



**APEX DEVS**  
SOFTWARE DEVELOPERS

## ITSL SOFTWARE PROJECT

2024  
**SOFTWARE  
ENGINEERING  
PROJECT**

CSIS3724  
9 October 2024

## Team Members

2019405725 – Neo Moloi

2019371950 – Letsie Letsie

2020591885 – Tshepiso Poolo

2020205213 – Phila Sophilase

2021332339 – Dominion Oamen

2017108776 – Boikutlo Rampora

2020619552 – Tsholofelo Motlhale

# Table of Contents

<b>Team Members</b>	2
<b>Table of Contents</b>	3
<b>Table of Figures</b>	5
<b>Project Proposal</b>	6
<b>Specifications Document</b>	8
	8
<b>Authentication and login</b>	9
<b>Classroom Chat</b>	13
<b>Boardroom</b>	16
<b>Staffroom</b>	19
<b>Registration</b>	24
<b>Reports and Statistics</b>	27
<b>Calendar</b>	33
<b>Study material access</b>	37
<b>Announcements</b>	41
<b>Study Material Access –Recording and Documentation</b>	47
<b>Assignments/Quiz</b>	50
<b>Testimony and Review</b>	54
<b>Additional Learning Opportunities</b>	58
<b>Upload Promotional and Marketing Material</b>	62
<b>Design and add curriculum content</b>	64
<b>Attendance</b>	67
<b>Software Project Management Plan</b>	71
<b>Quotation</b>	77
<b>Sequence Diagrams</b>	78
<b>Project Coordinator</b>	78
<b>Authentication</b>	78
<b>SL Student</b>	92
<b>Community Partner Organizations</b>	98
<b>Community Participant</b>	103
<b>Noun Extractions</b>	107
<b>CRC Cards</b>	108
<b>Class Diagrams</b>	112

<b><i>State Chart</i></b>	<b>113</b>
<b><i>Pseudo code</i></b>	<b>114</b>
<b><i>Record of Hours Spent</i></b>	<b>122</b>
<b><i>Overview of Development Effort</i></b>	<b>122</b>

# Table of Figures

<i>Figure 1.1: Login UML</i>	9
<i>Figure 1.2: Login UI</i>	9
<i>Figure 2.1: Classroom chat UML</i>	13
<i>Figure 2.2.1: Project Organiser and CSIS Student UI</i>	13
<i>Figure 3.1: Boardroom UML</i>	17
<i>Figure 3.2: Boardroom UI</i>	17
<i>Figure 4.1: Staffroom UML</i>	20
<i>Figure 4.2: Staffroom UI</i>	20
<i>Figure 5.1: Registration UML</i>	25
<i>Figure 5.2.1: Registration UI</i>	25
<i>Figure 5.2.2: Registration UI</i>	26
<i>Figure 6.1: Reports UML</i>	28
<i>Figure 6.2.1: Reports and Performance tracking UI</i>	29
<i>Figure 6.2.2: Reports and Performance tracking UI</i>	29
<i>Figure 7.1: Calendar UML</i>	33
<i>Figure 7.2.1: Calendar UI</i>	34
<i>Figure 8.1: Study material access UML</i>	37
<i>Figure 8.2: Study material UI</i>	38
<i>Figure 9.1: Announcements UI</i>	41
<i>Figure 9.2.1: Announcements UI</i>	41
<i>Figure 9.2.2: Announcements UI</i>	42
<i>Figure 10.1: Staffroom UI</i>	47
<i>Figure 10.2.1: Study Material access UI</i>	47
<i>Figure 10.2.2: Study Material access UI</i>	48
<i>Figure 11.1: Assignments/Quiz UML</i>	50
<i>Figure 11.2: Assignments/Quiz UI</i>	51
<i>Figure 12.1: Testimony and Review UML</i>	54
<i>Figure 12.2: Testimony and Review UI</i>	55
<i>Figure 13.1: Additional Learning Opportunities UML</i>	58
<i>Figure 13.2: Additional Learning Opportunities UI</i>	58
<i>Figure 14.1: Testimony and Review UML</i>	61
<i>Figure 14.2: Testimony and Review UI</i>	62
<i>Figure 15.1: Design and add curriculum UML</i>	65
<i>Figure 15.2: Design and add curriculum UI</i>	65
<i>Figure 16.1: Attendance UML</i>	68
<i>Figure 16.2: Attendance UI</i>	68

# Project Proposal

## Initial Specifications

This proposal outlines the development of a comprehensive mobile or web-based application designed to manage the ITSL program. The application will feature a user-friendly interface, robust backend functionality, and a secure database. It will serve various stakeholders, including project organizers, community partner organizations, and service-learning students, and will incorporate essential functionalities for both management and educational purposes. Key features will include:

- **User Registration:** Streamlined onboarding process for all users.
- **Login & Authentication:** Secure access control to protect user data.
- **Attendance Tracking:** Efficient management of participant attendance.
- **Assessments:** Tools for evaluating student performance.
- **Course Material Access:** Centralized repository for educational resources.
- **Progress Reports:** Comprehensive tracking of student and program progress.

## Final Specifications

### Client Feedback

Following a detailed review of the initial specifications, the project coordinator (the client) provided additional insights and requirements. It is imperative that the application comprehensively addresses all program needs without relying on third-party applications. The design must ensure that all functionalities are delivered seamlessly within the application itself.

Recognizing that community participants have limited access to data, it is crucial to integrate features that facilitate offline access. Additionally, the application must automate various tasks currently managed manually by the project organizer, enhancing efficiency.

To foster effective communication among users, the application should include robust communication features. Considering this feedback, the following additional features and functionalities will be incorporated:

### Appendix: Explanation of Additional Changes in Final Specifications

This appendix provides a brief overview of the additional features and functionalities incorporated into the final specifications based on client feedback.

#### 1. Student and Participant Management

This feature offers comprehensive tools for registering and managing both students and community participants. It streamlines the onboarding process and enables project coordinator to efficiently track participant details, ensuring organized management of all individuals involved in the ITSL program.

#### 2. Boardroom

The Boardroom feature serves as a dedicated virtual space for administrative discussions and decision-making. It facilitates collaboration among project coordinator and community partners, allowing them to hold meetings, share documents, and make strategic decisions in an efficient manner.

### **3. Staffroom**

The Staffroom provides a virtual area for staff members to collaborate and share resources. This feature fosters teamwork and communication among service-learning students and project coordinator, enabling them to exchange information and support each other in their roles.

### **4. Classroom Chat**

Classroom Chat enables instant messaging functionality, allowing real-time communication between participants and instructors. This feature enhances engagement and supports collaborative learning by providing a platform for questions, discussions, and immediate feedback.

### **5. Lecture Recordings**

The Lecture Recordings feature offers a repository for storing and accessing recorded lectures. This allows students to review course content at their convenience, reinforcing learning and accommodating various learning styles.

### **6. Study Materials**

A centralized hub for educational resources and materials, this feature ensures that all study materials are easily accessible to participants. It provides a structured environment for students to find relevant content necessary for their coursework and development.

### **7. Additional Learning Opportunities**

This feature includes resources and links to further learning experiences outside the core curriculum. It encourages continuous learning and professional development by connecting participants to workshops, webinars, and other educational events.

### **8. Announcements**

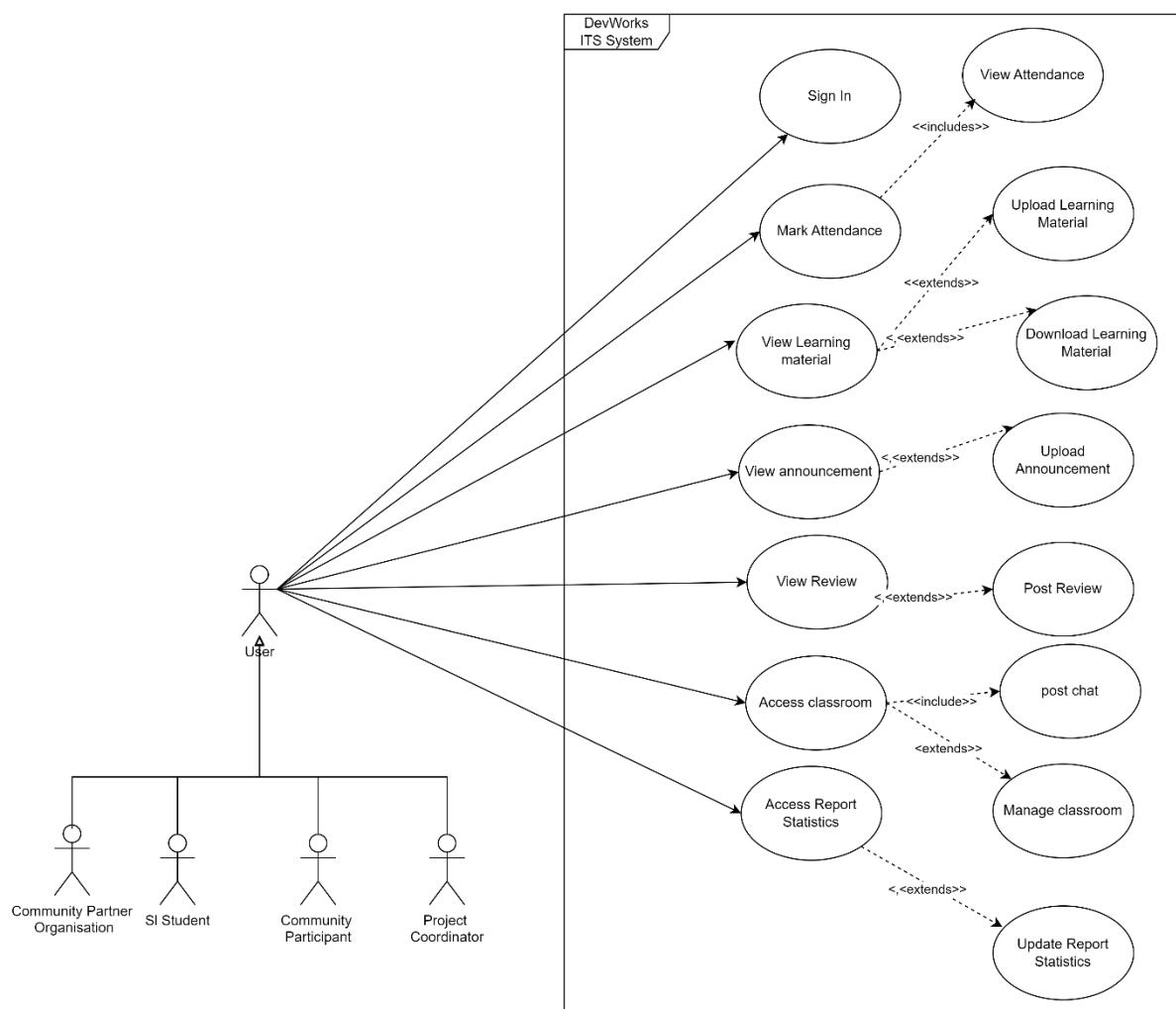
The Announcements feature allows the project coordinator to distribute important updates and information to all users within the application. This ensures that participants are kept informed about program developments, deadlines, and other relevant news.

### **9. Testimonies and Reviews**

This section enables users to share feedback, and experiences related to the ITSL program. By showcasing testimonies and reviews, this feature enhances community engagement and provides valuable insights for both current and prospective participants.

This enhanced proposal reflects a comprehensive understanding of the project requirements and aims to create an effective platform that meets the needs of all stakeholders involved in the ITSL program.

# Specifications Document



# Authentication and login

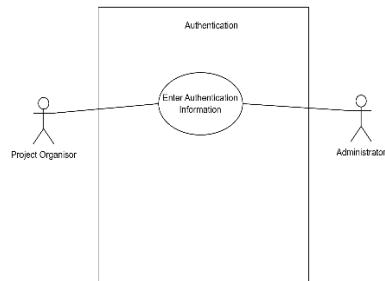


Figure 1.1: Login UML

## Brief Description:

The Authentication use case enables the project coordinator and system administration to login to the application/website using their credentials such as email and password

## User Interface:



Figure 2.2: Login UI

## step by step:

1. The user opens the application by clicking on the ITCL application icon in their device.
2. The applications welcome screen will then appear, prompting the user to enter their user account information to login.
3. The user will click on the email textbox to enter their email, then click on the password textbox to enter their password.
4. The application will verify the user's information against its database to ensure the information is valid.
5. If the entered information is valid, the app will proceed to log the student inside the application. However, if the information is invalid, the application will prompt the user to re-enter their details again or to click the register button to register their information.

### **Best-case Scenario**

1. Jonathon opens the application by clicking on the ITCL application icon in their device.
2. The applications welcome screen will then appear, prompting the Johnathon to enter their user account information to login.
3. Johnathon will click on the email textbox to enter their email, then click on the password textbox to enter their password.
4. The application will verify the Johnathon information against its database to ensure the information is valid.
5. If the entered information is valid, the app will proceed to log the student inside the application.

### **Worst-case Scenario**

1. Jonathon opens the application by clicking on the ITCL application icon in their device.
2. The applications welcome screen will then appear, prompting the Johnathon to enter their user account information to login.
3. The user will click on the email textbox to enter their email, then click on the password textbox to enter their password.

4. The application will verify the Johnathon's information against its database to ensure the information is valid.
5. If the entered information is valid, the app will proceed to log the Johnathon into the application.
6. However, if the information is invalid, the application will prompt the user to re-enter their details again.
7. After Johnathon re-enters his credentials if the credentials have been entered correctly.
8. The system will either proceed to log the user into the system.
9. The process of step 6 to 8 can be repeated 5 times.
10. If Johnathon's credentials are invalid after 5 attempts the system will log him out for 5 minutes before he can try again.
11. Johnathon enters his credentials in the textboxes for an additional 5 times and is unsuccessful each time.
12. The system generates a pop-up message box that allows the user to write a message to the admin. To verify the user's credentials in the database.

#### **Alternative Scenario 1:**

1. Jonathon opens the application by clicking on the ITCL application icon in their device.
2. The applications welcome screen will then appear, prompting the Johnathon to enter their user account information to login.
3. The user will click on the email textbox to enter their email, then click on the password textbox to enter their password.
4. The application will verify the Johnathon's information against its database to ensure the information is valid.
5. If the entered information is valid, the app will proceed to log the Johnathon into the application.
6. However, if the information is invalid, the application will prompt the user to re-enter their details again.

7. After Johnathon re-enters his credentials if the credentials have been entered correctly.
8. The system will either proceed to log the user into the system.
9. The process of step 6 to 8 can be repeated 5 times.
10. If Johnathon's credentials are invalid after 5 attempts the system will log him out for 5 minutes before he can try again

#### **Alternative Scenario 2:**

1. Mantwa Lebelo accesses the system by signing in, with her credentials. Entering her password and usernames in the provided spaces.
2. After entering all her details, Mantwa selects the sign in button.
3. The system authenticates and respond with an error message, stating that the password or username is incorrect.
4. Mantwa tries to enter the details again and the error persists. She decides to click on the forgot password link. Which opens a new page and request Mantwa's email to send recovery prompts.
5. Mantwa provides her email and click the submit button at the bottom of the page. The system sends Mantwa a reset password links via. Allowing Mantwa to creates a new password. Once she is done resetting. The log in page loads.
6. She enters her credentials in the provided fields. The system authenticates and validates her credentials and signs her in.
7. The systems load and lands Mantwa on the home dashboard.

# Classroom Chat

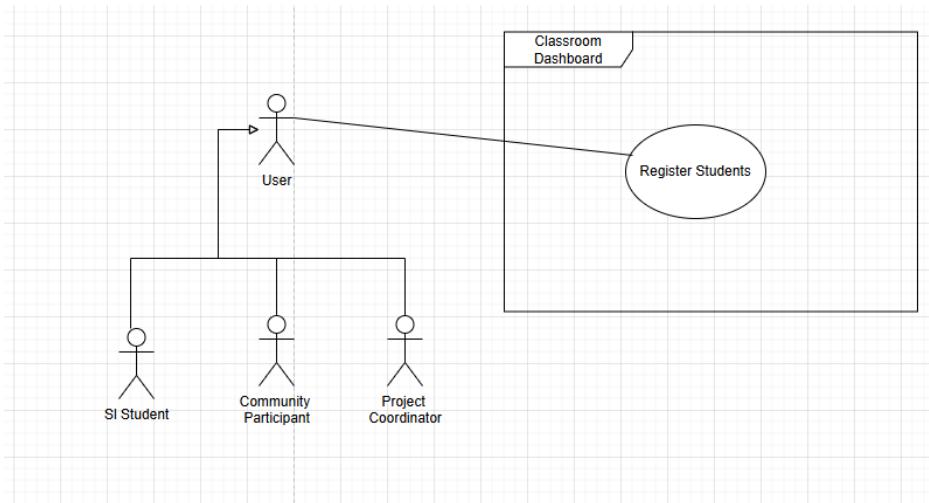


Figure 2.1: Classroom chat UML

## Brief Description:

The Classroom Chat use case allows the students, learners and administrators to communicate with one another without disrupting the live face to face lecture

## User Interface

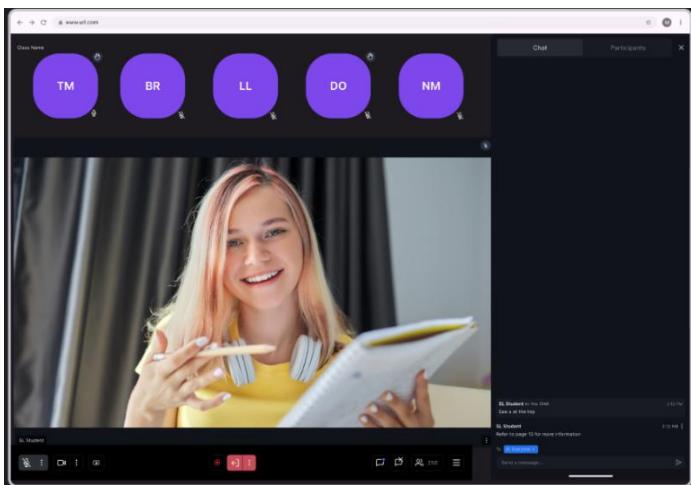


Figure 2.2.1: Project Organiser and CSIS Student UI

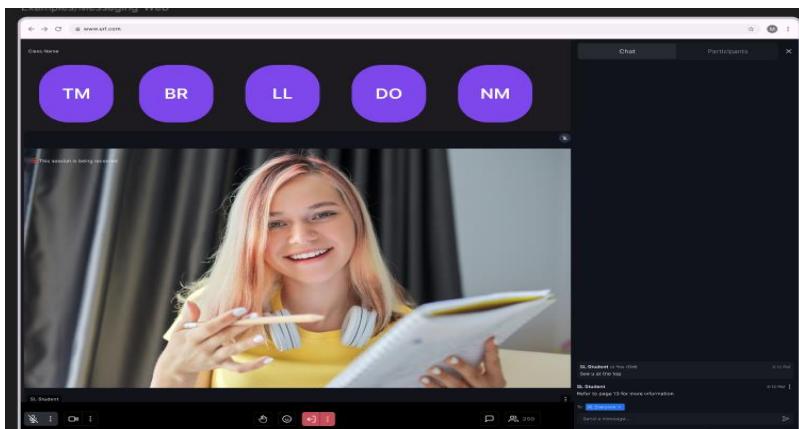


Figure 2.2.2: Community participant UI

#### Step by step:

1. After successful Login (with correct credentials)
  - a. Hover to the top left area the (Tab area) and click on the Classroom interface/dashboard and then click on the Chat Hub functionality.
2. Once user has entered Classroom Chat Hub, they enter the subject of the meeting in the textbox provided along with other information that will be required.

#### Project Coordinator and SL Student:

(During face-to-face classroom circumstances one individual will present while the other moderates the chatroom)

- 3.1 Once user has entered Classroom Chat Hub, they enter the subject of the meeting in the textbox provided along with other information that will be required.
- 3.2 Then proceed to click on the start session button.
- 3.3 Once inside the virtual meeting there is a chat function on the bottom right corner where in which the user will be able to type in conversations as an alternative to speaking out loud during a face-to-face classroom environment.
- 3.4 On the bottom left corner, the administrator has the right to
  - 3.4.1 Switch their mic on/off
  - 3.4.2 Record meeting
  - 3.4.3 Share the video with the other individuals in the virtual meeting.
- 3.5 The participants of the meeting are illustrated at the top of the screen, this is where the participants of the virtual environment can see who has their mic on and who is online.

#### Community Participant:

- 4.1 The user only has the option of joining the already created session.
- 4.2 The application will display the chat screen with a layout of all the messages that have been typed by other users as well as a tab on the bottom of the chat screen for typing and posting messages to the chat.
- 4.3 The Community Participant must then scroll to view and read other users' messages or click the bottom tab to type a message.
- 4.4 After typing their message, the student must press enter on their keyboard to post their message or click the paper airplane button located on the right-hand side of the message tab to send/post their message.

### **Best-case Scenario: Project Organiser**

1. After successful Login (with correct credentials)
2. Thulani hovers over to the top left area the (Tab area) and clicks on the “Classroom” interface/dashboard and then clicks on the Chat Hub functionality.
3. Thulani has entered Classroom Chat Hub, he enters the subject of the meeting in the textbox provided along with other information that will be required.
4. Thulani then proceeds to click on the start session button.
5. Once inside the virtual meeting there is a chat function on the bottom right corner where in which the user will be able to type in conversations as an alternative to speaking out loud during a face-to-face classroom environment.
6. On the bottom left corner, the Thulani has the right to
  - 6.1 Switch their mic on/off.
  - 6.2 Record meeting.
  - 6.3 Share the video with the other individuals in the virtual meeting.

### **Alternative Scenario: Community participant**

1. After successful Login (with correct credentials)
  - a. Thabo hovers over to the top left area the (Tab area) and clicks on the “Classroom” interface/dashboard and then clicks on the Chat Hub functionality.
2. Thabo has entered Classroom Chat-Hub; he enters the subject of the meeting in the textbox provided along with other information that will be required.
3. Thabo then proceeds to click on enter session button.
4. Once inside the virtual meeting there is a chat function on the bottom right corner where in which the user will be able to type in conversations as an alternative to speaking out loud during a face-to-face classroom environment.

5. Thabo clicks on the open mic button icon.
6. The system does not allow him use his mic.
7. Thabo will need to troubleshoot his device speaker and change his settings or type in a request to the individual who created the session to activate allow him to use his speaker.

#### **Alternative scenario 2: Community Participant:**

- 1 After Jane has successfully entered her credentials.
- 2 She focuses her cursor to the navigation on the left.
- 3 She clicks on the classroom button.
- 4 This directs her to click on the button for the open session and signs her in.
- 5 Jane has question and the tutor instruct her to unmute her mic and speak up.
- 6 Jane unmutes her mic by clicking on the red mic button at the bottom left of the classroom screen.
- 7 The button turns green and blinks as Jane speaks.
- 8 Once she is done, Jane clicks the mic icon again to mute herself.
- 9 Jane clicks at the icon with the door to leave the session.

#### **Worst case: Community participant**

- 1 Jane focuses her cursor to the navigation on the left.
- 2 She clicks on the classroom button. This directs her to an open session and signs her in. In the classroom there is chat to the right of the presenter's space.
- 3 Jane misses something when the presenter is speaking and attempts to type the question in the chat box. Once she finishes typing her question.
- 4 She tries to press the right arrow which then send her question.
- 5 The button is greyed out and she cannot send her comment.
- 6 She tries to unmute her mic, but it is also greyed out.
- 7 She cannot use her camera as the computers in the lab do not have camera.
- 8 She leaves the session with her questions unanswered.

## Boardroom

#### **UML:**

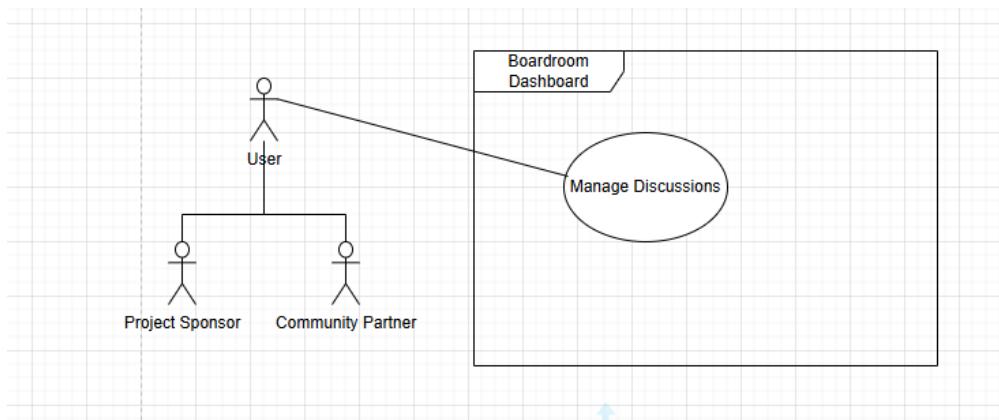


Figure 3.1: Boardroom UML

### Brief Description:

The Board use case allows the project coordinators and administrators to conduct a collaborative online meeting amongst themselves. This allows these individuals to have boardroom like interactions while located in different areas.

### User Interface:

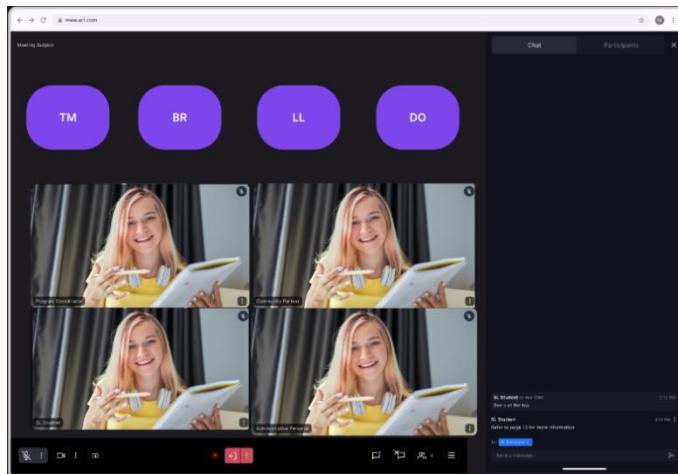


Figure 3.2: Boardroom UI

### Step by step:

1. After successful login, the user navigates the top left corner and clicks on the Boardroom tab.
2. The user clicks on the “Boardroom” tab.
3. User can click on the mic to speak.

4. Click on the chat to type their comments.

#### **Project Coordinator:**

5. Functionality they would enter the details of the meeting.
6. Once the details have been entered the user clicks on the ‘start meeting’ button and proceeds to the virtual environment
7. The users of the system will be able to share their screen with the rest of the participants of the virtual meeting.
8. The user can record the meeting by clicking on the record icon on the bottom left corner of the screen. The meeting will then be recorded and saved to the systems database.

#### **Community partner:**

9. Click on the start meeting button.
10. Add a response through a thumbs up for yes and thumbs down for no.

#### **Best Scenario**

1. After successful login, the Thando navigates the top left corner and clicks on the Boardroom tab.
2. Thando of the system will be able to share their screen with the rest of the participants of the virtual meeting.
3. Share their camera and audio with the rest of the participants.
4. Once Thando clicks on the “Boardroom” functionality they would enter the details of the meeting.
5. Once the details have been entered the user clicks on the ‘start meeting’ button and proceeds to the virtual environment
6. Thando can record the meeting by clicking on the record icon on the bottom left corner of the screen. The meeting will then be recorded and saved to the systems database.

#### **Worst Case Scenario**

1. Mr. Dlamini wants to be a part of a boardroom discussion with other community partner organizations and project coordinator, Mr. Dlamini must first login successfully on to the system.
2. Mr. Dlamini must click on boardroom icon to open up boardroom page.

3. When boardroom page is displayed there is no link available to join the boardroom discussion.
4. Mr. Dlamini cannot join the boardroom discussion.
5. The system Produces a message box “Browser is not compatible”.

### **Alternative Case Scenario 1**

1. Mr. Dlamini joins boardroom discussions after successfully login to the system.
2. Mr. Dlamini clicks on the boardroom icon to join boardroom discussion.
3. When Mr. Dlamini is in the meeting, he cannot hear other community partner organizations and project coordinator communicating.
4. Mr. Dlamini proceeds to click on the sound icon.
5. The system troubleshoots the computers sound system and produces a message box that has diagnosed the problem and a solution on how to rectify the problem.

### **Alternative Case Scenario 2**

1. After successful login, the Thando navigates the top left corner and clicks on the Boardroom tab.
2. Thando of the system will be able to share their screen with the rest of the participants of the virtual meeting.
3. Once Thando clicks on the “Boardroom” functionality they would enter the details of the meeting.
4. Once the details have been entered the user clicks on the ‘start meeting’ button and proceeds to the virtual environment
5. Thando can record the meeting by clicking on the record icon on the bottom left corner of the screen. The meeting will then be recorded and saved to the systems database.
6. Thando shares her camera and audio with the rest of the participants.
7. The system won’t share Thando’s screen with the rest of the participants.
8. The system will then provide Thando with multiple troubleshooting options including texting her mic and changing her screen sharing settings.

## **Staffroom**

**UML:**

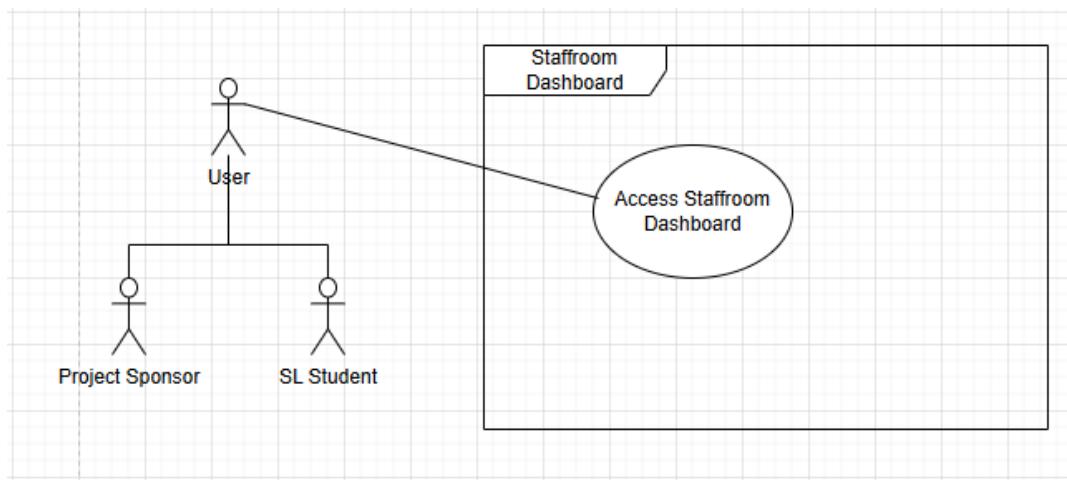


Figure 4.1: Staffroom UML

### Brief Description:

The staffroom use case has a functionality that allows the Project coordinator to have the final say regarding all assessments and course material. This allows them to then edit and finally upload these documents to the system.

### User Interface

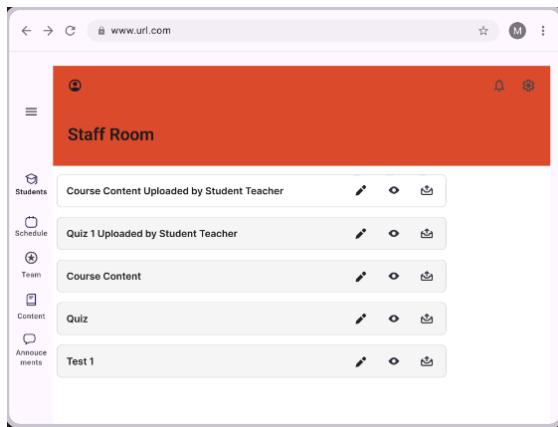


Figure 4.2: Staffroom UI

### Step by step:

1. After successfully logging in the user will then be directed to the home dashboard.

2. This is where the user hovers to the top left-hand side menu to find and click on the “Staff Room” tab.
3. Once inside Staff Room the system will allow the user access according to whether they are a Project Coordinator or SL Student.
4. This will determine whether they have the Upload Course Material, Manage course Material option.
5. The user has the option of creating and posting the course material, quizzes and tests.
6. The user has the option of clicking on the upload document button allowing them to post a file to the system.
7. These files can be created by clicking on the “Create Assessment” inside the “Upload Course Material” tab.
8. This is when a template is provided by the system allowing the user to populate the template with information about the test and its content.
9. Once completed the user clicks on the next button at the bottom of the screen where they populate the textboxes with information about the duration, time the test will be taken and the name of the assessment.
10. Finally, the user will click on the submit button.

**SL Student:**

11. After submission the file is sent to the Project Coordinator for authentication.

**Project Coordinator:**

12. After submission the file is posted onto the system with no need for authentication by a third party.

**Manage course Material:**

13. This is where all submissions by the SL Student are sent.
14. The user has three options which allows them to view the file
15. Edit the file and its content
16. And then upload the file onto the system.

**Best-case Scenario**

1. After successfully logging in Katleho will then be directed to the home dashboard.
2. This is where Katleho hovers to the top left-hand side menu to find and click on the “Staff Room” tab.

3. Once inside Staff Room the system will allow Katleho access the “Staffroom” dashboard
4. This will determine whether they have the Upload Course Material, Manage course Material option.
5. The Katleho has the option of creating and posting the course material, quizzes and tests.
6. The Katleho has the option of clicking on the upload document button allowing them to post a file to the system.
7. These files can be created by clicking on the “Create Assessment” inside the “Upload Course Material” tab.
8. This is when a template is provided by the system, allowing the Katleho to populate the template with information about the test and its content.
9. Once completed Katleho clicks on the next button at the bottom of the screen where they populate the textboxes with information about the duration, time the test will be taken and the name of the assessment.
10. Finally, Katleho will click on the submit button.

