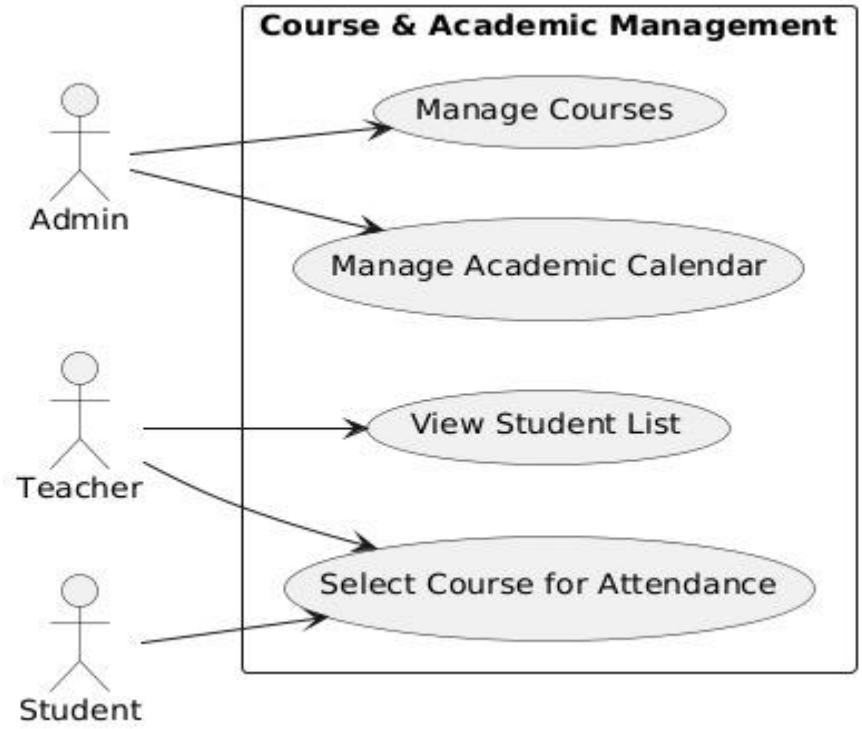
Course And Academic Management

Use Case Diagram:



Fully Dressed Use Case:

Use Case Name:

Course and Academic Management

Scenario:

Admin adds, updates, or removes course records in the academic system.

Triggering Event:

The admin selects the "Manage Courses" option from the system menu**.**

Brief Description:

The Admin manages the list of courses by adding new courses, updating existing ones, or removing outdated courses. Each course includes details such as course name, code, description, and assigned teacher**.**

Primary Actor:

Admin

Supporting Actors:

- Teacher: The course must be assigned to a valid teacher already in the system.
- Student: Students are indirectly affected, as they will enroll in or view the courses added/updated by the admin.

Stakeholders and Interests:

- Admin: Wants to ensure the course catalog is accurate and up-to-date.
- Teachers: Need to be correctly assigned to courses.
- Students: Need access to current and valid course offerings for enrollment and attendance.

Preconditions:

- Admin is logged into the system.
- Course Management module is available and functioning.
- Teacher data must already exist in the system to assign to courses**.**

Postconditions:

- Course data is saved, updated, or removed successfully.
- The changes are reflected in the system immediately.
- Teachers and students see the updated course list where applicable.

Main Flow (Basic Flow):

