Introduction to the OOAD Approach of E-Learning Portal

Submitted By:
SWASTI DEWAN (500121067)
MAYANK VIJ (500122674)
GARV SACHDEVA (500120203)

Purpose & Key Features

E-learning platform unlike Swayam/Coursera supporting Students, Instructors, Administrators

User roles:
Students, Instructors, Administrators

Core features:
Course browsing & enrollment,Acess Tto content

Learning tools:
Interactive lectures, quizzes, assignments

Outcomes:
Certificate management and progress tracking

Design goals:
Scalable, accessible, industry-standard UX

Problem & OOAD Solution: Flexible, Scalable Learning

Addressing access limits with modular, maintainable system design

Limited reach and rigid time constraints hinder flexible learning

Need for anytime access, progress tracking, certification, scalable infrastructure

Apply **OOAD** to create modular, reusable, and maintainable components

Benefits: easier updates, component reusability, efficient extension and scaling

Requirement Analysis: Functional & Non-Functional

Clear, testable needs to ensure reliable, secure, and engaging online learning

Functional Requirements

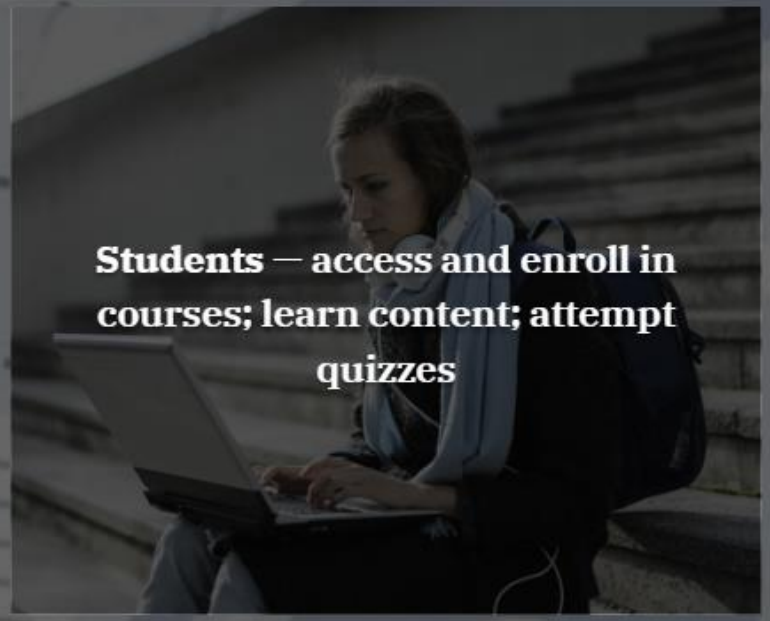
- User Login/Signup
- Role-based Access (Admin/User)
- Dashboard Display
- Add/Edit/Delete Records
- Search & Filter Data
- View Detailed Information
- Process Main Function (core operation)
- Generate Reports/Outputs
- Notifications & Alerts
- Data Storage & Retrieval
- Error Handling

Non-Functional Requirements

- High performance and low latency to ensure smooth video playback and fast page responses under normal and peak loads.
- Robust security and encryption for user data, secure authentication, role-based access control, and secure data storage and transmission.
- **Scalability for over 1000 users** simultaneously with horizontal scaling, load balancing, and efficient resource utilization.
- **User-friendly interface** to boost engagement with intuitive navigation, accessible design, and responsive layouts across devices.