### **Worst case**

1. After successfully logging in Tshepo will then be directed to the home dashboard.
2. This is where Tshepo hovers to the top left-hand side menu to find and click on the “Staff Room” tab.
3. Once inside Staff Room the system will allow Tshepo access the “Staffroom” dashboard
4. This will determine whether they have the Upload Course Material, Manage course Material option.
5. Then Tshepo has the option of creating and posting the course material, quizzes and tests.
6. The Tshepo has the option of clicking on the upload document button allowing them to post a file to the system.
7. These files can be created by clicking on the “Create Assessment” inside the “Upload Course Material” tab.
8. This is when a template is provided by the system, allowing the Tshepo to populate the template with information about the test and its content.

9. Once completed Tshepo clicks on the next button at the bottom of the screen where they populate the textboxes with information about the duration, time the test will be taken and the name of the assessment.
10. Tshepo submits but realises that he has submitted the wrong assessment and has then already sent the assessment to the project organiser for approval.

### **Alternative Scenario 1**

1. After successfully logging in Tshepo will then be directed to the home dashboard.
2. This is where Tshepo hovers to the top left-hand side menu to find and click on the “Staff Room” tab.
3. Once inside Staff Room the system will allow Tshepo access the “Staffroom” dashboard
4. This will determine whether they have the Upload Course Material, Manage course Material option.
5. Then Tshepo has the option of creating and posting the course material, quizzes and tests.
6. The Tshepo has the option of clicking on the upload document button allowing them to post a file to the system.
7. These files can be created by clicking on the “Create Assessment” inside the “Upload Course Material” tab.
8. This is when a template is provided by the system, allowing the Tshepo to populate the template with information about the test and its content.
9. Once completed Tshepo clicks on the next button at the bottom of the screen where they populate the textboxes with information about the duration, time the test will be taken and the name of the assessment.
10. Finally, Tshepo clicks on the submit button.
11. After submission the file is sent to the Katleho (Project Coordinator) for authentication.
12. This is where all submissions by the SL Student are sent.
13. The Katleho views
14. Edit the file and its content
15. The system opens the file template
16. Katleho can make changes to the quiz or test by clicking on and changing the labels, moving and or deleting textboxes, radio buttons (multiple choice questions and

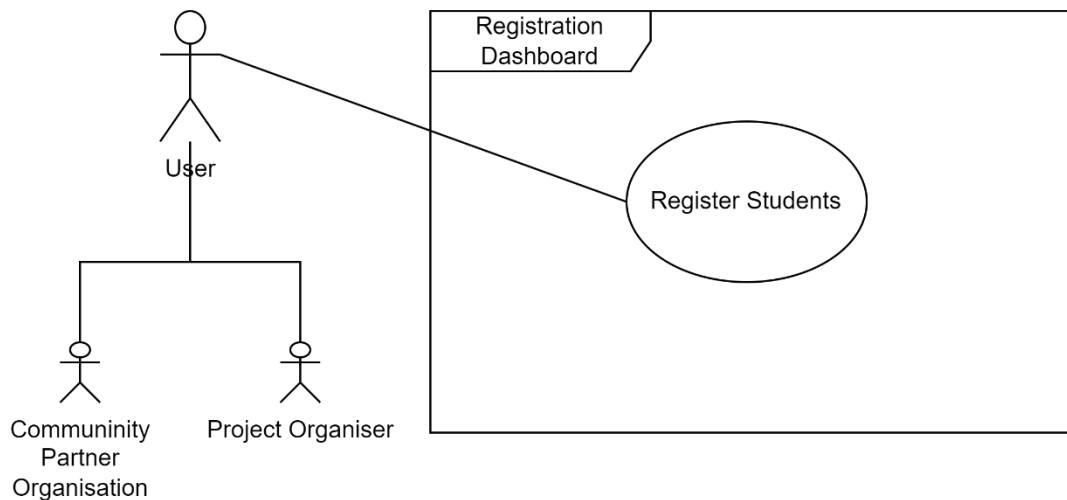
changing the name of the assessments or file. Edit the grammar on the word files before they upload the file as a PDF.

## Alternative Scenario 2

1. After successfully logging in Tshepo will then be directed to the home dashboard.
2. This is where Tshepo hovers to the top left-hand side menu to find and click on the “Staff Room” tab.
3. Once inside Staff Room the system will allow Tshepo access the “Staffroom” dashboard.
4. This will determine whether they have the Upload Course Material, Manage course Material option.
5. Then Tshepo has the option of creating and posting the course material, quizzes and tests.
6. The Tshepo has the option of clicking on the upload document button allowing them to post a file to the system.
7. These files can be created by clicking on the “Create Assessment” inside the “Upload Course Material” tab.
8. This is when a template is provided by the system, allowing the Tshepo to populate the template with information about the test and its content.
9. Tshepo then clicks on the submit button but the system will not allow him to because the information in one or more of the textbox's is entered incorrectly.
10. Tshepo then must proceed to fix his mistakes before using the submit button.

## Registration

**UML:**

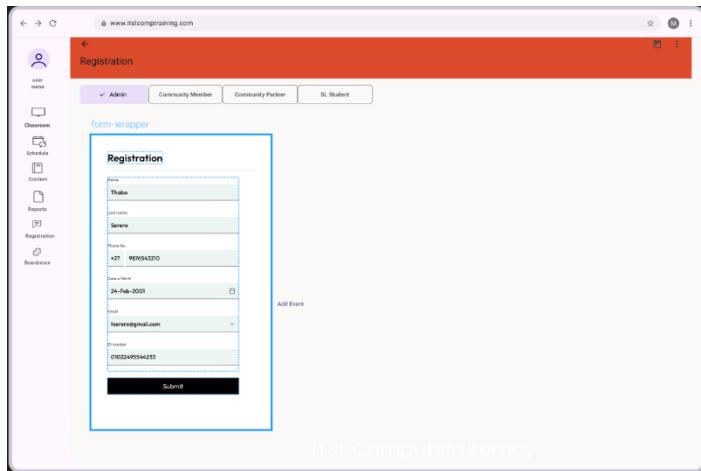


*Figure 5.1: Registration UML*

### Brief Description:

The Registration use case enables the project coordinator and system administrator to receive all the necessary personal information of the community members and register them onto the systems database allowing them to have access the system through the login screen.

### User Interface:



*Figure 5.2.1: Registration UI*

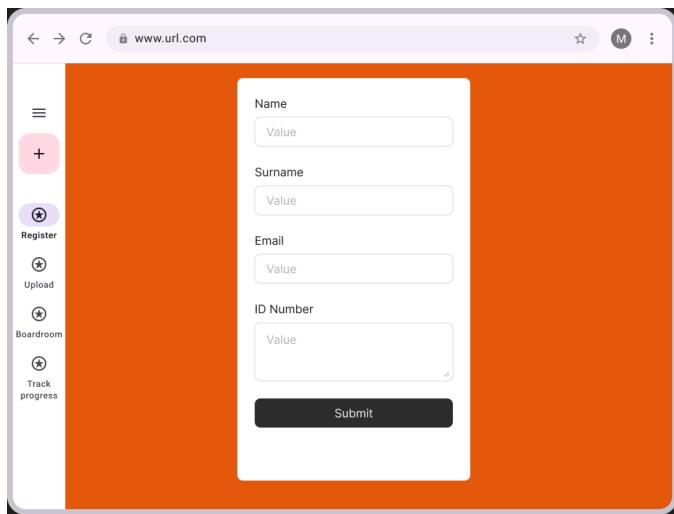


Figure 5.2.2: Registration UI

### Step by step:

1. After the user (project coordinator or system administrator) logs into the system they will hover over the top left-hand corner opening the menu option.
2. Select the Registration option.
3. Choose between the 4 available option tabs (System Admin, SL student, Community Member, Community Partner Organisations) buttons.
4. Fill in the individual's information in the textboxes available.
5. Click on the submit button to save the individuals data onto the database.

### Best-case Scenario:

1. After Mbali logs into the system they will hover over the top left-hand corner opening the menu option.
2. Select the Registration option.
3. Choose between the 4 available option tabs (System Admin, SL student, Community Member, Community Partner Organisations) buttons.
4. Mbali fills in the individual's information in the textboxes available.
5. Mbali clicks on the submit button to save the individuals data onto the database.

### Worst-case Scenario:

1. Daniel wants to register community participants to the service-learning.

2. Daniel must first login to the system successfully and click on the register icon from the navigation panel.
3. When the registration page opens.
4. Daniel can fully register community participants with missing personal information and without the approval of project coordinator.

### **Alternative Scenario 1**

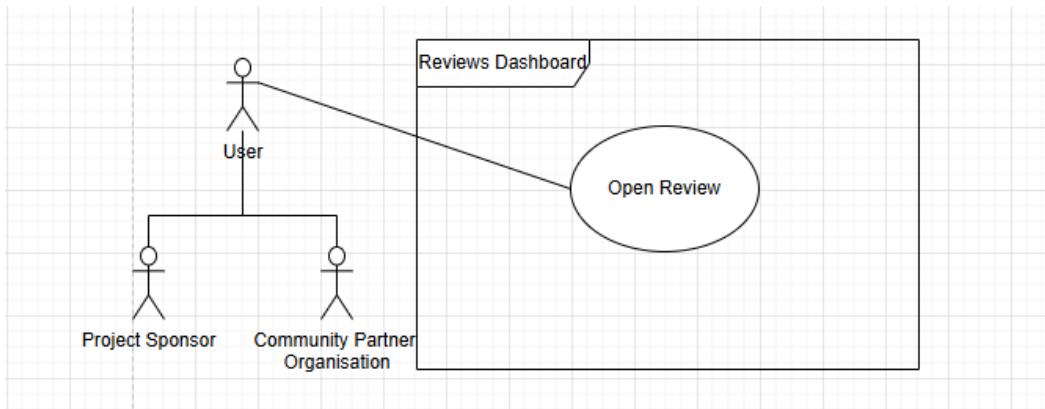
1. After Mbali logs into the system they will hover over the top left-hand corner opening the menu option.
2. Select the Registration option.
3. Choose between the 4 available option tabs (System Admin, SL student, Community Member, Community Partner Organisations) buttons.
4. Mbali fills in the individual's information in the textboxes available.
5. Mbali cannot type anything in the textboxes.
6. The system does not illustrate anything in the textboxes after Mbali has attempted to type Mbali has made the attempt to enter the information.
7. Mbali then is forced to refresh the page or if using an application logout and then log back in.

### **Alternative Scenario 2**

1. After Mabali logs into the system they will hover over the top left-hand corner opening the menu option.
2. Select the Registration option.
3. Choose between the 4 available option tabs (System Admin, SL student, Community Member, Community Partner Organisations) buttons.
4. Mbali fills in the individual's information in the textboxes available.
5. Mbali click on the submit button to save the individuals data onto the database.
6. The system rejects the submission because one or more of the fields has information that is entered incorrectly.
7. Mbali will have to re-enter the information in the textbox in the way that the system prompt suggests.

## **Reports and Statistics**

### **UML**



*Figure 6.1: Reports UML*

#### **Brief Description:**

This functionality allows the Project coordinator and the Community Partner Organisations to access reports and graphical representations that have been created from inputs entered by the system administrator and processed by the system.

#### **User Interface:**

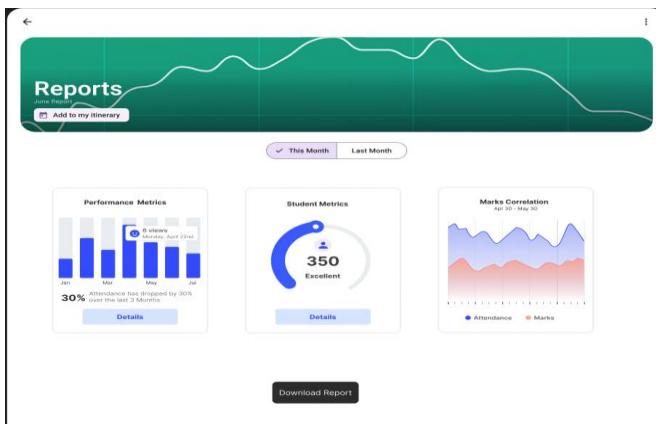


Figure 6.2.1: Reports and Performance tracking UI

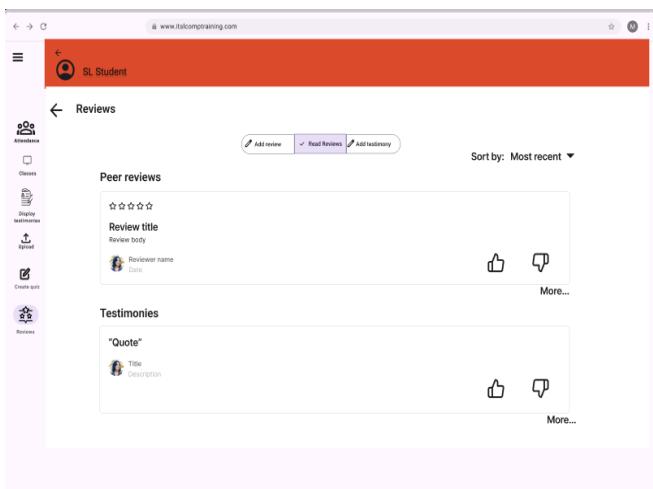


Figure 6.2.2: Reports and Performance tracking UI

### Step by step:

1. After successfully logging into the system. The user will then enter the home dashboard.
2. User will then hover over the top left-hand side of the screen and navigate the menu and click on the “Reports” tab.
3. There will be multiple options including (Create Report, View Report and Remove Report)
4. The user can click on the View report option and view the reports that are readily available on the system.

5. On the bottom of the screen there is a “Download Report” function that allows the user to download the report files onto their computer.

#### **Create Report:**

6. This functionality is only available to the Project coordinator and System Administrator.
7. This allows the user to enter (data) information into the available textboxes, this data includes Total number of students, students that have passed and attendance.
8. Once data is entered the user will then proceed to click on the generate report button at the bottom of the screen.
9. Once the system has created the report the user has the option of editing the report or delete the report and create a new one.
10. Once everything is to the user’s satisfaction then the user can post the report to the system.

#### **Remove Report:**

11. This functionality is only available to the Project coordinator and System administrator.
12. This functionality allows the user to click on the “Remove Button” functionality and select from the list of reports given to them and then click on the report they want to remove.
13. The system will then prompt the user if they are sure they want to remove the report and if they do, they will click on the “yes” button and the report will then be removed from the system.

#### **Best-case Scenario**

1. After successfully logging into the system. The Ntombi will then enter the home dashboard.
2. Ntombi will then hover over the top left-hand side of the screen and navigate the menu and click on the “Reports” tab.
3. There will be multiple options including (Create Report, View Report and Remove Report)
4. Ntombi clicks on the View report option and view the reports that are readily available on the system.

5. On the bottom of the screen there is a “Download Report” function that allows Ntombi to download the report files onto their computer.
6. This allows Ntombi to enter (data) information into the available textboxes, this data includes Total number of students, students that have passed and attendance.
7. Once data is entered the Ntombi will then proceed to click on the generate report button at the bottom of the screen.
8. Once the system has created the report Ntombi has the option of editing the report or delete the report and create a new one.
9. Once Ntombi satisfied, then she can post the report onto the system.

#### **Worst Case Scenario:**

1. Ben navigates to the login page but encounters repeated “Page Not Found” errors. After several attempts and long delays, Ben finally accesses the login page but faces a “Login Failed” message despite entering the correct credentials.
2. This forces Ben to reset the password, leading to further delays and frustration. Upon finally logging in, Ben is directed to a malfunctioning dashboard where the toolbar is unresponsive.
3. The “Testimonial Submission” button is absent, preventing Jasmine from writing any testimonials about the session. This leaves Ben without any student feedback to encourage future participants.
4. The “Review Management” button is broken: When Ben attempts to review or approve testimonials, the system fails to load the reviews.
5. Ben cannot manage appropriate feedback, and inappropriate testimonials get displayed publicly, damaging Ben's reputation.
6. The “Display Testimonial” button malfunctions, failing to display any testimonials. Positive testimonials are not highlighted on Ben's profile, leading to a lack of endorsements and decreased interest from potential new students.
7. The “Anonymous Feedback” button fails completely: Jasmine tries to leave anonymous feedback, but the system logs their name with the review.
8. This discourages honest feedback and makes Jasmine hesitant to provide any more testimonials, leaving Ben without valuable insights into their teaching effectiveness.

#### **Alternative Scenario**

1. After successfully logging into the system. The Ntombi will then enter the home dashboard.

2. Ntombi will then hover over the top left-hand side of the screen and navigate the menu and click on the “Reports” tab.
3. There will be multiple options including (Create Report, View Report and Remove Report)
4. Ntombi clicks on the View report option and view the reports that are readily available on the system.
5. On the bottom of the screen there is a “Download Report” function that allows Ntombi to download the report files onto their computer.
6. This allows Ntombi to enter (data) information into the available textboxes, this data includes Total number of students, students that have passed and attendance.
7. Once data is entered the Ntombi will then proceed to click on the generate report button at the bottom of the screen.
8. The system creates an incorrect report that requires modification.
9. Ntombi will thus needs to click on the “modify report” button. This will allow her to change the parameters on the reports and graphs. If the report still is not generated correctly, then Ntombi can delete the report and start the whole process again.
10. Once Ntombi is satisfied, then she can post the report onto the system.

#### **Alternative Scenario 1:**

1. Ben encounters significant delays and security warnings when trying to access the login page.
2. After finally logging in, the dashboard loads slowly and incompletely, with key buttons like the “Testimonial Submission” non-functional.
3. The “Testimonial Submission” button leads to a system crash each time Jasmine attempts to write a testimonial.
4. Any effort to recover the system fails, leaving Jasmine unable to submit any feedback. This lack of testimonials makes it difficult for Ben to gather positive reviews.
5. The “Review Management” button displays incorrect testimonials. When Ben tries to approve feedback, the system fails to save Ben’s approvals.
6. The “Display Testimonial” button is unreliable, Positive testimonials do not appear on Ben’s profile. Ben’s profile appears empty and poorly maintained, reducing the likelihood of attracting new students.
7. The “Anonymous Feedback” button logs names unintentionally: Jasmine’s attempts to leave anonymous feedback result in their identity being displayed with the testimonial, leading to a breach of trust.

8. This error discourages students from leaving candid reviews, causing Ben to lose out on valuable, honest feedback.

## Calendar

### UML:

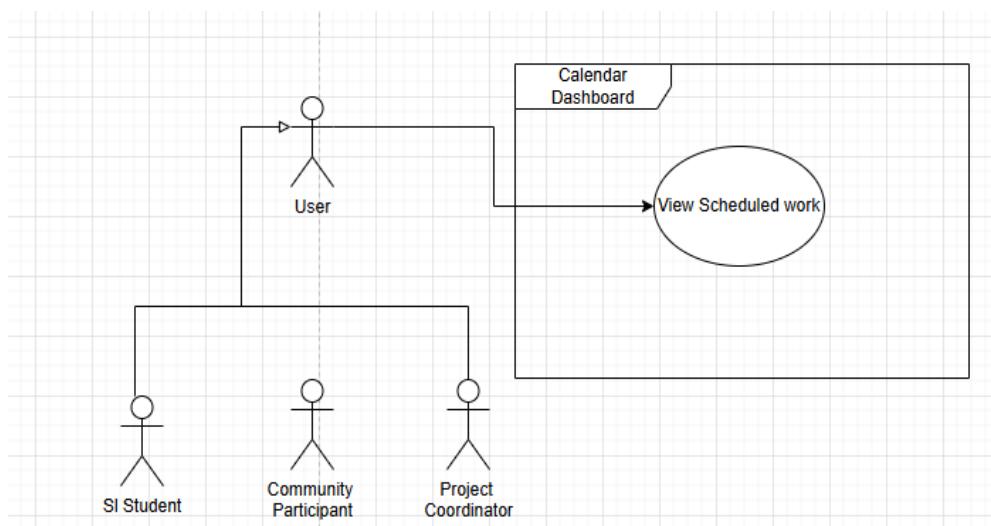


Figure 7.1: Calendar UML

### Brief description:

An integrated calendar helps users keep organized and track course progress. This tool makes sure that all course milestones are accomplished on time and that participants stay focused and on track during the course by enabling users to monitor completed work, examine future sessions and deadlines, and receive reminders.

### User Interface:

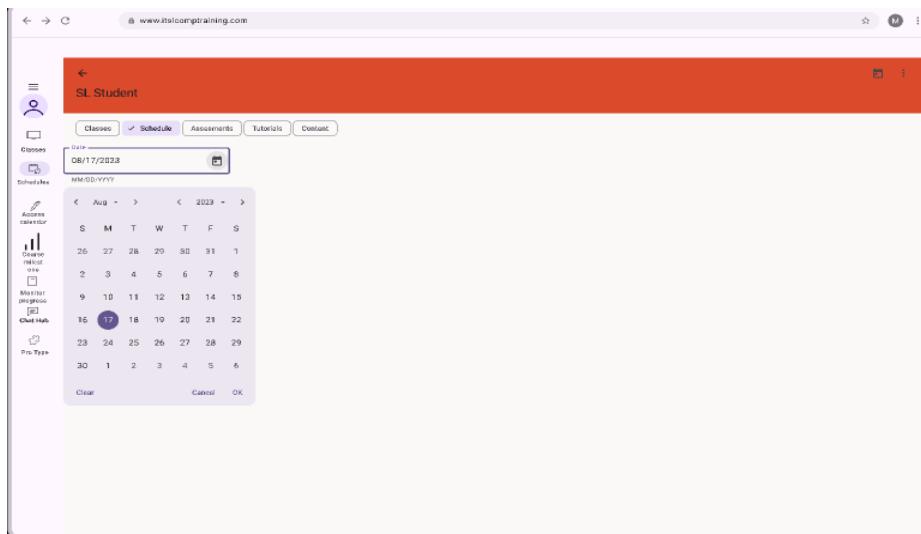


Figure 7.2.1: Calendar UI

### Step-by-Step:

1. The User navigates to the login page and enter their username and password to access the system.
2. After logging in, the tutor is directed to the dashboard and can access a tool bar on the top left of the screen for options such as:
3. Monitor Progress: By clicking this button the tutors can use monitor the overall progress of the class, identifying participants who may be falling behind and might need additional support. They can also see which topics or activities have been covered and plan future sessions accordingly.
4. Access Calendar: Both tutors and participants can view a calendar that displays upcoming sessions, deadlines for assignments or quizzes. The calendar also includes reminders for upcoming tasks
5. Course Milestones: By clicking this button, the system will highlight major milestones in the course, such as the completion of a module, helping participants feel a sense of achievement.

### Best Case Scenario:

1. Ben navigates to the login page and enter their username and password to access the system.
2. After logging in, Ben is directed to the dashboard and can access a tool bar on the top left of the screen for options: By clicking the 'Monitor Progress' button Ben can

use monitor the overall progress of the class, identifying Jasmine who may be falling behind and might need additional support.

3. Ben can also see which topics or activities have been covered and plan future sessions accordingly.
4. Both Ben and Jasmine can view a calendar by clicking ‘Access Calendar’ that displays upcoming sessions, deadlines for assignments or quizzes.
5. The calendar also includes reminders for upcoming tasks.
6. By clicking the ‘Course Milestones’ button, the system will highlight major milestones in the course, such as the completion of a module, helping Jasmine feel a sense of achievement.

#### **Worst Case Scenario:**

1. Upon finally logging in, Ben is redirected to a glitch dashboard where the toolbar is not fully loaded.
2. The “Monitor Progress” button is absent, preventing Ben from tracking the class’s progress. This leaves Ben unaware of which students, like Jasmine, might be struggling, and prevents any timely intervention.
3. The “Access Calendar” button is non-functional, leading to an error page when clicked. Both Ben and Jasmine are unable to view the calendar, missing important upcoming sessions, deadlines, and reminders. This results in missed assignments and poorly attended sessions.
4. The “Course Milestones” button inaccurately shows incomplete and incorrect data. Jasmine receives no acknowledgment of completed modules, leading to frustration and a lack of motivation as there’s no sense of progression or achievement in the course.
5. System instability leads to frequent crashes throughout the session, erasing any progress Ben’s efforts in attempting to monitor the class, update the calendar, and track milestones disappear.
6. This forces Ben to repeatedly log back in an attempt to recover lost work, disrupting the session and causing significant frustration for both Ben and Jasmine.

#### **Alternative Scenario 1:**

1. The “Monitor Progress” button is present but fails to load any data when clicked.

2. Ben is unable to see the progress of the class or identify students who may need additional support.
3. The system shows outdated information, misleading Ben into thinking the class is performing well when some students are actually falling behind.
4. The “Access Calendar” button displays an empty calendar with no upcoming sessions, deadlines, or reminders.
5. When Ben and Jasmine attempt to add events manually, the system refuses to save the entries. This results in a disorganized schedule, with both missing critical sessions and deadlines.
6. The “Course Milestones” button triggers a system error that logs Ben out or crashes the platform.
7. When Ben logs back in, previously tracked milestones are lost, and Jasmine receives no recognition for completed modules, leading to confusion and decreased morale.
8. The platform becomes increasingly unstable, with frequent timeouts and crashes, especially when attempting to access progress reports and update the calendar.
9. This forces Ben to restart the system multiple times, wasting valuable time and creating frustration for both Ben and Jasmine.

#### **Alternative Scenario 2:**

1. The “Monitor Progress” feature is severely limited, displaying incomplete data.
2. Ben cannot accurately track which students are struggling, leading to missed opportunities for intervention and support.
3. The system displays incorrect progress, causing Ben to misjudge the class’s performance.
4. The “Access Calendar” button opens a corrupted calendar view with overlapping events, missing deadlines, and incorrect dates.
5. Ben and Jasmine are unable to rely on the calendar for scheduling, resulting in missed assignments, poorly attended sessions, and overall disorganization.
6. The “Course Milestones” feature is entirely non-functional, with the button not responding.
7. Jasmine does not receive any notifications of achievements or module completions, leading to disengagement and a lack of motivation.
8. Frequent system crashes and data loss plague the session, particularly when trying to access or update the calendar and milestones.
9. Ben is forced to repeatedly restart the platform, leading to lost work and a disjointed learning experience for Jasmine.

10. The lack of reliable tools makes it difficult for Ben to manage the course effectively, and Jasmin's learning progress suffers as a result.

## Study material access

**UML:**

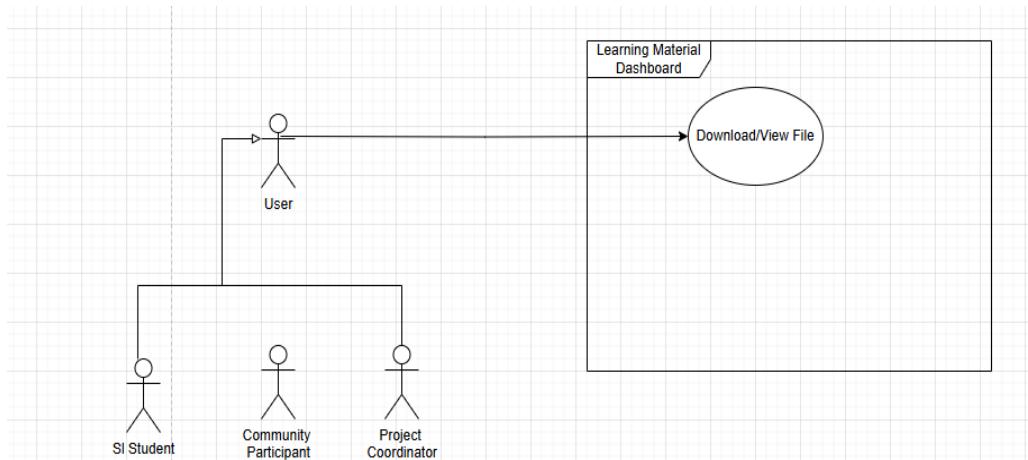


Figure 8.1: Study material access UML

**Brief description:** Study resources, such as session recordings and documentation, are accessible to and managed by users via a well-organized platform. This feature ensures that participants have quick access to vital resources, making it possible for them to download materials, review session content, and use a searchable archive. This ensures that participants receive full support throughout the learning process.

**User Interface:**

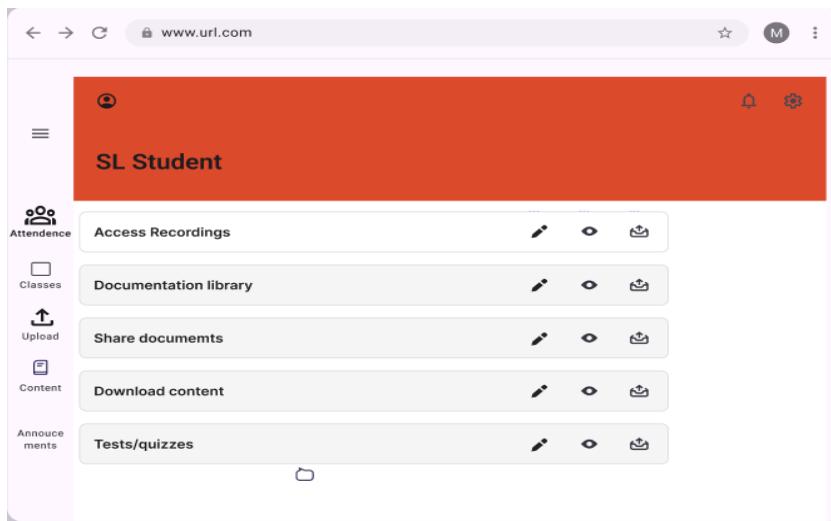


Figure 8.2: Study material UI

### Step-by-Step:

1. The User navigates to the login page and enter their username and password to access the system.
2. After logging in, the tutor is directed to the dashboard and can access a tool bar on the top left of the screen for options such as:
3. Documentation Library: By clicking this button the tutors can upload and various study materials, such as lecture notes, presentation slides, handouts in which participants can easily access these materials at any time.
4. Share documents: Users can click the 'share documents' button to share specific materials with participants, either during the live session or afterward. Participants can receive notifications when new materials are available.
5. Download Content: By clicking this button, the tutors can upload content which Participants can have the option to download session recordings and study materials for offline use, allowing them to study even without internet access.
6. Access Recordings: After a live session is recorded, participants can access these recordings through the website/app. This allows them to review the session content at their own pace, reinforcing their understanding of the material.

### Best Case Scenario:

1. Ben navigates to the login page and enter their username and password to access the system.
2. After logging in, Ben is directed to the dashboard and can access a tool bar on the top left of the screen for options.
3. By clicking the ‘Documentation Library’ button Ben can upload and various study materials, such as lecture notes, presentation slides, handouts in which Jasmine can easily access these materials at any time.
4. Ben clicks the ‘share documents’ button to share specific materials with Jasmine, either during the live session or afterward. Jasmine can receive notifications when new materials are available.
5. By clicking the ‘Download Content’ button, Ben can upload content which Jasmine can have the option to download session recordings and study materials for offline use, allowing them to study even without internet access.
6. Access Recordings: After a live session is recorded, Jasmine can access these recordings through the website/app.
7. This allows them to review the session content at their own pace, reinforcing their understanding of the material.

#### **Worst Case Scenario:**

1. Thevin Bohmer accesses the system by signing in, with his credentials. Entering his password and usernames in the provided spaces. Thevin focuses his cursor to the navigation on the left.
2. He clicks on the Content button. This allows him to access study material on the system.
3. Thevin selects the download button to save the resources on his device. Once Thevin clicks the download button the download process begins in the background.
4. Thevin clicks the download button twice and system downloads the selected document again.
5. Thevin continues clicking the three more time. After the last click. 5 documents open in the browser dividing filling Thevin’s screen.
6. Thevin opts to close the quiz and return to the main m

#### **Alternative Scenario 1:**

1. The “Documentation Library” button leads to a system crash each time Ben tries to upload materials.

2. Any attempt to recover the system fails, and Ben is unable to make study materials available to Jasmine.
3. The session is delayed or disrupted as a result.
4. The “share documents” button sends outdated files, when Ben attempts to share materials, the system sends the wrong documents, causing confusion and frustration for Jasmine.
5. Additionally, notifications fail to reach Jasmine, leaving her unaware of any added resources.
6. The “Download Content” button fails to allow downloads: Jasmine attempts to download study materials or session recordings, but the download will not start.
7. This prevents Jasmine from studying offline, leaving her at a disadvantage if she lacks consistent internet access.
8. The “Access Recordings” feature is unreliable: Recordings are missing, some incomplete, with key sections of the session not captured.
9. Jasmine struggles to review the session, as the recording cuts off unexpectedly or displays poor audio and video quality, making it difficult to follow along.

#### **Alternative Scenario 2:**

1. The “Documentation Library” button is non-functional, attempts to upload study materials are met with persistent error messages, preventing Ben from sharing critical resources with Jasmine.
2. The course is severely impacted as key materials cannot be accessed.
3. The “share documents” button is unreliable: When Ben shares materials, the system delays sending the documents and they arrive incomplete.
4. Notifications fail to alert Jasmine about new materials, leaving her without access to the necessary resources to prepare for sessions.
5. The “Download Content” button corrupts files: When Jasmine attempts to download study materials or session recordings, the files are corrupted, rendering them useless.
6. This leaves Jasmine unable to study offline, severely affecting her ability to keep up with the course.
7. The “Access Recordings” feature consistently fails: Recordings are inaccessible due to technical errors. Even when recordings are available, they suffer from poor quality, such as missing audio or video, making them unusable for review.
8. Jasmine is left without the ability to reinforce her learning, leading to gaps in her understanding of the material.

