

College of Engineering Guindy, Anna University

Department of Computer Science and Engineering

CS6110 Object Oriented Analysis and Design

Project Documentation

Social Networking Site

Srivatsav R 2019103066

Sachin Raghul T 2019103573

Sanjeev K M 2019103576

TABLE OF CONTENTS

S.NO.	TITLE	PAGE NO
1.	Introduction and Abstract	3
2.	Use Case Diagram	4
3.	Class Diagram	6
4.	Domain Model Diagram	8
5.	Refined Class Diagram	9
6.	Sequence Diagram	11
7.	Collaboration Diagram	13
8.	State Chart Diagram	15
9.	Activity Model	17
10.	Deployment Diagram	19
11.	Component Diagram	20
12.	Code Generation	22
13.	Conclusion	44

INTRODUCTION:

A social networking service or SNS is an online platform which people use to build social networks or social relationships with other people who share similar personal or career content, interests, activities, backgrounds or real-life connections. Also, it allows users to share ideas, digital photos and videos, posts, and to inform others about online or real-world activities and events with people within their social network.

ABSTRACT:

A social network is a website that allows people with similar interests to come together and share information, photos and videos. It shows the flows between the activity of Videos, Users, Social Network, Photos, Posts. The main activity involved in Social Networking Site are as follows:

- Videos activity
- Users activity
- Social Network Activity
- Photos Activity
- Posts Activity

Features of Social Networking Site

- Admin User can search Videos, view description of a selected Videos, add Videos, update Videos, and delete Videos.
- It shows the activity flow of editing, adding, and updating of Users.
- Users will be able to search and generate reports of Social Network, Posts, Photos.
- All objects such as (Videos, Users, Posts) are interlinked.
- It shows the full description and flow of Videos, Photos, Posts, Social Network, Users.

USE CASE DIAGRAM:

A UML use case diagram is the primary form of system/software requirements for a new software program underdeveloped. Use cases specify the expected behaviour (what), and not the exact method of making it happen (how). A key concept of use case modelling is that it helps us design a system from the end user's perspective. It is an effective technique for communicating system behaviour in the user's terms by specifying all externally visible system behaviour.

ACTORS AND THEIR USE CASES:

USER :-

- Login
- Chat
 - Group chat
 - Private chat
- Calls
 - Audio call
 - Video call
- Posts and Story
- Followers and Following
- View Profile

BUISNESS USER :-

- General user facilities
- Promotion posts

ADMIN :-

- Manage Users and Business users
- Manage Posts and Story
- Co-ordinate chats and calls
- Controls payment and transactions

PAYTM :-

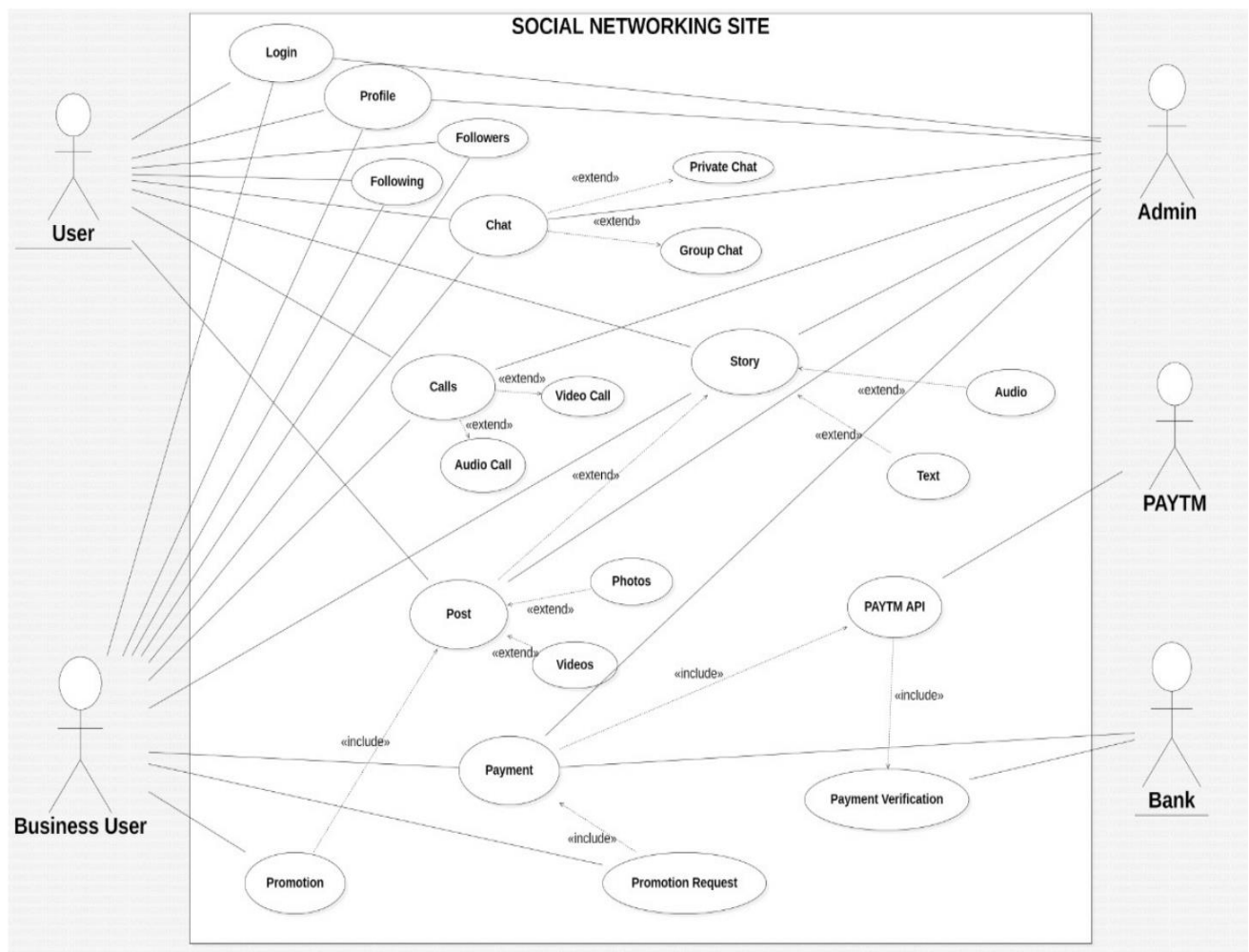
- Gateway for payment
- Provides PAYTM API

BANK :-

- Controls payment requests
- Verifies payments received

ACTORS	RESPONSIBILITY
ADMIN	Control and co-ordinate the overall activity of the network.
USER	Uses the general facilities of the website
BUSINESS USER	In addition to the general facilities, also the user is provided business facilities for business purpose
PAYTM	Gateway of transactions involved between user and the bank
BANK	Verifies the payments received

DIAGRAM:

