LMS SYSTEM — FULL PROJECT PLAN (NestJS + Next.js + Mantine + SCSS + MongoDB)

PHASE 1 — FOUNDATION

Sprint 1: Requirements & Planning

- Define user roles (Student, Instructor, Admin, Super Admin)

- Create complete feature list

- Prepare system architecture (frontend, backend, DB)

- Prepare API specification

- Plan database models (User, Course, Lesson, Quiz, Enrollment, Progress)

Sprint 2: Project Setup

- Initialize Next.js (frontend)

- Initialize NestJS (backend)

- Setup MongoDB connection

- Configure folder structure (admin panel + client)

- Install Mantine + SCSS + Zustand/Redux

PHASE 2 — AUTHENTICATION & ACCESS CONTROL

Sprint 3: Auth Module

- Register, Login, Logout

- JWT access + refresh tokens

- Forgot/Reset password

- Email verification

- Role assignment (Student, Instructor, Admin, Super Admin)

Sprint 4: Admin Panel Access Control

- RBAC middleware in backend

- Multi-role UI guard on frontend

- Navigation based on roles

- Admin dashboard prototype

PHASE 3 — COURSE MANAGEMENT

Sprint 5: Course Module (Backend)

- Course CRUD

- Category and Subcategory

- Course publish/unpublish

- Upload thumbnails (S3 optional, local fallback)

- Lessons CRUD (video, pdf, text)

Sprint 6: Course Management (Frontend)

- Course list page for admin

- Course builder UI

- Lesson editor

- Rich text editor for content

- SCSS + Mantine style integration

PHASE 4 — STUDENT FEATURES

Sprint 7: Learning Module Backend

- Enrollment system

- Watch tracking & progress tracking

- Course completion logic

- Certificate generation

Sprint 8: Student Dashboard Frontend

- My courses

- Continue learning

- Track progress

- Download certificates

PHASE 5 — QUIZ & ASSIGNMENTS

Sprint 9: Quiz System Backend

- Quiz CRUD

- Question bank

- Auto-grading

- Submission tracking

Sprint 10: Quiz System UI

- Instructor quiz creator UI

- Student quiz player

- Show results and analytics

PHASE 6 — PAYMENTS

Sprint 11: Payment Integration

- Razorpay/Stripe

- Create order

- Payment success callback

- Enrollment on success

Sprint 12: Payment UI

- Checkout screen

- Payment success/failure pages

PHASE 7 — ANALYTICS & ADMIN PANEL

Sprint 13: Analytics Backend

- Student progress stats

- Revenue analytics

- Course performance metrics

Sprint 14: Admin Panel UI

- User management

- Course management

- Sales dashboard

- Moderation tools

PHASE 8 — AI PHASE (PHASE 2 OF PROJECT)

Sprint A1: AI Course Recommendation

- ML-based suggestion engine

- Track learning behavior

Sprint A2: AI Chat Tutor (LLM)

- Student support chatbot

- Lesson-based Q&A;

Sprint A3: Auto Quiz Generation

- Generate quiz from course text/video transcript

Sprint A4: AI Assignment Feedback

- AI evaluates long descriptions

- Suggest improvements

PHASE 9 — DEPLOYMENT & MAINTENANCE

Sprint 15: Deployment

- Backend deploy (Render, Railway, ECS)

- Frontend deploy (Vercel)

- MongoDB Atlas setup

- CI/CD

Sprint 16: Load Testing & Security

- API rate limiting

- Mongo index optimization

- Vulnerability testing

This document provides the complete LMS blueprint.