

Lab – Chat GUI Extension

Learning Goals

1. Develop your understanding of JavaFX interfaces.
2. Continue to develop your understanding of client server applications.
3. Continue to develop your familiarity with socket communications.

Background

In your last lab, you extended your ChatServer to include two new functions – “list” and a function of your choice. In class, we have extended the user interface of our ChatClient to use a JavaFx GUI rather than the terminal. In this lab, you will combine those two elements.

[Instructions for the lab are provided in this video.](#)

You may work with a partner on this lab. In certain circumstances, groups of three may be permitted. Please ask.

To collaborate among teammates, you are encouraged to use private git repositories or Live Share.

Part A: List of Participants and Private Messaging

Upon joining the server, the client will request a list of all users currently participating in the chat. The client will then use this list to fill the list of Radio Buttons on the left side of the GUI. An additional option “Everyone” should also be listed. Ideally, names should be listed in alphabetical order.

Whenever a user leaves or joins the server, the displayed user list should update on all clients.

The current user should not be displayed on their own screen.

When sending a message, the user will have the option to send the message to Everyone or to private message an individual. The message will be a private message if the user has selected the radio button for a single user. To implement the private message, you will likely need to create new message types.

Part B: Integrate your choice from Prior Lab

In the prior lab, you designed a new user command for the terminal program.

Integrate this new command into the GUI. The user should not type any special escape characters. Instead, the user should be clicking on buttons or using some other clear graphical user interface item to initiate your command.

If members of your group had different extensions in the prior lab, choose one of those extensions to implement.

Grading

This assignment will be scored out of 100 points, each part being worth 50 points.

The points will be (roughly) allocated as follows:

- For Part A:
 - 10 points: The list of chat participants correctly list after logging in.
 - 10 points: The list of chat participants correctly updates whenever a new user joins.
 - 10 points: The list of chat participants correctly updates whenever a user leaves.
 - 20 points: A chat participant is able to privately message another individual or message then entire group.
- 50 pts: Part B
 - Grading similar to part A will apply, appropriate to the particulars of the design.