E-Learning Management System SRS

1. Introduction

Purpose

The purpose of this document is to define the functional and non-functional requirements for the **E-Learning Platform**. This platform aims to provide an interactive and structured learning environment for students and instructors. The system is designed to be reusable, maintainable, reliable, and secure while ensuring ease of use for all users.

Intended Audience and Reading Suggestions

This document is intended for:

Role	Responsibility
Developers	Understanding system architecture and requirements.
Project Managers	Tracking deliverables and system functionality.
End Users (Students, Instructors, Admins)	Understanding system capabilities and usage.
QA Testers	Validating system functionality against defined requirements.

Project Scope

The **E-Learning Platform** aims to provide:

- An interactive learning environment for students.
- Course creation and management capabilities for instructors.
- User authentication and role-based access control.

- A structured progress tracking and reporting system.

References

- IEEE 830-1998 Software Requirements Specification Standard.
- API documentation and relevant system integration guidelines.

2. Overall Description

Product Features

Feature	Description
User Authentication	Secure login with role-based access for students, instructors, and admins.
Course Management	Create, update, delete courses, and manage learning materials.
Progress Tracking	Students can monitor their learning progress, while instructors can track student performance.
Quiz and Assessment	Instructors can create and grade quizzes. Students can attempt assessments and receive results.
Notifications	Automated email and in-app notifications for deadlines and updates.
Reporting & Analytics	Generate reports on student performance and system activity.
Multi-Device Support	Fully responsive design accessible on desktop and mobile devices.

User Classes and Characteristics

- **Admin**: Full system control, including user management and performance monitoring.
- **Instructor**: Course creation, student performance tracking, and assessment management.

- **Student**: Course enrollment, material access, progress tracking, and assessments.

Operating Environment

- **Web-based (Compatible with Windows, macOS, Linux, and Mobile devices)**
- **Database**: PostgreSQL for secure data storage.
- **Backend**: Node.js
- **Frontend**: React.js
- **Cloud & On-Premise Support**: The system will support cloud-based and offline functionalities.

3. Functional Requirements

Admin:

- Manage users
- Monitor system performance and resolve issues.
- Generate system-wide reports.

Instruction:

- Register and log in to the system.
- Create, update, and delete courses.
- Upload learning materials.
- Create and grade quizzes.
- View student progress and performance reports

Student Management:

- Register and log in to the system.
- Browse and enroll in courses.
- Access learning materials and complete quizzes.

- Track progress and view grades.

Course Management

- Instructors can create courses with titles, descriptions, and categories.
- Upload learning materials.
- Set prerequisites for courses.

Progress Tracking

- Students can view their progress in each course.
- Instructors can view student performance and generate reports.

Notifications

- Email and in-app notifications for course updates, and quiz deadlines

Reporting and Analytics

- Instructors can view course performance reports.
- Admins can generate system-wide reports.

4. Non-Functional Requirements

Reusability

- The system will be developed using an object-oriented programming language (e.g., Java, Python).
- Modular design with reusable components.

Maintainability

- Detailed documentation will be provided.
- Clean coding practices and consistent naming conventions.

- The system will allow instructors to update course content without developer intervention.

Reliability

- The system will have high operational availability.
- Error handling and logging mechanisms will ensure smooth operation.

Security

- All user data will be encrypted for security.
- Passwords will be securely stored.
- Only authorized users will have access to sensitive data.

Scalability

- The system will be designed to handle an increasing number of users and courses.

Usability

- The user interface will be easy to navigate.
- The system will be accessible on multiple devices.

5. Project Management

- **Jira** will be used for task tracking.
- Development updates and version control will be maintained in **GitHub**.

6. Conclusion

This document defines the requirements for the **E-Learning Platform**, ensuring that it meets the needs of students, instructors, and administrators. With a strong focus on security, usability, scalability, and maintainability, the system aims to provide an efficient and interactive online learning environment.