

LMS SYSTEM — COMPLETE MEGA DOCUMENT

1. System Overview

A full-stack Learning Management System built using NestJS, Next.js, Mantine UI, SCSS, and MongoDB. Includes user roles, admin panel, course authoring, payments, quizzes, analytics, and AI features.

2. Tech Stack

Frontend

- Next.js
- Mantine UI
- SCSS
- Zustand/Redux

Backend

- NestJS
- JWT Authentication
- Role-Based Access Control

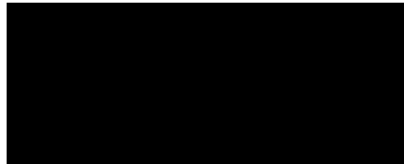
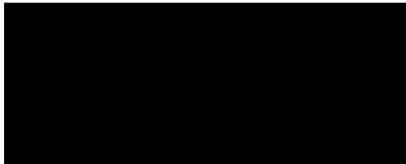
Database

- MongoDB (Atlas)

AI (Phase 2)

- LLM-based tutor
- Auto quiz generator
- Recommendation engine

3. ER Diagram



4. Wireframes (Low Fidelity)

Student Dashboard

| Header |

|-----|

| My Courses | Continue Learning |

|-----|

| Course List Cards |

Admin Dashboard

| Sidebar | Stats Tiles |

|-----|

| Course Table | User Table |

5. API Endpoints

Method	Endpoint	Description
POST	/auth/register	Create an account
POST	/auth/login	Login user
GET	/courses	List all courses
POST	/courses	Create course
POST	/enroll	Enroll student
GET	/progress/:id	Track progress
POST	/quiz/submit	Submit quiz

6. Full Sprint Plan

Sprint 1: Project Setup

- Next.js + Mantine + SCSS
- NestJS modules setup
- MongoDB connection

Sprint 2: Auth System

- JWT login/register
- Email verification
- Forgot/reset password

Sprint 3: Admin Panel + RBAC

- Roles: Student, Instructor, Admin, Super Admin
- Protected routes
- Admin dashboard

Sprint 4: Course Module

- CRUD for courses
- Lesson module
- Rich text editor

Sprint 5: Student Features

- Enrollment
- Progress tracking
- Certificates

Sprint 6: Quizzes

- Quiz creation
- Auto grading

Sprint 7: Payments

- Razorpay integration

Sprint 8: Analytics

- Revenue analytics
- Student performance

Sprint 9: AI Phase

- AI tutor chatbot
- Auto quiz generator
- Smart recommendations

7. System Architecture (High-Level)

Frontend (Next.js)

↓

Backend API (NestJS)

↓

MongoDB Atlas

CDN: Cloudflare

Deployment: Vercel (frontend), Render/Railway (backend)

8. Sequence Diagram (Text Representation)

User → Frontend: Request login

Frontend → Backend: Send credentials

Backend → Database: Validate user

Backend → Frontend: Return token

Frontend → Backend: Access protected endpoint

9. AI Features (Phase 2)

AI Tutor Chatbot

- Contextual learning
- Lesson-aware answers

Auto Quiz Generator

- Input: Lesson text
- Output: MCQs + descriptive questions

Personalized Recommendations

- ML model observes:
- Progress
- Weak areas
- Learning speed

10. Conclusion

This mega document covers the full architecture, sprints, diagrams, wireframes, and AI roadmap for the LMS system.