

Proxy-Resistant Real-Time QR-Based Smart Attendance System

¹Atharv Kadam, ²Vinayak Dhulubulu, ³Om Bhosale, ⁴Sneha Khot, ⁵Kiran Tangod

^{1,2,3,4}Student, Department of Computer Science and Engineering (Artificial Intelligence and Machine Learning), Kasegaon Education Society's Rajarambapu Institute of Technology, Affiliated to Shivaji University, Sakharale, MS-415414, India.

⁵Assistant Professor, Department of Computer Science and Engineering (Artificial Intelligence and Machine Learning), Kasegaon Education Society's Rajarambapu Institute of Technology, Affiliated to Shivaji University, Sakharale, MS-415414, India.

Email: atharvkadam1858@gmail.com¹, 2317049@ritindia.edu², 2467007@ritindia.edu³, 2317029@ritindia.edu⁴, Kiran.tangod@ritindia.edu⁵

Abstract:

The traditional methods of attendance tracking are slow, prone to mistakes and proxy attendances, and the use of static QR codes can be bypassed by taking screenshots. In this paper, we describe our developed Proxy-Resistant Real-Time QR-Based Smart Attendance System, a web-based application that uses role-based authentication, session-based QR codes generation, QR scanning through a mobile phone, face verification using a webcam along with a live prompt, attendance management, a dashboard for faculty in real-time, student management by administrators and spreadsheets for reporting. The faculty creates an attendance session and shows the QR code of it, the students scan the QR code through a mobile phone and proceed to face verification before being marked as present. We conducted testing using actual AIML student and faculty accounts and showed the whole process, dashboard update, reporting and multi-device capability. This paper gives an overview of our system description, implementation, deployment, testing results, limitations and future developments mentioned in our project report.

Keywords—Proxy Attendance, Dynamic QR Code, Face Verification, Liveness Detection, Attendance Dashboard, Deep Learning

I. INTRODUCTION

Attendance Management System is an academic activity for tracking the presence of students in classes, making decisions regarding academics, and keeping institution records. Traditional roll call system involves manual identification and data entry and takes up quite a bit of time from lectures. Proxy attendance has been stated in the report as one of the practical problems associated with the project since the student can try to mark his/her attendance using a third party or remotely using static credentials.

This system solves the problem with multi-stage authentication process. The faculty authenticates himself/herself, chooses the department, year and subject, and initiates a session. The server creates the session-specific QR code with the session context. The QR code is shown on the Teacher Dashboard. The student logs in through the mobile browser, scans the QR code and then goes through the facial recognition process. The attendance is marked when both stages of the session authentication process are successful.

The project was developed in two phases; Phase I consisted of problem identification, literature review, summary, SRS, architecture, UML & DFD design, frontend scaffolding, and backend API & database design. Phase II consists of development of authentication, QR generation/scanning, face verification, attendance management, admin dashboard, and reporting, followed by testing, deployment, and user-acceptance testing using live accounts.

The objectives documented include the reduction in proxy attendance through QR + face verification, reduction in faculty efforts through session-wise automation and one-click reports, hardware independent browser-accessible application, and increased transparency through live dashboards and downloadable attendance records. The report also includes relevance of the solution to SDG 4 (Quality Education) and SDG 9 (Industry, Innovation & Infrastructure).

II. RELATED WORK AND RESEARCH GAP

The project report considers twenty systems in attendance management, including attendance

based on QR codes, geo-location checking, face recognition, biometric attendance, dynamic QR approach, anomaly detection, and secure digital records. According to the literature review, a QR code can increase the efficiency of attendance tracking, but a static QR code can be transferred to another person. Face-based approach is more effective for identity verification, while location-based approaches can provide additional context-based constraint.

Reference [1] presents a QR-based attendance tracking with geo-location checking, while References [2] and [6] use a combination of QR and face recognition. Reference [12] presents a solution based on dynamic QR codes. Our system combines all these approaches with a session-specific QR code generated by the faculty members and verified with the AI-based face recognition.

Another requirement that was determined through the course of this project was visibility. The implemented solution provides faculty with a live attendance panel, override functionality, and monthly or defaulter report downloads. This allows for an audit-style process in which automatic verification is complemented with faculty intervention in the event of an unsuccessful capture of the legitimate individual.

