**DESIGN DOCUMENT**

**[STUDENT ATTENDANCE MONITORING SYSTEM WITH QR CODE]**

**Prepared By:**

**Camacho, Vien Mabee**

**Galvez, Sean Andrew**

**Java, Marjorie**

**Ortiz, Krisha Mae**

**INTRODUCTION**

**Overview of the Software System Being Developed**

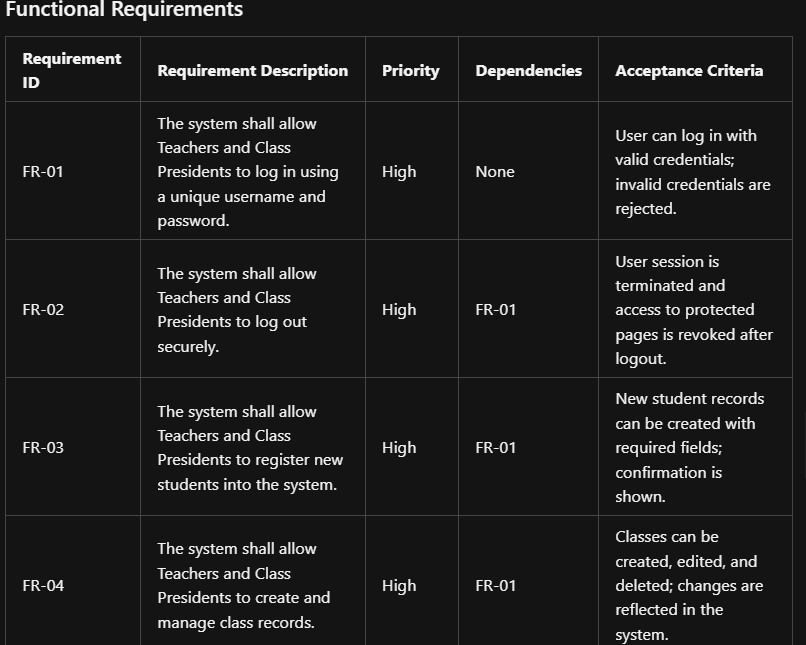
The Student Attendance Monitoring System with QR Code is a web-based application designed to facilitate and automate the process of recording student attendance in educational institutions. By leveraging QR code technology, the system enables fast, accurate, and contactless attendance tracking.

The application is intended for use exclusively by Class Presidents and Teachers, providing them with interfaces to manage students, classes, and attendance records, as well as to generate reports and visualize attendance data. The system aims to streamline administrative tasks, minimize manual errors, and provide real-time insights into student attendance patterns.

# **Scope of the Requirements Specification**

This requirements specification covers all functional and non-functional requirements for the Student Attendance Monitoring System with QR Code. Including

1. User roles and permissions (Class President and Teacher)
2. Student, class, and attendance management
3. QR code-based attendance tracking
4. Data visualization and reporting
5. Security, performance, and usability requirements

