IT347 – Analysis of implementation issues

Lehi Alcantara

In this section of the Project 1, I will describe the issues that I ran into, how I found them and how I fixed them:

1. The biggest issue by using java for this chat server-client is how do I read client or server input and output and get all the user intended server or client to get. Since java “readLine” function stops reading characters until then “\r\n” then it became hard to implement the help command since its output is a list of commands followed by “\r\n”. One way to work around it is by using “PrintWriter” data structure to send to client.
2. Then the other issue with java is like the previous item to read “help” command. Could not just do “readline()”, since that would mean only getting one of the help commands. The work around was to create a “char” array and get input from server and make a copy of the “InputStreamReader” into the “char” array. Then give it to the client to display.
3. With the push command, I had some trouble with getting the list and manipulate it as to create the right output meaning: “name: X” but after a few attempts I realized as to make a list using List data structure and include name in the list.
4. “getrange” command was also hard at first, because I was not sure how to implement, but since I had a list from push command, then I realized that I needed to just get the inputs and then make another list as copy of the original user with the parameters passed by user.
5. Checking the range was also tricky. I tried to make some comparisons of size of the list and the parameters passed in to determine if range was good, but was not so successful. So I just added a try catch which java provides where it will check for “IndexOutOfBounds” and if it was out of bounds I catch it and let the user know gracefully.
6. Another issue I had was to send a welcome to client before client sends anything to server. the work around it was to create an “int” counter equaling to zero at first send welcome message to client increment counter, then second and so forth interaction between client and server, don’t send an welcome message anymore.
7. Transitioning from command line to GUI version in java was hard at first because I was still in the command line mode, thinking about loops to determine the user input. I then realized that event-driven is for the GUI, setting event listeners and panels was not so hard, but thinking event-driven was the first block to conquered. Then I got rid of the loops that I had set up in command line. Used “textfield” and “getText” function to read the user input. Server code was not modified at all.
8. I had run some compatibility issues running with different languages such as python, but after a few research, I discovered that they handle sockets differently. We were able to identify the issue and modify python to work with java using asynchronous socket handler.