

Assignment No-6

Title- multi-threading chat system simulation using java sockets.

objectives-

To design and implement a multi-threading chat server in java can handle multiple client connections simultaneously. each client should be able to send and receive message in real-time through the server using socket programming and threading.

Theory-

A chat system is a communication platform that allows multiple users to exchange text msg in real-time. In this stimulation we implement a client server architecture using java socket programming and multithreading.

key concepts-

client-server model

A central server handles multiple clients connect to the server using IP and Port number.

socket programming

A socket represents a client connections

A server socket listens for incoming connections.

multithreading -

each client connection is handled in a separate thread using the runnable interface. This ensures that server can interact with multiple clients concurrently.

Broadcast mechanism -

The server receives message from one client and forwards (broadcasts) them to all other connected clients.

Algorithm -

server side (chatServer.java)

- 1) start the server and listen on a specific port.
- 2) create a data structure
- 3) Accept client connections in a loop.
- 4) for each connections
create a new client handle object.
start a new thread for the client handler
- 5) In each client handler thread:
Read msg from the client. Broadcast the msg to all other clients handle client disconnection and cleanup.

client side (chatclient.java)

connect to the server using IP address and port number.

- start a reader thread to listen for incoming msg from the server.
- In the main thread, continuously read user IP and send msg to the server
- handle disconnection gracefully.

conclusion -
The project successfully demonstrate a basic multithreading chat system in java using socket programming and thread to a central server and exchange message in real time.

✓ (C) psr