# Announcements

## UML:

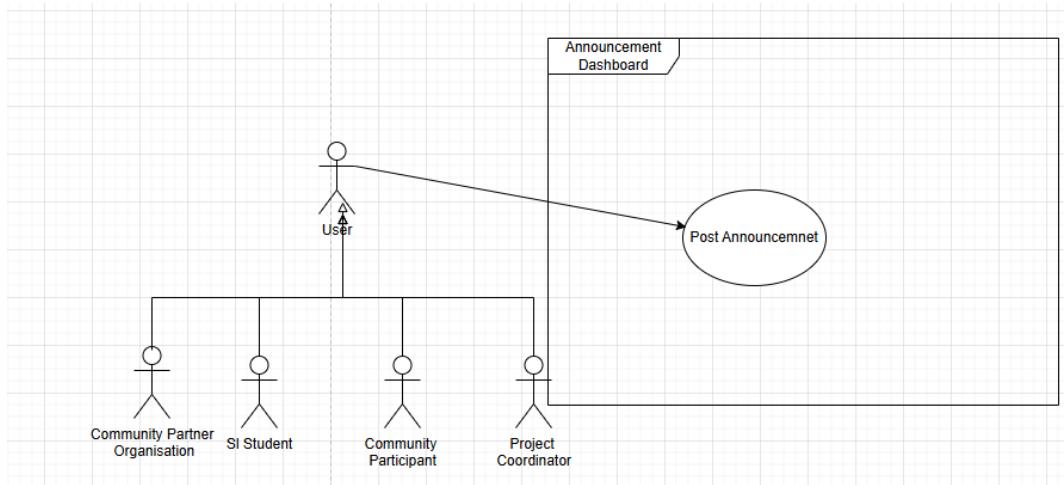


Figure 9.1: Announcements UI

**Brief Description:** Instructors have the ability to plan upcoming announcements, compose and distribute announcements to every participant in real-time sessions, and oversee critical alerts. This function improves group coordination and participation by making sure that important updates, reminders, and messages are distributed to everyone in the group.

## User Interface:

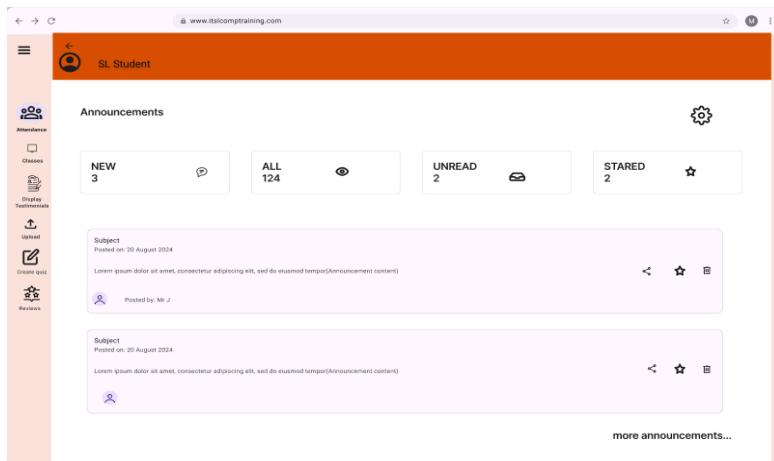


Figure 9.2.1: Announcements UI

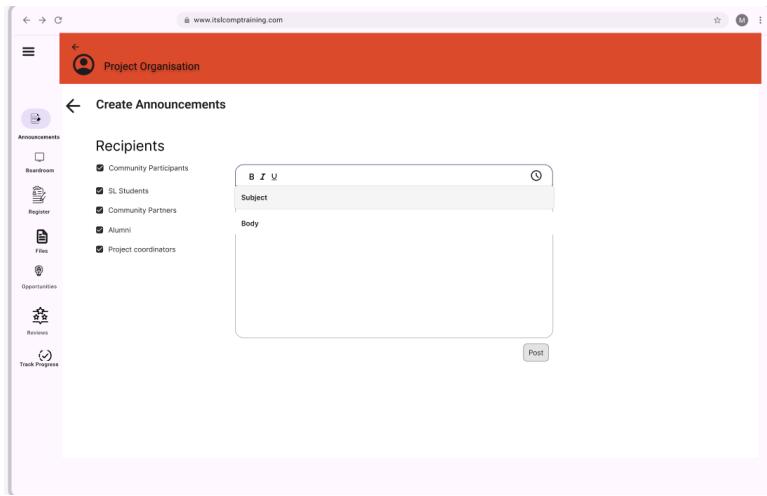


Figure 9.2.2: Announcements UI

### Step-by-Step:

1. After the user has successfully logged into the system:
2. The user has to click on the Announcement tab found on the menu at the top left of the window that is represented by a paper and pen icon.
3. Once clicked, the user will be directed to the Announcement dashboard, where they can view a comprehensive overview of their announcements. The dashboard displays metrics on the number of sent, scheduled, and all announcements made at the top of the page.
4. By clicking on the metrics, the user can access:
  - Sent announcements from the past month
  - All announcements, including the number of viewers
  - Scheduled announcements, along with their scheduled dates
5. Below the metrics, the user can see their most recent announcements. They can share, star, or delete these announcements by clicking on the corresponding icons on the far left of each announcement.
6. Clicking on the:
  - Share icon allows user to share the announcement with others
  - Delete icon represented by a bin allows user to delete the announcement, making it inaccessible to recipients

- Star icon represented by a star allows user to star the announcement, adding it to the "Starred" list, accessible by clicking on the "Starred" metrics at the top right of the dashboard
7. The Create icon, found at the top left of the window, allows the user to write and post an announcement.
  8. To create a new announcement, the user clicks the Create icon at the top left of the window and they will be directed to the Create Announcement page.
  9. Once the user has accessed the page they will click on checkbox to select the recipients of the announcement. A selected checkbox will be shown by a tick inside the box and the box will have shaded black
  10. Next, the user will double click on the Subject of the announcement to where a flickering cursor will appear allowing user to add the announcement subject and double clicks on the body to compose the announcement.
  11. The text editing tools found at the top of the Subject allow user to format and edit the announcement. The user can Click on the:
    - B icon – to write I bold
    - I icon – to write in italics
    - U icon- to underline text
  12. The user clicks on the schedule icon found at the top left of the announcement text editing tools bar, providing the user with the option to schedule the publication of the announcement.
  13. To post the announcement the user has to click the Post button, found below the announcement body. Once the user clicks the post button and the announcement has been sent the user will receive a pop-up message stating that the announcement has been posted successfully.

#### **Best Case Scenario:**

1. After Kamo has successfully logged into the system:
2. He clicks on the Announcement tab found on the menu at the top left of the window.
3. Once he has clicked, is directed to the Announcement dashboard, where he will view a comprehensive overview of his announcements.
4. The dashboard displays metrics on the number of sent, scheduled, and all announcements made at the top of the page.

5. By clicking on the metrics, he can access:
  - Announcement he sent from the past month
  - All his announcements, including the number of viewers
  - Scheduled announcements, along with their scheduled dates
6. Below the metrics, Kamo see his most recent announcements. He clicks on the share icon to share the announcement with his co-worker.
7. He clicks on the delete icon of an announcement he made a month ago so that he can delete the announcement. And he stars his recent sent announcement by clicking on the star icon.
8. He remembers that there is an important announcement he must share with project coordinators as well as Users.
9. He clicks on the Create Icon, clicks on the User and Project Coordinators checkboxes to select them as recipients of the announcement.
10. He proceeds to double clicks the Subject to add the subject and double clicks the body to write the announcement.
11. He clicks on the B icon to make important parts in the announcements bold.
12. Kamo schedules the announcement to be sent the following morning by clicking on the schedule icon. After he proofreads his announcement, he finally clicks the Post button.
13. After sending the announcement, his colleague asks him to share with him an announcement he made a month ago.
14. He then clicks on the settings icon and selects the option to show announcements for the last month, he finds the announcement and sends it to his colleague.
15. He then clicks on the back tab to go back to the main menu.

#### **Worst case scenario:**

1. John successfully logs into the system he clicks on the Announcement tab found on the menu at the top left of the window.
2. He is directed to the Announcement dashboard where he is presented with announcements made in the last year.
3. John wishes to share an announcement with the community members about applications of a new learning opportunity, he struggles to find the Create button and clicks on all icons on the window before finally clicking on the correct one.
4. He is now on the "Create Announcement" page, he forgets to click on the community member checkbox to select the community members as the recipients of the announcement. Next, he attempts to write the announcement but encounters

difficulties using the formatting tools, resulting in an announcement that is hard to read and understand.

Finally, he clicks the Post button. He keeps on getting an error message every time he clicks on the post button.

#### **Alternative Scenario 1:**

1. Ben struggles to access the login page due to server downtime, receiving “Service Unavailable” messages.
2. After multiple retries, the page loads, but the system is slow, causing significant delays in logging in.
3. The dashboard loads with missing elements: The toolbar appears, but the “upload announcement” button is missing, and the dashboard displays random errors.
4. Ben tries to refresh the page, but this causes the entire session to reset, forcing Ben to log in again.
5. The “upload announcement” button causes the system to freeze, making it impossible for Ben to send any announcements.
6. When the system finally unfreezes, all previously drafted announcements are lost, and the button disappears from the toolbar.
7. The “post-session” announcements” button triggers an unexpected logout when clicked.
8. Ben is forced out of the system before sending any wrap-up messages, leaving Jasmine without important follow-up information.
9. System security issues arise: After the session, Ben discovers that an unauthorized user accessed the dashboard due to a security breach.
10. Announcements and session data are compromised, requiring immediate intervention from IT support and causing significant concern for both Ben and Jasmine.

#### **Alternative Scenario 2:**

1. Zandi successfully logs into the system, she clicks on the Announcement tab found on the menu at the top left of the window. She is directed to the announcement dashboard.

2. She clicks on the settings icon to edit the announcement so she can view announcement she made from the last 7 days. After she went through the announcements, she realises that she did not share an important announcement with the community members.
  3. She proceeds to click the back button which directs her back to the Announcement page. She clicks the create icon. After she has access to the Create Announcement page, she tries to double click the Subject of the announcement, but the form is greyed out and she cannot type in the subject. She notices that she did not click on the community member's checkbox, and she does so. Soon after she has selected the recipients, she notices that the form is no longer greyed out.
  4. She double clicks the Subject textbox, composes the body of the announcement and clicks the post button.
- 
5. Nala logs into the system and clicks on the Announcement tab. She wants to view the announcements she sent to the community members last quarter to evaluate the engagement.
  6. She clicks on the settings icon and clicks on the option to show announcements from the last month.
- 
7. She clicks the back tab to go back to the Announcement tab. She then clicks on the "Sent" metric at the top of the page to view only the announcements she has sent. She scrolls through the list and finds an announcement with a high number of views.
  8. She clicks on the announcement to view the details, including the number of viewers. After reviewing the announcement, she clicks on the back button to return to the Announcement dashboard.
- 
9. She then decides to view the announcements she has scheduled for the next day and clicks on the "Scheduled" metric at the top of the page.

# Study Material Access –Recording and Documentation

## UML:

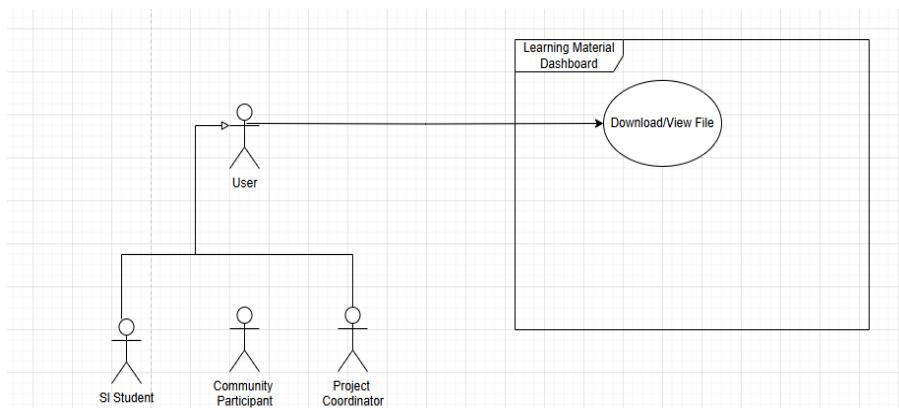


Figure 10.1: Study Material UML

## Brief Description

The study material access allows users to access study material which includes recording and documentation from the system.

## User Interface:

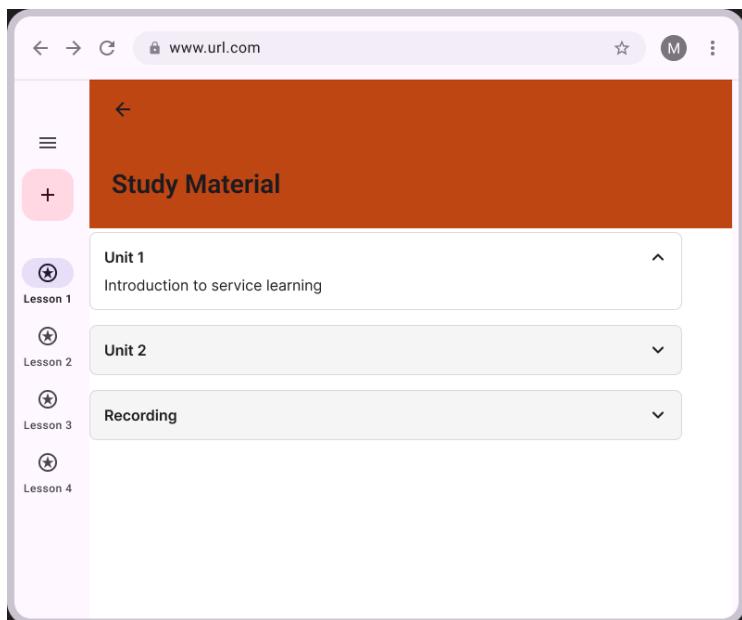


Figure 10.2.1: Study Material access UI

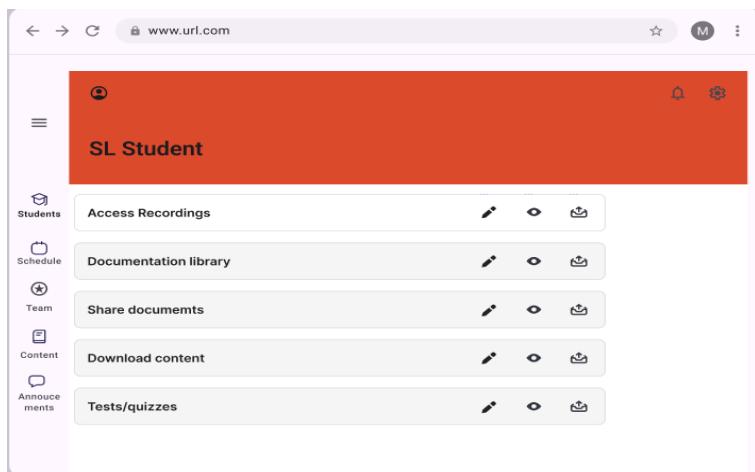


Figure 10.2.2: Study Material access UI

### Step-by-Step:

1. The users navigate to the login page and enter their username and password to access the system.
2. After logging in, the user is directed to the dashboard and can access a tool bar on the top left of the screen for options such as:
3. Documentation Library: By clicking this button, the user can view various study materials, such as lecture notes, presentation slides, handouts in which they can easily access these materials at any time.
4. Download Content: By clicking this button, the Participants have the option to download session recordings and study materials for offline use, allowing them to study even without internet access.
5. Access Recordings: After a live session is recorded, participants can access these recordings through the website/app. This allows them to review the session content at their own pace, reinforcing their understanding of the material.

### Best Case scenario

1. Sello Melokwe access the system by signing in, with his credentials. Entering his password and usernames in the provided spaces.
2. Sello focuses his cursor to the navigation on the left.

3. He clicks on the Content button. This allows him to access study material on the system.
4. Sello selects the download button to save the resources on his device.
5. Once they are done, they can click the back button to return to the dashboard.

### **Worst Case Scenario:**

1. Upon logging in, Ben is directed to a broken dashboard where the toolbar not fully functional.
2. The “Documentation Library” button is unresponsive and leads to a blank page, preventing Ben from uploading or accessing any study materials.
3. Jasmine is left without essential resources needed for the course.
4. The “share documents” button malfunctions, when Ben tries to share specific materials, the system fails to send the documents.
5. Jasmine does not receive any notifications of new materials, leaving her unaware of important resources she needs for the session or homework.
6. The “Download Content” button triggers errors, when Ben attempts to upload content, the system fails to save the files.
7. Jasmine cannot download the session recordings or study materials, leaving her unable to study offline.
8. Any attempt to access previously downloaded content results in incomplete that are unusable.
9. The “Access Recordings” feature fails to work: After a live session, the recording is not available and shows a “File Not Found” error when Jasmine tries to access it.
10. This leaves Jasmine without the ability to review the session, leading to gaps in understanding and missed learning opportunities

### **Alternative scenario 1**

1. Sello Melokwe access the system by signing in, with their credentials. Entering their password and usernames in the provided spaces.
2. Sello focuses his cursor to the navigation on the left.
3. He clicks on the Content button. This allows the him to access study material on the system.
4. Sello selects the eye icon button to view the uploaded material on the system.
5. Once they are done, they can click the back button to return to the dashboard.

### **Alternative scenario 2:**

1. The “Documentation Library” button leads to a system crash each time Ben tries to upload materials.
2. Any attempt to recover the system fails, and Ben is unable to make study materials available to Jasmine.
3. The session is delayed or disrupted as a result.
4. The “share documents” button sends outdated files, when Ben attempts to share materials, the system sends the wrong documents, causing confusion and frustration for Jasmine.
5. Additionally, notifications fail to reach Jasmine, leaving her unaware of any added resources.
6. The “Download Content” button fails to allow downloads: Jasmine attempts to download study materials or session recordings, but the download will not start.
7. This prevents Jasmine from studying offline, leaving her at a disadvantage if she lacks consistent internet access.
8. The “Access Recordings” feature is unreliable: Recordings are missing, some incomplete, with key sections of the session not captured.
9. Jasmine struggles to review the session, as the recording cuts off unexpectedly or displays poor audio and video quality, making it difficult to follow along.

## Assignments/Quiz

**UML:**

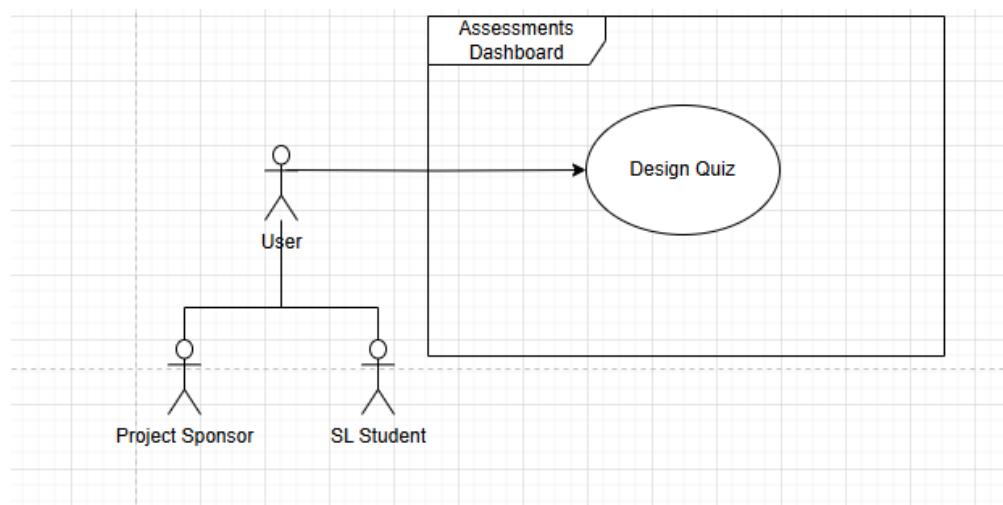


Figure 11.1: Assignments/Quiz UML

**Brief description:** During live sessions as well as afterward, users have the ability to generate, administer, and oversee homework assignments and tests for students. With the help of this tool, users can give scheduled or real-time evaluations, gauge participant understanding, and provide feedback to improve the learning process. The users will provide comments and grades to participants who do these exercises online and keep track of their progress.

### User Interface:

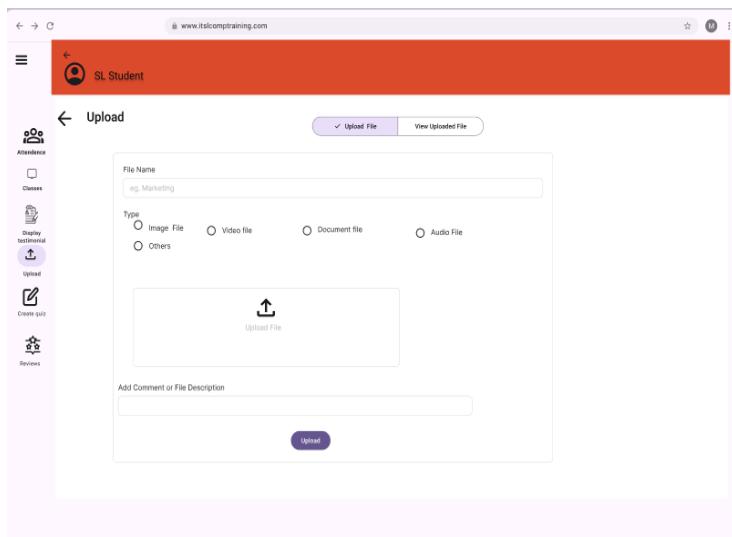


Figure 11.2: Assignments/Quiz UI

### Step-by-Step:

1. The User navigates to the login page and enter their username and password to access the system.
2. After logging in, the tutor is directed to the dashboard and can access a tool bar on the top left of the screen for options such as:
3. Create quiz: By clicking this button, the tutor can create assignments or quizzes directly within the platform. This includes a variety of question types such as multiple choice, true/false, short answer questions.
4. Upload quiz: Once the quiz is created, the tutor can click the 'upload quiz' button to upload a quiz to participants either during the live session or as homework.
5. Review and Retry: By clicking this button, the tutor can allow participants to review their answers after submission and retry the quiz or assignment if needed.

6. Submission Deadlines: By clicking this button, the tutor can set deadlines for assignment or quiz submissions, and the system can send reminders to participants as the deadline approaches.

#### **Best Case Scenario:**

1. Ben navigates to the login page and enter their username and password to access the system.
2. After logging in, Ben is directed to the dashboard and can access a tool bar on the top left of the screen for options: By clicking the Create quiz button, Ben can create assignments or quizzes directly within the platform.
3. This includes a variety of question types such as multiple choice, true/false, short answer questions. Once the quiz is created, Ben can click the 'upload quiz' button to upload a quiz to Jasmine either during the live session or as homework.
4. By clicking the Review and Retry button, Ben can allow Jasmine to review their answers after submission and retry the quiz or assignment if needed.
5. By clicking the 'Submission Deadlines' button, Ben can set deadlines for assignment or quiz submissions, and the system can send reminders to Jasmine as the deadline approaches.

#### **Worst Case Scenario:**

1. The "Create quiz" feature has bugs, which leads to question, like multiple choice and true/false, not save properly, and the platform randomly deletes questions after they've been created.
2. The interface crashes frequently, forcing Ben to restart the process multiple times.
3. The "upload quiz" button is non-functional, displaying an error message stating, "Quiz upload failed." Ben cannot share the quiz with Jasmine during the live session or assign it as homework, leaving Jasmine without critical assessment tools.
4. The "Review and Retry" button fails to work, preventing Jasmine from reviewing and retrying their answers.
5. The platform mistakenly marks all answers as incorrect, even when they are right, causing confusion and frustration for Jasmine.
6. The "Submission Deadlines" button does not save the deadlines properly. Reminders are not sent, causing Jasmine to miss crucial submission windows. At the end of the session, the system crashes completely, resulting in the loss of all progress made during the session.

7. The quiz data, submission deadlines, and reminders are not saved, forcing Ben to redo everything, wasting time, and leaving Jasmine without any assignments or quizzes.

### **Alternative Scenario 1**

1. After gaining access to the dashboard, Ben finds that the toolbar is partially loaded, but the “Create quiz” button does not function.
2. Clicking it results in a “Feature Temporarily Unavailable” message, preventing Ben from creating any quizzes. The platform crashes mid-quiz creation, causing Ben to lose all progress.
3. Upon restarting, the quiz content is incomplete, with missing questions and options.
4. The “upload quiz” button results in corrupted files: When the quiz is uploaded, Jasmine sees an incomplete version of the quiz, making it impossible to complete the assignment properly.
5. The “Review and Retry” feature incorrectly marks all answers as correct, making it impossible for Jasmine to identify and correct mistakes.
6. The retry option is also disabled due to a system bug, leading to a poor learning experience.
7. The “Submission Deadlines” button fails to send reminders, and deadlines disappear from the system, leaving Jasmine unaware of when assignments are due.
8. At the conclusion of the session, the system logs Ben out unexpectedly, and the quiz data is not saved.
9. Any attempts to recover the data result in an error, leaving Ben with no choice but to inform Jasmine that the quiz must be redone.

### **Alternative Scenario 2:**

1. Upon accessing the dashboard, Ben discovers that the toolbar and key features, like the “Create quiz” button, are inaccessible.
2. The system attempts to contact support are met with automated responses, and the session time is slipping away.
3. The “Create quiz” feature is unresponsive, taking too long to load and then crashing without saving any work. Ben tries multiple times, but each attempt results in the same failure.
4. The “upload quiz” button uploads the wrong version of the quiz: The platform mistakenly uploads an older, outdated quiz instead of the new one, confusing Jasmine and leading to poor results.

5. The “Review and Retry” button fails completely, giving Jasmine incorrect feedback on their answers. Attempts to retry the quiz cause the system to freeze, forcing a restart that doesn’t resolve the issue.
6. The “Submission Deadlines” feature sends incorrect reminders, sending them too early, too late, sometimes not at all.
7. This creates confusion and frustration to the participants. By the end of the session, the system crashes during logout, resulting in the loss of all quiz data and settings.
8. Ben is unable to recover the information, leading to a significant delay in the academic schedule, and Jasmine receives no assignments or feedback, severely disrupting their learning progress.

## Testimony and Review

**UML:**

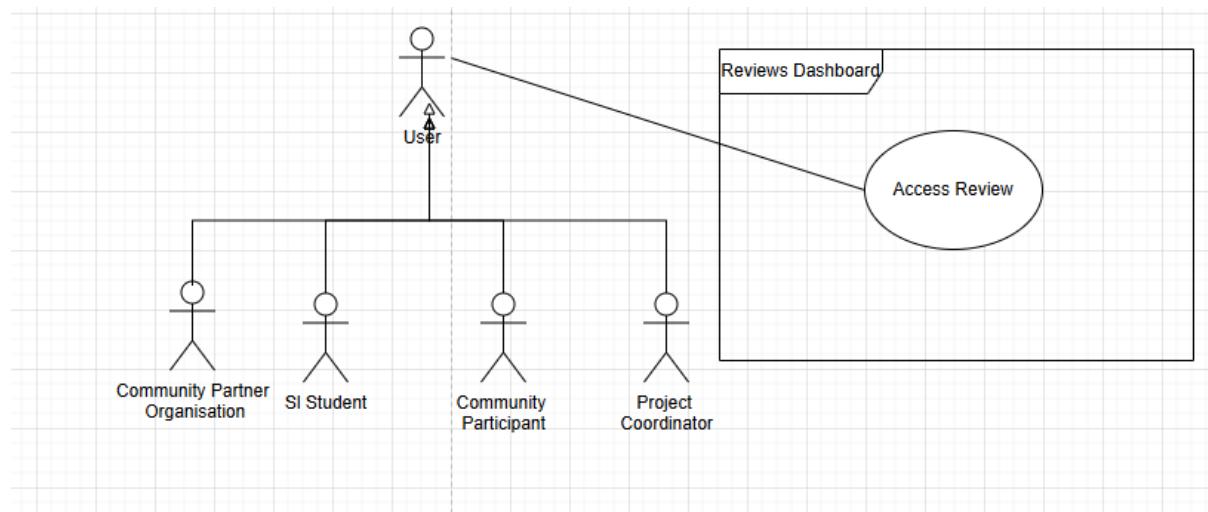


Figure 12.1: Testimony and Review UML

### Brief Description

This functionality allows the user to view the reviews left by students, and alumni's, it is mostly just a display of a list of reviews and has no special features.

## User Interface:

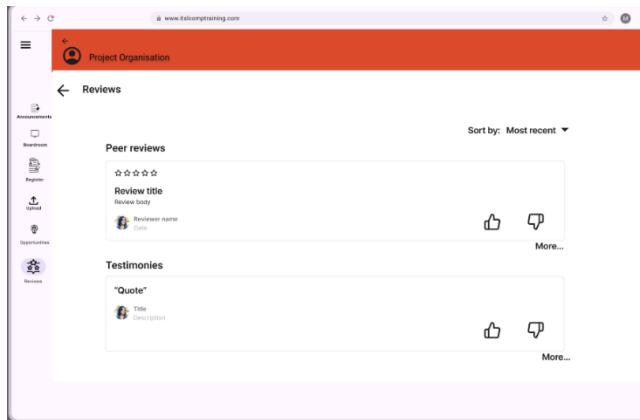


Figure 12.2: Testimony and Review UI

## Step-by-Step:

1. The User navigates to the login page and enter their username and password to access the system.
2. After logging in, the tutor is directed to the dashboard and can access a tool bar on the top left of the screen for options such as:
3. Testimonial Submission: By clicking this button, participants could have the option to write a short testimony about how the session helped them, which could be used to encourage other students to join future sessions.
4. Review Management: By clicking this button at the right side of the screen, the tutors can have the ability to review, approve, or highlight certain testimonies or reviews, which ensures that only appropriate feedback is displayed publicly.

5. Display Testimonials: When the user clicks 'Display Testimonial' button on the left side of screen, it will display Positive testimonials will be showcased on the tutor's profile, serving as endorsements for their teaching style and effectiveness.
6. Anonymous Feedback: By clicking this button the participants can't have the option to leave anonymous reviews or testimonies, which allows them to provide honest feedback without identifying themselves.

#### **Best Case Scenario:**

1. Ben navigates to the login page and enter their username and password to access the system. After logging in, Ben is directed to the dashboard and can access a tool bar on the top left of the screen for options.
2. By clicking the Testimonial Submission, button, Jasmine could have the option to write a short testimony about how the session helped them, which could be used to encourage other students to join future sessions.
3. By clicking the 'Review Management' button at the right side of the screen, Ben can have the ability to review, approve, or highlight certain testimonies or reviews, which ensures that only appropriate feedback is displayed publicly.
4. When the Ben clicks 'Display Testimonial' button on the left side of screen, it will display Positive testimonials will be showcased on Ben's profile, serving as endorsements for their teaching style and effectiveness.
5. By clicking the Anonymous Feedback button, the Jasmine cannot have the option to leave anonymous reviews or testimonies, which allows them to provide honest feedback without identifying themselves

#### **Worst Case Scenario:**

1. Ben navigates to the login page but encounters repeated "Page Not Found" errors. After several attempts and long delays, Ben finally accesses the login page but faces a "Login Failed" message despite entering the correct credentials.
2. This forces Ben to reset the password, leading to further delays and frustration. Upon finally logging in, Ben is directed to a malfunctioning dashboard where the toolbar is unresponsive.
3. The "Testimonial Submission" button is absent, preventing Jasmine from writing any testimonials about the session. This leaves Ben without any student feedback to encourage future participants.
4. The "Review Management" button is broken: When Ben attempts to review or approve testimonials, the system fails to load the reviews.

5. Ben cannot manage appropriate feedback, and inappropriate testimonials get displayed publicly, damaging Ben's reputation.
6. The "Display Testimonial" button malfunctions, failing to display any testimonials. Positive testimonials are not highlighted on Ben's profile, leading to a lack of endorsements and decreased interest from potential new students.
7. The "Anonymous Feedback" button fails completely: Jasmine tries to leave anonymous feedback, but the system logs their name with the review.
8. This discourages honest feedback and makes Jasmine hesitant to provide any more testimonials, leaving Ben without valuable insights into their teaching effectiveness.

#### **Alternative Scenario 1:**

1. Ben encounters significant delays and security warnings when trying to access the login page.
2. After finally logging in, the dashboard loads slowly and incompletely, with key buttons like the "Testimonial Submission" non-functional.
3. The "Testimonial Submission" button leads to a system crash each time Jasmine attempts to write a testimonial.
4. Any effort to recover the system fails, leaving Jasmine unable to submit any feedback. This lack of testimonials makes it difficult for Ben to gather positive reviews.
5. The "Review Management" button displays incorrect testimonials. When Ben tries to approve feedback, the system fails to save Ben's approvals.
6. The "Display Testimonial" button is unreliable, Positive testimonials do not appear on Ben's profile. Ben's profile appears empty and poorly maintained, reducing the likelihood of attracting new students.
7. The "Anonymous Feedback" button logs names unintentionally: Jasmine's attempts to leave anonymous feedback result in their identity being displayed with the testimonial, leading to a breach of trust.
8. This error discourages students from leaving candid reviews, causing Ben to lose out on valuable, honest feedback.

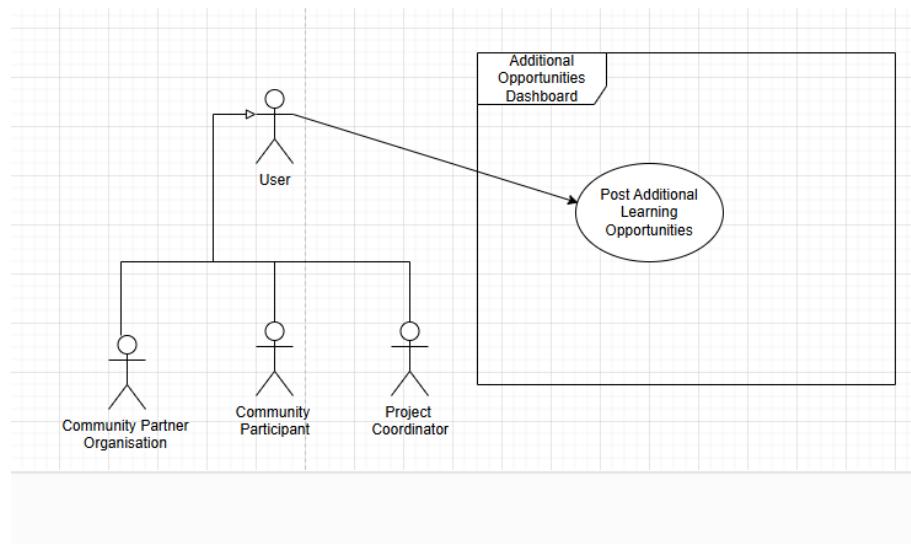
#### **Alternative Scenario 2:**

1. The "Testimonial Submission" button is unresponsive: Jasmine tries to submit a testimonial, but the button fails to work or leads to an error page.

