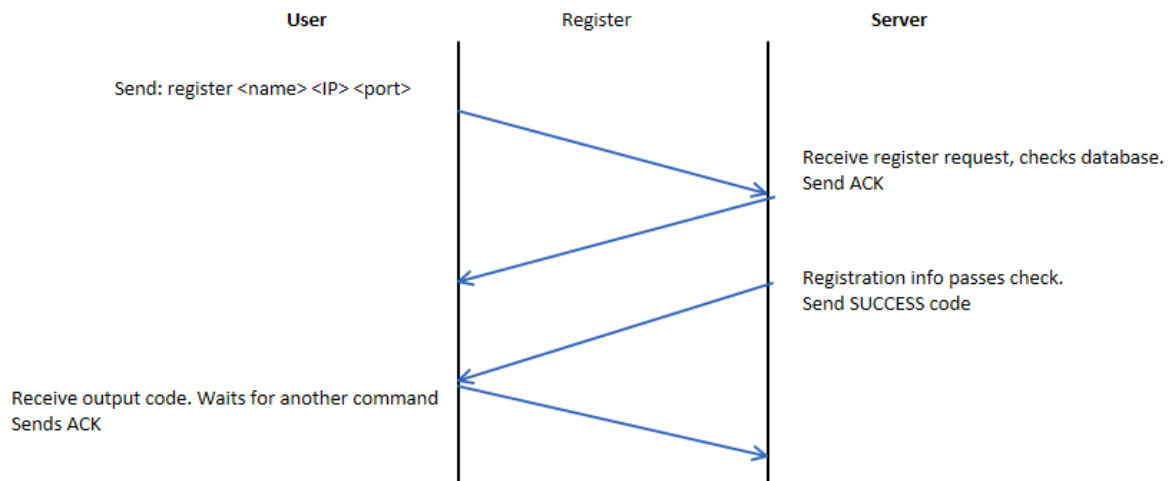


Socket Project Milestone Design Document

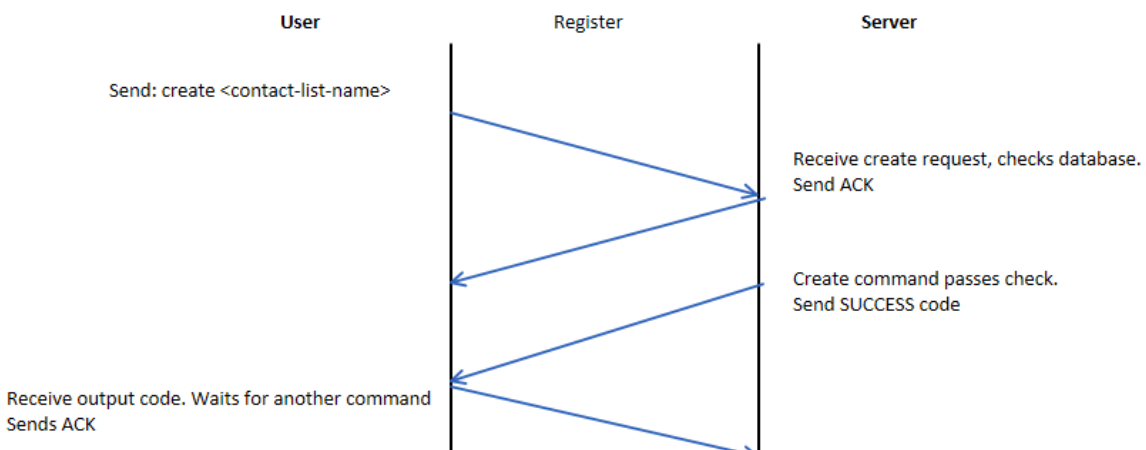
Design

- This socket project is written in Python, currently compatible with python (not python3).
- First, the server is started up, printing its own IP address and waits for a client to send requests. As the client connects to the server, a welcome message is shown along the server IP address, and the user is prompted to enter a command. **We also print the client's IP address for easy registration.** The server parses client's message for the command keyword first, then passes it to the proper function.
- There are **3** global databases: a dictionary for each registered user, a dictionary for all created lists with joined users, **and a dictionary for contact list and im-start contact name tuple.** Dictionary of registered user is a nested dictionary; each name entry has its own dictionary of IP address and port number. This allows easy implementation, as usernames are keywords to be looked up in the dictionary, we can also query either for the user's IP address or port number separately. The contact list is a triply nested dictionary. Each entry is the contact list name, they contain entries of joined user, defined similarly to the registered database. We chose this method so data can be looked up using keywords, allowing easy implementation. The databases are accessed using built-in algorithms and methods. **The third dictionary is to verify the same person that initiated the im-start command is the same name that sends im-complete. The dictionary's entries are keyed by the contact-list.**
- The server currently supports 20 users and 10 contact lists.
