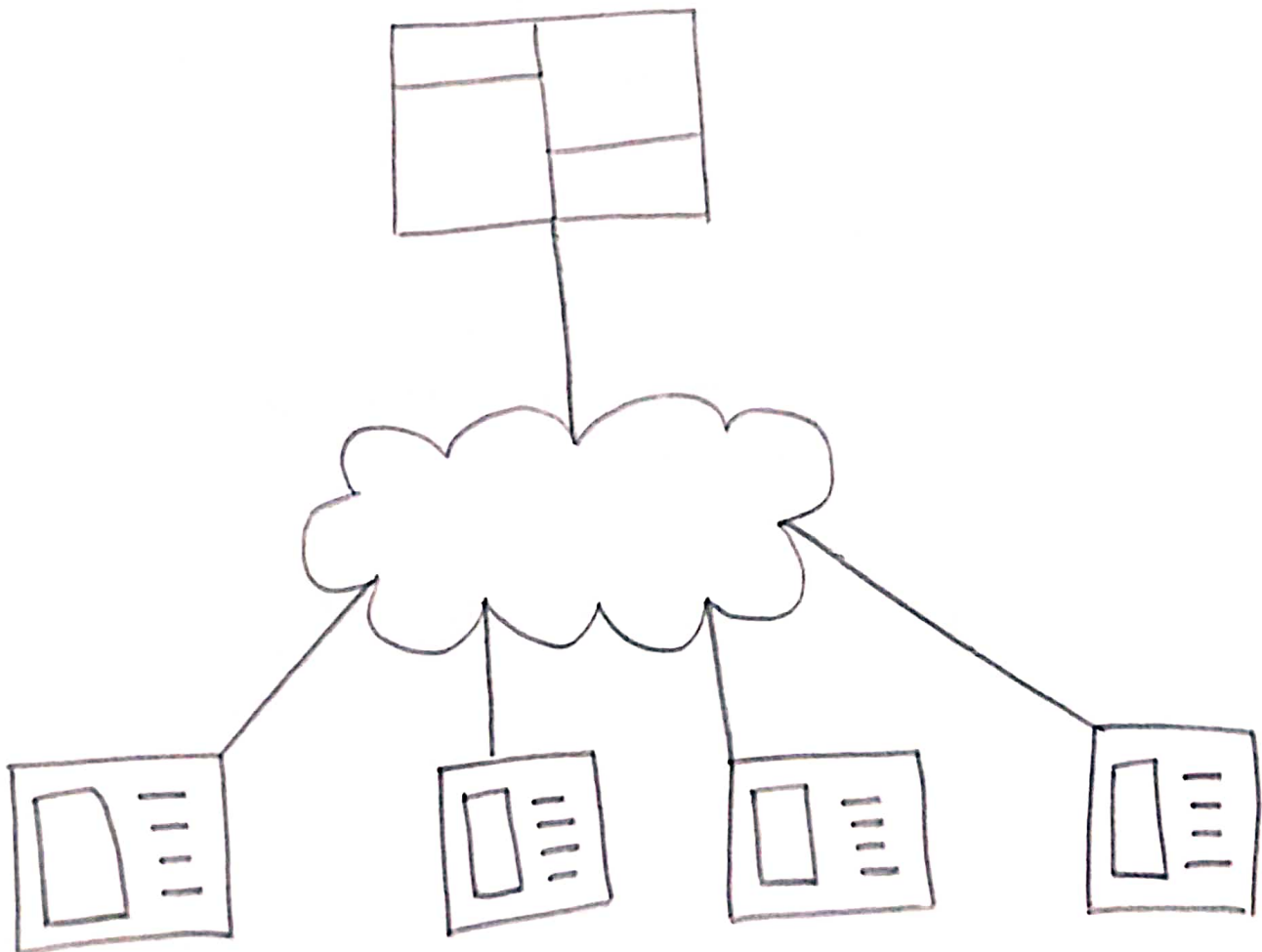# Activity - 02

Name : Om Shree

Roll no : 2006077

Section : IT - 05

Subject : Software Enginuring

Topic : SRS document of a multicast intranetwork chat application

- **Introduction :-**

→ Purpose of this project is to implement a Java based chat application that will allow users with an internet connection to engage in private & public conversations.

→ Development of this project is centred upon the development of a message protocol that would allow the application to properly log in users, send messages, and perform system maintenance.

- **Problem introduction :-**

→ Project has to create a chat application with a server and clients to enable the clients to chat with many other clients in the same common chat group.