2. This prevents Jasmine from providing feedback, and Ben is left without any testimonials to encourage future participants. The “Review Management” feature is malfunctioning.
3. When Ben attempts to manage testimonials, the system fails to save any changes, and it randomly deletes approved testimonials.
4. Inappropriate content is displayed, harming Ben’s reputation and professional image. The “Display Testimonial” button is completely non-functional: Testimonials are not shown on Ben’s profile.
5. This failure makes Ben’s profile look unprofessional and unappealing to potential students.
6. The “Anonymous Feedback” feature misfires: Jasmine’s anonymous feedback is mistakenly attributed to the wrong person. Additionally, the feedback that is displayed is incorrect, leading to misunderstandings and mistrust between Ben and the students.
7. This failure results in a lack of honest feedback and a missed opportunity for improvement.

## Additional Learning Opportunities

### UML



*Figure 13.1: Additional Learning Opportunities UML*

## Description

The Additional Learning Opportunities use case enables the user to view, approve and upload additional learning opportunities sent by the project sponsors.

### User Interface:

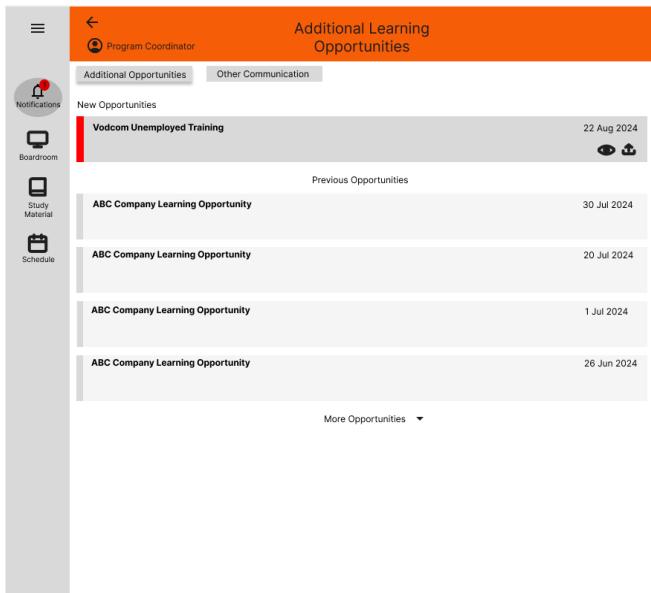


Figure 13.2: Additional Learning Opportunities UI

## Step-by-Step

1. The user must click the boardroom button on the left navigation bar inside the “Project Coordinator” dashboard.
2. The application will direct the coordinator to the “Boardroom” dashboard.
3. The coordinator must click the notifications button located on the left navigation bar inside the boardroom dashboard.
4. The application will display a list of all the additional learning opportunities that have been sent by the project sponsors.
5. The coordinator must click the view button to view the document the sponsor sent.
6. The application will display the document in view mode.
7. The coordinator must click the exit button to exit the documentation.
8. The application will redirect the coordinator back to the application.
9. The coordinator must click the upload button to upload the documentation onto the application.
  - a. If the coordinator does not approve of that learning opportunity, the coordinator must opt to not click the upload button.

- b. As part of a separate functionality, the coordinator will be able to send a message to the sponsor to communicate their reason for not approving the opportunity.
10. The application will share the document to the dashboards of the students and community members and send messages to their dashboards communicating that a new learning opportunity has been uploaded.

### **Best Case Scenario**

1. Ben clicks the boardroom button on the left navigation bar inside the “Project Coordinator” dashboard.
2. The application directs Ben to the “Boardroom” dashboard.
3. Ben clicks the notifications button located on the left navigation bar inside the boardroom dashboard.
4. The application displays a list of all the additional learning opportunities that have been sent by the project sponsors.
5. Ben clicks the view button to view the document the sponsor sent.
6. The application displays the document in view mode.
7. Ben clicks the exit button to exit the documentation.
8. The application redirects Ben back to the list of learning opportunities.
9. Ben clicks the upload button to upload the learning opportunity document onto the application.
10. The application shares the document to the dashboards of the students and community members and sends messages to their dashboards communicating that a new learning opportunity has been uploaded.

### **Worst Case Scenario**

1. Ben clicks the boardroom button on the left navigation bar inside the “Project Coordinator” dashboard.
2. The application directs Ben to the “Boardroom” dashboard.
3. Ben clicks the notifications button located on the left navigation bar inside the boardroom dashboard.
4. The application displays a list of all the additional learning opportunities that have been sent by the project sponsors.
5. Ben clicks the view button to view the document the sponsor sent.

6. The application displays the document in view mode.
7. Ben clicks the exit button to exit the documentation.
8. The application redirects Ben back to the list of learning opportunities.
9. Ben clicks the upload button to upload the learning opportunity document onto the application.
10. The application shares the document to the dashboards of the students and community members and sends messages to their dashboards communicating that a new learning opportunity has been uploaded.

**Alternative scenario 1:**

1. Ben clicks the boardroom button on the left navigation bar inside the “Project Coordinator” dashboard.
2. The application directs Ben to the “Boardroom” dashboard.
3. Ben clicks the notifications button located on the left navigation bar inside the boardroom dashboard.
4. The application displays a list of all the additional learning opportunities that have been sent by the project sponsors.
5. Ben clicks the view button to view the document the sponsor sent.
6. The application displays the document in view mode.
7. Ben clicks the exit button to exit the documentation.
8. The application redirects Ben back to the list of learning opportunities.
9. Ben does not approve of the learning opportunity and decides not to upload and send the document.
10. As part of a separate functionality namely “Boardroom”, Ben will be able to send feedback, during a boardroom meeting with the sponsors, on his reason for disapproving the opportunity.

**Alternative scenario 2:**

1. Ben clicks the boardroom button on the left navigation bar inside the “Project Coordinator” dashboard.
2. The application directs Ben to the “Boardroom” dashboard.
3. Ben clicks the notifications button located on the left navigation bar inside the boardroom dashboard.
4. The application displays a list of all the additional learning opportunities that have been sent by the project sponsors.
5. Ben clicks the view button to view the document the sponsor sent.

6. The application displays the document in view mode.
7. Ben clicks the exit button to exit the documentation.
8. The application redirects Ben back to the list of learning opportunities.
9. Ben clicks the upload button and clicks the checkbox next to community members to only share the document with the community members.
10. The application shares the document to the dashboards of the students and community members and sends messages to their dashboards communicating that a new learning opportunity has been uploaded.

## Upload Promotional and Marketing Material

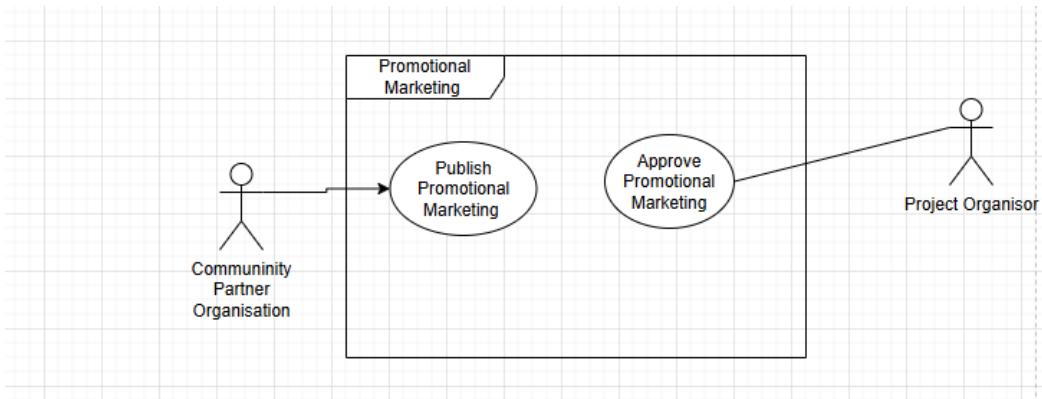


Figure 14.1: Testimony and Review UML

### Brief Description

This functionality allows user (community partners) to share or upload their promotional and marketing material on the system for brand exposure.

### User Interface:

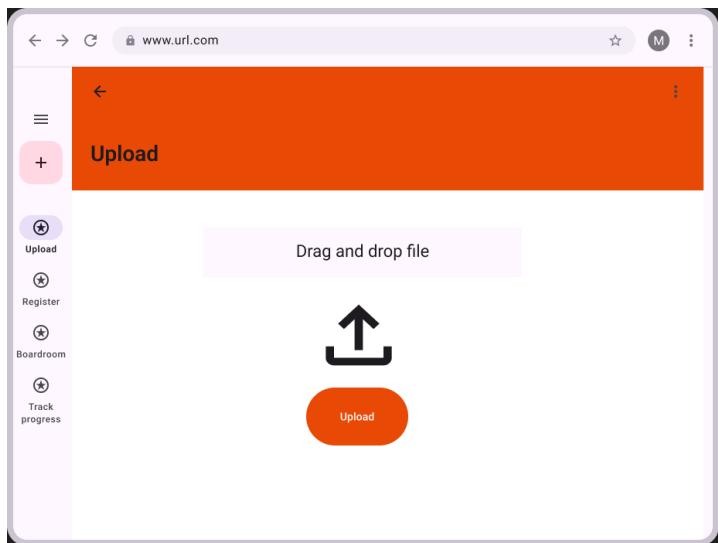


Figure 14.2: Testimony and Review UI

### Step-by-Step

1. After successfully logging in onto the system, community partner can click on the boardroom icon.
2. Boardroom page will open and display available functionalities.
3. Click on upload button in the boardroom page to upload promotional and marketing material.
4. Select material to upload.
5. Click upload to upload selected material.

### Project coordinator:

6. The user (Project coordinator) approves the uploaded material.
7. Selected material will be successfully uploaded by the system.

### Best Case Scenario

1. Amanda wants to upload promotional and marketing material for her community organisation, she must first login to the system successfully.
2. Amanda must click on navigation panel on left side of website page or app, then she must click on upload icon.

3. This will display a page where Amanda can upload a pictures, documents, audio, or videos of her organization for promotional and marketing purposes.

### **Worst Case Scenario**

1. Amanda wants to upload promotional and marketing material for her community organisation after successfully registering into the system.
2. Amanda will click on upload icon on the navigation panel.
3. When the upload page opens there is no functionality upload meaning Amanda cannot select any type of file to upload onto the system for promotional and marketing purposes.

### **Alternative scenario 1:**

1. Amanda wants to upload promotional and marketing material for her community organization after successfully registering into the system.
2. Amanda must click on upload icon from the navigation panel.
3. When upload page opens up Amanda can only upload picture files on to the system meaning a small part of the upload functionality is working properly

### **Alternative Case Scenario**

1. Amanda wants to upload promotional and marketing material for her community organization after successfully registering into the system.
2. Amanda must click on upload icon from the navigation panel.
3. Devin (Project coordinator) views the uploaded material but does not publish the material by clicking on the approve button.

## **Design and add curriculum content**

## UML

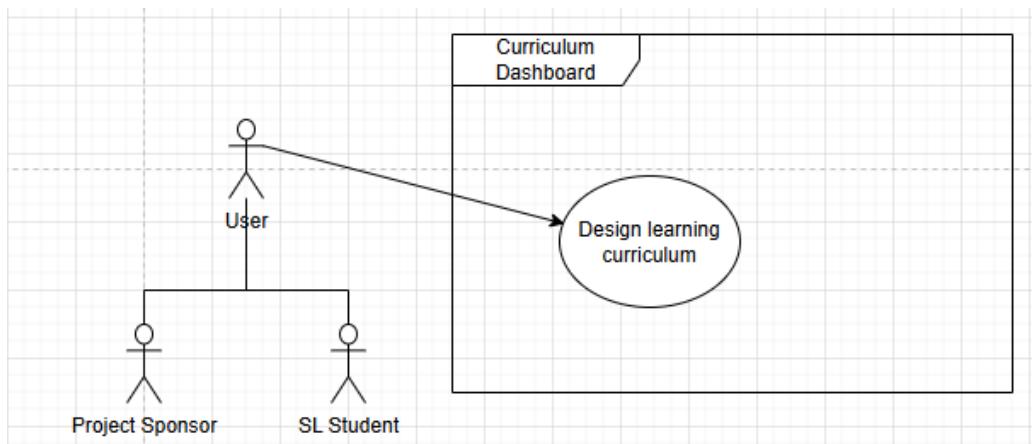


Figure 15.1: Design and add curriculum UML

## Brief Description

This functionality is a portal that allows the user to upload files from their computer unto the system making it available to the participant. It allows the user to upload, name, and describe the file.

## User Interface

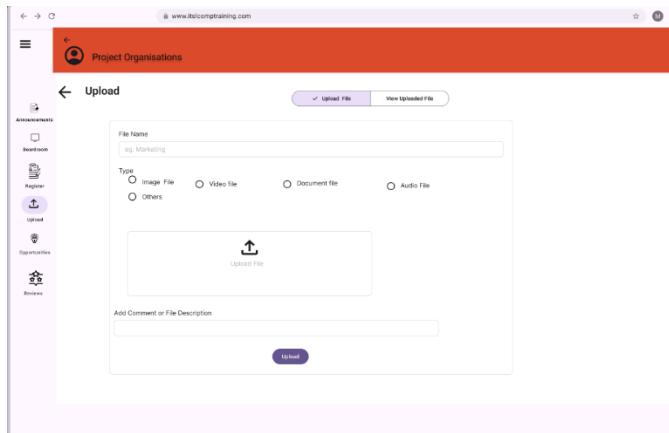


Figure 15.2: Design and add curriculum UI

## Step by Step.

1. User clicks on upload on the left navigation section.
2. System produces an Upload panel pops up

3. To upload course contents User can
  - 3.1. Click on the file name textbox to rename the uploaded file
  - 3.2. Choose what type of file it is by clicking one of the five radio buttons
  - 3.3. Click on the upload box, to trigger a dialog that allows the user to browse files on the user's computer.
  - 3.4. Add comments by clicking on the "Add Comment or File Description" text box, to add a comment or to describe the file.
  - 3.5. Click on the upload button to upload.
  - 3.6. Click on "View Uploaded File" to view past and most recent uploads.

#### **Best case scenario:**

1. Jenny clicks on upload on the left navigation section.
2. System produces an Upload panel pops up
3. To upload course contents Jenny can
  - 3.1. She clicks on the file name textbox to rename the uploaded file
  - 3.2. Choose what type of file it is by clicking one of the five radio buttons
  - 3.3. She clicks on the upload box, to trigger a dialog that allows the user to browse files on her computer.
  - 3.4. She adds comments by clicking on the "Add Comment or File Description" text box, to add a comment or to describe the file.
  - 3.5. She clicks on the upload button to upload.
  - 3.6. She clicks on "View Uploaded File" to view past and most recent uploads.

#### **Worst case scenario**

1. Jenny clicks on upload on the left navigation section.
2. System produces an Upload panel pops up
3. To upload course contents Jenny can
  - 3.1. She clicks on the file name textbox to rename the uploaded file
  - 3.2. Choose what type of file it is by clicking one of the five radio buttons
  - 3.3. She clicks on the upload box, to trigger a dialog that allows the user to browse files on her computer.
  - 3.4. Jenny did not save the file in the correct location and will thus be unable to upload the anything on to the system.

#### **Alternative scenario 1:**

1. Jenny clicks on upload on the left navigation section.
2. System produces an Upload panel pops up

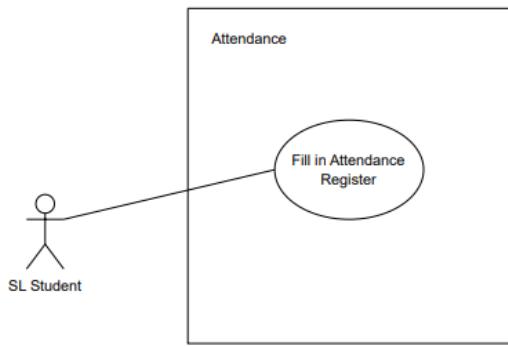
3. To upload course contents Jenny can
  - 3.1. She clicks on the file name textbox to rename the uploaded file
  - 3.2. Choose what type of file it is by clicking one of the five radio buttons
  - 3.3. She clicks on the upload box, to trigger a dialog that allows the user to browse files on her computer.
  - 3.4. She adds comments by clicking on the “Add Comment or File Description” text box, to add a comment or to describe the file.
  - 3.5. She clicks on the upload button to upload.
  - 3.6. She clicks on “View Uploaded File” to view past and most recent uploads.
  - 3.7. She then sees that she has uploaded the wrong file onto the system.

**Alternative scenario 2:**

1. Jenny clicks on upload on the left navigation section.
2. System produces an Upload panel pops up
3. To upload course contents Jenny can
  - 3.1. She clicks on the file name textbox to rename the uploaded file
  - 3.2. Choose what type of file it is by clicking one of the five radio buttons
  - 3.3. She clicks on the upload box, to trigger a dialog that allows the user to browse files on her computer.
  - 3.4. She adds comments by clicking on the “Add Comment or File Description” text box, to add a comment or to describe the file.
  - 3.5. She types in too many words and the system produces an error, prompting her to not exceed the word count of 100 words.

## Attendance

**UML**

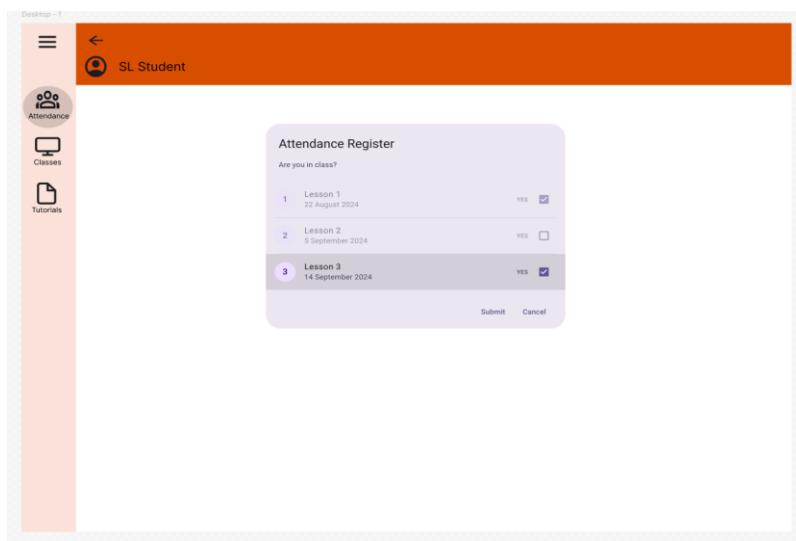


*Figure 16.1: Attendance UML*

### Brief Description

The attendance use case allows the SL Student to mark the attendance register on the application.

### User Interface:



*Figure 16.2: Attendance UI*

### Step-by-Step

1. The user must click the attendance button on the left side navigation bar within the "Student" dashboard.
2. The application will display a dialog box with the heading Attendance Register. The dialog box will display the current class/lesson number, topic of the lesson, the current date, a tab for entering student number along with a question along with check boxes that the student can tick to indicate if they are present. Lastly will be submit and cancel buttons.

3. The click inside the yes checkbox to indicate they are present, and then click submit to save their answer.
4. The application will save the users answer and update the systems database to mark the student as present for that lesson.
5. The application will then display a message indicating that the student has marked the register successfully.

### **Best Case Scenario**

1. Ben clicks on the attendance button on the left side navigation bar within the "Student" dashboard.
2. The application will display a dialog box with the heading Attendance Register. The dialog box will display the current class/lesson number, topic of the lesson, the current date, a tab for entering student number along with a question along with check boxes that the student can tick to indicate if they are present. Lastly will be submit and cancel buttons.
3. Ben clicks inside the yes checkbox to indicate that he is present, and then click submit to save his answer.
4. The application saves Ben's answer and updates the systems database to mark Ben as present for that lesson.
5. The application displays a message indicating that Ben has marked the register successfully.

### **Worst Case Scenario**

1. Ben clicks on the attendance button on the left side navigation bar within the "Student" dashboard.
2. The application displays a dialog box with the heading Attendance Register. The dialog box will display the current class/lesson number, the current date, a textbox for entering student number along with a question with check a box that the student can tick to indicate if they are present. Lastly will be submit and cancel buttons.
3. Ben clicks yes checkbox to indicate they are present, and then clicks submit to save their answer.
4. The application saves Ben's answer inside the applications database.
5. The application redirects Ben back to the "Student" dashboard without displaying any message that his answer was submitted successfully or that he marked the register successfully.

### **Alternative scenario 1**

1. Ben clicks on the attendance button on the left side navigation bar within the "Student" dashboard.
2. The application displays a dialog box with the heading Attendance Register. The dialog box will display the current lesson number, the current date, a textbox for entering student number along with a question with check a box that the student must tick to indicate if they are present. Lastly will be submit and cancel buttons.

3. Ben clicks the submit button without clicking the check box to answer the question and mark that he is present.
4. The application displays a message stating that Ben must click the check box to mark the register before submitting.
5. Ben clicks the checkbox.
6. The application saves Ben's answer and updates the systems database to mark Ben as present for that lesson.
7. The application displays a message indicating that Ben has marked the register successfully.

#### **Alternative scenario 2**

1. Ben clicks on the attendance button on the left side navigation bar within the "Student" dashboard.
2. The application displays a dialog box with the heading Attendance Register. The dialog box will display the current lesson number, the current date, a textbox for entering student number along with a question with check a box that the student must tick to indicate if they are present. Lastly will be submit and cancel buttons.
3. Ben clicks the checkbox to indicate that he is present. Then Ben clicks on the cancel button.
4. The application displays a message that Ben did not mark the register then redirects Ben back to the "Student" dashboard.
5. Ben clicks on the attendance button again.
6. The application displays the dialog box again.
7. Ben clicks the checkbox indicating that he is present then he clicks the submit button.
8. The application saves Ben's answer and updates the systems database to mark Ben as present for that lesson.
9. The application displays a message indicating that Ben has marked the register successfully.

# Software Project Management Plan

## Software Project Management Plan

### Overview

#### 1.1 Project summary

**1.1.1** The objective of this project is to develop a system that will enhance the management of participants' data, feedback, and communication for the Information Technology Service Learning (ITSL) project. The system will help streamline processes related to communication, training schedules, and participant progress tracking for community members involved in ITSL.

#### 1.1.2 Assumptions and constraints

- Participants will have access to smartphones and computers.
- All community members will have access to WhatsApp and other relevant messaging platforms for communication.
- Training and support will be conducted primarily in English.

### Constraints

- The system must be compatible with MS Office and allow integration with existing software, such as the UFS student management systems.
- The system must operate within the given budget.
- The product must be user-friendly and intuitive for both participants and service-learning students.
- The project must adhere to security standards; as sensitive personal information is involved.

#### 1.1.3 The project must be completed within 250 hours after commencement. The primary deliverables include:

- A communication system through the website and WhatsApp
- Participant management system such as database for tracking participant data, attendance, and progress

- Integration with MS Office SLP functionalities (MS Word, MS Excel training modules)
- Report generation and feedback system for participants and instructors
- User-friendly interface for tracking and grading participants' progress

## 2. Definitions and Acronyms

- UFS: The University of the Free State
- ITSL: Information Technology Service-Learning
- SLPs: Short Learning Programs (MS Word, MS Excel)

## 3. Project Organization

### 3.1 External Interfaces

The ITSL system will interact with several external tools and platforms such as WhatsApp Integration for communication between participants and service-learning students. MS Office integration will also be in use for training and document sharing. UFS student management system for tracking participants' progress and reporting.

### 3.2 Internal Structure

- **Development Team:**
  - Team leader: L Letsie
  - Developers: T Poolo, T Motlhale, N Moloi, P Sophilase, B Rampora, L Letsie and D Oamen
- **Client Interaction Team:**
  - Client interaction team will consist of L Letsie, T Poolo, T Motlhale, N Moloi, P Sophilase, B Rampora and D Oamen
- **Module Integration Team:**
  - The module integration team will consist of L Letsie, T Poolo, T Motlhale, N Moloi, P Sophilase, B Rampora and D Oamen
- **Testing Team:**
  - Each developer will test other members' modules and ensure their own modules function correctly.

### **3.3 Roles and Responsibilities**

- L Letsie: Team leader, responsible for module integration, system architecture, and overall project quality.
- N Moloi: Lead developer for participant management and communication modules.
- P Sophilase and B Rampora : Developer responsible for training material management and report generation modules.
- T Poolo and T Motlhale: Responsible for feedback collection and data analysis functionalities.
- D Oamen: Developer responsible for system interface and user experience.

## **4. Managerial Process Plans**

### **4.1 Start-up Plan**

#### **4.1.1 Estimation Plan**

This project's estimation was made using data from previous ITSL projects. The group decided on a development cost of R910,000 with a total effort of 250 man-hours using expert opinion and analogy.

#### **4.1.2 Staffing Plan**

The core development team consists of 7 members, led by L Letsie who will ensure that the project will meet all deadlines within budget. The entire team of T Poolo, T Motlhale, N Moloi, P Sophilase, B Rampora, L Letsie and D Oamen will be involved throughout all workflows of the project.

#### **4.1.3 Resource Acquisition Plan**

- Computers and testing devices will be provided by UFS IT services.
- Software required includes:
  - Visual Studio 2022
  - MS SQL Server 2019 for database management
  - MS Office 2023 (for integration and training content management)
  - WhatsApp Business API for communication integration

## **5.2 Work Plan**

### **5.2.1 Activity Schedule**

- **Requirements:** 50 hours, team gathers detailed requirements through meetings with UFS and community partners.
- **Analysis:** 75 hours, refining requirements and preparing specifications.
- **Design:** 100 hours, developers create design documents for the modules and test cases.
- **Implementation:** 25 hours, developing the system in phases, testing incrementally.
- **Testing:** Embedded within each phase.

### **5.2.2 Resource Allocation**

Each team member will use individual workstations with the necessary software. Testing will be done on both desktop and mobile platforms to ensure cross-device compatibility.

### **5.2.3 Budget Allocation**

- Requirements workflow: R 175,000
- Analysis workflow: R 262,000
- Design workflow: R 350,000
- Implementation workflow: R 1,000
- Resources: R 87,000

**Total Budget:** R 875,000

## **5.3 Quality Control Plan**

Testing will be ongoing throughout the project. Unit testing, integration testing, and user acceptance testing will be completed in each phase. Final testing will involve real-world scenarios with mock participants to ensure system robustness.

## **5.4 Risk Management Plan**

Regular backups of all code and data will be made to a cloud service in case of Data loss. Backup computers will be available in case of hardware failure issues . In the case of System failure during training, a rollback system will allow the previous version of the software to be reinstated quickly.

## **5.5 Project Close-out Plan**

After completion, the development team will focus on providing support to UFS and community partners. A final review will be held with stakeholders to ensure all deliverables are met.

# **6. Technical Process Plans**

## **6.1 Process Model**

The system uses Unified Process and applying an Iterative-and-Incremental life cycle model development approach.

## **6.2 Methods, Tools, and Techniques**

The workflows will be performed in accordance with the Unified Process. The product will be program will be written in the following languages and framework, ASP.NET, C#, JavaScript, HTML5, CSS3. Using the MS SQL Server 2019. Communication Integration will use WhatsApp Business API.

## **6.3 Infrastructure Plan**

The system will be developed using Visual Studio 2022 running on Windows 10 or later. MS SQL Server will be used for database management, and the team will also use WhatsApp Business API for communication integration.

## **6.4 Product Acceptance Plan**

Product acceptance will be achieved through iterative demonstrations and user acceptance testing (UAT). Each iteration will be reviewed with the stakeholders to ensure that the system meets the client's requirements and quality standards.

# **7. Additional Plans**

## **7.1 Security Plan**

User credentials are encrypted and stored in secured database. Access to the system will require authentication using unique usernames and passwords for each user.

## **7.2 Maintenance Plan**

For **12 months**, the team will provide corrective maintenance at no additional cost. This will cover bug fixes and performance issues. After this period, a separate contract will be drawn up for further maintenance and potential system enhancements

The first year of the system, the team will provide free corrective maintenance. Once the period lapse, a new maintenance contract should be established

# Quotation

DEV  
WORKS

INVOICE

**BILLED TO:**

ITSL Sponsors  
205 Nelson Mandela Dr, Park West,  
Bloemfontein, 9301

Invoice No. 12345  
11 August 2024

Item	Amount (R)
Requirements	175,000
Analysis	262,000
Design	350,000
Implementation	1,000
Resources	87,000
Overheard	15,000
<b>Subtotal</b>	<b>895,000</b>

Thank you!

**Tax (15%)** 133,500

**Total 1,028,500**

**PAYMENT INFORMATION**

Bank: FNB  
Account Name: DevWorks  
Account No.: 1254112356  
Pay by: 23 October 2024

DevWorks Consultants)  
72 Paul Kruger, Universitas

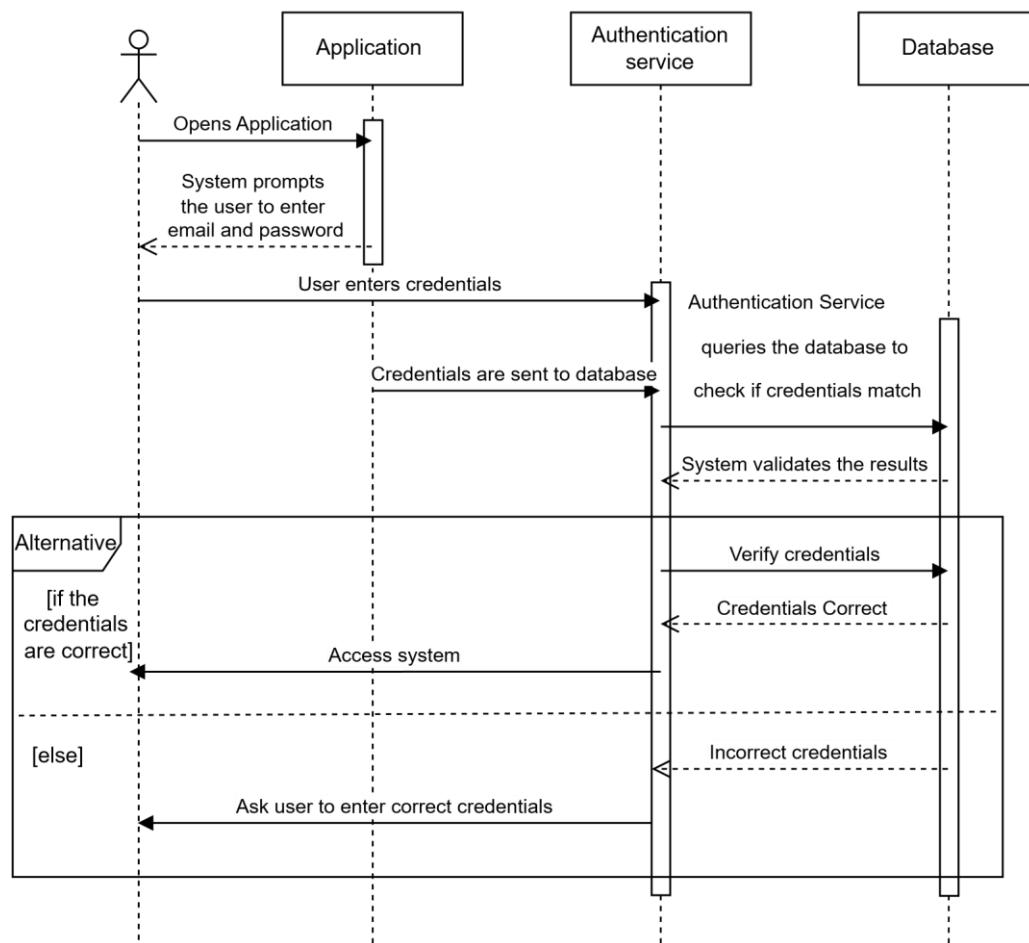
# Sequence Diagrams

Project Coordinator/SL Student /Community Partner organizations/Community Participant