FUNCTIONAL REQUIREMENTS

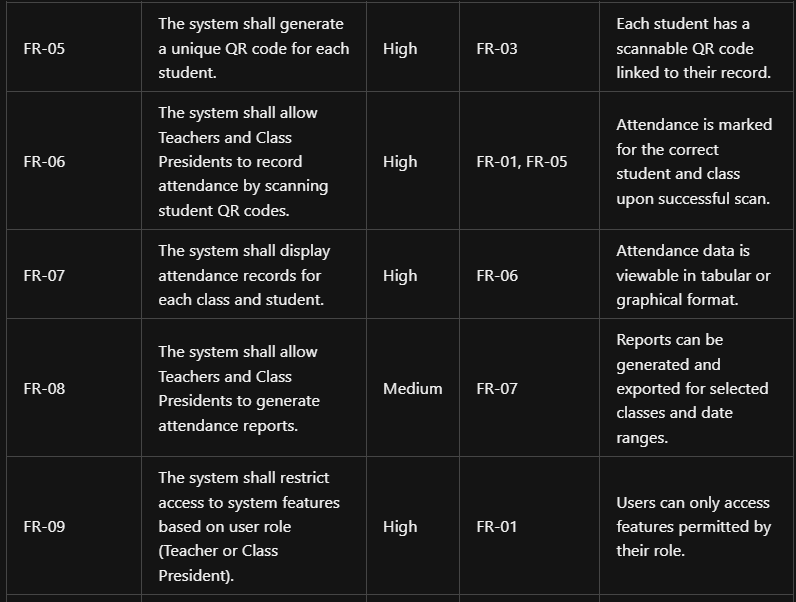
TABLE 1

Table 1

**NON-FUNCTIONAL REQUIREMENTS**

**Performance**

* 1. The system shall respond to user actions (login, attendance marking) within 2 seconds under normal load.
  2. The system shall support at least 20 concurrent users without significant degradation in performance.
  3. Attendance records shall be updated in real-time and reflected immediately in the user interface.

**Usability**

* 1. The system shall provide a clean, intuitive, and user-friendly interface suitable for both Teachers and Class Presidents.
  2. All primary functions (attendance marking, student/class management) shall be accessible within three clicks from the main dashboard.
  3. The system shall provide clear error messages and guidance for invalid actions or inputs.
  4. The system shall be accessible on both desktop and mobile browsers.

**Reliability**

* 1. The system shall be available 99% of the time, excluding scheduled maintenance.
  2. The system shall automatically recover from server restarts without data loss.
  3. All critical operations (Attendance marking, student registration) shall be logged for audit purposes.

**Security**

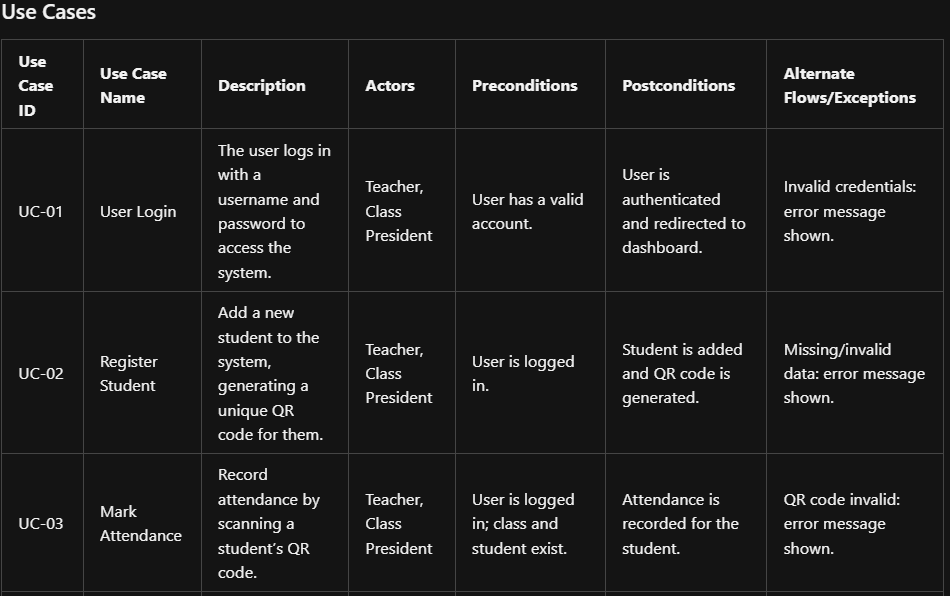
* 1. The system shall require authentication for all users.
  2. User passwords shall be stored using secure, salted hashing algorithms.
  3. The system shall enforce role-based access control, ensuring only authorized users can access or modify data.
  4. All data transmissions between client and server shall be encrypted using HTTPS in production.
  5. The system shall validate and sanitize all user inputs to prevent security vulnerabilities such as SQL injection and XSS.

