

CS484 - Computer Networks

Project #2 - Chat Client

In this programming assignment, you will build a small chat client that interacts with the user and chat server. There will be a limited set of commands the user can issue the client. Any chat messages that the user types will be sent to the chat server. The chat server will echo back to all connected clients any messages that it receives. You will continue to practice programming with sockets and also work with threaded processes. Write all of your code in Python 3.

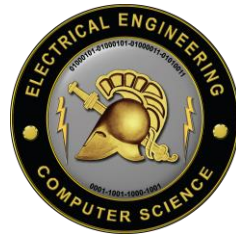
100 points; Due Lesson 20 (30OCT2018) at the beginning of class. We will test them live in class and have a group chat session.

Requirements

1. The chat client must connect to the remote server when it starts up. You may hard code this or allow the user to enter the server address and port number.
2. The chat client should prompt for a username. It should announce this username to the server.
3. Every message sent to the server should start with "<username>:"
4. The user will be able to quit the program with a command instead of with CTRL-C.
5. When the user quits, the client will announce its departure to the server before exiting.
6. The chat client should have a help command that shows the user all possible commands
7. The chat client must use threads to allow simultaneous input from the human user and from the remote chat server.
8. Block/Unblock a User Commands: The chat client will ignore all messages from this user. When the user is unblocked, their messages will no longer be ignored.
9. Change Username Command: The chat client should allow the user to specify a different username to be used in all future communications. When a username is changed, a message should be sent to the server to notify all parties of the change.
10. Block Word List: All occurrences of the following words should be converted to four asterisks: Thayer, Washington, Eisenhower, Lincoln. You should account for capitalization and possessive forms.
11. Timestamp: Every entry will be preceded by the time it was received by the client.

Submissions

You will submit a printed copy of your code with cover sheet, including documented sources and help as required. You must also email me a copy of your code in a .zip file. **IMPORTANT:** Include a "README:" comment section at the top of your code indicating which port number your server listens on.



Resources

A simple client shell is provided as `chatClientShell.py`. This should help you get started.

The `chatServer.py` program is available for you to test your client with. It will be running on `instructor.cs484.eecs.net:9000`

Two example chat transcripts are at the end of this brief. The two clients were talking to each other using the chat server.

Pro Tips

- We will test your client programs during Lesson 20. Make sure to test your implementations across the network and not just on your local computer.
- Python threading documentation: <https://docs.python.org/3.4/library/threading.html>
- Start with a simple working rudimentary chat client example and add features to it incrementally.
- Test your additions as you go.
- The “reader thread” can be abruptly stopped at shutdown. In other words, you do not need to call `join()` on the reader thread. In fact, calling `join()` on the reader thread is useless; the reader thread is designed not to end. If you read the Python threading documentation, this will make sense.



```
instructor@CS484: ~/18-1/Project2
instructor@CS484:~/18-1/Project2$ python3 chatClient.py
Welcome to Cadet Chat!
Please enter destination server (localhost):
Please enter port number (5000):
please enter user handle: John

Cadet Chat commands:
#user <handle> : changes user handle
#block <user> : blocks this user's messages
#unblock <user> : unblocks this user's messages
#help : shows this screen
#bye : quits chat program
Have fun!

John has joined the conversation
Caroline has joined the conversation
Hello Caroline
Caroline: hello john
how are you
Caroline: good u
im blocking you now
#block Caroline
User Caroline has been blocked
Caroline2: Caroline has changed to Caroline2
Caroline2: im back!!!
lol
#unblock Caroline
User Caroline has been unblocked
Caroline2: later!
Caroline2 has left the conversation
#bye
Disconnecting from Server

instructor@CS484:~/18-1/Project2$
```

```
instructor@CS484: ~/18-1/Project2
instructor@CS484:~/18-1/Project2$ python3 chatClient.py
Welcome to Cadet Chat!
Please enter destination server (localhost):
Please enter port number (5000):
please enter user handle: Caroline

Cadet Chat commands:
#user <handle> : changes user handle
#block <user> : blocks this user's messages
#unblock <user> : unblocks this user's messages
#help : shows this screen
#bye : quits chat program
Have fun!

Caroline has joined the conversation
John: Hello Caroline
hello john
John: how are you
good u
John: im blocking you now
im blocked
i can get around this
#user Caroline2
User handle changed to: Caroline2
im back!!!
John: lol
#help

Cadet Chat commands:
#user <handle> : changes user handle
#block <user> : blocks this user's messages
#unblock <user> : unblocks this user's messages
#help : shows this screen
#bye : quits chat program

later!
#bye
Disconnecting from Server
```