

Ryan Mill

Danny Kim

CS 3000

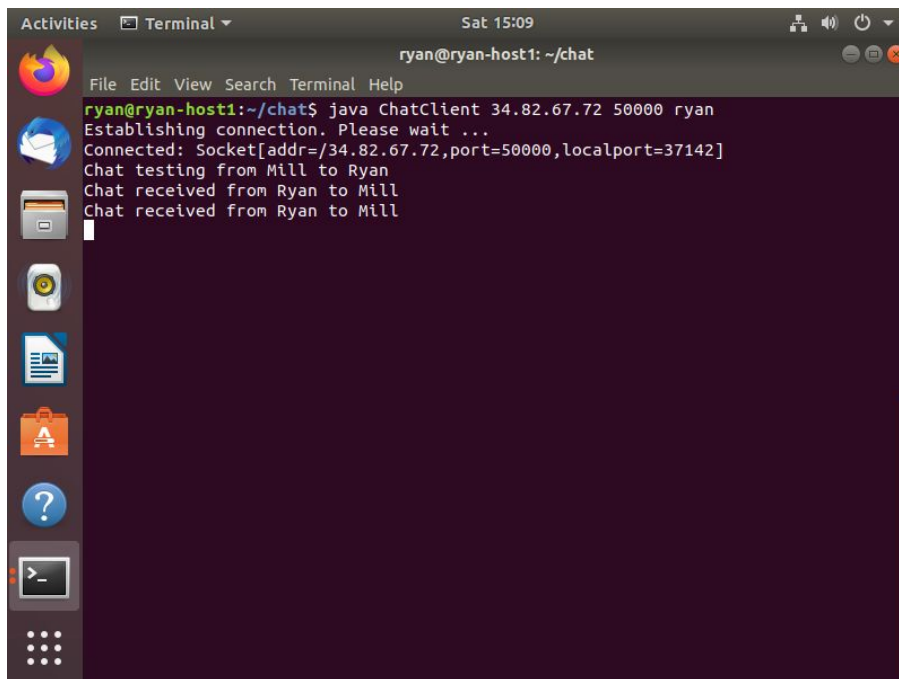
March 25, 2020

Google IP Address: 34.82.67.72

HW2 Report Document

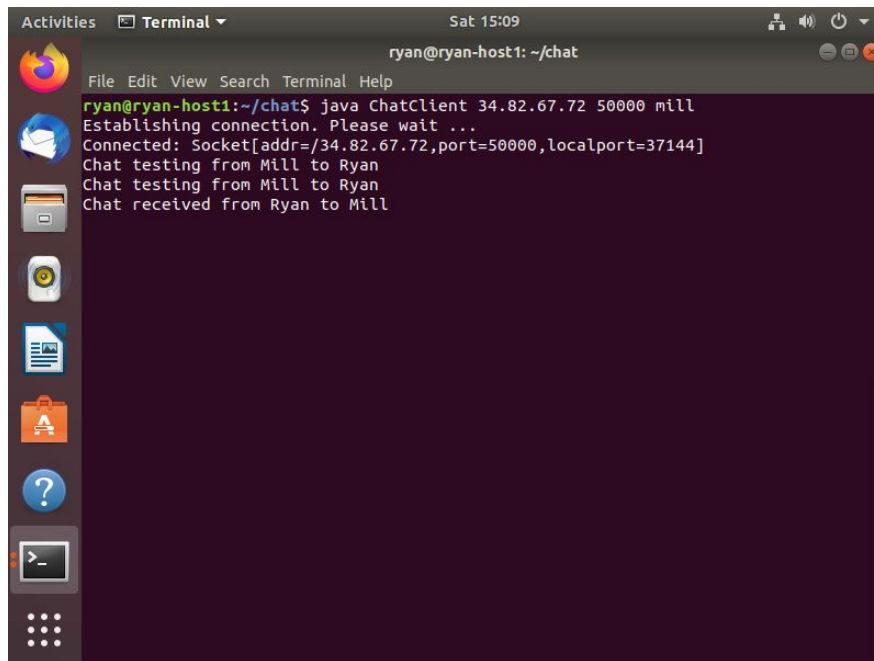
Part 1:

For this part I opened two terminals on the virtual machines. I went to the chat directory and created two different “users” by having two different display names when connecting to port 50000. I also ran “nohup java ChatServer 50000&” to run in the background on google cloud terminal. I learned from this part how to create remote connections using shared ports of an ip address.



