

Alumni Management System



PHASE 1 COVER SHEET

Helwan University - Faculty of Computers & Artificial Intelligence – Computer Science Department
Module: CS251 Software Engineering 1 – Spring “Semester 2” 2022-2023



THIS FORM IS FOR BOTH THE GENERAL & MEDICAL INFORMATICS PROGRAMMES

SE-I COURSE PROJECT (PHASE 1 COVER SHEET)

Discussions Scheduled for Week 8 (more details will be announced later).

- o Print 1 copy of this cover sheet and attach it to a printed copy of the documentation (SRS, ... etc.).
You must also submit softcopies of all your documents (as PDFs); details will be announced later.
- o Please write all your names in Arabic.
- o Please make sure that your students' IDs are correct.
- o Handwritten Signatures for the attendance of all team members should be filled in before the discussion.
- o Please attend the discussion on time (announced separately), late teams will lose 5 grades.

Project Name: _____

Team Information (typed not handwritten, except for the attendance signature):

	ID [Ordered by ID]	Full Name [In Arabic]	Attendance [Handwritten Signature]	Final Grade
1	20210424	سما مصطفى فزاع		
2	20210473	شونيز علاء مختار		
3	20210393	سارة اشرف		
4	20210399	سارة محمود عمر		
5	20210388	زينب هاشم		
6	20210387	زينب محمد الحسيني		
7	20210659	فاطمة احمد علي		

Grading Criteria:

PHASE 1 COVER SHEET

Helwan University - Faculty of Computers & Artificial Intelligence – Computer Science Department
Module: CS251 Software Engineering 1 – Spring “Semester 2” 2022-2023



10 Items		Grade	Notes
1. Functional Requirements	1		
2. Non-Functional Requirements	1		
3. Use-Case Diagram(s) <i>including general use-cases for the system, and the detailed use-cases description</i>	2		
4. System Architecture – <i>including applied Architectural Pattern(s)</i>	1		
5. Activity Diagrams	2		
6. Database Specification (ERD, Tables)	2		
7. Class Diagram (Interfaces, Classes, Relations) – An initial version based on the requirements and Use-Case/Activity diagrams.	2		
8. Object Diagrams (Including object diagrams that illustrate the preconditions and the post-conditions of selected functions)	1		
9. Package Diagram(s)	1		
10. Sequence Diagrams <i>including System Sequence Diagrams (SSDs)</i>	2		

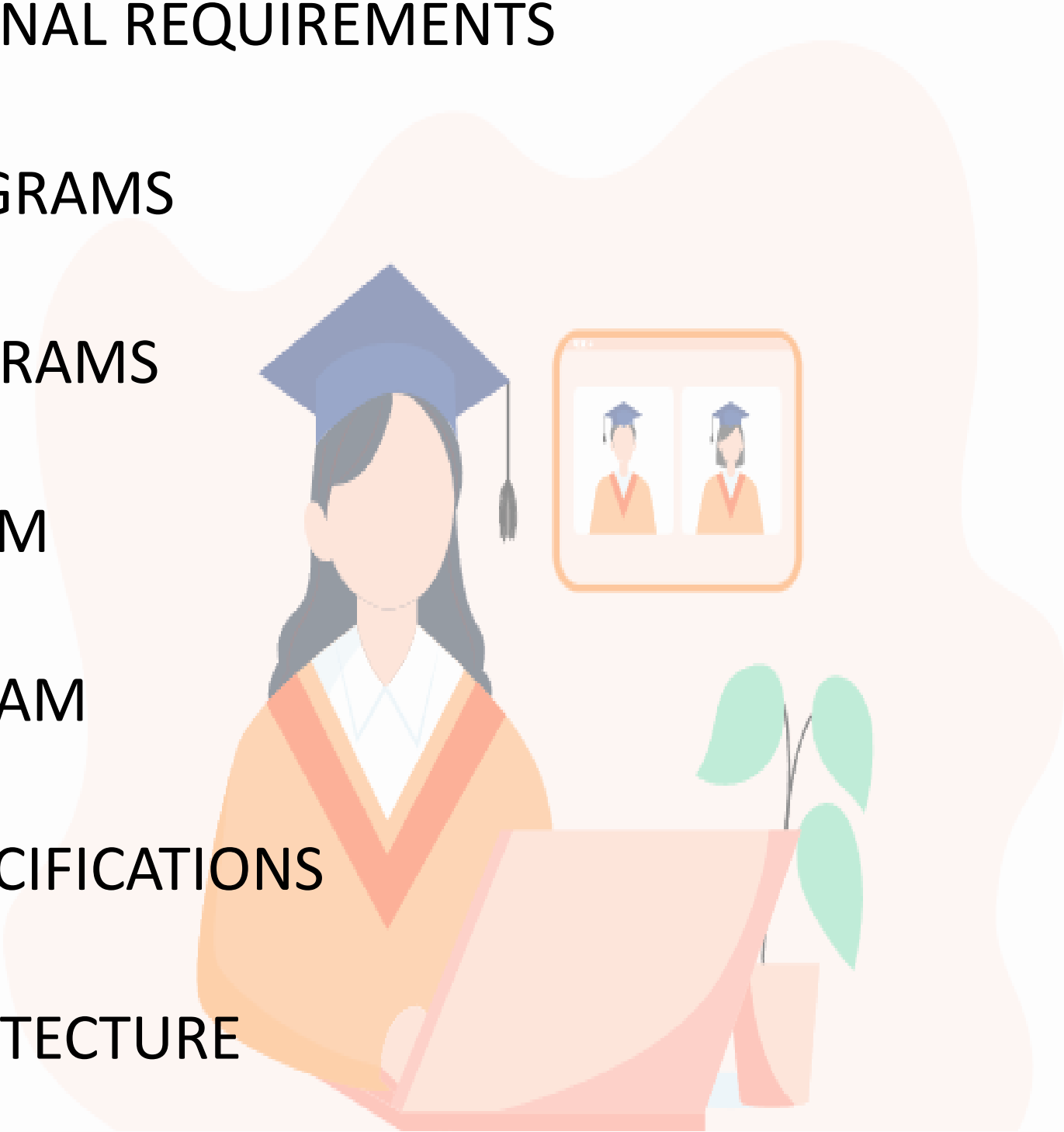
Teaching-Assistant's Signature: _____

15

TABLE OF CONTENTS

Phase 1

• INTRODUCTION	5
• FUNCTIONAL REQUIREMENTS	6
• NON- FUNCTIONAL REQUIREMENTS	12
• USE CASE DIAGRAMS	14
• ACTIVITY DIAGRAMS	39
• CLASS DIAGRAM	46
• OBJECT DIAGRAM	47
• DATABASE SPECIFICATIONS	49
• SYSTEM ARCHITECTURE	52
• PACKEGE DIAGRAM	53
• SEQUENCE DIAGRAMS	54



INTRODUCTION

PURPOSE

The purpose of the software Alumni management system is to make our college communicate and stay in touch with its alumni. Here the software gives you a platform where alumni can share their thoughts, ideas, contact and keep the interaction active.

PRODUCT PERSPECTIVE

Alumni management system is a website which has an easy and user-friendly interface that will help our college to retain long-term relationships with Alumni and providing a professional network for its Alumni.

PRODUCT SCOPE

Alumni management system has great features of building and maintaining an updated Alumni database every year, planning reunion events, involving our Alumni in the student development as a mentor and raising funds from Alumni. Developments and upgrades were to be done according to that.

OPERATING ENVIRONMENT

Alumni management system is a website that can run on any web browser. It can also run on any kind of operating system.

STACKHOLDERS

Developers : responsible for designing the software according to the Admin's request and developing it upon request.

Admin : Use website to manage Alumni's data and communicate with them.

Alumni: Use website to communicate with college/other alumni.

FUNCTIONAL REQUIREMENTS

OVERVIEW: The software has two modules: Admin and Alumni.

1-As an Admin I can:

- Login
- keep/update/delete Alumni's list
- Post about events /job opportunities /scholarships
- Manage a mentorship program
- Manage Alumni's posts
- View and interact with Alumni's posts
- Manage donations
- Contact Alumni
- View Alumni's feedback
- Logout

2-As an Alumni I can:

- Sign up
- Login
- Manage my account info
- Post about events /job opportunities /scholarships
- Post a question or topic to discuss
- Join as mentor
- Search for other Alumni
- View and interact with other Alumni's posts
- Donate
- Contact Admin
- Give feedback
- Logout



FUNCTIONAL REQUIREMENTS

1) Sign up

Description: Alumni can register an account if he isn't registered before and his national id in the data base.

1.1-> Alumni can sign up using national id, email and password.

1.2-> Verification:

1.2.1->if the verification done well, the system will create an account and transfer the alumni to the home page.

1.2.2->if the verification failed, the system will ask alumni to reenter right date.

2) Log in

Description: Alumni or admin can log in by entering email and password and the alumni shall be registered before.

2.1-> Alumni / Admin can log in using email and password or using Facebook account.

2.2-> Verification:

2.2.1->if the verification done well, the system will transfer the alumni / admin to the home page.

2.2.2->if the verification failed, the system will ask alumni to reenter right date.

3) Manage Alumni's list

Description: Admin can store/update/delete Alumni or Alumni's info in the database.

3.1-> Admin enter Alumni's list page

3.2->Admin add/update/delete Alumni record or Alumni's info and all changes is done in the database.

FUNCTIONAL REQUIREMENTS

4) Post events/jobs opportunities/scholarships/topic/question

Description: Alumni/Admin can make post about events, job opportunities or scholarships.

4.1-> Alumni/Admin click icon make post

4.2->Alumni/Admin write the post and provide links, add photos/videos or tag other users.

4.3->Alumni/Admin click button publish post.

5) Post question or topic to discuss

Description: Alumni ask question or write a topic to discuss

5.1-> Alumni click icon make post and write the post.

5.2-> revealing identities

5.2.1->if Alumni click icon allow to be anonymous, then his account name in the post will change to “participant”

5.2.2->if Alumni doesn't click icon allow to be anonymous, then his account name in the post will be as it is.

5.3->Alumni click button publish post.

6) Manage Alumni's posts

Description: in order to let the alumni's post be viewed by others, Admin must accept this post and publish it.

6.1-> Admin enter manage alumni's posts page and read the post.

6.2-> Following posting rules:

6.2.1->if the post is following the posting rules, Admin will accept the the post and publish it to be viewed by others.

6.2.1.1->System will send message to the alumni that his/her post is published successfully in the posts page.

6.2.2-> if the post is not following the posting rules, Admin will delete the post.

6.2.2.1->System will send message to the alumni that his/her post is violating the posting rules.

FUNCTIONAL REQUIREMENTS

7) View/Interact with others

Description: Alumni can view other users' posts and interact with it by adding react or commenting, participating, sharing post, saving it for later.

7.1-> Alumni enter posts page and view posts.

7.2-> Alumni can add react, write a comment, share post, save post to see later, participate if the post is about events or scholarship or job opportunity.

8) Manage account

Description: Alumni can manage his account by updating his profile photo, job, username, communication ways and...

8.1-> Alumni enter his/her account.

8.2-> Alumni click button manage account.

8.3-> Alumni update profile photo, job title, communication ways, delete post/comment/react.

9) Donate

Description: Alumni can donate choosing a payment method.

9.1-> Alumni click button donate.

9.2-> Alumni enter amount of money to be donated.

9.3-> Alumni choose a payment method: PayPal, credit card or debit card.

9.4-> Alumni click button process donation.

10) Manage donations

Description: Admin can view who donated and send a confirmation mail to the donor.

10.1-> Admin enter donations page.

10.2-> Admin view who donated.

10.3-> Admin write to the donor a confirmation mail that he/she donated successfully and to thank him/her.

FUNCTIONAL REQUIREMENTS

11) Join as mentor

Description: Alumni can become a mentor.

11.1-> Alumni enter mentor form page.

11.2-> Alumni fill the form by choosing the topic, suitable day and time.

11.3->Alumni click button send form.

12) Manage mentorship program

Description: Admin can view who filled the form and send to the mentor a confirmation mail and contact details.

12.1-> Admin enter mentorship program page.

12.2-> Admin view who filled the form

12.3-> Admin send confirmation mail to the mentor including contact details and thanking him/her.

13) Give feedback

Description: Alumni can send feedback to admin without revealing his/her identity.

13.1-> Alumni click icon feedback.

13.2-> Alumni write his/her feedback.

13.3->Alumni click send feedback.

14) View feedback

Description: Admin can view the feedback that was sent by alumni.

14.1-> Admin enter collected feedback page.

14.2-> Admin view the feedback.

FUNCTIONAL REQUIREMENTS

15) Contact

Description: Alumni can contact admin and wait for admin to reply.

15.1-> Alumni enter contact page.

15.2-> Alumni fill the contact form by writing the message and entering an email that admin will reply to.

15.3-> Alumni click send button.

16) Search

Description: Alumni can search for other alumni by entering a keyword and he/she can select the graduate year.

16.1-> Alumni click icon search.

16.2-> Alumni write a keyword and select a graduate year.

16.3-> Searching process:

16.3.1-> if the keyword is in the database, then the system will show him/ her the results.

16.3.2-> if the keyword isn't in the database, then the system will show him/her a message "sorry, there's no result."

17) Logout

Description: Alumni/Admin can logout if they're logged in.

17.1-> Alumni/Admin click button logout.

NON-FUNCTIONAL REQUIREMENTS

1) Look and feel REQs (Interface):

- The product shall use colors palette.
- The product should use a lot of animations.
- The product shall use trendy icons.
- The product shall be approachable, so users don't hesitate to use it.
- the product shall be authoritative, so users feel they can rely on it.

2) Usability & Humanity REQs:

- The product shall be easy to use and not be a complex software.
- The product shall be easy to search the information.
- The product shall be a -user friendly as The product shall be easy to use on the first attempt by a member of the public without training .

3) Performance requirements:

- The product shall be fast, and the processes shall not take a lot of times to execute.
- The web page load time shouldn't exceed 3 second.
- The product shall handle up to 15 users simultaneously.
- The product shall, on average, operate without failure for 30 days.
- the program shall do the tasks required from it accurately without editing or deleting, so that sharing or dealing with data shall be accurate.

4) Maintainability & support REQs:

- The product shall be able to be modified to cope with a new class of user.
- Any bugs or problem in executions shall be supported and fixed.

NON-FUNCTIONAL REQUIREMENTS

5) Operational & Environment REQs:

- If the user is using website, then all operation will be used.

6) Cultural REQs:

- The language used in the interface should be formal and polite..

7) Legal REQs:

- The product shall reserve copyrights .
- The product shall comply with our ISO 9001 certification.

8) Design Constraints:

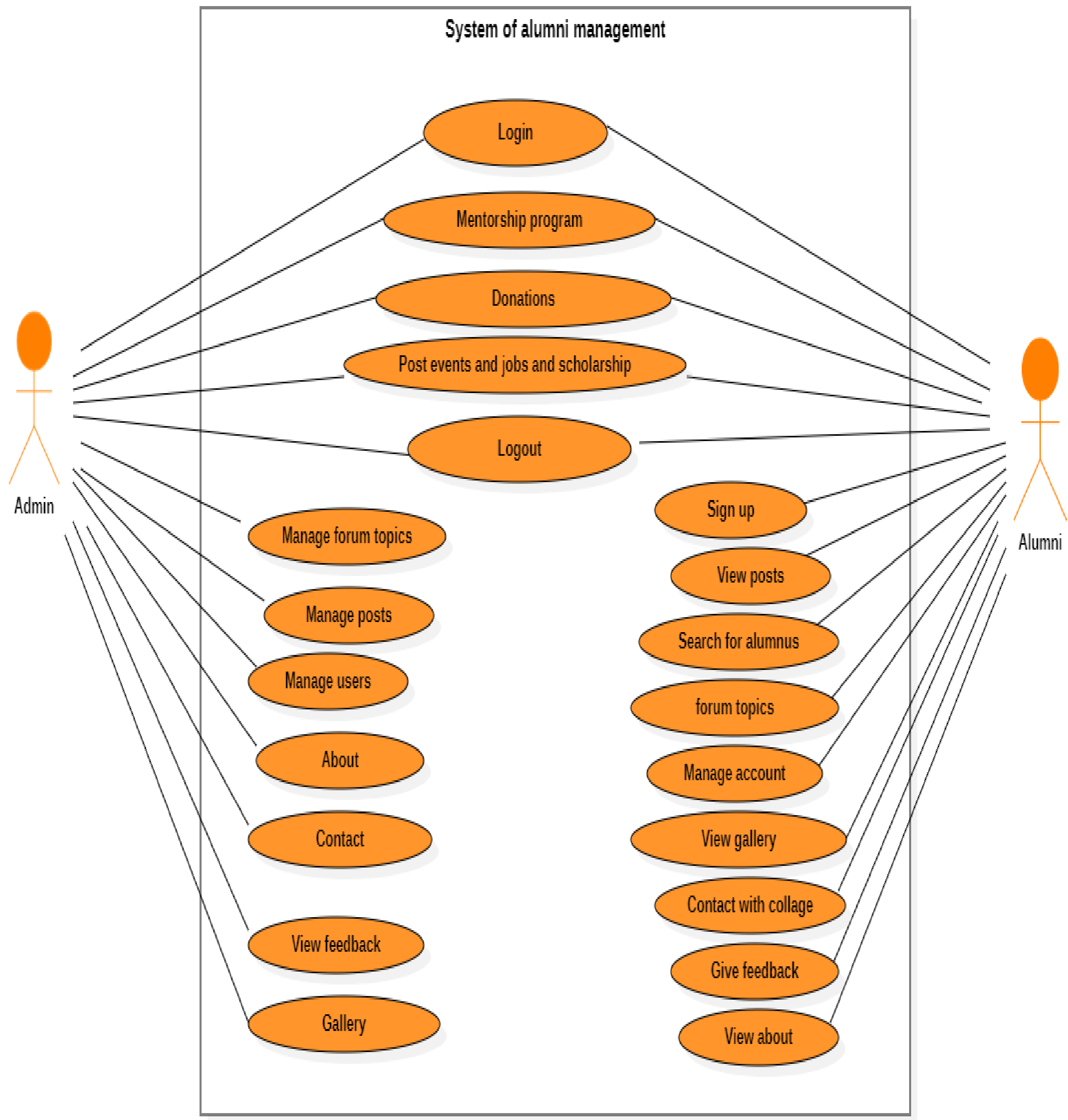
- Password must be at least 8 characters including at least one special character, uppercase letter and a number.

9) Security REQs:

- The product shall ensure that only authorized users are able to gain access.
- The product shall distinguish between authorized and non-authorized users.
- Access: uninterrupted/continual access to data & functionality by authorized users.
- Privacy: protection of data from unauthorized access/disclosure.
- Integrity: prevention of unauthorized modification/deletion of data (data consistency).
- Immunity: protection against threats and attacks.
- Use of database to maintain the data as well as actions that must be done in order to prevent data leak.
- The requirements regarding security, such as data leak or privacy issues, such as disclosing the details of a person to some other person.
- Identity authentication requirements such as while creating an account.

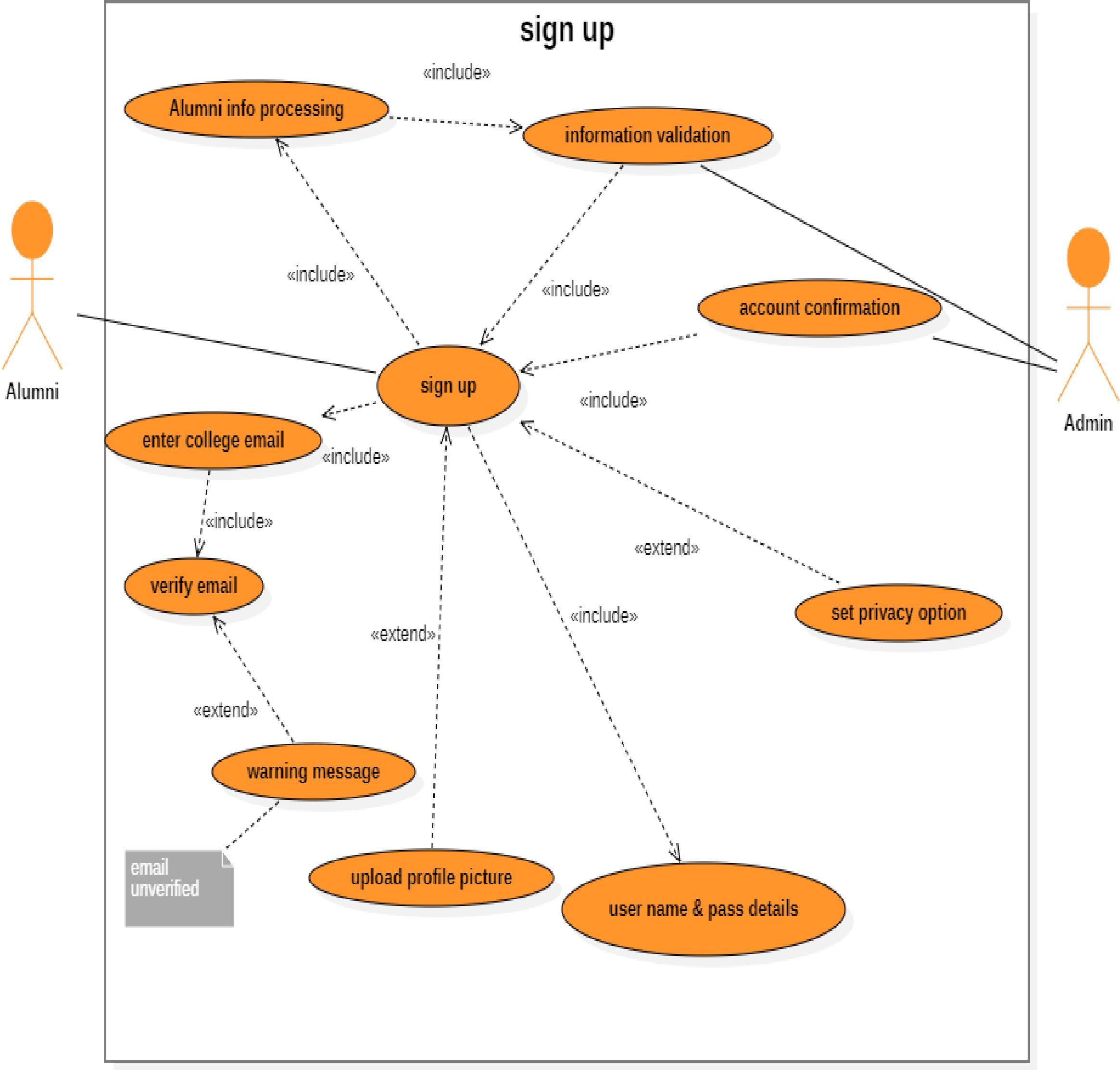
USECASE DIAGRAM

1) System of alumni management :



