

## Implementation Description

This project is a basic Python-based chat application that supports:

- **Private messaging** between two users
- **Group messaging** within named chat rooms
- The ability to **create** and **join groups**

The server handles multiple clients at once using threads and keeps track of:

- Connected users (via a dictionary of username → socket)
- Group memberships (via a dictionary of group\_name → list of usernames)

Messages are sent in JSON format with four fields:

- **status:** message type (private, group, create, join, system)
- **sender:** the user sending the message
- **receiver:** the recipient (username or group name)
- **text:** the message content

Clients can send commands like:

- /private <username> <message>
- /group <group\_name> <message>
- /create <group\_name>
- /join <group\_name>

## Instructions to Run the Code

### 1. Start the Server

- Open a terminal and run:
- `python Charlie_Meyers_C00513476_server.py`

### 2. Start a Client

- In a new terminal window, run:
- `python Charlie_Meyers_C00513476_client.py`
- You'll be asked to enter a username. You can then send commands as described above.

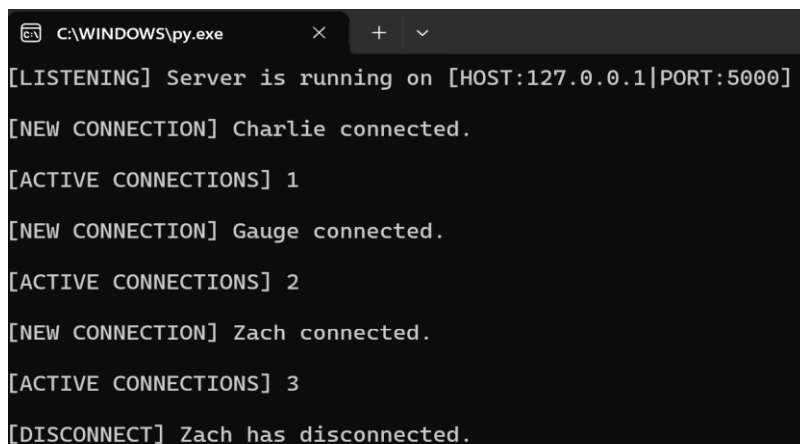
### 3. Supported Commands

- `/private <username> <message>` — Sends a private message to a user
- `/group <groupname> <message>` — Sends a message to a group
- `/create <groupname>` — Creates a new group
- `/join <groupname>` — Joins an existing group
- `/exit` — Disconnects the client

You can run multiple clients in separate terminal windows to test interaction.

## Screenshots

Note that all messages from User: Zach were successfully sent over before personal termination of his programs using the `/exit` command.



```
C:\WINDOWS\py.exe  X  +  v
[LISTENING] Server is running on [HOST:127.0.0.1|PORT:5000]
[NEW CONNECTION] Charlie connected.
[ACTIVE CONNECTIONS] 1
[NEW CONNECTION] Gauge connected.
[ACTIVE CONNECTIONS] 2
[NEW CONNECTION] Zach connected.
[ACTIVE CONNECTIONS] 3
[DISCONNECT] Zach has disconnected.
```

```
C:\WINDOWS\py.exe x + v - □ X

-----
Enter your username: Charlie
-----
Type /help to see the list of possible commands.
/help
-----

                        The List of Possible Commands
-----

1. Private chat: /private <username> <message> → Sent to a single recipient.
2. Group chat: /group <room_name> <message> → Sent to all users in a chat room.
3. Creating a Group: /create <room_name> → Creates a new group <room_name>.
4. Joining an Existing Group: /join <room_name> → Joins group <room_name>.
5. Exiting the Server: /exit will disconnect the current user from the server.
-----

/private Gauge wassup bro
[FROM: Gauge] hey man, im doing pretty good wbu.
[FROM: Zach] can you help me out with my CMPS150 PA
/private Zach yeah man when are you free
[FROM: Zach] how does tomorrow afternoon sound?
[FROM: Gauge] if you're still thinking about playing volleyball join the group Volleyball
ll
/join Volleyball
[SERVER] You have joined group Volleyball.
```

```
C:\WINDOWS\py.exe x + v - □ X

-----
                        Welcome to the ClassChat Client-Server Program!
-----
Enter your username: Gauge
-----
Type /help to see the list of possible commands.
[FROM: Charlie] wassup bro
/private Charlie hey man, im doing pretty good wbu.
/create Volleyball
[SERVER] Group Volleyball created successfully.
/private Charlie if you're still thinking about playing volleyball join the group Volleyball
ll
/group Volleyball our next practice will be saturday at 6:00pm
/private Zach did you still wanna play volleyball with the guys
[FROM: Zach] nah I think i'll pass, school has been busy.
[SERVER] Zach has left the chat.
.
```