Educational Question-Answer Forum for Bangladesh

Software Project Lab-3

1/9/2019

Submitted by

Tulshi Chandra Das-811

Supervised by

Md. Saeed Siddik

Lecturer, IIT, DU

IIT
University of Dhaka

EDUCATIONAL QUESTION-ANSWER FORUM

SUBMITTED TO

SPL3 Committee
Institute of Information Technology
University of Dhaka

SUBMITTED BY

Tulshi Chandra Das
BSSE0811

1 September 2019
The Coordinators, SPL3
Institute of Information Technology
University of Dhaka.

Subject: Submission of term report on "Educational Question Answer Forum for Bangladesh"

Sir
With due respect, I am submitting the report on the above topic you have assigned to us. In this report, I have given my best effort albeit some shortcomings.
I earnestly hope that you would excuse my errors and oblige thereby.

Sincerely yours
Tulshi Chandra Das- BSSE0811   Supervisor

                                                                              Md. Saeed Siddik
4rd Year, 8th Semester, 8th Batch   Lecturer, IIT, DU
Institute of Information Technology
University of Dhaka

ACKNOWLEDGEMENT

ABSTRACT

The study is made for Educational Question Answer Forum for Bangladesh. The scope of the study is to analyze on Question Answer Forum for Bangladesh and design the SRS of this system. The object of this study is to develop an SRS (Software Requirements Specification and Analysis) of this project

Table of Contents

TABLES OF FIGURES

LIST OF TABLES

CHAPTER-01: INTRODUCTION

This chapter is a part of my software requirement specification for the project "Question Answer Forum for Bangladesh". In this chapter, I focus on the intended audience for this project.

1.1 PURPOSE

This document briefly describes the Software Requirement Analysis of Question Answer Forum for Bangladesh. It contains functional, non-functional and supporting requirements and establishes a requirements baseline for the development of the system. The requirements contained in the SRS are independent, uniquely numbered and organized by topic. The SRS serves as an official means of communicating user requirements to the developer and provides a common reference point for both the developer team and the stakeholder community. The SRS will evolve over time as users and developers work together to validate, clarify and expand its contents.

1.2 INTENDED AUDIENCE

This SRS is intended for several audiences including the customers as well as the project designers, developers. The customer will use this SRS to verify that the developer team has created a product that the customer finds acceptable. The designers will use this SRS as a basis for creating the system's design. The designers will continually refer back to this SRS to ensure that the system they are designing will fulfill the customer's demands. The developers will use this SRS as a basis for developing the system's functionality. The developers will link the requirements defined in this SRS to the software they create to ensure that they have created a software that will fulfill all of the customer's documented requirements. When portions of the software are complete, the developer will run their tests on that software to ensure that the software fulfills the requirements documented in this SRS. The testers will again run their tests on the entire system when it is complete and ensure that all requirements documented in this SRS have been fulfilled.

1.3 CONCLUSION

This analysis of the audience helped me to focus on the users who will be using my analysis. This overall document will help each and every person related to this project to have a better idea about the project.

CHAPTER-02: INCEPTION

2.1 INTRODUCTION

Inception is the beginning phase of requirements engineering. It defines how a software project gets started and what the scope and nature of the problem to be solved are. The goal of the inception phase is to identify concurrent needs and conflicting requirements among the stakeholders of a software project. At project inception, I establish a basic understanding of the problem, the people who want a solution, the nature of the solution that is desired and the effectiveness of preliminary communication and collaborations between the other stakeholders and the software team.

To establish the groundwork, I have worked with the following factors related to the inception phases:

- ➢ List of stakeholders
- ➢ Recognizing multiple viewpoints
- ➢ Working towards collaboration
- ➢ Requirements questionnaire

2.1.1 LIST OF STAKEHOLDERS

According to Sommerville and Sawyer [Som97], "Anyone who benefits in a direct or indirect way from the system which is being developed is a stakeholder." This implies that stakeholders include the end users of the developed software as well as the people whose activities might be influenced by the tool. Towards the end of inception, the list of stakeholders is usually larger as every stakeholder is allowed to suggest one or more individuals who might be probable stakeholders for the given problem.

To identify stakeholders, I consulted some teachers and students of some universities of Bangladesh and asked them the following questions:

I identified following stakeholders for my Educational Question Answer Forum:

- ➢ Inquirer
- ➢ Responder

Inquirer: Inquirer is a person who ask question to this forum.

Responder: Responder is a person who answer to a question

2.1.2 MULTIPLE VIEWPOINTS

Different stakeholders achieve different benefits from the system. Consequently, each of them has a different view of the system. So, I have to recognize the requirements from multiple points of view, as well as multiple views of requirements. Assumptions are given below:

USER VIEWPOINTS

- ➢ Creating Question
- ➢ Answer to question
- ➢ Voting/Downvoting to question/answer
- ➢ Comment System
- ➢ Profile view
- ➢ Search user
- ➢ Rating system

2.1.3 WORKING TOWARDS COLLABORATION

Each of the stakeholder constituencies (and non-stakeholder constituency) contributes to the requirement engineering process. The greater the numbers of interactions with multiple stakeholders, the higher is the probability of inconsistency, conflicts, and clashes of viewpoints. In such circumstances, requirement engineers finalize the requirements following some steps, which are listed below.

- ➢ Find the common and conflicting requirements
- ➢ Categorize them
- ➢ List the requirements based on stakeholder's priority
- ➢ Make a final decision about requirements

COMMON REQUIREMENTS

- ➢ Creating Question

- ➢ Answering to question
- ➢ Commenting to question
- ➢ Rating system

CONFLICTING REQUIREMENTS

- ➢ Profile view

FINAL REQUIREMENTS

- ➢ Creating Question
- ➢ Answer to question
- ➢ Voting/Downvoting to question/answer
- ➢ Comment System
- ➢ Profile view
- ➢ Search user
- ➢ Message to teacher

2.1.4 REQUIREMENTS QUESTIONNAIRE

In requirements engineering, the involved individuals can be broadly divided into two clusters: the developers and the stakeholders. Coming from different backgrounds, it will be obvious that these two parties will have different points of views regarding the problem. The stakeholders have more knowledge on facing the problem. Meanwhile, the developers are experienced in providing computerized solutions. Thus, in order to obtain an efficient solution to the problem, it is important to 'loosen up' or 'break the ice' between the two groups.

Following the ideal guidelines of requirement engineering, some context-free questions were asked. The context-free questions help to throw light on the stakeholders of the project. The next set of questions includes the context itself so that a better understanding of the problem is obtained. The stakeholder is encouraged to voice out his/her opinions about an alternate solution and also provide recommendations to the developer's suggestions. The final set of questions focuses on the communication activity itself.

2.2 CONCLUSION

The Inception phase helped me to establish a basic understanding about the Educational Question Answer Forum, identify the stakeholders who will be benefited if this system becomes automated, define the nature of the system and the tasks done by the system, and establish a preliminary communication with my stakeholders.

In my project, I have established a basic understanding of the problem, the nature of the solution that is desired and the effectiveness of preliminary communication and collaboration between the stakeholders and the software team. More studies and communication will help both sides (developer and client) to understand the future prospect of the project. I believe that the full functioning document will help me to define that future prospect

CHAPTER-03: ELICITATION

After discussing on the inception phase, I need to focus on Elicitation phase. So, this chapter specifies the Elicitation phase.

3.1 INTRODUCTION

Requirements Elicitation is a part of requirements engineering that is the practice of gathering requirements from the users, customers, and other stakeholders. I have faced many difficulties, like understanding the problems, making questions for the stakeholders, problems of scope and volatility. Though it is not easy to gather requirements within a very short time, I have surpassed these problems in an organized and systematic manner.

3.2 ELICITING REQUIREMENTS

I have seen Question and Answer (Q&A) approach in the previous chapter, where the inception phase of requirement engineering has been described. Requirements Elicitation (also called requirements gathering) combines problem solving, elaboration, negotiation and specification. The collaborative working approach of the stakeholders is required to elicit the requirements. I have finished the following tasks for eliciting requirements-

➢ Collaborative requirements gathering
➢ Quality function deployment
➢ Usage scenario
➢ Elicitation work products

3.2.1 COLLABORATIVE REQUIREMENTS GATHERING

I have met with stakeholders in the inception phase. The stakeholders are Inquirer and Responder. Many different approaches to collaborative requirements gathering have been proposed by the stakeholders. To solve this problem, I have met with the stakeholders again to elicit the requirements. A slightly different scenario from these approaches has been found.

3.2.2 PROBLEM IN THE SCOPE

A number of the problems were encountered in the course of preparing the software requirement specification and analysis of the Question-Answer Forum.

What will not be done:

  ➢ In the forum user cannot upload any document file (docx, pdf etc.)
  ➢ The forum is only for educational purpose not other.

What will be done:

  ➢ In this forum image can be uploaded
  ➢ User can give answer of maximum 5000 character.

3.2.3 QUALITY FUNCTION DEPLOYMENT

Quality Function Deployment (QFD) is a technique that translates the needs of the customer into technical requirements for software. It concentrates on maximizing customer satisfaction from the software engineering process. So, I have followed this methodology to identify the requirements for the project. The requirements, which are given below, are identified successfully by the QFD.