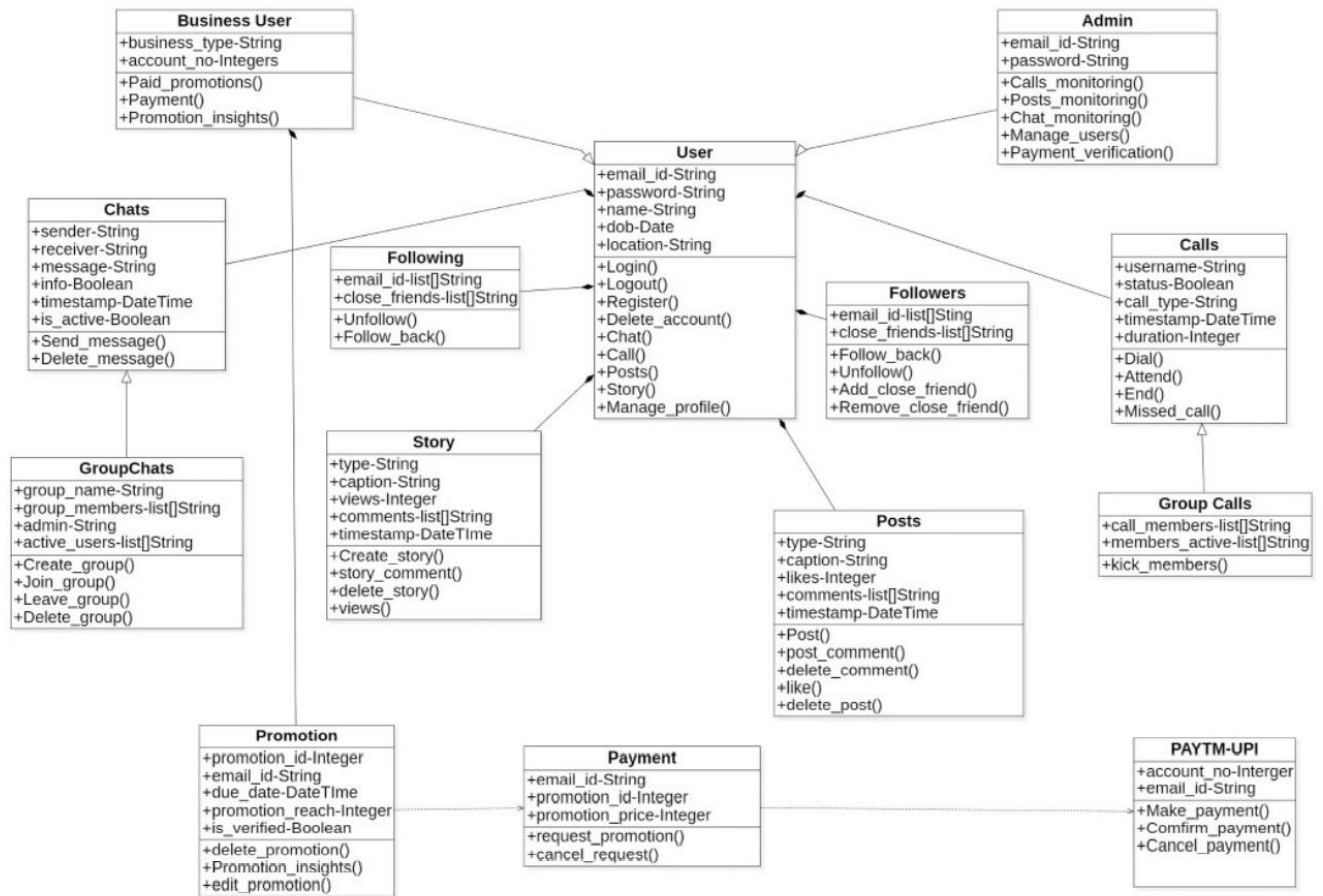


CLASS DIAGRAM:

The class diagram depicts a static view of an application. It represents the types of objects residing in the system and the relationships between them. A class consists of its objects, and also it may inherit from other classes. A class diagram is used to visualize, describe, document various different aspects of the system, and also construct executable software code.

CLASS	PURPOSE
ADMIN	Contains admin credentials and monitors all activities of user in general
USER	Contains user credentials and can perform all user-friendly activities
BUSINESS USER	Contains business user credentials and can promote their business products in addition to the general user activities
CHATS	Consists of information and meta data of every chat done between users
POSTS	Details of any type of posts posted by the users
STORY	Contains the story info and the log of the users who visited the story
CALLS	End to end encrypted call between user and maintains the call history
FOLLOWERS	List of followers and also the close friends of the user
FOLLOWING	List of other users the user follows
PROMOTION	Details of the product which is promoted by the business user with due-date
PAYMENT	Contains the details of the payment and transactions between user and the bank

DIAGRAM:

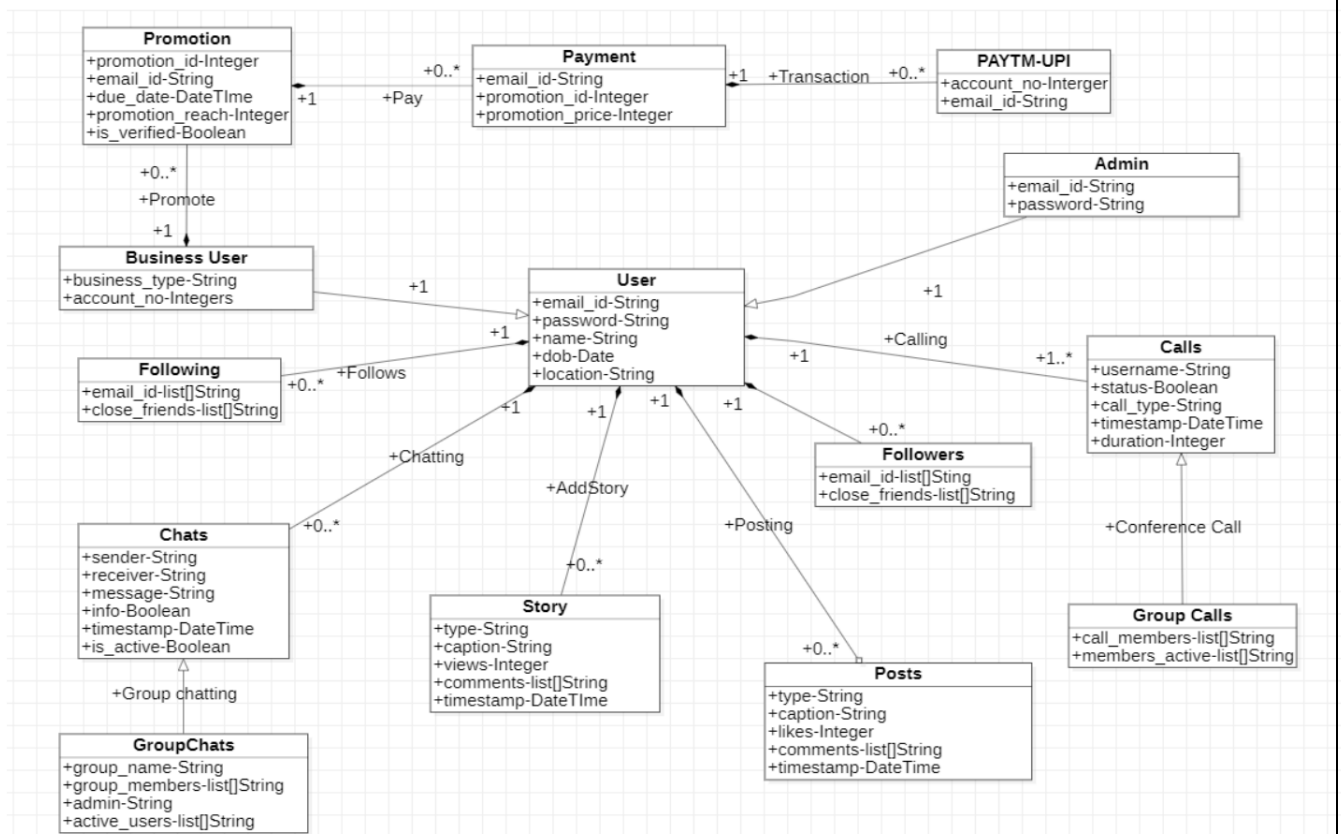


DOMAIN CLASS MODEL:

Domain modelling is a technique used to understand the project problem description and to translate the requirements of that project into software components of a solution. The software components are commonly implemented in an object-oriented programming language.

