Assignment 3: a3chat.c

Objective:

The objective of this project is twofold.

1. The first objective is to force the people that didn't quite complete Assignment 2 to complete it.

2. The second objective is to increase our understanding of TCP, and furthermore how servers/client interaction works.

This is what I believe are the two objectives of Assignment 3

Design Overview:

My project is set up a little something like this
There are two sections of the code.
The Client and the Server
The client is responsible for polling stdin and pollin the pipe (once its opened via the command)
The client only checks for 3 commands, exit, close and open (and deals with the tcp stuff correctly)
The server had the majority of the code.
It had to keep the activity tracker all correct
As well as monitor the master socket for new connections (and then would make said connection)
It would also need to keep track of the commands that the client sends to the server.
It keeps track of all commands, open, close, exit, < , to, who.
Open makes sure that the user name does not exist
Close makes sure all values used are released
Exit makes sure all values used are released
< sends to all the affiliated users
Who sends back a list of users online (although there's a bit of weird formmatting)
To adds the person to the sending list.

Project Status:

I encountered a lot of difficulty with the polling portion of the code (again) because there were some weird polling issues. Getting started with the project definitely took me a long time as well until i found server.c and client.c. Starting on a blank code is extremely hard.
The only thing currently not complete is the Keep Alive messages. I couldn’t figure out how to get that implementation to work there can only be one alarm.

Testing and Results:

I tested my implementation after every single function. Once a function was complete, I tested it to its fullest range, trying to make it fail, and trying to make it succeed. Once I was happy with it, I moved on to the next function and repeated the process. After I was finished the entire project, I ran through every function and made sure I haven't broken anything else in the process.

Acknowledgements:

The most helpful sources of code was the local directory on how TCP works between server and client. That server.c and client.c code was copied and used directly in my code.
The Unix textbook that we had been given was also a bit help
Stack overflow was also a major help when trying to troubleshoot odd behaviours of C.