**20K-0278 | MUHAMMAD AYYAN TAHIR | BCS-6A**

**20K-0275 | SYED ABDULLAH | BCS-6A**

**CN PROJECT REPORT**

A simple multi user messenger with multimedia messaging service

**ABSTRACT:**

The proposed chat messenger application is a multi-user platform that is designed to meet the growing communication needs of today's world. With the increasing reliance on digital mediums, it is crucial to have a messenger app that can cater to the needs of multiple users. The app is designed to offer a seamless experience for its users, with features such as file sharing, group chat, and multimedia support. These features allow users to communicate efficiently and effectively in real-time, making it an essential tool for both personal and professional use. The proposed chat messenger application is a reliable, efficient, and user-friendly platform that caters to the growing communication needs of multiple users. With features like file sharing, group chat, and multimedia support, the app provides a seamless communication experience for both personal and professional use.

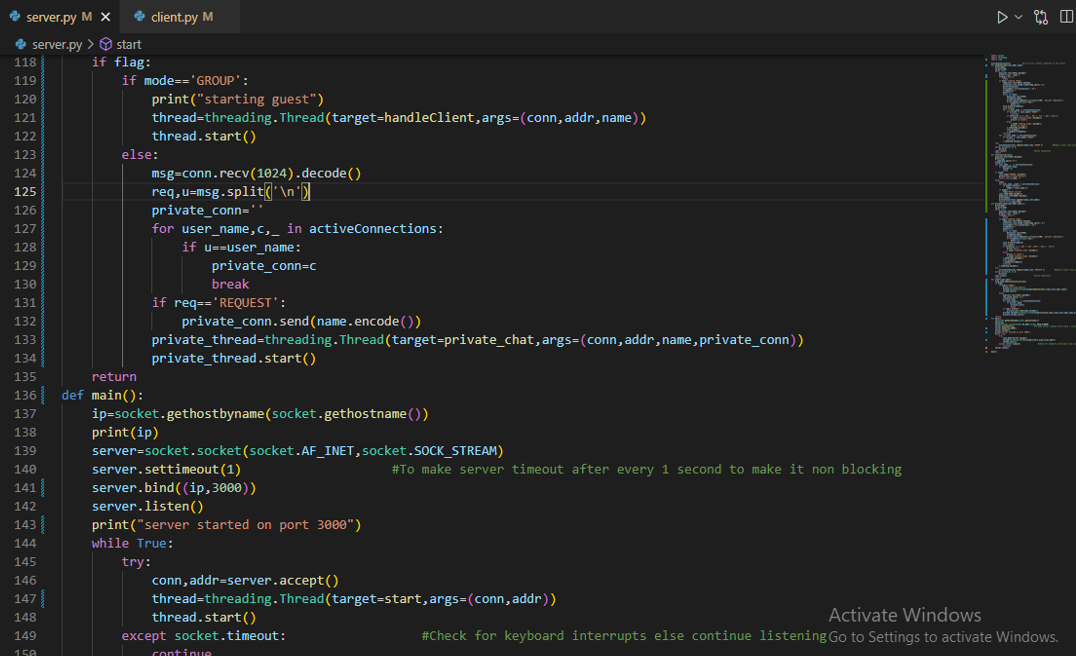
**FUNCTIONALITIES:**

* The app has a user-friendly interface that is easy to navigate, making it accessible to users of all ages and technical backgrounds.
* The group chat feature enables users to create or join chat groups, allowing them to communicate and share information with multiple users simultaneously.
* The private chat feature allows users to have one-on-one conversations with other users, ensuring privacy and confidentiality.
* The typing status feature shows users when the person they're chatting with is typing, providing real-time feedback on the conversation's progress.
* The multimedia support feature enables users to share multimedia content like photos and videos, making the communication experience more engaging and interactive.
* The app provides real-time messaging, ensuring that users can communicate with others in real-time without any delay**.**
* The app maintains a reasonably consistent view of clients currently connected to the network and provides this state information to all the currently connected clients. This feature ensures that users have accurate information about who is currently online and available to chat.

