SOFTWARE REQUIREMENT SPECIFICATION

**1. Introduction**

This project is a web-based application designed to help educational institutions track student academic performance and attendance. It allows teachers to upload grades, mark attendance, and generate reports. Students can view their own performance, and administrators can manage users and the overall system.

**2. Purpose**

The purpose of this software is to:

* Monitor and evaluate student performance easily.
* Provide role-based access for admins, teachers, and students.
* Automate report generation for exams and attendance.
* Alert students or teachers when performance is low.

**3. Scope**

This system will be accessible online via web browsers. It will:

* Allow teachers to mark attendance and upload grades.
* Allow students to log in and view their marks and attendance.
* Allow admins to manage teachers, students, and system data.
* Generate performance reports automatically.
* Be used by schools, colleges, and other educational institutions.

**4. System Overview**

The system includes:

* **Login system** for three types of users: admin, teacher, and student.
* **Admin dashboard** to manage users and generate reports.
* **Teacher dashboard** to manage students, enter marks, and mark attendance.
* **Student dashboard** to view personal performance and attendance.
* **Database** to store all records of users, attendance, and marks.

**5. Functional Requirements**

1. **User Login**: All users must log in to access the system.
2. **Admin Functions**:
   * Add/edit/delete student and teacher accounts.
   * View overall student reports.
3. **Teacher Functions**:
   * Mark attendance.
   * Enter marks for tests, assignments, and exams.
   * View and generate reports.
4. **Student Functions**:
   * View marks and attendance.
   * Download performance reports.
5. **Report Generation**: System should generate downloadable PDF reports.

**6. Non-Functional Requirements**

1. **Usability**: Simple and user-friendly interface.
2. **Performance**: Fast loading and quick response even with many users.
3. **Security**: Password protection and role-based access.
4. **Availability**: System should be available 24/7 without crashes.

**7. Design Constraints**

* **Frontend**: A responsive web interface using **React.js** that allows users (admin, teacher, student) to log in, view dashboards, enter/view performance and attendance data.
* **Backend**: A secure REST API built with **Spring Boot** (or Node.js) that handles authentication, data storage, and logic for student performance, grades, and attendance management.

**Conclusion**

The Student Performance Tracker web application provides a simple, efficient, and user-friendly solution for managing academic records. With a React-based frontend and a robust backend, it ensures smooth interaction, secure data handling, and real-time access to student performance and attendance for all users.