USECASE DIAGRAM

1.1) Sign Up:

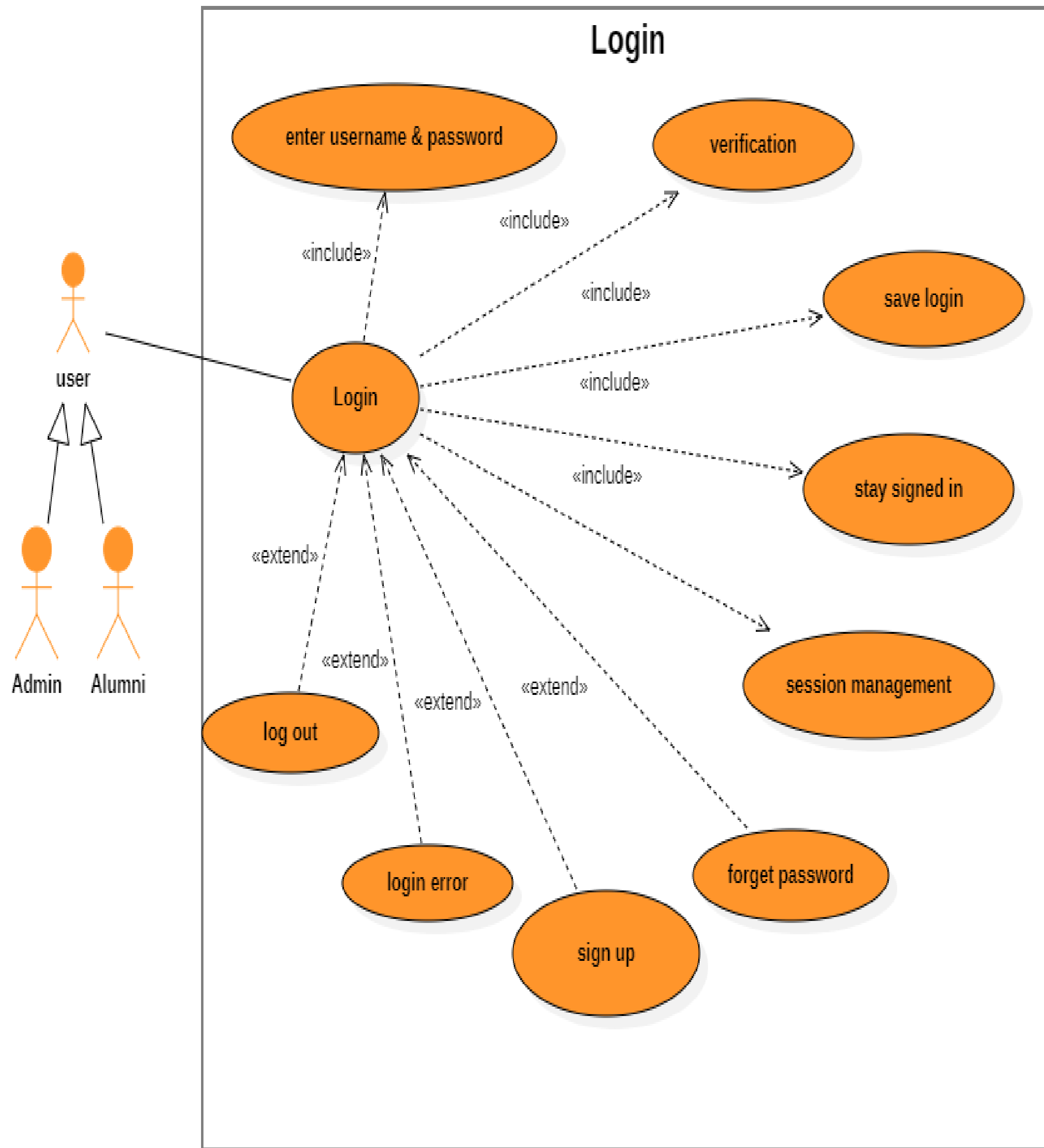


USECASE DIAGRAM

ID Number – Name:	UC.1-Sign up
Short Description:	If the Alumni does not have an account, the Admin will give the Alumni the opportunity to create an account.
Pre-conditions:	The Alumni not logged in to the system
Post-conditions:	The Alumni entered successful information and is returned to the home page as a Logged In Alumni
Error Situations:	Alumni is unable to sign up for one or more reasons and is returned to the home page as a Guest
Actors:	Alumni / Admin
Trigger:	User requests to sign up
Main Success Scenario	1.The Alumni selects sign up option. 2.The System prompts Alumni for registration information, username, password ,college email 3.The Alumni enters in their information. 4.System verifies information and creates account.
Alternative Scenarios:	1.Alumni clicks submit after entering information system asked for. 2.System displays information with warning message to correct invalid information.
Non-functional requirements:	must be :1.secure 2.fast

USECASE DIAGRAM

1.2) Login:

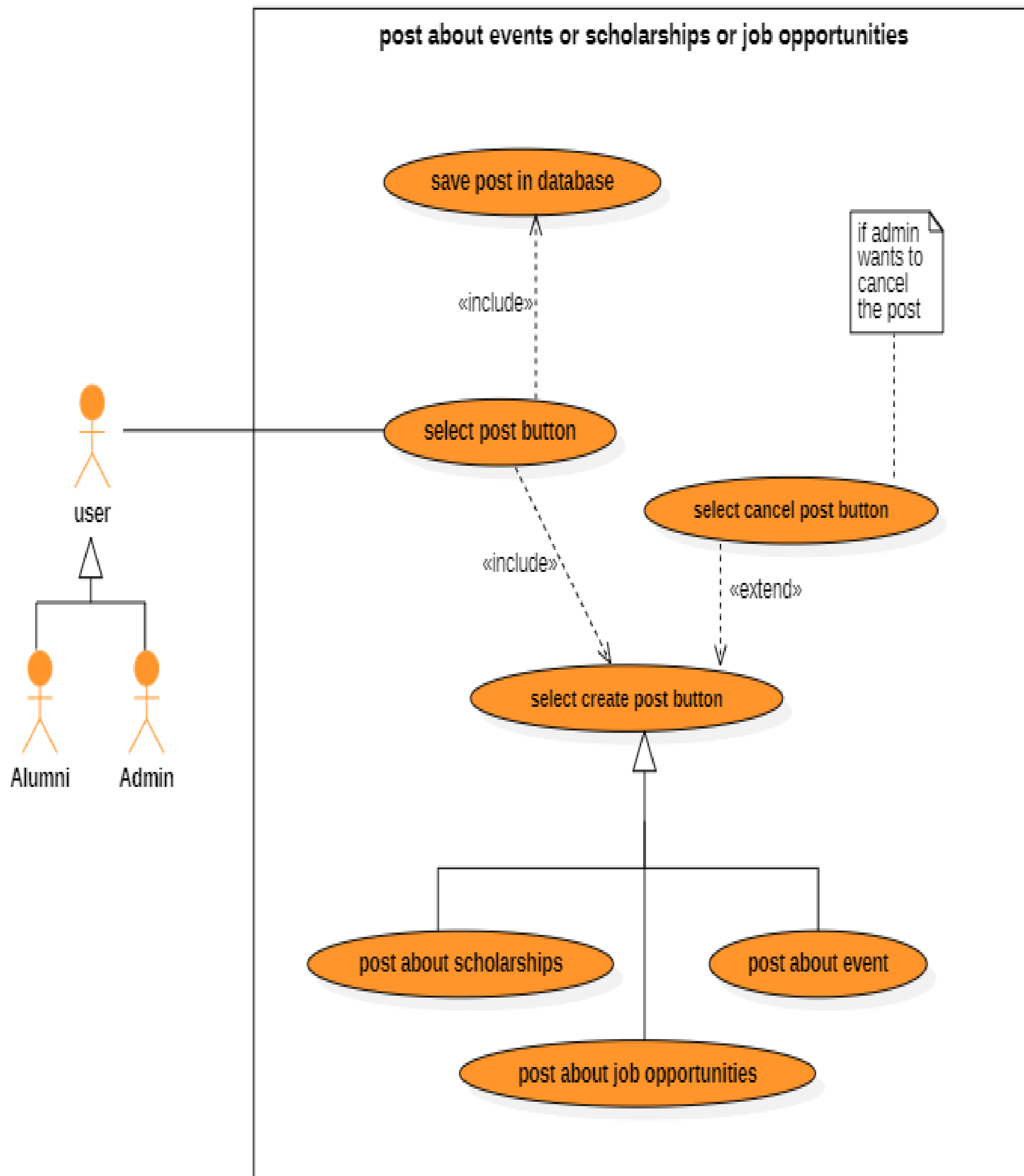


USECASE DIAGRAM

ID Number – Name:	UC.2-login
Short Description:	This use case allows users to login to the website
Pre-conditions:	open web interface and trying to log in with their account (must be register in a web first)
Post-conditions:	1. The user is logged in to the system 2. The user has access to the functions of the Alumni system
Error Situations:	the website is not opened
Actors:	Admin& Alumni
Trigger:	User requests to login
Main Success Scenario:	1.The user enters their username and password. 2.The user gains access to the systems functionality
Alternative Scenarios:	Wrong login credentials: 1. Users key in the wrong combination of username and password
Non-functional requirements:	must be : 1.secure 2.fast

USECASE DIAGRAM

1.3) Post about events/ scholarships/ job opportunities:

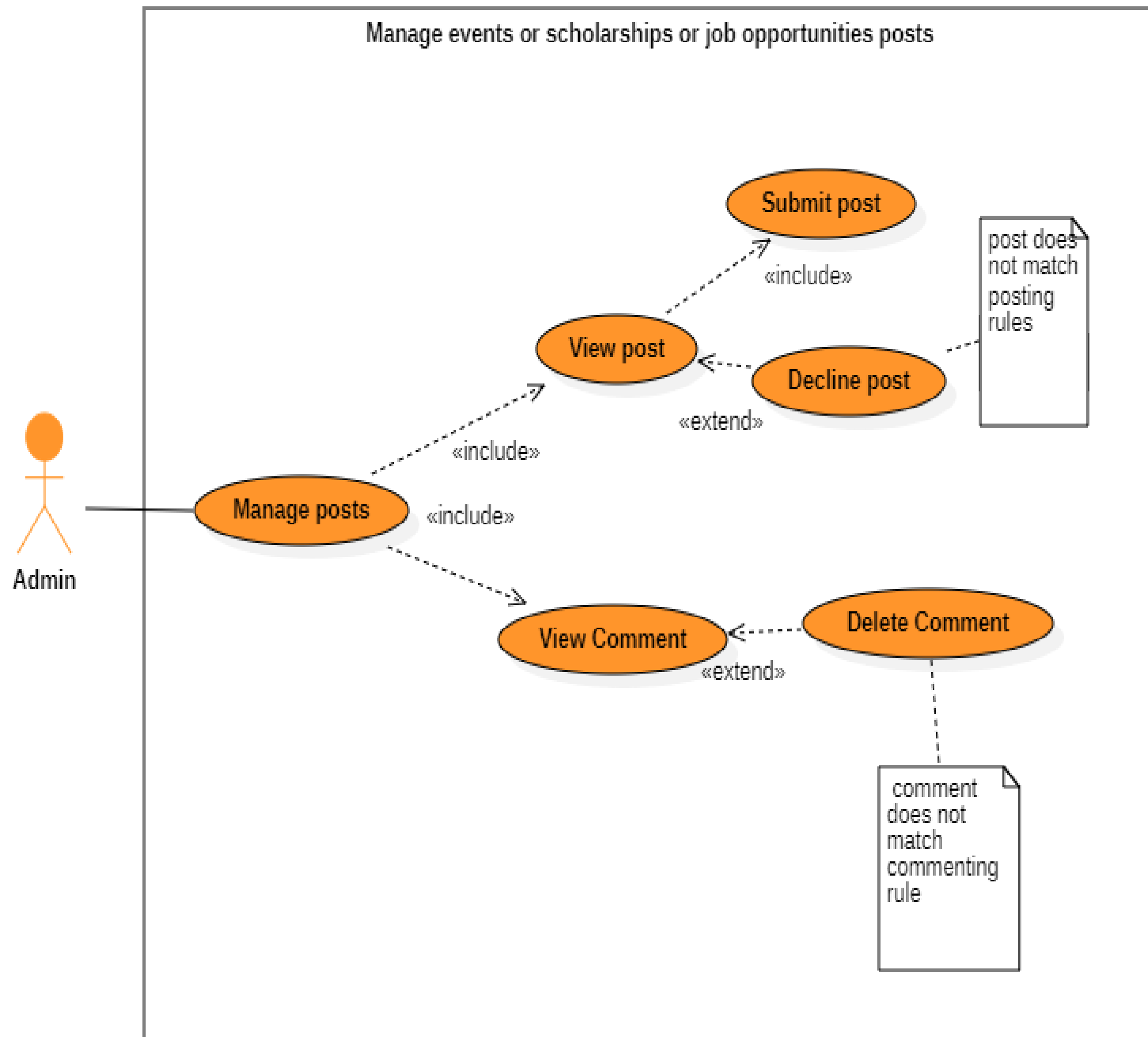


USECASE DIAGRAM

ID Number – Name:	UC.3-Post an event/ scholarships/ job opportunities
Short Description:	Admin post about event, scholarships or job opportunities
Pre-conditions:	Admin/Alumni logged in successfully
Post-conditions:	Admin/Alumni posted successfully, and the post is saved in the database
Error Situations:	There is no internet connection
Actors:	Admin/Alumni
Trigger:	Select create post button
Main Success Scenario:	1)open the website. 2) log in. 3) Select create post button. 4) write the post. 5) Select post button.
Alternative Scenarios:	5') select cancel button.
Non-functional requirements:	Usability and user friendly

USECASE DIAGRAM

1.4) Manage events or scholarships or job opportunities posts :

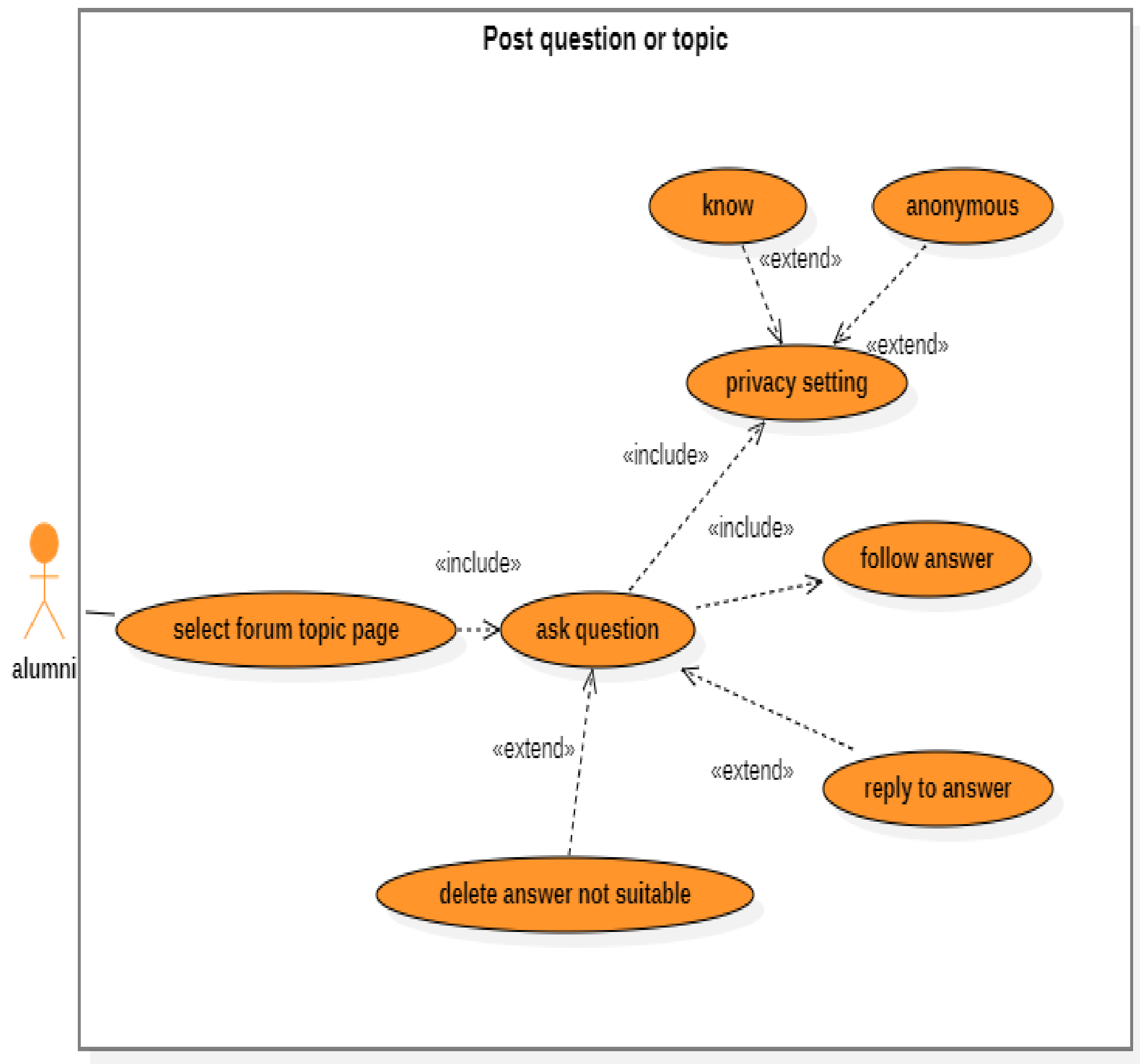


USECASE DIAGRAM

ID Number – Name:	UC.4-Manage events/scholarships/job posts
Short-Description:	Admin can manage almuni’s posts
Pre-conditions:	Admin logged into his account
Post-conditions:	Admin acceptes post/comment
Error Situations :	Admin Can’t log into his account
Actors:	Admin
Trigger:	Almuni adds post /comment
Main Success Scenario:	(1) Admin Views Almuni’s Post / Comment (2)Admin accepts post/ comment
Alternative Processes :	(2)’Admin deletes post / comment if it does not match rules.
Non-functional requirements :	Data is secure.

USECASE DIAGRAM

1.5) Post topics/ questions:

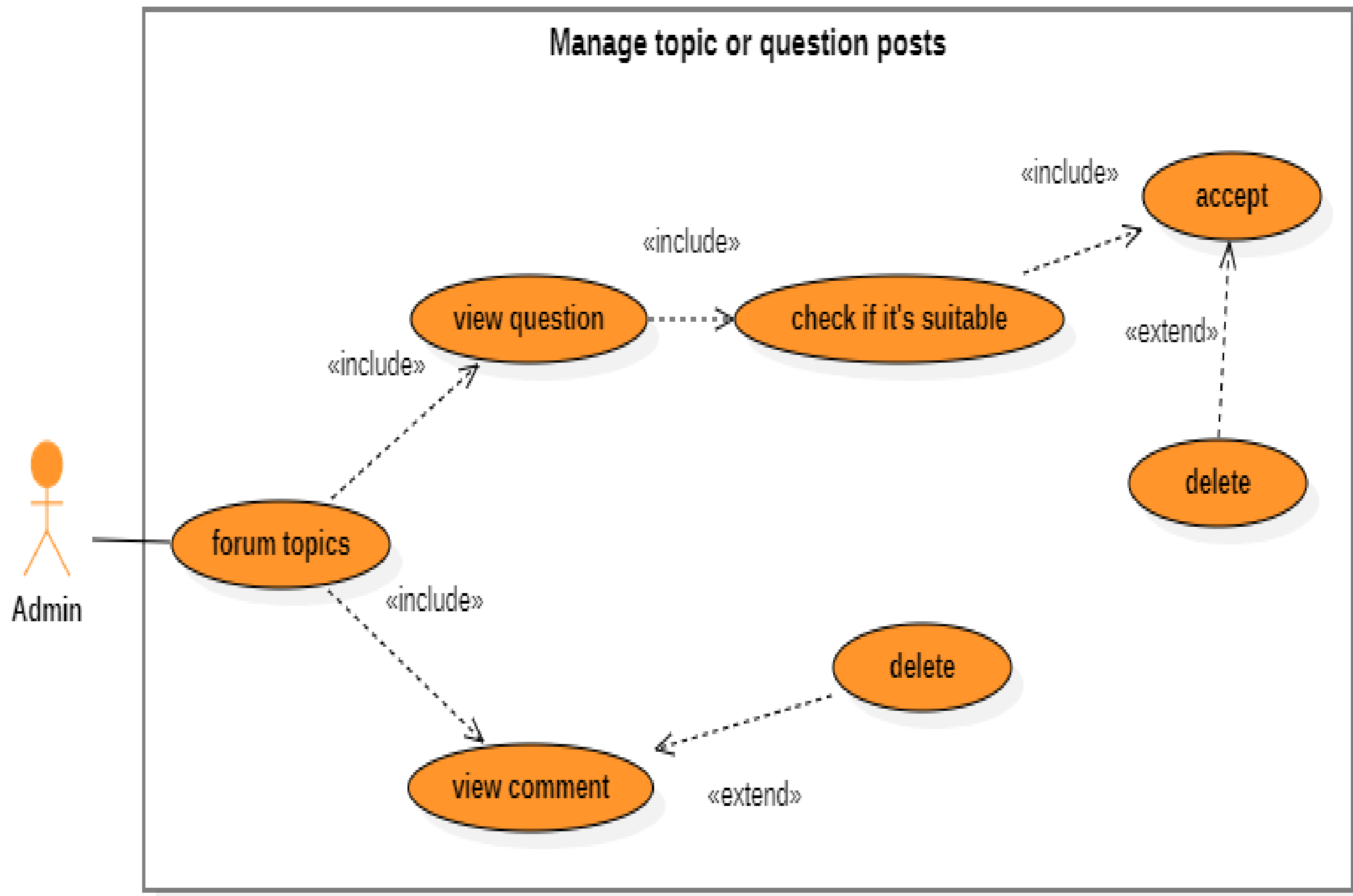


USECASE DIAGRAM

Id Number – Name	UC.5-Post question /topic
Short Description	Alumni post questions and topics
Pre-condition	Alumni has a question or help colleagues
Post-condition	Alumni has an answer to his question
Error Situation:	The site doesn't receive questions
Actors	Alumni
Trigger	Alumni moves to university website to ask his questions
main success scenario:	1. Alumni forum topic page 2.Alumni ask his questions
Alternative processes:	Alumni chose privacy settings if they want messages known or anonymous
Non-functional requirements:	Fast

