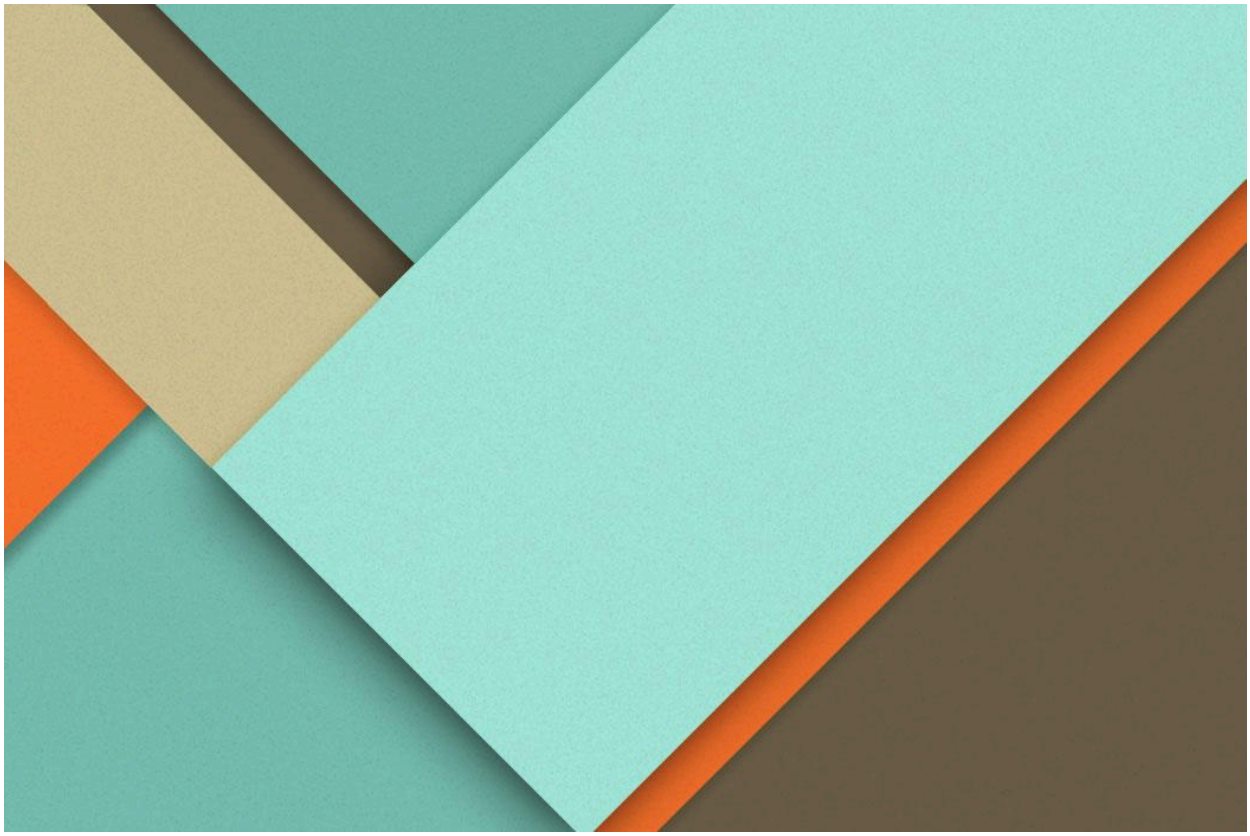


Attendance Tracking System

Members

Alforon, Jhamaica P.

Barbarona, Quennie Rose

Buhisan, Albert T.

Project Overview

The Attendance Tracking System is designed to simplify and secure the process of recording student attendance while ensuring data integrity and accessibility. By leveraging cloud-based storage and strong security measures, the system will provide a seamless experience for instructors, students, and administrators.

System Requirements

1. User Management Requirements

R01.

The system must allow authorized users to register students and instructors.

R01.01

Authorized users must be able to upload student ID photos along with personal details (full name, student ID, year level, program, and faculty) during registration.

R01.02

The system must allow authorized users to remove students or instructors from the system.

R01.03

Authorized user must have the authority to unregister an instructor's phone if necessary.

R01.04

Only authorized users can access specific features.

R01.05

Only authorized users can locate lost registered devices of the instructor.

2. Attendance Tracking Requirements

R02.

The system must provide attendance tracking functionality for instructors.

R02.01

Instructors must be able to take attendance using ID Barcode scanning via registered devices (phones or laptops).

R02.02

Instructors must be able to take attendance using ID Barcode scanning via registered devices (phones or laptops).

R02.03

The system must allow instructors to drop students who have accumulated three consecutive absences.

R02.05

The system must generate attendance reports for authorized users.

