



**KINGDOM OF CAMBODIA
NATION RELIGION KING**



Software Engineer

GROUP: I4-GIC-A

Report Progress Week3

Name of Students	ID of Students	Score
1. HAN Raby	e20220580
2. CHEA Chanminea	e20220335
3. HENG Oulong	e20221390
4. LO Bunleang	e20220722
5. HUN Lyhorng	e20220188

Lecturer: Mr. ROEUN Pacharoth

Academic year 2025-2026

Course Enrollment & Classroom Scheduling System

I. HAN Raby

➤ Student Module (Secondary CRUD)

- Implemented Student dashboard for authenticated student users
- Created StudentController to handle student-specific routes
- Student dashboard is role-protected and accessible only to users with STUDENT role
- Successfully integrated student navigation into the secured system
- Student dashboard rendered correctly via browser routes

➤ Student Profile Module (Secondary CRUD)

- Implemented Student Profile management functionality
- Students can: View personal profile information, Update personal details (First name, Last name, Email, Date of Birth), Upload and update profile image.
- Profile data is retrieved from the authenticated user session
- Implemented StudentProfileDto to: Separate entity from form data, Apply validation rules on user input
- Profile updates are processed via controller and persisted to database
- Uploaded profile images are stored and linked to the user entity
- Profile information is correctly saved to the users table
- Profile CRUD operations work correctly via browser routes

➤ Service Layer Implementation

- Implemented StudentProfileService to handle business logic
- Service responsibilities include: Retrieving currently authenticated user, Updating profile information securely, Handling profile image upload logic
- Controller layer remains thin and delegates logic to service layer
- Follows clean separation of concerns

➤ Student Profile Service & Data Handling

- Implemented StudentProfileService to handle student profile business logic
- Service handles: Retrieving the currently authenticated student, Updating profile information, Handling profile image upload
- Controller layer delegates logic to the service layer
- Maintains clear separation between controller, service, and entity layer

II. LO Bunleang

- Classroom Module (Secondary CRUD)
 - Implemented full CRUD operations for Classroom (Create classroom, View classroom list & details, Update classroom information, Delete classroom)
 - Classroom entity correctly stores
 - Integrated DTO (ClassroomDTO) to Separate entity from form data and apply validate rules
 - Implemented service layer for business logic
 - Implemented repository layer using Spring Data JPA
 - Data successfully saved to classrooms table
 - CRUD works via browser routes
- Schedule Module (Secondary CRUD)
 - Implemented full CRUD operations for Schedule (Create schedule, View schedules, Update schedule, Delete schedule)
 - Schedule correctly links to Course, Classroom, Academic year, Semester, Day of week, Start & end time, Status.
 - Used ScheduleDTO to: Accept form input, Map IDs (courseId, classroomId) to entities
 - Implemented service logic with: Course lookup, Classroom lookup, Validation before saving.
- Role-Based Access Control (STRICT)
 - Integrated with Spring Security
 - Admin capabilities (Full CRUD for Classroom, Full CRUD for Schedule)
 - Student capabilities (View classroom list & details, View schedules, No create/update/delete permissions)
- Quick Test (Temporary)

The image displays three browser screenshots and a terminal output, illustrating the Classroom application's functionality.

Classroom Form (localhost:8080/admin/classrooms/1/edit): Shows a form for editing a classroom. Fields include Building (J), Room Number (J602), Capacity (30), Projector (checked), Computer (checked), and Status (AVAILABLE). A 'Save' button and a 'Back' link are visible.

Classrooms (Admin) (localhost:8080/admin/classrooms): Shows a table of classrooms with columns Room, Building, Capacity, and Actions. A 'New Classroom' link is present. The table contains one row: J602, J, 30, with Edit and Delete links.

Available Classrooms (localhost:8080/student/classrooms): Shows a table of available classrooms with columns Room, Building, and Capacity. The table contains one row: J602, J, 30.

