

Java RMI Chat

1. Code Execution

I was using Github to store the code of my application. I find this method the easiest to launch my project and as you were suggesting using Eclipse,

You can either clone it from github or just here are the instructions to clone it and then run it.

Otherwise please open project that I uploaded folder in eclipse.

In Eclipse open File menu and click Import then choose > Projects from Git > Clone URI

URI is <https://github.com/yonasjonas/rmi-chat.git>

Then click Next and Finish with a default settings.

Server and Client are in the separate folders.

Server needs to be launched first.

I have hardcoded hostname variable as my machines private ip in the ChatServer.java line 43 and this needs to be changed to your own private ip address of the machine. Due to lack of time I couldn't fix this properly.

Solution I wanted to create was to get hostname and with it get private ip.

Then on successful start it should print a message in the console.

Group Chat RMI Server is running...

I am using a default port 1099 but it can be changed if needed in the ChatServer.java file line: 47 `LocateRegistry.createRegistry(1099);`

Once server launched successfully, please run RegisterLoginPopup.java file in the Client project as Java application and it will open a prompt window to enter your details. Once details are filled in correctly it will open main chat window.

Consider that the user may have Mac, Linux, or Windows.

Unfortunately, I couldn't produce instructions how to run code on different machines...

2. Design Description

Running Server

Contains of the ChatServer class which creates the Server instance that implement IChatServer interface which extends Remote from java.rmi for the communication with clients.

With Naming.rebind method I initiate a link for the clients to use to connect where server is listening and second argument is the server interface with object with remote methods defined.

In the server project I also have the interface which implements remote methods in order to invoke them remotely.

User class is a template for the User information to be stored. It has only get methods as set methods not needed to be applied as they are set when are added during the registration/login part.

Running Client

I was developing in the Eclipse IDE all the time and was running it as java application

Entry point of application is RegisterLoginPop.java but can be also triggered by running ChatUi.java

3. Issues

1. I do have a lot of issues which I am still facing and due to lack of time I can't finish it. As other assignmen

I have listed them below:

2. Did not finish to implement the method for blocking users.
3. I am not checking every 5 minutes if clients are still alive
4. If name is wrong re-display login popup with all details. Now when in login prompt I enter name that already exists in the Users Vector it shows new popup where I remotely invoke a method in ChatClient which invokes a method in RegisterLoginPopup (didn't want to create a separate interface for the RegisterLoginPopup) to display login prompt again with a message as an argument that username is taken but for some reason login prompt has nothing in it.
5. I have used a some global variables and I am not sure if it is good practice.
6. I found it very hard to create a pretty UI with swing manually and I regret I did not use GUI to create a UI as per Oracle suggestion

7. I would like to understand Java more in general so I would be able to better debug and optimise my application for speed and efficiency.

I had an idea to implement features from leaders like Messenger, Viber etc..

1. If JTextField for the message input is empty then instead of the disabled Send button have an icon (thumbs up) to be sent to the users.
2. When users starts typing in the JTextField show a text that user is typing in the other clients chat output area and then remove it once user stopped typing.
3. Attaching an image or an audio file.

4. Evaluation

I really enjoyed this assignment and would have loved to have more time to perfect it to work on it but other assignments due soon and I need to work on them.

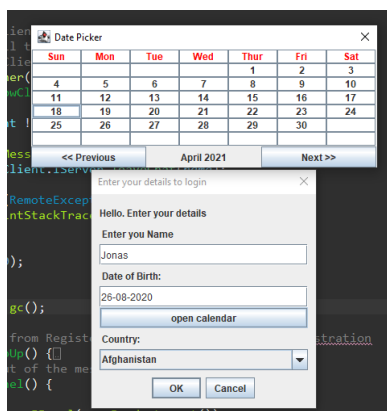
Even though I did learn a lot of new things and did familiarise myself with object oriented principles which I really enjoy coming from only functional programming background.

I was able to implement various classes. For example JButtonStateController to disable and enable button depending on the text in the input field and more...

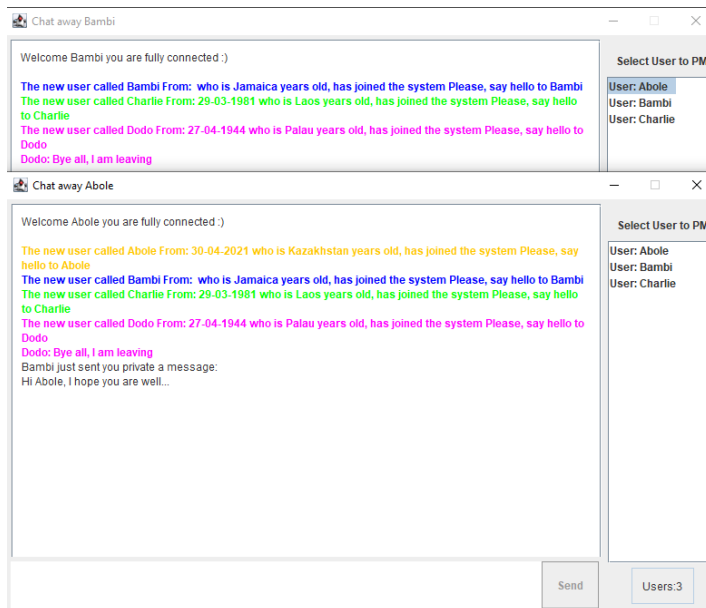
Even though I like my application and functionality I was able to achieve it is not comparing with any real life chat applications available on the market.

