SRS FOR CHAT APPLICATIONS AND WEBSITE

By,

1. SUSHMITHA.M - 2115a139
2. SUBHIKHSHA.C.N- 2115a138
3. RESHMA.B - 2115a127

DEPARTMENT OF ECE
Panimalar Institute of Technology
No.391, Bangalore Trunk Road,
Varadharajapuram, Poonamallee,
Chennai-600 123

1.0) Introduction:

- There are numerous products available that allow for real time “chatting” over the Internet.

- The purpose of this project is to implement a Java based chat application that will allow users with an internet connection to engage in private and public conversations.

- The development of this project centered on the development of a message protocol that would allow the application to properly log in users, send messages, and perform system maintenance.

1.1) Problem Introduction:

- This project is 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.
- This 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.
- The 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 together through LAN.

1.2) Innovative Ideas Of Project:

- GUI: Easy to use GUI (Graphical User Interface), hence any user with minimal knowledge of operating a system can use the software.

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

- Unlimited clients: „n‟ number of users can be connected without any performance degradation of the server.

1.3) 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.

1.4) Scope of the project:

- Broadcasting Chat Server Application is going to be a text communication software, it will be able to communicate between two computers using point to point communication.

- The limitation of Live Chat is it does not support audio conversations. To overcome this limitation we are concurrently working on developing better technologies.

- Companies would like to have a communication software wherein they can communicate instantly within their organization.

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

1.5) Related Previous Work:

- When the existing system was studied, it was found having some problems, existing system was very time consuming and was not very efficient.

- The drawback of the existing system has resulted in to the development of new system, which is very user friendly and effective. Existing system was also very low in performance.

- There is no private chatting option had been available.

- Client can connect with the server with the only IP Address.

2.1) Project Perspective:

- The system to be developed here is an Chat facility. It is a centralized system. It is Client-Server system with centralized database server. All local clients are connected to the centralized server via LAN.

- There is a two 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.

- **No need of Internet connection:** Existing system requires Internet connection; whereas in the proposed system only Intranet connection i.e. only a LAN connection is required. This system is useful for those who can not afford to have an Internet connection. For example: schools, colleges, small companies, etc.

- **Conference possible on LAN:** Usually on LANs connections conferencing is not possible. The proposed system allows the LAN users to create and participate in conference. This makes communications possible among number of LAN users simultaneously.

2.1.1) Interface:

- This application interacts with the user through G.U.I. The interface is simple , easy to handle and self-explanatory.

- Once opened, user will easily come into the flow with the application and easily uses all interfaces properly.

- However the basic interface available in our application is
  - Title panel
  - Content panel
  - Admin panel.

2.1.2) Hardware Interface:

Minimum requirements will be as follows:

- 128 MB RAM required.
- Processor with speed of 500 MHz.
- Internet or LAN connection.
- MOUSE: 2 or 3 button mouse
- KEYBOARD: 101 key Keyboard

2.1.3) Software Interface:

- Notepad++ is a text editor and source code editor and provides an environment for developing HTML, jsp, JavaScript many other editing purposes.
- Coding done in java so required JDK 1.4 and above for run java programs.
- Operating system (such as window 7, 8, xp, Linux etc).

2.1.4) Constraints:

- The application does not by any means open the web browser. If user wishes to open the web browser he must open it externally.
- The system need to be permanent connected with internet.

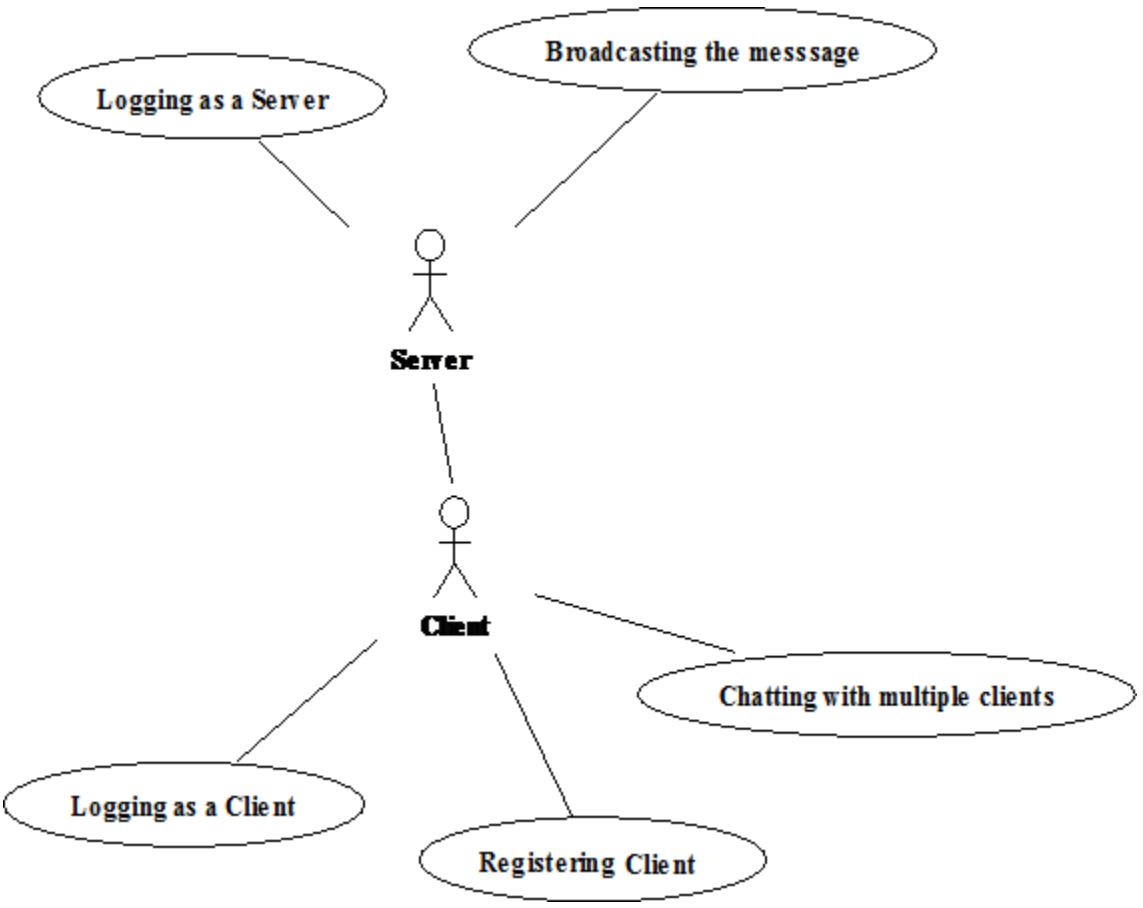
2.1.5) Product Functions:

- There is a two 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.

2.1.6) Assumption and dependencies:

- There should be LAN or internet connection.
- Clients should know each other.
- There can be multiple clients.

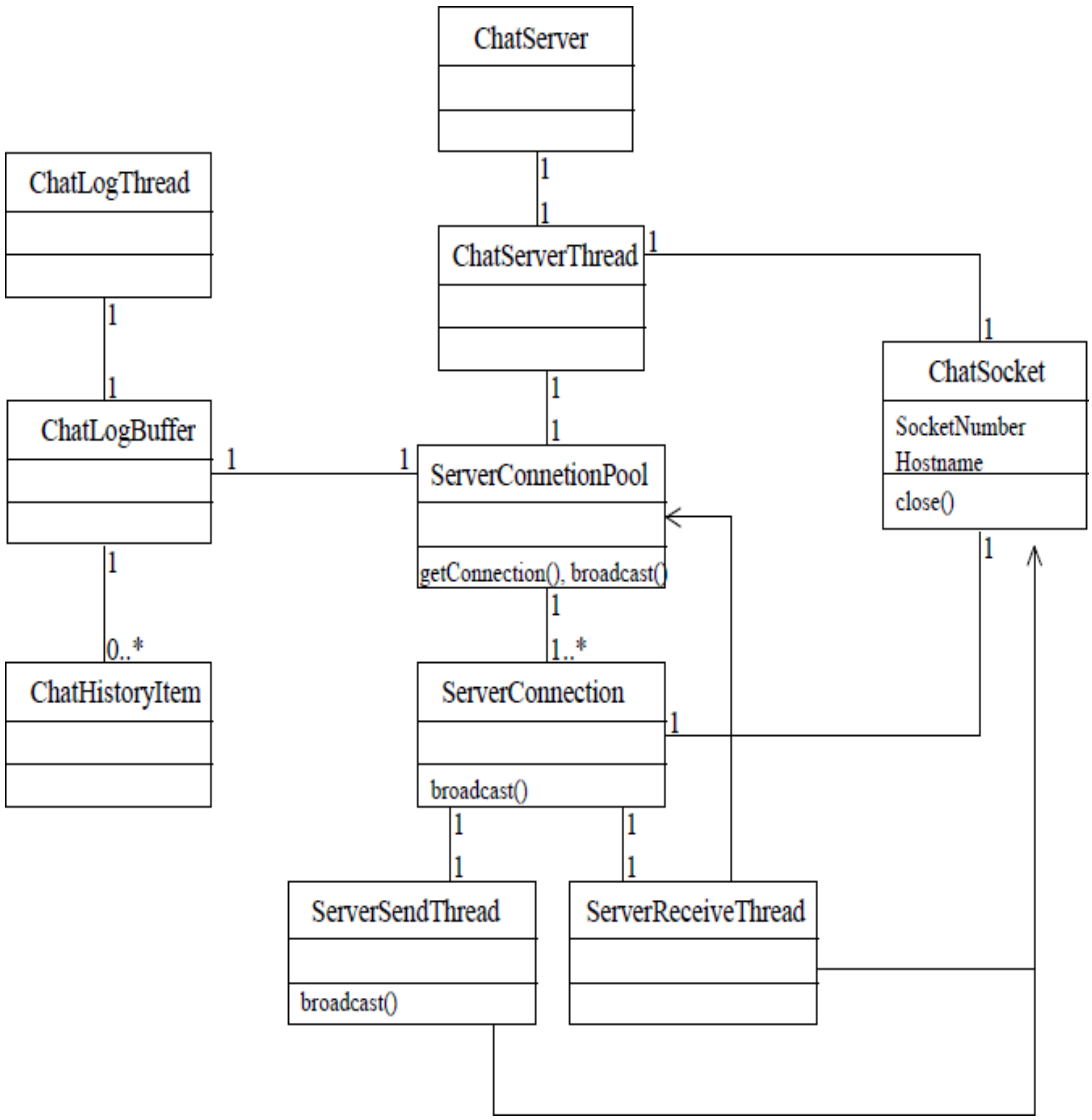
2.1.7) Use Case Diagram:



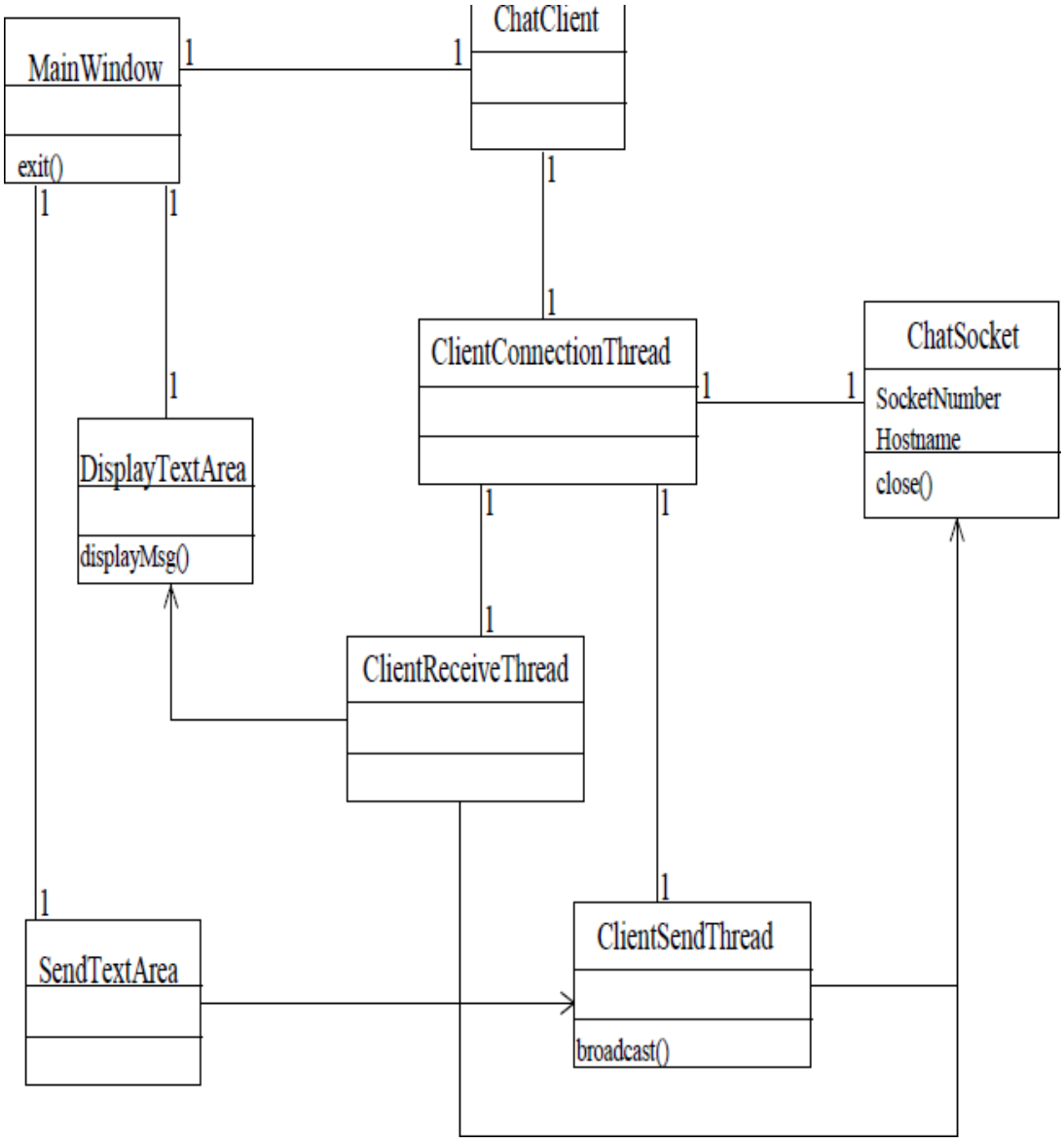
This use case model how client interacts with the server and start getting communicating with other clients.

2.1.8)