Terminal Output: Shows the SQL query and result for the classrooms table.

```
mysql> select * from classrooms;
+-----+-----+-----+-----+-----+-----+
| classroom_id | building | capacity | has_computer | has_projector | room_number | status |
+-----+-----+-----+-----+-----+-----+
| 1 | J | 30 | 0x01 | 0x01 | J602 | AVAILABLE |
+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

Since there is no data in the courses table yet, I insert data to it directly. If I don't, the schedule can't be create

Course	Classroom	Day	Time
Intro to CS	J602	MONDAY	08:55 - 00:55

ID	Course	Classroom	Day	Time	Actions
1	Intro to CS	J602	MONDAY	08:55 - 00:55	Edit Delete

```
mysql> INSERT INTO courses (
-> course_code,
-> course_name,
-> credits,
-> max_capacity,
-> lecturer_id,
-> status
-> )
-> VALUES (
-> 'CS101',
-> 'Intro to CS',
-> 3,
-> 40,
-> 1,
-> 'ACTIVE'
-> );
Query OK, 1 row affected (0.01 sec)

mysql> select * from courses;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| course_id | course_code | course_name | created_at | credits | description | max_capacity | status | updated_at | lecturer_id | course_image |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | CS101 | Intro to CS | NULL | 3 | NULL | 40 | ACTIVE | NULL | 1 | NULL |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

III. HUN Lyhorng

- Student Interface & Course UI Implementation (Weekly Progress)
 - Designed and implemented Student Home Page UI based on Figma design
 - Built UI using Thymeleaf and Tailwind CSS
 - Fixed controller logic to correctly redirect users after login
 - Implemented static course cards for UI testing purposes
 -

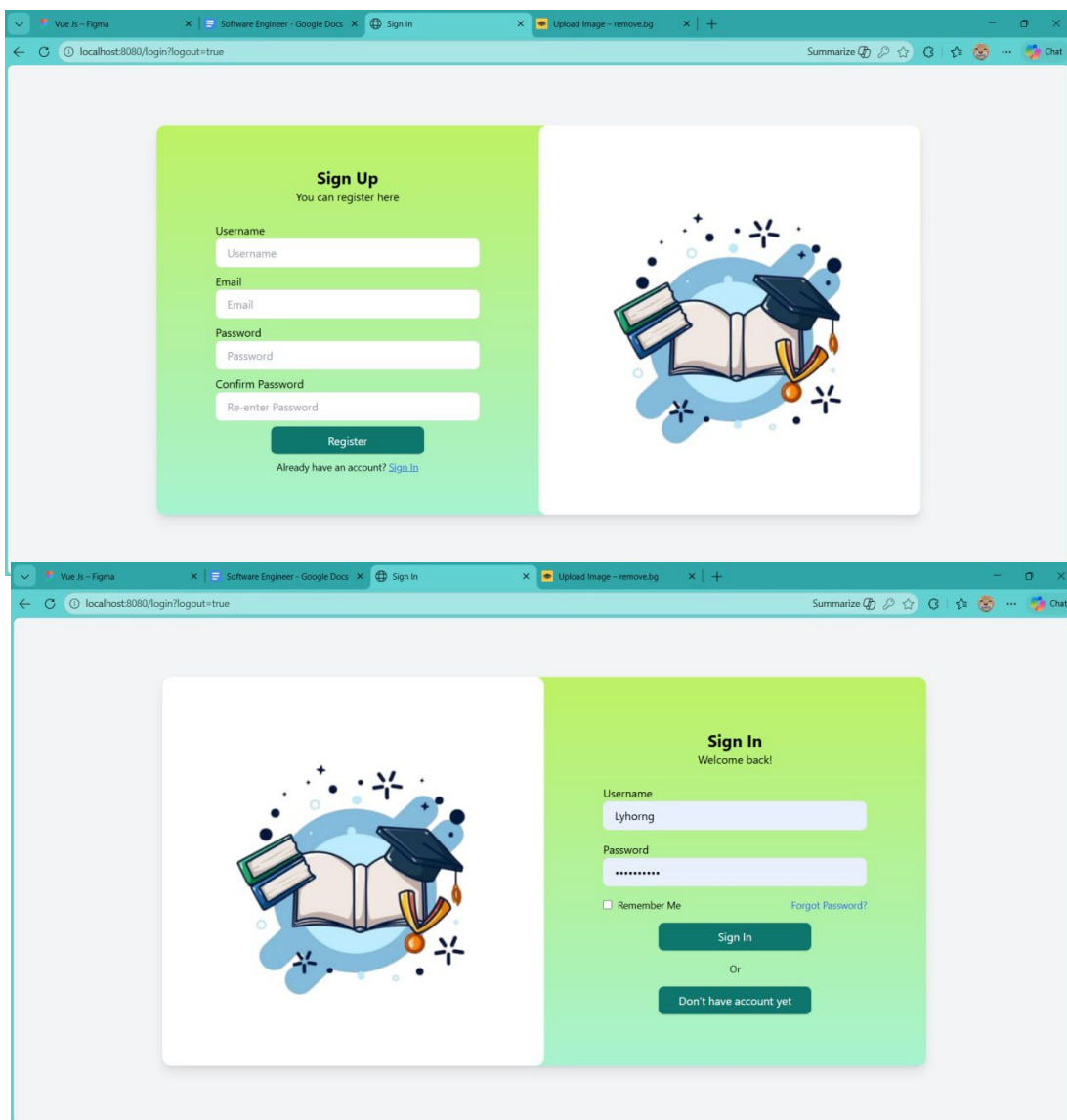
- Used static images and sample data to validate layout and responsiveness
- Created Course Details page UI
- Implemented navigation flow when user clicks on a course card
- Course details page is partially working and still requires bug fixing
- Implemented reusable Header and Footer layouts
- Integrated layouts across student-related pages for consistent UI
- Figma's Link:

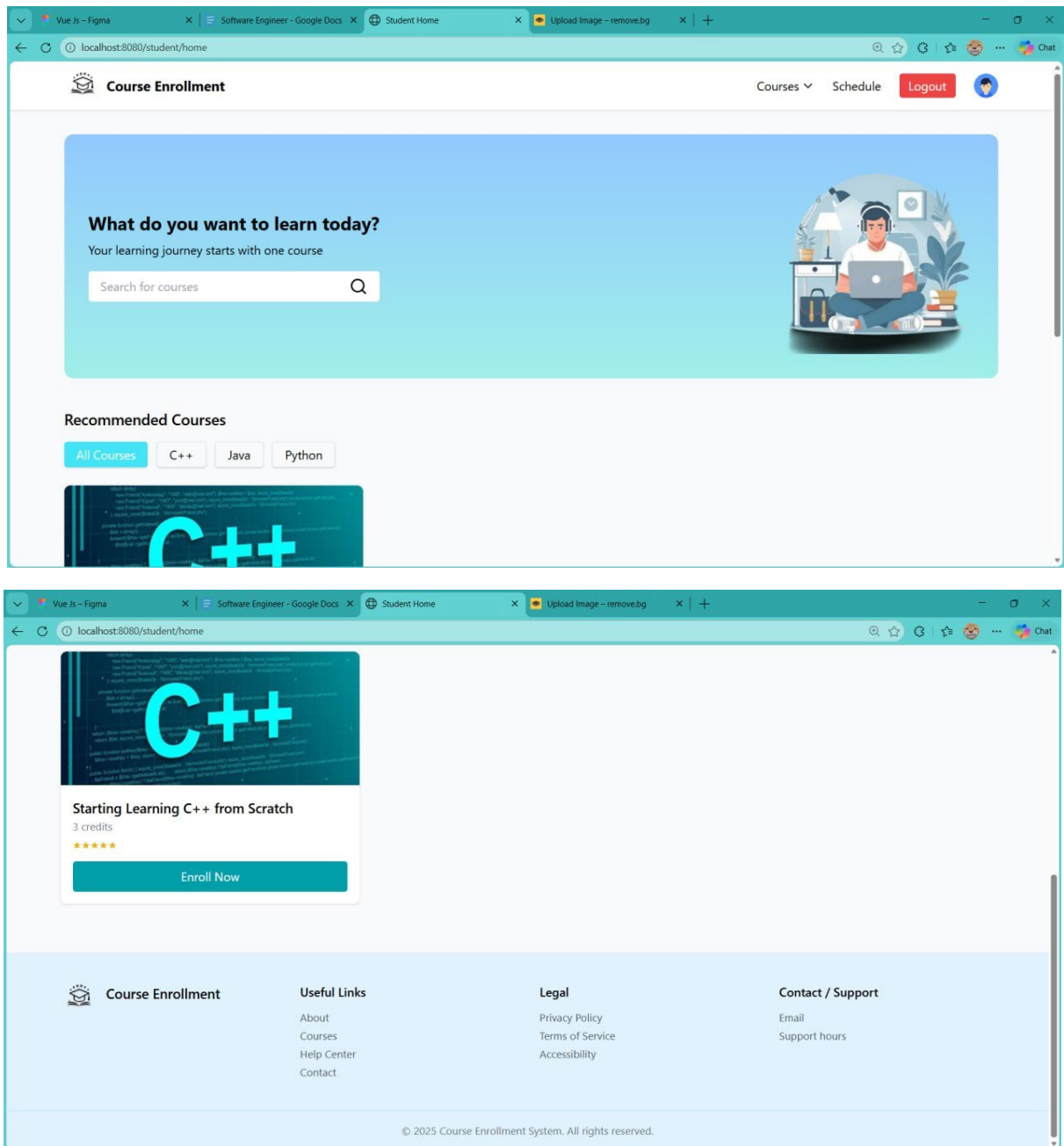
<https://www.figma.com/design/y2mzEZUAfgVZlEZw4WlzoC/Vue-Js?node-id=755-37&p=f&t=xdbTRA20zEcAN7nV-0>