→ Project is to simulate the multicast chatting In the case of multicasting when a message is sent to a group of clients, then only a single message is sent to the router.

→ Main purpose of this project is to provide multi chatting functionality through network.

→ This project can play an important role in organizational field where employees can connect through LAN.

→ Project doesn't provide encryption facility.

• **Innovative ideas of this project :—**

→ GUI : easy to use GUI (graphical user interface), hence any user with minimal knowledge of an operating system can use this software.

→ Platform independence : The messenger operates on any operating system irrelevant of the underlying operating system.

→ Unlimited clients : 'n' number of users can be connected without any performance degradation of the server.

• **Project objective :—**

→ Communication : to develop an instant messaging solution to enable users to seamlessly communicate with each other

→ User friendliness : The project should be very easy to use enabling even a novice person to use it.

• Scope of the project :-

→ Broadcasting chat server application is going to be a text communication software, it will be able to communicate between 2 computers using point to point communication.

→ The limitation of live chat is it doesn't support audio communication.

→ The fact that the software uses an internal network setup within the organization makes it very secure from outside attacks.

• Project perspective :-

→ There is 2 way communication between different clients and server. This chat application can be used for group discussions. It allows using to find other logged in users.

→ No need for internet communication:

Only intranet connection i.e. a LAN connection is required. This system is useful for those who can't afford to have an internet connection.

For example: schools, colleges & small companies

→ Conference possible on LAN:

Usually on LAN connections conferencing is not possible. The proposed system allows the LAN users to create & participate in conference. This makes communications possible among no. of LAN users simultaneously.

• <u>Interface</u> :—

→ This application interacts with the user through GUI. The interface is simple, easy to handle & self-explainatory.

→ Once opened, user will easily come into the flow of application & easily use all interfaces properly.

→ Basic interface available in our application is :—

① Title panel
② Content panel
③ Admin panel

- Hardware interface :–

  → Minimum requirements will be as follows :–

    - 1 GB RAM
    - Processor speed > 1 GHz
    - Internet or LAN connection
    - Mouse : 2 or 3 button mouse
    - Keyboard : 101 key keyboard

- Software interface :–

  → Notepad++ is a text editor and source code editor and provides an environment for developing HTML, jsp, JS and many other editing purposes.

  → Coding done in Java. JDK 1.4 or above is necessary.

  → Operating system : Linux

- Constraints :–

  → The application does not by any means open the web-browser. If user wishes to open the web-browser, he must do so externally.