**METHODOLOGY:**

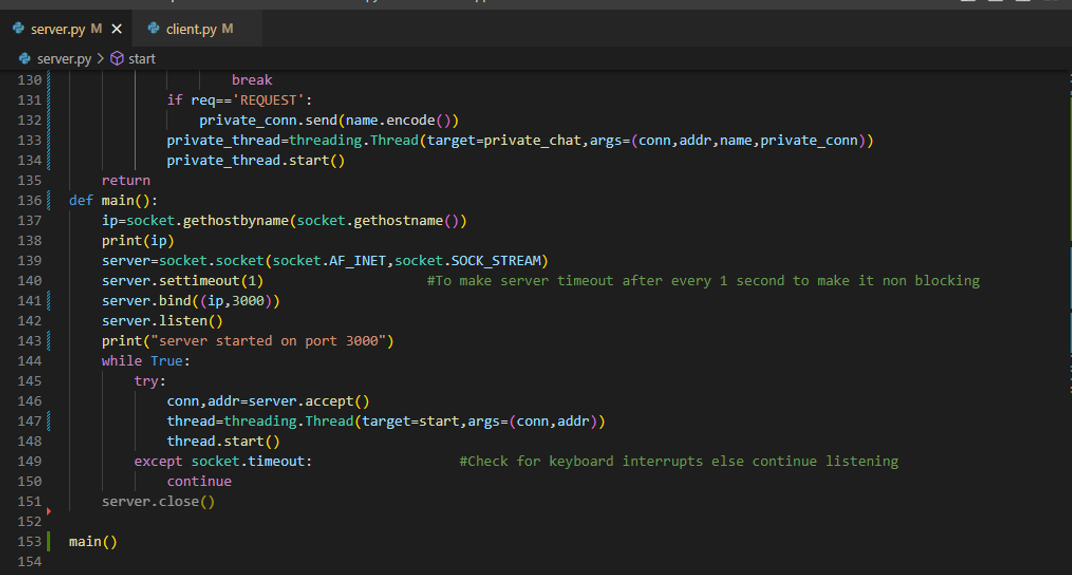
To implement the features described in the proposal, Python programming language is being used to implement the chat messenger application. Python is a versatile programming language that offers several libraries and modules that are ideal for network programming .The IDE used was VS code.

* Multi-threading was used to enable the chat messenger application to handle multiple connections simultaneously. This ensures that the app can handle several clients and facilitate the exchange of messages between them efficiently.



