

A dark blue vertical bar on the left side of the page. A blue arrow points to the right from the bar, containing the date 3/5/2016.

3/5/2016

Online Student Registration and Exam System

Software Requirement Specification
and Analysis

Software Project Lab - 2



Institute of Information Technology
University of Dhaka

Submitted by:

M.A. Nur Quraishi (BSSE-0615)

Feroze Al Mamun (BSSE-0634)

Supervisor

Md. Nurul Ahad Tawhid

Assistant Professor

Submitted to

Sheikh Muhammad Sarwar

Assistant Professor

And

Asif Imran Anik

Lecturer

Submission Date:

13th April, 2016

Institute of Information Technology

University of Dhaka

Letter of Transmittal

Sheikh Muhammad Sarwar
Assistant Professor,
and
Asif Imran
Lecturer,
Institute of Information Technology,
University of Dhaka
April 13, 2016

Honorable Sir,
We have submitted the enclosed report on Software Requirements Specification and Analysis of our proposed “Online Student Registration and Exam System” for your approval. This report includes all the requirements we have acquired for developing the project.

The primary purpose of this document is creating SRS report for the project what we are doing for our *software project lab-II*. This report includes the details of each steps we have followed to collect the requirements.

Sincerely yours,
M.A. Nur Quaraishi (BSSE-0615)
Feroze Al Mamun (BSSE-0634)
Institute of Information Technology
University of Dhaka
Session: 2013-14

Enclosure: Software Requirement Specification and Analysis Report

Exclusive Summary

The purpose of our proposed project is to reduce all the manual controversies for Registration and Exam system. In our proposed project, one part is called discussion forum which stands for establishing efficient learning system for the students where any student can ask questions (exam related) and rest of other can answer. As our proposed system is implemented as a web based portal, it will provide an easy way to access this site from anywhere via internet thus it will reduce time and cost.

Acknowledgement

I am expressing my heartiest gratitude to Almighty ALLAH to complete the designated SRS report in time and without hassles.

I would like to thank my project supervisor Md. Nurul Ahad Tawhid and course instructors Sheikh Muhammad Sarwar and Asif Imran Anik for their valuable supervision, advice, instruction and time throughout the project.

Table of contents

Chapter 1	8
Introduction.....	8
1.1 Purpose	8
1.2 Intended Audience.....	8
1.3 Scope	9
1.4 Timeline	9
Chapter 2.....	10
Inception	10
2.1 Introduction	10
2.1.1 Identifying Stakeholders.....	10
2.1.2 Asking The First Questions	11
2.1.3 Recognizing Multiple Viewpoints.....	11
2.1.4 Working towards Collaboration	11
2.2 Conclusion.....	12
Chapter 3.....	13
Elicitation.....	13
3.1 Introduction	13
3.2 Eliciting Requirements.....	13
3.3 Collaborative Requirements Gathering.....	13
3.4 Quality Function Deployment.....	13
3.4.1 Normal Requirements.....	14
3.4.2 Expected Requirements	14
3.4.3 Exciting Requirements	14
3.5 Usage Scenario.....	14
3.6 Elicitation Work Product.....	15
Chapter 4.....	16
Scenario Based Modeling	16
4.1 Definition of Use Case	16
4.2 Use Case Diagrams	17
4.2.1 Level-0 of the use case:	17
4.2.2 Level-1 of the use case:	18

4.2.3.1 Level-2 of the “Online registration” subsystem:	18
4.2.3.2 Level-2 of the “Authentication” subsystem:	19
4.2.3.3 Level-2 of the “Online exam” subsystem:	20
4.2.3.4 Level-2 of the “Discussion forum” subsystem:	21
4.2.4.1 Level-3 of the “Offering course” action:	22
4.3 Activity and Swim Lane Diagram.....	23
Use case 1: Online Registration.....	23
Use Case 2: Log in.....	25
Use Case 3: Log out.....	27
Use Case 4: Change password.....	29
Use Case 5: Change profile picture	31
Use Case 6: Offering course	33
Use Case 7: Requesting course.....	35
Use Case 8: Approving or rejecting student.....	37
Use Case 9: Uploading questions and answers	39
Use Case 10: Generating random question for each student	41
Use Case 11: Appearing at the exam.....	43
Activity Diagram:	43
Use Case 12: Viewing the scores	45
Use Case 13: Asking answer	47
Use Case 14: Answering question	49
Use Case 15: Replying each answer or reply	51
Use Case 16: Rating each answer or reply	53
Use Case 16: Approving correct answer	55
Chapter 5	57
Data Model.....	57
5.1 Data Modeling Concept	57
5.2 Data Objects	57
5.2.1 Grammatical Parsing (Noun Identity)	57
5.2.2 Identify Data Object	59
5.2.3 Pair Relation	61
5.2.4 ER-Diagram.....	64

5.2.5 Database Table	65
Chapter 6	71
Class Based Model.....	71
6.1 Class Based Modeling Concept.....	71
6.2 General Classification	71
6.3 Selection Criteria.....	71
6.4 Class Identification.....	72
6.5 Subject, Verb, Object/ Predicate Identification:.....	74
6.6 Class Card	75
6.6 CRC Diagram.....	81
Chapter 7	82
Flow Oriented Model.....	82
7.1 Introduction	82
7.2 Data Flow Diagram (DFD)	82
7.2.1 Level-0 of DFD	82
7.2.2 Level-1 of DFD	83
Chapter 8.....	84
Behavioral Model.....	84
8.1 State Transition Diagram	84
Event List:.....	84
Identifying the initiator and collaborator:.....	85
State Transition Diagrams (STD):	86
Class - Student:	87
Class – Exam:	88
8.2 Sequence Diagram.....	89
Chapter 9.....	92
Screenshot of the Project	92
Chapter 10.....	99
References.....	99
Chapter 11	100
Conclusion	100

Chapter 1

Introduction

1.1 Purpose

This software Requirement Specification (SRS) document stands for “Online Student Registration and Exam System”. In this document, there contains functional, non-functional and supported requirements that establishes a requirements baseline for developing our proposed project. The requirements having this document are independent, uniquely numbered and embodied by topic. The SRS serves as specifying user requirements to the developer by means of communication and provides a common reference point for both the developer team and stakeholder community. The SRS will evolve enough 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 managers, designers, developers, and testers.

The customer will use this SRS to verify that the developer team will produce a product which will meet all the requirements specified and approved by the customer.

This SRS document helps the project manager to plan milestones and a delivery date, and ensuring that the developing team is on the right track during developing the system.

The designers will use this SRS as a basis for designing the system. The designers will continually roll back to this SRS to ensure that the system they are designing, will fulfill the customer's requirements

The necessity of using this SRS documentation for the developer is in order to provide a basic for developing the system's functionality. The developers will link the requirements defined in this SRS so that they ensure that they are producing a software which meets all the requirements defined in SRS as satisfying the customer requirements.

SRS document provides an easy way for the tester to perform their task with more flexibility because in SRS document requirements must be clear and specific with no uncertainty, requirements should be measurable in terms of specific values, requirements should be testable having some evaluation criteria for each requirement, and requirements should be complete, without any contradiction.

1.3 Scope

The main aim of the project is to learn the intricacies of modeling the data base with the given requirements and using a web based interface to interact with the back end keeping in mind the data consistency and the stability of the entire system.

Since the system is developed with a web based interface, we can start executing the model with parallel executions with the confidence that the back end MySQL data store will take care of the concurrent transactions. The ACID (Atomicity, Consistency, Isolation, Durability) property of the database helps to keep the system consistent and stable.

The website to conduct online examination is “Pathaloy”. This website provides facility to institutes to conduct online exams by providing a unique id to each institute. The institute provides questions along with positive and negative marks. Institute also enters the list of eligible students who are registered previously. All the information entered can be later edited by the institute.

In turn student can login with their id, name and institute id to give the exams and can view their result then and there. Institutes can also view the result of their students.

The website provides an especial feature known as “FAQ”. In this section, Students can ask any question related to the exam or quiz. On the other hand, rest of the users can reply with their knowledge and logic. They can also rate the replies by the merit of the answer.

1.4 Timeline

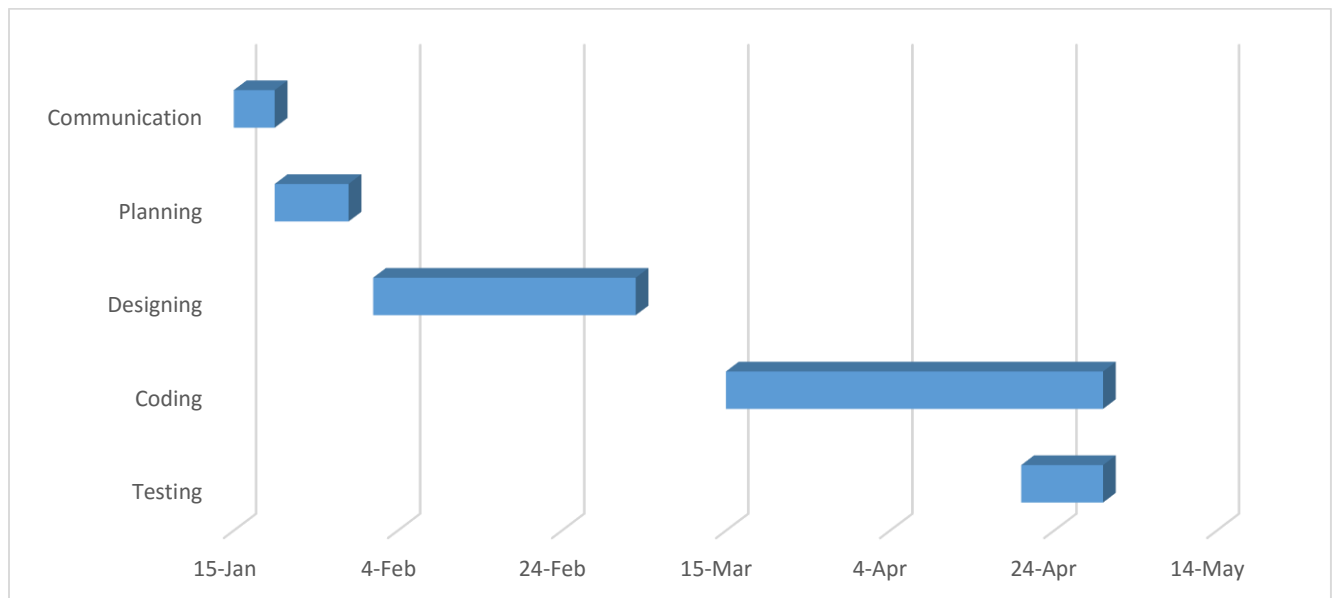


Figure 1.4: Timeline of “OSRES”

Chapter 2

Inception

2.1 Introduction

The goal of the inception phase is to achieve concurrence among all stakeholders on the lifecycle objectives for the project. The inception phase is of significance primarily for new development efforts, in which there are significant business and requirements risks which must be addressed before the project can proceed.

Inception is the beginning phase of requirements engineering. It defines how does a software project get started and what is the scope and nature of the problem to be solved. To establish the groundwork, we have worked with the following factors related to the inception phases:

- I. Identifying Stakeholders
- II. Recognizing multiple viewpoints
- III. Working towards collaboration
- IV. Asking the First Questions

2.1.1 Identifying Stakeholders

Stakeholders refer an individual, group, or organization, who may affect, be affected by, or perceive itself to be affected by a decision, activity, or outcome of a project. Stakeholders include end-users who interact with the system and everyone else in an organization that may be affected by its installation. To identify the stakeholders, we consulted with Assistant Librarian (Program) and asked her following questions:

- Who is paying for the project?
- Who will be using the project outcomes?
- Who gets to make the decisions about the project (if this is different from the money source)?
- Who has resources I need to get the project done?
- Whose work will my project affect? (During the project and also once the project is completed).

By analyzing the requirements, we select two stakeholders for our proposed “Online Student Registration and Exam System” project. These stakeholders are course teacher and student.

1. Course Teachers: Course Teacher offers course for the student and approves or rejects student for his/her course.
2. Student: Student requests for course and appears at the exam if he/she is approved.

2.1.2 Asking The First Questions

We predefine a set of context-free questions focuses on the customer and other stakeholders, overall project goals and benefits. The questions are mentioned above. These questions have helped us to identify all stakeholders, measurable benefit of the successful implementation and possible alternatives to custom software development. Next set of questions are helped us to gain a better understanding of problem and allows the customer to voice his or her perception about the solution. The final set of question focused on the effectiveness of the communication activity itself.

2.1.3 Recognizing Multiple Viewpoints

We gather these view points by discussing with our selected stakeholders including Course Teacher and student.

- 1) Course Teacher
 - i) Offering course
 - ii) Approving or rejecting student
 - iii) Generating and publishing exam information in the notice board
 - iv) Answering and replying the question which is asked by student in discussion forum
 - v) Approving the correct answer by marking it with tic sign
- 2) Student
 - i) Requesting course
 - ii) Appearing at the exam
 - iii) Asking exam related question in discussion forum
 - iv) Answering or replying each question or answer
 - v) Rating each answer or reply

2.1.4 Working towards Collaboration

Every stakeholder has their own requirements. We followed following steps to merge these requirements:

1. Identify the common and conflicting requirements
2. Categorize the requirements
3. Take priority points for each requirement from stakeholders and on the basis of this voting prioritize the requirements
4. Make final decision about the requirements

Common Requirements:

- Web-Based Interfaces.
- The application can be accessed from any computer that has Internet access.
- Easy Access.
- Maintain a database of all information.

Collecting Requirements:

- Easy access and Strong Authentication.
- Don't allow to access any unauthenticated user.

Final Requirements:

We finalized following requirements for the system by categorizing and prioritizing the requirements:

Error free system (Level of tolerance 5% that means maximum 5% error may be considerable).

- Web-based interfaces.
- Accessible via the Internet.
- Allow valid users to login and logout.
- Restrict access to functionality of the system based upon user roles.
- Allow administrators of the system to change users and configure parameters of the system.
- Allow valid users that log in to view his/her accessible content.
- Maintain a database of all information.
- Restrict access to functionality of the system based upon user roles. Suppose student never view Course Teacher profile.

2.2 Conclusion

Inception phase helped us to establish basic understanding about our proposed “Online Student Registration and Exam System” project; identify the people who will be benefited if online exam becomes automated, define the nature of the “Online Student Registration and Exam System” software and establish a preliminary communication with our stakeholders.

Chapter 3

Elicitation

3.1 Introduction

Requirement elicitation is the process of discovering, reviewing, documenting, and understanding the user's needs and constraints for the system. Elicitation is a step of requirement engineering that helps the customer to define what is required. To complete the elicitation step we face many problems like problems of scope, problems of volatility and problems of understanding. However, this is not an easy task. To overcome these problems, we have worked with the Eliciting requirements activity in an organized and systematic manner.

3.2 Eliciting Requirements

Unlike inception where Q&A (Question and Answer) approach is used, elicitation makes use of a requirements elicitation format that combines the elements of problem solving, elaboration, negotiation, and specification. It requires the cooperation of a group of end-users and developers to elicit requirements. To elicit requirements, we completed following four works.

- Collaborative Requirements Gathering
- Quality Function Deployment
- Usage Scenarios
- Elicitation work products

3.3 Collaborative Requirements Gathering

Many different approaches to collaborative requirements gathering have been proposed. Each makes use of a slightly different scenario. We completed following steps to accomplish this task.

- The meetings were conducted with Teachers. He was questioned about the requirements and expectations from the automated “Online Student Registration and Exam system”.
- He was asked about the problems he is facing with the current manual system. At last we selected our final requirement list from covering the meetings.

3.4 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. With respect to our project the following requirements are identified by a QFD.

3.4.1 Normal Requirements

Normal requirements consist of objectives and goals that are stated during the meeting with the customers. Normal requirements of our project are: -

- Accessible via the Internet.
- Allow valid user to log in and log out to the system.
- Restrict access to functionality of the system based on user roles
- Allow valid users that log in to renew, reserve, and view the items
- The application is accessible from any computer via internet.
- A product reference manual describing how to utilize this web based portal.

3.4.2 Expected Requirements

Selectable fields wherever possible instead of fields that require the user to type in data.

3.4.3 Exciting Requirements

These requirements go beyond the customer's expectations and prove to be very satisfying when present.

- The user interface should provide appropriate error messages for invalid input as well as tool-tips and online help.
- The user interface should follow standard web practices such that the web interface is consistent with typical internet applications.
- Offer log in with smart phone and tablets.
- The system's configuration shall be documented and updated as changes to the system are made due to patches, new releases, new Changes etc.

3.5 Usage Scenario

The Online Registration System allows the students to register as authenticated user and organize/update their course menu by adding or removing individual courses during the registration period through internet. It also allows the course teachers to register as authenticated user, offer courses online and approve or decline those previously made requests by the interested students. Online Exam System is a web-based portal that establishes an educational network between the course teachers and the students for taking course related exam with a discussion forum.

Online Registration System (ORS) begins with the registration process of the course teachers and students. Both of them need to submit their profile, email. The profile includes first and last names, gender, date of birth, local address, permanent address, institute name, enrolled year, username, password, contact info and a profile picture.

After successful logged in, the course teacher inserts course name and course id in the course list and fixes the exam date, exam information. Then, he/she post it on the websites notice board. After logged in, any interested student can make a request for appearing at the exam. Course teacher verify all the information of the student to approve or reject a student. Course teachers have the supreme authority to approve or reject any student. An approved student gets an email. An approved student must verify their email within 24 hours after receiving the mail. If they do not verify it within 24 hours, then they need to go through the registration process again. Both students and teachers can be involved more than one courses.

The course teacher enters into the website before the exam and uploads questions (MCQ) and solutions of these in the system database which will not be available for the students. Then, he/she sets the number of questions for each student, the positive and negative marks for each right/wrong answer. He/she also sets the exam time.

Questions are randomly generated for each approved student just before the exam. The exam begins at the schedule time and ends in due time. After completing the exam, the student can submit their script at any time before the exam time ends. If a student cannot submit his/her script during this period, then the system automatically submits the script. After submission, total scores are calculated and saved immediately in the system database. Later the system shows the correct answers and achieved score to the students. The course teacher also can view the mark sheets of the students.

Students can ask any exam related questions in the discussion forum. Other users including course teachers and rest of the students can answer it. There is an opportunity to reply against each answer or comment. The student who asked the question can rate each answer according to its merit. Course teacher approves the correct answers by marking it with a tic sign.

3.6 Elicitation Work Product

The output of the elicitation task can vary depending on size of the system or product to be built. Our elicitation work product includes:

- A statement of our requirements for automated Online Student Registration and Exam system.
- A bounded statement of scope for our proposed system.
- A list of customers, users and other stakeholders who participated in requirement specification.
- Set of usage scenarios.
- Description of the system's technical environment.

Chapter 4

Scenario Based Modeling

This chapter describes scenario based modeling of Online Student Registration and Exam System.

4.1 Definition of Use Case

A use case is a software and system engineering term that describes how a user uses a system to accomplish a particular goal. A use case acts as a software modeling technique that defines the features to be implemented and the resolution of any errors that may be encountered.

Use cases define interactions between external actors and the system to attain to particular goals. There are three basic elements that make up a use case:

1. **Actors:** Actors are the type of users interact with the system.
2. **System:** Use cases capture functional requirements that specify the intended behavior of the system.
3. **Goals:** Use cases are typically initiated by a user to fulfill goals describing the activities and variants involved in attending the goal.

Use cases are modeled using unified modeling language and are represented by ovals containing the names of the use case. Actors are represented using lines with the name of the actor written below the line. But here, we use the combination of lines and a circle to represent each actor. To represent an actor's participation in a system, a line is drawn between the actor and the use case. Boxes around the use case represent the system boundary.

There are two types of actor:

1. Primary actor.
2. Secondary actor.

Primary actor: Primary actor refers who is directly involved with the system in order to achieve required function and benefit from the system. They interact directly and frequently with the software.

In our proposed system, students and teachers are primary actor.

Secondary actor: Secondary actor refers who is indirectly involved with the system but necessary to support the system so that system can perform its functionality without any hinder. Secondary actor either produces or consumes information.

In our proposed system, there is no secondary actor.

4.2 Use Case Diagrams

Use case diagram provides the non-technical view of the system. A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify and organize system requirements.

4.2.1 Level-0 of the use case:

By analyzing the usages scenario, we select two primary actors. Figure 4.2.1 indicates that these two actors are directly interacted with the hole system. Basically the whole system directly depends on these two actors. But we cannot find any secondary actor by analyzing the usages scenario.

Here, these two actors who are directly involved with our proposed system:

1. Student
2. Teacher

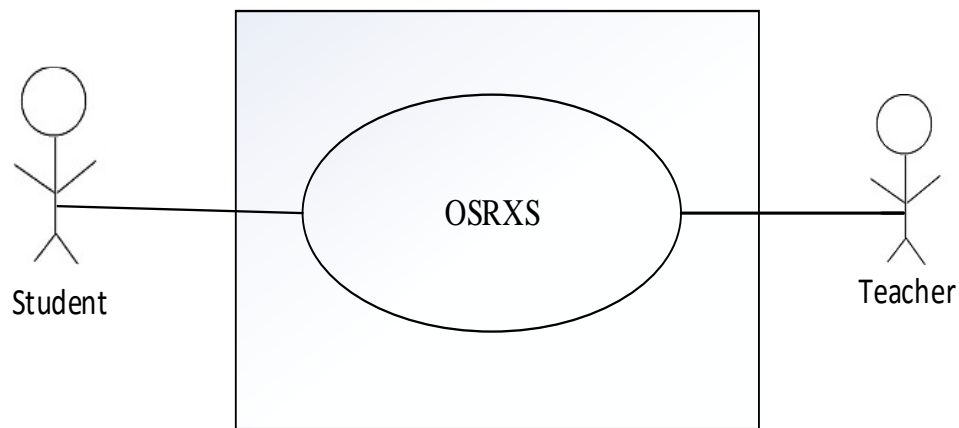


Figure 4.2.1: Level-0 of the use case.

4.2.2 Level-1 of the use case:

The proposed system is divided into four subsystems. These subsystems are online registration, authentication, online exam and discussion forum. Here, online registration, authentication and discussion forum are the heart of this proposed project and authentication subsystem is used only for protecting this system from unauthorized user. Both teacher and student are involved with all of the subsystems.

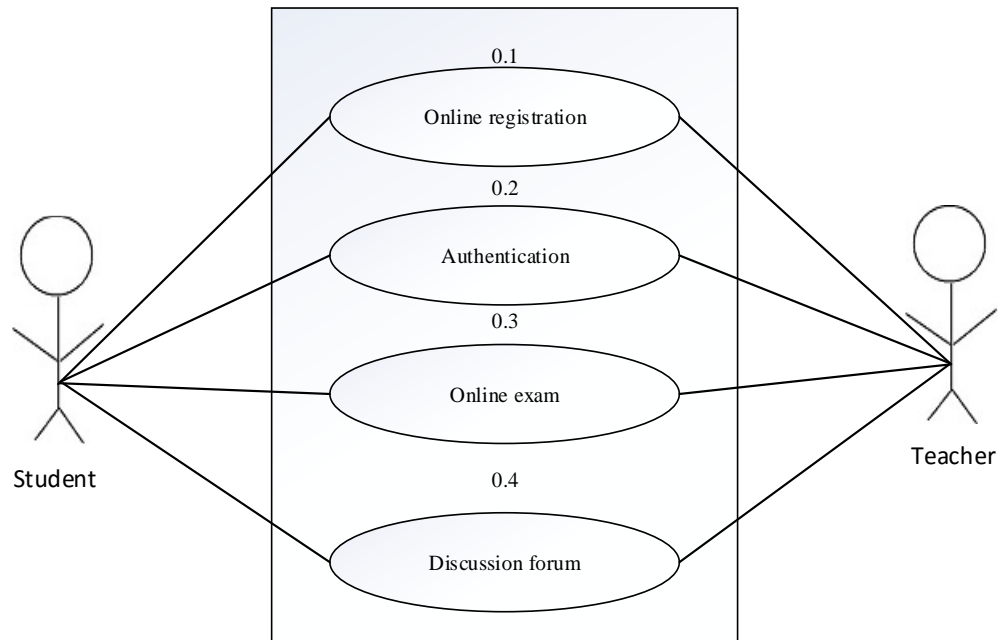


Figure 4.2.2: Level-1 of the use case

4.2.3.1 Level-2 of the “Online registration” subsystem:

The online registration subsystem is separated into two actions. These actions are registered as teacher and registered as student. These actions are registered as teacher and registered as student. Students are involved with the second action and teachers are interacted with the first action. Action registered as teacher allows any user to fill up form as a teacher of the proposed system. However, registered as student allows any user to fill up form as a student.

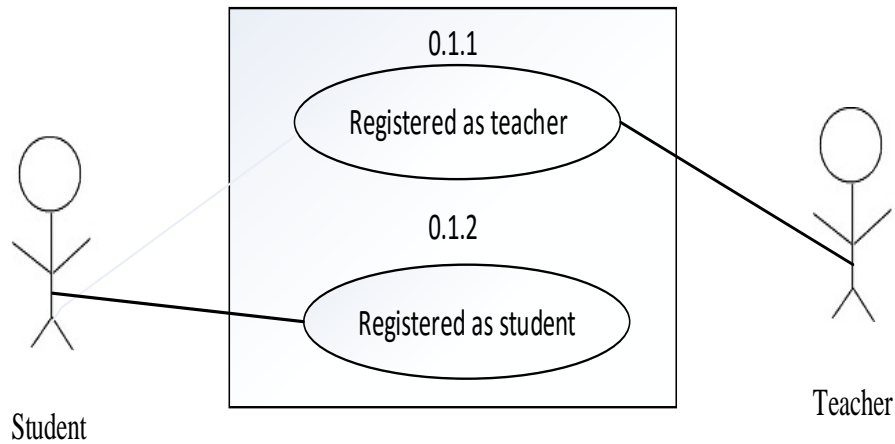


Figure 4.2.3.1: Level-2 of the “Online registration” subsystem

4.2.3.2 Level-2 of the “Authentication” subsystem:

The authentication subsystem is divided into four actions. These actions are log in, log out, change password and change profile picture. Both teachers and students can be interacted with all for actions specified at authentication subsystem. Action log in is used to enter the system, action log out is used to exit from the system, action change password and change profile picture are both used to change current password or profile picture.