The report determines the gap to be a lack of a hardware-less web browser solution that provides dynamic QR code verification, AI-assisted real-time facial verification with a liveness component, and a reporting dashboard. The project was conceived and built precisely to fill this gap.

III. PROPOSED SYSTEM AND METHODOLOGY

Methodology consists of the following sequential verification process. First, authentication by roles defines the type of user – either Student, Teacher or Admin. Second, teacher starts the QR session for the specified Department, Year and Subject. Third, server sends back the session ID and QR payload which is shown on teacher screen. Fourth, the student scans active QR using the mobile browser. Fifth, the application starts using the front camera and goes through the stages "Looking at camera..." and "Verifying Face..."

The state of session changes to Created, Active or Closed. The Created state occurs right after the teacher starts the session; Active is the time during which the QR is shown and scanned; Closed state

comes after the teacher closes the session. When session is closed, the QR and session ID can not be used any more in normal mode, but Mark Present override mode is still accessible.

The design uses five basic data persistence areas, as mentioned in the report, namely, Student, Faculty, Admin, QRSession, and Attendance. The Attendance area is used to tie up a student and a session and holds information about status, mark time, and other related details. The design includes the definition of an additional location data element, which will be used for the geofencing layer.

TABLE I. KEY IMPLEMENTED SYSTEM MODULES

Module	Implemented responsibility
Authentication	Role-based authentication as Student, Teacher, and Admin.
Dynamic QR Generation	Generation of a unique QR code per session based on subject, department, and year.
QR Scanning	Authentication of the ongoing session using the mobile browser.
Face Verification	Detection of live video feed, face verification, and liveness detection.
Attendance Management	Attendance marking and prevention of duplicate session entry.
Admin Dashboard	Creation and filtering of student data based on year and department.
Reports	Preparation of monthly attendance and defaulter report sheets.

IV. SYSTEM DESIGN AND ARCHITECTURE

The final architecture that was implemented has the form of a multi-tiered client-server application. The React.js/HTML5 front-end serves as a top tier and it can be used by users through their mobile and desktop browsers. Node.js/Express back-end performs all the authentication, session management, QR code creation and checking, attendance-related processes, and reports generation. It also communicates with the face verification service.

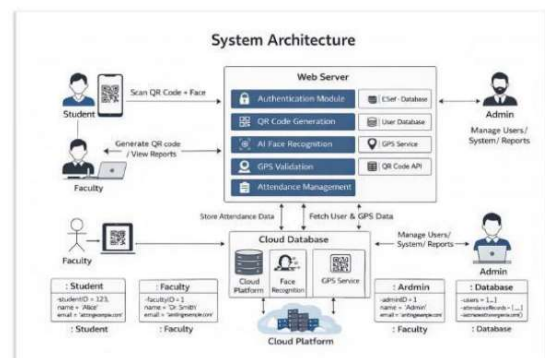


Fig. 1. System Architecture of the Smart Attendance System

The architecture is divided between presentation, logic, and data concerns. The React-based front end

generates Login, Admin, Teacher, and Student screens, along with capturing frames from the cameras for verification. The Express API Gateway handles authentication, routing, business logic, and input validation. The Face Verification service accepts the camera frame data and generates the verification/liveness output. The Reports engine handles attendance data and generates spreadsheets. The Data Store holds the essential data.

IV-A. TEACHER SESSION LIFECYCLE

The steps in the workflow of the Teacher Dashboard that are incorporated into the report include: Start → Log In to System → Choose Department/Year/Subject → Create QR Code → Show Live Session QR Code → Monitor Live Attendance → End Attendance/Download Reports → Stop. The live session panel displays the subject chosen, department, year, start time, session ID, QR code, and session status.

