الجامعة الاسلامية للتكنولوجيا
UNIVERSITÉ ISLAMIQUE DE TECHNOLOGIE
ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)
ORGANISATION OF ISLAMIC COOPERATION (OIC)

Online Coaching System

Software Requirements and Specifications Lab Report

Prepared for **Md. Jubair Ibna Mostafa**

Created by

Ayesha Afroza Mohsin 200042016

Nawsheen Mehreen 200042134

Abrar Mahmud Rahim 200042168

Table of Contents

1. Introduction

In recent years, the education industry has seen a significant shift towards online learning and coaching. With the widespread availability of the internet and the advancement of technology, coaching centers have recognized the benefits of offering their services online. The convenience, flexibility, and affordability of online coaching have made it an increasingly popular choice among students, and it is expected that the demand for such services will continue to grow in the future.

The Online Coaching System is a unique and integrated platform that provides a one-stop solution for students seeking coaching services. Unlike other online coaching platforms, our system offers a comprehensive range of features and tools, all within a single platform. Our system provides everything a student needs to succeed in their educational journey, including live interactive classes, video lectures, study materials and practice exams. With our integrated system, students can save time and effort by accessing all the necessary resources in one place, ultimately leading to improved learning outcomes.

2. Purpose

This document provides a brief overview of the requirements for an Online Coaching System, which includes the functions, performance, and support needs of the system. By establishing these requirements, we create a foundation for the development of the system. The purpose of this document is to serve as an official way of communicating what the users and stakeholders require from the developers. As users and developers work together to create the system, this document will be updated and expanded to reflect any changes or clarifications needed.

3. Intended Audience

This report on Software Requirement Specifications (SRS) is aimed at a variety of audiences. After discussing with all possible stakeholders, we have developed this SRS to provide clear guidelines and documentation for the following purposes:

- Users and administrators can use this SRS to ensure that the product created by the development team meets customer requirements.
- Project managers on the development team can use this SRS to plan milestones and delivery dates, and ensure that the team stays on track during development.
- Designers can use this SRS as a basis for designing the system, constantly referring back to it to ensure it meets customer needs.
- Developers can use this SRS as a basis for building system functionality, linking the requirements defined in the SRS to the software they create.
- Testers can use this SRS to create test plans and cases for each requirement documented in the SRS. Once portions of the software are complete, testers will run tests on it to ensure

that it meets the requirements in the SRS.

- Finally, testers will test the entire system when it is complete, ensuring that all requirements in the SRS have been met.

4. Conclusion

The analysis process was conducted systematically, which allowed us to identify the stakeholders and requirements in a clear and organized manner. Our focus was on understanding the users who will be utilizing the software. The resulting document will be useful to all stakeholders involved in the software development process, allowing them to better understand and ensure the software's ease of use. The developers and testers will have a clearer idea of what needs to be done, which will result in smoother development. Furthermore, ongoing communication between stakeholders and developers will help evolve and further enhance the software.

5. Inception

5.1. Establishing a Basic Understanding of the Problem

The system described is an online coaching platform with multiple user types: system admin, moderators, students, teachers, parents, and payment partners. The admin will be the only user with the ability to create accounts, and pre-generated admin credentials will be provided at system deployment. The admin can add and remove other users, view activity logs, generate reports, and perform all user activities. Students and parents can sign up and log in to access personal information, classes, exams, resources, and schedules. Moderators can do almost everything an admin can, but cannot delete the admin. Prospective teachers apply to become a teacher and will be hired by the admin or moderators after being reviewed. Teachers can host video conferences, upload study material, check copies, and update student marks, depending on their assigned role.

5.2. Identifying the Clients of the Solution

The clients are the people who have hired us to create this system for their Coaching Center. In order to let them bring their coaching center services online.

5.3. Identifying the Stakeholders of the Solution

To Identify stakeholders, the steps we took were

- First, communicate with the administrator.
- Who else do you think I should talk to? Teacher
- Who will use the solution? Admin, Student, Teacher, Parents

Further Inception aspects of the Project

- Business value
    - What will be the economic benefit of a successful solution?
        - No more third party softwares required for the users. Example: Zoom, test portal etc.
        - Since there is a parent supervision option, parents will be more inclined to admit their children in our coaching
        - All in one solution attracting students to get the smooth coaching experience at home saving them the cost and energy for commuting.
- Existing solution
    - Is there another source for the solution that you need?

        - In our country, currently there are very few websites like Udvash for coaching but they use a variety of paid websites to manage the coaching system, for example they take classes using zoom, sending SMS to parents separately after every quiz etc.

(if any, understand that solution to better understand your problem and features)

- Define the problem (from a broader sense)
    - What problem are you solving in your SPL-2 / or the project you have chosen?
        - Avoiding use of multiple softwares such as Google classroom for resource and information sharing, zoom for video conferencing, paid test portals for taking exams and quizzes and sending messages manually to the parents notifying them of the current status of the students.
    - Why did you choose this project?
        - Currently, we are trying to run a small coaching online but we are facing these challenges of third party websites and it is very hard to keep track of everything at once.
    - Is there any scenario or relationship of your project to the real world?
        - Online coachings are a hassle to many people due to lack of ease of usage of the existing softwares even though many people do not want to leave the comfort of their homes for doing offline coachings. Our website will make the process more efficient for the use of these people and many more.
- Feasibility
    - Is it possible to solve the problem? Yes

5.4. The Nature of the Solution that is Desired

The nature of the solution as we understood it thus far is an all in one platform that brings together all the aspects of a coaching center online in a web-based platform that provides a comprehensive and user-friendly learning experience for students. The website should offer a range of features and functionalities, such as live video classes, recorded lectures, learning based exams, progress tracking, and parental control modes, that enable students to learn at their own pace under their parent’s supervision. The website should also have a secure and reliable infrastructure, with robust content management and administrative capabilities, that ensures smooth and efficient operation and maintenance. The solution should be scalable, flexible, and customizable, allowing the client to adapt and modify the website to meet changing needs and preferences. Overall, the desired solution is a high-quality, cost-effective, and innovative online coaching system that meets the needs and expectations of the client and their customers.

5.5. Establishing Preliminary Communication

We have designed our solution and contacted the appropriate clients in order to further discuss if our imagined solution coincides with their perception of the solution and our preliminary views seem similar. The current solution was feasible and had the green signal from the client team so we will soon start its implementation. We have also informed the client and they understand that their input is necessary throughout the process for the implementation of a useful solution.

6. Elicitation

The process of requirements elicitation involves a combination of problem-solving, elaboration, negotiation, and specification. It aims to promote a team-oriented approach to identify the problem, suggest potential solutions, negotiate alternative approaches, and establish a preliminary set of solution requirements. To encourage collaboration among stakeholders, we completed various tasks during the elicitation phase, including

- **Quality Function Deployment**
- **Usage Scenarios Development.**

6.1. Quality Function Deployment

Quality Function Deployment (QFD) is a method that translates customer requirements into specific engineering characteristics and determines how well the final product or service meets those requirements. In the context of Online Coaching System, QFD can help identify the requirements of the system and prioritize them based on their importance to different stakeholders.

To apply QFD to this scenario, the requirements can be divided into three categories: normal, expected, and exciting. Normal requirements are essential for the system to function properly and meet the basic needs of the stakeholders. Expected requirements go beyond the basic functionality and provide additional features or benefits that stakeholders would expect from the system. Exciting requirements are innovative or unique features that could differentiate the system from other similar solutions and create a positive impression among stakeholders.

6.1.1. Normal requirements:

- The system should provide a secure login mechanism for all users (admin, moderators, students, teachers, parents and payment partner).
- The system should allow the admin to add, modify, and delete user accounts and assign appropriate roles to them.
- The system should have a centralized dashboard for the admin to view various activities of the coaching, including class and test schedules, attendance records, rankings, and notifications sent out to students and parents.
- The system should allow the admin to generate reports of coaching activities.
- The system should allow students to sign up by providing their personal information, academic results, and hsc registration number.

- The system should allow parents to sign up by providing their personal information and their child's/children's username(s) as reference.
- The system should allow prospective teachers to apply for a teacher's role by providing their personal information, valid credentials, and preferred role.
- The system should allow teachers to host video conferences, upload study materials and resources, update student marks, and manually give attendance to a student.

6.1.2. Expected Requirements:

- The system should have an approval mechanism for student and teacher accounts after payment and verification by the admin/moderator.
- The system should allow the admin to remove any user accounts if needed.
- The system should allow moderators to perform almost everything the admin can except deleting the admin.
- The system should allow admins and moderators to edit class schedules, attendance records, marksheets, and ranking lists of students.
- The system should have an application form for prospective teachers to sign up as a teacher and approval mechanism for their application.

6.1.3. Exciting Requirements:

- The system could use AI/ML algorithms to generate reports and analytics on coaching activities.
- The system could have a feature for students to ask questions and get answers from teachers in real-time during class
- The system could provide personalized recommendations for students based on their previous academic results and class activities.

6.2. Usage Scenario :

The system will be used for conducting online coaching. The end users will be the system admin, moderators, students, teachers, parents and the payment partner.

There won’t be an option to create an ‘admin’ account by an existing user as there will be only one system administrator. At the time of deployment of the system, the pre-generated admin credentials i.e. their username and password will be provided to the administrator.
System admin can add information such as the name, email, address, mobile number, username, password and role about the moderators, teachers, students and parents and also assign them their respective roles. There will be an option for teachers to apply for certain positions and the admin can approve these requests.The admin will also be able to remove any users if needed.

The system admin can view the activity log - the results, attendance records and rankings of the students, current activities of the coaching such as class and test schedules, the notifications about the classes, results and rankings sent out to the students and parents, etc. They can also generate reports of these activities.
The system admin can do all of the activities of other users.
They can also add moderators who will have the same privileges as the admin except being able to delete the admin.

Students can sign up by giving name,username,password,previous academic results,hsc registration number. After giving this information he/she will get a payment request. After payment the student has to wait till the verification of the admin/moderator. After their approval, the student will get a confirmation email.
Students can log in to the system using username or email address and password. After logging in to the system students can view their personal information,previous results, previous recorded classes,current rank among the students,upcoming class schedule,resources provided by the teachers/moderator in their dashboard.Personal information like name,session,institution,email,mobile number.Student can attend the class,give exam.

Parents will be able to sign up by providing their name, email, mobile number, preferred username and finally their child’s/childrens’ username(s) as reference. They will be able to login using their username and password.

Moderators are staff hired by the System Admin to help oversee the website. They can do almost everything an Admin can. The Admin and the moderators can both delete and designate Moderators but the Moderator cannot delete the Admin.
The class schedules can be created, edited and/or removed by the system admin or moderators. They will also be able to edit the attendance records, marksheets, and ranking lists of the students. The system admin can create, upload or delete study resources or materials. They can manually add a student or teacher account.

Prospective Teachers will apply to sign up as a Teacher. They will be redirected to the Teacher Application page where they will give their name, email, phone number, password, valid credentials, which role they want to apply for (eg: which subject they want to teach, do they want to check copies or take classes or make resource materials etc)
The Admin or Moderators will contact the Prospective Teacher based on their information (take interviews if they wish) to hire the Teacher or reject them. After hiring the Teacher, Admin or Moderators will approve that Teacher's Application on the Website, upon which an account will be created for the Teacher.
Admins and Moderators can ban specific email addresses from applying if they are abusing the Teacher Application Form even after being rejected multiple times.

Teachers can host Video Conferences for classes. They can upload study material, resources, check copies, update the marks of their students, and manually give attendance to a student despite the student being absent. A teacher's role determines what they have access to. For example a teacher who only checks copies cannot host Classes, or a teacher who takes Classes cannot update the student's marks. Admins and Moderators can change or add a Teacher's roles.

7. Requirements Modeling

Requirements modeling is the process of understanding and documenting the needs and expectations of stakeholders for a software system. It involves gathering, analyzing, and prioritizing requirements, and then creating visual representations of these requirements using diagrams, charts, or other techniques.

Requirements modeling helps ensure that all stakeholders have a clear understanding of what the software system needs to do, and provides a framework for developers to design and build the system in a way that meets these requirements. It also helps identify potential conflicts or problems early on, allowing for more efficient and effective development.

7.1. Scenario Based Modeling

7.1.1. Use Case Diagrams

Use case diagrams are a visual representation of a software system's functionality and the different ways that users or external systems can interact with it. They are used to model the system's behavior from the user's perspective, and they help ensure that the system is designed to meet user needs. They are an important communication tool for software developers as they provide a clear and concise way to communicate the system's functionality and requirements to all stakeholders involved in the development process.

In a use case diagram, a **primary actor** is the main user or stakeholder who initiates the use case and triggers the system to perform a certain action or behavior.

A **secondary actor** is an external entity that provides a service to the system being modeled or **receives** a service from the system. The secondary actor is not the main user or stakeholder, but they are still important to the use case.

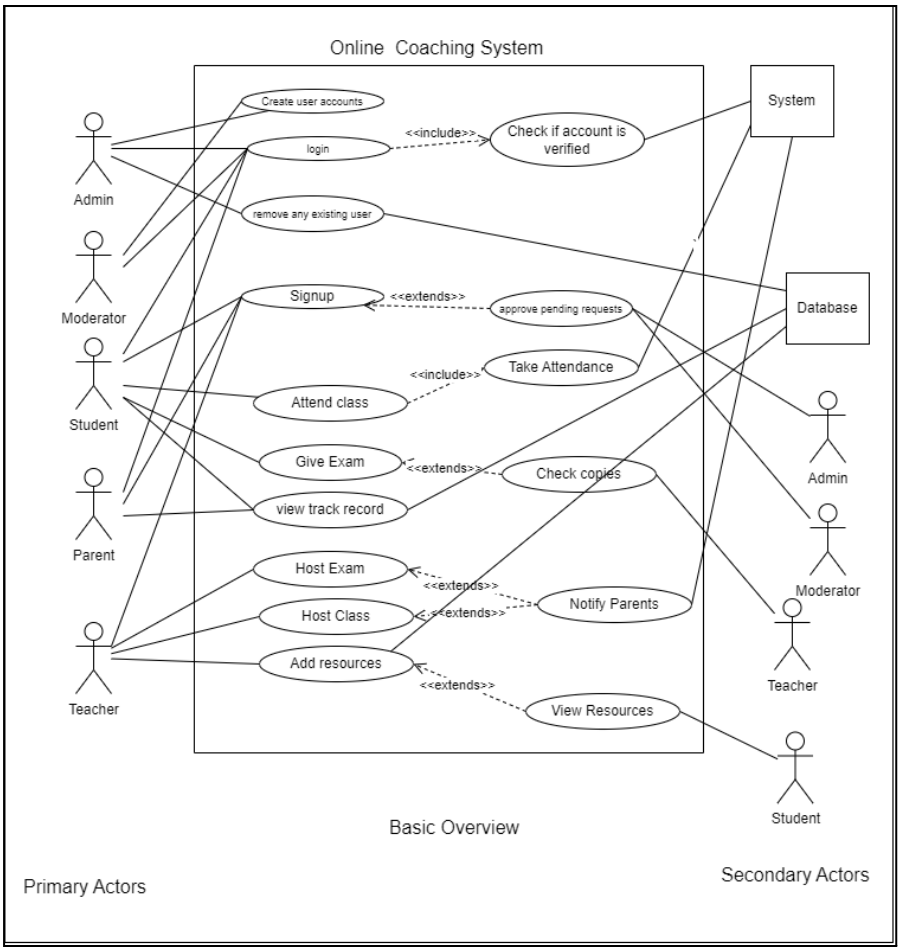
7.1.1.1. Level: 1

Name : Online Coaching System

Primary Actors : Admin, Moderator, Student, Parent, Teacher

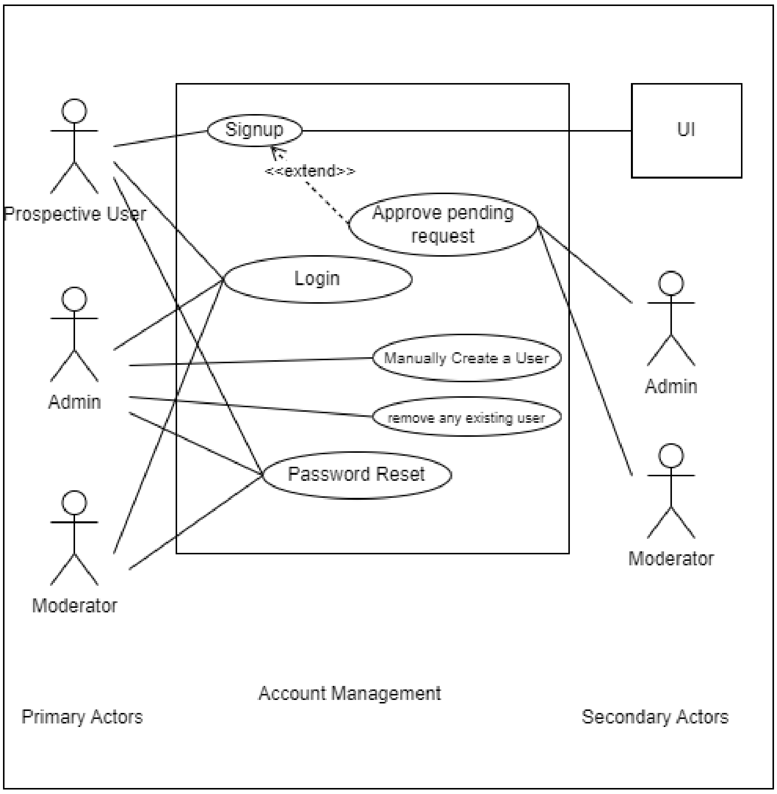
Secondary Actors : System, Database, Admin, Moderator, Teacher, Student

This is a basic overview of the entire system.



7.1.1.2. Level : 1.1

Name : Account Management
Primary Actors : Prospective user, Admin, Moderator.
Secondary Actors : UI, Admin, Moderator.



Description:

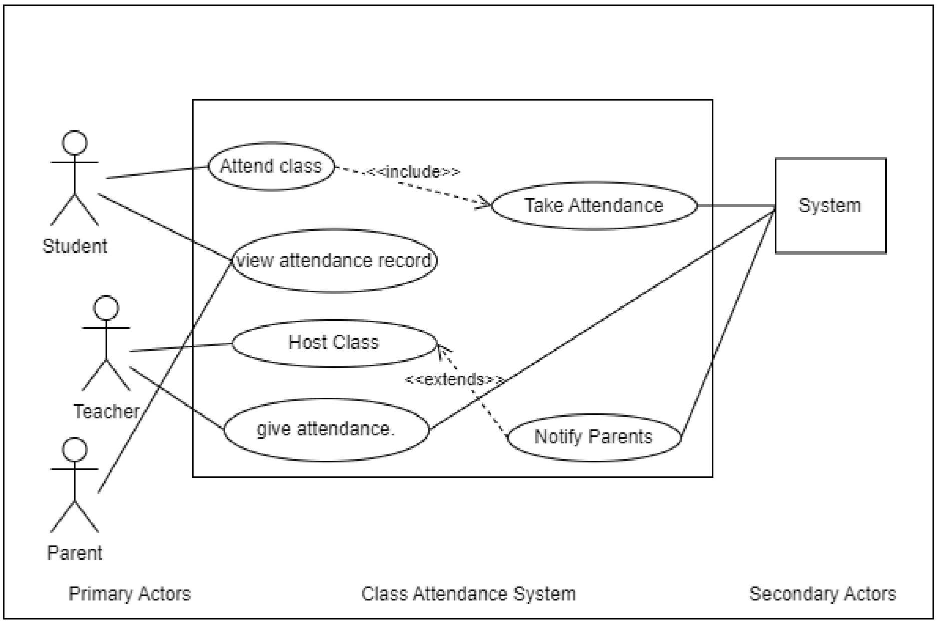
- **Signup:** Prospective users such as students, teachers and parents will be able to create accounts to access the system's features after they are approved by either the admin or a moderator. Students will be able to sign up by providing their name, preferred username, password, previous academic results, and hsc registration number. Parents will be able to sign up by providing their name, email, mobile number, preferred username and their child's name as reference. As for teachers, they will be able to sign up by providing their name, email, phone number, password and the role they want to apply for - their preferred subjects, whether they want to teach, check copies, etc. Once a user has created an account, a request will be sent to the admin and/or moderator(s) to verify the account. For students this will be sent only after they have completed their payment.
- **Login:** All users will be able to login to the system using their username and password.
- **Password reset:** In case a user - a student, parent or teacher forgets their password, there will be an option to reset their password using the email or mobile number they provided at the time of creating their account.
- **Approve pending requests:** The admin and moderators will be able to verify an account

after it is created successfully by an user.