A domain model contains conceptual classes, associations between conceptual classes, and attributes of a conceptual class. "Informally, a conceptual class is an idea, thing, or object".

DIAGRAM:



REFINED CLASS DIAGRAM:

Refining the Class Diagram From Activity Diagram and Sequence diagram, add the classes needed. Identify more attributes (instance variables) and methods

ADMIN :-

- Contains admin credentials and monitors all activities of user in general

USER :-

- Contains user credentials and can perform all user-friendly activities

BUSINESS USER :-

- Contains business user credentials and can promote their business products in addition to the general user activities

CHATS :-

- Consists of information and meta data of every chat done between users
- Also, contains the status of every message sent by the user

POSTS :-

- Details of any type of posts posted by the user
- Consists of the comments commented by other users to the post

STORY :-

- Contains the story info and the log of the users who visited the story

CALLS :-

- End to end encrypted call between user and maintains the call history
- In case of group calls, it manages the participants attending the conference

FOLLOWERS :-

- List of followers and also the close friends of the user

FOLLOWING :-

- List of other users the user follows

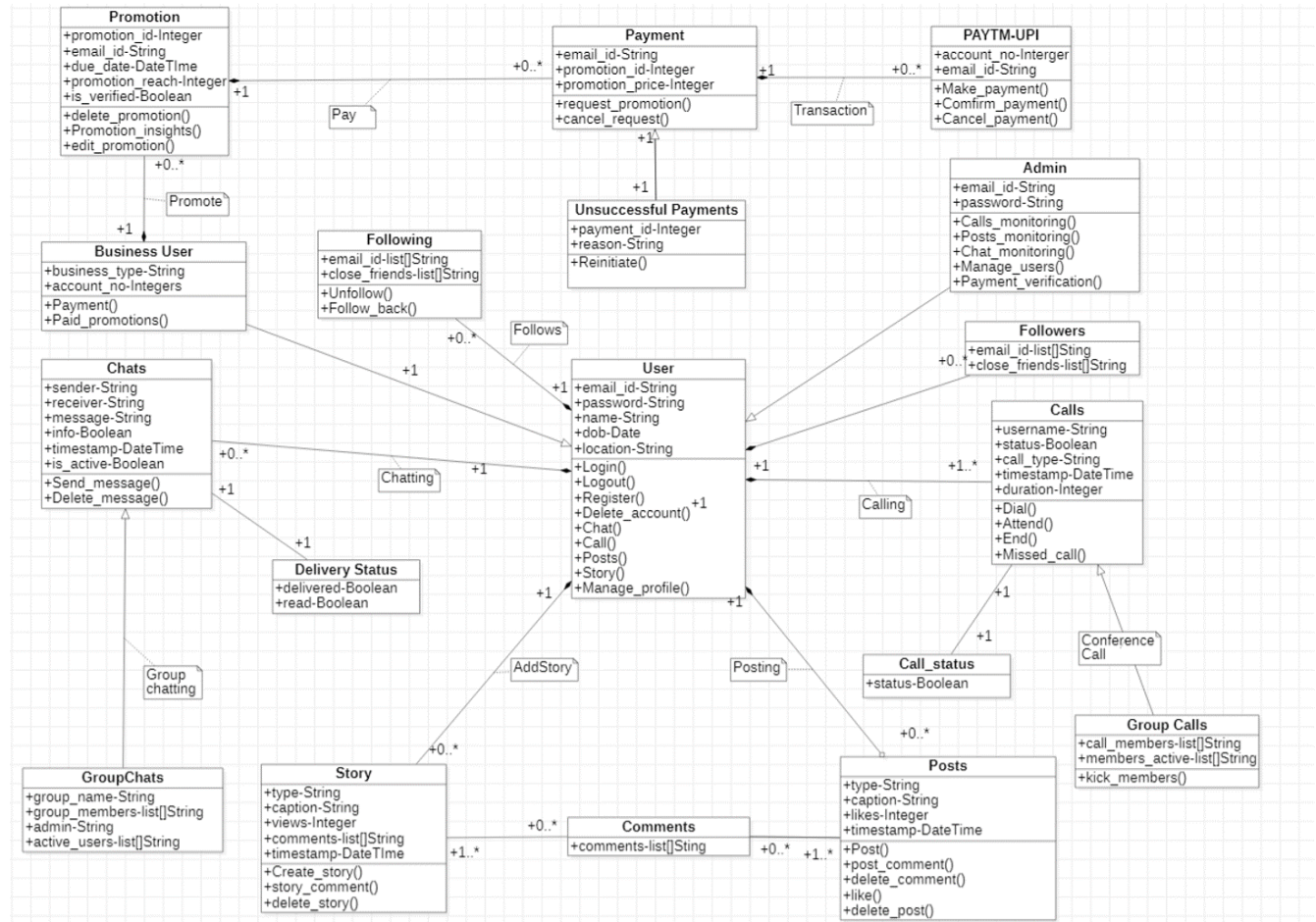
PROMOTION :-

- Details of the product which is promoted by the business user with due-date

PAYMENT :-

- Contains the details of the payment and transactions between user and the bank
- Maintains a log of unsuccessful payments

DIAGRAM:



SEQUENCE DIAGRAM:

A sequence diagram simply depicts interaction between objects in a sequential order i.e., the order in which these interactions take place. We can also use the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system function.

LIFELINES :-

1. Client:

The user of the chatting system. Can utilise the chatting system and can perform multiple tasks.

2. Chatting System:

Manages all the tasks that the user needs to perform Acts as an interface between the tasks and the user.

3. Client Database:

Stores details of clients.