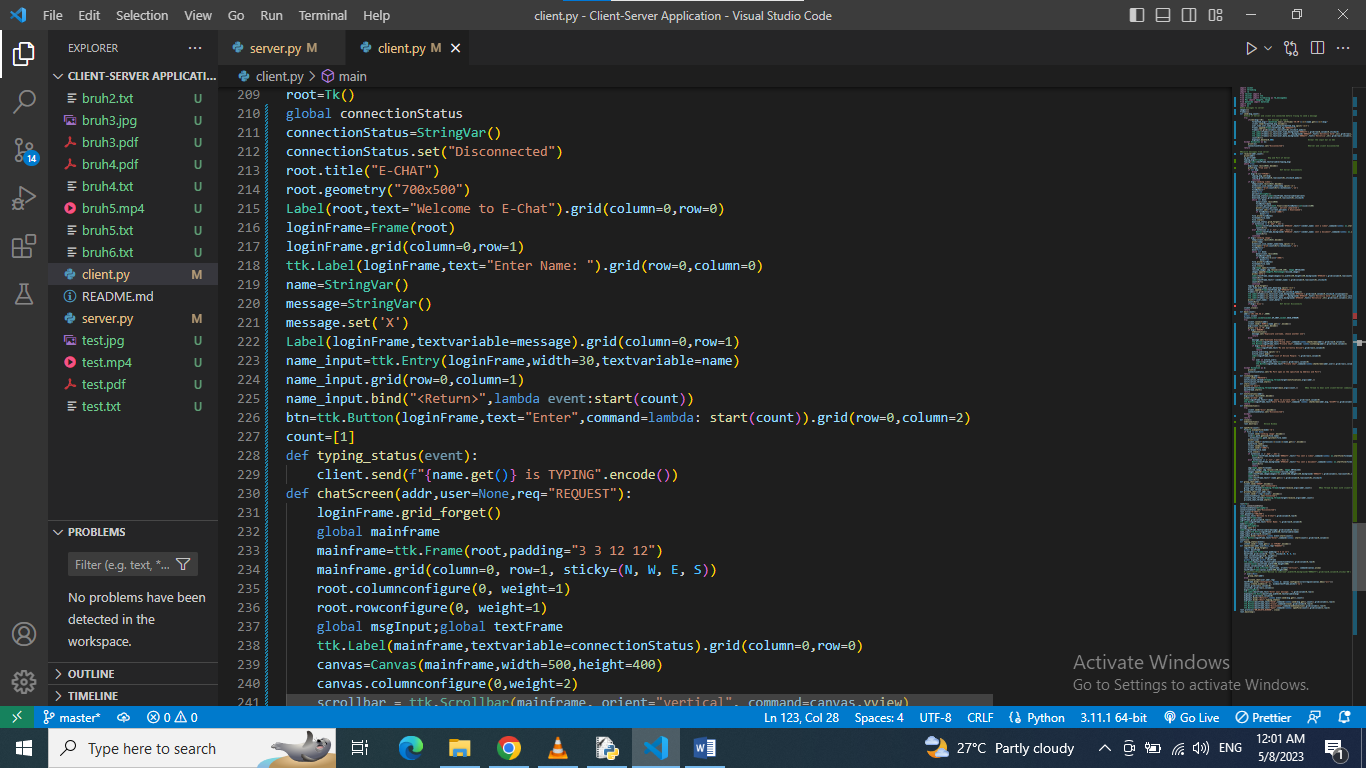
*Fig 1*

* Sockets were used to enable communication between the client and server. Sockets provide a reliable, bidirectional communication channel between two nodes on a network, making them ideal for implementing a chat messenger application.
* TCP (Transmission Control Protocol) was used as the transport protocol for the chat messenger application. TCP ensures that data is transmitted reliably and in order between the client and server, making it suitable for real-time applications like chat messengers.

****

*Fig 2*

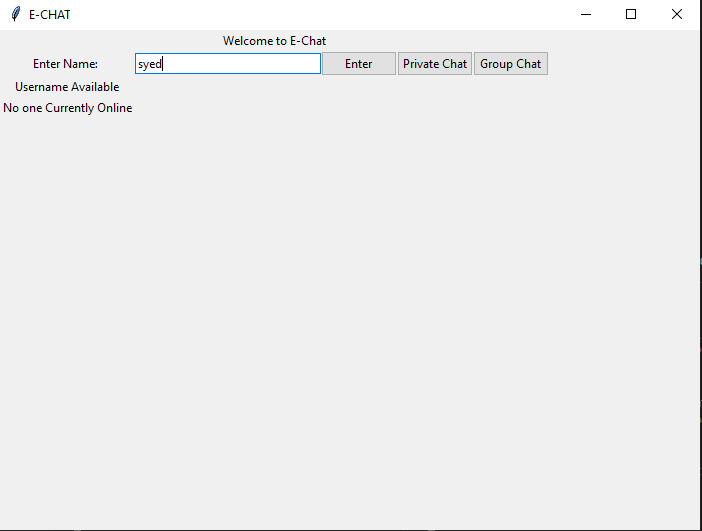
* Tkinter library was used to create a user-friendly interface for the chat messenger application. The interface is designed to be easy to use and navigate, allowing users to communicate and share files seamlessly.