3.2.3.1 NORMAL REQUIREMENTS

Normal requirements are generally the objectives and goals that are stated for a product or system during meetings with the stakeholders. The presence of these requirements fulfills stakeholders' satisfaction. The normal requirements of my project-

  ➢ Answering to question
  ➢ Creating a question
  ➢ Commenting to answer or a question
  ➢ Upvote/Downvote to question/answer

3.2.3.2 EXPECTED REQUIREMENTS

  ➢ Question/Answer modification deletion

➢ User-friendly

➢ Efficient question recommendation system

➢ Efficient question similarity check

3.2.3.3 EXCITING REQUIREMENTS

➢ Badge popularity system in user rating

3.2.4 USAGE SCENARIO

Question-Answer Forum for Bangladesh is an automated system for the following purposes:

➢ Question

➢ Answer

➢ Comment management

➢ User management

3.2.4.1 QUESTION MANAGEMENT

QUESTION MANAGEMENT

User will create a question going throw a set of defined steps. First user will provide the category of his/her question. Then user will give the title of the question. Then he/she will provide the tags related the topics of question. In the next step the system will show some similar question related to his asking question. If user not satisfied user will go to next step and give description and can upload image. At last user will submit question. To submit user must have to be signed. If he/she is new user he/she signup using OAuth auth0 authentication system.

RECCOMMENDING QUESTION TO USERS

In this forum different user from different will create question on different study topics. So, all user will not be recommended to all types of question. The system will use user profile data and activity to recommend questions.

3.2.4.2 ANSWER MANAGEMENT

Any user can answer to any question. User can upload image with his/her answer. User will be able to write maximum 5000-character description to an answer.

3.2.4.3 COMMENT MANAGEMENT

Any user can comment to an answer/question. The question or answer is considered the comment source. User will have to gain minimum 50 reputation to comment. Any user can upvote/downvote to a comment. Any user can mark the comment as usable or as problematic. If the negative rating crosses the limit of 50 then comment will be blocked. User reputation also increase/decrease based on user reaction.

3.2.4.4 USER MANAGEMENT

Managing user and proofing their activity is one of the major parts of this project. This system will store the activities like answers, question, popularity, following tags, answer count to corresponding to tags, question count corresponding to tags. Popularity is the exciting thing for users to be motivated active in question-answer discussion. User will gain more popularity based on upvote to his/her question or answer or comment. He/she will gain different badge (silver, gold, platinum) at different level of popularity. For each event or user will get notification status. As a result, user will appear in the search of users more in front position. User will use the online authentication system to be authenticated(auth0). User will be identified by a unique id.

3.2.5 ELICITATION WORK PRODUCT

At first, I have to know whether the output of the Elicitation task may vary because of the dependency on the size of the system or the product to be built. Here, the Elicitation work product includes:

➢ Making a statement of my requirements for the Educational Question Answer Forum.

➢ Making a bounded statement of scope for my system.

- ➢ Making a list of users and other stakeholders who participated in the requirements elicitation.

- ➢ A set of usage scenarios that provide insight into the use of the system.

- ➢ Description of the system's technical environment

CHAPTER-04: SCENARIO BASED MODELING

This chapter describes the Scenario-Based Model for the project.

4.1 INTRODUCTION

Although the success of a computer-based system or product is measured in many ways, user satisfaction resides at the top of the list. If I understand how end users (and other actors) want to interact with a system, my software team will be better able to properly characterize requirements and build meaningful analysis and design models. Hence, requirements modeling begins with the creation of scenarios in the form of Use Cases, activity diagrams.

4.2 DEFINITION OF USE CASE

A Use Case captures a contract that describes the system behavior under various conditions as the system responds to a request from one of its stakeholders. In essence, a Use Case tells a stylized story about how an end user interacts with the system under a specific set of circumstances. A Use Case diagram simply describes a story using corresponding actors who perform important roles in the story and makes the story understandable for the users. The first step in writing a Use Case is to define that set of "actors" that will be involved in the story. Actors are the different people that use the system or product within the context of the function and behavior that is to be described. Actors represent the roles that people play as the system operators. Every user has one or more goals when using the system.

PRIMARY ACTOR

Primary actors interact directly to achieve required system function and derive the intended benefit from the system. They work directly and frequently with the software.

SECONDARY ACTOR

Secondary actors support the system so that primary actors can do their work. They either produce or consume information.

4.3 USE CASE DIAGRAM

Use case diagrams give the non-technical view of the overall system.

4.3.1 LEVEL- 0 USE CASE DIAGRAM-EQAF(EDUCATIONAL QUESTION-ANSWER FORUM)
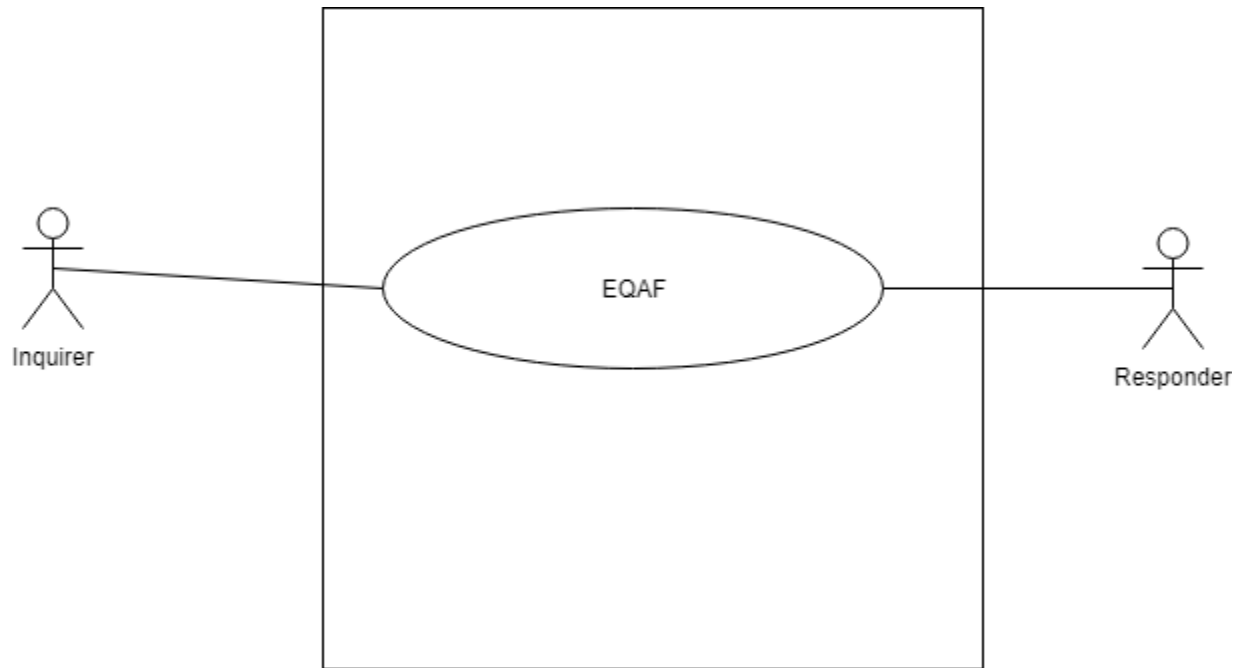
Figure 1: level 0 use case diagram- EQAF

Name: Educational Question-Answer Forum

Primary actor: Inquirer, Responder

Secondary actor: N/A

DESCRIPTION OF USE CASE DIAGRAM LEVEL-0

After analyzing user story, I found five actors who will directly use the system as a system operator. Primary actors are those who will play action and get a reply from the system whereas secondary actors only produce or consume the information.

Following the actors of "Educational Question-Answer Forum":

4.3.2 LEVEL -1 USECASE DIAGRAM-SUBSYSTEM OF EQAF
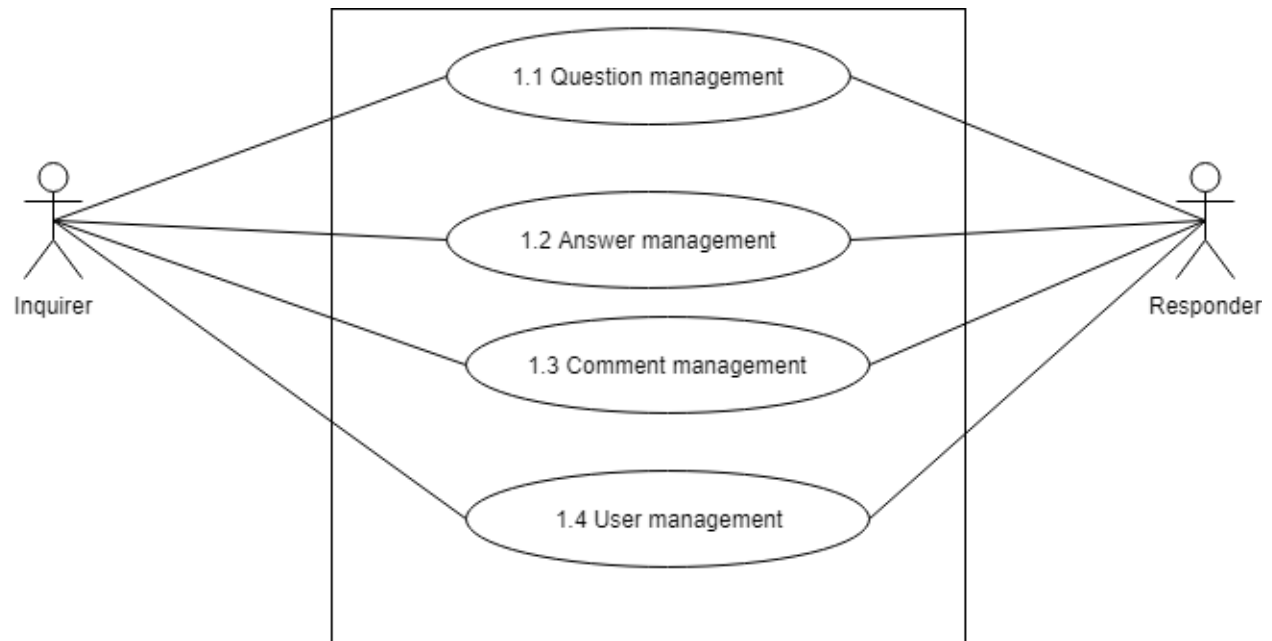
Figure-2: Figure 2level 1 use case diagram - Subsystem

