## Overview

The Student Management Application is designed to streamline the management of a kindergarten's daily operations. The app will facilitate efficient handling of student information, attendance, staff management, communication between teachers and parents, and various other administrative tasks.

## Objectives

1. Improve Efficiency: Automate and simplify administrative tasks.
2. Enhance Communication: Facilitate smooth communication between parents and staff.
3. Streamline Management: Provide a centralized platform for managing students, staff, and activities.

## Features

1. **User Management**
    1. **Admin**

        Permissions:

        - Full access: Complete access to all application features.
        - User Management: Create, edit, delete, and manage all user accounts (Admins, Principals, Teachers, Parents, Teaching Assistants).
        - Financial Reports: Access and manage financial reports and budgets.
        - Curriculum and Schedule Settings: Configure and update curriculum guidelines and schedules.
        - Communication: Communicate with all users across the platform.
        - Emergency Alerts: Send and manage emergency alerts and notifications.

        Responsibilities:

        - Administrative Management: Manage the preschool at an administrative level.
        - Educational Standards: Ensure that educational standards and practices are being followed.
        - Leading Teaching Staff: Provide guidance and support to the teaching staff.

    2. **Principal**

        Permissions:

        - Classroom Data: Access all classroom-related data.
        - Teacher Performance Reports: View and analyze reports on teacher performance.
        - Student Progress Reports: Access detailed reports on student progress.
        - Parent Communications: Communicate with parents and address their concerns.
        - Modify Schedules and Curriculum Guidelines: Update class schedules and curriculum guidelines as needed.

        Responsibilities:

        - Administrative Management: Manage the preschool at an administrative level.
        - Educational Standards: Ensure that educational standards and practices are being followed.
        - Leading Teaching Staff: Provide guidance and support to the teaching staff.

    3. **Lead Teacher**

        Permissions:

        - Classroom Data: Access data specific to their own classroom.
        - Student Progress Reporting: Report on student progress and performance.

- Modify Lesson Plans: Update and modify lesson plans within set curriculum guidelines.
- Communication: Communicate with teachers, teaching assistants, and parents.

Responsibilities:

- Classroom Oversight: Oversee and manage classroom activities and environment.
- Mentoring: Mentor and support other teachers.
- Curriculum Implementation: Coordinate with the principal on implementing the curriculum.

4. **Teacher**

Permissions:

- Classroom Data: Access data specific to their own classroom.
- Update Student Records: Modify and update student records.
- Parent Communication: Communicate with parents regarding their child's progress.
- Student Assessments: Input assessments and track student performance.

Responsibilities:

- Classroom Management: Conduct classes and ensure a safe and engaging learning environment.
- Lesson Preparation: Prepare lesson materials and educational activities.
- Student Assessment: Assess and monitor student progress and development.

5. **Teaching Assistant**

Permissions:

- Classroom Data: Limited access to view classroom data and student progress.
- Update Records: Assist with updating student records under the supervision of a teacher.

Responsibilities:

- Support Role: Support the teacher with classroom management and lesson activities.
- Student Care: Help with the care and supervision of students.

6. **Parent**

Permissions:

- Child's Progress Reports: View progress reports and updates for their child.
- School Announcements: Access school announcements and updates.
- Communication: Communicate with the school and teaching staff.

Responsibilities:

- Engagement: Engage with their child's education and school activities.
- Meeting Attendance: Attend parent-teacher meetings and school events.
- Feedback: Provide feedback through appropriate channels.

2. **Student Management**
   - Registration:
     - Add, edit, and remove student profiles.
     - Capture personal details, medical history, and emergency contacts.
   - Attendance:
     - Mark and view daily attendance.
     - Generate attendance reports.
   - Performance Tracking:
     - Track and update student performance and behavior.
     - Generate and view performance reports.
3. **Class and Schedule Management**
   - Class Scheduling:

- Create and manage class schedules.
- Assign teachers and classrooms.

- Timetable:
  - View and update daily timetables.
  - Notify parents and students of schedule changes.

4. **Communication**
   - Announcements:
     - Post announcements and updates.
     - Notify parents and staff via push notifications.
   - Messages:
     - Enable direct messaging between parents and teachers.
     - View message history.
5. **Event Management**
   - Event Creation:
     - Create and manage events (e.g., parent-teacher meetings, school functions).
     - Notify attendees and manage RSVPs.
   - Event Calendar:
     - View and manage upcoming events in a calendar format.
6. **Reports and Analytics**
   - Generate Reports:
     - Attendance, performance, and administrative reports.
     - Export reports in various formats (PDF, Excel).
   - Analytics:
     - Analyze trends and insights from student performance and attendance data.
7. **Data Security and Backup**
   - User Authentication:
     - Secure login and role-based access control.
   - Data Backup:
     - Regular data backups to prevent data loss.

## Technology Stack

**Frontend**

- For Android-Java and iOS-Swift
  - App development for Android and iOS.
  - Responsive and intuitive user interface design.

**Backend**

- Framework: React Native
  - Robust and scalable backend development.
  - RESTful API development.
- Database: PostgreSQL
  - Reliable and scalable database solution.
- Authentication:
  - JWT (JSON Web Tokens) for secure user authentication.
- Hosting:
  - Deploy backend on platforms like AWS, DigitalOcean etc.

## System Architecture

1. Frontend (Java and Swift)
   - User Interface: Handles all user interactions and displays data retrieved from the backend.
   - State Management: Manages application state and updates the UI accordingly.
2. Backend (React Native)
   - API Endpoints: Provides RESTful endpoints for all operations (CRUD operations for students, staff, etc.).
   - Business Logic: Handles core functionalities like user management, scheduling, and reporting.
   - Database: Stores all application data, including user information, student records, and event details.
3. Integration • HTTP Requests: Frontend communicates with the backend via RESTful API endpoints.
   - Authentication: Secure authentication and authorization using JWT tokens.

## Project Phases

1. Planning • Define requirements and specifications.
   - Design system architecture and user interfaces.
2. Design • Create wireframes and mockups for the Android and iOS frontend.
   - Design database schema and API endpoints.
3. Development • Develop the frontend and backend.
   - Implement features and integrate frontend with backend.
4. Testing • Perform unit tests, integration tests, and user acceptance tests.
   - Ensure cross-platform compatibility and security.
5. Deployment • Deploy the backend to a cloud service provider.
   - Publish the app to Google Play Store and Apple App Store.
6. Maintenance • Monitor performance and user feedback..
   - Implement updates and improvements as needed.