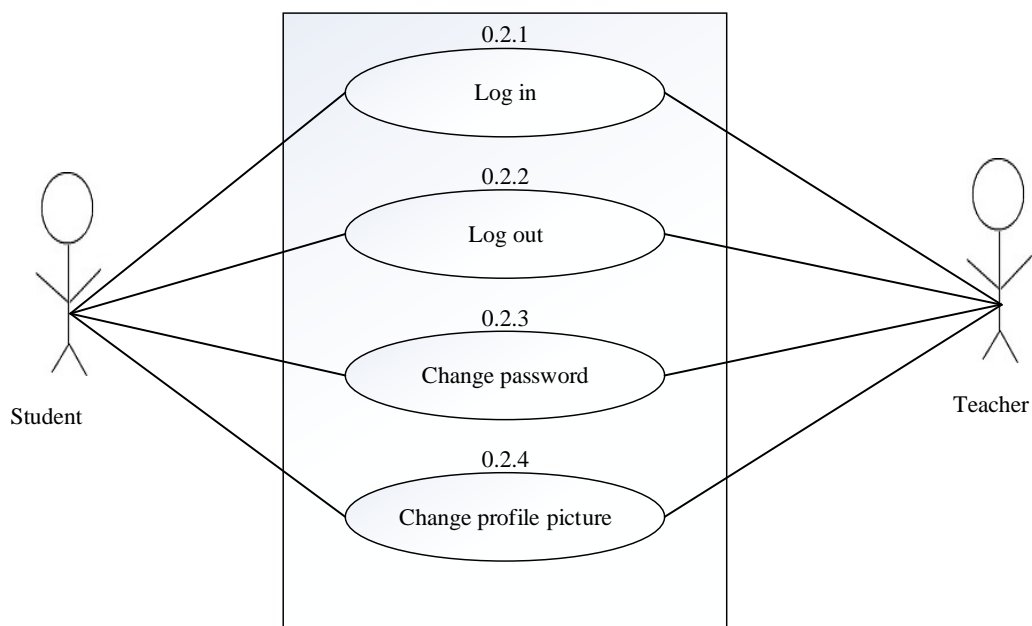


Figure 4.2.3.2: Level-2 of the “Authentication” subsystem

4.2.3.3 Level-2 of the “Online exam” subsystem:

The online exam subsystem is composed of seven actions. Teachers are interacted with five actions and these actions are offering course, approving or rejecting student, uploading questions and solutions, generating random question for each students and viewing the scores. On the contrary, students are connected to three actions and these actions are requesting course, appearing at the exam and viewing the scores.

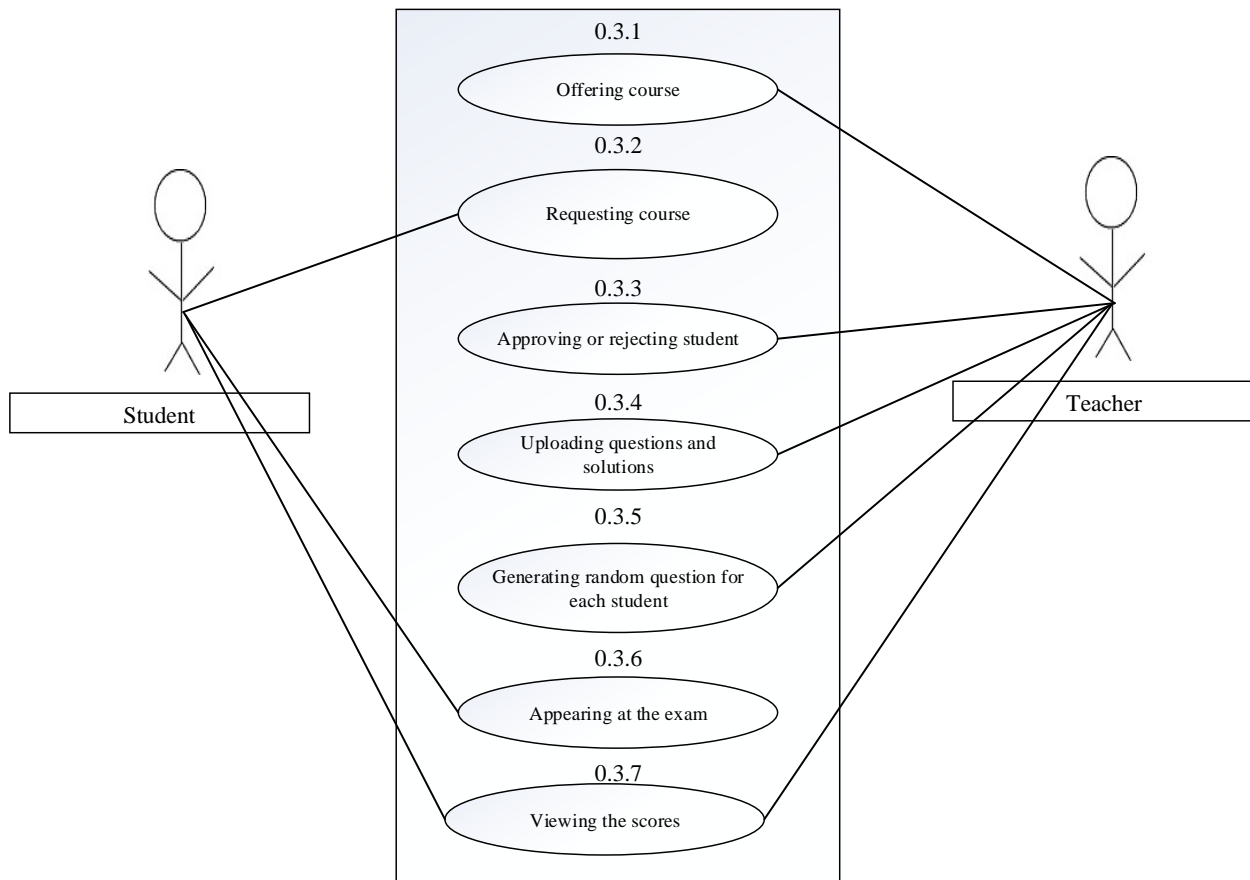


Figure 4.2.3.3: Level-2 “Online exam” subsystem

4.2.3.4 Level-2 of the “Discussion forum” subsystem:

The discussion forum subsystem is divided into five actions. These actions are asking question, answering question, replying each answer, rating each answer/reply, approving correct answer. Students are interacted with first four actions and teachers involve with all these actions except first action.

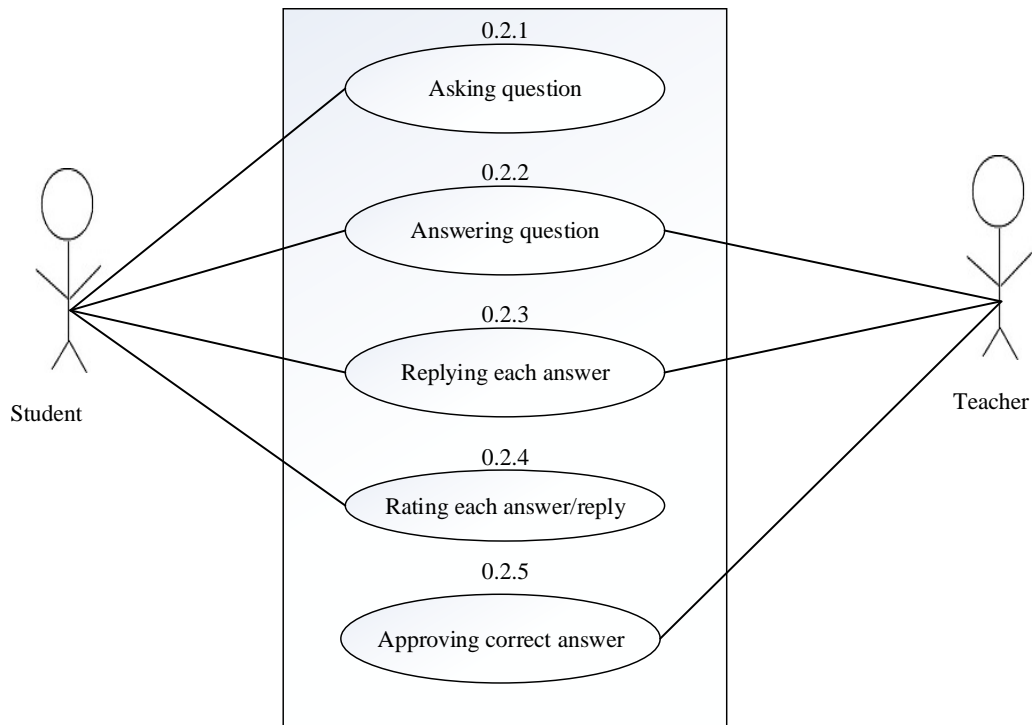


Figure 4.2.3.4: Level-2 of the “discussion forum” subsystem

4.2.4.1 Level-3 of the “Offering course” action:

Offering course action is composed of three actions and these actions are selecting and notifying exam schedule, selecting and notifying total exam marks, selecting and notifying positive/negative mark for each answer. Only teachers are interacted with all the actions.

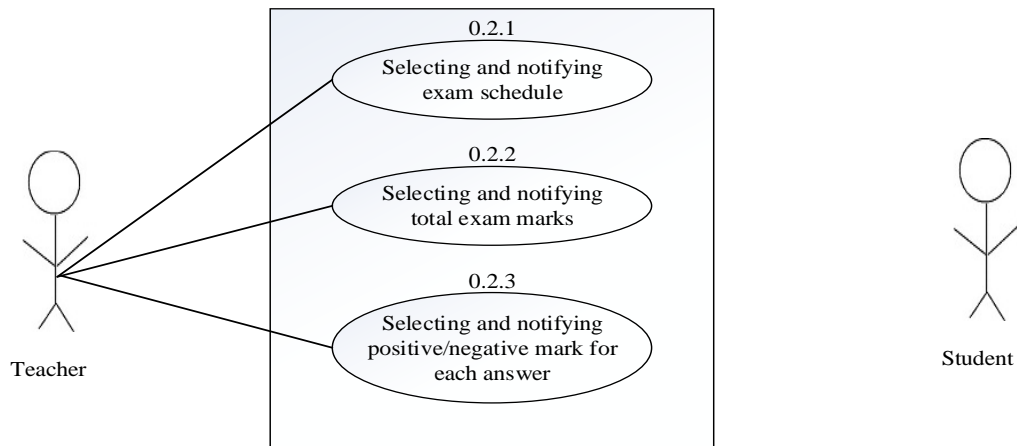


Figure 4.2.3.4: Level-2 of the “Offering course” action

4.3 Activity and Swim Lane Diagram

Activity diagram shows the technical view of the system for every use case from which we can understand how the system actually works and how the actors interact with the system.

Here is the activity diagram for “Online Registration” use case:

Use case 1: Online Registration

Activity Diagram:

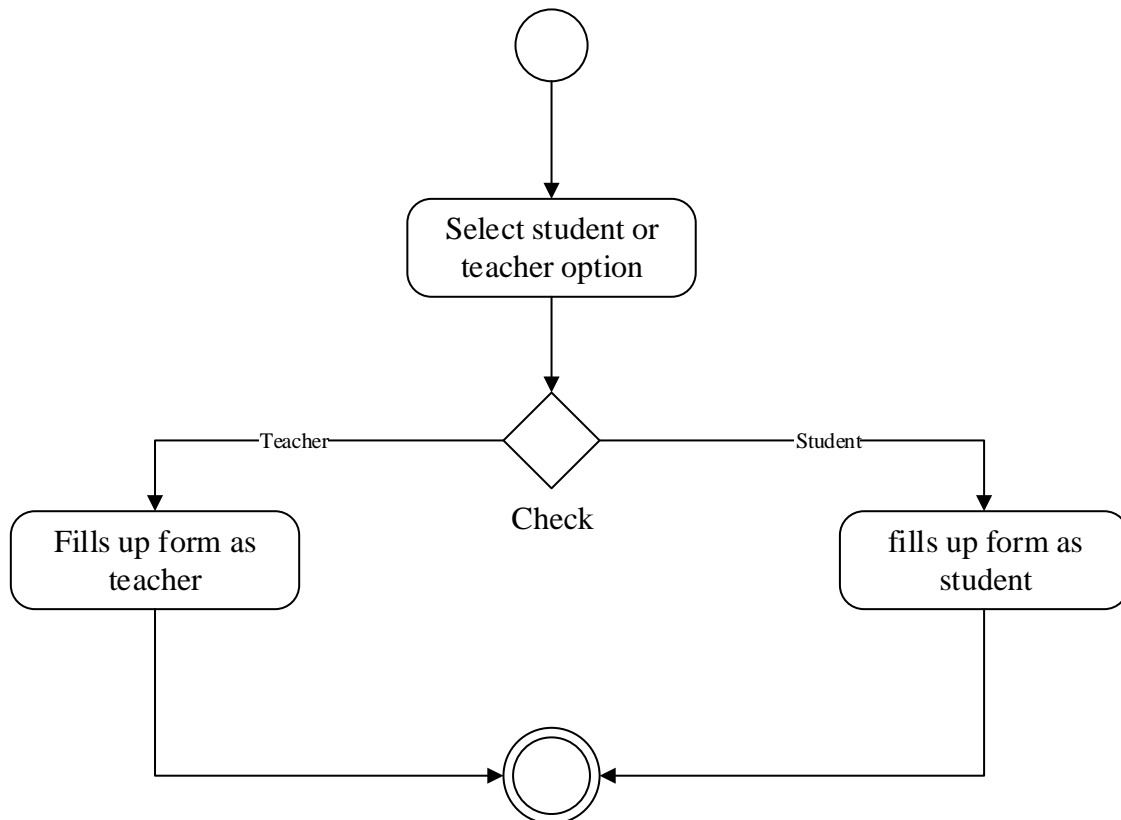


Figure 4.3.1: Activity Diagram for “Online Registration” use case

Swim Lane Diagram:

Swim lane diagram of a specific activity diagram shows the responsibilities of each actor dividing them into lanes. From this diagram we can improve our understanding about how the system works and which actors play what role.

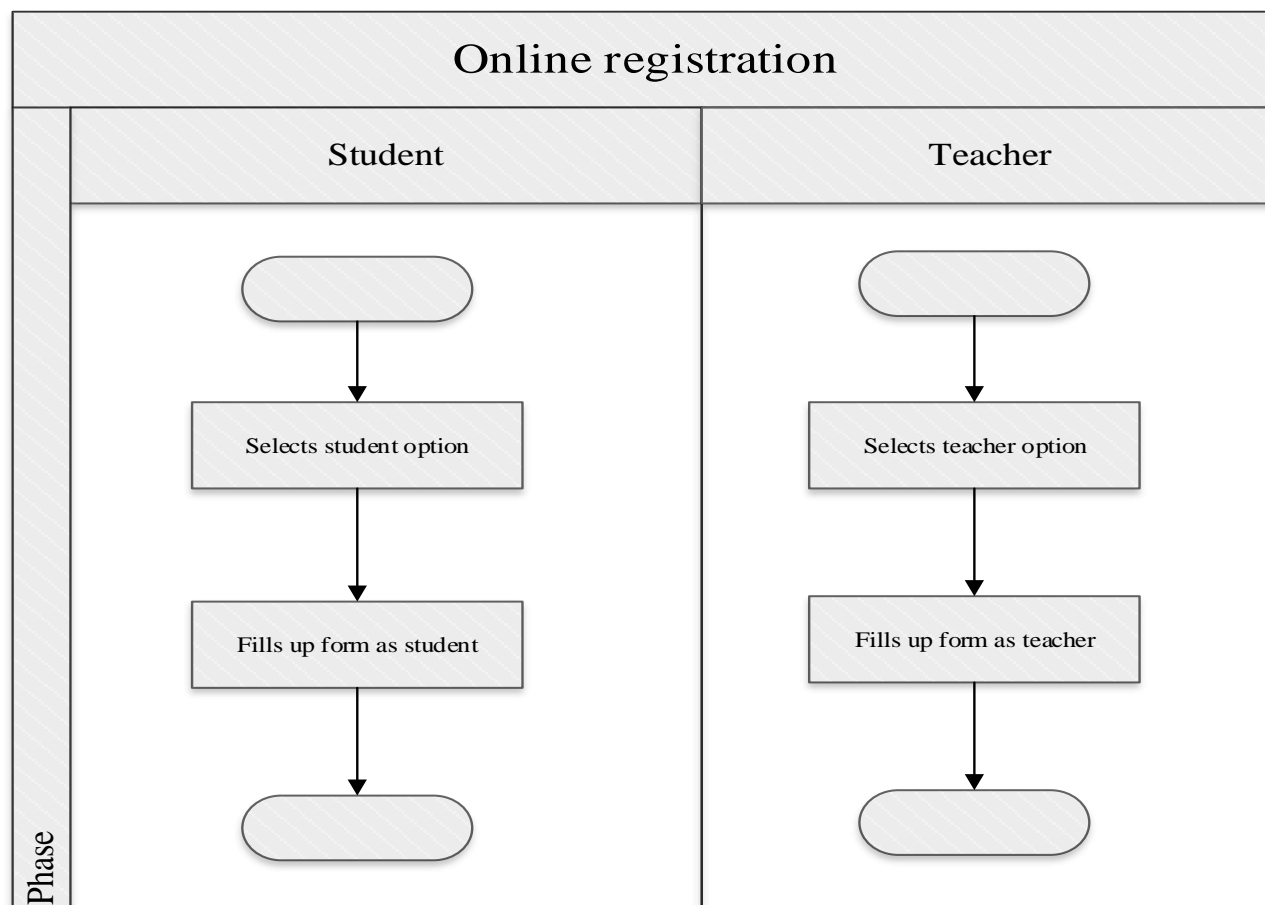


Figure 4.3.2: Swim Lane Diagram for “Online Registration” use case

Use Case 2: Log in
Activity Diagram:

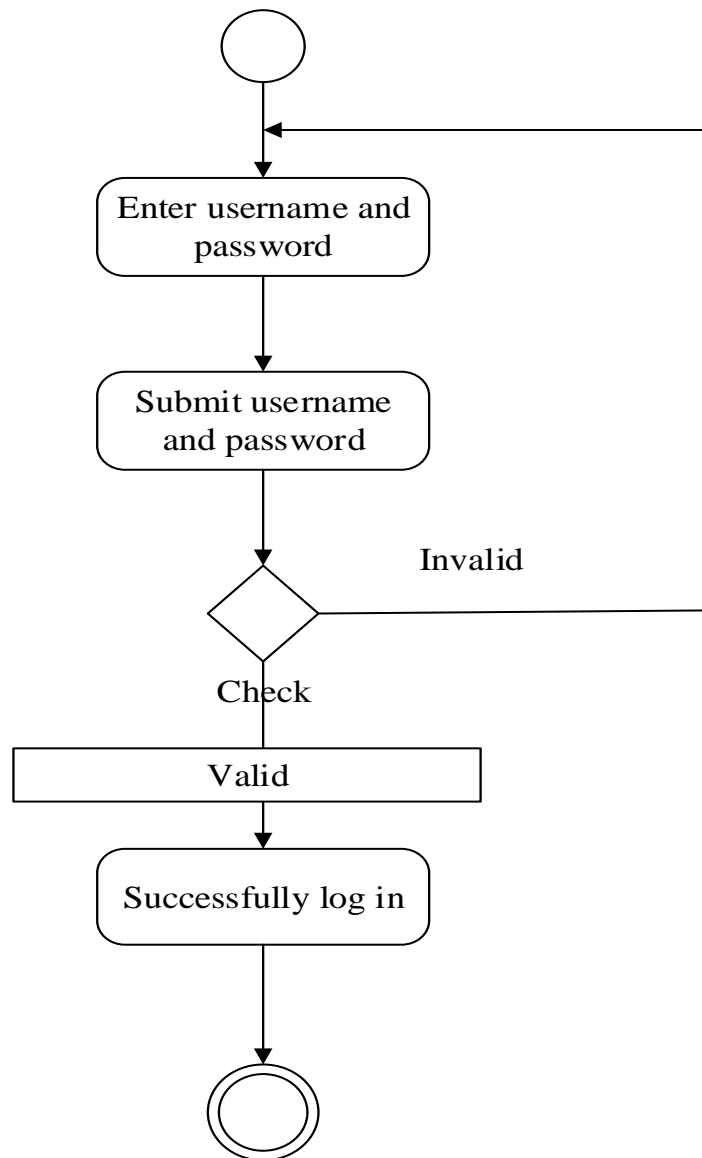


Figure 4.3.3: Activity Diagram for “Log in” use case

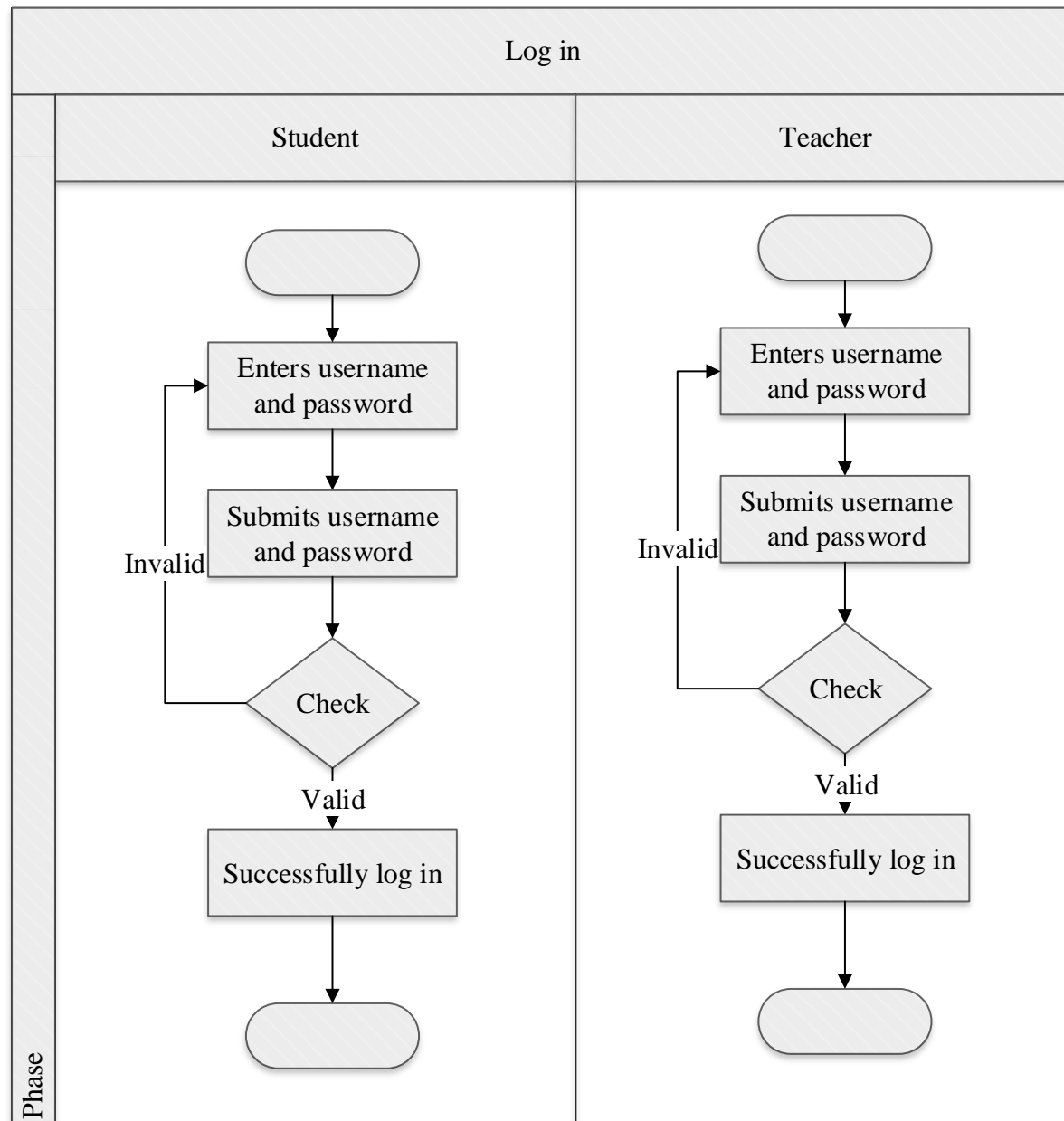
Swim Lane Diagram:

Figure 4.3.4: Swim Lane Diagram for “Log in” use case

Use Case 3: Log out
Activity Diagram:

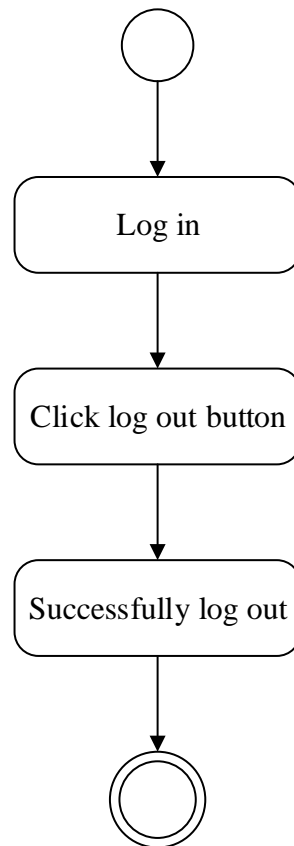


Figure 4.3.5: Activity Diagram for “Log out” use case

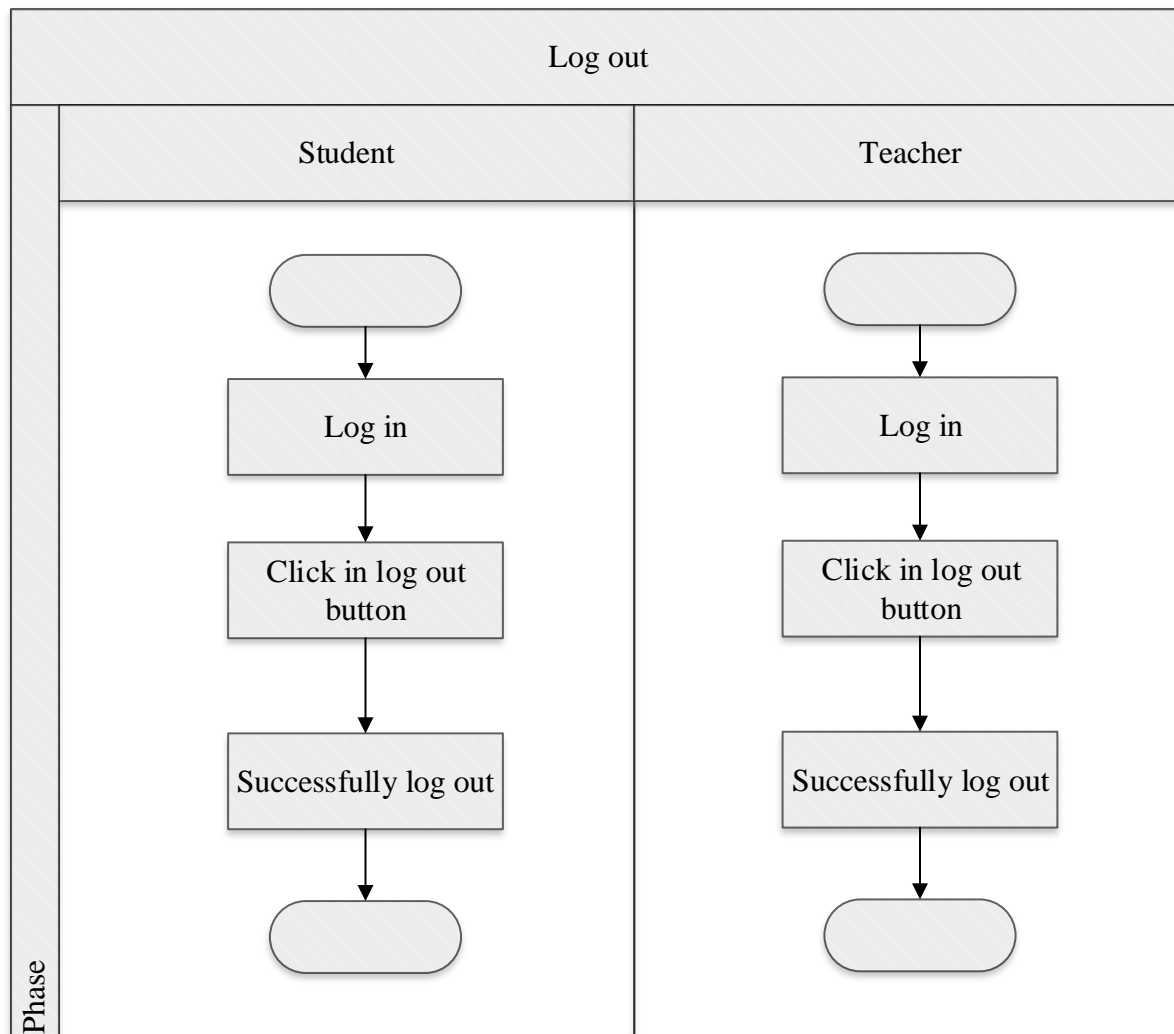
Swim Lane Diagram:

Figure 4.3.6: Swim Lane Diagram for “Log out” use case

Use Case 4: Change password
Activity Diagram:

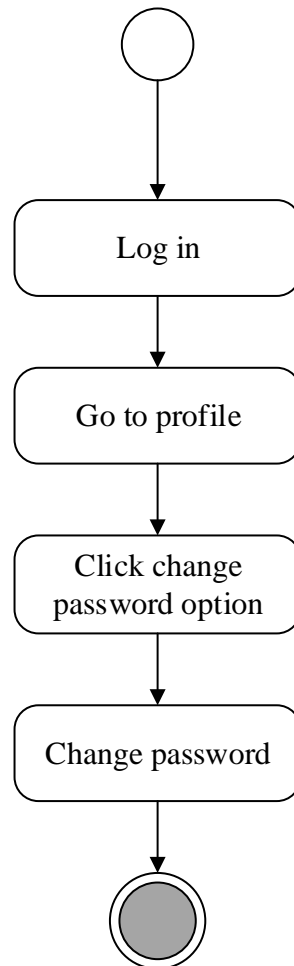


Figure 4.3.7: Activity Diagram for “Change password” use case

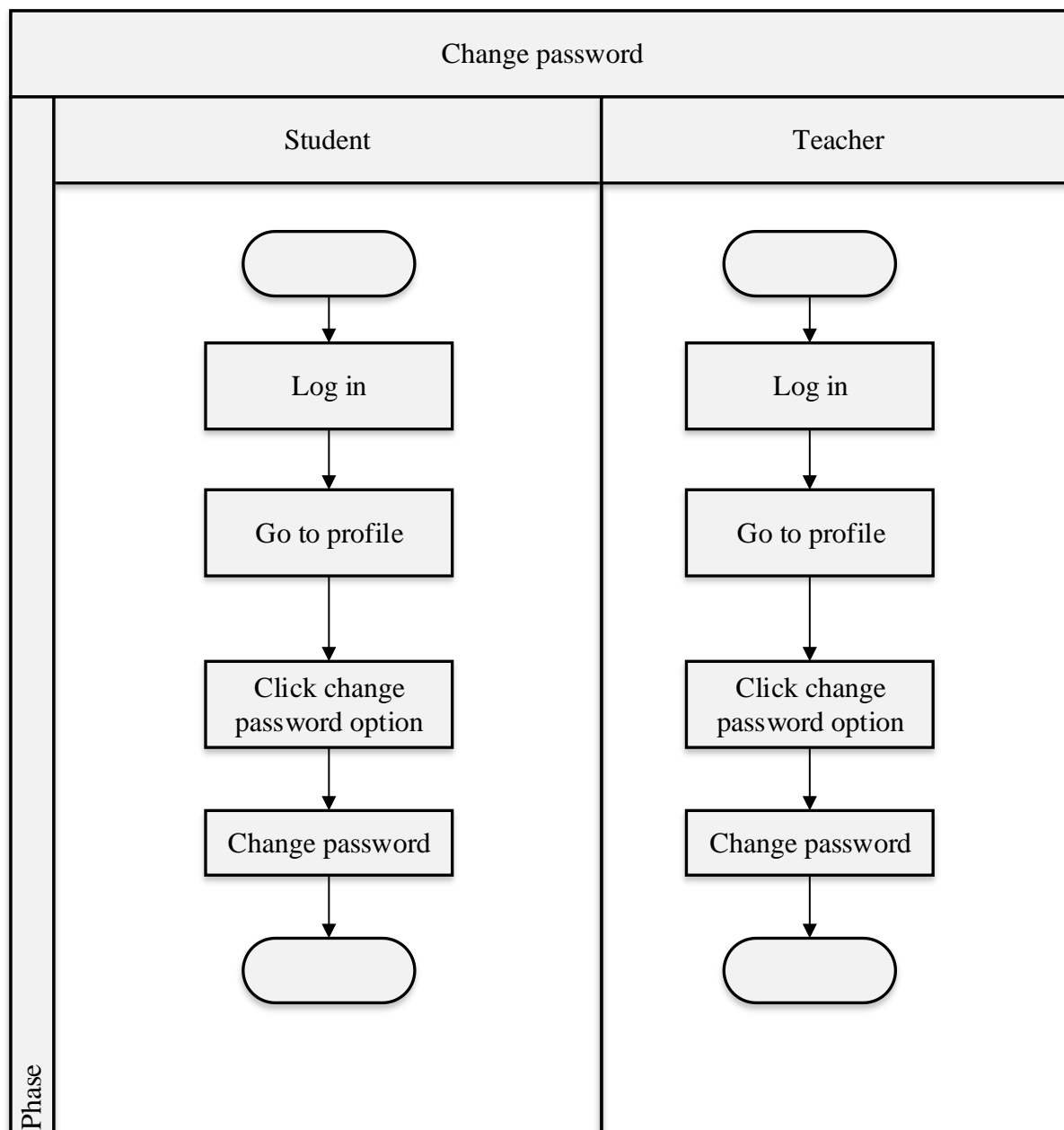
Swim Lane Diagram:

Figure 4.3.8: Swim Lane Diagram for “Change password” use case

Use Case 5: Change profile picture
Activity Diagram:

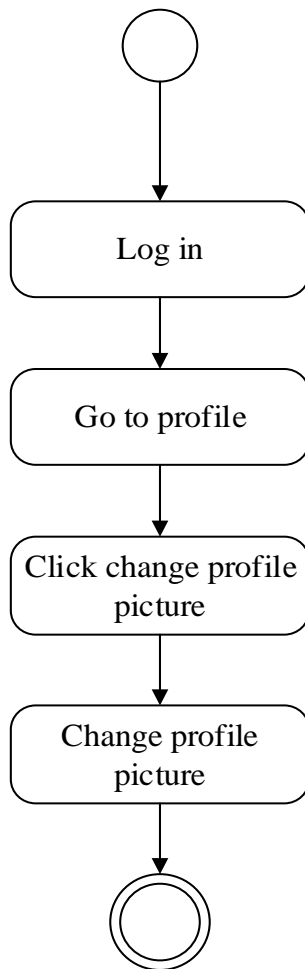


Figure 4.3.9: Activity Diagram for “Change profile picture” use case

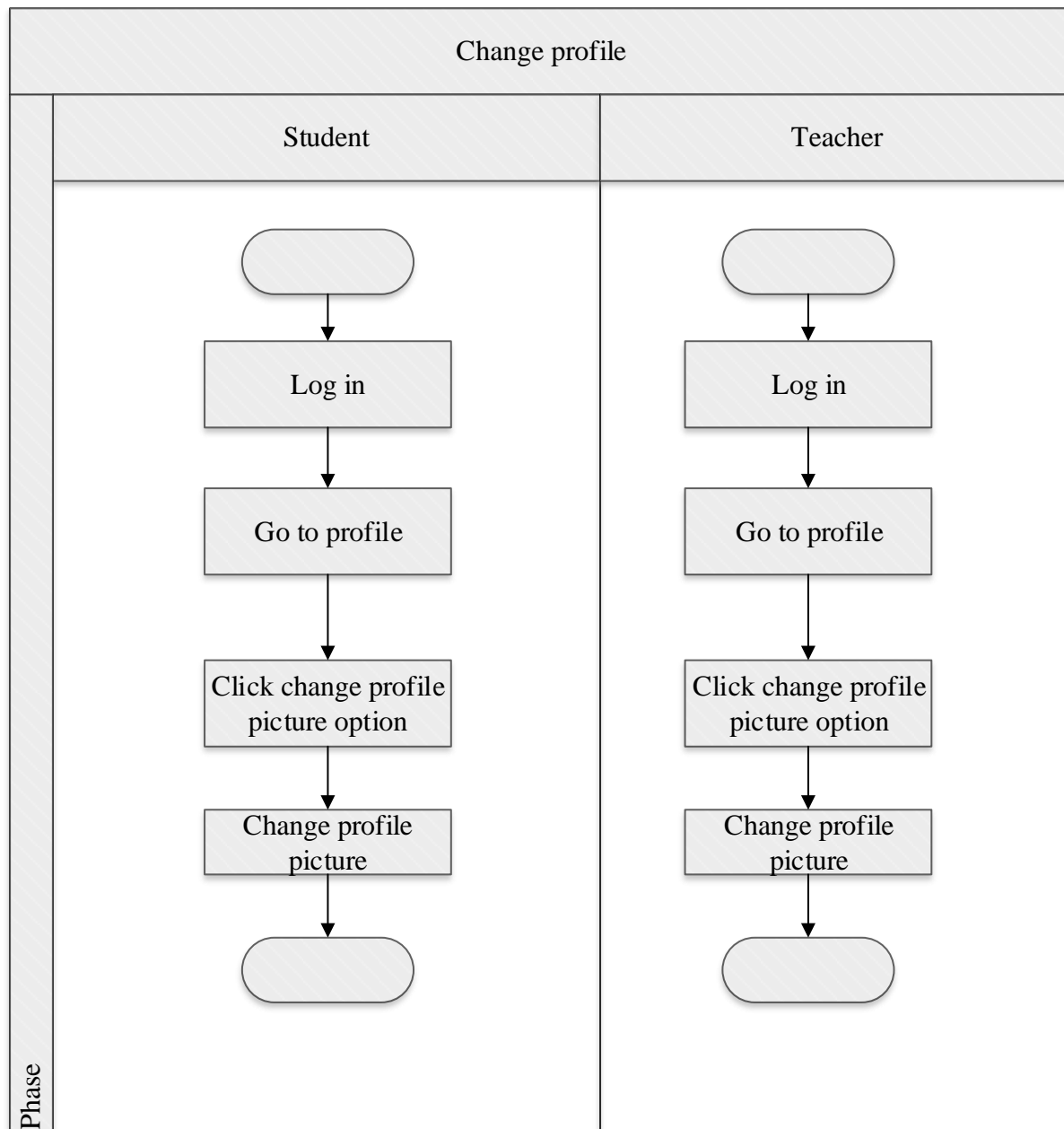
Swim Lane Diagram:

Figure 4.3.10: Swim Lane Diagram for “Change profile picture” use case

Use Case 6: Offering course
Activity Diagram

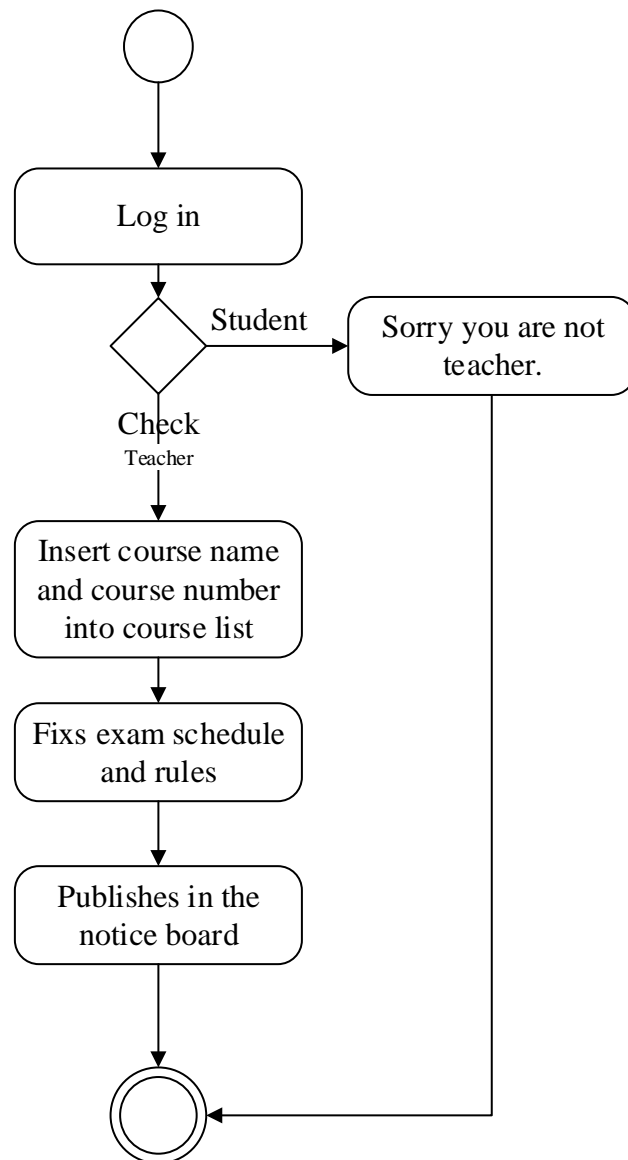


Figure 4.3.11: Activity Diagram for “Offering course” use case

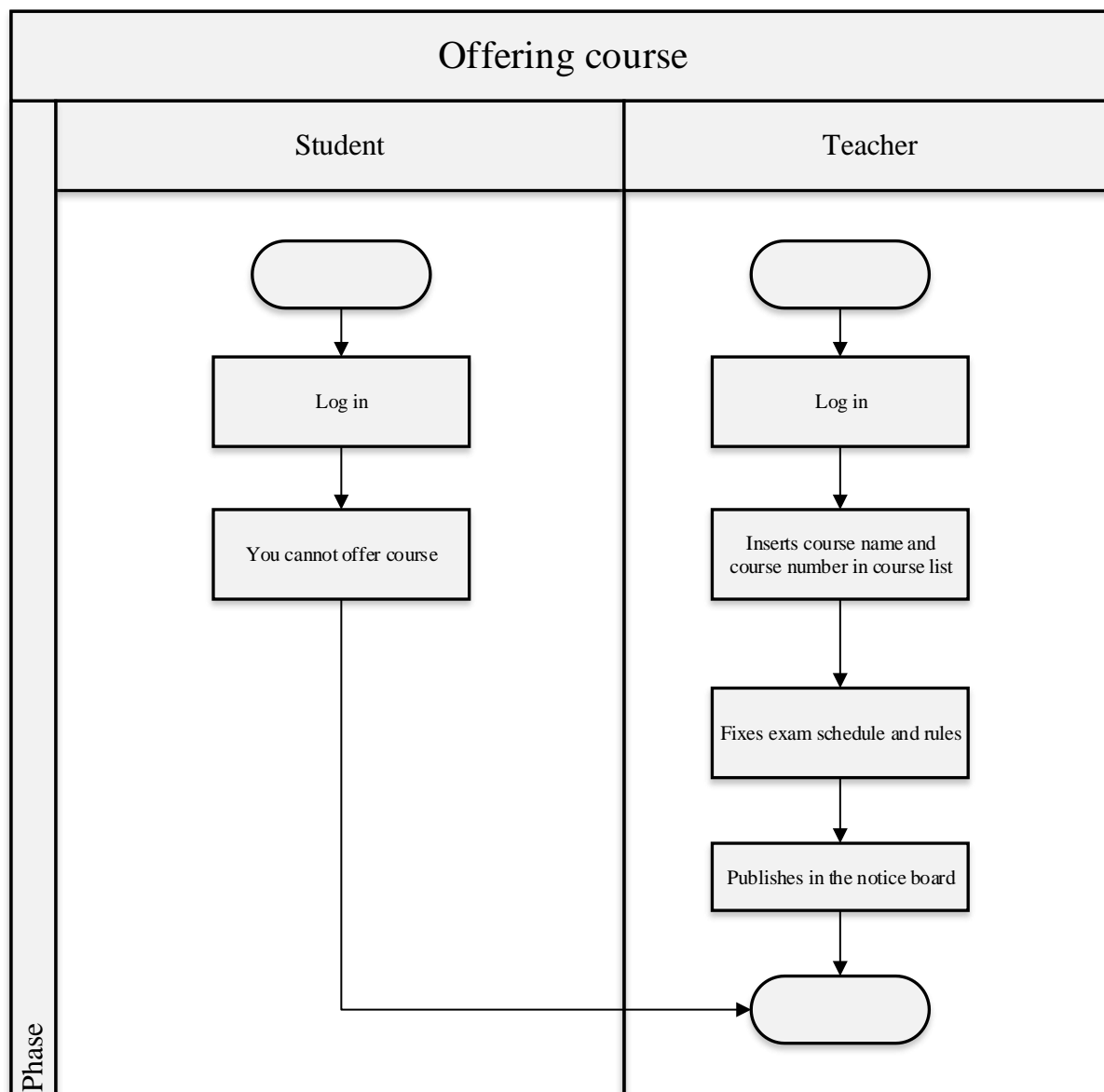
Swim Lane Diagram:

Figure 4.3.12: Swim Lane Diagram for “Offering course” use case

Use Case 7: Requesting course
Activity Diagram

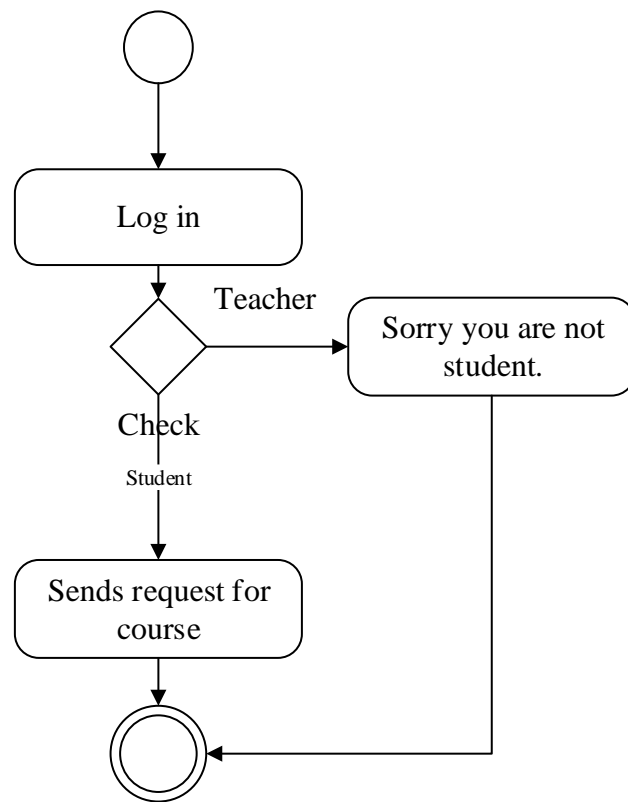


Figure 4.3.13: Activity Diagram for “Requesting course” use case

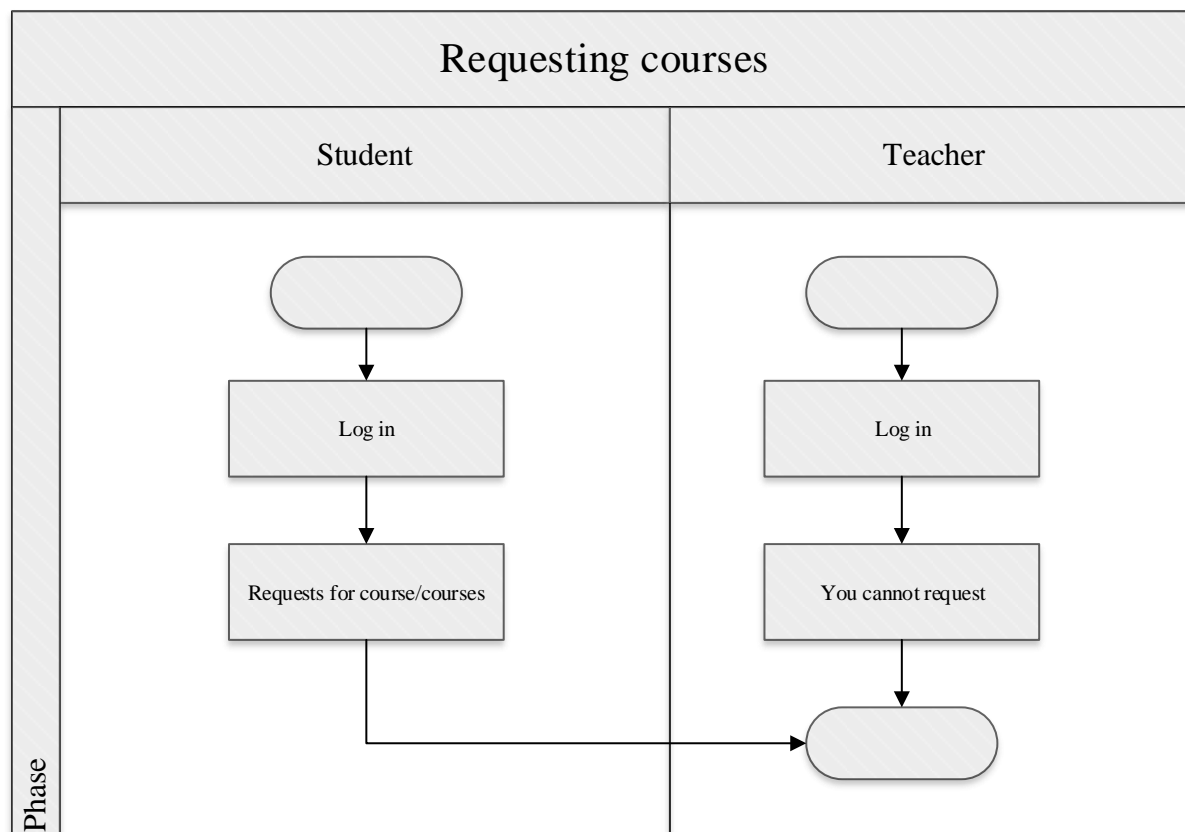
Swim Lane Diagram:

Figure 4.3.14: Swim Lane Diagram for “Requesting course” use case

Use Case 8: Approving or rejecting student
Activity Diagram

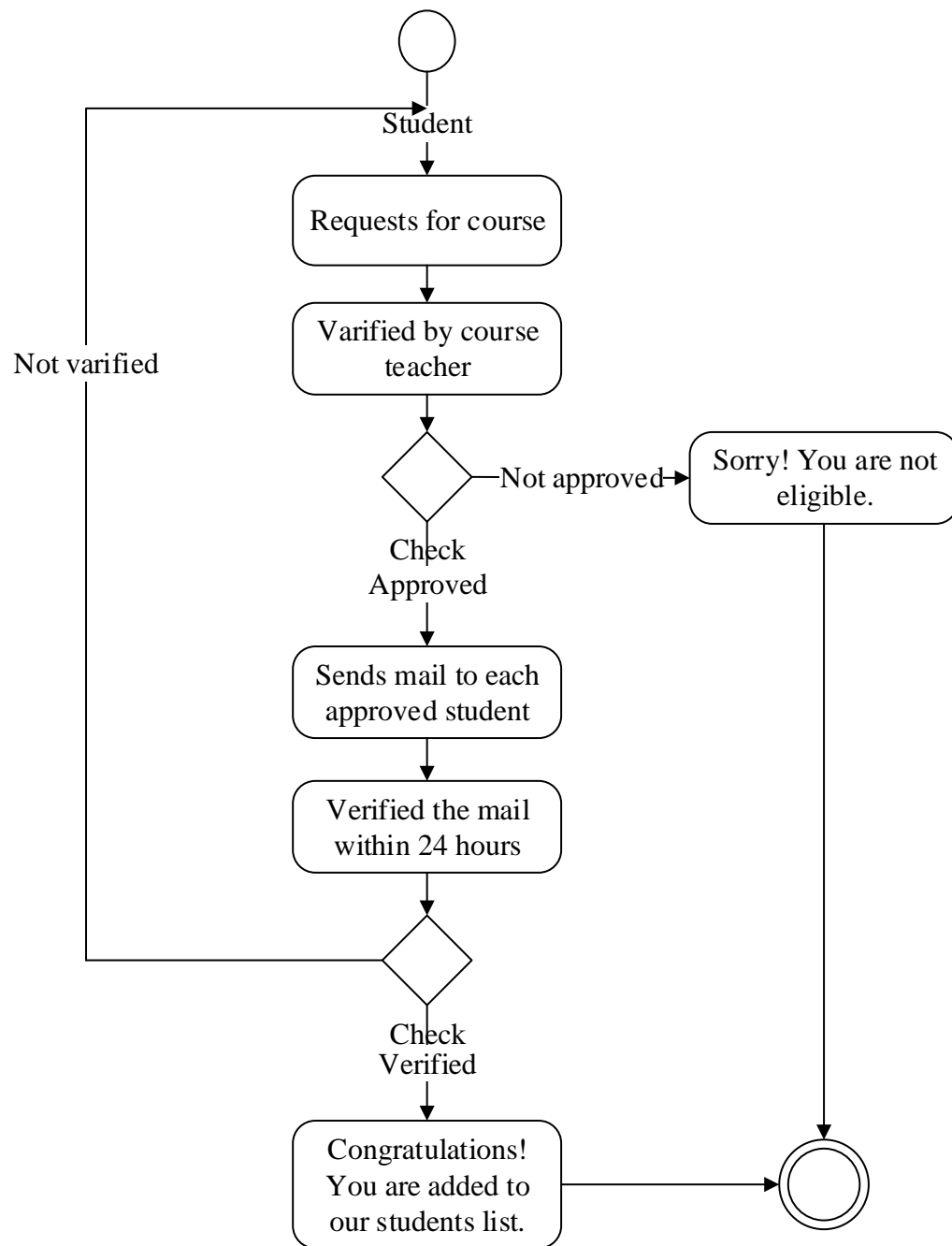


Figure 4.3.15: Activity Diagram for “Approving or rejecting student” use case

Swim Lane Diagram:

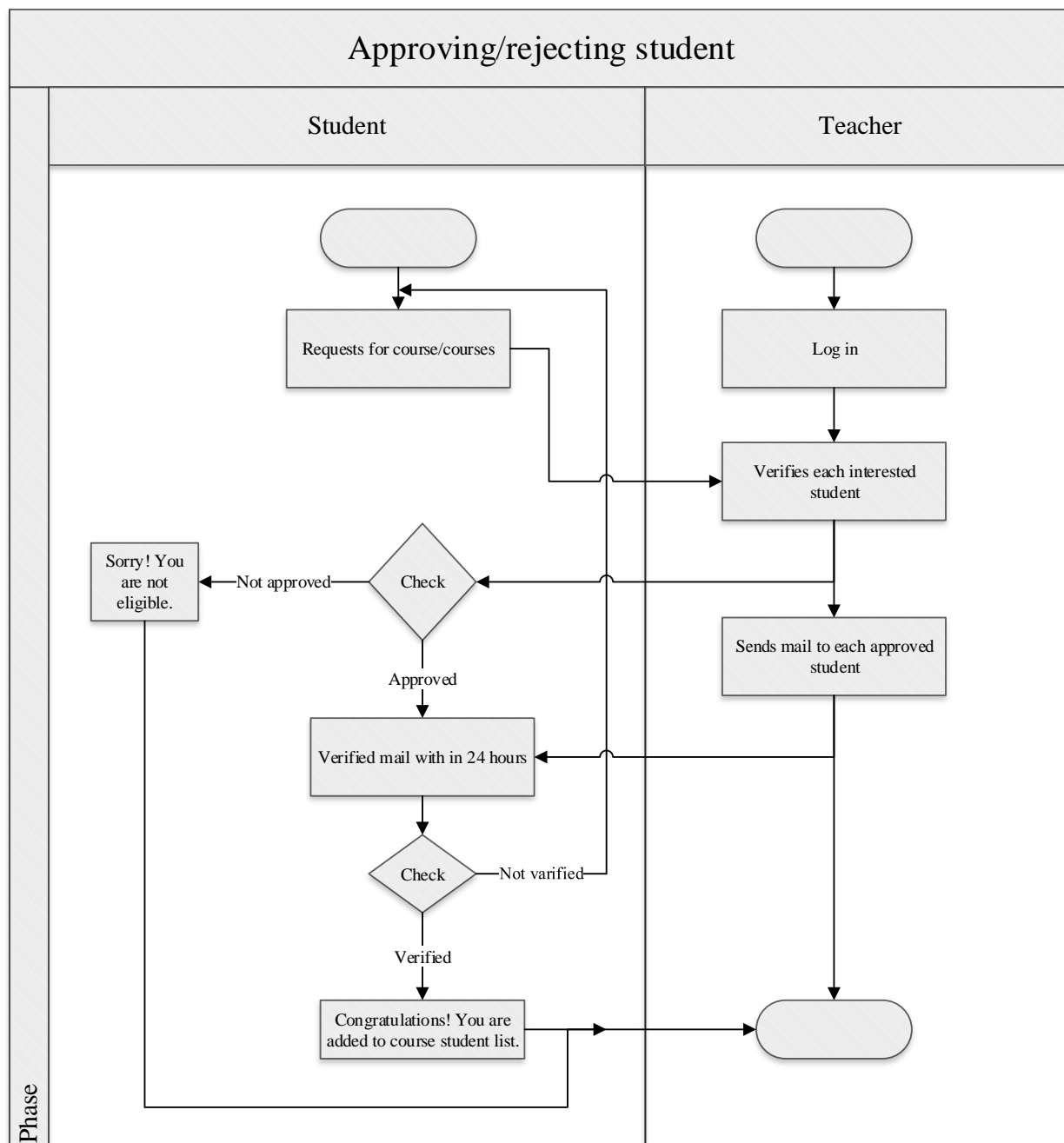


Figure 4.3.16: Swim Lane Diagram for “Approving or rejecting student” use case

Use Case 9: Uploading questions and answers
Activity Diagram

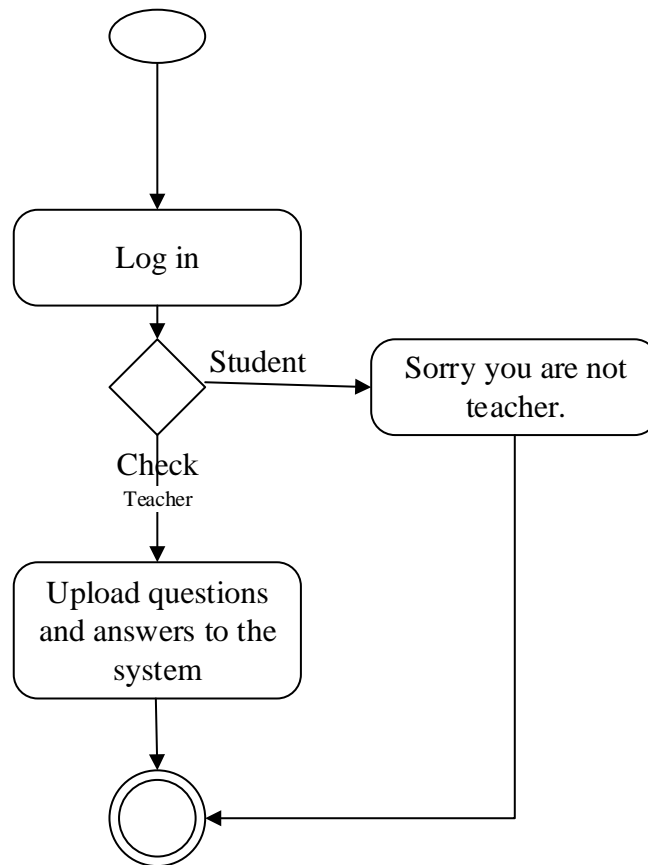


Figure 4.3.17: Activity Diagram for “Uploading questions and solutions” use case

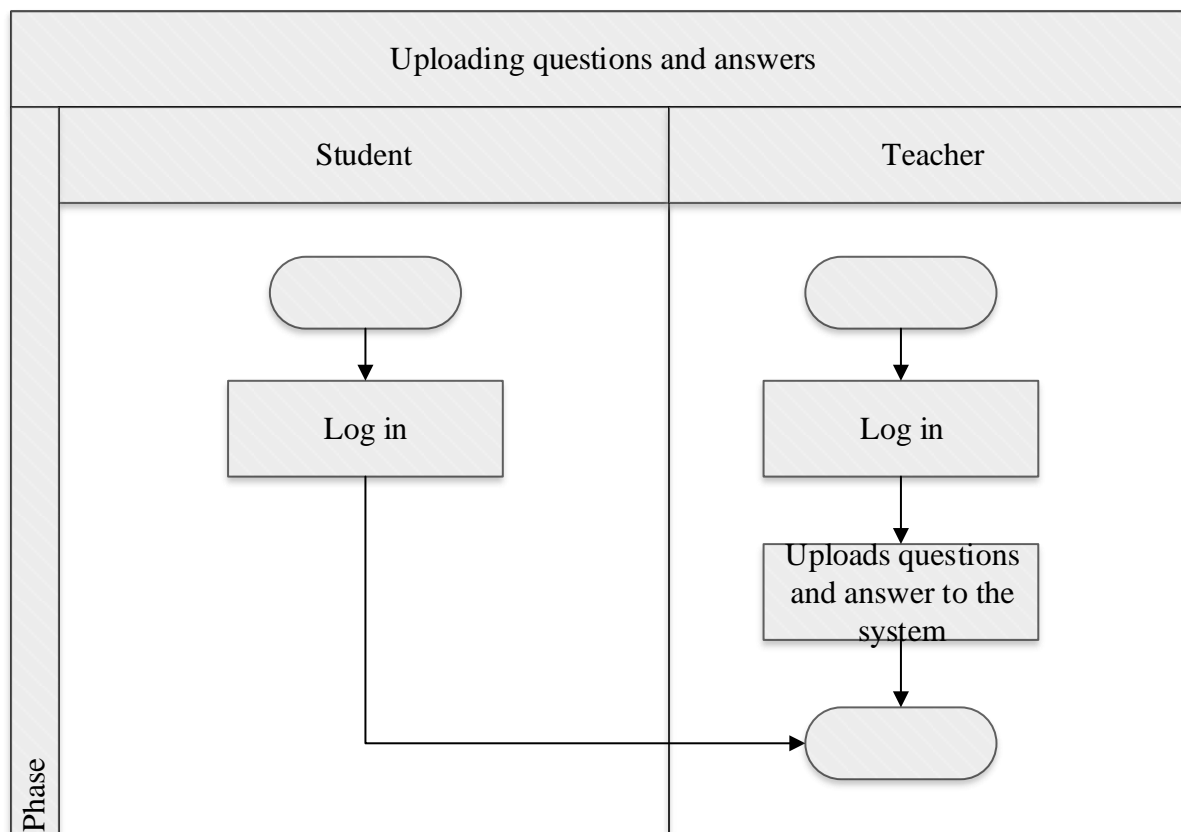
Swim Lane Diagram:

Figure 4.3.18: Swim Lane Diagram for “Uploading questions and solutions” use case

Use Case 10: Generating random question for each student
Activity Diagram:

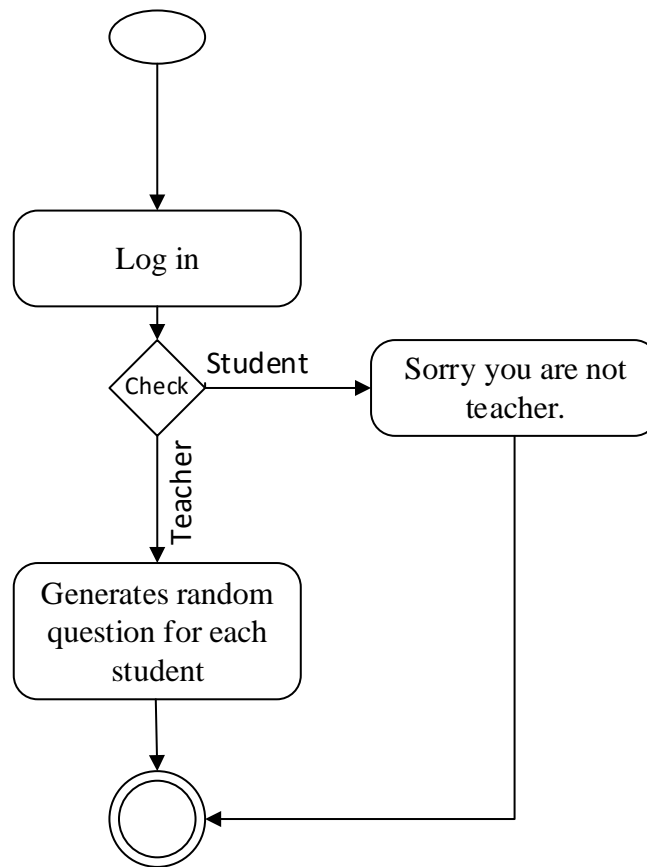


Figure 4.3.19: Activity Diagram for “Generating random question for each student” use case

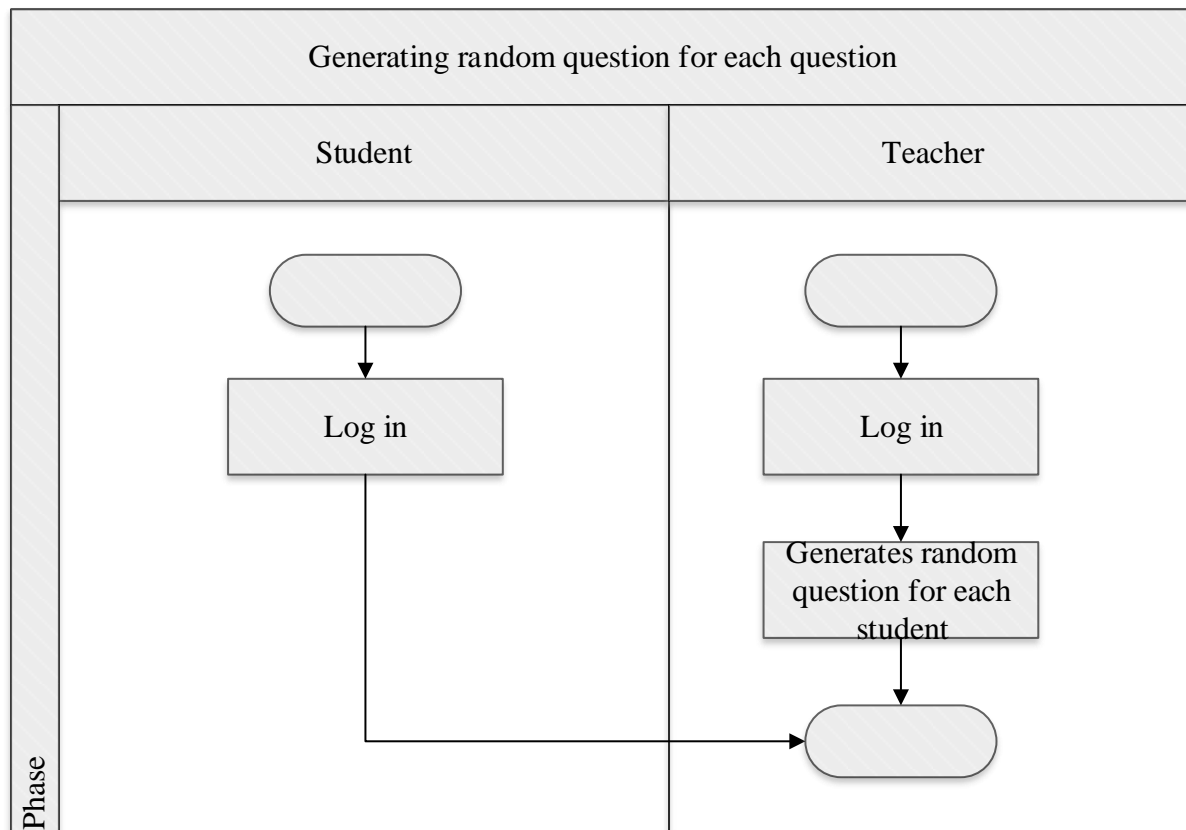
Swim Lane Diagram:

Figure 4.3.20: Swim Lane Diagram for “Generating random question for each student” use case

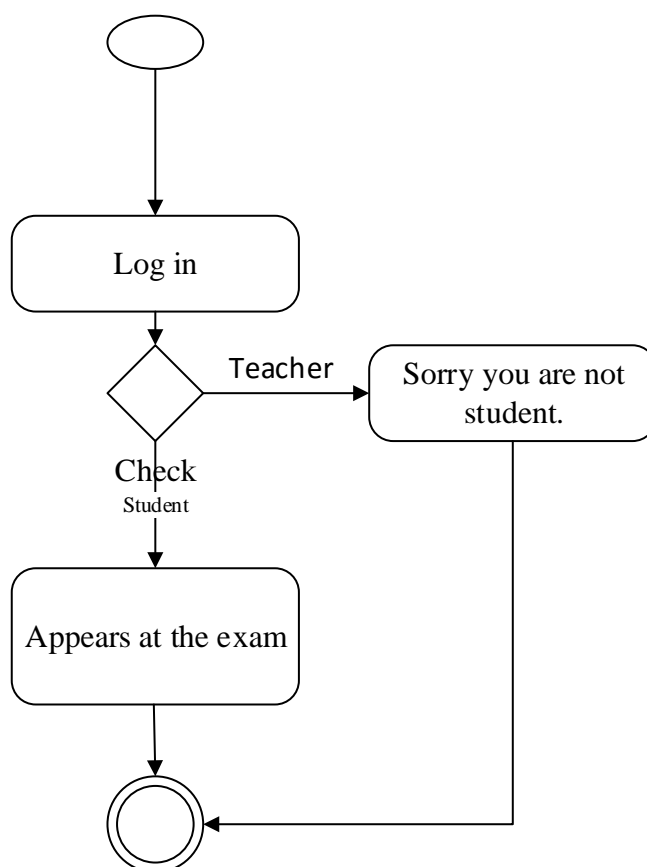
Use Case 11: Appearing at the exam**Activity Diagram:**

Figure 4.3.21: Activity Diagram for “Appearing at the exam” use case

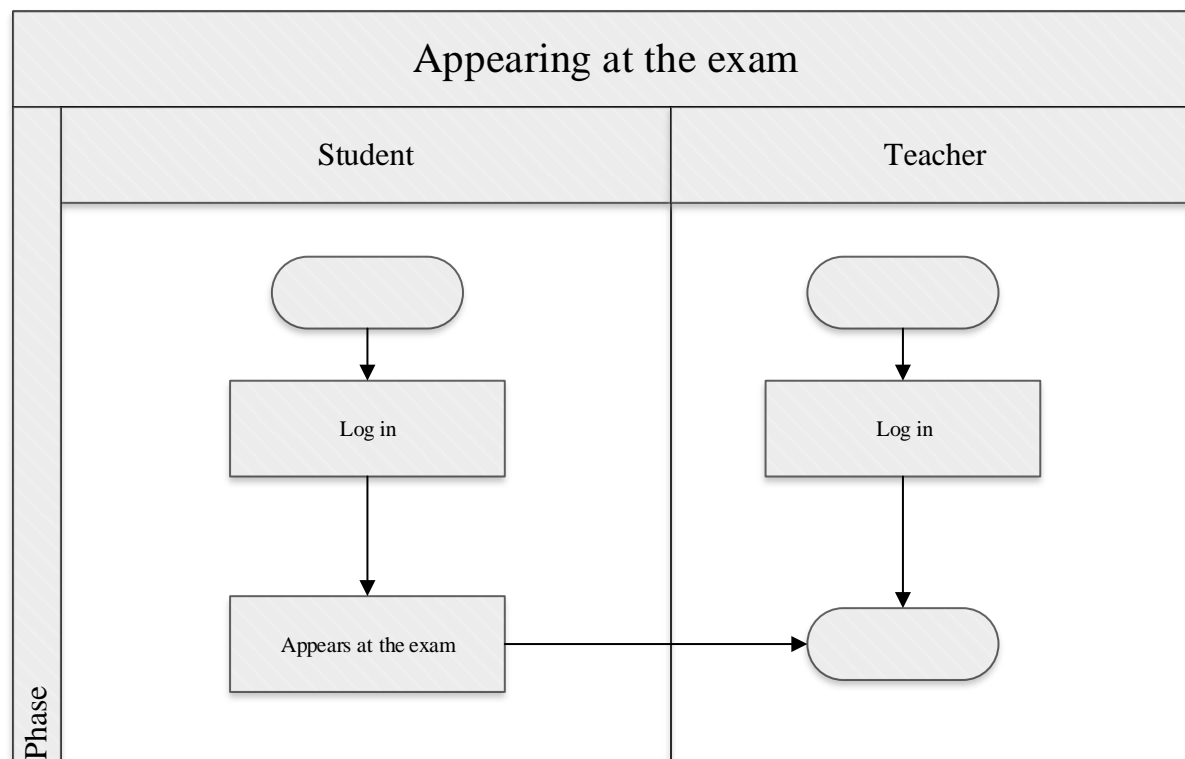
Swim Lane Diagram:

Figure 4.3.22: Swim Lane Diagram for “Appearing at the exam” use case

Use Case 12: Viewing the scores
Activity Diagram

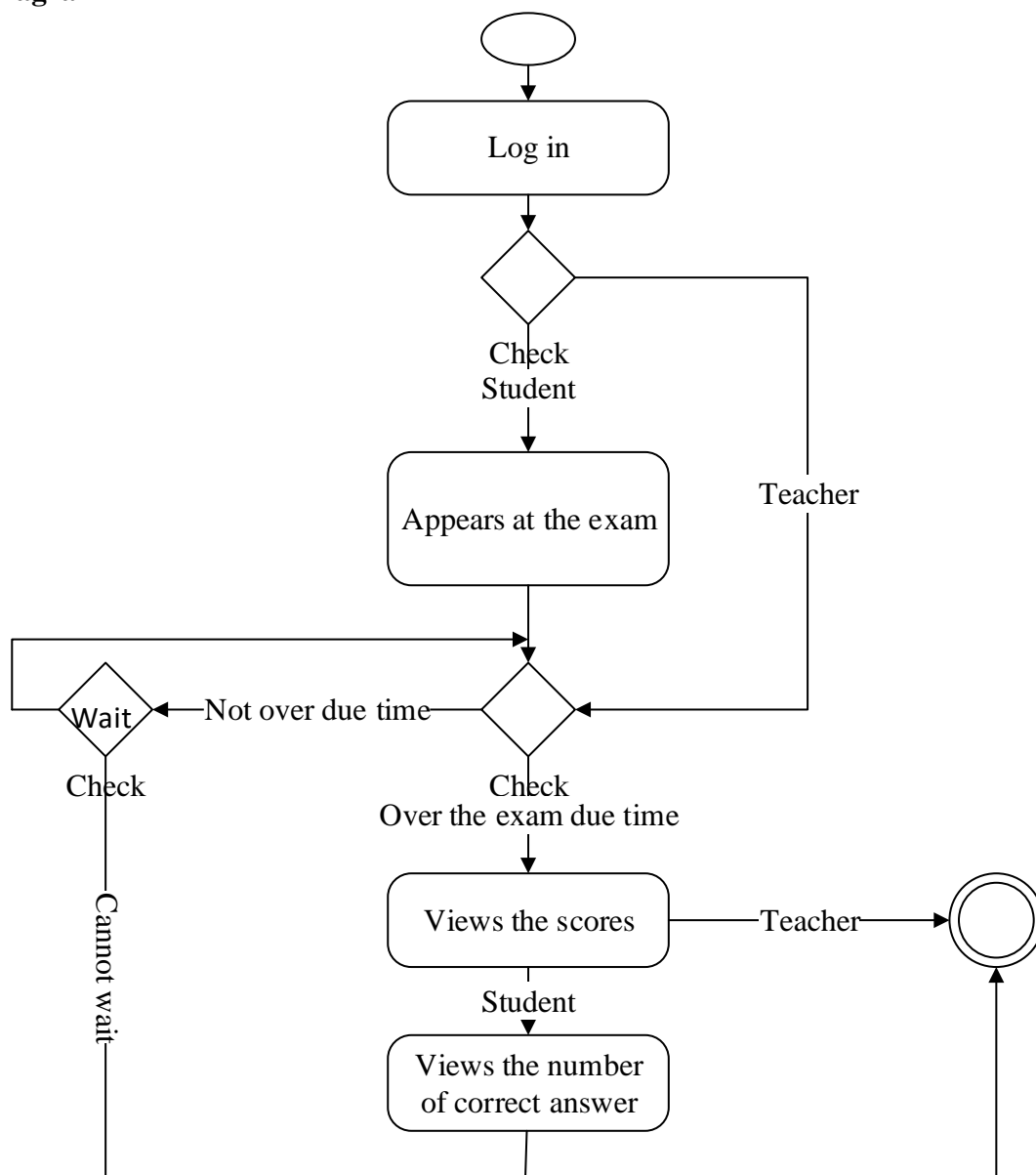


Figure 4.3.23: Activity Diagram for “Viewing the scores” use case

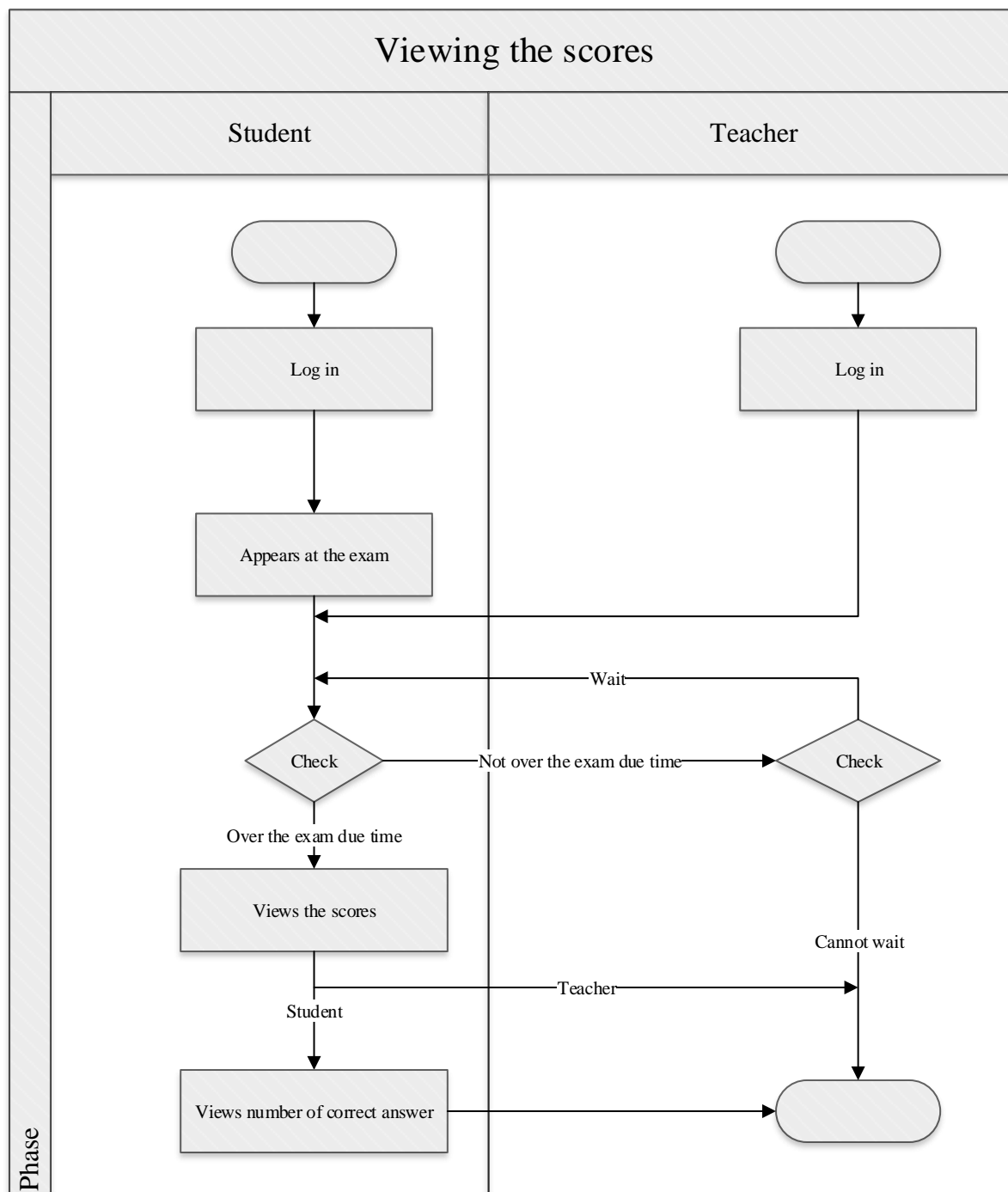
Swim Lane Diagram:

Figure 4.3.24: Swim Lane Diagram for “Viewing the scores” use case

Use Case 13: Asking answer
Activity Diagram

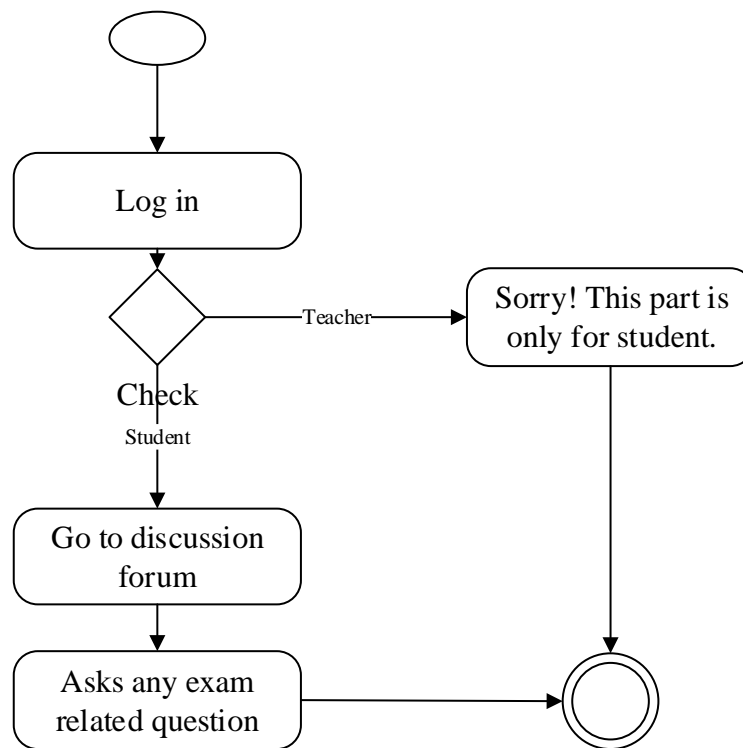


Figure 4.3.25: Activity Diagram for “Asking question” use case

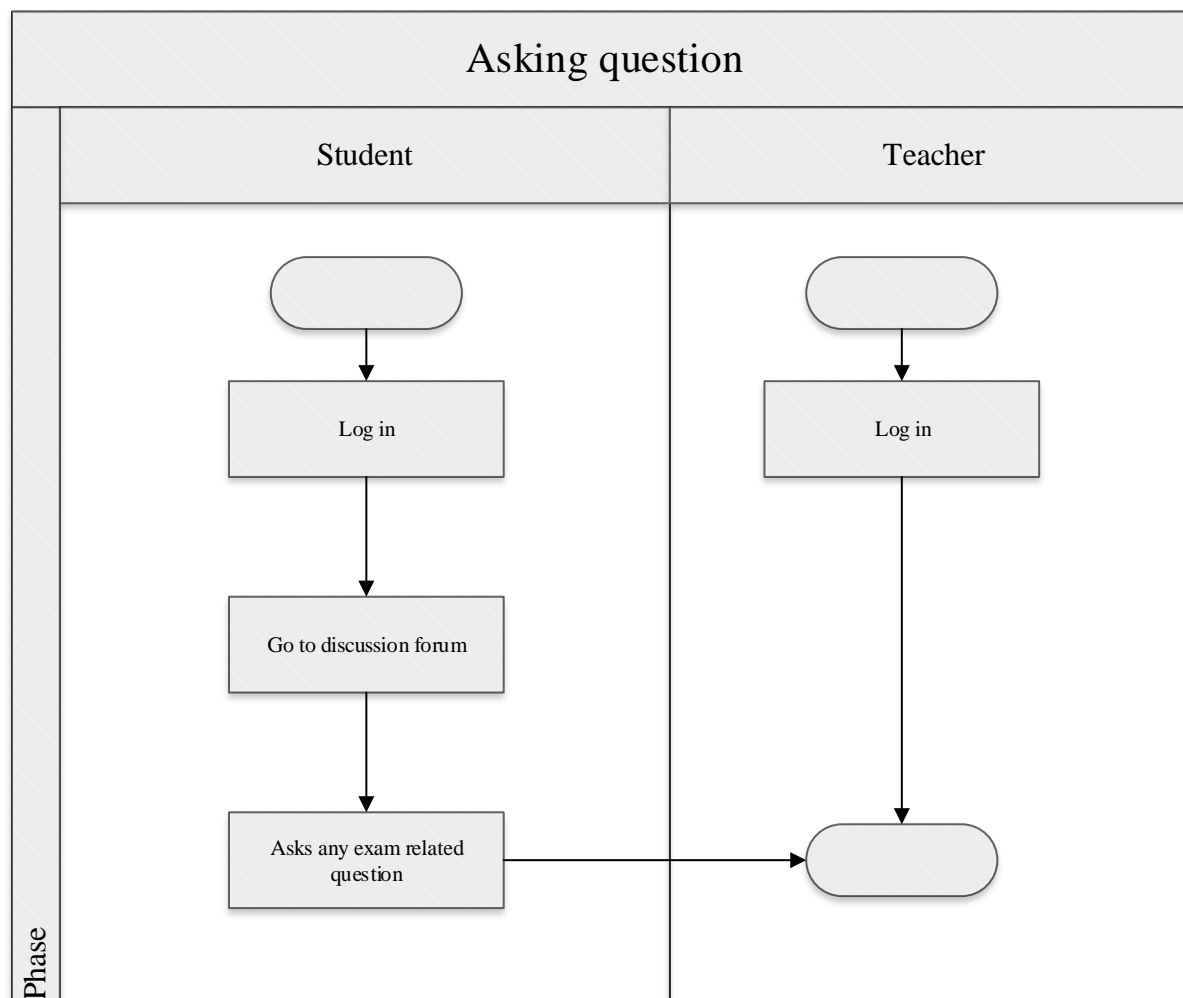
Swim Lane Diagram:

Figure 4.3.26: Swim Lane Diagram for “Asking question” use case

Use Case 14: Answering question
Activity Diagram

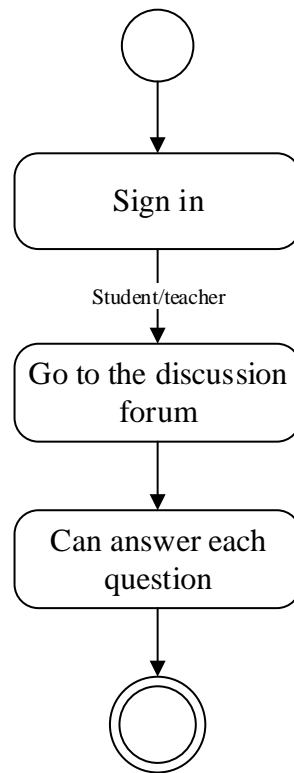


Figure 4.3.27: Activity Diagram for “Answering question” use case

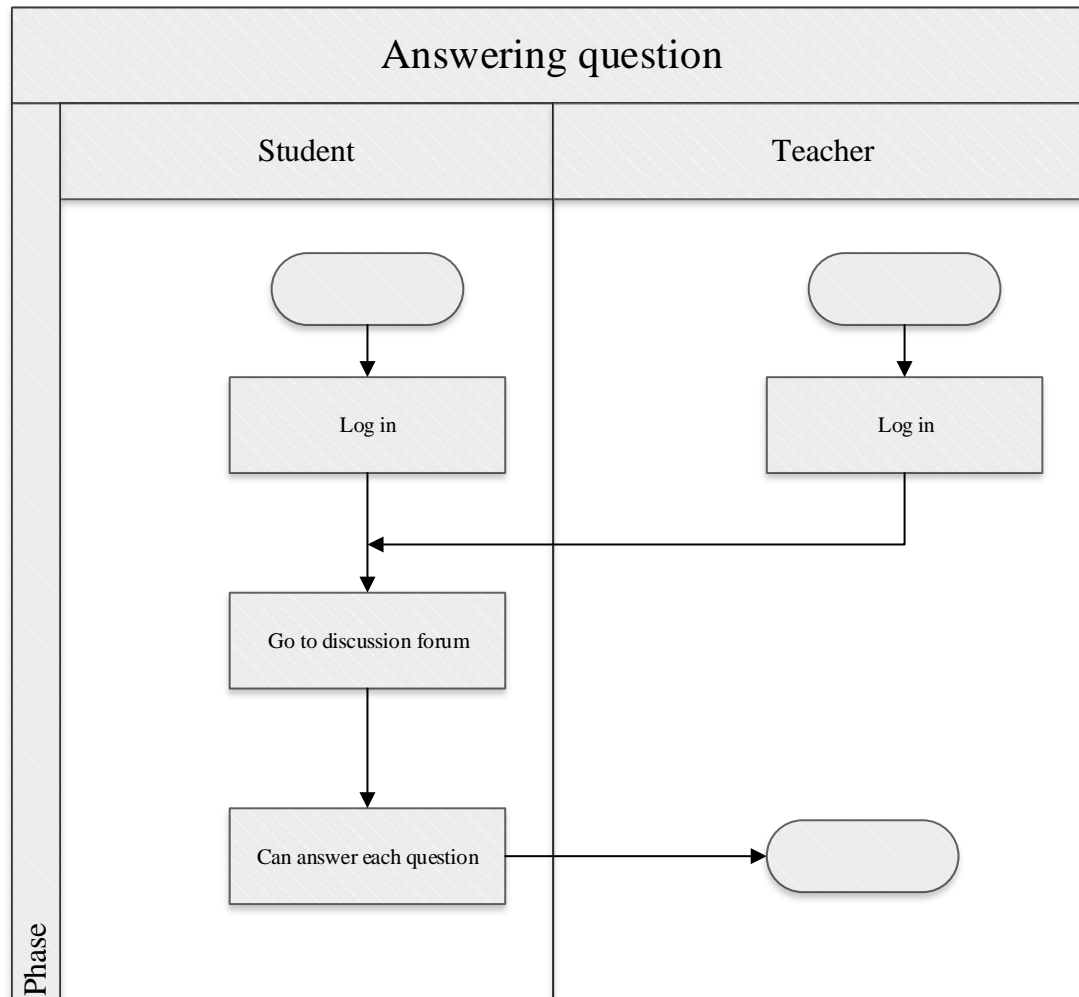
Swim Lane Diagram:

Figure 4.3.28: Swim Lane Diagram for “Answering question” use case

Use Case 15: Replying each answer or reply
Activity Diagram:

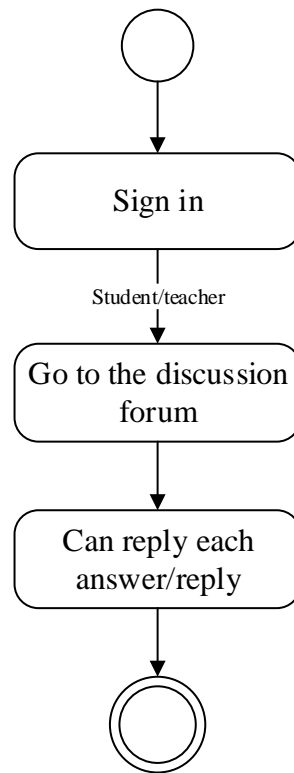


Figure 4.3.29: Activity Diagram for “Replying each answer/reply” use case

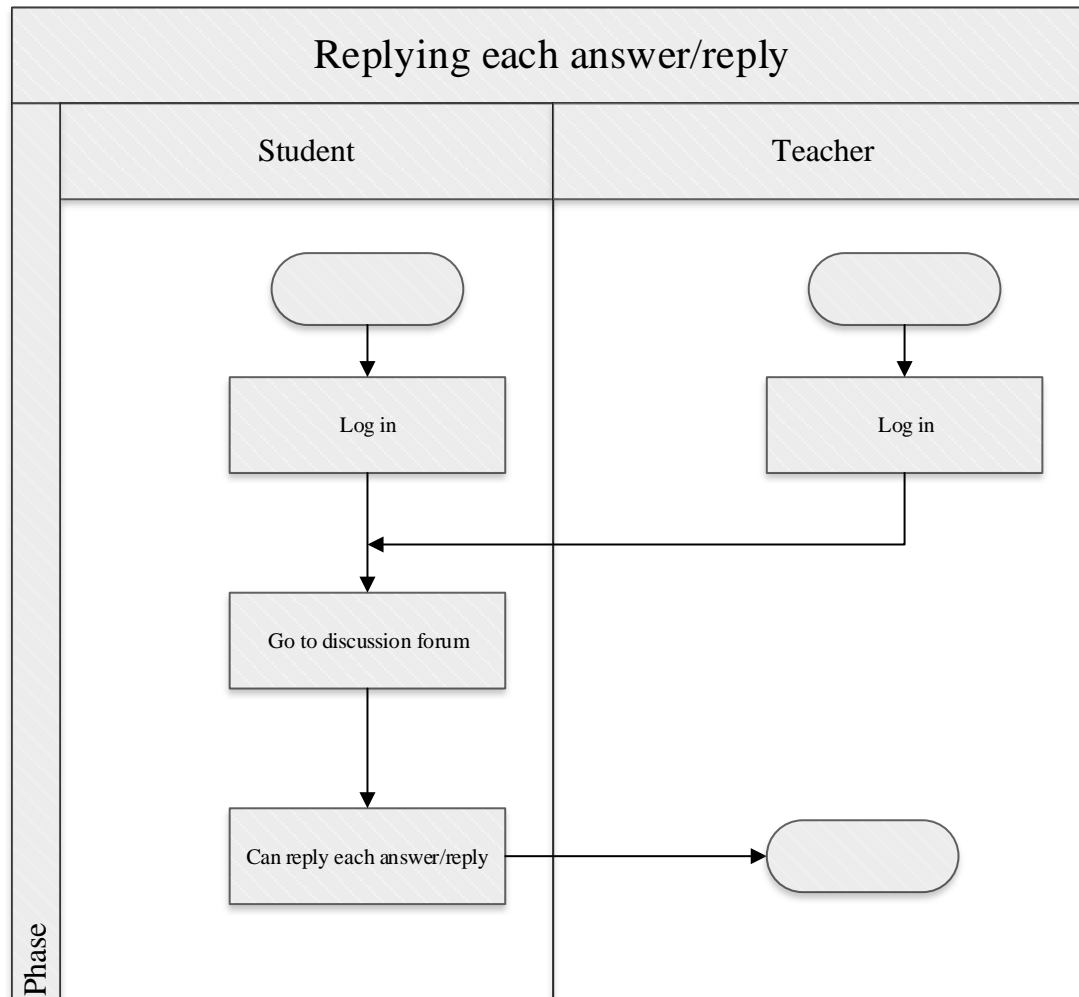
Swim Lane Diagram:

Figure 4.3.30: Swim Lane Diagram for “Replying each answer/reply” use case

Use Case 16: Rating each answer or reply
Activity Diagram:

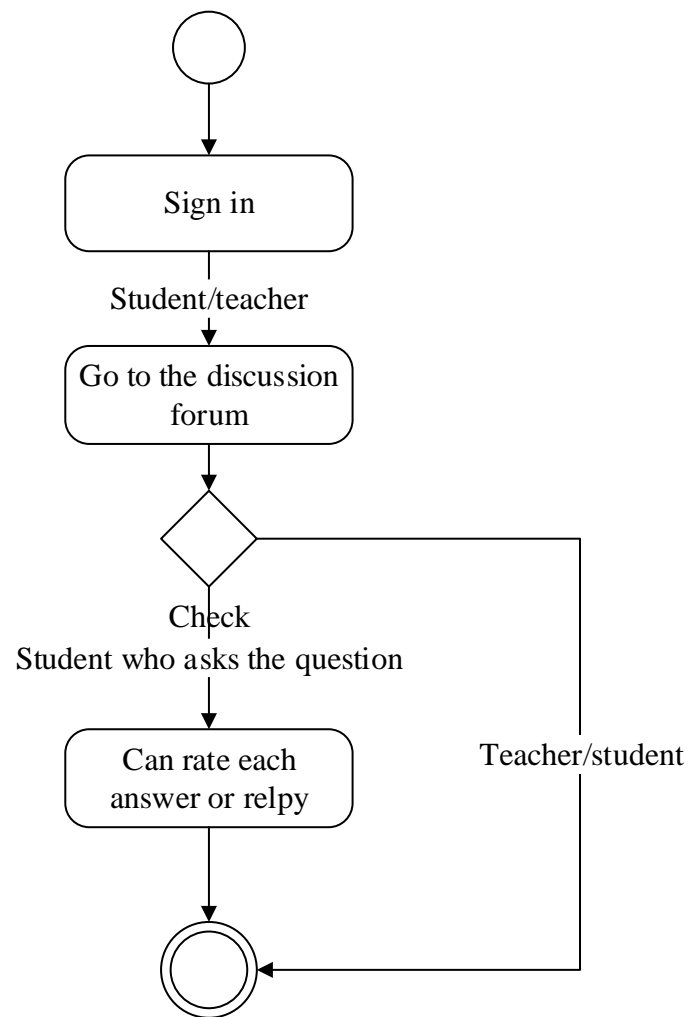


Figure 4.3.31: Activity Diagram for “Rating each answer/reply” use case

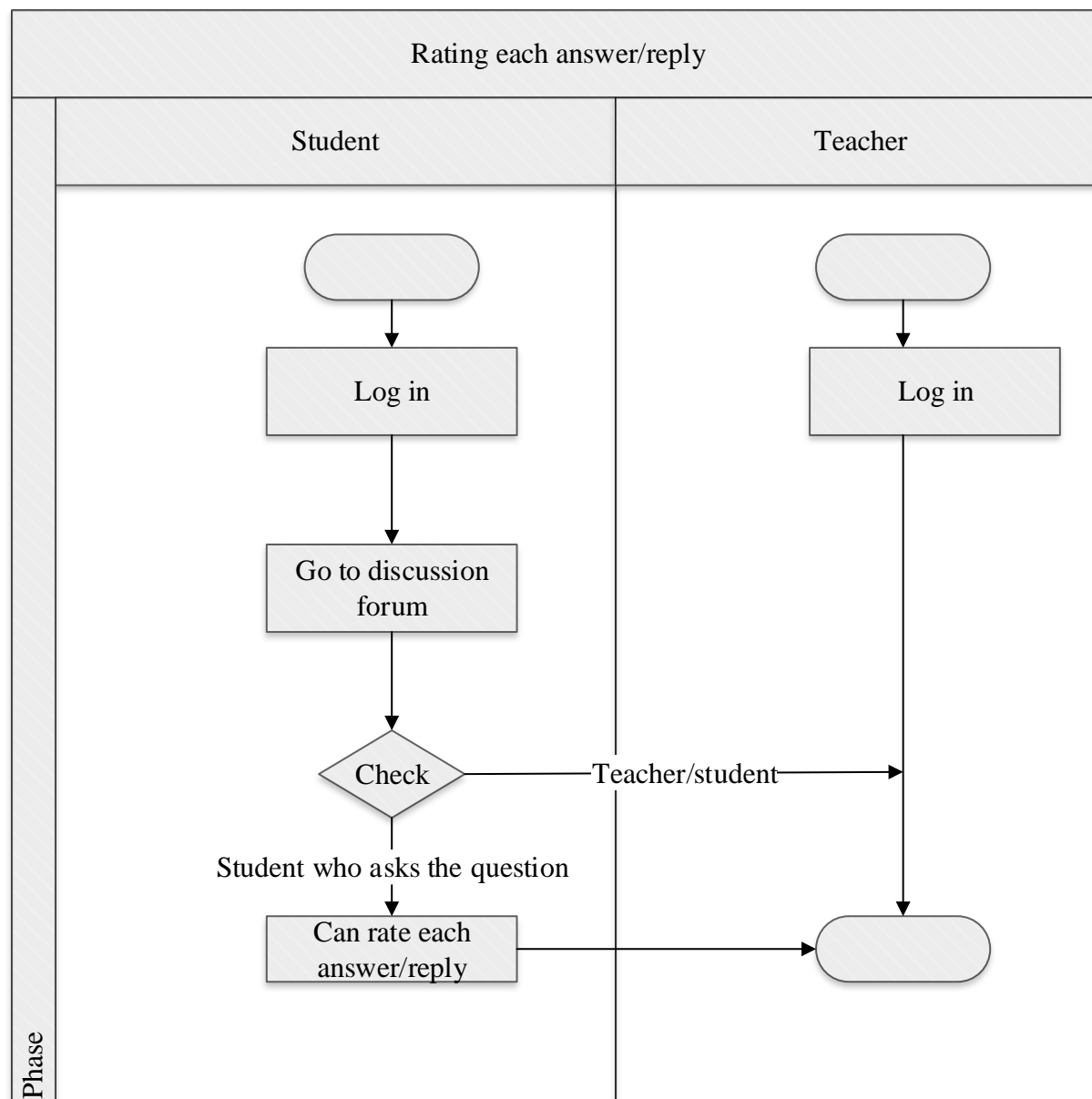
Swim Lane Diagram:

Figure 4.3.32: Swim Lane Diagram for “Rating each answer/reply” use case

Use Case 16: Approving correct answer
Activity Diagram:

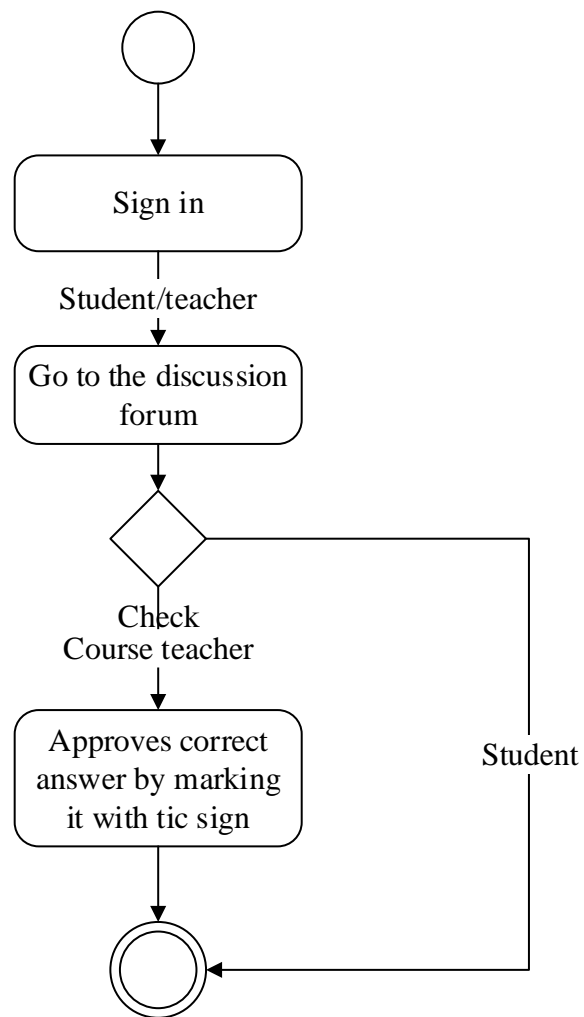


Figure 4.3.33: Activity Diagram for “Approving correct answer” use case

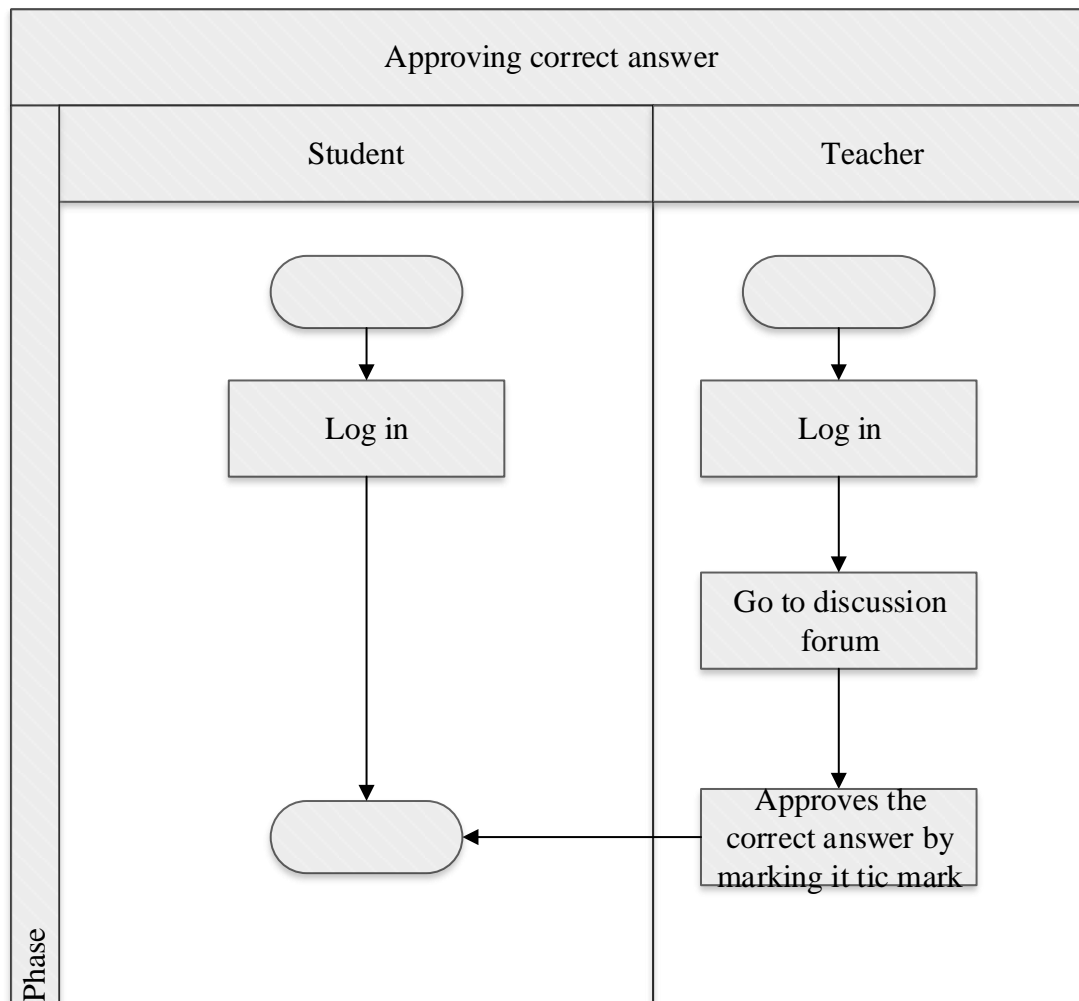
Swim Lane Diagram:

Figure 4.3.34: Swim Lane Diagram for “Approving correct answer” use case

Chapter 5

Data Model

In this chapter we discuss the data model of our proposed system.

5.1 Data Modeling Concept

A data model visually represents the nature of data, business rules governing the data, and how it will be organized in the database. A data model is comprised of two parts logical design and physical design. Data Models are created in either Top Down Approach or Bottom-Up Approach. In Top-Down Approach, data models are created by understanding and analyzing the business requirements. In Bottom Up Approach, data models are created from existing databases. For our proposed system we are used Top-Down Approach.

5.2 Data Objects

A database object in a relational database is a data structure used to either store or reference data. The most common object that people interact with is the table. Other objects are indexes, stored procedures, sequences, views and many more.

5.2.1 Grammatical Parsing (Noun Identity)

We have identified all nouns by parsing from our proposed user scenario and each noun may be in either problem space or solution space. A noun which is in solution space can be attribute or data object.