User Roles and Responsibilities

Clear role boundaries to boost security, UX, and efficiency

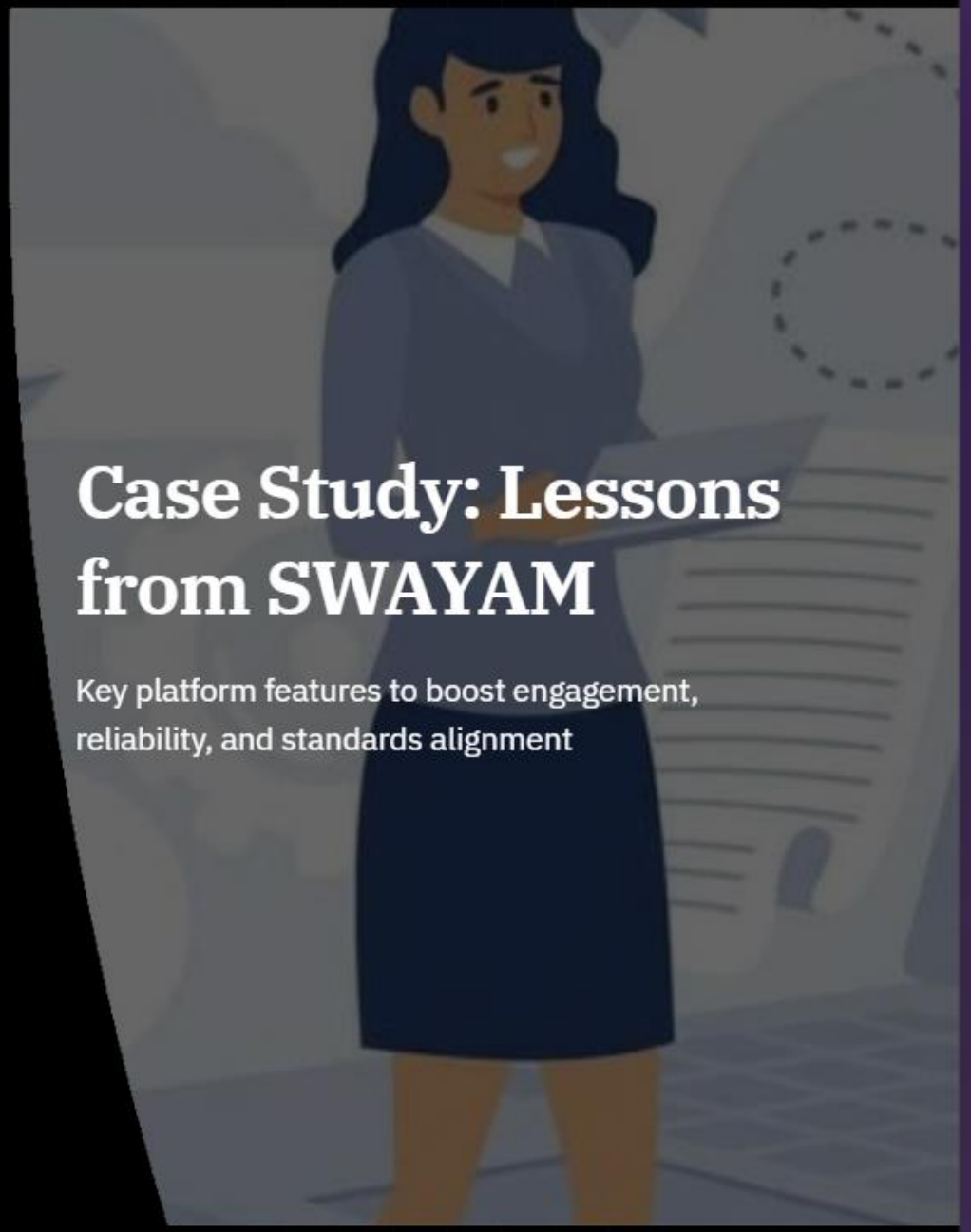


Students — access and enroll in courses; learn content; attempt quizzes

Instructors — create/manage materials; grade assignments; ensure course quality

Administrators — monitor operations; user management (CRUD); generate reports; maintain system health

Clear differentiation improves security, user experience, and operational efficiency



Case Study: Lessons from SWAYAM

Key platform features to boost engagement, reliability, and standards alignment

Massive Open Online Courses (MOOCs) with certification options to validate learning

Peer collaboration features to increase engagement and learning retention

Simple UI/UX design for accessibility and lower learner friction

Comprehensive progress tracking for learners and instructors

Robust content hosting ensuring reliability and scalability

Takeaway: Adopt certifications, collaboration, simple UX, tracking, and resilient hosting to align with global e-learning standards

Structural Modeling: Class Diagram Overview

Static OOAD model showing core classes, roles, and relationships

User (Student, Instructor, Admin)

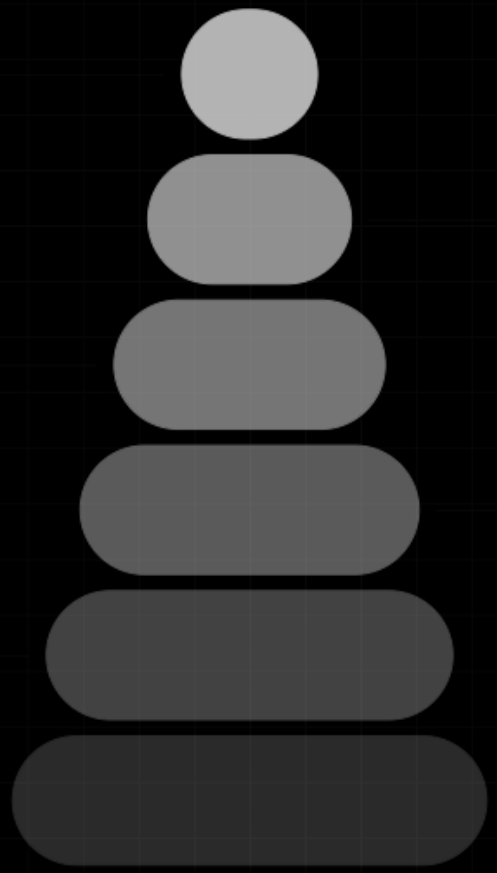
Primary actor hierarchy; Student, Instructor, Admin as User subtypes

Lecture & Assignment

Course components; Lectures and Assignments belong to Course

Certificate

Award entity tied to Course completion for Students



Course

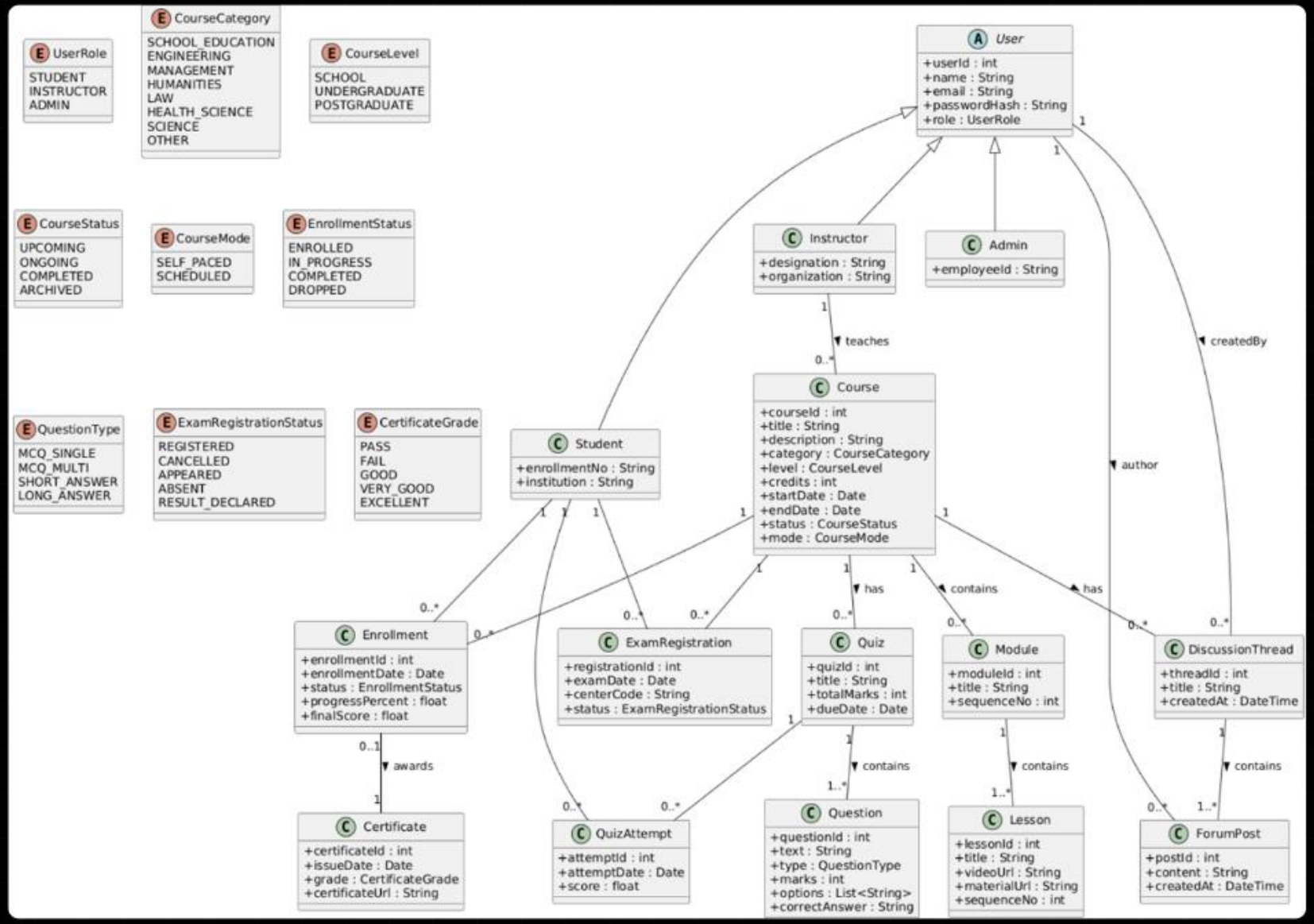
Container for content created by Instructor; central system entity