- **Remove any existing user:** Only the admin has the ability to remove any user i.e. the user will not be able to login and access any functionality of the system - to the user it would seem like the account never existed.

7.1.1.3. Level : 1.2

Name : Class Attendance System
Primary Actors : Student, Teacher, Parent
Secondary Actors : System



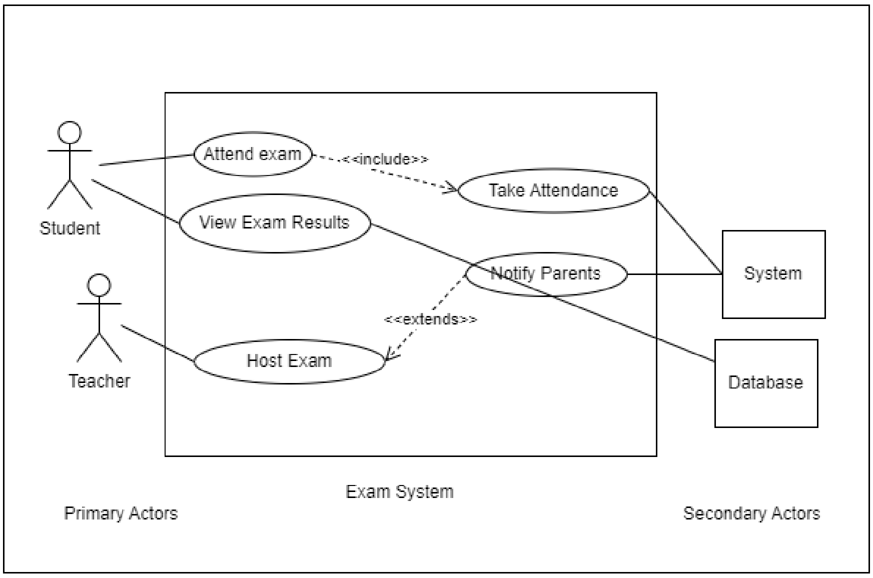
Description:

- **Attend class:** To attend a class, a student will have to log in to the system and go to the link to the class provided in their dashboard. This link can be provided by the admin, a teacher or a moderator.
- **Take attendance:** The system can automatically mark a student as present for a class if the student goes to the class link and is present in the list of attendees for the video conference of the class. The system can also mark a student as absent if they are not present in the list of attendees after a certain time interval.
- **View attendance record:** Students and Parents can view their/their child's attendance record from their respective dashboard. This record is generated by the system according to whether or not a student was present in the list of attendees in the classes the student is enrolled in.
- **Host class:** Teachers can host classes by providing a link to the class's video conference.
- **Notify Parents:** Once a class has started, the system will notify parents via sms and email.

- **Give attendance:** A teacher can manually mark a student as present or absent, while a class is being held. The attendance status of the student in the system will get updated.

7.1.1.4. Level : 1.3

Name : Exam system
Primary Actors : Teacher, Student
Secondary Actors : System, Database

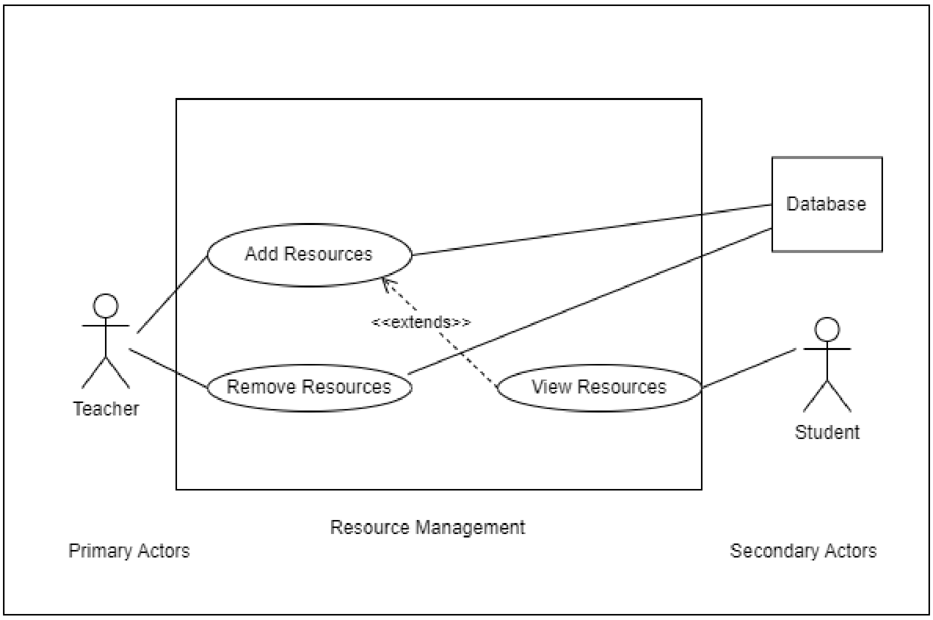


Description:

- **Attend exam:** To attend an exam, a student will have to log in to the system and go to the exam link provided in their dashboard. This exam will be created by a teacher and the link will also be provided by them.
- **Take attendance:** The system can automatically mark a student as present or absent, if they have attended the exam within the specified deadline or otherwise, respectively.
- **View exam results:** Students can view their exam results after the teachers have checked the copies. The students will see they have received no marks if they were absent for the exam. Their exam results will be stored in the database.
- **Host exam:** Teachers will be able to host an exam by adding questions, generating the exam link, setting a duration for the exam and setting a deadline up to which a student can give that exam.
- **Notify Parents:** Once an exam's link has been published the system will notify the parents via their sms and email.

7.1.1.5. Level : 1.4

Name : Resource Management
Primary Actors : Teacher
Secondary Actors : Student, Database



Description:

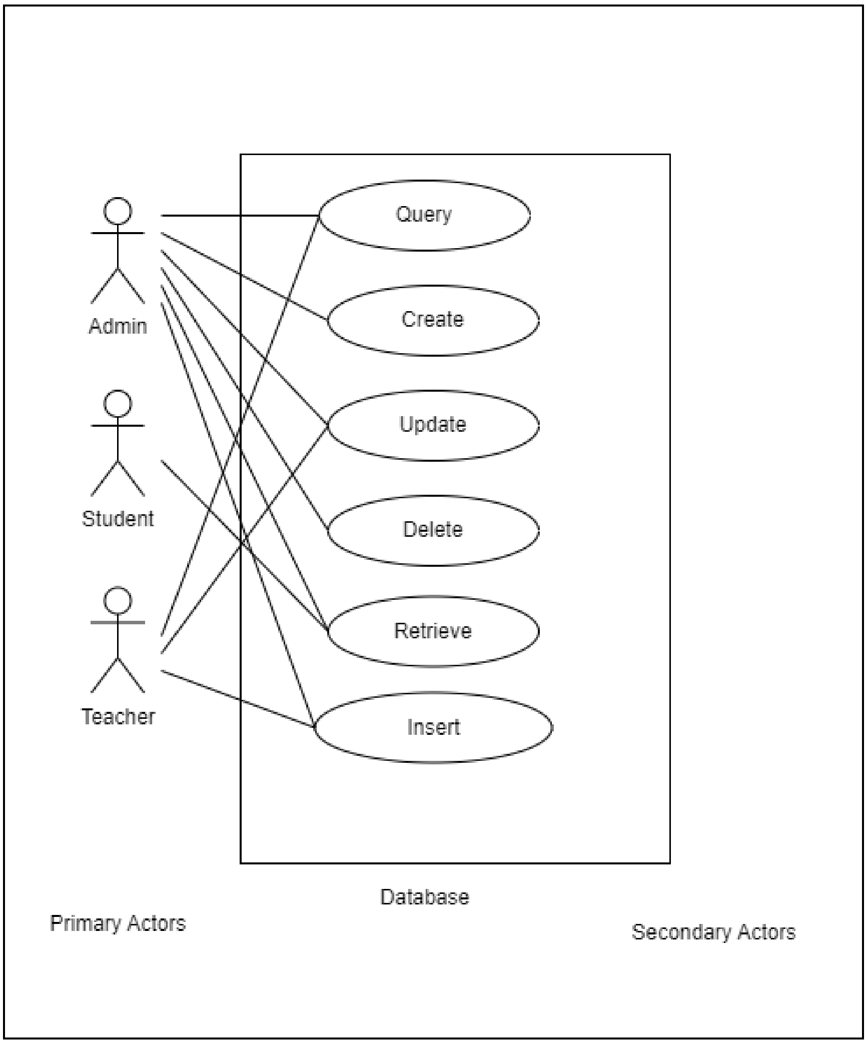
- **Add Resources:** A teacher can upload resources such as study materials, such as books, notes, etc. for the student to view. The formats include pdfs, documents, slides, sheets, or provide links for video reference. They can also provide the class recordings for the student's reference. These resources will be stored in the database.
- **Remove Resources:** Teachers can also remove any resources. These removed resources will be deleted from the database.
- **View resources:** Students can view the resources uploaded by the teachers by going to their dashboard and viewing the resources for a particular subject. These resources will be retrieved from the database.

7.1.1.6. Level : 1.5

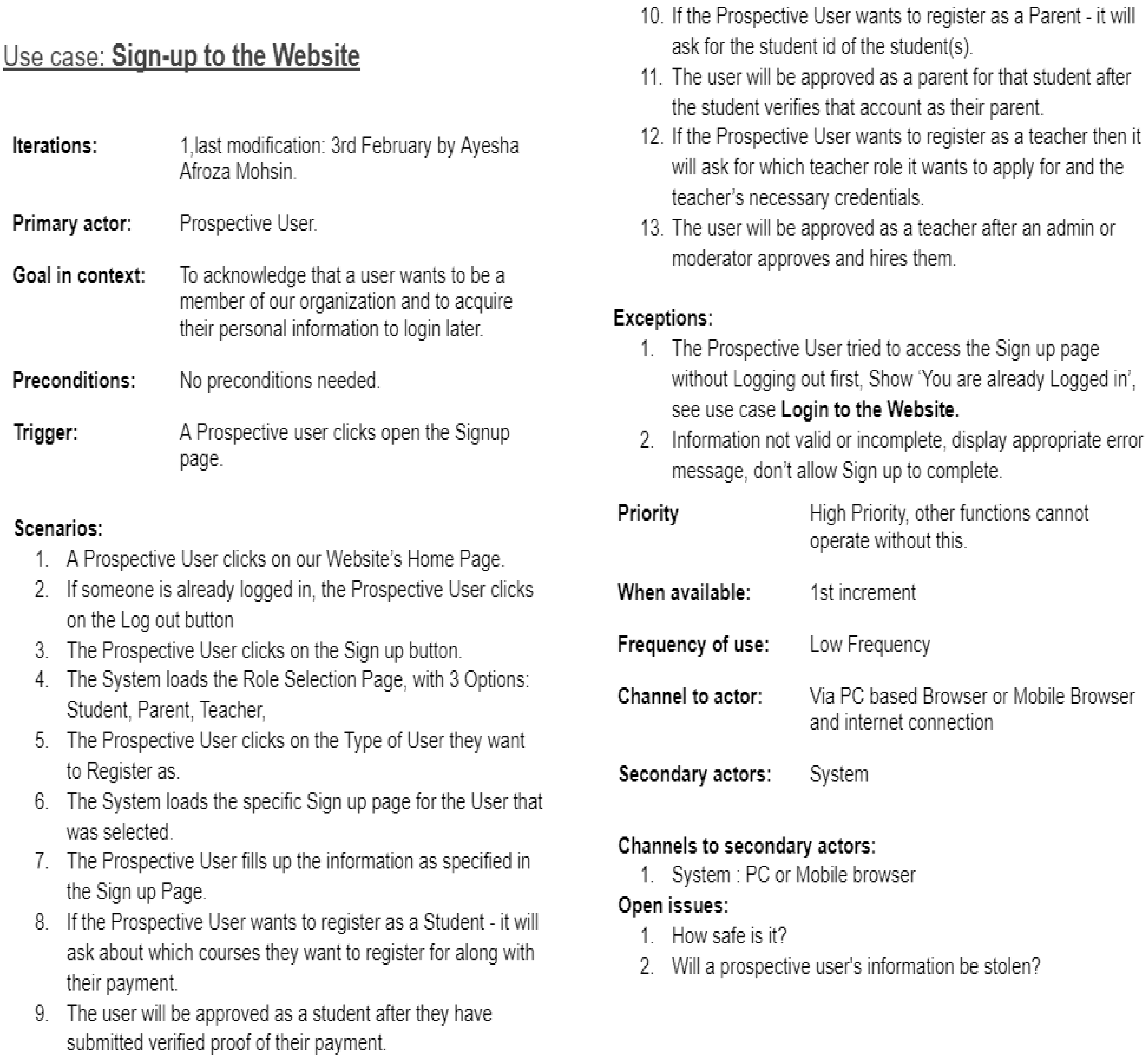
Name : Database

Primary Actors : Admin, Student, Teacher

Secondary Actors : Database.



7.1.2. Detailed Use Case



Use case: **Sign-up to the Website**

Iterations: 1,last modification: 3rd February by Ayesha Afroza Mohsin.

Primary actor: Prospective User.

Goal in context: To acknowledge that a user wants to be a member of our organization and to acquire their personal information to login later.

Preconditions: No preconditions needed.

Trigger: A Prospective user clicks open the Signup page.

Scenarios:

1. A Prospective User clicks on our Website's Home Page.
2. If someone is already logged in, the Prospective User clicks on the Log out button
3. The Prospective User clicks on the Sign up button.
4. The System loads the Role Selection Page, with 3 Options: Student, Parent, Teacher,
5. The Prospective User clicks on the Type of User they want to Register as.
6. The System loads the specific Sign up page for the User that was selected.
7. The Prospective User fills up the information as specified in the Sign up Page.
8. If the Prospective User wants to register as a Student - it will ask about which courses they want to register for along with their payment.
9. The user will be approved as a student after they have submitted verified proof of their payment.
10. If the Prospective User wants to register as a Parent - it will ask for the student id of the student(s).
11. The user will be approved as a parent for that student after the student verifies that account as their parent.
12. If the Prospective User wants to register as a teacher then it will ask for which teacher role it wants to apply for and the teacher's necessary credentials.
13. The user will be approved as a teacher after an admin or moderator approves and hires them.

Exceptions:

1. The Prospective User tried to access the Sign up page without Logging out first, Show 'You are already Logged in', see use case **Login to the Website.**
2. Information not valid or incomplete, display appropriate error message, don't allow Sign up to complete.

Priority High Priority, other functions cannot operate without this.

When available: 1st increment

Frequency of use: Low Frequency

Channel to actor: Via PC based Browser or Mobile Browser and internet connection

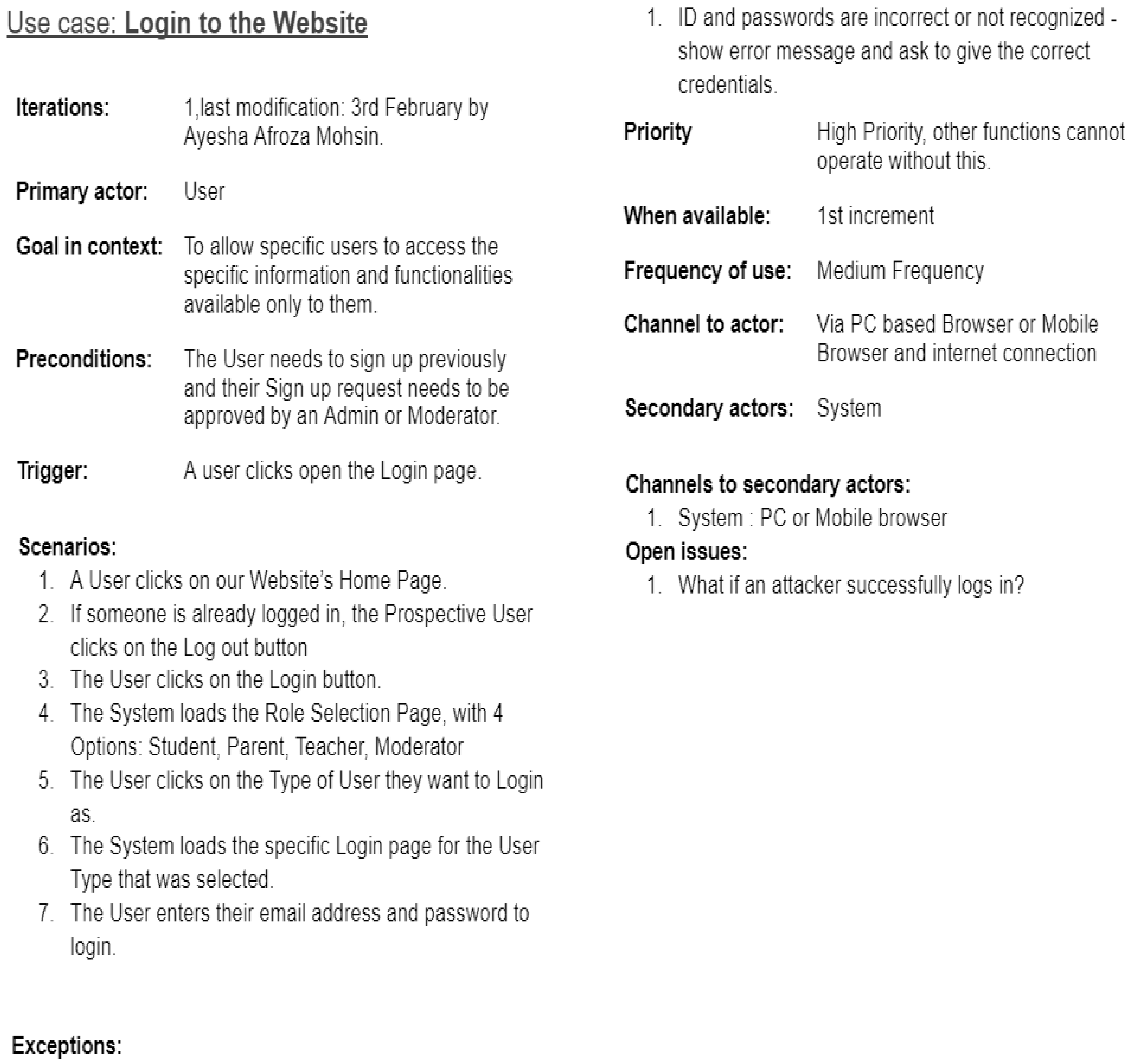
Secondary actors: System

Channels to secondary actors:

1. System : PC or Mobile browser

Open issues:

1. How safe is it?
2. Will a prospective user's information be stolen?



Use case: **Login to the Website**

Iterations: 1,last modification: 3rd February by Ayesha Afroza Mohsin.

Primary actor: User

Goal in context: To allow specific users to access the specific information and functionalities available only to them.

Preconditions: The User needs to sign up previously and their Sign up request needs to be approved by an Admin or Moderator.

Trigger: A user clicks open the Login page.

Scenarios:

1. A User clicks on our Website's Home Page.
2. If someone is already logged in, the Prospective User clicks on the Log out button
3. The User clicks on the Login button.
4. The System loads the Role Selection Page, with 4 Options: Student, Parent, Teacher, Moderator
5. The User clicks on the Type of User they want to Login as.
6. The System loads the specific Login page for the User Type that was selected.
7. The User enters their email address and password to login.

Exceptions:

1. ID and passwords are incorrect or not recognized - show error message and ask to give the correct credentials.

Priority High Priority, other functions cannot operate without this.

