because I did not have the time and I feel it was not a requirement thing and that I would not get any extra credit for solving it anyways.

Reflection on the learning experience:

I learned a lot. So much. I learned networkvariables, multiplayer thinking and netcode. T was interesting and I would like to continue learning about it. I probably learned a bit slowly though and was stuck for too long at multiple times. At some times I got it right and did not know why and just left it like that and then when it got broken I had big problems solving it so I then had to really understand it to be able to fix it and in these situations I feel I learned the most.