

Snake - P03 – Alex Denford - Post Mortem – 27/06/14

I found this snake project to be very enjoyable. At first I was quite worried about implementing networking with a game because of the complexity it adds but after stepping into it quite early with Michael and reading about Raknet, I thought I would be able to do it. We starting off creating a menu system which allowed us to step into Raknet to setup a lobby system. We created the game in the same order you would face each element in the actual game. This meant that we had a good understanding of Raknet and setting up clients / server etc. before we got into handling gameplay. I am happy with the result of this project and would like to spent even more time implementing more features and ruling out any other problems I could find etc.

I found working with Michael went well. He jumped straight into Raknet with no hesitation and then once he had the basics setup, he caught me up to speed. We employed a tool called VS anywhere which allowed us to code simultaneously which gave us a good advantage. We found most of the system to be quite easy to implement and had little troubles leading up to gameplay.

We had a small bit of trouble with connecting clients to the lobby, initially, but we quickly ruled out the issues and managed the different connections. The only big issue we had in our game was the movement system. We wanted to create a system where the head position is stored as well as each turning point. These would be sent across the network to the server and received the render the other people. We had a lot of problems with creating a render function for this system. In the end (after redoing the render function twice) we decided to change to the easier system of storing each position. This made it easy to render and we just had to update our process to move each square and manage turning points etc.

Overall this was a fun project and I am happy with the result. I would have liked to make a more efficient system and I also found that towards the end of the project our solution was getting quite complex. Better planning would help avoid “criss cross” function calls and keep the classes more tidy. I look forward to further networking projects and understanding more about networking efficiency and management.