

# **Software Requirement Specification (SRS) - Student Information System (SIS)**

## **1. Introduction:**

### **1.1 Purpose:**

The purpose of this document is to outline the software requirements for the development of a Student Information System (SIS) for ABC school. The SIS will serve as a comprehensive platform for managing student information, academic performance, communication, and administrative processes.

### **1.2 Scope:**

The SIS will include modules for student login, academic performance tracking, leave management, timetable display, notification center, student profile management, contact class teacher, and fees and payments. The system will be developed using AngularJS for the frontend, Python Flask for the backend, and MS SQL Server for the database.

### **1.3 High-Level Architecture:**

The high-level architecture will consist of a three-tier architecture:

- **Presentation Layer:**
  - AngularJS for the user interface, ensuring a responsive and dynamic user experience.
- **Application Layer:**
  - Python Flask framework for server-side logic, handling data requests and responses.
- **Data Layer:**
  - MS SQL Server for secure, scalable, and relational data storage.

## **2. Modules:**

### **2.1 Student Login:**

#### **2.1.1 Functional Requirements:**

- Secure authentication using unique credentials.
- Password recovery mechanism with multi-factor authentication.
- Session management for secure user interactions.
- Password policies implementation (complexity, expiration).

#### **2.1.2 Technical Requirements:**

- Implementation of secure token-based authentication.
- Integration with password recovery services.

- Session handling using secure tokens or cookies.

## **2.2 Academic Performance:**

### 2.2.1 Functional Requirements:

- Display of subject-wise marks and grades for each test.
- Historical performance analysis with graphical representation.
- Download/print academic reports.
- Notification system for upcoming assessments and results.

### 2.2.2 Technical Requirements:

- Data retrieval from the database based on student ID and assessment parameters.
- Integration with charting libraries for graphical representation.
- Real-time notification integration with the notification center.

## **2.3 Leave Management:**

### 2.3.1 Functional Requirements:

- Leave request submission form with reason, dates, and documents.
- Real-time notifications to parents and teachers.
- Approval and rejection options with comments.
- Calendar view displaying approved leave dates.

### 2.3.2 Technical Requirements:

- Database storage and retrieval of leave requests.
- Real-time notification integration with the notification center.
- Calendar integration for displaying leave dates.

## **2.4 Time Table:**

### 2.4.1 Functional Requirements:

- Dynamic display of the weekly class schedule.
- Ability for administrators to update and modify the timetable.
- Overview of daily/weekly schedules.
- Integration with other modules to avoid conflicts.

### 2.4.2 Technical Requirements:

- Database storage and retrieval of class schedules.
- Admin interface for timetable updates.
- Integration with other modules through API endpoints.

## **2.5 Notification Center:**

### **2.5.1 Functional Requirements:**

- Centralized dashboard for sending notifications.
- Categorization of notifications.
- Delivery of real-time notifications to parents.
- Customization of notification preferences.

### **2.5.2 Technical Requirements:**

- Database storage and retrieval of notifications.
- Real-time delivery using WebSocket or polling.
- User-specific preferences stored in the database.

## **2.6 Student Profile:**

### **2.6.1 Functional Requirements:**

- Detailed personal information.
- Academic history and extracurricular activities.
- Uploading and updating of student photos.
- Privacy settings for information visibility.

### **2.6.2 Technical Requirements:**

- Secure storage and retrieval of student profile data.
- Image upload and storage capabilities.
- Privacy settings stored in the database.

## **2.7 Contact Class Teacher:**

### **2.7.1 Functional Requirements:**

- Direct messaging system between parents and teachers.
- Scheduling system for virtual or physical meetings.
- Notification system for new messages and meeting requests.
- Archive or history of communication.

### **2.7.2 Technical Requirements:**

- Secure messaging system with message encryption.
- Calendar integration for meeting scheduling.
- Database storage and retrieval of messages and meeting requests.

## **2.8 Fees and Payments:**

### **2.8.1 Functional Requirements:**

- Display of fees, outstanding payments, and payment history.
- Secure payment gateway integration.
- Generation of digital receipts.
- Automated alerts for upcoming fee due dates.

### **2.8.2 Technical Requirements:**

- Integration with a secure payment gateway API.

- Database storage and retrieval of fee-related data.
- Automation of fee alerts based on due dates.

### **3. Software Tools Requirements:**

#### **3.1 Frontend Development:**

- AngularJS for a responsive and dynamic user interface.
- Integration of UI components for an intuitive user experience.

#### **3.2 Backend Development:**

- Python Flask framework for server-side logic.
- Efficient handling of data requests and responses.

#### **3.3 Database Management:**

- MS SQL Server for secure, scalable, and relational data storage.
- Implementation of database triggers for data integrity.

#### **3.4 Security:**

- SSL/TLS encryption for secure data transmission.
- Implementation of secure coding practices to prevent vulnerabilities.
- Regular security audits and updates.

#### **3.5 Notification Services:**

- Integration with notification services for real-time alerts and communication.

#### **3.6 Payment Gateway:**

- Integration with a reputable and secure payment gateway for fee transactions.

#### **3.7 Version Control:**

- Git for version control, allowing for collaboration and code management.

#### **3.8 Development Environment:**

- Code editors (e.g., Visual Studio Code), development servers, and debugging tools.

This Software Requirement Specification (SRS) document provides a comprehensive outline of the functional and technical requirements for the development of the Student Information System. It serves as a guide for the development team to ensure the successful implementation of the system as per the school's needs and expectations. Continuous collaboration with the school administration is recommended to address any emerging requirements or adjustments during the development process.

•