Name: Subsystem of EQAF

Primary actor: Inquirer, Responder

Secondary actor: N/A

There are 4 subsystems in the Educational Question Answer Forum. They are-

➢ Question Management

➢ Answer Management

➢ Comment Management

➢ User Management

4.3.3 LEVEL- 1.1 USE CASE DIAGRAM- QUESTION MANAGEMENT
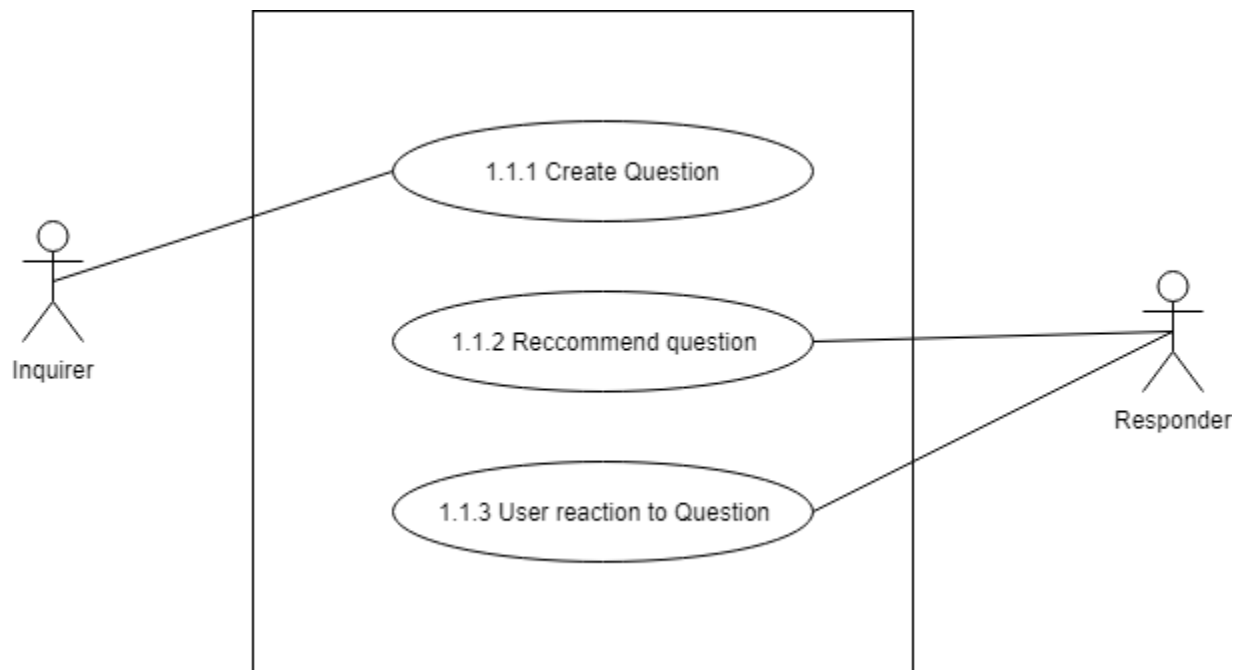
Figure-3: Figure 3level 1.1 use case diagram – Question management

Name: Question management

Primary actor: Inquirer, Responder

Secondary actor: N/A

DESCRIPTION OF LEVEL- 1.1 USE CASE DIAGRAM-

Creating question refers the process of creating question. User will create a question following some defined steps. After creating question system will recommend the asked question to appropriate user. Users can react by giving upvote or downvote to the question. If downvote count cross a specific limit question will be blocked. The subsystems are:

➢ Create question
➢ Recommend question
➢ User reaction to question

1.1.1 CREATE QUESTION

- ➢ Primary actor: Inquirer
- ➢ Secondary actor: N/A

INQUIRER ACTION/REPLY

- ➢ Action: Inquirer will ask the question.
- ➢ Reply: System will recommend the question to appropriate users.

1.1.2 RECOMMEND QUESTION

- ➢ Primary actor: N/A
- ➢ Secondary actor: Responder

RESPONDER ACTION/REPLY

- ➢ Action: Responder will view the question.
- ➢ Reply: Responder can respond to the question

1.1.3 USER REACTION TO QUESTION

- ➢ Primary actor: Users
- ➢ Secondary actor: N/A

RESPONDER ACTION/REPLY

- ➢ Action: Responder will upvote/downvote to question
- ➢ Reply: Inquirer will be notified of it

4.3.4 LEVEL -1.2 USE CASE DIAGRAM- ANSWER MANAGEMAGEMENT

Figure 4: level 1.2 use case diagram- Answer management

Name: Answer management of EQAF

Primary actor: Responder, Inquirer

Secondary actor: N/A

DESCRIPTION OF LEVEL- 1.2 USE CASE DIAGRAM-

There are 2 subsystems in Answer management subsystem. These are-

➢ Answer to question
➢ User reaction to answer

1.2.1 ANSWER TO QUESTION

➢ Primary actor: Responder
➢ Secondary actor: N/A

RESPONDER ACTION/REPLY

➢ Action: Responder answer to question.

➢ Reply: Answer will be view by other users.

1.2.2 USER REACTION TO ANSWER

➢ Primary actor: Responder, Inquirer

➢ Secondary actor: N/A

INQUIRER ACTION/REPLY

➢ Action: Inquirer can vote/downvote and identify as accepted answer

➢ Reply: Reputation of responder will increase/decrease.

RESPONDER REACTION

➢ Action: Other Responders can vote/downvote to answer

➢ Reply: Reputation of responder will increase/decrease.

4.3.5 LEVEL -1.3 USE CASE DIAGRAM- COMMENT MANAGEMENT
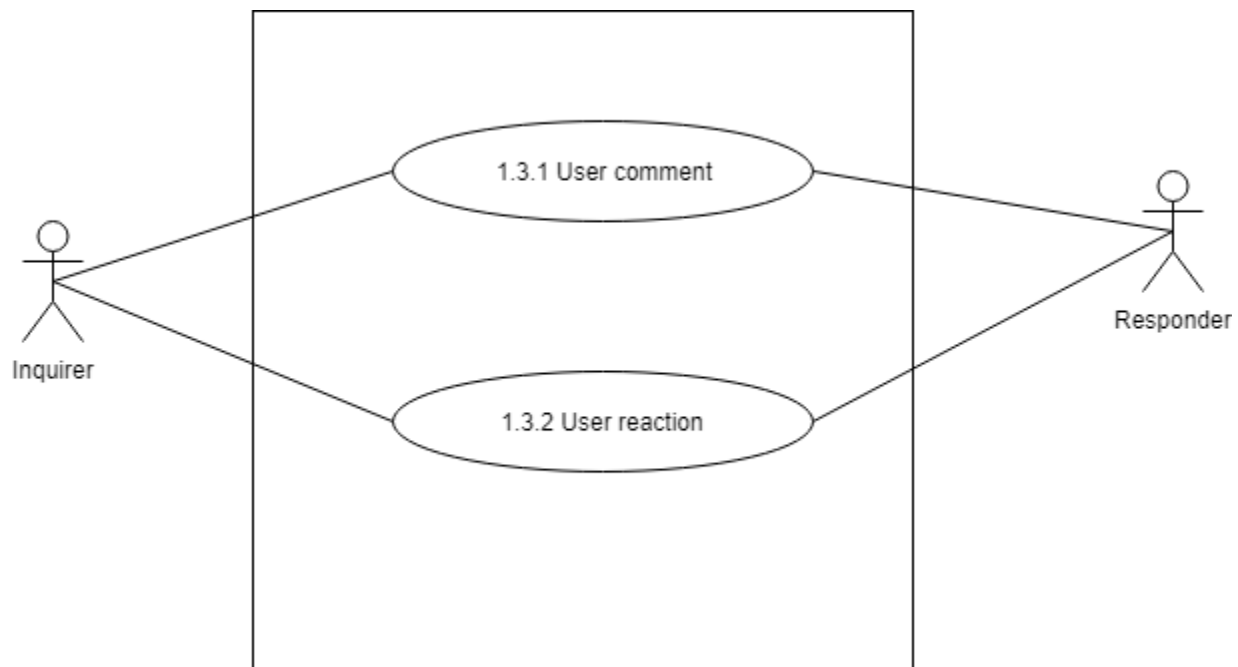
Figure 5: level 1.3 use case diagram- Comment management

Name: Comment management of EQAF

Primary actor: Inquirer, Responder

Secondary actor: N/A

DESCRIPTION OF LEVEL -1.3 USE CASE DIAGRAM

There are 2 subsystems in comment management subsystem. These are-

- ➢ User comment
- ➢ User reaction

1.3.1 USER COMMENT

- ➢ Primary actor: Inquirer, Responder
- ➢ Secondary actor: N/A

INQUIRER/RESPONDER ACTION/REPLY

- ➢ Action: A user can comment to others answer/question
- ➢ Reply: Comment will be seen by other

1.3.2 USER REACTION

- ➢ Primary actor: Inquirer, Responder
- ➢ Secondary actor: N/A

INQUIRER/RESPONDER ACTION/REPLY

- ➢ Action: User can mark the comment as useful
- ➢ Reply: Reputation of commenter will change
- ➢ Action: User can mark the comment as problematic
- ➢ Reply: Reputation of commenter will change and comment will be block after crossing a limit of flag

