



TEAM MEMBERS

name	id
مطاوع محمد مطاوع عبدالهادي	20210950
ياسين أحمد محمد علي	20211047
محمد ناصر رفعت على	20210841
مصطفى محمد حسان احمد	20210940
معتز أسامة محمد السيد	20210953
مصطفى محمد فوزي محمود	20210944



INDEX

1- Functional requirements

Page (4)

2-System requirements

Page (7)

3- Non-functional requirements

Page (14)

4-use case diagram

Page (21)

5- Sequence & system sequence diagrams

Page (34)

6- Activity diagrams

Page (60)

7- class diagrams

Page (77)

8- object diagram

Page (79)

9- package diagram

Page (81)

10-system architecture

Page (83)

11- database specification

Page (85)

- ERD **Page (86)**
- Tables **Page (89)**

PURPOSE

25) A Microblogging Application [inspired by Twitter]

Microblogging is an online broadcast medium that exists as a specific form of blogging. A micro-blog differs from a traditional blog in that its content is typically smaller in size. Micro-blogs "allow users to exchange small elements of content such as short sentences, individual images, or video links", which may be the major reason for their popularity. These small messages are sometimes called micro posts. Twitter (as an example of a Microblogging & Social Networking Service) is an application on which users post and interact with messages known as "tweets". Registered users can post, like, and retweet tweets, but unregistered users can only read those that are publicly available. Tweets are publicly visible by default, but senders can restrict message delivery to only their followers. Users can mute users they do not wish to interact with, block accounts from viewing their tweets, and remove accounts from their followers' list. Users may subscribe to other users' tweets—this is known as "following" and subscribers are known as "followers". Individual tweets can be forwarded by other users to their own feed, a process known as a "retweet". A "quote tweet" (originally called

Page 14 of 16



Helwan Uni. | Faculty of Computers & Artificial Intelligence | Computer Science Dept.
Module: CS251 Software Engineering 1 – Spring "Semester 2" 2022-23



"retweet with comment") is a feature that allows users to add a comment to their retweet, nesting one tweet in the other. Users can also "like" (formerly "favorite") individual tweets. The counters for "likes", "retweets", and replies appear next to the respective buttons in timelines such as on profile pages and search results. Counters for likes and retweets exist on a tweet's standalone page too. Quote tweets, formerly known as "retweet with comment", have their own counter on their tweet page. For ideas on more features, you may check: https://en.wikipedia.org/wiki/List_of_Twitter_features



FUNCTIONAL REQUIREMENTS

The software has two modules, Admin, User and system.

ADMIN

- 1. Log in**
- 2. Log out**
- 3. Send message**
- 4. Receive message**
- 5. Remove user**
- 6. search**
- 7. View profile**
- 8. Edit profile**
- 9. Make admin account**



USER

- **Sign up**
- **Log in**
- **Log out**
- **View profile**
- **Edit profile**
- **Posting a tweet**
- **Like tweet**
- **Retweet tweet**
- **Create hashtag**
- **Follow user**
- **Follow hashtag**
- **Search**
- **Receive notification**
- **Block & Mute user**
- **Chat**
- **Delete tweet**
- **Dark mode**
- **Feedback**



SYSTEM REQUIREMENTS

System requirements

ADMIN

Login

Admin can login the system with his own account

Logout

Admin can logout the system

Send message

Admin have permission to send message to users

receive message

Admin maybe receive message from user

View profile

Admin can view his profile

Edit profile

Admin can Edit his profile information



- Remove user

Admin can remove user account if he use the system in bad way...

- Search

Admin can search about user or tweet

- Make admin account

Admin can make an admin to another admin



1. User Registration and Login:

- i. The application should provide a registration process for new users
- ii. allow them to log in to the system securely.

2. Posting Tweets:

- i. The application should allow users to post micro-posts in the form of text, images, and videos.

3. Following/Follower System:

- i. Users should be able to follow other users
- ii. view their tweets in their timeline,
- iii. users should be able to see who is following them.



4. Like/Retweet System:

- i. Users should be able to like other users' tweets
- ii. Users should be able to dislike other users' tweets
- iii. Users should be able to retweet other users' tweets.

5. Search Functionality:

- i. Users should be able to search for tweets based on keywords and hashtags.
- ii. Users should be able to search for other users based on keywords and hashtags.

6. Notification System:

- i. Users should be notified of new tweets, likes, and retweets related to their account.



7. User Blocking/Muting:

- i. **Users should be able to block and mute other users to prevent them from viewing their tweets or interacting with them.**

8. User Profile:

- i. **Each user should have a profile page that displays their name, bio, profile picture, and other relevant information.**
- ii. **User can edit his profile picture**
- iii. **User can edit his profile information**
- iv. **User can edit his country**

9. Hashtag System:

- i. **The application should allow users to create which can be used to categorize tweets.**
- ii. **user can follow this hashtags**



10. Chat:

- Users should be able to send direct messages to each other, allowing for private conversations outside of the public timeline.
 - i. **User can send messages to another user.**
 - ii. **User can receive message from another user.**
 - iii. **Create a private conversation channel.**

11. Analytics:

- The application should provide users with analytics on their tweets such as:
 - i. **engagement rates**
 - ii. **impression numbers**
 - iii. **other relevant metrics.**



NON-FUNCTIONAL REQUIREMENTS

Scalability

- The application should be able to handle a large number of users and tweets without slowing down.
- The application should be able to scale up or down depending on the number of users and tweets.

Reliability

- The application should be reliable and available at all times with minimum downtime.
- The application should be reliable and able to recover quickly from any errors or crashes.

SECURITY

- - To use the system, it is a must to have an account
- - The system must distinguish between admin and user roles.
- - Each role has his own functions which can perform. So that, no roles can be overlapped.
- - Password encryption to enhance privacy and security.
- - There is a problem of use fake accounts
- - The application should ensure that user data is secured.
- - Access to the

Availability

- The application should be available 24/7, with minimal downtime for maintenance or updates.

Look and feel

- The system shall less of use colors
- the system should use a lot of animation
- This system is user friendly.
- Approachable, so that people do not hesitate to use it.
- Authoritative, so that users feel they can rely on it and trust it.

USABILITY AND HUMANITY

- - Centralized maintain all information.
- - Easy to search the information
- - Facilitate communication
- - Easy to use on the first attempt by someone without training.
- - The application should be easy to use and navigate, with a simple and intuitive user interface.

PERFORMANCE REQUIREMENTS

TIME PERFORMANCE:

- -The application should have fast response times and load pages quickly
- -The application should be able to handle a high volume of traffic and user activity without experiencing lag or slow response times.
- -The web page load time should not exceed 1 second.
- - Any operation that will be done by any system user will not exceed 1 second.

ACCURACY PERFORMANCE:

- Accurate in search
- Accurate in share data
- Accurate in check user account
- Accurate in Dealing with data



MAINTAINABILITY AND SUPPORT REQUIREMENTS

- - The system is a website. So, it can be opened in any browser.
- - The system shall be able to be modified to cope with minor changes to different countries' rules.

CULTURAL REQUIREMENTS

- The language used in the interface should be formal and polite which includes Political Requirements

DESIGN CONSTRAINTS

- - Each password must has a pattern to follow (At least 8 characters, a mixture of both uppercase and lowercase letters, a mixture of letters and numbers, inclusion of at least one special character and don't have , ',').
- - inputted information like email, phone and date of birth must follow their pattern.
- - Each username must be unique.

Use Case Diagram

Use Case Description

Use Case Name	Sign up
Actors	User
Main scenario	User will enter the register interface , and user should fill his first name and last name and new password then confirm his password and enter his phone then press submit then he can create account.
Pre-condition	User is not logged in to the system.
Post-condition	System verifies information and creates account and is returned to the home page as a logged in user.
exception	Data not valid, display rejected message

Use Case Name	Login
Actors	User & Admin
Main scenario	The actor will open home page, and login if the password is verified.
Pre-condition	Open app interface
Post-condition	The actor can do all his operations .
exception	Fail to login and stay in the home page,display login error

Use Case Name	Logout
Actors	User & Admin
Main scenario	The actor will click the logout button, and the App interface will be opened instead of current one.
Pre-condition	Actor is logged in
Post-condition	The actor cannot do any operation until he login again.
exception	Fail to logout as an operation is working, still in current page.

Use Case Name	View profile
Actors	User & Admin
Main scenario	The actor will click the profile button, the profile data page will be opened and can edit or update his/her data and password.
Pre-condition	Actor is logged in
Post-condition	Open and see profile data
exception	

Use Case Name	Edit profile (edit country, edit info, edit profile picture)
Actors	User & Admin
Main scenario	The actor will click the profile button, then click on Edit profile and can edit or update his/her data and password.
Pre-condition	Actor is logged in
Post-condition	Actor profile has been edited
exception	Data not valid, display rejected message

Use Case Name	Search (hashtag, user, tweet)
Actors	User & Admin
Main scenario	The System displays the search submission box ,then The User enters search criteria and submits ,then The system displays results
Pre-condition	Actor is logged in
Post-condition	The search is successfully executed and system displays results.
exception	If Info not found, display error message.

Use Case Name	Posting
Actors	User
Main scenario	After logging in, the user can post a tweet And save this tweet to the data base
Pre-condition	User is logged in
Post-condition	The tweet must be saved in the database and the users can show this tweet in timeline.
exception	posting failed, show error message/warning.

Use Case Name	Delete tweet
Actors	User
Main scenario	After logging in, the user show post , select a tweet and delete it then save changes to the data base
Pre-condition	User is logged in and have at least one tweet
Post-condition	The tweet has been deleted and can not be seen in timeline
exception	deleting failed, show error message/warning. Or user don't have tweets

Use Case Name	Block \ Mute user
Actors	User
Main scenario	The user select another user and block or mute him then the blocked user can not see the first user's profile and tweets and can not follow him
Pre-condition	User is logged in
Post-condition	The blocked user can't interact with the first user.
exception	blocking failed, show error message. Or blocked user not found.

Use Case Name	Follow user
Actors	User
Main scenario	After logging in, the user can follow another user and see his profile and tweets.
Pre-condition	User is logged in
Post-condition	The user can see the followed user profile and tweets and the followed user's followers increase by one. and the second user notified
exception	following failed, show error message. Or the second user blocked the user.

Use Case Name	Retweet tweet
Actors	User
Main scenario	The user see a tweet by another user then can retweet it to his user page
Pre-condition	User is logged in
Post-condition	This tweet becomes in the user page as a retweet tweet and the second user notified
exception	retweeting failed, show error message / Second user don't allow retweeting

Use Case Name	Like tweet
Actors	User
Main scenario	The user see a tweet by another user then can like this tweet
Pre-condition	User is logged in
Post-condition	This tweet likes increase by one. and the second user notified
exception	Like failed, show error message

Use Case Name	Receive notification
Actors	User
Main scenario	User press on the button notification, user can show notification And know who interact with it's page
Pre-condition	User is logged in
Post-condition	The notifications must be saved in the database
exception	There are no notification

Use Case Name	Chat
Actors	Users
Main scenario	user will search about another user, then send message and the other user will notified, and the other user can receive this message and reply and so on
Pre-condition	Both users is logged in
Post-condition	Messaging session established
exception	The server will not be available and User and his friend will have to wait till connection is established. or one of users blocked another

Use Case Name	Make admin account
Actors	Admin
Main scenario	Admin logged in and make admin account then this account can logged in as a admin and do all admins operations
Pre-condition	Admin logged in
Post-condition	There is a new admin account has been created
exception	Make admin account failed, return to home page

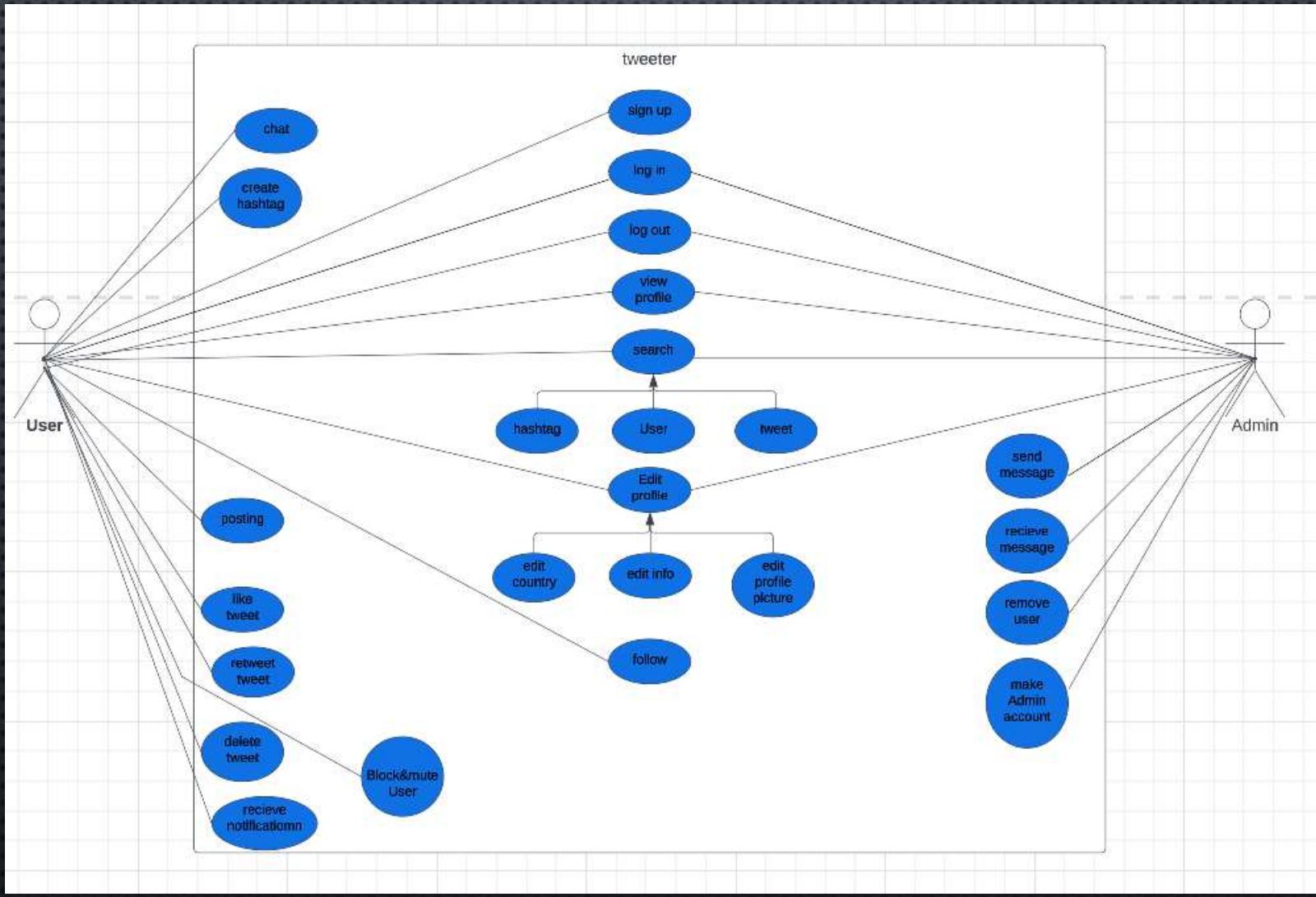
Use Case Name	Send message
Actors	Admin
Main scenario	Admin select user and send message like a warning message
Pre-condition	Admin logged in
Post-condition	User has notified
exception	Message sending error ,display error message

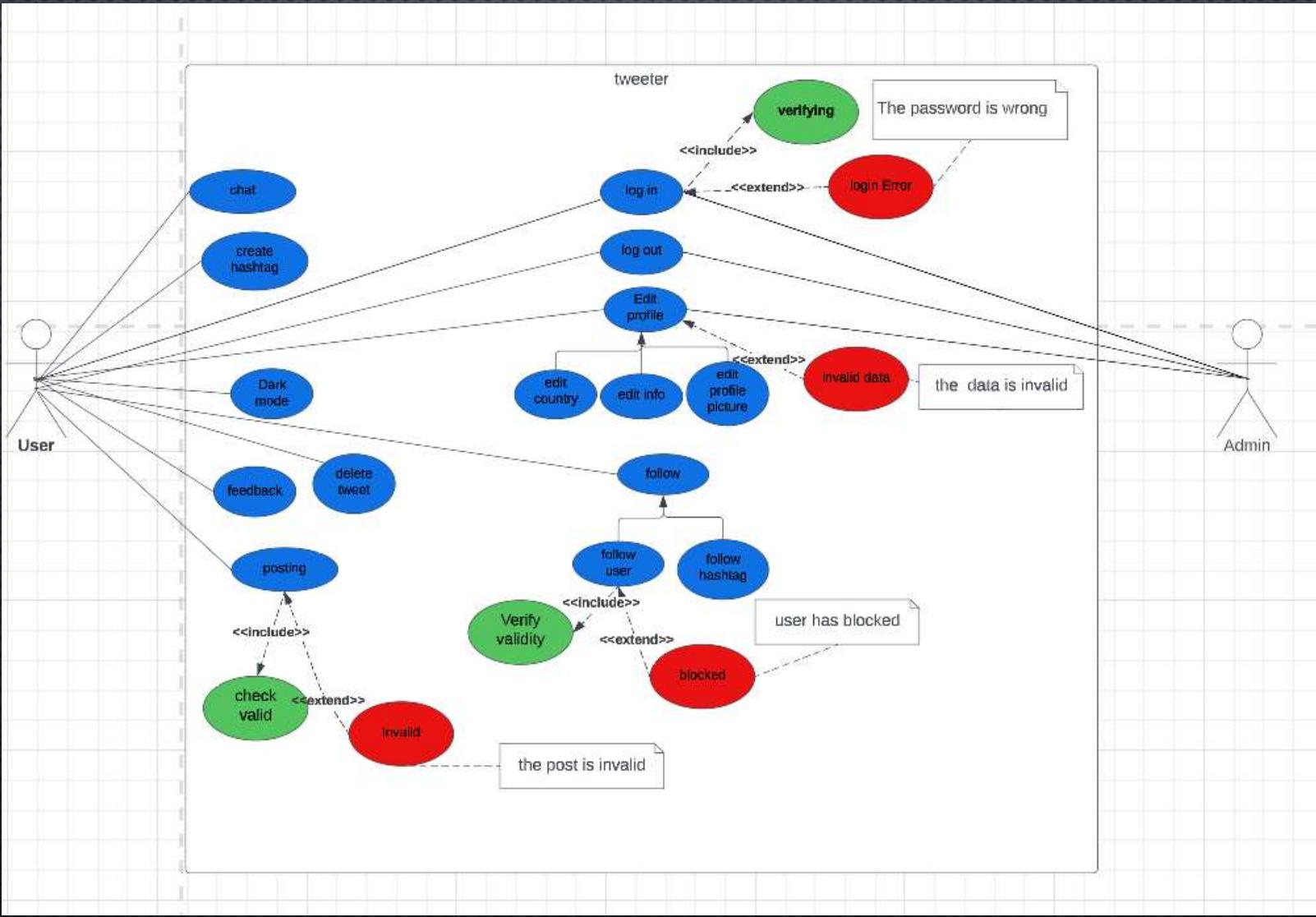
Use Case Name	Receive message
Actors	Admin
Main scenario	User want a help from the system then click on help button and send the problem description.
Pre-condition	user is logged in
Post-condition	Problem are sending to admins and the admins notified
exception	Message receive error ,display error message

Use Case Name	Remove user
Actors	Admin
Main scenario	Admin can remove any user breaks the rules
Pre-condition	Admin logged in
Post-condition	The user has been removed from the system
exception	

Use Case Name	feedback
Actors	user
Main scenario	After logging in, the User will be wanted to put feedback about the system and if he have a complain write it to us.
Pre-condition	User logged in successfully and choose to give feedback to system.
Post-condition	The gives us feedback and be saved in the database and will return to the current page to be able to select another operation to do
exception	Message sending error ,display error message

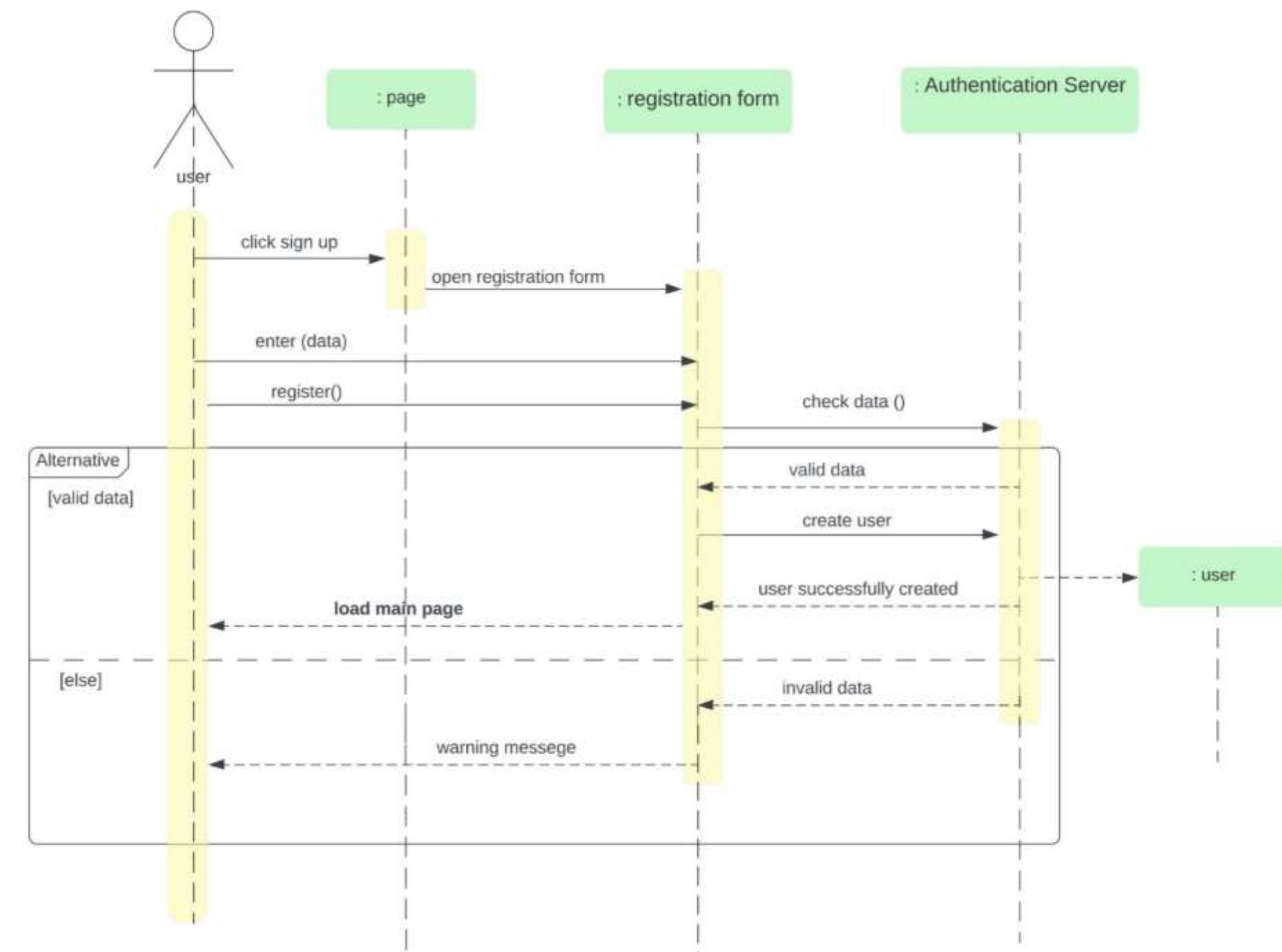
Use Case Name	Dark mode
Actors	User
Main scenario	User want to enable dark mode then click on the dark mode button
Pre-condition	user is logged in
Post-condition	The interface will switch to the dark mode
exception	



