Report

Student name: Ellis Floriani

Student Number: 12139801

Course name: Applied Distributed Systems

Course code: COIT13229

# How to Operate System

To start with we must activate the TCP and UDP servers. To this we will right click on the java files and run them both.

A screenshot of a computer

Description automatically generated

Once these are active, we will then start the TCP client by using the same method we did for the servers. Once running you will be instantly prompted to fill out the information asked for in the console.

A screenshot of a computer

Description automatically generated

These will be first name, last name, address, and phone number. Once filled the console will then prompt the user to enter a number to either continue with another entry or to stop. This will ask to enter 1 to continue and enter 2 to stop.

A screenshot of a computer

Description automatically generated

After this we can move onto using the UDP client. To run this file, we use the same action as we did for all the previous java files by right clicking and clicking run file. Once done we will immediately. Be prompted with the console asking for a name. once entered it will then prompt the user with an option to display the current object file data. To will ask the user to type 1 to do this.

A screenshot of a computer

Description automatically generated

# UML Class Diagram

Your Diagram goes here.

# Client Server System Architecture

Your client server architecture illustrating multiple clients connecting to server goes here

A diagram of a group of people

Description automatically generated

# Test Plan

Table 1: Test plan with test data and results

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Reference | Functionality | Input List | Expected Result | Actual Result | Evidence |
| TCPServer | Convert all text data to object data. | Running TCPServer with both files in folder | Stores all text file entries in object. | Only stores latest and overwrites last entry. | Refer Figure 1 |
| TCPServer & TCPClient | Makes each entry a unique number even after client TCP is closed and reopened. | Running TCPServer & running TCPClient after closing it and running it again. | Increments the value of entry as to not have multiple of the same numbers. | If TCP client is closed and reopened it will reset its counter allowing repeats of client numbers. | Refer Figure 2 |
| UDPServer & UDPClient | After requesting the object data displays full list of the entries made in from the TCP. | Running UDPServer and UDPClient. | Displays all the entries made in TCPServer. | Only shows latest entry that was made on the TCPServer. | Refer Figure 3 |

# Evidence of Testing

A screenshot of a computer

Description automatically generated

Figure 1

Shows both memberList.txt and memberlistobject failing to covert the two entries. As can be seen in this screenshot there is only one entry saved in memberlistobject (that being the lastest entry).

A screenshot of a computer

Description automatically generated

Figure 2

After closing and reopening the TCPClient for detail entries the number of the client entry will reset creating this example above.

A screenshot of a computer

Description automatically generated

Figure 3

Figure 3 shows the output of only the latest entry instead of the list data that is shown in figure 2. After the user requests to see the full list they will only instead receive the latest entry (this issue relates to issue shown in figure 2).

# Client Server Communication Sequence for Customer Registration

### 