Tech On-boarding Project - coursework management

Objective - To build an application where educators could conduct online courses and students can enroll for such courses.

Tech Stack

1. Node/Python
2. MongoDB
3. Git - Version Control
4. GCP - Deployment

Expectation - After completing this course you are expected to have the hands-on knowledge on following procedures

1. Authentication
2. CRUD Operations
3. File Upload
4. State Management
5. Background Jobs
6. Socket Implementation

Project Specifications

Users

The system will have two types of users

1. **Educator** - User will have the following responsibilities:
    a. Create course
    b. Add assignments
    c. Review assignments
2. **Student** - User will have the following responsibilities:
    a. Enroll for coursework
    b. Upload assignment solutions

User Stories

1. As an educator, I should be able to create a course so that students can enroll for it
2. As an educator, I should be able to upload the assignments for a course.
3. As an educator, I should be able to view students enrolled in my course.
4. As an educator, I should be able to view submissions for an assignment to grade them.
5. As an educator, I should be able to grade and assignment.
6. As a student, I should be able to view all active and upcoming courses.
7. As a student, I should be able to enroll for a course if a seat is available.
8. As a student, I should be able to view my currently enrolled courses.
9. As a student, I should be able to view the assignments for an enrolled course.
10. As a student, I should be able to submit solutions for an assignment.
11. As a student, I should be able to view my grade when the course ends.
12. As a student, I should be able to view my completed courses.
13. The course should automatically expire after completion date.
14. The students should be notified when an assignment submission slot opens.

Entities

User

User Model should have following fields

Field	Type	Description
_id	ObjectId	default id field for mongo
name	string	Name of the user
phone	number	Phone of the user
active	boolean	Whether user is active in our system or not
created_at	Date	Time at which user was created
updated_at	Date	Last time at which user was updated
type	Enum	admin/educator/student

API operations on User Model

1. Create (only an admin can create educators or other admins)
2. Login
3. Logout
4. Delete (only an admin can delete a user)

Course

Field	Type	Description
_id	ObjectId	default id field for mongo
educator_id	ObjectId	id of the educator who owns this course
name	string	Name of the course
description	string	course description
status	enum	upcoming, running, completed
created_at	date	time when created
updated_at	date	time last updated
capacity	number	number of students who can enroll for a course
start_date	date	course start date
end_date	date	course completion date

API operations on Course Model [Restriction - Only an educator can take action on his courses]

1. Create
2. Update Description/name, start_date, end_date
3. View Students enrolled in the course

API operation (for a student)

1. View all upcoming/active courses
2. View all completed courses for the student

State Management of the course status

upcoming running completed

Updates to the course should not break the state.

Enrollment

Field	Type	Description
_id	ObjectId	default id field for mongo
course_id	ObjectId	id of the course
student_id	ObjectId	id of the student
created_at	date	time of enrollment

API Operations on Enrollment model [Restriction - Only Student can perform]

1. Create Enrollment

Assignment

Field	Type	Description
_id	ObjectId	default id field for mongo
course_id	ObjectId	id of the course
description	string	text description of the assignment

start_time	date	time after which submissions would be accepted
end_time	date	time until which submissions would be accepted

API Operations on Assignment model [Restriction - Only Educator can perform]

1. Create Assignment
2. Update Assignment
3. Delete Assignment
4. View all the submissions against an assignment

Open APIs

1. View assignment

Submission

Field	Type	Description
_id	ObjectId	default id field for mongo
student_id	ObjectId	id of student
assignment_id	ObjectId	id of the assignment
submission_time	date	submission time
file_url	string	blob url
grade	enum[A,B,C,D,F]	the grade - can only be updated by educator

API Operations on Submission Model [Restriction - Only Student can perform]

1. Create submission

API operation on submission mode [restriction - only educator can perform]

1. Update grade on submission

Assignment Notification

There will be an hourly job running which will send notification via web socket to all the students enrolled in a course whenever the assignment is open for submission.

There will be an hourly job running which will update the status of the course to completed after the end_time of the course.