NID	Noun	Problem/solution space	Attributes
1.	Online Registration System	P	---
2.	Students	S	s_id, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 49, 50,
3.	Authenticated user	P	---
4.	Course menu	P	---
5.	Registration Period	P	---
6.	Internet	P	---
7.	Course Teachers	S	ct_id, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27
8.	Online Exam System	P	---

9.	Web-based portal	P	---
10.	Educational network	P	---
11.	Exam	S	e_id, 31, 32, 39, 40, 41, 44
12.	Discussion forum	P	---
13.	Registration process	P	---
14.	Profile	P	---
15.	Email info	S	---
16.	First name	S	---
17.	Last name	S	---
18.	Gender	S	---
19.	Date of birth	S	---
20.	Local address	S	---
21.	Permanent address	S	---
22.	Institute name	S	---
23.	Enrolled year	S	---
24.	Username	S	---
25.	Password	S	---
26.	Contact info	S	---
27.	Profile info	S	---
28.	Course name	S	---
29.	Course id	S	---
30.	Course list	S	28, 29
31.	Exam date	S	---
32.	Exam information	S	---
33.	Websites notice board	P	---
34.	Information of the student	P	---
35.	Supreme authority	P	---
36.	Email	S	em_id, content, source, destination
37.	Questions (MCQ)	S	q_id, question, option1, option2, option3, option4, 38
38.	Solution	S	---
39.	Number of questions	S	---
40.	Positive marks	S	---
41.	Negative marks	S	---
42.	Right answer	P	---
43.	Wrong answer	P	---
44.	Exam time	S	---
45.	Schedule time	P	---
46.	Script	S	sc_id, q_id, mark_answer
47.	Period	P	---
48.	Submission	P	---

49.	Total scores	S	---
50.	Number of correct answer	S	---
51.	Mark sheets	S	s_id, e_id, 49
52.	Exam related questions	S	erq_id, er_question
53.	Answer	S	ans_id, 54, 55, 57
54.	Reply	S	a_id, comment
55.	Merit	S	---
56.	Correct answer	P	---
57.	Tic sign	S	---

5.2.2 Identify Data Object

Nouns having attributes are selected as data object. So the data objects are

1. Student

- i. s_id
- ii. email info
- iii. first name
- iv. last name
- v. gender
- vi. date of birth
- vii. local address
- viii. permanent address
- ix. institute name
- x. enrolled year
- xi. username
- xii. password
- xiii. contact info
- xiv. profile picture
- xv. total scores
- xvi. number of correct answer

2. Course Teachers

- i. ct_id
- ii. email info
- iii. first name
- iv. last name
- v. gender
- vi. date of birth
- vii. local address
- viii. permanent address
- ix. institute name
- x. enrolled year

- xi. username
 - xii. password
 - xiii. contract info
 - xiv. profile picture
- 3. Exam
 - i. e_id
 - ii. exam date
 - iii. exam information
 - iv. exam time
 - v. positive marks
 - vi. negative marks
 - vii. number of questions
- 4. Course list
 - i. course_id
 - ii. course name
- 5. Questions
 - i. q_id
 - ii. question
 - iii. option1
 - iv. option2
 - v. option3
 - vi. option4
 - vii. solution
- 6. Script
 - i. sc_id
 - ii. q_id
 - iii. marked answer
- 7. Mark sheets
 - i. e_id
 - ii. s_id
 - iii. total scores
- 8. Exam related questions
 - i. erq_id
 - ii. er_question

9. Answer

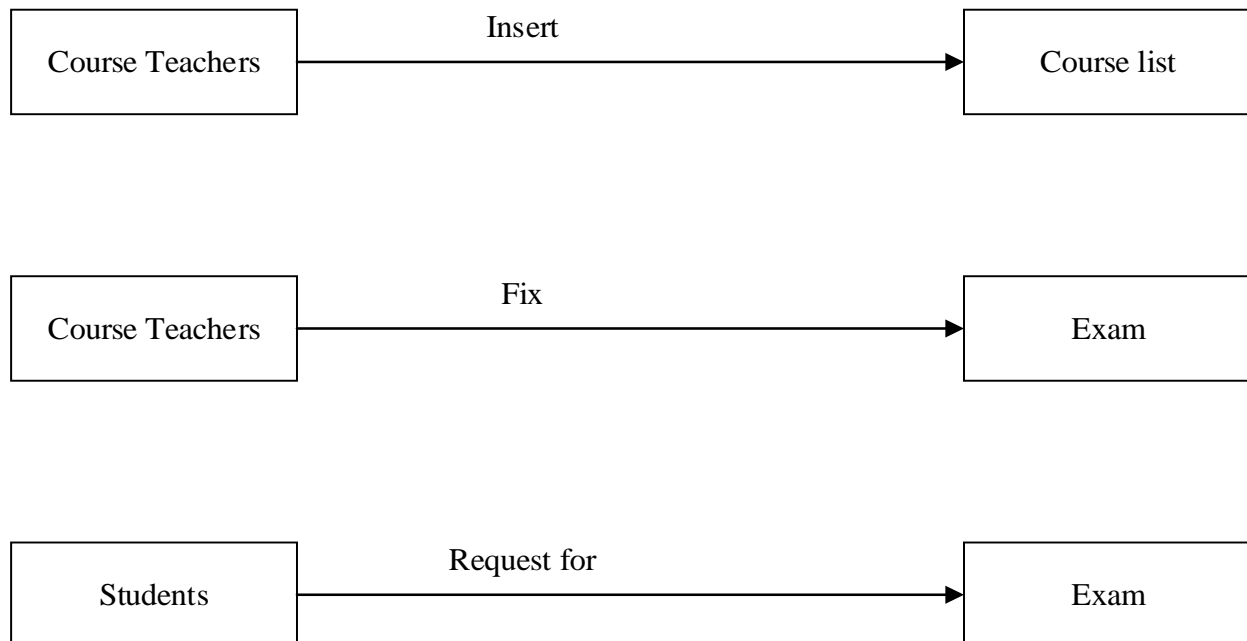
- i. a_comment
- ii. merit
- iii. tic sign
- iv. ans_id

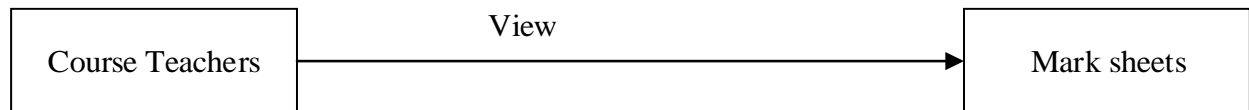
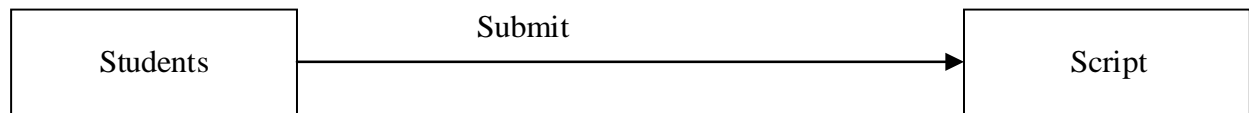
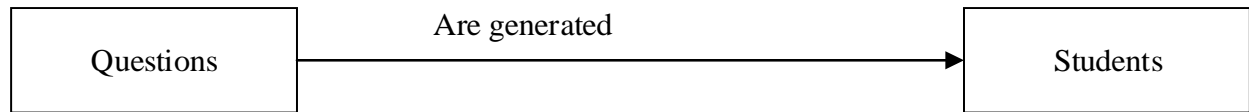
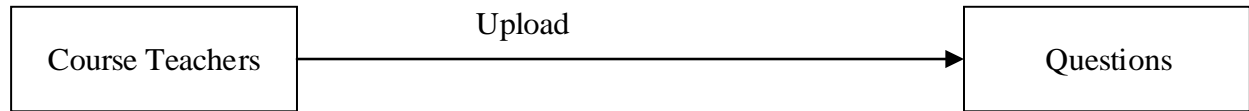
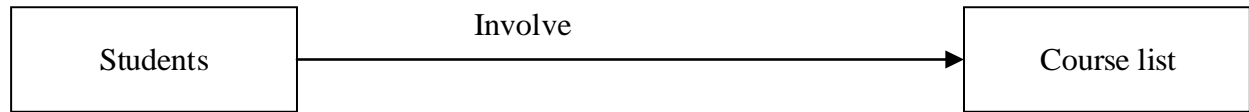
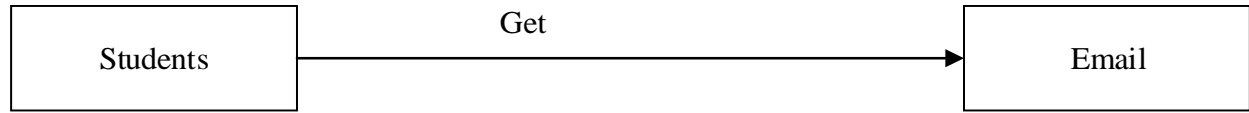
10. Email

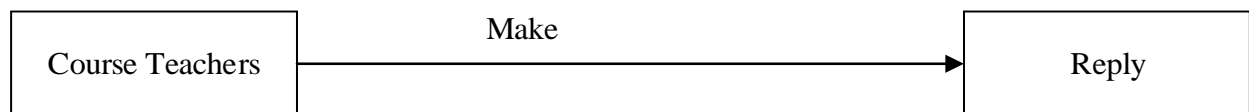
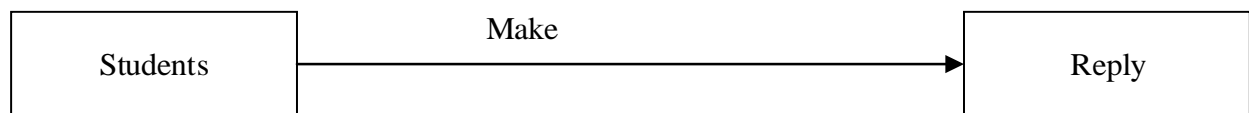
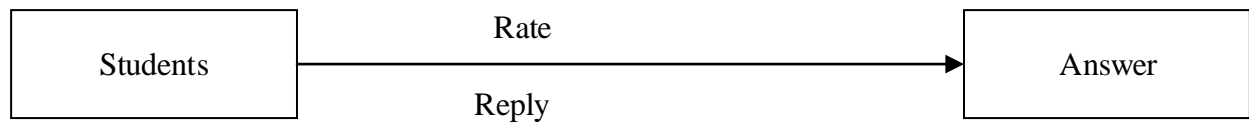
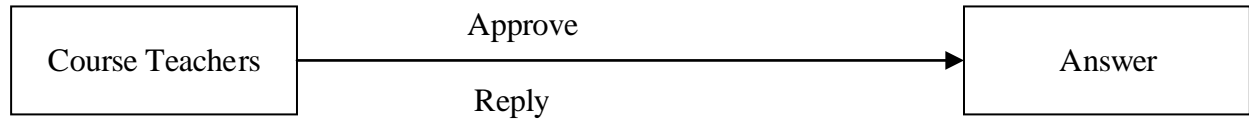
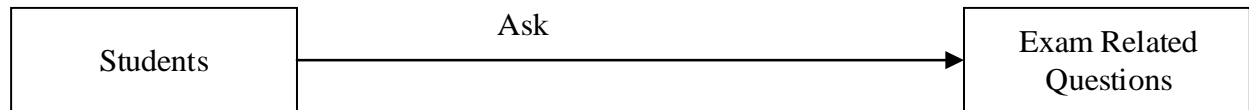
- i. em_id
- ii. content
- iii. source
- iv. destination

11. Reply

- i. r_id
- ii. comment

5.2.3 Pair Relation





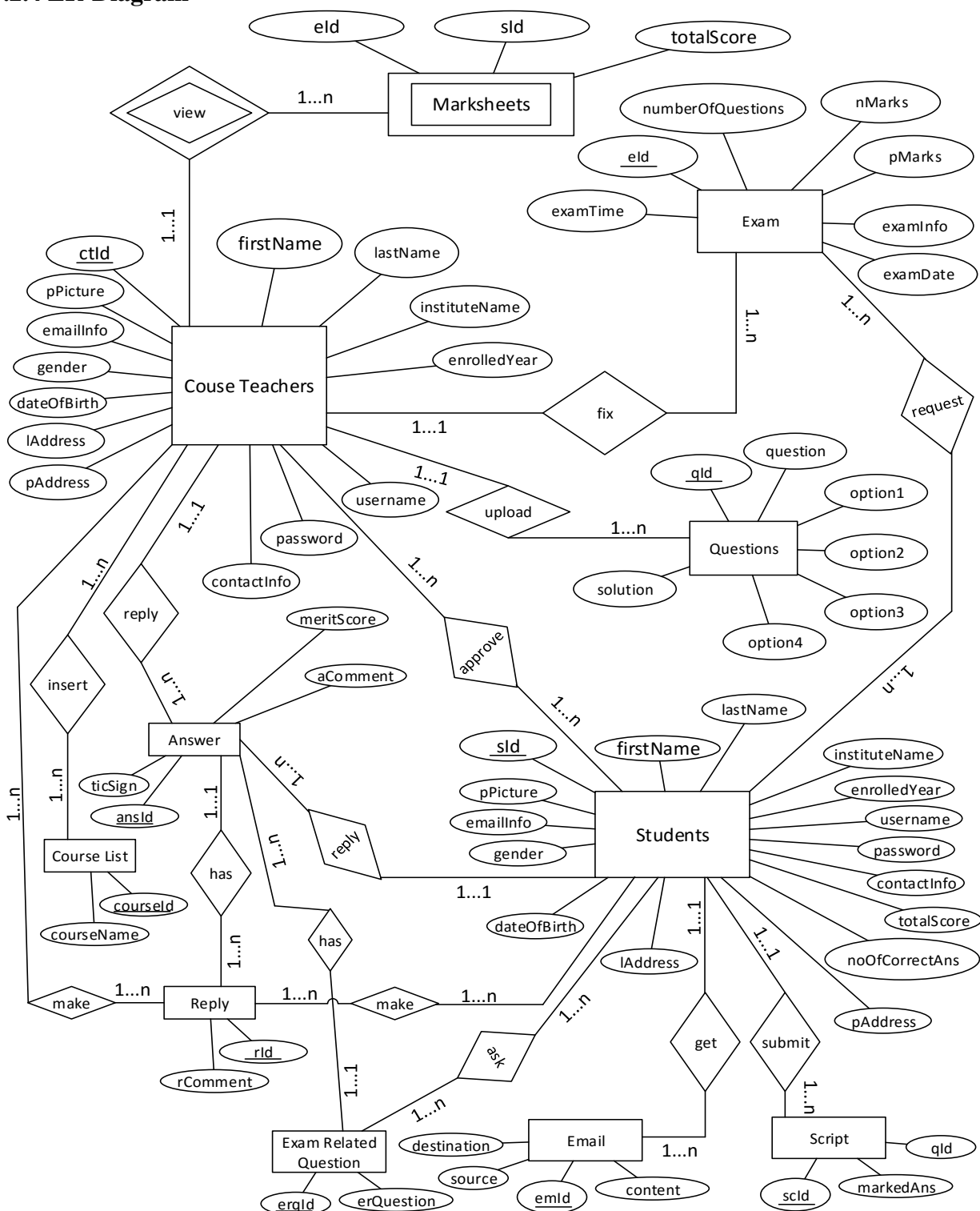


Figure 5.2.4: ER-diagram of the proposed system

5.2.5 Database Table

Students	
Attribute	Type (Size)
student_id	Varchar2(10)
first_name	Varchar2(20)
last_name	Varchar2(20)
email_info	Varchar2(25)
gender	Boolean
institute_name	Varchar2(20)
date_of_birth	Date
local_address	Varchar2(50)
permanent_address	Varchar2(50)
username	Varchar2(20)
password	Varchar2(10)
enrolled_year	Integer (4)
total_scores	Float(5)
number_of_correct_answer	Integer(3)
profile_picture	Image
answer	Varchar2(10)

Teachers	
Attribute	Type (Size)
course_teacher_id	Varchar2(10)
first_name	Varchar2(20)
last_name	Varchar2(20)
email_info	Varchar2(25)
gender	Boolean
institute_name	Varchar2(20)
date_of_birth	Date
local_address	Varchar2(50)
permanent_address	Varchar2(50)
username	Varchar2(20)
password	Varchar2(10)
enrolled_year	Integer (4)
profile_picture	Image

Mark sheets	
Attribute	Type (Size)
course_teacher_id exam_id total_scores student_id	Varchar2(10) Varchar2(10) Floats(5) Varchar2(10)

Course list	
Attribute	Type (Size)
course_id course_name	Varchar2(10) Varchar2(20)

Exam	
Attribute	Type (Size)
exam_id exam_date exam_info exam_time positive_marks negative_marks number_of_questions course_teacher_id	Varchar2(10) Date Varchar(1000) Time Float(3) Float(3) Integer(3) Varchar2(10)

Questions	
Attribute	Type (Size)
question_id	Varchar2(10)
question	Varchar2(100)
option1	Varchar2(100)
option2	Varchar2(100)
option3	Varchar2(100)
option4	Varchar2(100)
solution	Varchar2(100)
course_teacher_id	Varchar2(10)

Answers	
Attribute	Type (Size)
answer_id	Varchar2(10)
answer_comment	Varchar2(1000)
merit_score	Float(3)
tic_sign	Boolean
course_teacher_id	Varchar2(10)
student_id	Varchar2(10)
exam_related_question	

Email	
Attribute	Type (Size)
email_id	Varchar2(10)
content	Varchar2(1000)
source	Varchar2(20)
destination	Varchar2(20)
student_id	Varchar2(10)

Exam Related Questions	
Attribute	Type (Size)
exam_related_question_id exam_related_question	Varchar2(10) Varchar2(1000)

Script	
Attribute	Type (Size)
script_id question_id marked_answer student_id	Varchar2(10) Varchar2(10) Varchar2(100) Varchar2(10)

Insert	
Attribute	Type (Size)
course_teacher_id course_id	Varchar2(10) Varchar2(10)

Approve	
Attribute	Type (Size)
course_teacher_id student_id	Varchar2(10) Varchar2(10)

Ask	
Attribute	Type (Size)
exam_related_question	Varchar2(10)
student_id	Varchar2(10)

Request	
Attribute	Type (Size)
student_id	Varchar2(10)
email_id	Varchar2(10)

Reply	
Attribute	Type (Size)
reply_id	Varchar2(10)
reply_comment	Varchar2(1000)
answer_id	Varchar2(10)

Make by Teacher	
Attribute	Type (Size)
reply_id	Varchar2(10)
course_teacher_id	Varchar2(10)

Make by Teacher	
Attribute	Type (Size)
reply_id	Varchar2(10)
student_id	Varchar2(10)

Chapter 6

Class Based Model

In this chapter, we describe class based model of our proposed project “Online Registration and Exam System”.

6.1 Class Based Modeling Concept

Class-based modeling represents the objects that the system will manipulate, the operations that will applied to the objects, relationships between the objects and the collaborations that occur between the classes that are defined.

6.2 General Classification

Property Name	ID (G.C)
External entity	1
Things	2
Occurrences	3
Roles	4
Organizational units	5
Places	6
Structures	7

6.3 Selection Criteria

Property Name	ID (S.C)
Retained information	1
Needed service	2
Multiple attributes	3
Common attributes	4
Common operations	5
Essential requirements	6

6.4 Class Identification

NID	Noun (Potential Class)	Problem/solution space	General Classification	Special Criterial
1.	Online Registration System	P	---	---
2. *	Students	S	4, 5	1, 2, 3, 4, 5, 6
3.	Authenticated user	P	---	---
4.	Course menu	P	---	---
5.	Registration Period	P	---	---
6.	Internet	P	---	---
7. *	Course Teachers	S	4, 5	1, 2, 3, 4, 5, 6
8.	Online Exam System	P	---	---
9.	Web-based portal	P	---	---
10.	Educational network	P	---	---
11.	Exam	S	2,3	2,3,4,5,6
12.	Discussion forum	P	---	---
13.	Registration process	P	---	---
14.	Profile	P	---	---
15.	Email info	S	---	---
16.	First name	S	---	---
17.	Last name	S	---	---
18.	Gender	S	---	---
19.	Date of birth	S	---	---
20.	Local address	S	---	---
21.	Permanent address	S	---	---
22.	Institute name	S	---	---
23.	Enrolled year	S	---	---
24.	Username	S	---	---
25.	Password	S	---	---
26.	Contract info	S	---	---
27.	Profile info	S	---	---
28.	Course name	S	---	---
29.	Course id	S	---	---
30. *	Course list	S	2,3	2,3,4,5
31.	Exam date	S	---	---
32.	Exam information	S	---	---

33.	Websites notice board	P	---	---
34.	Information of the student	P	---	---
35.	Supreme authority	P	---	---
36. *	Email	S	2,3	2,3,4,5
37. *	Questions (MCQ)	S	2,3	2,3,4,5
38.	Solution	S	---	---
39.	Number of questions	S	---	---
40.	Positive marks	S	---	---
41.	Negative marks	S	---	---
42.	Right answer	P	---	---
43.	Wrong answer	P	---	---
44.	Exam time	S	---	---
45.	Schedule time	P	---	---
46. *	Script	S	2,3	2,3,4,5
47.	Period	P	---	---
48.	Submission	P	---	---
49.	Total scores	S	---	---
50.	Number of correct answer	S	---	---
51. *	Mark sheets	S	2,3	1,2,3,4,5
52. *	Exam related questions	S	2,3	1,2,3,4,5
53. *	Answer	S	2,3	1,2,3,4,5
54. *	Reply	S	2,3	1,2,3,4,5
55.	Merit	S	---	---
56.	Correct answer	P	---	---
57.	Tic sign	S	---	---

Star (*) sign indicates that there are selected for class. So the classes are

1. Students
2. Course Teachers
3. Course List
4. Email
5. Questions (MCQ)
6. Script
7. Mark Sheets
8. Exam Related Questions
9. Answer
10. Reply

6.5 Subject, Verb, Object/ Predicate Identification:

Subject	Verb	Object/Predicate
Course Teachers	submit	email, username, password, contract info, profile picture, first name, last name, gender, date of birth, present address, permanent address, enrolled year
Students	submit	email, username, password, contract info, profile picture, first name, last name, gender, date of birth, present address, permanent address, enrolled year
Course Teachers	insert	course name, course id in course list
Course Teachers	fix	the exam date and exam information
Course Teachers	post	it on websites noticeboard
Course Teachers	approve	students
Students	get	an email
Students	verify	their email
Course Teachers	upload	questions (M.C.Q) and solutions in the database
Course Teachers	set	The number of questions, positive, negative marks, exam time
Course Teachers	generate	Questions for students
Exam	begin	at schedule time
Exam	end	in due time
Students	submit	their script
Students	show	correct answer
Course Teachers	view	mark sheets of students
Student	ask	exam related question
Course Teachers	answer	it
Students	answer	it
Course Teachers	make	reply against answer
Students	make	reply against answer
Students	rate	answer
Course Teacher	approve	answer with tic sign

6.6 Class Card

1. Students	
Attributes	Methods
<ol style="list-style-type: none"> 1. student_id 2. email info 3. first name 4. last name 5. gender 6. date of birth 7. local address 8. permanent address 9. institute name 10. enrolled year 11. username 12. password 13. contract info 14. profile picture 15. total scores 16. number of correct answer 	<ol style="list-style-type: none"> 1. submit_info() 2. submit_script() 3. verify() 4. show() 5. ask() 6. answer() 7. make_reply() 8. rate()
Responsibilities	Collaborator Class
<ol style="list-style-type: none"> 1. Submission of profile info 2. Submission of script 3. Getting email 4. Verifying email 5. Showing number of correct answer and score 6. Answering exam related questions 7. Answering it 8. Making reply 9. Rating answer 	<ol style="list-style-type: none"> 1. Database 2. Script 3. Email 4. Email 5. Database 6. Exam related question 7. Answer 8. Reply 9. Answer

2. Course Teachers	
Attributes	Methods
<ol style="list-style-type: none"> 1. course_teacher_id 2. email info 3. first name 4. last name 5. gender 6. date of birth 7. local address 8. permanent address 9. institute name 10. enrolled year 11. username 12. password 13. contract info 14. profile picture 	<ol style="list-style-type: none"> 1. submit_info() 2. insert() 3. fix() 4. post() 5. approve_student() 6. approve_answer() 7. upload() 8. set() 9. generate() 10. view() 11. answer() 12. make_reply()
Responsibilities	Collaborator Class
<ol style="list-style-type: none"> 1. Submission of profile info 2. Insertion of the course info 3. Fixing exam info 4. Posting on website 5. Approving student 6. Uploading question 7. Setting exam properties 8. Generation of questions 9. Viewing the mark sheets 10. Answering exam related questions 11. Making reply 12. Approving answer 	<ol style="list-style-type: none"> 1. Database 2. Course list 3. Exam 4. ---- 5. Students 6. Questions 7. Exam 8. Questions 9. Mark sheets 10. Answer 11. Reply 12. Answer

3. Exam	
Attributes	Methods
1. exam_id 2. exam_date 3. exam_info 4. exam_time 5. positive_marks 6. negative_marks 7. number_of_questions	1. begin() 2. end()
Responsibilities	Collaborator Class
1. Beginning the exam 2. Ending the exam	1. Database 2. Database

4. Course List	
Attributes	Methods
1. course_id 2. course_name	1. getter/setter
Responsibilities	Collaborator Class
-----	-----

5. Questions	
Attributes	Methods
2. question_id 3. question 4. option1 5. option2 6. option3 7. option4 8. solution	1. getter/setter
Responsibilities	Collaborator Class
-----	-----

6. Script	
Attributes	Methods
1. script_id 2. question_id 3. marked_answer	1. getter/setter
Responsibilities	Collaborator Class
-----	-----

7. Mark Sheets	
Attributes	Methods
1. exam_id 2. student_id 3. total_score	1. getter/setter
Responsibilities	Collaborator Class
-----	-----

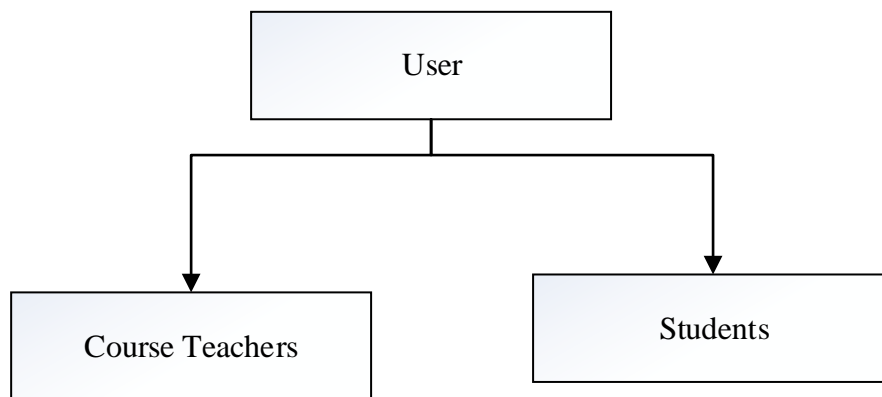
8. Exam Related Question	
Attributes	Methods
1. exam_related_question_id 2. exam_question	1. getter/setter
Responsibilities	Collaborator Class
-----	-----

9. Answer	
Attributes	Methods
1. answer_comment 2. merit_rating 3. tic_sign 4. answer_id	1. getter/setter
Responsibilities	Collaborator Class
-----	-----

10. Email	
Attributes	Methods
1. email_id 2. content 3. source 4. destination	1. getter/setter
Responsibilities	Collaborator Class
-----	-----

11. Reply	
Attributes	Methods
1. reply_id 2. reply_comment	1. getter/setter
Responsibilities	Collaborator Class
-----	-----

12. Database	
Attributes	Methods
	1. create() 2. read() 3.update() 4. delete()
Responsibilities	Collaborator Class
-----	-----



Many attributes and methods of Course Teachers and Students are same, so they can be merged into User class.

