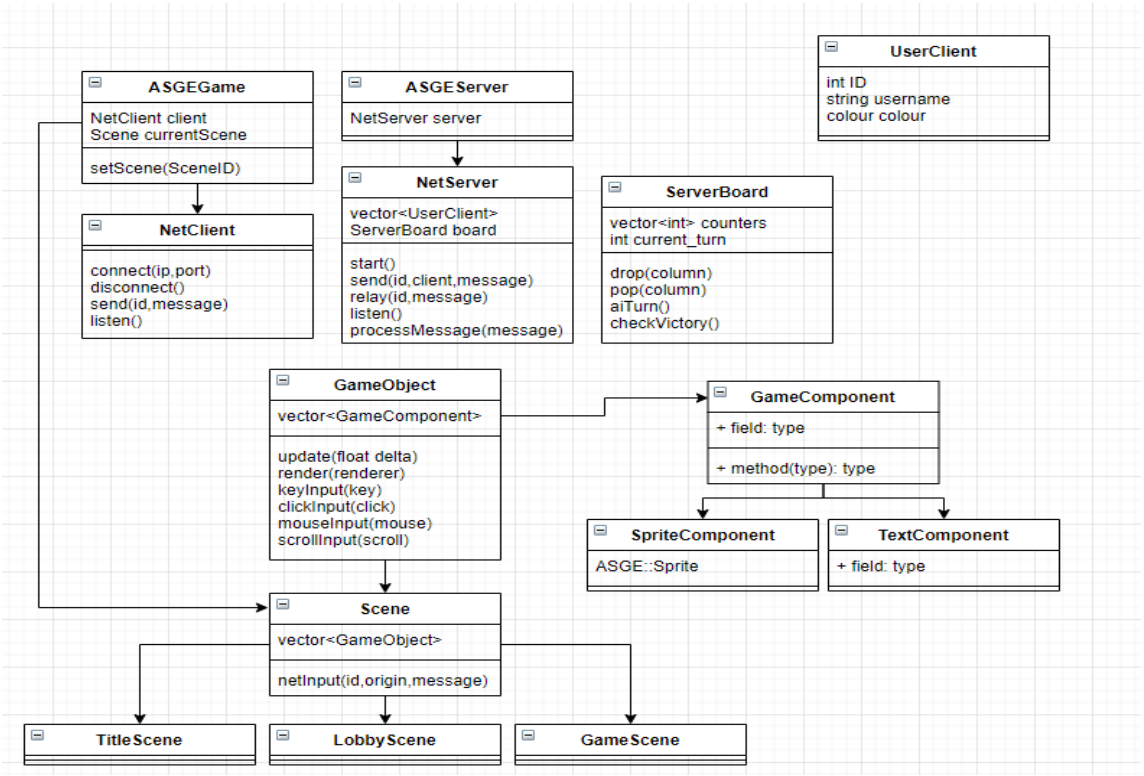
Developer Diary:

Pre-Production stage:

During the pre-production stage, research began on how it could be possible to take Connect Four to a new level and make the game more entertaining for the players. While researching we came across that there is a different game-mode called “Pop Out” where one of the players is able to take one of their chips out from the bottom during their turn; this style of play allows players to utilise tactical thinking when placing the following chip it can lead into a strategical gameplay. This will allow the game to be more flexible and interesting by introducing multiple game-modes.

After understanding the mechanic for pop-out, the development plan took a different route which allowed the team to implement the classic game-mode and pop out. The vision for the game had to be clarified and what each one of us would be working on, and what the final product would look like.

To start the assignment off we first thought about how we would go about implementing the game into C++, in an efficient way, which will allow us to utilise the implemented systems in future assignments which can make workload easier. To start off me and Ollie have both assigned each other tasks on what we would work on and the deadlines we would set us. Each week we would do meetings where we would talk about what needs to be implemented, the bugs that need to be resolved and the possible additional features. While working on the project Ollie handled sending and interpreting messages on the client side, and I handled the front-end and most of the game scenes. While planning out the project, we have also developed an UML diagram where we



showcased how our system would actually be implemented.

Development Stage:

Handling the project went swimmingly, both of us decided to split what tasks we were doing for the networking aspect of the game. Ollie was working with the ability of sending messages from the client to the server, while I was working on handling relaying of the messages back to the client. Ollie than implemented a scene system which allowed us to get better workflow in each scene. I was working on putting the scenes together, implemented the lobby game and the win scene early on which allowed us to work on the mechanics quicker once the networking had been finished. Ollie focused on his own version of a net string; this feature allowed commands to be passed between the client and server.

After the previous tasks have been completed Ollie than shifted his focus on to the title screen and allowing the clients to connect to the server and choosing an online username. Meanwhile, I was working on implemented a visual connect four board inside the game scene. In the lobby scene Ollie implemented a chatroom where the players able to interact with each other while waiting in the lobby.

Both of us then worked together on a working version of the game, a point where we we're both able to connect and play a simple game of connect four. Upon reaching this stage, I decided to work on Reconnection/Disconnection support while Ollie was handling A.I. implementation to the game. A lot of the mechanics have been shifted to the server instead of them being client based. Ollie has also handled the server-side shift and implemented a victory detection algorithm.

Most of the game has now been handled so it was time to implement the personal features me and Ollie decided on.

To start the process of safely reconnecting and disconnecting, I've implemented a function that generates a string which contains the state of the board represented by a sequence of the counter user ID's, each row separated by commas. Then if a player decides to reconnect to the game it will send the data over. The game-scene will then follow up and send the data received from the re-connected player to the board which then iterates character by character to construct the board. The counter sprites are then created, looking up the connected player's selected counter colour.

Post-production stage:

The development of the game was an enjoyable learning experience; while developing this project not only did, we improve our technical skills but also our leadership and communication skills. Me and Oliver dedicated days / hours where would we plan out and track what needs doing or how tasks can be achieved. We stayed organised and managed to hit the deadlines we both have set for each other. While working with Oliver I have learnt a lot about programming and managed to utilise his techniques for my own project. The best thing about this project was the workflow and how fast we were at achieving tasks and working on them. One powerful tool I wanted to take advantage of during this project was git, the whole project we built was made on one singular branch instead of it being spread out which would have helped massively, there were a lot of merging errors but we've managed to solve them. If we we're more experience with git we could have taken this project to a better development workflow. If I was to approach the same or a similar project, I would first now start off by creating systems. Scene Managers, or pre-built button functions helped immensely and sped up the process.