- **Class Diagram Of Server:**

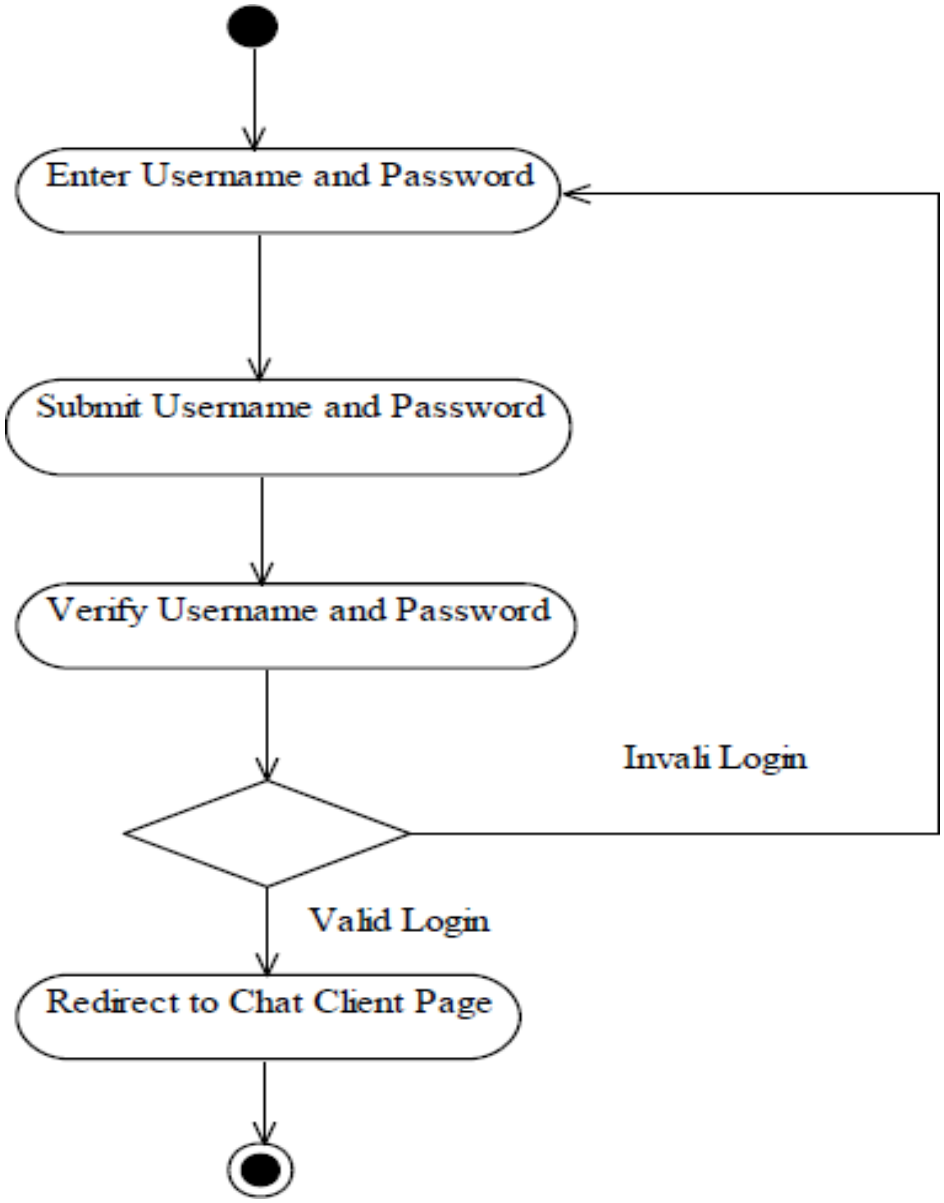


- **Class Diagram for Client:**

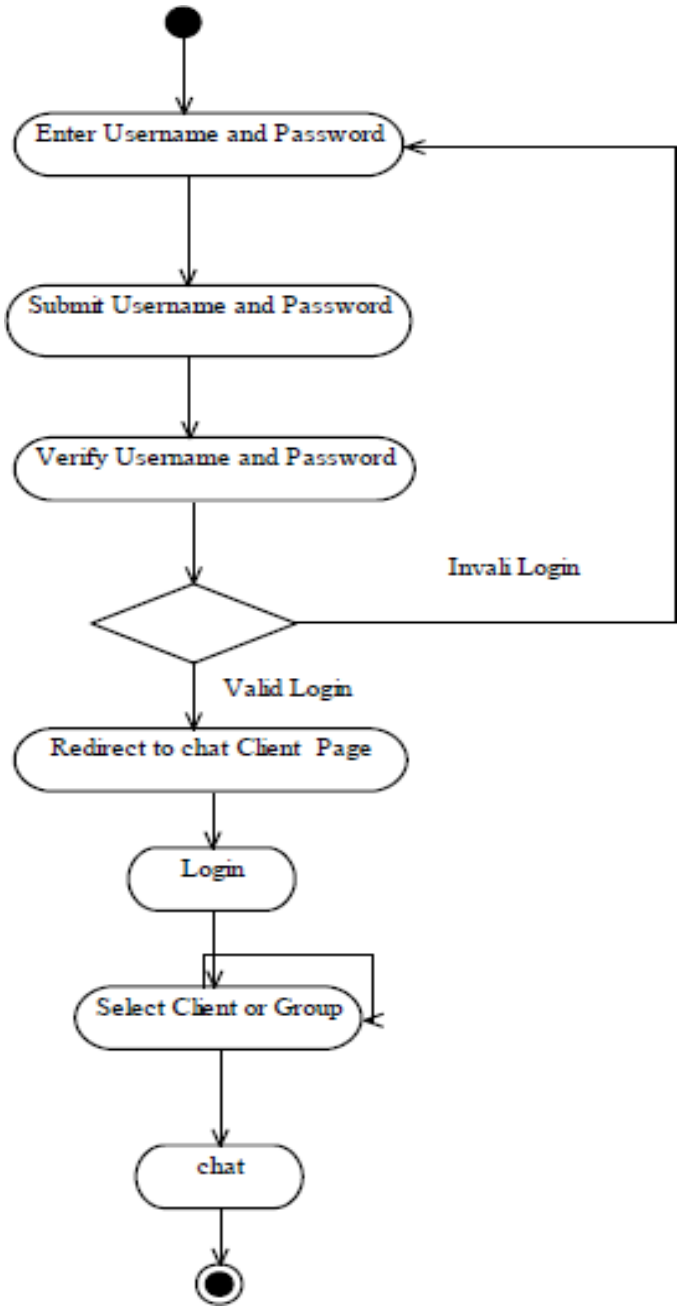


2.1.9)

