

PREMIER UNIVERSITY, CHATTOGRAM

Department of Computer Science & Engineering



Project proposal for School Management System.

SUBMITTED BY

Name: Md.Ariful Islam

ID: 2104010202249

Name: Tinni Banik

ID: 2104010202260

Name: Saira Jahan

ID: 1903710202050

Name: Zarin Tasnima Azad

ID: 2104010202293

In partial fulfillment for the degree of
Bachelor of Science in Computer Science & Engineering under the

Supervision of

Tahsin Hossain

Lecturer,

Department of Computer Science & Engineering

Premier University, Chattogram

24 October 2024

School Management System

1. Abstract:

This project outlines the development of School Management System, A School Management System is a comprehensive software solution designed to automate and streamline the administrative and academic processes within educational institutions. The system serves as a centralized platform that integrates various functions such as student enrollment, attendance tracking, student management, class scheduling, and communication between teachers, students, and parents. a user-centric academic platform catering to the growing demand for accessible acquisition of the system. The School Management System is an integrated platform developed to enhance the efficiency and effectiveness of school administration. It addresses the diverse needs of educational institutions by automating routine tasks, facilitating communication, and providing real-time access to information.

2. Introduction:

A School Management System using Laravel, MySQL, and XAMPP is a comprehensive web-based solution designed to simplify the administration and management of school operations. Laravel, a robust PHP framework, offers efficient routing, authentication, and easy-to-manage MVC architecture, making it ideal for developing scalable applications. MySQL is used as the database management system to store student records, staff information, attendance, grades, and other essential data securely. XAMPP serves as a local development environment, providing an integrated platform with Apache and MySQL to test and deploy the application. The system allows administrators to manage student enrollment, staff details, and academic records, while teachers can access class schedules and grade submissions. Parents and students can view progress reports, attendance, and announcements through a user-friendly interface. With this system, schools can automate daily tasks, reduce paperwork, and improve communication among staff, students, and parents.

3. Project Organization:

he project was a collaborative effort, involving the following team members:

1. Md. Ariful Islam:

- Focus on developing the **Admin** features:
 - Admin authentication, role management, and permissions.
 - CRUD operations for managing student, teacher, and parent profiles.
 - Report generation for attendance, performance, and financial records.
 - Database design and structure with MySQL.

2. **Tinni Banik:**

- Focus on developing the **Teacher** and **Parent** features:
 - Teacher authentication, profile management, and class schedules.
 - Attendance submission and grade management.
 - Parent dashboard, including viewing student performance and announcements.
 - Notification system for parent-teacher communication.

Frontend Developers

3. **Saira Jahan:**

- Focus on designing and implementing the **Student** user interface:
 - Student dashboard and login pages.
 - Displaying class schedules, assignments, and grades.
 - Responsive design for student-related views.
 - Integration with backend API for fetching student data.

4. **Zarin Tasnima Azad:**

- Focus on designing and implementing the **Admin** and **Parent** user interfaces:
 - Admin dashboard for managing users, settings, and reports.
 - User-friendly parent dashboard, including announcements and student progress.
 - Responsive design for all views and ensuring cross-browser compatibility.
 - Integration with backend API for admin and parent functionalities.

The team utilized various tools and methodologies to ensure smooth collaboration and the successful completion of the project. Communication was maintained through daily stand-ups and weekly team reviews to track progress and resolve any development challenges.

Key Features:

☐ **User Management (Admin, Teachers, Students, Parents):**

- Role-based access control for different user types.
- Profile management for students, teachers, parents, and administrators.
- Secure login and authentication system.

☐ **Attendance Management:**

- Digital attendance tracking for students and staff.
- Automated notifications to parents for student absenteeism.
- Attendance reports generation for class-wise and school-wide analysis.

☐ **Academic Management:**

- Subject and curriculum management for various classes.
- Timetable scheduling and management for classes and exams.
- Assignment and homework management for teachers and students.

☐ **Grade and Exam Management:**

- Gradebook management, allowing teachers to record student grades.
- Report card generation with performance analysis.
- Online examination and result publishing.

☐ **Student Admission and Enrollment:**

- Online admission forms and registration processes.
- Student database management with detailed records.
- Fee collection, payment tracking, and receipt generation.

☐ **Financial Management:**

- Fee management with online payment options.
- Payroll management for staff salaries.
- Financial reporting and budgeting tools.

❑ **Event and Notice Management:**

- Calendar integration for school events, holidays, and activities.
- Notice board for announcements and circulars.
- Event registration and tracking for students and staff.

4. Hardware and Software Requirements:

❑ **Server-side Software:**

- **Web Server:** Apache (XAMPP for local development includes Apache).
- **Database:** MySQL for data management.
- **Programming Language:** PHP (required for Laravel).
- **Framework:** Laravel (version 11).
- **Backup Software:** Tools like Acronis or custom backup scripts for regular database backups.

❑ **Client-side Software:**

- **Web Browser:** Latest versions of Chrome, Firefox, Edge, or Safari.
- **PDF Reader:** For viewing reports and downloadable resources.
- **Antivirus Software:** To secure client machines.

❑ **Development Environment:**

- **XAMPP:** For local development and testing (includes Apache, PHP, and MySQL).
- **IDE/Code Editor:** Visual Studio Code
- **Version Control:** Git for source code management and collaboration (GitHub, GitLab, or Bitbucket).

5. System Architecture:

The **MVC (Model-View-Controller)** architecture for a **School Management System** using Laravel involves three main components: the **Model**, **View**, and **Controller**. Each of these components has a specific role in handling data, user interactions, and presentation. Here's how the MVC architecture can be applied to a **School Management System**:

1. Model:

- **Role:** Handles the database interactions, business logic, and data management.
- **Examples in the School Management System:**
 - **Student Model:** Manages data related to students, such as enrollment details, personal information, grades, and attendance records.
 - **Teacher Model:** Manages teacher profiles, class schedules, subjects taught, and grading information.
 - **Admin Model:** Handles data related to user roles, permissions, and system settings.
 - **Parent Model:** Stores information about parents and their association with students.
 - **Attendance Model:** Tracks attendance records for students and teachers.
 - **Grade Model:** Manages grades and examination results for students.
 - **Fee Model:** Manages fee structures, payments, and transactions.
- These models interact with the database using Laravel's **Eloquent ORM** to perform operations like **CRUD** (Create, Read, Update, Delete).

2. View:

- **Role:** Responsible for displaying the data to users. It handles the presentation layer.
- **Examples in the School Management System:**
 - **Student Dashboard:** Displays class schedules, grades, assignments, and attendance reports.
 - **Teacher Dashboard:** Shows class schedules, student lists, and assignment submission status.
 - **Admin Dashboard:** Provides views for managing students, teachers, parents, and generating reports.
 - **Parent Portal:** Allows parents to view their child's progress, attendance, and school announcements.
 - **Templates:** HTML pages with embedded **Blade templates** (Laravel's templating engine) for dynamic content rendering.
- Views are typically designed using Laravel's **Blade templates**, enabling dynamic data display and easier layout management.

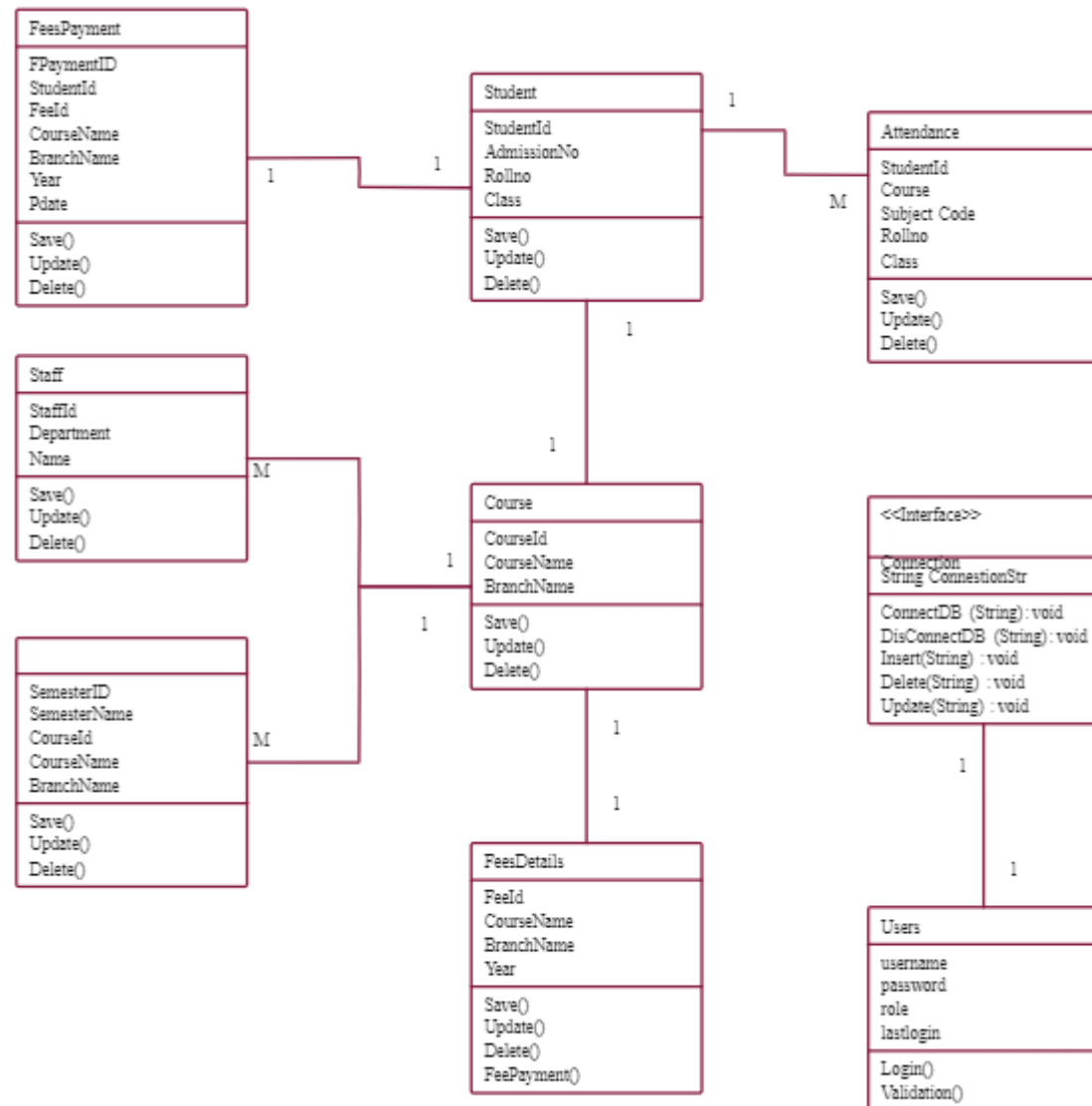
3. Controller:

- **Role:** Acts as an intermediary between Models and Views. It processes user inputs, communicates with Models, and returns the appropriate Views.
- **Examples in the School Management System:**
 - **StudentController:** Handles student-related actions such as registration, profile updates, and retrieving grades or attendance data.
 - **TeacherController:** Manages actions like class scheduling, grade submission, and viewing student performance.
 - **AdminController:** Oversees user management, permissions, report generation, and system configuration.
 - **ParentController:** Allows parents to log in, view their child's information, and communicate with teachers.
 - **AttendanceController:** Manages marking attendance, viewing attendance records, and generating attendance reports.
 - **FeeController:** Handles fee payment processes, transaction records, and receipt generation.

Flow of MVC in the School Management System:

1. **User Request:** A user (e.g., a teacher) makes a request through the web interface (e.g., view a student's attendance).
2. **Controller Processing:** The request is handled by a specific controller (e.g., AttendanceController) which processes the input.
3. **Model Interaction:** The controller interacts with the relevant model (e.g., Attendance Model) to retrieve or modify data in the database.
4. **Data Retrieval:** The model retrieves the necessary data from the database using **Eloquent** and returns it to the controller.
5. **View Rendering:** The controller then selects the appropriate view (e.g., attendance.blade.php) and sends the data to be displayed.
6. **Response to User:** The view renders the data and returns a response (e.g., a list of attendance records) to the user's browser.

6. Class Diagram of the System:



7. Project Demonstration:

☐ Use Case 1: Admin Management

- **Scenario:** An admin adds a new student.
 - Navigate to the "Manage Students" section.
 - Fill in the student's details and save.
 - Show the updated student list.

☐ Use Case 2: Teacher Functionality

- **Scenario:** A teacher marks attendance.
 - Navigate to the class roster for the day.
 - Select students present and absent.
 - Save the attendance record and show confirmation.

☐ Use Case 3: Student Enrollment

- **Scenario:** A student enrolls in a new subject.
 - Log in as a student.
 - Go to the "Enroll in Subjects" section.
 - Select the desired subjects and submit.

☐ Use Case 4: Parent Monitoring

- **Scenario:** A parent views their child's grades.
 - Log in as a parent.
 - Navigate to the "Child's Progress" section.
 - View grades and attendance records.

8. Testing

□ Admin Functions

- Test if createUser() adds a user correctly with valid data.
- Test if manageUsers() updates user details accurately.
- Test for error handling when trying to create a duplicate user.

□ Student Functions

- Test if viewAttendance() correctly retrieves attendance records for a student.
- Test if enrollInSubject() successfully adds a student to a subject.

□ Teacher Functions

- Test if assignGrades() assigns grades correctly to students.
- Test if takeAttendance() records attendance properly for a class.

9. Result and Output

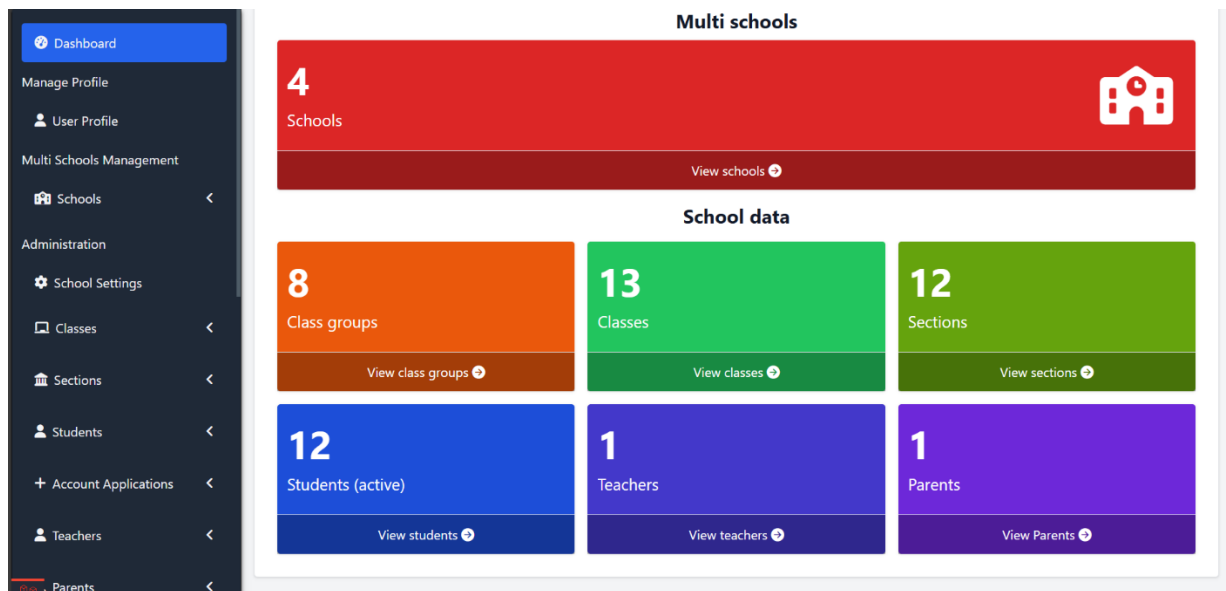


Figure-1

Dashboard

Manage Profile

User Profile

Multi Schools Management

Schools

Administration

School Settings

Classes

Sections

Students

Account Applications

Teachers

Parents

Admins

Profile Information

Update your account's profile information and email address.

All fields marked * are required

A

New Photo

Name

Admin

Email

super@admin.com

BirthDay *

04/04/2000

Gender *

Male

Nationality *

Nigeria

State *

Abia

Figure-2

Dashboard

Manage Profile

User Profile

Administration

Fees

Academics

Notices

Syllabi

Timetables

Exams

Grade Systems

Dashboard

Dashboard

Download Profile

Notices

Search

10

S/N	Title	Action
1	Accusantium itaque labore tempora expedita accusantium.	Actions

Figure-3

School-Management

Dashboard

Manage Profile

User Profile

Multi Schools Management

Schools

Administration

School Settings

Classes

Sections

Students

Account Applications

Teachers

View teachers

Create teacher

Teachers

You are currently on Default School - School of Computing

Dashboard > Teachers

Teachers List

Search 10

S/N	Name	Email	Gender	Locked	Actions	Delete
1	John Doe	teacher@teacher.com	male	<input type="checkbox"/> Unlocked	<div>Actions</div>	<div>Delete</div>

Figure-4



Email

Password

☐ Remember Me

Log in

Dont Have An account? [Create Account](#)

[Forgot your Password?](#)

Figure-5

School-Management

Dashboard

Manage Profile

User Profile

Multi Schools Management

Schools

Administration

School Settings

Classes

Sections

Students

View students

Create student

Promote students

Manage promotions

Graduate students

Students

You are currently on Default School - School of Computing

Dashboard > Students

Students List

Search

10

S/N	Profile Photo Url	Name	Email	Admission Number	Class	Section	Locked
1	AH	Aracely Hyatt	halvorson.ariane@example.com	ZTvQwttG3T	Nicklaus Spinka	Letha Monahan	Unlocked
2	AA	ariful islam arif	arif@gmail.com	DSI/24/382512	Kindergarten 1	Gold	Unlocked
3	BI	Baylee Kessler IV	denesik.jake@example.net	g5bDzVjze7	Kindergarten 1	Gold	Unlocked
4	CM	Chaim Dibbert MD	roxanne.kassulke@example.org	Uz2zeQw8fp	Nursery 1	Mr. Donnie Legros	Unlocked
5	DN	Damon Nienow	vern74@example.net	5kmYD5nd70	Nursery 1	Mr. Donnie Legros	Unlocked
6	DI	Dr. Brock Ullrich I	crist.gina@example.net	6BRwI3Y0xQ	Kindergarten 1	Gold	Unlocked
7	DO	Dr. Estrella Oberbrunner	chelsey.kihn@example.org	sEGK0ajcdy	Nursery 1	Mr. Donnie Legros	Unlocked
8	EP	Earlene Reichert PhD	henri.mitchell@example.org	SZ1745tiXy	Kindergarten 1	Gold	Unlocked
9	HA	Hosea Abernathy	carroll.manley@example.org	cNhqjdJdqY7	Kindergarten 1	Gold	Unlocked
10	JD	Jane Doe	student@student.com	sUjK0qP811	Kindergarten 1	Gold	Unlocked

Showing 1 to 10 of 12 results

< 1 2 >

Figure-6

School-Management

Admin

Dashboard

Manage Profile

User Profile

Multi Schools Management

Schools

Administration

School Settings

Classes

Sections

Students

View students

Create student

Promote students

Manage promotions

Graduate students

Create Student

You are currently on Default School - School of Computing

Dashboard > Students > Create

Create Student

All fields marked * are required

Select Profile image

First name *

Last name *

Other names *

Email address *

Password *

Confirm password *

Personal information

Birthdate *

Gender *

Blood group *

Phone number

Address *

Religion *

Nationality *

State *

City *

Class information

Choose a class *

Choose a section *

Admission number

Date of admission *

Create

Figure-7

School-Management

Admin

Schools

Administration

School Settings

Classes

Sections

Students

View students

Create student

Promote students

Manage promotions

Graduate students

Manage graduations

Account Applications

Teachers

Manage Promotions

You are currently on Default School - School of Computing

Dashboard > Students > Promotions For 2024 - 2025

Promotion List

Search 10

S/N	Old Class	New Class	Old Section	New Section	New Section	Actions	Delete
1	Kindergarten 1	Kindergarten 2	Gold	Ruby	1	Actions	Delete
2	Kindergarten 1	Kindergarten 2	Gold	Ruby	1	Actions	Delete
3	Kindergarten 1	Kindergarten 2	Gold	Ruby	1	Actions	Delete
4	Kindergarten 1	Kindergarten 2	Gold	Ruby	1	Actions	Delete

Figure-8

School-Management

Dashboard

Manage Profile

User Profile

Multi Schools Management

Schools

Administration

School Settings

Classes

Sections

Students

Account Applications

Teachers

Parents

Admins

Profile

You are currently on Default School - School of Computing

Dashboard > Profile

Profile Information

Update your account's profile information and email address.

