Student ERP System Software Requirements Specification

1. Introduction

1.1 Purpose

The purpose of this document is to define the functional and non-functional requirements of the Student ERP system, which is designed to streamline administrative and academic processes within educational institutions.

1.2 Scope

The Student ERP system will cover various aspects of student and administrative management, including student records, course management, attendance tracking, grade management, communication, and reporting.

1.3 Document Conventions

Use "Must" to indicate mandatory requirements.

Use "Should" to denote preferred or desirable features.

Use "May" for optional or future enhancements.

1.4 Intended Audience

The intended audience for this document includes developers, designers, project managers, quality assurance teams, and all stakeholders involved in the development and deployment of the Student ERP system.

2. System Overview

2.1 System Description

The Student ERP system is a comprehensive web-based application that serves educational institutions. It offers a centralized platform for managing student-related information, faculty interactions, and administrative tasks.

2.2 System Architecture

The system will be built using a three-tier architecture, with a front-end developed using Flutter, a back-end using [Your Preferred Technology], and a database system using [Database Management System].

3. Functional Requirements

3.1 User Authentication

3.1.1 Must

Users must be able to register for accounts with unique usernames and passwords.

Users must authenticate using their credentials.

3.2 User Management

3.2.1 Must

Administrators must be able to manage user roles (e.g., student, faculty, administrator).

Users should be able to update their profiles.

3.3 Student Information Management

3.3.1 Must

Users must be able to add, edit, and delete student records.

Student records should include personal information, contact details, and enrollment details.

3.4 Course Management

3.4.1 Must

Administrators should be able to create, edit, and delete courses.

Users must be able to view course details and enroll in courses.

3.5 Attendance Tracking

3.5.1 Must

Faculty members should be able to record student attendance.

Students must be able to view their attendance records.

3.6 Grade Management

3.6.1 Must

Faculty members should be able to record and update grades for students.

Students must be able to view their grades.

3.7 Communication

3.7.1 Must

Users must have the ability to send and receive messages within the system.

Notifications must be sent for important events (e.g., grade updates, course announcements).

3.8 Reporting

3.8.1 Must

The system should generate and provide access to various reports (e.g., student transcripts, attendance reports) for authorized users.

4. Non-Functional Requirements

4.1 Performance

4.1.1 Must

The system must handle a minimum of [number] concurrent users.

Response times for critical functions should be within [time limit].

4.2 Security

4.2.1 Must

User data and passwords must be securely stored and transmitted.

Access control mechanisms must be implemented to ensure data privacy.

4.3 Usability

4.3.1 Must

The user interface should be intuitive and user-friendly.

Usability testing should be conducted with actual users.

4.4 Reliability

4.4.1 Must

The system should have an uptime of at least [uptime percentage].

Regular backups and data recovery procedures must be in place.

4.5 Compatibility

4.5.1 Must

The system should be compatible with modern web browsers (e.g., Chrome, Firefox, Safari).

Mobile responsiveness is required for access on various devices.

5. Constraints

The project budget is limited to [budget amount].

Development must be completed within [timeline].

6. Assumptions

Users have basic computer literacy.

The system will be hosted on [hosting provider].

7. Dependencies

The system will integrate with [external systems or APIs].

8. Appendices

Include any supplementary materials, such as wireframes, diagrams, or mockups.

9. Glossary

Define any technical terms or acronyms used in the document.

10. Revision History

Maintain a record of document revisions and updates.

This SRS document provides a structured framework for defining the requirements of a Student ERP system. Be sure to collaborate closely with stakeholders to gather and refine requirements throughout the project's lifecycle.