

Networks Assignment 2 GROUP COMMENT

Student A 170017147/1 GRIGORE COSTIN

Student B 170017397/1 IBRYAMOV AYLIN

Student C 170016164/1 CESNIK NEJA

Bot

In the server log private messaging works the following way: `/privmsg ProBot <message>`. That will send a private message to the bot and get a reply of a random fact in the ProBot private channel. If you message the Bot in his private channel it also works the same way. The problem we get is because the bot is listening to all of the messages in the channels he's part of (in this case only `#test`) and privately replies to any comment by any user with a random fact to the user that sent the public message. This could be fixed by making the bot ignore messages in a channel unless they include commands `!day` or `!time`.

Server

When a user connects through HexChat, it already prompts him to choose a nickname and real name. Nickname can be changed with the command `/nick`. The command prompt shows that the nickname was changed but this is not updated in HexChat. We could solve this by doing more research on how to update nicknames in the user interface. We tried implementing nickname validation but didn't manage to.

We implemented PING/PONG handlers so the server registers the commands but it doesn't return an expected reply. If we implemented timestamps, we would be able to remove clients that have quit and disconnected from the server of the client list. That is also why we get some errors in the log when `/quit` command is executed. These do not affect the functionality of the quit function.

When the server connects to HexChat, the server sends automatic reply messages in the terminal (001, 002, 003, 004, 251, 422) that are part of the IRC Protocol. Before each reply, a GARBAGE label shows up unexpectedly. We don't know why this is happening but it does not hinder the flow of the program.

If we had more time, we would focus on solving the problems but also implementing some additional features. One of them would be joining and parting multiple channels at once. In the official IRC commands, you can use `/join` and `/part` with multiple parameters which are in this case channels separated by commas. This would be implemented by splitting the argument by commas and creating a channel for each one of them. In our implementation, you can only part from within the channel by using `/part`. Another feature we would implement is the ability to set a topic for the channel. We could also add keys, which are passwords for channels that are needed to enter a private channel. Another useful command would also be `/list` which is used to list all the channels that exist on this server.