Distributed Systems Lab - Assignment # 2

Q.1. Use TCP sockets to build a simple client-server system, where you use the client to chat with a dummy server. The protocol between the client and server is as follows:

a) The server is first started on a known port.

b) The client program is started (server IP and port are provided on the command line).

c) The client connects to the server, and then asks the user for input. The user types his message on the terminal (e.g., "Hi", "Bye", "How are you"). The user's input is sent to the server via the connected socket.

d) The server reads the user's input from the client socket. If the user has typed "Bye" (without the quotes), the server must reply with "Goodbye". For any other message, the server must reply with "OK".

e) The client then reads the reply from the server, and checks that it is accurate (either "OK" or "Goodbye").

f) If the user had typed "Bye", and the server replied with a "Goodbye" correctly, the client quits. Otherwise, the client asks the user for the next message to send to the server.

If the server sends a wrong reply, say, "NO" instead of "OK", the client exits with error message.

Q.2. Use TCP Sockets to create a simple client-server based chat application. In the application, two clients connect to the same server and exchange messages between them. The server relays these messages from one client to the other. Both the clients must have proper interfaces for sending and receiving chat messages.

NOTE: The client and server codes can initially be tested locally using the loopback address 127.0.0.1, but eventually the codes are to be tested on different machines.