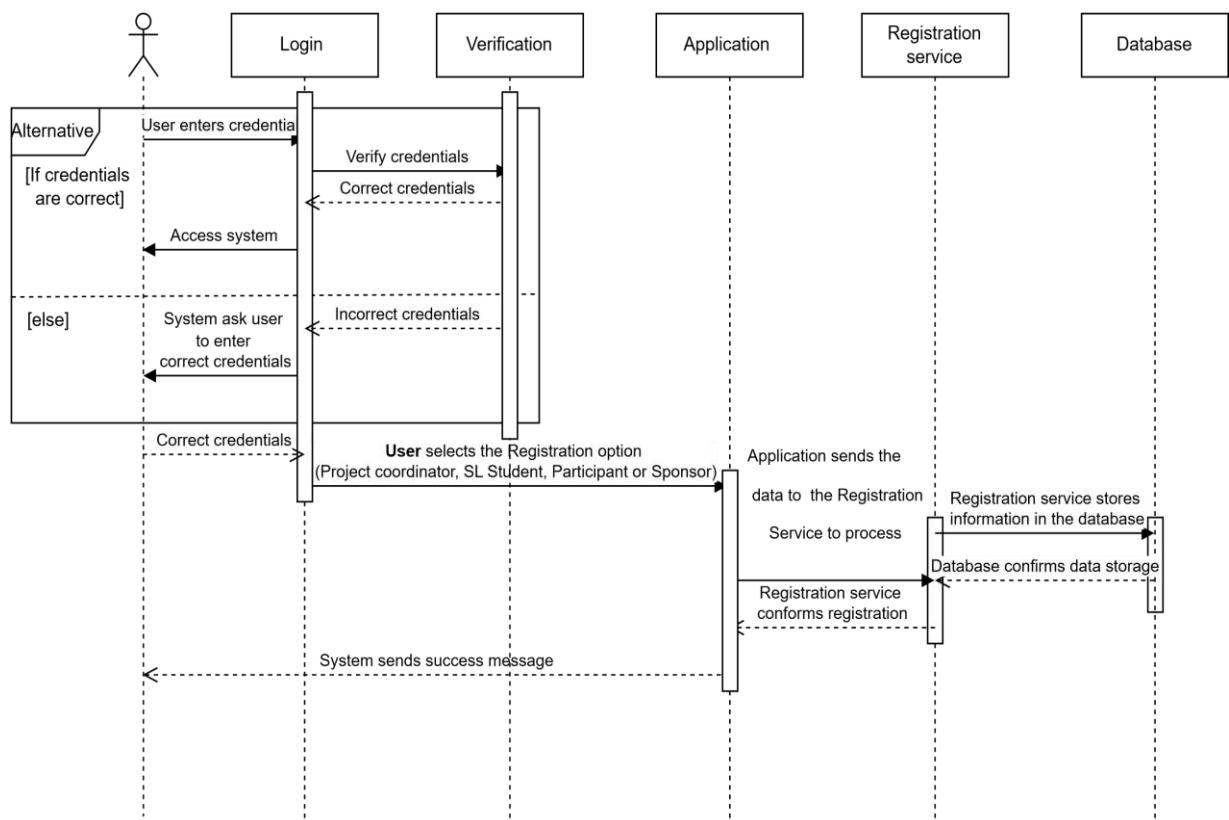
## Project Coordinator

### Authentication

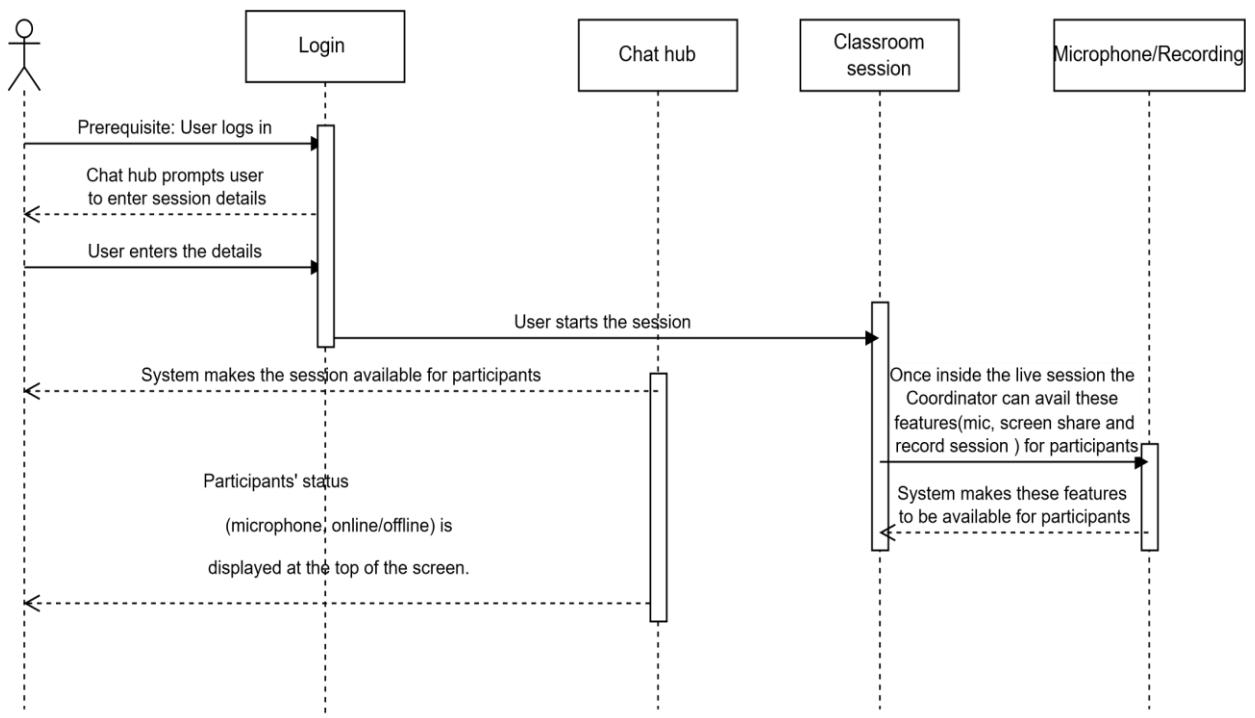
#### 1. Authentication



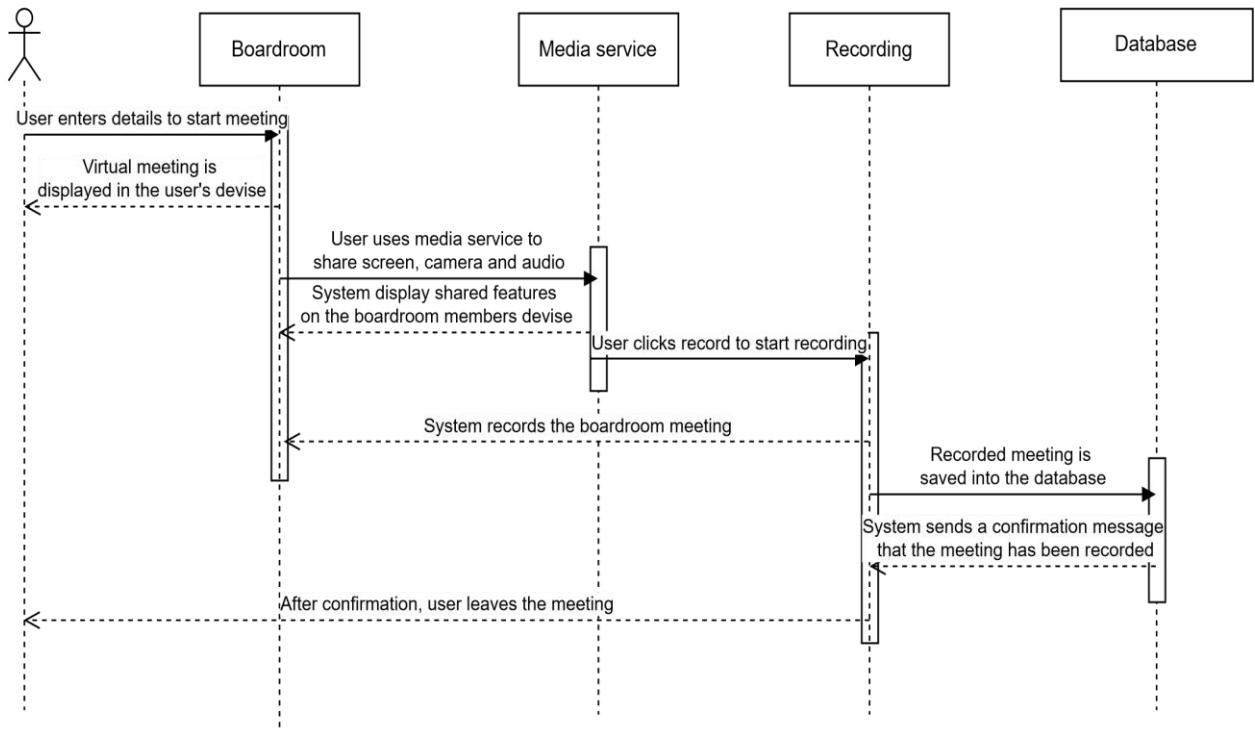
## Registration



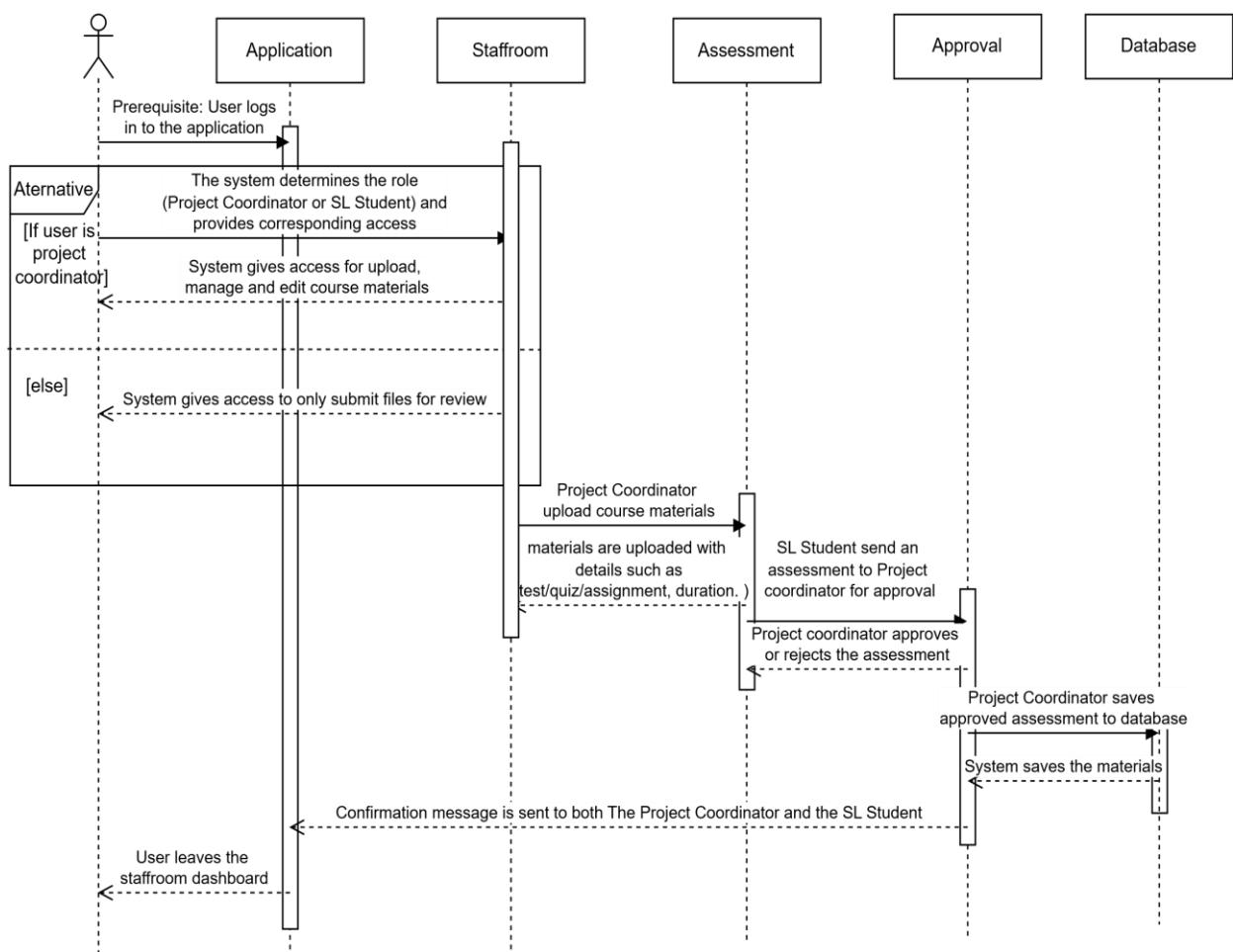
## Classroom Chat



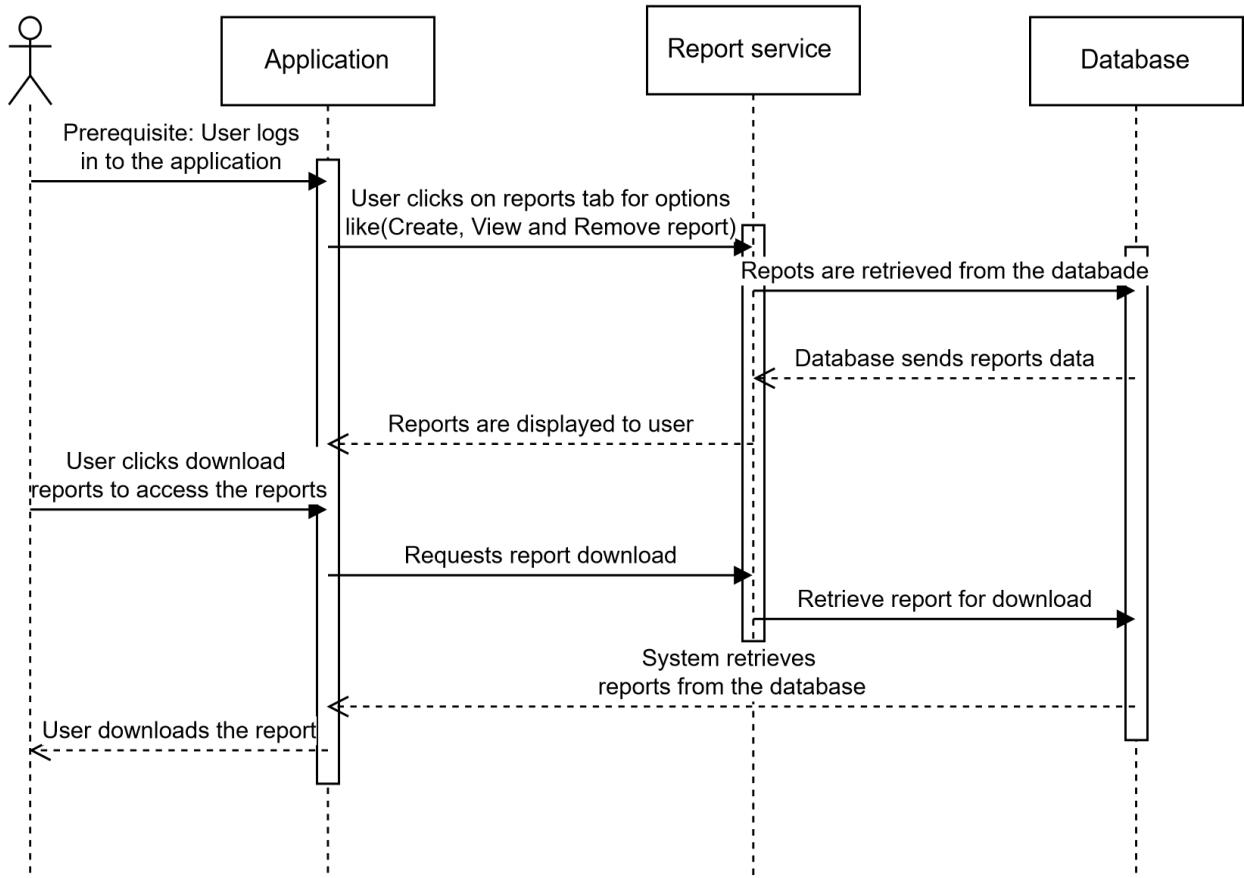
## Boardroom



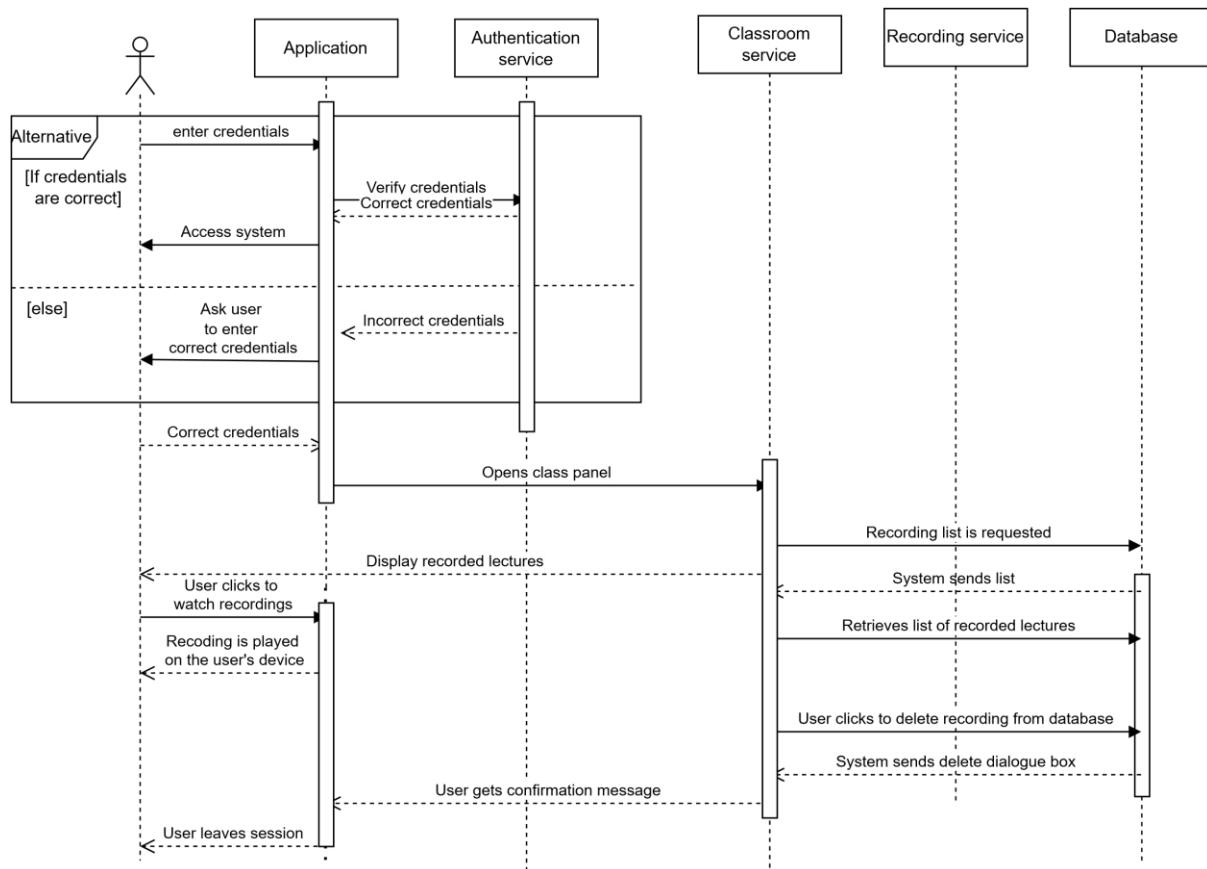
## Staffroom



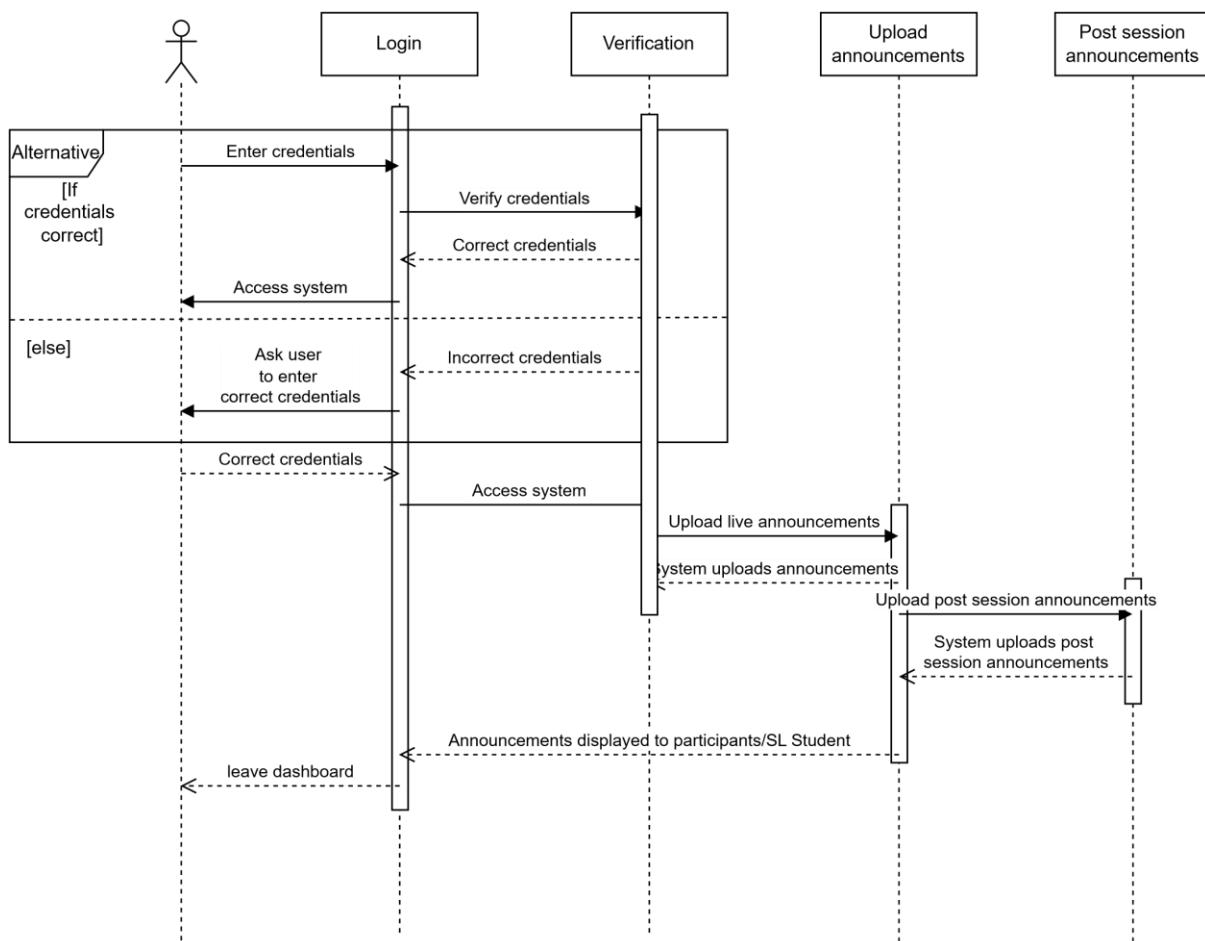
## Report statistics



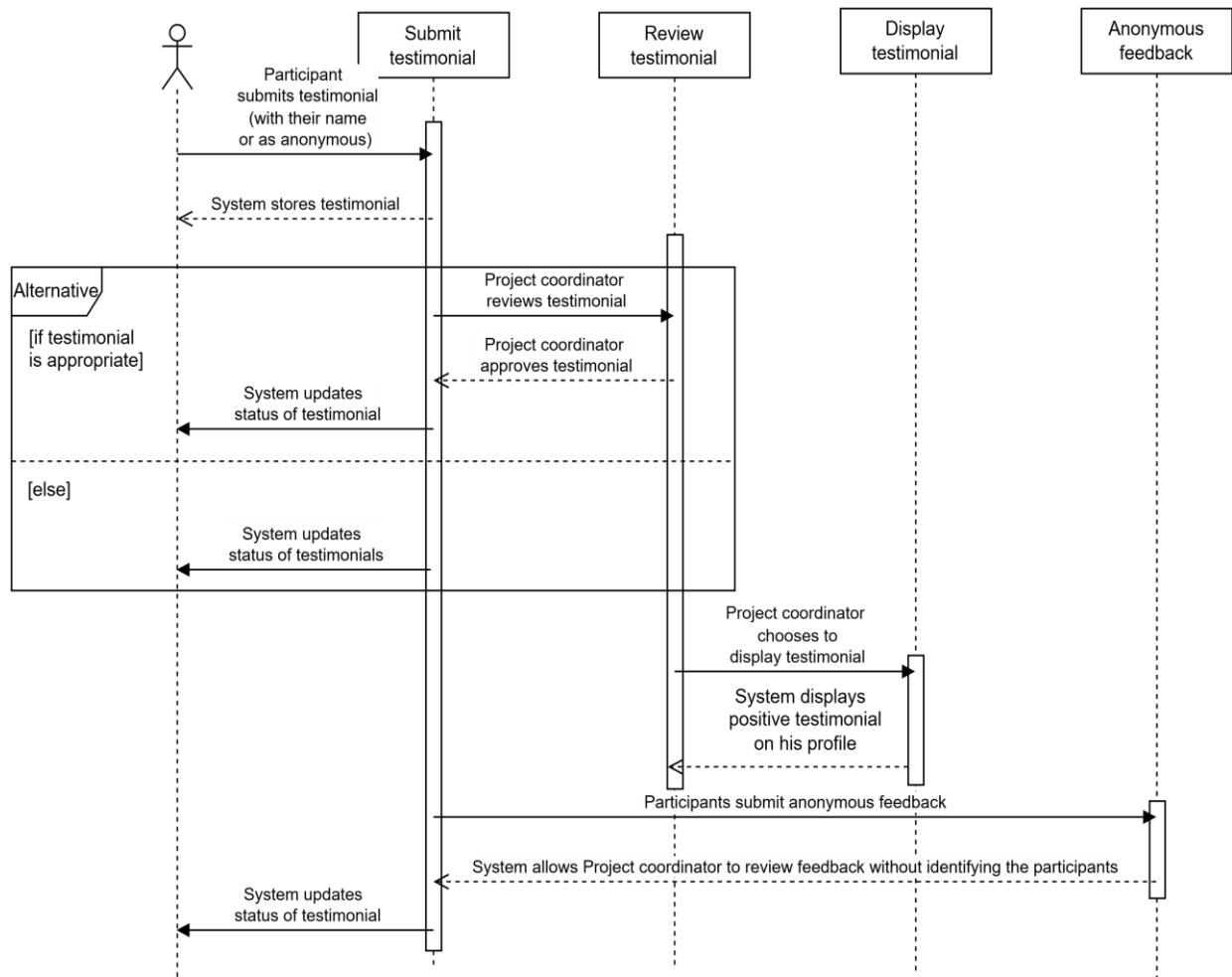
## Classroom + Recordings



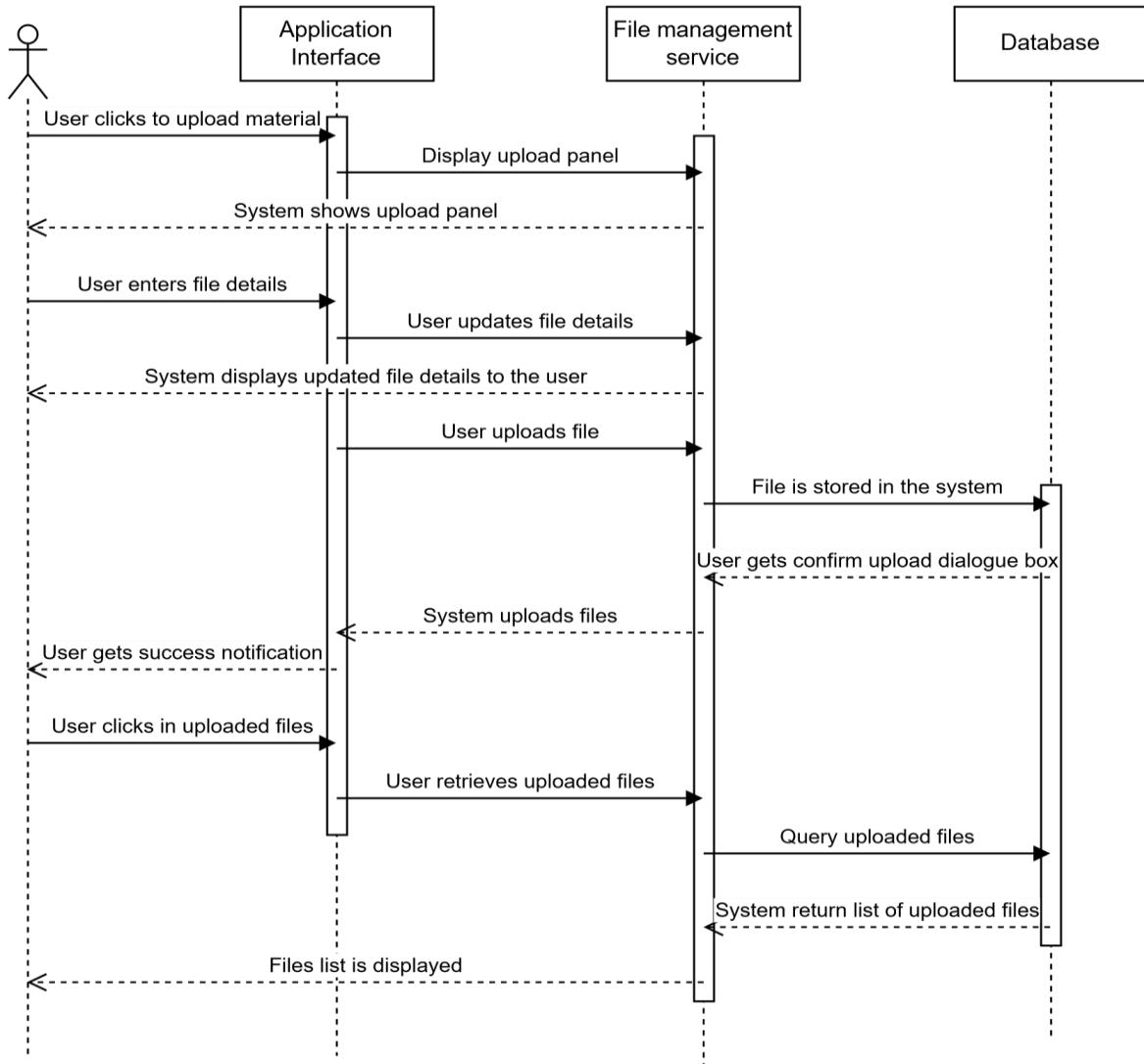
## Announcements



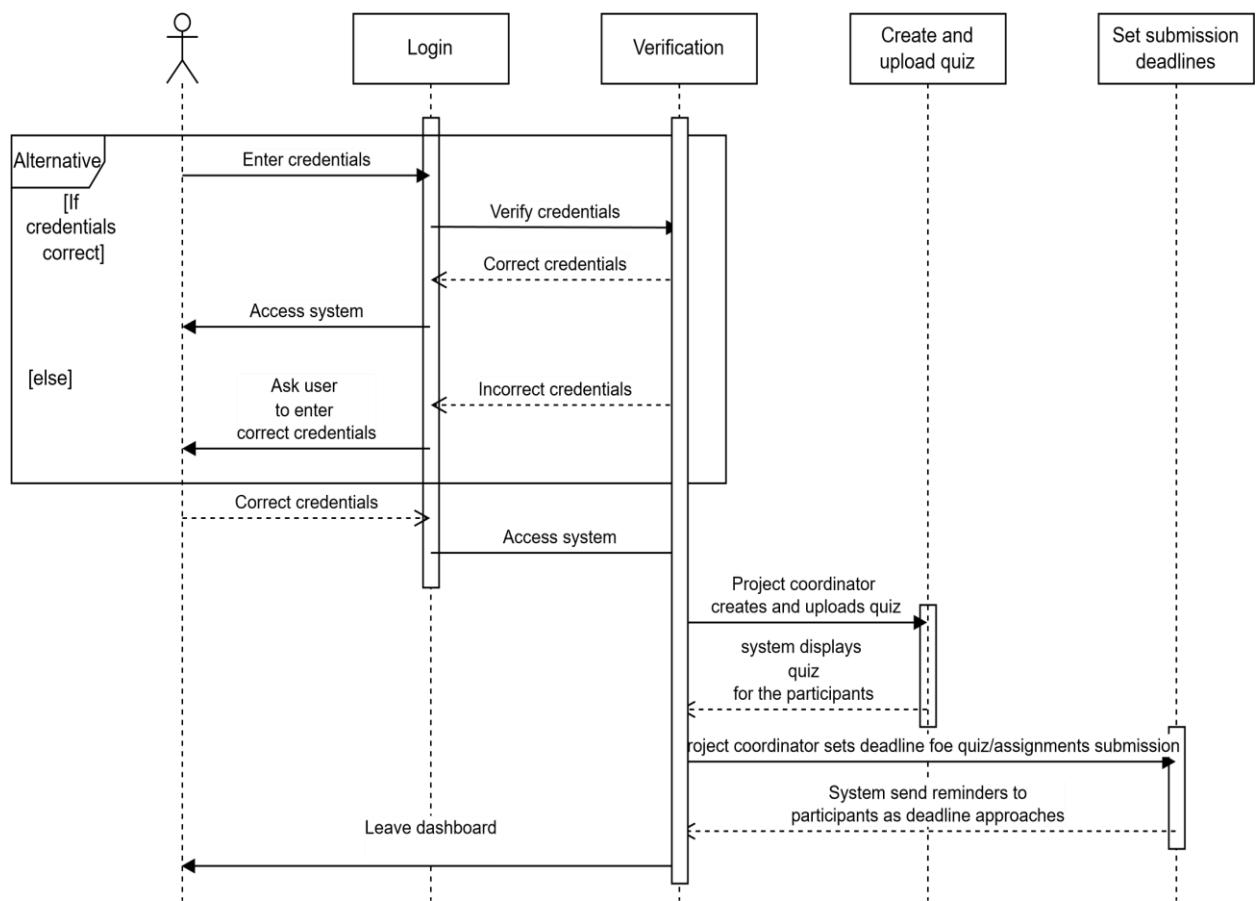
## Testimony/Review



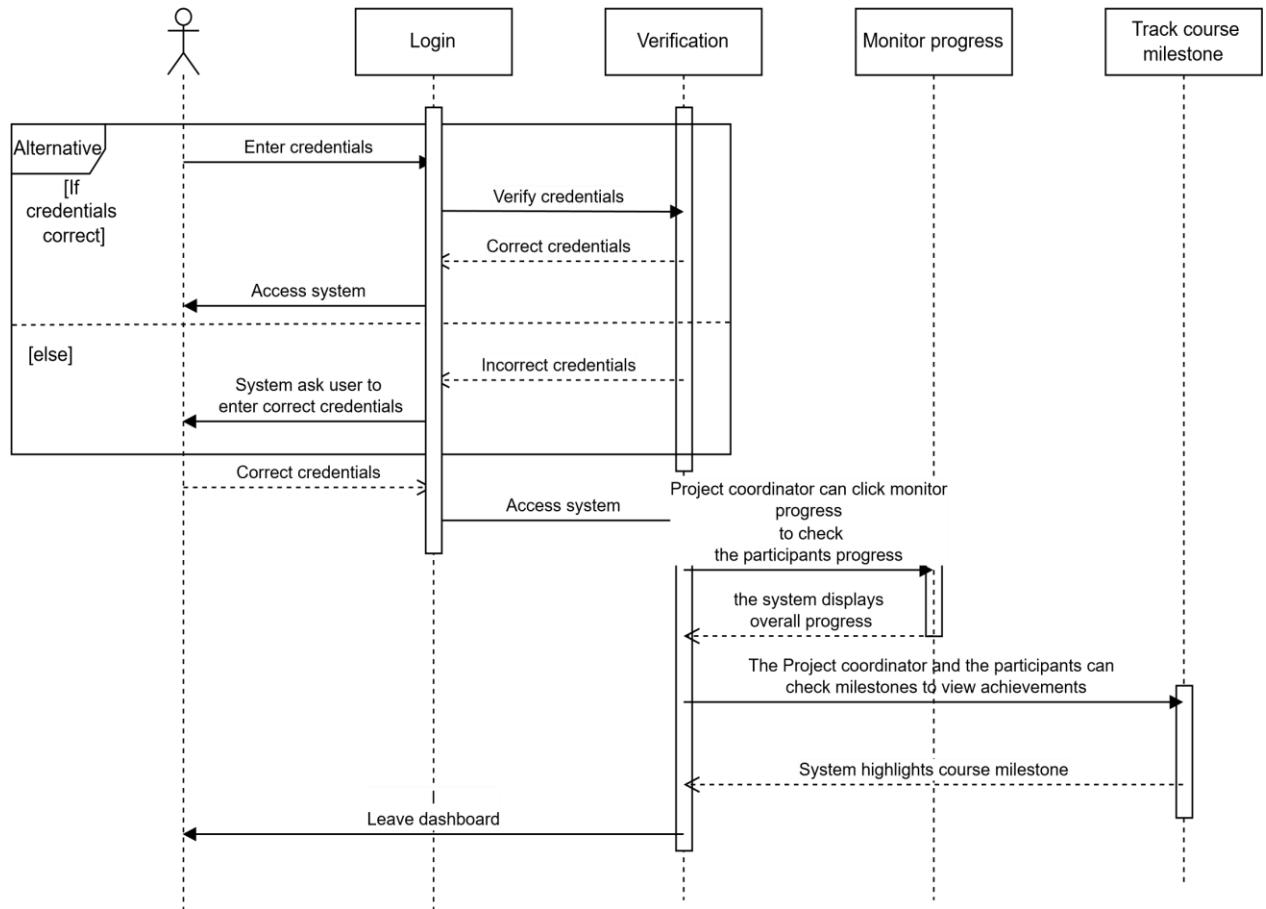
## Design and add curriculum



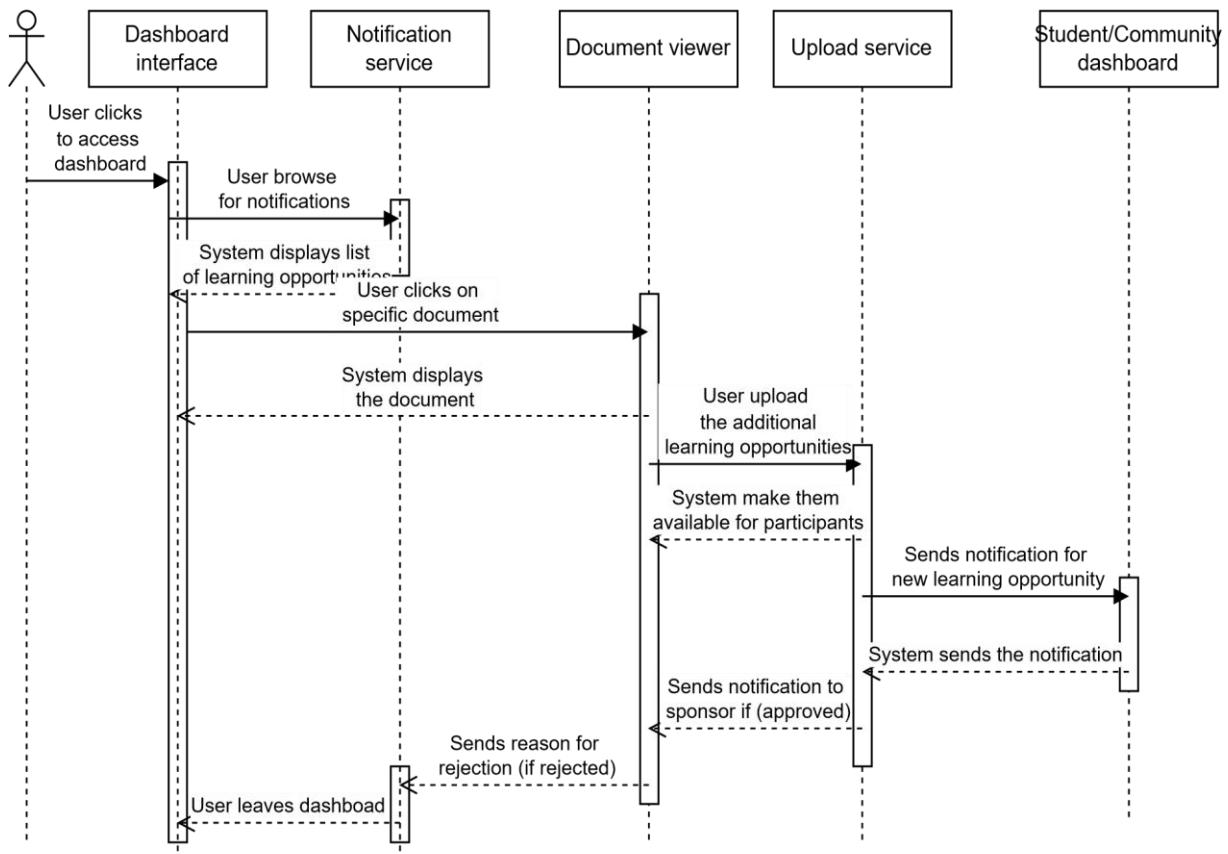
## Assessments/Quiz



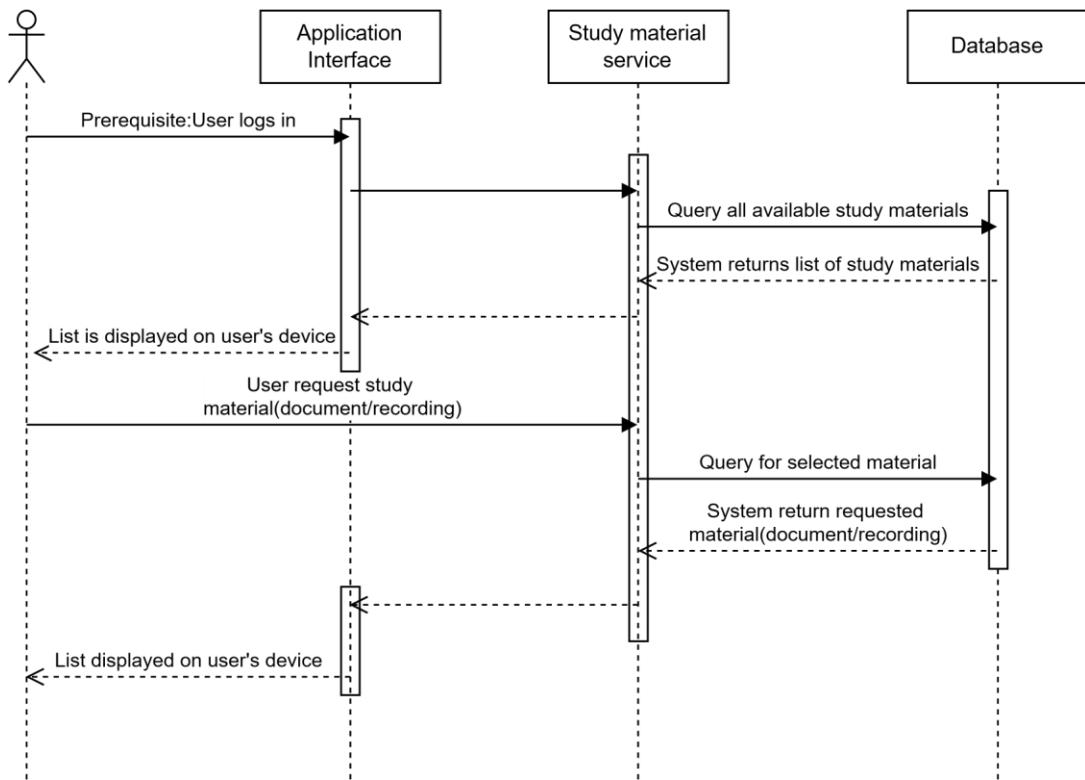