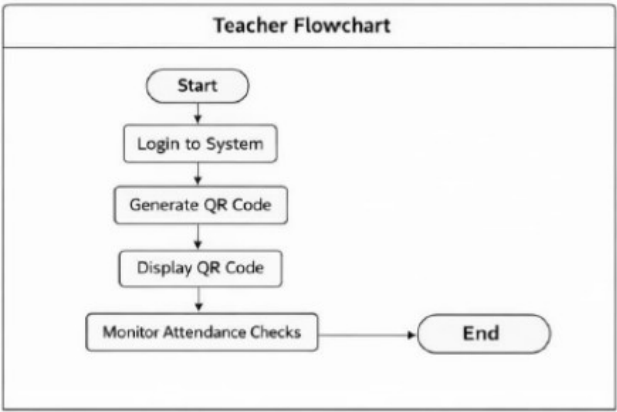


Fig. 2. Teacher Flowchart

IV-B. STUDENT VERIFICATION LIFECYCLE

Student flow has been made sequential. In case the user is logged in, the program verifies if the scanned QR code has any valid session or not. In case there is an invalid session or an inactive session, the transaction gets rejected. Otherwise, the student goes through the process of face capture and the liveness check. Only after that attendance can be recorded.

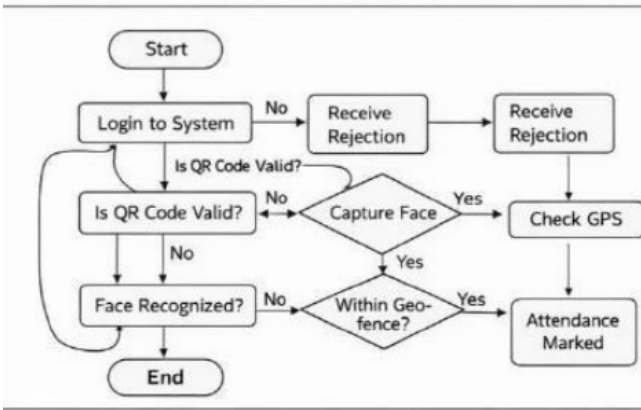


Fig. 5.4: Student Flowchart

Fig. 3. Student Verification Flowchart

This sequence distinguishes between having access to the QR code in the live session and verification of identity. The sole QR code thus does not serve as proof of attendance. This report also indicates that there is a process in place to prevent duplication by checking the student-session pair before scanning.

V. IMPLEMENTATION

The front end has been developed using the React.js, HTML5, CSS3 and JavaScript. This part of the program offers the role based login page, Admin dashboard, Teacher dashboard and Student dashboard. The Student flow that is mobile first allows the use of QR code scanning and camera, while Admin and Teacher flows that are desktop based allow the use of forms, tables, sessions and actions.

The backend has been created using the Node.js and Express.js. The report discusses a REST API which is consumed by the React frontend. The authentication process provides a session token, and the remaining processes that are restricted to certain roles require the authentication. The creation of a session creates a QRSession with Subject, Department, Year, Start Time and ID.

V-A. KEY API INTERFACES

Endpoint	Caller	Purpose
POST /api/auth/login	All roles	Role authentication and session token.
POST /api/sessions/start	Teacher	Creates QRSession and returns QR payload.
POST /api/sessions/stop	Teacher	Closes the active session.
GET /api/sessions/id	Student/Teacher	Returns session details and status.
POST /api/attendance/scan	Student	Validates session and starts face step.
POST /api/attendance/verify-face	Student	Submits camera frame for face/liveness check.
POST /api/attendance/mark	System	Records verified student/session attendance.
POST /api/attendance/override	Teacher	Manual Mark Present action.
GET /api/attendance/live/sessionId	Teacher	Returns live attendance list.
POST /api/students	Admin	Adds a student record.
GET /api/students	Admin	Filters student list by Year and Department.
GET /api/reports/monthly	Teacher	Creates monthly attendance spreadsheet.
GET /api/reports/defaulters	Teacher	Creates defaulter spreadsheet.

V-B. DATABASE AND SECURITY

The report identifies five main tables namely Student, Faculty, Admin, QRSession and Attendance. Student table contains PRN number, student name, email id, password hash, phone number, year, department, division, optional gender field, and path to picture. Similarly, the Faculty and Admin tables contain role-specific credentials. QRSession contains sessionID, facultyID, course name, QR payload, start time, and end time. Table Attendance contains attendanceID, studentID, sessionID, status, marktime, location ID and creation timestamp.