When available: 1st increment

Frequency of use: Medium Frequency

Channel to actor: Via PC based Browser or Mobile Browser and internet connection

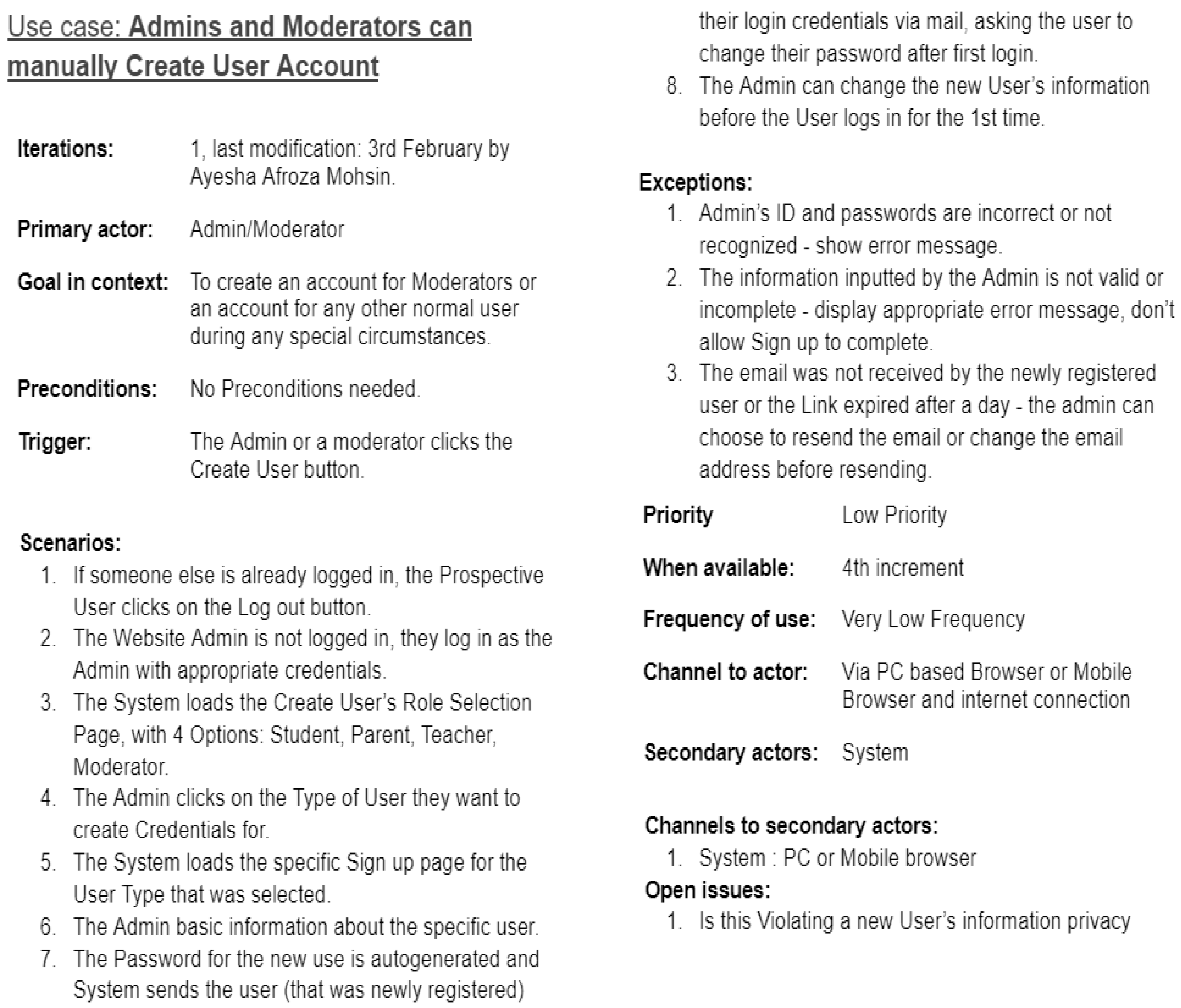
Secondary actors: System

Channels to secondary actors:

1. System : PC or Mobile browser

Open issues:

1. What if an attacker successfully logs in?



Use case: **Admins and Moderators can manually Create User Account**

Iterations: 1, last modification: 3rd February by Ayesha Afroza Mohsin.

Primary actor: Admin/Moderator

Goal in context: To create an account for Moderators or an account for any other normal user during any special circumstances.

Preconditions: No Preconditions needed.

Trigger: The Admin or a moderator clicks the Create User button.

Scenarios:

1. If someone else is already logged in, the Prospective User clicks on the Log out button.
2. The Website Admin is not logged in, they log in as the Admin with appropriate credentials.
3. The System loads the Create User's Role Selection Page, with 4 Options: Student, Parent, Teacher, Moderator.
4. The Admin clicks on the Type of User they want to create Credentials for.
5. The System loads the specific Sign up page for the User Type that was selected.
6. The Admin basic information about the specific user.
7. The Password for the new use is autogenerated and System sends the user (that was newly registered) their login credentials via mail, asking the user to change their password after first login.
8. The Admin can change the new User's information before the User logs in for the 1st time.

Exceptions:

1. Admin's ID and passwords are incorrect or not recognized - show error message.
2. The information inputted by the Admin is not valid or incomplete - display appropriate error message, don't allow Sign up to complete.
3. The email was not received by the newly registered user or the Link expired after a day - the admin can choose to resend the email or change the email address before resending.

Priority Low Priority

When available: 4th increment

Frequency of use: Very Low Frequency

Channel to actor: Via PC based Browser or Mobile Browser and internet connection

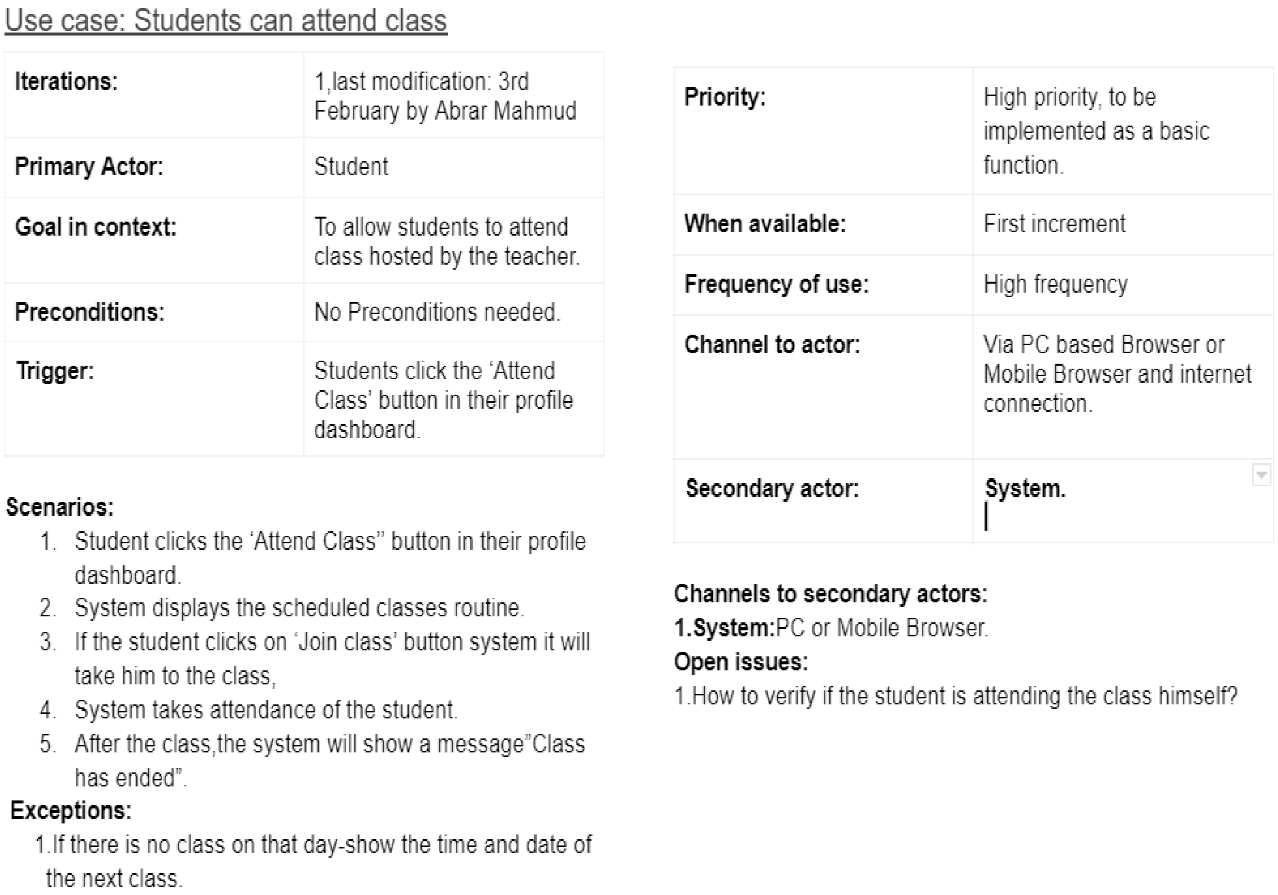
Secondary actors: System

Channels to secondary actors:

1. System : PC or Mobile browser

Open issues:

1. Is this Violating a new User's information privacy



Use case: Students can attend class

Iterations:	1,last modification: 3rd February by Abrar Mahmud
Primary Actor:	Student
Goal in context:	To allow students to attend class hosted by the teacher.
Preconditions:	No Preconditions needed.
Trigger:	Students click the 'Attend Class' button in their profile dashboard.

Scenarios:

1. Student clicks the 'Attend Class'' button in their profile dashboard.
2. System displays the scheduled classes routine.
3. If the student clicks on 'Join class' button system it will take him to the class,
4. System takes attendance of the student.
5. After the class,the system will show a message"Class has ended".

Exceptions:

1.If there is no class on that day-show the time and date of the next class.

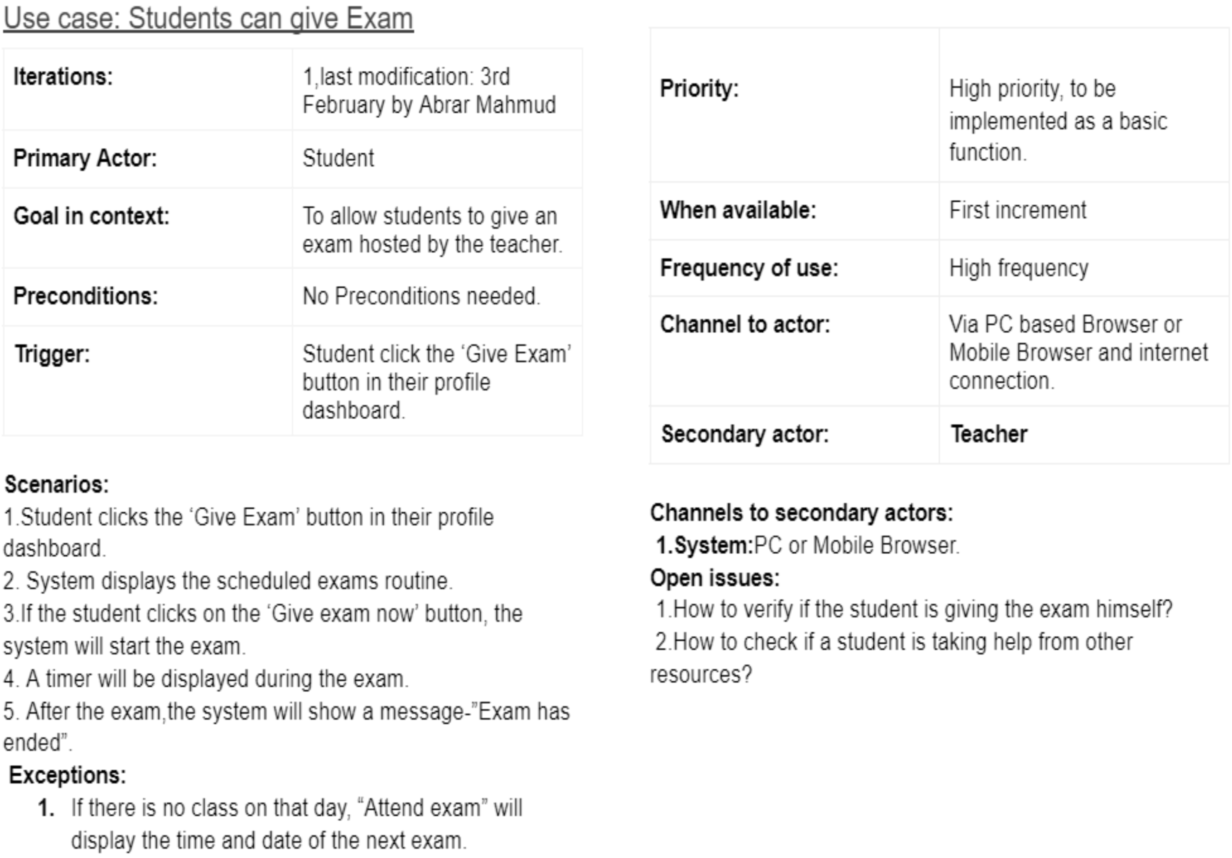
Priority:	High priority, to be implemented as a basic function.
When available:	First increment
Frequency of use:	High frequency
Channel to actor:	Via PC based Browser or Mobile Browser and internet connection.
Secondary actor:	**System.**

Channels to secondary actors:

1.System:PC or Mobile Browser.

Open issues:

1.How to verify if the student is attending the class himself?



Use case: Students can give Exam

Iterations:	1,last modification: 3rd February by Abrar Mahmud
Primary Actor:	Student
Goal in context:	To allow students to give an exam hosted by the teacher.
Preconditions:	No Preconditions needed.
Trigger:	Student click the 'Give Exam' button in their profile dashboard.

Scenarios:

1.Student clicks the 'Give Exam' button in their profile dashboard.

2. System displays the scheduled exams routine.

3.If the student clicks on the 'Give exam now' button, the system will start the exam.

4. A timer will be displayed during the exam.

5. After the exam,the system will show a message-"Exam has ended".

Exceptions:

1. If there is no class on that day, "Attend exam" will display the time and date of the next exam.

Priority:	High priority, to be implemented as a basic function.
When available:	First increment
Frequency of use:	High frequency
Channel to actor:	Via PC based Browser or Mobile Browser and internet connection.
Secondary actor:	**Teacher**

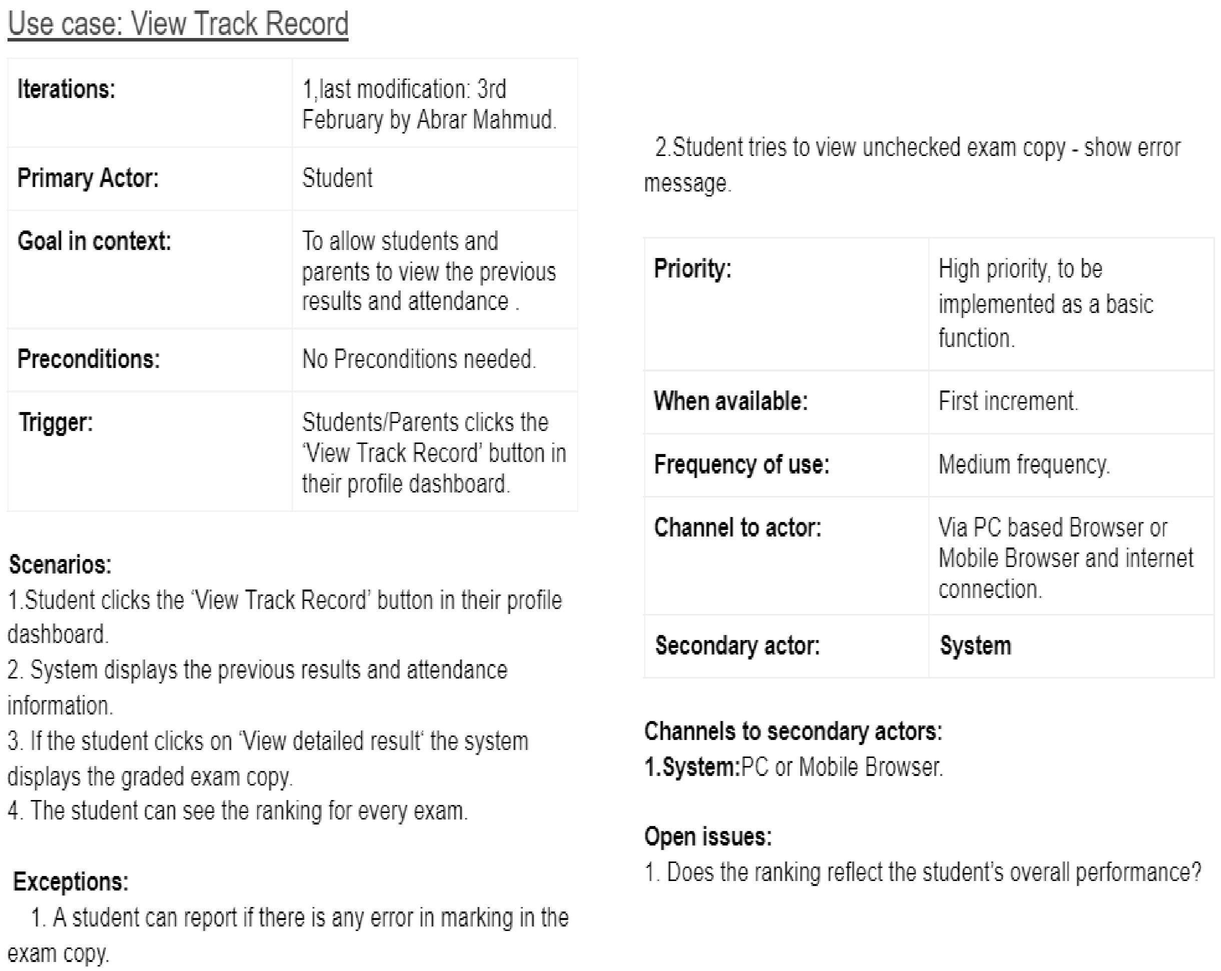
Channels to secondary actors:

1.System:PC or Mobile Browser.

Open issues:

1.How to verify if the student is giving the exam himself?

2.How to check if a student is taking help from other resources?



Use case: View Track Record

Iterations:	1,last modification: 3rd February by Abrar Mahmud.
Primary Actor:	Student
Goal in context:	To allow students and parents to view the previous results and attendance .
Preconditions:	No Preconditions needed.
Trigger:	Students/Parents clicks the 'View Track Record' button in their profile dashboard.

Scenarios:

1.Student clicks the 'View Track Record' button in their profile dashboard.

2. System displays the previous results and attendance information.

3. If the student clicks on 'View detailed result' the system displays the graded exam copy.

4. The student can see the ranking for every exam.

Exceptions:

1. A student can report if there is any error in marking in the exam copy.

2.Student tries to view unchecked exam copy - show error message.