5. Screenshots

Initial Login Screen



UI



6. Implementation comments

1. The ChatServer should periodically (every 10 seconds) verify whether clients are still reachable. And, send a message to all clients when a client is lost or disconnected.
2. Individual clients must register with the server when they are initialised. In addition to purposefully leaving the chat system, clients may 'die' at any given time (For example, they do not post any message for 5 minutes but set this feature to 60 seconds for the testing).

After user successfully enters their unique username, date of birth and country information, those details are added as the arguments in the method and it is sent back to the server for the remote invocation of the method that creates a stub which contains an URI for the later communication with the server then it bundles all of the new client details into an object that adds the information to the User class that is a custom data structure of the user information.

If chat user closes the window of the chat, it invokes a method in ChatUI class that invokes a method on the server to remove a user object from the Vector data structure of the User by name and then return and append a text message to the UI JTextPane noting that user has left the chat. Also, another method in the leaveChat function is invoked to update a user's list panel unless this was the only user using chat client GUI.

3. Each client should be initialised with a username, country, date of birth in the ChatServer. It is recommended to use a "list of countries" for selecting the country and a "Display calendar" to select the date of birth.

When client application is launched initial login screen popup is presented to the user. Following popup is implemented in the RegisterLoginPopup.java file that contains Java class. I have 3 fields that capture chat user information.

Name: Is a simple JTextField

Date of birth: I am using a date picker that I found online which have the include of the jar file that implements the date picker. I wanted to change it to another one that is not an include but unfortunately I don't have any more time left. I hope it is ok.

Country: I am using Locale.getISOCountries() with a custom function that I found on stackoverflow that generates a list of value pairs of countries: country codes and it is used in the combobox to populate a dropdown with countries.

I have set name, country, dob as the global variables in the ChatUI class and then assigned the values from RegisterLoginPopup on the successful submission of the JOptionPane.

I am not sure if this was the best practice to be used...

4. The program should check if the name is already taken or it is free. When initialised, clients should first register themselves with the server.

I am taking the name and

5. A registration server (ChatServer) must store information on which chat instances are connected to the chat system at any given point of time. The ChatServer must also be able to supply clients with a list of all known chat clients when requested. So, implement a function ClientList() that displays to a given Client the list with all clients when a button is called.

I am using a scrollpane which holds defaultListModel that contains usernames connected at this moment of time. When new user joins or existing user leaves a method is invoked to update users panel with a new list that was supplied by the server. I had a bug that I could not solve for days where it was adding a new panel on top of the old users panel and I could not click into the users to select one in order to send pm message and it looked wrong.

Finally, I was able to find that I need to remove the panel completely and add it again.

Because repaint or revalidate did not work.

6. When a user types a message, the ChatClient code should send that message on to all chat clients with the name, country and age of the client and a string with the content of the message.

Once user types a message into the JTextField dedicated for the input of the message and clicks "Send" button, actionPerformed method is triggered to invoke local method sendMessage("message from JTextField goes here") which invokes remote method on the server to distribute message to all clients held in the in Vector User data structure with a for loop. Each user has an uri as held which is the address of the user.

I am using one button to send message to all users or if a user is selected in the users defaultListModel scrollpane then button has different SEND_ACTION and sends private message to the selected user.

7. There will be a counter indicating the number of clients in the chat. Every time a user is in the chat this counter will be increased by one. Additionally, every time a new users logs into the system a message to all members saying: "The new user called (name) from (country) who is (age) years old, has joined the system. Please, say hello to (name)."

While I am adding users to the list to be displayed in the usersPanel I increment by one for each addition to a variable int usersCount and then I update JTextField dedicated to display the information of currently active in clients.

I could have used JLabel for this instead of JTextField.setEditable(false).

8. Include the option, block users by which a user can select to filter the message from certain users.

I haven't implemented the following feature because of the time constrain but the idea would be create a separate array of blocked users by this user and store it in the User object and when sending a message to all skip to send to those usernames that have added this username to the blocked list.

9. Use different colours font for each users comments so they can be more easily identified.

I have been stuck for 2 evenings with this and was delighted when I got this working.

In the beginning I used JTextArea and was appending text from users to it. Only when I figured out that JTextPane allows to have styling I was able to colour the messages.

I have implemented a solution to assign font colour for the user according to the first letter of the name.

If I would have more time, I would have implemented a method to assign the font colours by the position in the Vector. For example if position is zero then colour is Blue if 1 the green and so on. I wanted to create a list of colours and iterate it to assign various colors to userstub.

Another idea I had is to implement a color picker in the registerloginpopup class and pass it via arguments and I did with a name, dob and country but have struggled with a Color to be passed and then assigned with StyleConstants.

10. Create a button that when it pressed, prints the number of messages each user send to the chat.

Not completed.