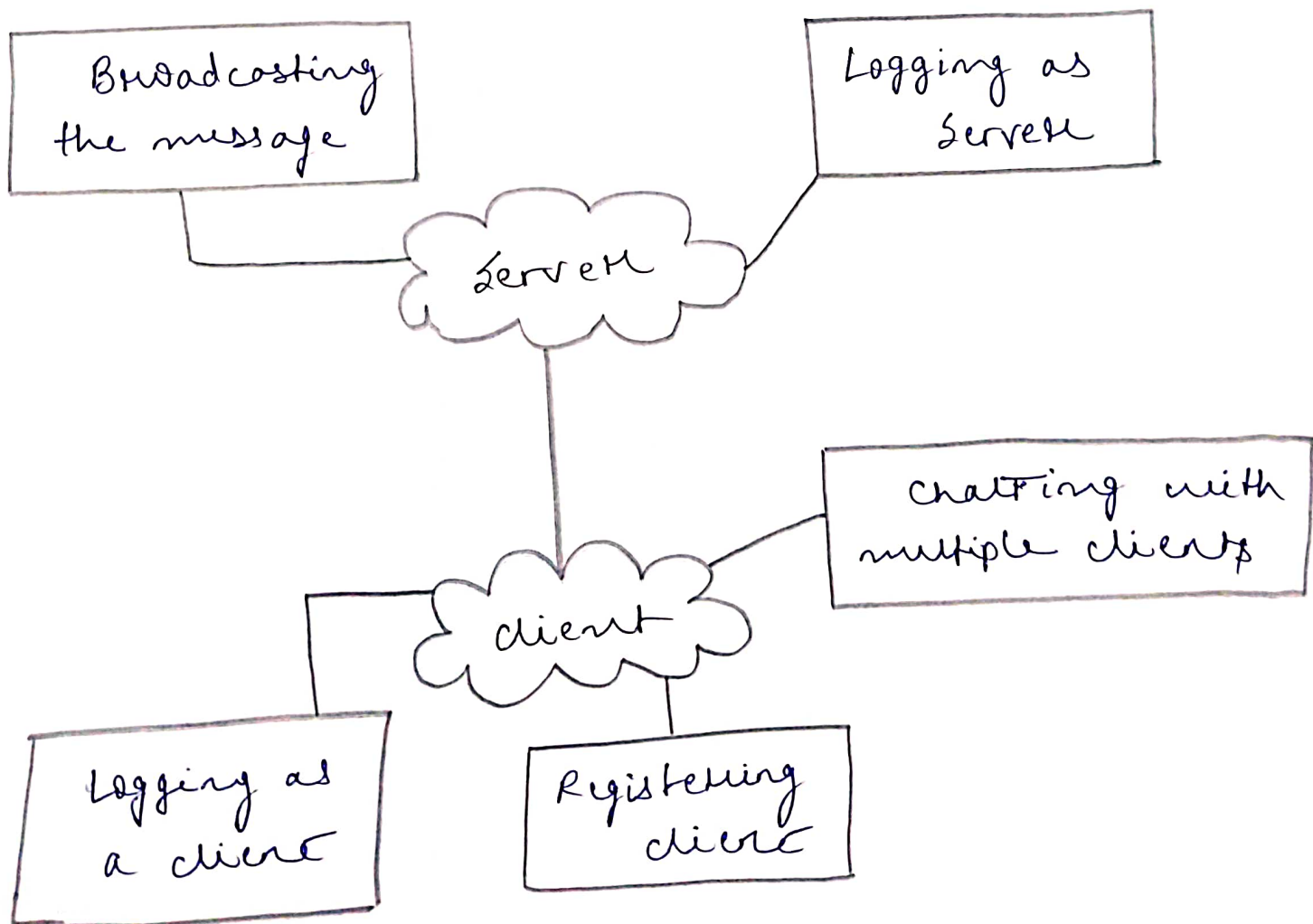 → The system needs to be permanently connected with internet.

- **Product Functions :—**

→ There is 2 way communication between different clients and server.

→ This chat application can be used for group discussion.

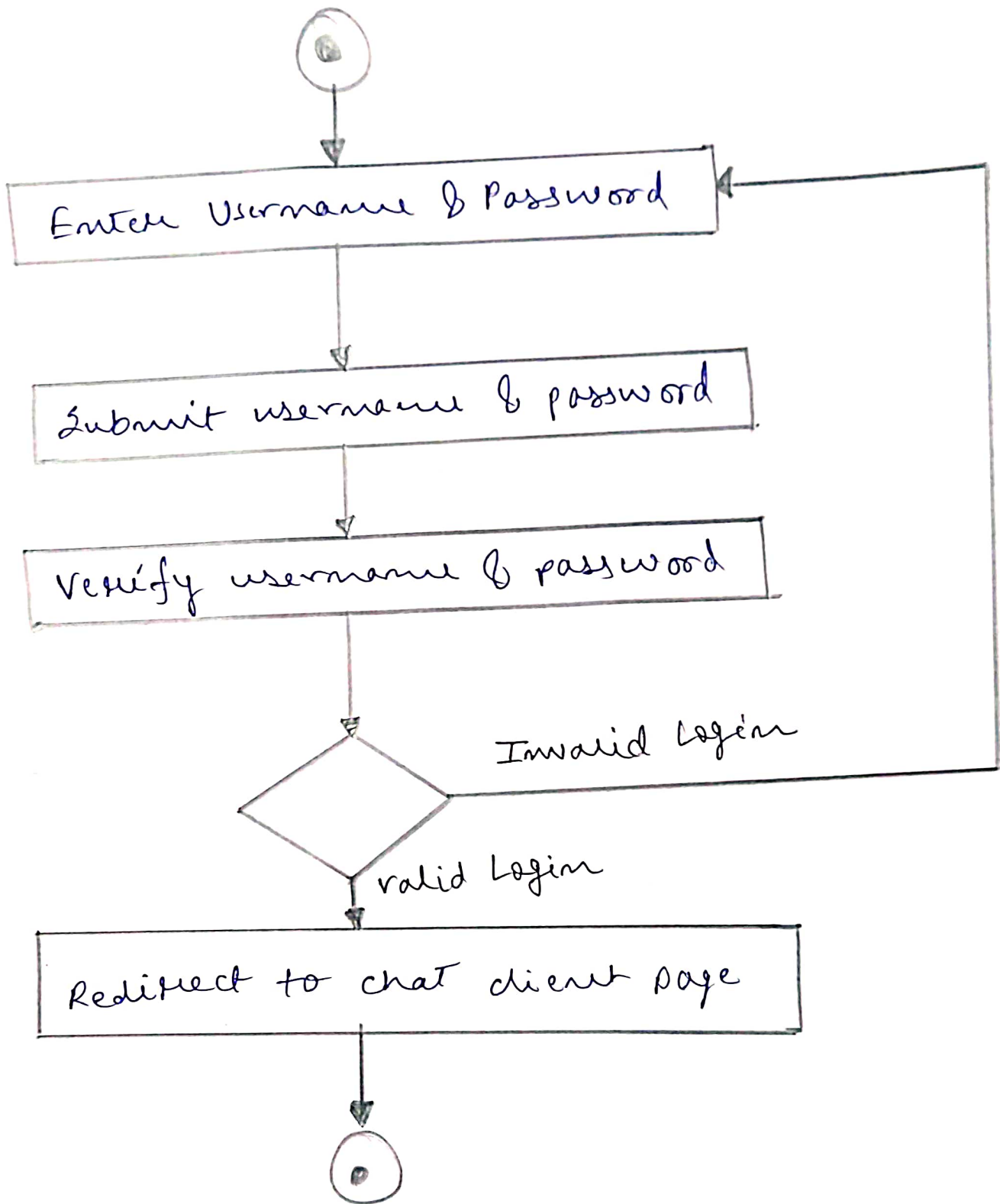→ It allows users to find other logged in users.

