

Program Design:

Server.java File

- The Server.java file has two main classes in it; the server class and the client thread class.
- The **server class**;
 - Read the credentials.txt file from working directory,
 - Open a server socket on the the port number given in the arguments and
 - Open multiple client sockets for each client thread.
- The **client thread class**;
 - Create input and output streams for this client,
 - Validate username and password,
 - Block the ip/username after three consecutive wrong tries,
 - Check if there are any offline messages for the client and print them on client screen,
 - Logout the client ,
 - Block and unblock users,
 - Send private message and broadcast messages,
 - Show the user if he wants to see who else is online,
 - Show who else is online since past time,
 - Check if the statement from client is invalid,
 - If user want to log out, closes its input stream, out put stream and socket.

Client.java File

- The client file open the socket on the host and port being provided in the arguments.
- Establish input and output stream.
- Opens connection to the server.
- Read what is being said by the server and send standard input data from user to server.
- Closes the input stream, output stream and connection after user wants to logout.

Transport Layer Protocol:

I established a TCP connection from client to server to send messages. The tcp connection is being established by client

```
clientSocket = new Socket(host, portNumber);
inputLine = new BufferedReader(new InputStreamReader(System.in));
os = new PrintStream(clientSocket.getOutputStream());
is = new DataInputStream(clientSocket.getInputStream());
```

Assumption:

- The maximum client number will not increase 30.

Doesn't work under following circumstances and Improvements to make it work:

- When instead of proper logging out, a client breaks the connection from server and tries to login again, the server will consider this client as already logged in from another machine.
- When client type broadcast and doesn't type any message, it brings a null pointer to server, and make some problems for the server. This can be removed easily by some more controlled conditions in broadcast if statement.

References:

In the beginning I struggled a lot in coming up some workable design of this project. I tried many different designs but as the project becomes more dynamic and complex, the designs failed. I watched a lot of online youtube videos and go through a lot of online codes. After that I came up with the design format, and proceeded with the project. Following are the some links where I took help to execute the project:

- <https://www.mkylong.com/java/how-to-get-client-ip-address-in-java/>
- <http://stackoverflow.com/questions/22703409/java-code-to-get-the-difference-between-two-timestamp-stored-in-arraylist>
- <http://alvinalexander.com/java/java-timestamp-example-current-time-now>
- <http://stackoverflow.com/questions/4044726/how-to-set-a-timer-in-java>
- <http://stackoverflow.com/questions/8229473/hashmap-one-key-multiple-values>
- <http://stackoverflow.com/questions/12315023/java-console-based-client-server-login-application-error-in-database-authent>
- <https://codereview.stackexchange.com/questions/78230/tcp-chat-server-client>
- <http://stackoverflow.com/questions/31345462/client-server-username-password-authentication>
- <http://stackoverflow.com/questions/1840420/how-to-find-the-ip-address-of-client-connected-to-server>
- https://www.youtube.com/watch?v=vagEh_Y99uY
- <https://www.youtube.com/watch?v=2cQJJwoSNLk>
- <https://www.youtube.com/watch?v=uYRTpMGdf1g>
- <https://github.com/TheCherno/ChernoChat/issues>
- <http://stackoverflow.com/questions/9008883/best-way-for-line-by-line-reading-stdin>
- <http://stackoverflow.com/questions/9963743/client-ends-connection-or-server>
- <https://de8964361f4bb909de8d-fe8b524ce0801bda0a4b2a48b0c06837.ssl.cf4.rackcdn.com/hEfEJLvPH2rBpiw6kTLyqKvCMowJNkDGS5obMkWyWwkHJ4HM4us9aVzkG29icAxF.1485476814/TCPServer.java>
- <https://de8964361f4bb909de8d-fe8b524ce0801bda0a4b2a48b0c06837.ssl.cf4.rackcdn.com/cqVsuH3TFBJkg4WpwfVWWnSGNCJq76kPnUvd7qzxSaHmc3UopJcQjtHbnwkDEkHW.1485476902/TCPClient.java>