USECASE DIAGRAM

1.6) Manage topic/ question posts:

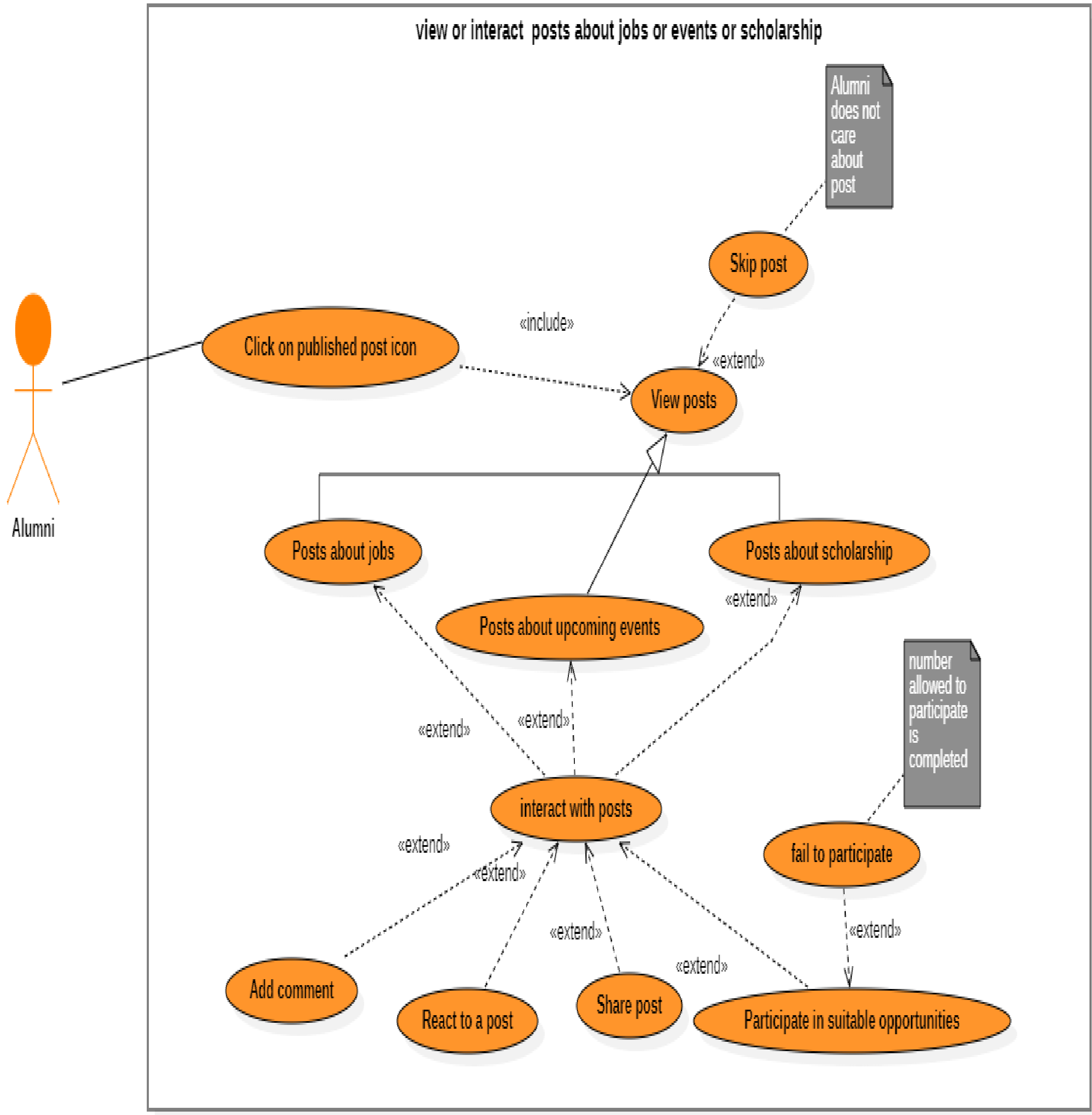


USECASE DIAGRAM

ID Number – Name	UC.6-Manage topic/ question posts
Short description	Admin mange topics and questions posts
Pre-condition	Check for all alumni’s posts
Post-condition	I answered all posts
Error Situation :	If an error occurs ,the post isn’t followed
Actors	Admin
Trigger	Admin see forum topics
main success scenario:	1.Admin view questions and accept it or no 2.Admin view comment and interact and answer or delete
Alternative processes:	Admin if questions are suitable
Non-functional requirements :	Fast

USECASE DIAGRAM

1.7) View/Interact posts:

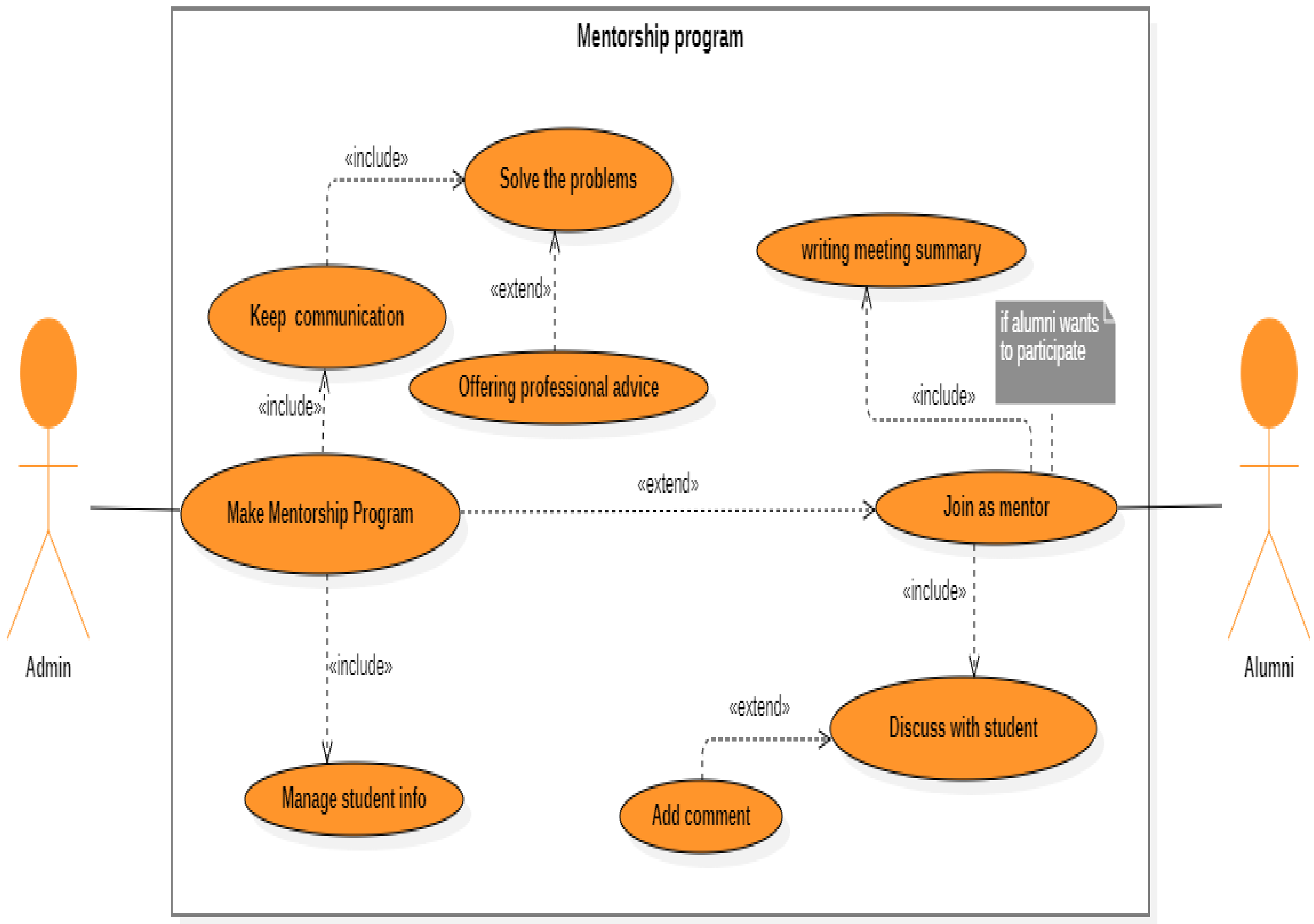


USECASE DIAGRAM

ID Number - Name:	UC.7-view / interact posts
Short Description :	Alumni view posts that admin post them and interact with them.
Pre-conditions :	Alumni logged in successfully and admin post posts in the published post page.
Post-conditions :	Alumni can interact with published posts.
Error situations:	There is no internet connection.
Actors:	Alumni
Trigger:	Click on published post icon.
Main success scenario:	1)View published post. 2)Interact with post by add a comment or react to a post or share post. 3)Participate in suitable job opportunities or events or scholarship.
Alternative scenarios:	2') skip a post. 3') if number allowed to participate completed , the alumni cant participate .
Non-functional requirements:	User friendly and save change in database.

USECASE DIAGRAM

1.8) Mentorship program:

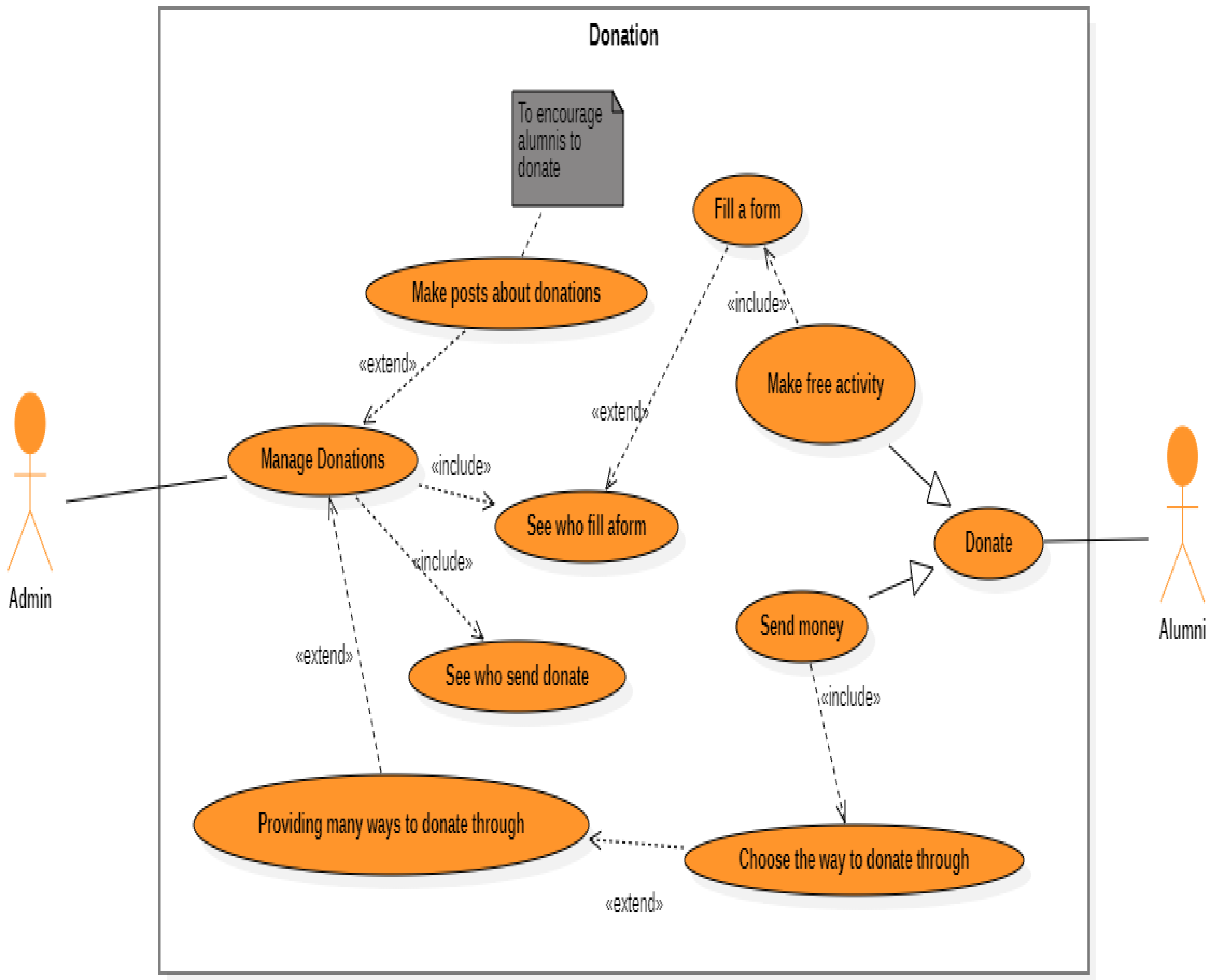


USECASE DIAGRAM

ID Number- Name:	UC.8-Mentorship program
Short Description	Admin control to connect alumni and mentor Alumni participate as mentor
Pre- condition	Make sure 1) Alumni is a mentor 2) Admin Make Mentorship Program
Post- condition	Mentor communicate with student Admin covers all aspects of the program
Error Situations	Alumni failed to be mentor Failed to Create mentorship program
Actors	Admin / Alumni
Trigger	There are mentorship between admin and alumni
Main success scenario:	1)Mentors communicate with Alumni 2) Mentors can access data 3) Mentors offering professional advice
Alternative scenario:	Alumni fails to be a mentor
Non-functional Requirements:	Communication between them must be secure

USECASE DIAGRAM

1.9) Donation:

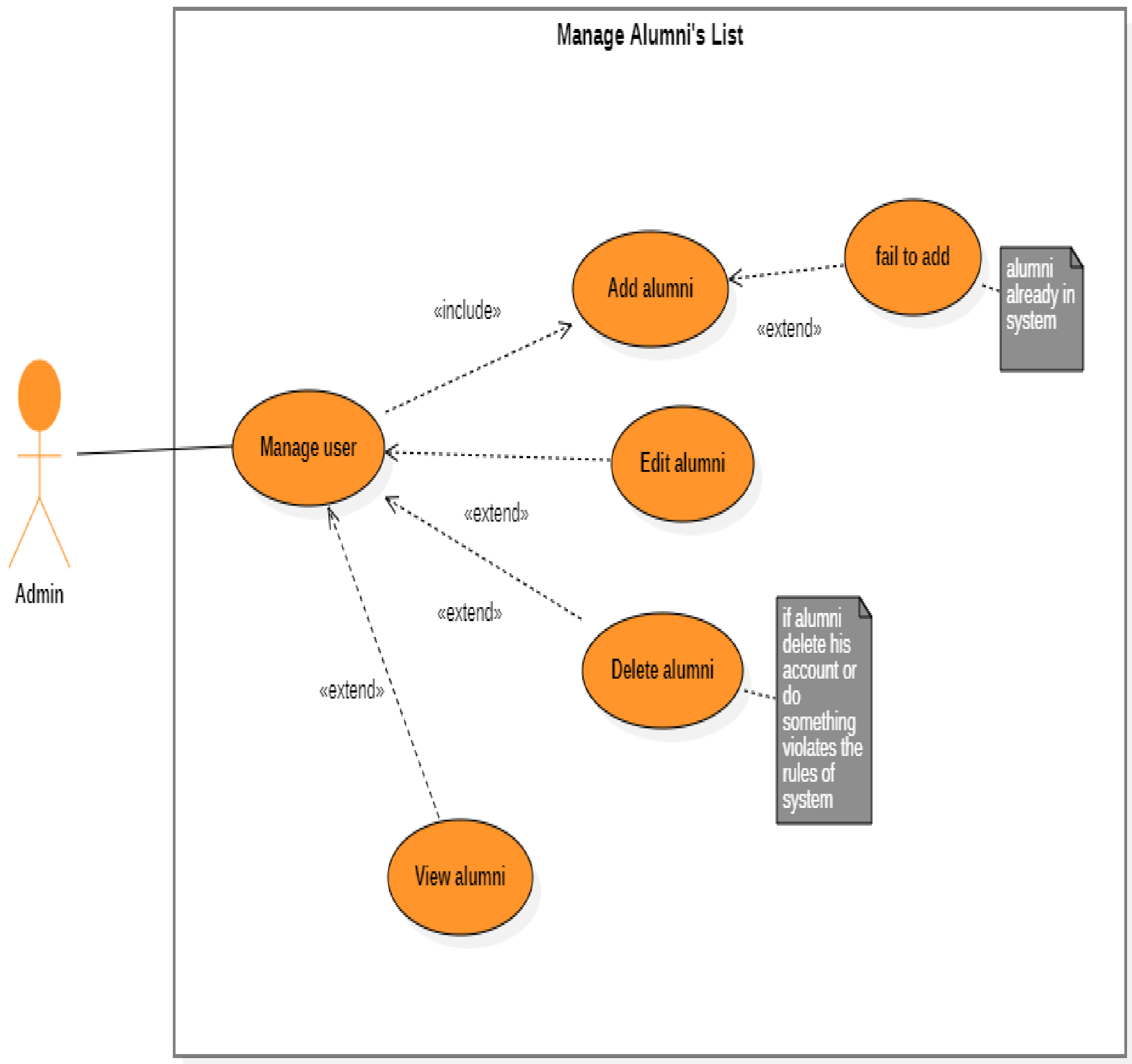


USECASE DIAGRAM

ID Number- Name:	UC.9-Donation / Donate
Short Description:	Admin manage donations / Alumni send donate
Pre-conditions:	Admin have changes to do Alumni wanna support his faculty
Post- conditions:	Admin: Changes done successful Alumni : donate successful
Error Situations:	Admin: Failed to manage donations Alumni: Error in data entered by alumni
Actors:	Admin / Alumni
Trigger:	Admin want to manage donations / Alumni want to donate
Main success scenarios:	Admin: 1)Open Donation Page 2)See who donated or who fill the form 3)Add/Update ways to make donation easier Alumni: 1)Open Donate page 2)Choose Donate by money 3) Choose best way to send money 4)Fill the form
Alternative processes:	Admin: 3`) Make post to encourage other alumni to donate Alumni: 2`)Choose Free Activity
Non-functional requirements:	Secure

USECASE DIAGRAM

1.10) Manage Alumni's List:

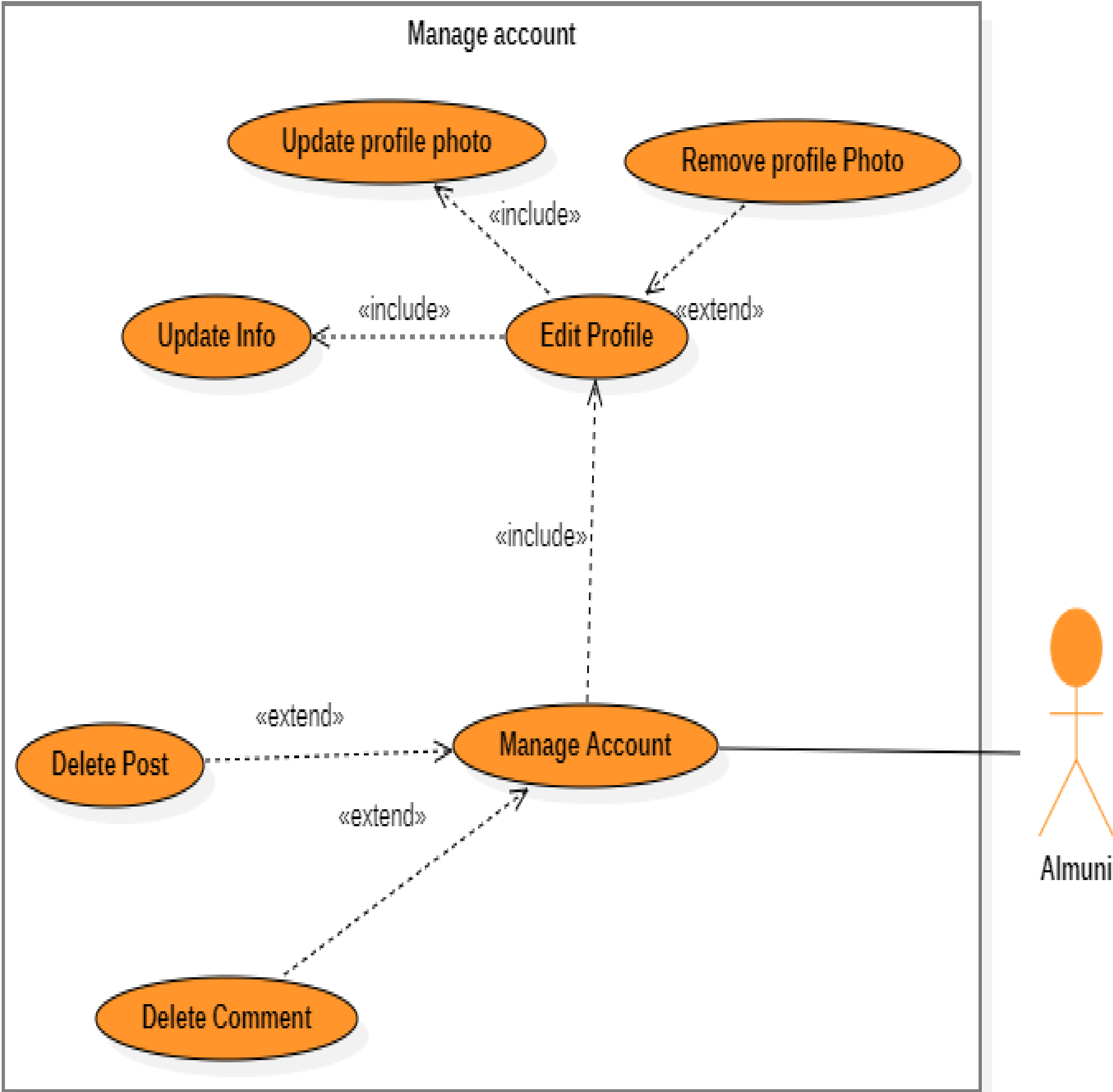


USECASE DIAGRAM

ID Number - Name:	UC.10-Manage Alumni's List
Short Description :	After admin logged in system, he can manage user information through add other alumnus , delete others , update information of others and view who use the system.
Pre-conditions :	Admin logged in successfully.
Post-conditions :	The change must save in database.
Error situations:	There is nothing.
Actors :	Admin
Trigger :	select manage user icon
Main success scenario:	1)Add new alumni 2)Edit alumni information if needed 3>Delete alumni 4)View alumnus in system
Alternative scenarios :	1') Admin fail to add alumni if he/she in system and already has an account
Non-functional requirements :	Data must be secure and retrieve it fast