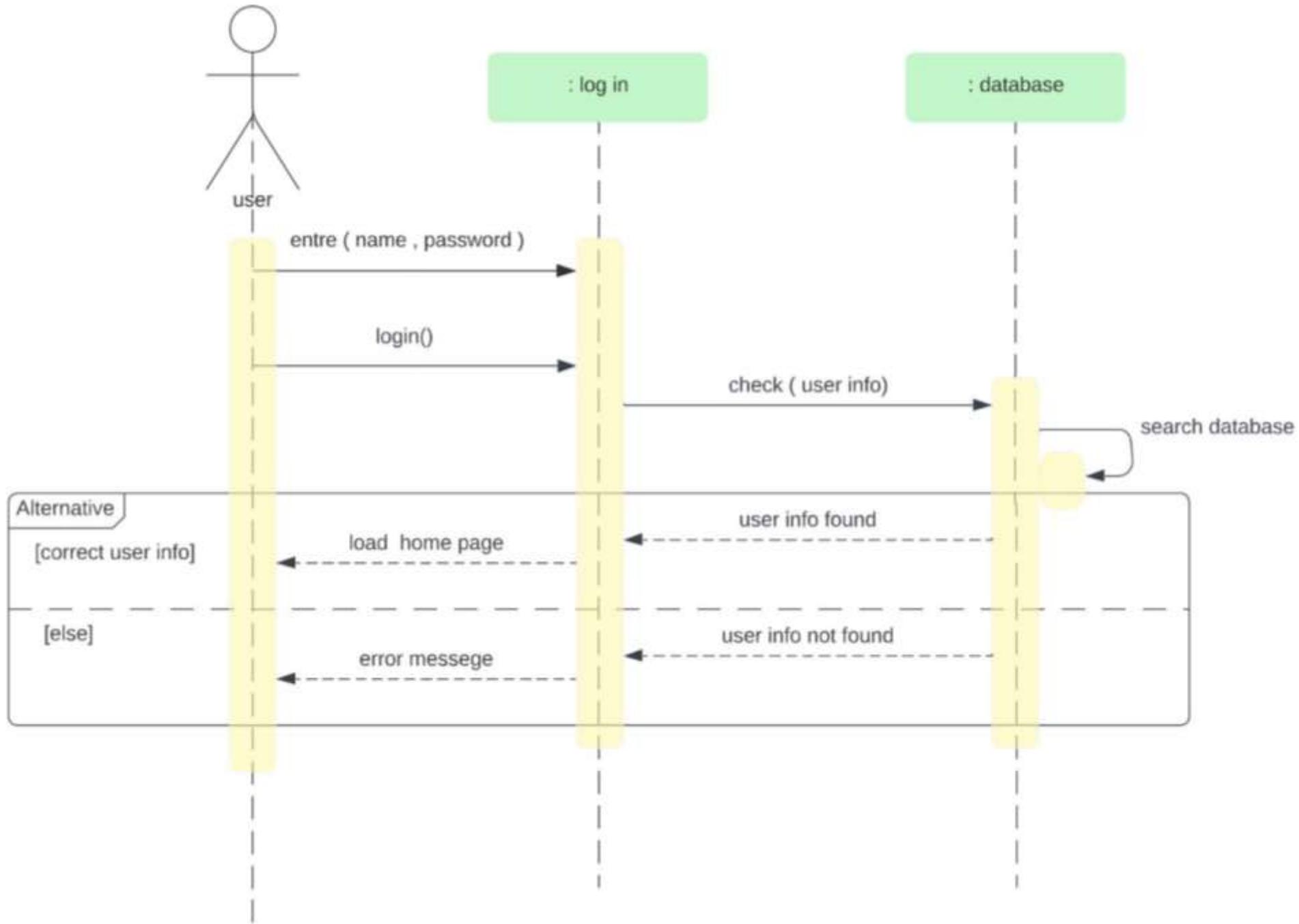


Sequence & System Sequence Diagram

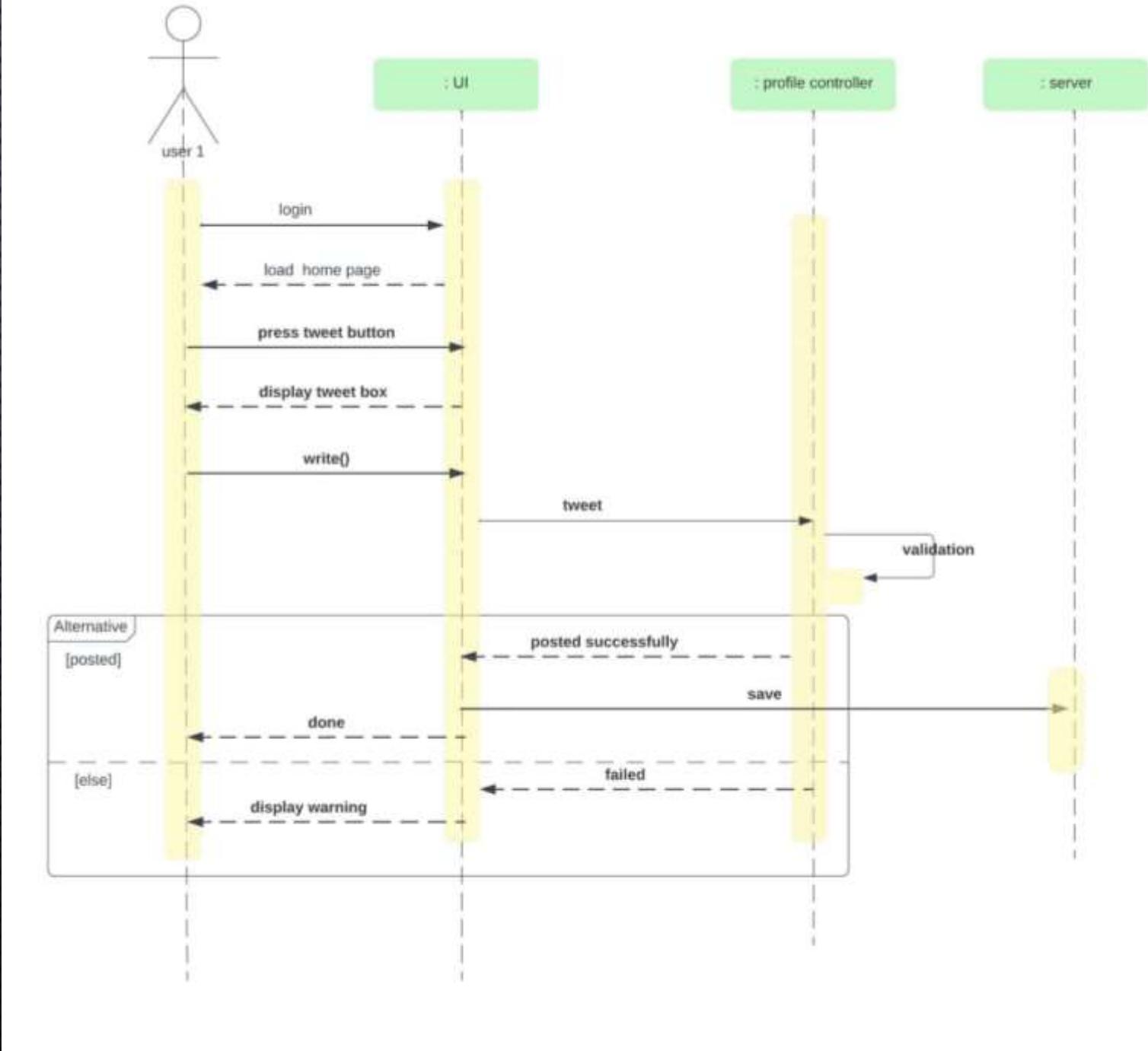
Signup



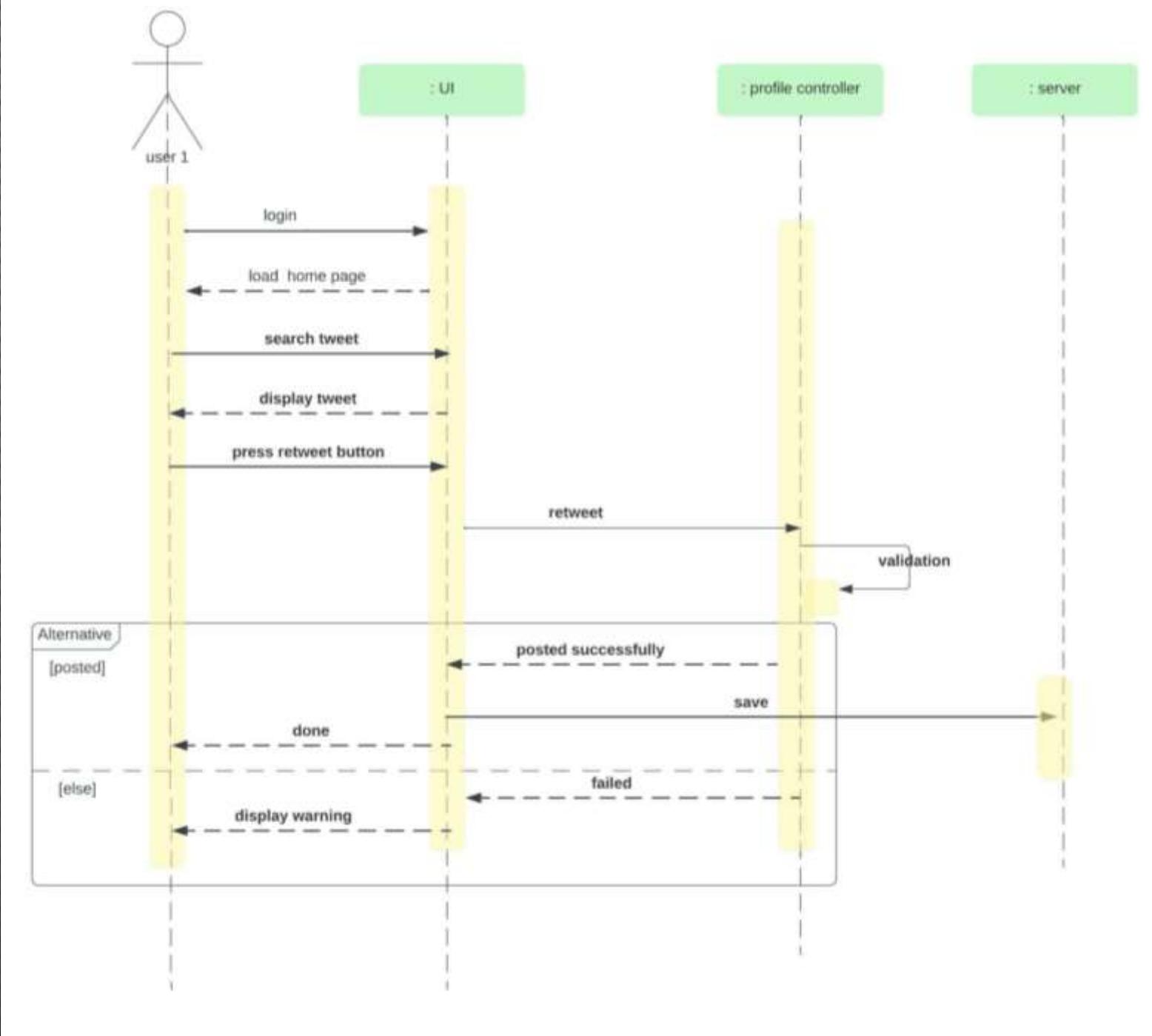
Login



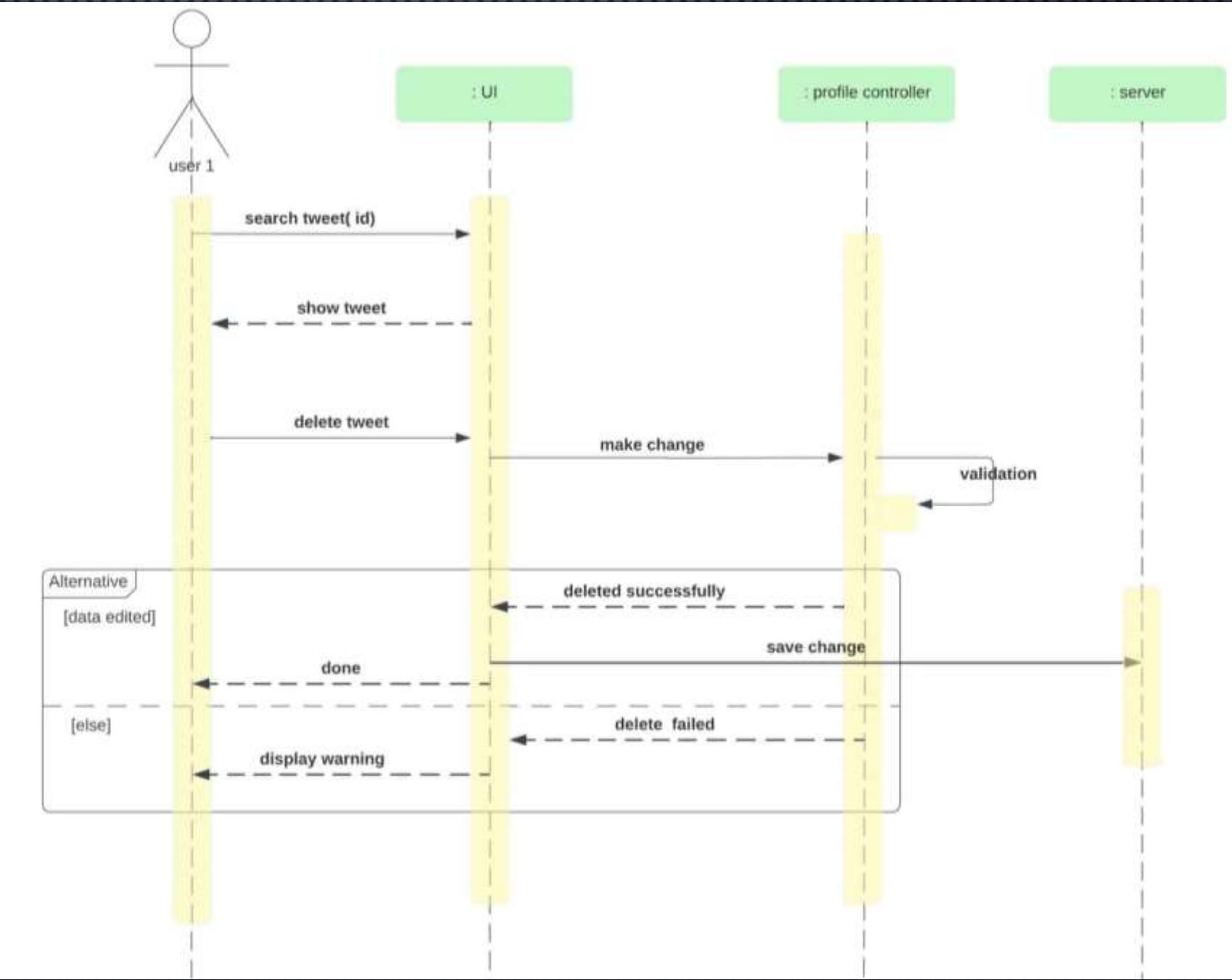
Tweet



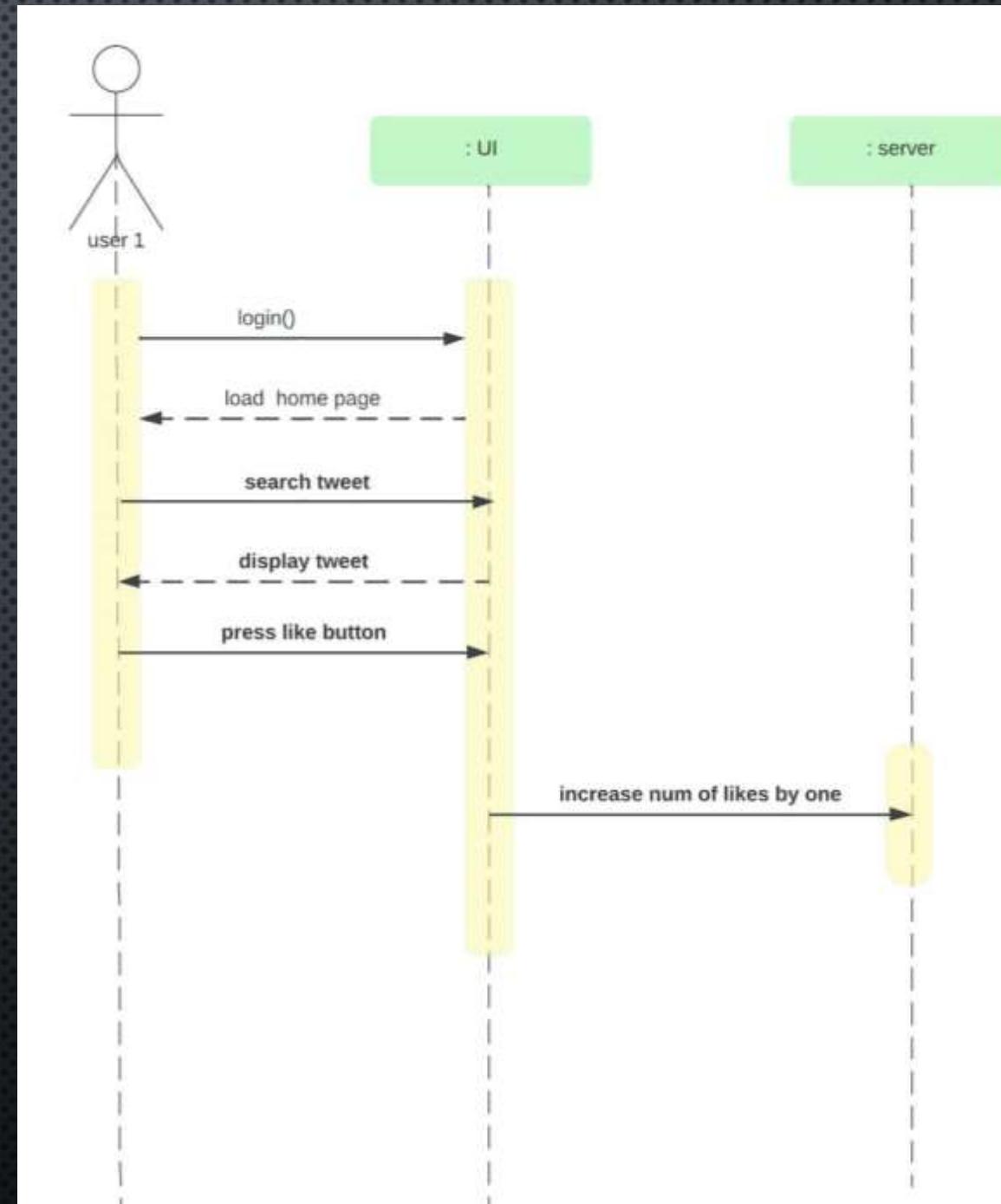
Retweet



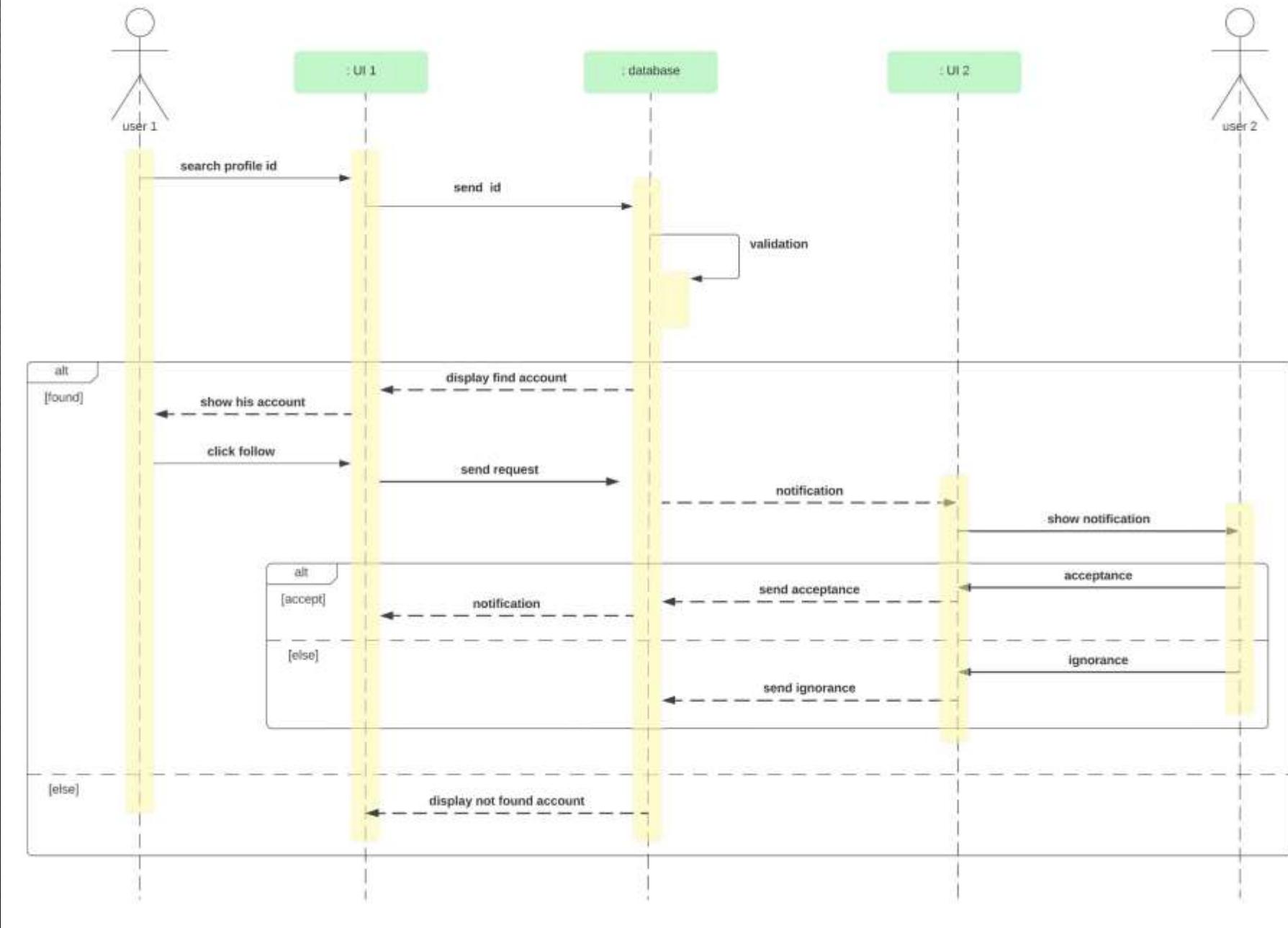
Delete tweet



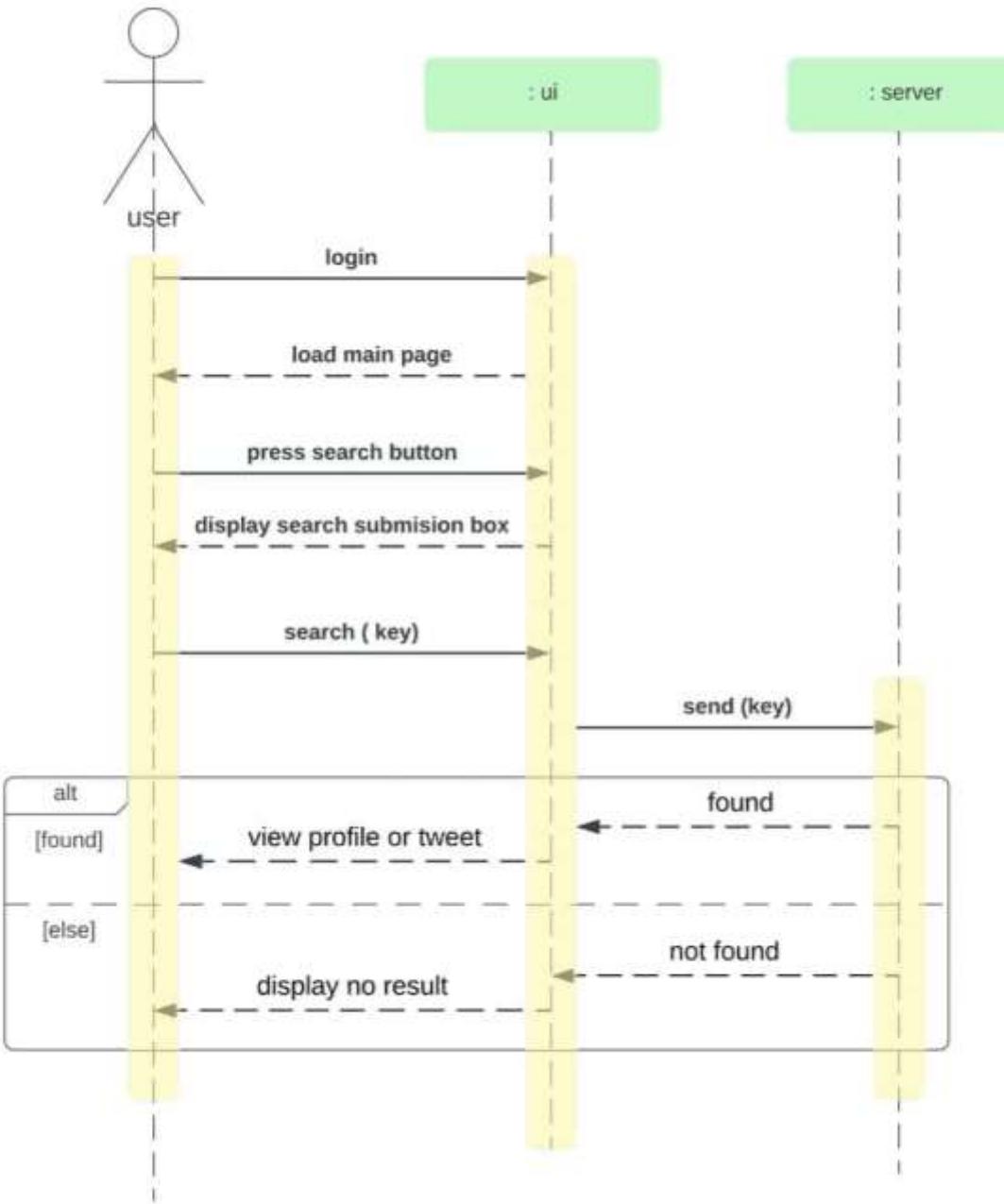
Like Tweet