Enrollment

Association linking Student to Course; models registration

Design Value

Promotes modular, maintainable code aligned with OOAD principles


UserRole
STUDENT
INSTRUCTOR
ADMIN
CourseCategory
SCHOOL_EDUCATION
ENGINEERING
MANAGEMENT
HUMANITIES
LAW
HEALTH_SCIENCE
SCIENCE
OTHER
CourseLevel
SCHOOL
UNDERGRADUATE
POSTGRADUATE
CourseStatus
UPCOMING
ONGOING
COMPLETED
ARCHIVED
CourseMode
SELF_PACED
SCHEDULED
EnrollmentStatus
ENROLLED
IN_PROGRESS
COMPLETED
DROPPED
QuestionType
MCQ_SINGLE
MCQ_MULTI
SHORT_ANSWER
LONG_ANSWER
ExamRegistrationStatus
REGISTERED
CANCELLED
APPEARED
ABSENT
RESULT_DECLARED
CertificateGrade
PASS
FAIL
GOOD
VERY_GOOD
EXCELLENT
User
+userId : int
+name : String
+email : String
+passwordHash : String
+role : UserRole
Instructor
+designation : String
+organization : String
Admin
+employeeId : String
Student
+enrollmentNo : String
+institution : String
Course
+courseId : int
+title : String
+description : String
+category : CourseCategory
+level : CourseLevel
+credits : int
+startDate : Date
+endDate : Date
+status : CourseStatus
+mode : CourseMode
Enrollment
+enrollmentId : int
+enrollmentDate : Date
+status : EnrollmentStatus
+progressPercent : float
+finalScore : float
ExamRegistration
+registrationId : int
+examDate : Date
+centerCode : String
+status : ExamRegistrationStatus
Quiz
+quizId : int
+title : String
+totalMarks : int
+dueDate : Date
Module
+moduleId : int
+title : String
+sequenceNo : int
DiscussionThread
+threadId : int
+title : String
+createdAt : DateTime
Certificate
+certificateId : int
+issueDate : Date
+grade : CertificateGrade
+certificateUrl : String
QuizAttempt
+attemptId : int
+attemptDate : Date
+score : float
Question
+questionId : int
+text : String
+type : QuestionType
+marks : int
+options : List<String>
+correctAnswer : String
Lesson
+lessonId : int
+title : String
+videoUrl : String
+materialUrl : String
+sequenceNo : int
ForumPost
+postId : int
+content : String
+createdAt : DateTime
teaches
createdBy
author
has
contains
awards

Behavioral Modeling — Use Case Diagram

Map actors to core system functions to validate requirements

Actors

1

Students, Instructors, Admins who interact with the system.

Authentication

2

Login and registration use cases for user access control.

Administration

6

System management: user roles, system settings, reports.

Enrollment & Courses

3

Course registration and enrollment workflows for students.