USECASE DIAGRAM

1.11) Manage Account:

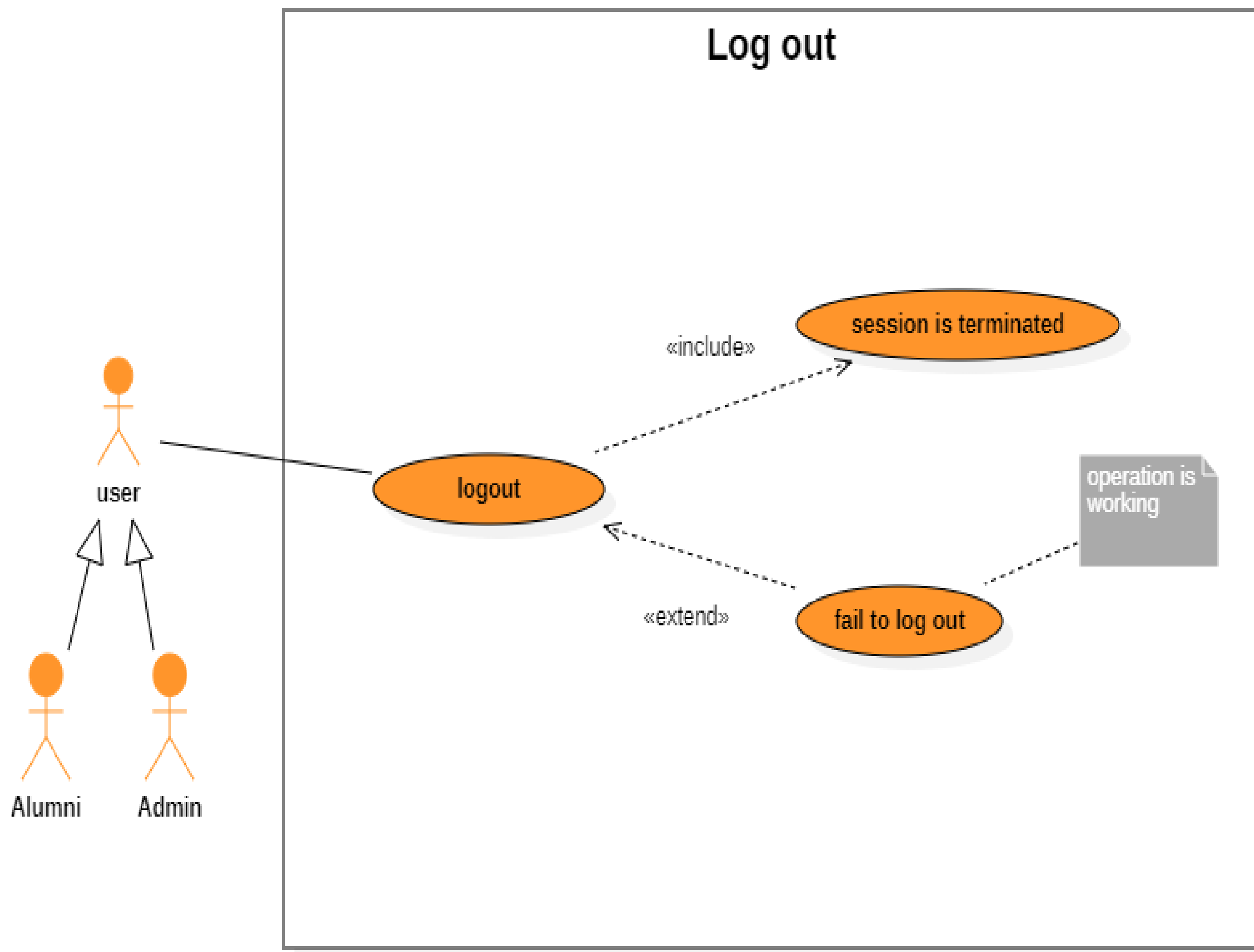


USECASE DIAGRAM

ID Number – Name:	US.11-Manage account
Short-Description:	Almuni manages his account
Pre-conditions:	Almuni logged in to his account
Post-conditions:	Almuni Added/ removed profile photo , added/deleted post/comment
Error Situations :	Almuni Can't log into his account
Actors:	Almuni
Trigger:	Almuni logged into his account
Main Success Scenario:	(1)Almuni adds post / comment/profile photo or updates info (2) Admin accepts posts / comments
Alternative Processes :	Admin deletes posts /comments
Non-functional requirements :	User friendly and save change in database.

USECASE DIAGRAM

1.12) Logout:

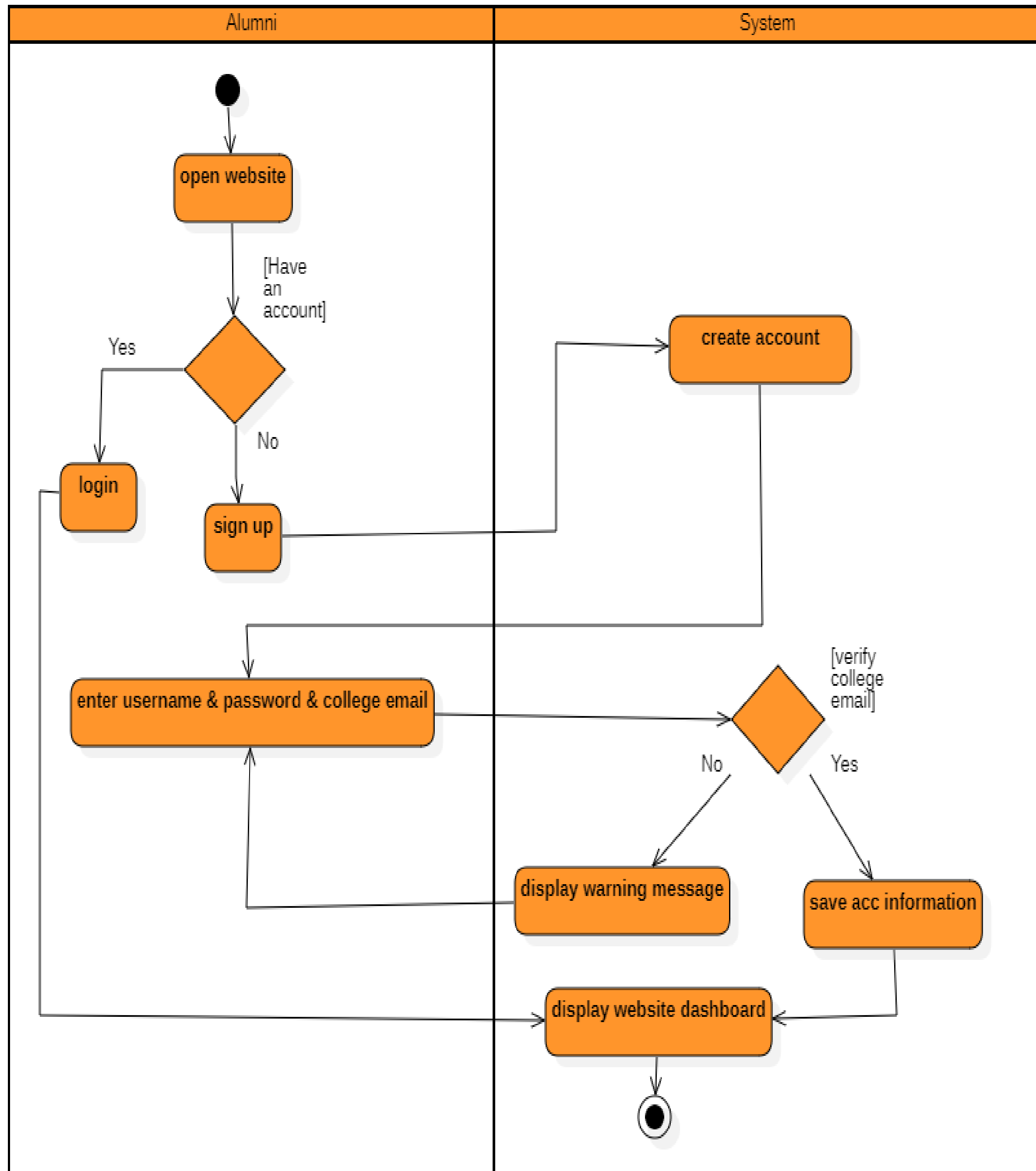


USECASE DIAGRAM

ID Number – Name:	UC.12-log out
Short Description:	The user clicks on “Logout” and their session is terminated
Pre-conditions:	1. The user is logged in 2. The user no longer wants to be logged in
Post-conditions:	1. The user is logged out 2. The user can not do any operation until login again
Error Situations:	internet problem while logging out
Actors:	Admin& Alumni
Trigger:	User is done using the web application
Main Success Scenario:	1.User is done using the web application 2.The user clicks on the logout button 3.The system logs the user out and invalidates the session
Alternative Scenarios:	fail to log out -> because an operation is working
Non-functional requirements:	must be secure

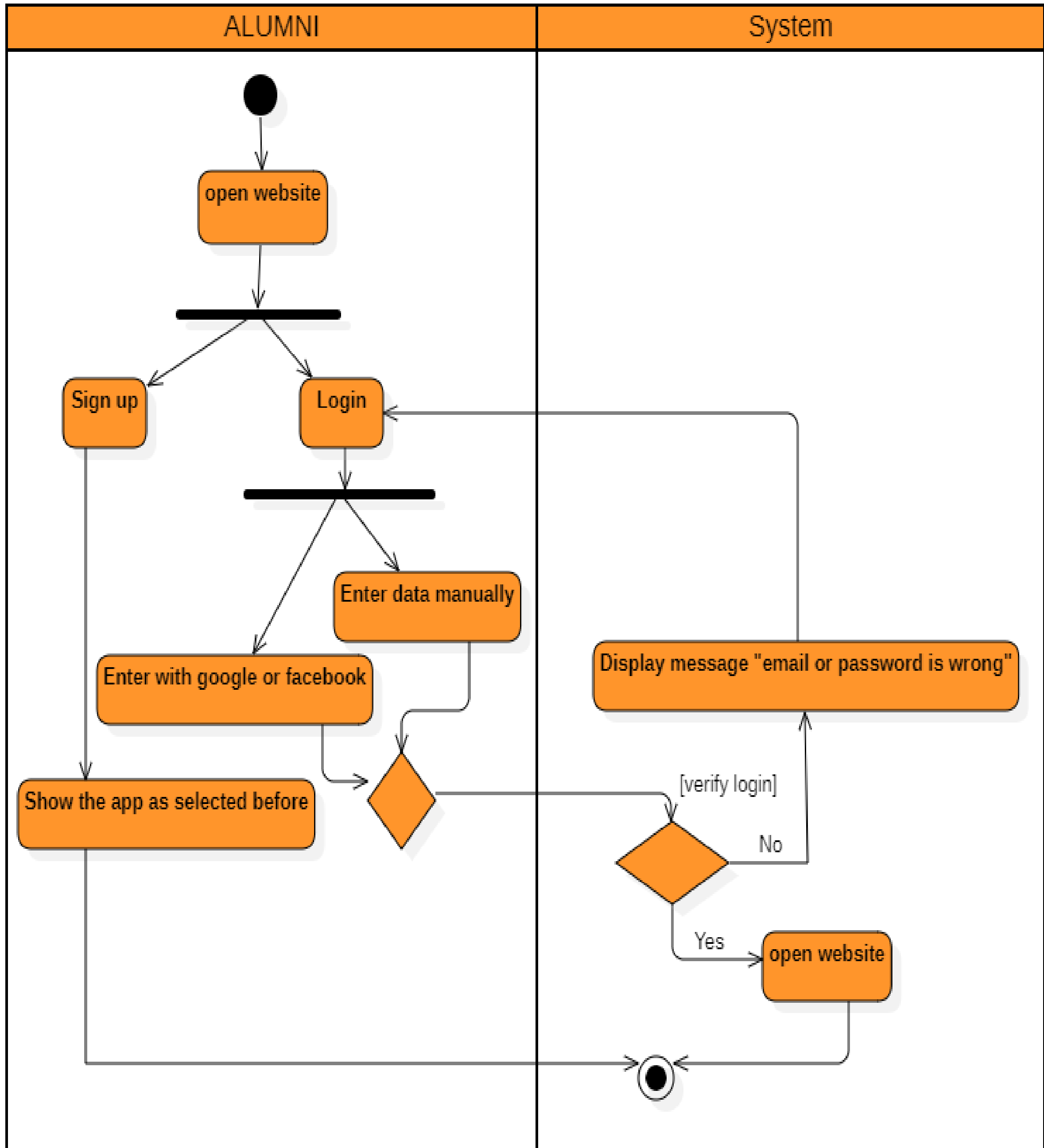
ACTIVITY DIAGRAM

1.1) Sign up:



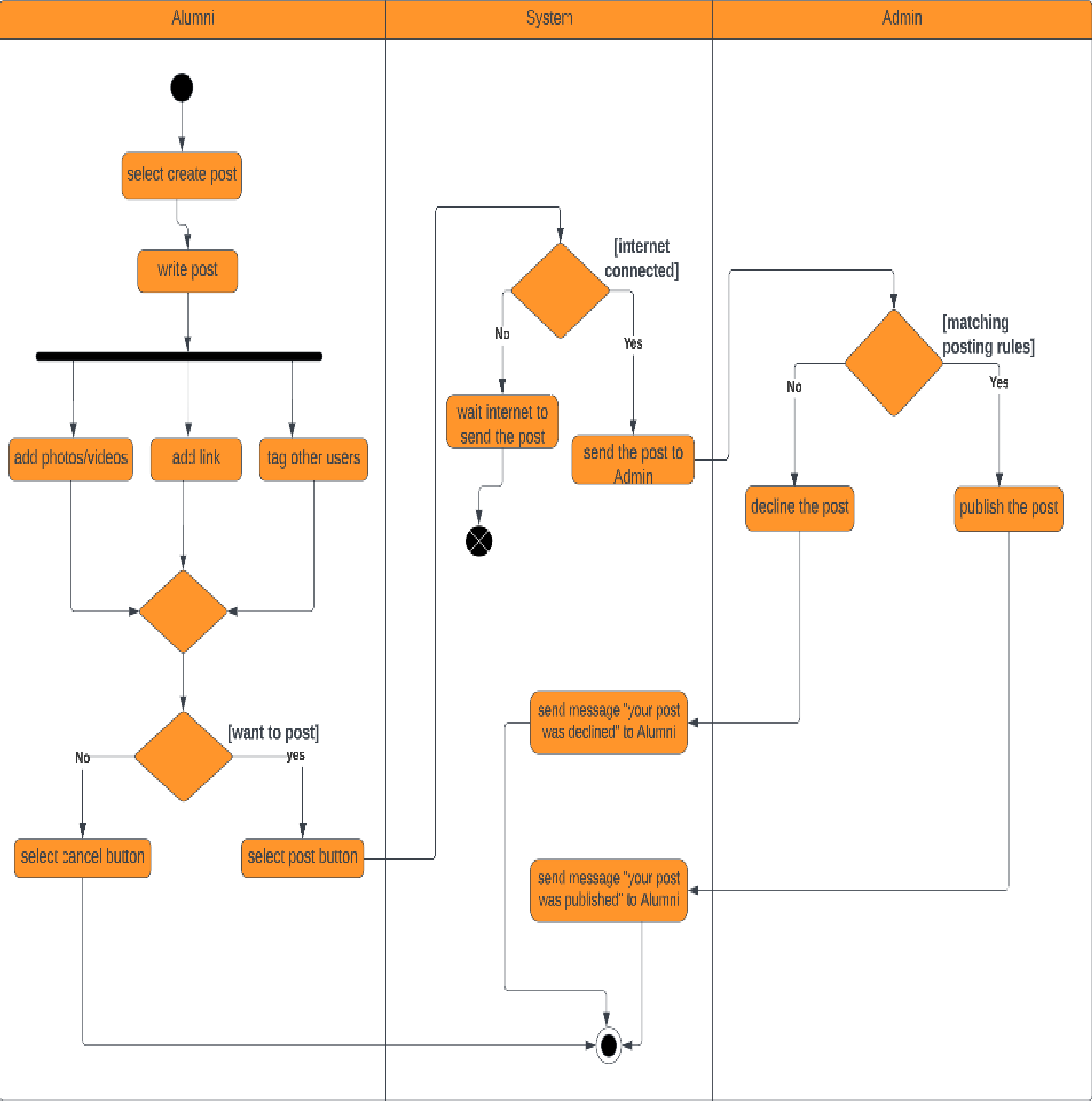
ACTIVITY DIAGRAM

1.2) Login:



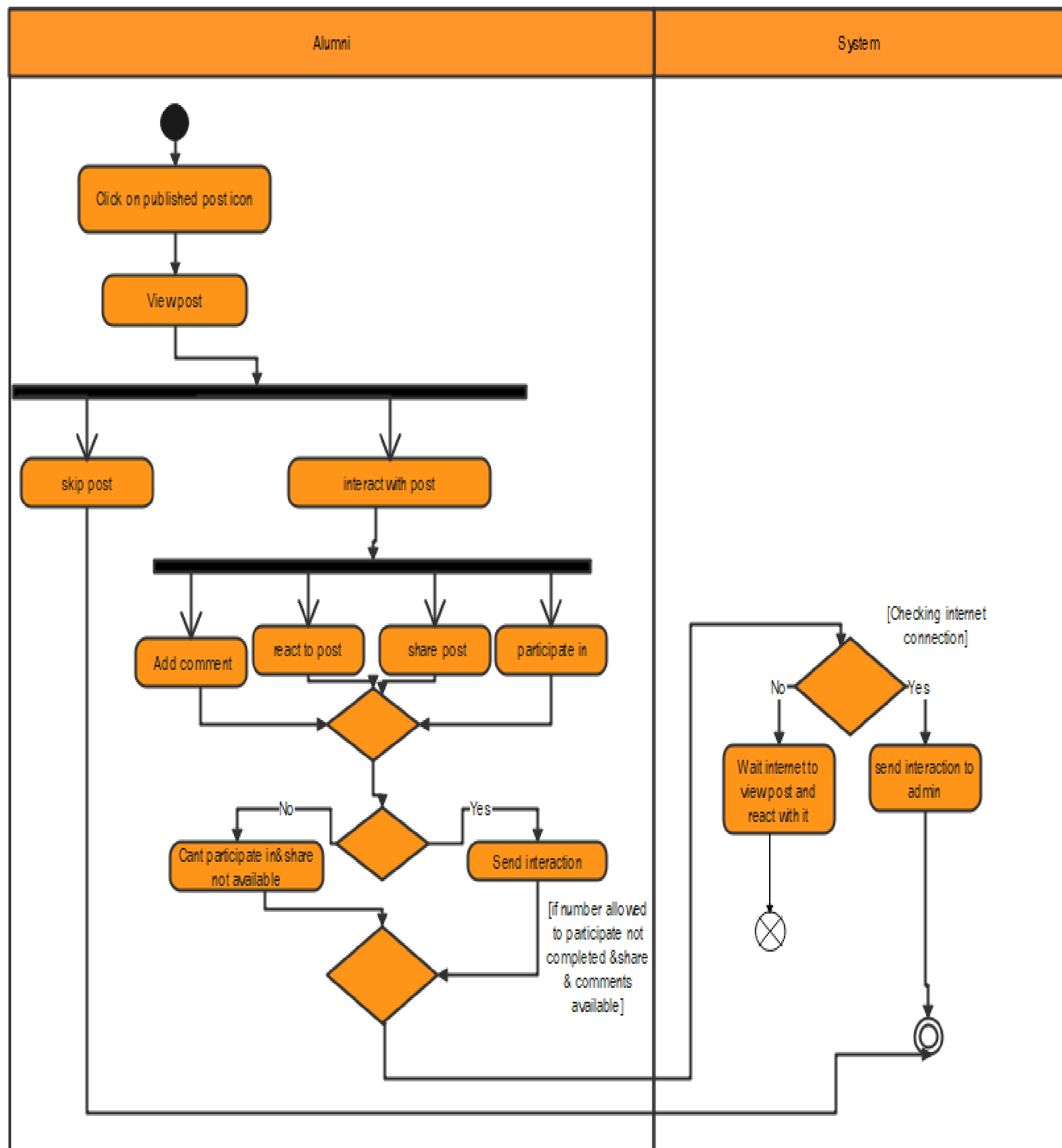
ACTIVITY DIAGRAM

1.3) Post events/ jobs/ scholarships:



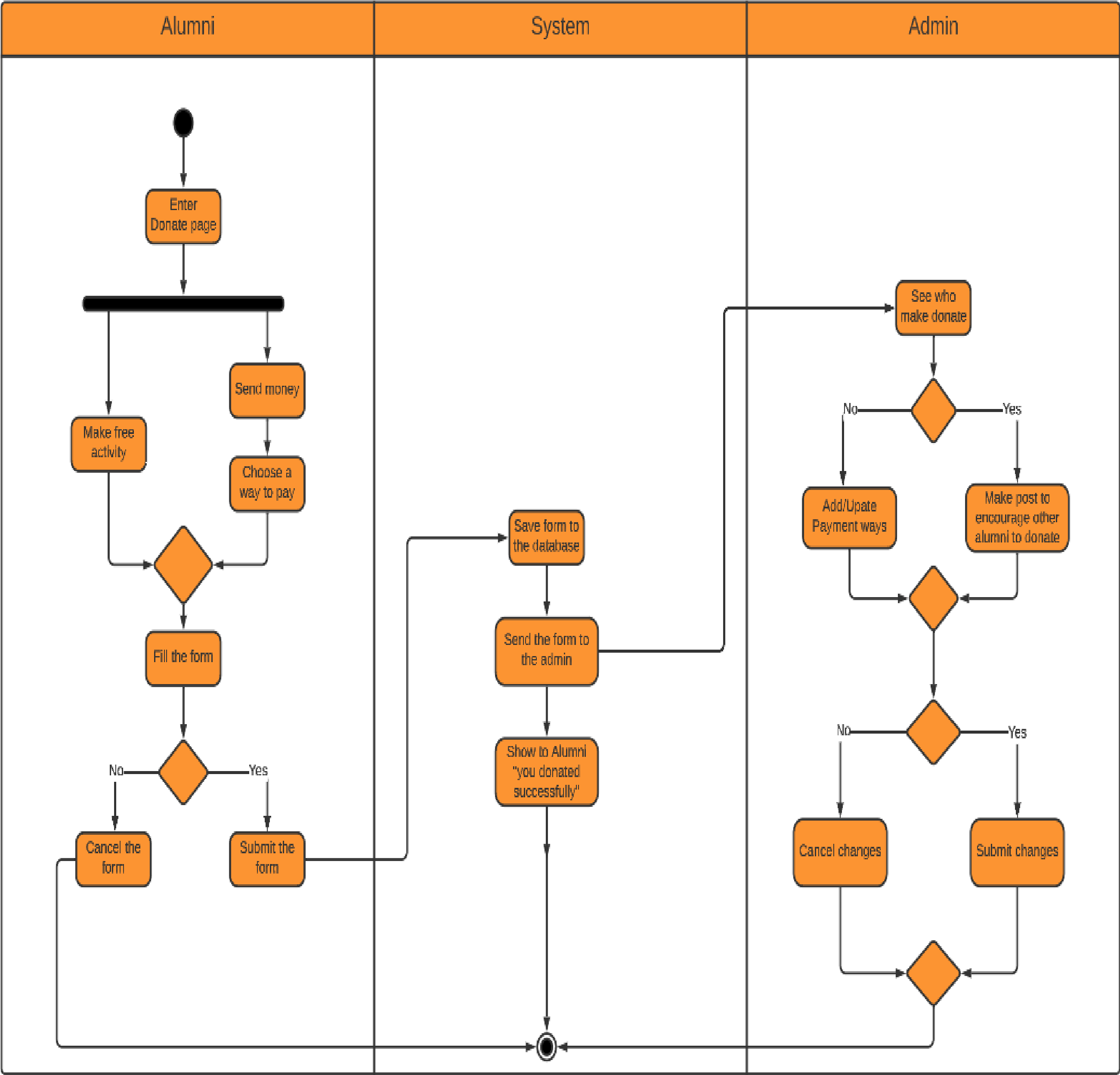
ACTIVITY DIAGRAM

1.4) View/ interact posts:



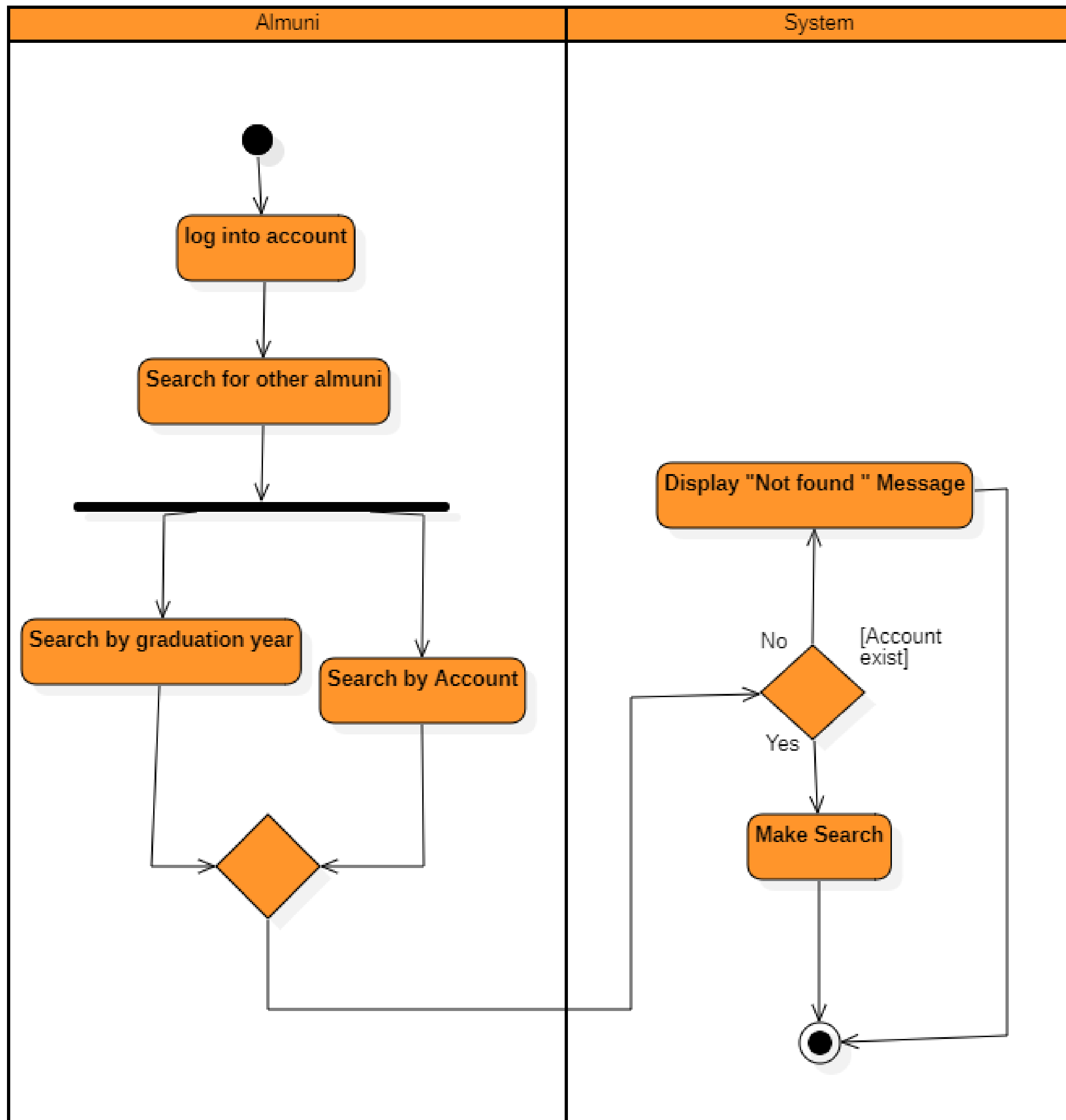
ACTIVITY DIAGRAM

1.5) Donation:



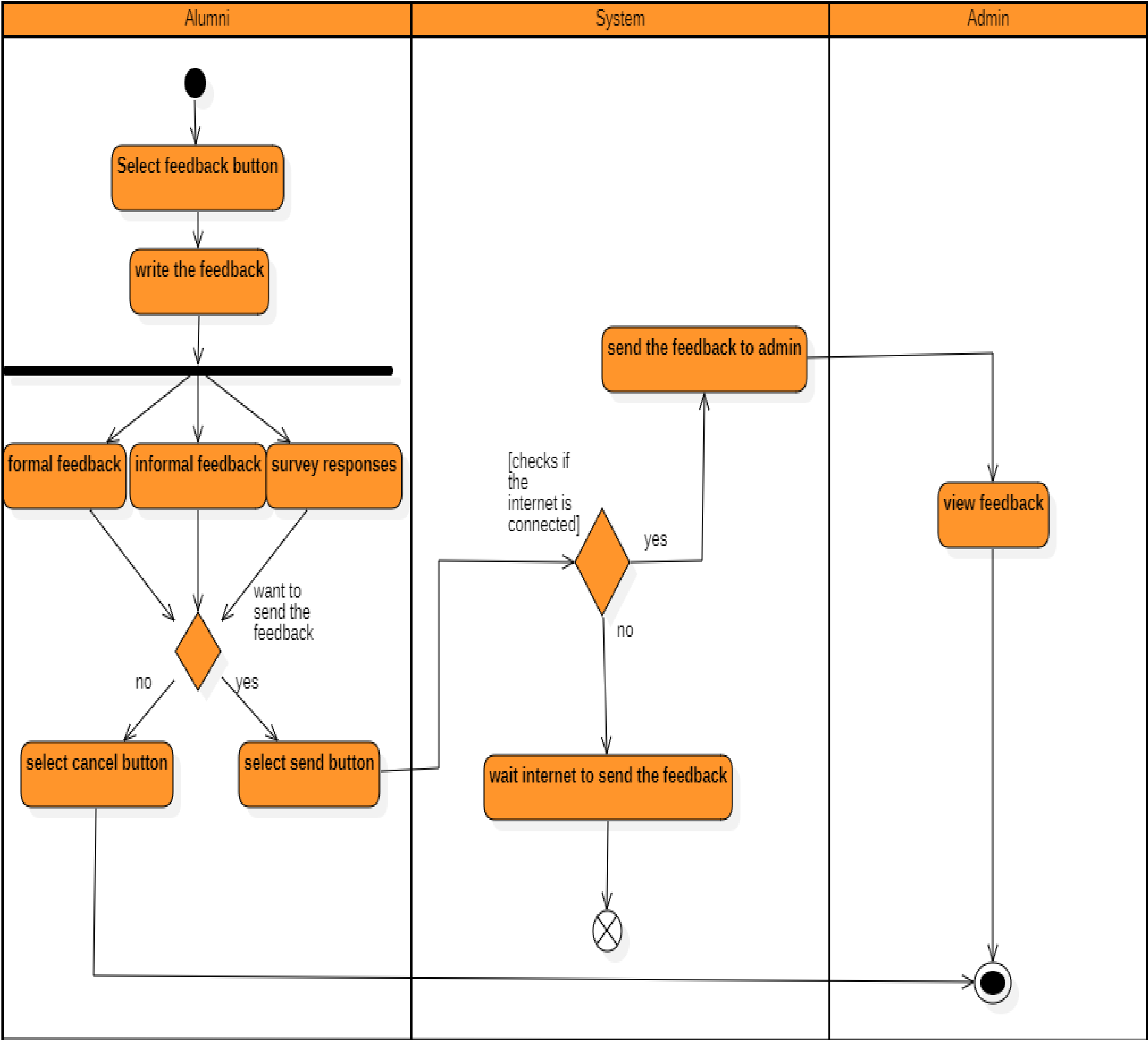
ACTIVITY DIAGRAM

1.6) Search:

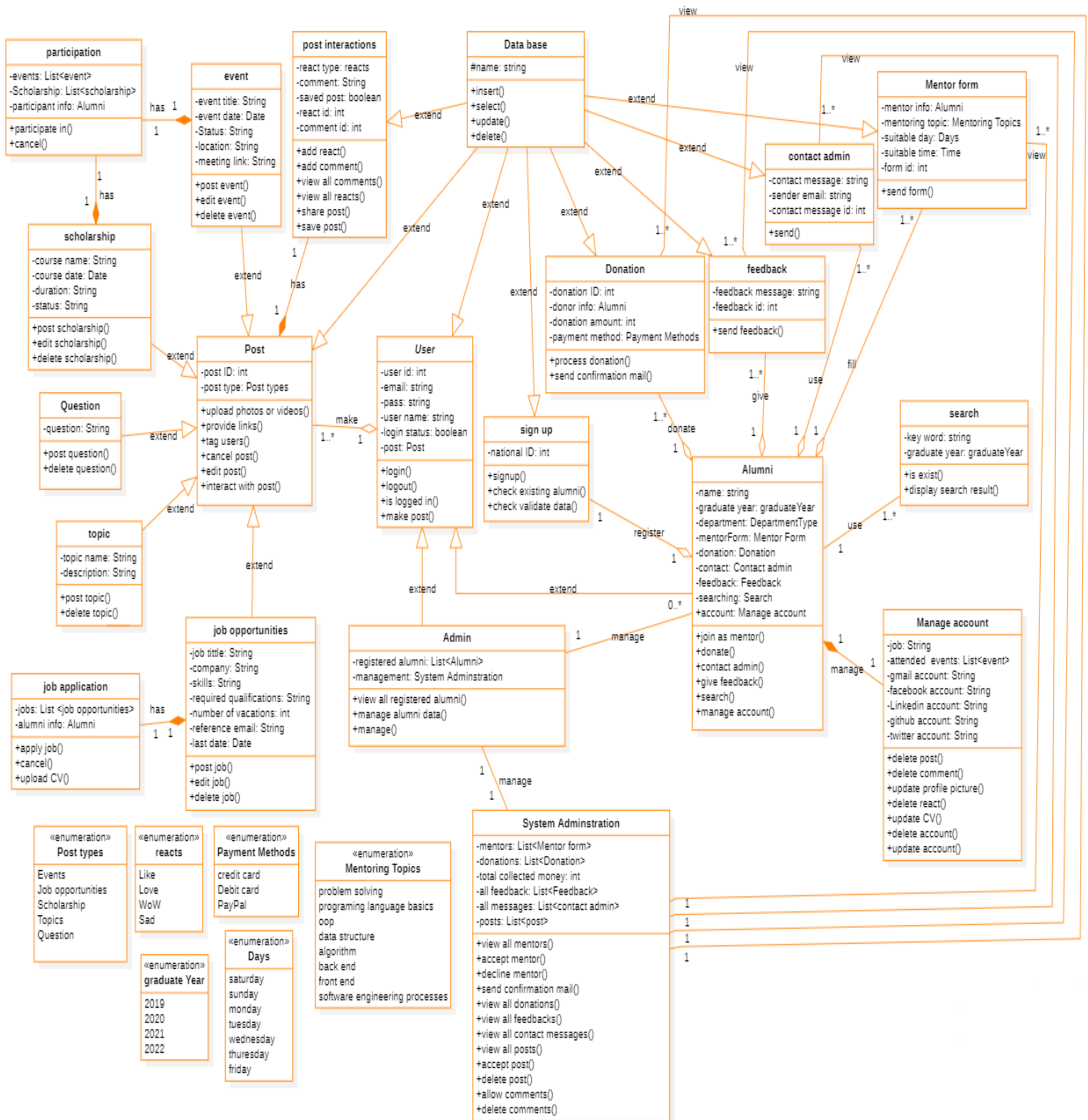


ACTIVITY DIAGRAM

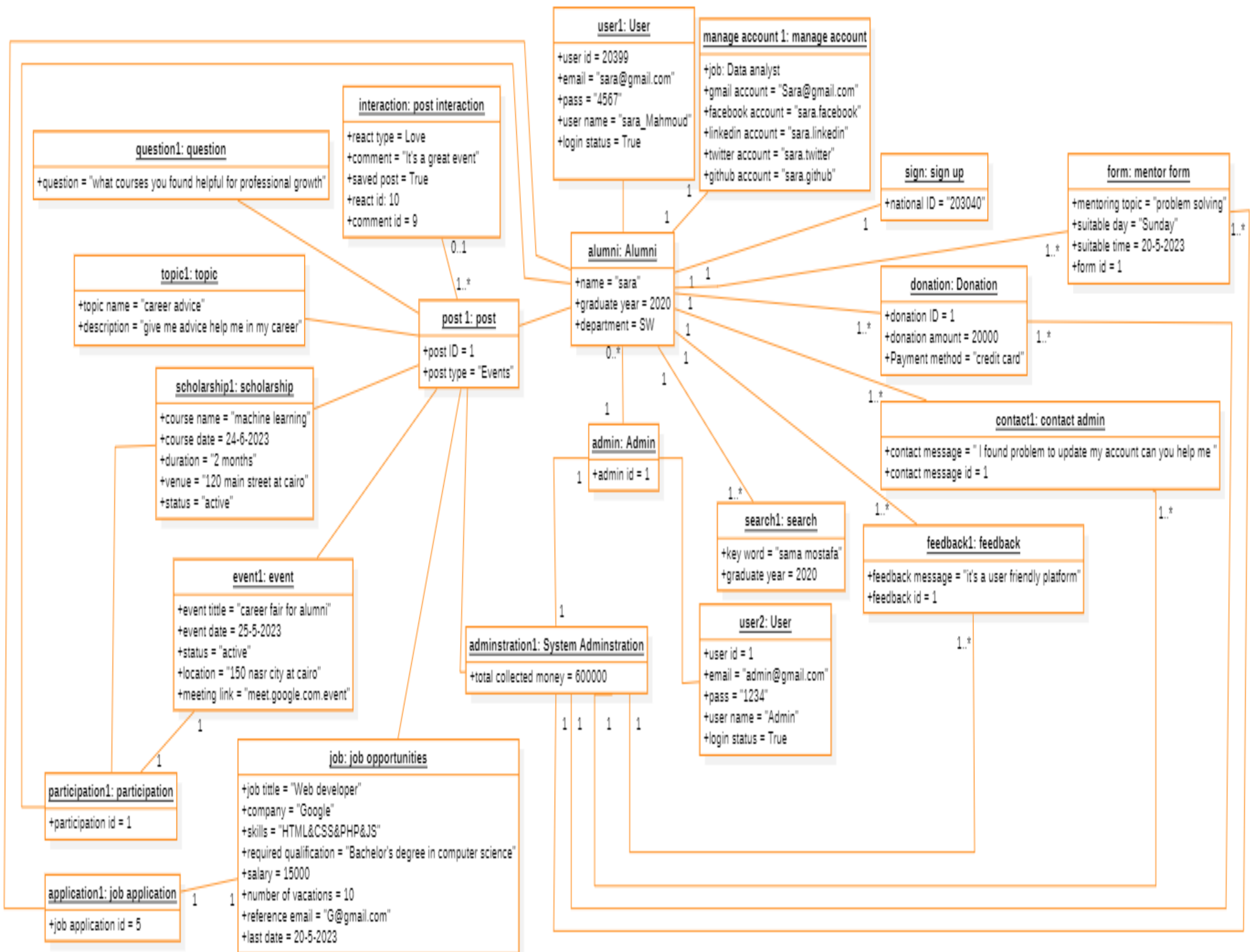
1.7) Feedback:



CLASS DIAGRAM

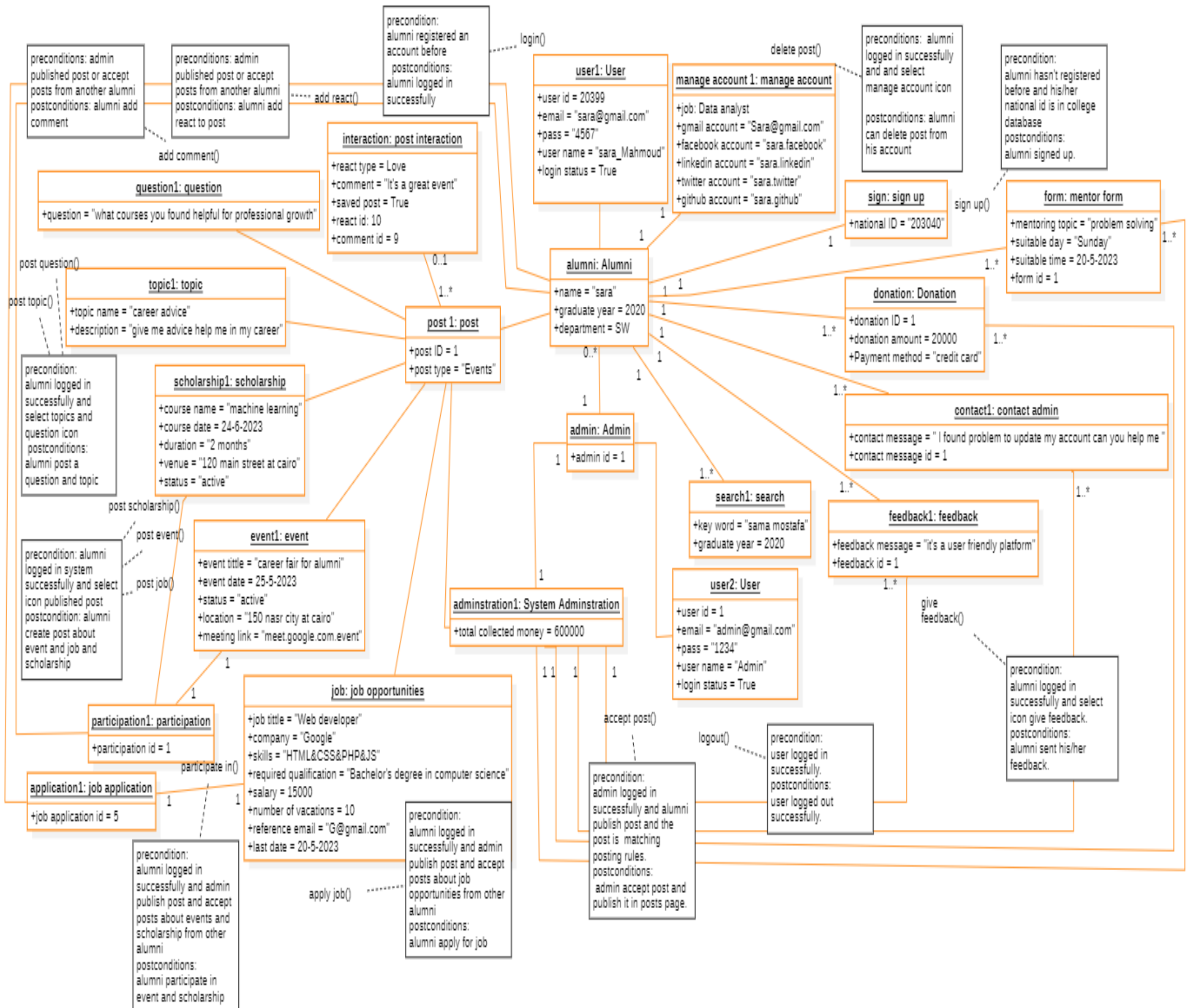


OBJECT DIAGRAM

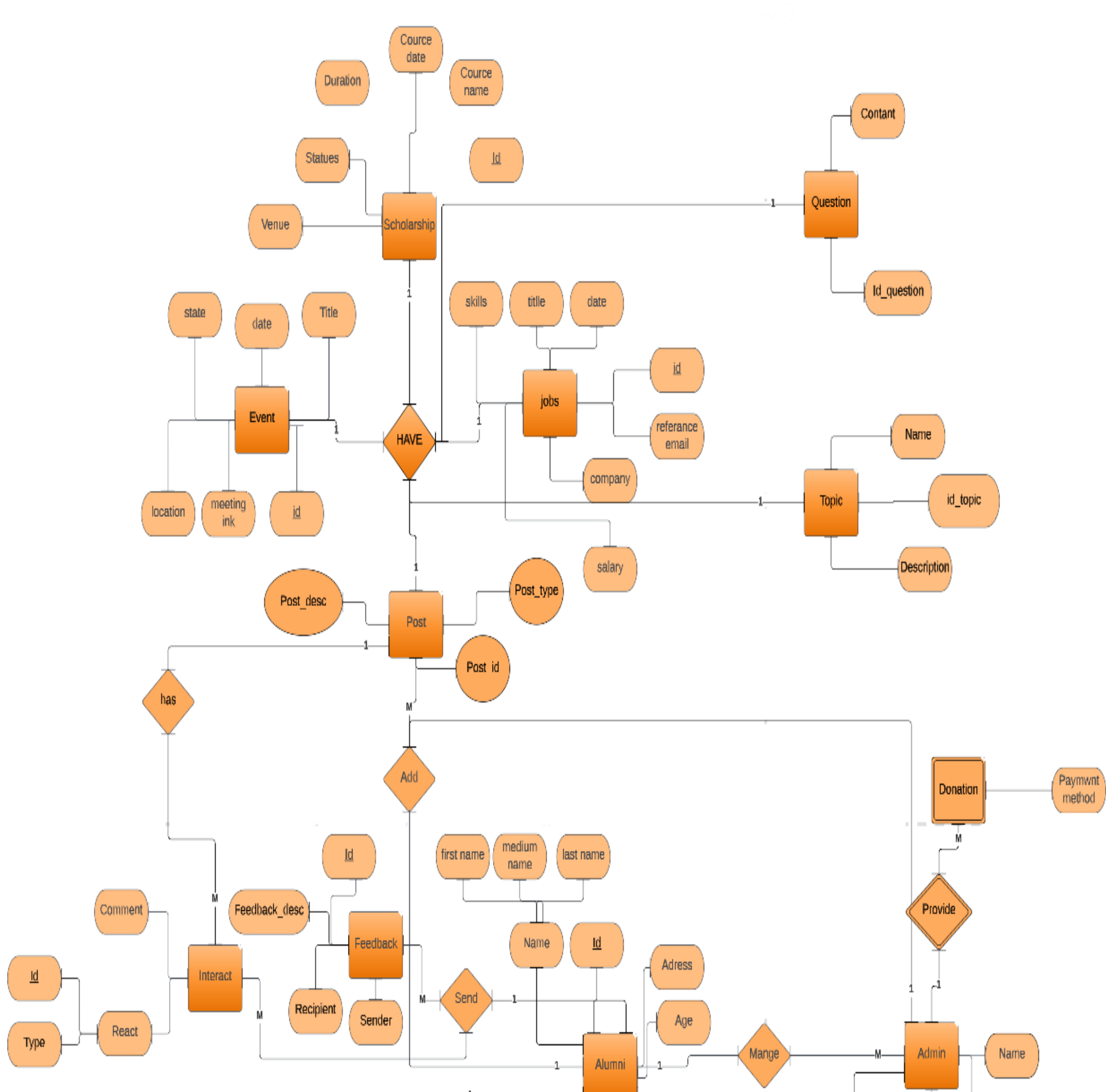


OBJECT DIAGRAM

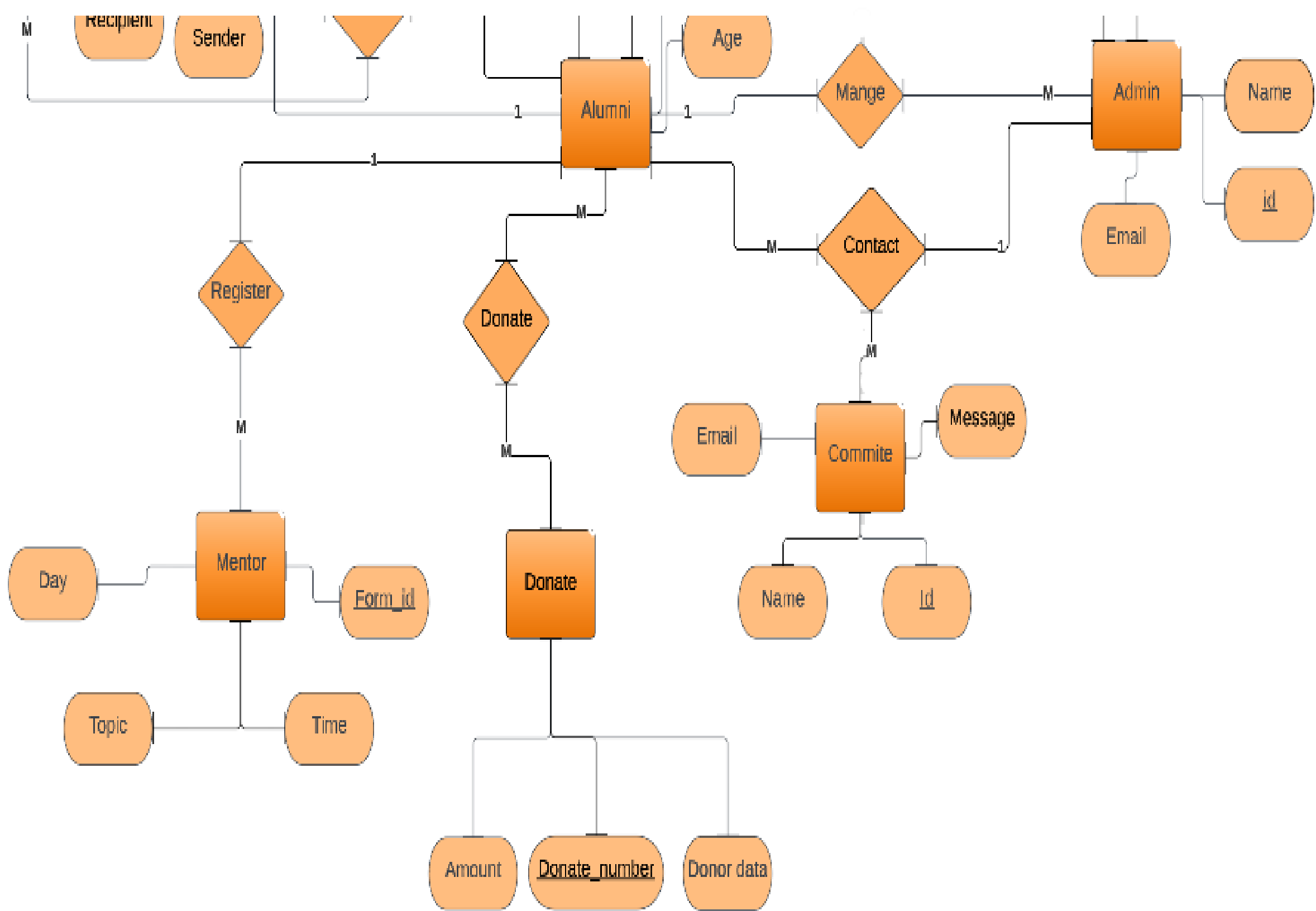
Pre & Post Conditions:



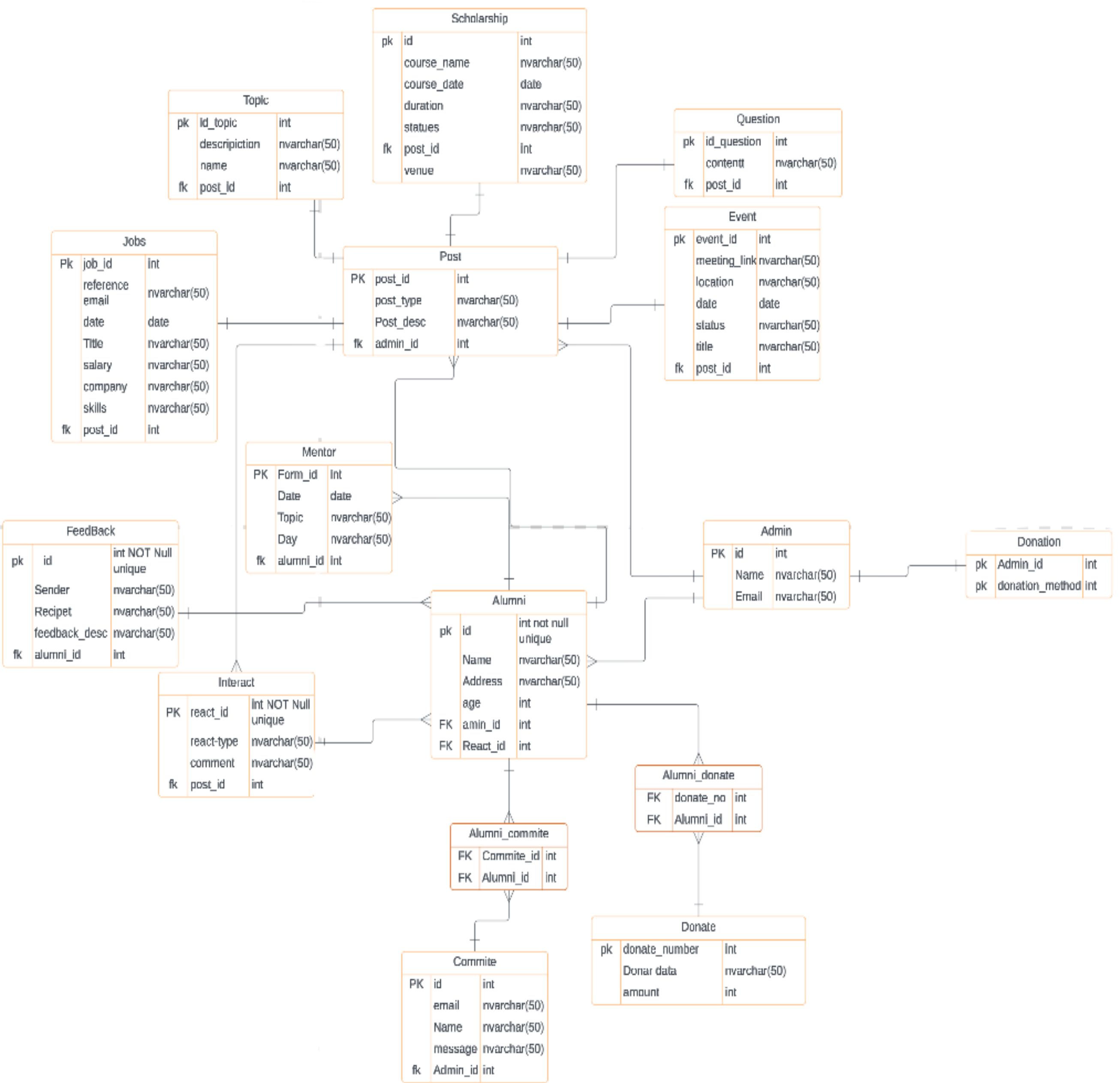
ER DIAGRAM



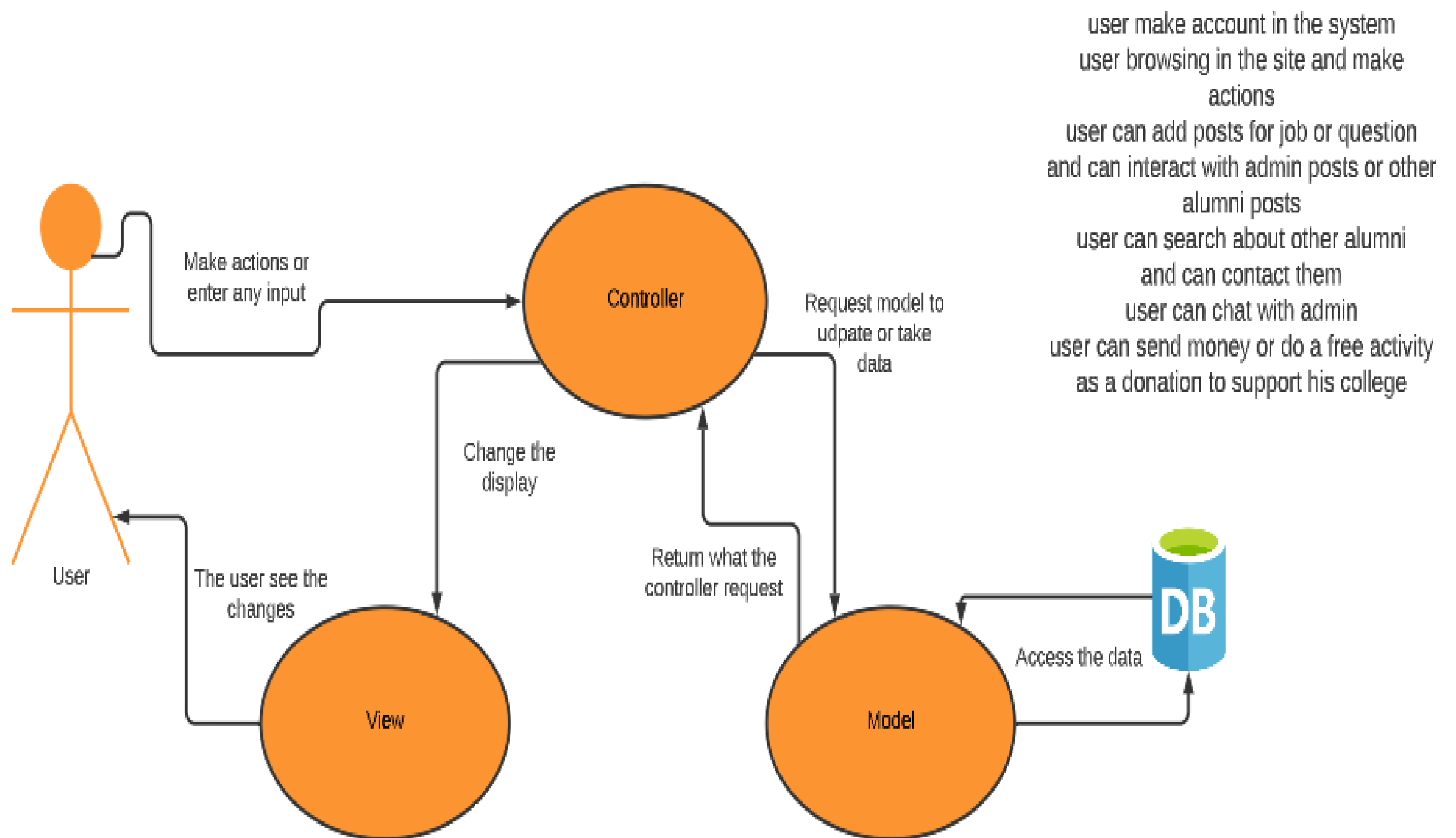
ER DIAGRAM



ER TABLES DIAGRAM



SYSTEM ARCHITECTURE



The view is the interface, user interacts with the system and make actions or enter inputs
The controller take the inputs or this actions to the model then make the view change him self to what the user need
The model(classes and database clases) take or update the data then tell the controller that data had updated or return the data that needed

PACKAGE DIAGRAM

Components:

Alumni

Post

Post
interaction

Donation

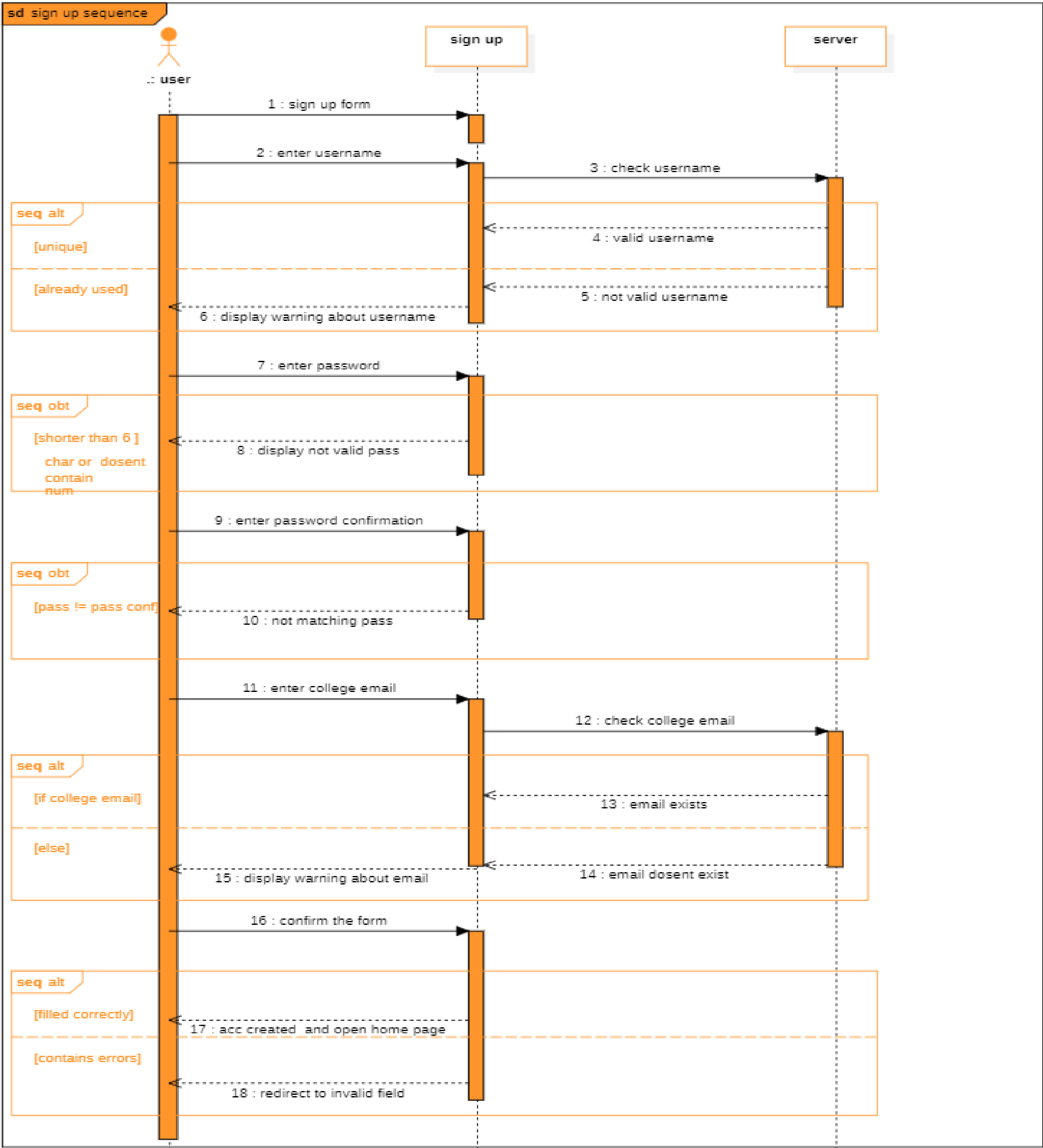
Mentor
form

Feedback



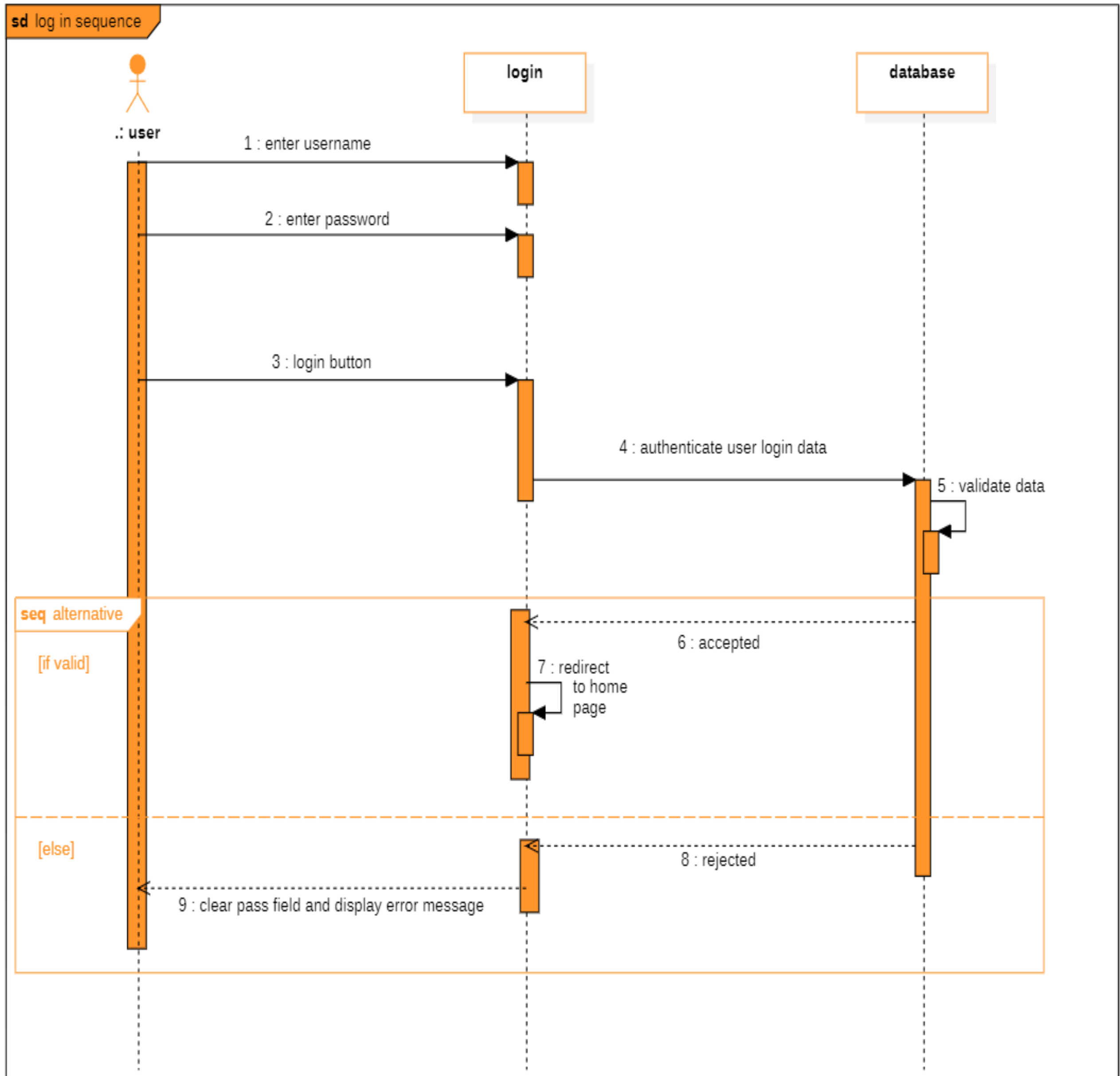
SEQUENCE DIAGRAM

1)Sign up:



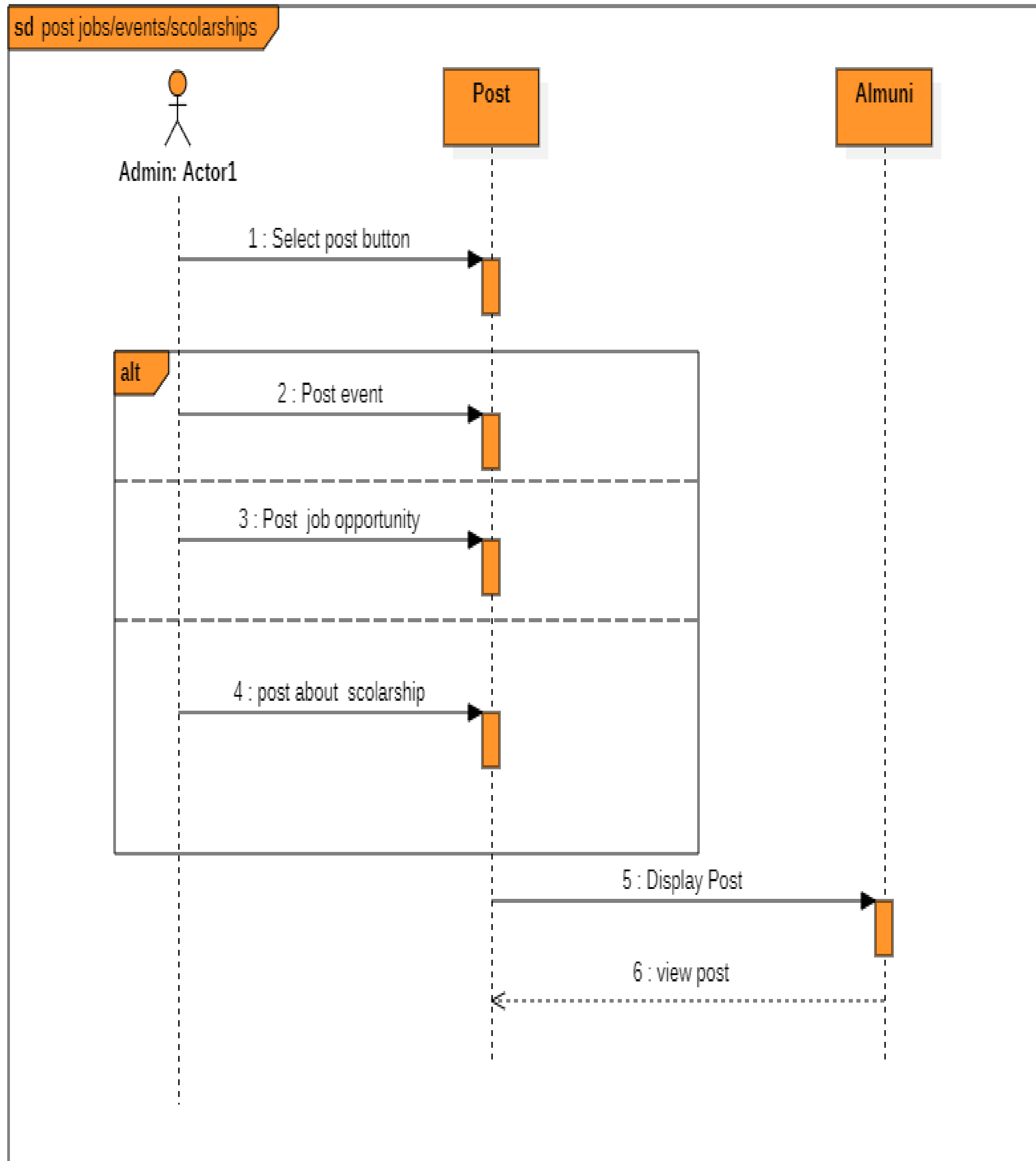
SEQUENCE DIAGRAM

2)Login:



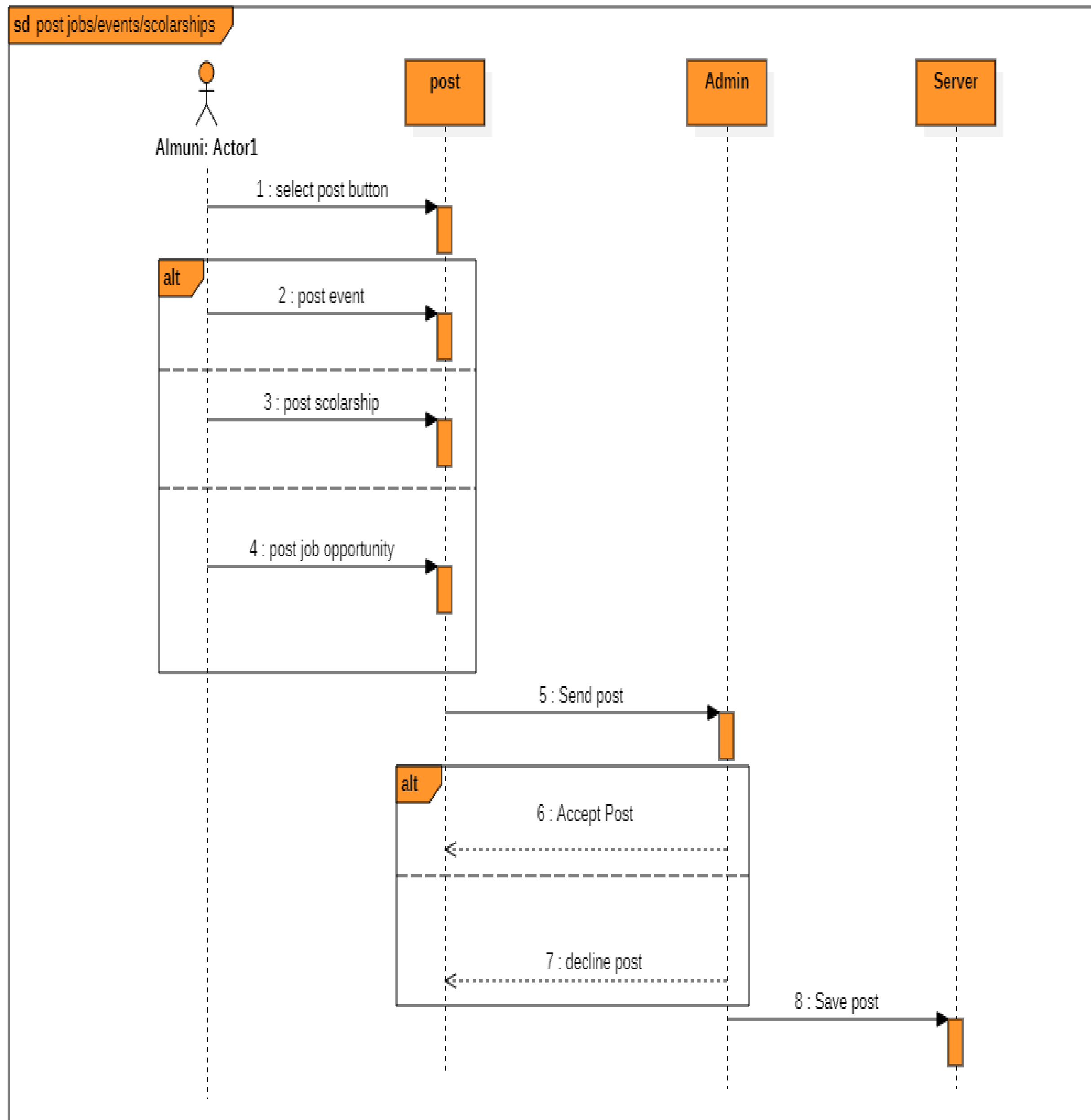
SEQUENCE DIAGRAM

3) Admin Post:



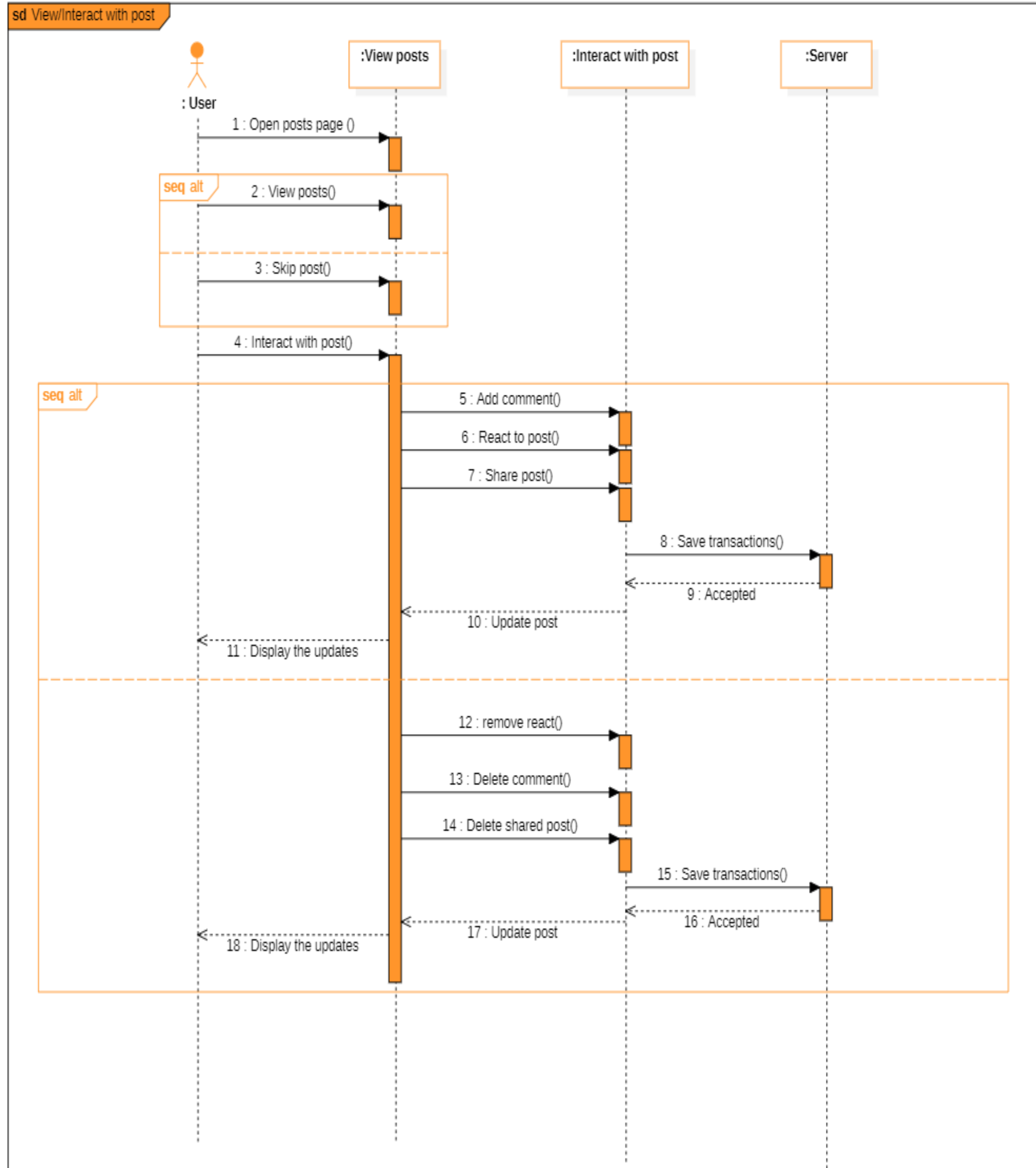
SEQUENCE DIAGRAM

4) Alumni Post:



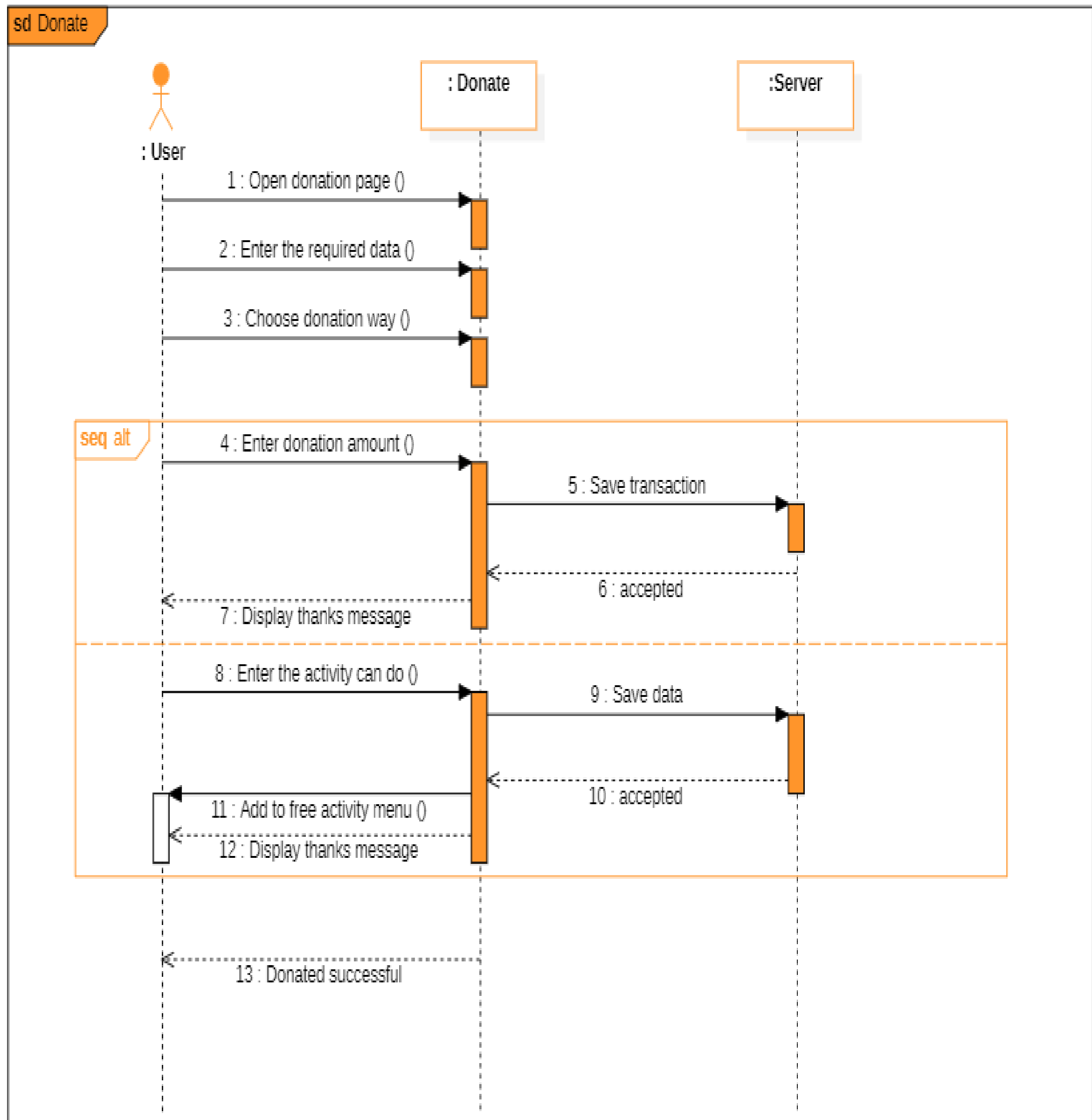
SEQUENCE DIAGRAM

5) Interact with post:



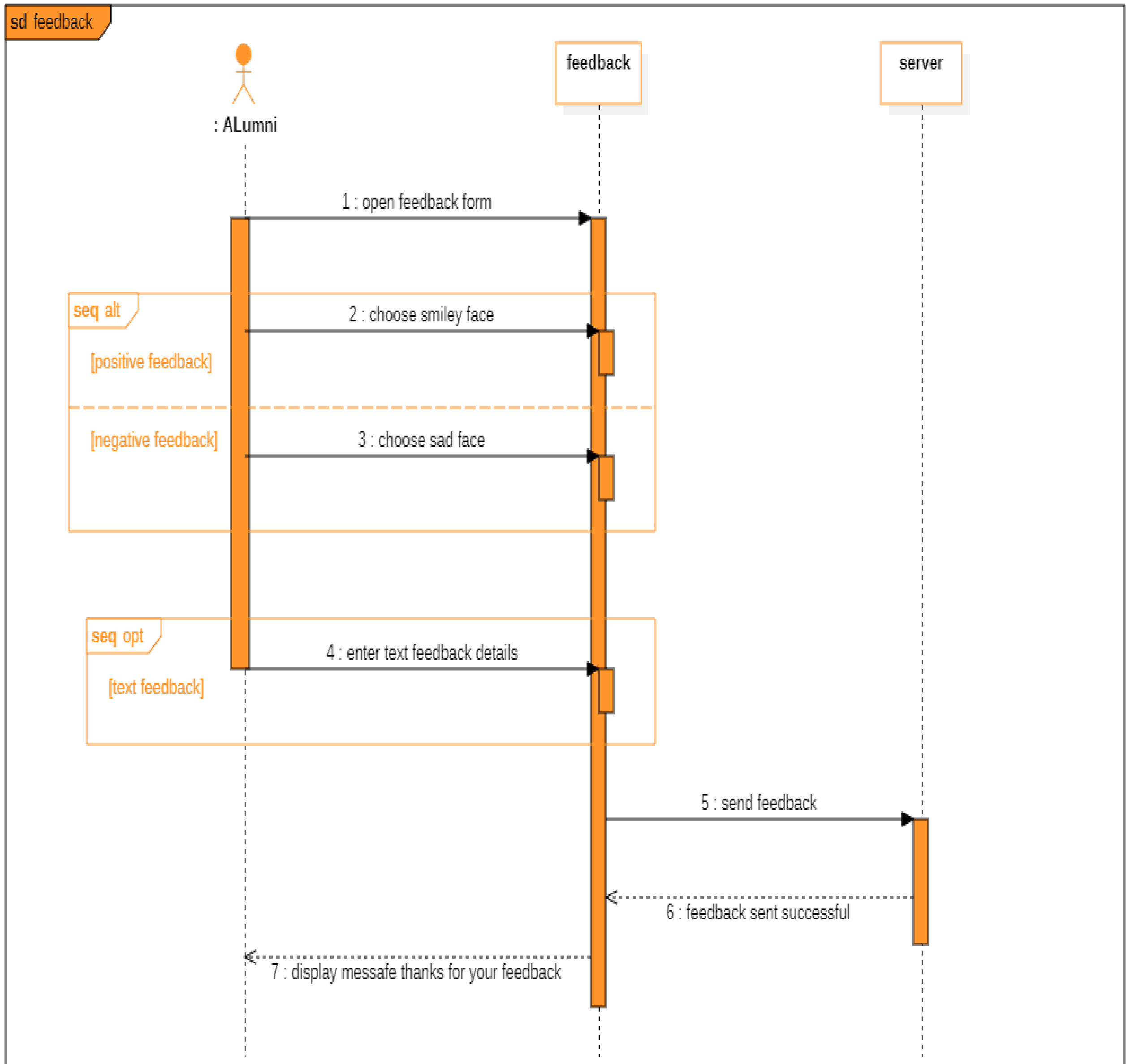
SEQUENCE DIAGRAM

6) Donate:



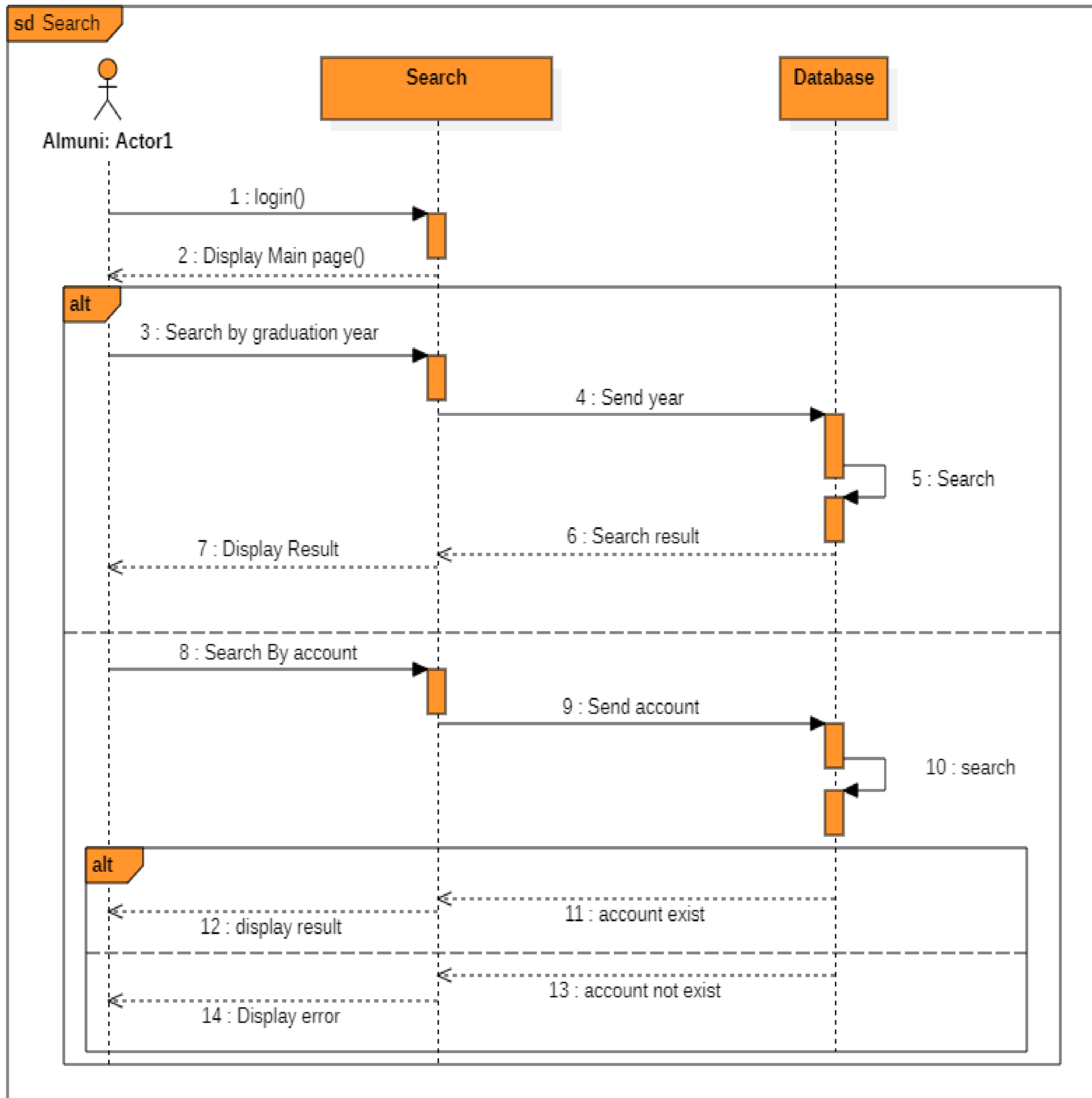
SEQUENCE DIAGRAM

7) Feedback:



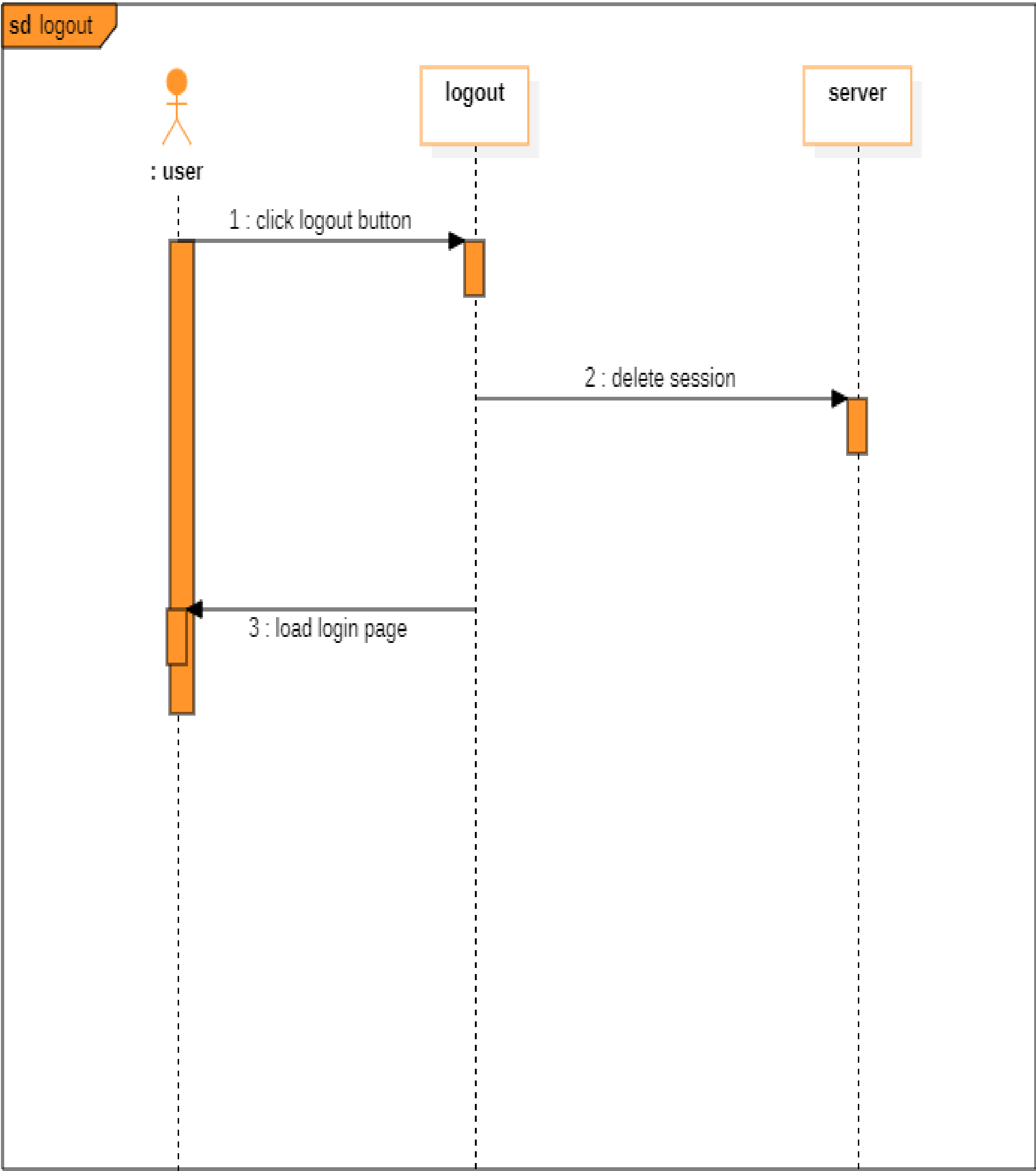
SEQUENCE DIAGRAM

8) Search:



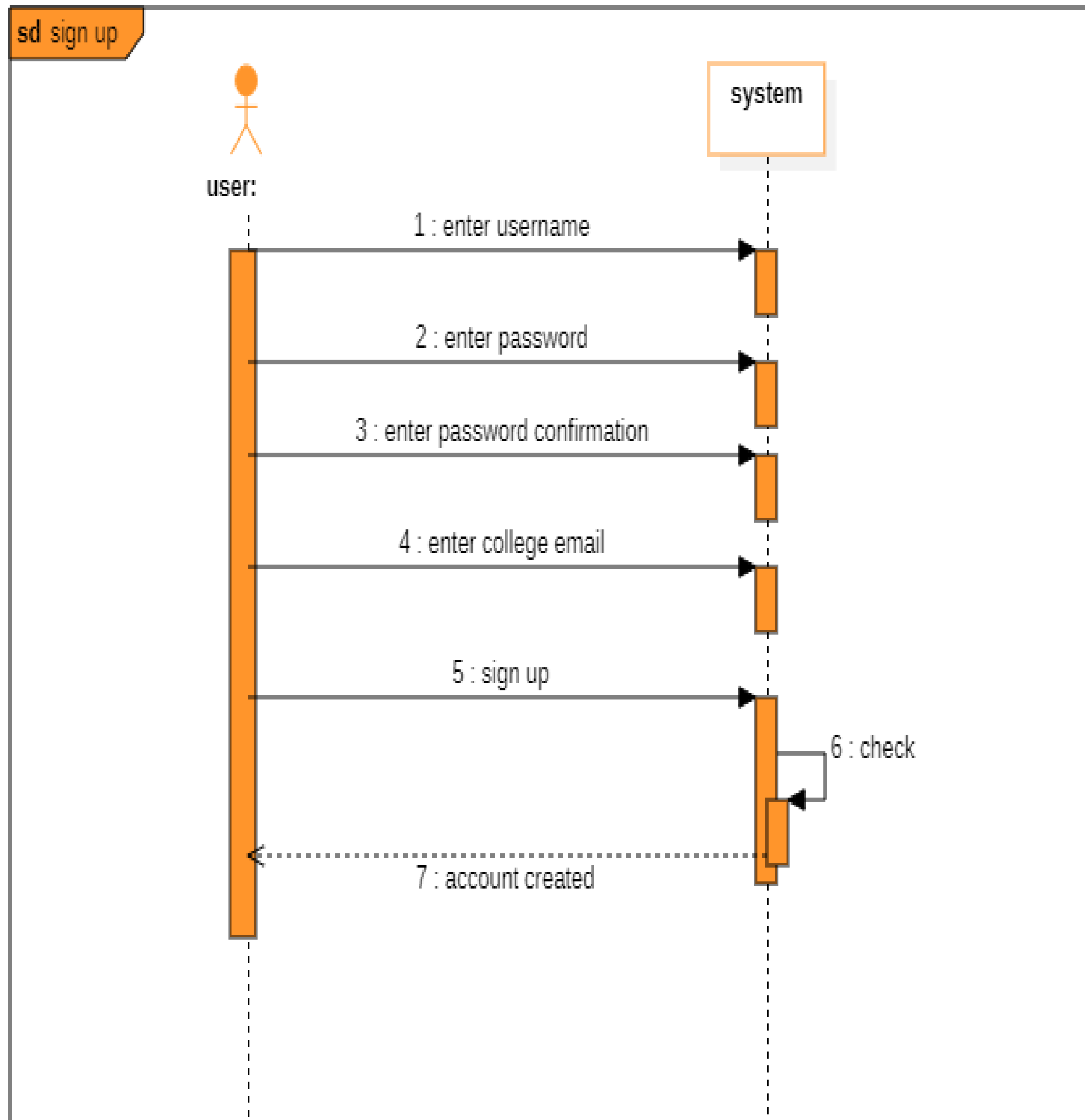
SEQUENCE DIAGRAM

9) Logout:



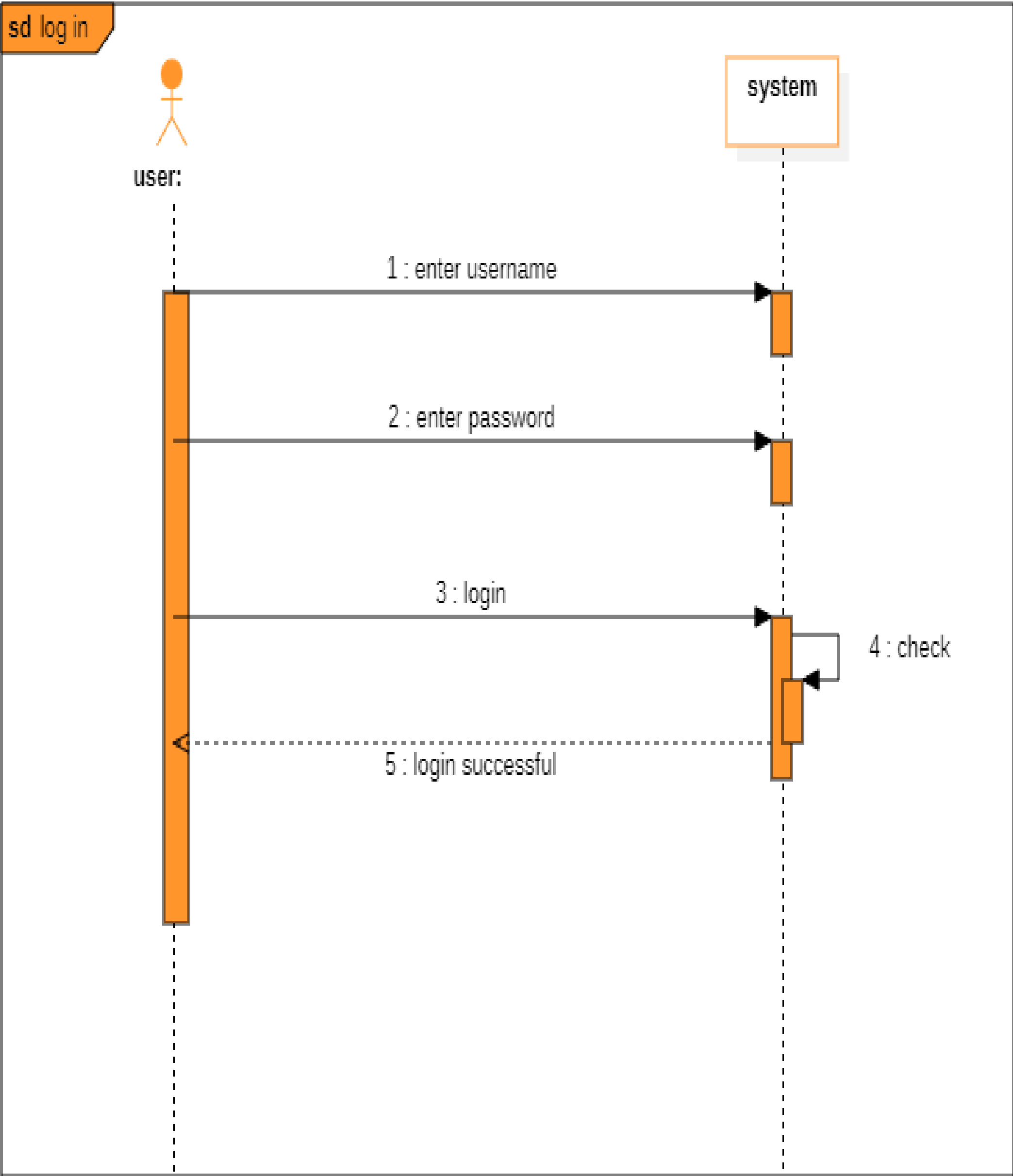
SYSTEM SEQUENCE DIAGRAM

1)Sign up:



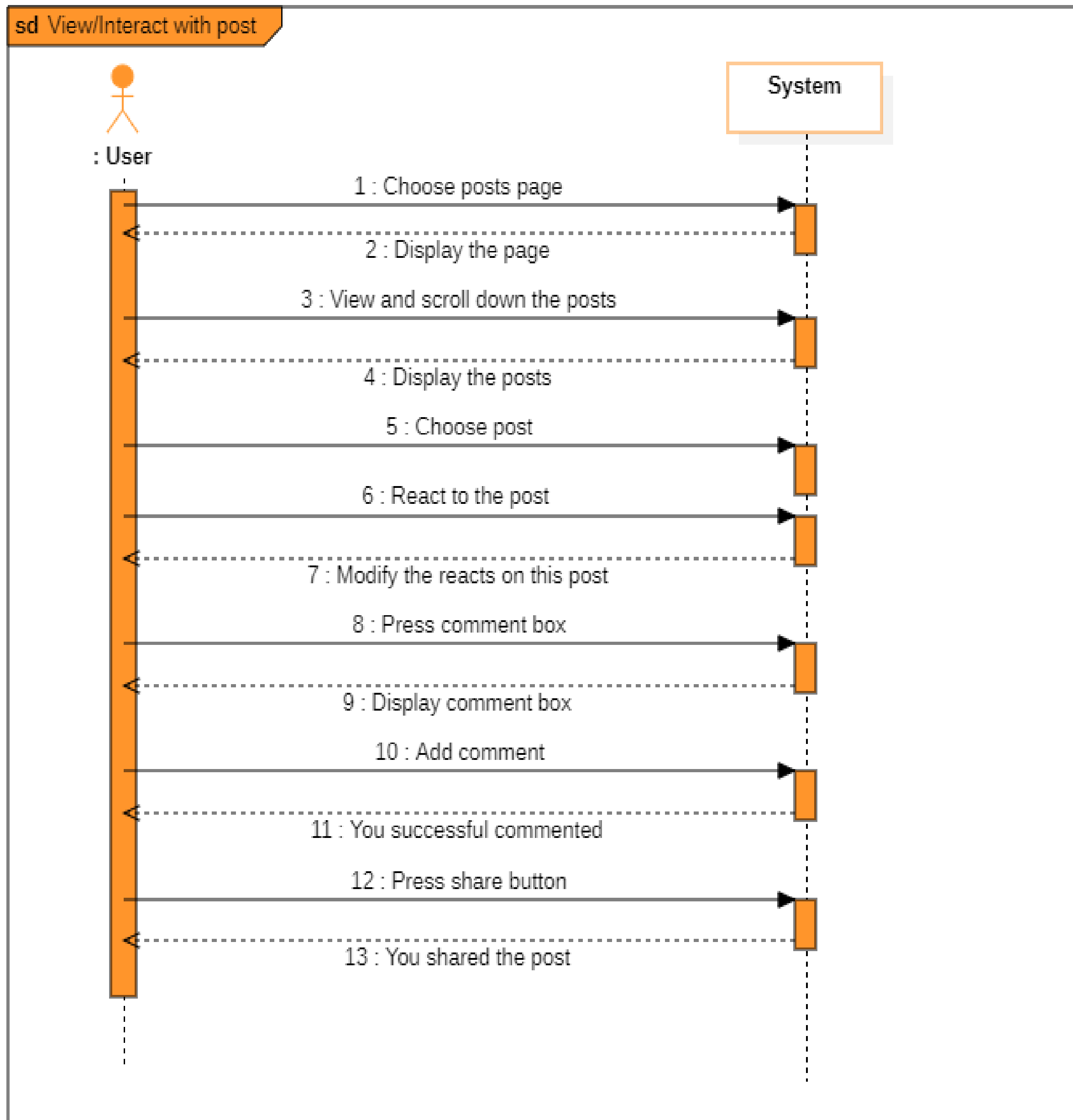
SYSTEM SEQUENCE DIAGRAM

2)Login:



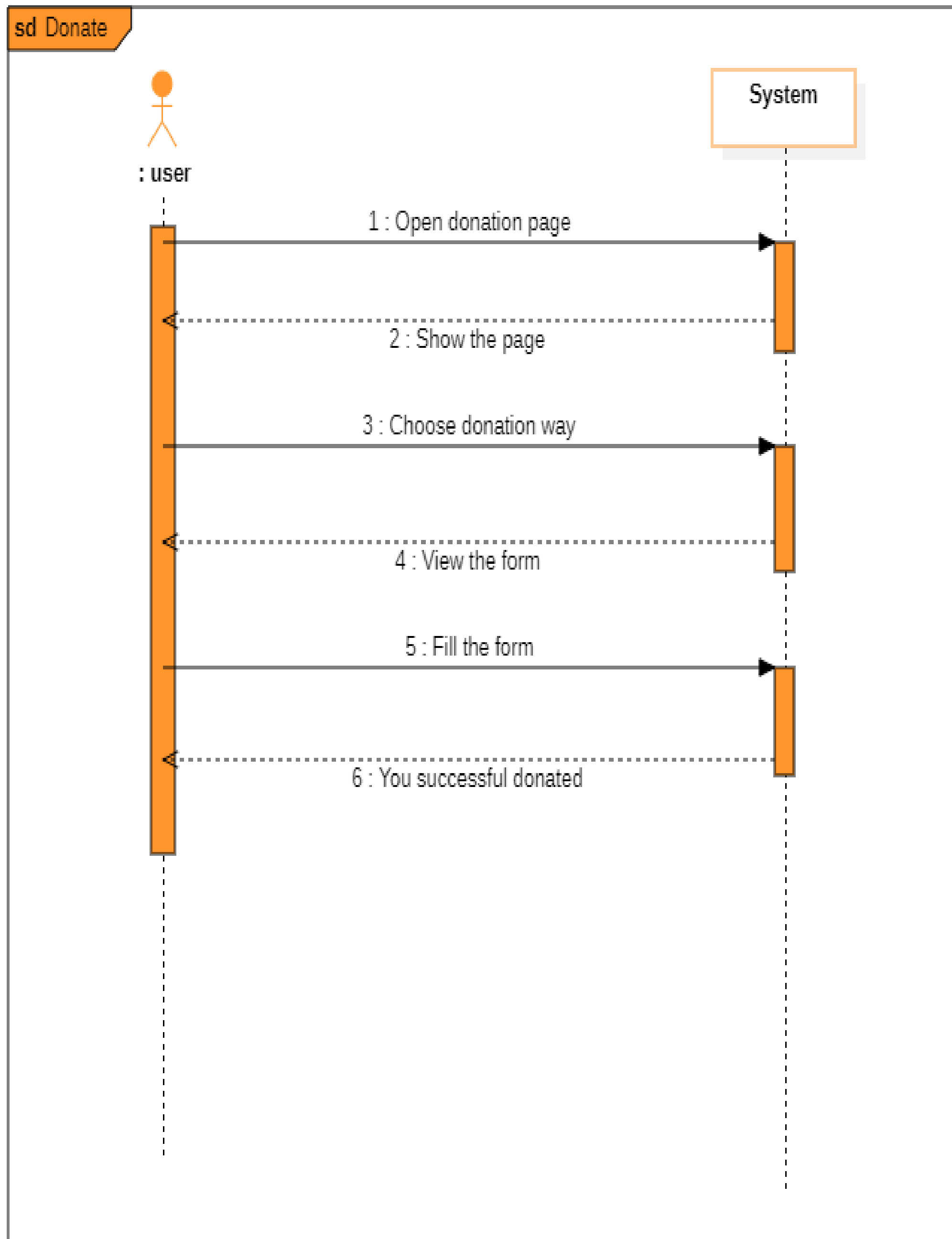
SYSTEM SEQUENCE DIAGRAM

3) Interact with post:



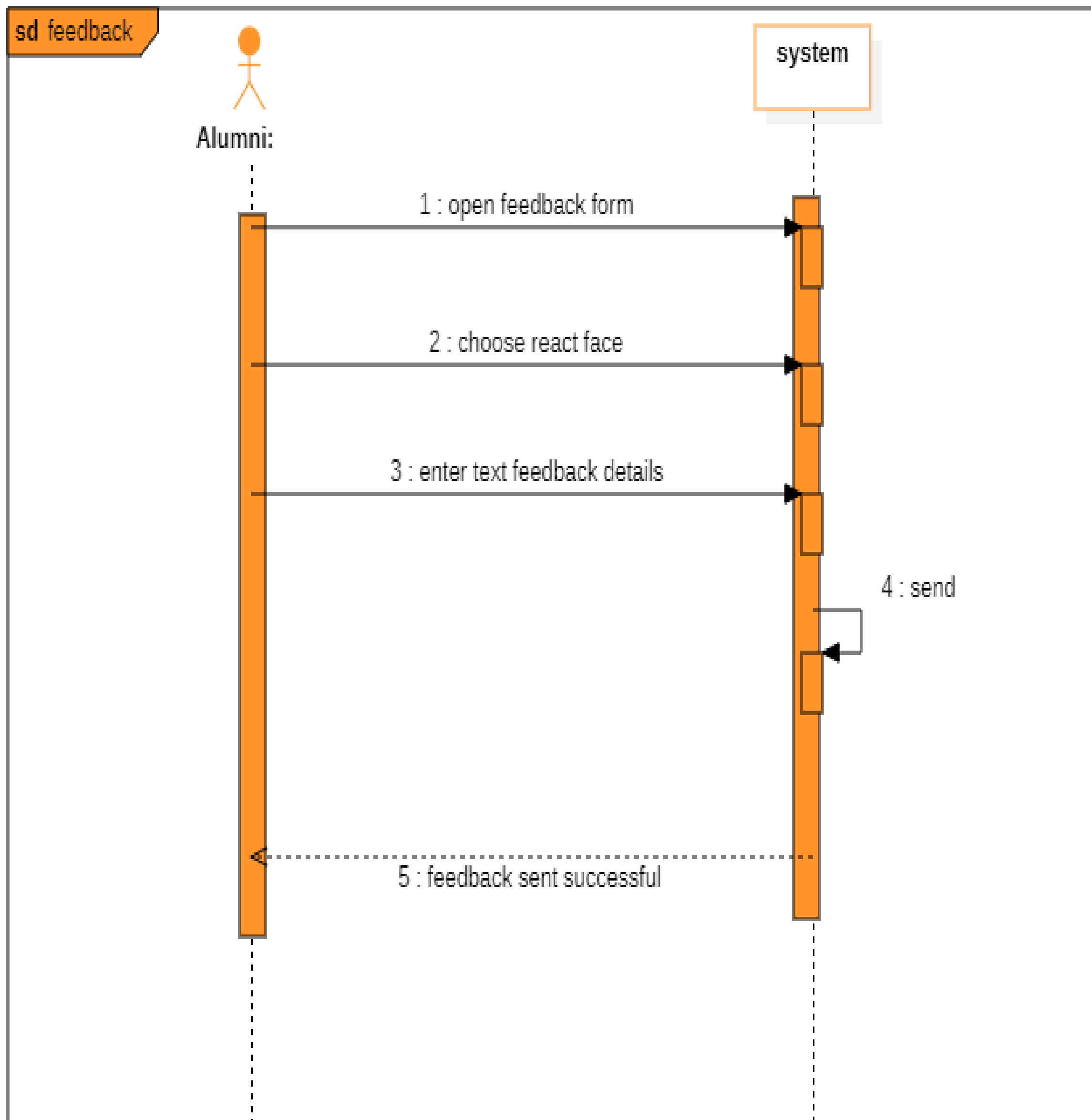
SYSTEM SEQUENCE DIAGRAM

4) Donate:



SYSTEM SEQUENCE DIAGRAM

5) Feedback:



SYSTEM SEQUENCE DIAGRAM

6) Logout:

