System Overview and Architecture

# School Management Information System of Lumban National High School

The School Management Information System (SMIS) of Lumban National High School is a comprehensive digital platform designed to streamline school operations and enhance administrative efficiency. The system offers functionalities such as student information management, faculty records, class schedules, and performance tracking, making it a vital tool for modern educational administration.

A significant feature of this system is its modular design, ensuring scalability and adaptability. It integrates various school processes into one cohesive platform, reducing manual workload and enabling better data-driven decision-making.

## System Architecture

The system architecture follows a modular and layered approach, combining a web-based frontend, cloud-hosted backend, and integrated analytics:

1. 1. Frontend (React + Bootstrap)

a) Built with React.js and styled with Bootstrap, the frontend ensures a user-friendly and responsive interface for staff and administrators.

b) Handles dashboard views, user authentication, student and faculty record entry, class and grade management, and analytics display.

1. 2. Backend (Node.js + Firebase)

a) Authentication: Firebase Authentication manages secure login for users.

b) Database: Firestore stores student records, grades, schedules, and logs.

c) Hosting and Storage: Firebase Hosting deploys the web app, while Firebase Storage manages uploaded documents (e.g., grade sheets, IDs).

1. 3. Data Analytics Module (JavaScript + Chart.js)

a) Uses historical student performance data to generate insights into academic trends.

b) Displays data visualization (bar graphs, pie charts, etc.) using Chart.js to assist in administrative reporting.

1. 4. Role-Based Access Control

a) Admins can manage all data.

b) Teachers can update grades, attendance, and schedules.

c) Students and parents can view performance and announcements.

## Component Description

|  |  |
| --- | --- |
| Component | Description |
| Input | Student records, grades, class schedules, teacher assignments, attendance logs |
| Process | Data validation, access control, analytics generation, and file storage management |
| Output | Reports (grades, attendance), visual graphs, student performance summaries, and user dashboards |

## Summary of Enhancements and Rationale

To improve school operations, the SMIS of Lumban National High School has introduced key enhancements:

* a. Role-based dashboard views for Admin, Teacher, and Student.
* b. Automated grade computation and report card generation.
* c. Graphical performance reports to identify at-risk students.
* d. Cloud-based data storage ensuring accessibility and backup.
* e. Simplified schedule management with conflict detection.

Rationale:

1. 1. Improved Administrative Efficiency – Automating core school functions saves time and ensures data consistency.
2. 2. Enhanced Data Accuracy – Centralizing data entry and retrieval reduces human error.
3. 3. Data-Driven Insights – Visual analytics help in early intervention for underperforming students.
4. 4. Scalability – Modular architecture allows future additions like online enrollment or learning modules.
5. 5. Accessibility and Transparency – Teachers, students, and parents gain access to real-time performance data.

## Testing Approach and Results

|  |  |  |  |
| --- | --- | --- | --- |
| Test Type | Description | Applied To | Result |
| Unit Testing | Tested individual logic modules | Grade calculator, login auth | All units returned expected results |
| Integration Testing | Verified module interoperability | Login → Dashboard → Student Profile | Smooth navigation, no data loss |
| Functional Testing | Ensured all features work end-to-end | Class creation, grade reports | Features performed as expected |
| Performance Testing | Measured app speed under load | Bulk uploads, dashboard load | Acceptable performance, no crashes |

## Technologies and Frameworks Used

|  |  |
| --- | --- |
| Technology | Description |
| React.js | Used for building the interactive user interface |
| Bootstrap | UI components and responsive layout |
| Node.js | Backend scripting and logic handling |
| Firebase Authentication | User identity management |
| Firestore | Cloud-hosted NoSQL database |
| Firebase Hosting | For web deployment |
| Firebase Storage | For document uploads and storage |
| Chart.js | To render performance charts and graphs |

## Developer Notes / Installation Instructions

1. 1. Install Node.js and Firebase CLI.
2. 2. Clone the SMIS repository from GitHub.
3. 3. Navigate to the project directory using terminal:  
    cd lumban-smis
4. 4. Install dependencies:  
    npm install
5. 5. Start the development server:  
    npm start
6. 6. Use Firebase login to connect to your project and deploy:  
    firebase login  
    firebase init  
    firebase deploy
7. 7. Open the hosted URL in a browser to use the application.