Concurrent Network Applications – Max Stockton 19017767

# Contents

Contents

[Contents 1](#_Toc59201347)

[Explanation of Strategy 1](#_Toc59201348)

[Critical Reflection 1](#_Toc59201349)

[User Guide 2](#_Toc59201350)

# Explanation of Strategy

Using multi-threaded code meant that the code had to be kept organised and easy to read. This was done in the form of using relevant names for variables, functions and classes and consistently using industry standard naming conventions. More complicated parts of the code were commented on so that it is easier to read & track of. TCP was used to send off message data as the data from the client was being sent to the server via a packets system. It is also guaranteed that the data will arrive at the destination server. UDP, however, is normally used for graphical data, which isn’t present in this project, since it sends data across before the server can receive the data. It’s faster but the data isn’t guaranteed to arrive in order or not at all. Threads were used to send & receive data from each client currently connected to the server. It kept track of how many clients were connected and started a thread every time a new client connected which would run the Client Method which would receive the bytes sent from the client & determine the packet type. Threads were also used in the client project to process the responses received from the server by using the Process Server Response method. The method also reads the incoming bytes from the server and determines the packet type of the data received.

# Critical Reflection

This assignment has allowed me to learn the basics of C# which will be useful for working on a variety of projects whether it’s in the industry itself or my own personal projects in Unity or other network based projects, as well as the fundamentals and basic principles of network-based application development. I will also be able to apply these principles & fundamentals to future projects in the industry or my own personal projects. I also learned the basics of using Windows Forms which has allowed me to create simple interactive GUIs for my programs.

I was able to set up a packet-based chat program that connected to a server project. The client gave the user the ability to send and receive messages from the server. I was able to create a client & server that transferred data between one another. I was able to send packets across and have the client & server recognise what kind of data was being transmitted & act differently accordingly. I was also able to use Windows Forms to create a simple interactive GUI that the user could use to send data across to the server.

However, I struggled with a lot of issues with the program. Due to working from home I was afraid to ask for help from tutors for a good portion of the semester & I only asked for help closer to the assignment deadline. This halted my progress by a good amount which meant I fell behind with my project quickly. I’ve come across many problems in my project including messages not being sent to the server and having issues with putting code in the right places.

If I was to attempt this project again, I would have done more reading into C# in my own time, so I wasn’t struggling as much with the syntax. I also would have asked for help earlier when I encountered a problem but also taken more of an initiative to investigate problems myself properly before asking for help with my work. I would’ve also liked to have moved onto the security & encryption section of the project as this is a fundamental part of any server-based application & I would like to learn how to do it. I will practise doing this in my free time, so I don’t fall behind. I would also be more organised with my code so I can find certain sections of my code easier & apply the same systems to other parts of my project if it’s necessary. I also would’ve liked to take the time to create a graphics-based program that sent data to the server via UDP, so I have experience in developing graphical network based projects & using different methods of sending data to and from the server. Whilst I used Windows Forms to create my GUI, I would also like to learn how to use WPF to create programs too, so I have experience in using different GUI libraries.

# User Guide

The user will be able to type messages into the chat box at the bottom & hit “Submit”. The message typed into the message box will be sent to the server. The chat box will be updated whenever it receives a message from the server, it will display the message typed in the chat box.