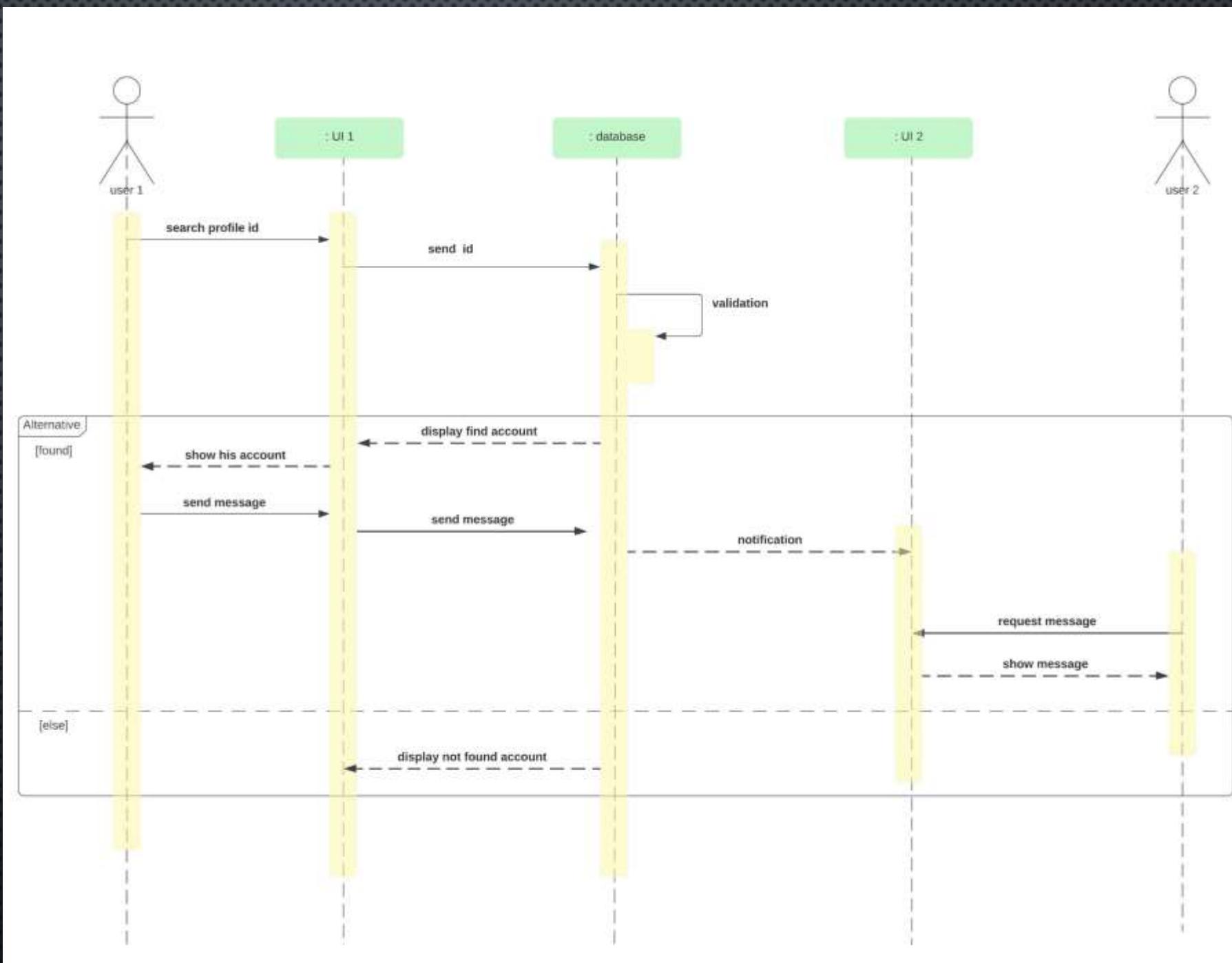
Follow



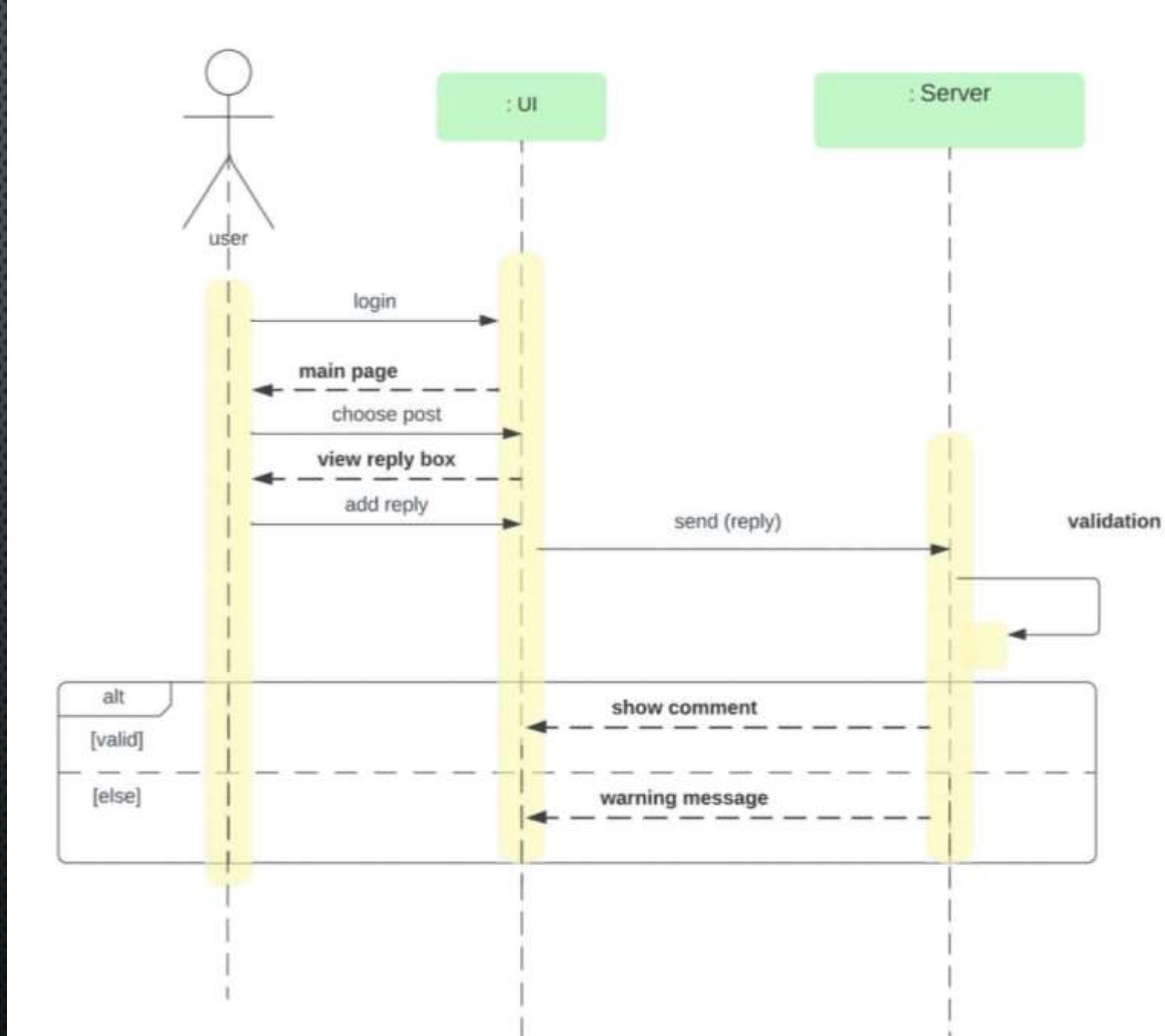
Search



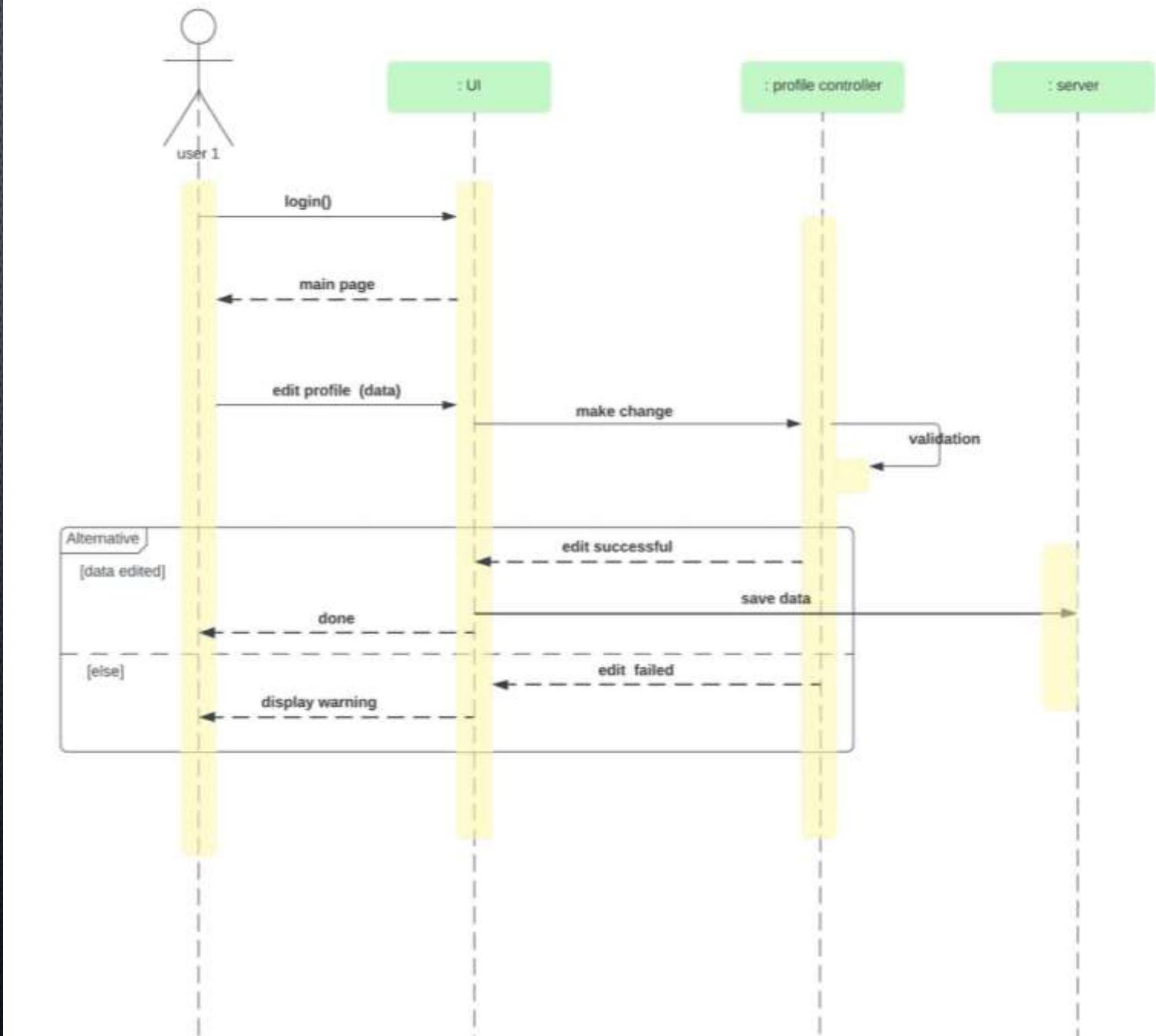
Chat



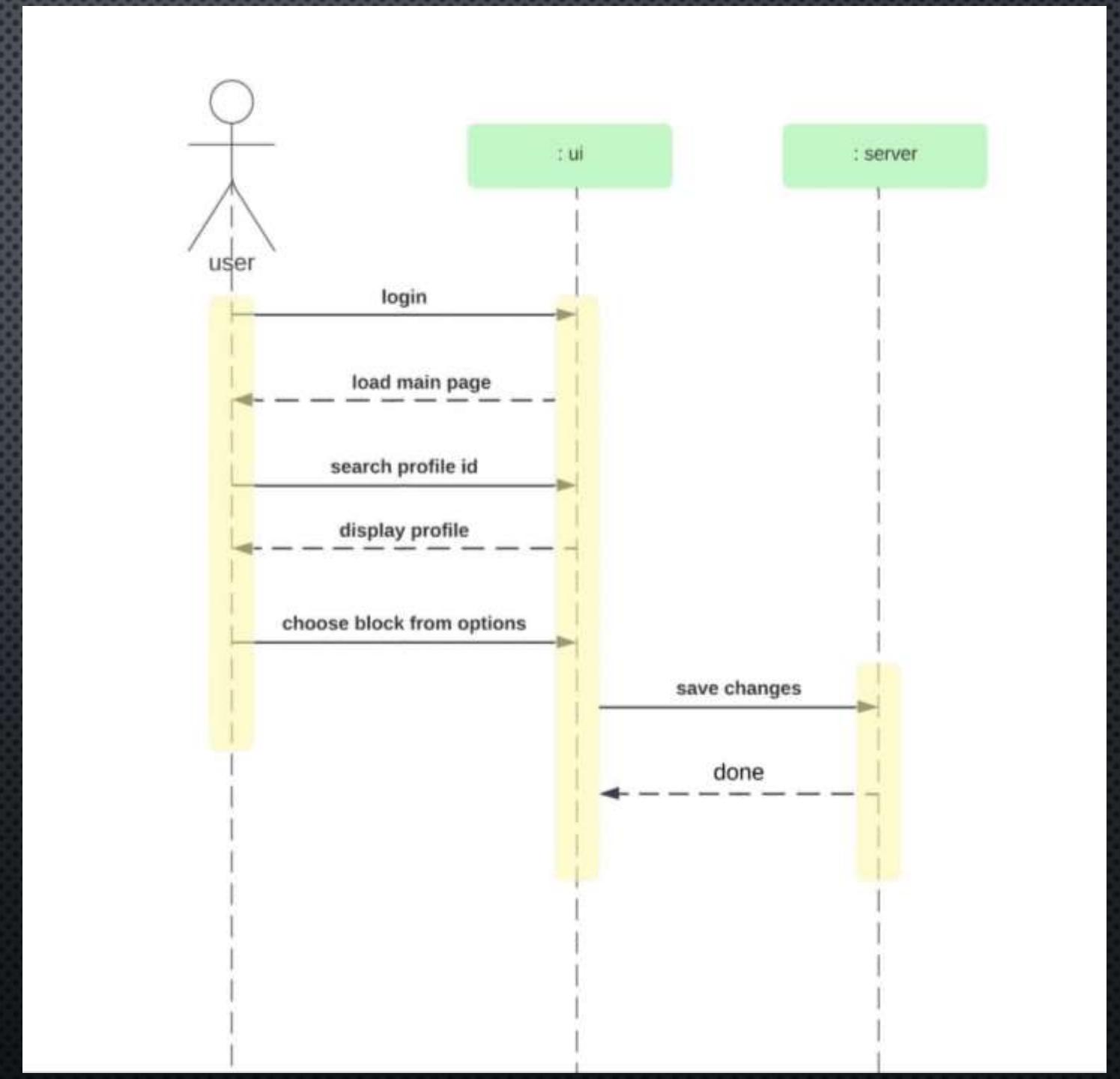
Reply



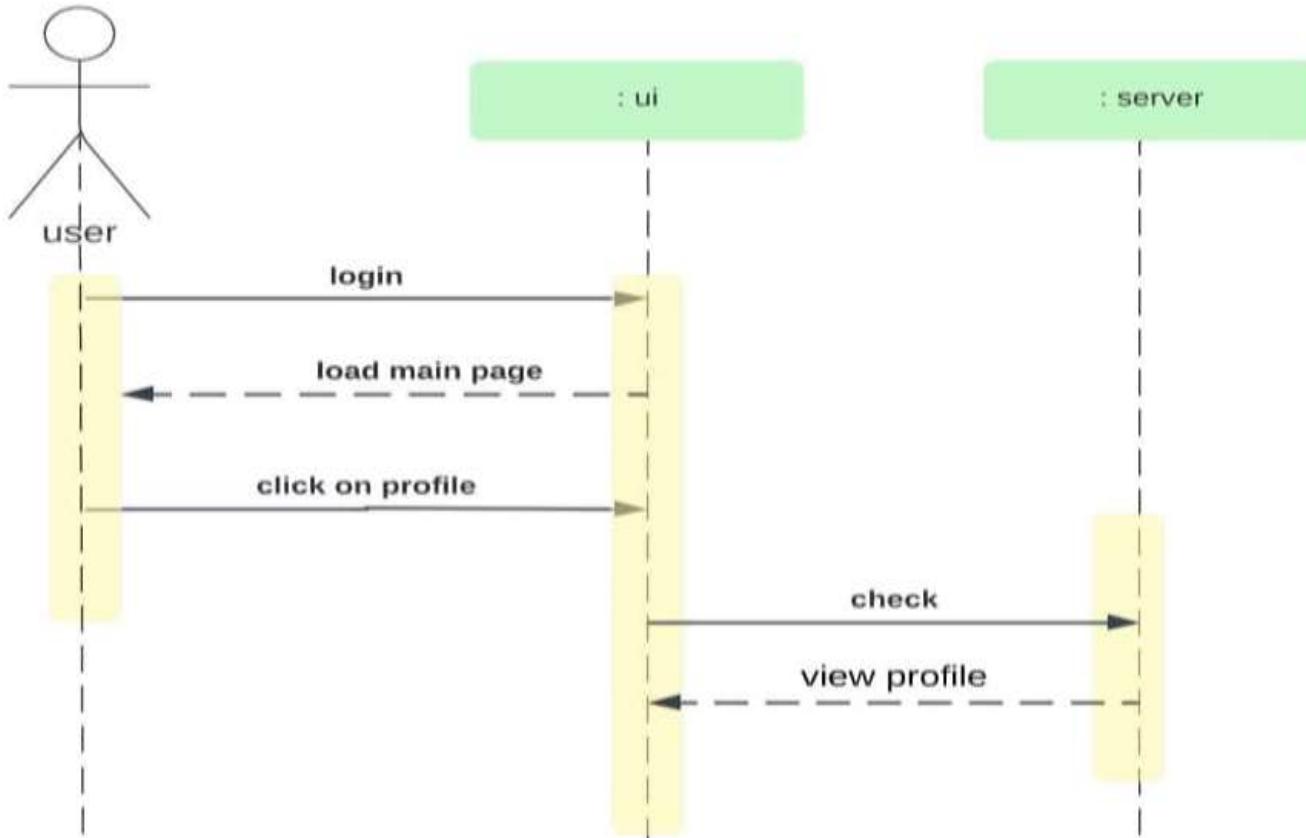
Edit Profile



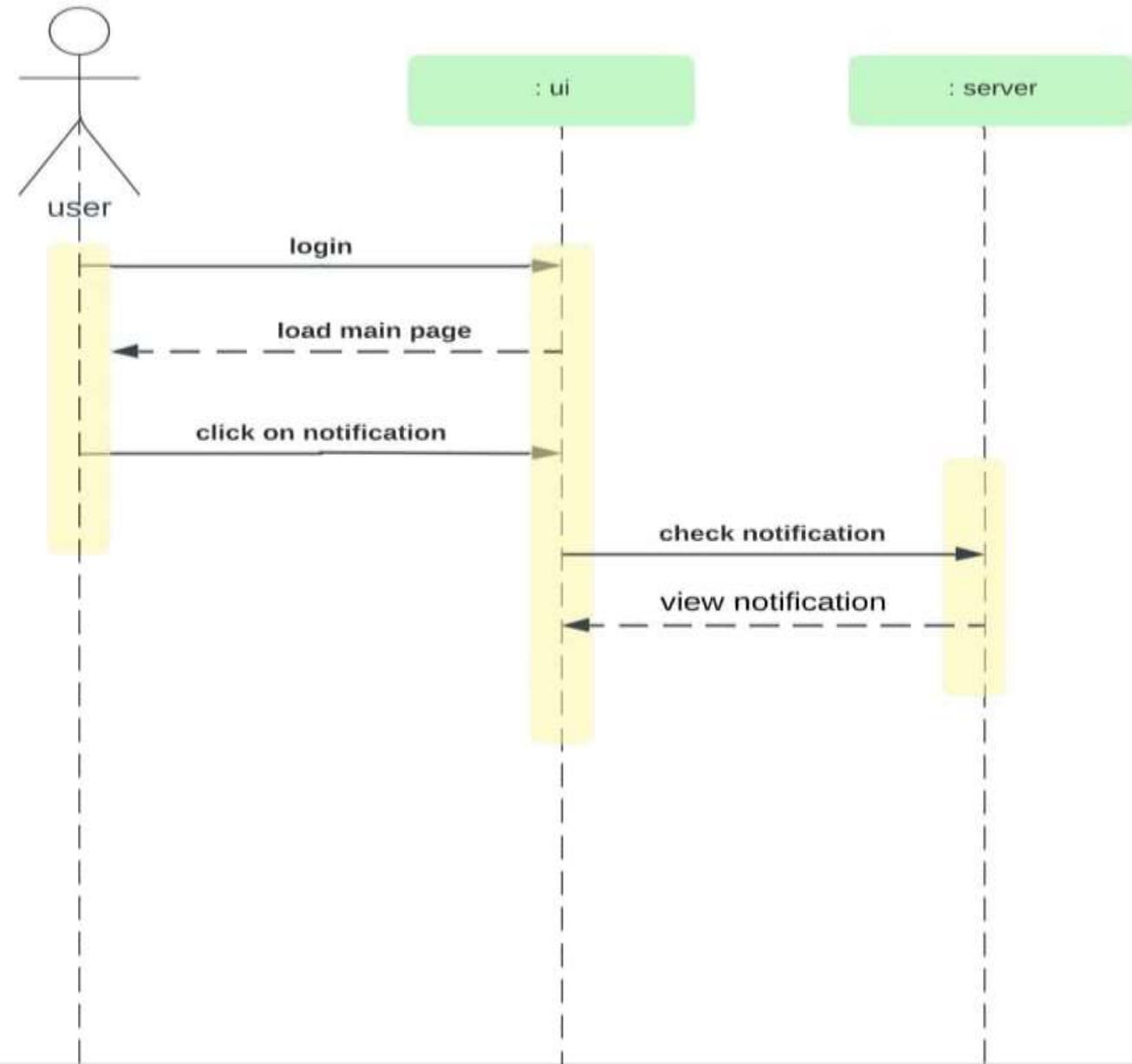
Block User



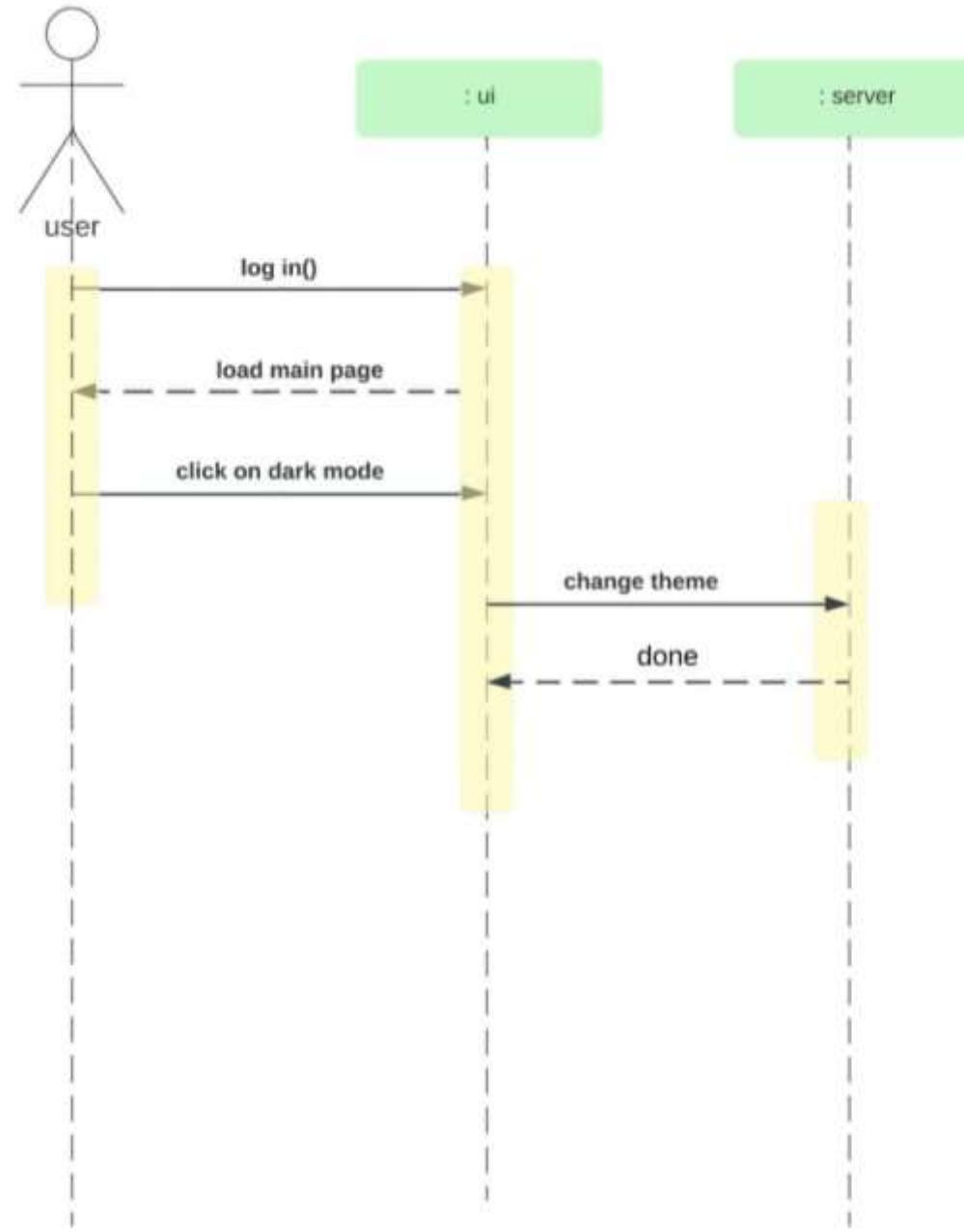
View Profile



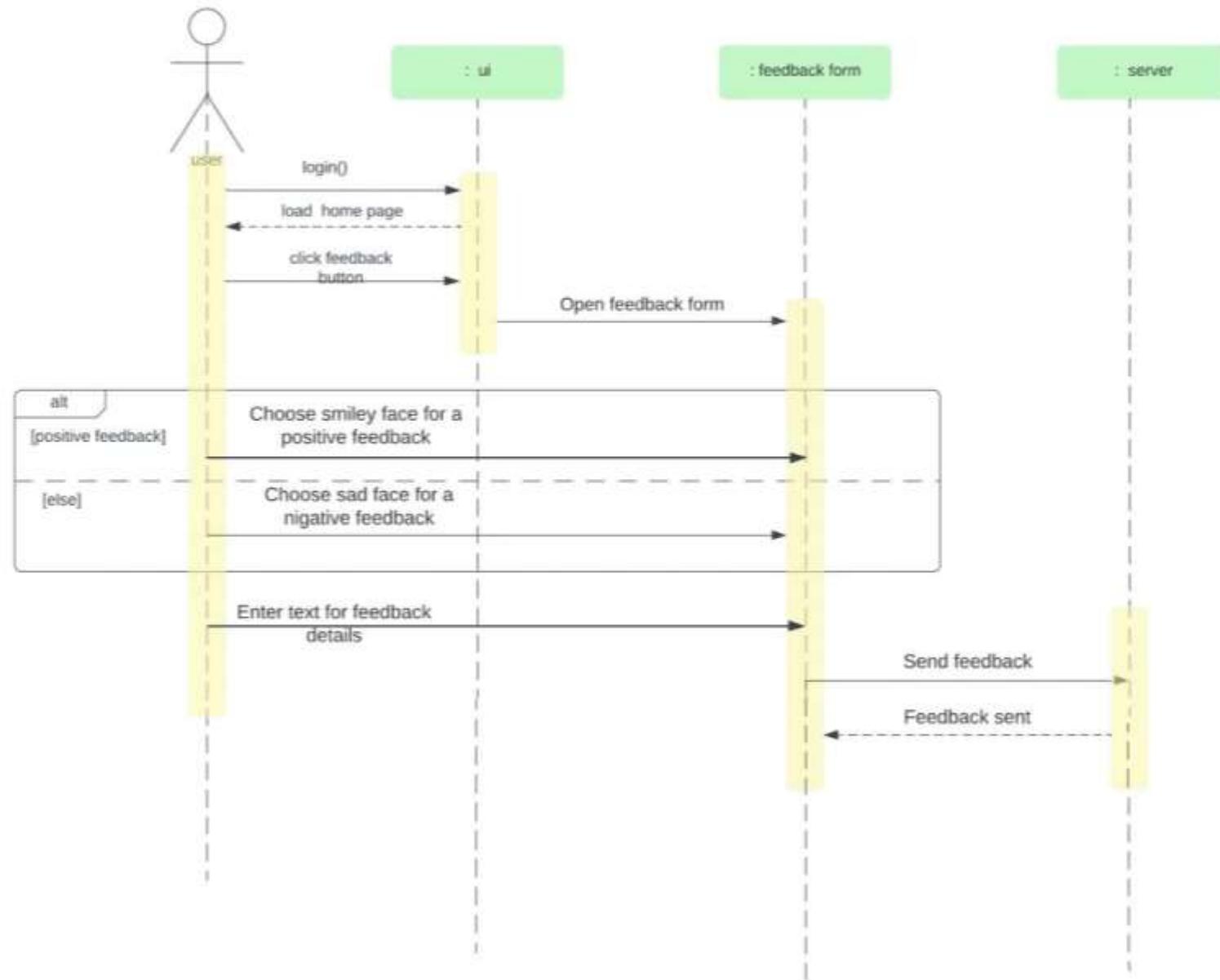
Notification



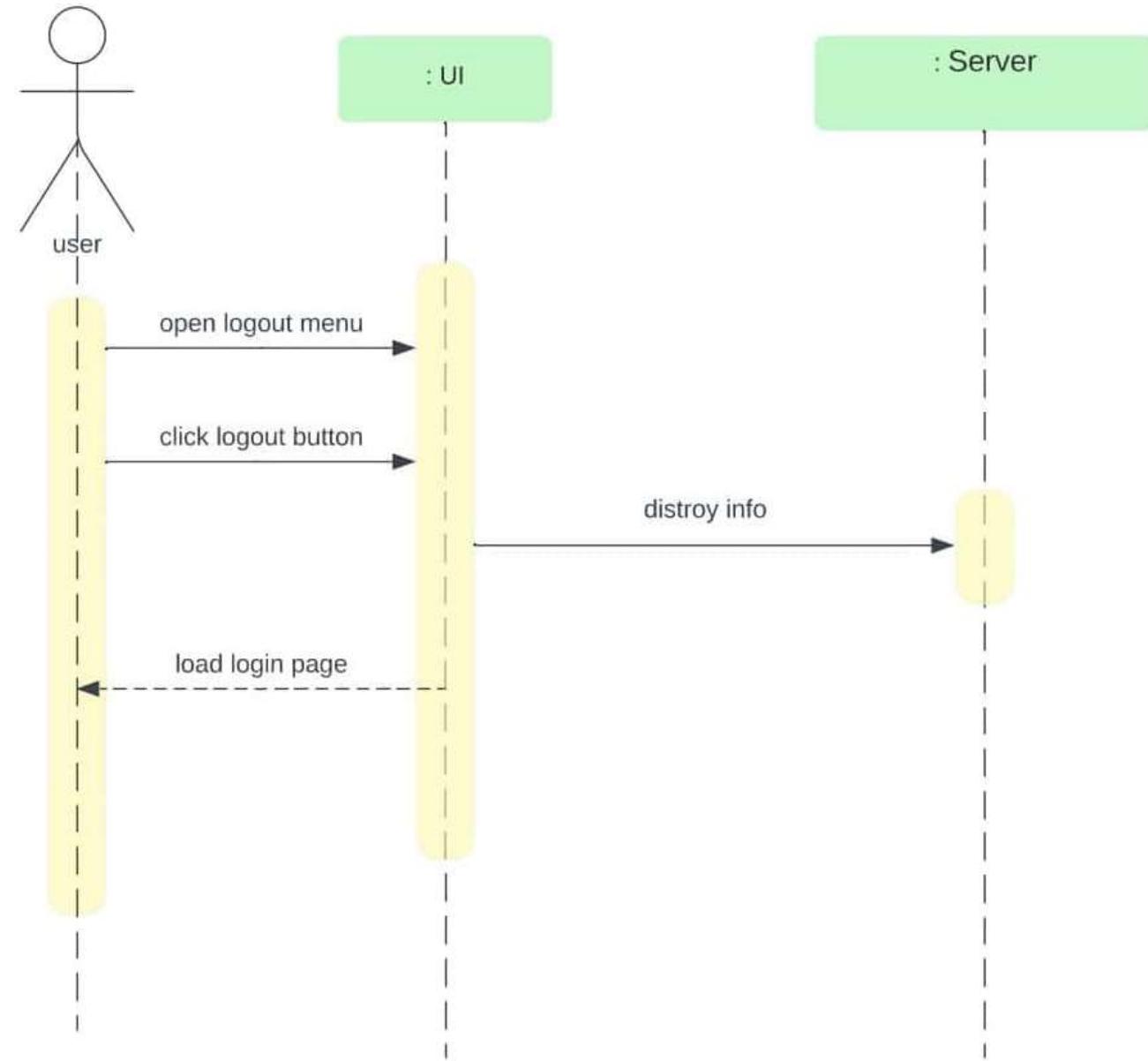
Dark mode



Feedback

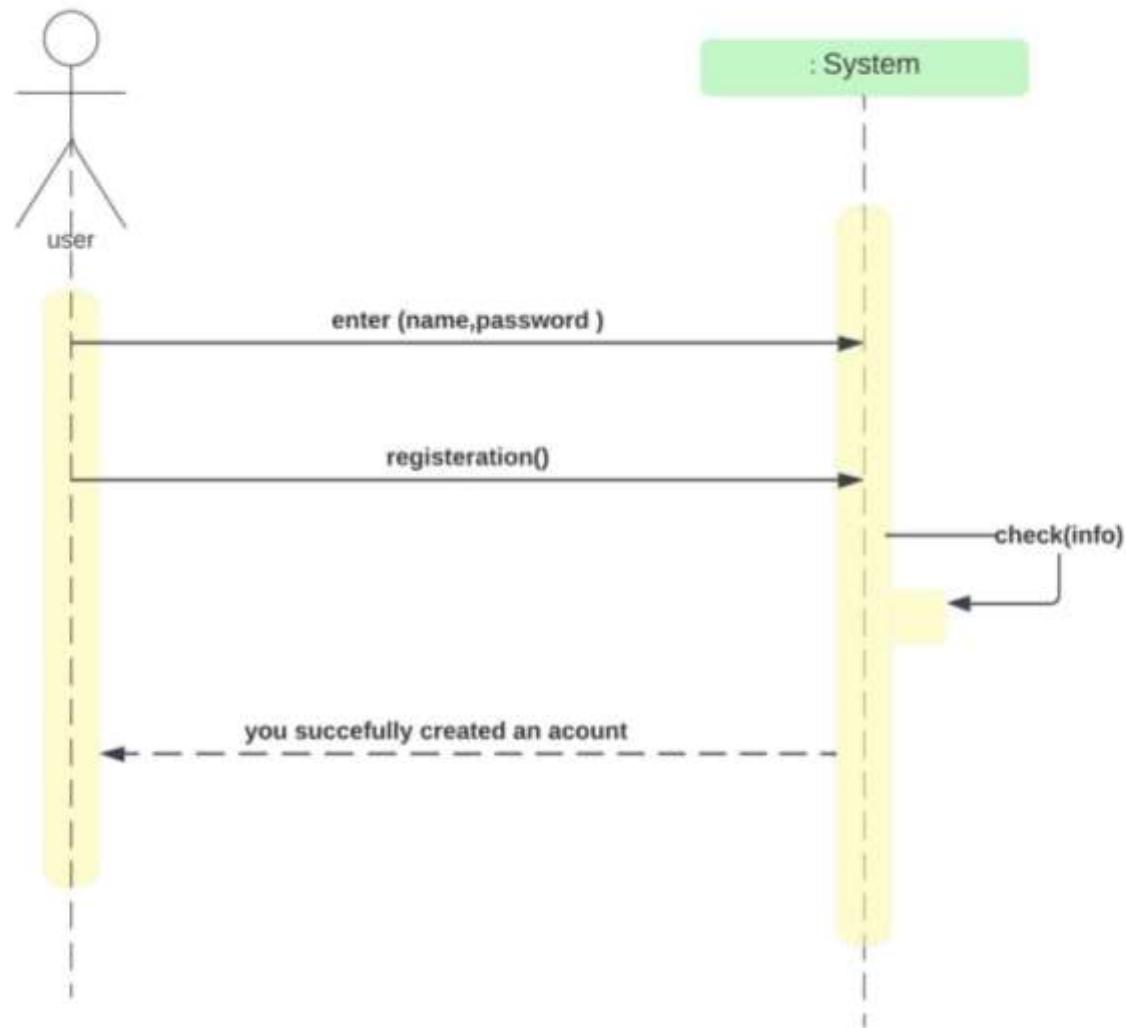


Logout



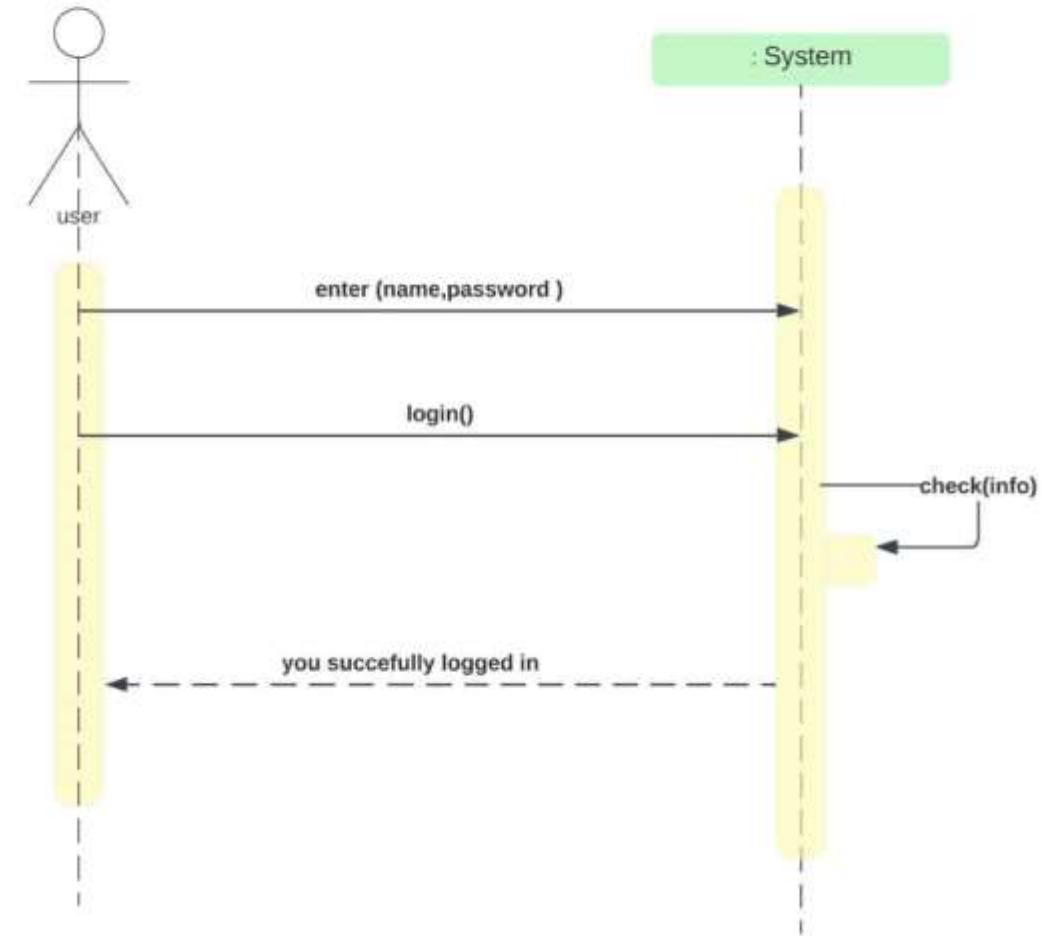
system Sequence diagram

Sign up



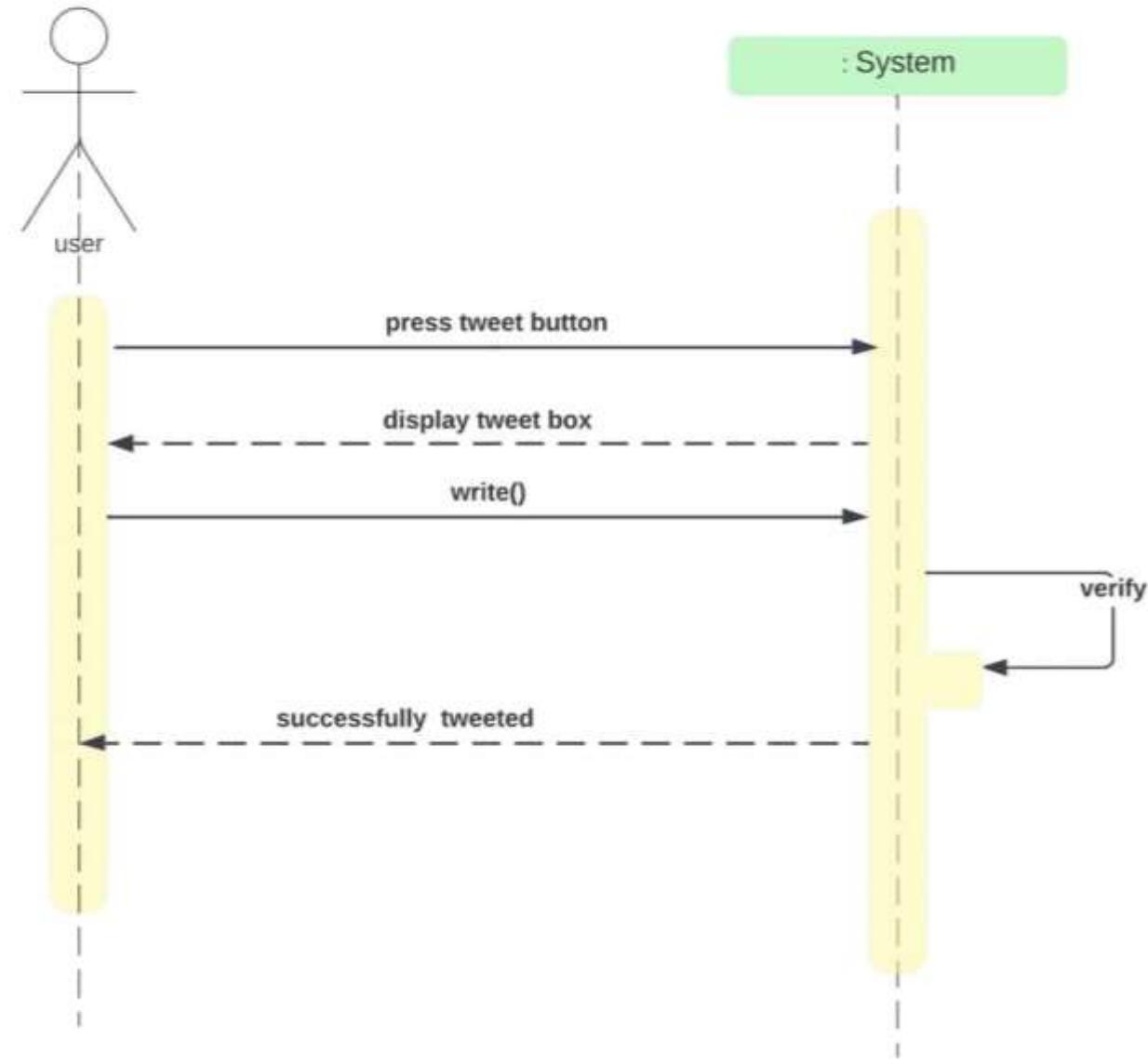
system Sequence diagram

Login



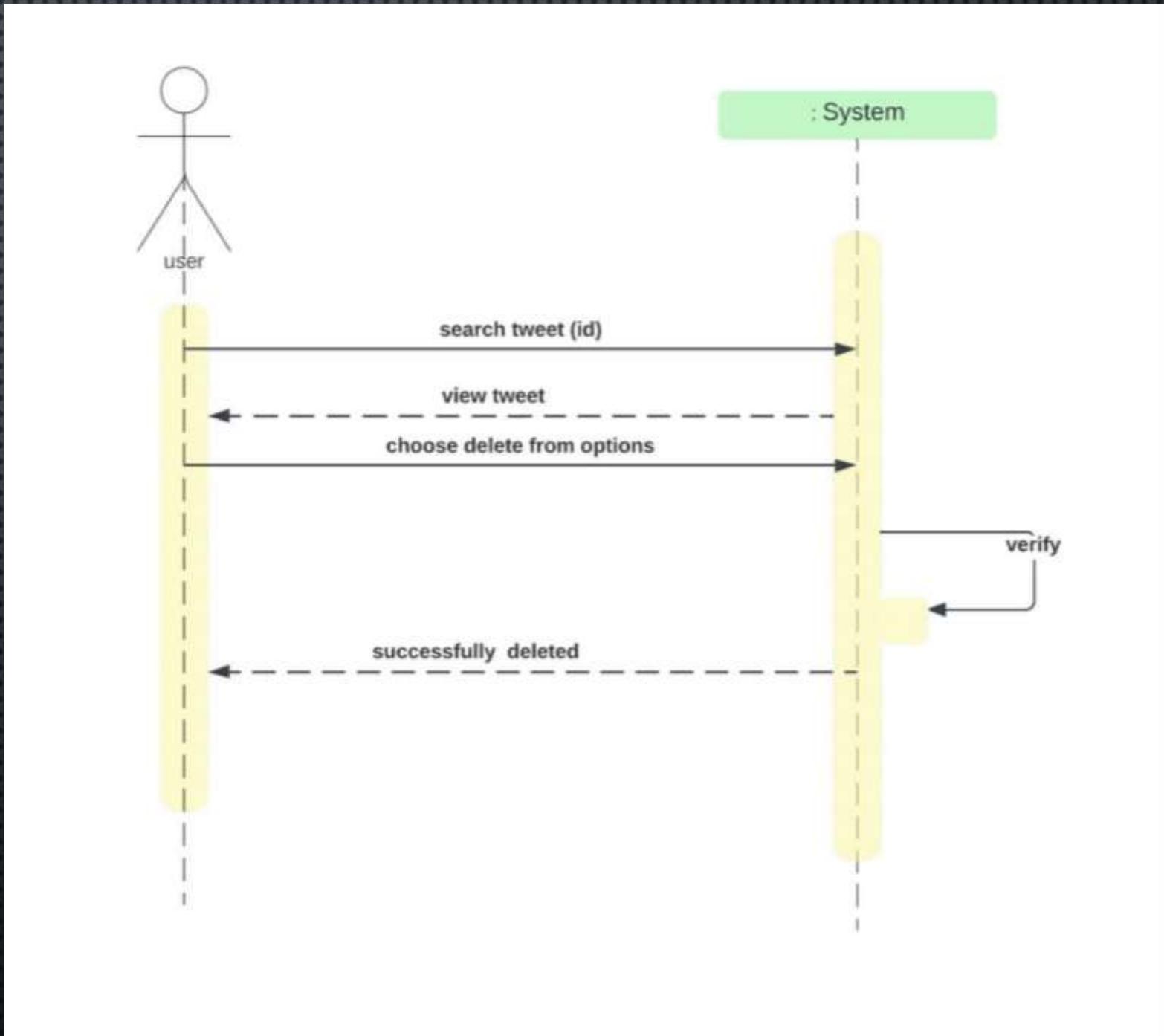
system Sequence diagram

Tweet



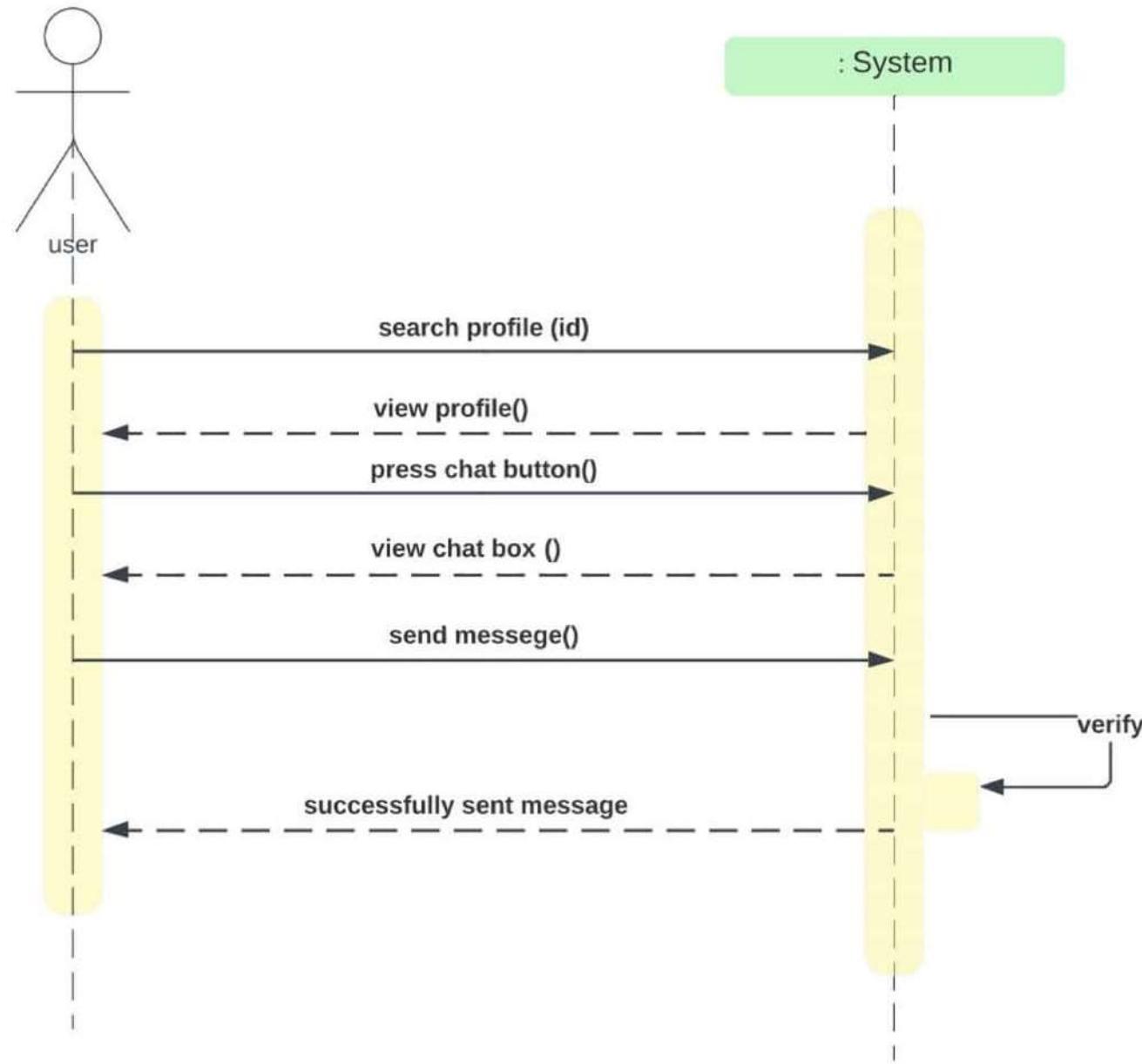
system Sequence diagram

Delete Tweet



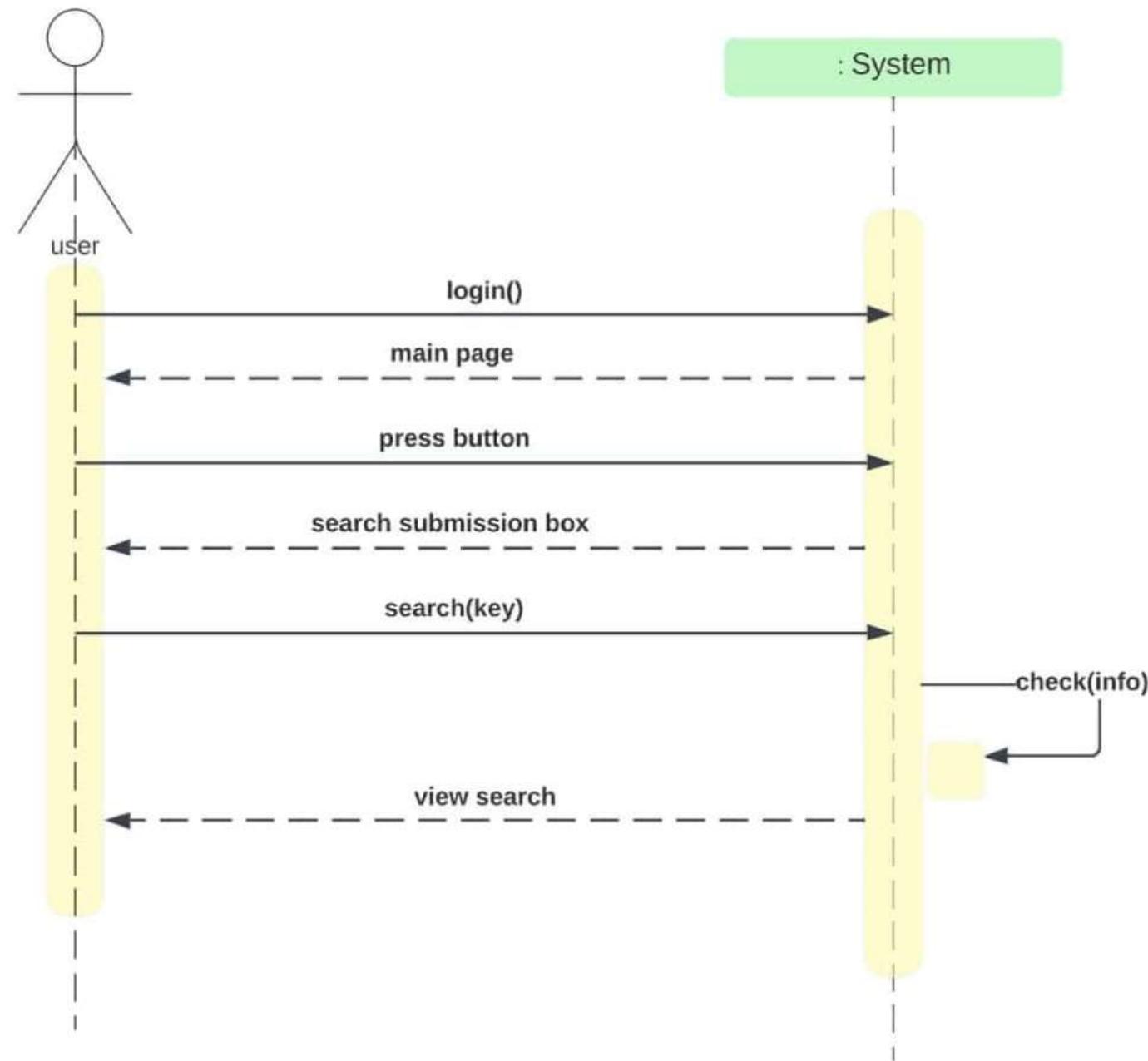
system Sequence diagram

Chat



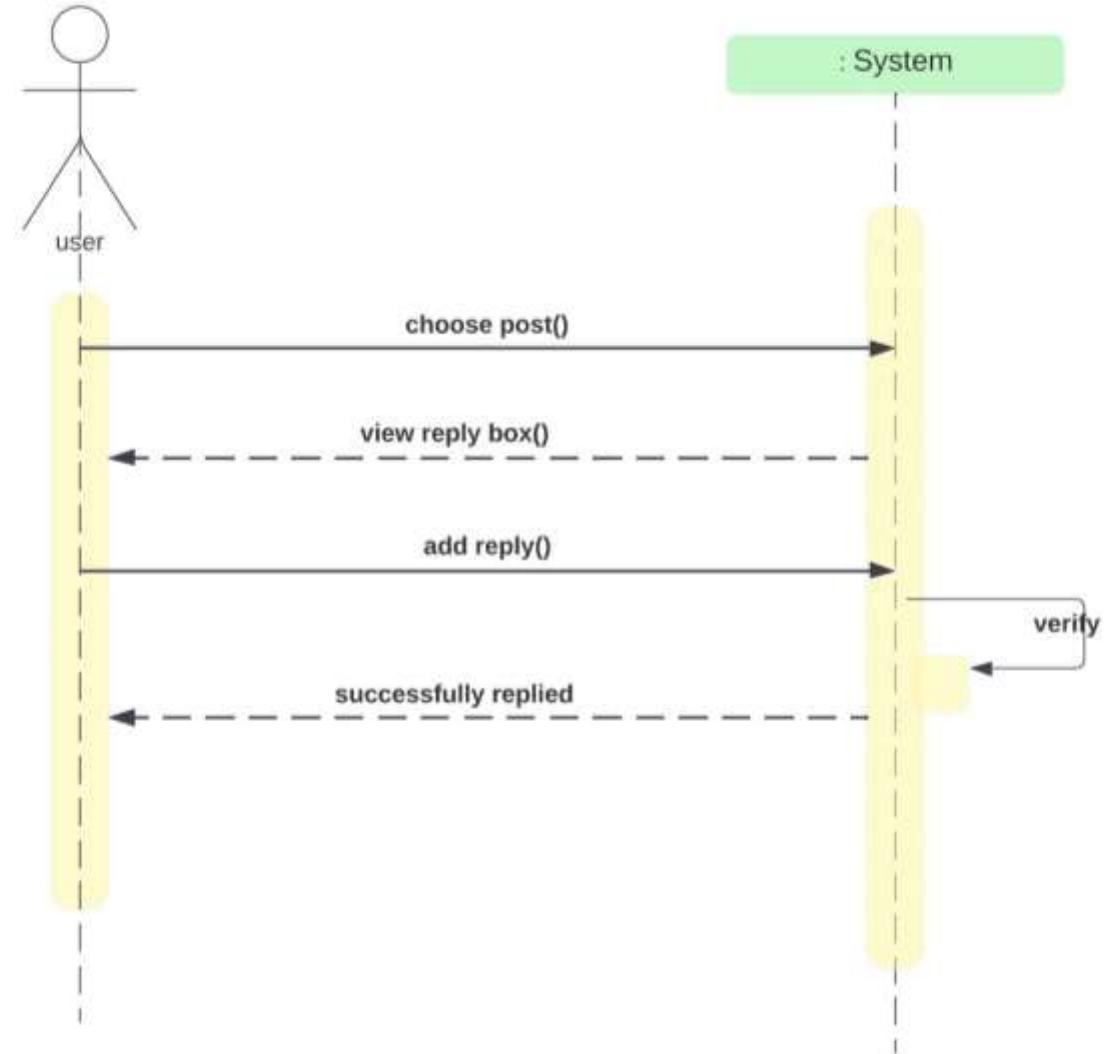
system Sequence diagram

Search



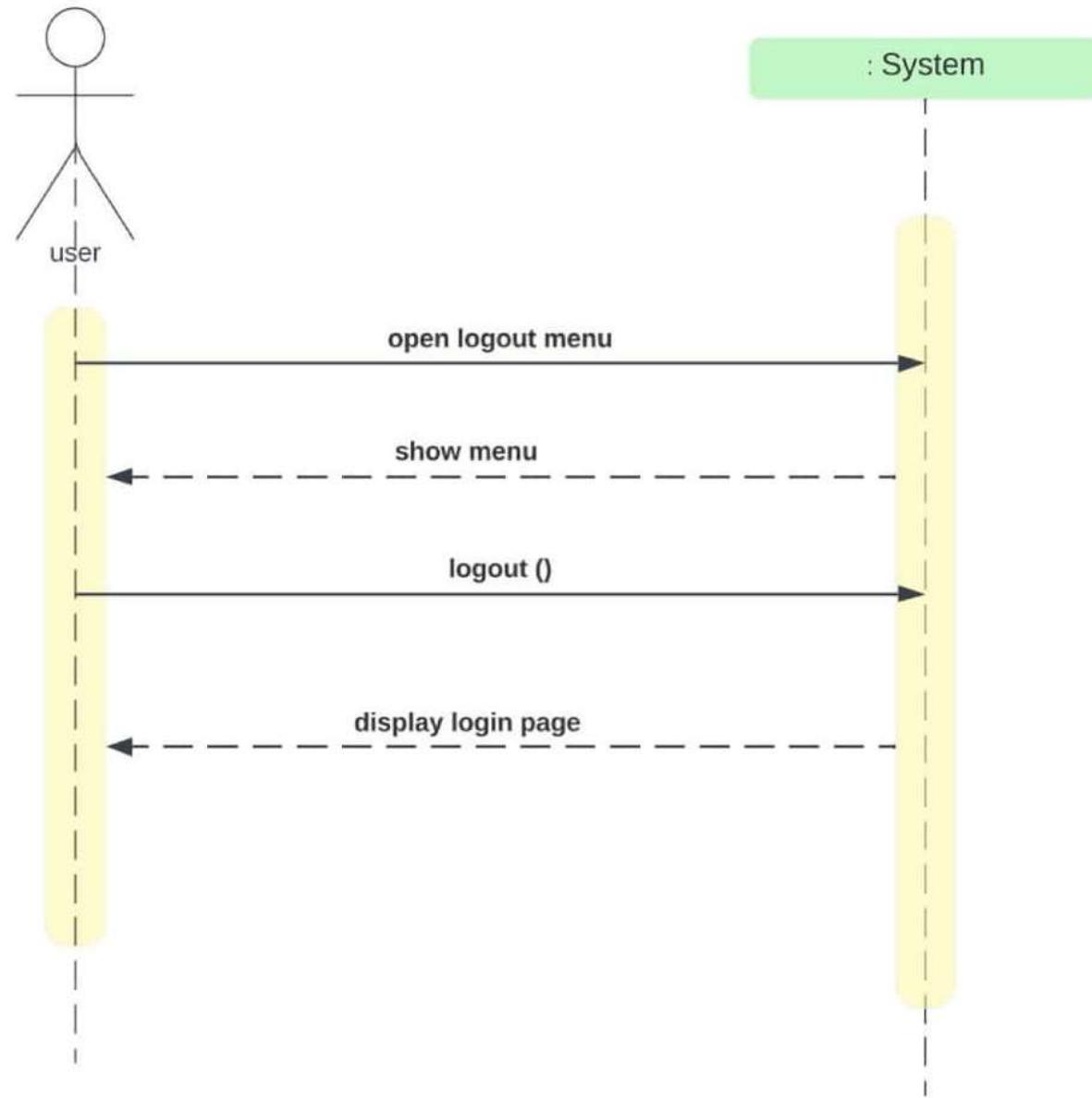
system Sequence diagram

Reply



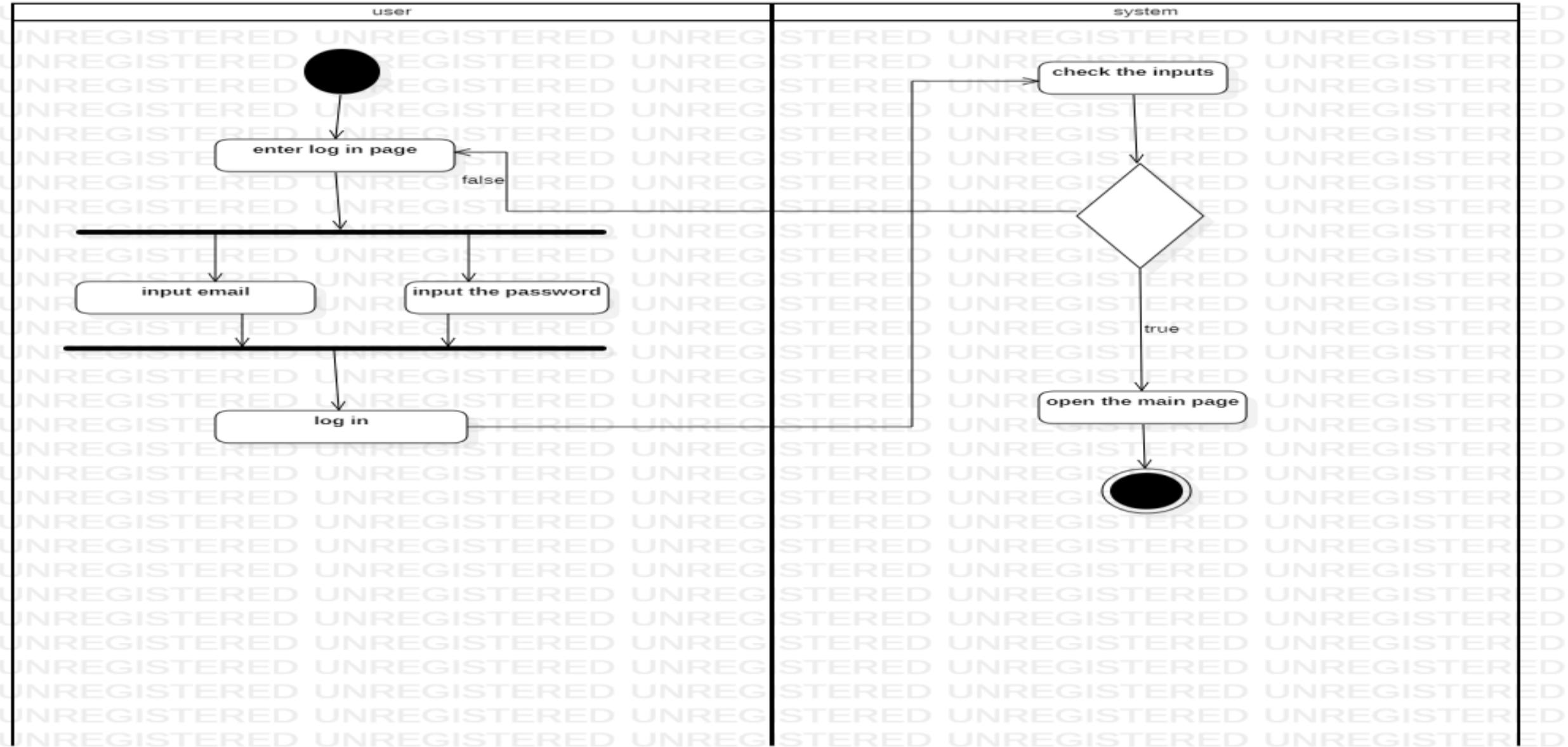
system Sequence diagram

Logout

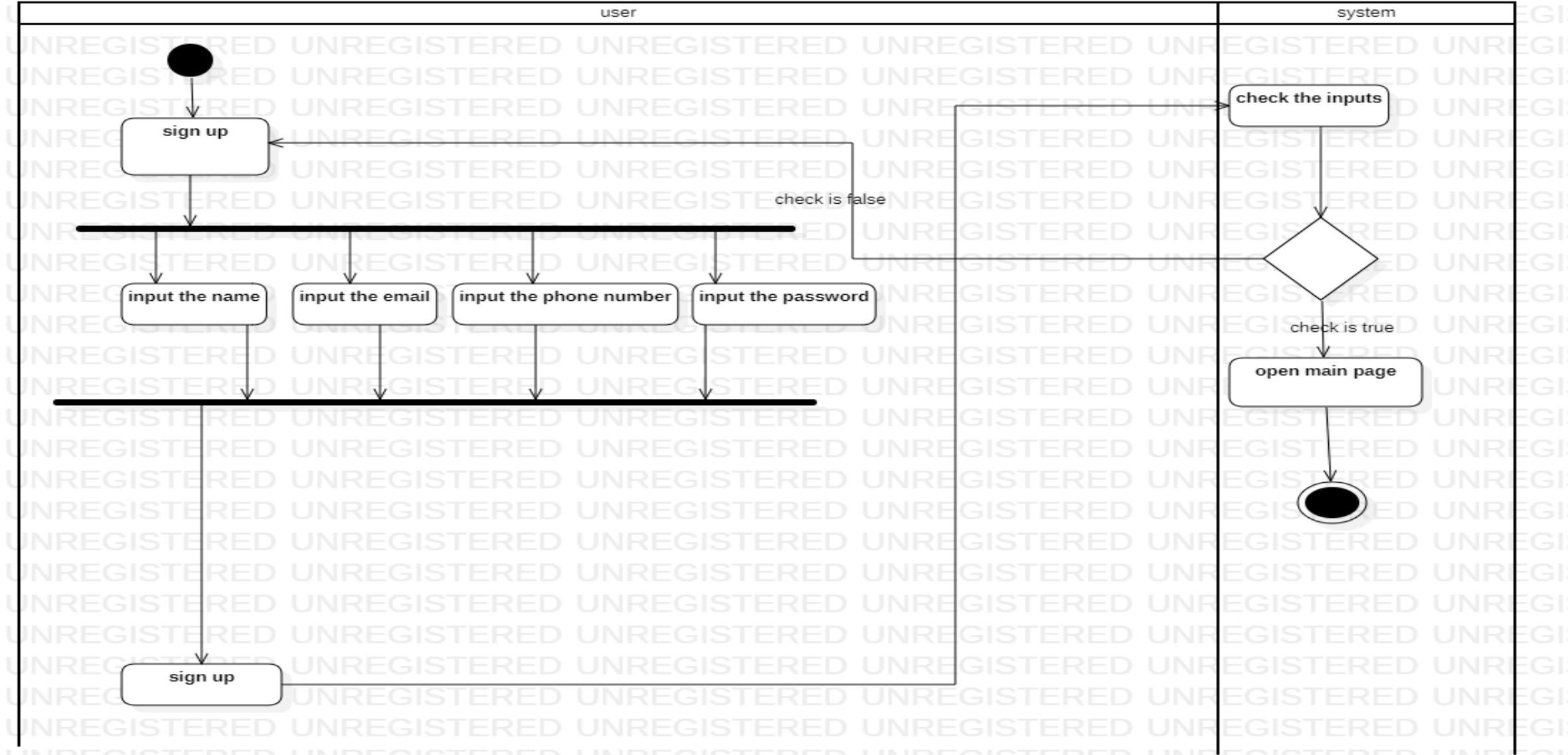


Activity Diagram

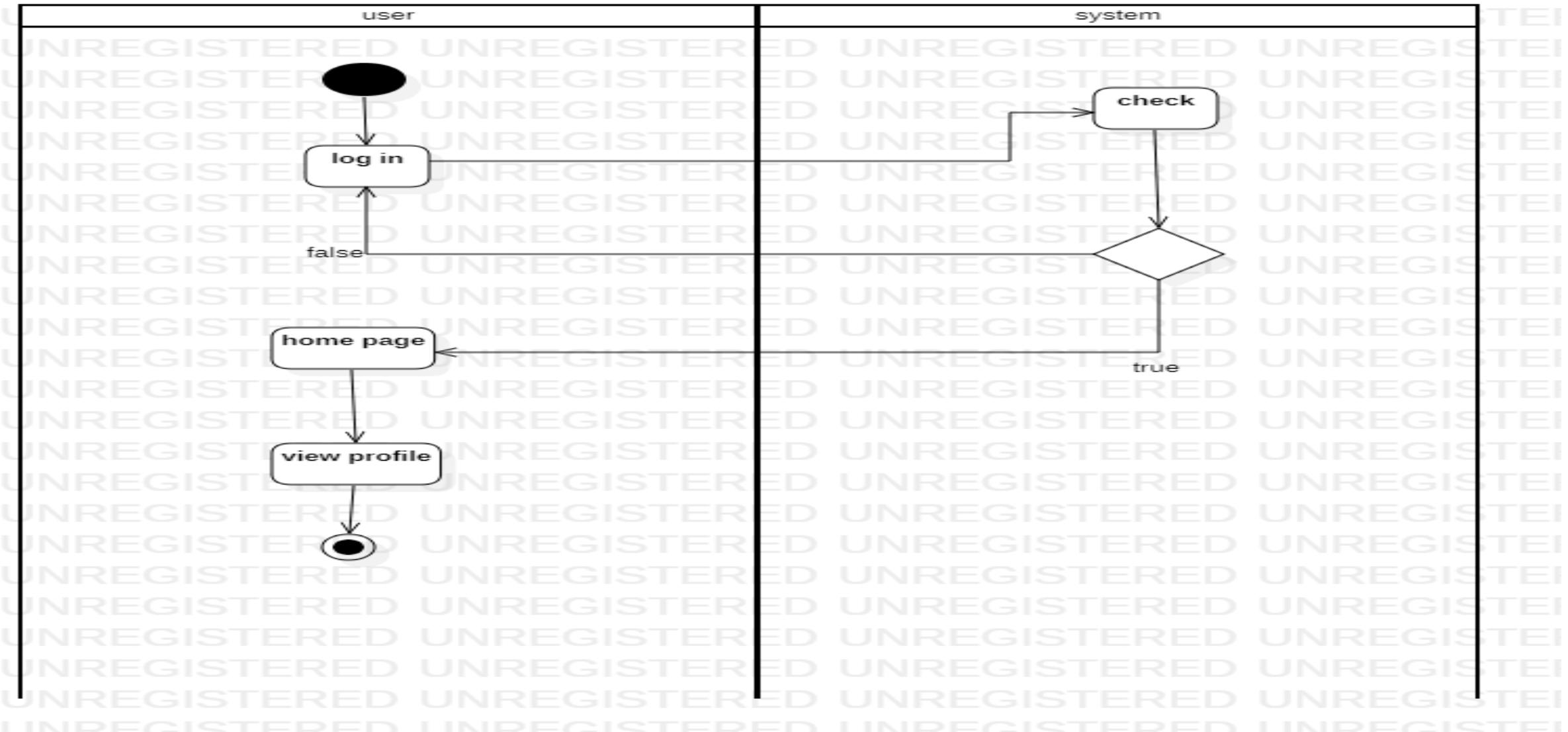
Log in



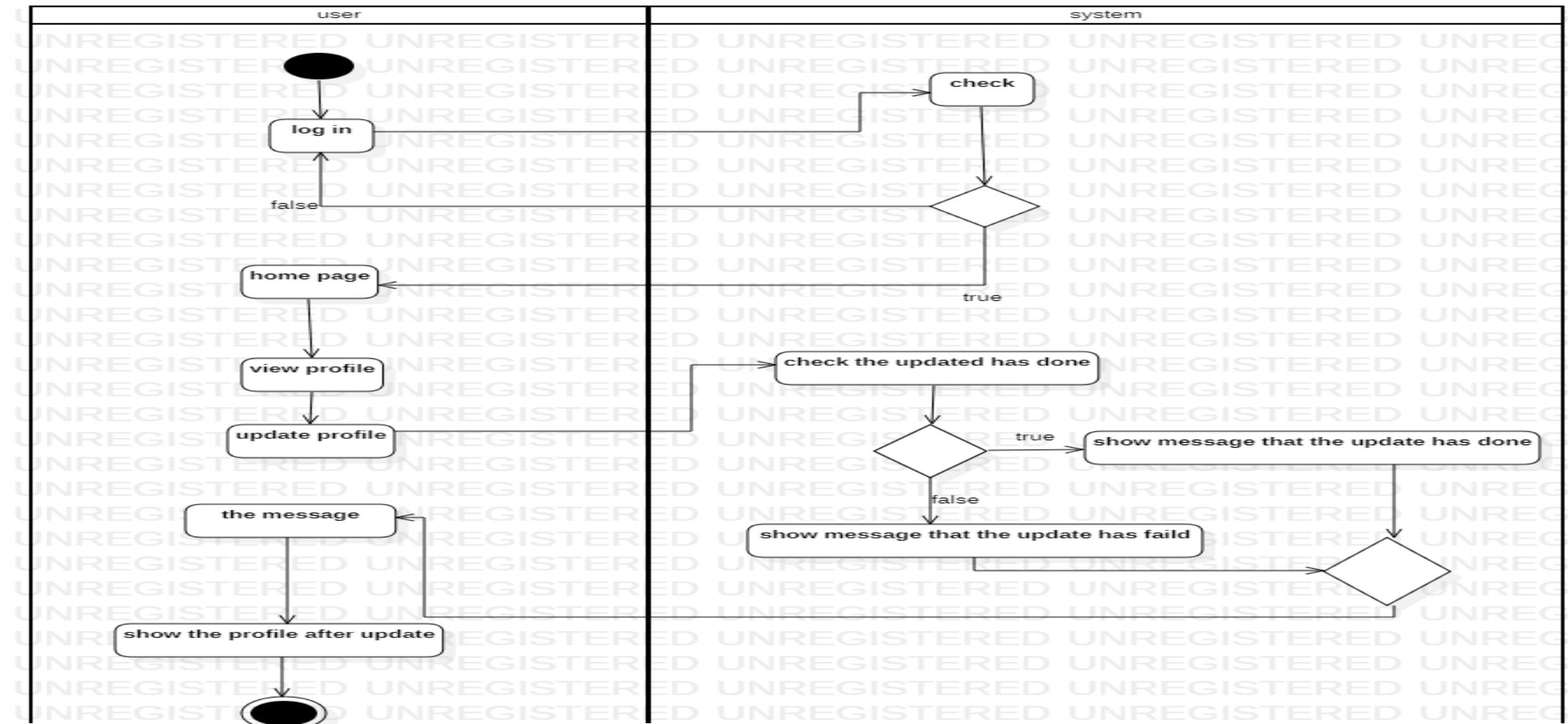
Sign up



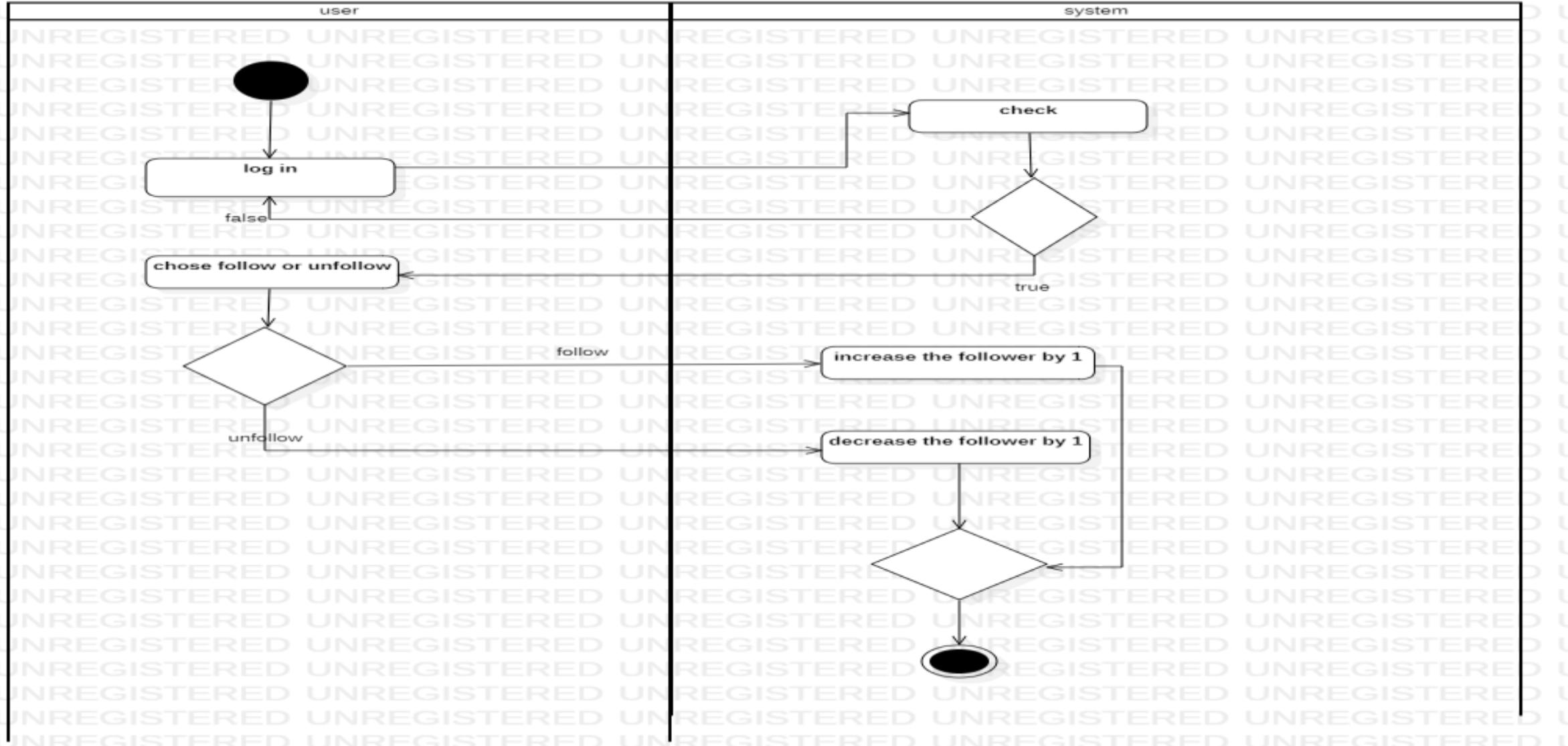
View profile



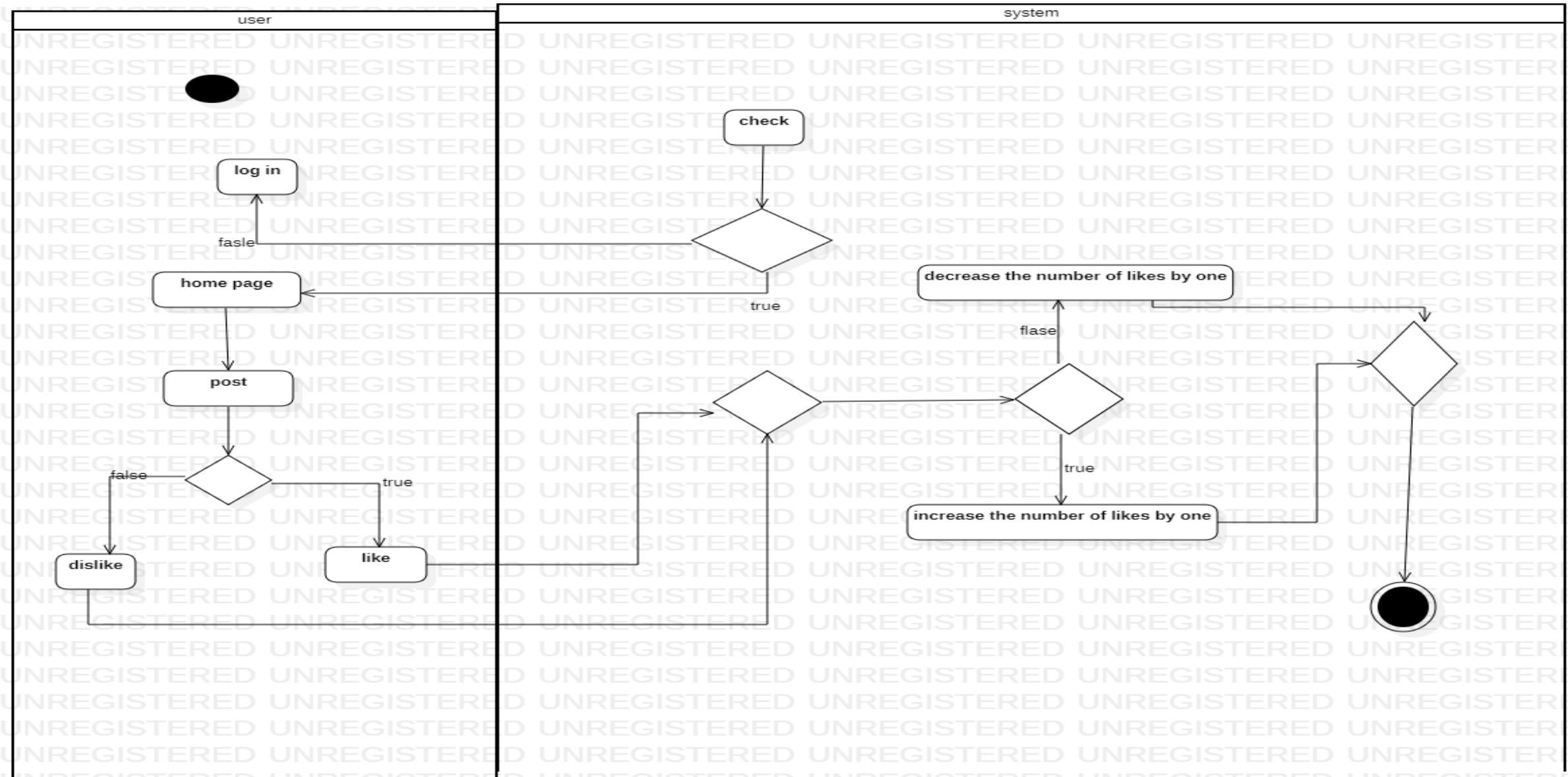
Edit profile



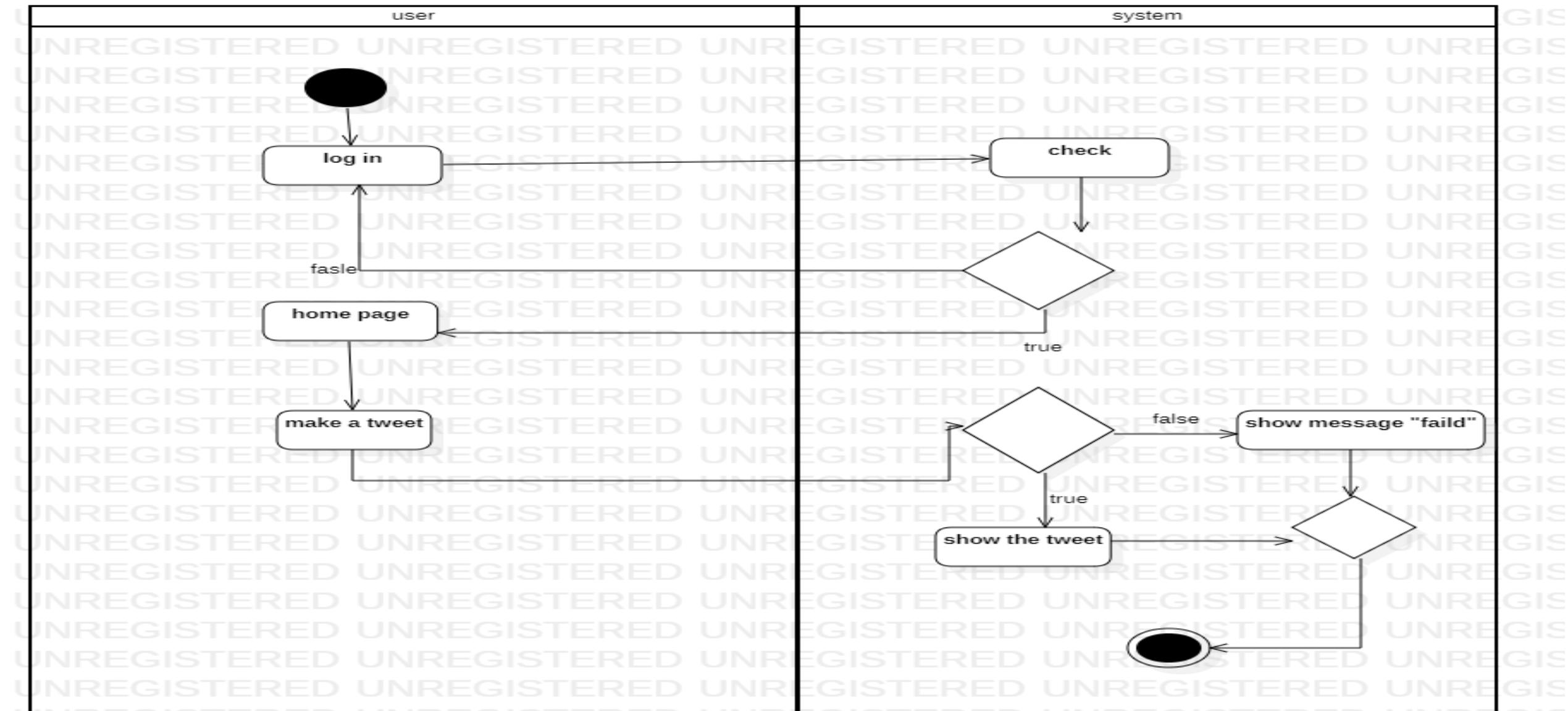
Follow and Unfollow



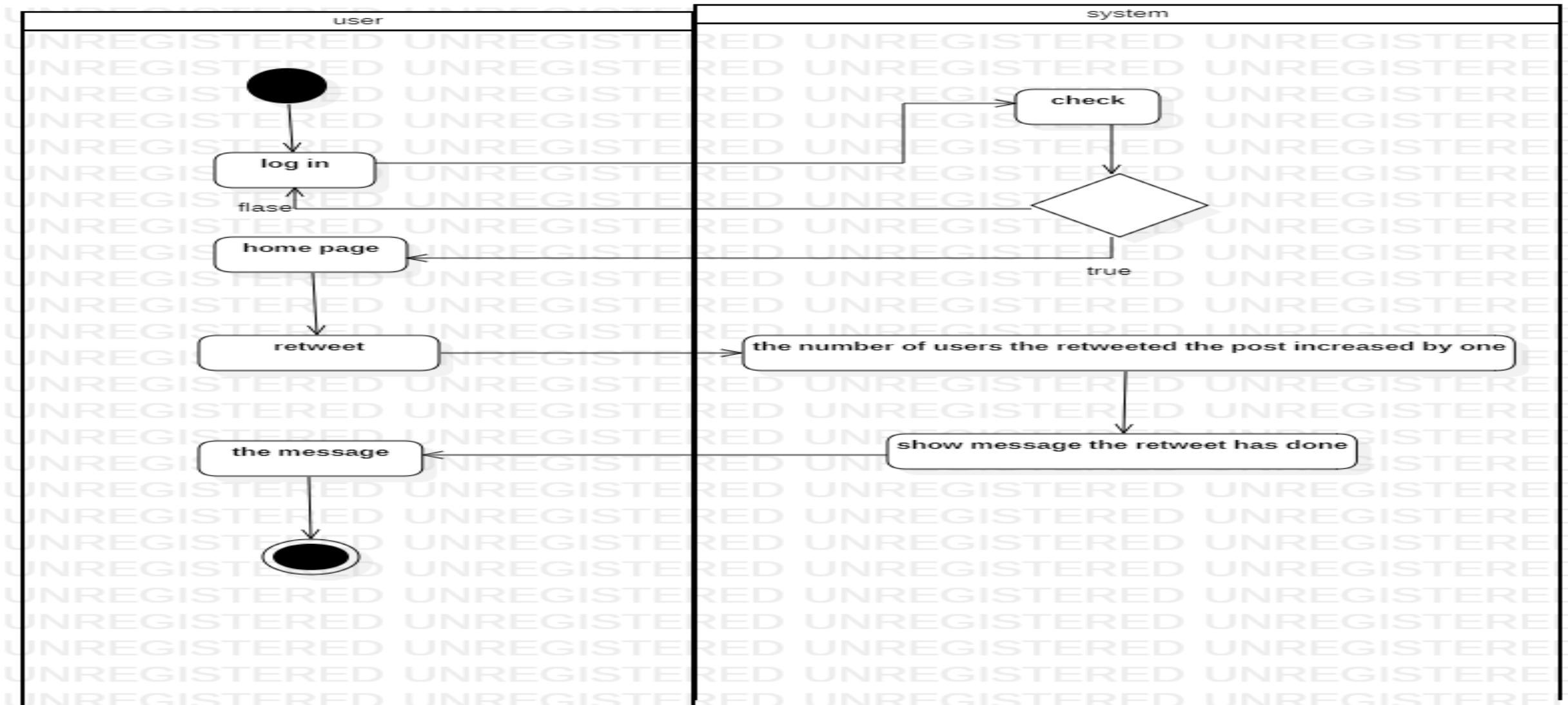
like and dislike



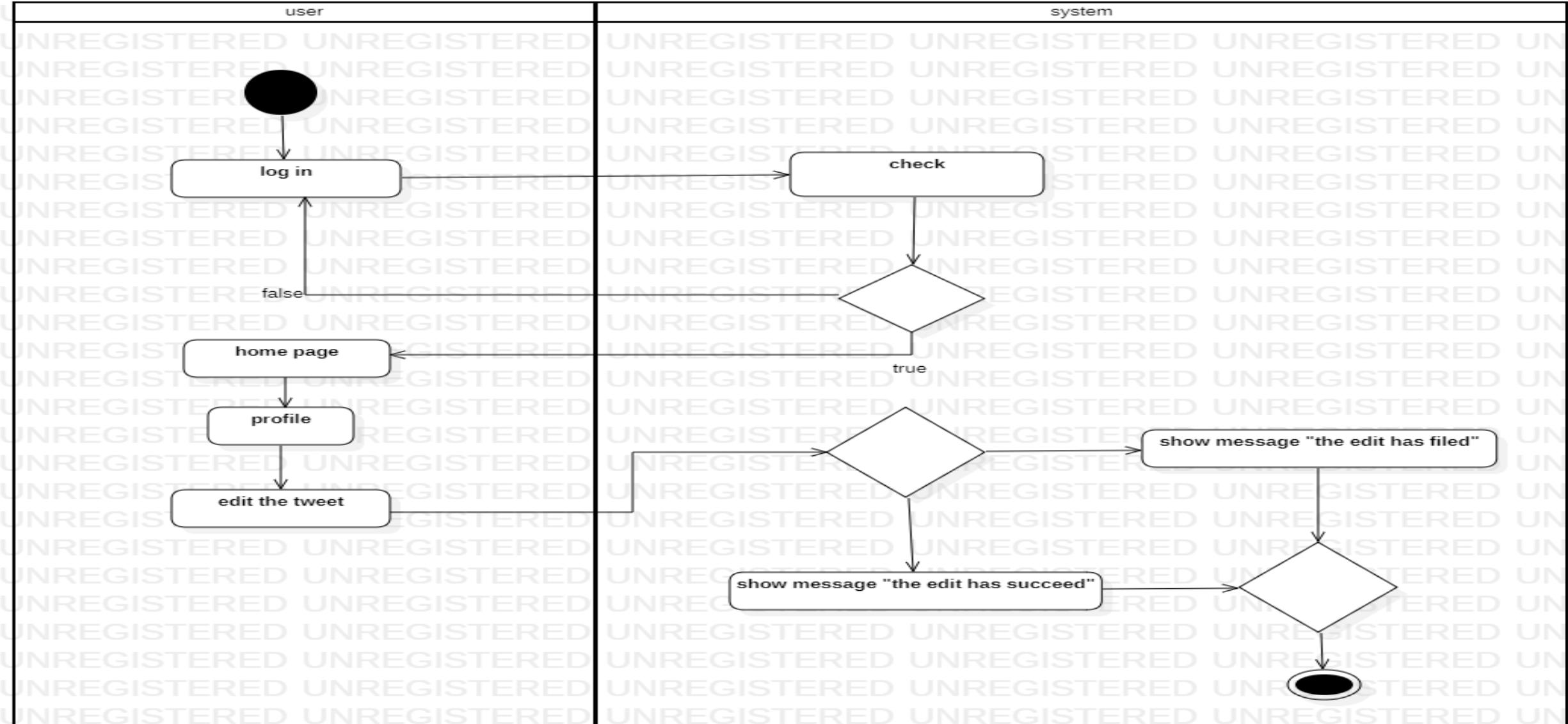
Tweet



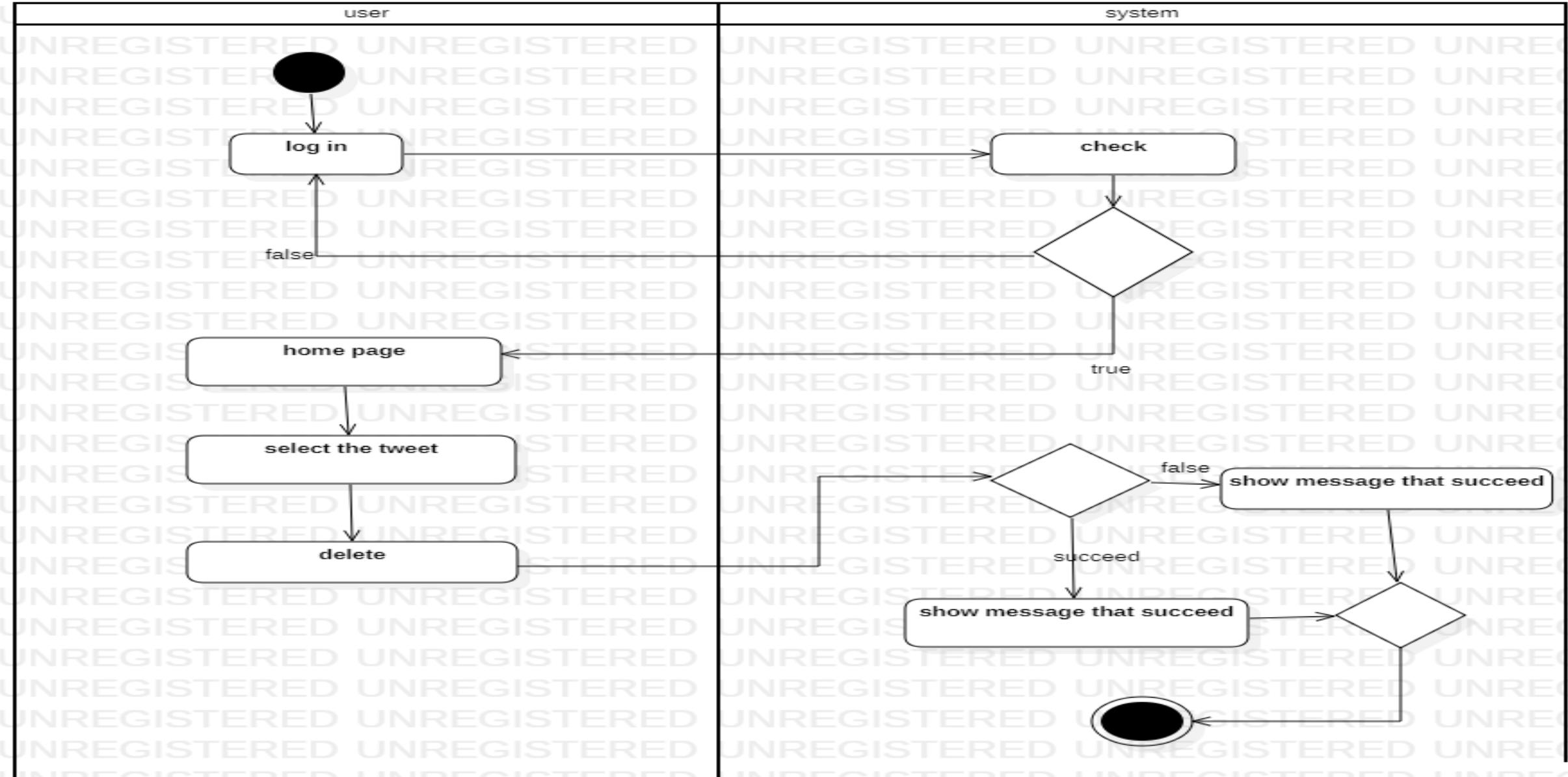
Retweet



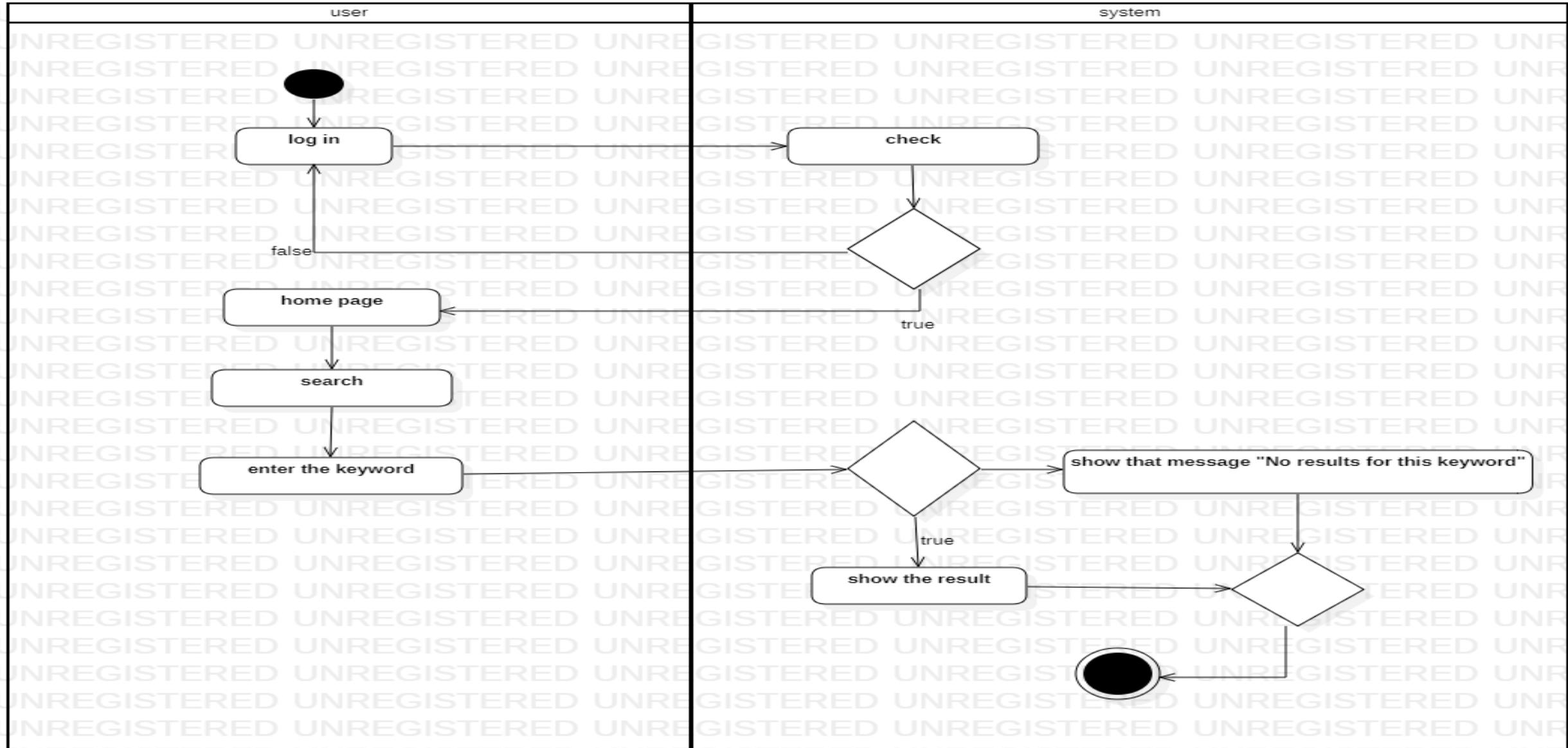
Edit Tweet



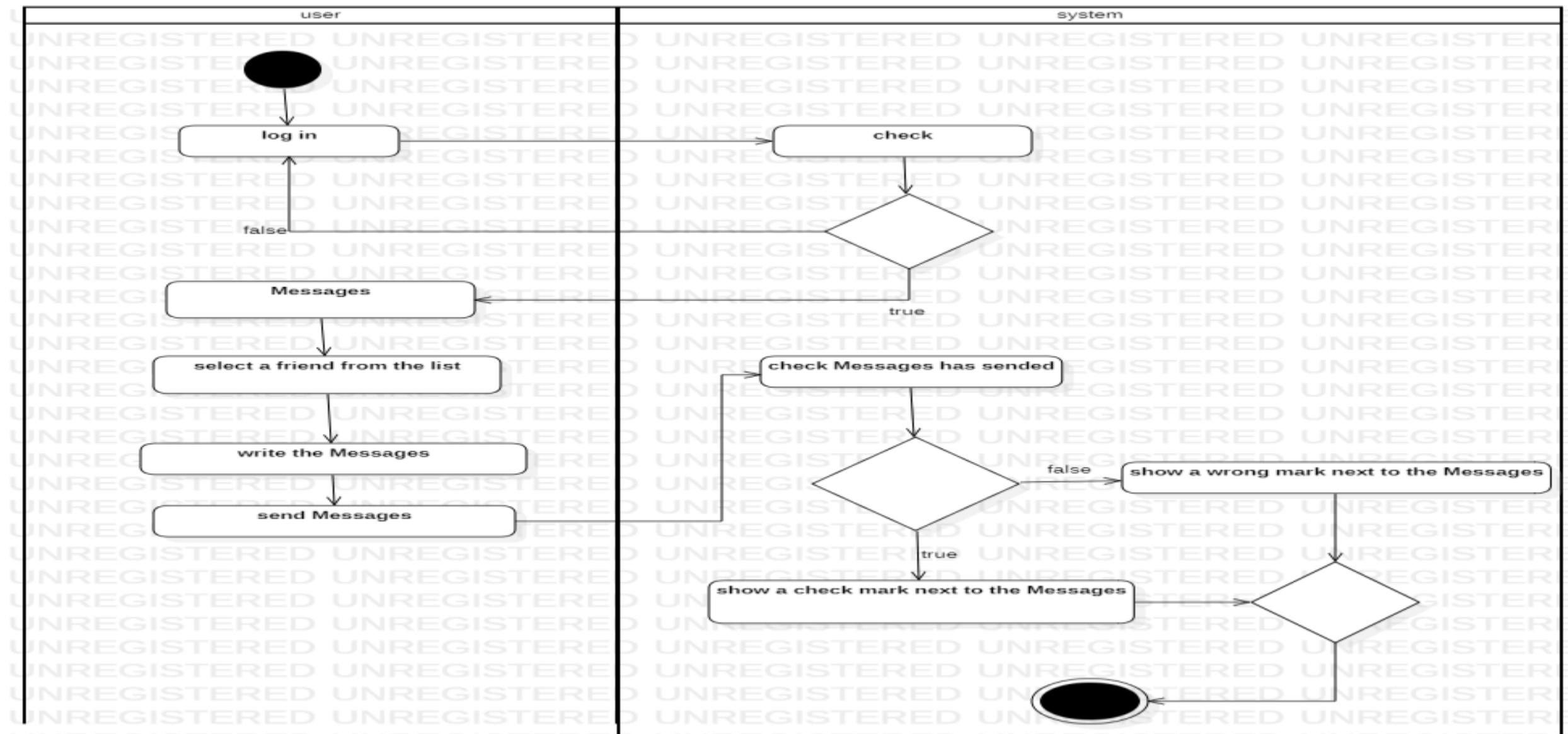
Delete



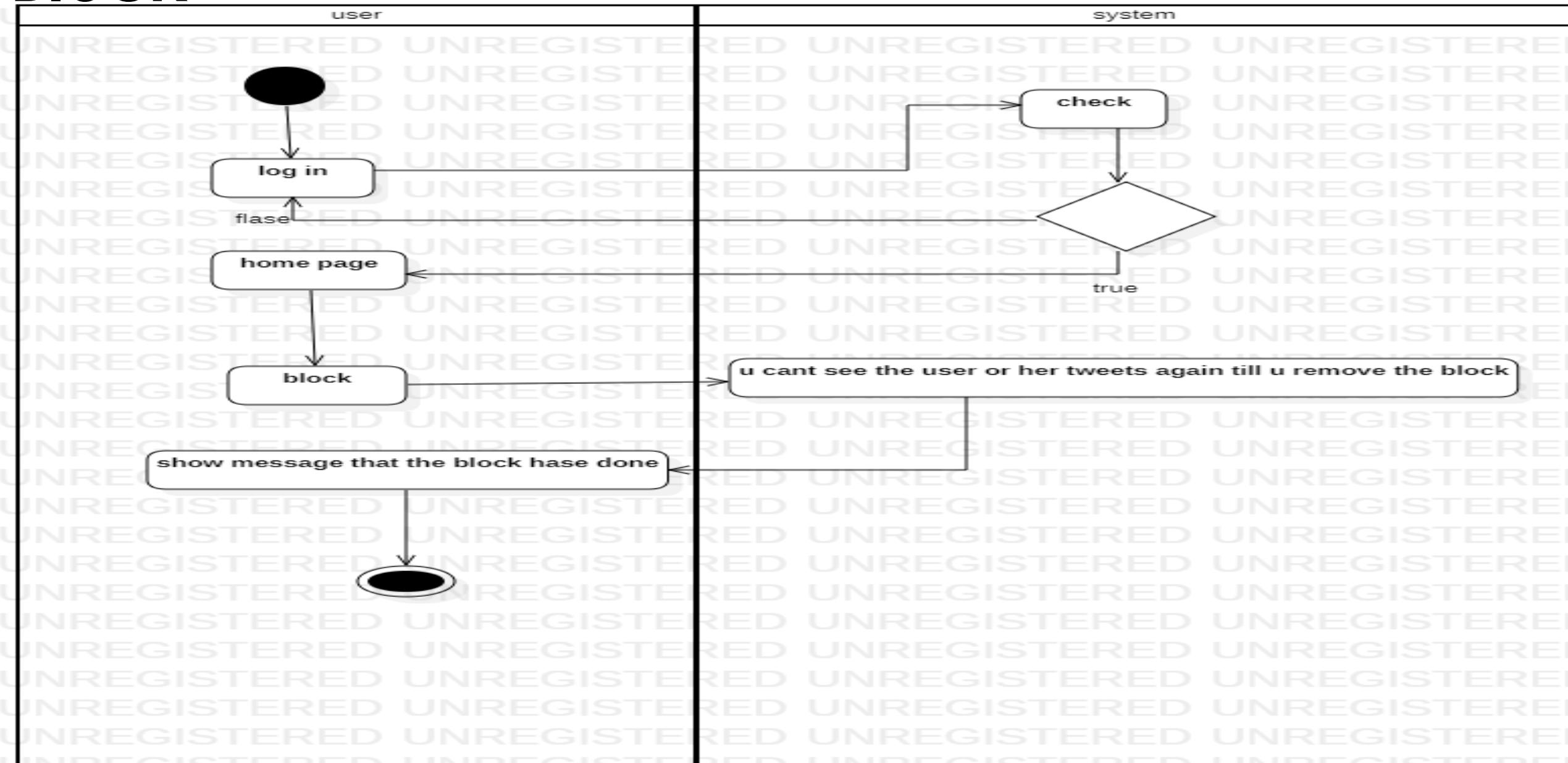
Search



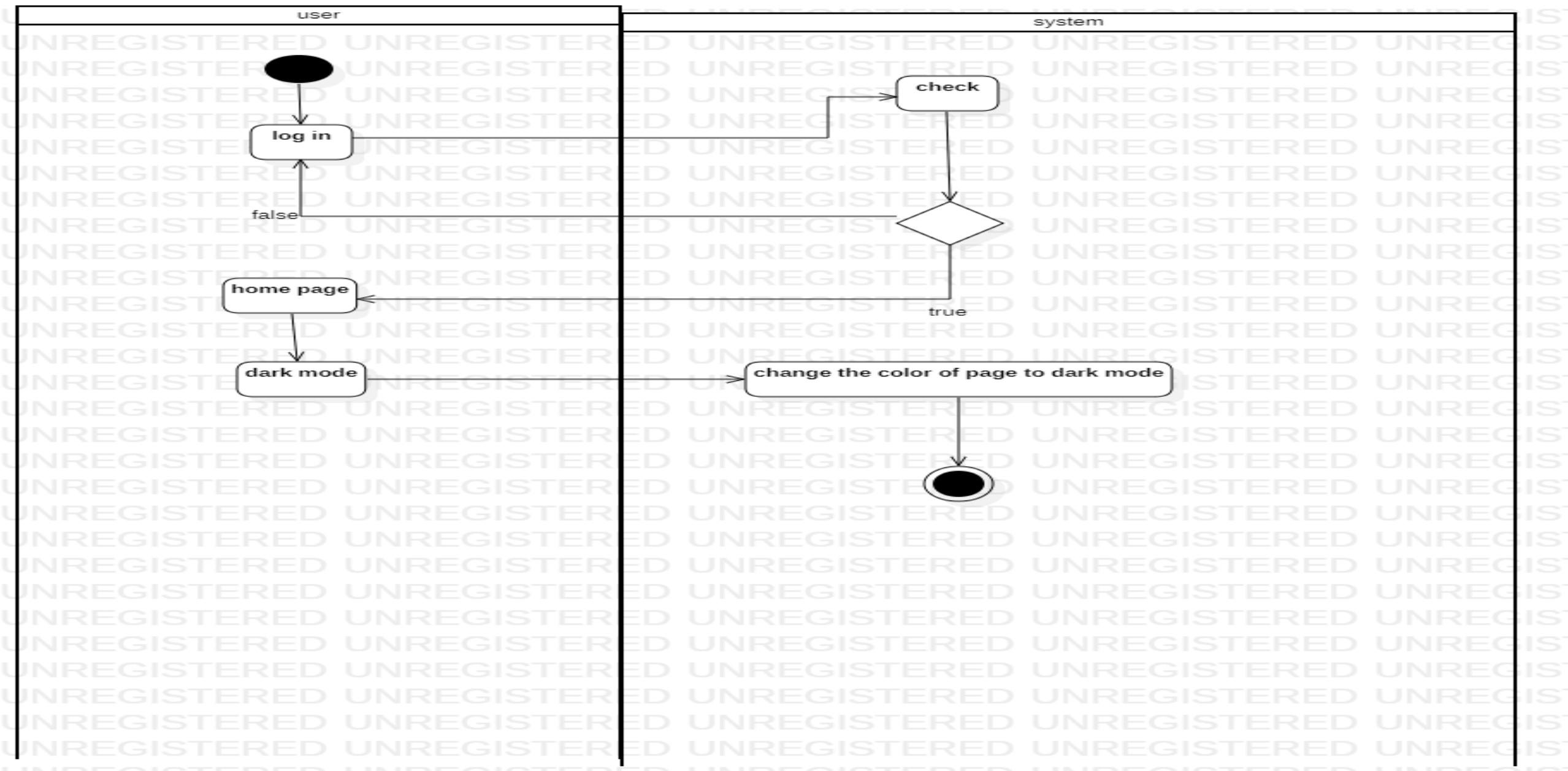
Chat



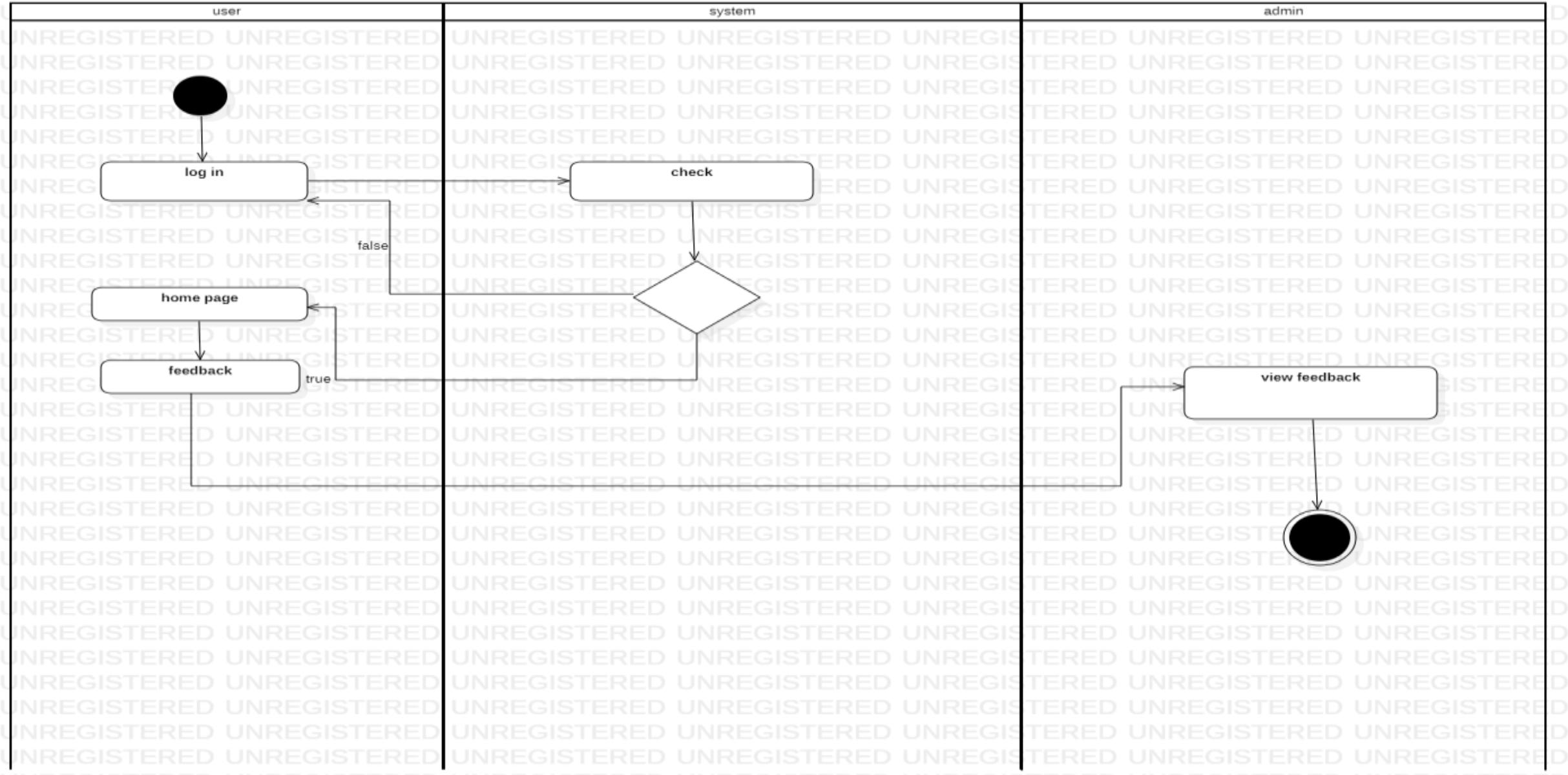
Block



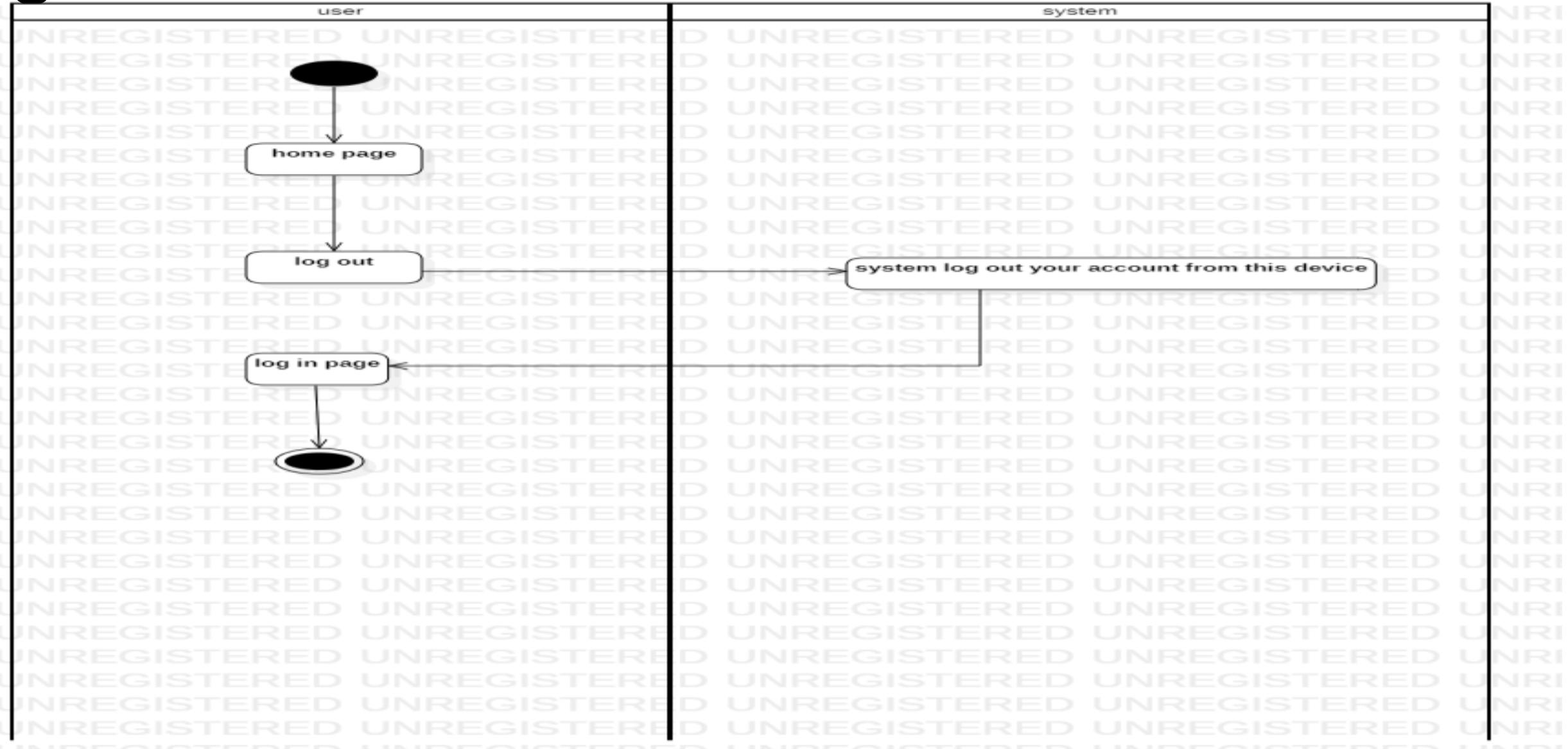
Dark Mode



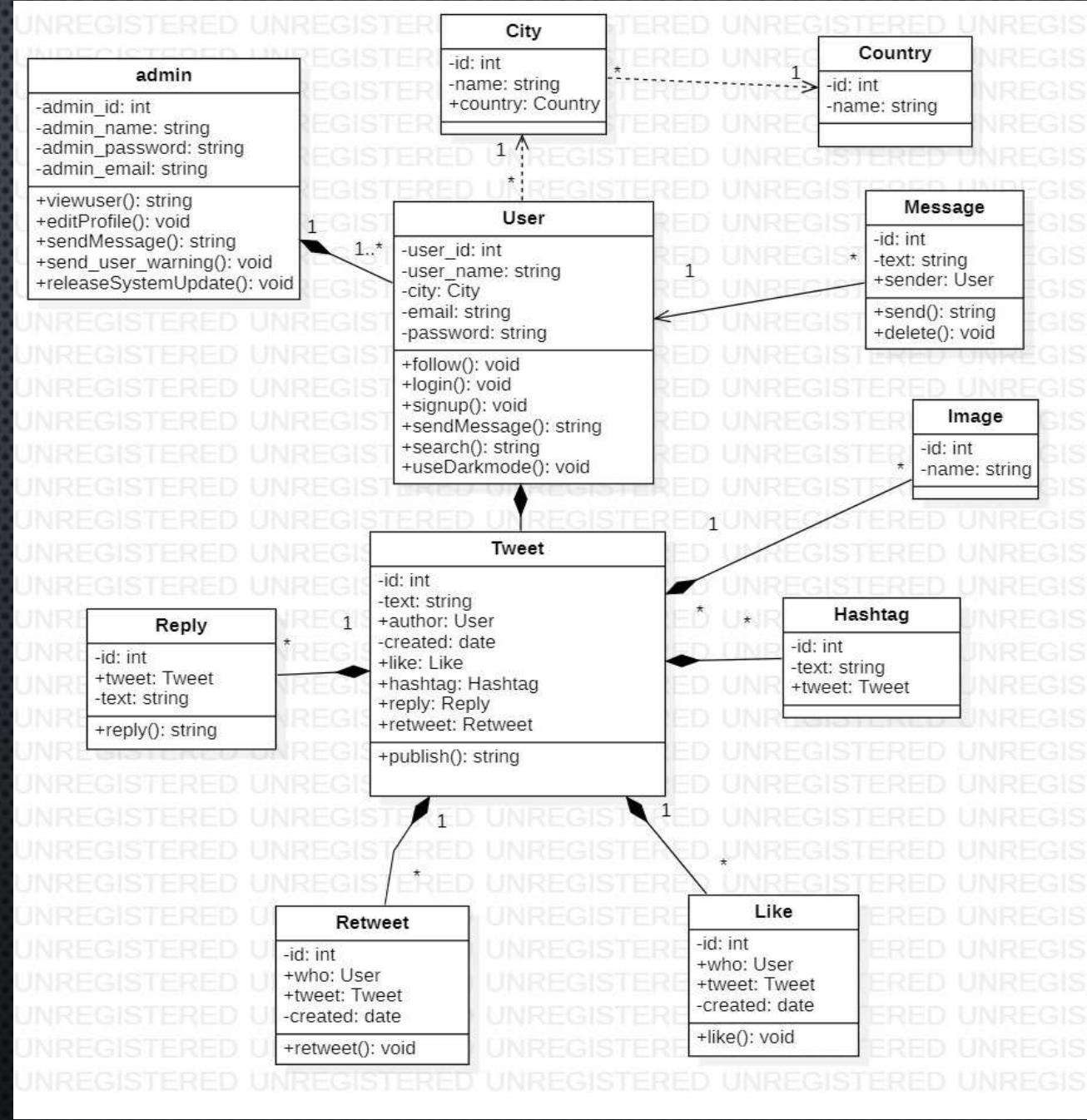
Feed Back



