

■ Project Requirement Document: New School – Modern Online Learning Platform

New School is a web-based educational platform designed to provide interactive, multimedia learning experiences for students around the world. It aggregates educational materials, online courses, and resources for different school categories — Elementary, Middle, and High School.

1. Objectives

- Provide students with an easy-to-navigate interface for accessing online educational content.
- Allow users to register, sign in, and start learning instantly.
- Offer categorized learning paths (Elementary, Middle, High School).
- Present an appealing and motivating design that encourages engagement.
- Integrate multimedia materials and teacher-student interaction capabilities.

2. Target Audience

- Students (ages 6–18)
- Parents (looking for online learning resources)
- Teachers (uploading or managing content)

3. Core Features

User Features:

- Landing Page with hero section (“Your Modern Online School”), CTA Buttons: “Start Today” and “Registration”.
- Category selection: Elementary, Middle, High School.
- “Sign In” button on top right.
- User Authentication with role-based login (Student / Teacher / Admin).
- Student Dashboard: Personalized courses, progress tracking.
- Course Pages: Video lessons, reading materials, quizzes.
- Teacher Dashboard: Upload video lectures and materials.
- Admin Panel: Manage users, teachers, content, and analytics.

4. Technical Requirements

Frontend:

- React.js / Next.js, Tailwind CSS or SCSS, Shadcn/UI, Framer Motion, Lucide Icons.
- Responsive for Mobile, Tablet, and Desktop.

Backend:

- Flask / Node.js / Spring Boot, MySQL / MongoDB / PostgreSQL.
- JWT-based authentication, RESTful APIs.

Integrations:

- Video hosting (YouTube API or AWS S3), Email notifications (SendGrid / SMTP), Admin Analytics Dashboard.

5. UI/UX Requirements

- Dark background with colorful capsules highlighting student photos.
- Rounded buttons, vibrant color palette (blue, purple, yellow).
- Typography: Modern sans-serif (Poppins, Inter).
- Responsive, animated, and accessible design (WCAG 2.1).

6. Non-Functional Requirements

- Performance: Optimized loading and caching.
- Security: Encrypted passwords, HTTPS, secure sessions.
- Scalability: Cloud-ready (AWS, Render, Railway).
- Accessibility: WCAG 2.1 compliant.

7. Future Enhancements

- AI-powered learning recommendations.
- Gamified learning with badges and leaderboards.
- Integration with Google Classroom or Microsoft Teams.
- Live class scheduling and video conferencing (Zoom API).

8. Deliverables

- Fully functional frontend and backend.
- Responsive UI per provided prototype.
- Admin, Teacher, and Student dashboards.
- Cloud deployment setup.