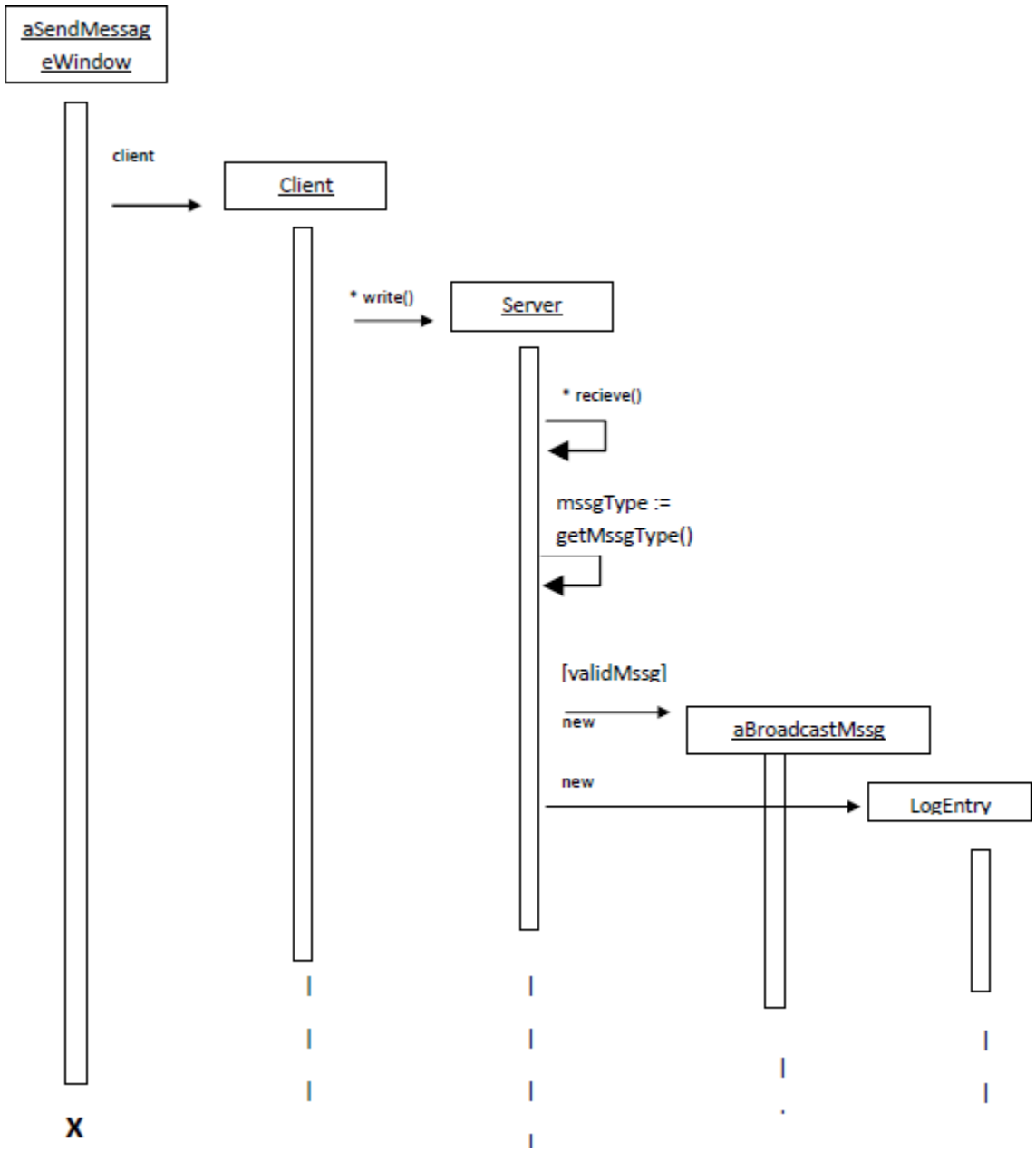
- **Activity Diagram For Login:**



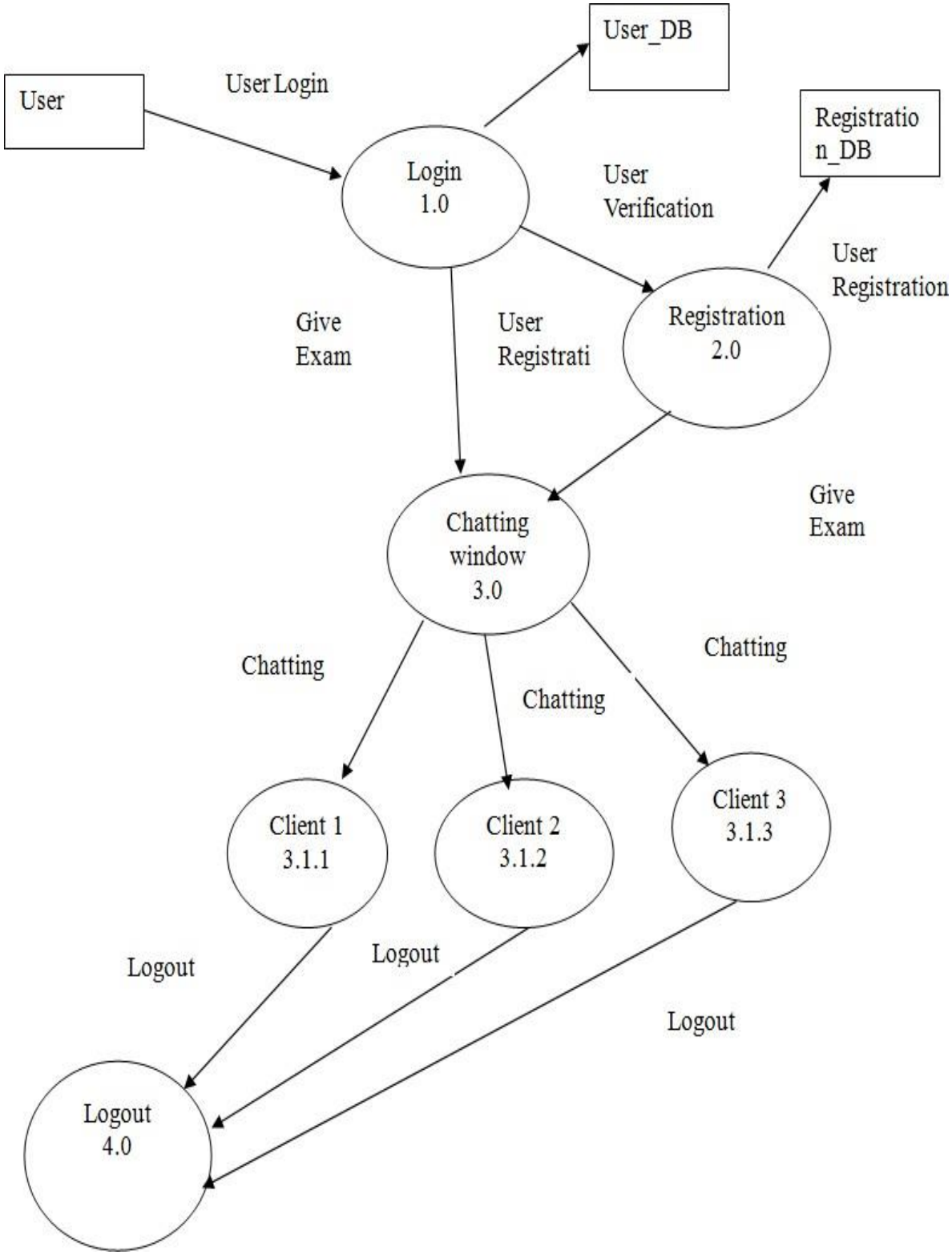