R02.06

Students must be able to view their attendance records per course.

3. Security Requirements

R03.

The system must implement strong security measures to protect data integrity.

R04.01 Role-Based Access Control (RBAC) must be enforced to restrict access to sensitive data.

R03.02

Admin-controlled data encryption must be used to ensure data confidentiality.

R03.03

Multi-Factor Authentication (MFA) must be required for instructor logins.

R03.04

The system must operate over HTTPS using SSL security protocols.

4. Infrastructure & Deployment Requirements

R04.

The system must be cloud-based and ensure scalability.

R04.01

The system must use MongoDB Atlas or MySQL for cloud-based data storage.

R04.02

The system must be accessible and scalable.

R04.03

The system must not provide an API for integration with other university systems.

5. Deployment Timeline

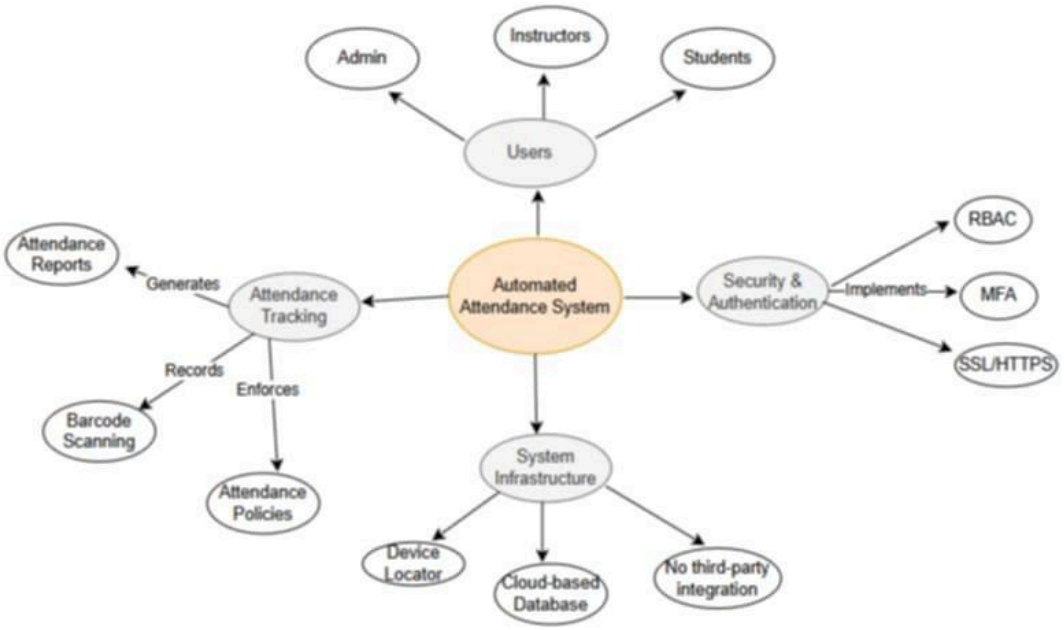
R05.

On May 15, 2025 is the initial checking for the system.

R06.

The system must be fully operational and deployed on May 22, 2025.

Conceptual Model



Overview

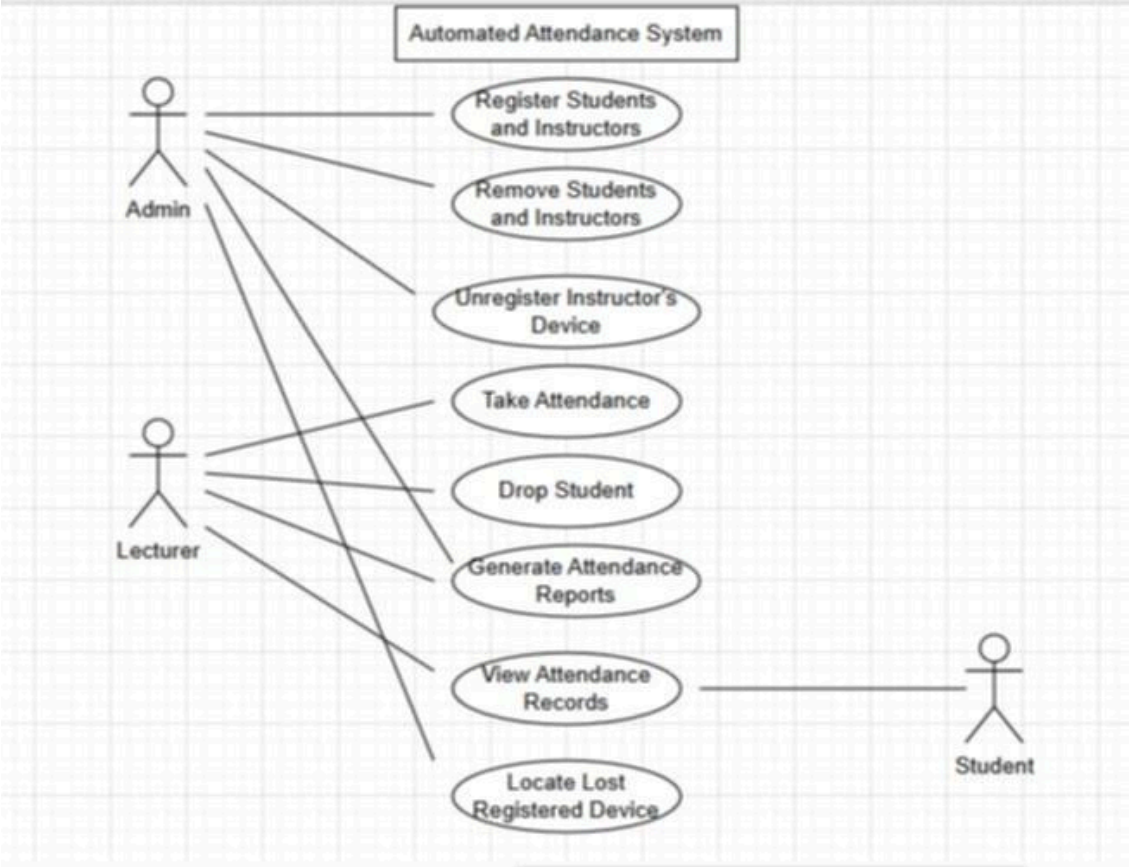
The **Automated Attendance System** consists of three primary components:

- **Users**: Admin, Instructors, and Students
- **Attendance Tracking**: Barcode scanning, attendance policies, and attendance reports
- **Security & Authentication**: RBAC, MFA, SSL/HTTPS
- **System Infrastructure**: Cloud-based database, device locator, and no third-party integration

High-Level Use Case

Actors and Use Cases:

- **Admin**: Register/Remove students and instructors, Unregister instructor's device, Locate lost registered devices.
- **Lecturer**: Take attendance, Drop students, Generate attendance reports, View attendance records.
- **Student**: View attendance records.



Traceability Matrix

Requirement ID	Use Case	Feature Implemented
R01.01	Register Students/Instructors	Upload student ID photos and details
R01.02	Remove Students/Instructors	Authorized user can delete students/instructors
R01.03	Unregister Instructor's Device	Remove device from system
R01.05	Locate Lost Device	Track registered instructor devices
R02.01	Take Attendance	Barcode scanning via registered devices
R02.03	Drop Student	Remove students with 3 consecutive absences
R02.05	Generate Attendance Reports	Export attendance data
R02.06	View Attendance Records	Students access attendance logs
R03.01	Implement RBAC	Role-based permissions enforced
R03.02	Encrypt Data	Secure database with encryption
R03.03	Enable MFA	Multi-factor authentication for logins
R03.04	Use HTTPS	Secure communication using SSL

Project Plan

Milestones and Deliverables

Milestone	Target Date	Deliverable
Requirement Analysis	March 2025	System Requirements Document
System Design	April 2025	Conceptual Model, High-Level Use Case Diagram
Development Phase 1	April 2025	User Management, Attendance Tracking
Development Phase 2	May 2025	Security Implementation, Reports Generation
System Testing	May 10, 2025	Test Cases and Bug Fixes
Initial System Review	May 15, 2025	Beta Version Deployment
Full Deployment	May 22, 2025	Operational System

Git Project Setup

Repository Structure

```
/attendance-tracking-system
├── /docs           # Documentation files (requirements, project plan)
├── /frontend       # React-based front-end
├── /backend        # Node.js with Express API
├── /database       # MongoDB schemas & scripts
├── /tests          # Unit and integration tests
├── README.md       # Project overview
├── .gitignore      # Files to exclude from Git
```

Git Workflow

1. **Initialize Repository:** `git init`
2. **Create Main Branch:** `git checkout -b main`
3. **Feature Branching:**
    - `git checkout -b feature-user-management`
    - `git checkout -b feature-attendance-tracking`
4. **Commit Changes:** `git commit -m "Added user registration module"`
5. **Push to GitHub:** `git push origin feature-user-management`
6. **Create Pull Requests (PRs)** for code review before merging.

Conclusion

This document provides a comprehensive breakdown of the **Attendance Tracking System** covering system requirements, models, use cases, a traceability matrix, project plan, and Git setup. The next steps involve refining the system architecture and starting the initial development phase.