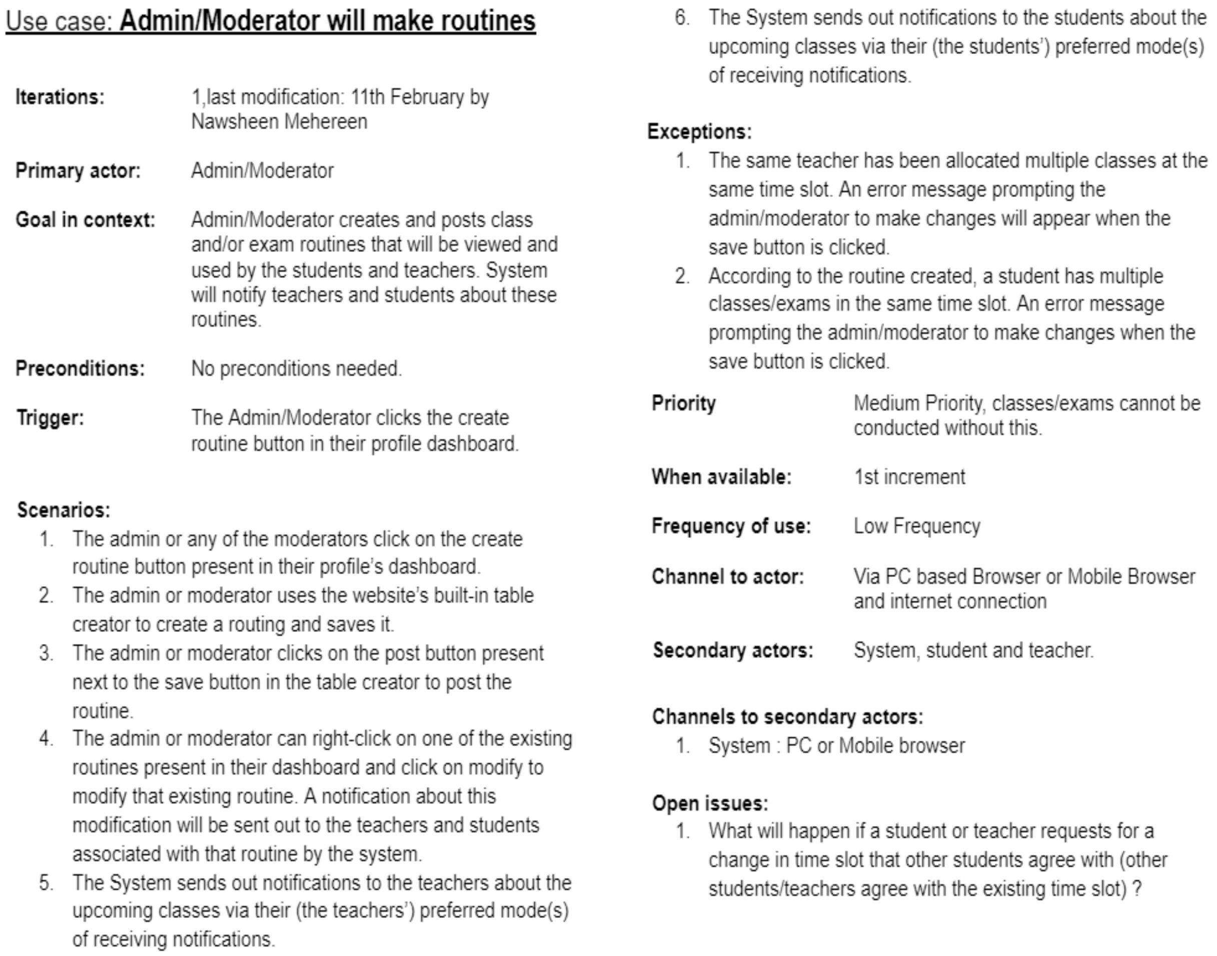
Priority:	High priority, to be implemented as a basic function.
When available:	First increment.
Frequency of use:	Medium frequency.
Channel to actor:	Via PC based Browser or Mobile Browser and internet connection.
Secondary actor:	**System**

Channels to secondary actors:

1.System:PC or Mobile Browser.

Open issues:

1. Does the ranking reflect the student's overall performance?



Use case: **Admin/Moderator will make routines**

Iterations: 1,last modification: 11th February by Nawsheen Mehereen

Primary actor: Admin/Moderator

Goal in context: Admin/Moderator creates and posts class and/or exam routines that will be viewed and used by the students and teachers. System will notify teachers and students about these routines.

Preconditions: No preconditions needed.

Trigger: The Admin/Moderator clicks the create routine button in their profile dashboard.

Scenarios:

1. The admin or any of the moderators click on the create routine button present in their profile's dashboard.
2. The admin or moderator uses the website's built-in table creator to create a routing and saves it.
3. The admin or moderator clicks on the post button present next to the save button in the table creator to post the routine.
4. The admin or moderator can right-click on one of the existing routines present in their dashboard and click on modify to modify that existing routine. A notification about this modification will be sent out to the teachers and students associated with that routine by the system.
5. The System sends out notifications to the teachers about the upcoming classes via their (the teachers') preferred mode(s) of receiving notifications.
6. The System sends out notifications to the students about the upcoming classes via their (the students') preferred mode(s) of receiving notifications.

Exceptions:

1. The same teacher has been allocated multiple classes at the same time slot. An error message prompting the admin/moderator to make changes will appear when the save button is clicked.
2. According to the routine created, a student has multiple classes/exams in the same time slot. An error message prompting the admin/moderator to make changes when the save button is clicked.

Priority Medium Priority, classes/exams cannot be conducted without this.

When available: 1st increment

Frequency of use: Low Frequency

Channel to actor: Via PC based Browser or Mobile Browser and internet connection

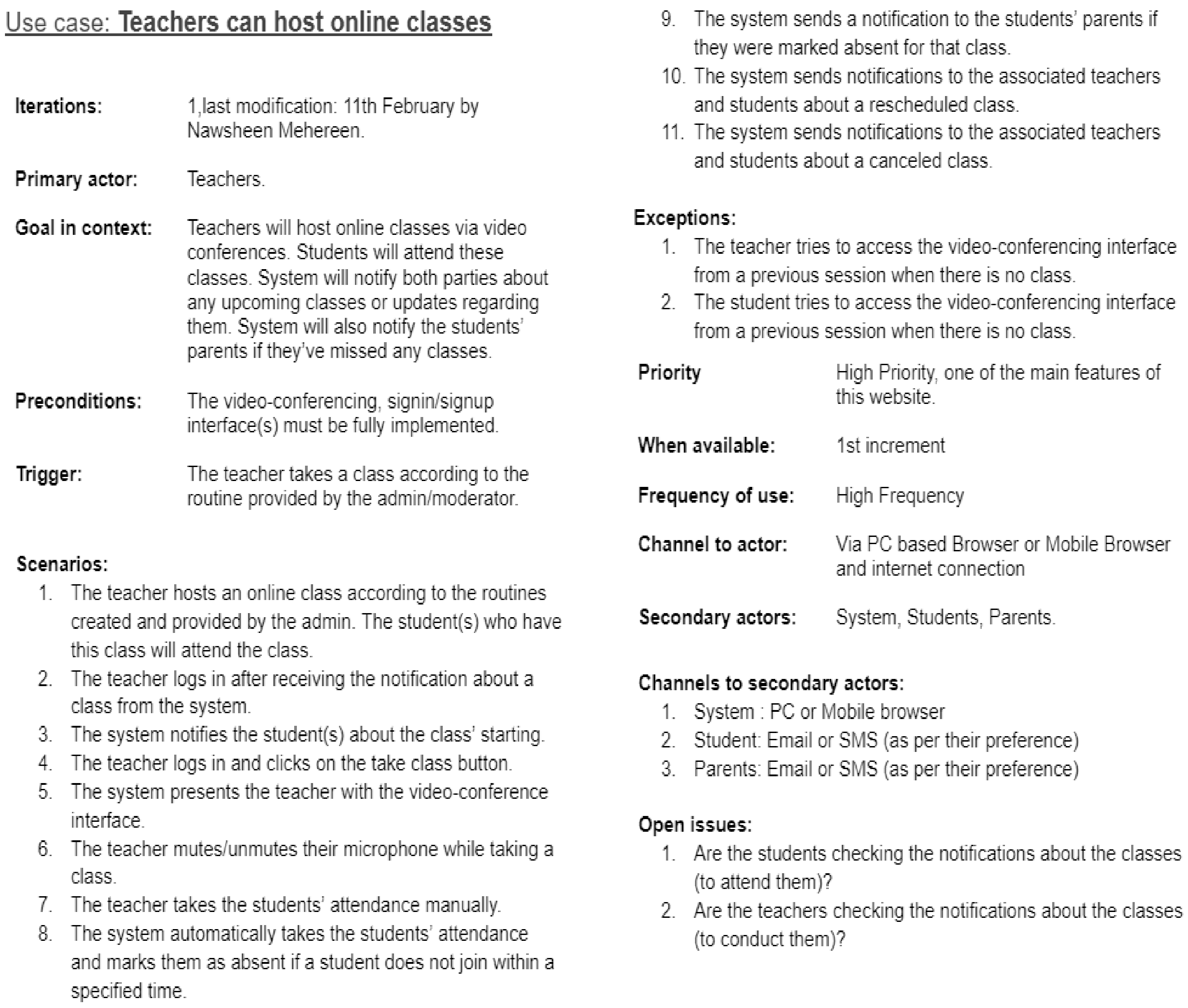
Secondary actors: System, student and teacher.

Channels to secondary actors:

1. System : PC or Mobile browser

Open issues:

1. What will happen if a student or teacher requests for a change in time slot that other students agree with (other students/teachers agree with the existing time slot) ?



Use case: **Teachers can host online classes**

Iterations: 1,last modification: 11th February by Nawsheen Mehereen.

Primary actor: Teachers.

Goal in context: Teachers will host online classes via video conferences. Students will attend these classes. System will notify both parties about any upcoming classes or updates regarding them. System will also notify the students' parents if they've missed any classes.

Preconditions: The video-conferencing, signin/signup interface(s) must be fully implemented.

Trigger: The teacher takes a class according to the routine provided by the admin/moderator.

Scenarios:

1. The teacher hosts an online class according to the routines created and provided by the admin. The student(s) who have this class will attend the class.
2. The teacher logs in after receiving the notification about a class from the system.
3. The system notifies the student(s) about the class' starting.
4. The teacher logs in and clicks on the take class button.
5. The system presents the teacher with the video-conference interface.
6. The teacher mutes/unmutes their microphone while taking a class.
7. The teacher takes the students' attendance manually.
8. The system automatically takes the students' attendance and marks them as absent if a student does not join within a specified time.
9. The system sends a notification to the students' parents if they were marked absent for that class.
10. The system sends notifications to the associated teachers and students about a rescheduled class.
11. The system sends notifications to the associated teachers and students about a canceled class.

Exceptions:

1. The teacher tries to access the video-conferencing interface from a previous session when there is no class.
2. The student tries to access the video-conferencing interface from a previous session when there is no class.

Priority High Priority, one of the main features of this website.

When available: 1st increment

Frequency of use: High Frequency

Channel to actor: Via PC based Browser or Mobile Browser and internet connection

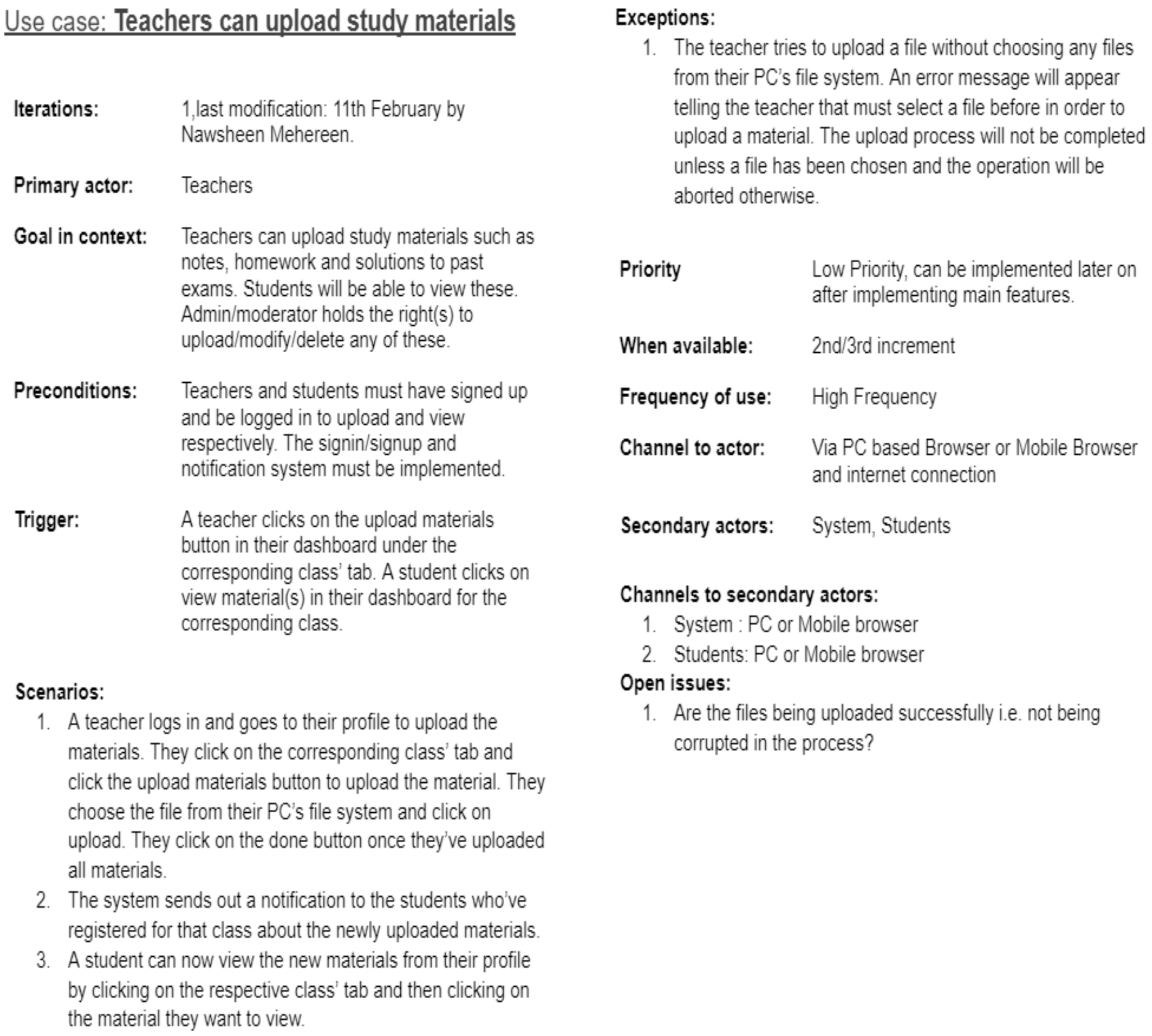
Secondary actors: System, Students, Parents.

Channels to secondary actors:

1. System : PC or Mobile browser
2. Student: Email or SMS (as per their preference)
3. Parents: Email or SMS (as per their preference)

Open issues:

1. Are the students checking the notifications about the classes (to attend them)?
2. Are the teachers checking the notifications about the classes (to conduct them)?



Use case: **Teachers can upload study materials**

Iterations: 1,last modification: 11th February by Nawsheen Mehereen.

Primary actor: Teachers

Goal in context: Teachers can upload study materials such as notes, homework and solutions to past exams. Students will be able to view these. Admin/moderator holds the right(s) to upload/modify/delete any of these.

Preconditions: Teachers and students must have signed up and be logged in to upload and view respectively. The signin/signup and notification system must be implemented.

Trigger: A teacher clicks on the upload materials button in their dashboard under the corresponding class' tab. A student clicks on view material(s) in their dashboard for the corresponding class.

Scenarios:

1. A teacher logs in and goes to their profile to upload the materials. They click on the corresponding class' tab and click the upload materials button to upload the material. They choose the file from their PC's file system and click on upload. They click on the done button once they've uploaded all materials.
2. The system sends out a notification to the students who've registered for that class about the newly uploaded materials.
3. A student can now view the new materials from their profile by clicking on the respective class' tab and then clicking on the material they want to view.

Exceptions:

1. The teacher tries to upload a file without choosing any files from their PC's file system. An error message will appear telling the teacher that must select a file before in order to upload a material. The upload process will not be completed unless a file has been chosen and the operation will be aborted otherwise.

Priority Low Priority, can be implemented later on after implementing main features.

When available: 2nd/3rd increment

Frequency of use: High Frequency

Channel to actor: Via PC based Browser or Mobile Browser and internet connection

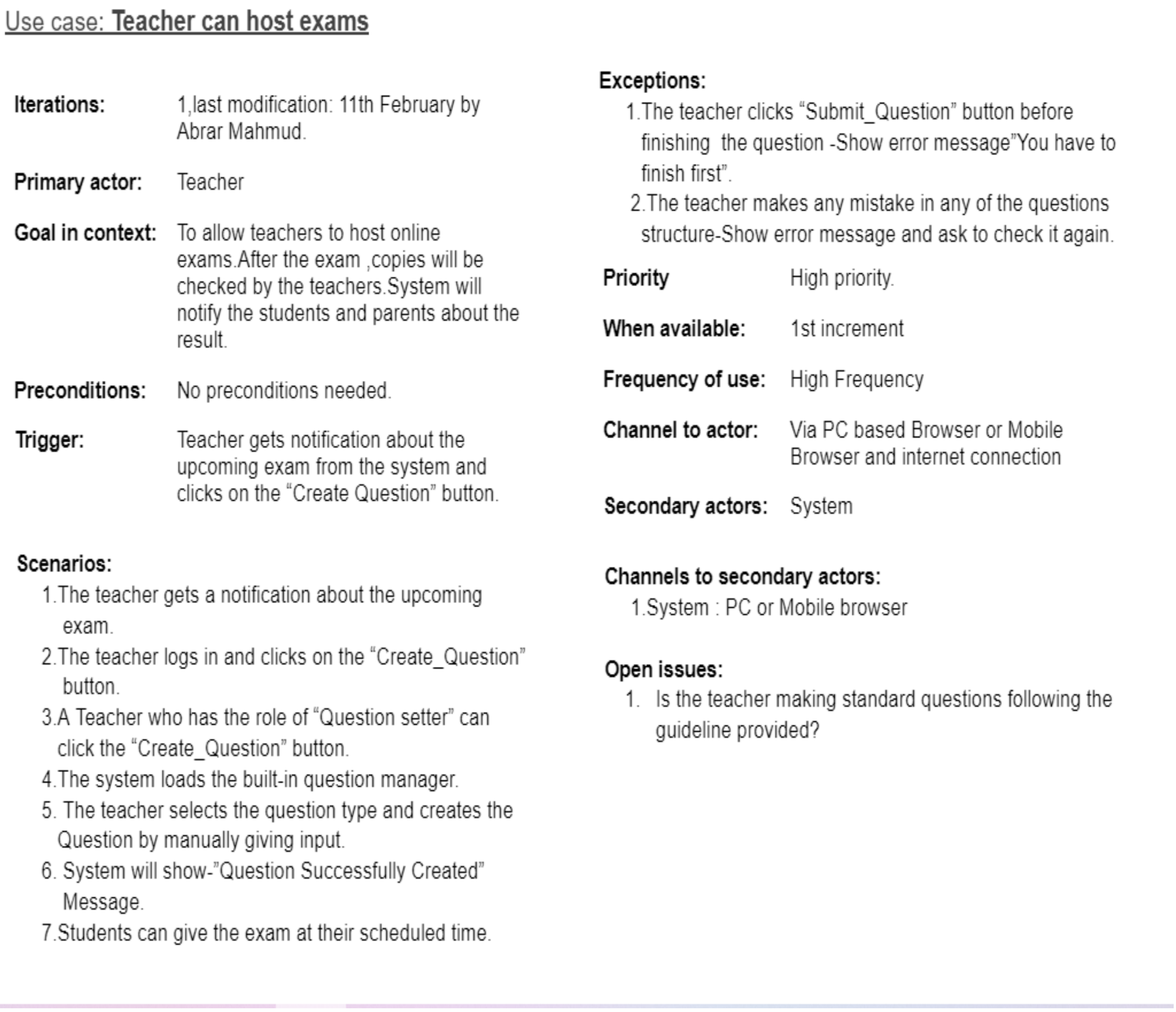
Secondary actors: System, Students

Channels to secondary actors:

1. System : PC or Mobile browser
2. Students: PC or Mobile browser

Open issues:

1. Are the files being uploaded successfully i.e. not being corrupted in the process?



Use case: **Teacher can host exams**

Iterations:	1,last modification: 11th February by Abrar Mahmud.
Primary actor:	Teacher
Goal in context:	To allow teachers to host online exams.After the exam ,copies will be checked by the teachers.System will notify the students and parents about the result.
Preconditions:	No preconditions needed.
Trigger:	Teacher gets notification about the upcoming exam from the system and clicks on the “Create Question” button.

Scenarios:

1.The teacher gets a notification about the upcoming exam.

2.The teacher logs in and clicks on the “Create_Question” button.

3.A Teacher who has the role of “Question setter” can click the “Create_Question” button.

4.The system loads the built-in question manager.

5. The teacher selects the question type and creates the Question by manually giving input.

6. System will show-”Question Successfully Created” Message.

7.Students can give the exam at their scheduled time.