- **Assumption and dependencies :—**

→ There should be LAN or internet connection

→ Client should know each other.

→ There can be multiple clients.

- **Use - case diagram :—**

```
┌─────────────────┐                    ┌─────────────────┐
│  Broadcasting   │                    │  Logging as     │
│  the message    │                    │    Server       │
└─────────────────┘                    └─────────────────┘
         │                 ╭─────╮            │
         └─────────────────┤Server├───────────┘
                           ╰──┬──╯
                              │
                              │                ┌──────────────────┐
                              │                │  Chatting with   │
                        ╭─────┴─────╮──────────│  multiple clients│
                ┌───────┤  client   ├───────┐  └──────────────────┘
                │       ╰───────────╯       │
        ┌───────┴──────┐          ┌─────────┴────────┐
        │  Logging as  │          │   Registering    │
        │  a client    │          │     client       │
        └──────────────┘          └──────────────────┘
```

• Activity diagram for login function :—

```
                    ( ◎ )
                      │
                      ▼
   ┌──────────────────────────────────┐◀──────────┐
   │  Enter Username & Password        │           │
   └──────────────────────────────────┘           │
                      │                            │
                      ▼                            │
   ┌──────────────────────────────────┐           │
   │  Submit username & password       │           │
   └──────────────────────────────────┘           │
                      │                            │
                      ▼                            │
   ┌──────────────────────────────────┐           │
   │  Verify username & password       │           │
   └──────────────────────────────────┘           │
                      │                            │
                      ▼                            │
                    ◇─────── Invalid Login ────────┘
                    ◇
                      │  valid Login
                      ▼
   ┌──────────────────────────────────┐
   │  Redirect to chat client page     │
   └──────────────────────────────────┘
                      │
                      ▼
                    ( ● )
```
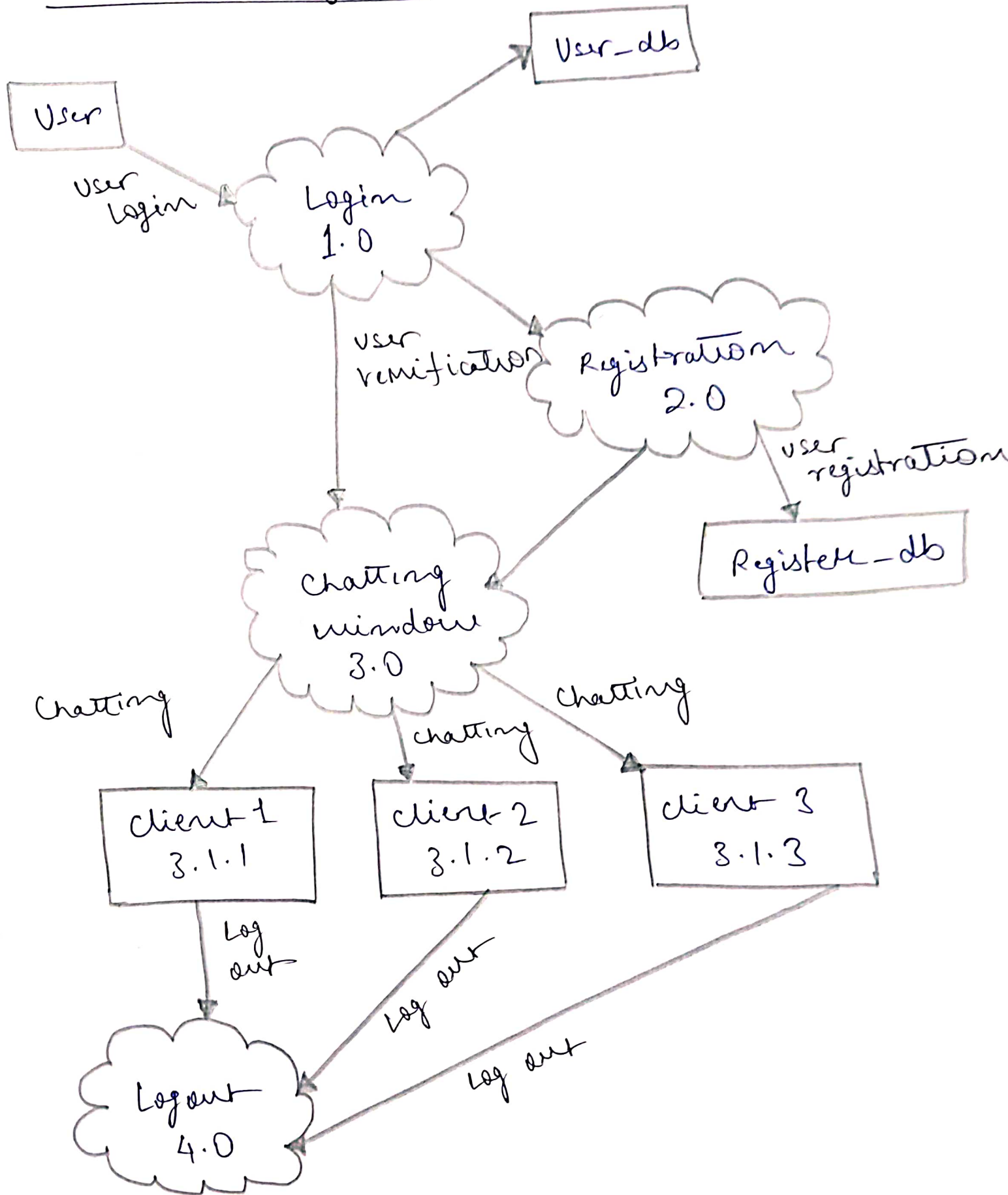
- Activity diagram for chatting :—
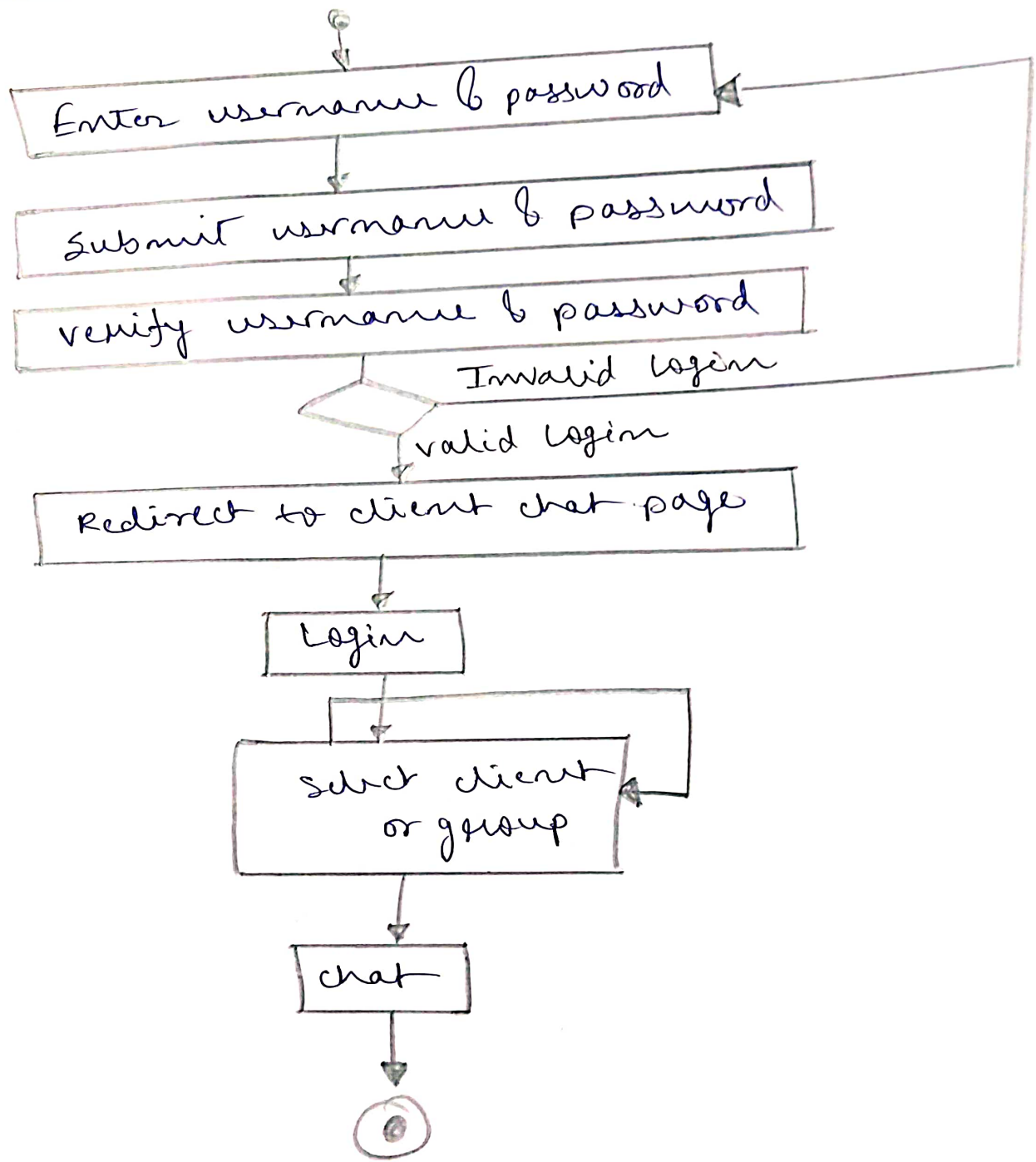
→ The diagram represents an interdependen
-cy between the login & chatting
facilities.

