CS232 Course Project

Project Title: Hybrid Chat Application

Project Description: This project involves creating a chat application that supports both client-server and peer-to-peer (P2P) communication. The client-server model can be used for user authentication, maintaining the list of online users, and initial connection setup. The P2P model can be utilized for direct messaging between users to reduce server load and improve scalability.

Key Features and Requirements:

1. **Client-Server Model for User Management:**
    - Implement a server that manages user registrations, sign-ins, and maintains the list of currently online users.
    - The server should handle client connections, authentication, and direct clients on how to initiate P2P connections.
2. **Peer-to-Peer Communication for Direct Messaging:**
    - Once the initial connection is established through the server, allow direct messaging between clients using a P2P architecture.
    - Implement functionality for sending and receiving messages in real-time between two clients without server intervention.
3. **User Interface:**
    - Develop a simple GUI (Graphical User Interface) for the client application, allowing users to log in, view online users, and initiate chats.
    - The interface should include separate chat windows for different conversations, with the ability to send text messages and possibly file attachments.
4. **Networking Concepts:**
    - Apply TCP/IP sockets for client-server communication and for establishing P2P connections.
    - Explore NAT traversal techniques to enable P2P connections between clients behind NATs (Network Address Translators).
5. **Security Features (Optional):**
    - Implement basic encryption for the messages sent between clients in the P2P mode to ensure privacy.

- Include authentication mechanisms to secure user logins and registrations.

Learning Outcomes:

- Understand and apply client-server and peer-to-peer networking models.
- Gain practical experience with socket programming and managing network connections.
- Explore NAT traversal techniques and their importance in real-world networking.
- Develop an understanding of basic network security concepts and encryption.

Assessment Criteria:

- Code Quality: Organization, readability, and adherence to networking best practices.
- Functionality: The implementation of both client-server and P2P architectures, including user authentication, online presence, and direct messaging.
- Usability: The intuitiveness and responsiveness of the user interface.
- Innovation: The incorporation of additional features or improvements to the basic requirements.

This project not only allows students to delve into the practical aspects of network programming but also provides them with an opportunity to understand and compare different networking architectures in a real-world application context.

Note: Students can also work on any other idea if it covers the same concepts.

Deadline

- **Sunday May 12, 2024**