The screenshot shows a terminal window titled "Terminal" with a menu bar (File, Edit, View, Search, Terminal, Help) and a status bar (Sat 15:09, ryan@ryan-host1: ~/chat). The terminal output shows the user running the command `java ChatClient 34.82.67.72 50000 ryan`. The output includes: "Establishing connection. Please wait ...", "Connected: Socket[addr=/34.82.67.72,port=50000,localport=37142]", "Chat testing from Mill to Ryan", "Chat received from Ryan to Mill", and "Chat received from Ryan to Mill". The terminal has a dark purple background and a light blue cursor.

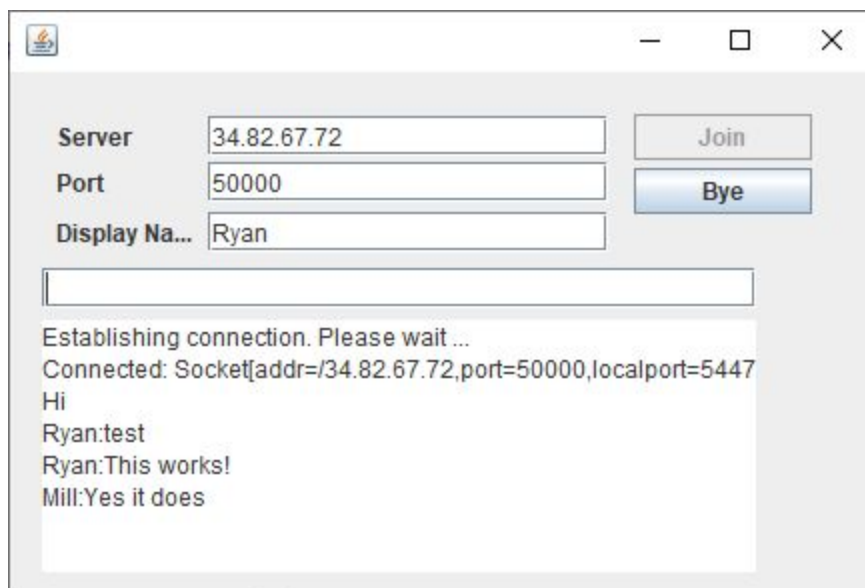
```
ryan@ryan-host1: ~/chat
ryan@ryan-host1:~/chat$ java ChatClient 34.82.67.72 50000 ryan
Establishing connection. Please wait ...
Connected: Socket[addr=/34.82.67.72,port=50000,localport=37142]
Chat testing from Mill to Ryan
Chat received from Ryan to Mill
Chat received from Ryan to Mill
```


A terminal window titled 'Terminal' with a dark background. The prompt is 'ryan@ryan-host1: ~/chat'. The user has entered the command 'java ChatClient 34.82.67.72 50000 mill'. The output shows the client establishing a connection to the server at 34.82.67.72 on port 50000, with a local port of 37144. It then shows a series of chat messages: 'Chat testing from Mill to Ryan', 'Chat testing from Mill to Ryan', and 'Chat received from Ryan to Mill'.

```
ryan@ryan-host1: ~/chat
ryan@ryan-host1:~/chat$ java ChatClient 34.82.67.72 50000 mill
Establishing connection. Please wait ...
Connected: Socket[addr=/34.82.67.72,port=50000,localport=37144]
Chat testing from Mill to Ryan
Chat testing from Mill to Ryan
Chat received from Ryan to Mill
```

Part 2:

This part I created the GUI client and set the IP connections to my google cloud IP Address through port 50000. I learned how to turn a java application to implement shared ports and IP addresses in an easy to use format.



— □ ×

Server	<input type="text" value="34.82.67.72"/>	<input type="button" value="Join"/>
Port	<input type="text" value="50000"/>	<input type="button" value="Bye"/>
Display Na...	<input type="text" value="Mill"/>	

Establishing connection. Please wait ...
Connected: Socket[addr=/34.82.67.72,port=50000,localport=5448
Ryan:test
Ryan:This works!
Mill:Yes it does