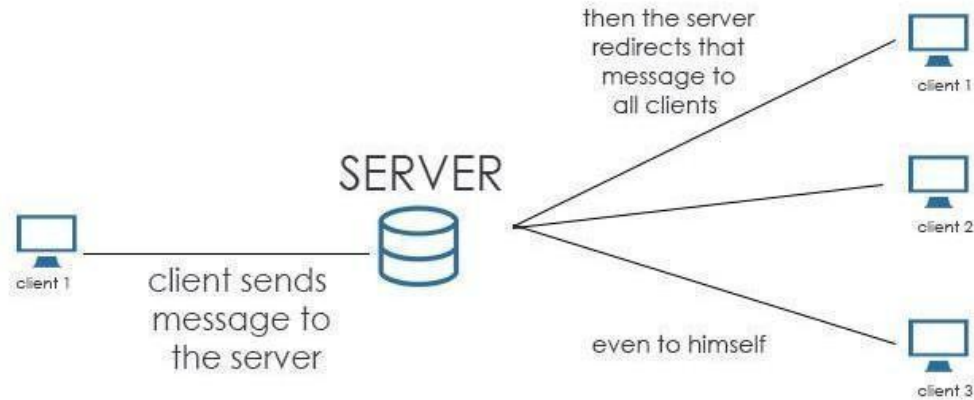


## **INSTANT CHATTERS APPLICATION**

### **HIGH LEVEL DESIGN**

#### **Flow Diagram of IC:**

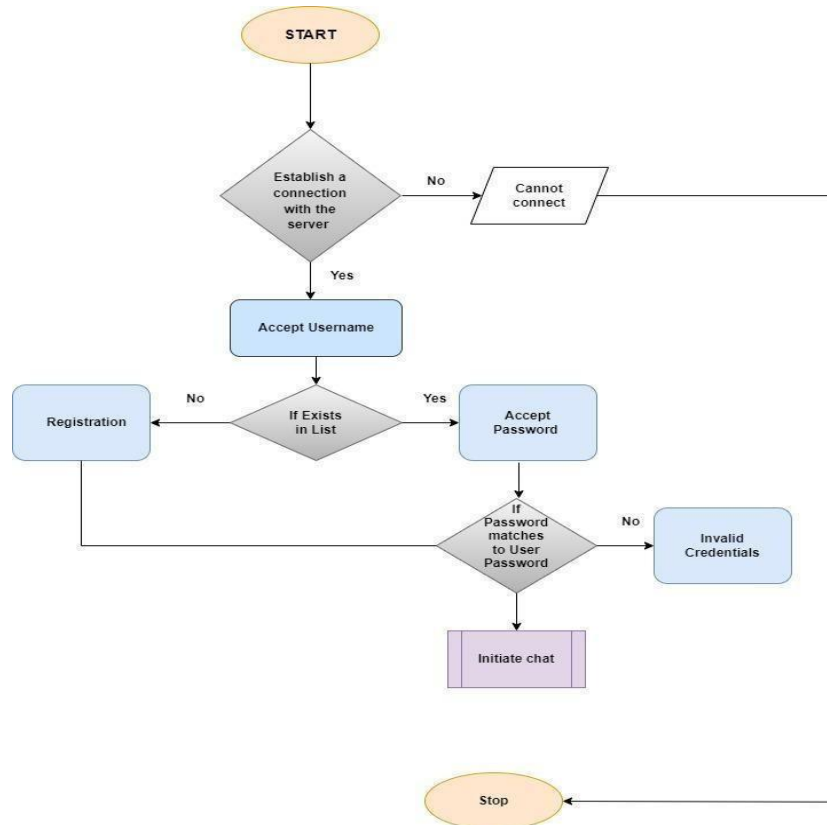


#### **IC Process Overview:**

The purpose of this project is to implement a socket programming-based chat application that will allow to create a multiple client to chat in private conversations.

- This project is to create a chat application with the server and clients to enable the clients to chat with many other clients. This project can play an important role in organization field where employees can be connected through internet.

### Procedure flow:



### **For User registration:**

The user registration is to be created for clients and the data of the registered user is stored in server in a file. For login the details from the clients get compared with the data file stored in server.

### **User Validation For Client:**

The main function is to Compare the entered username and password with registered username and password stored in the data file in the server.

### **Message Exchange:**

Sending messages from client and receiving messages from other users and sending messages from user (server side), receiving messages from the user (server side) and send to specified client. The data is saved for chat history of two clients with the file name (client1\_client2\_date) and displaying active users (server side).

### **New User:**

The user registration is to be created for clients and the data of the registered user is stored in server in a file. For login the details from the clients get compared with the data file stored in server.

### **Registered User:**

The registered user is to Compare the entered username and password with registered username and password stored in the data file in the server.

### **Login:**

Login failure of the client results to successive 3 attempts and more than 3 attempts leads to delay of 15 minutes and failure to log in provides client some specific attempts to recover his login id credentials.

### **Active Users:**

It shows the active users who are all logged in the application. And the logged in user can chat with any of active users.

### **Message Exchange:**

Sending messages from client and receiving messages from other users and sending messages from user (server side), receiving messages from the user (server side) and send to specified client.

### **Chat history Search:**

The data is saved for chat history of two clients with the file name (user1\_user2\_date) and displaying active users (server side). The data is searched by the given user input and date.

### **Logout:**

The logout is used to exit from the chat application by the logged in users. After logged out, the users can also able to login again