## Check course progress



## Additional learning opportunities

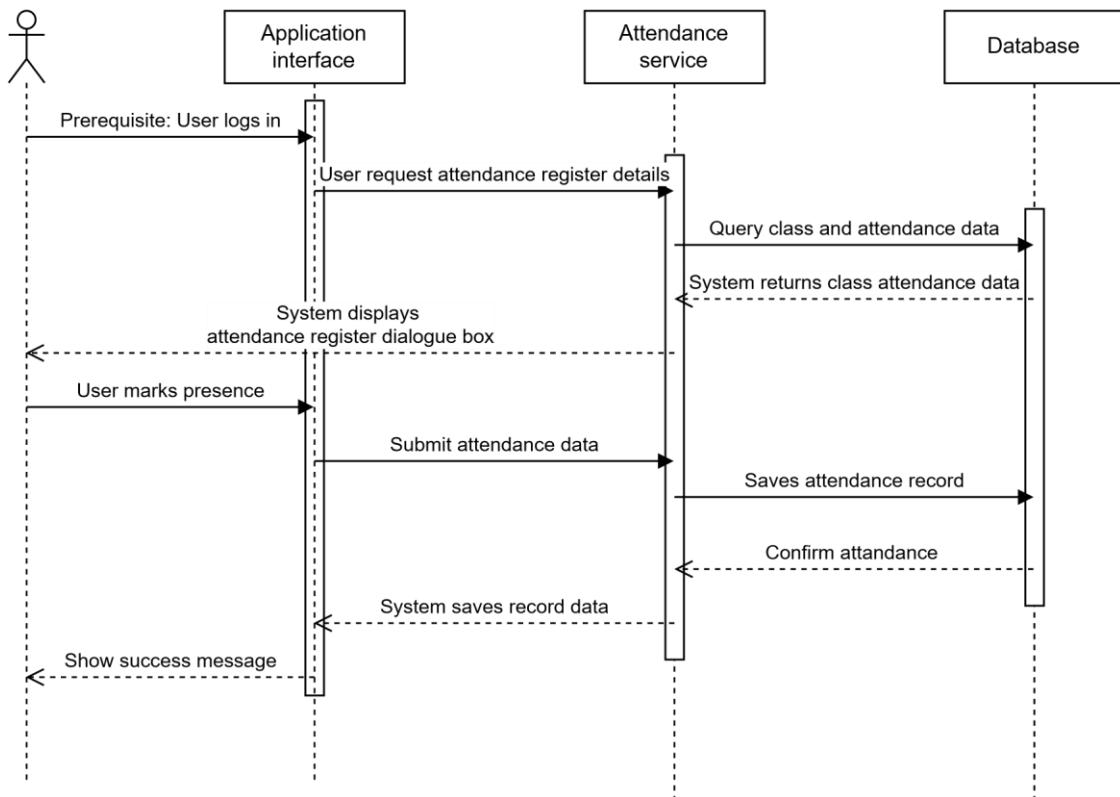


## Study material access

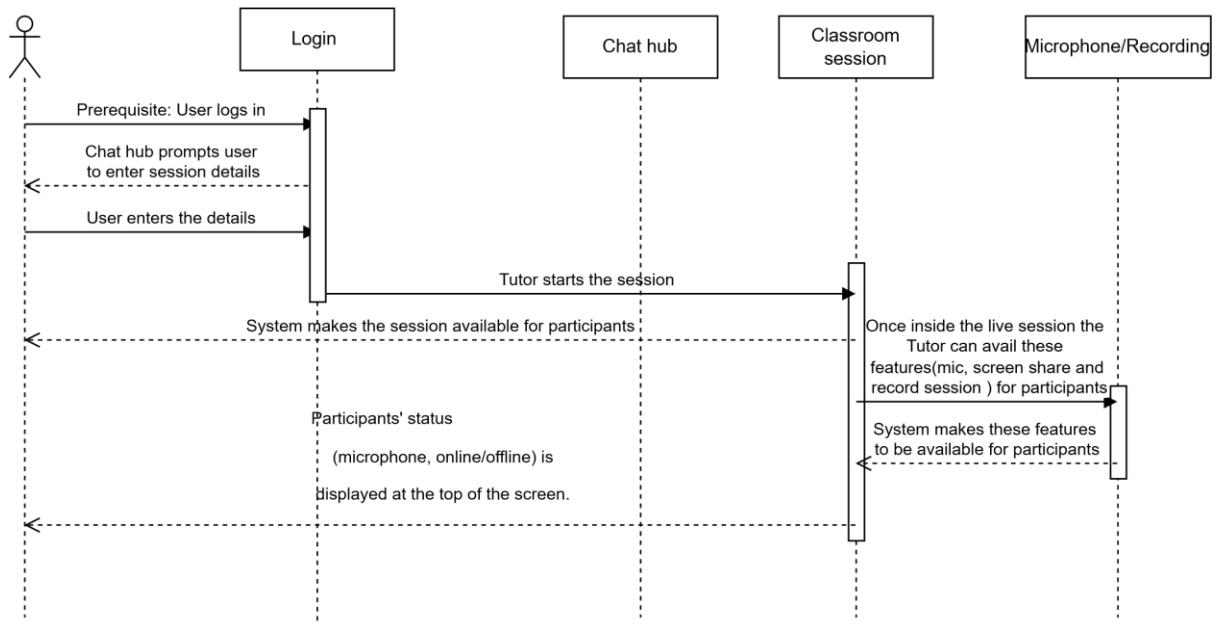


## SL Student

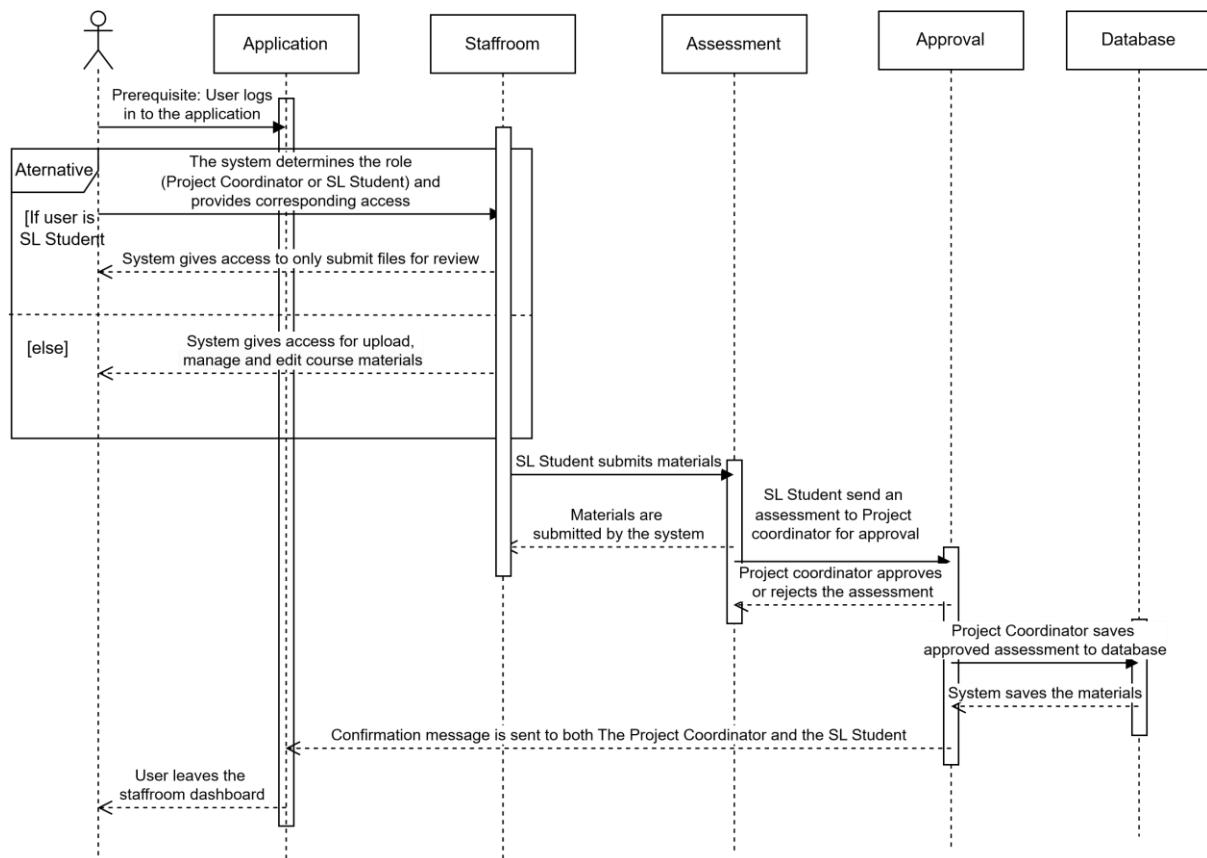
### Attendance



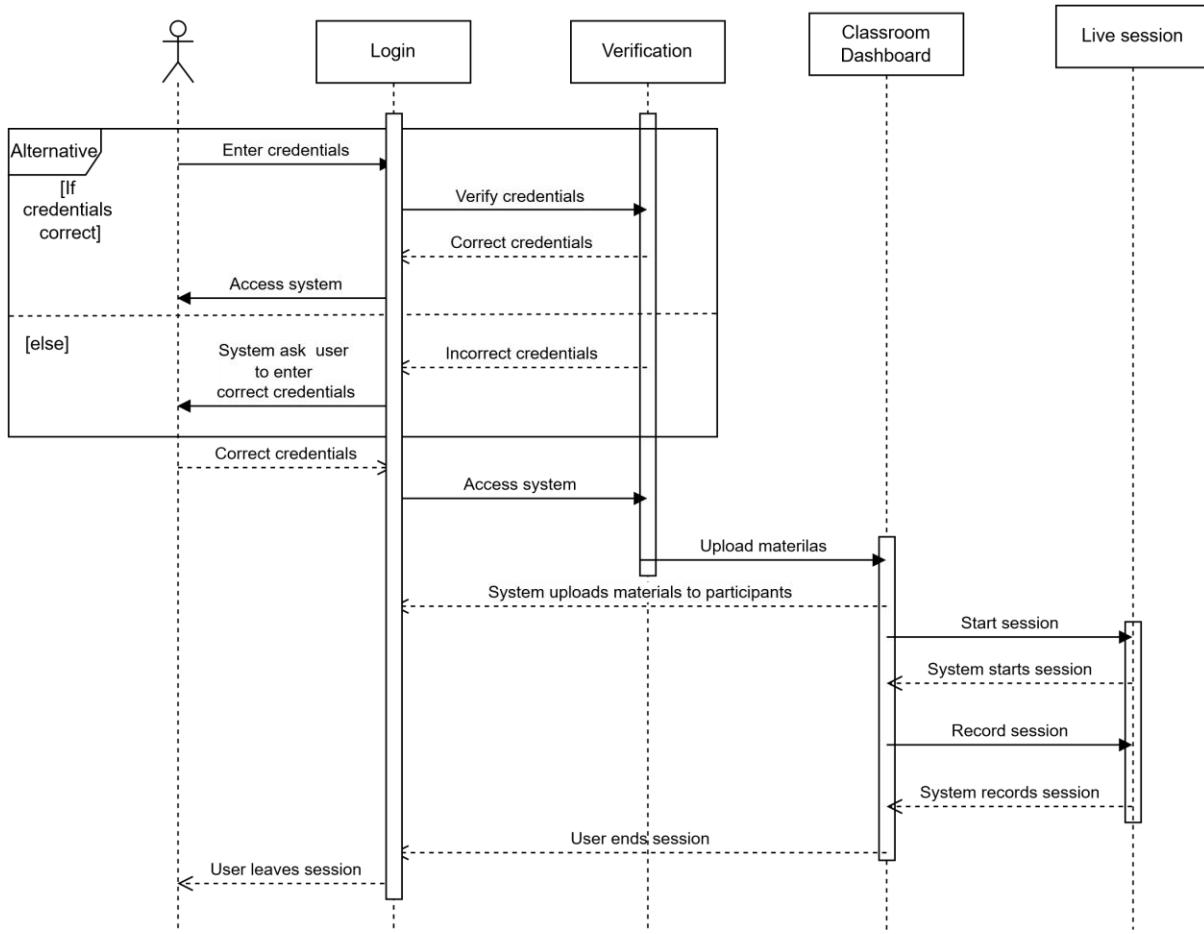
## Classroom Chat



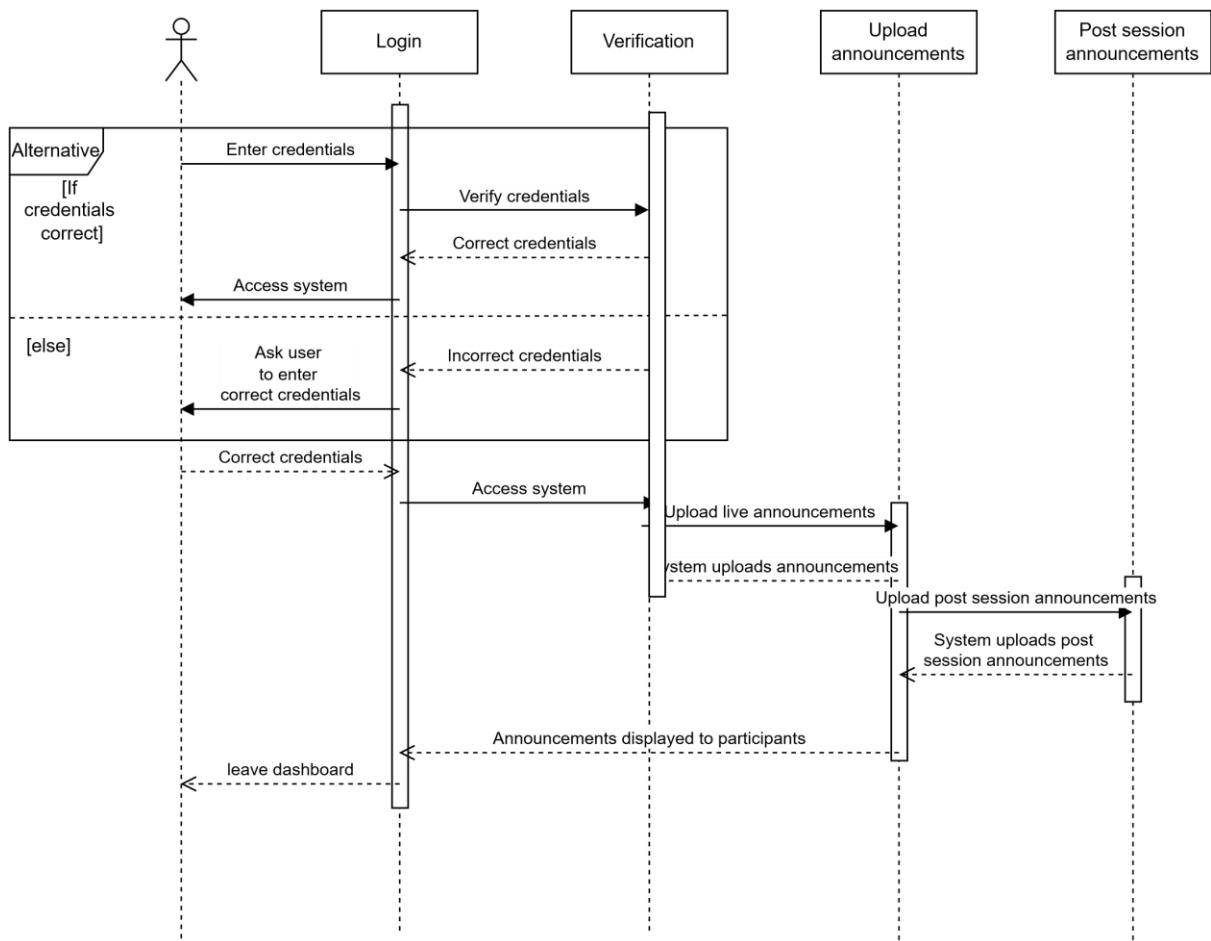
## Staffroom



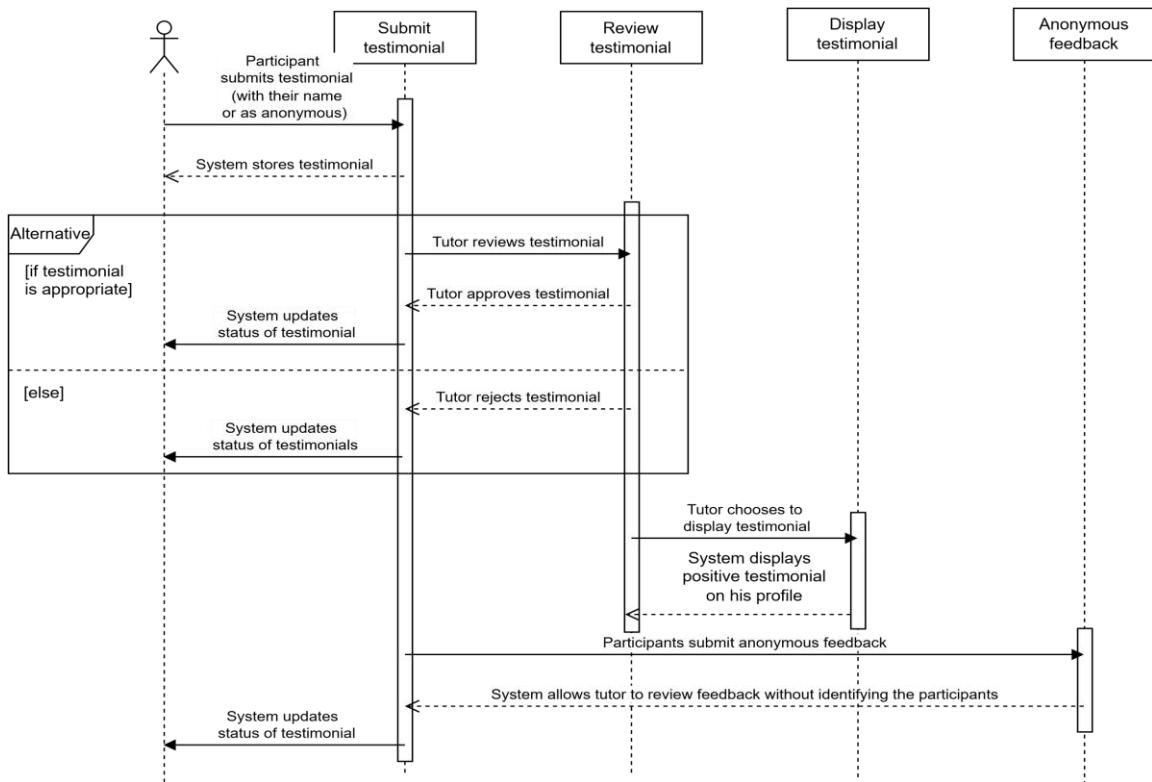
## Classroom + Recordings



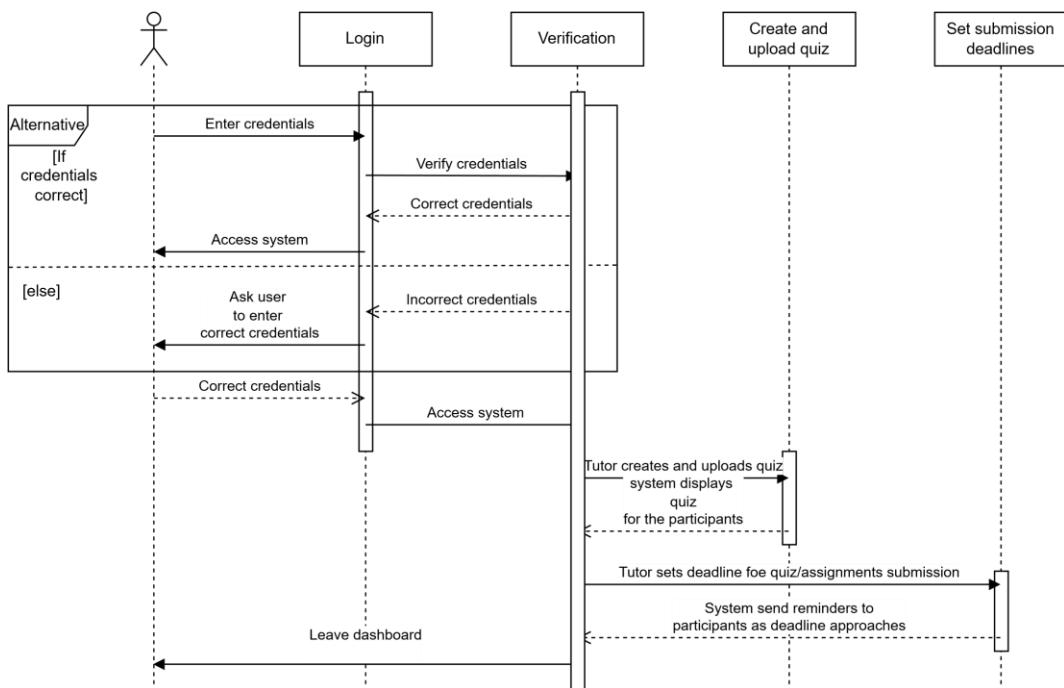
## Announcements



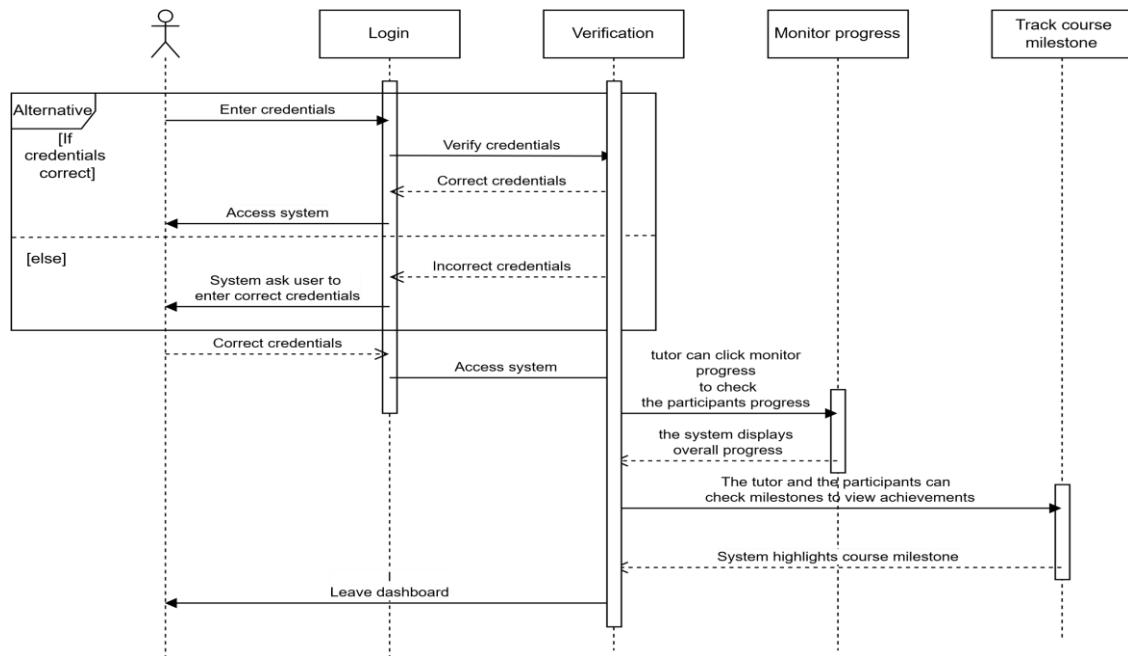
## Testimony/Review



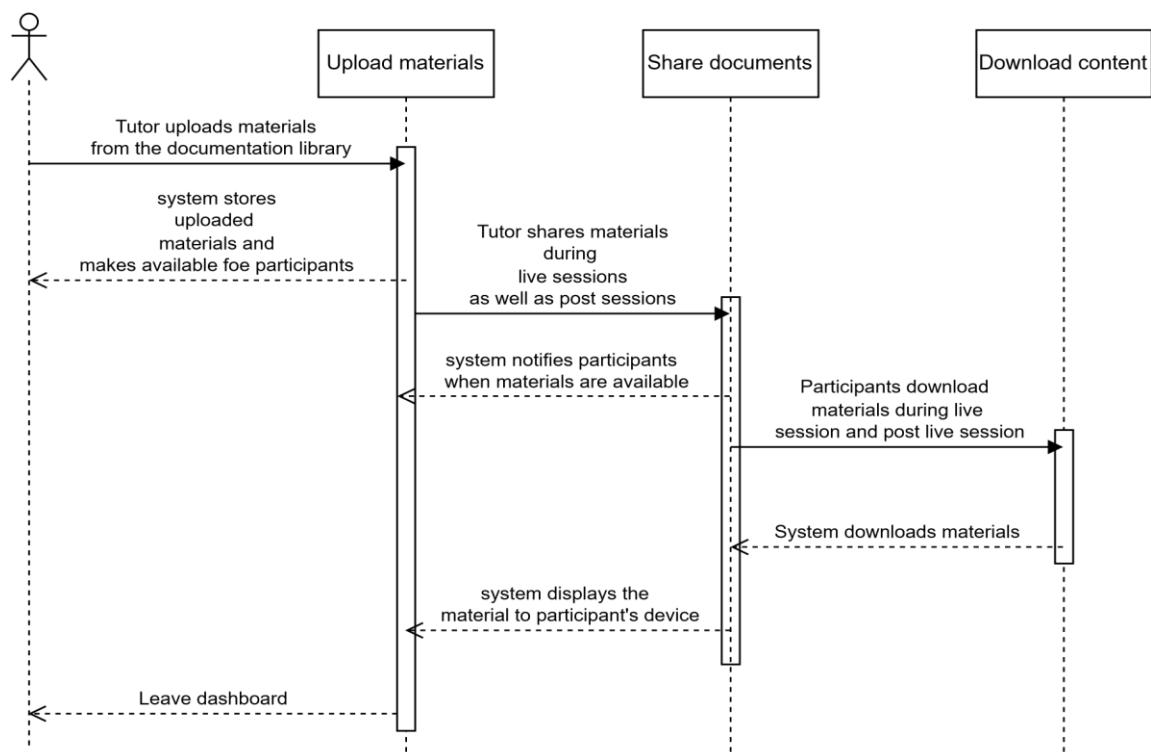
## Assessments/Quiz



## Check course progress/calendar



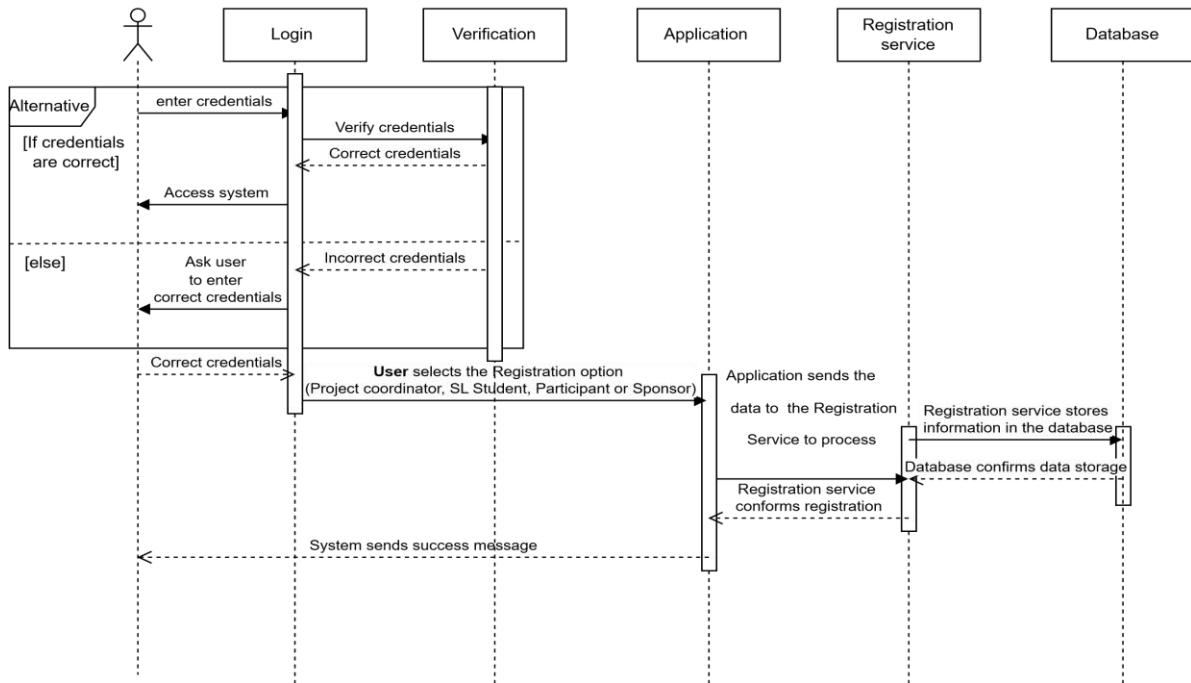
## Study materials access



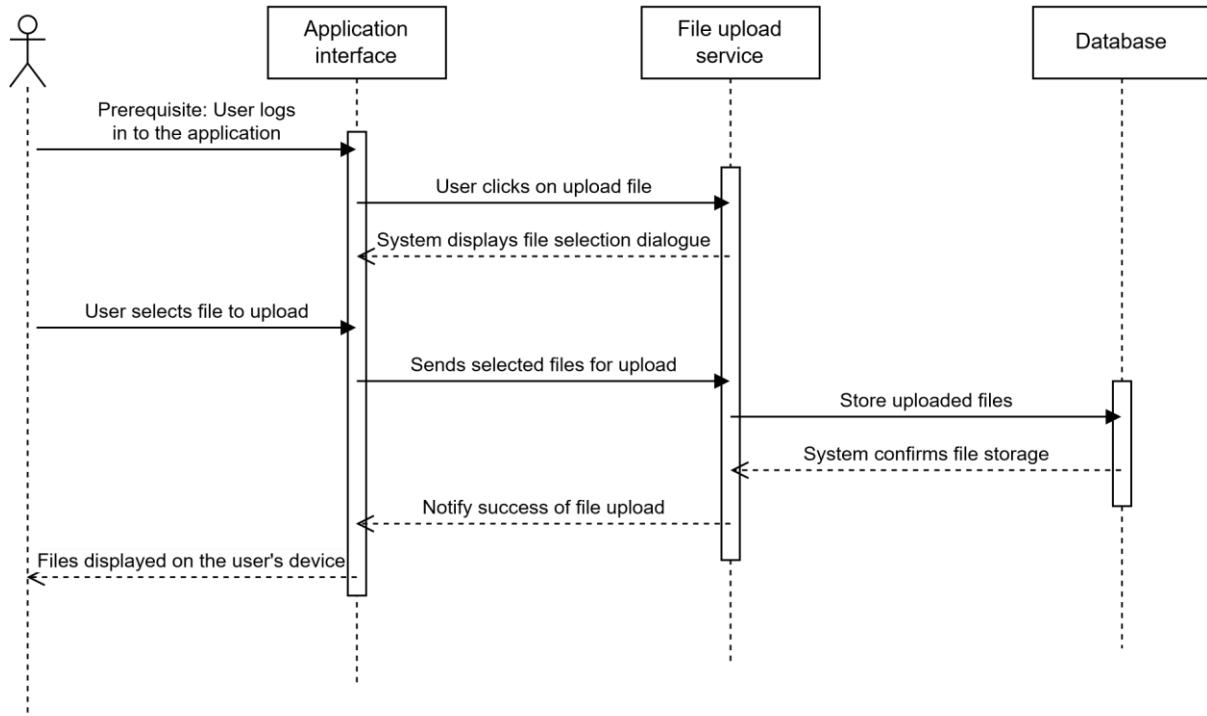