- Clients are sent feedback messages reflecting the outcome of their request. Server end also has internal print messages, showing client's IP addresses to differentiate sources of requests. Other internal messages are printed to reflect results of different functions to maintain developer's sanity as we build this program.
- With the implementation of IM functionality, the client program is added a chat socket. When the user registers, the port registered is bound to the chat socket. The chat socket runs in a separate process to accept peer messages and process them.
- **How we send P2P message: message, list name, contact list, number of contacts, wrapped in a list that gets encoded. Upon receiving, the list is decoded. The contact list itself is an encoded dictionary. We rotate the list until the current peer is on top and prints it. We rotate once more to extract the address and port of the next host. The counter is decremented for every peer the message is sent to, once the message reaches back to the original sender, it prints SUCCESS.**
- For the final submission, the following commands are implemented:
 - o **Register:** The command format is `register <contact-name> <IP-address> <port>`. The command will result in FAILURE under these following circumstances: insufficient parameter (it will also send back usage instruction), registered name already exists, invalid IP address, invalid port number, or maximum number of registered users already met. A short explanation follows. SUCCESS is sent

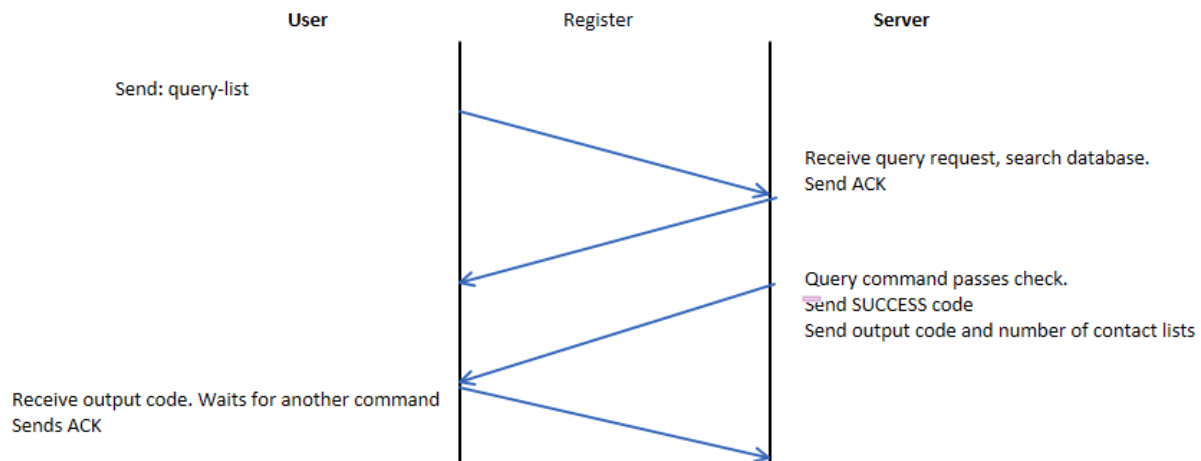
back if user's information is added to the database successfully. The client's port that is entered is then bound to the chat socket to listen to incoming message. This process is running in a separate processing thread.



