

SOFTWAR ENGINEERING

Managing student courses

ID:4

Lama bander alsaeedi 444003323 s444003323@st.uqu.edu.sa

Rimas Hassan Alshehri 44411873 s44411873@st.uqu.edu.sa

Ghala Abed 444002510 s444002510@st.uqu.edu.sa

Lama kudish 44410529 s44410529@st.uqu.edu.sa

Ghadi Abdullah Alharbi 444002299 s444002299@st.uqu.edu.sa

TABLE OF CHANGING DIAGRAMS

Diagram name	Page number	Modification explanation
COMPONENT DIAGRAMS	38	<ol style="list-style-type: none">1. We modified the database shape.2. We added a port to the learning component.3. When saving to the database, we also enabled data retrieval from the Student Training Department's database.
DEVELOPMENT DIAGRAMS	39	WE INITIALLY PLACED THE COMPONENTS ON THE WEB SERVER, BUT SINCE THEY ARE SOFTWARE THEY NEED TO BE INSTALLED ON THE CLIENT PC.

1. BUSINESS REQUIREMENTS

1.1 Needs Statement (Problem)

- Students frequently forget course dates, leading to missed sessions.
- Lack of technical support to solve problems during course delivery
- Delayed delivery of presentation content from organizing bodies to course providers impacts the quality and effectiveness of teaching.
- Low attendance of the target group
- Lack of direct communication channels between students and course supervisors hinders timely assistance and support.
- Absence of alternative methods to access or view attendance courses limits accessibility for students.

1.2 Business Goals

Our main goal is to develop the delivery of Training courses at Umm Al-Qura University by:

- alerting students about course timings.
- providing a technical support team for the course to solve problems during the course.
- Ensure timely delivery of course content.
- providing easier and more effective ways to communicate between students and course supervisors .
- Increase accessibility to courses for the target group.
- providing other ways to present the course at the same time.

2. STAKEHOLDERS

Name: Ruqaya Eid Abdul Khair Abu Dhara

email: reabuthiraa@uqu.edu.sa

Organization: Training Department - Vice
Deanship of Student Affairs for Student
Activities and Training -

Role: Member

Requirements Gathering Methods :

One-on-one interview.

QUESTION

Q: How are the courses delivered to the students?

A: The courses can be delivered to the students by sending emails.

Q: What are the common problems that affect the delivery of courses to the targeted students?

A: One common problem is that the emails may not reach all the targeted students due to issues with email addresses or filtering of spam messages by email servers.

Q: What types of courses are offered?

A: A variety of courses are offered, some are available to all university students, while others are offered based on their academic level or specialization.

Q: Who is the target audience for these courses?

A: The courses are targeted towards male and female students only.

Q: What do the students wish for in the application?

A: The students wish for the courses to reach as many students as possible and for registered students to be reminded when the course dates are approaching.

2.STAKEHOLDERS

Name: Fawzia Hassan Mohammed Ghazawi

email:fhgazzawe@uqu.edu.sa

Organization: Assistant professor

Software Engineering Department

College of Computers

Role: Member

Name: Afnan Muhammad Al-Qathami

email:amqethami@uqu.edu.sa

Organization: Assistant professor

Department of Biology - Faculty of Science

College of science

Role: Member

Requirements Gathering Methods:

One-on-one interview.

QUESTION

Q:What are the problems you faced in providing the courses?

A:Appointments are not compatible with students.

Q:What are the problems you faced in providing the courses?

1- The place is not ready with all the tools

2- The content does not reach the course provider sufficient time

3- Non-presence of the target category

4- Lack of other ways to attend the attendance course

2. STAKEHOLDERS

Name: Ghala Abdullah Al Hashemi Al Ameer, Software Engineering
Sabah Al-Anazi, cyber security
Juri Al-Harbi, Chemistry
Mayaser Sindhi, Arabic language
Judy Millibary, Artificial Intelligence

Role: student

Requirements Gathering Methods:
One-on-one interview.

QUESTION

Q: Would you prefer to receive a message about the course, such as an email announcement or an application, from which you can choose the appropriate course for you?

A: Definitely an application

Q: What problems did you encounter?

A: Forgetting the time of the course you are registered for

3.GLOSSARY

Term	Definition
Criteria	It refers to the standards or requirements that something must meet in order to be judged or evaluated.
Initial	This means something that happens or exists at the beginning.
Certificate	This is a formal document that verifies something.
Conflicting	This describes things that are incompatible or disagree with each other.
Synchronized	This means things are happening together in a coordinated way.

4.FUNCTIONAL REQUIREMENTS

1. The user can log in using his email and password.

- 1.1 The system should prompt the user to enter their email and password during login.
- 1.2 The system should validate the entered credentials against the stored records in the application database.
- 1.3 The system should display the main interface, showing available courses, if the login credentials are correct.
- 1.4 The system should display a message that the email or password is wrong if the login is incorrect.

2. The user can search for courses by browsing

- 2.1 The system shall display a list of all available courses by searching the database, then arrange them in ascending order.
- 2.2 The system shall display the course title, description, date, day, time, course presenter, and the number of available seats when the user selects the course.

3. The user can search for courses according to the course topic

- 3.1 The system should obtain from the user the course topic.
- 3.2 The system displays a list of all available courses after sorting through searching the database.
- 3.3 The system shall display the course title, description, date, day, time, course presenter, and the number of available seats when the user selects the course.

4. The user can register in the course by course name.

- 4.1 The user can reserve only one seat in the specified course by using the course name
- 4.2 If no seats are available in the selected course, a message should be displayed to the user indicating seat unavailability.
- 4.3 Upon successful booking, the system should send a confirmation message to the user via email.
- 4.4 The available seat count in the database should be updated based on the confirmed bookings.
- 4.5 The system should send a notification one day prior to the booked course date, containing course details such as the title, time, location, and course type.

5. The user can obtain the certificate if he completes the course

- 5.1 The system should record the user's completion status based on course requirements.
- 5.2 The system shall automatically generate certificates that include
The user's name, course title, The name of the course presenter, The date of course completion
- 5.3 The system must send certificates to the user via email after successful completion of the course.

6. The user can create a course by entering the topic, goal, target group, requirements.

- 6.1 The system shall take from the user the topic, objective, target group, and requirements and store them in the database
- 6.2 The system shall allow the user to attach presentations or related files to the course.
- 6.3 The system shall prompt the user to choose the date and time of the course and specify the number of attendees expected.
- 6.4 The system should store the entered course details in database

7. The user can receive a notification 24 hours before the course:

7.1 The system should have a notification state for each user, allowing them to opt-in or opt-out of receiving notifications for courses.

7.2 The system should compare the current time with the scheduled time of each course.

7.3 If the notification state of a user is set to "True" and the course time is at least 24 hours or more before the scheduled time, the system should send a notification to the user.

7.4 The notification should include relevant information about the course, such as the course title, date, time, location, and any additional details.

7.5 The system should ensure the successful delivery of the notification to the user.

4.2 NON-FUNCTIONAL REQUIREMENTS

Security:

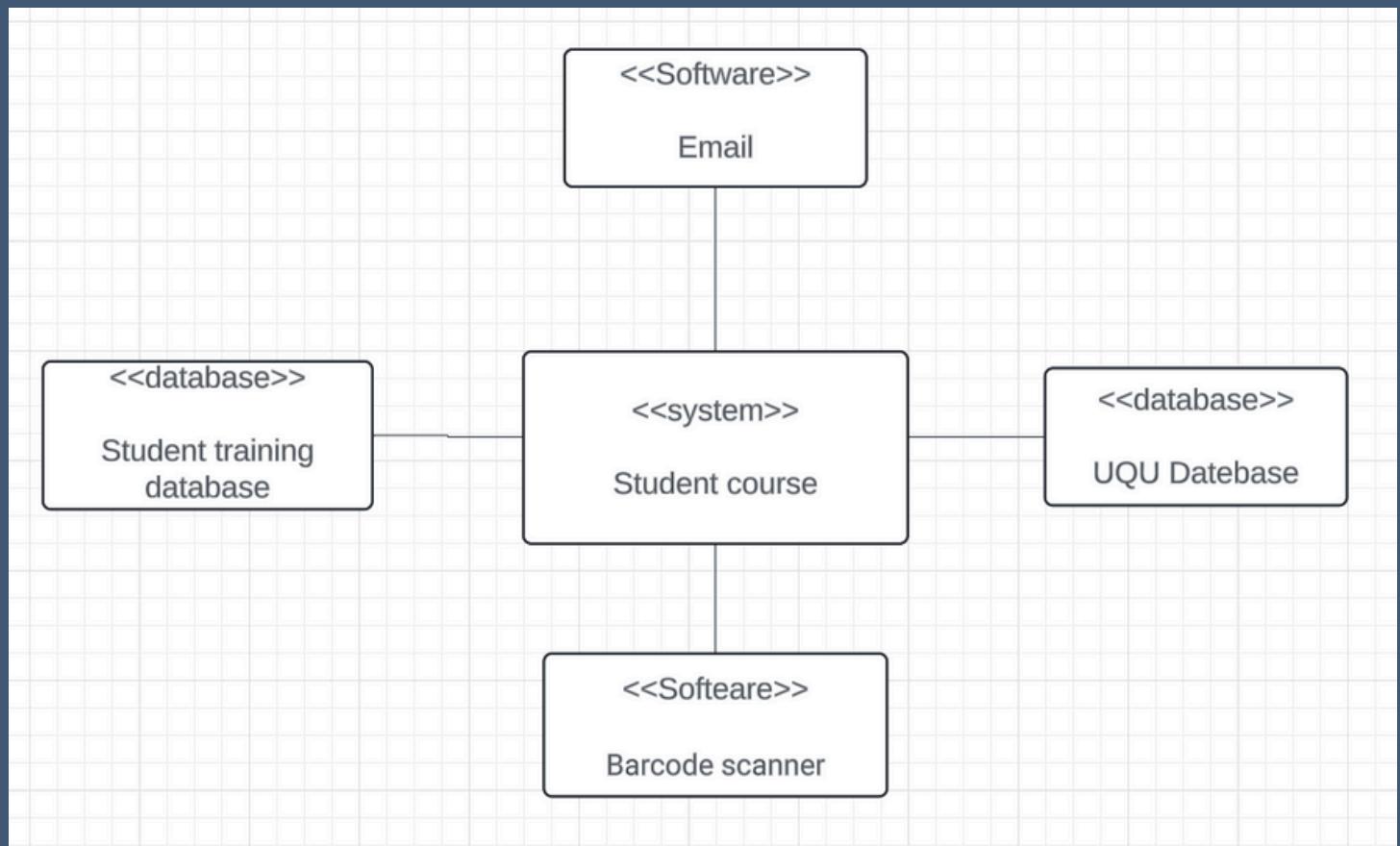
-The system should guarantee the security and confidentiality of user account information, including encryption and access controls.

Internationalization:

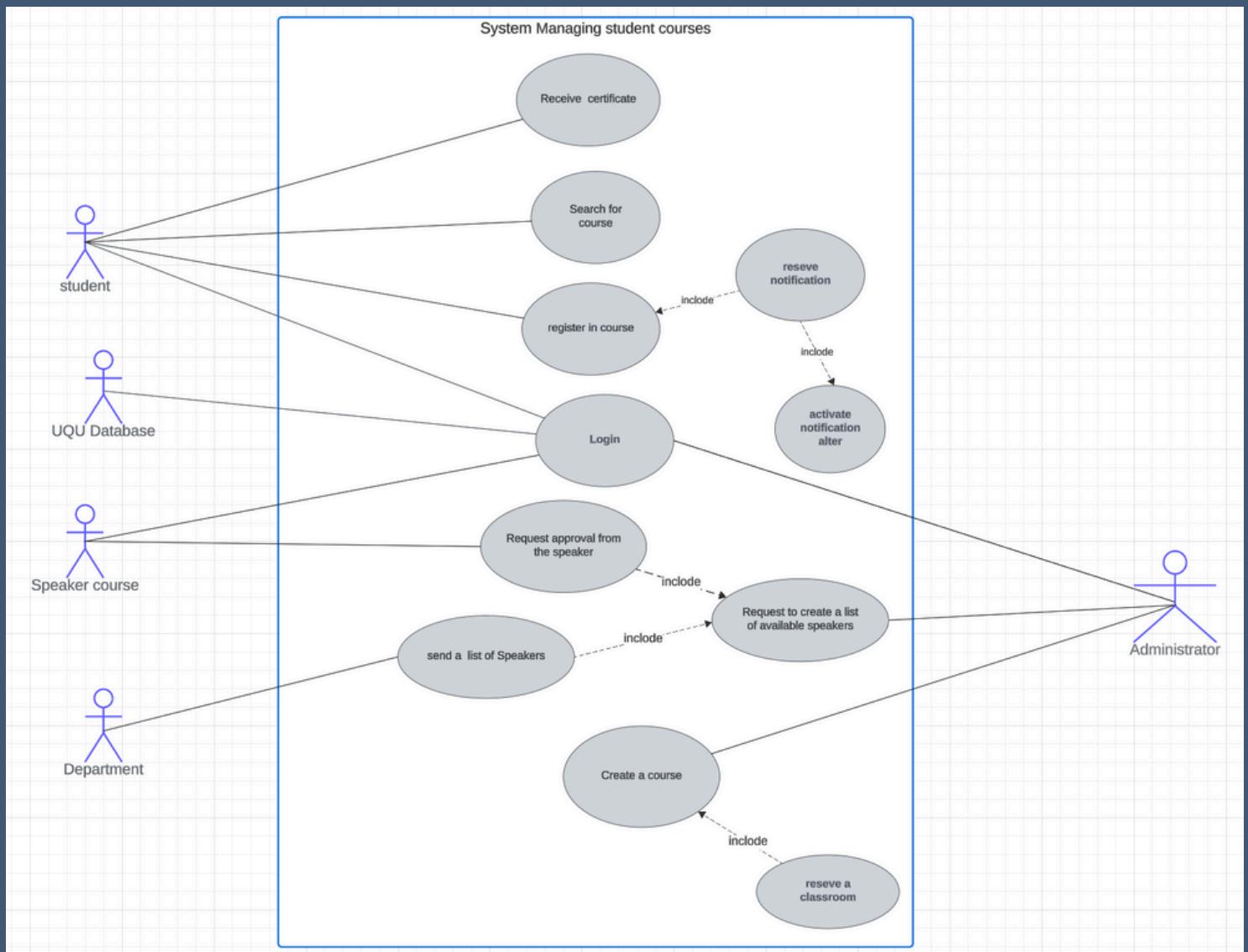
The system is available in both Arabic and English.