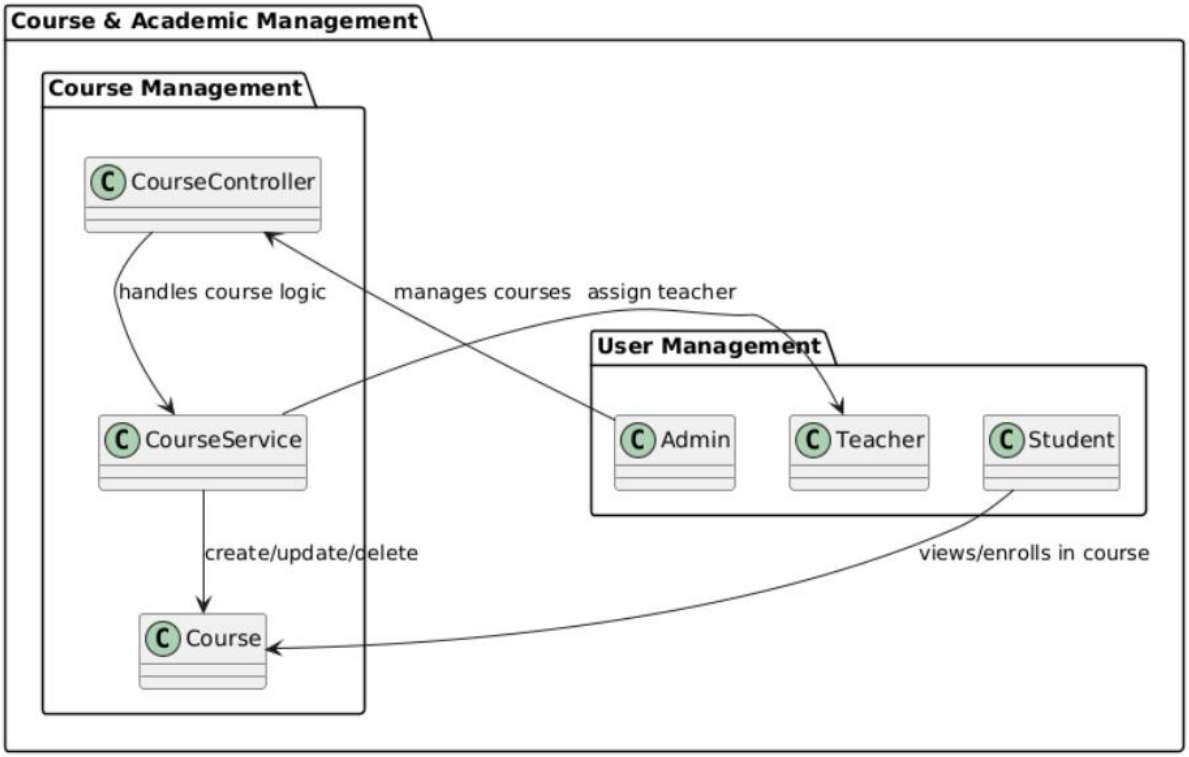
Step	**Actor (Admin)**	**System Response**
1	Selects “Manage Courses” option.	1.1 Displays course management interface.

Step	Actor (Admin)	System Response
2	Chooses to add a new course.	2.1 Displays form to enter course details.
3	Enters course name, code, description, and assigns a teacher.	3.1 Validates the input fields.
4	Confirms submission.	4.1 Saves course data to the database. 4.2 Shows success message.
5	(Optional) Selects a course to update or delete.	5.1 Displays selected course information. 5.2 Updates or removes course as requested.

