# High-Level Design Document

## **Project Title:** Project X

## **1.Introduction**

* 1. Purpose

Project X is designed to modernize and simplify attendance tracking in educational settings through the use of mobile devices and QR codes. Instructors generate a unique QR code for each session, which students scan with their devices to log their attendance. The system provides a secure, accurate, and real-time solution for monitoring attendance.

## 2.**System Overview**

Project X supports three main roles:

**Instructor**: Creates sessions, generates QR codes, and views attendance logs.

**Student**: Scans QR codes and sends scanned data to mark attendance.

**Admin**: Manages users, courses, and generates reports

## **Architecture Overview**

The system is composed of the following key components:

* **Mobile Applications**: Separate apps for Instructors, Students, and Admins to manage their respective roles and functionalities.
* **REST API**: Facilitates secure authentication and communication between the mobile apps and the backend.
* **Backend Logic**: Responsible for QR code validation, session timing, and attendance tracking.
* **MongoDB Database**: Stores all essential data, including user profiles, session details, and attendance records.

### 3.1 Architecture Diagram Reference

## 4. **Component Description**

## **4.1 Instructor Module**

## **Create Session** – Set up a class session with details such as date, time, and course metadata.

## **Generate QR Code** – Generate a time-sensitive QR code for students to scan and mark their attendance.

## **View Attendance Log** – Access and review a log of students who attended each session.

## **4.2 Student Module**

## **Scan QR Code** – Use the app to scan the instructor's QR code during class.

## **Submit Attendance** – Automatically send scanned data to the backend for validation and attendance recording.

## **4.3 Admin Module**

## **Manage Users** – Add, update, or remove instructors and students.

## **Manage Courses** – Assign courses and manage scheduling.

## **Generate Reports** – Export attendance data filtered by course, date, or student.

## **4.4 API Layer**

## **User Authentication** – Manages login sessions and role-based access.

## **Backend Communication** – Interfaces with the backend logic and MongoDB database.

## **Attendance Processing** – Handles QR code validation and attendance submission.

## **4.5 Backend Logic**

## **Validate QR Code** – Confirms the QR code is valid and has not expired.

## **Check Session Timing** – Ensures attendance is submitted within the allowed timeframe.

## **Verify Enrollment** – Confirms the student is enrolled in the relevant course.

## **Record Attendance** – Stores attendance records in the database.

## **4.6 Database (MongoDB)**

## **Collections include:**

## **users** – Contains instructor, student, and admin profiles.

## **sessions** – Stores session details and generated QR codes.

## **attendance** – Logs of time-stamped student attendance.

## **courses** – Holds course information and enrollment records.

## 5. **Data Flow Summary**

**Instructor** logs in → creates a session → generates QR

**Student** logs in → scans QR → data is sent to API

**API** passes data to backend → checks validity → attendance is saved

**Admin** accesses reports, manages users/courses

## 6. **Security Measures**

Credential-based login for all users

QR codes expire after a short period

All sensitive data encrypted (especially passwords)

Only enrolled students can mark attendance