4.3.6 LEVEL -1.4 USE CASE DIAGRAM -USER MANAGEMENT

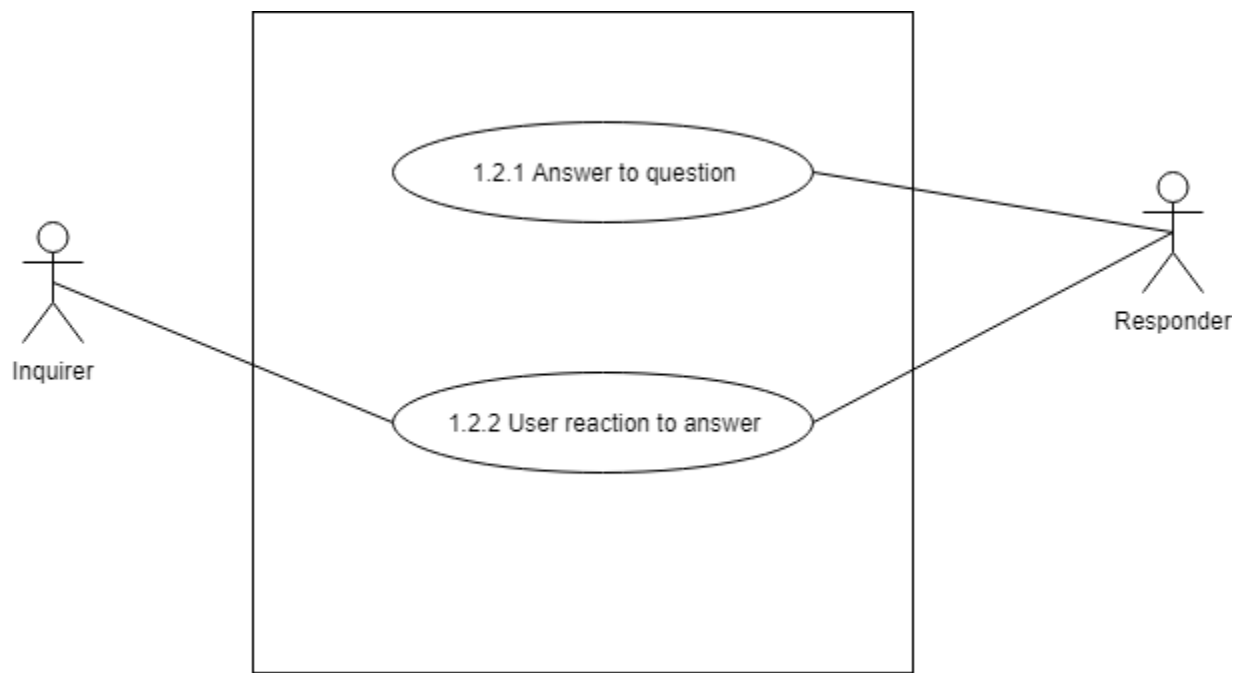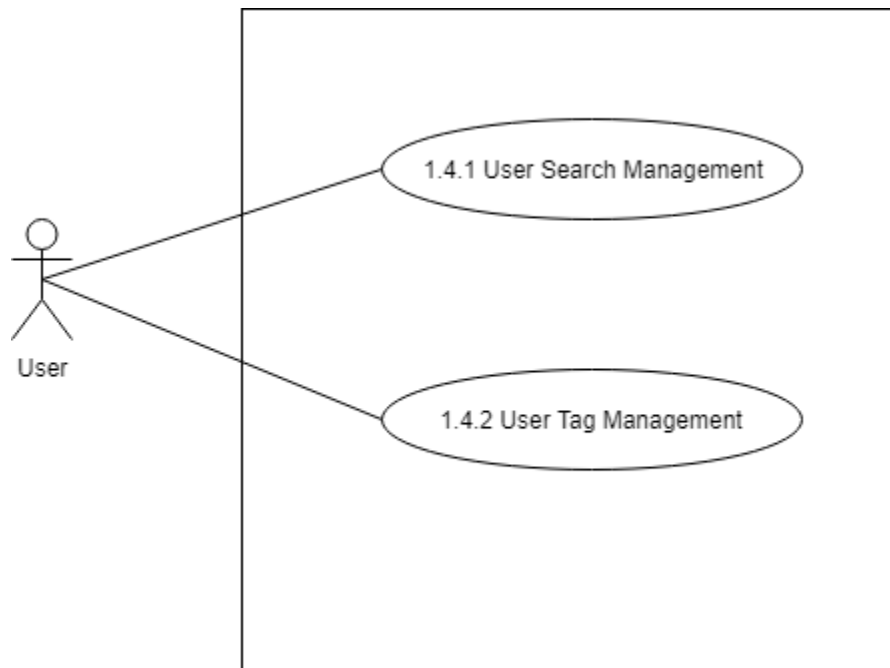*Figure 6: level 1.4 use case diagram- user management*

Name: User Management of EQAF

Primary actor: User

Secondary actor: N/A

DESCRIPTION OF LEVEL- 1.4 USE CASE DIAGRAM

There are 2 subsystems in comment management subsystem. These are-

- ➢ User search management
- ➢ User tag management

1.4.1 USER SEARCH MANAGEMENT

- ➢ Primary actor: User
- ➢ Secondary actor: N/A

USER ACTION/REPLY

- ➤ Action: A user can search or filter users by reputation, tag, location.
- ➤ Reply: Search result

4.4 ACTIVITY DIAGRAMS

ACTIVITY DIAGRAM: CREATE QUESTION
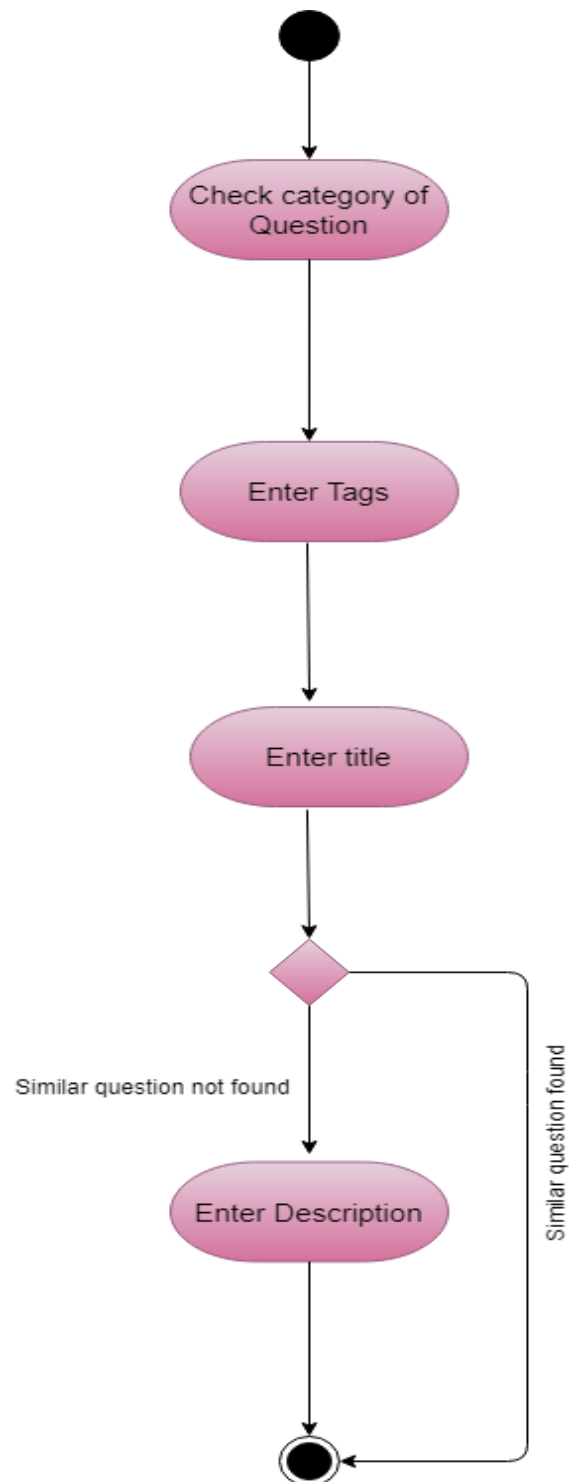
Figure 7: Level 1.1 Activity diagram – Create question

ACTIVITY DIAGRAM: USER REACTION
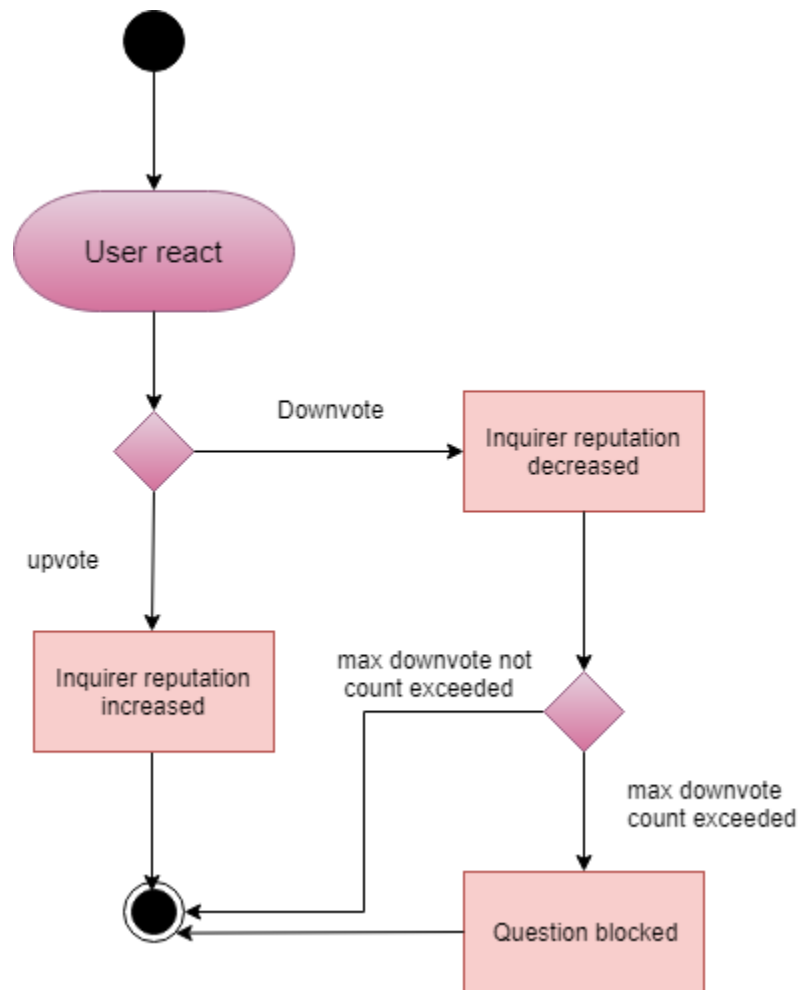
Figure 8: Level 1.1.1 Activity diagram – User Reaction