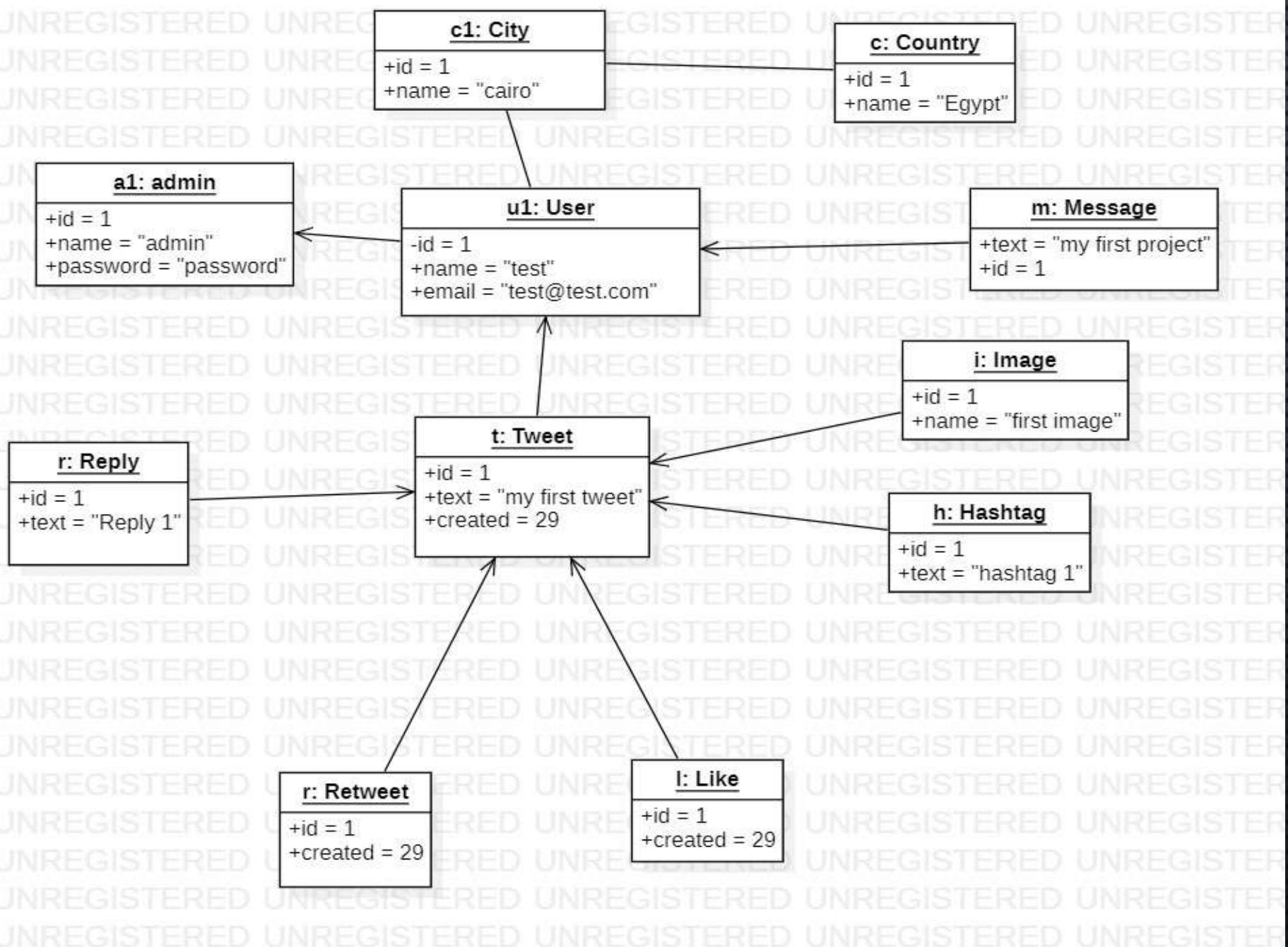
LogOut



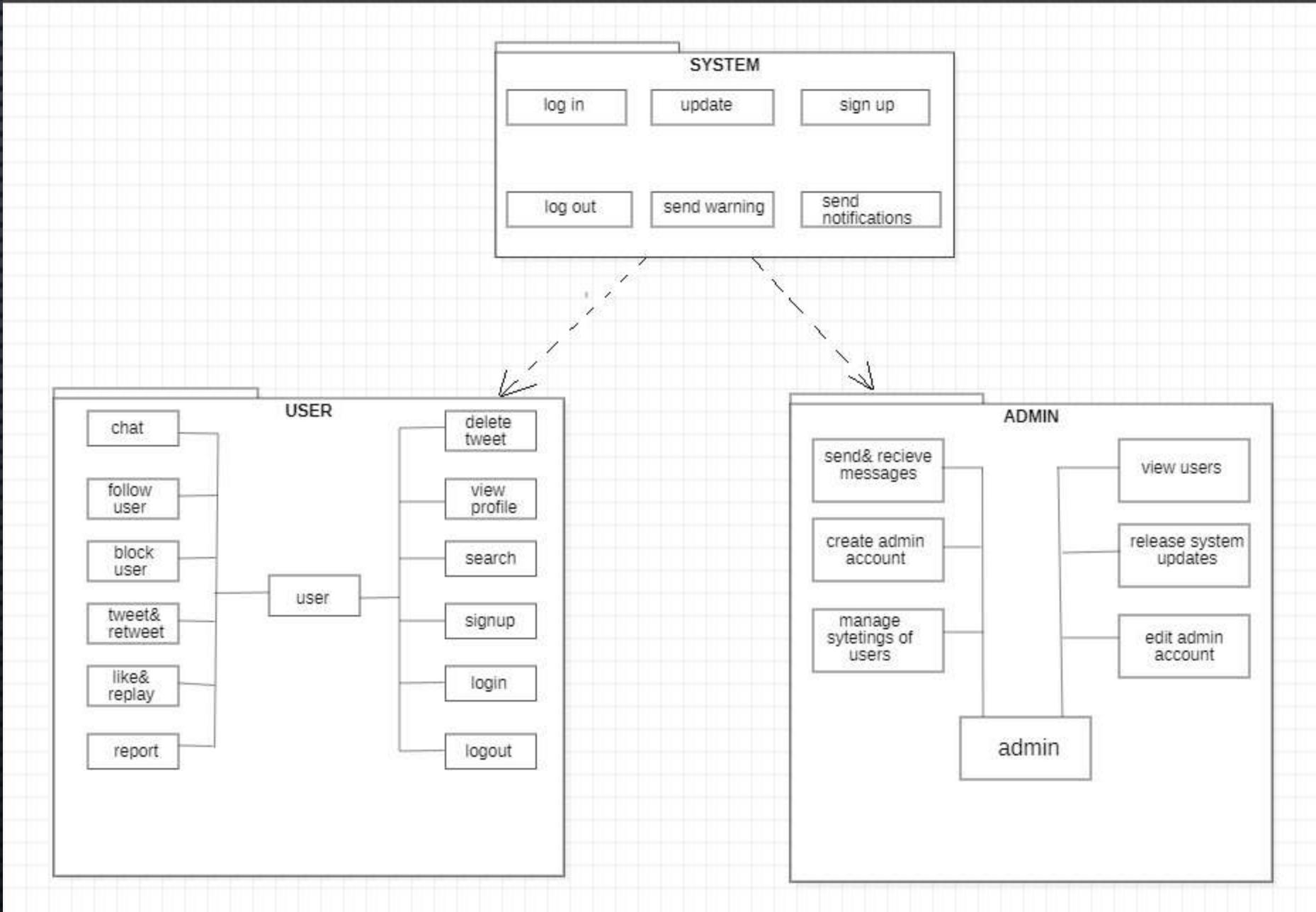
Class Diagram



Object Diagram

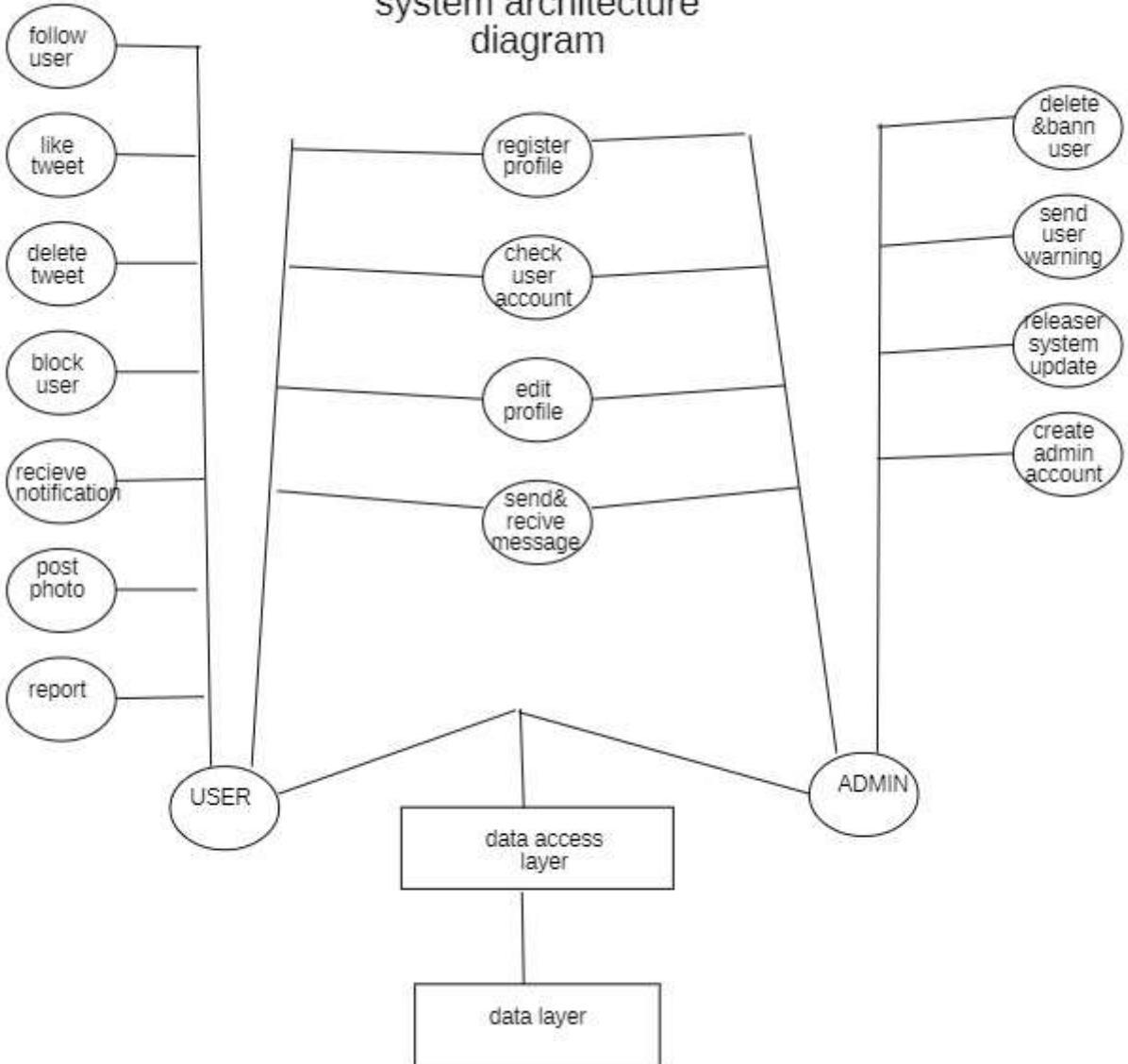


Package Diagram



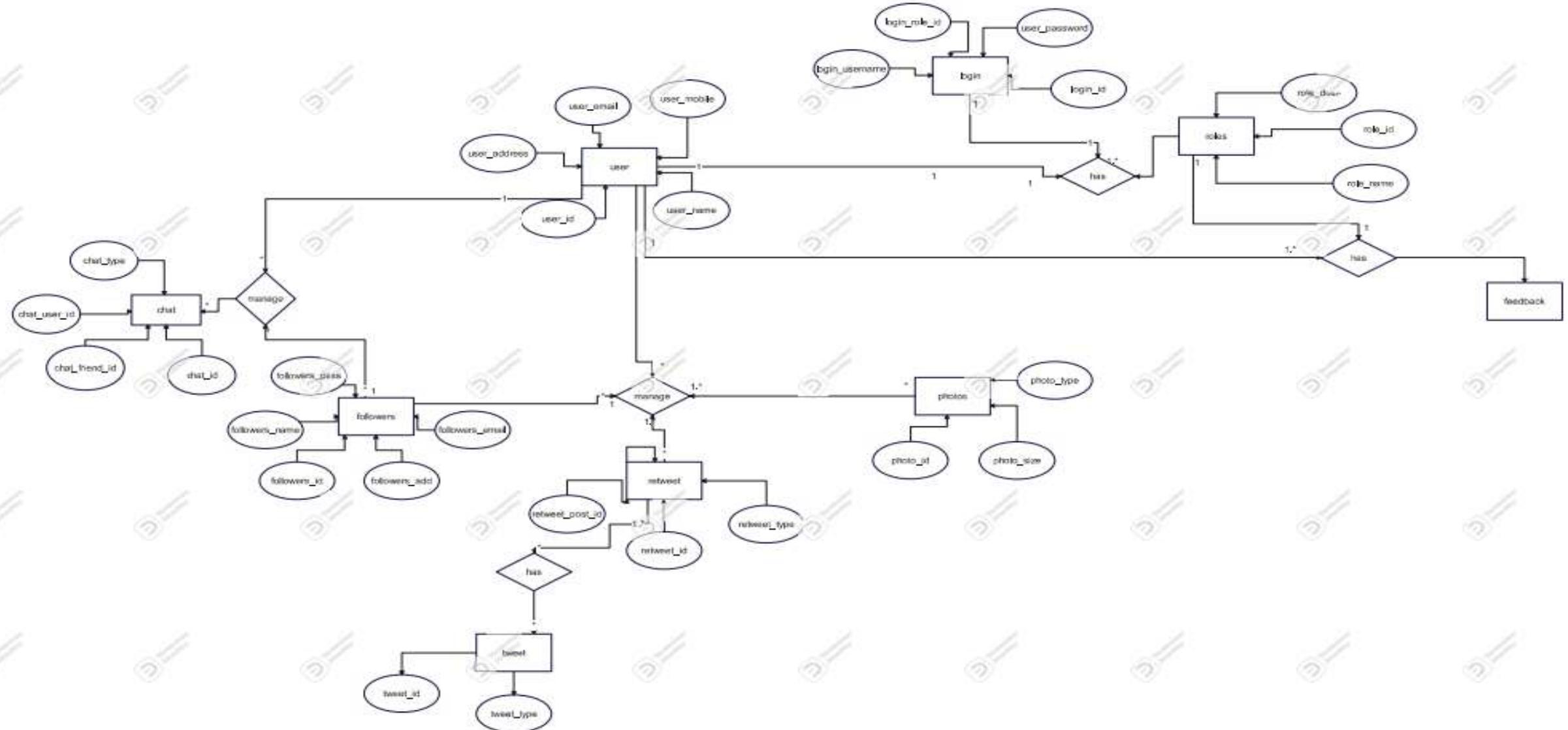
System Architecture Diagram

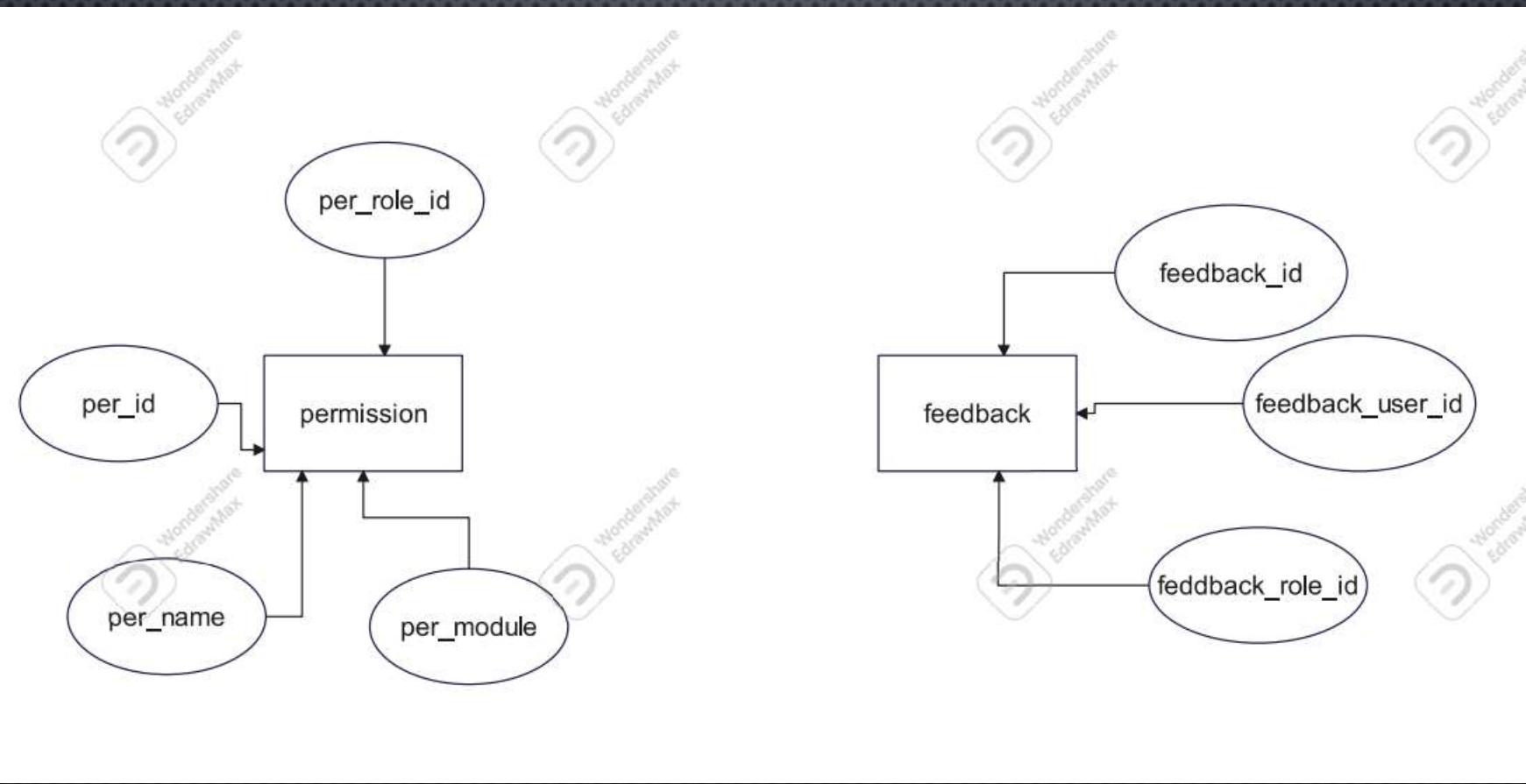
system architecture diagram



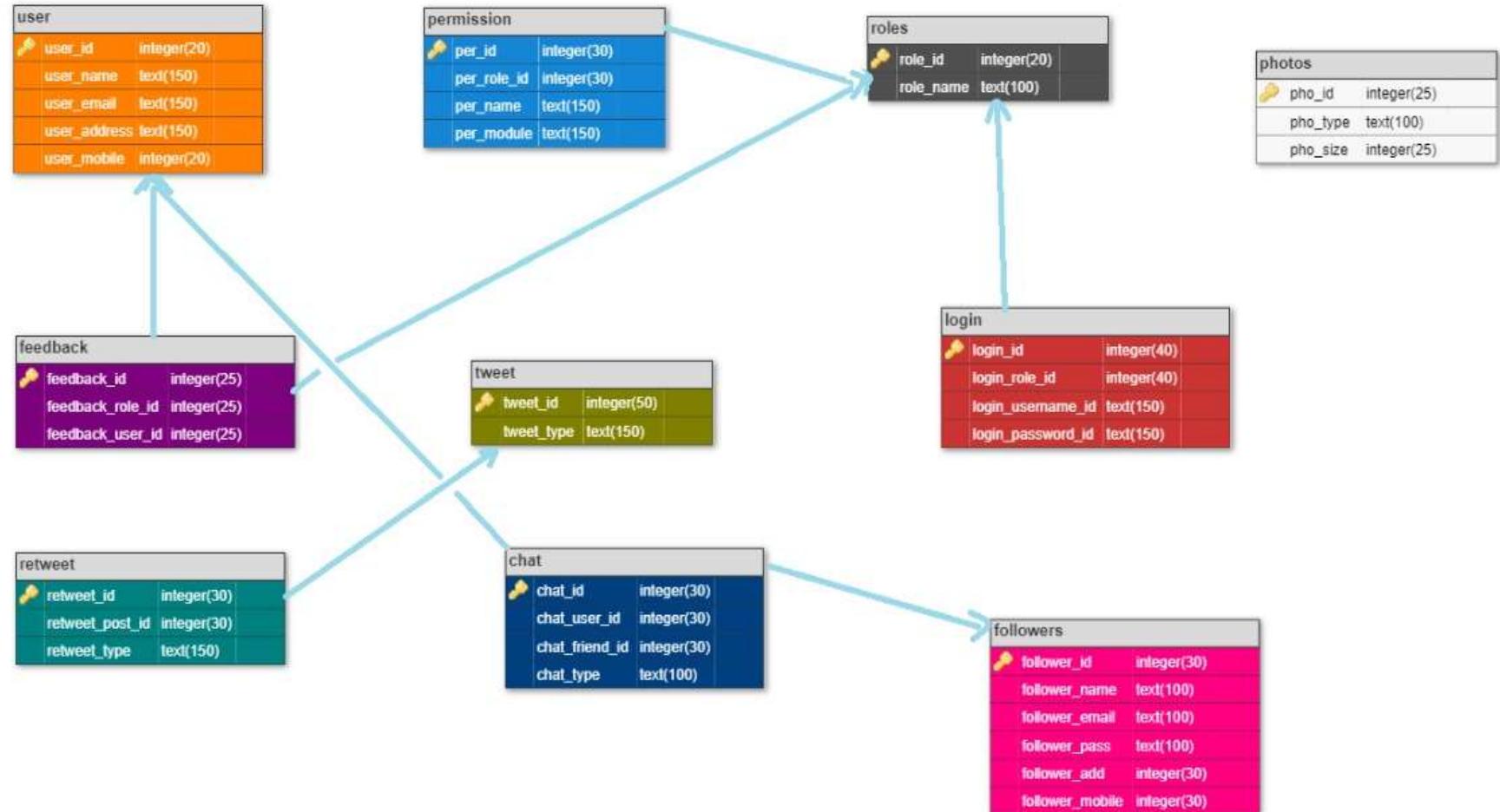
Database Specification

ERD





Tables



Thank you