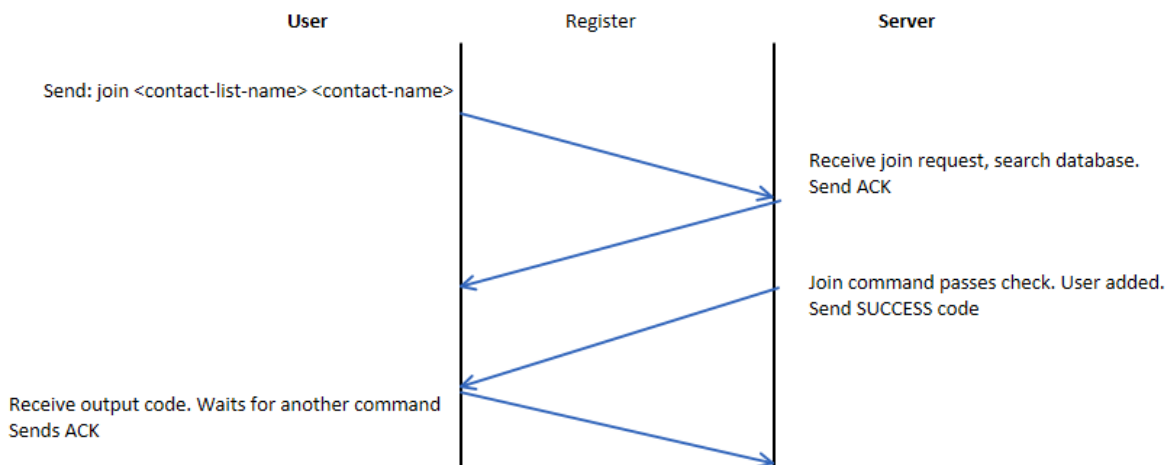
- **Create:** Command format is `create <contact-list-name>`. The command sends back FAILURE if: insufficient parameter entered (sends usage instruction), list name already exists, or exceeded number of existing lists. Sends SUCCESS if list is successfully added to database.



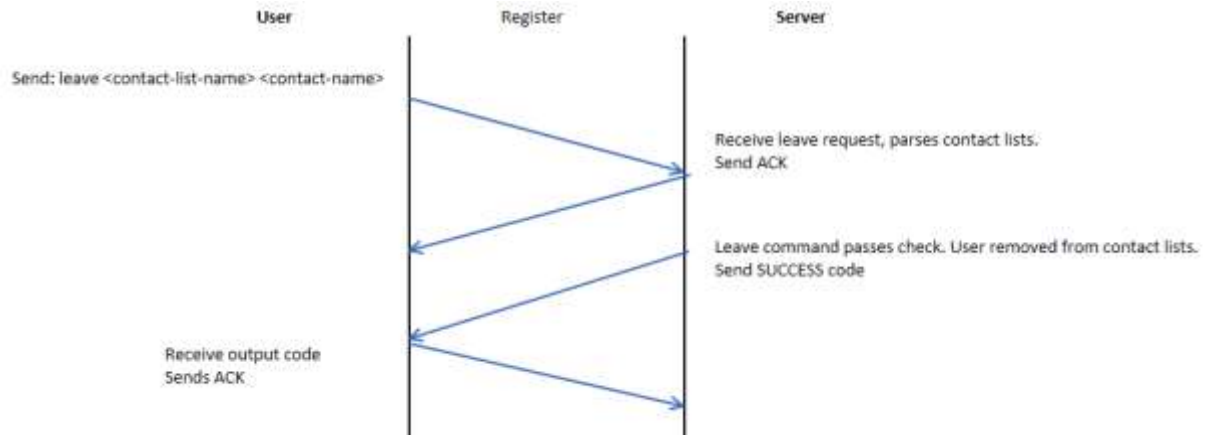
- **Query-lists:** Command is `query-lists`. Sends back number of contact lists along with their names. If no list exists, 0 is sent.



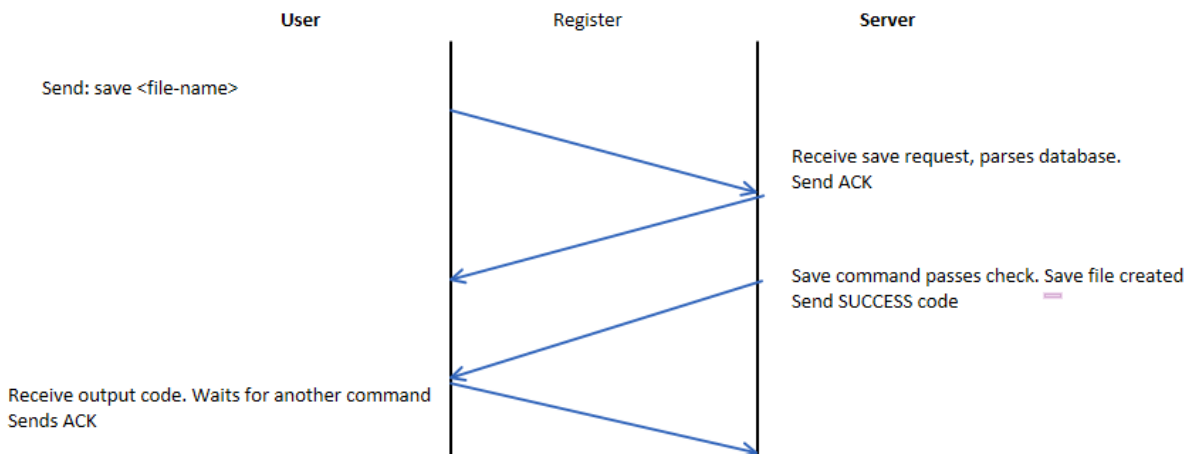
- o **Join:** Command format is `join <contact-list-name> <contact-name>`. FAILURE is sent if: contact list does not exist, contact name does not exist in registered database, or contact name already exists in the requested list, or insufficient parameter sent for this command. It will also now fail if there is an IM process taking place with this list. Otherwise SUCCESS is sent.



