# Software Requirements Specification

**for**

# <Education System>

**Version <1.1>**

**Prepared by**


| | |
|---|---|
| **<202220150>** | **<أندرو فيكتور فهيم بشارة>** |
| **<202220299>** | **<فيرونيا كامل عدلي كامل>** |
| **<202220118>** | **<لوقا باسم وجدي حبيب>** |
| **<202220308>** | **<أسامة أسامة السيد ربيع>** |
| **<202220274>** | **<معاذ عمرو أحمد أحمد>** |
| **<202320042>** | **<كيرلس عادل مسعد جرس>** |

**Instructor:** **<Dr: Mohamed Reda>**

**Course:** **< Software Engineering >**

**Teaching Assistant:** ***< Eng : Mohamed Karam >***

**Date:** ***<14/11/2024>***

# Table of Contents



# Revision History

| Name | Date | Reason For Changes | Version |
|---|---|---|---|
| | | | |
| | | | |

# 1. Introduction

## 1.1 Purpose

The purpose of this SRS document is to provide a comprehensive overview of the requirements for the development of an Education Management System (EMS). This system will support educational institutions in managing student information, courses, assessments, and communications.

## 1.2 Scope

The EMS will allow administrators, teachers, students, and parents to manage educational activities, including course registration, attendance tracking, assignments, grades, and performance analytics. The system will be accessible via web and mobile platforms.

## 1.3 Definitions, Acronyms, and Abbreviations

- **EMS**: Education Management System
- **Admin**: Administrator responsible for system management
- **Student**: Learner enrolled in the institution
- **Teacher**: Educator who creates, manages, and grades coursework
- **Parent**: Guardian who can view student performance and attendance

## 1.4 Overview

This document will detail the functional and non-functional requirements of the EMS.

# 2. Overall Description

## 2.1 Product Perspective

The EMS is a standalone system that integrates with other educational tools, such as Learning Management Systems (LMS), communication tools, and payment gateways.

## 2.2 Product Functions

- **User Management**: Role-based access for admins, teachers, students, and parents.
- **Course Management**: Create and manage courses, enroll students, and assign teachers.
- **Attendance Management**: Track and report attendance.
- **Assessment Management**: Create quizzes, assignments, and exams; automate grading.
- **Communication Module**: Facilitate announcements, messaging, and notifications.
- **Reporting & Analytics**: Generate performance, attendance, and course progress reports.

## 2.3 User Classes and Characteristics

- **Admin**: Manages system settings, user roles, and access permissions.
- **Teacher**: Manages courses, uploads study materials, tracks student performance.
- **Student**: Accesses course materials, submits assignments, views grades.
- **Parent**: Monitors student attendance, assignments, and prog

## 2.4 Operating Environment

- Web and mobile platforms.
- Supports major browsers (Chrome, Firefox, Safari).
- Compatible with Android and iOS mobile devices.

## 2.5 Design and Implementation Constraints

- Compliance with data privacy laws (e.g., GDPR).
- System should support up to 100,000 simultaneous users.
- Regular backups to prevent data loss.

## 2.6 Assumptions and Dependencies

- Users have internet access.
- Institution provides necessary digital infrastructure.

# 3. Specific Requirements

## 3.1 Functional Requirements

### 3.1.1 User Management

- **User Registration**: The system must allow new users to register with appropriate roles.
- **Role-Based Access Control**: Different roles (Admin, Teacher, Student, Parent) have access to specific modules.
- **Authentication & Authorization**: Secure login and role-based access.

### 3.1.2 Course Management

- **Course Creation**: Admins and teachers can create and configure courses with titles, descriptions, schedules, and prerequisites.
- **Enrollment Management**: Students can enroll in courses, and admins can manage enrollments.
- **Content Management**: Teachers can upload study materials, such as documents, videos, and presentations.

### 3.1.3 Attendance Management

- **Automated Attendance Tracking**: Record attendance using digital check-ins.
- **Attendance Reports**: Generate daily, weekly, and monthly attendance summaries for teachers and admins.

### 3.1.4 Assessment Management

- **Assignment & Quiz Creation**: Teachers can create assignments, quizzes, and exams.
- **Grading System**: Automated grading for quizzes and manual grading options for assignments.
- **Feedback Mechanism**: Teachers can provide feedback on student submissions.

### 3.1.5 Communication Module

- **Announcements**: Admins and teachers can post announcements visible to relevant users.
- **Messaging**: Secure messaging between teachers, students, and parents.
- **Notifications**: Real-time notifications for assignments, announcements, and other updates.

### 3.1.6 Reporting & Analytics

- **Performance Reports**: Generate reports on student performance in assessments.
- **Attendance Reports**: View and download attendance data.
- **Course Progress Tracking**: Track course completion percentages for individual students.

## 3.2 Non-Functional Requirements

### 3.2.1 Performance Requirements

- **Scalability**: System should handle up to 100,000 users.
- **Response Time**: Average page load time should not exceed 2 seconds.

### 3.2.2 Security Requirements

- **Data Encryption**: Encrypt sensitive data, such as passwords and personal information.
- **Backup and Recovery**: Regular backups and disaster recovery plan to prevent data loss.

### 3.2.3 Usability Requirements

- **User Interface**: Clean, intuitive design for easy navigation.
- **Accessibility**: Ensure the system complies with accessibility standards (WCAG 2.1).

### 3.2.4 Reliability Requirements

- **Uptime**: System should maintain 99.9% uptime.
- **Error Handling**: System should handle errors gracefully and provide informative messages.

### 3.2.5 Maintainability Requirements

- **Code Documentation**: Ensure all code is well-documented.
- -**Modular Design**: Use modular architecture for easy updates and maintenance.

# 4. External Interface Requirements

## 4.1 User Interfaces

- **Web Interface**: Accessible from major browsers.
- **Mobile Interface**: Responsive and optimized for iOS and Android devices.

## 4.2 Hardware Interfaces

- Integrates with institution's existing digital infrastructure for attendance tracking (if using biometric or card-based attendance).

## 4.3 Software Interfaces

- **Integration with LMS**: Should support integration with popular LMS like Moodle or Google Classroom.

- **Payment Gateway Integration**: To facilitate fee payments for students (Stripe, PayPal).

## 4.4 Communication Interfaces

- **Email and SMS API Integration**: For notifications and alerts to students and parents.

# 5. Other Requirements

## 5.1 Legal and Regulatory Requirements

Compliance with

- **GDPR** (General Data Protection Regulation).
- **COPPA** (Children’s Online Privacy Protection Act) compliance if the platform is used by minors.

## 5.2 Future Enhancements

- AI-driven analytics to provide insights into student learning patterns.

- Integration with external content providers for additional learning resources.