



# **Automated Attendance System**

*Faculty of Computing Data Sciences Engineering and Technology*

**Mike A. Misoles**

**Kenneth Jay Paragoso**

**Kristine Mae Bonotan**

**Bachelor of Science and Information Technology**

**May 2025**

## Table of Content

1. Introduction .....	3
1.1 Purpose.....	3
1.2 Scope.....	3
1.3 Definitions, Acronyms, and Abbreviations .....	3
1.4 References.....	3
2. System Overview .....	<b>Error! Bookmark not defined.</b>
2.1 System Description .....	<b>Error! Bookmark not defined.</b>
2.2 User Roles and Personas.....	<b>Error! Bookmark not defined.</b>
3. Architecture Design .....	<b>Error! Bookmark not defined.</b>
3.1 System Architecture Diagram .....	<b>Error! Bookmark not defined.</b>
3.2 Component Diagram .....	<b>Error! Bookmark not defined.</b>
4. Module Description .....	<b>Error! Bookmark not defined.</b>
4.1 QR Code Generator .....	<b>Error! Bookmark not defined.</b>
4.2 Attendance Module.....	<b>Error! Bookmark not defined.</b>
4.3 User Management.....	<b>Error! Bookmark not defined.</b>
4.4 Reporting Module .....	<b>Error! Bookmark not defined.</b>
4.5 Authentication Module .....	<b>Error! Bookmark not defined.</b>
5. Data Design .....	<b>Error! Bookmark not defined.</b>
5.1 Data Model Diagram .....	<b>Error! Bookmark not defined.</b>
5.2 Data Flow .....	<b>Error! Bookmark not defined.</b>
6. Technology Stack.....	<b>Error! Bookmark not defined.</b>
7. Security Considerations .....	<b>Error! Bookmark not defined.</b>
8. Scalability & Performance .....	<b>Error! Bookmark not defined.</b>
9. Integration.....	<b>Error! Bookmark not defined.</b>
10. Assumptions & Constraints .....	<b>Error! Bookmark not defined.</b>
11. Appendices.....	<b>Error! Bookmark not defined.</b>

## Introduction

### Purpose

The Automated Attendance System aims to streamline and automate the process of tracking student attendance in educational institutions by leveraging QR code technology. It eliminates manual attendance logging, reduces errors, and provides real-time attendance data to instructors and administrators.

### Scope

- Manage user roles: Admin, Instructor, and Student.
- Each student is issued a unique QR code representing their identity.
- Instructors scan student QR codes to record attendance instantly.
- Admin manages user accounts, views attendance reports, and system configurations.
- The system is web and/or mobile-based, accessible to all user roles.

### Definitions, Acronyms, and Abbreviations

- **QR Code:** Quick Response Code, a two-dimensional barcode storing student info.
- **Admin:** User who manages the system.
- **Instructor:** User who records attendance by scanning QR codes.
- **Student:** User who possesses a unique QR code for attendance.

### References

<https://www.mongodb.com/docs/manual/>

[https://www.researchgate.net/publication/342343678 Smart Attendance System using QR Code](https://www.researchgate.net/publication/342343678_Smart_Attendance_System_using_QR_Code)

<https://nodejs.org/docs/latest/api/>

## 2. System Overview

### 2.1 System Description

The system consists of three main user roles interacting with a centralized backend database. Students are registered and assigned unique QR codes, which serve as their digital identity for attendance. Instructors use scanning devices (mobile app or web camera) to scan QR codes during class, which instantly records attendance. Admin monitors attendance data and manages users.

### 2.2 User Roles and Personas

Role	Description	Key Actions
------	-------------	-------------

Admin	Manages users and system settings	Create/update users, view reports
-------	-----------------------------------	-----------------------------------

Instructor	Takes attendance by scanning QR codes	Scan QR code, view attendance history
------------	---------------------------------------	---------------------------------------

Student	Receives unique QR code to be scanned	View own attendance records
---------	---------------------------------------	-----------------------------

## 3. Architecture Design

### 3.1 System Architecture Diagram

(Here, insert a diagram showing: Student devices → QR code generation → Instructor scanning device → Backend server → Admin dashboard)

### 3.2 Component Diagram

Component	Description
-----------	-------------

QR Code Generator	Generates unique QR codes for each student
-------------------	--

Attendance Module	Processes QR code scans, validates and logs attendance
-------------------	--

Component	Description
-----------	-------------

User Management	Handles creation and management of Admin, Instructor, Student
-----------------	---

Reporting Module      Generates attendance reports accessible by Admin Authentication  
Manages login and access control for all users

#### 4.      Module Description

##### 4.1      QR Code Generator

- Description: Generates unique QR codes linked to student IDs.
- Responsibilities: Create, store, and distribute QR codes securely.
- Interfaces: Student profile creation UI, QR code download/print options.

##### 4.2      Attendance Module

- Description: Scans and validates QR codes, marks attendance.
- Responsibilities: Real-time attendance recording, duplicate scan prevention.
- Interfaces: QR scanner interface for instructors, backend API for attendance logging.

##### 4.3      User Management

- Description: Admin module for user account lifecycle.
- Responsibilities: CRUD operations for Admin, Instructor, and Student accounts.

##### 4.4      Reporting Module

- Description: Provides attendance statistics and reports.
- Responsibilities: Generate daily, weekly, monthly reports; export to CSV/PDF.

##### 4.5      Authentication Module

- Description: Ensures secure login and access control.
- Responsibilities: Role-based authentication, session management.

## 5. Data Design

### 5.1 Data Model Diagram

Entities:

- User (Admin, Instructor, Student)
- StudentQR (StudentID, QRCodeData)
- AttendanceRecord (AttendanceID, StudentID, InstructorID, Timestamp, Status)

### 5.2 Data Flow

- Student data → QR Code generated → Student receives QR code
- Instructor scans QR code → System verifies and logs attendance → Admin views reports

## 6. Technology Stack

Layer Technology/Tools

Frontend React (Web) or React Native (Mobile)

Backend Node.js

Database MongoDB

QR Code QR code generation libraries (e.g., qrcode.js)

Authentication JWT-based authentication

## **7. Security Considerations**

- Secure storage of QR codes to prevent forgery.
- Authentication and role-based access control.
- SSL/TLS encryption for all communications.
- Prevent replay attacks on QR code scans by timestamp validation.

## **8. Scalability & Performance**

- Use efficient indexing on student IDs for quick attendance logging.
- Support multiple concurrent instructors scanning simultaneously.
- Optionally, deploy backend with load balancing for high availability.

## **9. Integration**

- Integration with existing student information systems (SIS) for syncing student data (optional).
- API endpoints for mobile apps and web clients.

## **10. Assumptions & Constraints**

- Students have access to their unique QR codes (printed or digital).
- Instructors have devices capable of scanning QR codes.
- Network connectivity is available during attendance scanning.

## **11. Appendices**

- Sample QR code format specification
- User interface wireframes (optional)