**Professional Document: School Management System Project**

**1. Introduction**

This document outlines the features and technical aspects of a School Management System developed using the MERN stack. The system caters to the needs of administrators, teachers, and students, providing functionalities such as user authentication, class management, attendance tracking, grade assignment, and notice distribution.

**2. Features**

**User Roles:**

* **Admin**: The main administrator with the highest level of access.
* **Teacher**: Faculty members responsible for managing classes, taking attendance, and assigning grades.
* **Student**: Enrolled individuals who can view their attendance, grades, and notices.

**Functionalities:**

1. **User Authentication**:
   * Admin credentials required for signing up new teachers and students.
   * Encrypted password storage using bcrypt.js for enhanced security.
2. **Class Management**:
   * Admin can create classes and assign teachers to specific classes.
   * Teachers can view the classes they're assigned to.
3. **Attendance Management**:
   * Admin and teachers can take attendance for students.
   * Students can view their attendance records.
4. **Grade Assignment**:
   * Admin can assign marks to students for subjects they are enrolled in.
   * Teachers can also assign grades to students.
5. **Notice Distribution**:
   * Admin can create notices visible to both teachers and students.

**3. Frontend Technologies**

**React:**

A JavaScript library for building user interfaces, allowing for efficient component-based development.

**React Router DOM:**

Enables navigation and routing in a React application, facilitating multi-page functionality.

**Material-UI (MUI):**

A React UI framework offering pre-designed components and styles for a cohesive and responsive design.

**Axios:**

A promise-based HTTP client for making asynchronous requests to the backend server.

**ReactCharts:**

Provides reusable chart components for visualizing data in the application.

**Redux & Redux Toolkit:**

State management libraries for managing application state across components and facilitating predictable state updates.

**Styled-components:**

Enables component-level styling using tagged template literals in JavaScript.

**4. Backend Technologies**

**bcrypt.js:**

A library for hashing passwords, enhancing security by encrypting user credentials.

**Mongoose:**

An Object Data Modeling (ODM) library for MongoDB and Node.js, simplifying interactions with the MongoDB database.

**Express.js:**

A web application framework for Node.js, providing robust features for building APIs and handling HTTP requests.

**Nodemon:**

A utility that monitors changes in the backend code and automatically restarts the server during development.

**dotenv:**

A module for loading environment variables from a .env file, enhancing security and configuration management.

**MongoDB:**

A NoSQL database for storing and retrieving data, chosen for its flexibility and scalability in handling school management system requirements.