

CISC 5597 Distributed Systems

Lab 1: Networking — Multiple users chatting

Sample codes:

The sample code is an example of a networked TCP server and associated client. The server opens a passive socket and listens for connection requests from clients.

DEMO-1. It accepts at most one connection request. Once a client connects, the server sends the message of the system time to the client and shuts down.

DEMO-2. It accepts multiple clients by using threads. Once a client connects, the server can send messages to different clients.

Lab Assignment:

Modify the server and client so that the system meets the following requirements:

1. The server can accept multiple clients and assign each of them a unique ID. When connected, the server sends back this ID to the client.
2. The server can accept several different commands:
 - a. **list**: The server sends back all the active client IDs.
 - b. **Forward ID string**: The server should be able to understand that this client wants to send the msg(string) to the other client with the ID that listed the command. The server should be able to forward the message to the target in the following format: source ID: message_content
 - c. **history ID**: The server should send back the chatting history between the requested client and the client with the ID listed in the command.
 - d. **exit**: The server should send back a message "Goodbye" and close the connection.

Grading Rubric

Your code should be elegant and well-documented (with comments).

- Video demonstration (Required).

1. The submission can not run successfully. (0 points)
2. The submission works for a 1-1 connection. (10%)
3. The submission can accept multiple connections from clients. (20%)
4. The commands "list" and "exit" work. (10%)
5. The command "forward ID string" works. (25%)
6. The command "history id" works. (25%)
7. The submission, along with a report, is clear in logic and easy to read. (10%)