# Community Partner Organizations

## Register and manage students

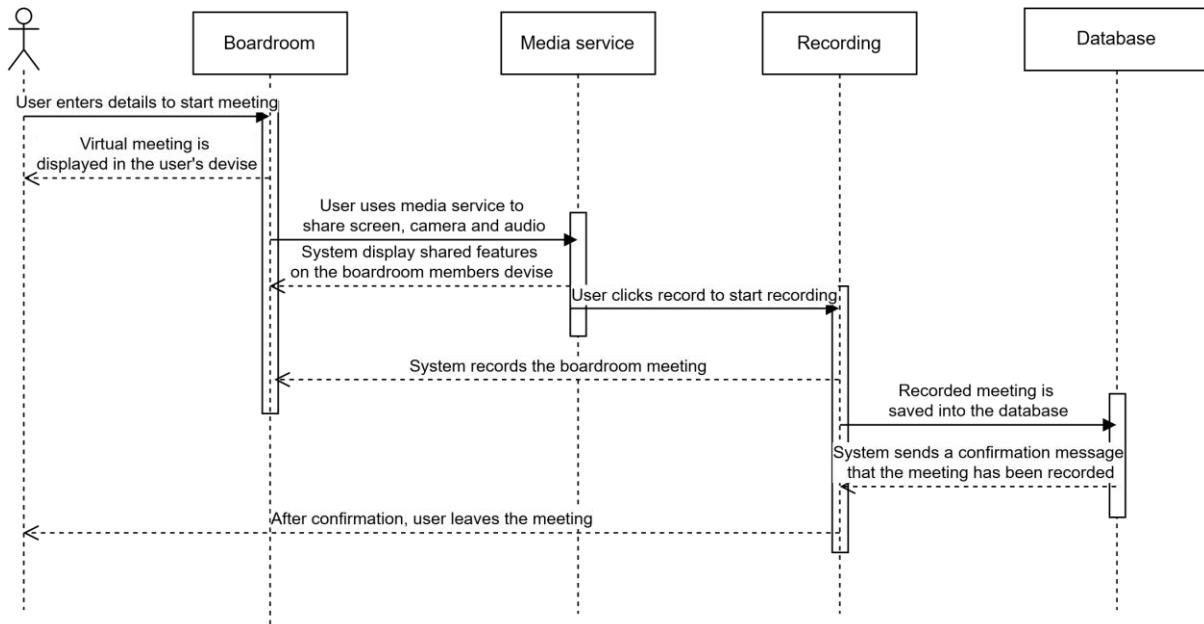
26. Register/manage participants



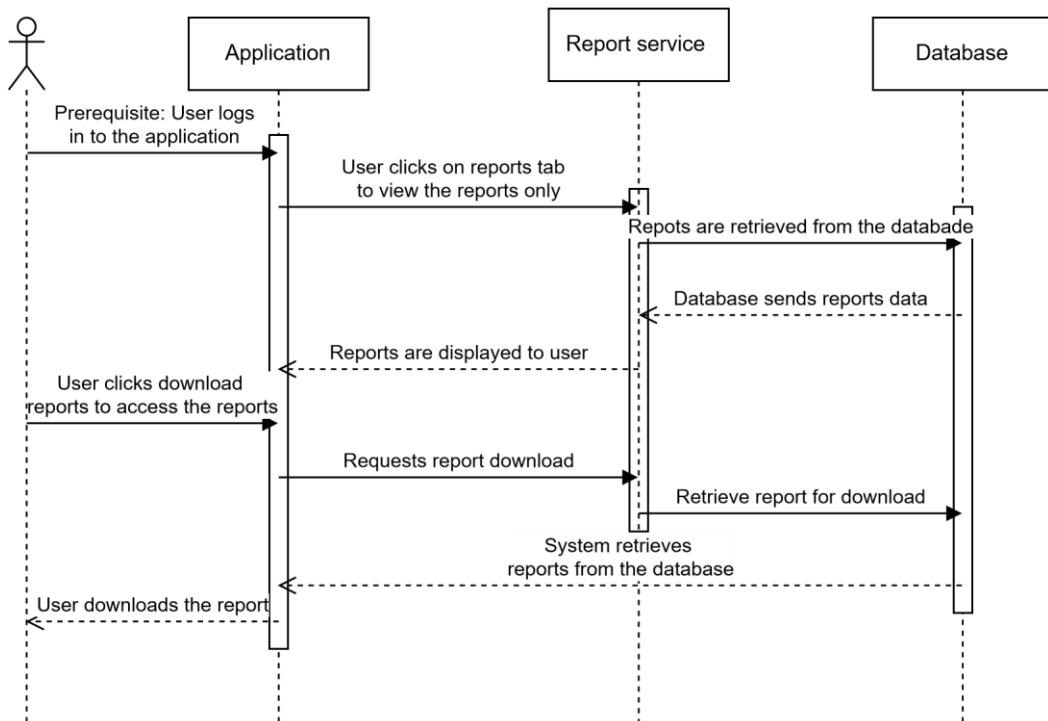
## Upload promotional materials



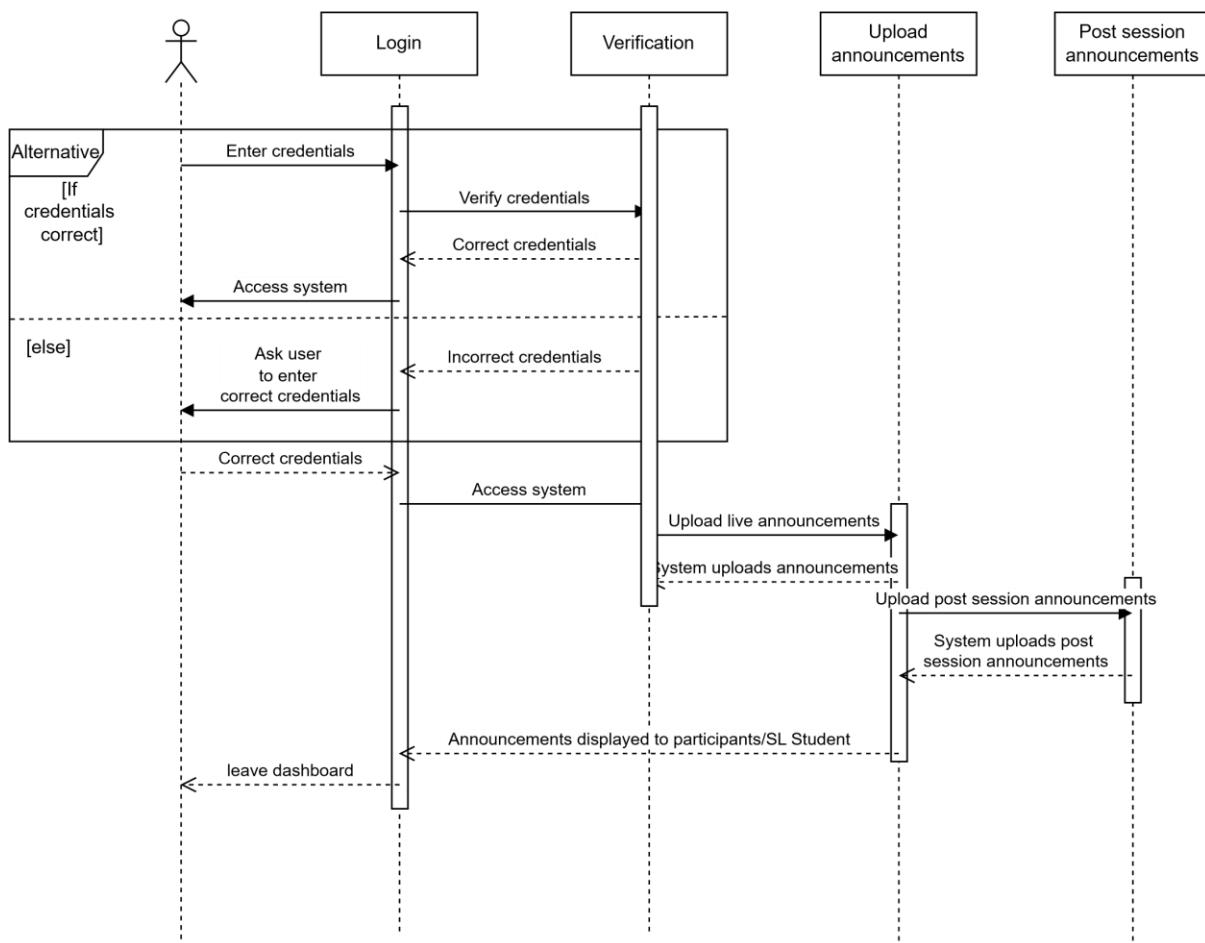
## Boardroom



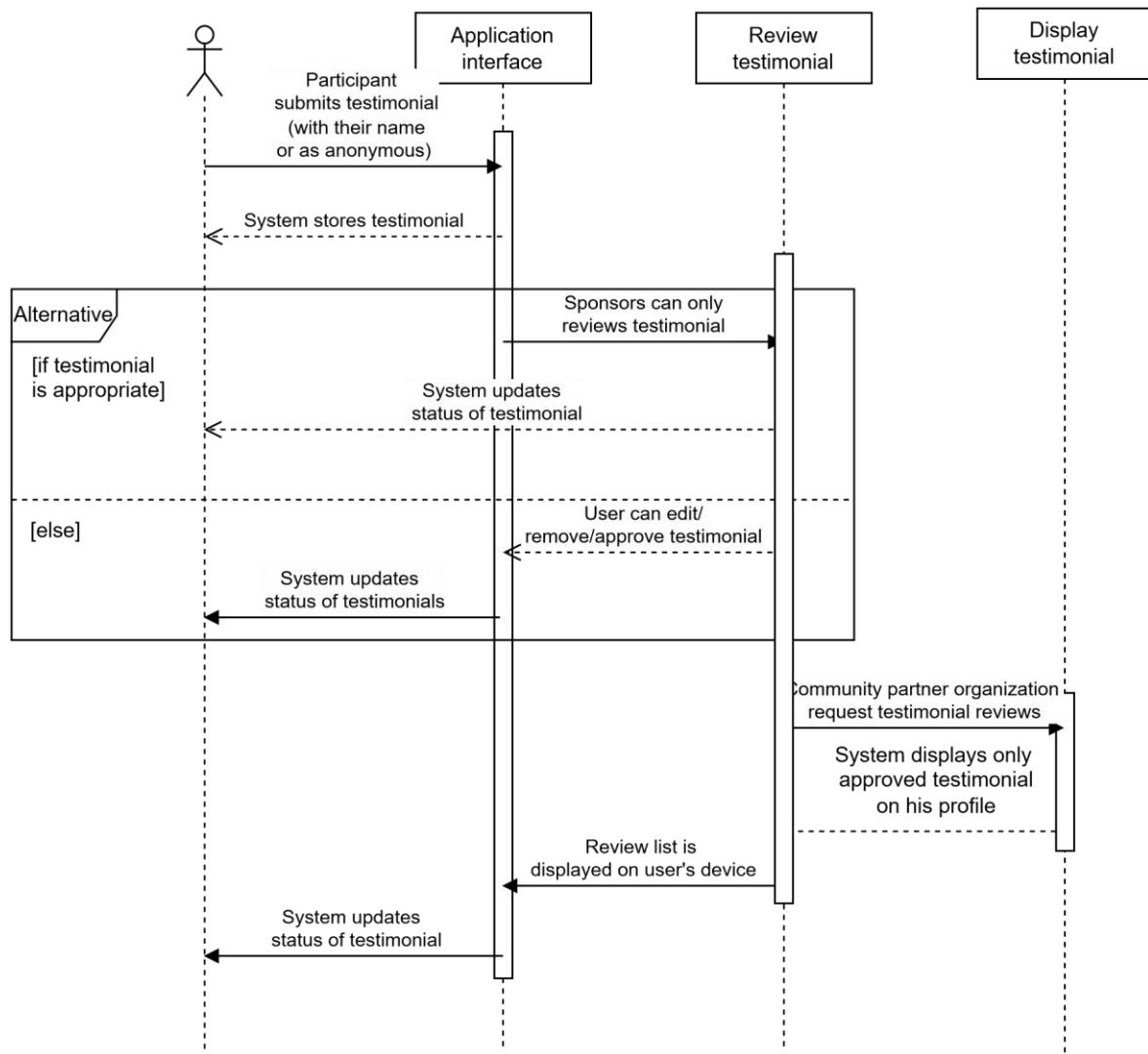
## Report statistics



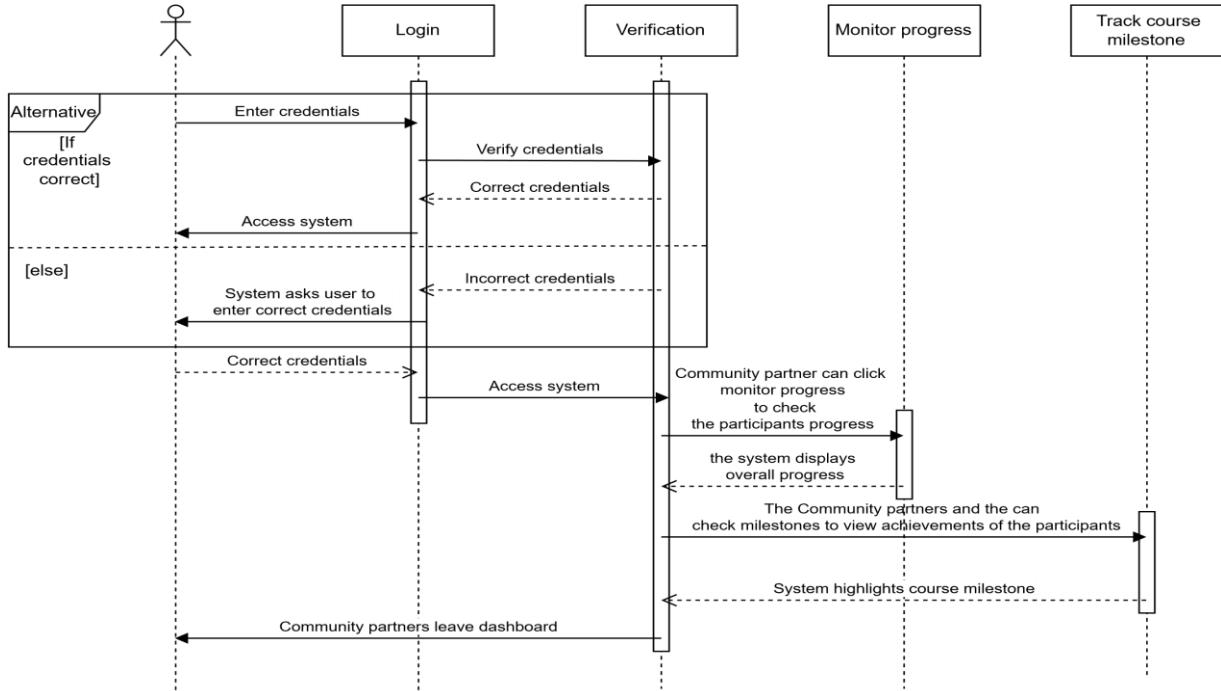
## Announcements



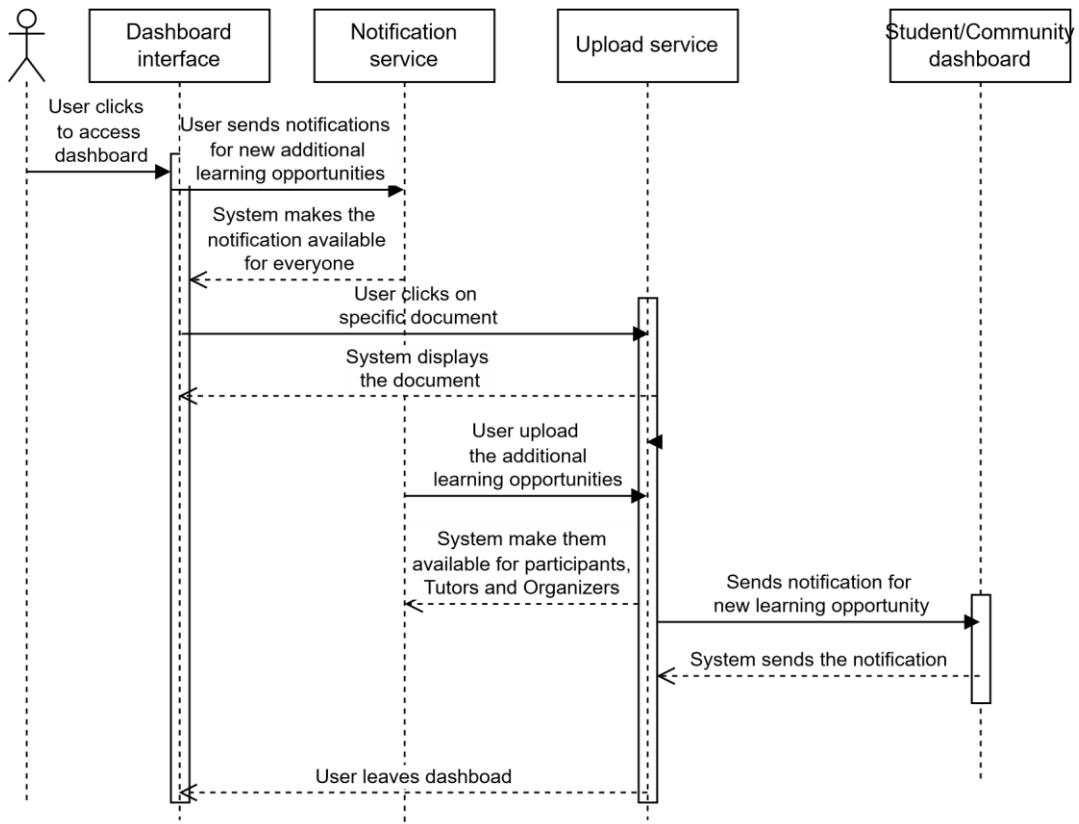
## Testimony/Review



## Check course progress/calendar

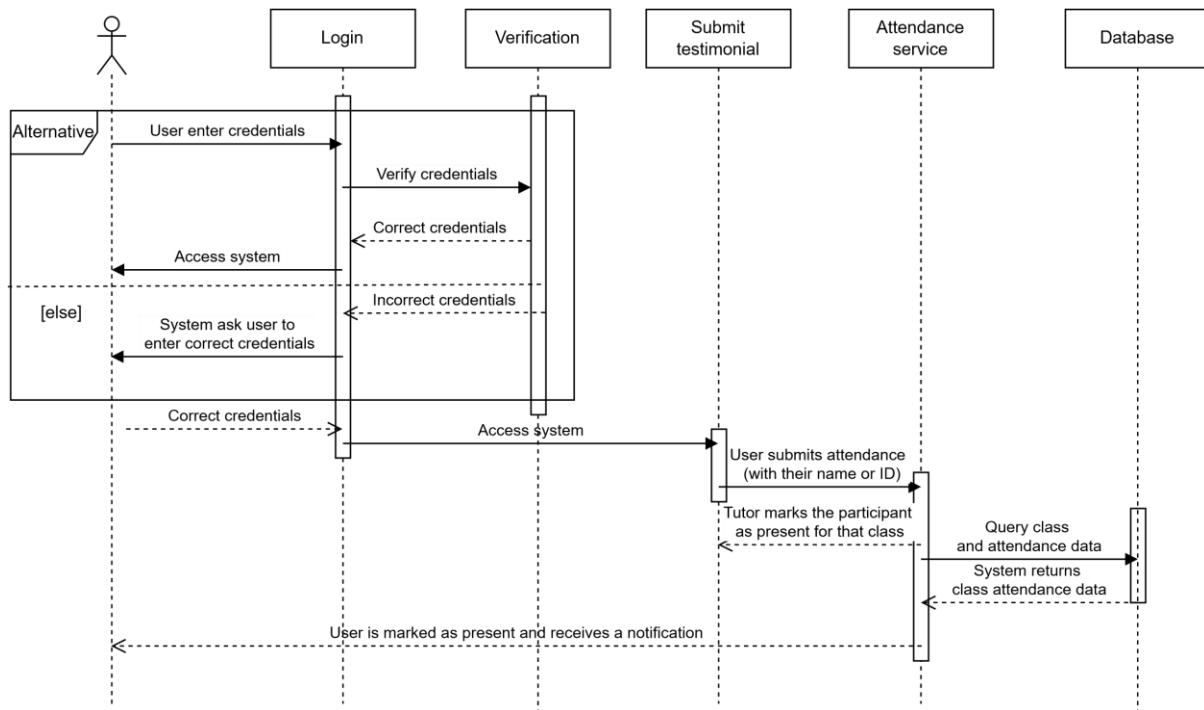


## Additional learning opportunities

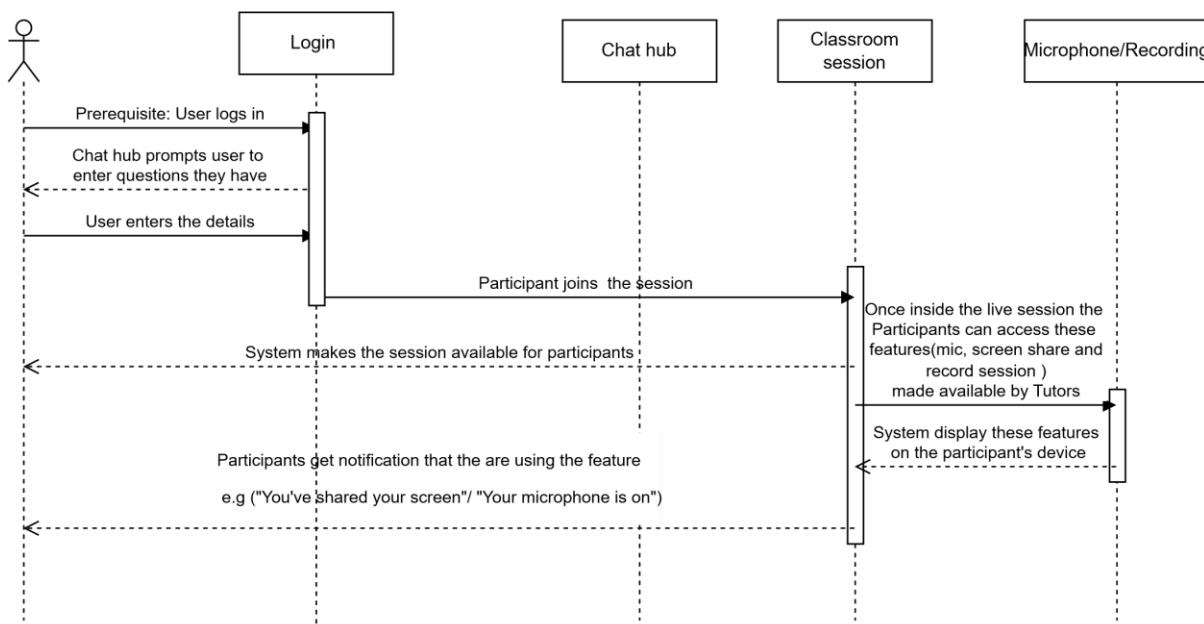


## Community Participant

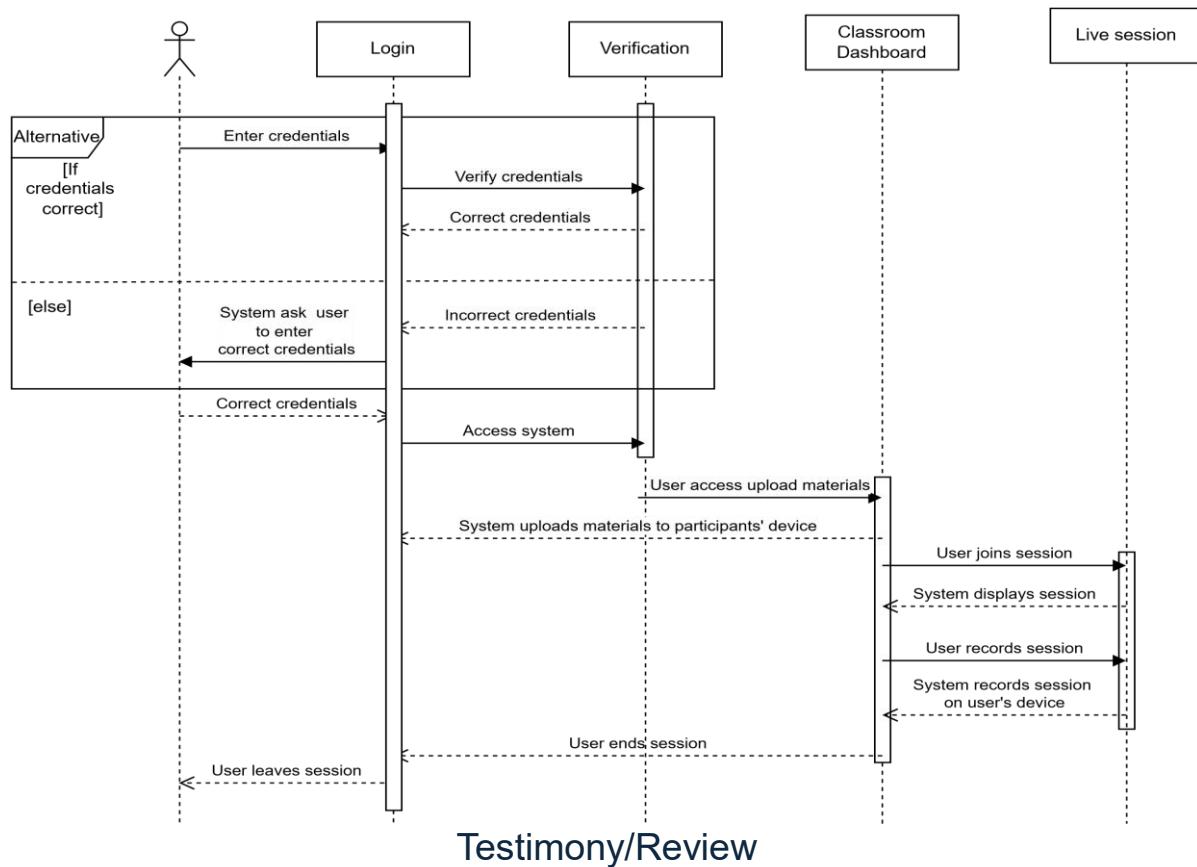
### Attendance



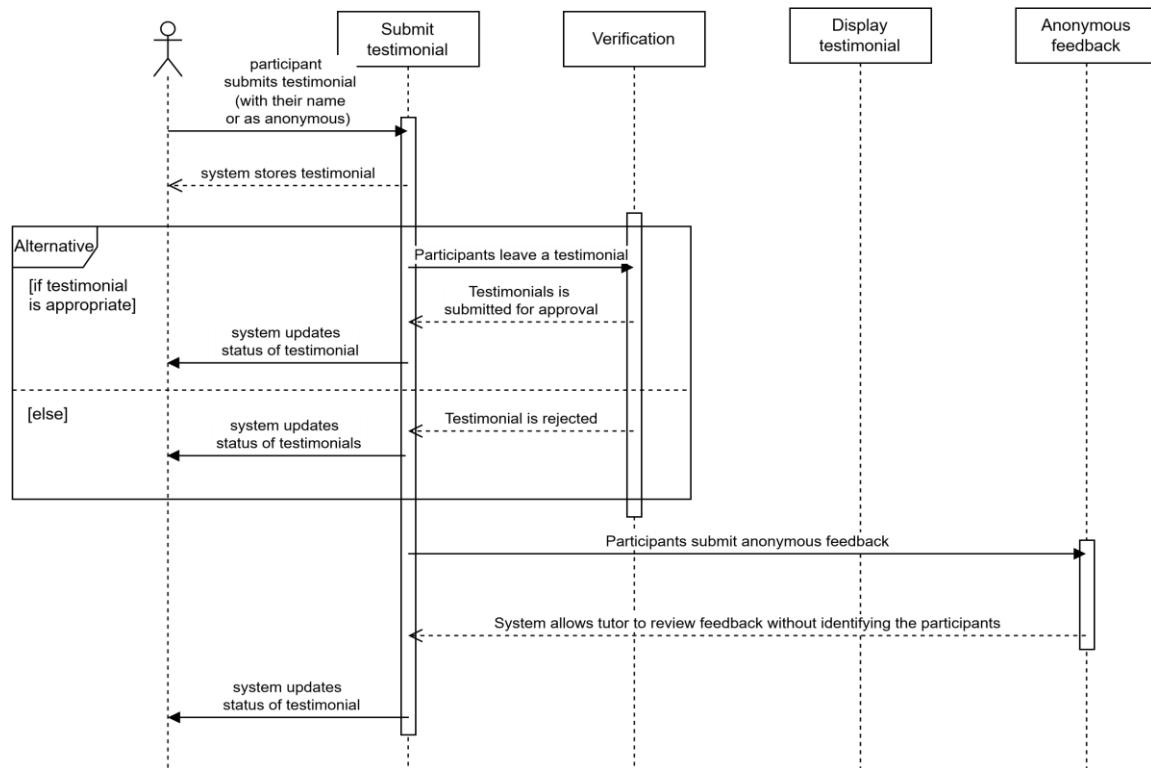
### Classroom chat



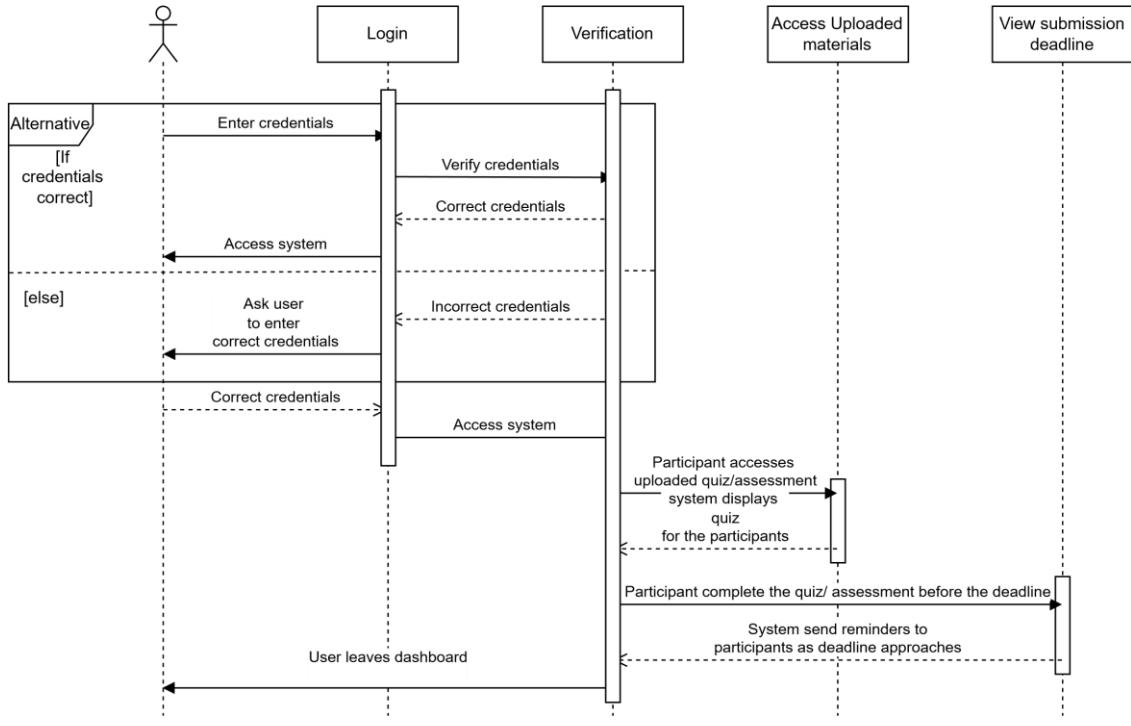
## Classroom + Recordings + Online (live session)