ACTIVITY DIAGRAM: ANSWER TO QUESTION
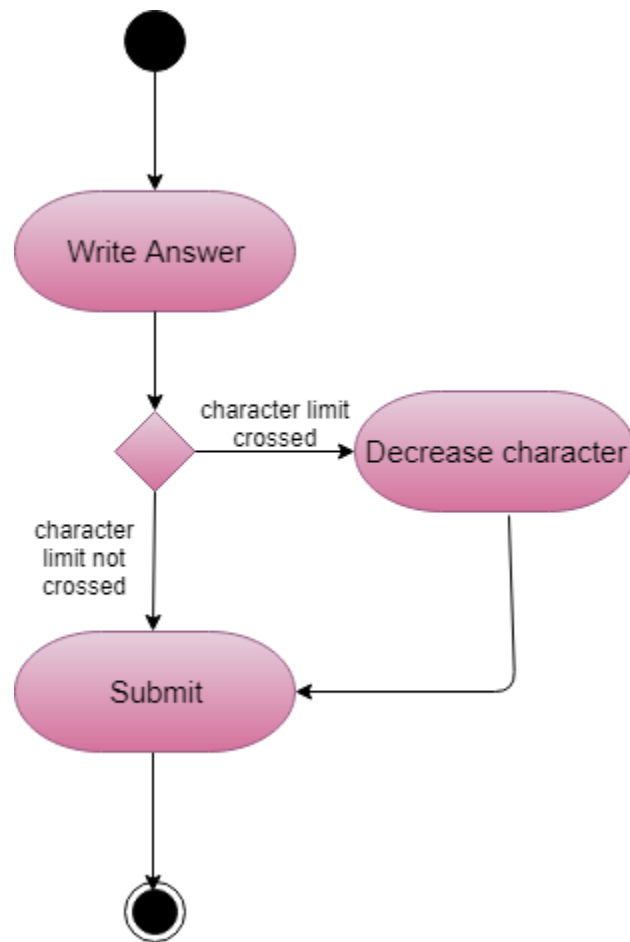
Figure 9: Activity diagram of Answer to question

ACTIVITY DIAGRAM: USER REACTION TO ANSWER
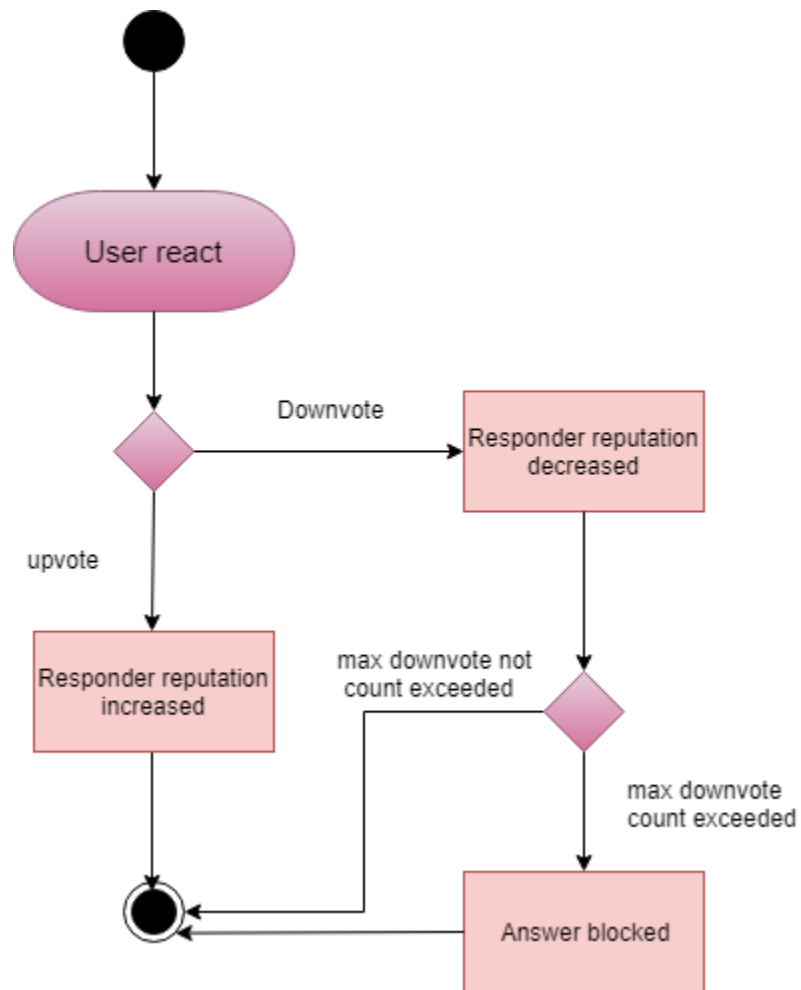
Figure 10: Level 1.1.2 Activity diagram – User reaction to answer.

ACTIVITY DIAGRAM: USER COMMENT
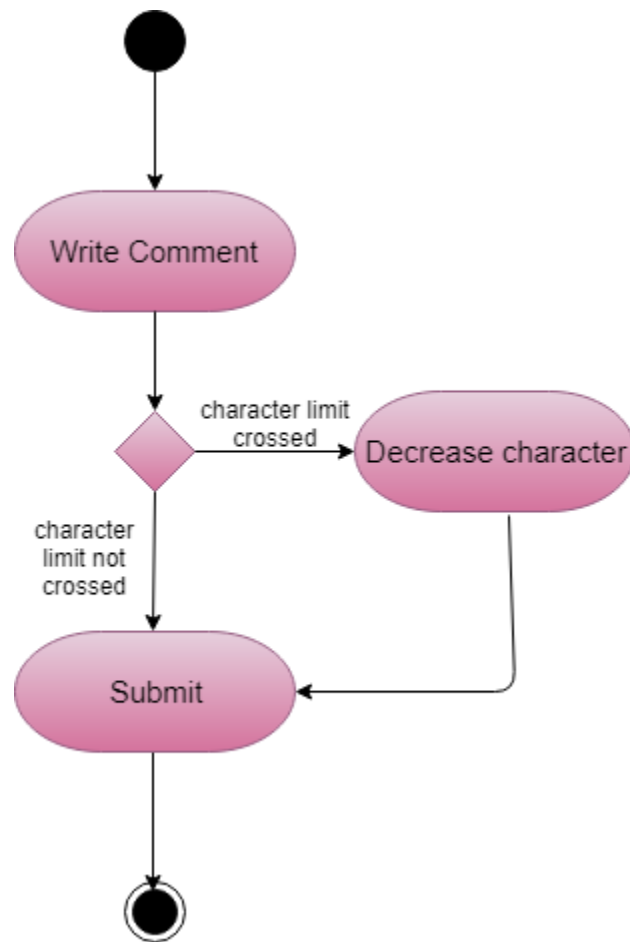
Figure 11: Level 1.1.3 Activity diagram –User comment

ACTIVITY DIAGRAM: USER REACTION TO COMMENT
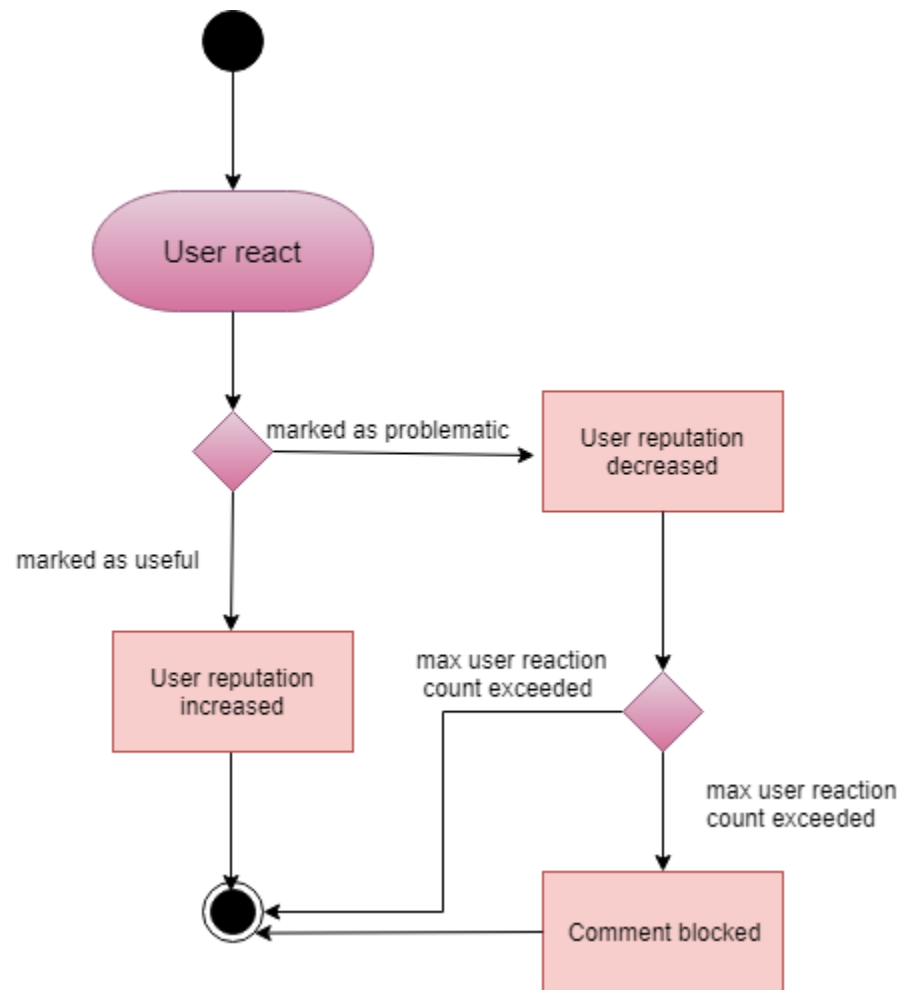
Figure 12: Level 1.2 Activity diagram – User Reaction to Comment