➤ Notes / Current Status

- Course data is currently static (testing phase)
- Course details page still has routing/controller errors and will be fixed next

➤ Quick Test





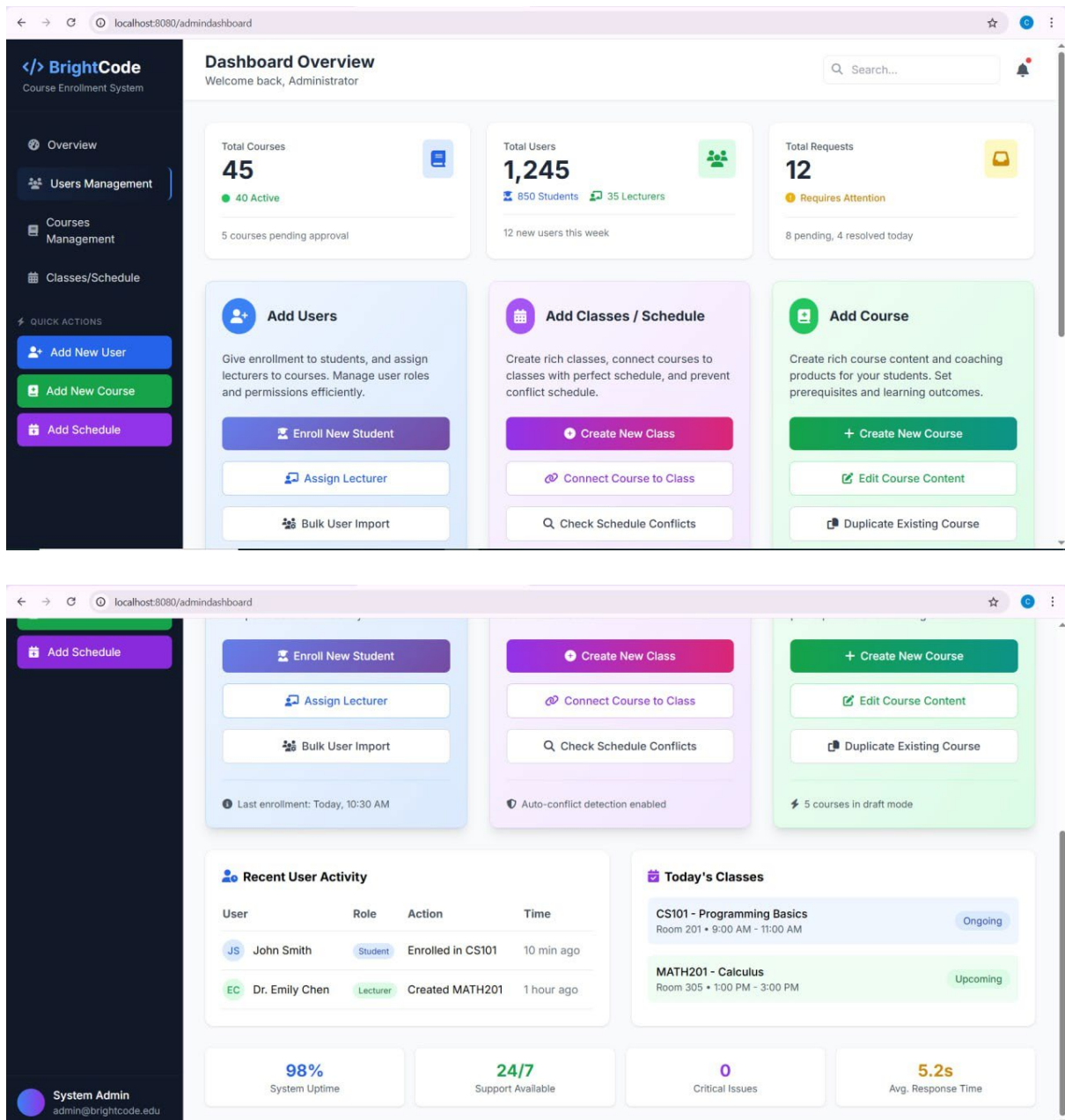
IV. CHEA Chanminea

Admin Dashboard Module (UI Implementation)

➤ Statistics Panel

- Displayed system overview statistics: Total Courses: 45 (40 Active), Total Users: 1,245, Total Requests: 12
- Used statistic cards with icons and visual status indicators
- Designed for quick system monitoring

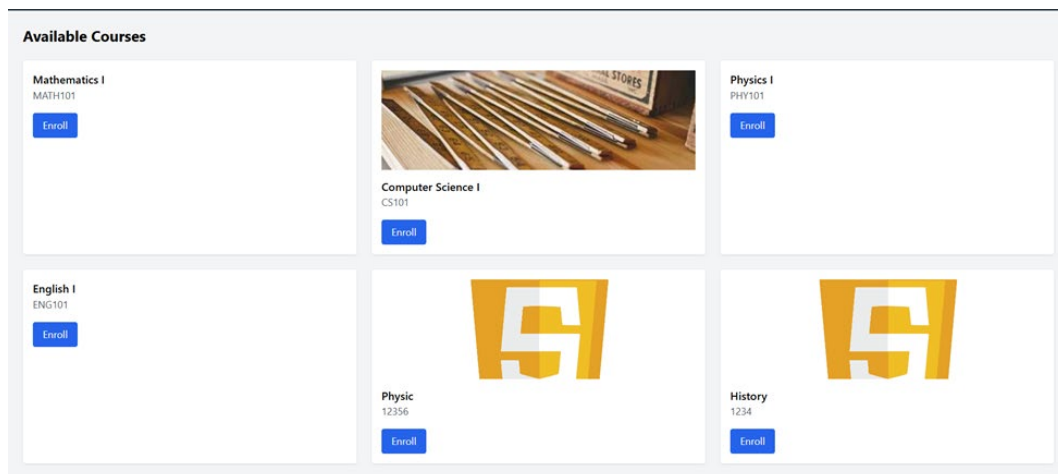
- Quick Action Cards
 - Implemented action cards for common admin tasks: Add Users: Enroll students and assign lecturers, Add Classes / Schedule: Create schedules and prevent conflicts, Add Course: Create and manage course content
 - Buttons prepared for future backend integration
- Navigation & Sidebar
 - Implemented sidebar navigation for: Users Management, Courses Management, Classes / Schedule
 - Active state handling for better user experience
 - Included quick action shortcuts in sidebar
- Recent Activity & Schedule Overview
 - Displayed recent user activity timeline
 - Showed today's class schedule with status indicators
 - Included basic system status metrics (uptime, response time)
- Technologies Used
 - Tailwind CSS for responsive styling
 - Thymeleaf for server-side templating
 - Fully responsive and mobile-friendly layout
 - Figma link:
<https://www.figma.com/design/y2mzEZUAfgVZIEZw4WlzoC/Vue-Js?node-id=755-37&p=f&t=oKuAbrhzNZhWKFH4-0>
- Notes
 - Dashboard currently uses static data for UI demonstration
 - Backend integration will be added in later development phases
- Quick Test



V. HENG Oulong

- Refactored the entire project to use @Controller with Thymeleaf instead of @RestController, following team agreement.
- Create another crud for enrollment: Admin and Student
- Separated the system clearly into Admin and Student flows:
 - Admin controllers for course and enrollment management
 - Student controllers for course viewing and enrolment
- Implemented Admin Course CRUD with Thymeleaf UI.

- Added and integrated course_image field into Course entity, DTOs, service, and database.
- Inserted course data directly into MySQL to validate database and UI integration
- Refactored CourseService:
 - Removed ModelMapper
 - Implemented manual DTO ↔ Entity mapping
 - Fixed enum (CourseStatus) handling
 - Prevented optimistic locking and null ID issues
- Fixed multiple 500 Internal Server Errors caused by:
 - Ambiguous ModelMapper mappings
 - DTO and entity mismatches
 - Incorrect form binding
- Verified that:
 - Admin can successfully create course
 - Delete courses
 - Edit course
 - Student can view all courses
 - Courses inserted in MySQL display correctly in Admin and Student views
- Student view all courses



```
mysql> select * from courses;
```

course_id	course_code	course_name	created_at	credits	description	max_capacity	status	updated_at	lecturer_id	course_image
1	MATH101	Mathematics I		3		30			5	
2	CS101	Computer Science I		4		40	ACTIVE		5	https://picsum.photos/400/300
3	PHY101	Physics I		4		25	ACTIVE		5	
5	ENG101	English I		2		35	DRAFT		5	
6	12356	Physic	2025-12-29 09:23:03.393024	3		5	DRAFT	2025-12-29 09:23:03.393852	5	https://repository-images.githubusercontent.com/539560750/278d8bbf-6aaf-4f38-a18f-e328305bcd7b
7	1234	History	2025-12-29 09:29:44.724091	5		10	DRAFT	2025-12-29 09:29:44.725039	5	https://repository-images.githubusercontent.com/539560750/278d8bbf-6aaf-4f38-a18f-e328305bcd7b

Admin view all courses and its Crud

Course Management			+ Add Course
Name	Code	Actions	
Mathematics I	MATH101	Edit	Delete
Computer Science I	CS101	Edit	Delete
Physics I	PHY101	Edit	Delete
English I	ENG101	Edit	Delete
Physic	12356	Edit	Delete
History	1234	Edit	Delete

[←](#) [→](#) [↺](#) [ⓘ](#) localhost:8080/admin/courses/create

Create Course

[Create Course](#)

After adding

localhost:8080/admin/courses

Course Management

+ Add Course

Name	Code	Actions
Mathematics I	MATH101	Edit Delete
Computer Science I	CS101	Edit Delete
Physics I	PHY101	Edit Delete
English I	ENG101	Edit Delete
Physic	12356	Edit Delete
History	1234	Edit Delete
Springboot	12345	Edit Delete

```
mysql> select * from courses;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| course_id | course_code | course_name | created_at | credits | description | max_capacity | status | updated_at | lecturer_id | course_image |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | MATH101 | Mathematics I | NULL | 3 | NULL | 30 | NULL | NULL | 5 | NULL |
| 2 | CS101 | Computer Science I | NULL | 4 | NULL | 40 | ACTIVE | NULL | 5 | https://picsum.photos/400/300 |
| 3 | PHY101 | Physics I | NULL | 4 | NULL | 25 | ACTIVE | NULL | 5 | NULL |
| 5 | ENG101 | English I | NULL | 2 | NULL | 35 | DRAFT | NULL | 5 | NULL |
| 6 | 12356 | Physic | 2025-12-29 09:23:03.393024 | 3 | NULL | 5 | DRAFT | 2025-12-29 09:23:03.393852 | 5 | https://repository-images.githubusercontent.com/539560750/278d0bbf-6aaf-4f38-a18f-e328305bcd7b |
| 7 | 1234 | History | 2025-12-29 09:29:44.724091 | 5 | NULL | 10 | DRAFT | 2025-12-29 09:29:44.725039 | 5 | https://repository-images.githubusercontent.com/539560750/278d0bbf-6aaf-4f38-a18f-e328305bcd7b |
| 9 | 12345 | Springboot | 2025-12-29 09:51:04.161978 | 4 | NULL | 12 | DRAFT | 2025-12-29 09:51:04.161978 | 5 | https://repository-images.githubusercontent.com/539560750/278d0bbf-6aaf-4f38-a18f-e328305bcd7b |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

After Deleting

localhost:8080/admin/courses

Course Management

+ Add Course

Name	Code	Actions
Mathematics I	MATH101	Edit Delete
Computer Science I	CS101	Edit Delete
Physics I	PHY101	Edit Delete
English I	ENG101	Edit Delete
Physic	12356	Edit Delete
History	1234	Edit Delete

```
mysql> select * from courses;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| course_id | course_code | course_name | created_at | credits | description | max_capacity | status | updated_at | lecturer_id | course_image |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | MATH101 | Mathematics I | NULL | 3 | NULL | 30 | NULL | NULL | 5 | NULL |
| 2 | CS101 | Computer Science I | NULL | 4 | NULL | 40 | ACTIVE | NULL | 5 | https://picsum.photos/400/300 |
| 3 | PHY101 | Physics I | NULL | 4 | NULL | 25 | ACTIVE | NULL | 5 | NULL |
| 5 | ENG101 | English I | NULL | 2 | NULL | 35 | DRAFT | NULL | 5 | NULL |
| 6 | 12356 | Physic | 2025-12-29 09:23:03.393024 | 3 | NULL | 5 | DRAFT | 2025-12-29 09:23:03.393852 | 5 | https://repository-images.githubusercontent.com/539560750/278d0bbf-6aaf-4f38-a18f-e328305bcd7b |
| 7 | 1234 | History | 2025-12-29 09:29:44.724091 | 5 | NULL | 10 | DRAFT | 2025-12-29 09:29:44.725039 | 5 | https://repository-images.githubusercontent.com/539560750/278d0bbf-6aaf-4f38-a18f-e328305bcd7b |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```