

Taallum: Learn Anywhere, Anytime.

UMM AL-QURA UNIVERSITY

Software Systems Implementation Project

Rimas Hassan Alshehri - 44411873

lana bander alsaedi - 444003323

Farah Ibrahim Alotaibi - 443000063

Lana kudish - 44410529

Waad Mohammad Al lehebi - 444001927

project idea

A comprehensive course management application that enables users (students, professionals, or organizations) to browse, register, and manage educational courses. The system provides tools for course creators to schedule sessions, manage attendees, and deliver certificates. It supports online, in-person, or hybrid courses and facilitates global accessibility.

problem statement

Educational institutions, companies, and training providers face challenges in efficiently managing course delivery and ensuring optimal user engagement. Common issues include:

Missed Opportunities: Users often miss enrolling in suitable courses due to lack of timely notifications or reminders.

Inefficient Communication: Limited communication channels hinder interactions between course providers and participants.

Fragmented Platforms: A lack of centralized systems for course discovery and management results in a poor user experience.

Scalability Issues: Existing solutions often fail to cater to diverse user groups, such as students, corporate trainees, and freelance learners.

Administrative Overhead: Course organizers struggle to manage scheduling, venue reservations, and attendee tracking effectively.

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, and course presenter 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.

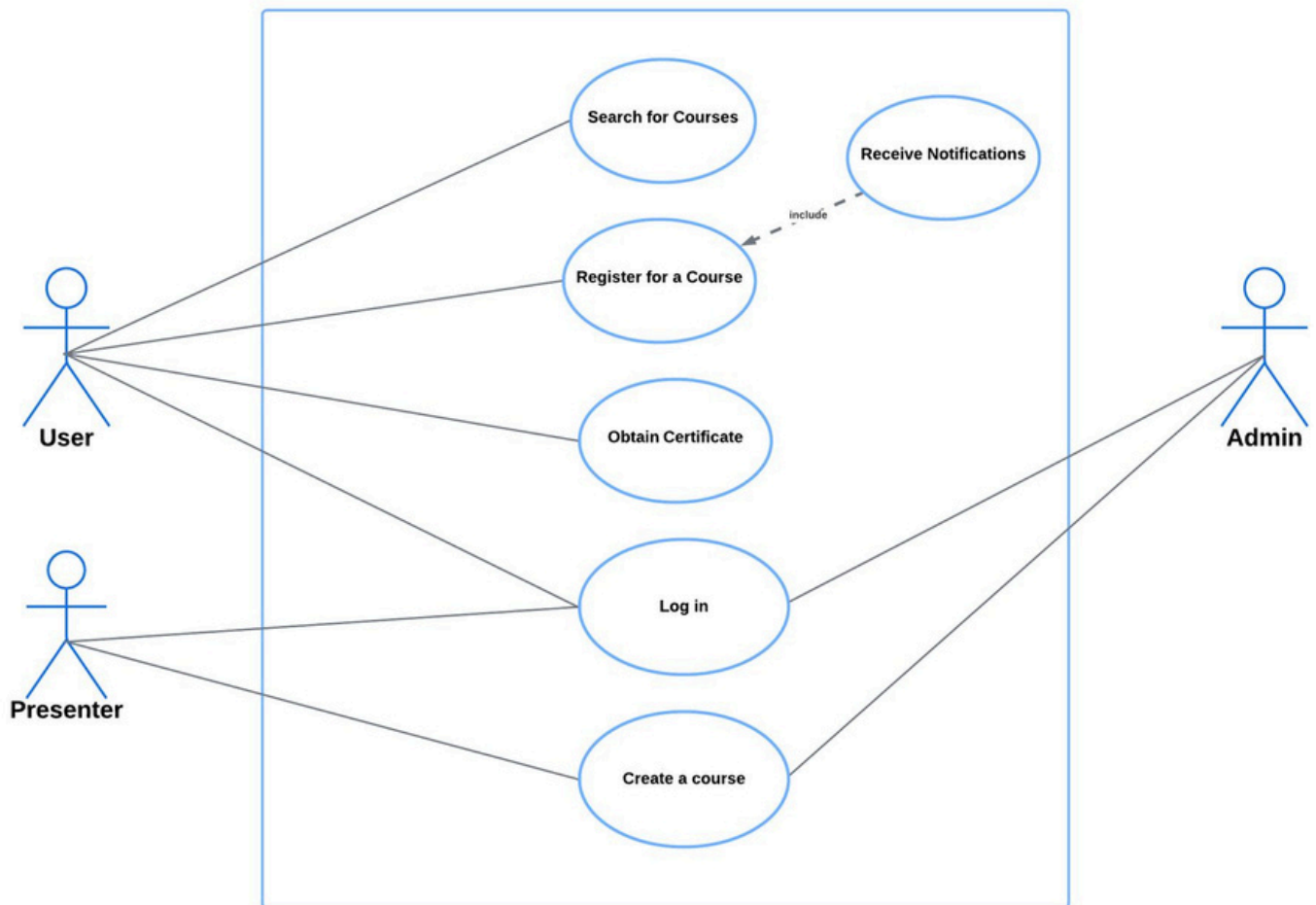
Non-Functional Requirements

Security:

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

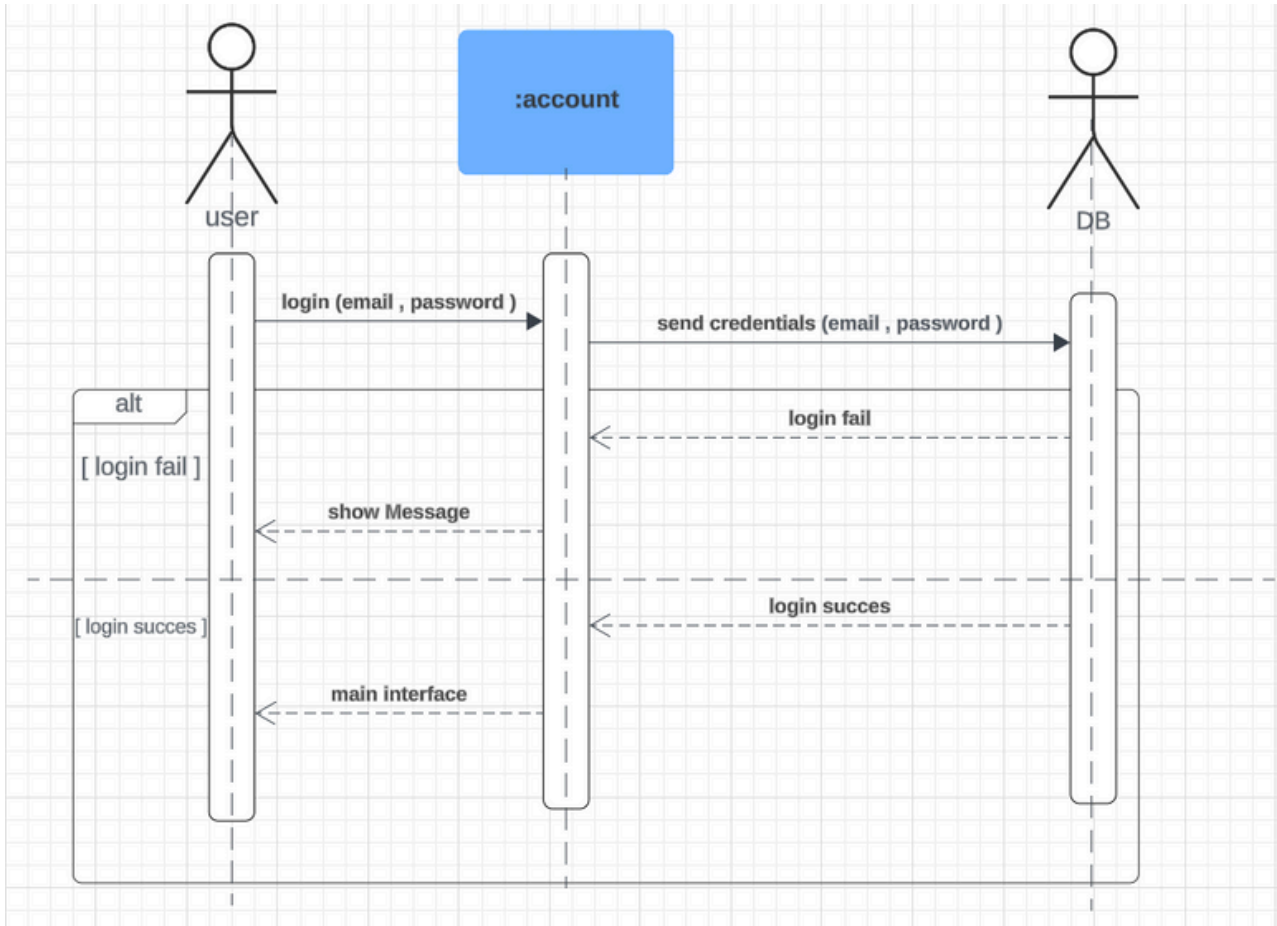
UML diagrams

1-Use Case Diagram

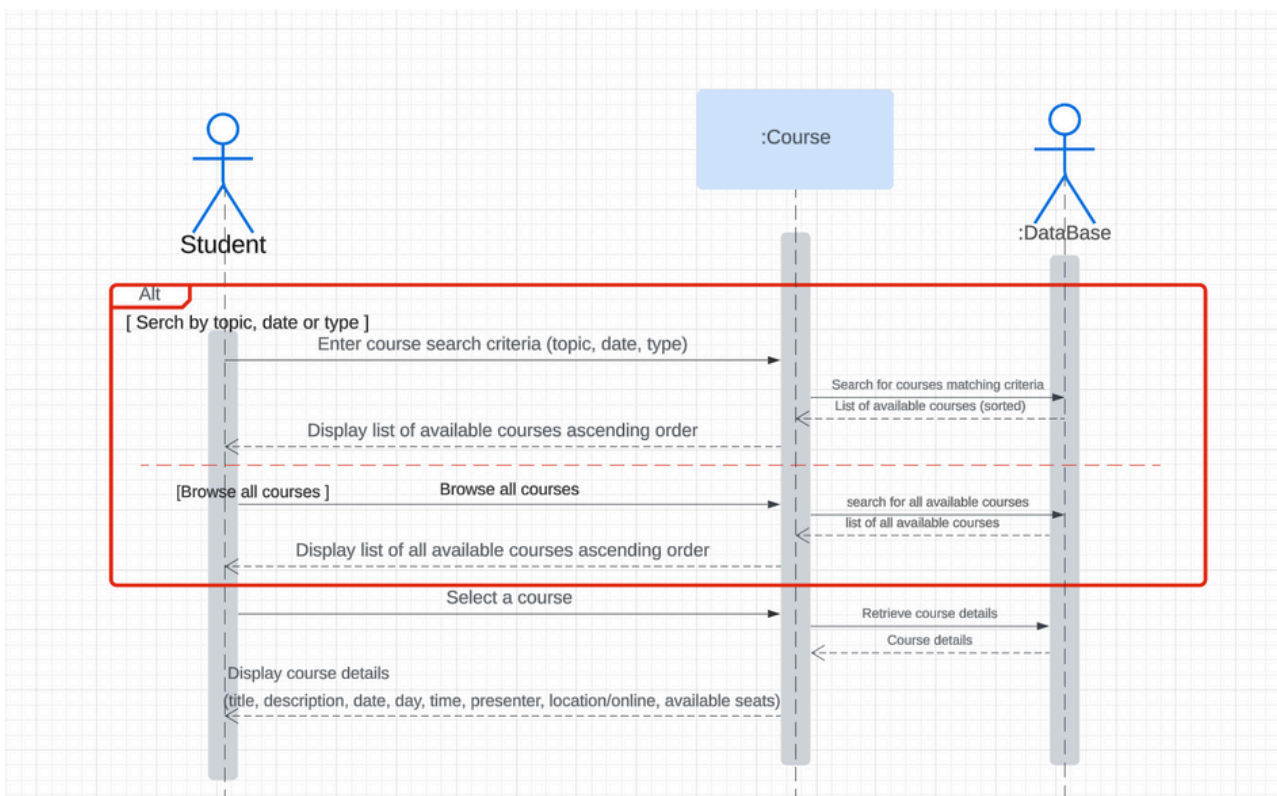


2- Sequence Diagram

Login process

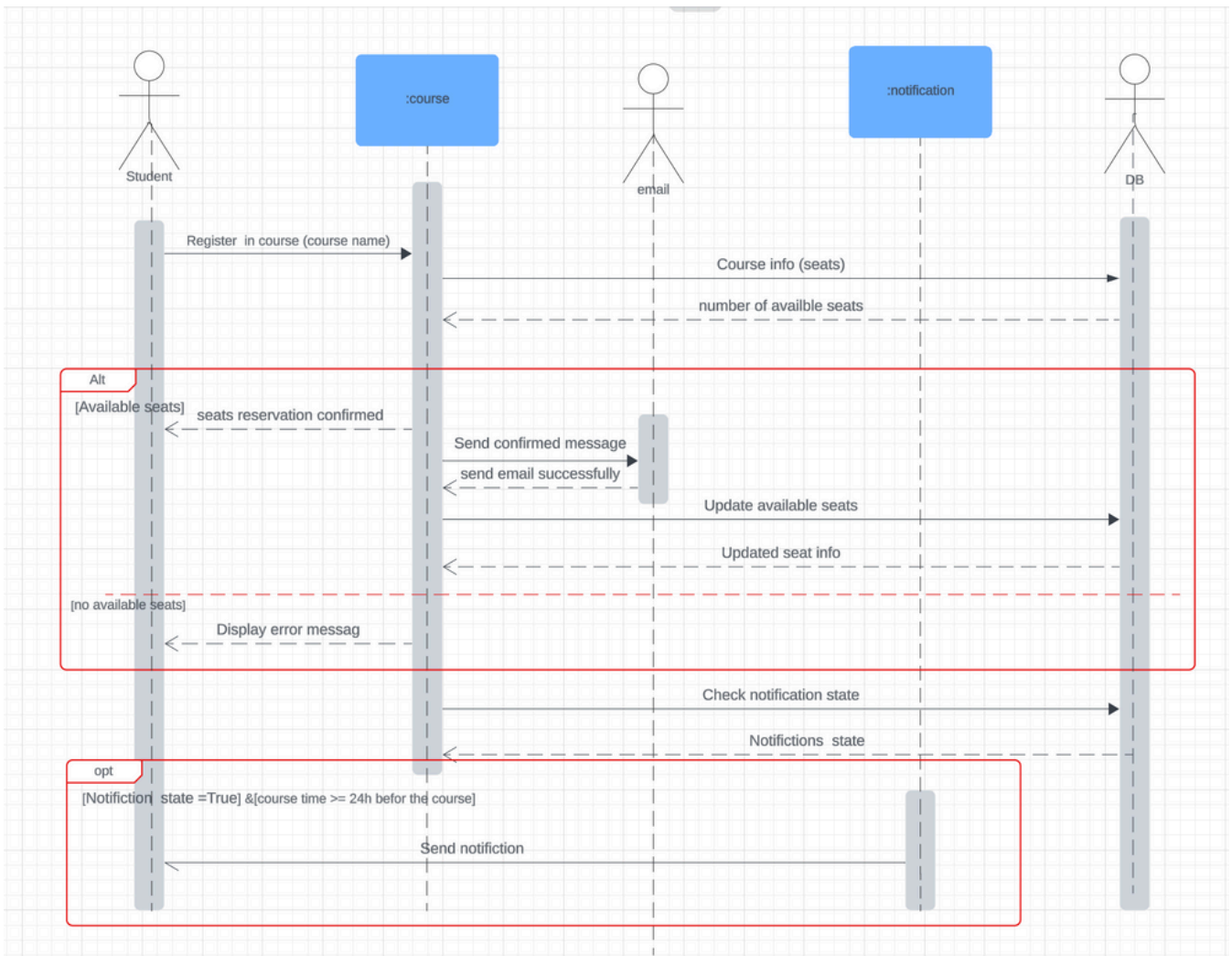


Search for a course

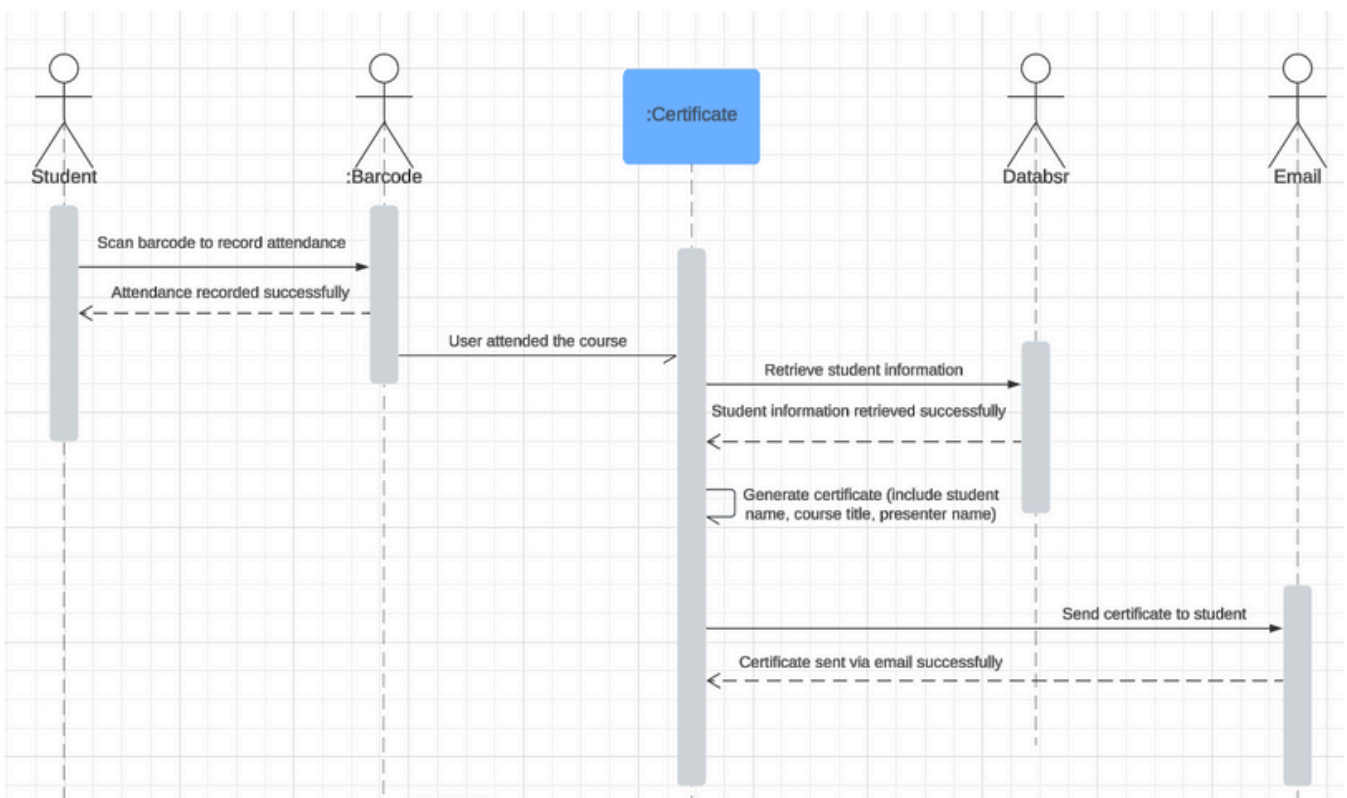


2- Sequence Diagram

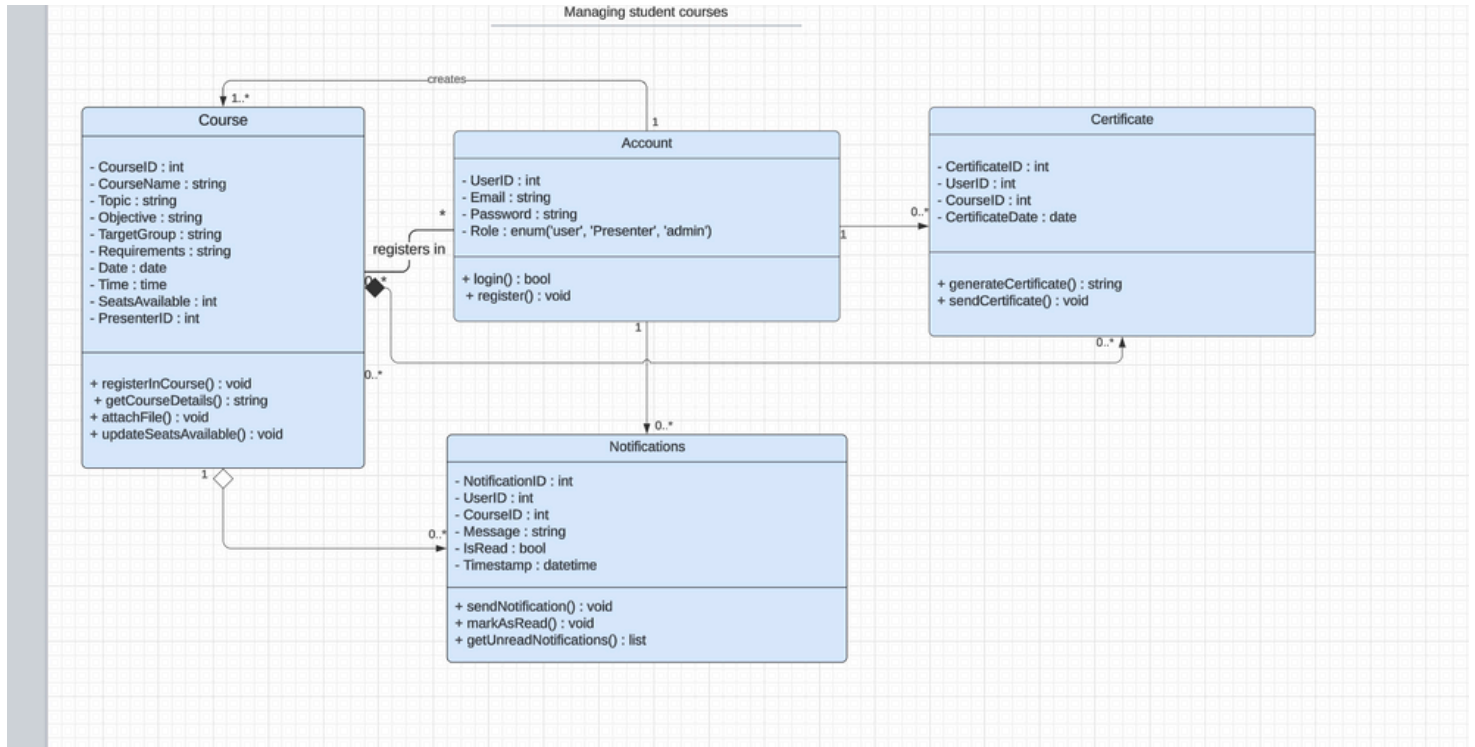
Book a course and activate notifications



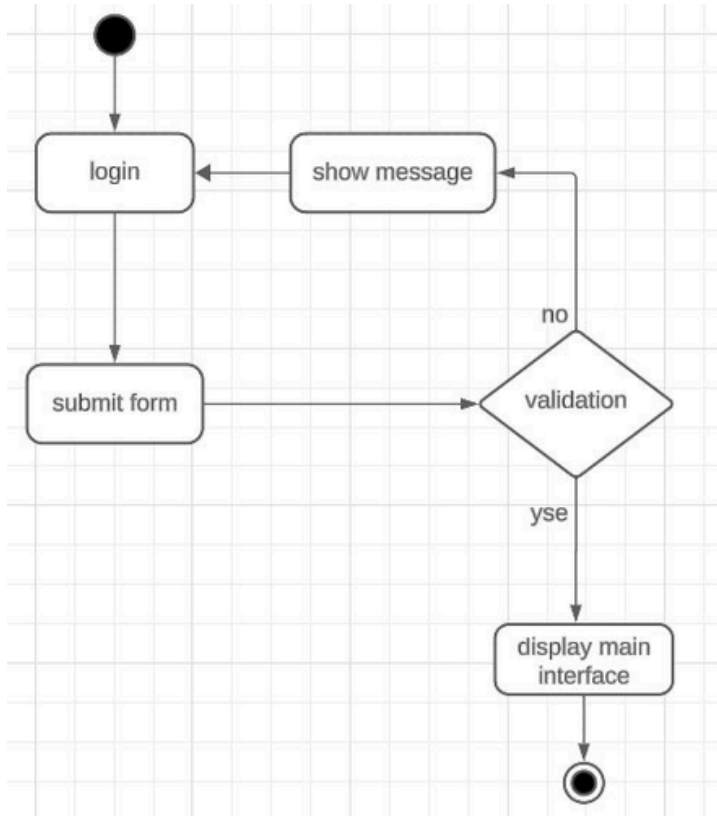
Create and send certificate



3- Class Diagram

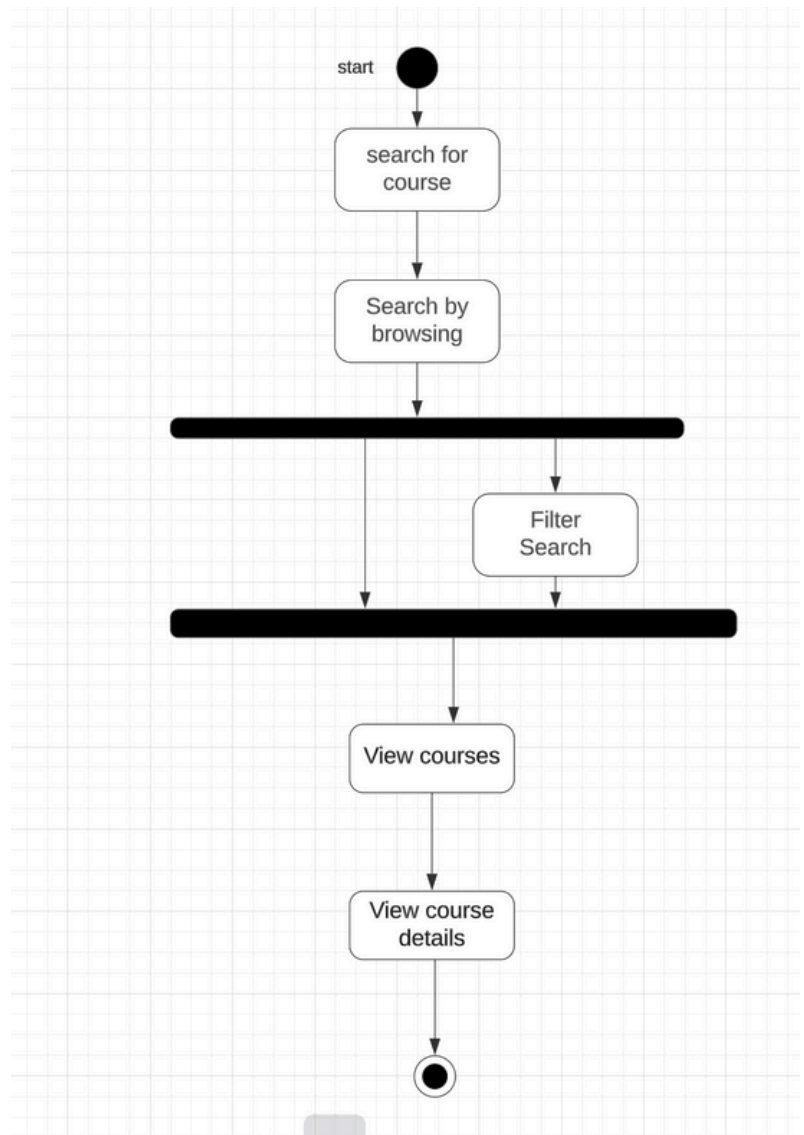


3- Activity Diagram



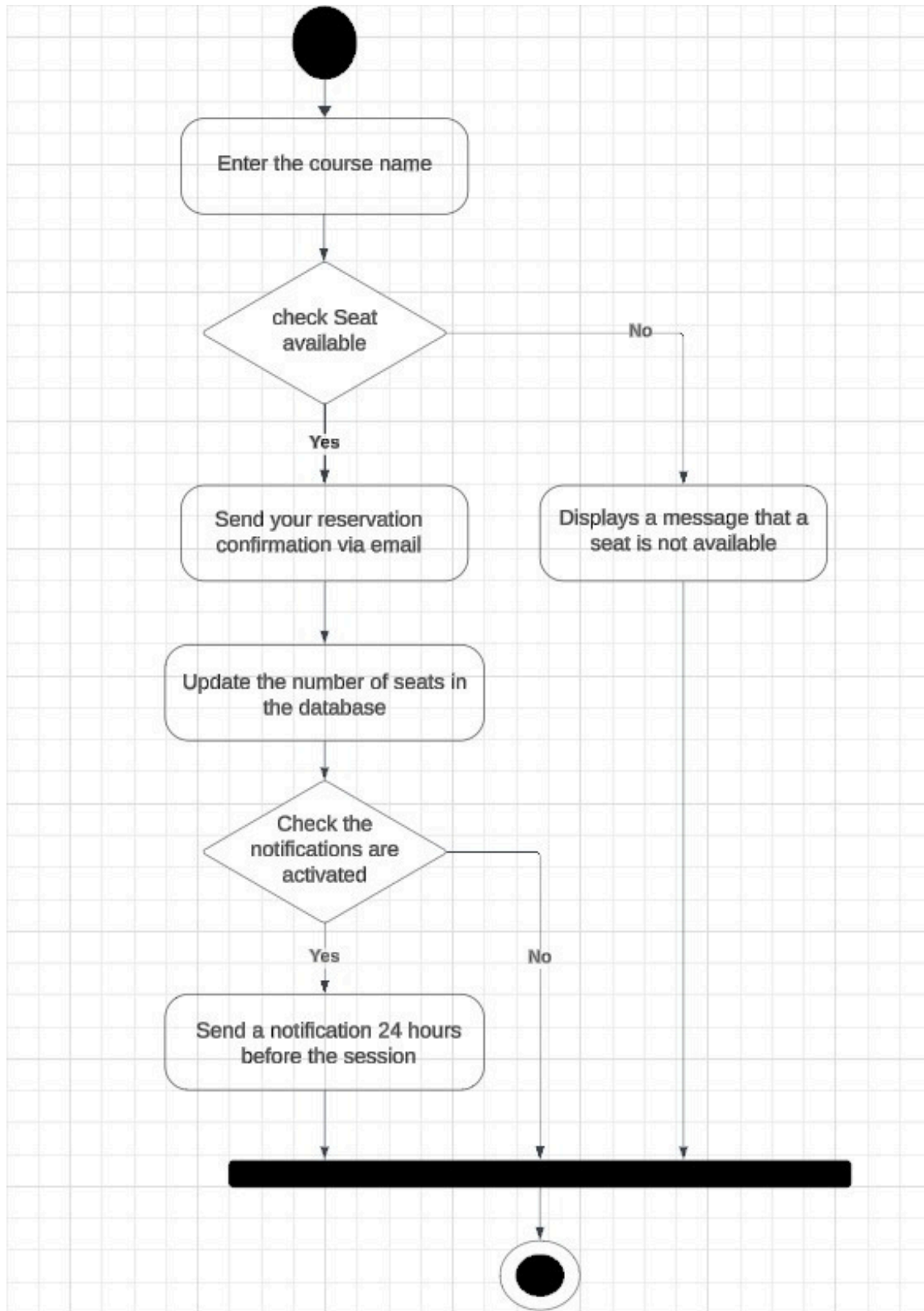
Login

Search for a course



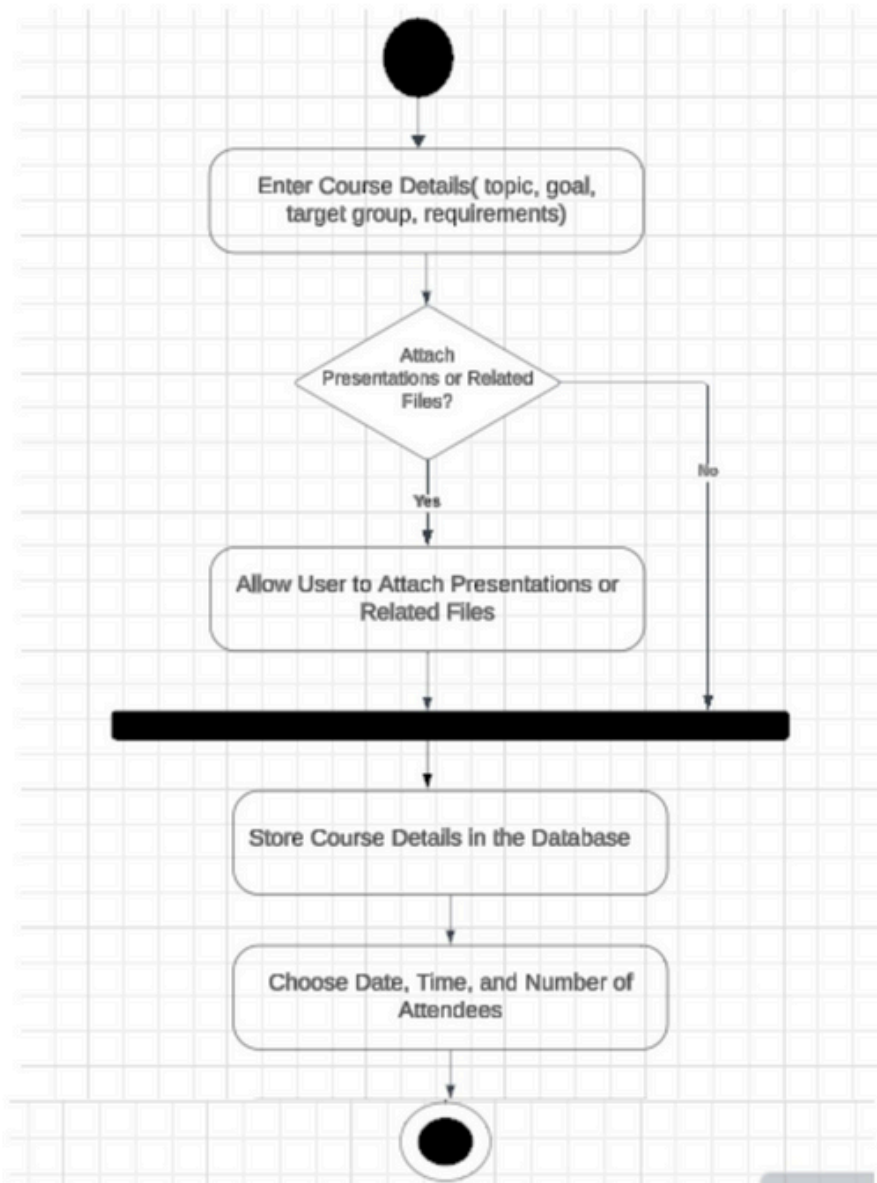
3- Activity Diagram

Book a course

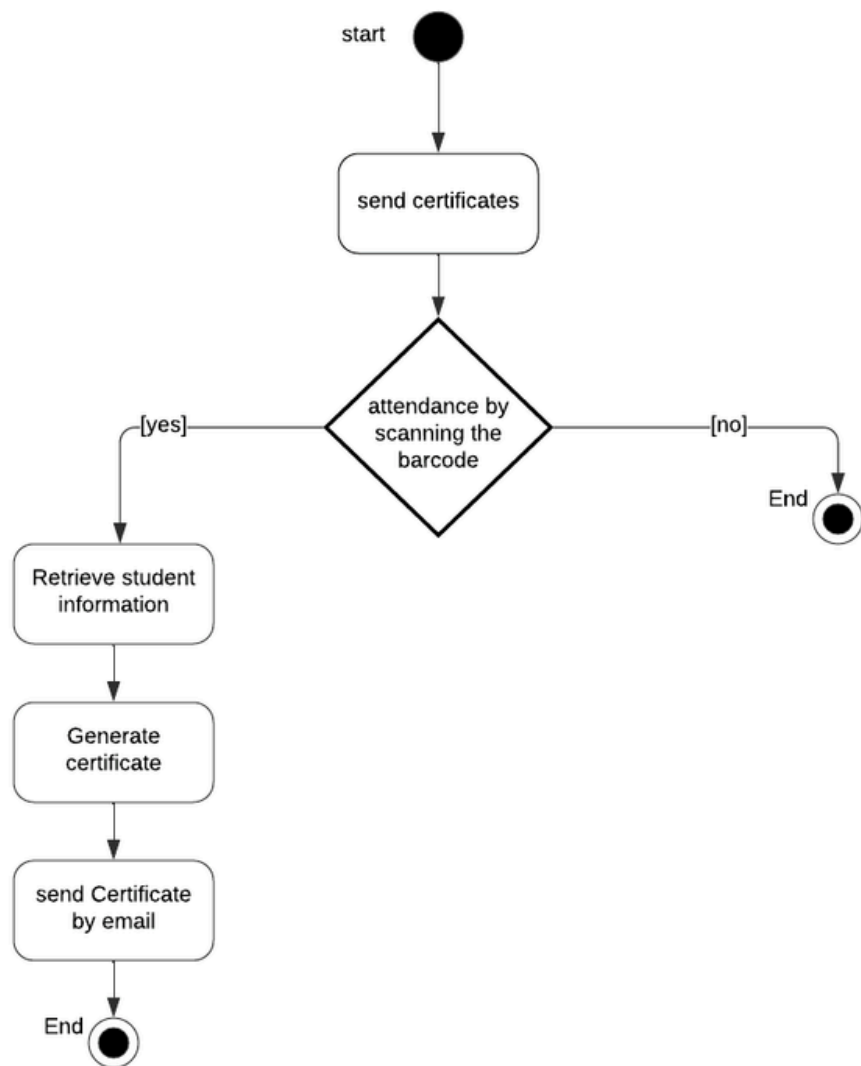


3- Activity Diagram

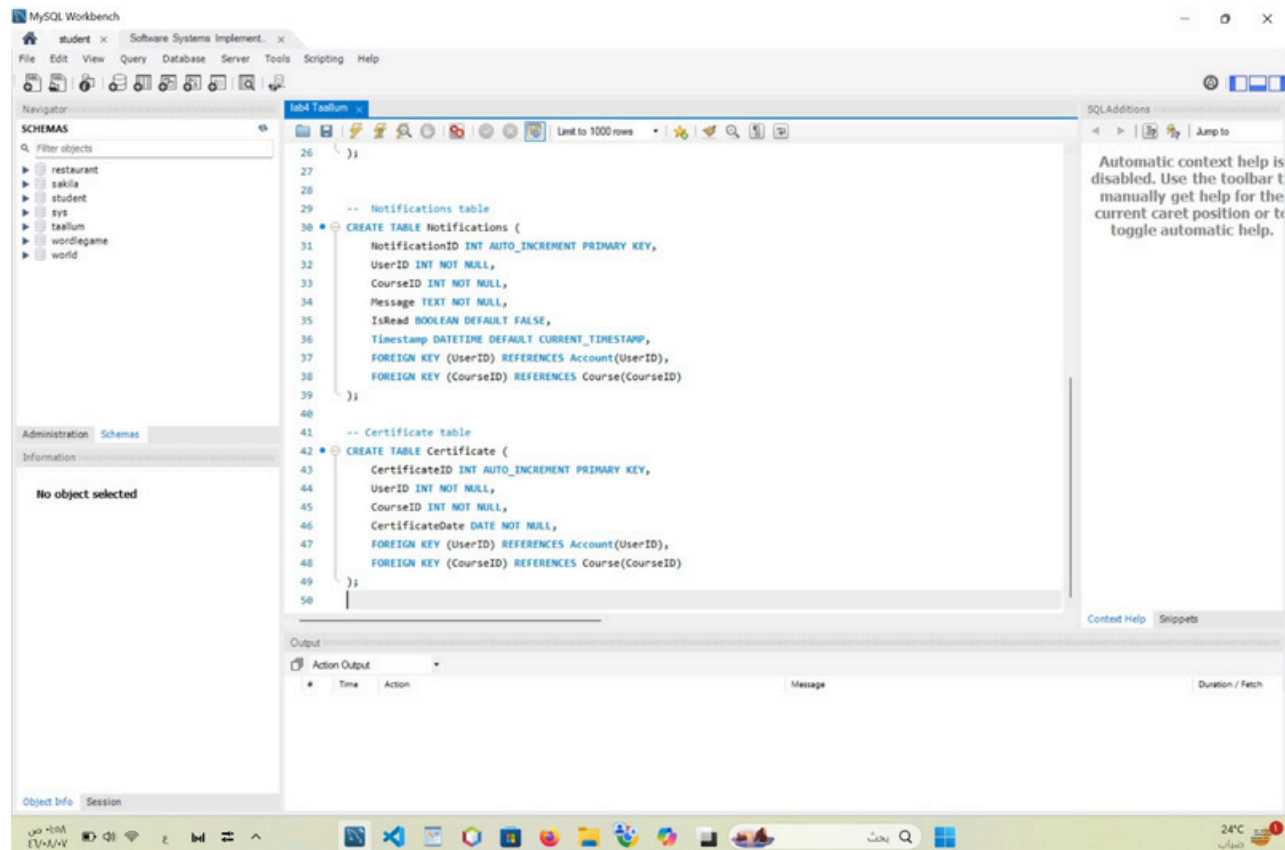
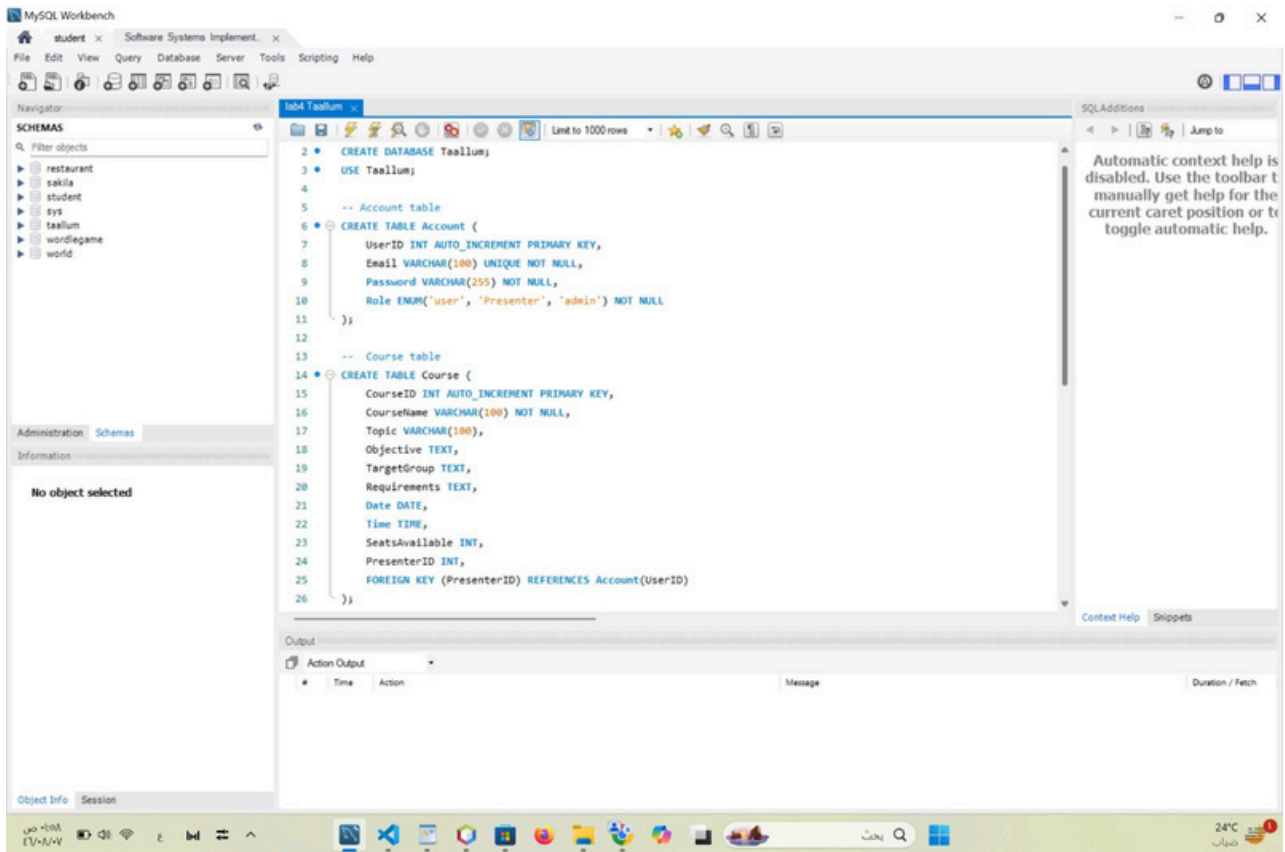
Create a course



Send certificate



4- DataBase



4- DataBase

```
← → project
taallum.sql ×
project > models > taallum.sql
2 CREATE DATABASE Taallum;
3 USE Taallum;
4
5 -- Account table
6 CREATE TABLE Account (
7     UserID INT AUTO_INCREMENT PRIMARY KEY,
8     Email VARCHAR(100) UNIQUE NOT NULL,
9     Password VARCHAR(255) NOT NULL,
10    Role ENUM('user', 'Presenter', 'admin') NOT NULL
11 );
12
13 -- Course table
14 CREATE TABLE Course (
15     CourseID INT AUTO_INCREMENT PRIMARY KEY,
16     CourseName VARCHAR(100) NOT NULL,
17     Topic VARCHAR(100),
18     Objective TEXT,
19     TargetGroup TEXT,
20     Requirements TEXT,
21     Date DATE,
22     Time TIME,
23     SeatsAvailable INT,
24     PresenterID INT,
25     FOREIGN KEY (PresenterID) REFERENCES Account(UserID)
26 );
27
```

```
29 -- Notifications table
30 CREATE TABLE Notifications (
31     NotificationID INT AUTO_INCREMENT PRIMARY KEY,
32     UserID INT NOT NULL,
33     CourseID INT NOT NULL,
34     Message TEXT NOT NULL,
35     IsRead BOOLEAN DEFAULT FALSE,
36     Timestamp DATETIME DEFAULT CURRENT_TIMESTAMP,
37     FOREIGN KEY (UserID) REFERENCES Account(UserID),
38     FOREIGN KEY (CourseID) REFERENCES Course(CourseID)
39 );
40
41 -- Certificate table
42 CREATE TABLE Certificate (
43     CertificateID INT AUTO_INCREMENT PRIMARY KEY,
44     UserID INT NOT NULL,
45     CourseID INT NOT NULL,
46     CertificateDate DATE NOT NULL,
47     FOREIGN KEY (UserID) REFERENCES Account(UserID),
48     FOREIGN KEY (CourseID) REFERENCES Course(CourseID)
49 );
50
```

4- DataBase

The screenshot shows a database management interface with a table named 'Account'. The table has four columns: Role, Password, Email, and UserID. The data is as follows:

Role	Password	Email	UserID
			1
sss	12244	lama	3
student	1234	...lamab	4

The screenshot shows the 'Database Structure' tab of a database management tool. It lists the following tables and their columns:

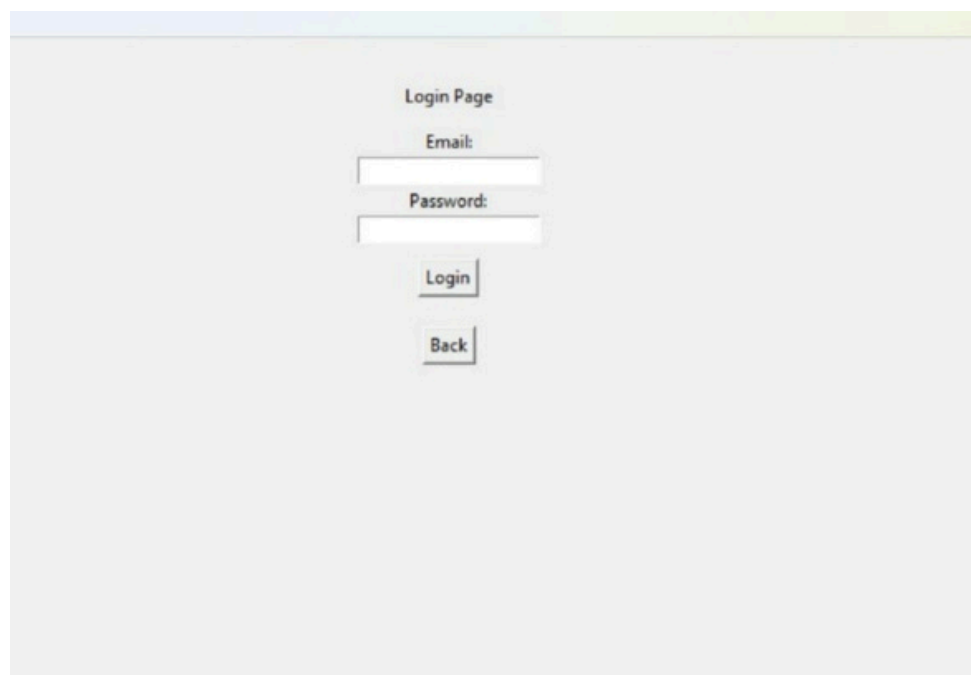
الاسم	النوع	المخطط
Account	INTEGER	INCREMENT, Email TEXT UNIQUE NOT NULL, Password TEXT NOT NULL, Role TEXT NOT NULL
Certificate	INTEGER	UserID* INTEGER
Course	INTEGER	UserID* INTEGER NOT NULL, CourseID* INTEGER NOT NULL, CertificateDate* TEXT NOT NULL
Notification	INTEGER	UserID* INTEGER NOT NULL, CourseID* INTEGER NOT NULL, Message* TEXT NOT NULL, IsRead* BOOLEAN DEFAULT FALSE, Timestamp* TEXT DEFAULT CURRENT_TIMESTAMP
UserCourses	INTEGER	UserID* INTEGER NOT NULL, CourseID* INTEGER NOT NULL
sqlite_sequence	TEXT	CREATE TABLE sqlite_sequence(name,seq)

5- Interfaces

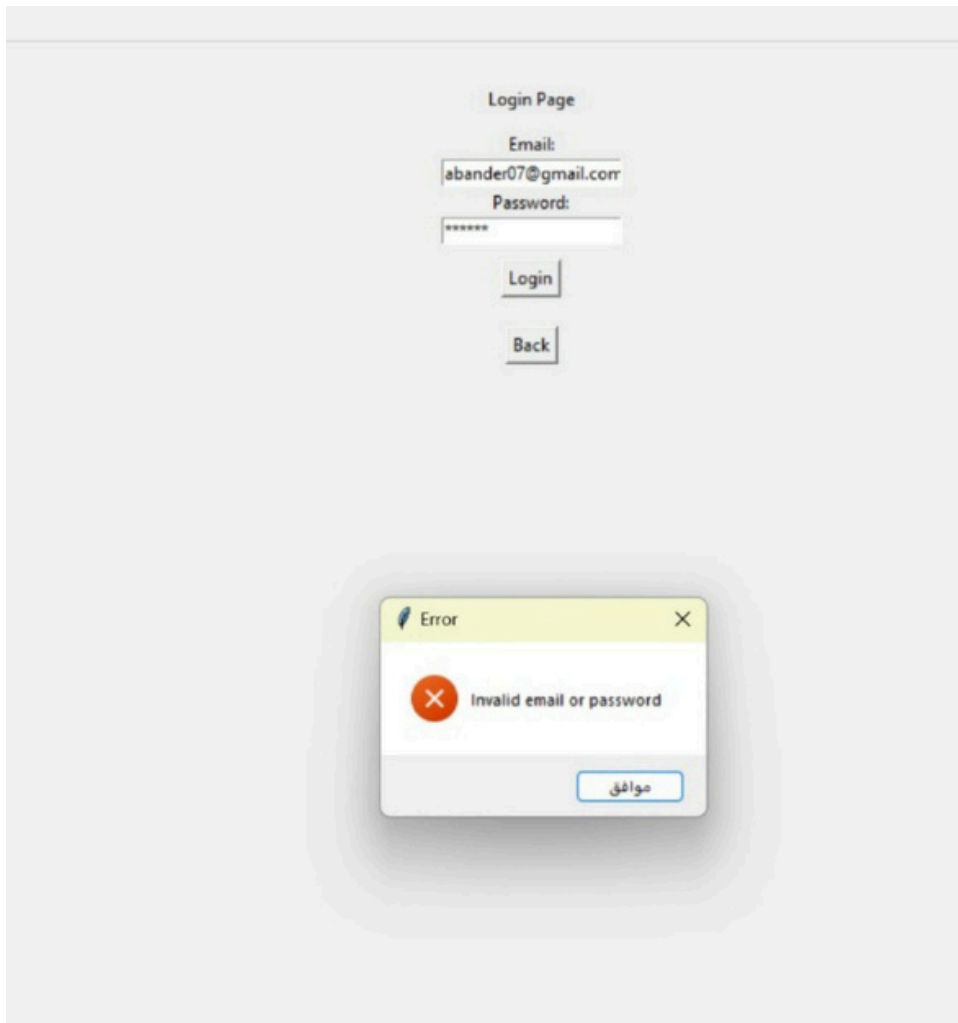
This is the main interface of the Taallum system, containing three options: Login to access the account, Register to create a new account, and Admin Controller to manage the system.



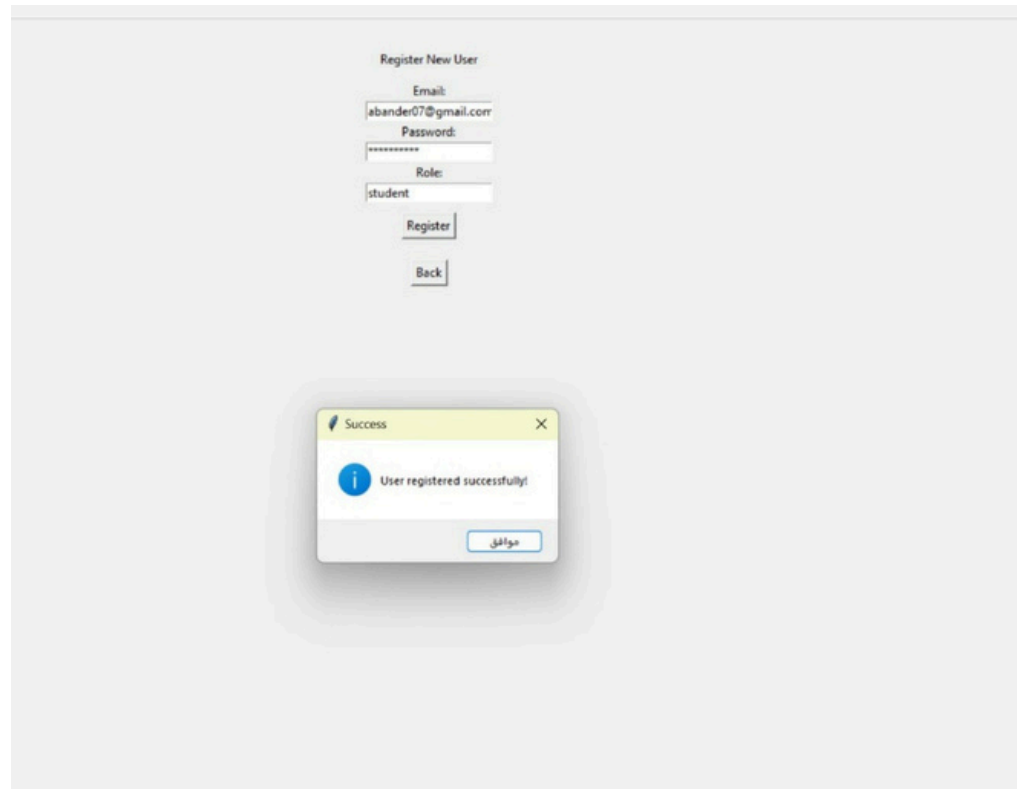
If the login button is pressed, this is the login interface in the Taallum system, where the user is required to enter the email and password to log in. It contains a "Login" button to confirm entry and a "Back" button to return to the previous page.



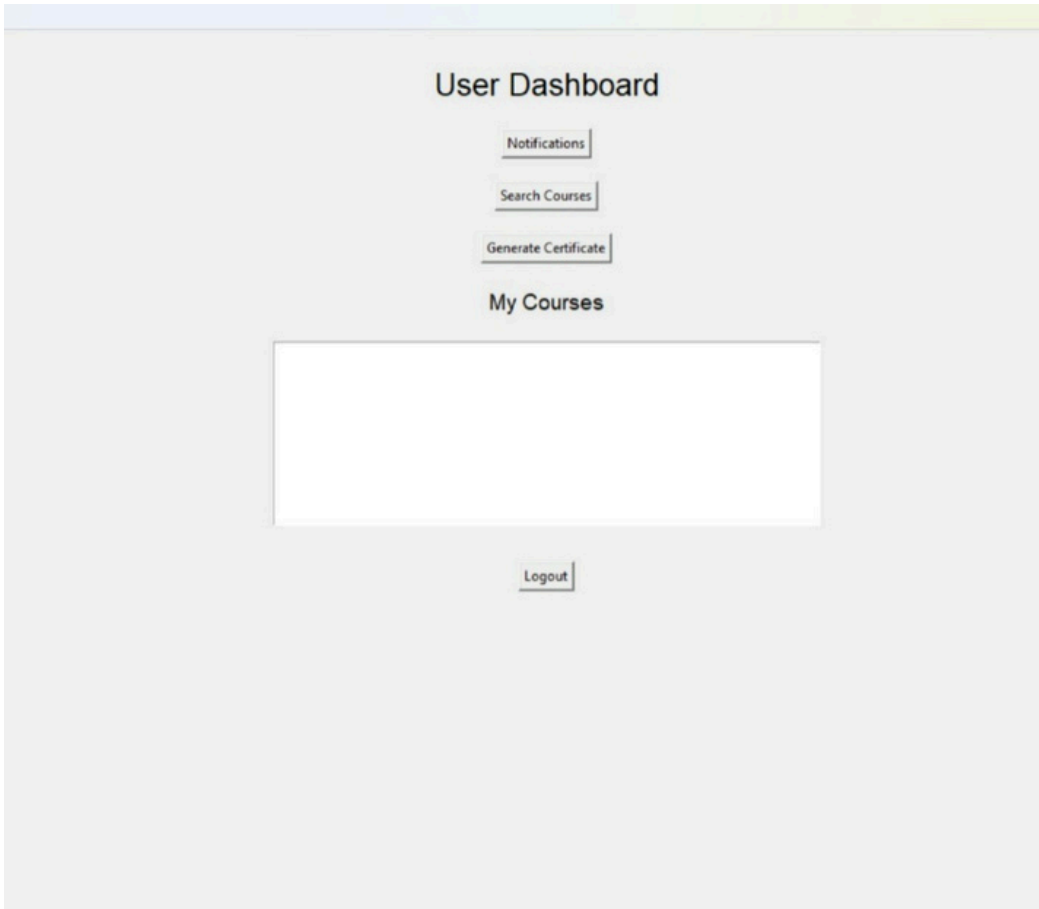
If the entered data is not registered, an error message appears alerting the user that there is no account with this email and password.



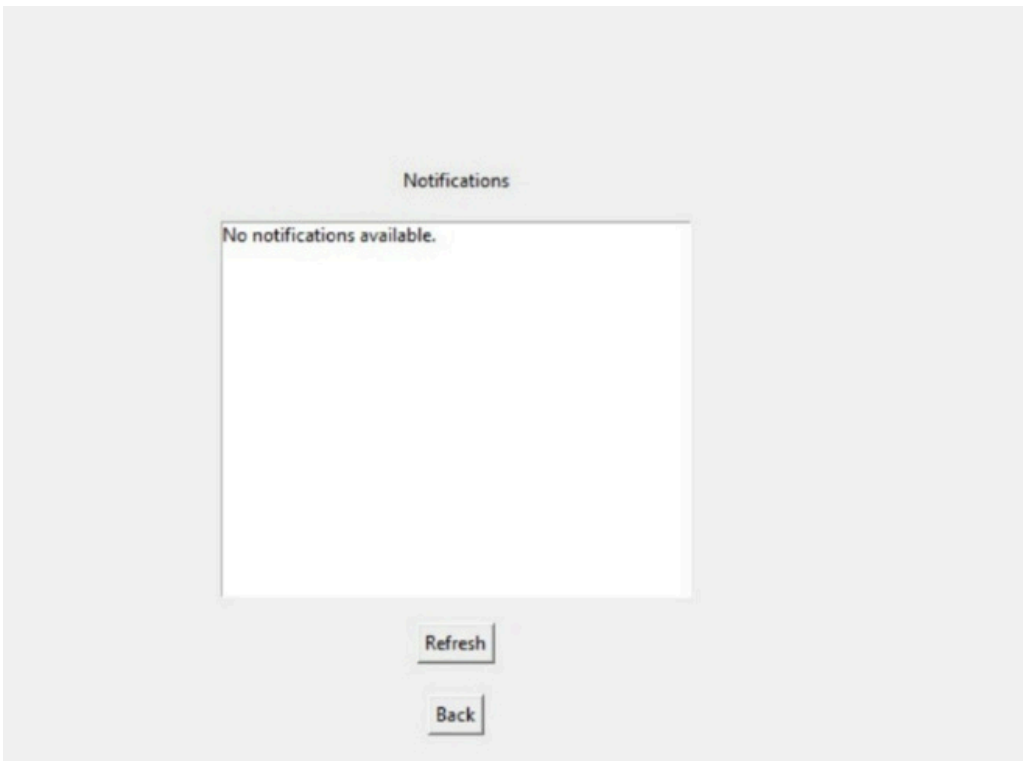
This interface is a new user registration page where the user is asked to enter his email and password and select his role (eg Student). When the user is registered successfully, a confirmation message appears stating that the user has been registered successfully.



This interface is the user's control panel, where he can access several options such as notifications, searching for courses, and generating certificates. Under the “My Courses” section, the courses in which the user has registered can be viewed, and there is also an option to log out of the system.



Interface This is the notifications page, where user notifications are displayed. In this case, a message appears stating that there are no available notifications. The user can refresh the page or return to the previous page via the buttons provided.

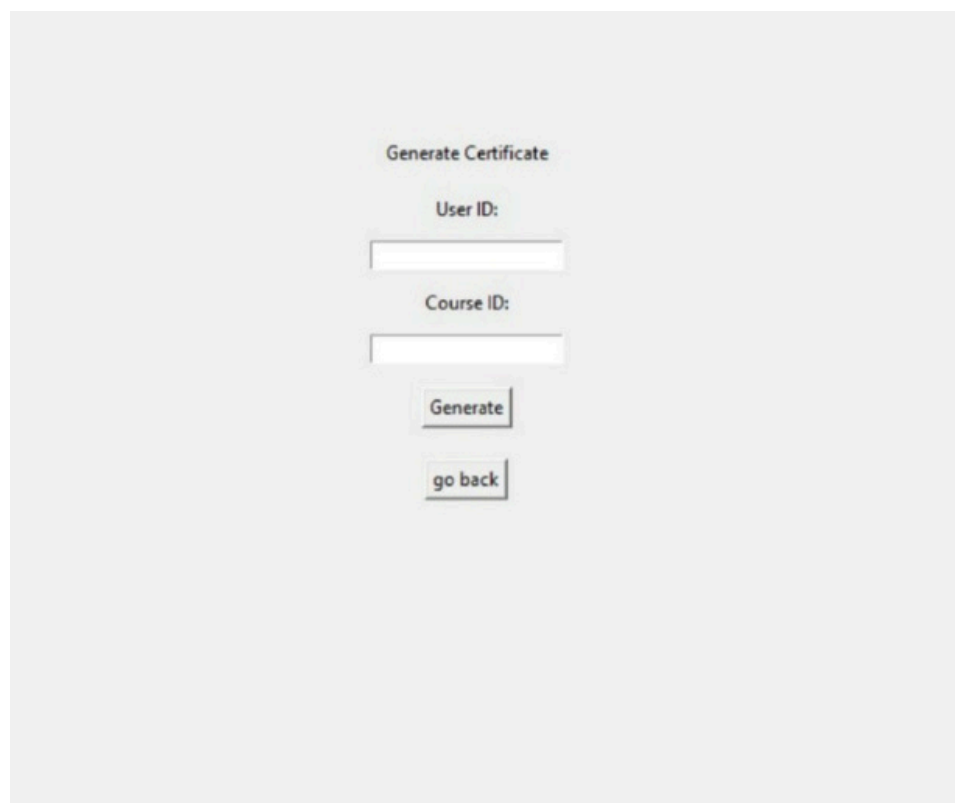


This interface is the courses search page, where the user can enter the search term in the search box. After that, he can view the results and add the chosen course to his list. There is also an option to return to the previous page.



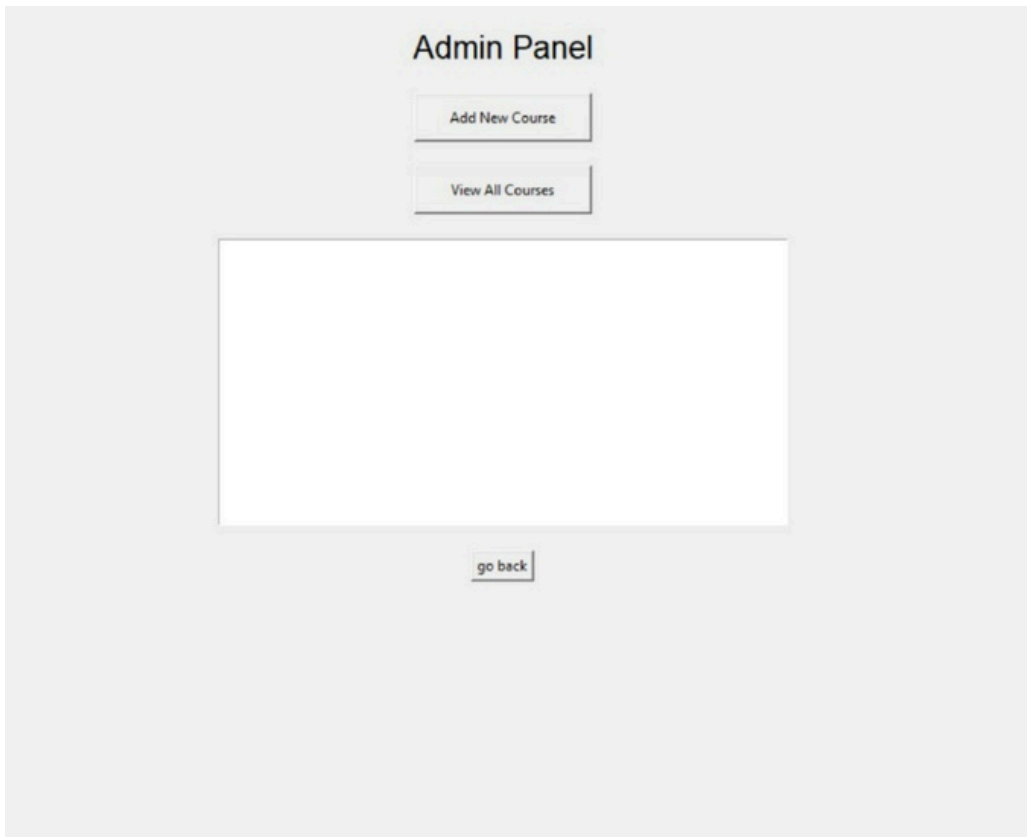
The image shows a web interface for searching courses. It features a title "Search Courses" at the top. Below the title is a text input field for the search term. To the right of the input field is a "Search" button. Below the input field is a large, empty rectangular box, likely intended for displaying search results. At the bottom of the interface are two buttons: "Add Selected Course" and "Back".

This interface is the certificate generation page, where the user is asked to enter the user ID and course ID. After entering the information, he can press the “Generate” button to generate the certificate. There is also an option to return to the previous page.

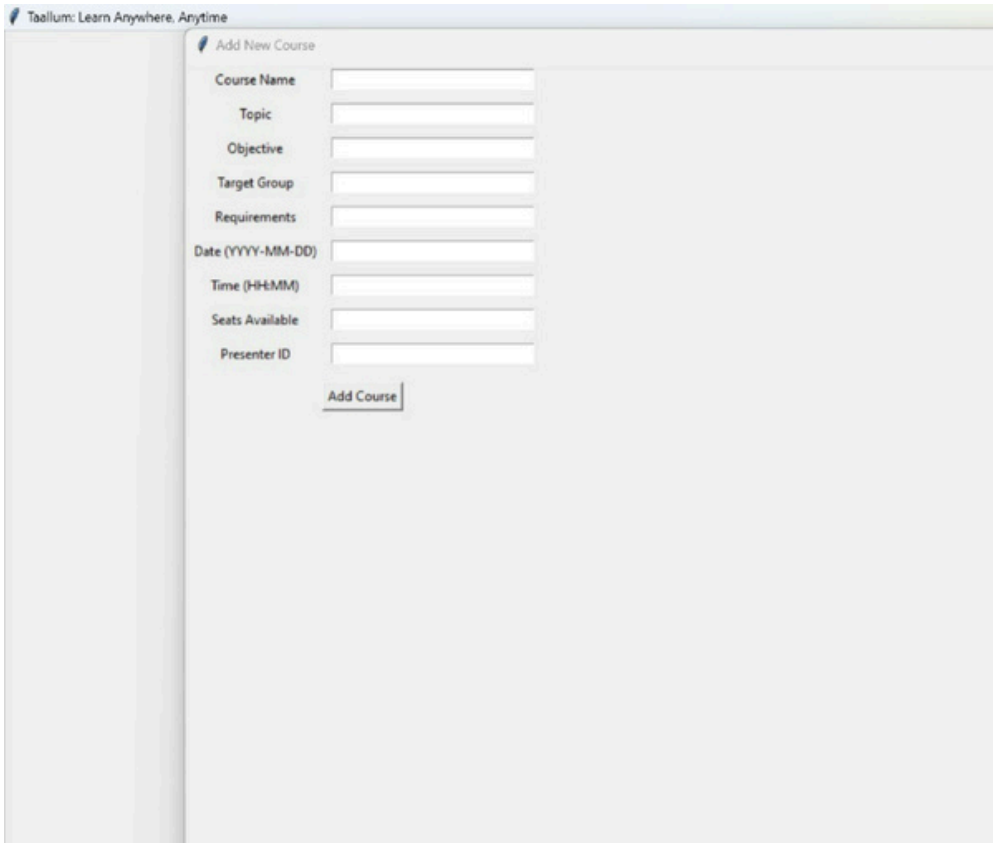


The image shows a web interface for generating a certificate. It features a title "Generate Certificate" at the top. Below the title are two text input fields. The first field is labeled "User ID:" and the second field is labeled "Course ID:". Below the input fields are two buttons: "Generate" and "go back".

This interface is the administrator's control panel, where the administrator can add a new course or view all available courses. There is also an option to return to the previous page.



This interface is the “Add New Course” page, where the administrator is asked to enter course information such as course name, subject, objective, target group, requirements, date, time, and number of seats available. After filling out the information, he or she can click on the “Add Course” button to enter the new course into the system.

A screenshot of a web application titled "Taallum: Learn Anywhere. Anytime". The main heading is "Add New Course". Below this heading is a form with the following fields: "Course Name", "Topic", "Objective", "Target Group", "Requirements", "Date (YYYY-MM-DD)", "Time (HH-MM)", "Seats Available", and "Presenter ID". Each field has a corresponding text input box. At the bottom right of the form, there is an "Add Course" button.