ACTIVITY DIAGRAM: USER SEARCH
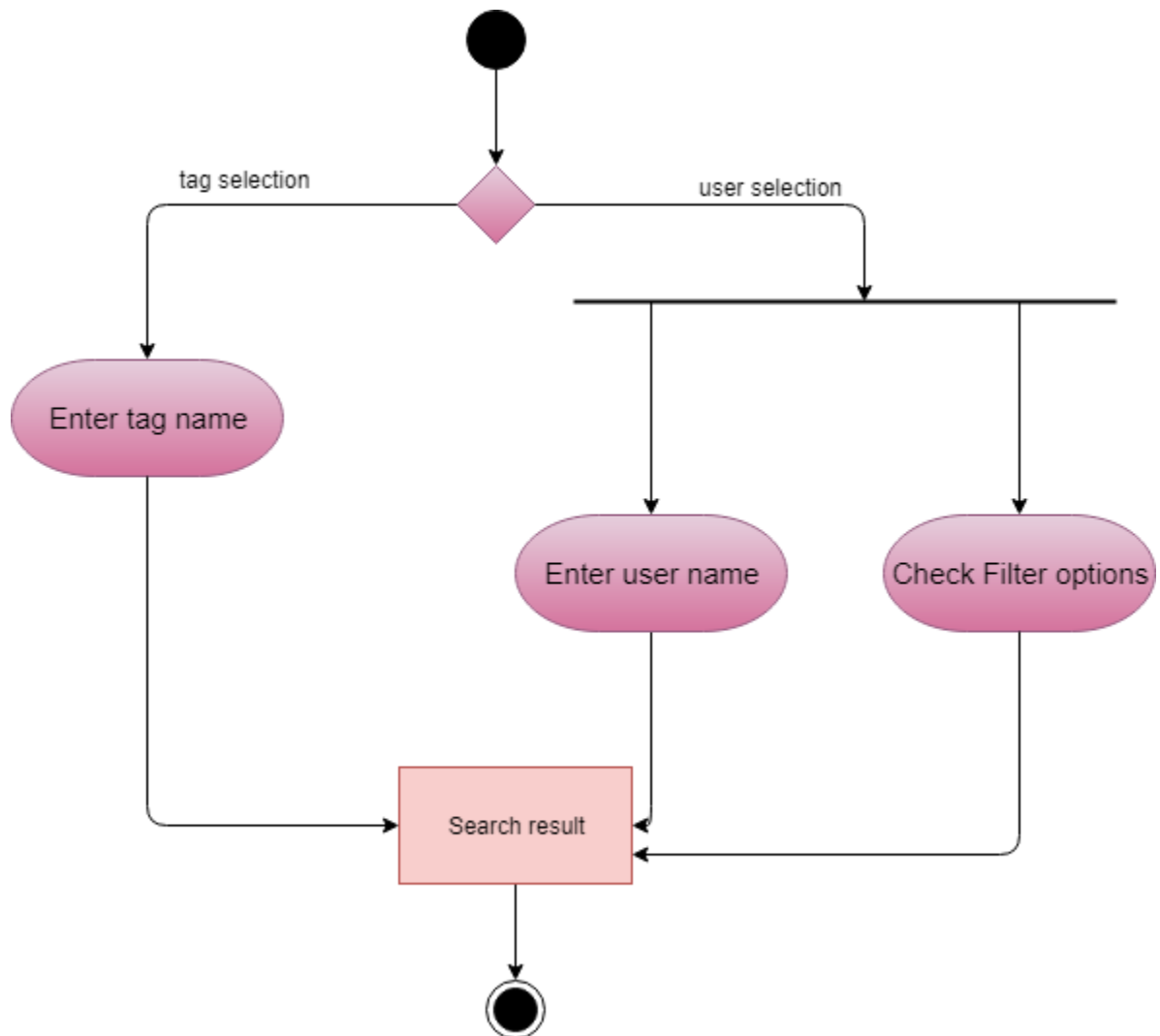
Figure 13: Level 1.2.1 Activity diagram – User search

CHAPTER-05: DATA BASED MODELING OF EQAF

This chapter describes the Scenario-Based Model for the Educational Question Answer Forum.

5.1 INTRODUCTION

Sometimes software requirements include the necessity to create, extend or interact with a database or complex data structures need to be constructed and manipulated. The software team chooses to create data models as a part of overall requirements modeling. The entity-relationship diagram (ERD) defines all data objects that are processed within the system, the relationships between the data objects and the information about how the data objects are entered, stored, transformed and produced within the system.

5.2 DATA OBJECTS

A data object is a representation of composite information that must be understood by the software. Here, composite information means an information that has a number of different properties or attributes. A data object can be an external entity, a thing, an occurrence, a role, an organizational unit, a place or a structure.

5.2.1 NOUN IDENTIFICATION

I identified all the nouns whether they are in problem space or in solution space from my usage scenario.

Table 1: Noun Identification for Data Modelling

Serial	Noun	S/P	Attributes
1	User	S	7,23,31,32
2	Inquirer	S	7,23,31,32
3	Responder	S	7,23,31,32
4	Question	S	9 ,12,13,14,17,23,24,31

5	Answer	S	9,12,14,17,23,24,27
6	Forum	P	
7	Reputation	S	
8	Comment	S	12,14,17,23,29
9	fileName	S	
10	Tag	S	12,22,28
11	Notification	S	12,14,23,30,33
12	Description	S	
13	Title	S	
14	Time	S	
15	Recommendation	S	
16	Similar Question	S	
17	Reaction	S	
18	User activity	S	
19	UserManagement	S	
20	Character	P	
21	Number	P	
22	tagName	S	
23	userId	S	
24	questionId	S	

27	answerId	S	
28	tagId	S	
29	source	S	
30	status	S	
31	tags	S	23
32	location	S	
33	notificationId	S	

5.2.2 POTENTIAL DATA OBJECTS

➢ **User:** 7,23,31,32
➢ **Inquirer:** 7,23,31,32
➢ **Responder:** 7,23,31,32
➢ **Question:** 9 ,12,13,14,17,23,24,31
➢ **Answer:** 9,12,14,17,23,24,27
➢ **Comment:**37,42,67
➢ **Tags:** 12,14,17,23,29
➢ **Notification:** 12,14,23,30,33

5.2.3 ANALYSIS FOR FINAL DATA OBJECT

➢ User store user information such as userId, tags, reputation, location
➢ Inquirer and Responder are all users of EQAF and thus common attributes stored as data object User.
➢ Question stores question information such as title, tags, time, user, description, file, reaction
➢ Answer stores answer information such as answer description, file, time, question,reactions.
➢ Comment stores information such as comment userId, description, reaction, source, time.
➢ Tags are the Question topics which store name, description, tagId.