Exceptions:

1.The teacher clicks “Submit_Question” button before finishing the question -Show error message”You have to finish first”.

2.The teacher makes any mistake in any of the questions structure-Show error message and ask to check it again.

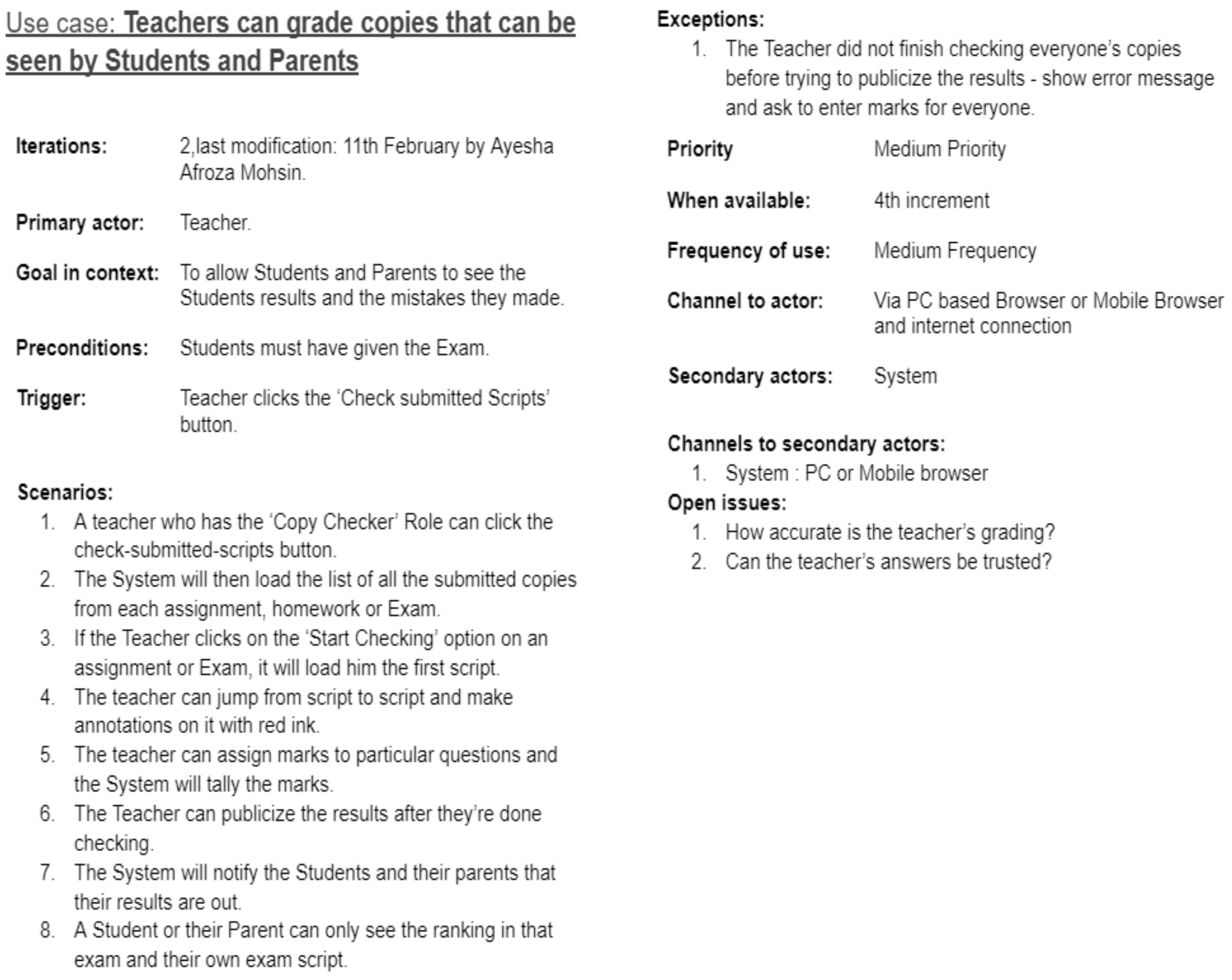
Priority	High priority.
When available:	1st increment
Frequency of use:	High Frequency
Channel to actor:	Via PC based Browser or Mobile Browser and internet connection
Secondary actors:	System

Channels to secondary actors:

1.System : PC or Mobile browser

Open issues:

1. Is the teacher making standard questions following the guideline provided?



Use case: **Teachers can grade copies that can be seen by Students and Parents**

Iterations:	2,last modification: 11th February by Ayesha Afroza Mohsin.
Primary actor:	Teacher.
Goal in context:	To allow Students and Parents to see the Students results and the mistakes they made.
Preconditions:	Students must have given the Exam.
Trigger:	Teacher clicks the ‘Check submitted Scripts’ button.

Scenarios:

1. A teacher who has the ‘Copy Checker’ Role can click the check-submitted-scripts button.
2. The System will then load the list of all the submitted copies from each assignment, homework or Exam.
3. If the Teacher clicks on the ‘Start Checking’ option on an assignment or Exam, it will load him the first script.
4. The teacher can jump from script to script and make annotations on it with red ink.
5. The teacher can assign marks to particular questions and the System will tally the marks.
6. The Teacher can publicize the results after they’re done checking.
7. The System will notify the Students and their parents that their results are out.
8. A Student or their Parent can only see the ranking in that exam and their own exam script.

Exceptions:

1. The Teacher did not finish checking everyone’s copies before trying to publicize the results - show error message and ask to enter marks for everyone.

Priority	Medium Priority
When available:	4th increment
Frequency of use:	Medium Frequency
Channel to actor:	Via PC based Browser or Mobile Browser and internet connection
Secondary actors:	System

Channels to secondary actors:

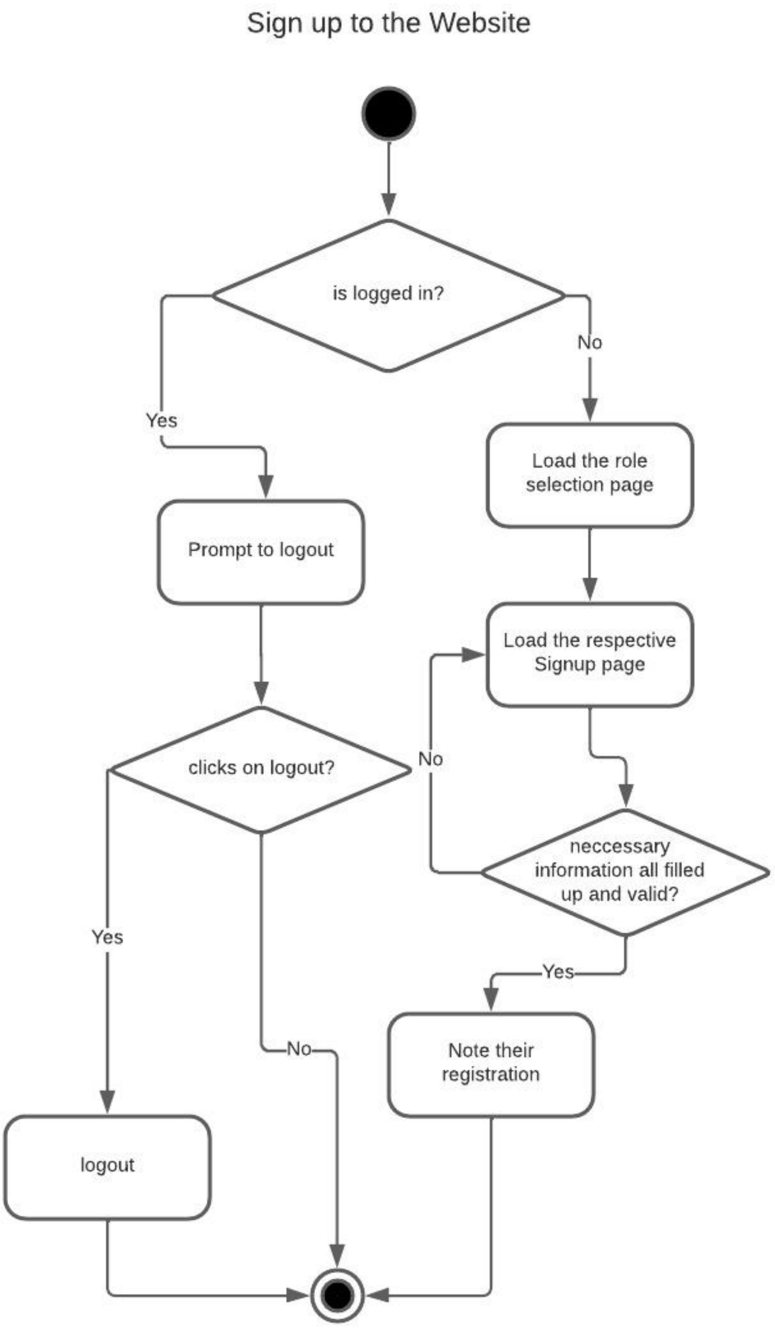
1. System : PC or Mobile browser

Open issues:

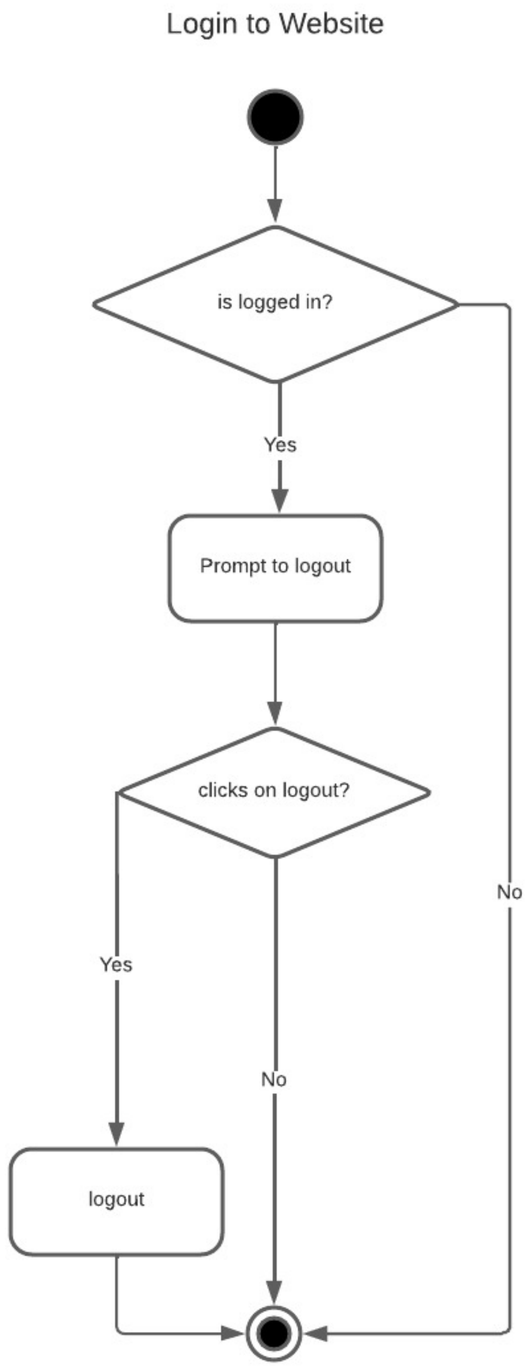
1. How accurate is the teacher’s grading?
2. Can the teacher’s answers be trusted?

7.1.3. Activity Diagram

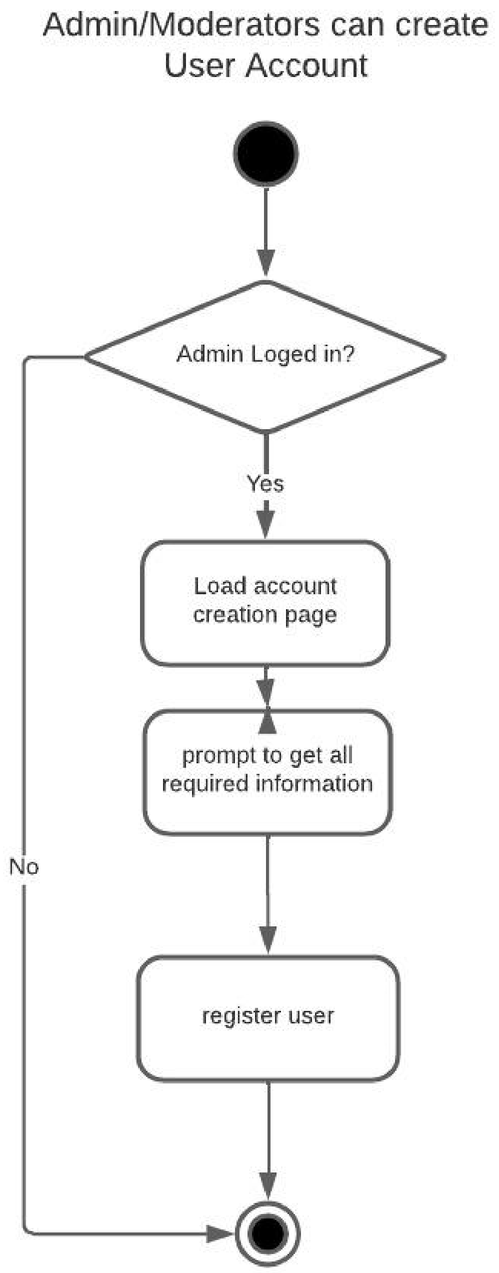
7.1.3.1. Sign up to the Website



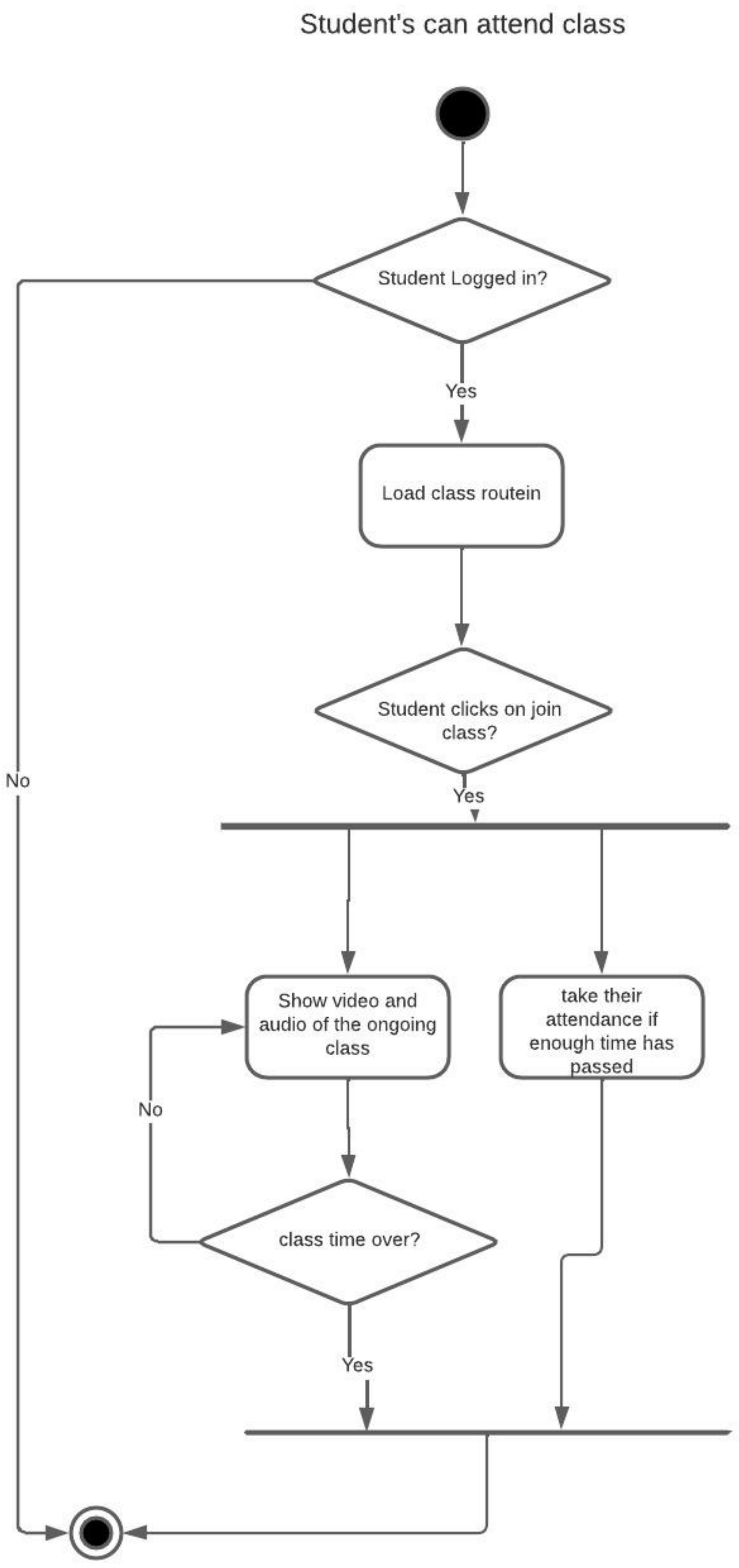
7.1.3.2. Log in


Login to Website
is logged in?
Yes
No
Prompt to logout
clicks on logout?
Yes
No
logout

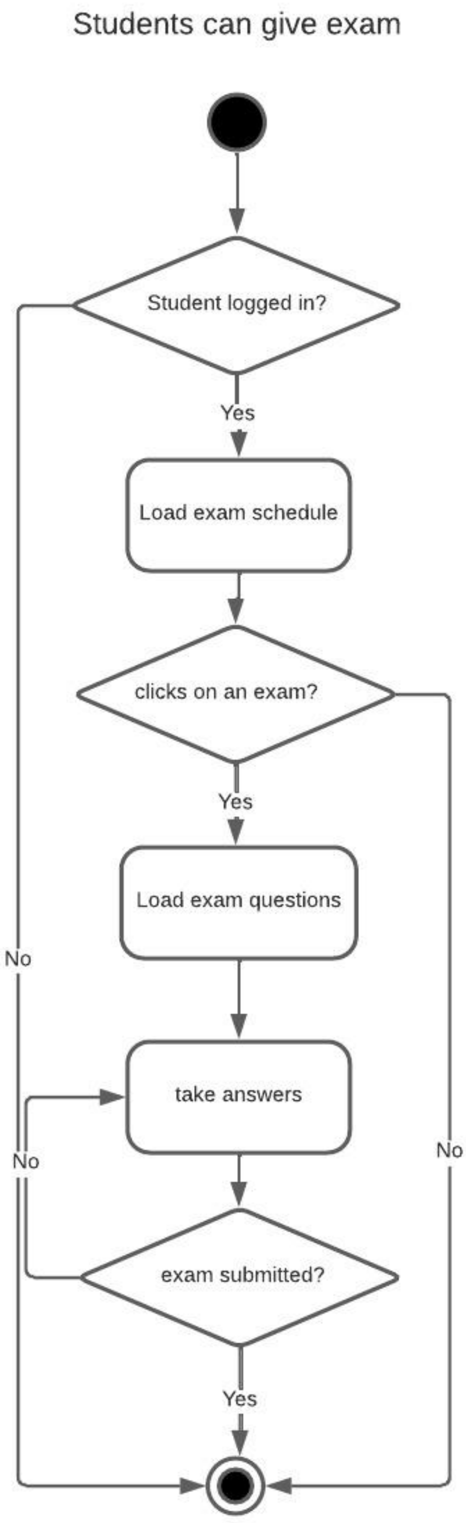
7.1.3.3. Admin/Moderators can create User Account


Admin/Moderators can create User Account
Admin Loged in?
Yes
No
Load account creation page
prompt to get all required information
register user