- **Data flow diagram :—**

User_db

User

User Login

Login
1.0

User verification

Registration
2.0

user registration

Register_db

Chatting window
3.0

Chatting

chatting

Chatting

client 1
3.1.1

client 2
3.1.2

client 3
3.1.3

Log out

Log out

Log out

Log out

Logout
4.0

- ER-diagram :—

Server

Server sends the message to the respective client or broadcasts in every client screen
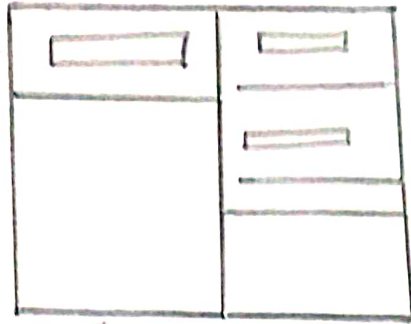
message passes on to the server

Intranetwork

client sends message to server via LAN
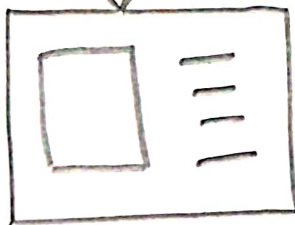
message comes to the respective client again through LAN
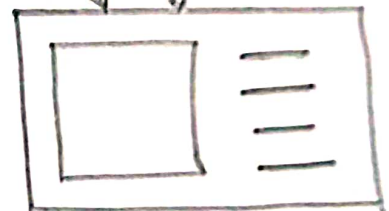
client 1

client N

- Checklist of various types of errors that can exist in an SRS document :-

① Omission : Some user requirements are not included in the SRS document. Omitted requirements may be related to features, performance, constraints or any other factor.

② Inconsistency : Can be due to contradictions within the requirements themselves or incompatibility of stated requirements with actual requirements of the client.

③ Incorrect fact : Some facts recorded on the SRS document are incorrect.

④ Ambiguity : error occurs when there are some requirements that have multiple meanings, that is, their interpretation is not unique.