5.2.4 FINAL DATA OBJECT

1   User: userId, tags, reputation, location

2   Question: fileName, description, title, time, reaction, userId, questionId, tags

3   Answer: fileName, description, time, reaction, userId, questionId, answerId

4   Comment: description, time, reaction, userId, source

5   Tag: description, tagname, tagId

6   Notification: description, time, userId, status, notificationId

5.3 DATA OBJECT RELATIONS

User	create	Question
User	make	Answer
User	react to	Answer
User	react to	Question
User	make	Comment
User	react to	Comment
Comment	created to	Question
Comment	create to	Answer
User	get	Notification
User	follow	Tags
Tag	is added to	Question
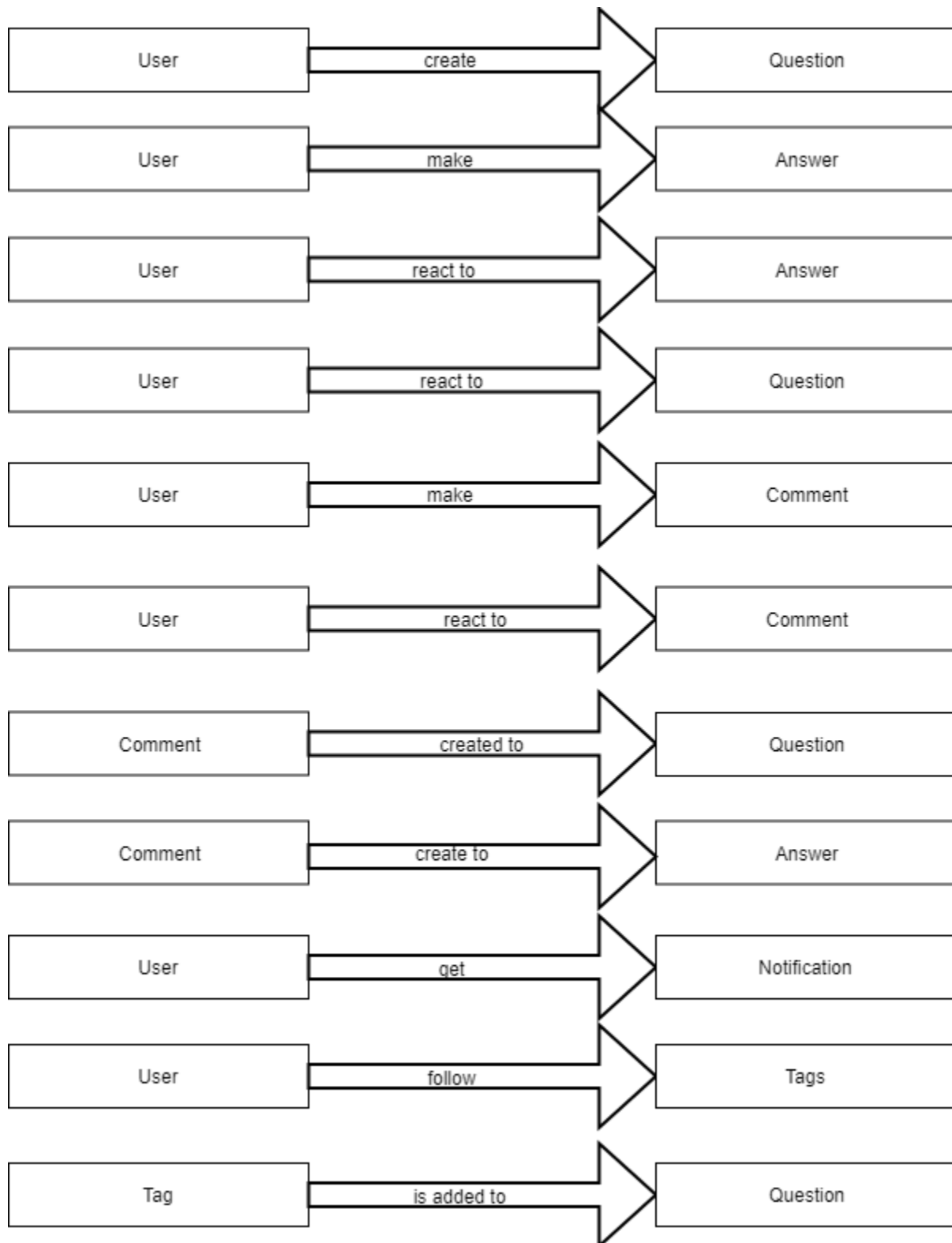
Figure 14: Relationships between data objects
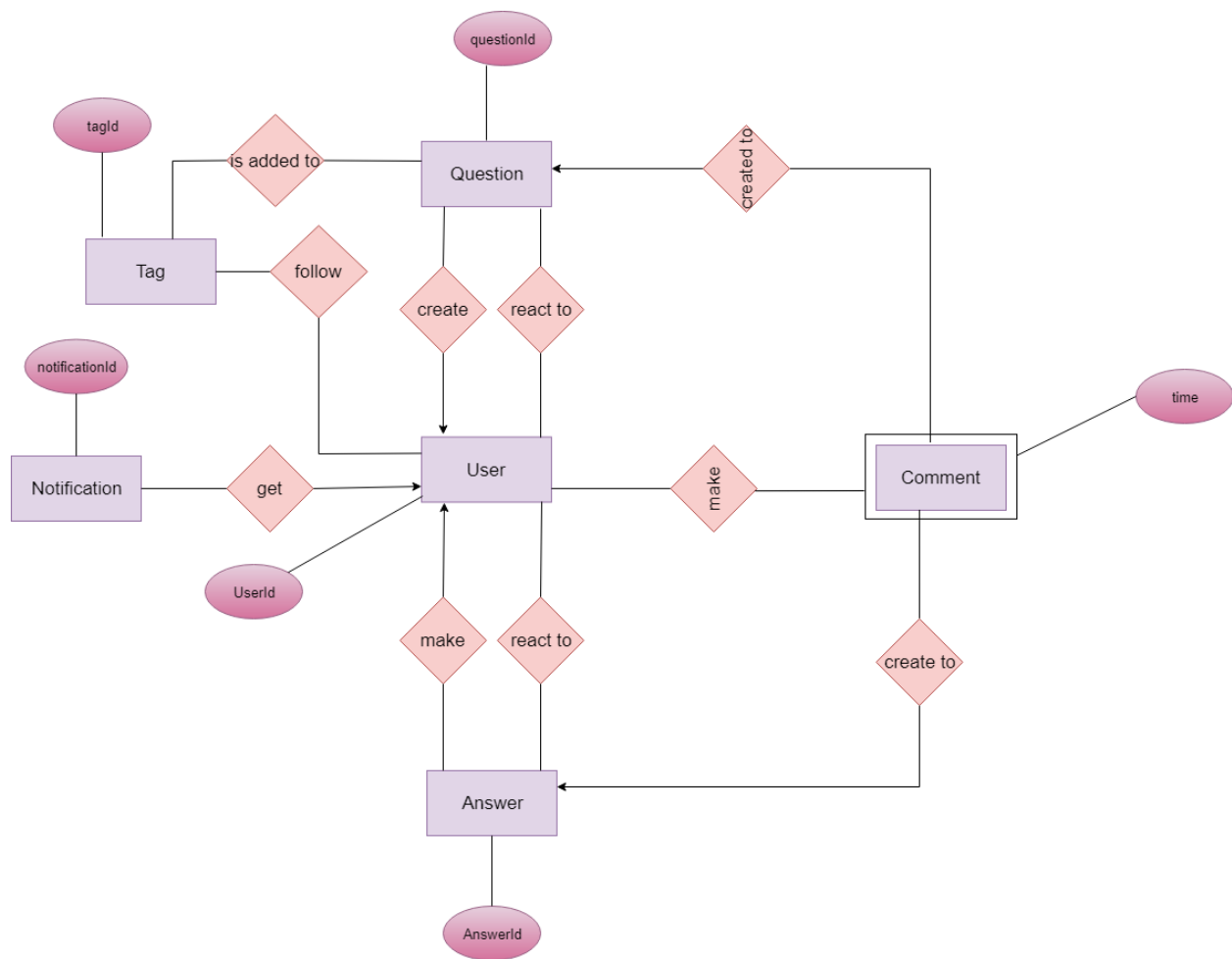
Figure 15: Entity Relationship of EQAF

5.5 SCHEMA DIAGRAM

A schema is a structure behind data organization. In a schema diagram, all database table are designated with unique columns and special features, e.g., primary key, foreign keys.

Table 2: schema table of User data object

User		
Attributes	**Type**	**Size**
userId	VARCHAR	40
tags	VARCHAR	80
reputation	NUMBER	40
location	VARCHAR	50

Table 3: schema table of Question data object

Question		
Attributes	**Type**	**Size**
questionId	VARCHAR	40
tags	VARCHAR	80
fileName	VARCHAR	40
description	VARCHAR	50
title	VARCHAR	30
time	VARCHAR	15
reaction	NUMBER	80

Answer		
Attributes	**Type**	**Size**
answerId	VARCHAR	40
fileName	VARCHAR	80
description	VARCHAR	40
time	VARCHAR	50
reaction	VARCHAR	30

Table 5: schema table of Comment data object

Comment		
Attributes	**Type**	**Size**
userId	VARCHAR	40
time	VARCHAR	80
source	VARCHAR	40
reaction	VARCHAR	50
description	VARCHAR	30

Table 6: schema table of Tag data object

Tag		
Attributes	**Type**	**Size**
tag_Id	VARCHAR	40
description	VARCHAR	80
tagName	VARCHAR	40

Table 7: schema table of Notification data object

Notification		
Attributes	**Type**	**Size**
notificationId	VARCHAR	40
userId	VARCHAR	40
time	VARCHAR	80
description	VARCHAR	40
status	BOOLEAN	40

CHAPTER–06 CLASS BASED MODELING

This chapter describes the class-based model for Educational Question-Answer Forum for Bangladesh.

6.1 INTRODUCTION

Class-based methods for requirements modeling use common concepts of object-oriented programming to craft an impression of an application that can be understood by nontechnical stakeholders. As the requirements model is refined and expanded, it evolves into a specification that can be used by software engineers in the creation of the software design. Class-based modeling represents:

1. The objects the system will manipulate
2. The operations (methods or services) that will be applied for effective manipulation
3. The relationships between the objects
4. The collaboration that occurs between the classes

6.2 IDENTIFYING ANALYSIS CLASS

Classes are identified by underlining each noun or noun phrase and plotting it into a simple table. If the class (noun) is required to implement a solution, then it becomes a part of the solution space. Otherwise, if the noun is used only to describe a solution, it is regarded as a part of the problem space. Once all the nouns have been isolated, General classification and Selection is done.

6.2.1 GENERAL CLASSIFICATION

Nouns belonging to the solution space should exhibit any of the following criteria to be considered as a class. The 7 general characteristics are stated below:

1. *External entities:* Other systems, devices, people that produce or consume information to be used by a computer-based system

2. *Things*: Reports, displays, letters, signals that are a part of the information domain for the problem.

3. *Events*: Actions or transfers (a property transfer or the completion of a series of robot movements) that occur within the context of system operation.

4. *Roles*: Responsibilities played by the people who interact with the system.

5. *Organizational units:* Divisions, groups, teams that are relevant to an application.

6. *Places:* Platform that establishes the context of the problem and overall function of the system.

7. *Structures*: Something that defines a class of objects or related classes of objects.