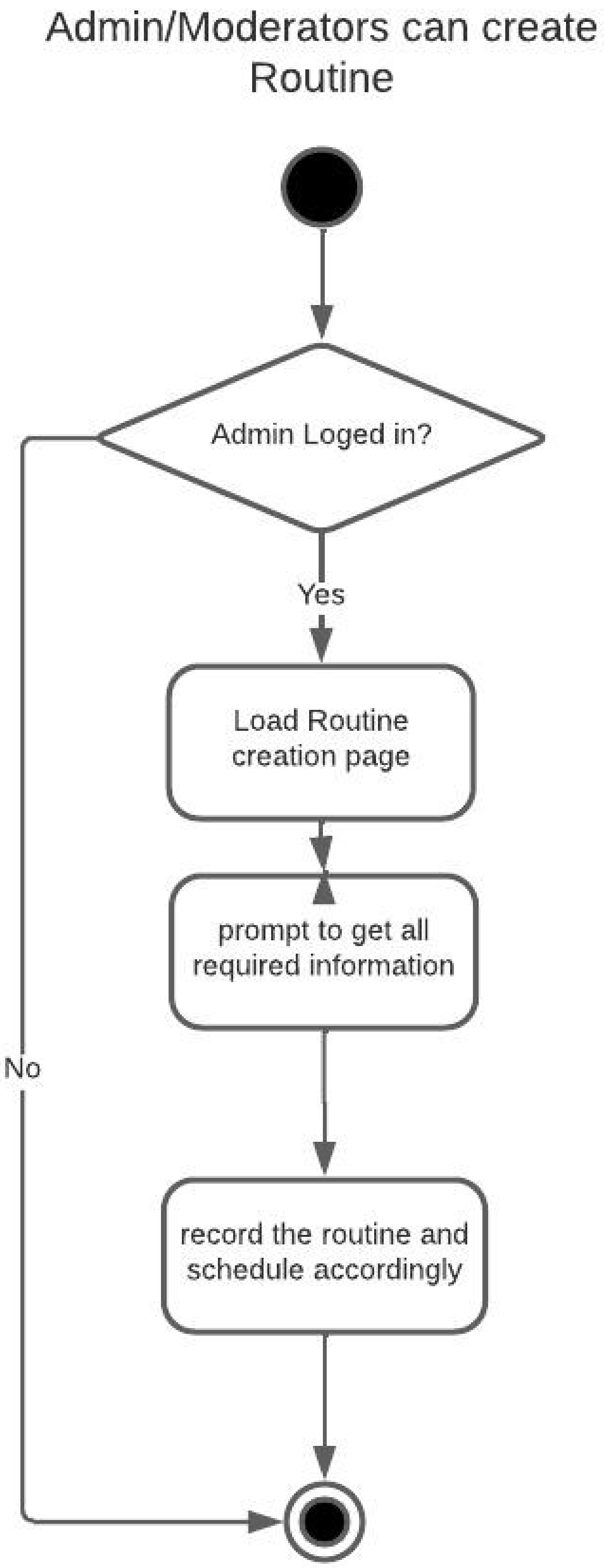
7.1.3.4. Students can attend classes


Student's can attend class
Student Logged in?
Yes
No
Load class routein
Student clicks on join class?
Yes
Show video and audio of the ongoing class
take their attendance if enough time has passed
No
class time over?
Yes

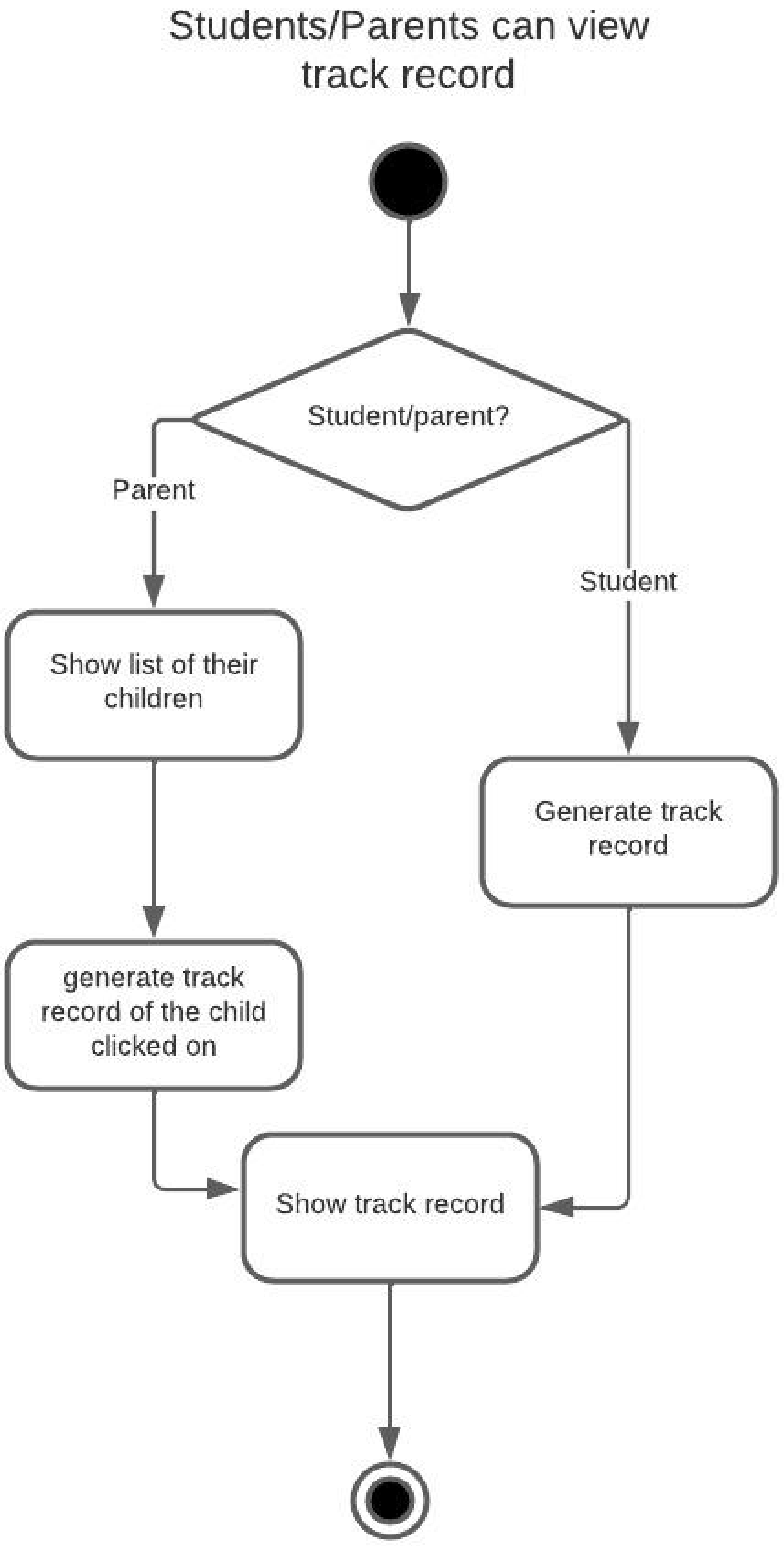
7.1.3.5. Students can give Exam



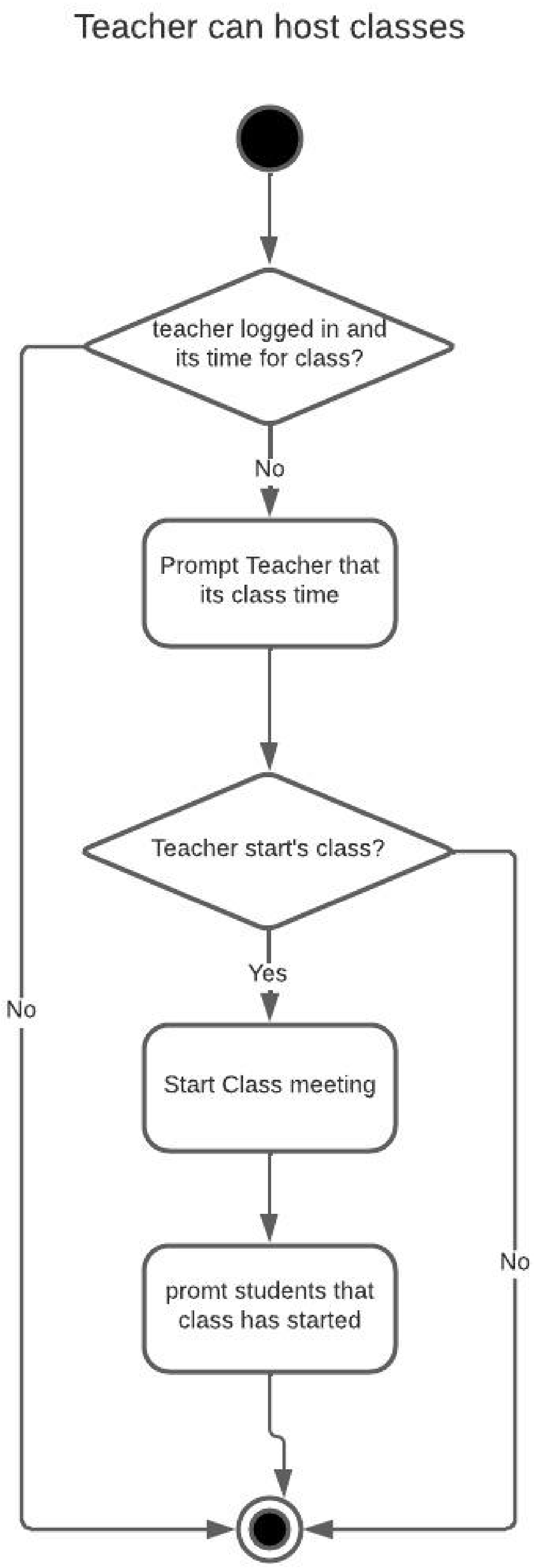
7.1.3.6. Students/Parents can view track records


Admin/Moderators can create Routine
Admin Loged in?
Yes
No
Load Routine creation page
prompt to get all required information
record the routine and schedule accordingly

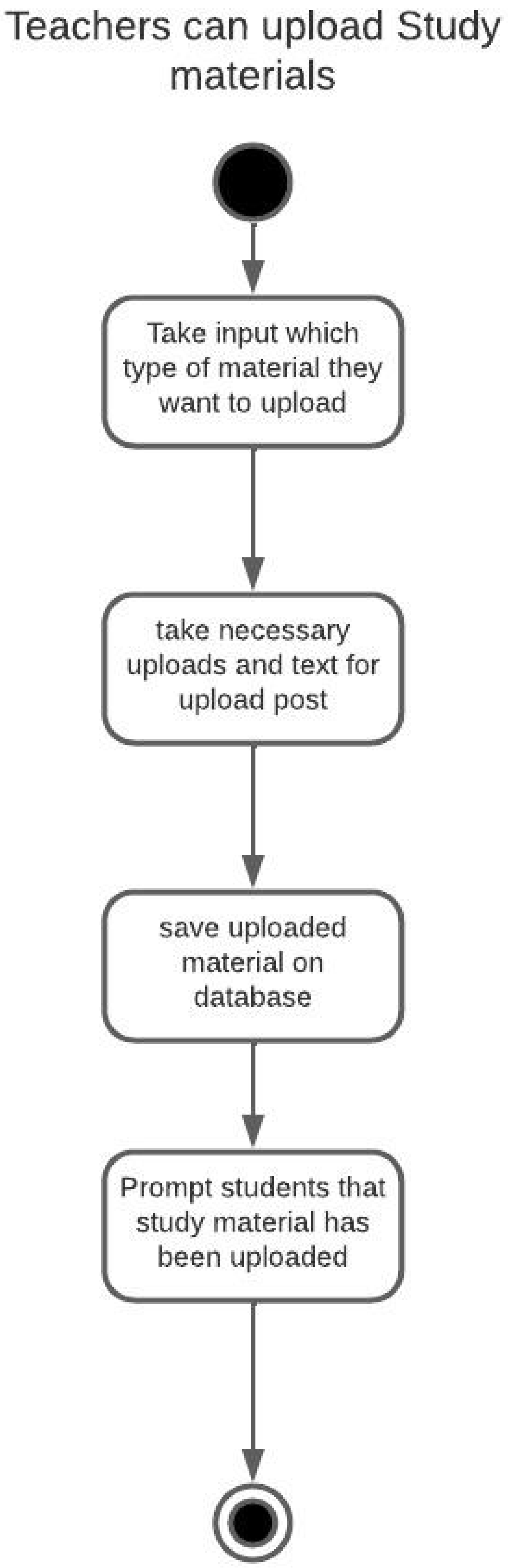
7.1.3.7. Admin/Moderators can create Routine


Students/Parents can view track record
Student/parent?
Parent
Student
Show list of their children
Generate track record
generate track record of the child clicked on
Show track record

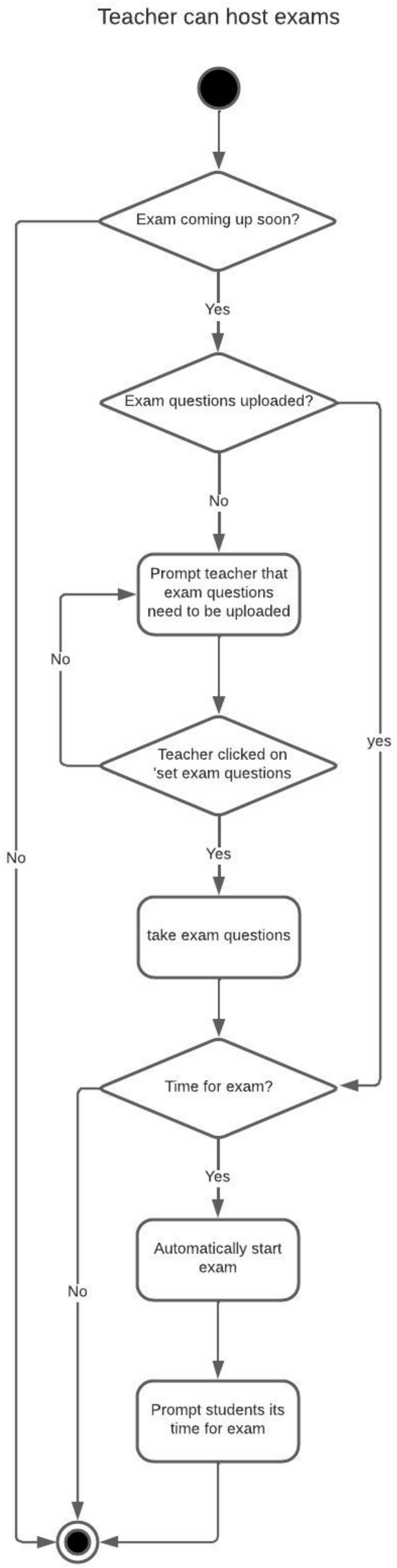
7.1.3.8. Teacher can host classes


Teacher can host classes
teacher logged in and its time for class?
No
Prompt Teacher that its class time
Teacher start's class?
Yes
Start Class meeting
promt students that class has started
No
No

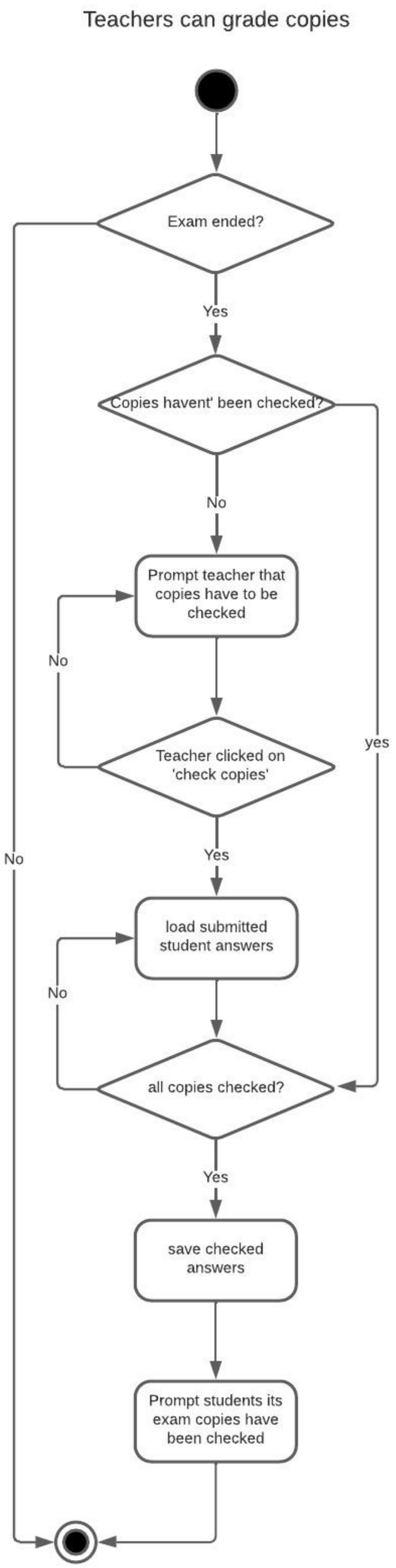
7.1.3.9. Teacher can upload Study materials


Teachers can upload Study materials
Take input which type of material they want to upload
take necessary uploads and text for upload post
save uploaded material on database
Prompt students that study material has been uploaded

7.1.3.10. Teacher can host exams



7.1.3.11. Teachers can grade copies


Teachers can grade copies
Exam ended?
Yes
No
Copies havent' been checked?
No
yes
Prompt teacher that copies have to be checked
No
Teacher clicked on 'check copies'
Yes
load submitted student answers
No
all copies checked?
Yes
save checked answers
Prompt students its exam copies have been checked

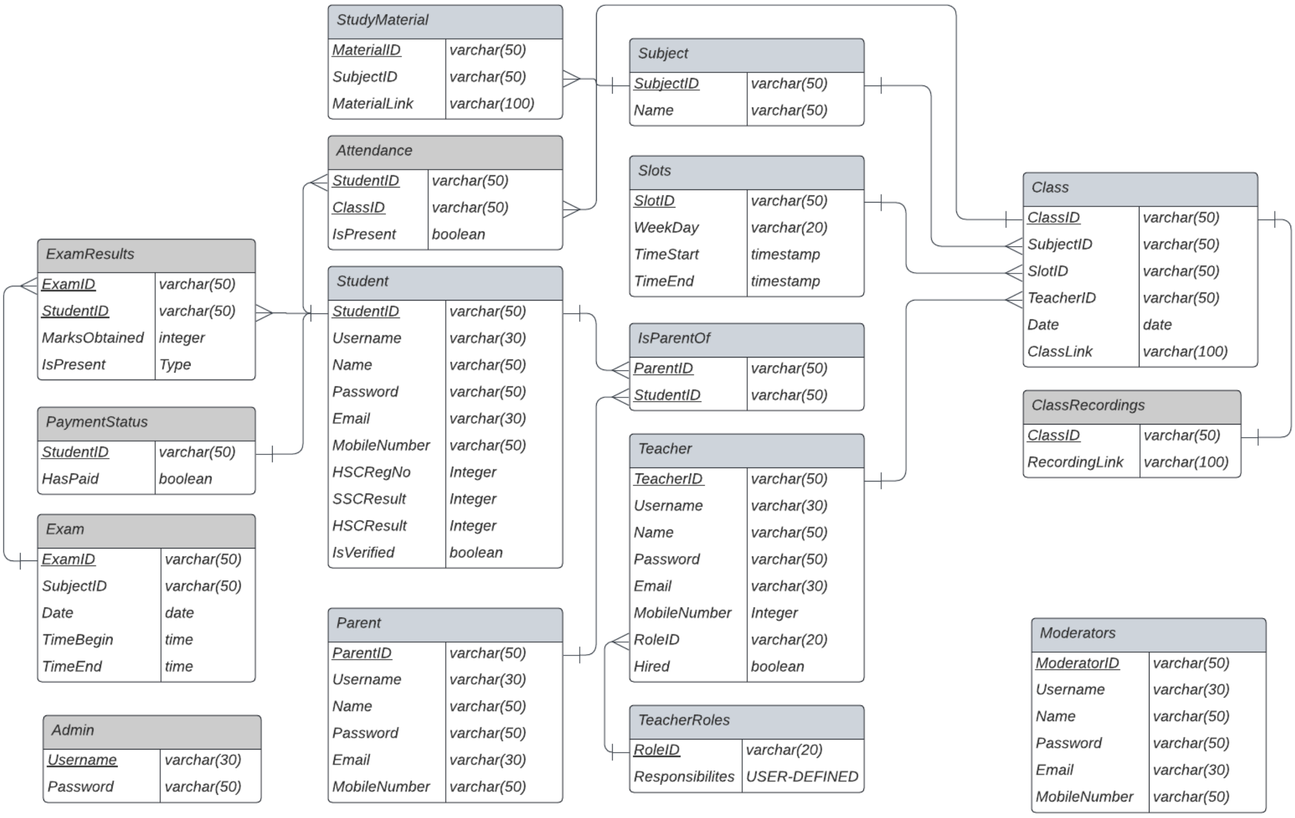
7.2. Data-Based Modeling

7.2.1. Categorizing the nouns into solution/problem space:

S1	Noun	Problem Space/Solution Space	Attribute/Field
1	Student	S	6, 7, 9, 10, 11, 22, 23, 17
2	Teacher	S	6, 7, 9, 10, 11, 12
3	Moderator	S	6, 7, 9, 10, 11
4	Parent	S	6, 7, 9, 10, 11
5	Admin Credentials	S	-
6	Name	S	-
7	Email	S	-
8	Address	S	-
9	Mobile Number	S	-
10	Username	S	-
11	Password	S	-
12	Role	S	-
13	Activity Log	P	-
14	Results	S	1
15	Attendance	S	-
16	Records	S	-
17	Ranking	S	-
18	Class	S	2,
19	Test Schedule	S	-
20	Reports	P	-
21	Privileges	P	-
22	Previous Academic Results	S	-
23	Hsc registration Number	S	-