Security is incorporated on several layers. HTTPS is employed for the deployed application to prevent transmission of credentials, camera frames, and session tokens in plaintext form. Passwords are hashed. Authorization is performed by role-based methods preventing access of Admin methods by Student account. Environmental variables store configuration values like DATABASE_URL, JWT_SECRET, SESSION_QR_EXPIRY and DEFAULTER_THRESHOLD.

VI. DEPLOYMENT AND USER WORKFLOW

The application is designed such that there is a login page with tabs for Teacher, Student and Admin roles. The Admin portal enables one to add student details as well as filter the list of students using the Year and Department tabs. The Teacher portal begins the session, shows the QR code and reports on the attendance. The Student portal shows the profile details as well as Scan QR for Attendance and View Attendance.

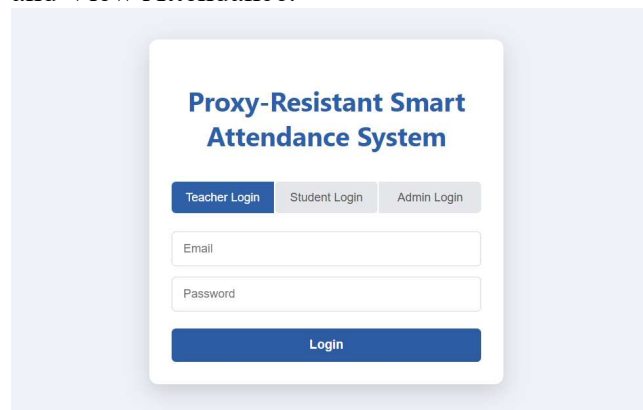


Fig. 4. Role-Based Login Screen from the Deployed System

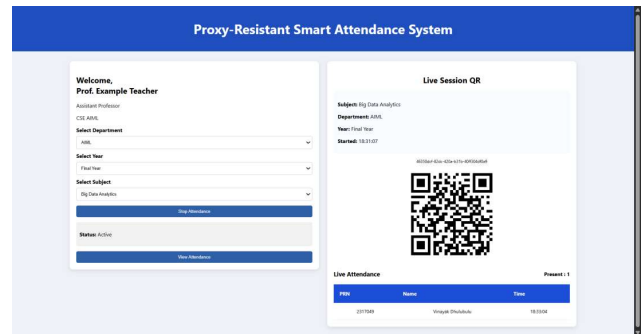


Fig. 5. Teacher Dashboard with Live Session QR Generation

The Teacher Dashboard shows the selected context of the class as well as the QR code currently being used. In the test run carried out on September 10, 2026, the Big Data Analytics class was captured between 18:31:07 and 18:33:19.

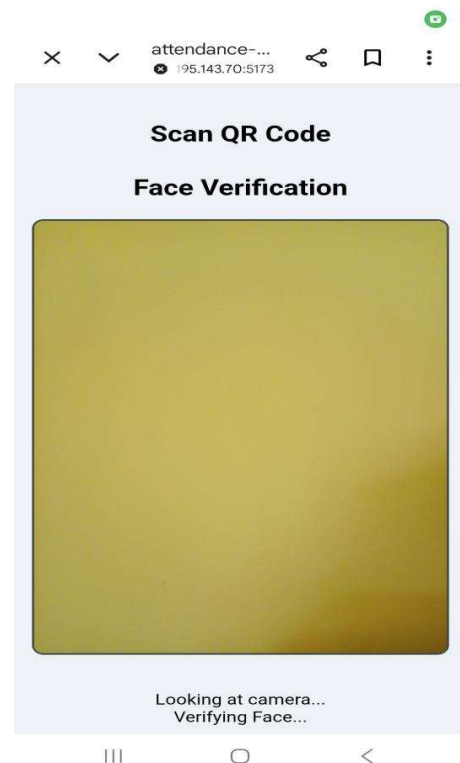


Fig. 6. Mobile Face Verification and Liveness Prompt

As soon as the scan through QR code is done, the Student application uses the front camera and shows “Looking at camera...” then “Verifying face...”. If the face is verified successfully, attendance will be registered for the active session. In the report, it was mentioned that permission for camera use and sufficient light are mandatory.

