Software Requirements Specification (SRS)

This SRS document outlines the necessary specifications for the online course authoring and teaching web application. It serves as a comprehensive guide for the development team to build a robust and user- friendly system.

Online Course Authoring and Teaching Web Application

# Introduction

* 1. Purpose This document specifies the requirements for an online course authoring and teaching web application that allows instructors to create and manage courses, including uploading pre-recorded videos and creating pre-written quizzes. Students will be able to access these courses, view the videos, take quizzes, and proceed to the next sections based on their quiz results.
  2. Scope The web application will include functionalities for course creation, video management, quiz management, user management, progress tracking, and reporting. It will cater to instructors and students, providing a seamless and interactive learning experience.

1.3 Definitions, Acronyms, and Abbreviations SRS: Software Requirements Specification Admin: Administrator of the system Instructor: User who creates and manages courses Student: User who takes courses 1.4 References IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications Online Course Platforms: Coursera, Udemy, edX 1.5 Overview This document contains detailed descriptions of the system's functionalities, interfaces, and constraints. It is organized into several sections, including system features, external interface requirements, system requirements, and other non- functional requirements.

# Overall Description

* 1. Product Perspective The application will be a standalone web-based system, accessible via standard web browsers. It will integrate with third-party video hosting services and payment gateways.

2.2 Product Functions Course Management: Instructors can create, edit, and delete courses. Video Management: Instructors can upload and manage pre-recorded videos. Quiz Management: Instructors can create, edit, and delete quizzes. Student Progress Tracking: The system tracks student progress and quiz results. User Management: Admins can manage user accounts and roles. Access Control: The system controls access to course content based on quiz results. 2.3 User Classes and Characteristics Admins: Manage the overall system, user roles, and permissions. Instructors: Create and manage courses, videos, and quizzes. Students: Enroll in and take courses, view videos, and take quizzes. 2.4 Operating Environment Client Side: Any modern web browser (Chrome, Firefox, Safari, Edge) Server Side: Linux-based server with a web server (e.g., Apache or Nginx), a relational database (e.g., MySQL or PostgreSQL), and a backend framework (e.g., Django, Flask, or Node.js). 2.5 Design and Implementation Constraints Compliance with W3C standards: The application must adhere to web standards. Security: The system must ensure data protection and user privacy. 2.6 User Documentation User Manual: Detailed guide for using the system. Online Help: Context-sensitive help within the application. 2.7 Assumptions and Dependencies Reliable internet connection for accessing the web application. Third-party services for video hosting and payment processing are operational.

# System Features

* 1. Course Management 3.1.1 Description and Priority Instructors can create and manage courses. This is a high-priority feature.

3.1.2 Functional Requirements

FR1: Instructors can create new courses. FR2: Instructors can edit existing courses. FR3: Instructors can delete courses. FR4: Instructors can view a list of all their courses. 3.2 Video Management 3.2.1 Description and Priority Instructors can upload and manage pre-recorded videos. This is a high-priority feature.

3.2.2 Functional Requirements

FR1: Instructors can upload videos. FR2: Instructors can organize videos into course sections. FR3: Instructors can delete videos. FR4: Instructors can view a list of all uploaded videos. 3.3 Quiz Management

* + 1. Description and Priority Instructors can create and manage quizzes. This is a high-priority feature.
    2. Functional Requirements

FR1: Instructors can create quizzes with multiple choice, true/false, and short answer questions. FR2: Instructors can edit existing quizzes. FR3: Instructors can delete quizzes. FR4: Instructors can set passing criteria for quizzes. 3.4 Student Progress Tracking 3.4.1 Description and Priority The system tracks student progress and quiz results. This is a high-priority feature.

3.4.2 Functional Requirements

FR1: The system records student progress through course sections. FR2: The system tracks quiz results. FR3: The system allows students to view their progress and quiz results. 3.5 User Management 3.5.1 Description and Priority Admins can manage user accounts and roles. This is a medium-priority feature.

3.5.2 Functional Requirements

FR1: Admins can create, edit, and delete user accounts. FR2: Admins can assign roles to users (Admin, Instructor, Student). FR3: Admins can view a list of all users. 3.6 Access Control 3.6.1 Description and Priority The system controls access to course content based on quiz results. This is a high-priority feature.

3.6.2 Functional Requirements

FR1: The system restricts access to subsequent course sections until a student passes the current section's quiz. FR2: The system grants access to the next section upon passing the quiz.

# External Interface Requirements

* 1. User Interfaces UI1: Course creation and management interface for instructors. UI2: Video upload and management interface for instructors. UI3: Quiz creation and management interface for instructors. UI4: Course browsing and progress tracking interface for students. UI5: Admin dashboard for managing users and roles. 4.2 Hardware Interfaces Not applicable.

4.3 Software Interfaces SI1: Integration with video hosting services (e.g., YouTube, Vimeo). SI2: Integration with payment gateways (e.g., PayPal, Stripe). 4.4 Communications Interfaces CI1: HTTPS for secure communication between clients and the server.

# System Requirements

* 1. Functional Requirements Covered in section 3.
  2. Non-Functional Requirements 5.2.1 Performance Requirements

The system should support at least 1000 concurrent users. The system should load pages within 2 seconds. 5.2.2 Security Requirements

The system must protect user data with encryption. The system must have secure authentication and authorization mechanisms. 5.2.3 Usability Requirements

The system must be easy to navigate for all user roles. The system must provide a responsive design for use on various devices. 5.2.4 Reliability Requirements

The system should have an uptime of 99.9%. 5.2.5 Maintainability Requirements

The system should have modular code for ease of maintenance. The system should have comprehensive documentation for developers. 5.2.6 Portability Requirements

The system should be deployable on various web servers.

# Other Requirements

* 1. Legal and Regulatory Requirements The system must comply with data protection regulations (e.g., GDPR, CCPA). The system must comply with accessibility standards (e.g., WCAG 2.1). 6.2 Ethical Requirements The system should ensure fair use and prevent academic dishonesty.

# Appendices

* 1. Glossary Course: A set of instructional materials including videos and quizzes.

Quiz: A set of questions to assess student understanding.

Section: A subdivision of a course, typically containing a video and a quiz.