6.6 CRC Diagram

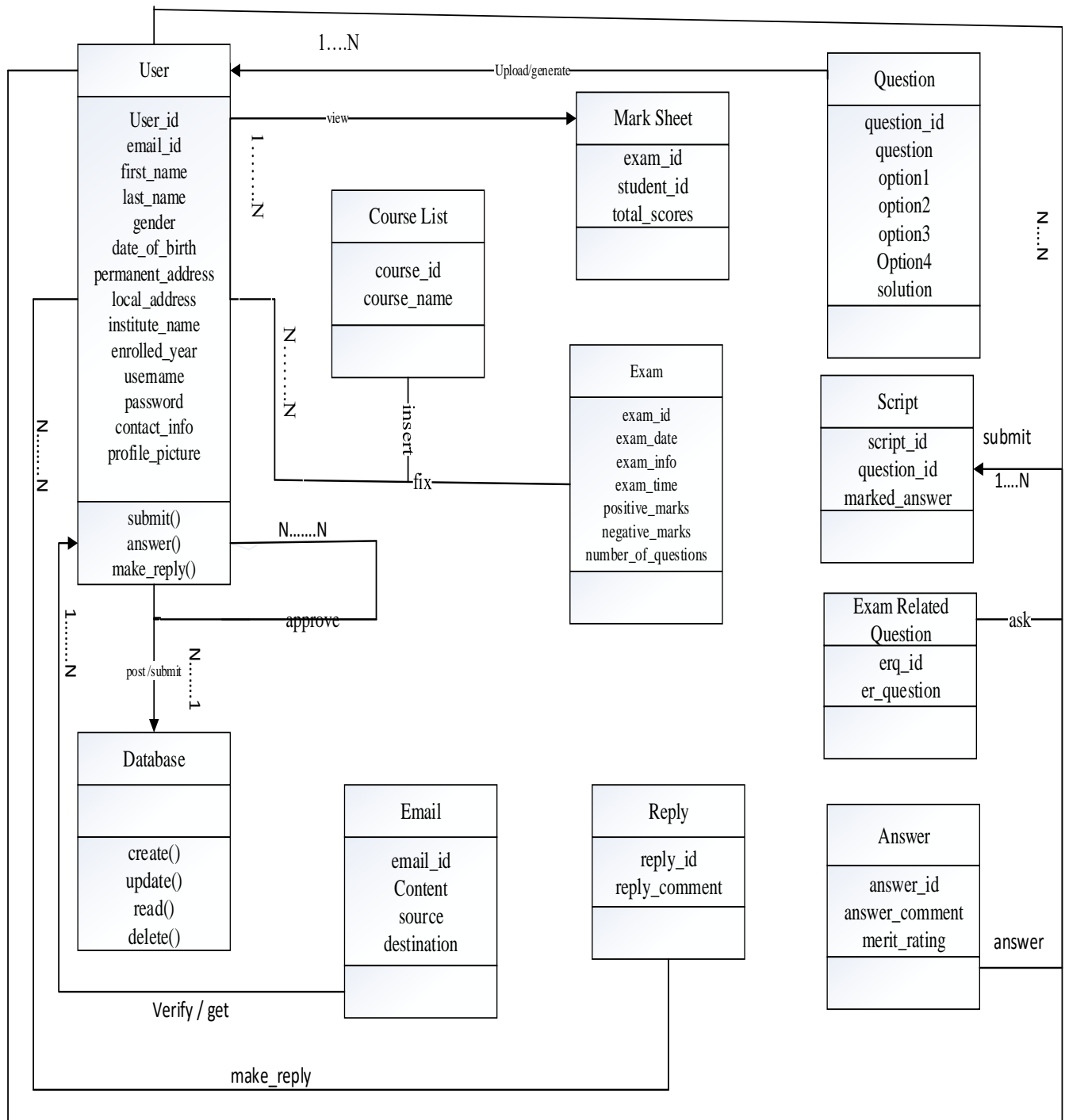


Figure 6.6: CRC diagram of our proposed system

Chapter 7

Flow Oriented Model

This chapter is intended to describe “Flow Oriented Model” of our proposed system.

7.1 Introduction

low models focus on the flow of data objects as they are transformed by processing functions. Derived from structured analysis, flow models use the data flow diagram, a modeling notation that depicts how input is transformed into output as data objects move through the system. Each software function that transforms data is described by a process specification or narrative. In addition to data flow, this modeling element also depicts control flow.

Data flow oriented modeling is the most widely used analysis notation. Flow oriented modeling focuses on structured analysis and design, follows a top to down methodology and uses a graphical technique depicting information flows and the transformations that are applied as data moves from input to output.

7.2 Data Flow Diagram (DFD)

A data flow diagram (DFD) is a significant modeling technique for analyzing and constructing information processes. DFD literally means an illustration that explains the course or movement of information in a process. DFD illustrates this flow of information in a process based on the inputs and outputs. A DFD can be referred to as a Process Model.

7.2.1 Level-0 of DFD

By analyzing the requirements and user scenario, we have found three inputs and two output. Theses inputs are

- I. Students information
- II. Course teacher information
- III. Questions and solutions

And outputs are

- I. Scores
- II. Database

Level-0 of the DFD diagram indicates these inputs and outputs interacted with the system.

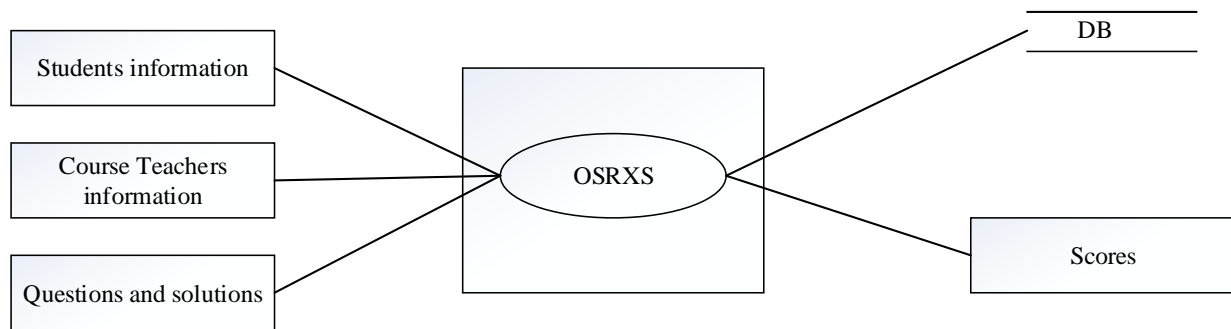


Figure 7.2.1: Level-0 of the DFD

7.2.2 Level-1 of DFD

Here, level-1 of the DFD shows how these three inputs flow and store in database or produce output of the system.

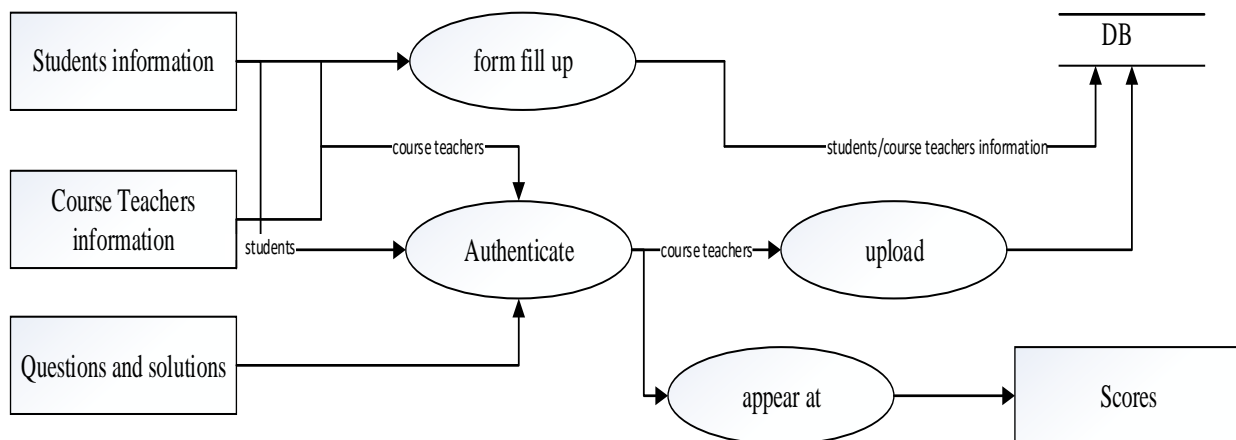


Figure 7.2.2: Level-1 of DFD

Chapter 8

Behavioral Model

The behavioral model indicates how software will respond to external events.

8.1 State Transition Diagram

State diagram represents active states for each class the events (triggers). For performing this task, we have identified all the events, their initiators and collaborators.

Event List:

1. Begins with registration process (same as- 2)
2. Submit first name, last name, gender, date of birth, local address, permanent address, institute name, enrolled year, username, password, contact info, email, profile picture
3. Insert course name and course id in the course list
4. Fix exam date and information
5. Post on the websites notice board
6. Make request for appearing at the exam
7. Approve or reject student
8. Get an email
9. Verify the email
10. Upload questions (MCQ) and solutions
11. Set number of questions, exam time, positive and negative marks
12. Generate questions randomly for each approved student
13. Begin at schedule time
14. End in due time
15. Submit their script at any time before the exam ends
16. Show correct answers and obtained score
17. View mark sheets
18. Ask any exam related questions
19. Answer this questions
20. Reply against each answer or comment
21. Rate each answer according to its merit
22. Approve the correct answers by marking it with a tic sign

Identifying the initiator and collaborator:

No.	Initiator	Collaborator
1.	Student, Course Teacher	
2.	Student, Course Teacher	
3.	Course Teacher	Course List
4.	Course Teacher	Exam
5.	Course Teacher	
6.	Student	Exam
7.	Course Teacher	Student
8.	Student	Email
9.	Student	Email
10.	Course Teacher	Question
11.	Course Teacher	Exam
12.	Course Teacher	Question
13.	Exam	
14.	Exam	
15.	Student	Script
16.	Student	Exam
17.	Course Teacher	Mark Sheet
18.	Student	Exam Related Question
19.	Student, Course Teacher	Answer
20.	Student, Course Teacher	Reply
21.	Student	Answer
22.	Course Teacher	Answer

State Transition Diagrams (STD):

Course Teacher:

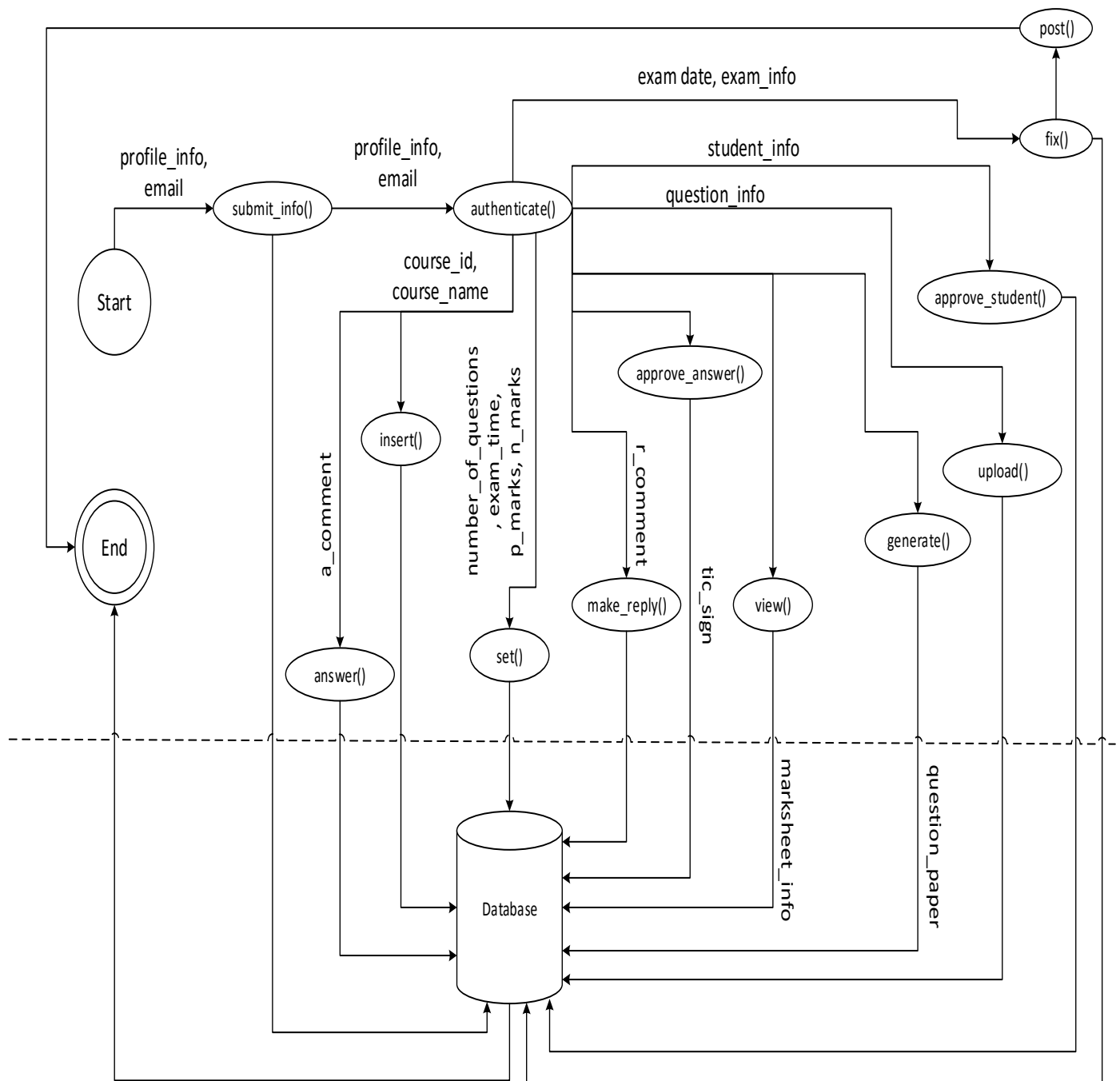


Figure 8.1.1: State Transition Diagram for “Course Teacher” class

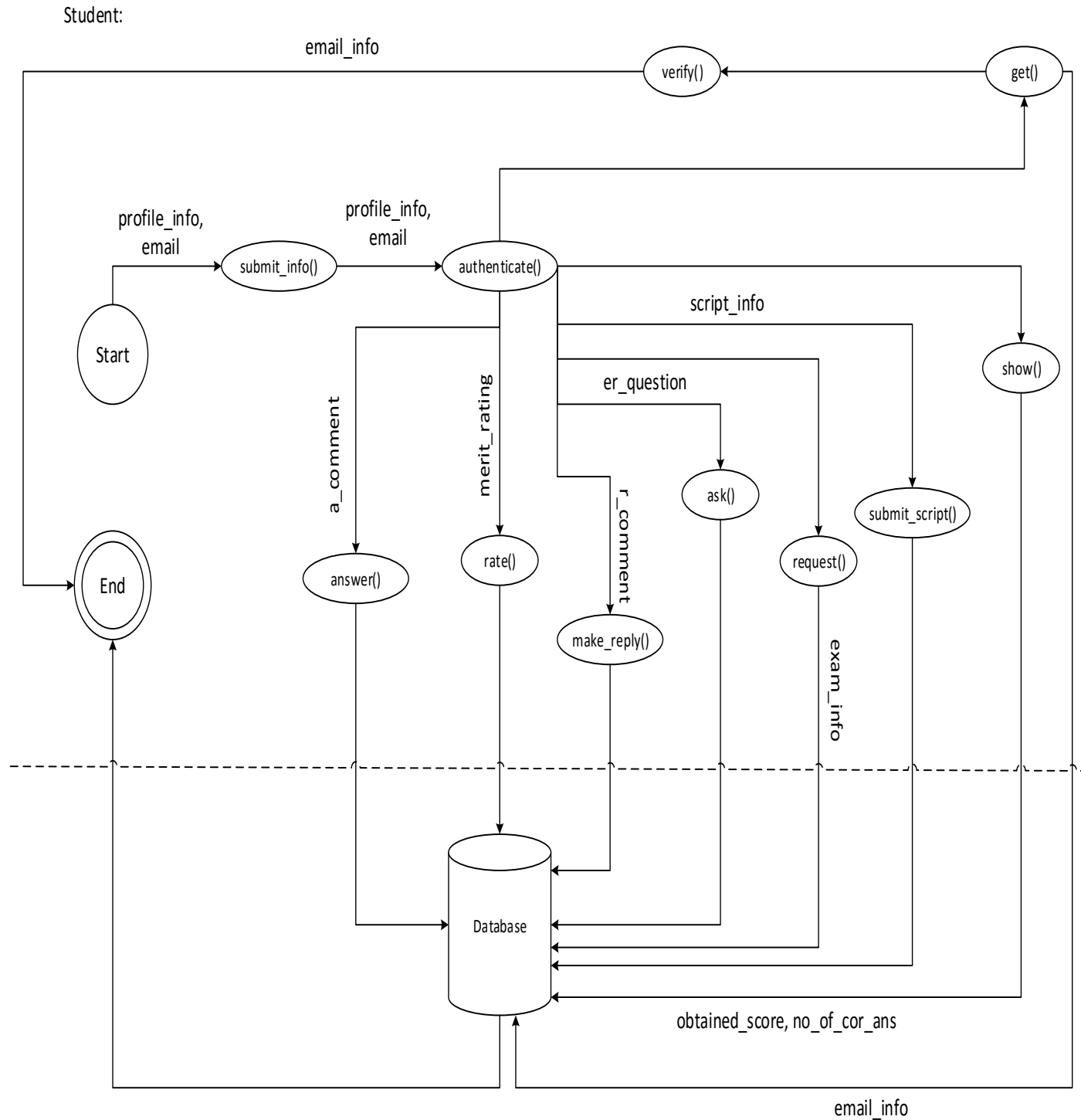
Class - Student:

Figure 8.1.2: State Transition Diagram for "Student" class

Class – Exam:

Exam:

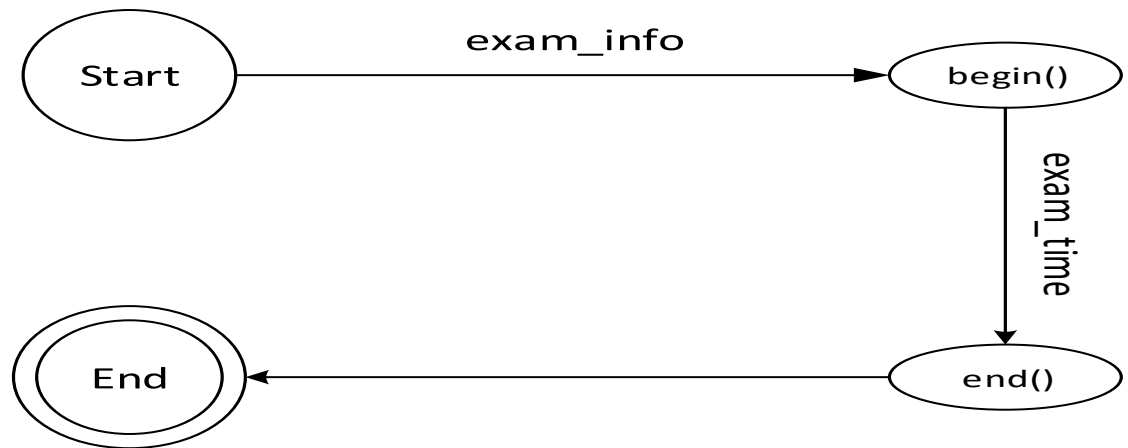
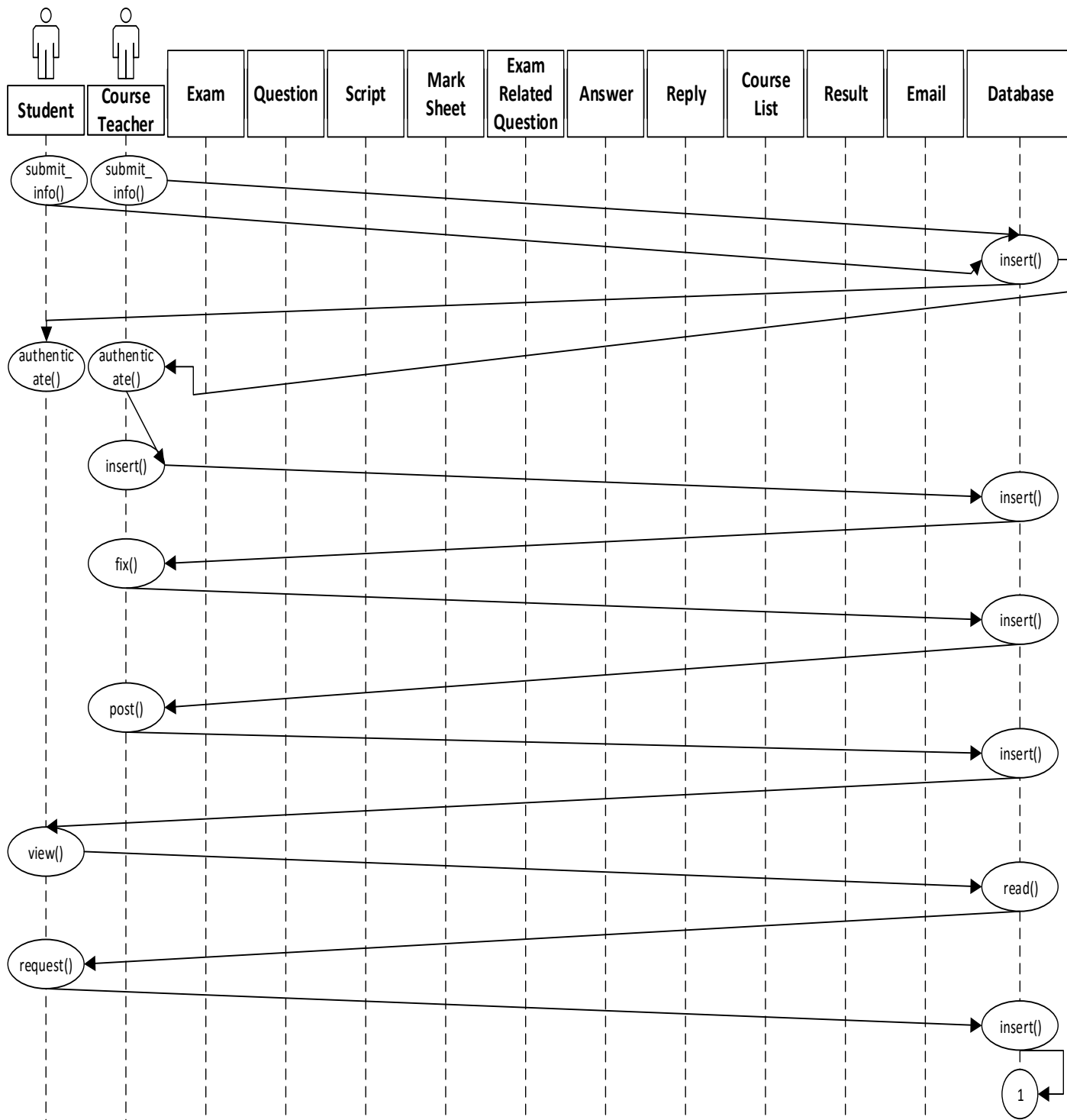


Figure 8.1.3: State Transition Diagram for “Exam” class

8.2 Sequence Diagram

Sequence diagram indicates how events cause transitions from object to object.