VI-A. ATTENDANCE ANALYTICS AND MANAGEMENT

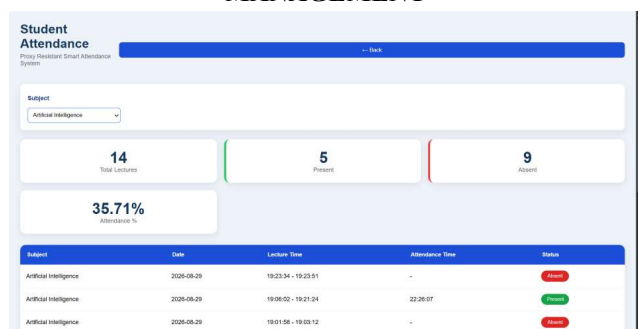


Fig. 7. Student Dashboard — Subject-wise Attendance Analytics

Analysis on the attendance is done subject-wise using the Student Dashboard. The document outlines an analysis of one test case where the Attendance Analysis for the Artificial Intelligence subject indicated that the number of lectures was 14, while the number of times present was 5, not present 9 and attendance percentage 35.71%.

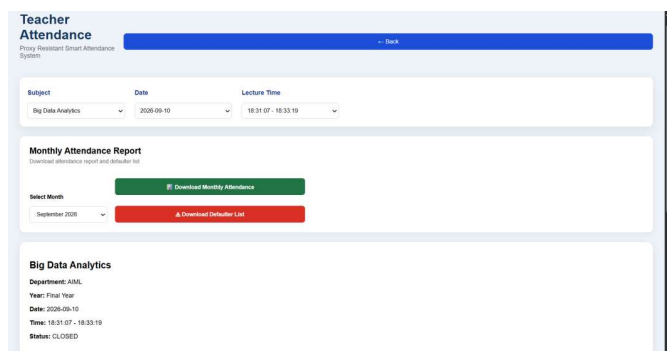


Fig. 8. Teacher Attendance Dashboard and Report Controls

In the Teacher Attendance Screen, teachers have the option to sort the attendance by subject, date, and lecture time. The Monthly Report section consists of activities relating to the Monthly Attendance Report and the Defaulter List. Upon selecting a session, the class list with the number of total students, present students, and absent students pops up along with the activity "Mark Present".

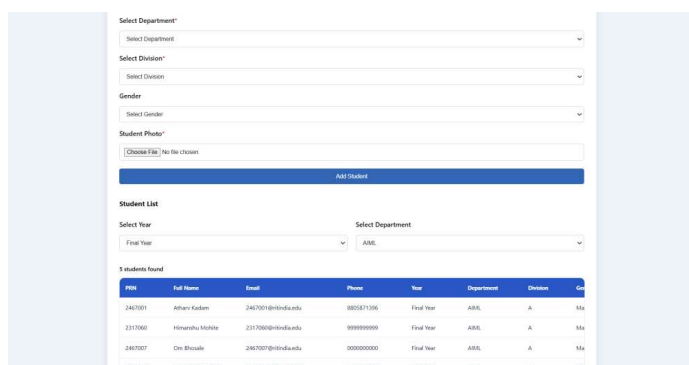


Fig. 9. Admin Student Management and Filtering Interface

Admin Form contains PRN, Full Name, Email, Password, Phone, Year, Department, Division,

Gender, and Photo. Student List is searchable on the basis of Year and Department. The report demonstrates that the filtering of students by the combination of Final Year and AIML gave matched results.

VII. TESTING AND VALIDATION

Integration testing of the combination of manual functional testing, user acceptance testing (role based UAT), and integration testing through Admin, Teacher and Student portals. The testing was carried out on desktop browsers for the Admin and Teacher modules and Android mobile browser for the Student QR-scan & Face Verification module. Seed data for the AIML department, Final year batch and subjects such as Artificial Intelligence and Big Data Analytics have been provided.