- **Activity Diagram For Chatting:**



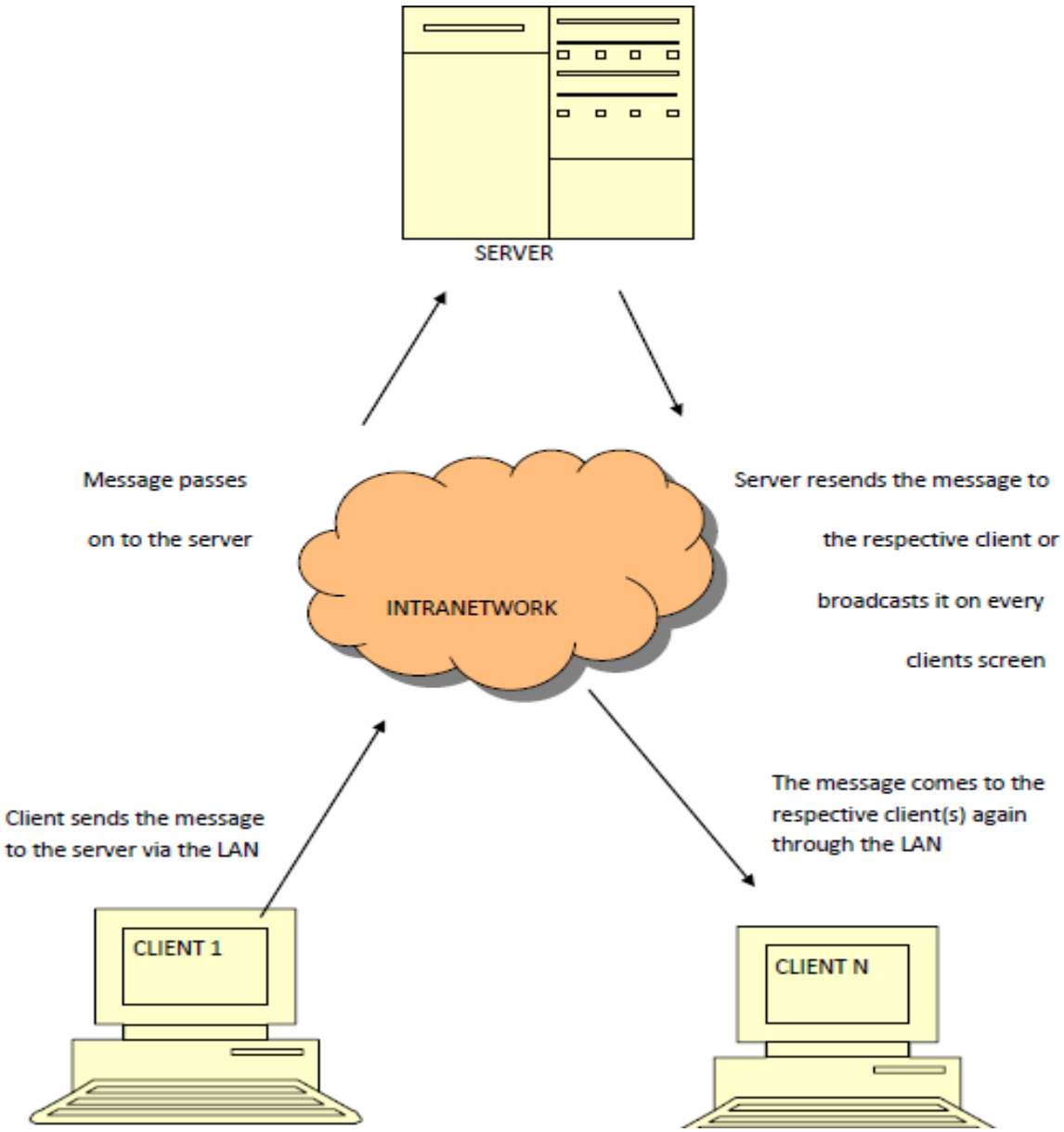
2.11) Sequence Diagram:



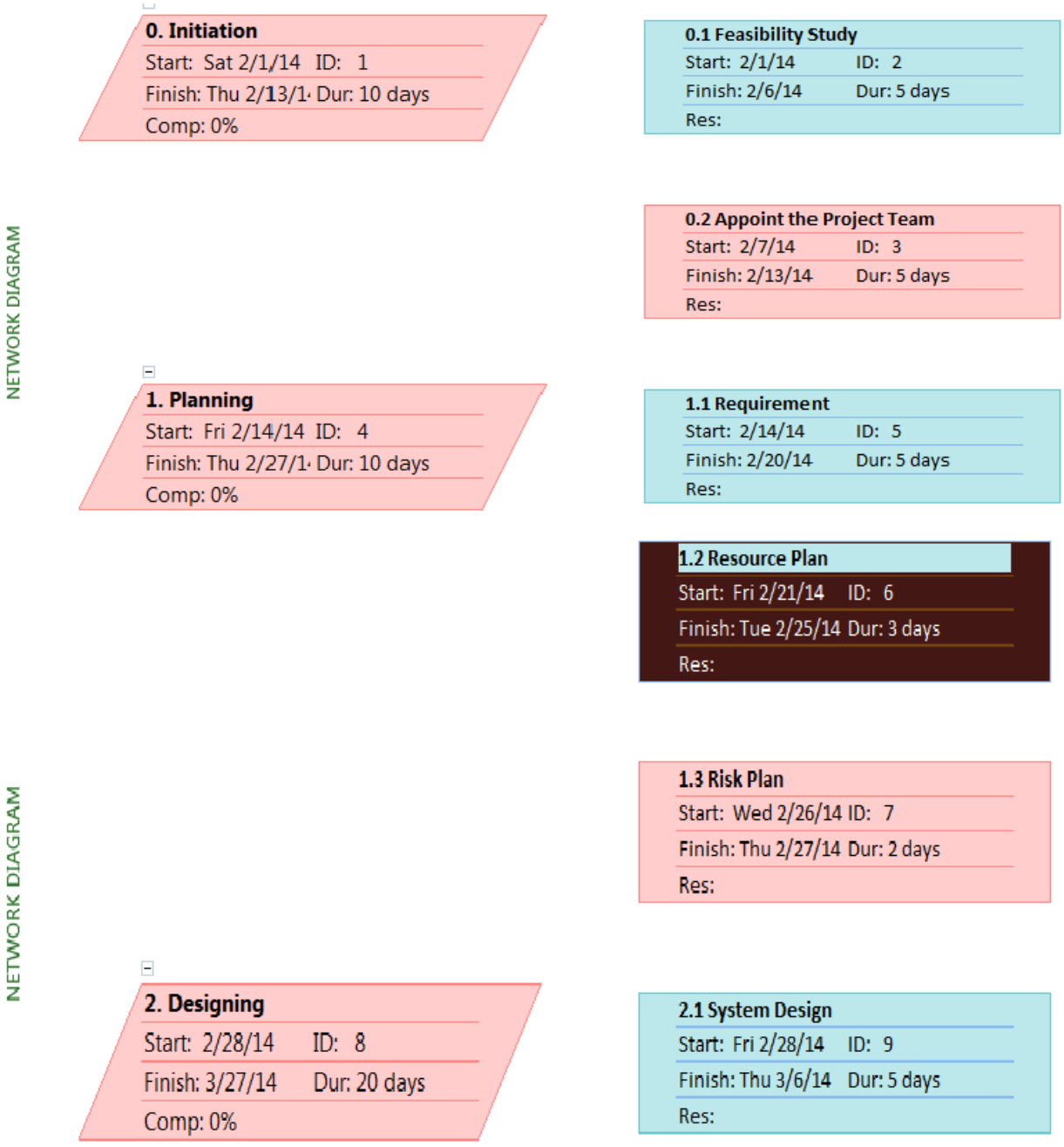
2.9) Data Flow Diagram:

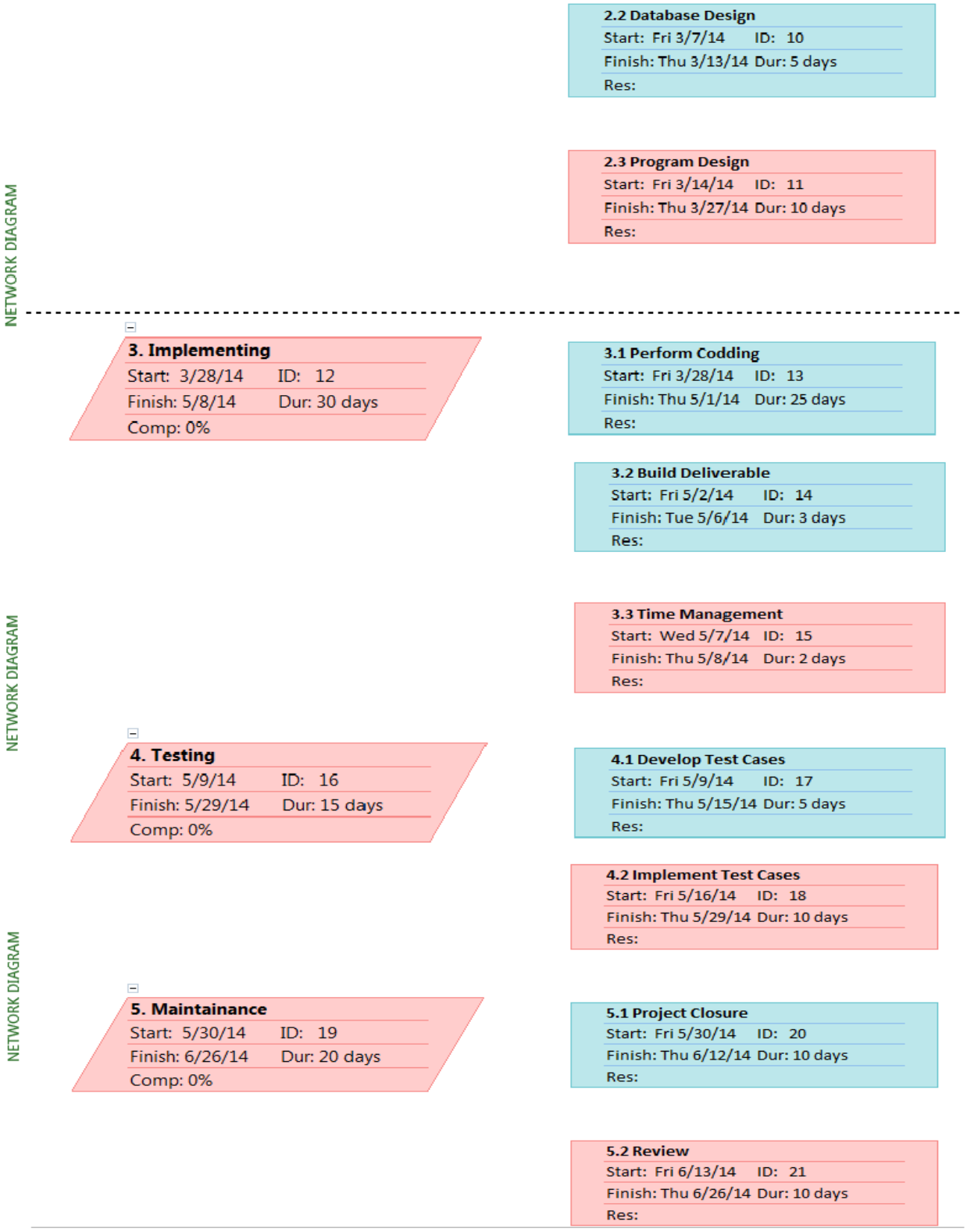


2.10) E-R Diagram:



3.1) Network Diagram:




NETWORK DIAGRAM
2.2 Database Design
Start: Fri 3/7/14 ID: 10
Finish: Thu 3/13/14 Dur: 5 days
Res:
2.3 Program Design
Start: Fri 3/14/14 ID: 11
Finish: Thu 3/27/14 Dur: 10 days
Res:
3. Implementing
Start: 3/28/14 ID: 12
Finish: 5/8/14 Dur: 30 days
Comp: 0%
3.1 Perform Codding
Start: Fri 3/28/14 ID: 13
Finish: Thu 5/1/14 Dur: 25 days
Res:
3.2 Build Deliverable
Start: Fri 5/2/14 ID: 14
Finish: Tue 5/6/14 Dur: 3 days
Res:
NETWORK DIAGRAM
3.3 Time Management
Start: Wed 5/7/14 ID: 15
Finish: Thu 5/8/14 Dur: 2 days
Res:
4. Testing
Start: 5/9/14 ID: 16
Finish: 5/29/14 Dur: 15 days
Comp: 0%
4.1 Develop Test Cases
Start: Fri 5/9/14 ID: 17
Finish: Thu 5/15/14 Dur: 5 days
Res:
4.2 Implement Test Cases
Start: Fri 5/16/14 ID: 18
Finish: Thu 5/29/14 Dur: 10 days
Res:
NETWORK DIAGRAM
5. Maintainance
Start: 5/30/14 ID: 19
Finish: 6/26/14 Dur: 20 days
Comp: 0%
5.1 Project Closure
Start: Fri 5/30/14 ID: 20
Finish: Thu 6/12/14 Dur: 10 days
Res:
5.2 Review
Start: Fri 6/13/14 ID: 21
Finish: Thu 6/26/14 Dur: 10 days
Res:

3.2) Gantt Chart: