System Design Document

Learning Management System

Prepared By

Jeevansree E

Table of Contents

1. Introduction....3
2. System Overview....3
3. Technology Stack....3
4. System Architecture Diagram....4
5. Component Diagram....4
6. Data Flow Diagram....4
7. Sequence Diagrams....5
    7.1 User & Course Management....5
    7.2 Content Delivery & Tests....6
    7.3 Forum, Compiler & Analytics....7
    7.4 Question Of The Day & Chat....8
8. Database Design....8
    8.1 Key entities:....8
    8.2 ER Diagram....9
8. Security Considerations....10
9. Scalability and Maintainability....10
10. Conclusion....10

1. Introduction

The Learning Management System (LMS) is a comprehensive platform that facilitates the administration, documentation, tracking, reporting, and delivery of educational courses. It supports three primary user roles: Super Admin, Admin (Instructor), and Student.

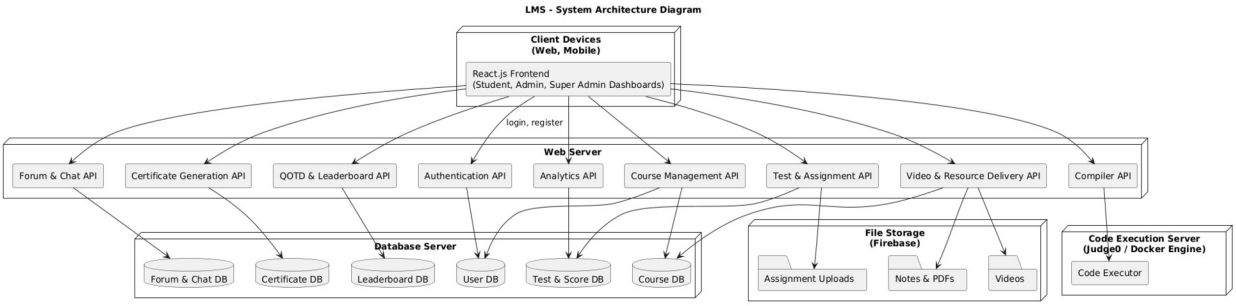
2. System Overview

- **Super Admin**: Manages courses, assigns instructors and students, and oversees platform operations.
- **Admin (Instructor)**: Manages course content, tests, student progress, and interacts via forums and chat.
- **Student**: Consumes learning content, participates in assessments, interacts via forums, and receives certificates upon completion.

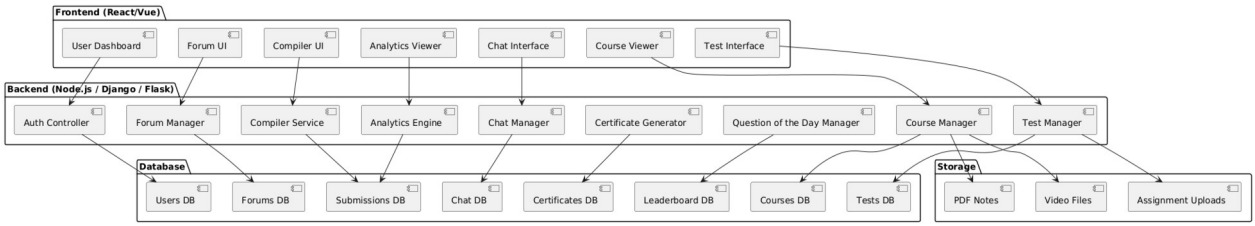
3. Technology Stack

- **Frontend**: React.js
- **Backend**: Node.js / Express.js
- **Database**: MongoDB
- **Authentication**: Firebase Auth / JWT
- **Video Handling**: Cloudinary or AWS S3
- **Compiler API**: Judge0 or Sphere Engine
- **Analytics**: Chart.js / D3.js
- **Chat & Messaging**: Socket.IO / Firestore

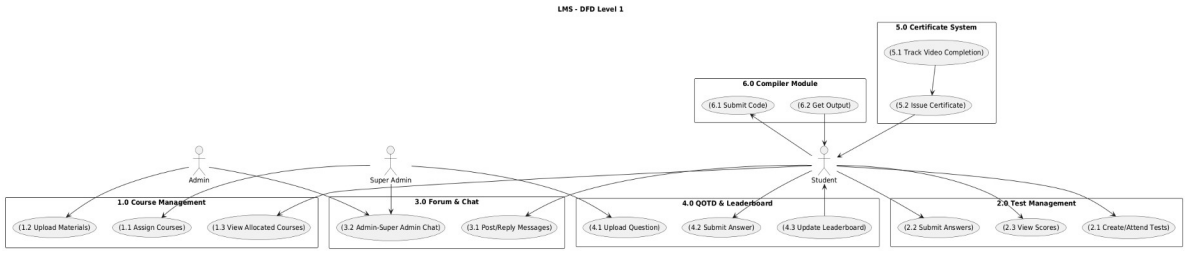
4. System Architecture Diagram



5. Component Diagram

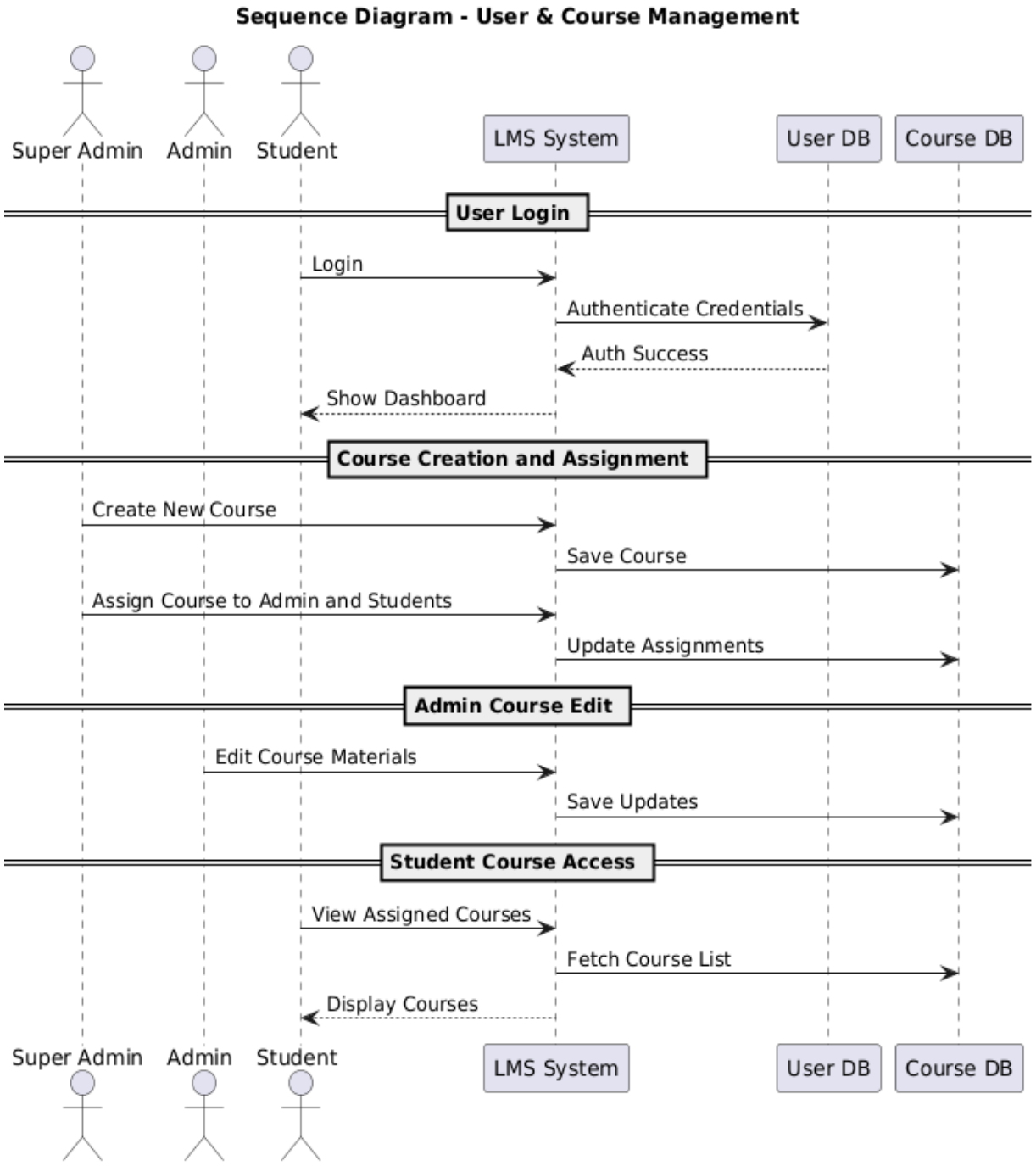


6. Data Flow Diagram

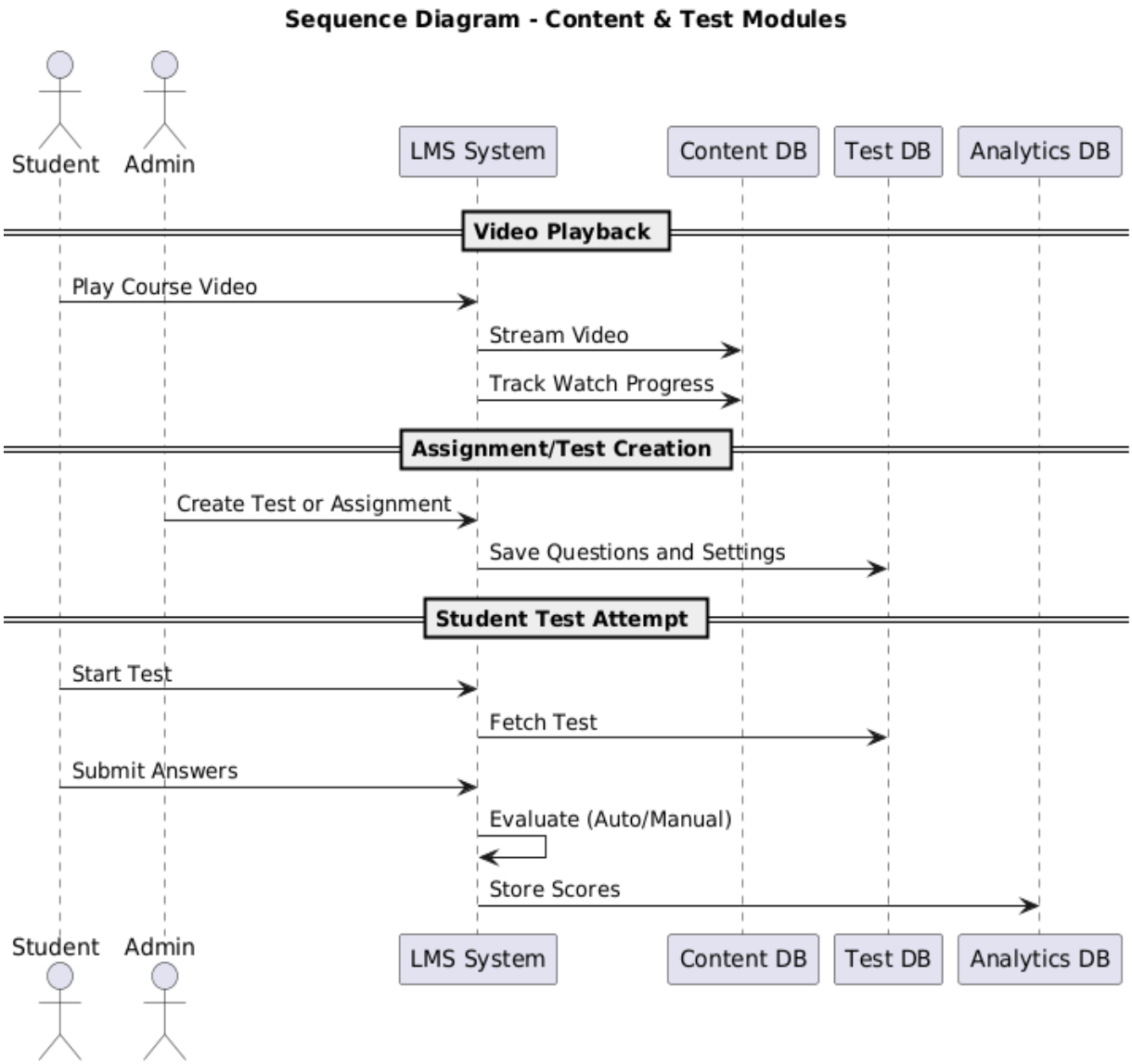


7. Sequence Diagrams

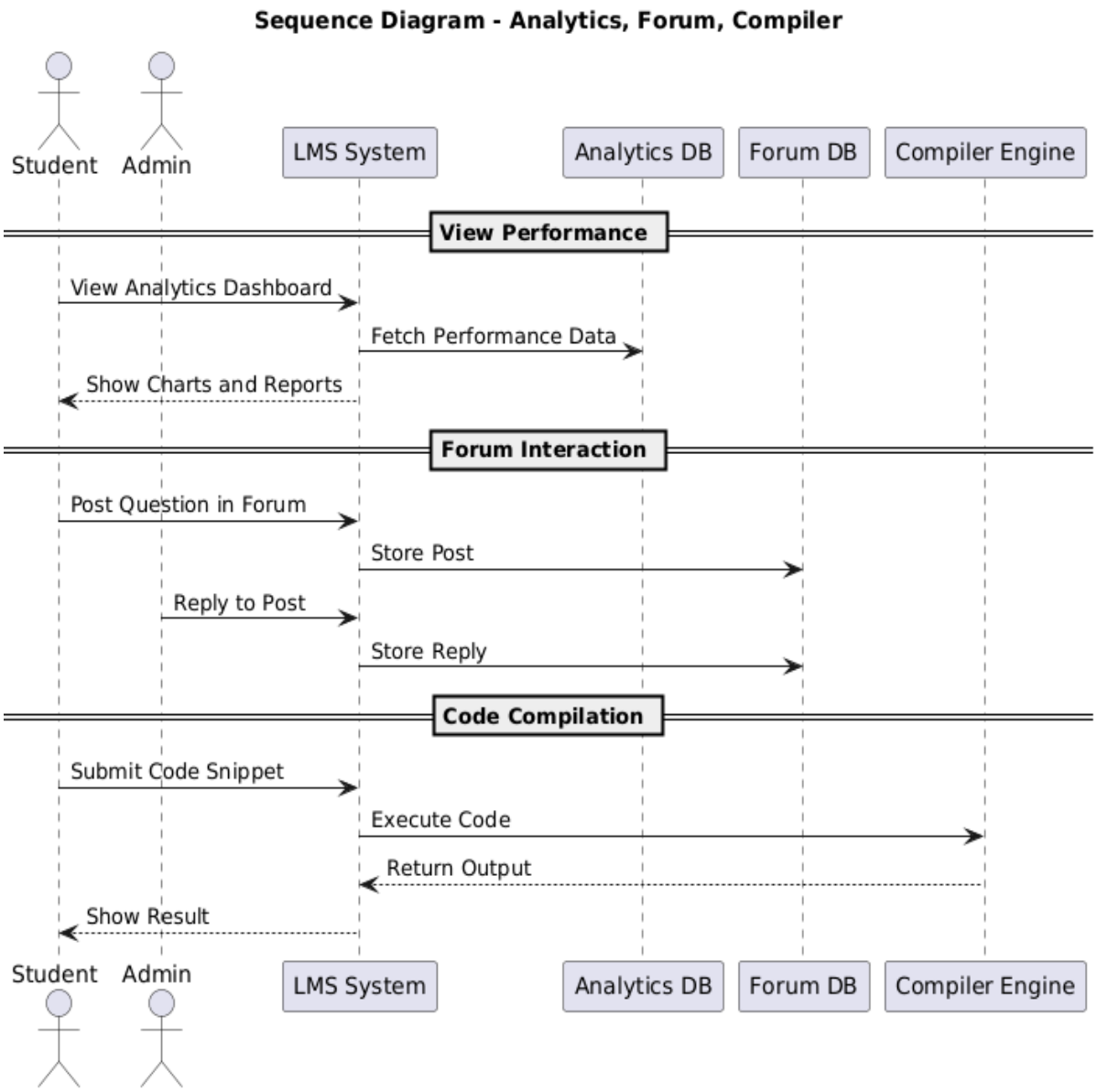
7.1 User & Course Management



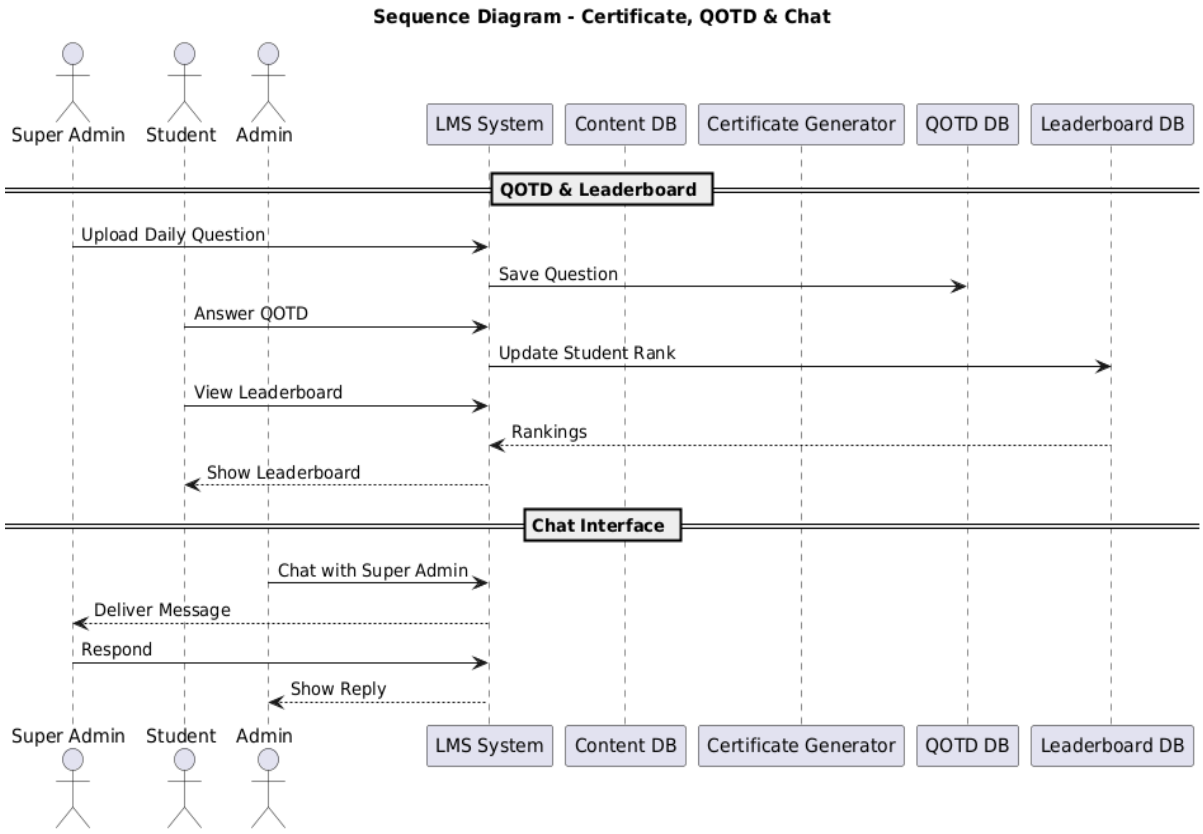
7.2 Content Delivery & Tests



7.3 Forum, Compiler & Analytics



7.4 Question Of The Day & Chat



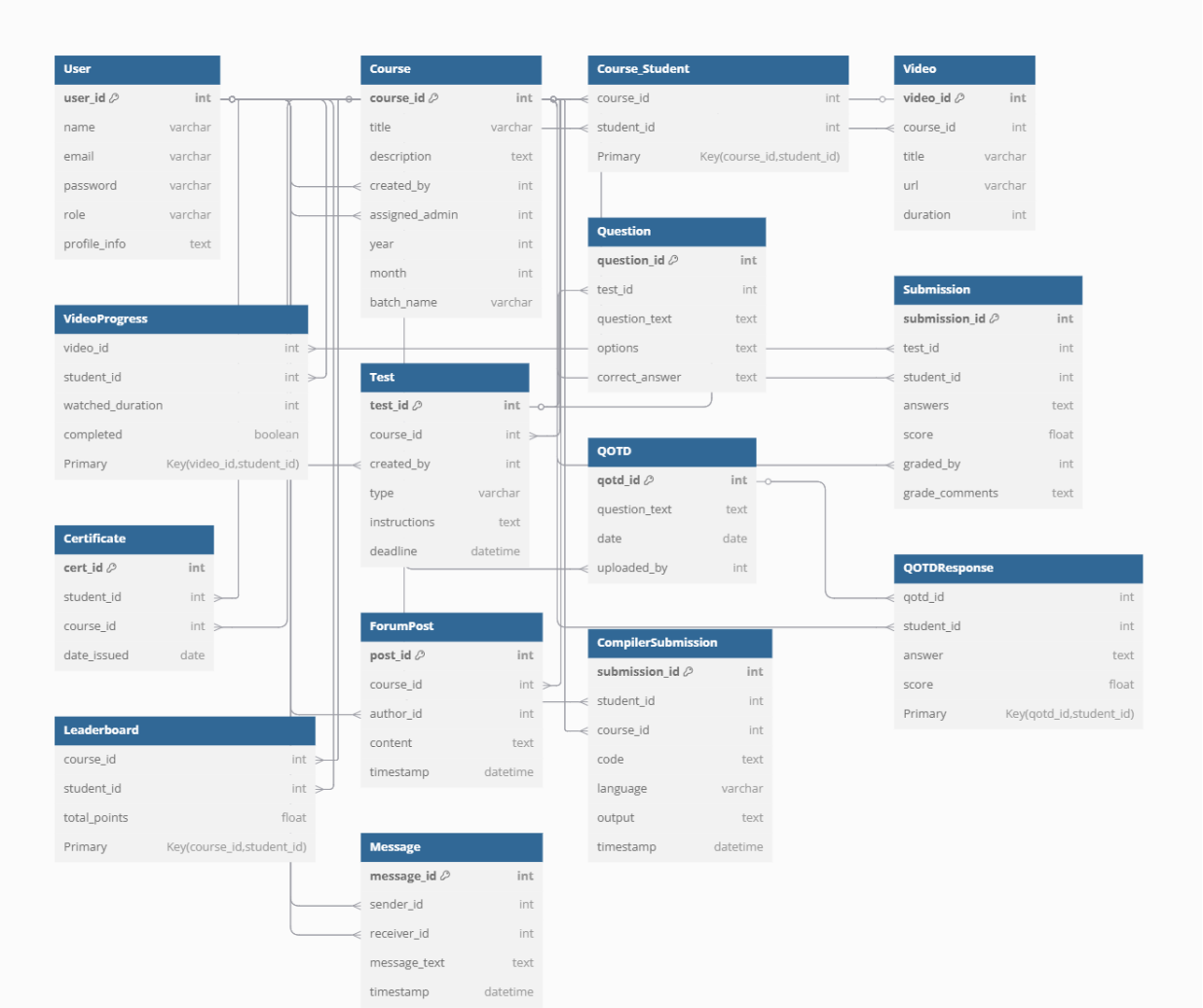
8. Database Design

8.1 Key entities:

- Users (id, name, role, auth info)
- Courses (id, title, instructor, student list)
- Videos (course_id, video_id, watched_by)
- Tests & Assignments (test_id, type, questions)
- Submissions (student_id, test_id, answers, score)

- Certificates
- QOTD & Leaderboard
- Forum Posts
- Messages (chat system)

8.2 ER Diagram



8. Security Considerations

- JWT/Firebase for secure login
- Role-based access control
- Encrypted video access links
- Test integrity via protected mode
- Secure chat & forum access

9. Scalability and Maintainability

- Modular structure
- Scalable backend APIs
- Microservices-ready design
- Cloud storage for media
- Stateless sessions for load balancing

10. Conclusion

The LMS is designed to provide a seamless and interactive learning experience. By automating content delivery, performance tracking, and communication, it replaces fragmented systems like WhatsApp-based test sharing. The architecture ensures modularity, security, and future scalability, supporting a professional educational ecosystem.