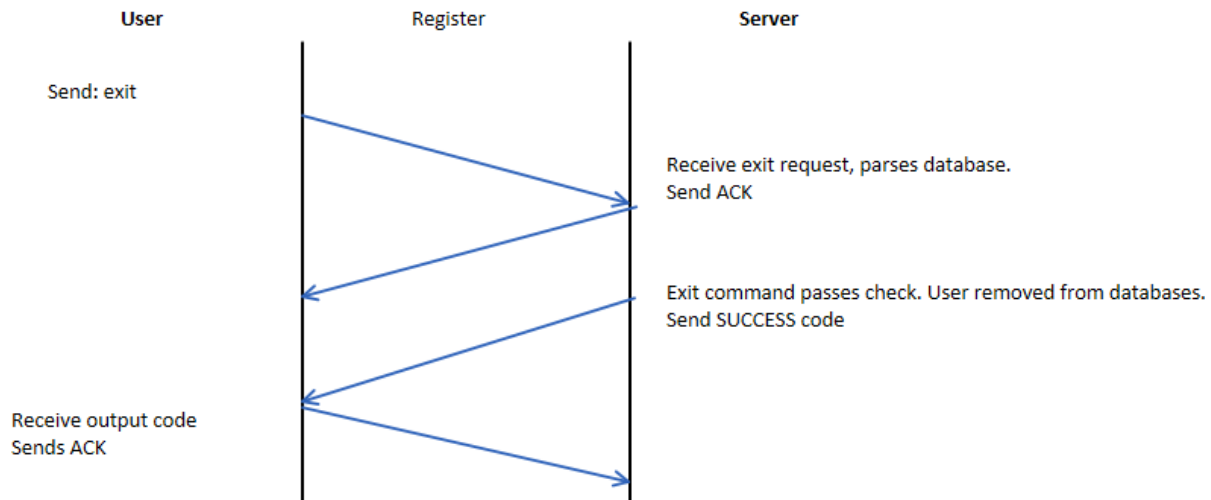
- o **Leave:** Command format is `leave <contact-list-name> <contact-name>`. FAILURE is sent if user sends insufficient parameters, IM is in process, or username is not registered, not in the list or the list itself does not exist. SUCCESS is sent otherwise.



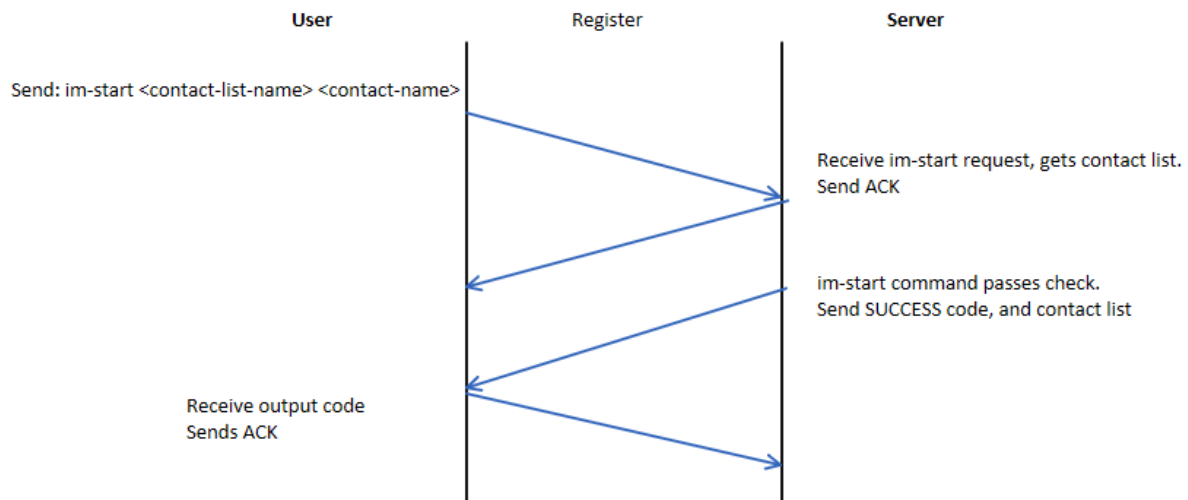
- **Save:** Command format is `save <file-name>` and a text file is save in the format of `file-name.txt`. FAILURE is sent if command's parameters are not provided. Otherwise SUCCESS is sent.

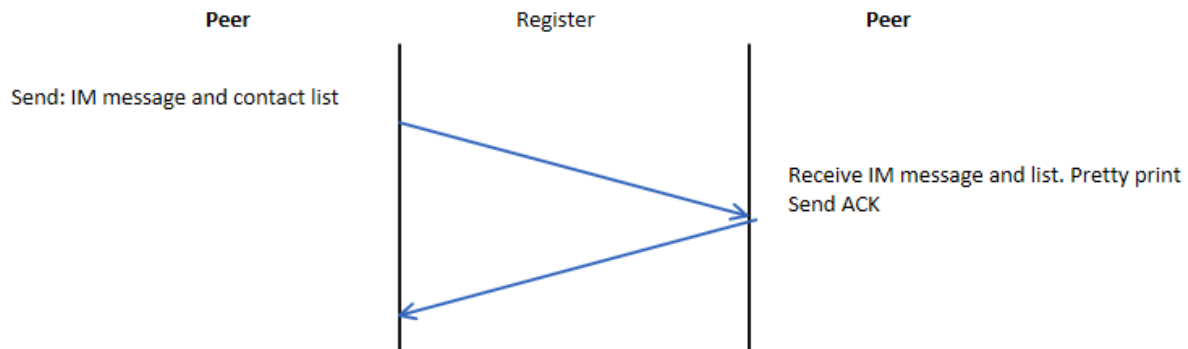


- **Exit:** Format of command is `exit <contact-name>`. Currently, the command will remove the contact-name from the registered database and all joined lists. FAILURE code is sent if the user doesn't provide sufficient parameters, or name does not exist in registered database or contact lists. SUCCESS is sent otherwise. **In addition, it will fail if the user is a part of the IM process.**

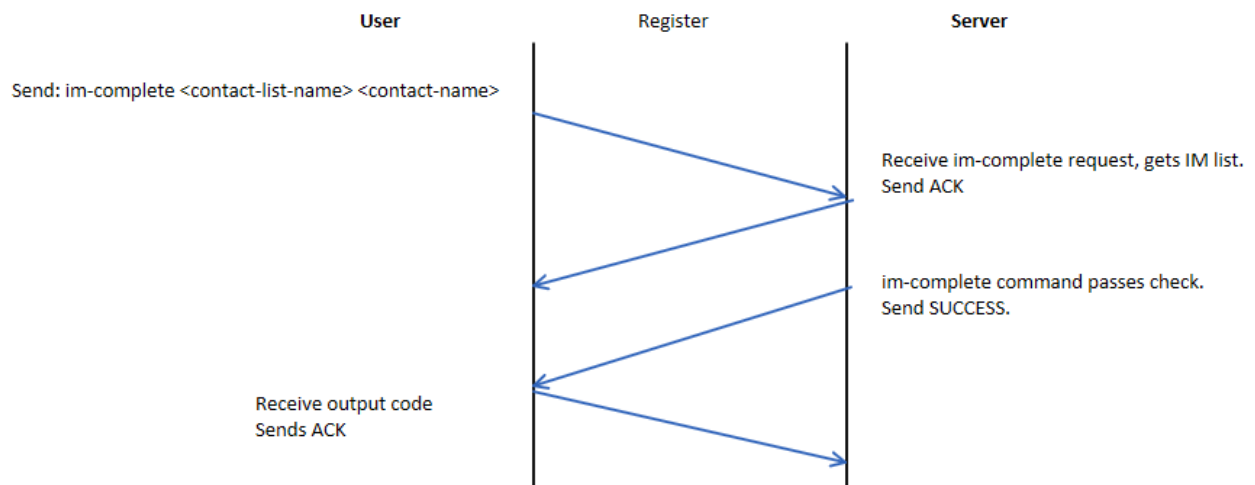


- o **im-start**: Format of command is `im-start <contact-list-name> <contact-name>`. Server checks if command parameters are met, sends **FAILURE** if not met. Also checks if contact list exists, and not currently under an IM process. Also, if contact name is not a part of the contact list, send **FAILURE**. Otherwise, server sends the client the contact list, and saves the contact list/ client name tuple in a list for verification later. Client program then prints the current client information on top, including name, IP address and port number. Then prompts that client for a message to send to their peers. The message and list will then be sent and printed through the contact circle, each time with the current peer information on top. As all peers receive the message, **SUCCESS** is printed to the initiator.



















o `im-complete`: Format of command is `im-complete <contact-list-name> <contact-name>`. The user that initiated `im-start` sends `im-complete` to the server. Server checks if it's the correct contact name / contact list tuple. Sends `FAILURE` if the list or contact name doesn't exist, or contact name is not in the contact list, or it's not the contact name that started the IM process, or command parameters are not met. Sends `SUCCESS` otherwise.




Git history


 master ▾	 1 branch	 0 tags	Go to file	Add file ▾	 Code ▾
	thydo Updated return code		a888e05 1 hour ago  91 commits		
	SourceCode	Updated return code	1 hour ago		
	Group18.zip	added im-complete and im checks for leave, join and exit	15 days ago		
	README.md	Update README.md	27 days ago		
	Writeup.docx	Updated return code	1 hour ago		
	push	added a get ip function	4 hours ago		
	run	added a get ip function	4 hours ago		
	~\$riteup.docx	Updated return code	1 hour ago		
	~WRL2750.tmp	Updated return code	1 hour ago		


History for CSE434SocketProj / SourceCode

Commits on Feb 10, 2021

Removed unnecessary checks in Server


 slleng3ater committed 1 hour ago


 f78f974




Commits on Feb 9, 2021


Added additional messaging getting printed out to server console window


 slleng3ater committed yesterday


 31ba94c



Updated messages from Server App to user


 slleng3ater committed 2 days ago


 e83a56




Commits on Feb 8, 2021


added register limit and contact list limit


 thydo committed 2 days ago


 68932ea




Updated server-client responses


 thydo committed 2 days ago


 372d4ea



Added IP and port call for Server, display in Server Console; Added a...


 slleng3ater committed 2 days ago


 94831e7




Commits on Feb 6, 2021


Added condition for port range 10000 to 10499 checking; made client a...


 slleng3ater committed 4 days ago


 74c186d




finished Save; Database and Contact Lists Save; Exit functional


 slleng3ater committed 5 days ago


 8d953ae




exitsave.py file creation; modification to database use in register; ...


 slleng3ater committed 5 days ago


 bc3d1fc



added join and leave

 thydo committed 5 days ago

 272e7b1



Commits on Feb 5, 2021		
Corrected Print out for testing in create.py slient3ster committed 5 days ago	f442235	
Added utils.py for global variables database and contactList set cre... slient3ster committed 5 days ago	251c96b	
finished register and added create thyde committed 5 days ago	c3383c3	
Added register function and main database for register. thyde committed 6 days ago	1b6a7fc	
Commits on Feb 4, 2021		
commented out line 22 slient3ster committed 6 days ago	9f8e6e8	
Tested peer 2 peer communication; added keyword detection for registe... slient3ster committed 6 days ago	1625395	
client and server files with argv code for command line argument acca... slient3ster committed 6 days ago	9e1c26b	
Socket code in py thyde committed 7 days ago	8a6a737	
Commits on Mar 8, 2021		
Updated return code thyde committed 1 hour ago	a855a95	
Commits on Mar 7, 2021		
added a get ip function thyde committed 4 hours ago	2b6b111	
Add comments thyde committed 21 hours ago	f12b366	
Commits on Mar 6, 2021		
concurring p2p!!!! thyde committed 2 days ago	9543786	
Added additional loop, removed return statement. slient3ster committed 2 days ago	8c8bc22	
Commits on Mar 5, 2021		
it sends message to 3 ppl!!!!!!!!!!!!!! thyde committed 3 days ago	1c94664	
Commits on Mar 4, 2021		
Removed breaks in message_thread, Reformatted Code slient3ster committed 3 days ago	9d51c28	
message prints to peers thyde committed 4 days ago	2f3d74f	

Link to milestone demo video: <https://youtu.be/W7vGow48-sl>

Link to final demo video: <https://youtu.be/1F6UJgu7E9c>