All fields marked * are required

A

New Photo

Name

Admin

Email

super@admin.com

Birth day *

04/04/2000

Gender *

Male

Nationality *

Nigeria

State *

Abia

City *

ctg

Phone number

01711111111

Address *

super admin street

Religion *

Islam

Blood group *

B+

Save

Update Password

Ensure your account is using a long, random password to stay secure.

Current Password

New Password

Confirm Password

Save

Two Factor Authentication

Add additional security to your account using two factor authentication.

You have not enabled two factor authentication.

When two factor authentication is enabled, you will be prompted for a secure, random token during authentication. You may retrieve this token from your phone's Google Authenticator application.

Enable

Browser Sessions

Manage and log out your active sessions on other browsers and devices.

If necessary, you may log out of all of your other browser sessions across all of your devices. Some of your recent sessions are listed below; however, this list may not be exhaustive. If you feel your account has been compromised, you should also update your password.

Windows - Edge

127.0.0.1, This device

Windows - Edge

127.0.0.1, Last active 9 hours ago

Log Out Other Browser Sessions

Figure-9

7. Limitations

1. High Initial Costs:

- **Implementation Expenses:** Setting up a comprehensive SMS can be expensive due to software licensing, hardware requirements, and initial configuration.
- **Training Costs:** Training staff and users to effectively use the system can add to the overall expense.

2. Complexity:

- **User Interface:** If the system is not user-friendly, it can lead to confusion and frustration among users, especially those who are not tech-savvy.
- **Overwhelming Features:** An abundance of features might overwhelm users, leading to underutilization of the system.

3. Dependence on Technology:

- **Technical Issues:** Reliance on technology can lead to disruptions if there are system failures, server downtimes, or software bugs.
- **Data Security:** Storing sensitive student and staff information digitally raises concerns about data breaches and unauthorized access.

4. Resistance to Change:

- **Cultural Barriers:** Staff and parents may resist adopting new technology or processes, leading to potential conflicts and reduced system effectiveness.
- **Adaptation:** Existing workflows and processes might need to be adjusted, causing temporary disruptions during the transition.

5. Limited Customization:

- **Rigid Systems:** Some SMS solutions offer limited customization options, making it challenging to tailor the system to specific institutional needs or workflows.
- **Scalability Issues:** As the school grows or changes, the system may not easily adapt to new requirements or larger user bases.

6. Data Management Challenges:

- **Data Entry Errors:** Manual data entry can lead to inaccuracies, which can affect decision-making and reporting.
- **Integration Issues:** Integrating the SMS with other existing systems (e.g., accounting software, learning management systems) can be complex and may require additional resources.

7. User Limitations:

- **Access Issues:** Not all stakeholders may have equal access to the system, leading to communication gaps (e.g., parents not having access to information).
- **Skill Disparities:** Variability in users' technical skills can affect overall system efficiency and usage.

8. Maintenance and Updates:

- **Ongoing Costs:** Regular updates, maintenance, and support can incur additional costs over time.
- **Version Compatibility:** Upgrading to newer versions may require retraining and can introduce compatibility issues with existing data or workflows.

9. Inadequate Support:

- **Customer Support:** If the vendor does not provide adequate technical support, resolving issues can become difficult and time-consuming.
- **Documentation:** Poor or insufficient documentation can hinder effective usage and troubleshooting of the system.

10. Data Privacy Concerns:

- **Regulatory Compliance:** Schools must ensure compliance with data protection regulations (e.g., FERPA, GDPR), which can add complexity to data handling and storage practices.
- **Parental Concerns:** Parents may have concerns about how their children's data is being stored and used, impacting trust in the system.

Conclusion:

The **School Management System** is a comprehensive tool designed to streamline and enhance the administration of educational institutions. By integrating various functions such as student enrollment, attendance tracking, grading, and fee management, it improves operational efficiency and reduces administrative burdens. The system facilitates seamless communication among administrators, teachers, students, and parents, fostering a collaborative educational environment. Utilizing technologies like Laravel, MySQL, and XAMPP ensures a robust and secure platform for data management. The intuitive user interface enhances user experience, making it accessible for all stakeholders. Moreover, the system allows for data-driven decision-making through real-time reporting and analytics. Its modular design enables easy scalability and customization to meet the unique needs of different schools. Overall, the School Management System enhances productivity, improves educational outcomes, and supports the growth and development of students and staff alike.