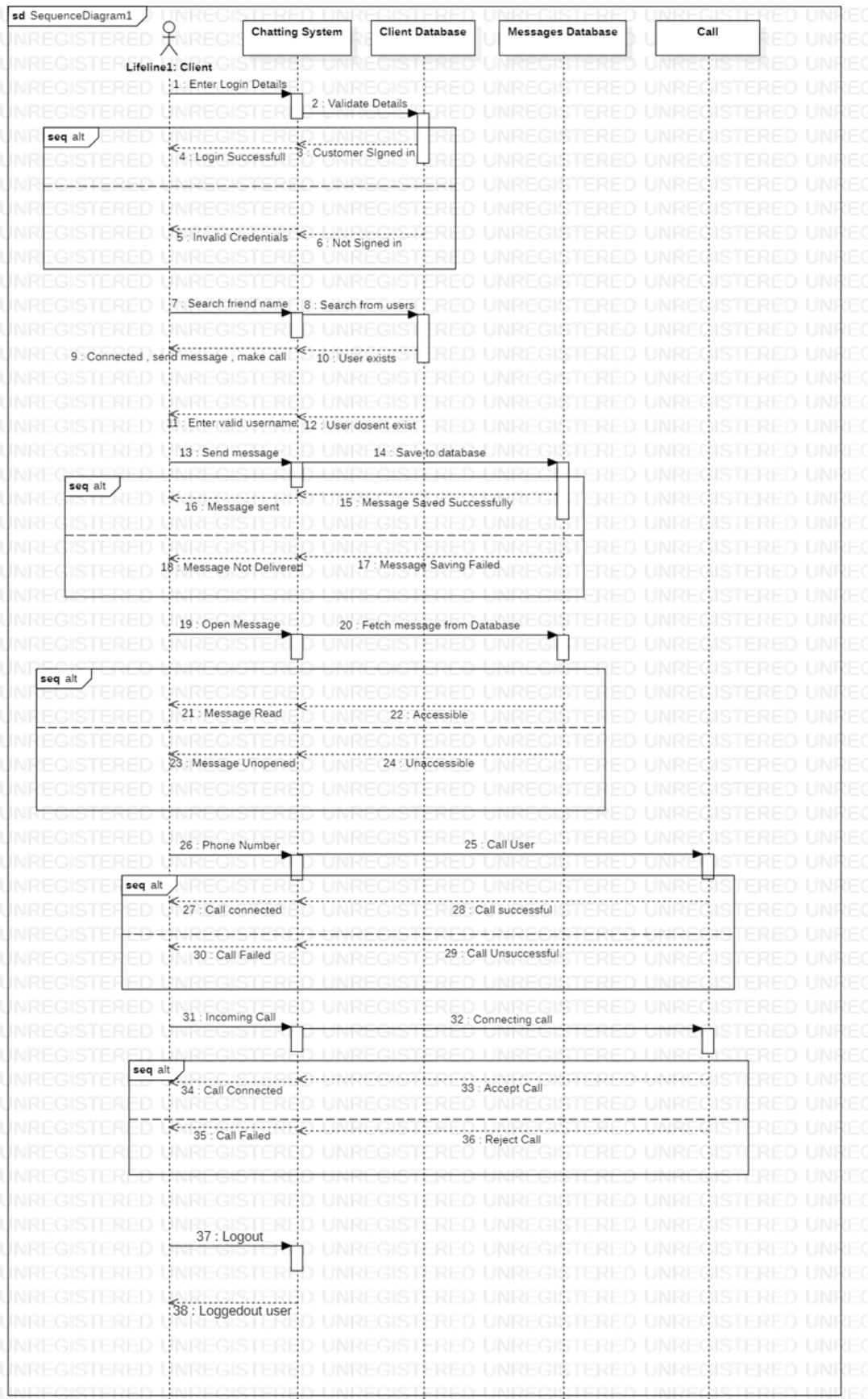
4. Message Database:

Stores the messages sent to different users and received from different users.

5. Call:

Process of calling users

DIAGRAM:



COMMUNICATION/ COLLABRATION DIAGRAM:

Communication diagrams, like the sequence diagrams are a kind of interaction diagram, shows how objects interact. A communication diagram is an extension of object diagram that shows the objects along with the messages that travel from one to another. In addition to the associations among objects, communication diagram shows the messages the objects send each other.

LIFELINES :-

1. Client:

Can perform chats and calls using the provided interface.

2. Chatting System:

Manages all chats, both private and group chat and contains read receipts of the chat.

3. Client Database:

Stores all information of both user and business user

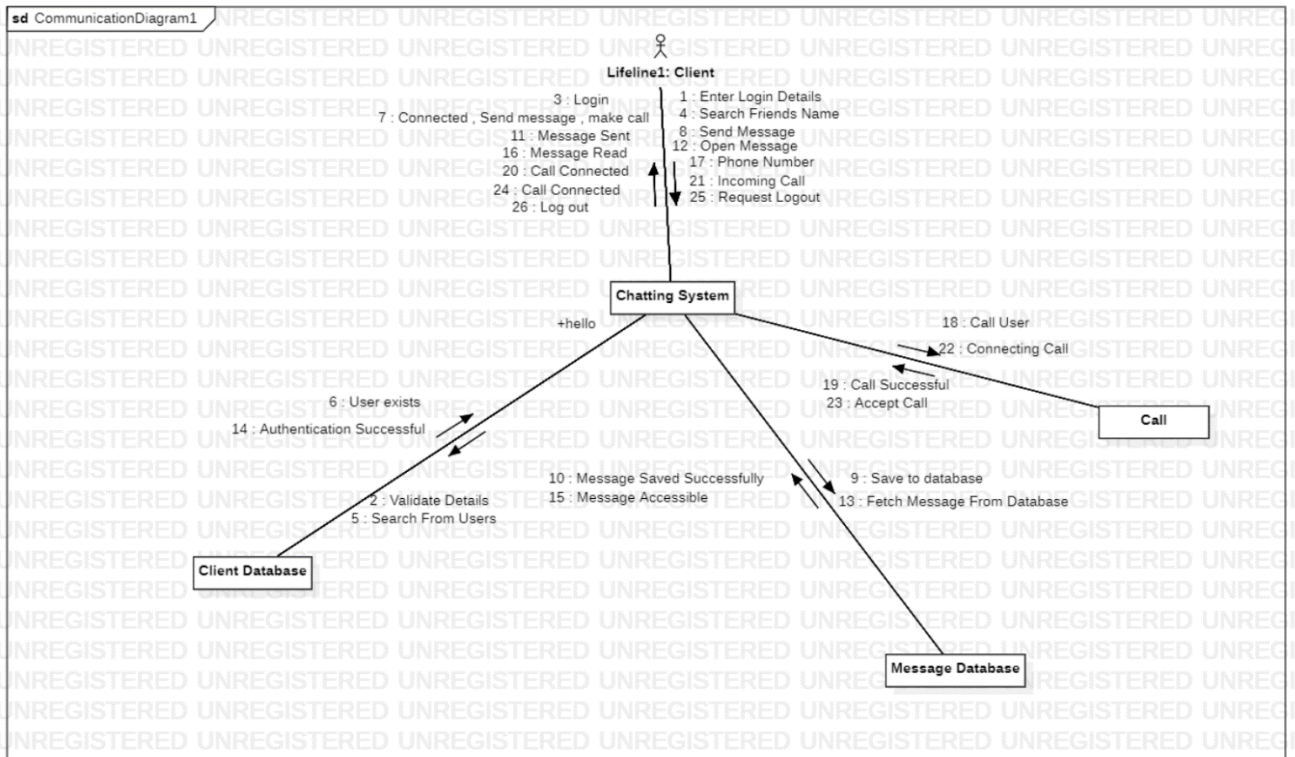
4. Message Database:

Stores all end-to-end encrypted information of messages sent to different users and received from different users.

5. Call:

The whole call process and details of the participants attending the conference.

DIAGRAM:



STATE DIAGRAM

A state diagram consists of states, transitions, events, and activities. You use state diagrams to illustrate the dynamic view of a system. They are especially important in modelling the behaviour of an interface, class, or collaboration. State diagrams emphasize the event-ordered behaviour of an object, which is especially useful in modelling reactive systems.

STATES :-

1. User credential validation:

User had to enter the user-id and the password.

2. Logged in :

After entering the user credentials the credentials are verified and user is allowed to get logged in.

3. Selection of other User:

In this state user searches for the other username with whom the user wishes to perform chat or call.

4. Ready to perform chat or call:

In this state user is connected with other user to send or receive messages and call.

5. Connected with friend, Place Call:

In this state after user got connected with the other user friend, the user can call or chat.

6. Message delivered:

User's message is delivered and the user can send next message.

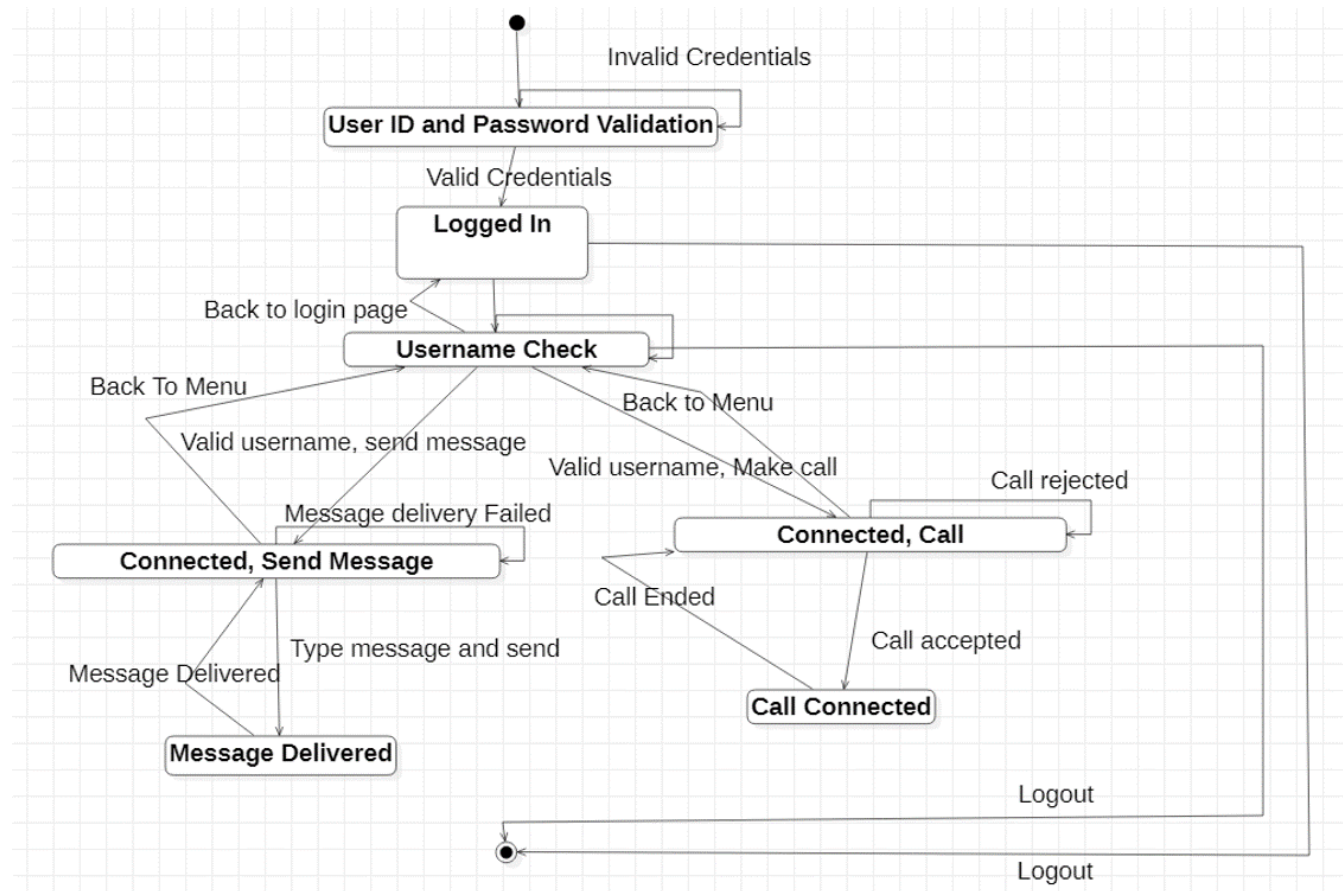
7. Call connected:

User's call is connected and so the user can speak with the other user friend.

8. Promotions :

Business user can advertise the products that they need.

DIAGRAM:



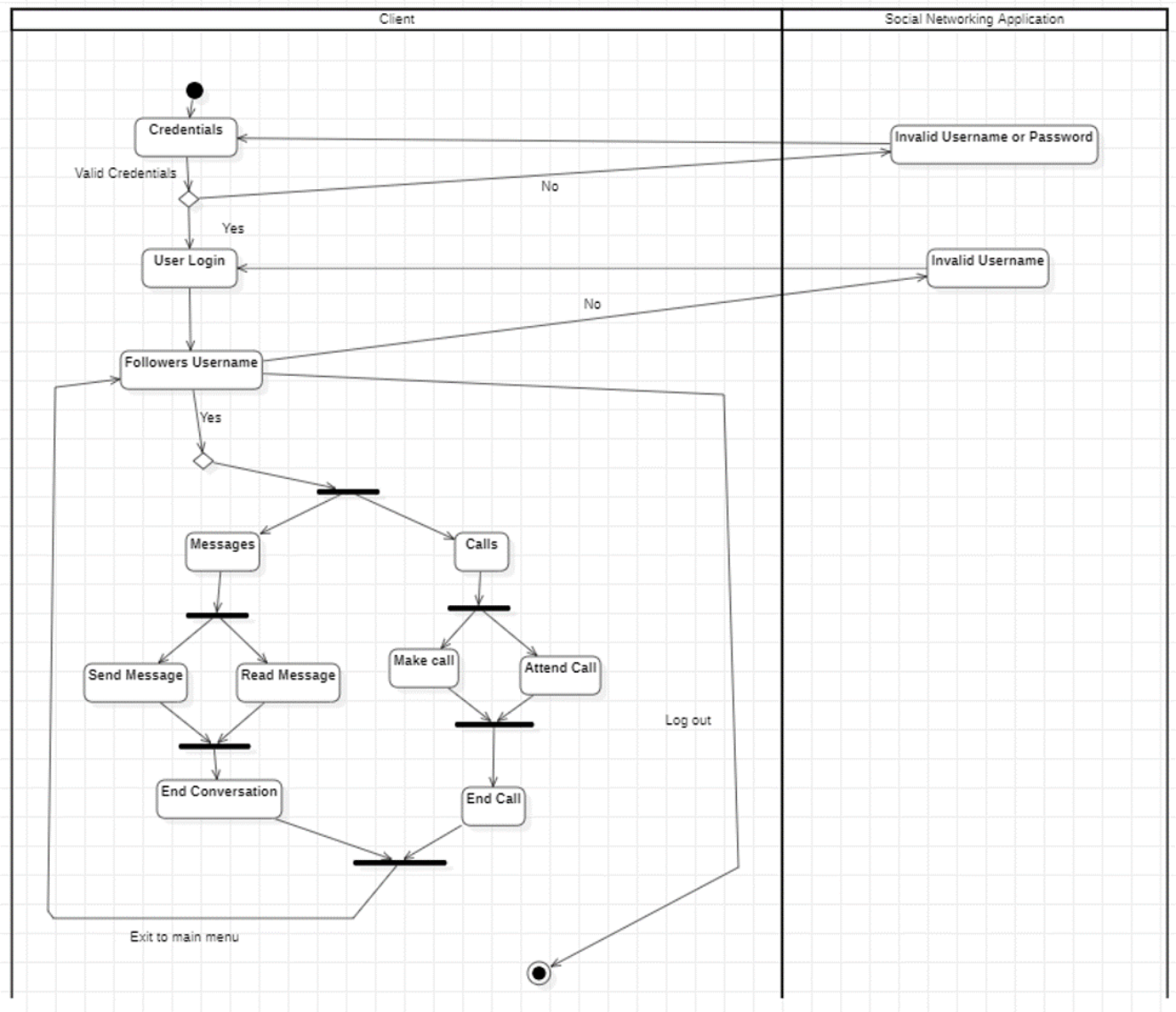
ACTIVITY DIAGRAM:

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent

FLOW :-

- User enters the login credentials.
- The credentials are validated by the chatting application.
- User gets logged in after validation.
- Searches for the other user to whom the user wants to perform some activity.
- User gets connected with the other user who was searched.
 - Chat
 - Call
 - Payment
- Conversation or conference is in process.
- End of the conversation/conference.
- At last, user can return to main menu else get logged out.

DIAGRAM:



DEPLOYMENT DIAGRAM:

A UML deployment diagram is a diagram that shows the configuration of run time processing nodes and the components that live on them. Deployment diagrams is a kind of structure diagram used in modelling the physical aspects of an object oriented system. They are often be used to model the static deployment view of a system (topology of the hardware).

SOCIAL SITE APPLICATION :-

- The application that manages all user activities and functions.

USER SERVICE :-

- Manages all user activities and credentials.

CHAT SERVER :-

- Holds information of end-to-end encrypted chats and logs.

POSTS SERVER :-

- Holds information of all types of posts and the comments commented by the other user under the particular post.

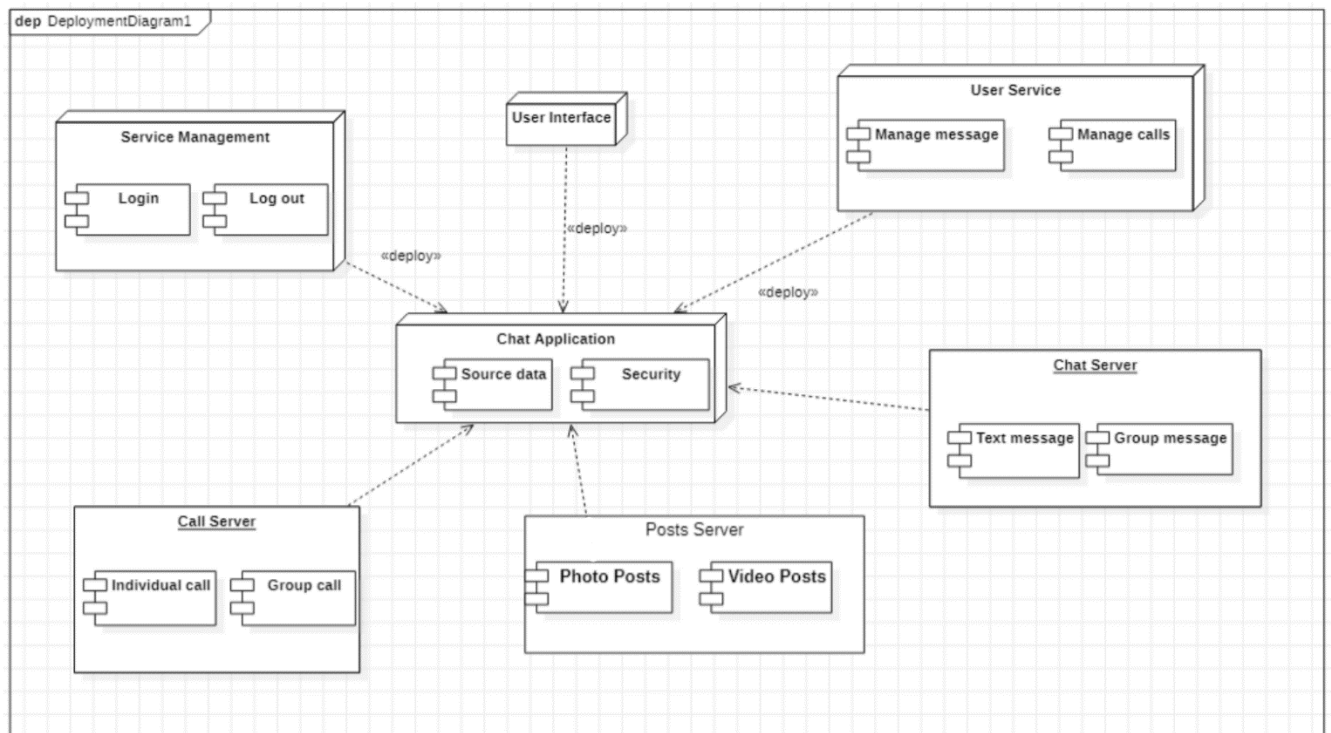
CALL SERVER :-

- Holds information of end-to-end encrypted call and list of participants of the conference.

SERVICE MANAGEMENT:-

- Manages user login and logout activity and other such general activities.

DIAGRAM:



COMPONENT DIAGRAM:

UML Component diagrams are used in modelling the physical aspects of object oriented systems that are used for visualizing, specifying, and documenting component-based systems and also for constructing executable systems through forward and reverse engineering. Component diagrams are essentially class diagrams that focus on a system's components that often used to model the static implementation view of a system.

Chatting Application :

Application that manages all functions.

