Software Requirements Specification (SRS)

# Online Education Platform

# 1. Introduction

## 1.1 Purpose

The purpose of this document is to specify the requirements for an online education platform. The platform will allow students to register, enroll in courses, attend lectures, complete quizzes, and interact with instructors. Instructors will be able to upload content, manage courses, and evaluate students’ progress.

## 1.2 Scope

The system will provide:  
- A web-based interface for students and instructors.  
- User authentication (login/register).  
- Course management (create, edit, delete).  
- Learning content delivery (videos, PDFs, assignments).  
- Assessment (quizzes, exams, grades).  
- Payment integration for premium courses.  
- Optional: Admin dashboard to manage users and courses.  
  
The system will not initially include mobile apps or advanced AI-driven recommendations (these may be added in future versions).

## 1.3 Definitions, Acronyms, Abbreviations

- SRS – Software Requirements Specification.  
- RESTful API – Representational State Transfer Application Programming Interface.  
- UI/UX: User Interface / User Experience.  
- JSON – JavaScript Object Notation.  
- JWT: JSON Web Token (used for authentication).

## 1.4 References

- IEEE SRS Template.  
- Existing platforms: Coursera, Udemy, Khan Academy (as benchmarks).

# 2. Overall Description

## 2.1 Product Perspective

- Frontend: Built using Angular for responsiveness and interactivity.  
- Backend: JSON-server for prototyping or Node.js/Express with MongoDB for scalability.  
- Communication: RESTful APIs.  
- Security: Authentication and authorization using JWT.

## 2.2 Product Functions

- Student Functions:  
 - Register and log in.  
 - Browse and enroll in courses.  
 - Access course materials (videos, notes, assignments).  
 - Take quizzes and view grades.  
 - Participate in forums/discussions.  
- Instructor Functions:  
 - Create and manage courses.  
 - Upload materials (videos, PDFs, links).  
 - Create and evaluate quizzes/assignments.  
 - Monitor student progress.  
- Admin Functions:  
 - Approve/reject instructors.  
 - Manage users and courses.  
 - Handle payments and reports.

## 2.2.1 Non-Functional Requirements

- Performance: Load course list within 3 seconds.  
- Usability: User-friendly UI and responsive design.  
- Security: JWT authentication and data encryption.  
- Availability: 99.5% uptime (when deployed).  
- Scalability: Support thousands of concurrent users.

## 2.3 User Classes and Characteristics

- Students – Register, browse courses, enroll, access lessons, and track progress.  
- Instructors – Create and manage courses, upload lessons, and review feedback.  
- Administrators – Manage platform settings, users, and course approvals (optional).

## 2.6 Operating Environment

- Angular 16+ (Frontend)  
- JSON-server / Node.js + Express + MongoDB (Backend)  
- Supported Browsers: Chrome, Firefox, Edge, Safari

## 2.5 Constraints

- The platform must support at least 1000 concurrent users.  
- Videos will be hosted externally (e.g., YouTube/Vimeo API).  
- Secure payment processing must be handled via PayPal/Stripe.

# 3. System Features

## 3.1 User Authentication

- Users can register/login with email and password.  
- Passwords must be hashed.  
- JWT for authentication.

## 3.2 Course Management

- Instructors can create/edit/delete courses.  
- Students can enroll in courses.  
- Course content includes videos, documents, and quizzes.

## 3.3 Learning Content

- Video streaming via YouTube/Vimeo API.  
- Downloadable resources (PDF, docs).  
- Assignments submission.

## 3.4 Assessments

- Quizzes (MCQ, True/False, short answers).  
- Auto-grading for objective questions.  
- Manual grading for assignments.

## 3.5 Communication

- Forum for each course.  
- Direct messaging between students and instructors.

## 3.6 Payment System

- Integration with PayPal/Stripe.  
- Students can buy premium courses.

# 4. User Interface Design

The platform will have the following pages:  
1. Home Page – Featured courses, search bar.  
2. Login & Register – User authentication forms.  
3. Courses List and path lines– Browse and filter courses.  
4. Course Details – Course description, enrollment option.  
5. Lesson Viewer – Watch video lessons and track progress.  
6-profile for students – My program , My assignments , Payment , My setting , log out.

## 4.1 Performance

- The system must handle 1000+ concurrent users.  
- Page load time < 3 seconds.

## 4.2 Security

- Encrypted passwords.  
- HTTPS support.  
- Role-based access control (student/instructor/admin).

## 4.3 Usability

- Responsive design for mobile/tablet.  
- Simple and intuitive UI.

## 4.4 Maintainability

- Modular architecture.  
- Documentation for developers.