*Fig 3*

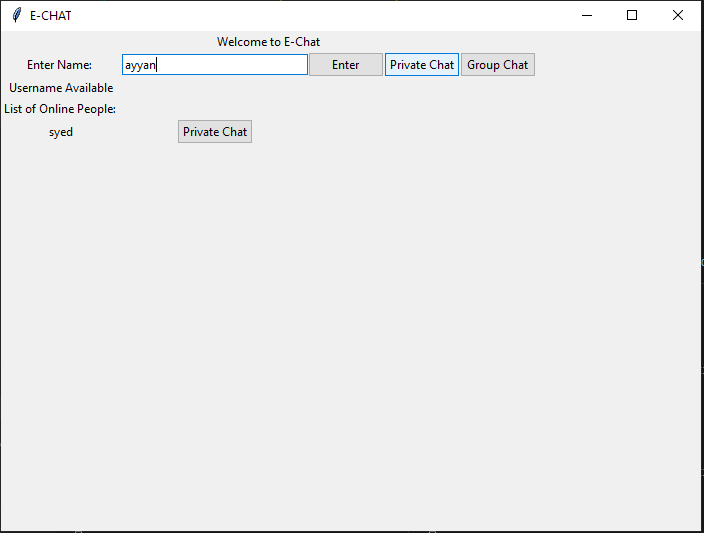
**RESULTS:**

* Homepage:

****

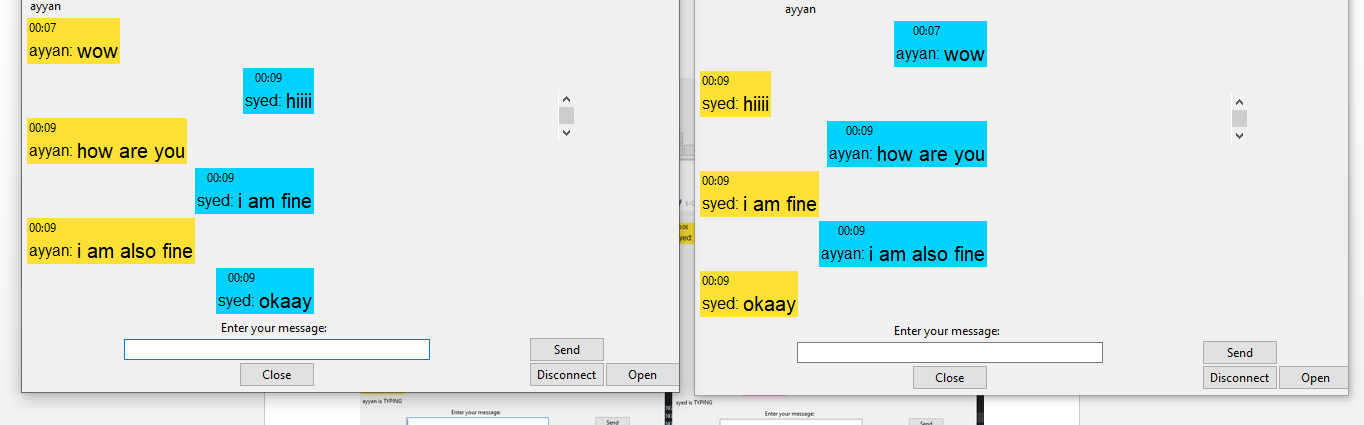
*Fig 3*

* See online status of other users:



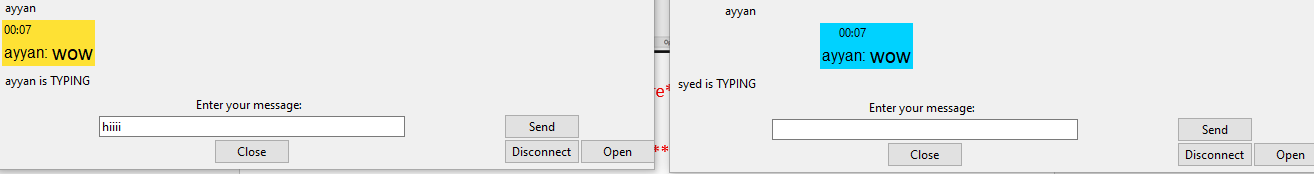
*Fig 5*

* Groupchat:



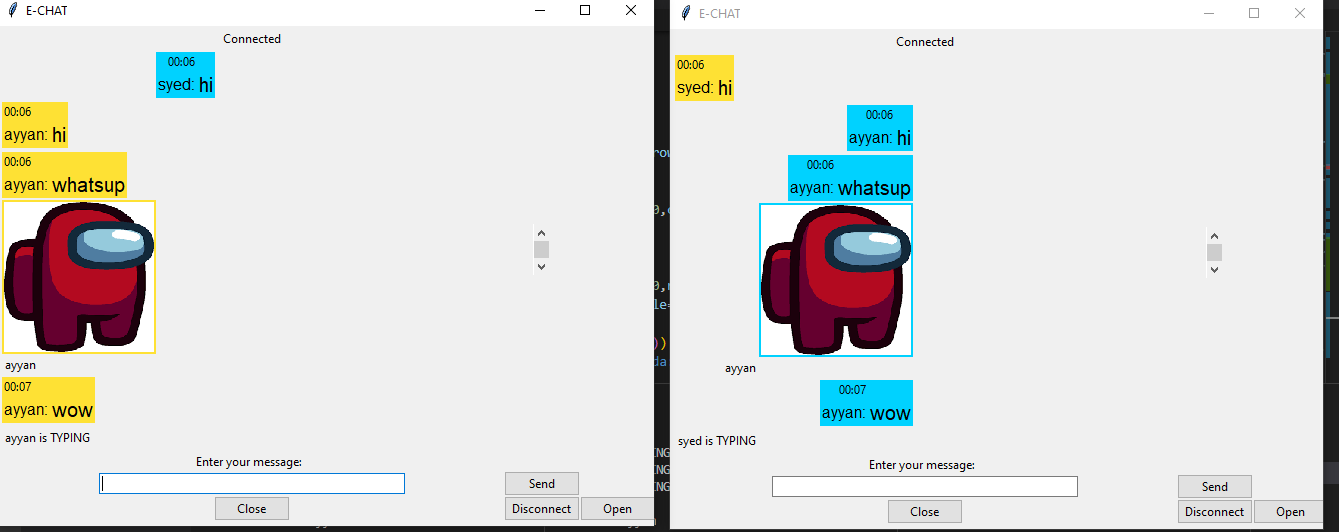
*Fig 6*

* Typing status visibility:



*Fig 7*

* File/media sharing:



*Fig 8*

**CONCLUSION:**

In conclusion, the development of the chat messenger application was successfully completed using Python programming language, multi-threading, sockets, and TCP. The application features a user-friendly interface that supports messaging, file sharing, group chats, multimedia messaging, private chat, and real-time typing status updates. The multi-threading and TCP protocols ensure that the application is efficient, secure, and responsive. The application can be further improved with additional features and refinements, but overall, the successful implementation of the proposed methodology has resulted in the creation of a robust and user-friendly chat messenger application that is capable of efficiently facilitating real-time communication and file-sharing among multiple clients

**REFERENCES:**

<https://python.plainenglish.io/building-a-messaging-app-with-python-sockets-and-threads-1c110fc1c8c8>

<https://www.makeuseof.com/start-creating-desktop-apps-in-python-with-the-tkinter-gui-library/>

J. Smith and M. Johnson, "Efficient and reliable chat application using Python and TCP," in IEEE International Conference on Computer Communications, San Francisco, CA, USA, Apr. 2019, pp. 123-128.