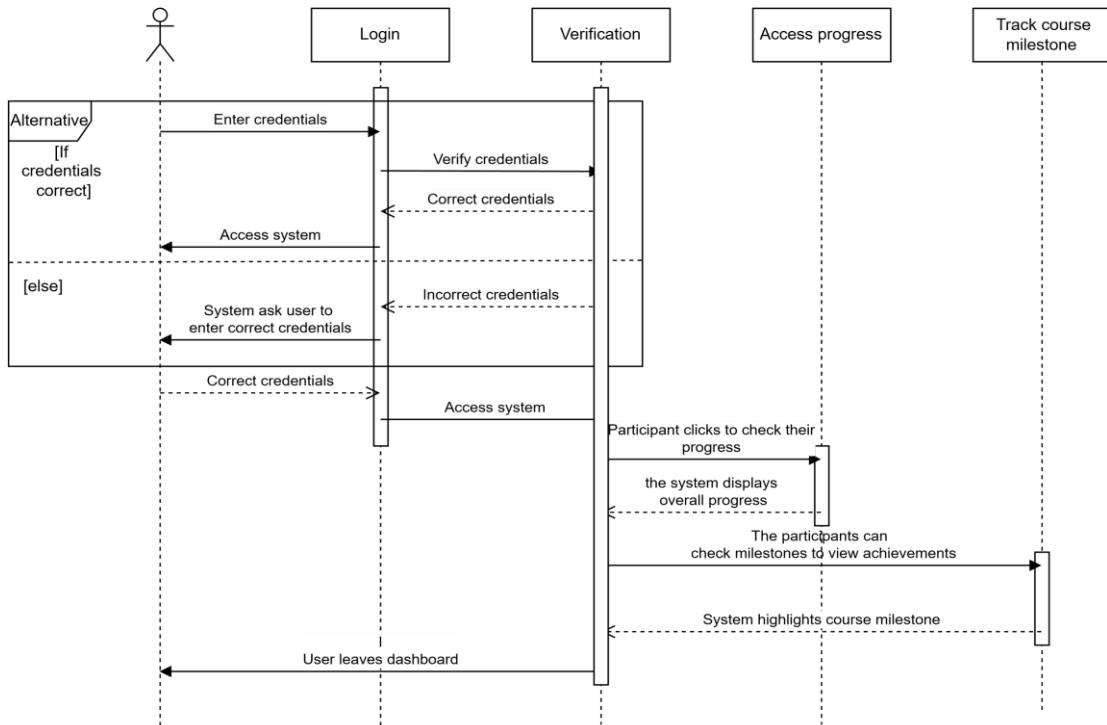
Testimony/Review



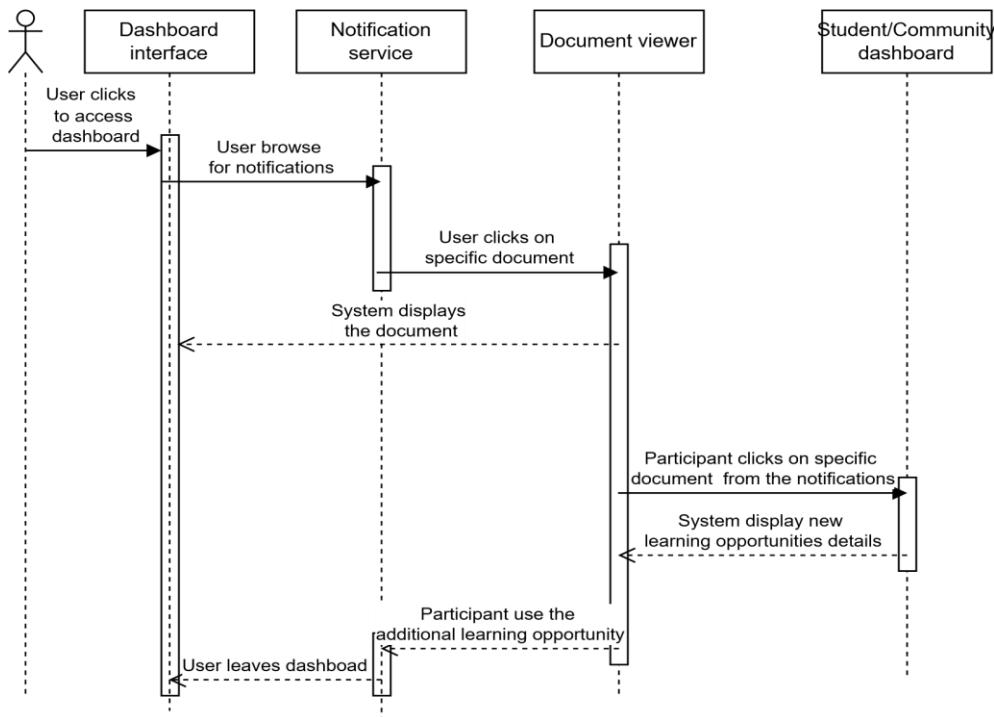
## Assessment/Quiz



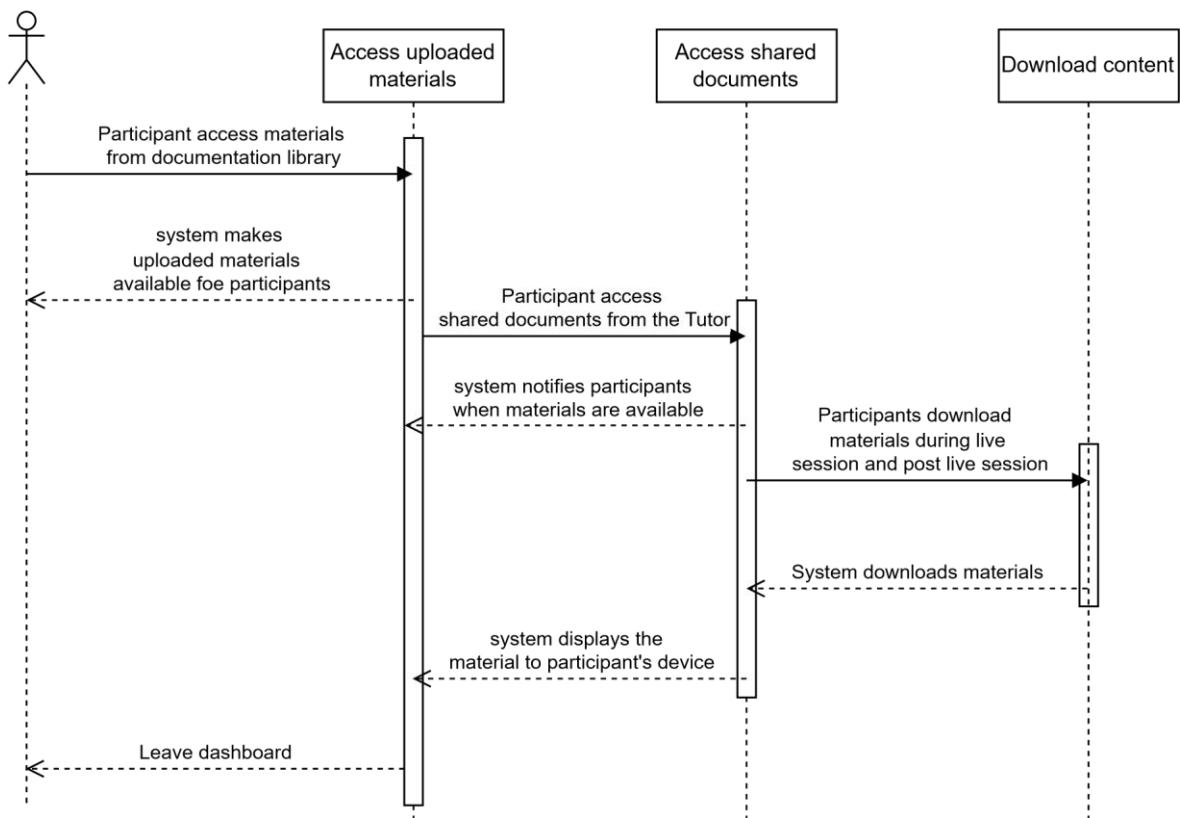
## Check course progress/calendar



## Additional learning material



## Study material access



# Noun Extractions

## Description

Users log into their dashboards. Project coordinators and community partner organizations access the registration dashboard to manage student and participant registrations. They use the classroom dashboard to send messages, manage presentations, and attend live lessons. The boardroom dashboard facilitates discussions among coordinators and partner organizations. The staffroom dashboard allows coordinators and students to upload and manage presentations, recordings, and documentation. The reports dashboard is for viewing and managing statistical reports. The assessments dashboard enables creation and management of assessments. The announcement button lets users manage and view announcements, while the course progress button shows progress reports. Community partner organizations send additional learning documents for review and upload by the coordinator, and all users view these opportunities. Finally, study materials are managed and uploaded by the coordinator and students, with participants viewing the content.

## Nouns Identified

Proper Nouns	Abstract Nouns
• Project Coordinators	• Management
• Community Partner Organizations	• Messages
• Registration Dashboard	• Presentations
• Classroom Dashboard	• Lessons
• Boardroom Dashboard	• Discussions
• Staffroom Dashboard	• Documentation
• Reports Dashboard	• Reports
• Assessments Dashboard	• Assessments
• Announcement Button	• Announcements
• Course Progress Button	• Progress
	• Opportunities
	• Content

- 
- |   |   |
|---|---|
| <ul style="list-style-type: none"> <li>• Additional Learning Opportunities</li> <li>• Study Materials features</li> </ul> | <ul style="list-style-type: none"> <li>• Users</li> </ul> |
|---|---|
- 

## CRC Cards

<b>Class: Dashboard</b>	
<b>Responsibility</b>	<b>Collaborations</b>
<ul style="list-style-type: none"> <li>• Provide access to all the other buttons (functionalities)</li> </ul>	<ul style="list-style-type: none"> <li>• Study Material button</li> <li>• Boardroom button</li> <li>• Announcement's button</li> <li>• Registration button</li> <li>• Staffroom button</li> <li>• Users(Project Coordinators, Community Partner Organizations, SL students)</li> <li>• Additional learning opportunities button</li> <li>• Course progress button</li> <li>• Progress reports button</li> </ul>

<b>Class: Announcement's</b>	
<b>Responsibility</b>	<b>Collaborations</b>
<ul style="list-style-type: none"> <li>• Create announcement</li> <li>• Upload announcement</li> <li>• Edit announcement</li> <li>• View/read announcement</li> </ul>	<ul style="list-style-type: none"> <li>• Users (Project Coordinators, SL students)</li> <li>• Announcement Button(TAB)</li> <li>• Main Dashboard</li> </ul>

<b>Class: Study Material</b>	
<b>Responsibility</b>	<b>Collaborations</b>
<ul style="list-style-type: none"> <li>• Upload study material</li> <li>• Edit study material</li> </ul>	<ul style="list-style-type: none"> <li>• Main Dashboard</li> <li>• Classroom Dashboard</li> </ul>

<ul style="list-style-type: none"> <li>• Read/view study material</li> </ul>	<ul style="list-style-type: none"> <li>• Assessments Dashboard</li> <li>• Staffroom Dashboard</li> <li>• Study Materials features</li> <li>• Users (Project Coordinators, Community Participant , SL students)</li> <li>• Presentations</li> <li>• Documentation</li> <li>• Assessments</li> <li>• Content</li> </ul>
--	---

<b>Class: Staffroom</b>	
<b>Responsibility</b>	<b>Collaborations</b>
<ul style="list-style-type: none"> <li>• Upload presentations</li> <li>• Edit presentations</li> <li>• View presentations</li> <li>• Send feedback on presentation</li> </ul>	<ul style="list-style-type: none"> <li>• Main Dashboard</li> <li>• Users (Project Coordinators, Community Participant , SL students)</li> <li>• Study Materials features</li> <li>• Presentations</li> <li>• Documentation</li> </ul>

<b>Class: Boardroom</b>	
<b>Responsibility</b>	<b>Collaborations</b>
<ul style="list-style-type: none"> <li>• Create message</li> <li>• Send message</li> <li>• Start live meeting</li> <li>• Join live meeting</li> <li>• Send documentation</li> <li>• Manage documentation</li> </ul>	<ul style="list-style-type: none"> <li>• Main Dashboard</li> <li>• Users (Project Coordinators, Administration)</li> <li>• Messages</li> <li>• Boardroom button(TAB)</li> <li>• Presentations</li> </ul>

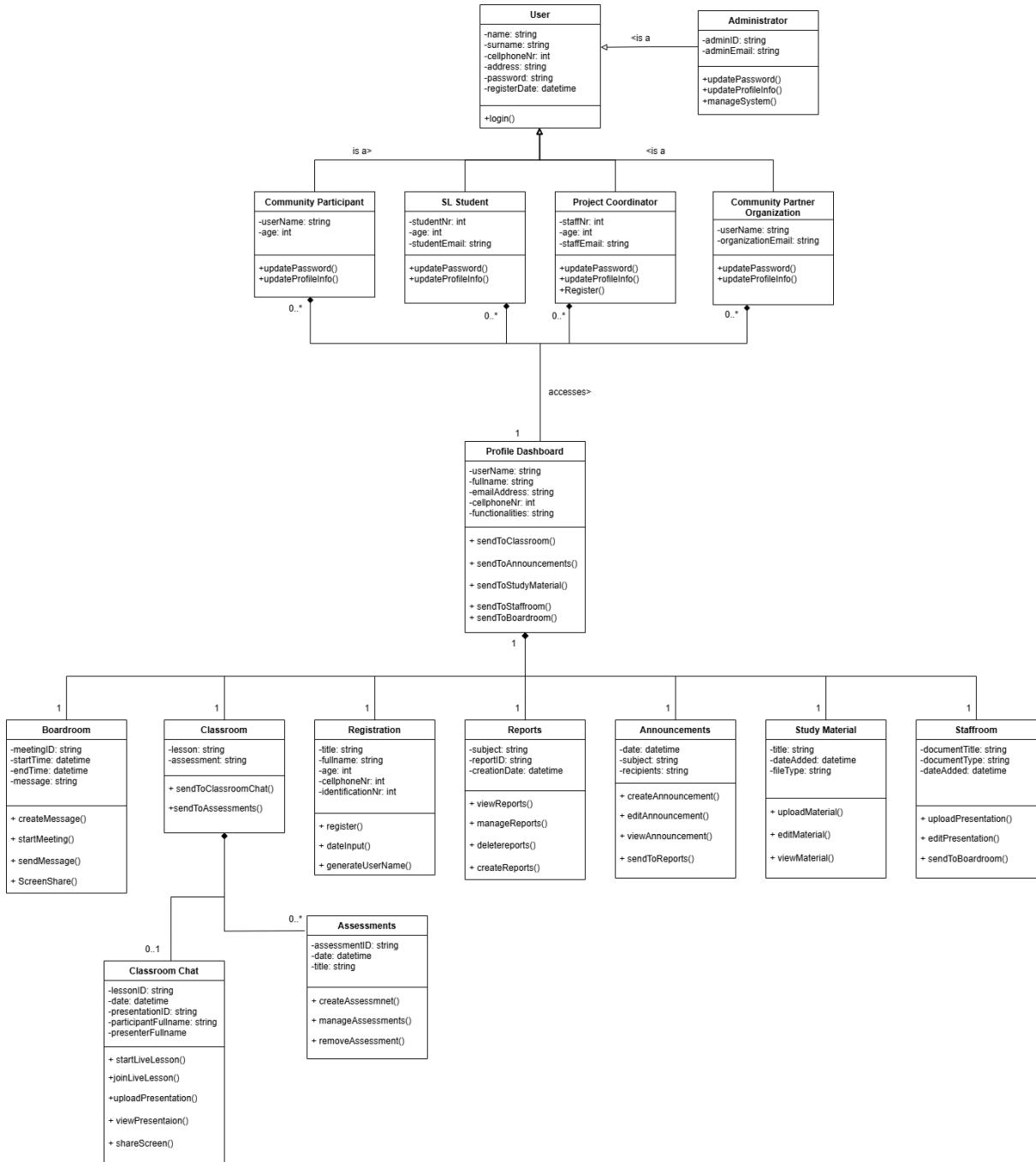
<b>Class: Classroom Chat</b>	
<b>Responsibility</b>	<b>Collaborations</b>
<ul style="list-style-type: none"> <li>• Start live-lesson</li> <li>• Join live lesson</li> <li>• Upload presentation</li> <li>• View presentation</li> <li>• Create message</li> <li>• Send message</li> <li>• Manage messages</li> </ul>	<ul style="list-style-type: none"> <li>• Main Dashboard</li> <li>• Users (Project Coordinators, Community Participant , SL students)</li> <li>• Classroom Dashboard</li> <li>• Presentations</li> <li>• Content</li> <li>• Lessons</li> <li>• Messages</li> </ul>

<b>Class: Assessments</b>	
<b>Responsibility</b>	<b>Collaborations</b>
<ul style="list-style-type: none"> <li>• Creates assessment opportunities</li> <li>• Manages assessments</li> <li>• Remove assessments</li> </ul>	<ul style="list-style-type: none"> <li>• Users (Project Coordinators, Community Participant, SL students)</li> <li>• Assessments button(TAB)</li> <li>• Main Dashboard</li> <li>• Classroom Button(TAB)</li> <li>• Classroom Dashboard</li> <li>• Assessments</li> </ul>

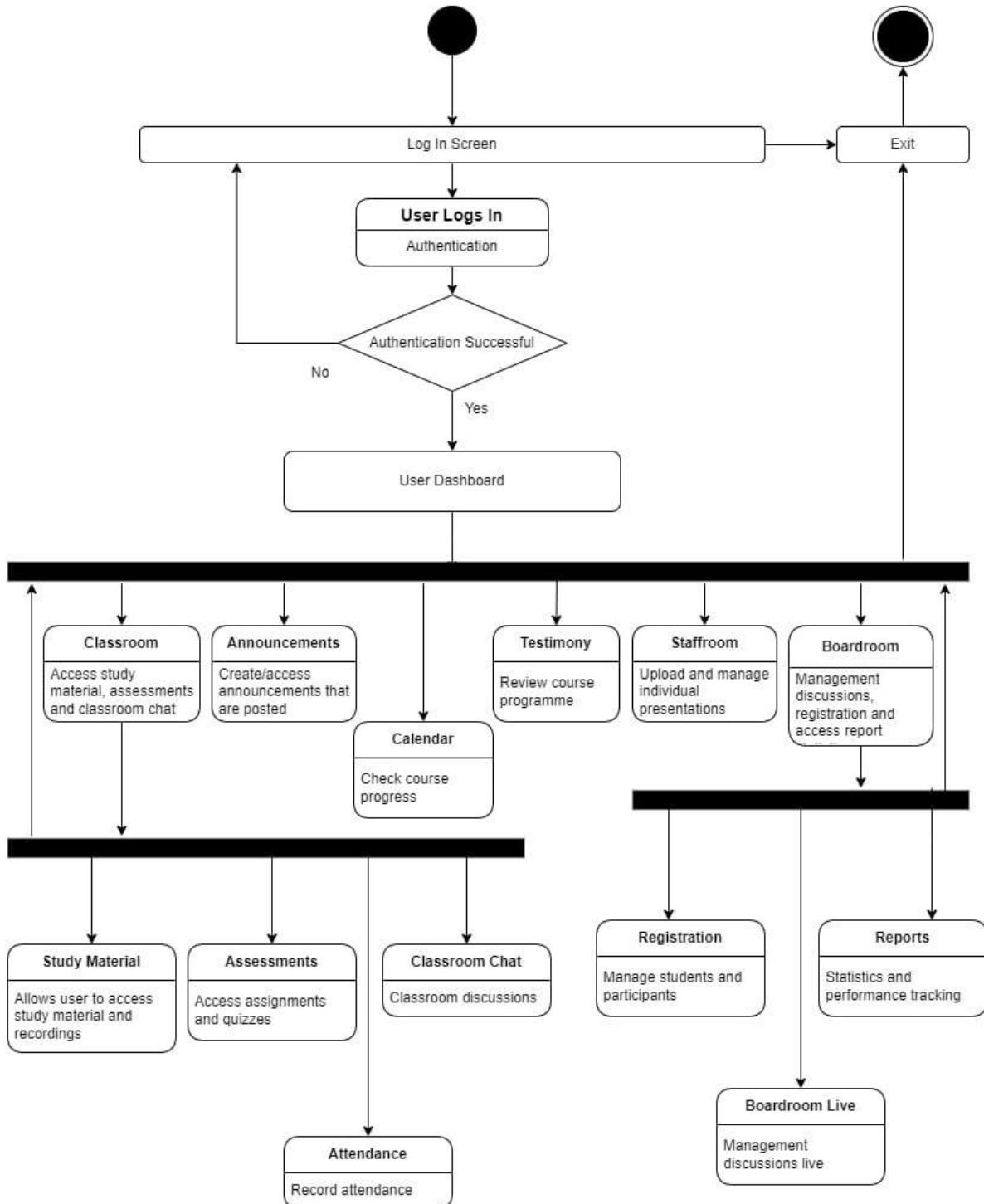
<b>Class: Reports Dashboard</b>	
<b>Responsibility</b>	<b>Collaborations</b>
<ul style="list-style-type: none"> <li>• View and manage statistical reports</li> <li>• Display progress reports</li> <li>• Create statistical reports</li> </ul>	<ul style="list-style-type: none"> <li>• Users (Project Coordinators, Community Participant)</li> <li>• Main Dashboard</li> <li>• Reports</li> <li>• Reports Dashboard</li> <li>• Reports button(TAB)</li> <li>• Progress</li> </ul>

<b>Class: Registration</b>	
<b>Responsibility</b>	<b>Collaborations</b>
<ul style="list-style-type: none"> <li>• Register individuals to the system</li> <li>• Input individual data on to the system</li> </ul>	<ul style="list-style-type: none"> <li>• Users (Project Coordinators, Administration, Community Partner Organizations)</li> <li>• Main Dashboard</li> <li>• Registration Button(TAB)</li> </ul>

# Class Diagrams



# State Chart



# Pseudo code

## Class : Registration

Method: Register

```
Void Register()
{
    Display Registration Screen
    Input Registration Details(name, email, password)
    Validate Registration info
    If (details== valid)
    {
        Save registration Details to database
        Display Success Message
        REDIRECT TO Login Screen
    }
    Else
        Display error message
        Loop back to Input registration details
}
```

## Class: Login & Authentication

Method: VerificationUserInfo

```
Void VerifyUserInfo(email, password)
{
    Connect to database
    Query database for matching email and password
    If (match== found)
        Return true
        Display dashboard
    Else
        Return false
        Disconnect from database
}
```

## Class: Classroom

Method: StartSession

Void StartSession()

{

    Display meeting interface

    Display chat hub

    Display input for meeting information

    If(startsession button == clicked && user == administrator)

{

    Display mic toggle button

    Display recordmeeting button

    Display share video button

    Display participants

}

}

### **Class: Boardroom**

Method: enterMeetingDetails()

Void enterMeetingDetails()

{

    Display “Enter meeting details”

    Input details

    If(startmeeting button==clicked)

{

        Display virtual meeting with meeting details

        If(shareschreen==clicked)

            Enable screen sharing

        If(sharecamera==clicked)

            Enable camera

```

        If(shareaudio==clicked)
            Enable audio
        If(record==clicked)
            Start recording
            Save recording to database
    }
}

```

### **Class: Staffroom**

1. Method: Checkuserroles()

```

Void CheckUserRoles()
{
    If( user=="Project Coordinator")
        Display Upload Course Material or manage material
    if(user=="SL Student ")
        Display Create Assessment and Upload Document
        Call CreateAssessments()
}

```

2. Method CreateAssessments()

```

{
    If(UploadCourseMaterial==clicked)
    {
        Display template for assessment
        Input assessments
        If(next==clicked)
        {
            Input assessment details (name, duration)
            If(Submit==clicked)
                Send assessment file to Project Coordinator
        }
    }
}

```

### **Class: Registration**

### **Method: Register()**

```
{
```

```
    Display options: "System admin, SL Student, Community member, Community  
    Partner Organisation"
```

```
    Get user selection
```

```
    If(user selected option)
```

```
{
```

```
        Display form for selected option
```

```
        Display input form
```

```
        If (submit button== clicked)
```

```
            If (user info saved to database)
```

```
                Display "Already registered"
```

```
            Else if (user not saved to database)
```

```
                Save Information to database
```

```
                Display "Registered successfully"
```

```
}
```

```
}
```

### **Class: Reports**

#### **1. Method: ViewReports**

```
Void ViewReports
```

```
{
```

```
    Display list of available reports
```

```
    If(download==clicked)
```

```
        Download selected file to user's device
```

```
}
```

## 2. Method: CreateReport()

```
Void CreateReport()
{
    If(user==Project Coordinator || user== System Administrator)
    {
        Display "Enter report data"
        Input total students, studentpassed , attendance

        Display options to edit, delete or post report
        If(edit==clicked)
            Enable user to edit document
        If(delete==clicked)
        {
            Display message ("Are you sure you want to delete this report?
            ")

            If(yes==clicked)
                Delete document from database
            Else
                Redirect to reports dashboard.
        }
        If(post==clicked)
            Post report
    }
    Else
        Display "Access Denied"
}
```

Class: Calender

Method : MonitorProgress()

```
Void MonitorProgress()
{
    Display class progress overview
    Display covered topics and activities
```

Display calendar with: upcoming sessions, assignment deadlines, reminders for tasks

}

Class: StudyMaterial()

1. Method: ShareDocument()

{

Display list of available materials

If(sharedocument button== clicked)

{

Notify participants about new material

Display "Documents shared successfully"

}

}

1. Method: DownloadMaterial()

Void DownloadMaterial

{

Display list of available material

If( downloadMaterial button == clicked)

{

Allow participants to download material into their device

Display" Downloaded successfully"

}

}

2. Method: AccessRecordings ()

Void AccessRecording()

{

Display list of recorded sessions

If( user select recording)

Play selected recording

}

### **Class: Attendance**

#### **Method: MarkAttendance()**

{

Display attendance register

Enter Identity number

Select Presence status( Yes/No)

If(Submit button== clicked)

{

Save attendance information on database

Update participants attendance status to present

Display Confirmation message

}

Else if( Cancel button ==clicked)

{

Cancel Attendance Registration

Display Cancellation message

}

### **Class: Announcement**

#### **Method: CreateAnnouncement()**

Void CreateAnnouncement()

{

Display “Select recipients :”

Create checkbox to select recipients

Display “Type announcement:”

Create textbox for announcement

```

If(textBox!=null&& postAnnouncement==clicked)

{
    Send announcement to selected recipient

    Display “Announcement posted!”

}

Else

Display “Announcemet cannot be empty ”

}

```

### **Class: Reviews**

1. Method: SortReviews()

```

{
    Display reviews content

    If(user click sort)

    Sort reviews by selected format

}

```

2. Method: StarReview()

```

{
    If(star==clicked)

        Rate the review with selected stars

    If(thumbDown==clicked)

        Display”Are you sure you want to delete this review”

    If(yes==clicked)

        Delete the review

    If(thumbUp==clicked)

        Increase like count of reviews

```

}

## Record of Hours Spent

Workflow	Hours spent
Requirements Workflow	50
Analysis Workflow	100
Design Workflow	70
Implementation	30
Other(delays)	5
<b>Total: 255 hours</b>	

## Overview of Development Effort

### Overview of Development effort

#### Requirements Workflow

Our approach to developing functionalities for the Interactive Learning Software, which includes a repository of academic content accessible to beneficiaries throughout their study, was collaborative and iterative. We adopted a roundtable method, fostering an open, democratic environment where ideas were freely shared and critically evaluated by all team members.

During this process, we drafted targeted questions for the client (Lecturer), which provided valuable insights into the functionalities that should be prioritized and those that could be deprioritized. Following the client's feedback, we reconvened for an additional roundtable discussion to refine our list of functionalities. This phase also included the creation of detailed use cases for visual representation, helping us to better understand how these features would be used in practice.

In conjunction with this, we applied rapid prototyping techniques to design User Interfaces, offering a preliminary view of how the proposed functionalities would be presented. These combined efforts allowed us to finalize the most critical features for the software.

Weekly meetings were held with stakeholders to confirm that client's requirements were met.

The team worked 50 hours on this workflow.

## **Analysis Workflow**

After receiving the final feedback from the client, we conducted a feasibility study to evaluate the practicality of our proposed approaches. During this phase, we carefully examined whether each functionality should be automated or manually operated, considering factors such as user roles, restrictions, privacy and security (in line with POPIA compliance), recruitment strategies, and the UI architecture.

Like our previous workflows, we adopted a collaborative and iterative approach, encouraging open discussions to share ideas, provide constructive criticism, and reach consensus. This roundtable format allowed for a comprehensive evaluation of all aspects of the project.

By the conclusion of this workflow, we were able to clearly visualize:

- How each functionality would operate,
- The distinction between automatic and manual functionalities,
- The roles and restrictions tied to different users,
- Recruitment strategies,
- Step-by-step implementation processes,
- Best case, worst case, and alternative scenarios,
- Noun extraction for object identification,
- CRC (Class-Responsibility-Collaboration) cards for role analysis.

The team dedicated 75 hours to this analysis workflow, ensuring a thorough and well-structured foundation for the next phases of development.

## **Design Workflow**

During this workflow, we were fortunate to have team members with diverse strengths across various design aspects, including use cases, sequence diagrams, user interfaces (UIs), and state charts, as well as others with a deep interest in the backend and mechanical aspects of the software. This blend of expertise fostered a collaborative environment where team members could share their skills, leading to valuable knowledge exchange throughout the process.

Thanks to the team's well-rounded expertise, we were able to efficiently make decisions regarding key elements of the project. These included the direction of the UI architecture, aesthetics, logic flow, and object-oriented programming (OOP) principles such as class diagrams, abstraction, and encapsulation. Techniques like noun extraction helped further refine our approach to developing abstracted objects and system logic.

By the end of this workflow, we had produced a comprehensive collection of deliverables, including:

- Detailed use cases,
- Consistent UIs with a cohesive look and feel,

- Abstracted objects,
- Class diagrams,
- Sequence diagrams,
- A state chart.

These deliverables laid a solid foundation for the next stages of development, ensuring clarity and alignment within the team. The workflow took 100 combined hours to complete, with an additional 25 hours dedicated to drafting pseudo code for system implementation.

In total, the team spent 250 hours on the product, covering analysis, design, and preparation for implementation.