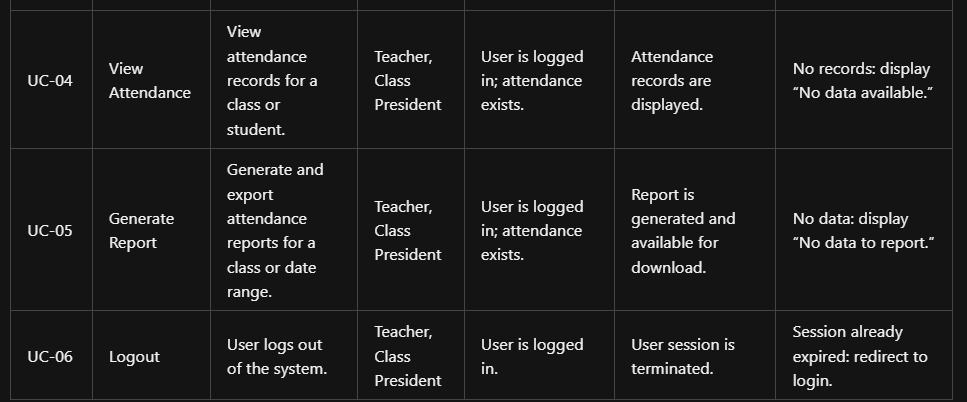
**Scalability**

1. The system shall be designed to allow migration from SQLite to more robust databases (PostgreSQL, MySQL) if user load increases.
2. The system architecture shall support horizontal scaling of the backend server if required.
3. Static assets and frontend code shall be cacheable and deliverable via a CDN if needed.

**Maintainability**

* 1. The system codebase shall be modular and well-documented to facilitate future enhancements and bug fixes.
  2. Configuration files and environment-specific settings shall be separated from the source code.
  3. The system shall include automated tests for critical functionality.
  4. Database migrations shall be managed using a version control tool (Alembic).

**USE CASES**



**TABLE 2**

**DATA REQUIREMENTS**

**Data Entities**

**The main data entities managed by the system are**

1. User
2. Student
3. Class
4. Attendance

**Attribute User**

* 1. Use id (Primary Key)
  2. username
  3. password
  4. role (Teacher, Class President)

**Student**

* 1. student id (Primary Key)
  2. name
  3. class id (Foreign Key)
  4. photo path
  5. QR code path

**Class**

* 1. Class Id (Primary Key)
  2. name
  3. description

**Attendance**

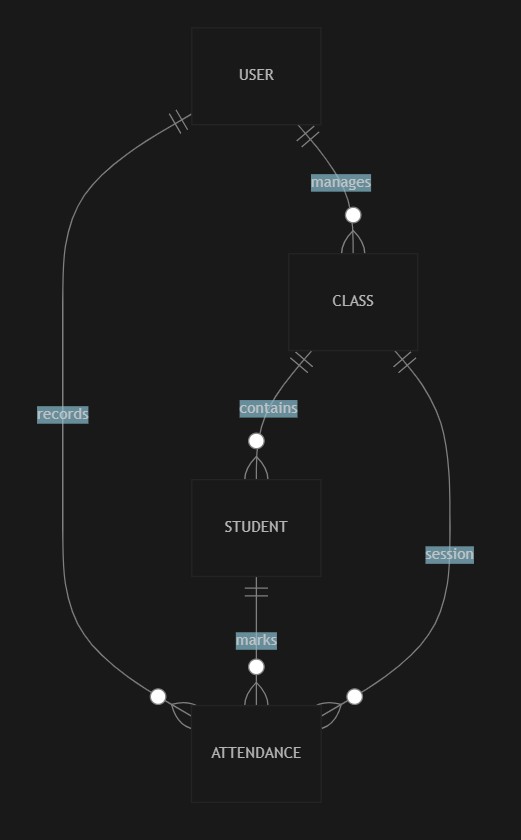
1. attendance id (Primary Key)
2. student id (Foreign Key)
3. class id (Foreign Key)
4. date
5. status (Present, Absent, Late, etc.)
6. scan time (timestamp of QR scan)

Relationships

1. **User**: Can manage multiple Classes and Attendance records.
2. **Class**: Contains multiple Students.
3. **Student**: Belongs to one Class; can have multiple Attendance records.
4. **Attendance**: Links a Student, a Class, and a Date (many-to-one relationships with Student and Class).

ER Diagram (Textual Representation)

1. A **User** manages many **Classes** and **Attendance** records.
2. A **Class** contains many **Students**.
3. A **Student** belongs to one **Class** and has many **Attendance** records.
4. An **Attendance** record references one **Student** and one **Class**.

Figure 1

**Legend**

1. ||--o means "one-to-many" relationship.
2. Entities: USER, CLASS, STUDENT, ATTENDANCE.
3. Relationships:
4. A USER manages many CLASSES and ATTENDANCE records.
5. A CLASS contains many STUDENTS and ATTENDANCE sessions.
6. A STUDENT belongs to one CLASS and has many ATTENDANCE records.

**ASSUMPTIONS AND CONSTRAINTS**

**Assumptions**

* 1. Only Class Presidents and Teachers will use the system; students will not have direct access.
  2. All users (Class Presidents and Teachers) have basic computer literacy and access to a device with a camera for QR code scanning.
  3. Each student will be assigned a unique QR code, and QR codes will be available and scannable during attendance sessions.
  4. The system will be deployed in an environment with a stable internet connection for both frontend and backend communication.
  5. The number of concurrent users will not exceed the expected load for a typical class or school setting (e.g., up to 20 users at a time).
  6. The database (initially SQLite) will be sufficient for the expected data volume; migration to a more robust DBMS is possible if needed.
  7. All required data (student info, class lists) will be entered and maintained by authorized users.

**Constraints**

1. The system must support only the roles of Class President and Teacher; no other user roles are considered.
2. The application is web-based and requires a modern browser (e.g., Chrome, Firefox, Edge) with JavaScript enabled.
3. QR code scanning functionality depends on device camera access and browser compatibility.
4. The system must comply with institutional data privacy and security policies.
5. The initial deployment uses SQLite, which may limit scalability and concurrent write operations.
6. The system must not store plain-text passwords; all credentials must be securely hashed.
7. Integration with external systems (e.g., SMS, email, school portals) is not included in the initial scope.
8. The system is designed for English language use; localization is not required in the initial version.

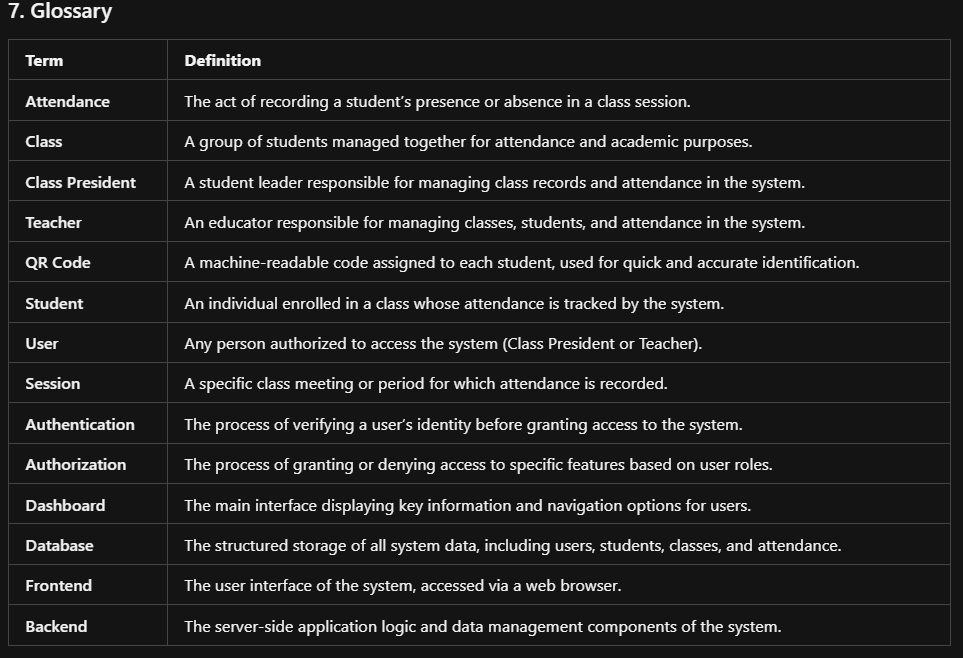
**GLOSSARY**

Table 1