Assessment & Grading

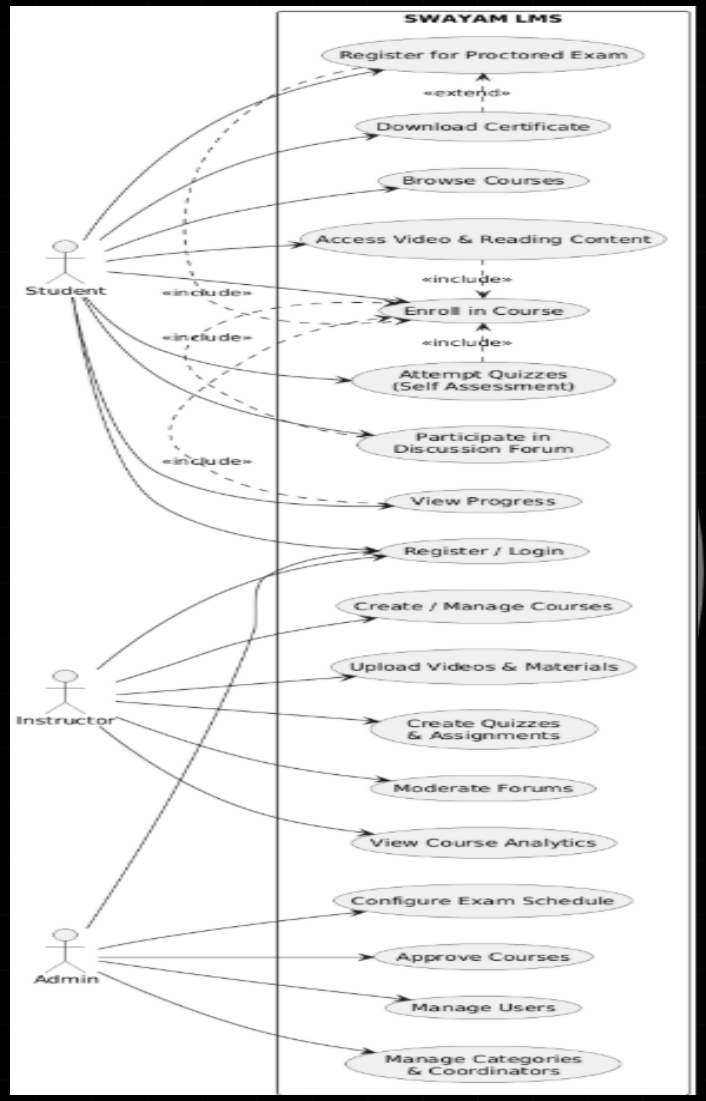
5

Grading, grade publication, and feedback loops for learners.

Content Management

4

Instructors upload materials; admins oversee content policies.



Enrollment Process Flow

Sequence diagram of real-time interactions from login to enrollment confirmation

Student Login

Student initiates login and submits credentials to the system

Authenticate Credentials

System validates credentials and returns success or failure

Request Enrollment

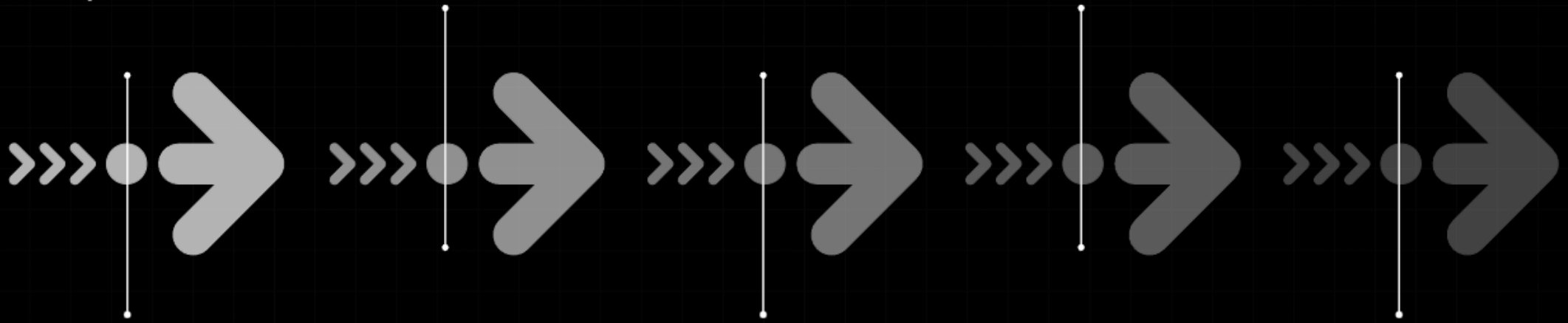
Authenticated student requests enrollment in chosen course

Update Database

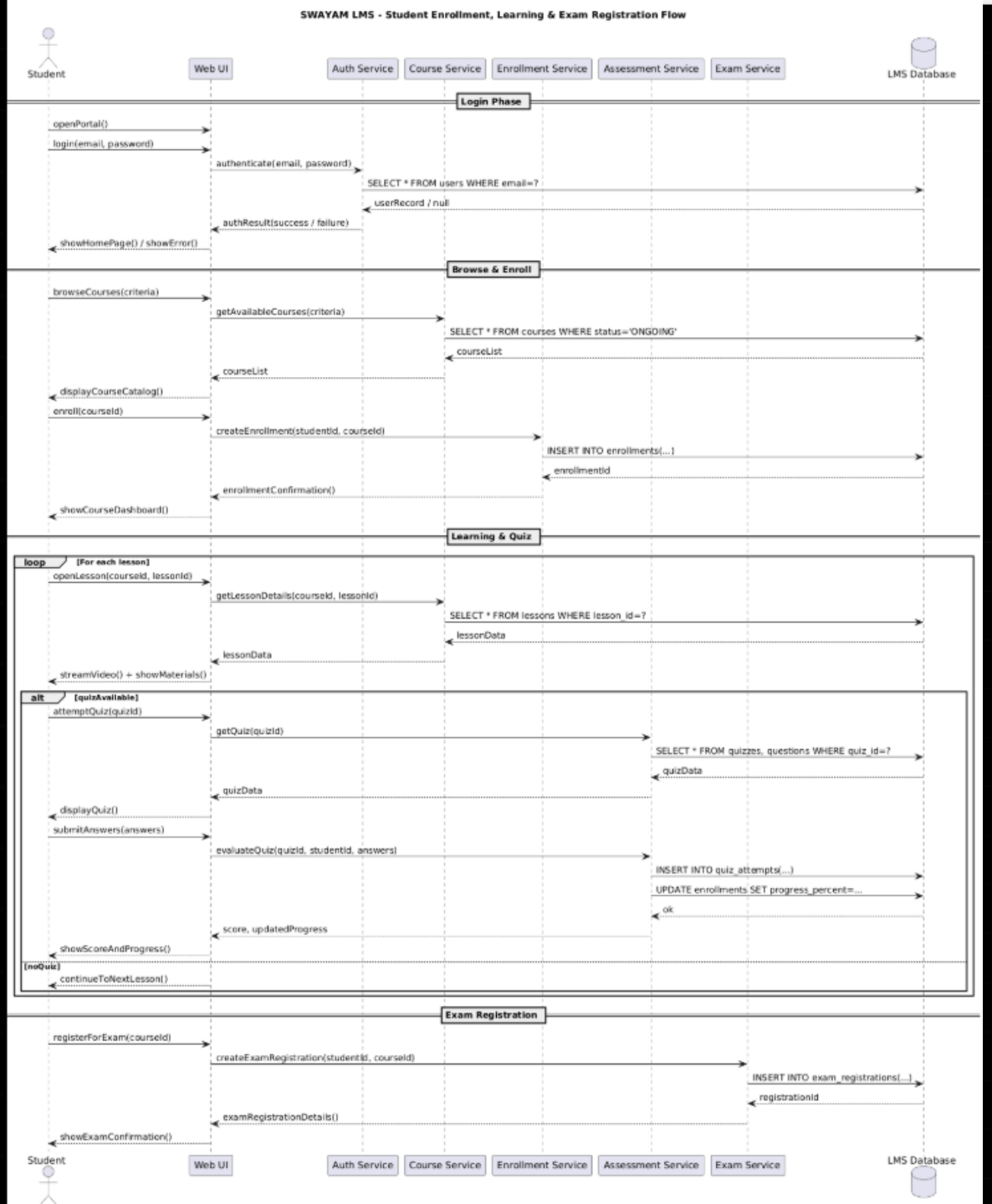
System updates enrollment records in the database

Confirm Enrollment

System sends enrollment confirmation back to the student



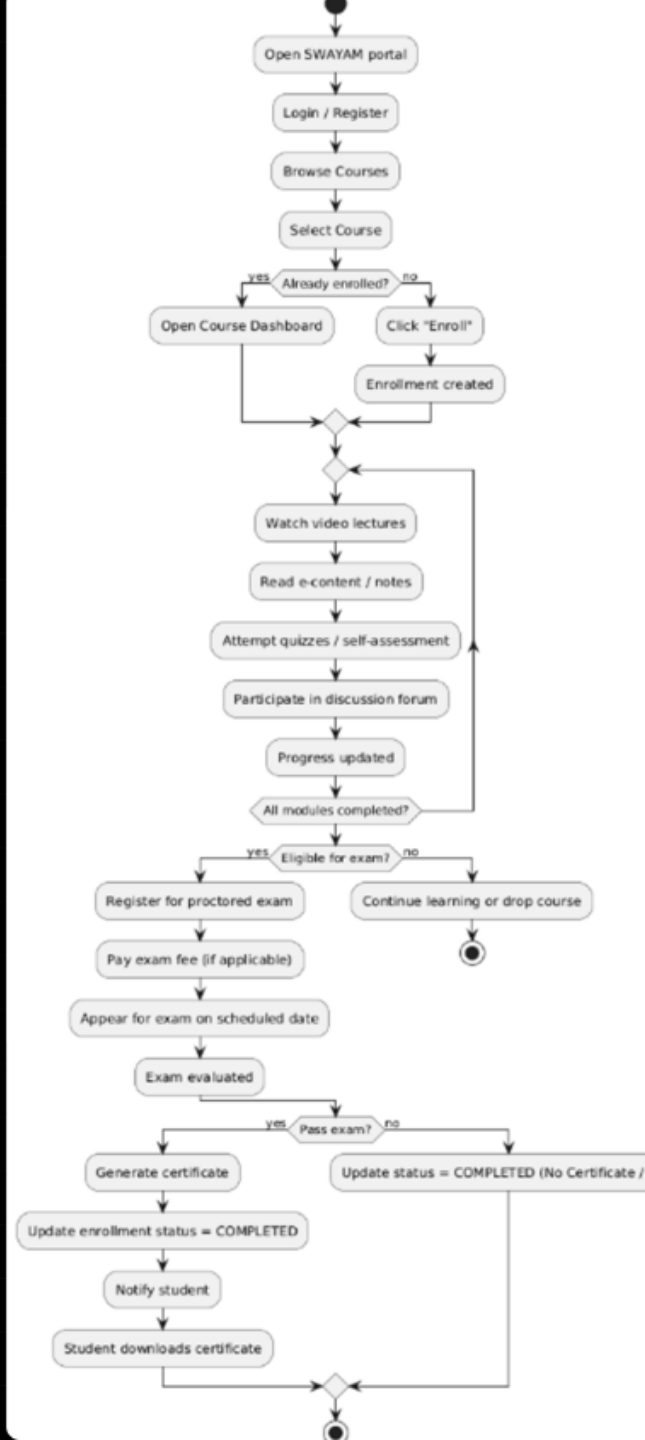
Student Learning :Sequence Diagram



Student Learning Flow :Activity Diagram

1. Login (Authenticate and access the learning dashboard)
2. Browse Courses (Explore course catalog and preview syllabi)
3. Enroll (Register for selected course and confirm enrollment)
4. Watch Lectures (Stream or download lecture videos and resources)
5. Attempt Quizzes (Complete formative or summative quizzes to assess learning)
6. Submit Assignments (Upload homework or projects for evaluation)
7. Receive Certificate (Earn certificate upon meeting course completion criteria)

Visual sequence of student interactions



DFD Level 0: Overview of Data Movements

Top-level flows between students, instructors, admins, core processes, and data stores

External Entities

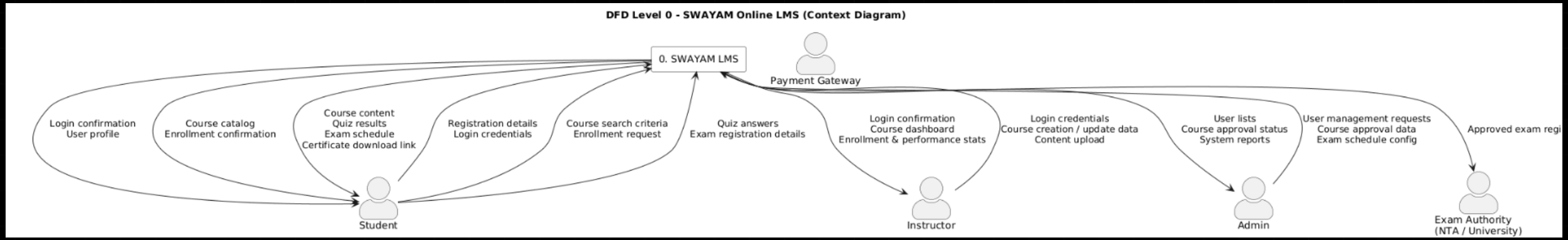
Students, Instructors, Admins exchanging requests and responses with the system

Data Stores

User DB, Course DB, Storage holding profiles, course data, and files

Purpose

Clarify system boundaries and main data pathways for secure, efficient operation



Core Processes

Login, Enrollment, Submission, Certificate Issuance as central services

Primary Data Flows

Authentication, enrollment requests, submissions, certificates and storage interactions

Data Flow Diagram Level 1: Detailed Data Processes

Breakdown of login, enrollment, submission and certification flows for optimization

Authentication (Login)

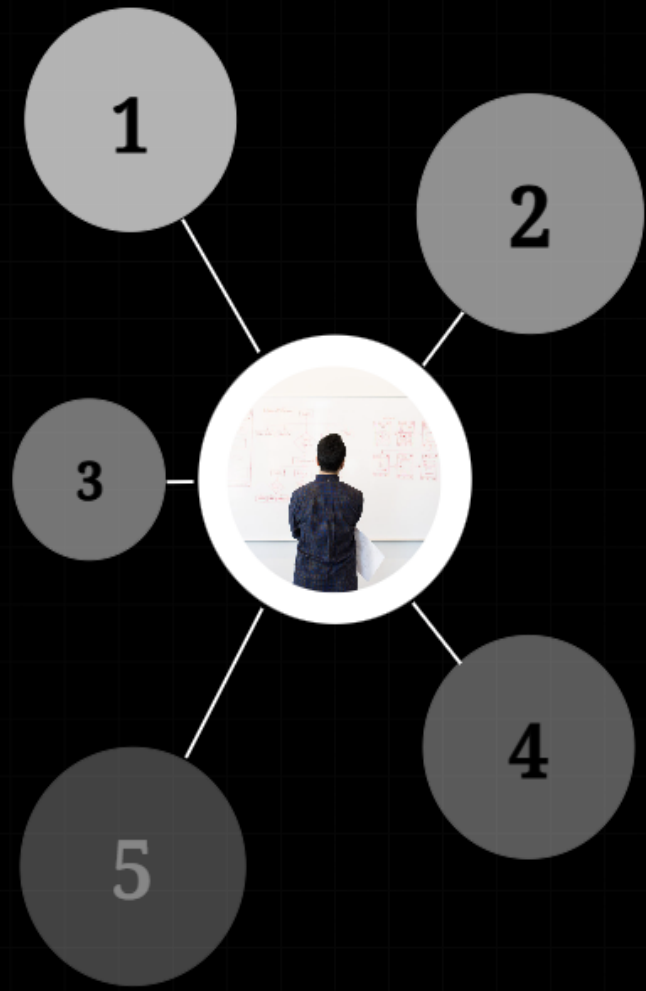
User credentials validated; session tokens issued; audit logs recorded.

Enrollment

Course selection and profile data stored; prerequisites checked; enrollment DB updated.

Submission

Assignments/uploads routed to grading process; timestamps and versioning recorded.

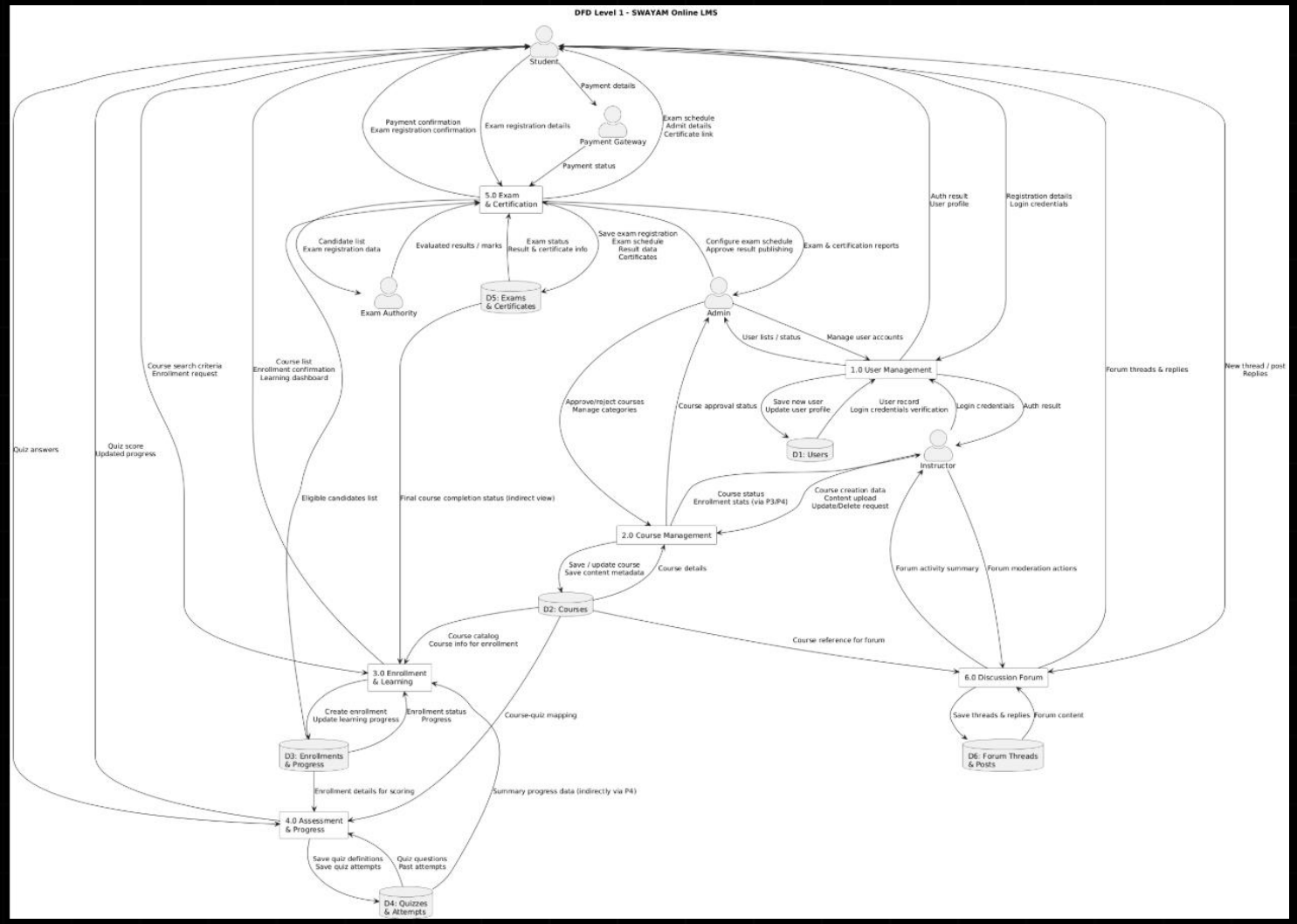


Certification

Completion verified, certificates generated and archived; notification sent to user.

Datastores & APIs

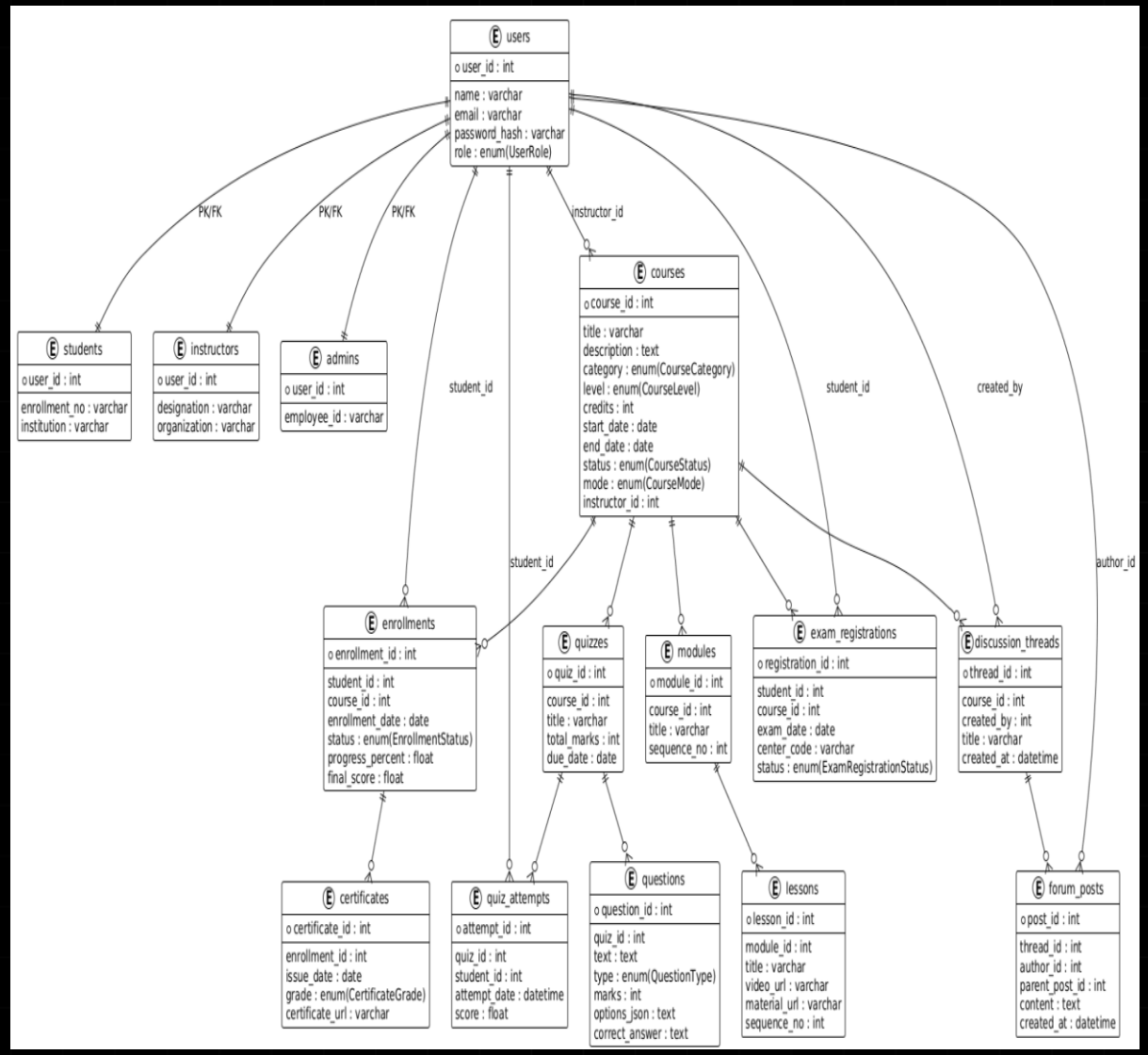
Central databases and APIs mediate data exchanges and ensure consistency.


DFD Level 1 - SWAYAM Online LMS
Student
Payment details
Payment Gateway
Payment confirmation
Exam registration confirmation
Exam registration details
Exam schedule
Admit details
Certificate link
Payment status
5.0 Exam & Certification
Auth result
User profile
Registration details
Login credentials
Candidate list
Exam registration data
Evaluated results / marks
Exam status
Result & certificate info
Save exam registration
Exam schedule
Result data
Certificates
Configure exam schedule
Approve result publishing
Exam & certification reports
Exam Authority
D5: Exams & Certificates
Admin
User lists / status
Manage user accounts
1.0 User Management
Course search criteria
Enrollment request
Course list
Enrollment confirmation
Learning dashboard
Approve/reject courses
Manage categories
Course approval status
Save new user
Update user profile
User record
Login credentials verification
Login credentials
Auth result
Forum threads & replies
New thread / post
Replies
Quiz answers
Quiz score
Updated progress
D1: Users
Instructor
Eligible candidates list
Final course completion status (indirect view)
Course status
Enrollment stats (via P3,P4)
Course creation data
Content upload
Update/Delete request
2.0 Course Management
Save / update course
Save content metadata
Course details
Forum activity summary
Forum moderation actions
D2: Courses
Course catalog
Course info for enrollment
Course reference for forum
3.0 Enrollment & Learning
6.0 Discussion Forum
Create enrollment
Update learning progress
Enrollment status
Progress
Course-quiz mapping
Save threads & replies
Forum content
D3: Enrollments & Progress
D6: Forum Threads & Posts
Enrollment details for scoring
Summary progress data (indirectly via P4)
4.0 Assessment & Progress
Save quiz definitions
Save quiz attempts
Quiz questions
Past attempts
D4: Quizzes & Attempts

Database Schema & ENUM Lookup Design

Relational tables, keys, and ENUMs for consistent e-learning data

Table	Primary Key	Foreign Keys / Relations	Notes
Users	user_id	none	Stores student/instructor profiles
Courses	course_id	owner_id → Users.user_id	Course metadata and ownership
Lectures	lecture_id	course_id → Courses.course_id	Lecture content linked to course
Enrollments	enrollment_id	user_id → Users.user_id, course_id → Courses.course_id	Tracks student progress; uses ENUM status
Assignments	assignment_id	course_id → Courses.course_id, lecture_id → Lectures.lecture_id	Assessment records and deadlines
ENUM Lookup	status_id	used by Enrollments.status	Values: **Enrolled**, **In-Progress**, **Completed**



Enums

UserRole
- STUDENT
- INSTRUCTOR
- ADMIN

CourseCategory
- SCHOOL_EDUCATION
- ENGINEERING
- MANAGEMENT
- HUMANITIES
- LAW
- HEALTH_SCIENCE
- SCIENCE
- OTHER

CourseLevel
- SCHOOL
- UNDERGRADUATE
- POSTGRADUATE

CourseStatus
- UPCOMING
- ONGOING
- COMPLETED
- ARCHIVED

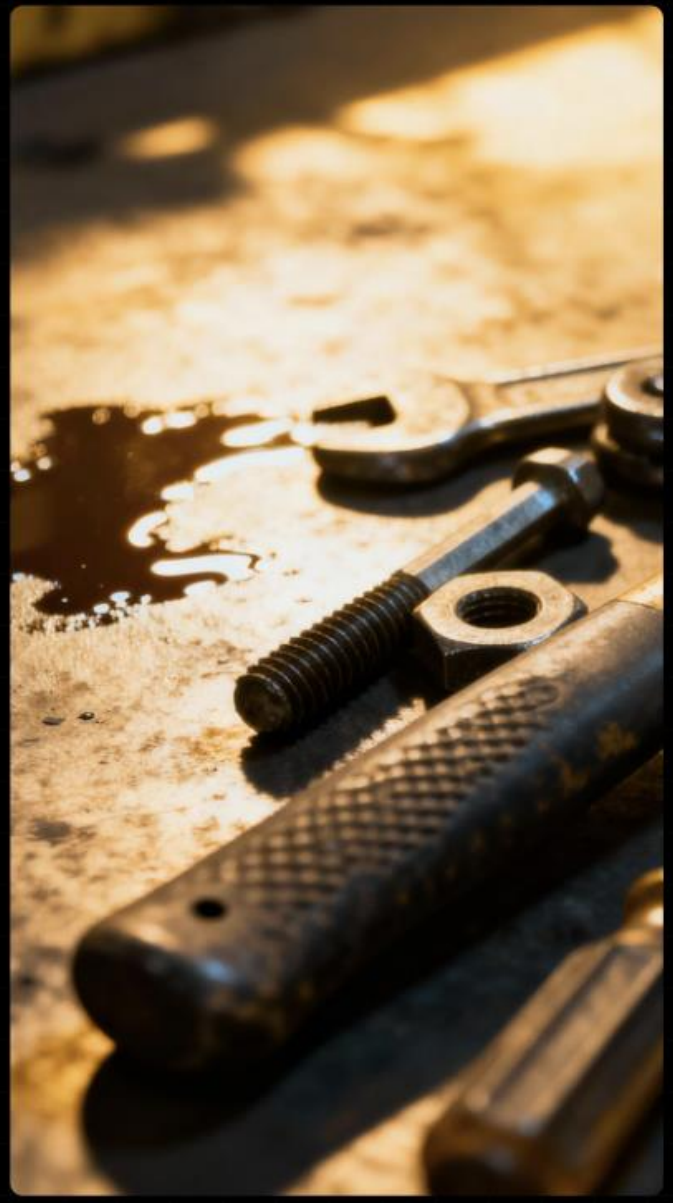
CourseMode
- SELF_PACED
- SCHEDULED

EnrollmentStatus
- ENROLLED
- IN_PROGRESS
- COMPLETED
- DROPPED

QuestionType
- MCQ_SINGLE
- MCQ_MULTI
- SHORT_ANSWER
- LONG_ANSWER

ExamRegistrationStatus
- REGISTERED
- CANCELLED
- APPEARED
- ABSENT
- RESULT_DECLARED

CertificateGrade
- PASS
- FAIL
- GOOD
- VERY_GOOD
- EXCELLENT



Testing, Maintenance & Future Enhancements

Validate functionality, sustain performance, and enrich learner experience

Unit, **Integration**, **System**, and **Acceptance** testing to validate components, data flow, overall functionality, and user satisfaction

Ongoing maintenance: performance tuning, scalability improvements, and bug fixing to sustain optimal operations

Future features: **AI-driven course recommendations**, **gamified learning**, **mobile app integration**, and **real-time chat support**

SECTION3

END