Table 8: Noun with general classification

Serial Number	Noun	General classification
1	User	4,5,7
2	Inquirer	4,5,7
3	Responder	4,5,7
4	Question	3
5	Answer	3
6	Forum	
7	Reputation	2
8	Comment	3
9	File	2
10	Tags	2
11	Notification	3
12	Description	2
13	Title	2
14	Time	2
15	Recommendation	3
16	Similar Question	3
17	Reaction	3
18	User activity	3,5
19	UserManagement	3,5
20	Name	2

21	UserId	2

6.2.2 SELECTION CRITERIA

Classes that fulfilled at least 3 characteristics of general classification are again reconsidered by six Selection Criteria. The six characteristics for the selection criteria are:

1. *Retained information:* The potential class will be useful during analysis only if information about it must be remembered so that the system can function.

2. *Needed services:* The potential class must have a set of identifiable operations that can change the value of its attributes in some way.

3. *Multiple attributes:* During requirement analysis, the focus should be on "major" information; a class with a single attribute may, in fact, be useful during design, but is probably better represented as an attribute of another class during the analysis activity.

4. *Common attributes:* A set of attributes can be defined for the potential class and these attributes apply to all instances of the class.

5. *Common operations:* A set of operations can be defined for the potential class and these operations apply to all instances of the class.

6. *Essential requirements:* External entities that appear in the problem space and produce or consume information essential to the operation of any solution for the system will almost always be defined as classes in the requirements model.

To be considered a legitimate class for inclusion in the requirements model, a potential object should satisfy all (or almost all) of these characteristics. The decision for inclusion of potential classes in the analysis model is somewhat subjective, and later evaluation may cause an object to be discarded or reinstated.

Table 9: selection criteria of nouns

Serial number	Noun	Selection criteria
1	User	1,2,3,4,5

2	Answer	3
3	Question	3
4	Comment	3
5	Tag	3
6	Recommendation	2
7	UserActivity	2,5
8	UserManagement	2,5

6.2.3 ASSOCIATE NOUN WITH VERB

I now identify the nouns and verbs associated with the potential classes to better find out the attributes and methods of each class.

Table 10: Associate noun and verb identification

No	Class name	Nouns	Verbs
1	User	Reputation, Location, Tags, UserId	N/A
2	Answer	Description, Image, Tags, User, Vote count, Comments, Time	N/A
3	Question	Title, Description, Image, Tags, User, Reactions, Comments, Time	N/A
4	Comment	Reactions, User, Description	N/A
5	Tag	Name, Description, Users	N/A

	6	UserActivity	N/A	Answer, ask, comment, vote
	7	Recommendation	Question, users	Recommend, findSimilarQuestions
	8	UserManagement	N/A	Get, Set, Search, Filter

6.2.4 ATTRIBUTE SELECTION

Table 11: class and attributes selection

No	Class	Attributes
1	User	UserId, Reputation, Location, Tags, Answers, Questions, Comments
2	Answer	Description, Image, Tags, User, , Comments, Reactions, Time
3	Question	Description, Image, Tags, User, Reactions, Comments, Time
4	Comment	Reaction, User, Description
5	UserActivity	N/A
6	Tag	Name, Description, Users
7	Recommendation	N/A
8	UserManager	N/A

6.2.5 METHOD IDENTIFICATION

Table 12: Method identification

No	Class	Methods
1	User	N/A
2	Answer	N/A
3	Question	N/A
4	Comment	N/A
5	UserActivity	Answer(), ask(), comment(), vote()
6	Recommendation	Recommend(), findSimilarQuestions()
7	Tag	N/A
8	UserManager	Get(), Set(), Search(), Filter(), addTag(), removeTag()

6.2.6 CLASS CARDS

After identifying my final classes, I have generated following class cards

Table 13: User

User	
Attributes	Methods
UserId, Reputation, Location, Tags, Answers, Questions, Comments	N/A

Responsibilities	Collaborative class
N/A	Tag, Comment, Answer, Question, Recommendation

Question	
Attributes	Methods
Description, Image, Tags, User, Reactions, Comments, Time	N/A
Responsibilities	Collaborative class
N/A	Answer, User, Tag, Comment

Answer	
Attributes	Methods
Description, Image, Tags, User, Reactions, Comments, Time	N/A
Responsibilities	Collaborative class
N/A	Question, User, Tag, Comment

Comment

Attributes	Methods
Reaction, User, Description	N/A
Responsibilities	Collaborative class
N/A	Question, Answer, User

UserActivity	
Attributes	Methods
N/A	Answer (), ask (), comment (), vote ()
Responsibilities	Collaborative class
➢ Save the answer information ➢ Save the question information ➢ Save the comment information ➢ Send the answer/question formation	Answer, Question, Comment,

Recommendation	
Attributes	Methods
N/A	Recommend(), findSimilarQuestions()
Responsibilities	Collaborative class
➢ Recommend question to users ➢ Find similar question	Question, Answer

Tag	
Attributes	Methods
Name, Description, Users	N/A
Responsibilities	Collaborative class
N/A	N/A

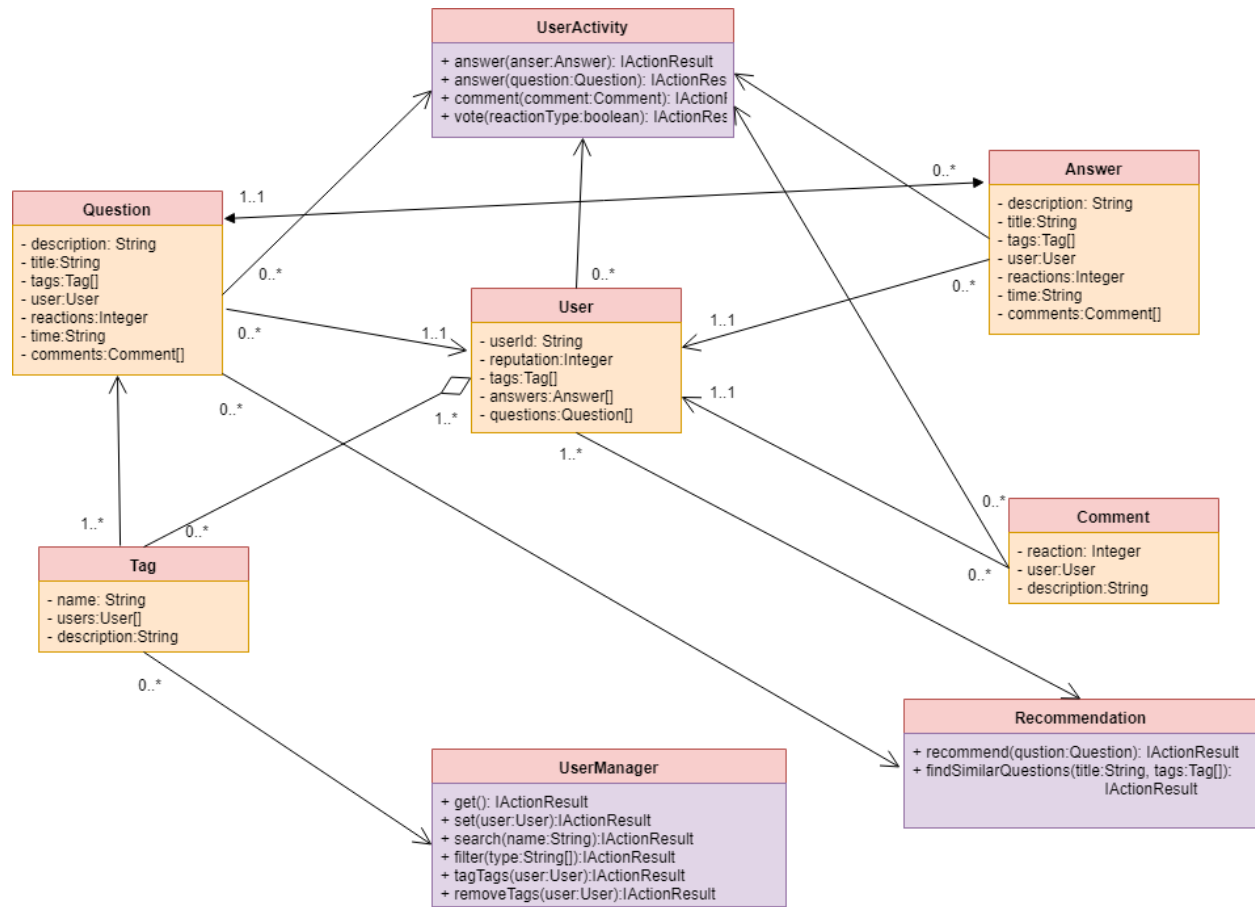| UserManager	
Attributes	Methods
N/A	Get(), Set(), Search(), Filter()
Responsibilities	User
➢ Send the user list based on reputation ➢ Save new user ➢ Search a user ➢ Filter users	N/A

6.2.7 CLASS COLLABORATION DIAGRAM

Figure 16: Class collaboration diagram of EQAF

CHAPTER–07 ALGORITHMS IMPLEMENTATION

7.1 INTRODUCTION

In this project there exist technical challenges. To get solution to these challenges I need to implement some algorithms.

7.2 TECHNICAL CHALLENGES

There are two technical challenges

1. Question recommendation to appropriate users
2. Similar Question suggestion to user while creating question

7. 3 ALGORITHMS

1. Apriori: To recommend/send asked question to appropriate users

2. Cosine Similarity measurement: To get similar question while user creating question

CHAPTER–08: CONCLUSION

I am pleased to submit the final SRS report on Educational Question Answer Forum. From this, the readers will get a clear and easy view of the overall system. This SRS document can be used effectively to maintain the software development cycle. It will be very easy to conduct the whole project using this SRS. Hopefully, this document can also help my junior BSSE batch students. I tried my best to remove all dependencies and make an effective and fully designed SRS. I believe that the reader will find it in order.