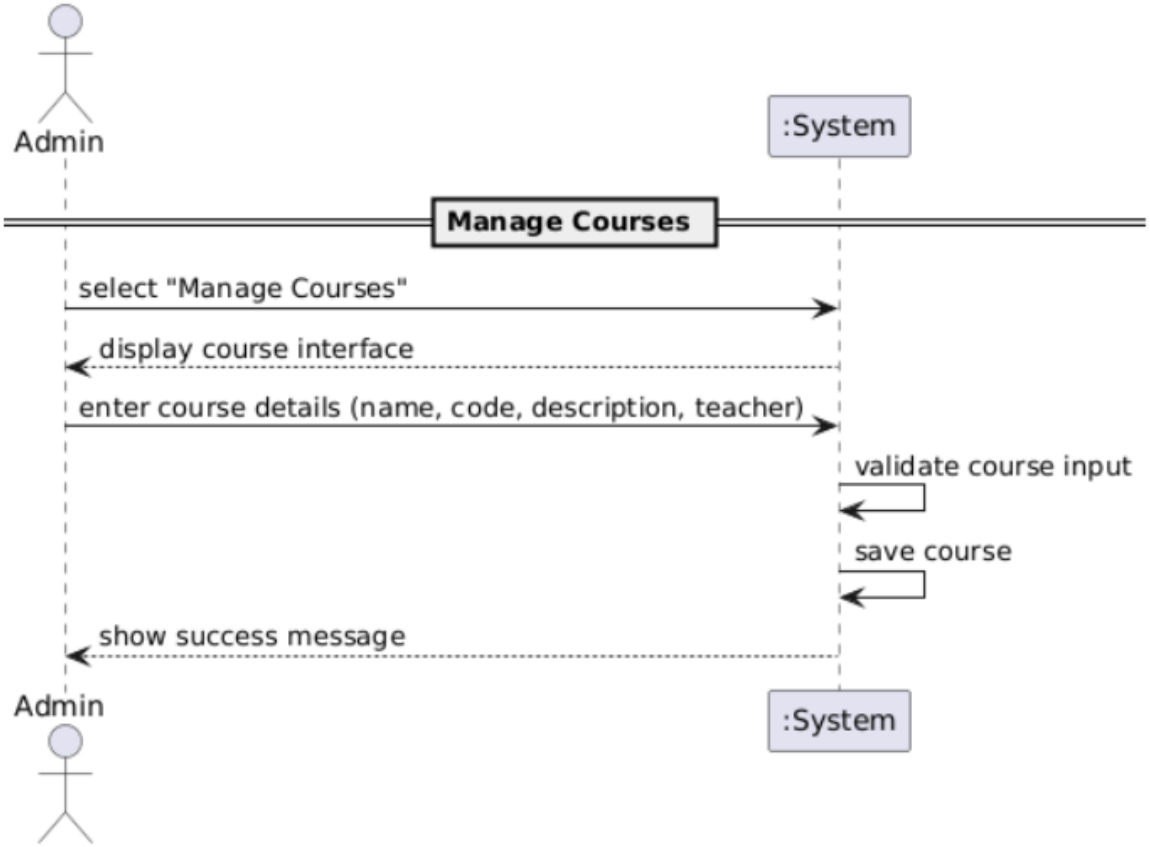
Alternate Flows / Exceptions:

- 3a. If any required field is missing → system shows an error and highlights missing data.
- 3b. If assigned teacher ID does not exist → system rejects input and asks to select a valid teacher.
- 4a. If database error occurs → system shows failure message and logs the error.
- 5a. If the course is currently assigned to students, deletion may be restricted → system warns admin.

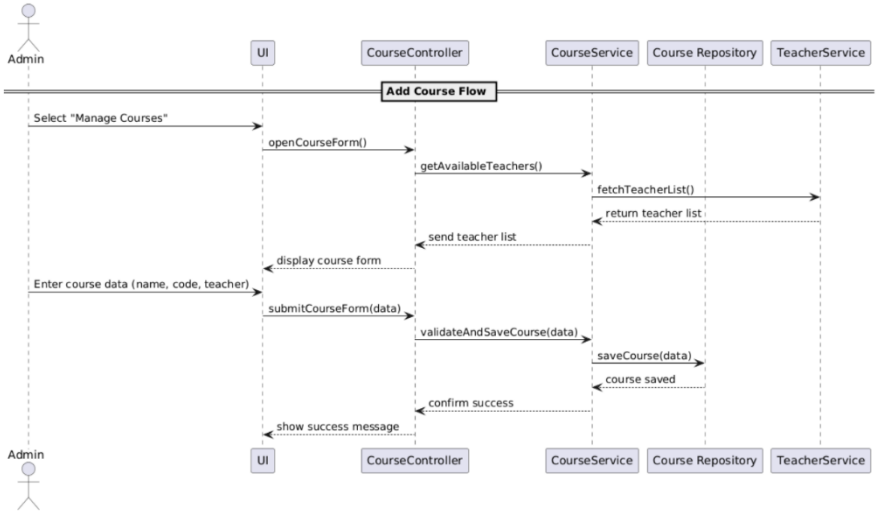
Package Diagram:


Course & Academic Management
Course Management
CourseController
handles course logic
manages courses
assign teacher
User Management
CourseService
Admin
Teacher
Student
create/update/delete
views/enrolls in course
Course

System Sequence Diagram:



Sequence Diagram Code:


Admin
UI
CourseController
CourseService
Course Repository
TeacherService
Add Course Flow
Select "Manage Courses"
openCourseForm()
getAvailableTeachers()
fetchTeacherList()
return teacher list
send teacher list
display course form
Enter course data (name, code, teacher)
submitCourseForm(data)
validateAndSaveCourse(data)
saveCourse(data)
course saved
confirm success
show success message
Admin
UI
CourseController
CourseService
Course Repository
TeacherService