Java Chat and File Transfer System

Course Project – IIT (BHU) MNC
Contributors:

- **Aviral Shukla** (Roll No: 24124012)
- **Arnav Raghuvanshi** (Roll No: 24124007)
- **Aryan Raj** (Roll No: 24124009)

Project Title:

Real-Time Java-Based Chat and File Transfer Application

Project Overview:

This project is a **Java-based desktop application** that mimics the core functionalities of a messaging platform like WhatsApp. It allows users to communicate in real-time using a Graphical User Interface (GUI) and also send files seamlessly over a local network. It demonstrates client-server communication using Java Sockets and includes custom-built components for chat handling and file transfers.

The primary objective is to provide a simple, efficient, and easy-to-use messaging solution that supports multi-client communication and file sharing.

Key Features:

- **Real-time Chat Messaging** using TCP sockets
- **File Transfer Support** between clients
- **Modern GUI** using Swing and JTextPane
- **Message Formatting** with colors for different users
- **Threaded Server** to handle multiple clients concurrently
- **Clean, user-friendly interface**
- **Feedback messages** for file sending and chat status

Tech Stack Used:

- **Language:** Java
- **GUI:** Java Swing
- **Networking:** Java Sockets, Input/Output Streams
- **Java Extensions Used:**

- javax.swing.* for GUI
- java.io.*, java.net.* for networking
- javax.swing.text.* for text formatting

Architecture:

The application follows a **Client-Server architecture**:

1. **Server**
    - Waits for connections using ServerSocket
    - Spawns a new ClientHandler thread for every client
    - Broadcasts messages to all connected clients
2. **Client**
    - Connects to server with Socket
    - Sends messages and files
    - Receives and displays messages from others in the chat GUI

Screenshots/Illustrations:

- Chat GUI with color-coded messages
- File transfer confirmation message
- Terminal showing server logs
- Multi-client communication in action

Individual Contributions:

Aviral Shukla – Roll No: 24124012

Component: Server and Client Handler

- Implemented multi-threaded server using Server Socket
- Developed Client Handler class to manage each client connection
- Handled broadcasting of messages and file data to all clients
- Worked with input/output streams and server-side message logic

Skills Used: Java Networking, Multithreading, Input/Output Streams

---

Arnav Raghuvanshi – Roll No: 24124007

Component: Chat GUI

- Designed GUI using Java Swing
- Used JTextPane and StyledDocument for styled messages
- Integrated buttons for chat and file features
- Ensured responsiveness and real-time updates
- Handled UI updates from multiple threads safely

Skills Used: Java Swing, Event Handling, UI/UX with JTextPane, JPanel, JButton

---

Aryan Raj – Roll No: 24124009

Component: File Transfer System

- Implemented file selection using JFileChooser
- Sent file bytes over the socket in a separate thread
- Received files and saved them correctly on the receiver side
- Managed large file handling, stream buffering, and exceptions

Skills Used: Java IO, File Handling, Threads, JFileChooser

How to Run the Project:

1. Start the server (Server.java)
2. Run multiple clients (ChatGUI.java) with different usernames
3. Chat in real-time and use the **File** button to transfer files
4. The receiver gets a confirmation, and the file is saved in the working directory

---

Conclusion:

This project combines GUI design, socket programming, multithreading, and file handling in Java. It showcases our understanding of object-oriented programming and real-world software design. The application is simple but demonstrates the foundational principles behind professional messaging platforms.