24	Information	P	-
25	Payment Request	P	-
26	Personal Information	P	-
27	Recorded Classes	S	-
28	Resources	S	27, 29
29	Reference	S	-
30	Marksheets	S	-
31	Teacher Application Page	S	-
32	Valid Credentials	S	-
33	Prospective Teacher	P	-
34	Interviews	P	-
35	Video Conference	S	-
36	Users	S	-
37	System Admin	S	5
38	Payment Partner	P	-
39	System	P	-
40	Online Coaching	P	-
41	Exam	S	42, 19, 31,
42	Subject	S	6

7.2.2. ER diagram:



StudyMaterial	
MaterialID	varchar(50)
SubjectID	varchar(50)
MaterialLink	varchar(100)

Attendance	
StudentID	varchar(50)
ClassID	varchar(50)
IsPresent	boolean

Subject	
SubjectID	varchar(50)
Name	varchar(50)

Slots	
SlotID	varchar(50)
WeekDay	varchar(20)
TimeStart	timestamp
TimeEnd	timestamp

Class	
ClassID	varchar(50)
SubjectID	varchar(50)
SlotID	varchar(50)
TeacherID	varchar(50)
Date	date
ClassLink	varchar(100)

ExamResults	
ExamID	varchar(50)
StudentID	varchar(50)
MarksObtained	integer
IsPresent	Type

Student	
StudentID	varchar(50)
Username	varchar(30)
Name	varchar(50)
Password	varchar(50)
Email	varchar(30)
MobileNumber	varchar(50)
HSCRegNo	Integer
SSCResult	Integer
HSCResult	Integer
IsVerified	boolean

IsParentOf	
ParentID	varchar(50)
StudentID	varchar(50)

ClassRecordings	
ClassID	varchar(50)
RecordingLink	varchar(100)

PaymentStatus	
StudentID	varchar(50)
HasPaid	boolean

Teacher	
TeacherID	varchar(50)
Username	varchar(30)
Name	varchar(50)
Password	varchar(50)
Email	varchar(30)
MobileNumber	Integer
RoleID	varchar(20)
Hired	boolean

Exam	
ExamID	varchar(50)
SubjectID	varchar(50)
Date	date
TimeBegin	time
TimeEnd	time

Parent	
ParentID	varchar(50)
Username	varchar(30)
Name	varchar(50)
Password	varchar(50)
Email	varchar(30)
MobileNumber	varchar(50)

Moderators	
ModeratorID	varchar(50)
Username	varchar(30)
Name	varchar(50)
Password	varchar(50)
Email	varchar(30)
MobileNumber	varchar(50)

Admin	
Username	varchar(30)
Password	varchar(50)

TeacherRoles	
RoleID	varchar(20)
Responsibilites	USER-DEFINED

7.3. Class Based Modeling

7.3.1. Potential class Identification:

S1	Noun	Characteristics Number That Applies	Remarks
1	Student	1,2,3,4,5,6	Accepted
2	Teacher	1,2,3,4,5,6	Accepted
3	Moderator	1,2,3,4,5,6	Accepted
4	Parent	1,3,4,5	Accepted
5	Admin Credentials	1	Rejected
6	Name	1	Rejected
7	Email	1,2	Rejected
8	Address	1	Rejected
9	Mobile Number	1	Rejected
10	Username	1,	Rejected
11	Password	1,	Rejected
12	Role	1,	Rejected
13	Activity Log	1,2,3,4,5	Accepted
14	Results	1,2,3,4,5	Accepted
15	Attendance	1,	Rejected
16	Records	1,	Rejected
17	Ranking	1,2	Rejected
18	Class	1,2,3,4,5	Accepted
19	Test Schedule	1,2,3,4	Accepted
20	Reports	1,2,3	Accepted
21	Privileges	1,	Rejected
22	Previous Academic Results	1	Rejected
23	Hsc registration Number	1,	Rejected

24	Information	1,	Rejected
25	Payment Request	1, 2,3,4,5	Accepted
26	Personal Information	1,4	Rejected
27	Recorded Classes	1,4,	Rejected
28	Resources	1,2,3,4,5,6	Accepted
29	Reference	1,	Rejected
30	Marksheets	1,4	Rejected
31	Ranking List	1,4	Rejected
32	Teacher Application Page	1,5	Rejected
33	Valid Credentials	1,4	Rejected
34	Prospective Teacher	1,2,3,4,5,6	Accepted
35	Interviews	-	Rejected
36	Video Conference	-	Rejected
37	Resource	1,2,3,4,5,	Accepted
38	Users	-	Rejected
39	System Admin	1,2,3,4,5,6	Accepted
40	Moderators	1,2,3,4,5,6	Accepted
41	Payment Partner	1,2,3,4,5,6	Accepted
42	System	1,	Rejected
43	Online Coaching	1,	Rejected

7.3.2. Method Identification:

Verb	**Remark**
Conducting	Yes
Create	Yes
Deployment	Out of scope
Pre-generated	Out of scope

provided	Out of scope
Assign	Yes
Apply	Yes
Approve	Yes
View	Yes
Add	Yes
Delete	Yes
attend	Yes
give	Yes
sign-up	Yes
sign-in	Yes
oversee	Out of scope
designate	Yes
Edit	Yes
Upload	Yes
Redirected	Out of scope
Contact	Out of scope
hire	Out of scope
reject	Yes
ban	Yes
abusing	Out of Scope
host	Yes
check	Yes
update	Yes

Class Name	Attributes	Methods

Student	Name, email, address, Mobile number, Username, Password,	giveExam(),viewResources(),attendClass(),viewTrackRecord(),signUp()
Teacher	Name, email, address, Mobile number, Username, Password, Role	hostClass(),uploadMaterials(),checkCopies().
Parent	Name, email, address, Mobile number, Username, Password	viewTrackRecord()
Admin	Name, email, address, Mobile number, Username, Password	addUser(),approveRequest(),createRoutine(),deleteUser(),editRoutine(),uploadMaterials(),
Moderator	Name, email, address, Mobile number, Username, Password,	addUser(),approveRequest(),createRoutine(),deleteUser(),editRoutine(),uploadMaterials(),
Prospective Teacher	Name, email, address, Mobile number, Username, Password, Wanted Roles	apply()
Class	Class_id,Class_Time	takeAttendance()
Routine	Class,classTime,examTime	viewRoutine()
Exam	Exam_id,Time	saveExamCopies(),viewExamCopies()
Resource	Time,Class,Type	viewResources(),editResource()

Common Methods:SignIn(),SignOut().

7.3.3. Class Card:

Student
Name Email Address Mobile number Username Password,

giveExam()
viewResources()
attendClass()
viewTrackRecord()

Teacher
Name Email Address Mobile_Number UsernamePassword Role

hostClass()
uploadMaterials()
checkCopies()

Parent

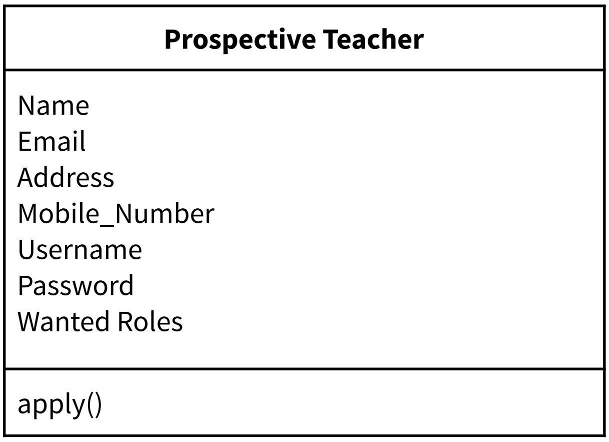
Name Email Address Mobile_Number Username Password
viewTrackRecord()

Admin
Name Email Address Mobile_Number Username Password

addUser()
approveRequest()
createRoutine()
deleteUser()
editRoutine()
uploadMaterials()

Moderator
Name Email Address Mobile_Number Username Password

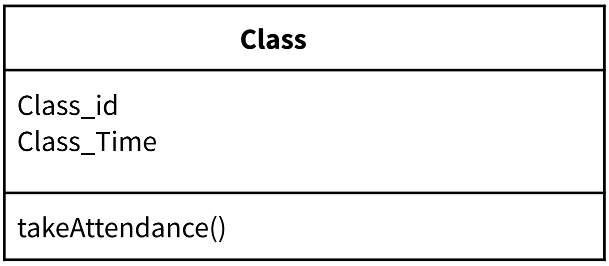
addUser()
approveRequest()
createRoutine()
deleteUser()
editRoutine()
uploadMaterials()



Prospective Teacher

Name
Email
Address
Mobile_Number
Username
Password
Wanted Roles

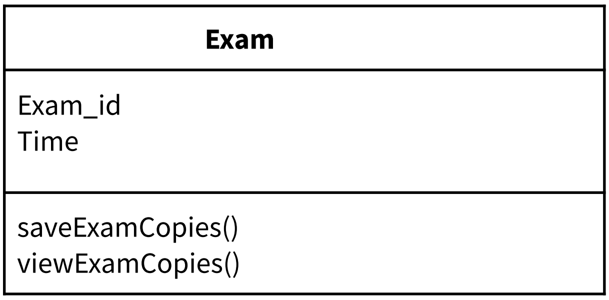
apply()



Class

Class_id
Class_Time

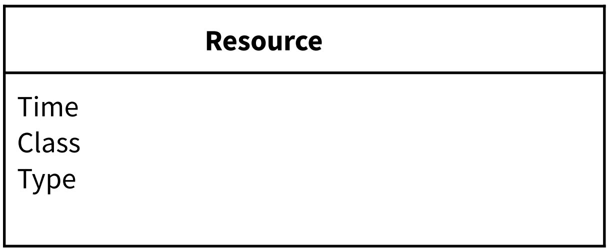
takeAttendance()



Exam

Exam_id
Time

saveExamCopies()
viewExamCopies()



Resource

Time
Class
Type

viewResources()
editResource()

Routine
Class Class_Time Exam_Time
viewRoutine()

7.3.4. CRC:

Student	
Methods	**Collaborator**
giveExam()	Exam
viewResources()	Resource
attendClass()	Class
viewTrackRecord()	Exam, Class

Parent	
Method	**Collaborator**
viewTrackRecord()	Student

Teacher

Methods	**Collaborator**
hostClass()	Class
uploadMaterials()	Resource
checkCopies()	Exam

Admin	
Methods	**Collaborator**
addUser()	Moderator,Student,Parent,Teacher
approveRequest()	Parent,Teacher
createRoutine()	Routine
deleteUser()	Moderator,Student,Parent,Teacher
editRoutine()	Routine
uploadMaterials()	Resource

Moderator	
Methods	**Collaborator**
addUser()	Student,Parent,Teacher
approveRequest()	Parent,Teacher
createRoutine()	Routine
deleteUser()	Student,Parent,Teacher
editRoutine()	Routine
uploadMaterials()	Resource

Prospective Teacher	
Methods	**Collaborator**
apply()	

Class	
Method	**Collaborator**
takeAttendance()	Student

Exam	
Methods	**Collaborator**
saveExamCopies()	Student
viewExamCopies()	

Resource	
Methods	**Collaborator**
viewResources()	
editResource()	Admin,Moderator,Teacher.

Routine	
Method	**Collaborator**
viewRoutine()	

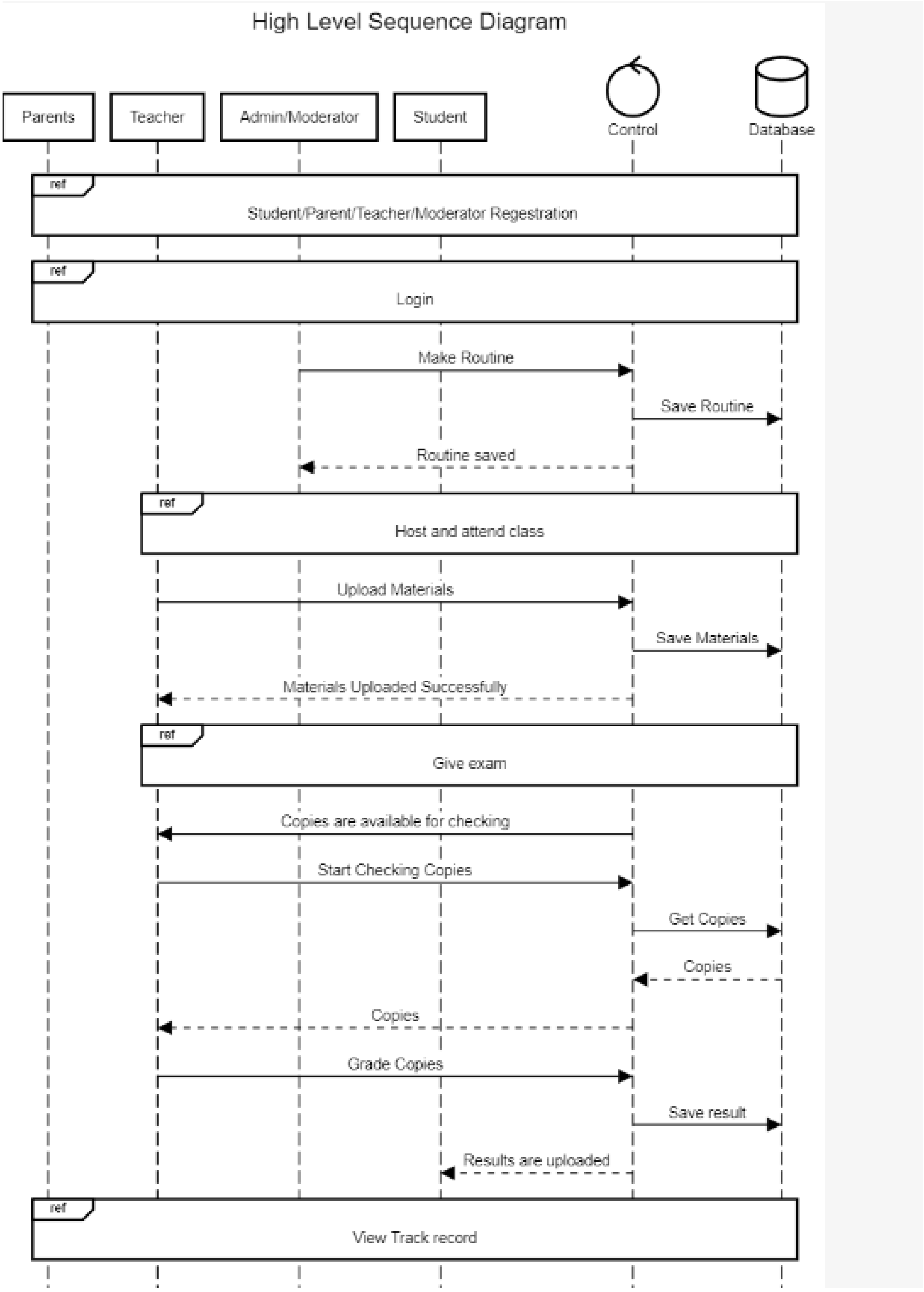
7.4. Behavioral Modeling

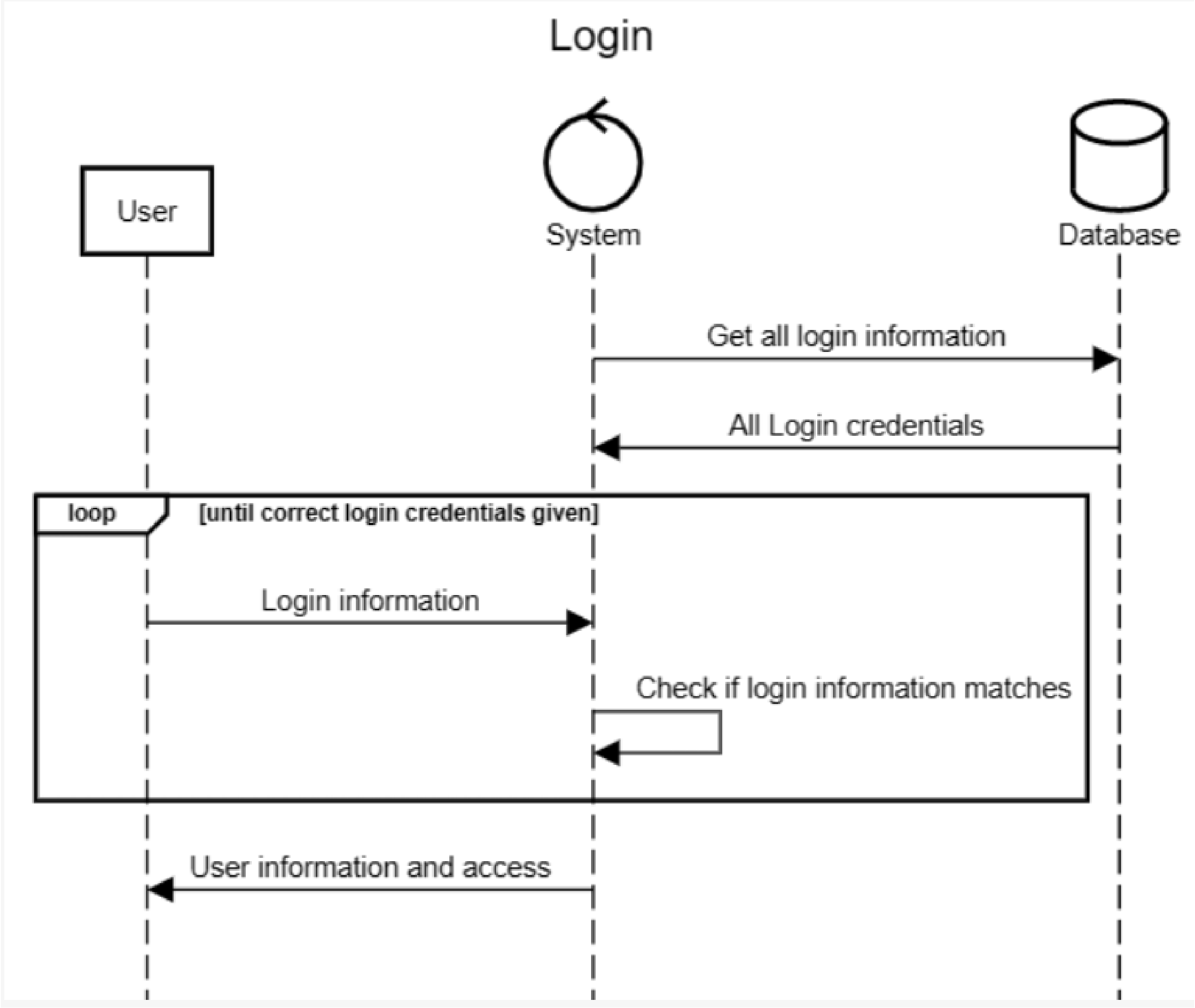
The behavioral model of software reveals its response to external events or stimuli. In this context, it's important to consider two types of state characterizations:

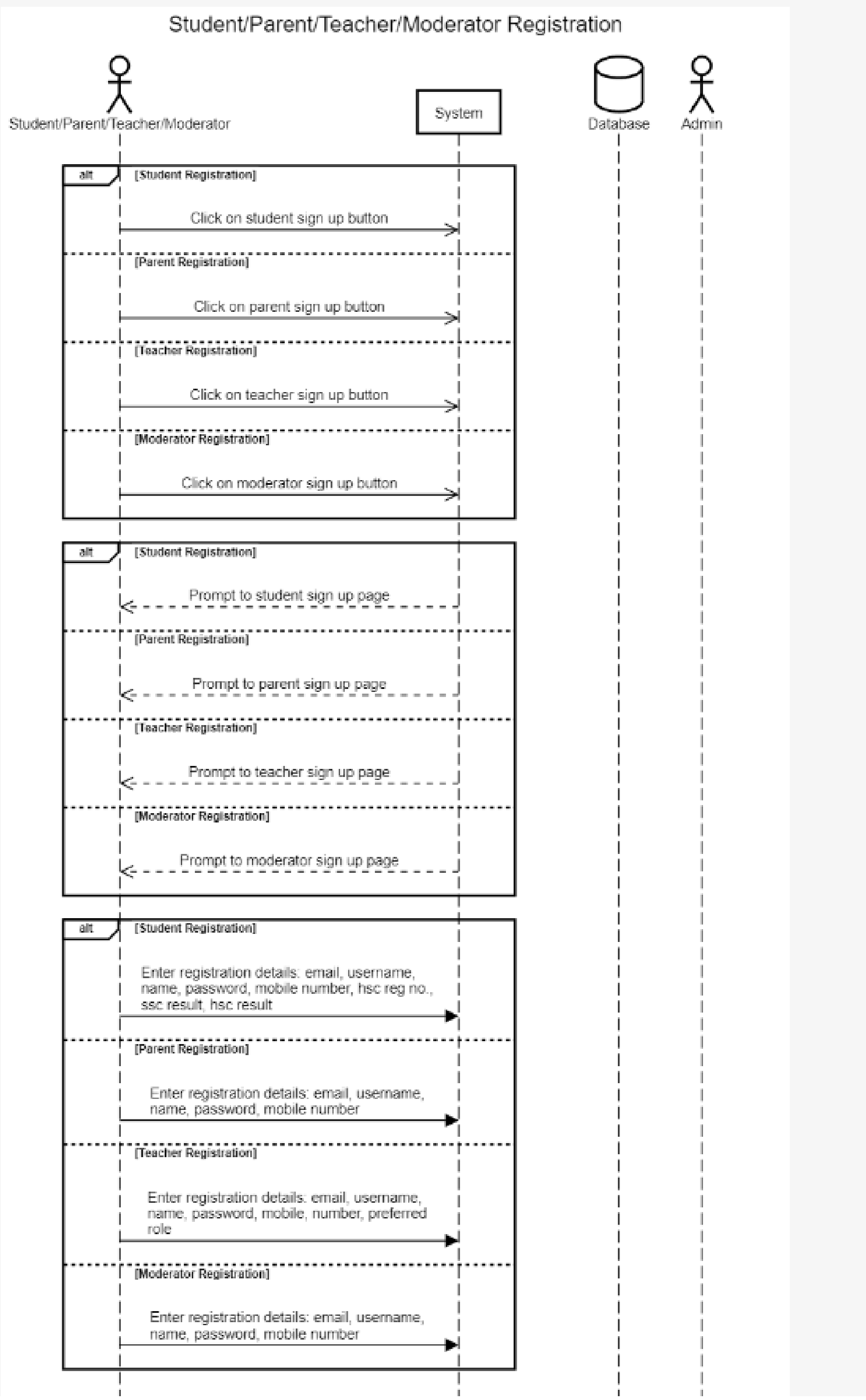
1. the state of each class within the system during its operation and
2. the state of the system as a whole, which can be observed from the outside during its operation.

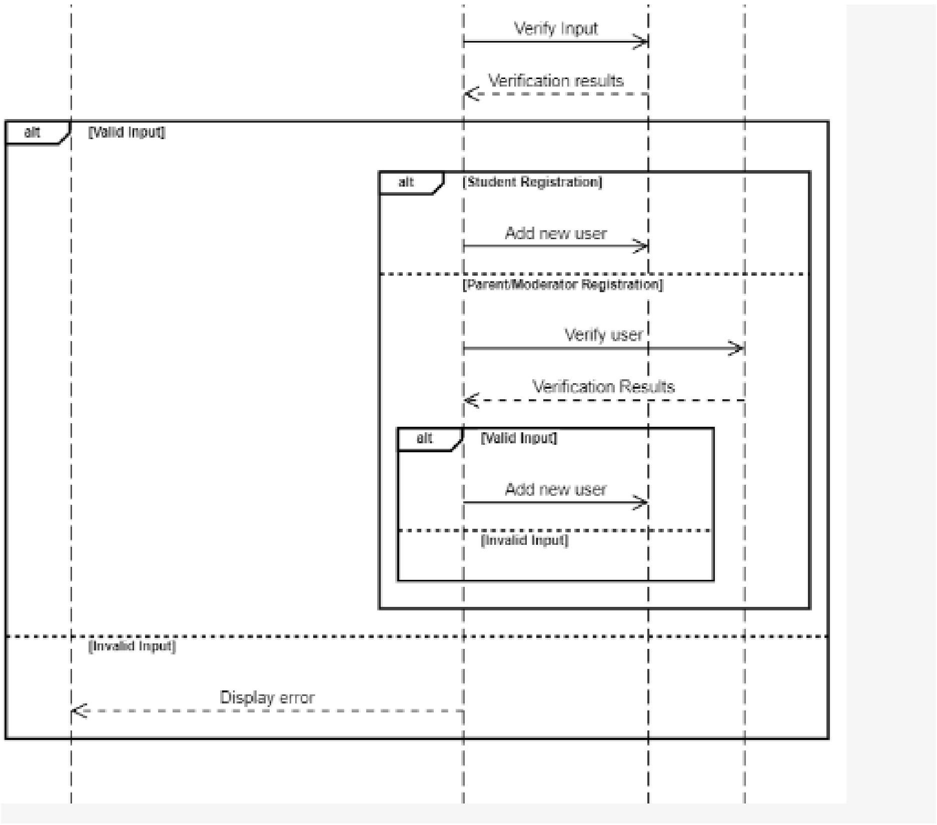
7.4.1. Sequence Diagram

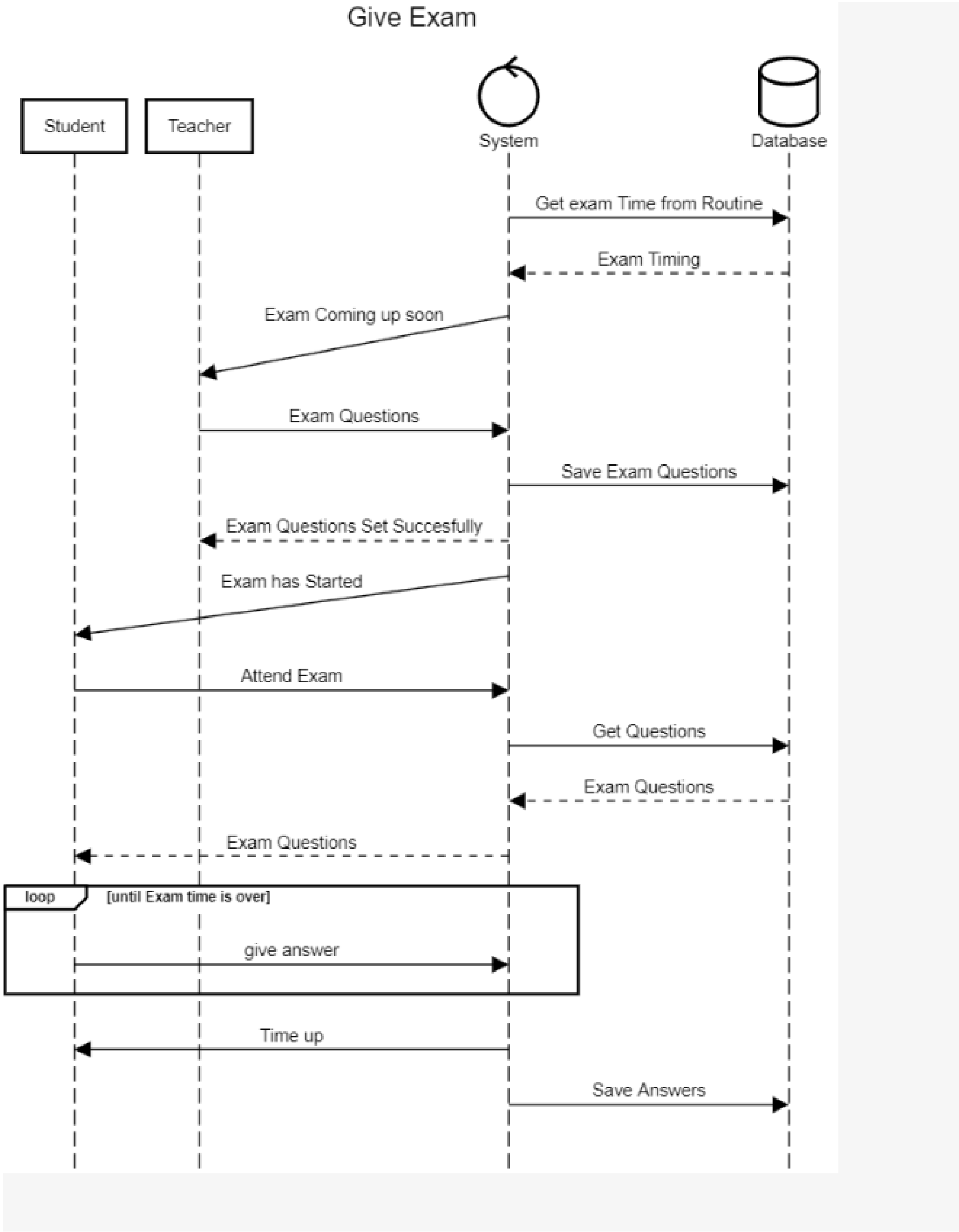
A sequence diagram is a type of interaction diagram that illustrates the interactions and messages exchanged between different objects or actors in a system. It represents the flow of messages and the order in which they are exchanged between the different components of a system over time.

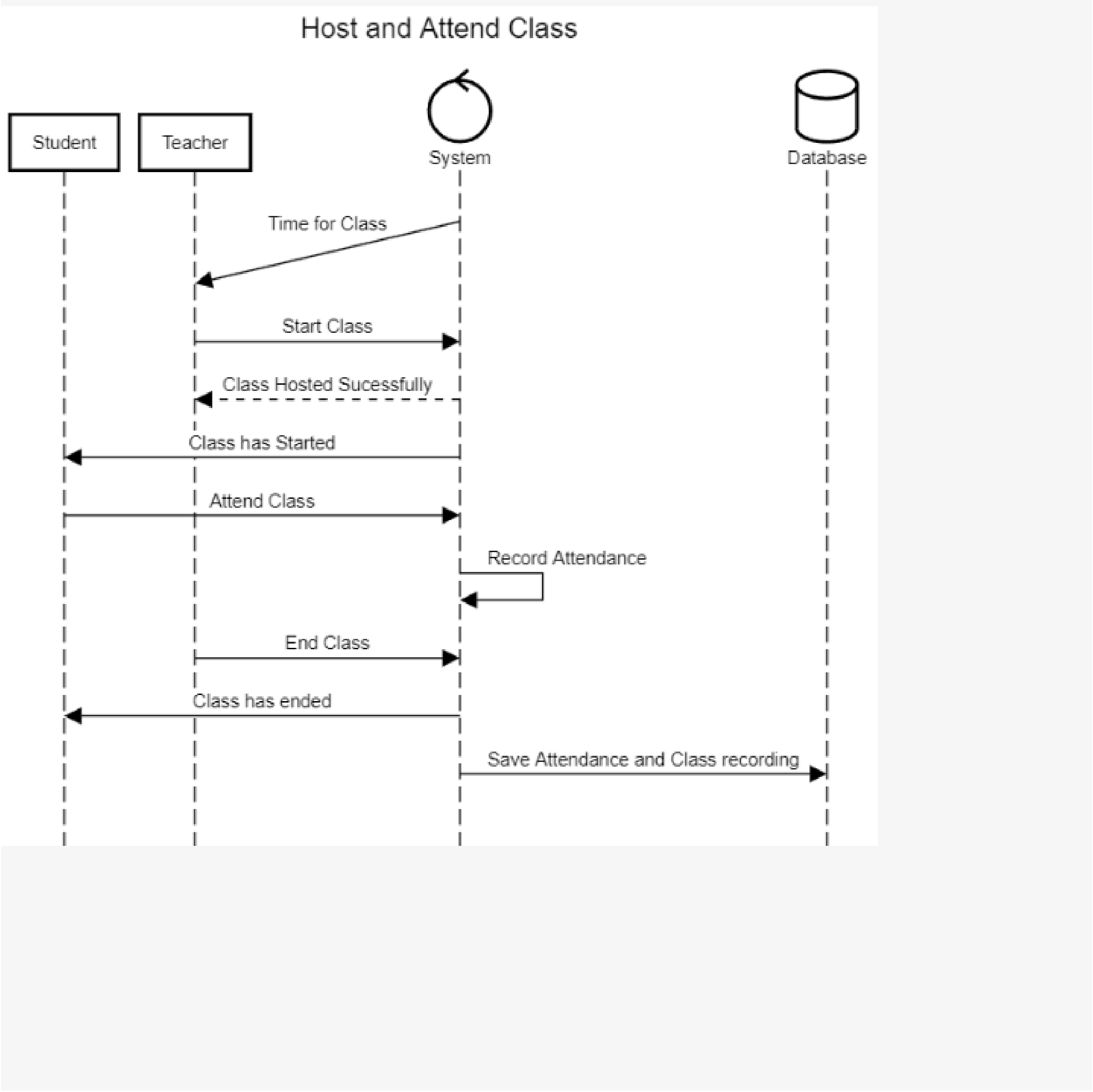

High Level Sequence Diagram
Parents
Teacher
Admin/Moderator
Student
Control
Database
ref
Student/Parent/Teacher/Moderator Regestration
ref
Login
Make Routine
Save Routine
Routine saved
ref
Host and attend class
Upload Materials
Save Materials
Materials Uploaded Successfully
ref
Give exam
Copies are available for checking
Start Checking Copies
Get Copies
Copies
Copies
Grade Copies
Save result
Results are uploaded
ref
View Track record

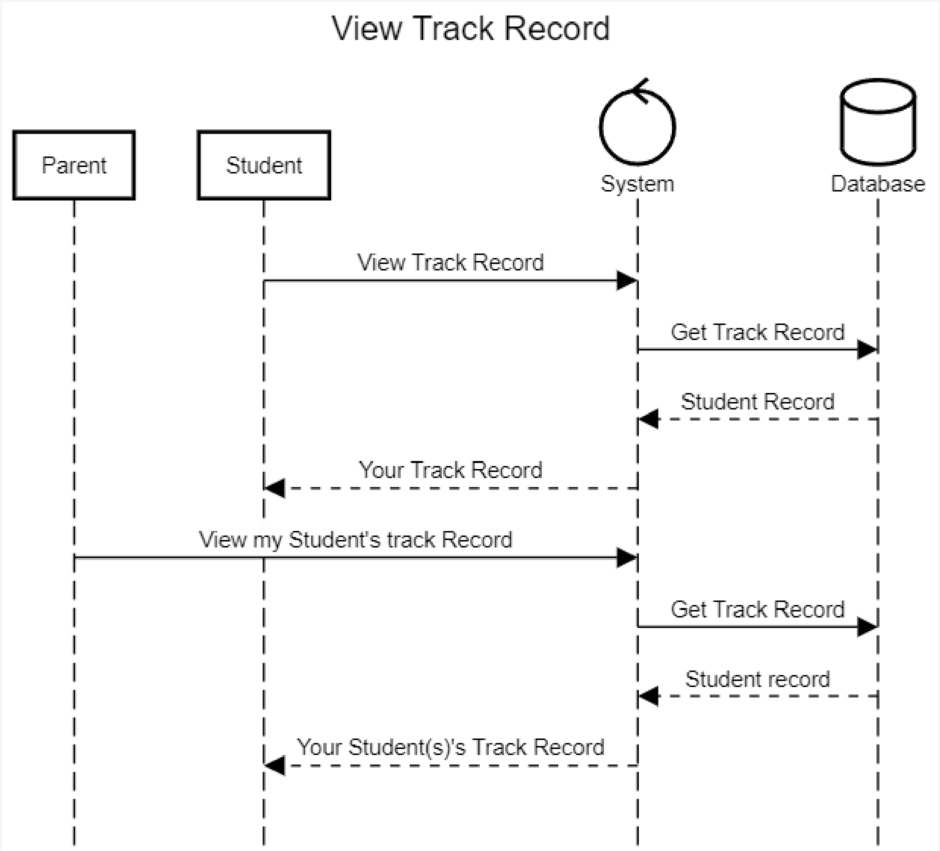

Login
User
System
Database
Get all login information
All Login credentials
loop
[until correct login credentials given]
Login information
Check if login information matches
User information and access


Student/Parent/Teacher/Moderator Registration
Student/Parent/Teacher/Moderator
System
Database
Admin
alt
[Student Registration]
Click on student sign up button
[Parent Registration]
Click on parent sign up button
[Teacher Registration]
Click on teacher sign up button
[Moderator Registration]
Click on moderator sign up button
alt
[Student Registration]
Prompt to student sign up page
[Parent Registration]
Prompt to parent sign up page
[Teacher Registration]
Prompt to teacher sign up page
[Moderator Registration]
Prompt to moderator sign up page
alt
[Student Registration]
Enter registration details: email, username, name, password, mobile number, hsc reg no., ssc result, hsc result
[Parent Registration]
Enter registration details: email, username, name, password, mobile number
[Teacher Registration]
Enter registration details: email, username, name, password, mobile, number, preferred role
[Moderator Registration]
Enter registration details: email, username, name, password, mobile number


Verify Input
Verification results
alt
[Valid Input]
alt
[Student Registration]
Add new user
[Parent/Moderator Registration]
Verify user
Verification Results
alt
[Valid Input]
Add new user
[Invalid Input]
[Invalid Input]
Display error


Give Exam
Student
Teacher
System
Database
Get exam Time from Routine
Exam Timing
Exam Coming up soon
Exam Questions
Save Exam Questions
Exam Questions Set Succesfully
Exam has Started
Attend Exam
Get Questions
Exam Questions
Exam Questions
loop
[until Exam time is over]
give answer
Time up
Save Answers


Host and Attend Class
Student
Teacher
System
Database
Time for Class
Start Class
Class Hosted Sucessfully
Class has Started
Attend Class
Record Attendance
End Class
Class has ended
Save Attendance and Class recording


View Track Record
Parent
Student
System
Database
View Track Record
Get Track Record
Student Record
Your Track Record
View my Student's track Record
Get Track Record
Student record
Your Student(s)'s Track Record

Contribution Table	
Ayesha Afroza Mohsin(200042106)	Introduction, Purpose, Intended Audience, Conclusion, Detailed Use Case (partial), Sequence Diagram (partial), Inception, Activity Diagram, Usage Scenario (partial), Use Case Diagram (partial),
Nawsheen Mehreen(200042134)	Use Case Diagram (partial), Detailed Use Case (partial), Sequence Diagram (partial), Data-based modeling, Usage Scenario (partial), Formatting, Editing, Activity Diagram(partial)
Abrar Mahmud Rahim(200042168)	Elicitation, Quality Function Deployment,Normal Requirements,Expected Requirements,Exciting Requirements, Usage Scenario (partial), Use Case Diagram (partial), Sequence Diagram (partial), Detailed Use Case (partial), Class-Based Modeling.