5. UML SYSTEM MODELS

1. CONTEXT MODELS:



2.USE CASE DIAGRAM



3.USE CASE TEXT(SCENARIO)

1.LOG IN

Actors:	User: The person who interacts with the system .
Description	This use case describes the process by which a UQU student logs into the training courses system using their university email and password. The system verifies the credentials against the university database and grants access to the main interface displaying available training courses if the credentials are correct. If the credentials are incorrect, the system provides an error message.
Data	<ul style="list-style-type: none">- Input Data:<ul style="list-style-type: none">-Email:The university email address provided by the user.- Password: The password for the university email account. - Output Data:<ul style="list-style-type: none">- User Information: Retrieved from the university database upon successful login.- Error Message:Displayed if the login attempt fails.
Stimulus	The stimulus for this use case is the user's attempt to log in to the training courses system. The stimulus occurs when the user submits their email and password through the login form.
Response	<ul style="list-style-type: none">- Successful Login:<ul style="list-style-type: none">- The system verifies the email and password against the database.- If the credentials are correct, the system displays the main interface with training courses. - Failed Login:<ul style="list-style-type: none">- The system checks the credentials and finds them incorrect.- The system displays an error message indicating that the email or password is incorrect.
Comments	<ul style="list-style-type: none">- The system should ensure that the email and password provided by the user are validated and securely transmitted.- Proper error handling and message display should be implemented to guide the user in case of login failures.- It is recommended to implement security measures such as password encryption and account lockout policies to enhance the security of the login process.

2. SEARCH FOR COURSES BY BROWSING

Actors:	User: The person who interacts with the system and performs the course search.
Description	The user can search for courses by browsing through the available options. The system retrieves the course information from the database and presents it to the user. The courses are listed in ascending order based on certain criteria. When the user selects a specific course, the system displays detailed information including the course title, description, date, day, time, speaker and the number of available seats.
Data	non.
Stimulus	The user initiates the course search by browsing through the available options.
Response	<ul style="list-style-type: none"> - System response to browsing: <ul style="list-style-type: none"> - The system searches the database for available courses. - The system arranges the courses in ascending order based on certain criteria (e.g., course title, date, or time). - The system displays the list of available courses to the user. - System response to course selection: <ul style="list-style-type: none"> - The system retrieves and displays detailed information about the selected course. - The displayed information includes the course title, description, date, day, time, course presenter and the number of available seats.
Comments	<ul style="list-style-type: none"> - The specific criteria for arranging the courses in ascending order should be determined and implemented based on the system's requirements. - The course information displayed should be accurate, up-to-date, and synchronized with the database.

3.FILTERED SEARCH

Actors:	User: The person who interacts with the system and performs the course search.
Description	The user can search for courses based on specific criteria such as course topic, date, or course type (in-person or online). The system then searches the database, retrieves the relevant courses, and arranges them in ascending order. The user is presented with a list of available courses that match the specified criteria. When the user selects a specific course, the system displays detailed information including the course title, description, date, day, time, course presenter and the number of available seats.
Data	Course information stored in the database: Course title, Description, Date, Day, Time, Course presenter and Number of available seats.
Stimulus	The user initiates the course search by providing the desired course topic, selecting a date, or specifying the course type.
Response	<ul style="list-style-type: none">- System response to obtaining search criteria:<ul style="list-style-type: none">- The system prompts the user to enter the desired course topic, select a date from a displayed calendar, or choose the course type.- System response to searching and sorting:<ul style="list-style-type: none">- The system searches the database for available courses based on the provided search criteria.- The system retrieves the relevant courses and arranges them in ascending order based on certain criteria (e.g., course title, date, or time).- The system displays the list of available courses that match the specified criteria to the user.- System response to course selection:<ul style="list-style-type: none">- The system retrieves and displays detailed information about the selected course.- The displayed information includes the course title, description, date, day, time, course presenter and the number of available seats.
Comments	<ul style="list-style-type: none">- The system should provide appropriate user interfaces for entering the course topic, selecting a date, or choosing the course type.- The specific criteria for arranging the courses in ascending order should be determined and implemented based on the system's requirements.- The course information displayed should be accurate, up-to-date, and synchronized with the database.- The system should handle cases where there are no available courses matching the specified search criteria.

4. REGISTER IN COURSE

Actors:	User: The person who interacts with the system and performs the booking a course .
Description	This use case process by which a UQU student registers for a course using the course name. The student can reserve only one seat in the specified course. If the course is full, the system notifies the student of the unavailability. Upon successful booking, a confirmation email is sent to the student, and the available seat count is updated. Additionally, the system sends a reminder notification one day before the course.
Data	<ul style="list-style-type: none"> - Input Data: <ul style="list-style-type: none"> - Course Name: The name of the course the user wishes to register for. - Output Data: <ul style="list-style-type: none"> - Confirmation Email: Sent to the user upon successful registration. - Notification Email: Sent to the user one day prior to the course date. - Error Message: Displayed if no seats are available.
Stimulus	The stimulus for this use case is the user's action to register for a course by providing the course name.
Response	<ul style="list-style-type: none"> - Successful Registration: <ul style="list-style-type: none"> - The system reserves a seat for the user. - The system sends a confirmation email to the user. - The system updates the available seat count in the database. - The system schedules a notification email to be sent one day before the course. - Failed Registration: <ul style="list-style-type: none"> - The system checks the availability of seats and finds none available. - The system displays a message indicating that no seats are available.
COMMENTS	<ul style="list-style-type: none"> - CONCURRENCY: ENSURE THAT THE SYSTEM HANDLES CONCURRENT REGISTRATION ATTEMPTS PROPERLY TO AVOID OVERBOOKING. - EMAIL RELIABILITY: ENSURE THE EMAIL NOTIFICATION SYSTEM IS RELIABLE AND CAN HANDLE SENDING CONFIRMATION AND REMINDER EMAILS PROMPTLY. - USER EXPERIENCE: PROVIDE CLEAR AND INFORMATIVE MESSAGES FOR BOTH SUCCESSFUL AND FAILED REGISTRATION ATTEMPTS. - SCALABILITY: THE SYSTEM SHOULD EFFICIENTLY MANAGE A LARGE NUMBER OF USERS AND COURSE REGISTRATIONS.

5. RESEVE NOTIFICATION

Actors:	User: The person who interacts with the system.
Description	This use case process by which the system sends a notification to a UQU student 24 hours before a registered course. The system checks if the user has opted in to receive notifications and compares the current time with the course time. If the conditions are met, the system sends a notification with relevant course details.
Data	<ul style="list-style-type: none">- Input Data:<ul style="list-style-type: none">- User Notification State: Indicates whether the user has opted in or out of notifications.- Course Schedule: Date and time of the course. - Output Data:<ul style="list-style-type: none">- Notification Message: Includes course title, date, time, location, and additional details.
Stimulus	The stimulus for this use case is the system's scheduled check for upcoming courses 24 hours before their start time.
Response	<ul style="list-style-type: none">-Successful Notification:<ul style="list-style-type: none">-The system verifies that the user has opted in to receive notifications.-The system sends a notification with the relevant course details to the user. -Failed Notification:<ul style="list-style-type: none">-The system identifies issues in sending the notification and logs the error for troubleshooting.
Comments	<ul style="list-style-type: none">- User Preferences: Allow users to manage their notification preferences easily.- Reliability: Ensure the notification system is robust and capable of handling large volumes of notifications.- User Experience: Notifications should be clear and provide all necessary information in a concise format.

6. REQUEST TO CREATE A LIST OF AVAILABLE SPEAKERS

Actors:	- User: Student training department
Description	This use case outlines the process by which a UQU student training department requests a list of available updaters from a target department. The system prompts the user to provide the necessary information and facilitates sending a request message to the target department.
Data	<p>- Input Data:</p> <ul style="list-style-type: none">- Target Department: The department to which the request will be sent.- Message: The message written by the user to request an interview with the speaker. <p>- Output Data:</p> <ul style="list-style-type: none">- Confirmation Message: Indicates successful sending of the request to the target department.
Stimulus	The stimulus for this use case is the user's request to obtain a list of available updaters from a specific department.
Response	<p>- Successful Request:</p> <ul style="list-style-type: none">- The system prompts the user to provide the target department and compose a message.- The user provides the required information.- The system sends the message to the target department.- The system displays a confirmation message indicating the successful sending of the request. <p>- Failed Request:</p> <ul style="list-style-type: none">- If there are issues sending the message, the system displays an error message and prompts the user to try again later.
COMMENTS	<ul style="list-style-type: none">- User Input: Ensure the system guides the user effectively in providing the required information.- Message Composition: Provide a user-friendly interface for composing the request message.- Error Handling: Implement error handling mechanisms to address issues such as network errors during message transmission.

7. SEND A LIST OF SPEAKERS

Actors:	- User: Target department
Description	This use case allows the target department to request a list of available speakers. The system facilitates the process by delivering the list creation request to the user, allowing them to input a list of available speakers, and then sending the completed list to the system administrator.
Data	<ul style="list-style-type: none">- Input Data:<ul style="list-style-type: none">- List of Available Speakers: Typed by the user.- Output Data:<ul style="list-style-type: none">- List of Available Speakers: Sent to the system administrator.
Stimulus	The stimulus for this use case is the user's request to create a list of available speakers.
Response	<ul style="list-style-type: none">- Successful List Creation:<ul style="list-style-type: none">- The system delivers the list creation request to the user.- The user types a list of available speakers.- The system sends the completed list to the system administrator.- Failed List Creation:<ul style="list-style-type: none">- If there are issues sending the list or if the user encounters any errors during list creation, the system displays an error message and prompts the user to try again.
Comments	<ul style="list-style-type: none">- User Input: Ensure the system provides clear instructions to the user for creating the list.- Administrator Notification: Implement a mechanism to notify the administrator upon receiving the list.- Error Handling: Provide appropriate error messages and guidance to the user in case of errors or issues during list creation.

8. SEND INVITATION TO SPEAKER FOR COURSE APPROVAL

Actors:	- User: Student training department
Description	This use case outlines how a UQU Student training department sends an invitation to a speaker, requesting approval to offer a course. The system prompts the user to compose and send a message to the list of candidates. It then monitors the response from the speaker, waiting for two days. If there is no response within the specified time, the system prompts the user to choose another speaker. The system also notifies administrators about the progress and outcomes of each attempt to obtain approval.
Data	<ul style="list-style-type: none"> - Input Data: <ul style="list-style-type: none"> - Message: Written by the user to request approval from the speaker. - Response Time: Two days. - Output Data: <ul style="list-style-type: none"> - Notification: Sent to administrators about the progress and outcomes of each attempt to obtain approval.
Stimulus	The stimulus for this use case is the user's request to send an invitation to a speaker for course approval.
Response	<ul style="list-style-type: none"> - Successful Invitation and Approval: <ul style="list-style-type: none"> - The system displays a text box and prompts the user to write a message to the list of candidates. - The user composes and sends the invitation message to the speaker. - The system monitors the response from the speaker for two days. - If the speaker approves the course within the specified time, the system proceeds with course offering. - No Response from Speaker: <ul style="list-style-type: none"> - If the speaker does not respond within two days, the system prompts the user to choose another speaker.
Comments	<ul style="list-style-type: none"> - User Input: Ensure the system guides the user effectively in composing the invitation message. - Response Time: Define a clear timeframe for the speaker to respond to the invitation. - Administrator Notification: Implement a mechanism to notify administrators about the progress and outcomes of each attempt to obtain approval. - User Experience: Provide clear instructions and notifications to the user throughout the process.

