

## **SE3020 – Distributed Computing**

### **Lab 1 – Introduction to Socket Programming**

1. Copy/Import the ChatServer.java and ChatClient.java files to an Eclipse project.
2. Run the chat server class on your PC
3. Run multiple (at least more than 2) clients as well. Give the server IP name as 'localhost'.  
Note that you could connect to a chat server running in some other PC in the network as well, in which case all your messages will be broadcast among all the clients connected to that particular server and vice versa.
4. When you are running each client, the client program will ask for a screen-name. It will keep repeating until you enter a unique name.
5. Write a message in one client and it should get broadcast to the other clients, along with the respective client name.
6. Debug the code to see whether you understand the code.
7. Ensure the thread safety of the shared variable 'names' by using the synchronized keyword, as this shared variable is updated by multiple threads.
8. Now, modify the Server code to enable one client to send a message to a specific receiver client, in addition to the broadcasting feature. You may use the same message in a specific format to send such point-to-point messages. For instance, when the message "Sunil>>hi!" is input from Nimal's chat client, only Nimal's and Sunil's windows should get the message "Nimal: hi!" and the other clients who are logged in should not get any message. Leave the broadcast feature as it is for general messages which are not targeted to a specific client.
9. Add a list box in the client window to list all the logged in clients. The list box should automatically update whenever a certain client joins or leaves his or her chat session. Using the list box, enable sending messages to multiple selected clients (multicast), in addition to the point-to-point and broadcast messaging features that are already there.

You may use the listbox to send point-to-point messages as well, by selecting one specific recipient, instead of using the message format suggested in step 6.

10. You may add a checkbox to the client window to ignore selecting specific client(s) and enable broadcasting when required.

*Tips:*

- *In the code, 'To do:' comments are added in places that may be useful in identifying the places to edit/add new code. However, you may edit the code as you wish.*
- *You may use the `java.util.HashMap` to store the `PrintWriter` objects using the Screen-name as the key.*
- *`javax.swing.JList` control could be used to create a list, which can be customized to select multiple list items.*

## **Submission**

Add detailed comments to the newly added code. Compress the resulting project to a zip file and rename the zip file name to your registration ID. Upload the zip file to the provided link on courseweb.

e.g. ITXXXXXXXX.zip