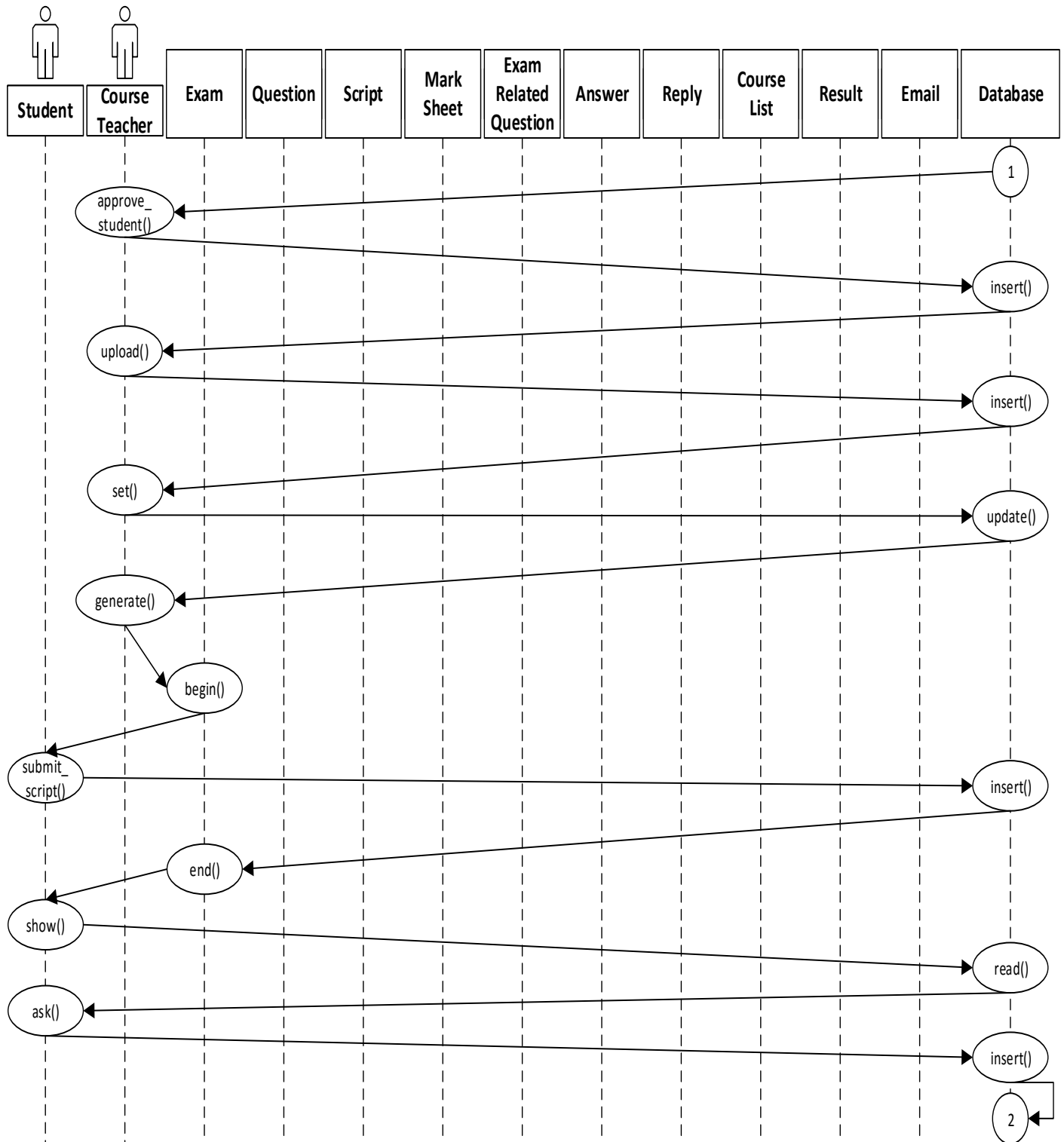
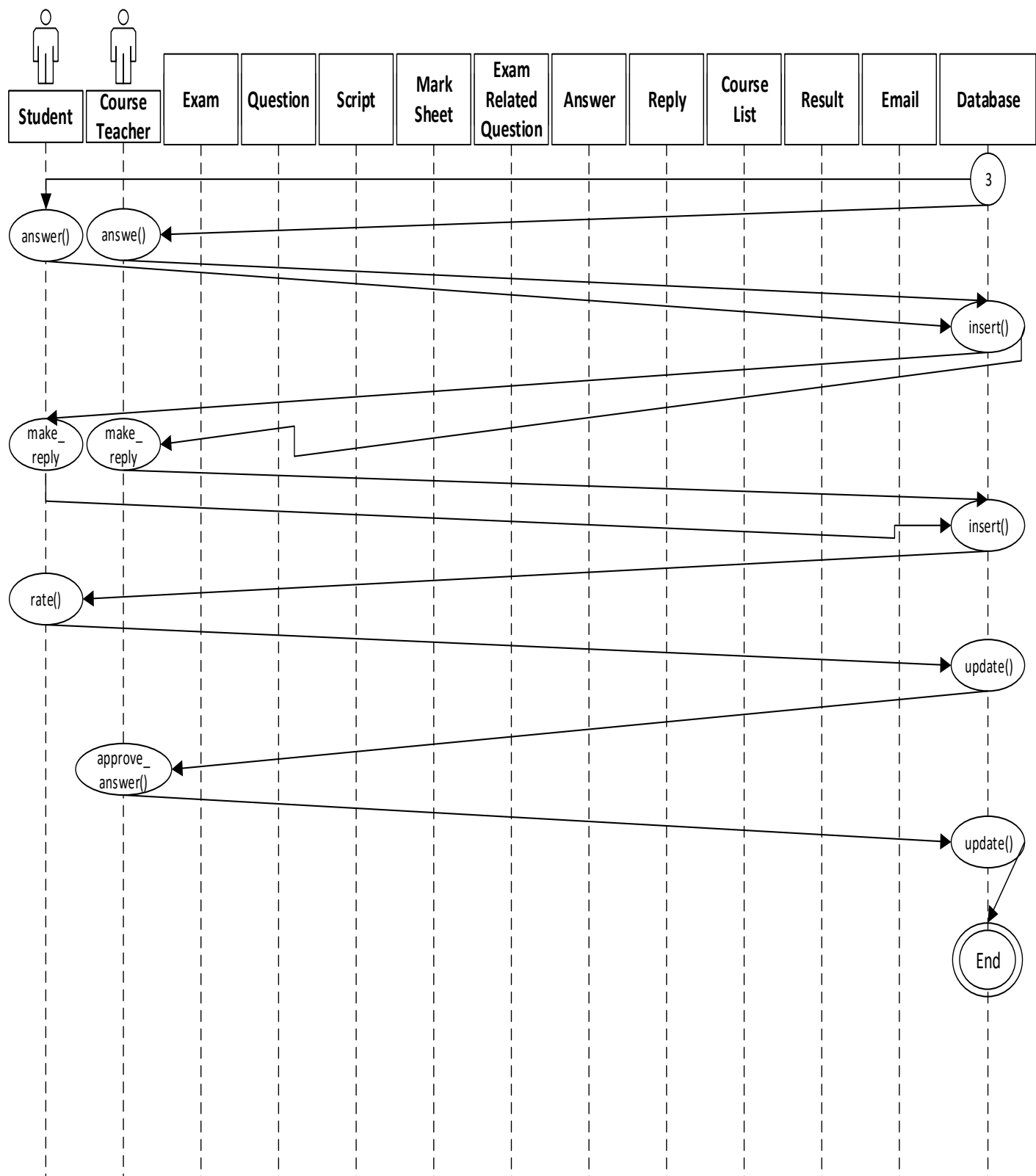


Figure 8.2.1: Sequence Diagram of our proposed system



Chapter 9

Screenshot of the Project

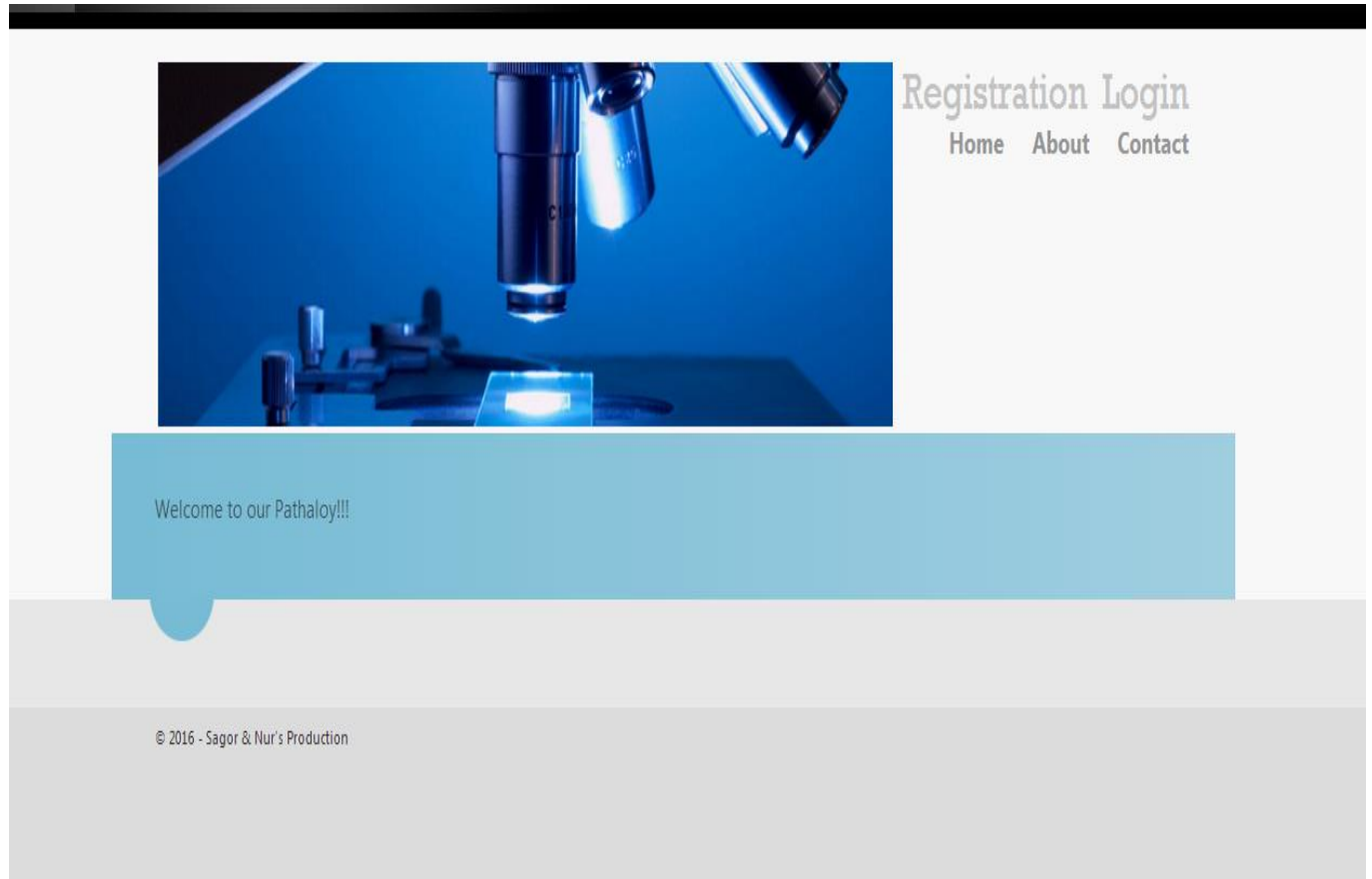


Figure 9.1.1: Screenshot of the Home Page

Select One Option

Registration As Student



Registration As Teacher



Figure 9.1.2: Screenshot of the Registration Page

Please Fill Up Form

first_name

last_name

email_info

gender

Male ▼

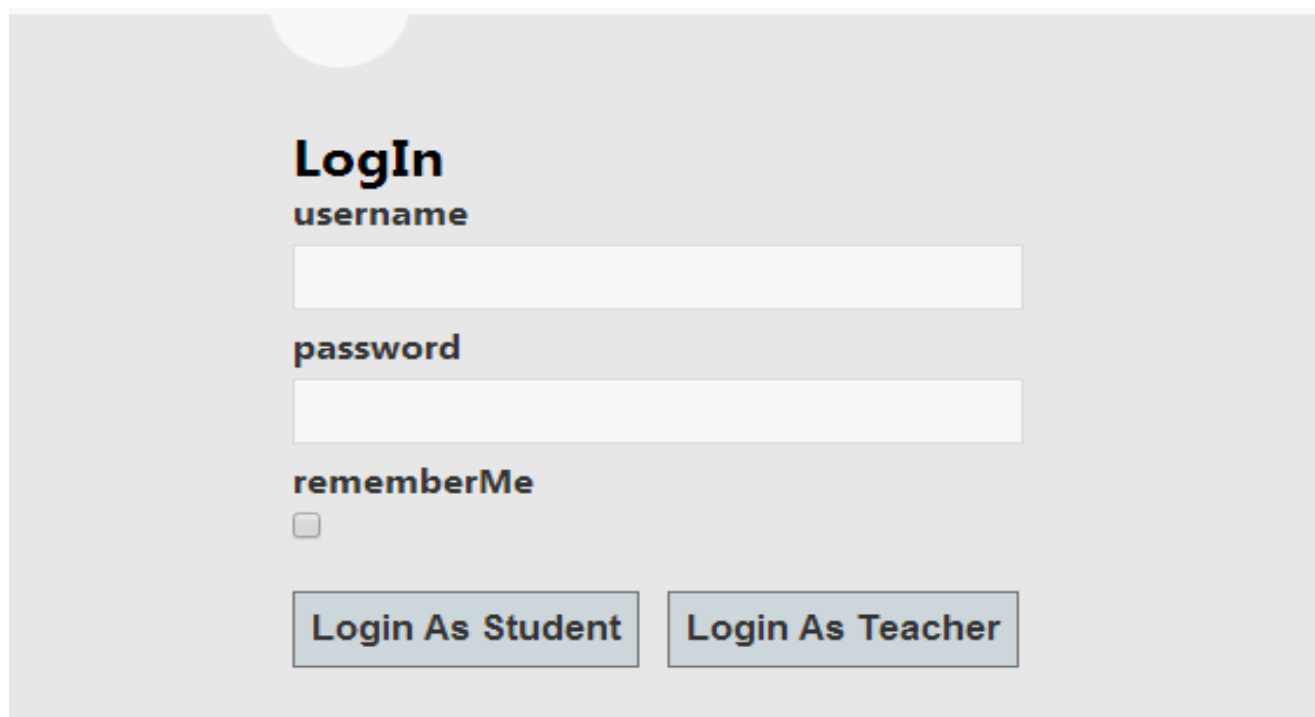
institute_name

Enter your date of birth :

local_address

permanent_address

Figure 9.1.3: Screenshot of the Registration as User Page



The screenshot shows a login interface on a light gray background. At the top, the word "Login" is displayed in a large, bold, black font. Below it, the label "username" is followed by a white rectangular input field. Underneath the username field, the label "password" is followed by another white rectangular input field. Below the password field, the label "rememberMe" is followed by a small, empty square checkbox. At the bottom of the form, there are two rectangular buttons with a light blue gradient and black borders. The left button is labeled "Login As Student" and the right button is labeled "Login As Teacher".

Figure 9.1.4: Screenshot of the Login Page

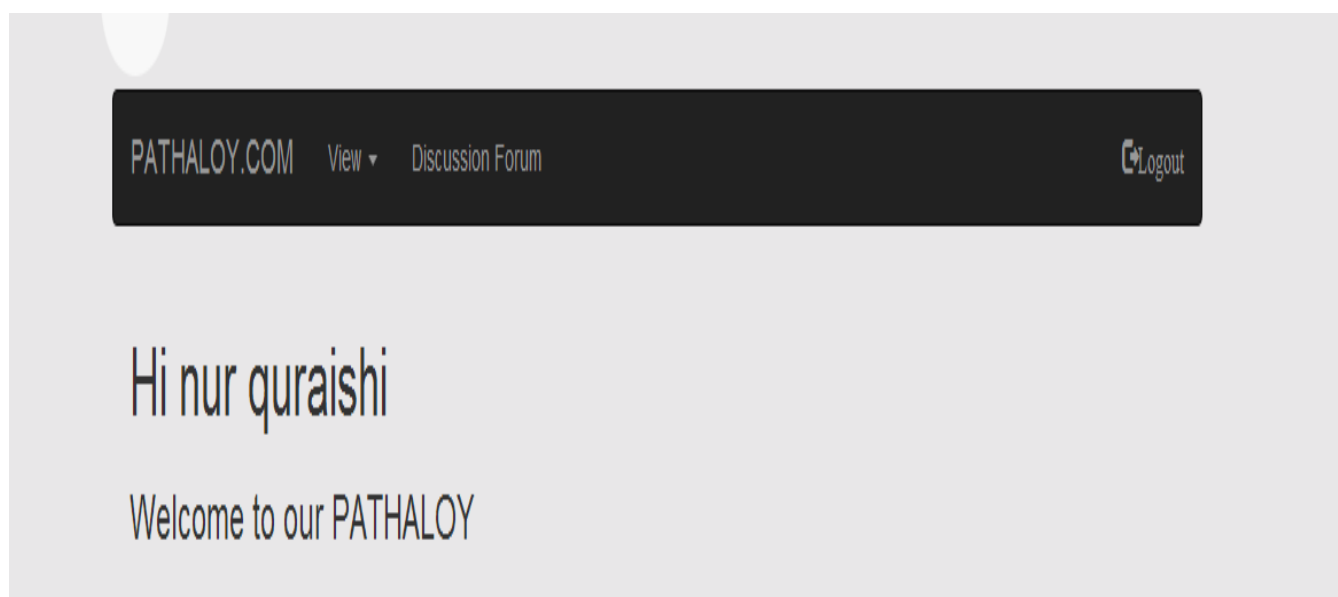


Figure 9.1.5: Screenshot of the Window for Student Page

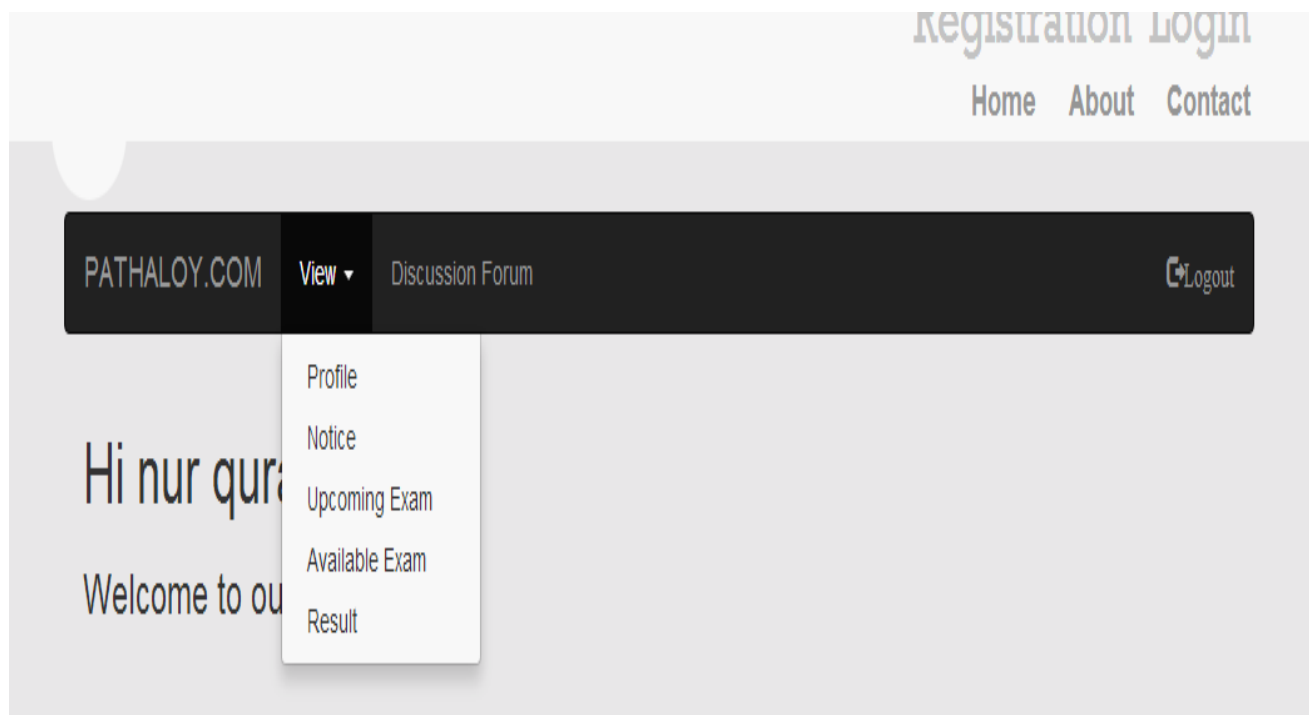


Figure 9.1.6: Screenshot of the Features of Window for Student Page

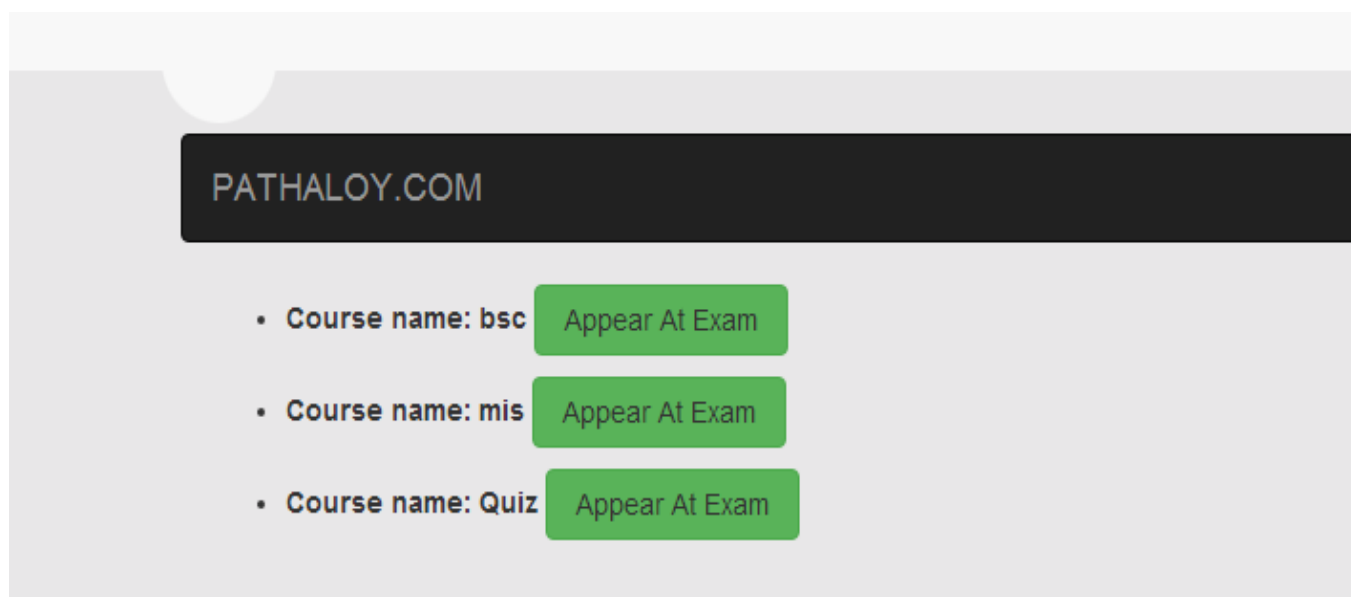


Figure 9.1.7: Screenshot of the Upcoming Exam Page

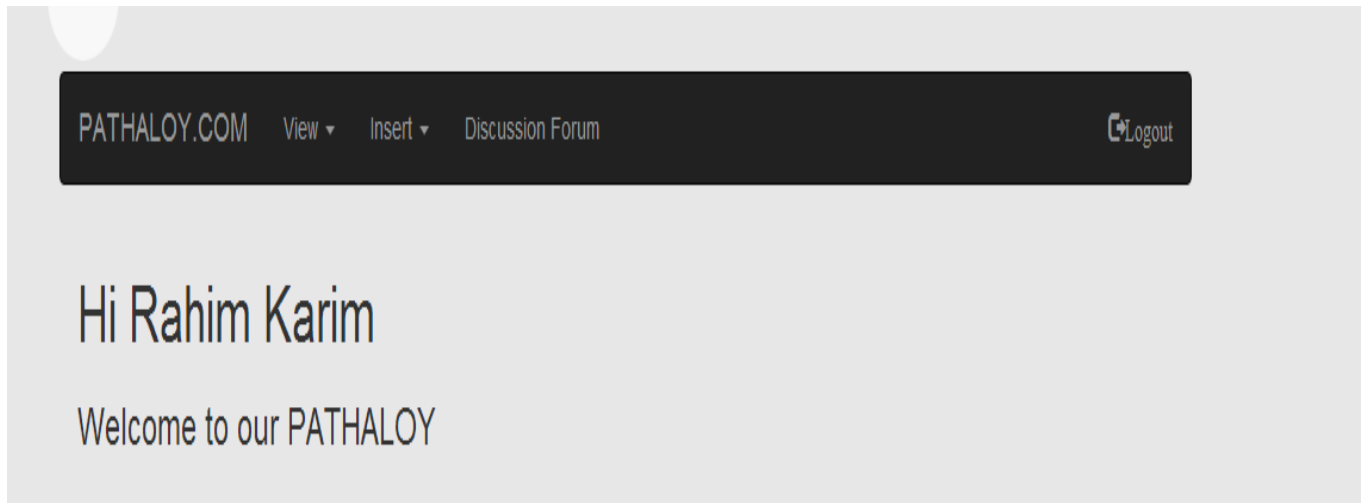


Figure 9.1.8: Screenshot of the Window for Course Teacher Page

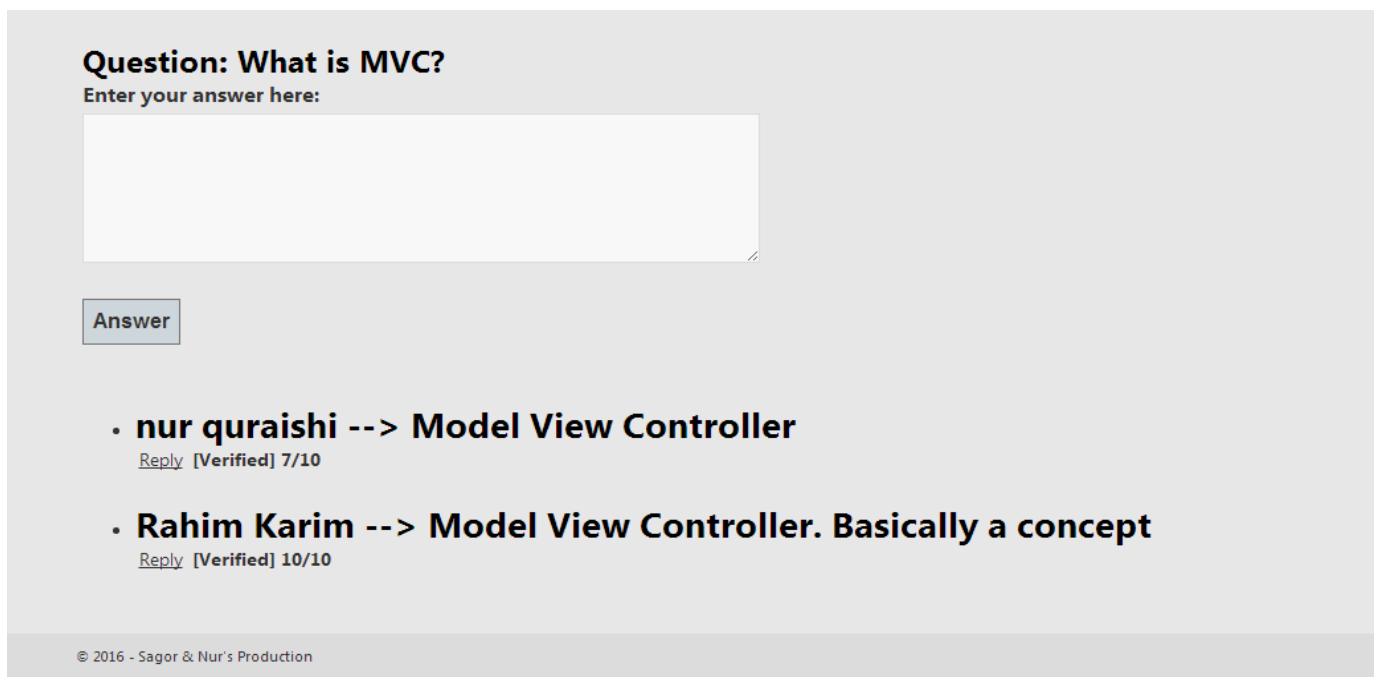


Figure 9.1.9: Screenshot of the Discussion Forum Page with related answers

Answer: Model View Controller. Basically a concept

Enter your reply here:

Reply

- **abcdsdafwsdse sadszfasFDFas --> Thank you, Sir :)**
- **Rahim Karim --> You are welcomed**

Figure 9.1.10: Screenshot of the Discussion Forum Page with replies

Chapter 10

References

- ❖ <http://mogadalai.wordpress.com/2007/05/28/howto-write-a-project-proposal>
(Accessed on 07/02/2016)
- ❖ http://www.w3schools.com/aspnet/mvc_intro.asp
(Accessed on 01/05/2016)
- ❖ <http://www.w3schools.com/bootstrap/default.asp>
(Accessed on 01/05/2016)
- ❖ <https://msdn.microsoft.com/en-us/library/gg416515%28v=vs.108%29.aspx>
(Accessed on 01/05/2016)
- ❖ <http://www.asp.net/mvc>
(Accessed on 01/05/2016)
- ❖ <http://stackoverflow.com/questions/5052752/adding-your-own-htmlhelper-in-asp-net-mvc-3>
(Accessed on 01/05/2016)

Chapter 11

Conclusion

We are really very glad to submit the final SRS report on our proposed “Online Student Registration and Exam System” project. From this, the readers will get a clear and easy view of “Online Student Registration and Exam System”. This SRS document can be used effectively to maintain software development cycle. It will be very easy to conduct the whole project using this SRS. We tried our best to remove all dependencies and make effective and fully designed SRS. We believe that reader will find it in order.