9. RESERVE A CLASSROOM

Actors:	- User: Student training department
Description	This use case describes the process by which a UQU Student training department reserves a classroom by entering the date and time of the course and the expected number of attendees. The system retrieves the necessary information from the user, searches the database for available classrooms, and updates the classroom's status to "Reserved" to ensure no conflicting bookings.
Data	<ul style="list-style-type: none">- Input Data:<ul style="list-style-type: none">- Course Date and Time: Provided by the user.- Expected Number of Attendees: Provided by the user. - Output Data:<ul style="list-style-type: none">- Available Classroom: Displayed to the user based on the search criteria.- Reservation Confirmation: Updates the database to indicate the classroom is reserved.
Stimulus	The stimulus for this use case is the user's request to reserve a classroom for a course.
Response	<ul style="list-style-type: none">- Successful Reservation:<ul style="list-style-type: none">- The system obtains the course date, time, and expected number of attendees from the user.- The system searches the database for available classrooms that match the criteria.- The system displays an available classroom to the user.- Upon user confirmation, the system updates the classroom's status to "Reserved" and saves the reservation details in the database. - Failed Reservation:<ul style="list-style-type: none">- If no available classrooms match the criteria, the system informs the user and prompts them to adjust their criteria or try again later.
Comments	<ul style="list-style-type: none">- User Input: Ensure the system provides clear instructions for entering the course date, time, and number of attendees.- Conflict Avoidance: Implement mechanisms to prevent double bookings and ensure accurate status updates.- User Experience: Provide a user-friendly interface for searching and reserving classrooms.

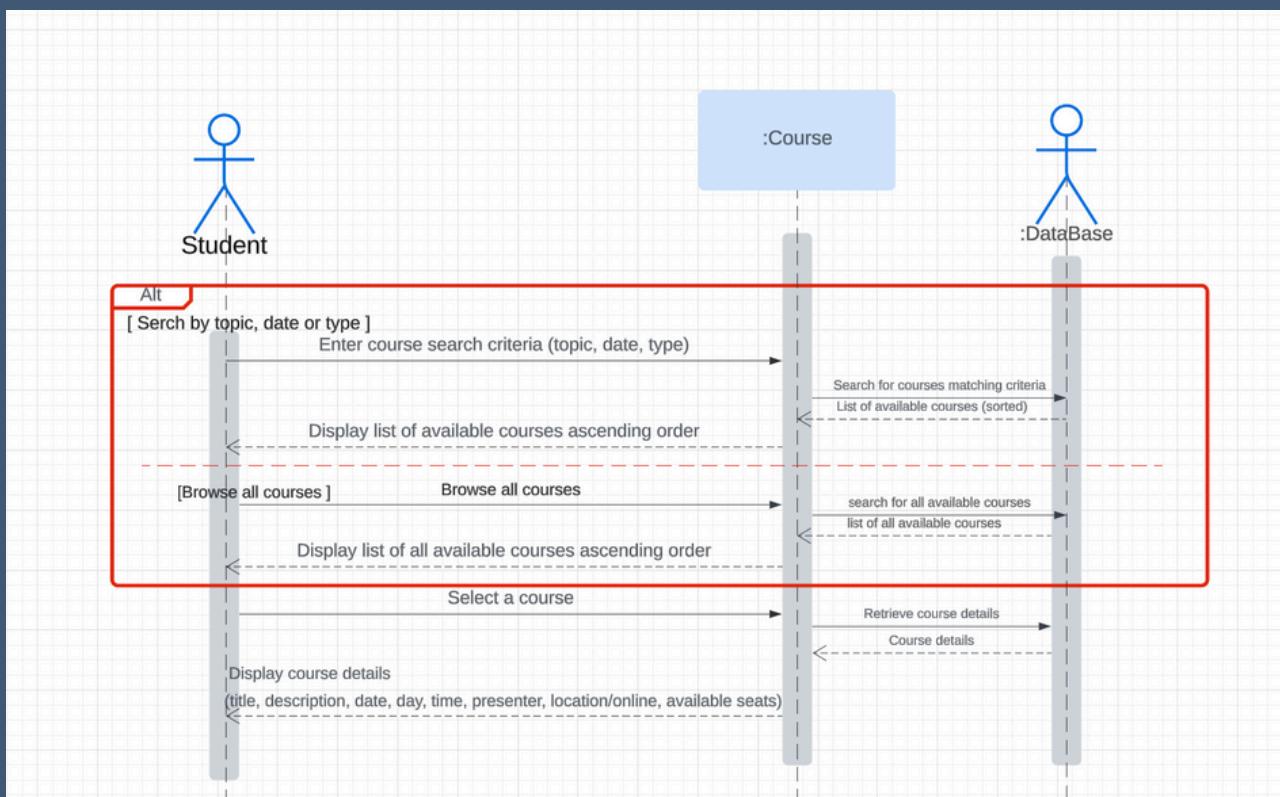
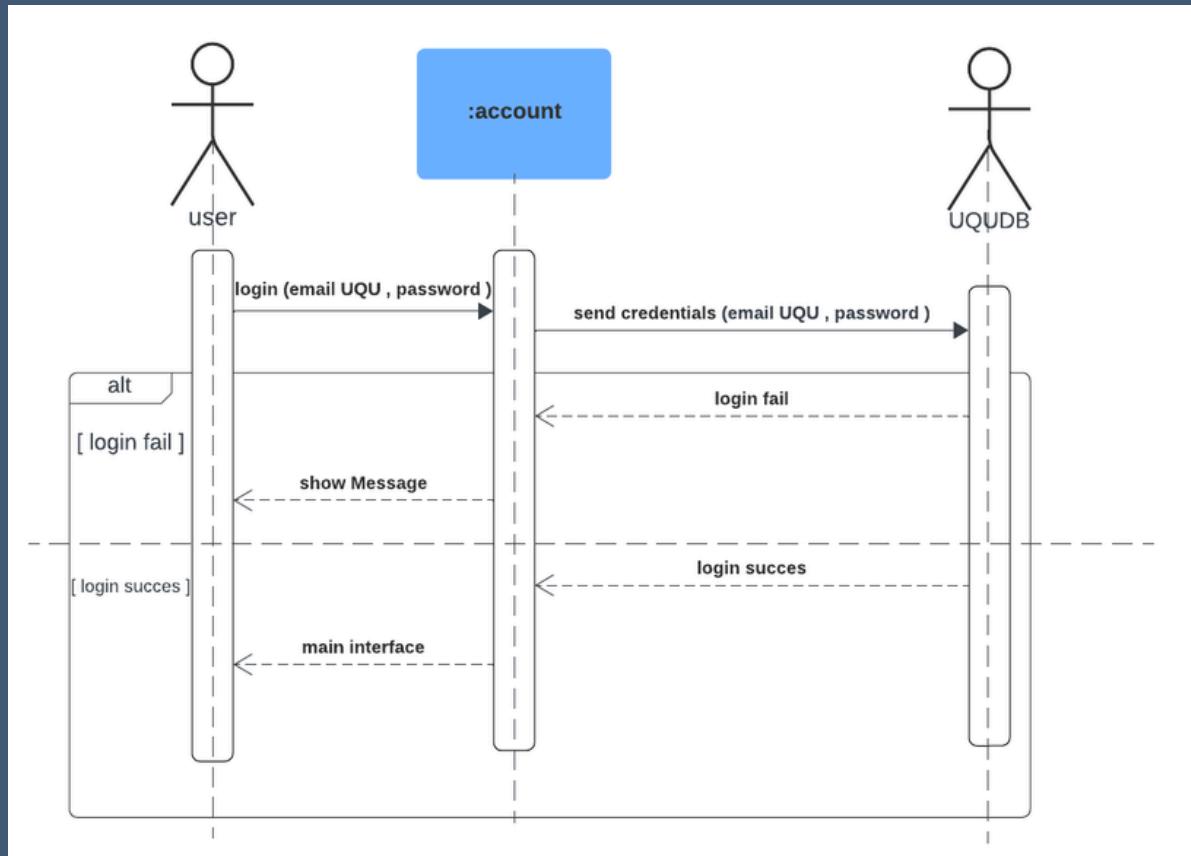
10.CREATE A COURSE

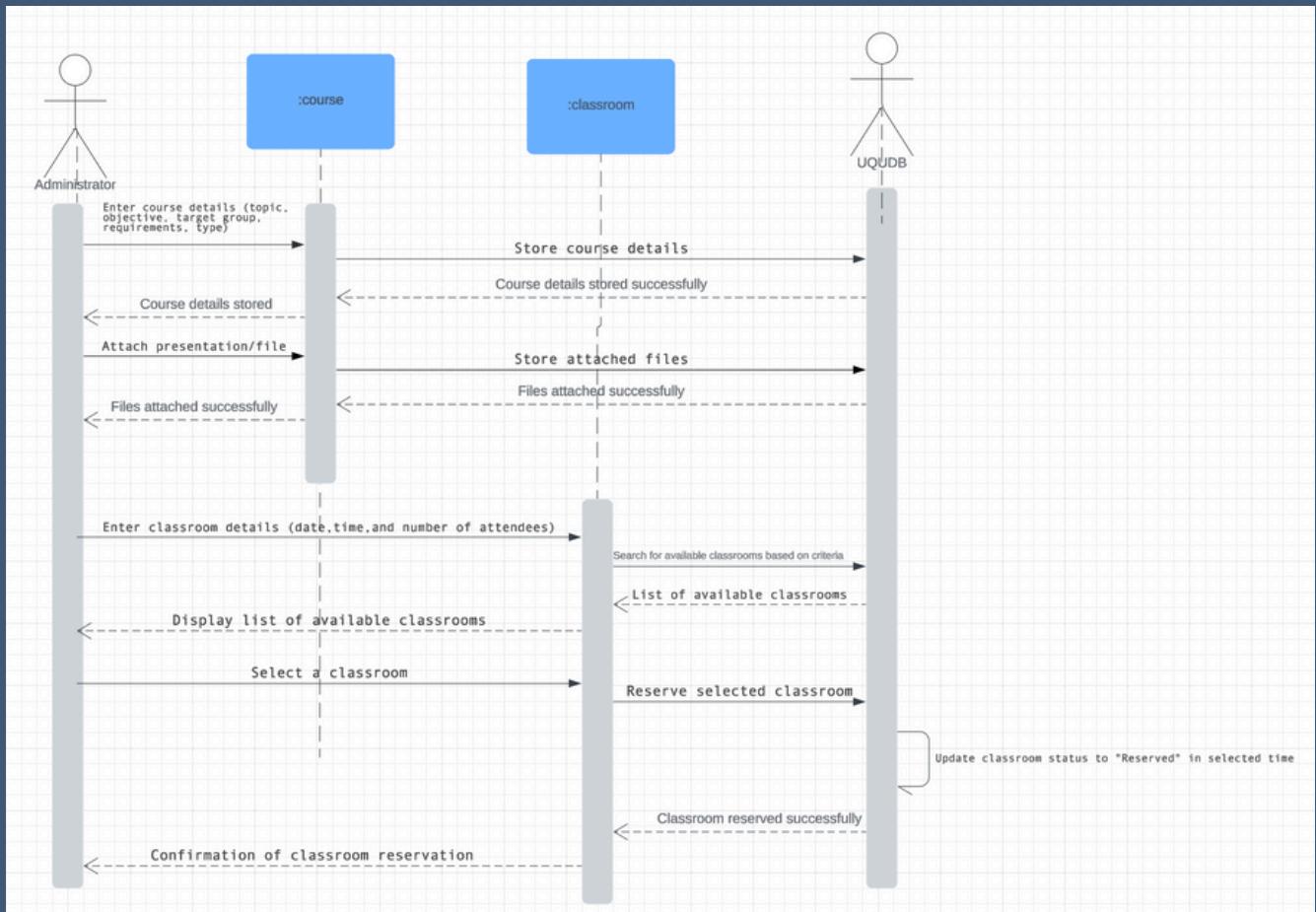
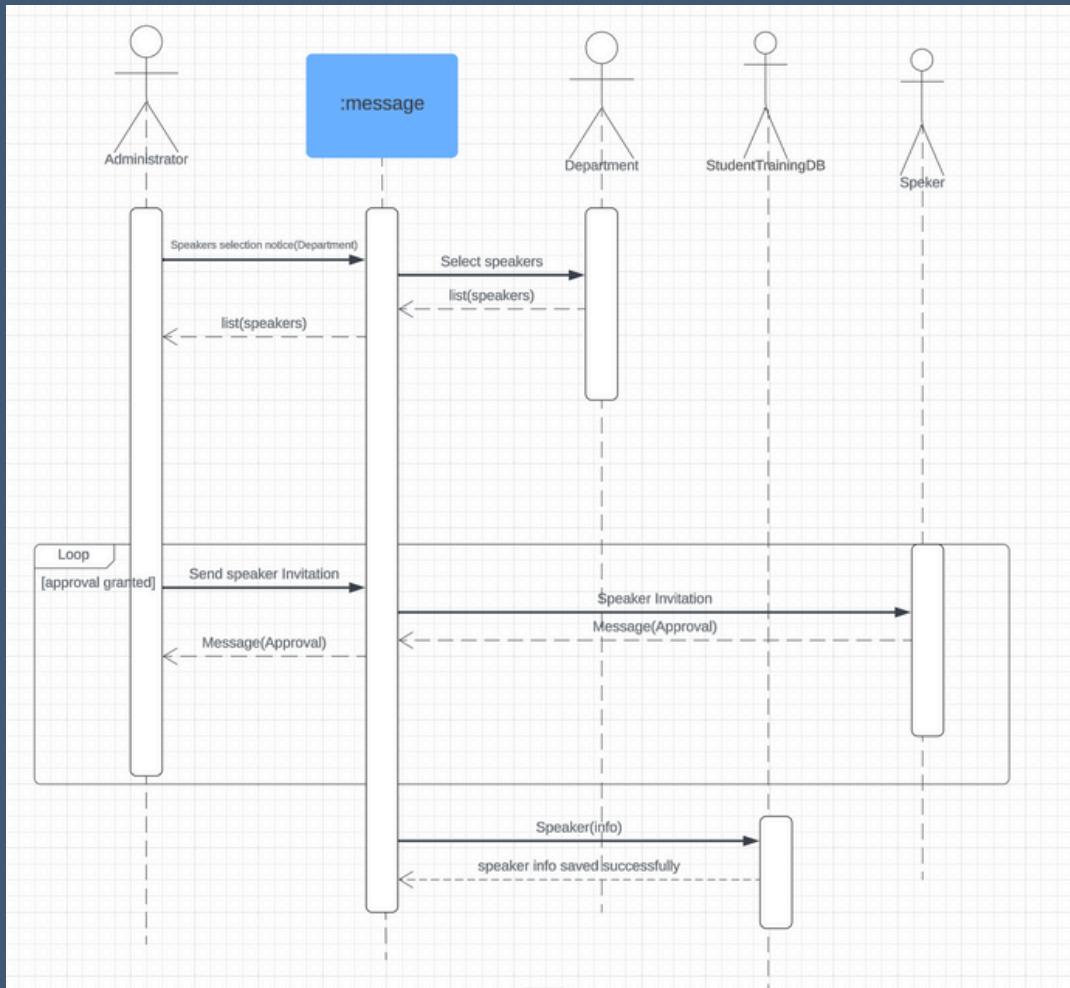
Actors:	- User: Student training department
Description	This use case describes how a UQU Student Training Department creates a new course by entering the topic, goal, target group, and requirements. The system also allows the user to attach presentations or related files, choose the date and time of the course, and specify the expected number of attendees. All entered details are stored in the Student Training Department.
Data	<ul style="list-style-type: none"> - Input Data: <ul style="list-style-type: none"> - Course Topic: Provided by the user. - Course Objective: Provided by the user. - Target Group: Provided by the user. - Course Requirements: Provided by the user. - Attached Files: Presentations or related files attached by the user. - Course Date and Time: Chosen by the user. - Expected Number of Attendees: Specified by the user. - Output Data: <ul style="list-style-type: none"> - Course Details: Stored in the Student Training Department database.
Stimulus	The stimulus for this use case is the user's request to create a new course.
Response	<ul style="list-style-type: none"> - Successful Course Creation: <ul style="list-style-type: none"> - The system prompts the user to enter the topic, objective, target group, and requirements. - The user provides the required course details. - The system allows the user to attach presentations or related files. - The user attaches the necessary files. - The system prompts the user to choose the date and time of the course and specify the expected number of attendees. - The user selects the date, time, and specifies the expected number of attendees. - The system stores all entered course details in the Student Training Department database. - Failed Course Creation: <ul style="list-style-type: none"> - If there are issues during the course creation process, the system displays an error message and prompts the user to correct the issues.
Comments	<ul style="list-style-type: none"> - User Input: Ensure the system provides clear instructions and validations for entering course details. - File Attachments: Support various file formats for presentations and related files. - Conflict Avoidance: Implement mechanisms to avoid scheduling conflicts when choosing the course date and time. - User Experience: Provide a user-friendly interface for creating and managing courses.

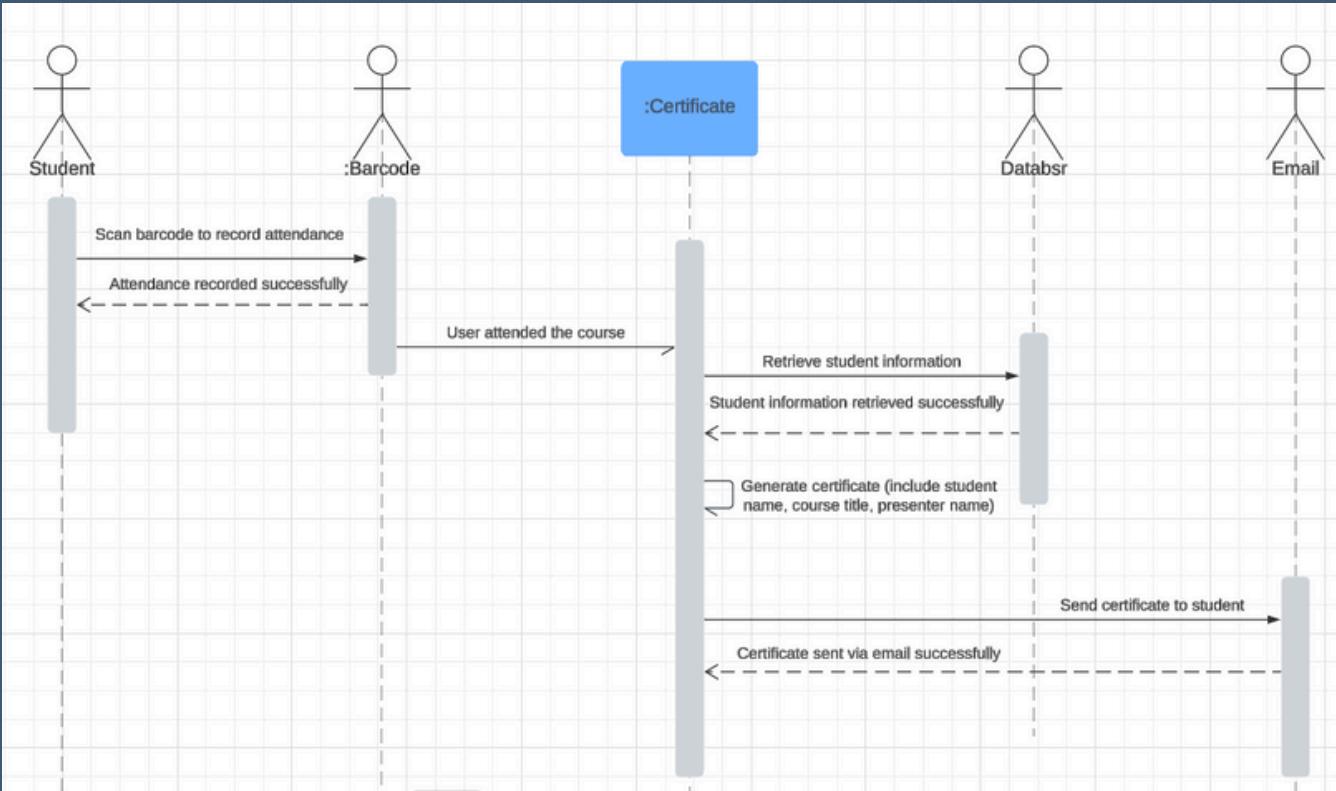
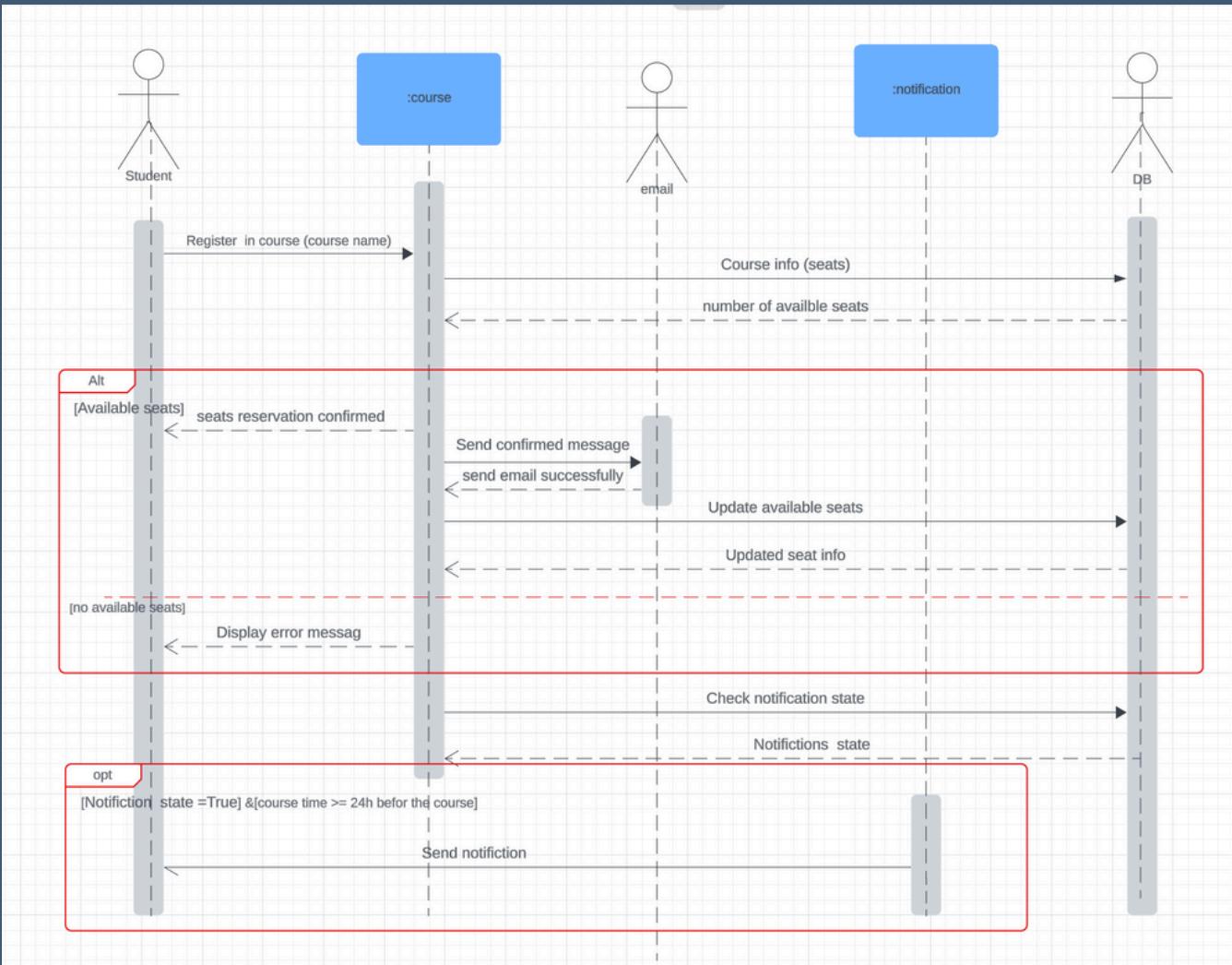
11.RECEIVE CERTIFICATE

Actors:	- User:UQU Student.
Description	This use case describes the process by which a UQU student obtains a certificate upon completing a course. The system records attendance by scanning a barcode, automatically generates certificates with relevant details, and sends them to participants via email.
Data	<ul style="list-style-type: none"> - Input Data: <ul style="list-style-type: none"> - Student Attendance: Recorded by scanning the barcode. - Course Details: Course title, presenter name, and completion status. - Output Data: <ul style="list-style-type: none"> - Generated Certificate: Includes the student's name, course title, and course presenter. - Email with Certificate: *Sent to the student upon course completion.
Stimulus	The stimulus for this use case is the student's completion of the course and their attendance being recorded.
Response	<p>Successful Certificate Generation:</p> <ul style="list-style-type: none"> - The system records the student's attendance by scanning the barcode. - Upon course completion, the system generates a certificate with the student's name, course title, and course presenter. - The system sends the generated certificate to the student's email. <p>Failed Certificate Generation:</p> <ul style="list-style-type: none"> - If there are issues with recording attendance or generating the certificate, the system displays an error message.
Comments	<ul style="list-style-type: none"> - Attendance Recording: Ensure accurate and reliable recording of student attendance through barcode scanning. - Certificate Generation: Implement a robust certificate generation process that includes necessary details and ensures no errors. - Email Delivery: Ensure reliable email delivery with appropriate error handling in case of delivery failures. - User Experience: Provide clear instructions and notifications to the user throughout the process.

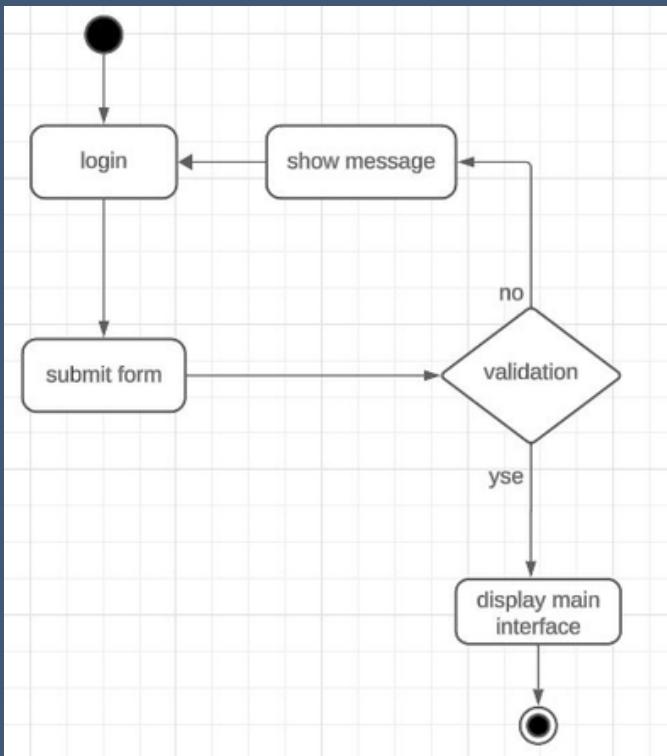
4. SEQUENCE DIAGRAMS





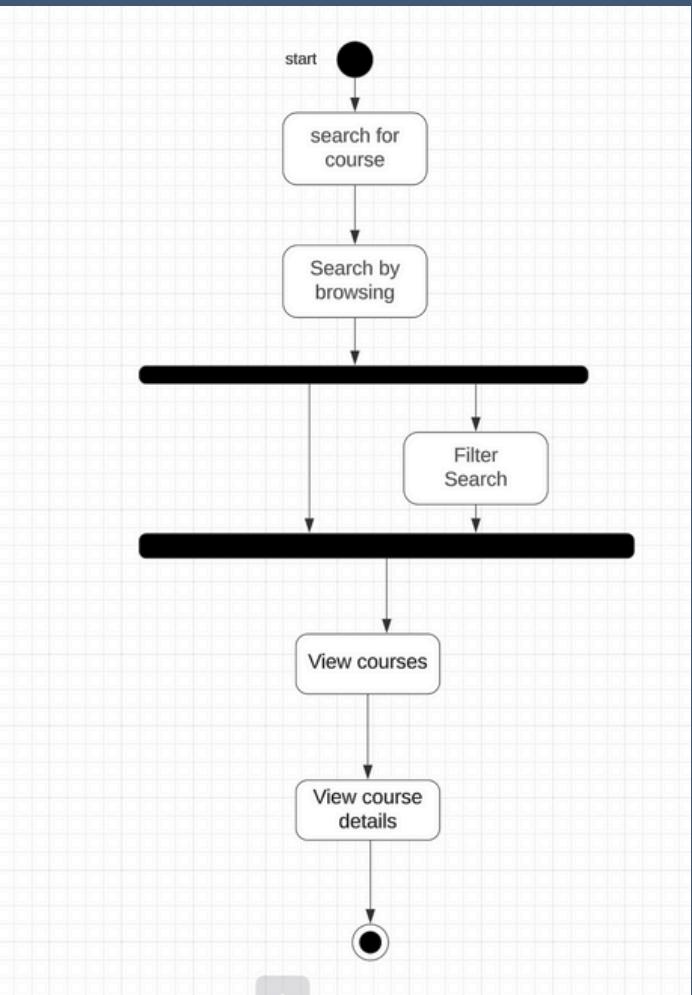


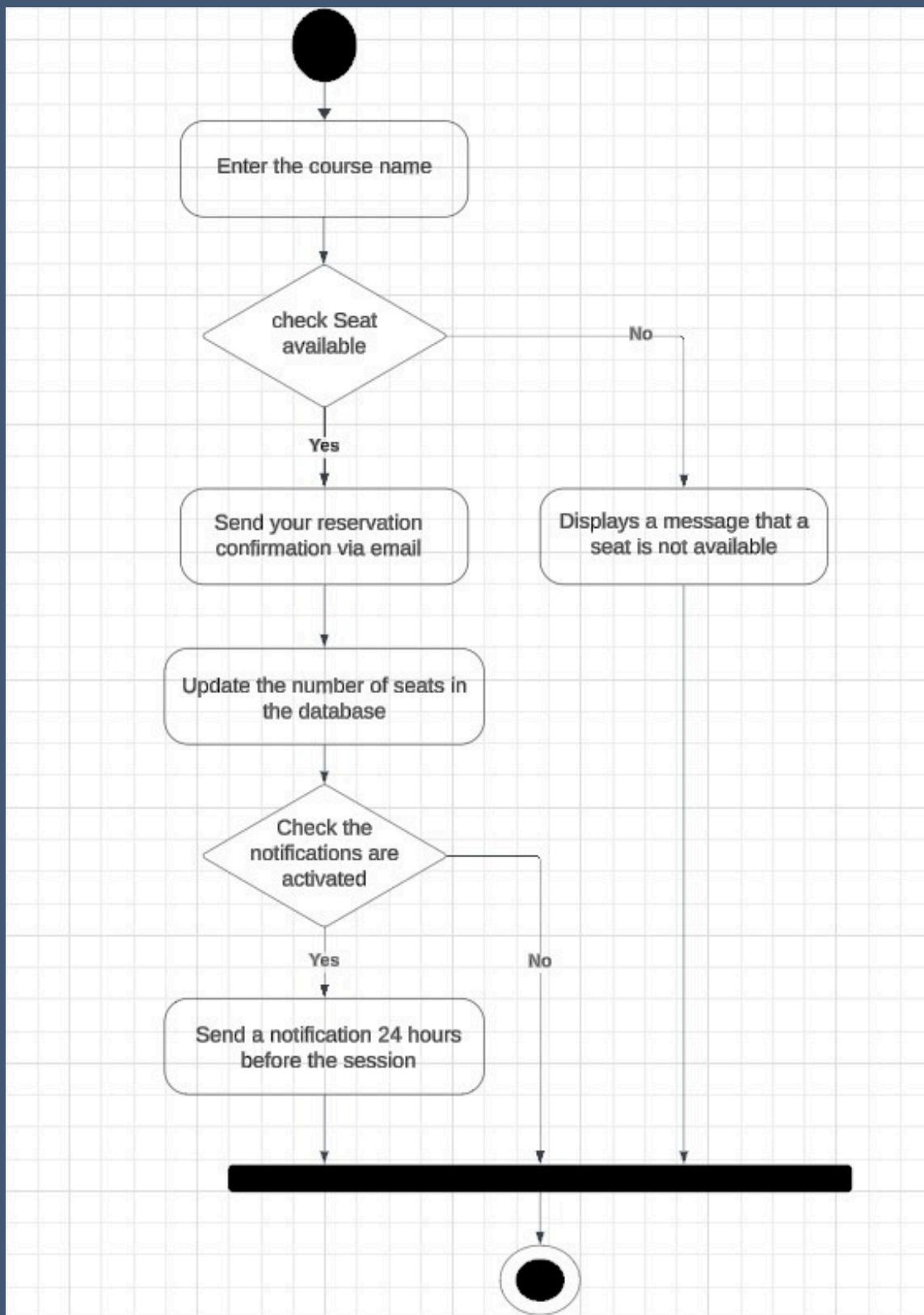
5.ACTIVE DIAGRAM:

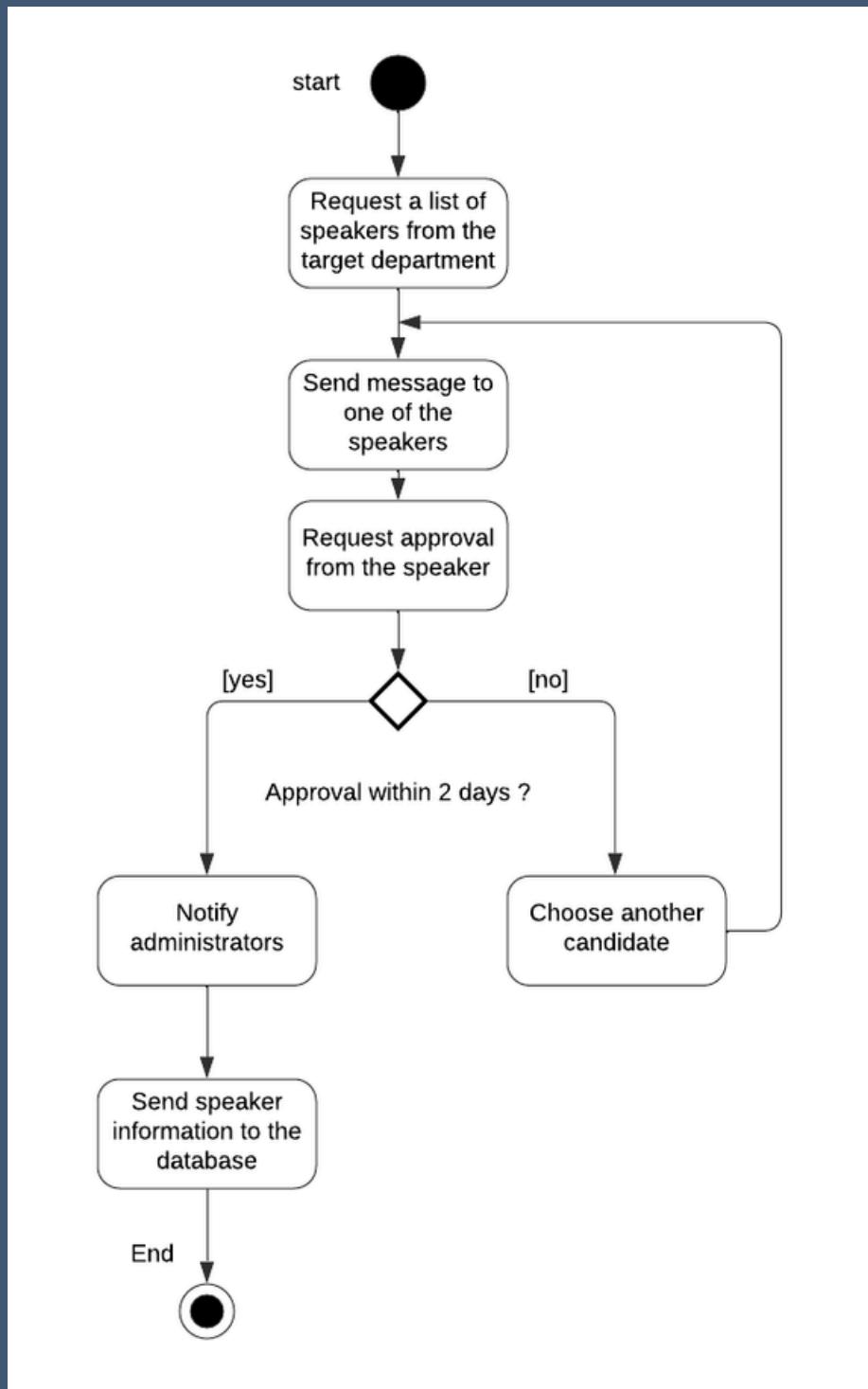


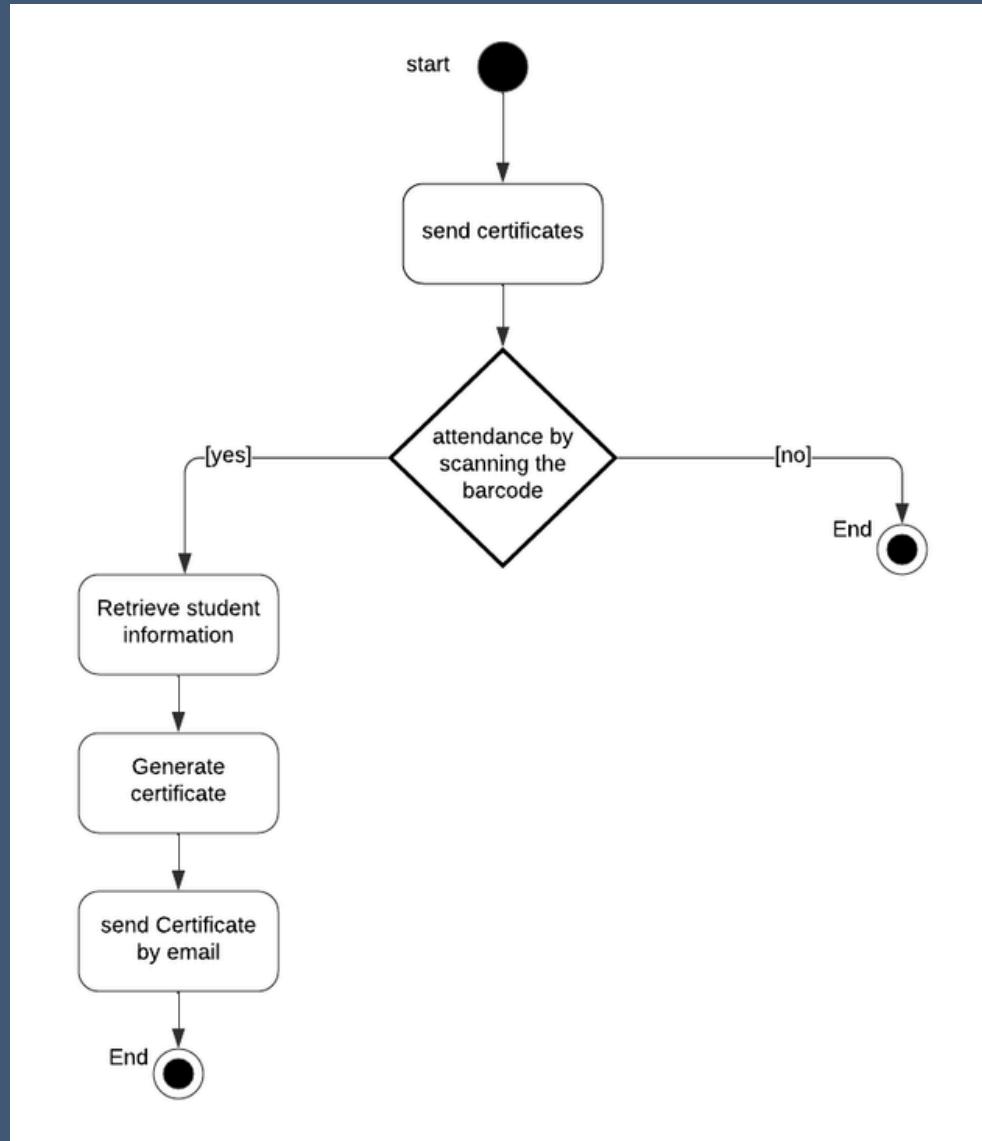
1

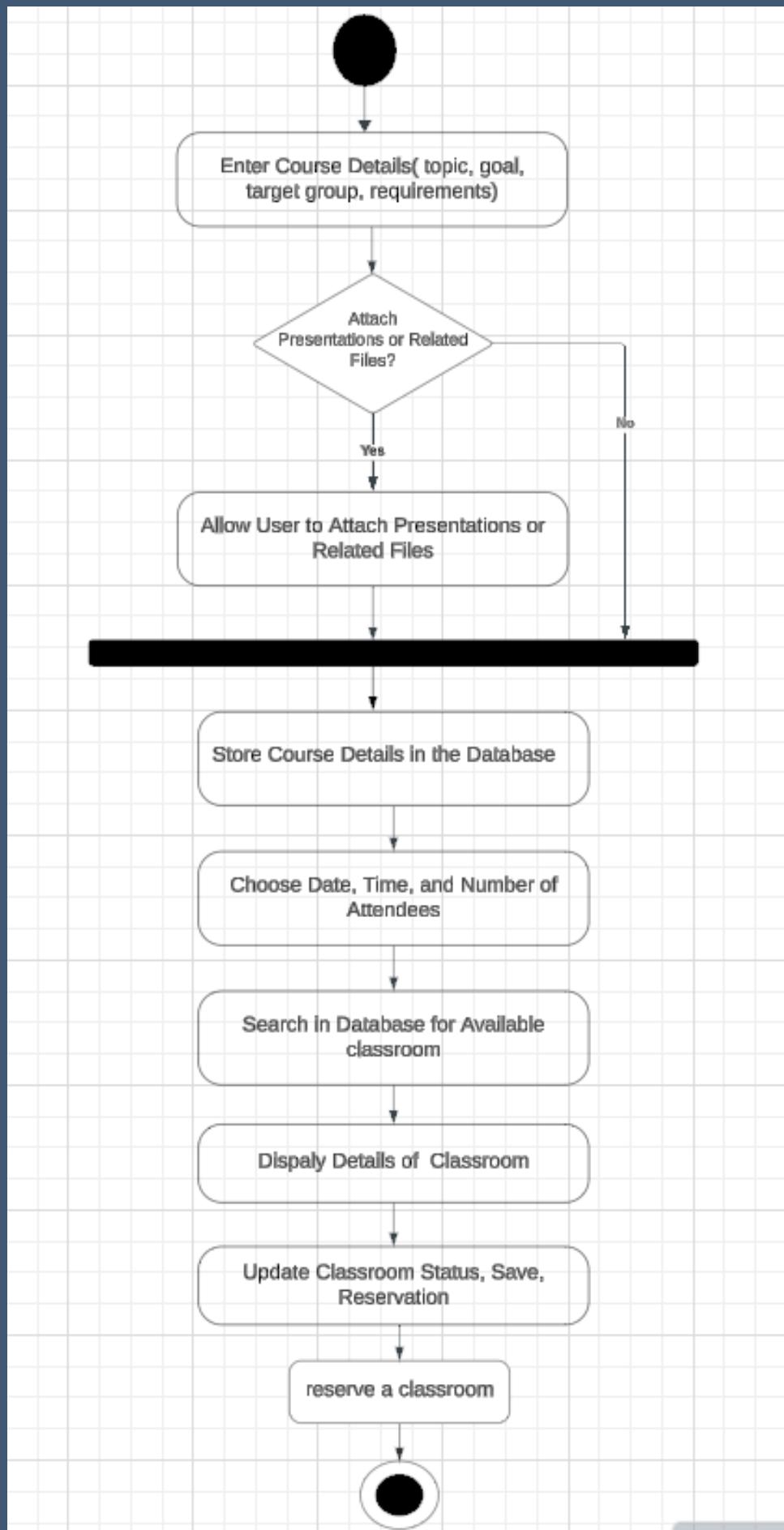
2



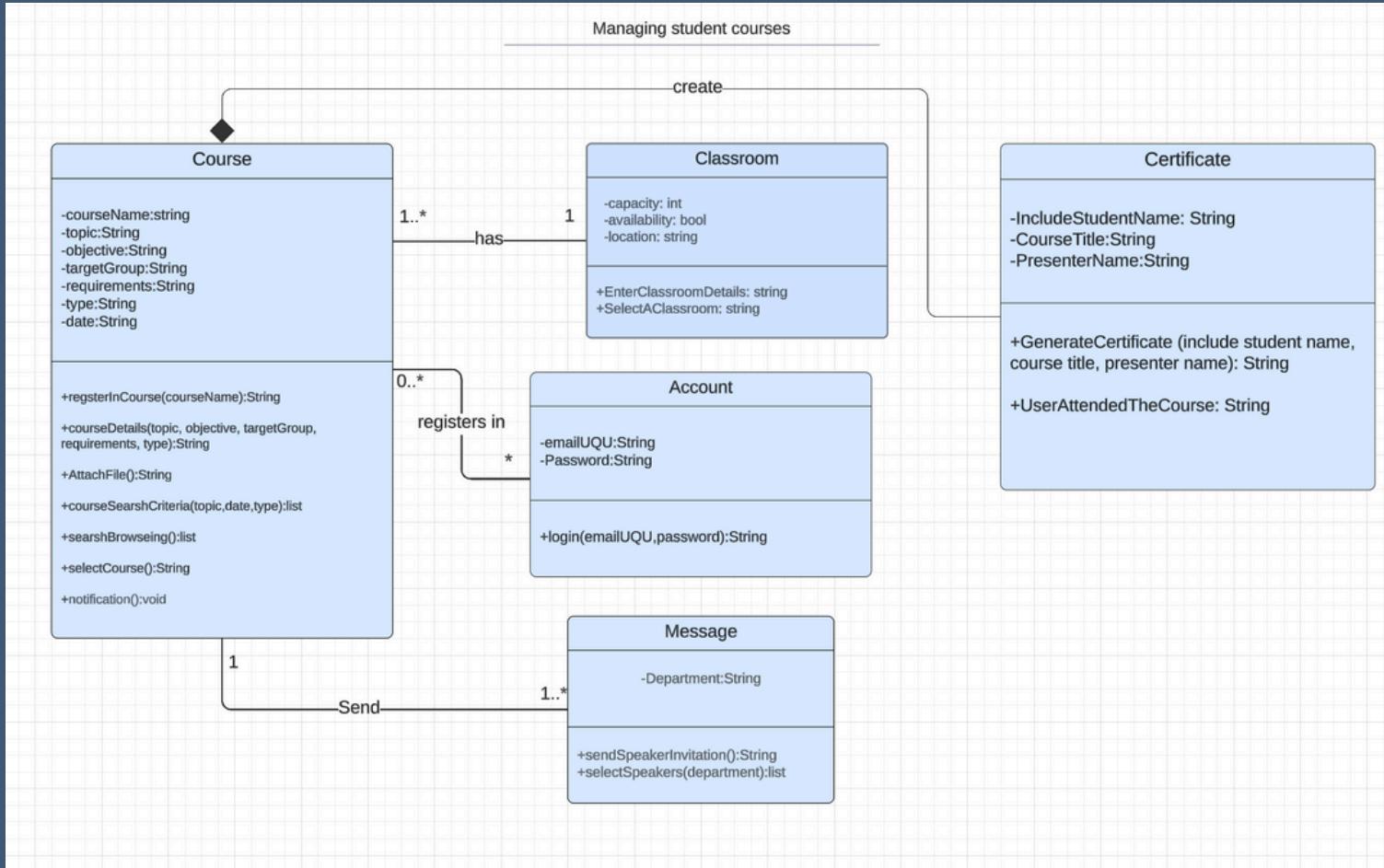




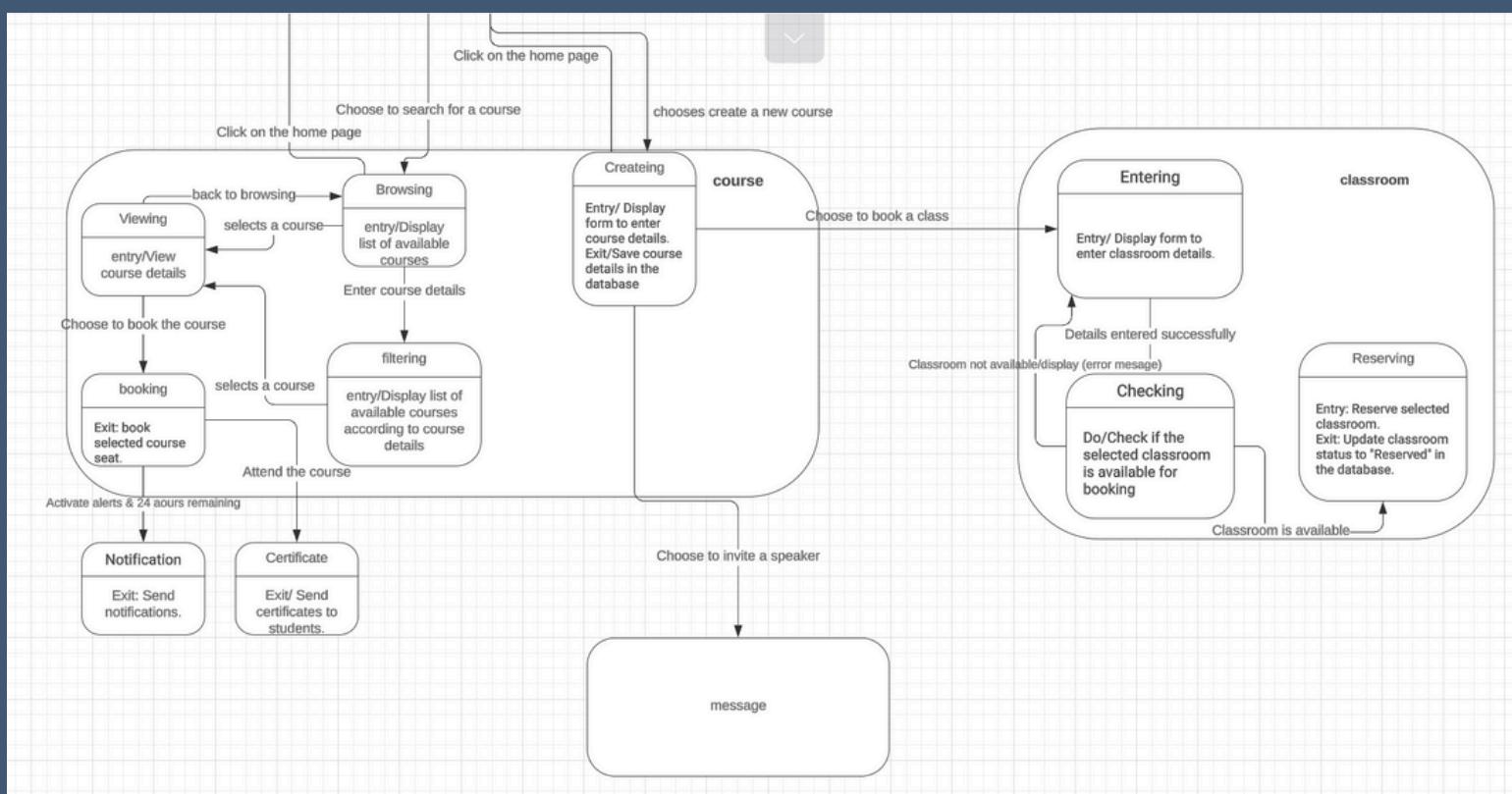
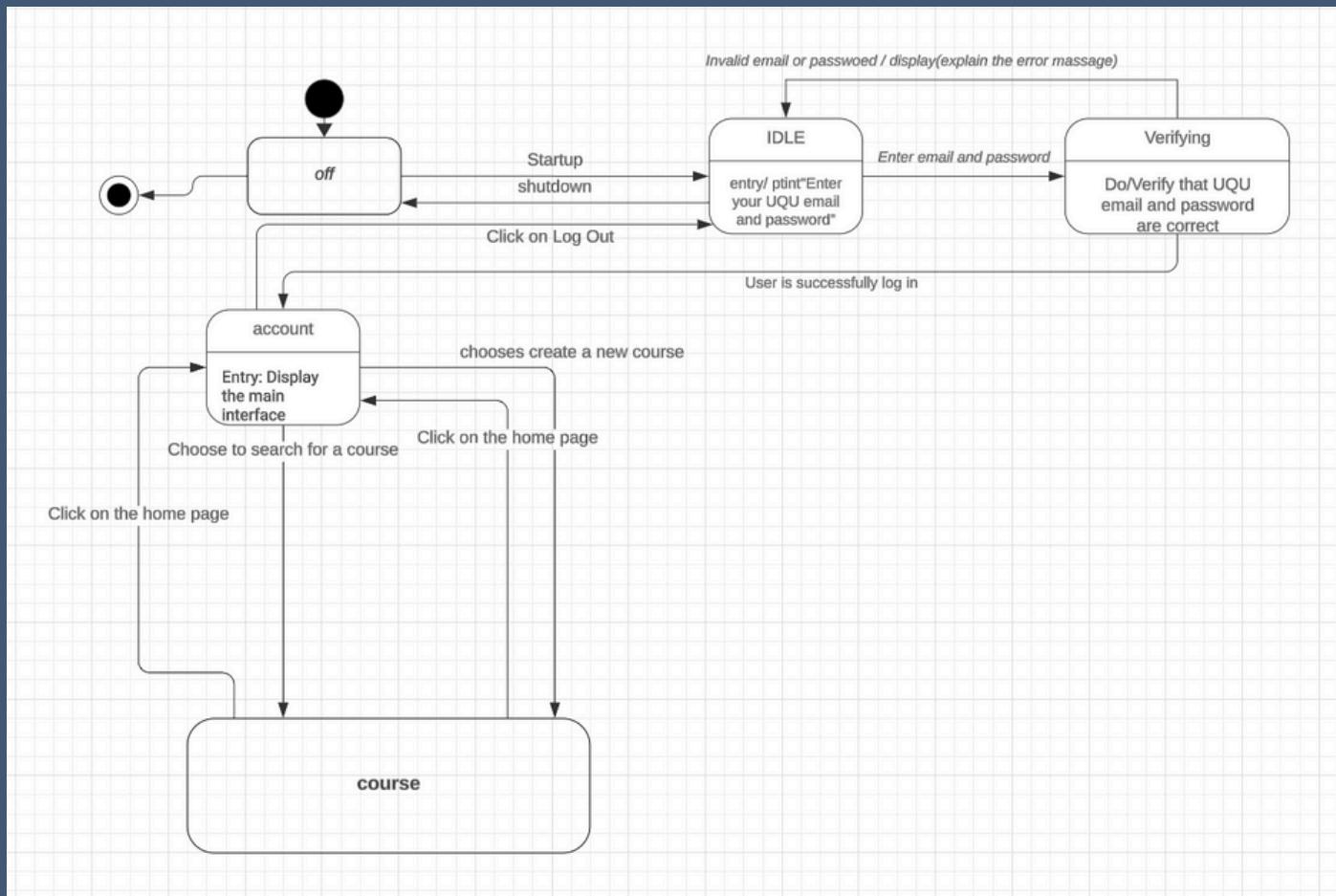


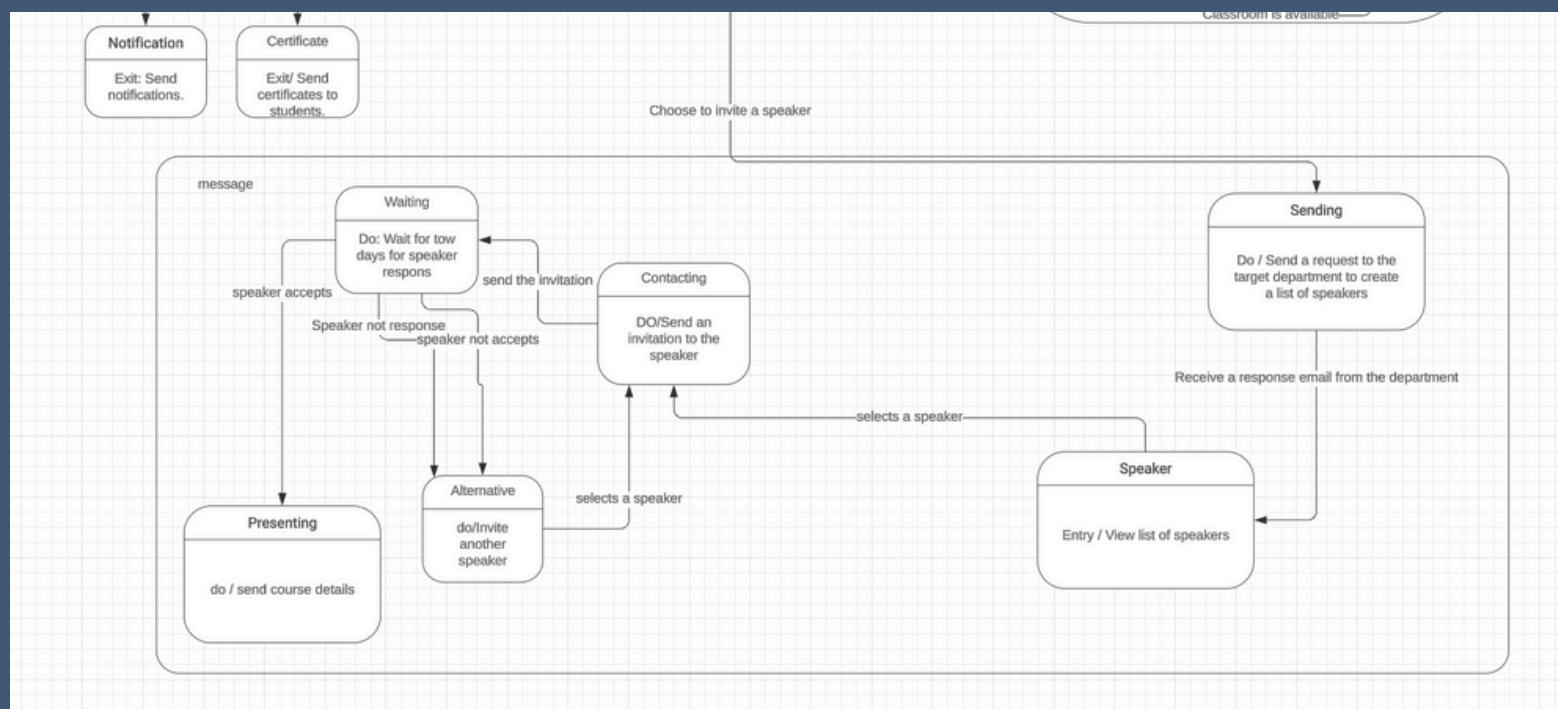


6.CLASS DIAGRAM

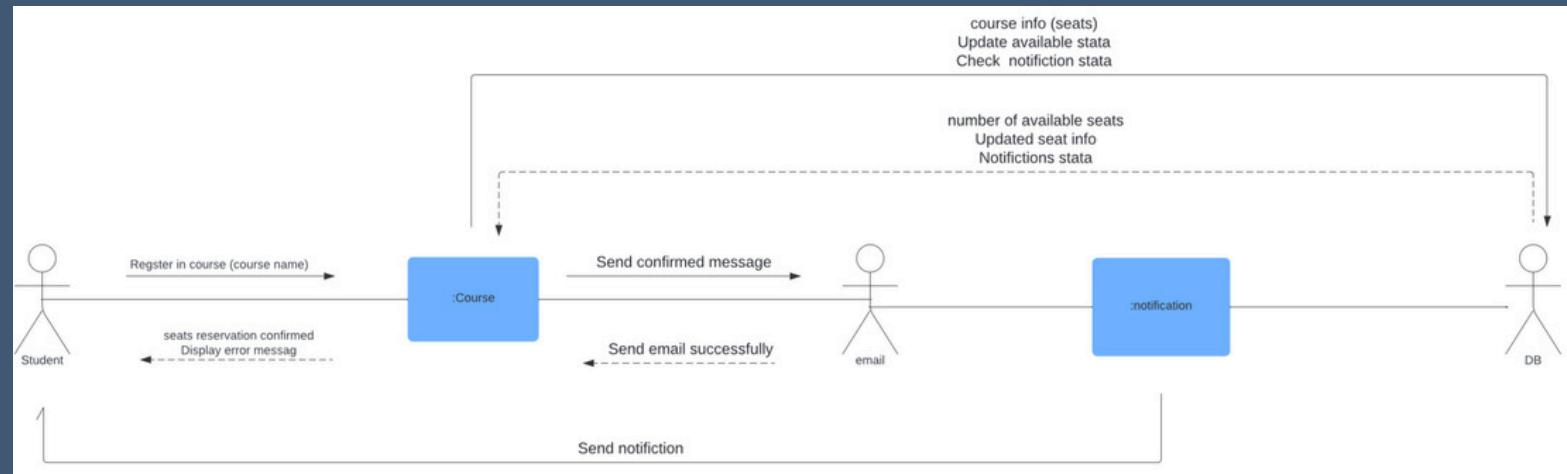
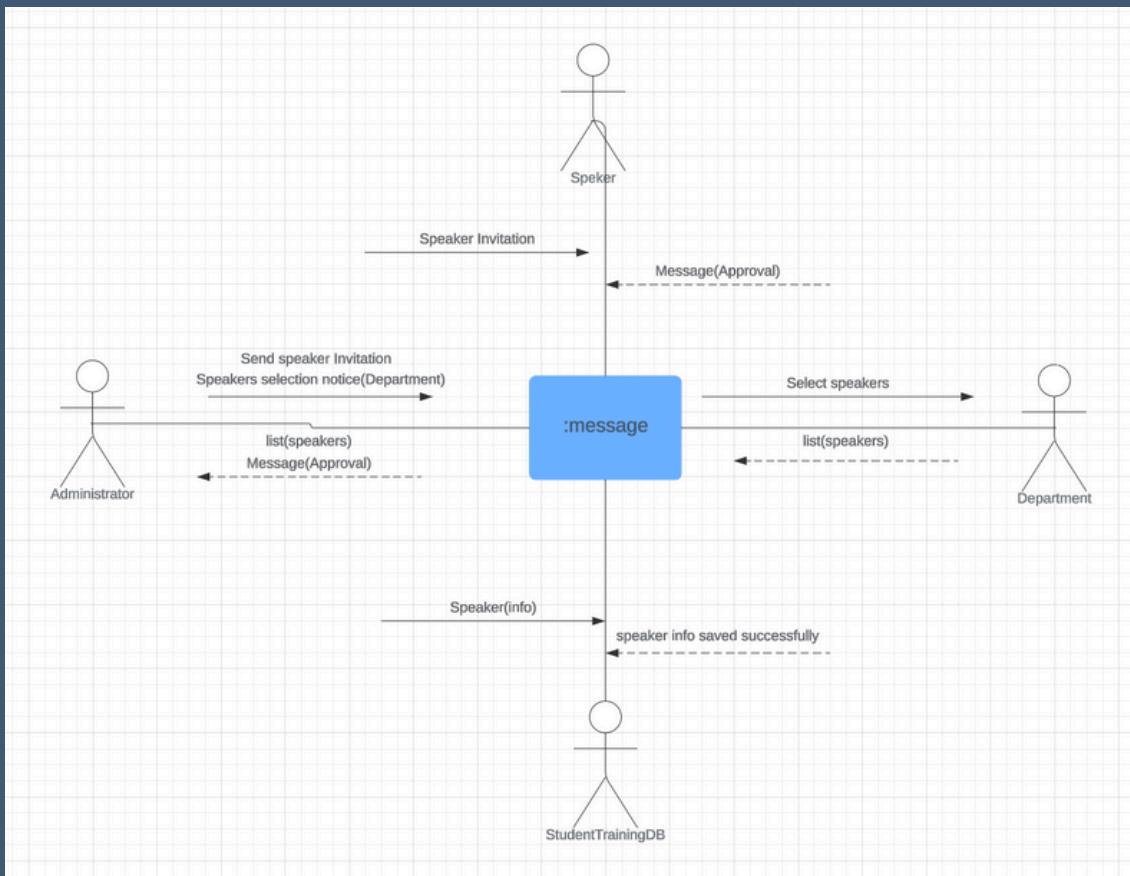
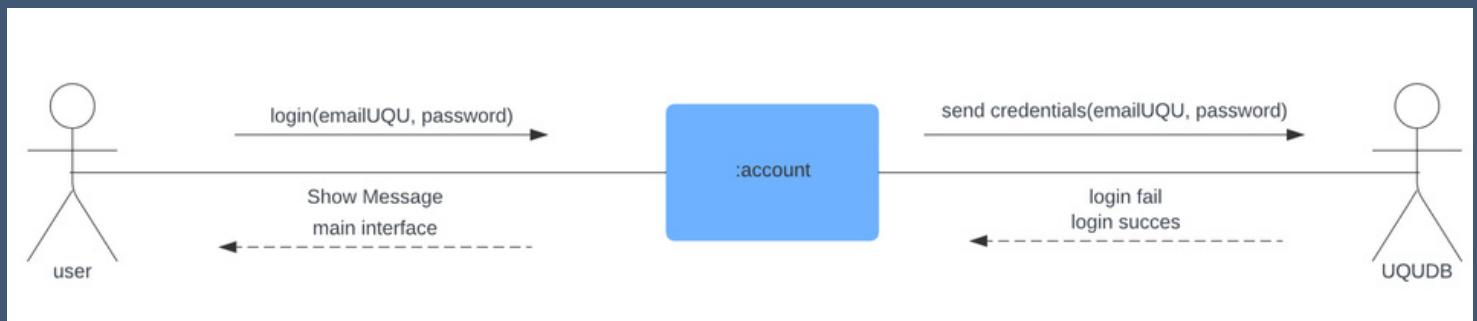


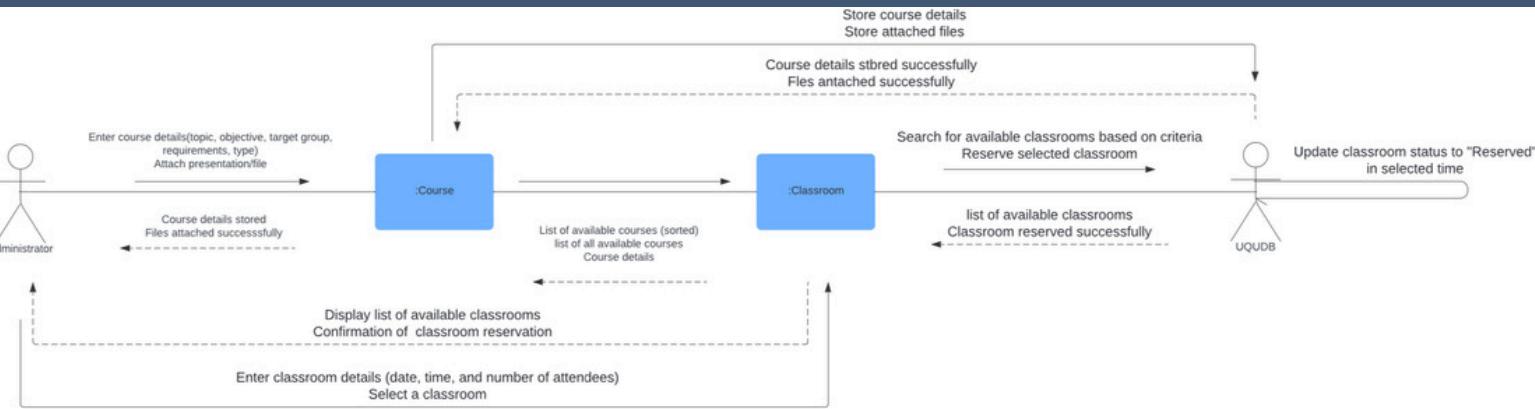
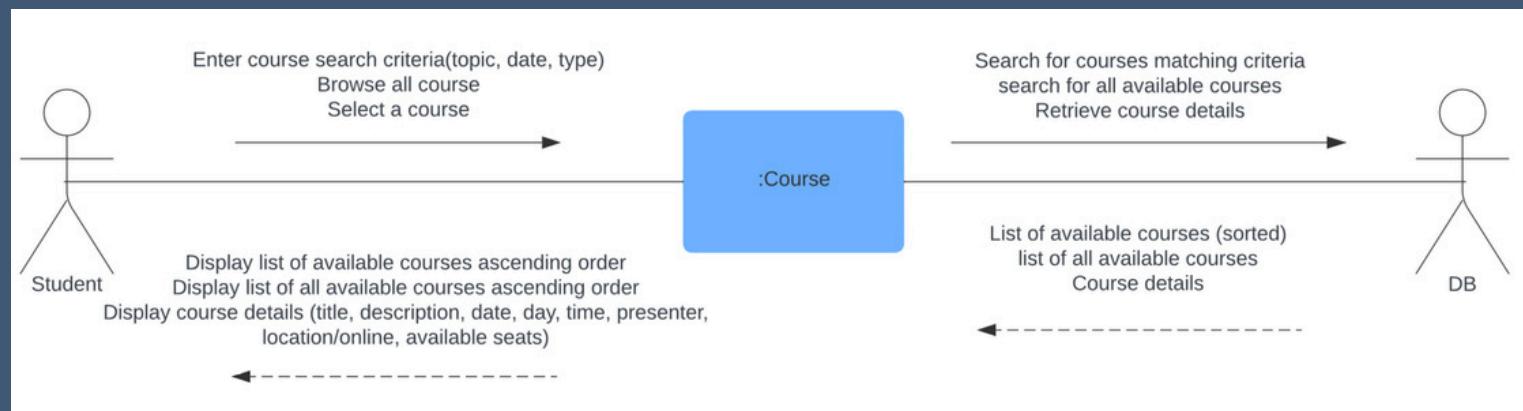
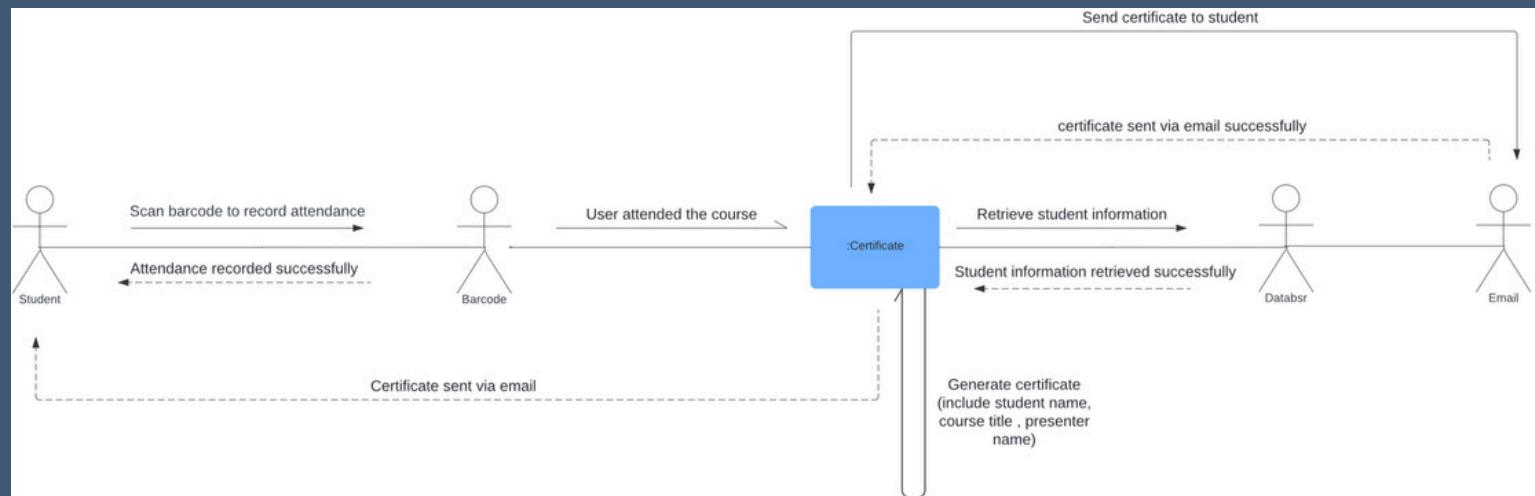
7.STATE DIAGRAM



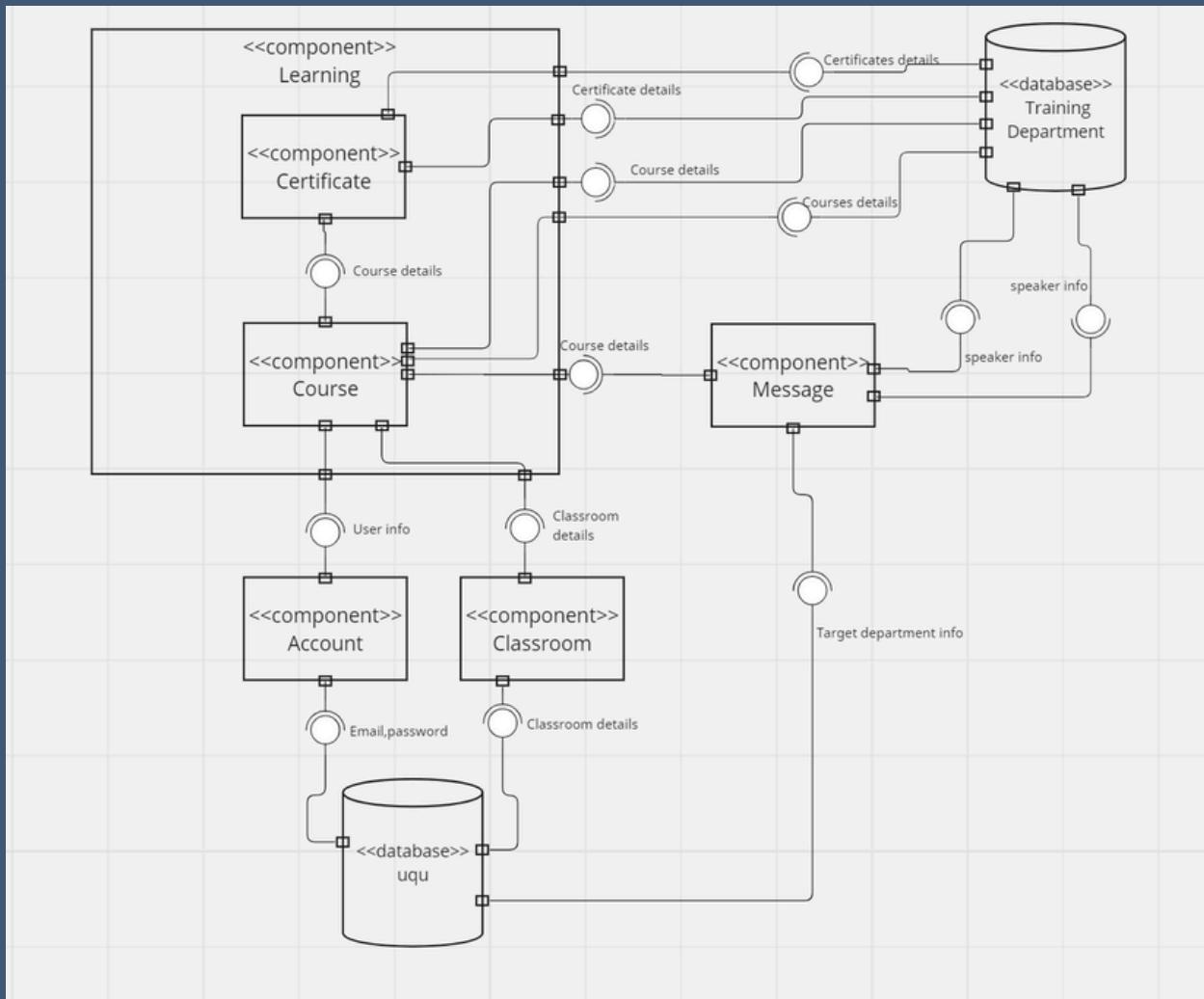


8.COMMUNICATION DIAGRAMS

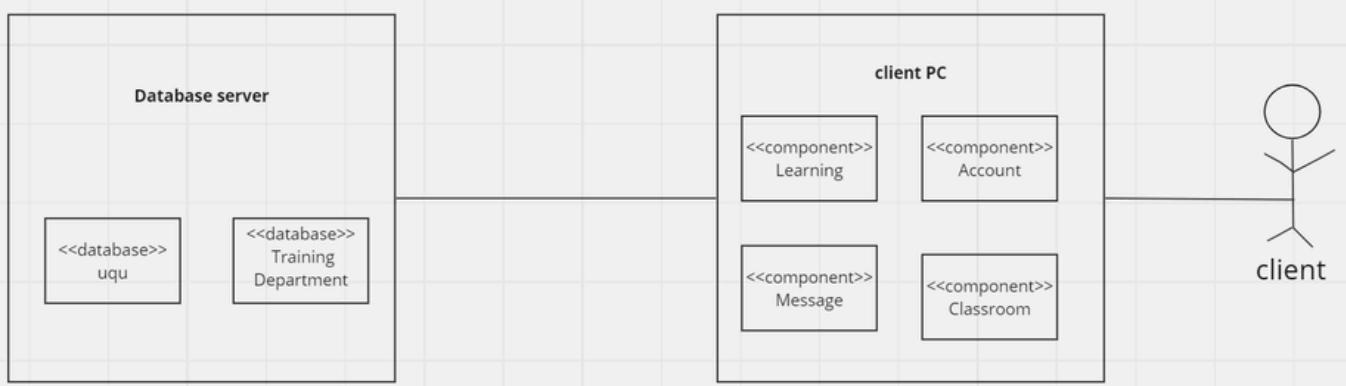




9.COMPONENT DIAGRAM



10.DEVELOPMENT DIAGRAM



TASKS TABLE

	lama bander alsaeidi	Rimas Hassan Alsnehri	Ghala Abed	Lama kudish	ghadi alharbi
business requirements:	✓	✓			
Functional requirements:	✓	✓	✓	✓	✓
on-Functional requirements		✓	✓		
Use case:	✓	✓	✓	✓	✓
Sequence diagram:	✓	✓	✓		✓
Communication diagram:	✓	✓	✓	✓	
Deployment diagram	✓				
Component diagram	✓		✓		
State diagram	✓		✓		✓
class diagram	✓	✓		✓	✓
Active diagram:	✓	✓	✓	✓	✓
Use case text (Scenario)	✓	✓	✓		