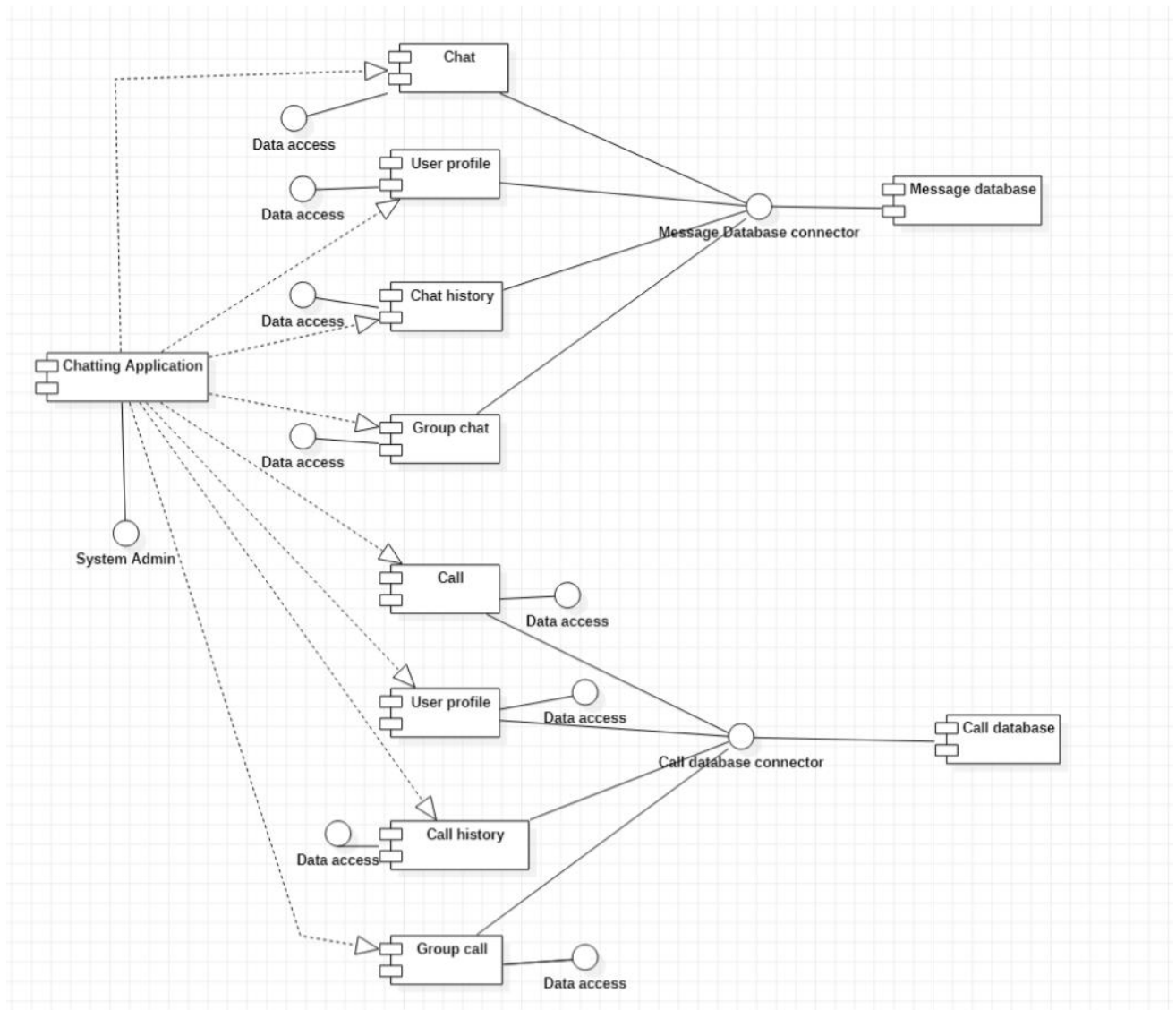
Message Database :

Stores all the messages and information about the messages.

Call Database :

Holds information about the calls and call history.

DIAGRAM:



CODE GENERATION:

ADMIN:

```
package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class Admin extends User {

    /**
     * Default constructor
     */
    public Admin() {
    }

    /**
     *
     */
    public void email_id-String;

    /**
     *
     */
    public void password-String;

    /**
     *
     */
    public void Calls_monitoring() {
        // TODO implement here
    }

    /**
     *
     */
    public void Posts_monitoring() {
        // TODO implement here
    }

    /**
     *
     */
    public void Chat_monitoring() {
        // TODO implement here
    }
}
```

```

/**
 *
 */
public void Manage_users() {
    // TODO implement here
}

/**
 *
 */
public void Payment_verification() {
    // TODO implement here
}
}

```

BUSINESS USER:

```

package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class Business User extends User {

    /**
     * Default constructor
     */
    public Business User() {
    }

    /**
     *
     */
    public void business_type-String;

    /**
     *
     */
    public void account_no-Integers;

    /**
     *
     */

```

```

    public Promotion Promote;

    /**
     *
     */
    public void Paid_promotions() {
        // TODO implement here
    }

    /**
     *
     */
    public void Payment() {
        // TODO implement here
    }

    /**
     *
     */
    public void Operation2() {
        // TODO implement here
    }

    /**
     *
     */
    public void Payment() {
        // TODO implement here
    }

    /**
     *
     */
    public void Paid_promotions() {
        // TODO implement here
    }
}

```


CALL STATUS:

```
package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class Call_status {

    /**
     * Default constructor
     */
    public Call_status() {
    }

    /**
     *
     */
    public void status-Boolean;
}
```

CALLS:

```
package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class Calls extends Group Calls {

    /**
     * Default constructor
     */
    public Calls() {
    }

    /**
     *
     */
    public void username-String;

    /**
     *
     */
}
```

```

    public void status-Boolean;

    /**
     *
     */
    public void call_type-String;

    /**
     *
     */
    public void timestamp-DateTime;

    /**
     *
     */
    public void duration-Integer;

    /**
     *
     */
    public void Dial() {
        // TODO implement here
    }

    /**
     *
     */
    public void Attend() {
        // TODO implement here
    }

    /**
     *
     */
    public void End() {
        // TODO implement here
    }

    /**
     *
     */
    public void Missed_call() {
        // TODO implement here
    }
}

```

CHATS:

```
package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class Chats {

    /**
     * Default constructor
     */
    public Chats() {
    }

    /**
     *
     */
    public void sender-String;

    /**
     *
     */
    public void receiver-String;

    /**
     *
     */
    public void message-String;

    /**
     *
     */
    public void info-Boolean;

    /**
     *
     */
    public void timestamp-DateTime;

    /**
     *
     */
    public void is_active-Boolean;
}
```

```

    public void Send_message() {
        // TODO implement here
    }

    /**
     *
     */
    public void Delete_message() {
        // TODO implement here
    }

    /**
     *
     */
    public void Send_message() {
        // TODO implement here
    }

    /**
     *
     */
    public void Delete_message() {
        // TODO implement here
    }
}

```

COMMENTS:

```

package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class Comments {

    /**
     * Default constructor
     */
    public Comments() {
    }

    /**
     *
     */
    public void comments-list[]String;

}

```

DELIVERY STATUS:

```
package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class Delivery Status {

    /**
     * Default constructor
     */
    public Delivery Status() {
    }

    /**
     *
     */
    public void delivered-Boolean;

    /**
     *
     */
    public void read-Boolean;

}
```

FOLLOWERS:

```
package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class Followers {

    /**
     * Default constructor
     */
    public Followers() {
    }

    /**
     *
     */
}
```

```

    public void email_id-list[]String;

    /**
     *
     */
    public void close_friends-list[]String;

    /**
     *
     */
    public User 1;

    /**
     *
     */
    public void Follow_back() {
        // TODO implement here
    }

    /**
     *
     */
    public void Unfollow() {
        // TODO implement here
    }

    /**
     *
     */
    public void Add_close_friend() {
        // TODO implement here
    }

    /**
     *
     */
    public void Remove_close_friend() {
        // TODO implement here
    }

}

```

FOLLOWING:

```

package SOCIAL NETWORKING SITE;
import java.util.*;

/**
 *
 */

```

```

public class Following {

    /**
     * Default constructor
     */
    public Following() {
    }

    /**
     *
     */
    public void email_id-list[]String;

    /**
     *
     */
    public void close_friends-list[]String;

    /**
     *
     */
    public User 1;

    /**
     *
     */
    public void Unfollow() {
        // TODO implement here
    }

    /**
     *
     */
    public void Follow_back() {
        // TODO implement here
    }

}

```

GROUP CALLS:

```

package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */

```

```

public class Group Calls extends Calls {

    /**
     * Default constructor
     */
    public Group Calls() {
    }

    /**
     *
     */
    public void call_members-list[]String;

    /**
     *
     */
    public void members_active-list[]String;

    /**
     *
     */
    public void kick_members() {
        // TODO implement here
    }

}

```

GROUP CHATS:

```

package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class GroupChats extends Chats {

    /**
     * Default constructor
     */
    public GroupChats() {
    }

    /**
     *
     */
    public void group_name-String;

    /**

```



```

    *
    */
    public void group_members-list[]String;

    /**
     *
     */
    public void admin-String;

    /**
     *
     */
    public void active_users-list[]String;

    /**
     *
     */
    public void Create_group() {
        // TODO implement here
    }

    /**
     *
     */
    public void Join_group() {
        // TODO implement here
    }

    /**
     *
     */
    public void Leave_group() {
        // TODO implement here
    }

    /**
     *
     */
    public void Delete_group() {
        // TODO implement here
    }
}

```

PAYMENT:

```

package SOCIAL NETWORKING SITE;

import java.util.*;

```

```

/**
 *
 */
public class Payment extends Unsuccessful Payments {

    /**
     * Default constructor
     */
    public Payment() {
    }
    /**
     *
     */
    public void email_id-String;

    /**
     *
     */
    public void promotion_id-Integer;

    /**
     *
     */
    public void promotion_price-Integer;

    /**
     *
     */
    public PAYTM-UPI Transaction;

    /**
     *
     */
    public Promotion 1;

    /**
     *
     */
    public void request_promotion() {
        // TODO implement here
    }

    /**
     *
     */
    public void cancel_request() {
        // TODO implement here
    }
}

```

PAYTM-UPI:

```
package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class PAYTM-UPI {

    /**
     * Default constructor
     */
    public PAYTM-UPI() {
    }

    /**
     *
     */
    public void account_no-Interger;

    /**
     *
     */
    public void email_id-String;

    /**
     *
     */
    public void Make_payment() {
        // TODO implement here
    }

    /**
     *
     */
    public void Comfirm_payment() {
        // TODO implement here
    }

    /**
     *
     */
    public void Cancel_payment() {
        // TODO implement here
    }
}
```

POSTS:

```
package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class Posts {

    /**
     * Default constructor
     */
    public Posts() {
    }

    /**
     *
     */
    public void type-String;

    /**
     *
     */
    public void caption-String;

    /**
     *
     */
    public void likes-Integer;

    /**
     *
     */
    public void timestamp-DateTime;

    /**
     *
     */
    public User 1;

    /**
     *
     */
    public void Post() {
        // TODO implement here
    }

    /**
```

```

        *
        */
    public void post_comment() {
        // TODO implement here
    }

    /**
     *
     */
    public void delete_comment() {
        // TODO implement here
    }

    /**
     *
     */
    public void like() {
        // TODO implement here
    }

    /**
     *
     */
    public void delete_post() {
        // TODO implement here
    }
}

```

PROMOTION:

```

package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class Promotion {

    /**
     * Default constructor
     */
    public Promotion() {
    }

    /**
     *
     */
    public void promotion_id-Integer;
}

```

```

/**
 *
 */
public void email_id-String;

/**
 *
 */
public void due_date-DateTime;

/**
 *
 */
public void promotion_reach-Integer;

/**
 *
 */
public void is_verified-Boolean;

/**
 *
 */
public Payment Pay;

/**
 *
 */
public void delete_promotion() {
    // TODO implement here
}

/**
 *
 */
public void Promotion_insights() {
    // TODO implement here
}

/**
 *
 */
public void edit_promotion() {
    // TODO implement here
}
}

```

STORY:

```
package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class Story {

    /**
     * Default constructor
     */
    public Story() {
    }

    /**
     *
     */
    public void type-String;

    /**
     *
     */
    public void caption-String;

    /**
     *
     */
    public void views-Integer;

    /**
     *
     */
    public void comments-list[]String;

    /**
     *
     */
    public void timestamp-DateTime;

    /**
     *
     */
    public User 1;

    /**
     *
     */
}
```

```

    public void Create_story() {
        // TODO implement here
    }

    /**
     *
     */
    public void story_comment() {
        // TODO implement here
    }

    /**
     *
     */
    public void delete_story() {
        // TODO implement here
    }

    /**
     *
     */
    public void views() {
        // TODO implement here
    }
}

```

UNSUCCESSFUL PAYMENT:

```

package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class Unsuccessful Payments extends Payment {

    /**
     * Default constructor
     */
    public Unsuccessful Payments() {
    }

    /**
     *
     */
    public void payment_id-Integer;

    /**

```



```

        *
        */
    public void reason-String;

    /**
     *
     */
    public void Reinitiate() {
        // TODO implement here
    }
}

```

USER:

```

package SOCIAL NETWORKING SITE;

import java.util.*;

/**
 *
 */
public class User {

    /**
     * Default constructor
     */
    public User() {
    }

    /**
     *
     */
    public void email_id-String;

    /**
     *
     */
    public void password-String;

    /**
     *
     */
    public void name-String;

    /**
     *
     */
    public void dob-Date;
}

```

```

/**
 *
 */
public void location-String;

/**
 *
 */
public Chats Chatting;

/**
 *
 */
public Calls Calling;

/**
 *
 */
public Posts 0..*;

/**
 *
 */
public Following Follows;

/**
 *
 */
public void Login() {
    // TODO implement here
}

/**
 *
 */
public void Logout() {
    // TODO implement here
}

/**
 *
 */
public void Register() {
    // TODO implement here
}

/**
 *
 */

```

```
public void Delete_account() {  
    // TODO implement here  
}  
  
/**  
 *  
 */  
public void Chat() {  
    // TODO implement here  
}  
  
/**  
 *  
 */  
public void Call() {  
    // TODO implement here  
}  
  
/**  
 *  
 */  
public void Posts() {  
    // TODO implement here  
}  
  
/**  
 *  
 */  
public void Story() {  
    // TODO implement here  
}  
  
/**  
 *  
 */  
public void Manage_profile() {  
    // TODO implement here  
}  
}
```

CONCLUSION:

Hence, a Social Networking Site or SNS is designed and modelled such that it is dynamic and robust in nature, so that it is user friendly to both our general users and business users, which allows them to perform tasks like chatting, calling, etc for socialising with other users in public and also to promote the business product over online platform. Thus, people of similar interests connect with each other easily and smoothly by our Social Networking Site.