Government Engineering College, Thrissur

CS334 – Network Programming Lab

Documentation -

Exp3 – Multi User Chat Server - TCP

Date of Submission

17 April 2021

Submitted By

Kowsik Nandagopan D

Roll No 31

TCR18CS031

GECT CSE S6

Experiment 3

Implement a multi user chat server using TCP as transport layer protocol. *(Automated Server)*

Executing program

- Code is provided in **Client.java** and **Server.java** along with this documentation. You can open the terminal in Linux (Ubuntu 20.04 tested). Then run the command

  javac Server.java
  java Server

- Open a **new terminal** and then run as many Clients as possible (3 Clients in test)

  javac Client.java
  java Client

- Server and Client will start their respective sockets and work on the same machine (IP 127.0.0.1, Port 5000).

- This is a dis-continuous message sending system. So we can send data as the console prompts us to do so.

- We can see output side-by-side in the console

- To terminate program type "**Exit**" in both program

- **Note: As per test case, we need to implement automatic server. So server will accept only following messages**

  Hi, I'm Client *<Number>*.

  May I know who else are chatting with you?

  *Message from Client *<Number>**

  Exit

P .T .O

Output / Screenshots

Server	Client 1	Client 2	Client 3

```
┌─hp@hp ~/Documents/S6/Network Lab/Ex
p3 ‹master*›
└─$ javac Server.java
┌─hp@hp ~/Documents/S6/Network Lab/Ex
p3 ‹master*›
└─$ java Server
Server socket created
Client1 Connected
Client2 Connected
Client3 Connected
From Client1: Hi, I'm Client 1.
To Client1: Hello Client 1.
From Client1: May I know who else are
 chatting with you?
To Client1: 2
From Client3: Hi, I'm Client 3.
To Client3: Hello Client 3.
From Client2: Hi, I'm Client 2.
To Client2: Hello Client 2.
From Client3: *Message from Client 3*
To Client3:
From Client1: *Message from Client 1*
To Client1:
From Client2: *Message from Client 2*
To Client2:
From Client3: Exit
To Client3: Exit
From Client1: Exit
To Client1: Exit
From Client2: Exit
To Client2: Exit
▯
```

```
┌─hp@hp ~/Documents/S6/Network Lab/Exp
3 ‹master*›
└─$ java Client
Client socket created
To Server: Hi, I'm Client 1.
From Server: Hello Client 1.
To Server: May I know who else are cha
tting with you?
From Server: 2
To Server: *Message from Client 1*
From Server:
To Server: Exit
From Server: Exit
Client Exit ...
┌─hp@hp ~/Documents/S6/Network Lab/Exp
3 ‹master*›
└─$ ▯
```

```
┌─hp@hp ~/Documents/S6/Network Lab/Exp
3 ‹master*›
└─$ java Client
Client socket created
To Server: Hi, I'm Client 2.
From Server: Hello Client 2.
To Server: *Message from Client 2*
From Server:
To Server: Exit
From Server: Exit
Client Exit ...
┌─hp@hp ~/Documents/S6/Network Lab/Exp
3 ‹master*›
└─$ ▮
```

```
┌─hp@hp ~/Documents/S6/Network Lab/Ex
p3 ‹master*›
└─$ java Client
Client socket created
To Server: Hi, I'm Client 3.
From Server: Hello Client 3.
To Server: *Message from Client 3*
From Server:
To Server: Exit
From Server: Exit
Client Exit ...
┌─hp@hp ~/Documents/S6/Network Lab/Ex
p3 ‹master*›
└─$ ▯
```

3