VII-A. FUNCTIONAL TESTING

Test	Expected / observed result	Status
TC-01: Role login	Students, Teachers and Admin log into the same login screen.	Pass
TC-02: Start session	The Teacher creates a unique session id and QR for the selected class.	Pass
TC-03: QR scan	The Student's mobile browser detects an active session QR.	Pass
TC-04: Face verification	The Camera takes a picture using the face liveness recognition.	Pass
TC-05: Attendance mark	An attendance entry gets created once the authentication succeeds.	Pass
TC-07: Manual override	Teachers may use Mark Present for a valid excuse.	Pass
TC-08: Add student	An Admin-created student gets listed in the Students List.	Pass
TC-09: Filter students	Filters on Year and Department sort out the students list.	Pass
TC-10: Student analytics	The subject tab displays total, present, absent, and percent.	Pass
TC-11/12: Reports	Defaulter and Monthly spreadsheet get generated correctly.	Pass

VII-B. INTEGRATION TESTING

Integration	Expected behaviour	Result
QR Generation → QR Scanning	Teacher-created session will appear on Student scanning screen.	Pass
QR Scanning → Face Verification	Valid scanned session hands over correct session context to Face Verification.	Pass
Face Verification → Attendance	Successful face verification will result in the creation of the Attendance record.	Pass
Attendance → Live Dashboard	Student verified is added to the Teacher live list.	Pass
Attendance → Reports	Attendance records are exported into monthly/defaulter reports.	Pass

VII-C. DEFECTS AND MITIGATIONS

This report provides a limited number of problems encountered during the process of implementation and their solutions. A problem with the live attendance refresh not updating in case of slow Internet connection has been solved using a short polling interval. Defaulter list calculations initially considered students with zero number of lectures; the solution to this problem was excluding such records from the list. The face verification could freeze in case of too dim light.

VIII. RESULTS AND PERFORMANCE

All seven modules specified in the SRS were successfully functional and tested via actual accounts from the AIML department and FY batch by the time Capstone Phase II ended. The application produced unique QR codes for sessions of Artificial Intelligence and Big Data Analytics, made students pass the process of face verification before marking attendance, updated the dashboard for faculty members, and provided monthly and defaulters' reports.

VIII-A. OBSERVED END-TO-END RESULTS

For the Big Data Analytics session conducted on 2026-09-10 between 18:31:07 and 18:33:19, the report identifies five students in total. Out of these five, one student was confirmed to be present using the QR + Face process at 18:33:04, and the rest of the four were able to be checked and corrected using the Mark Present override feature.

In the student perspective, the Artificial Intelligence dashboard under the same test account had a total number of 14 lectures, out of which 5 were attended, 9 were missed and the percentage attendance was 35.71%. In the departmental perspective, the August 2026 Monthly Attendance Report & Defaulter Attendance List had a total of six students in Final Year & Third Year AIML.

Monthly Attendance Report - 2026-08							
Teacher: Prof. Example Teacher							
PRN	Student Name	Branch	Year	Total Lectures	Present	Absent	Attendance %
2467001	Atharv Kadam	AIML	Final Year	18	3	15	16.67%
2317060	Himanshu Mohite	AIML	Final Year	18	0	18	0.00%
2467007	Om Bhosale	AIML	Final Year	18	3	15	16.67%
2317047	Prathmesh Salunke	AIML	Final Year	18	0	18	0.00%
2317049	Vinayak Dhulubulu	AIML	Final Year	18	9	9	50.00%
2417001	Rahul Patil	AIML	Third Year	1	0	1	0.00%

Fig. 10. Exported Monthly Attendance Report

TABLE II. ACHIEVED VS. EXPECTED PERFORMANCE

Metric	Target	Observed result	Status
End-to-end QR → Face → Marked	< 10 s	Completed within a few seconds per test run	Met
QR generation	< 2 s	Near-instant on Teacher Dashboard	Met
Live attendance update	Real/near-real time	Present count and row appeared promptly	Met
Monthly/Defaulter report	On-demand export	Both generated correctly as spreadsheets	Met
Cross-device usability	Android/iOS + desktop	Android student and desktop Admin/Teacher verified	Met

VIII-B. FEATURE COMPARISON WITH PRIOR WORK

Feature	Completed system evidence
Dynamic session-bound QR	Yes; created per active faculty session.
Face/identity verification	Yes; verification follows QR validation.
Liveness prompt	Yes; mobile screen guides live camera verification.
Location / geo-fencing	Designed in architecture; not yet connected to live scan flow.
Dedicated hardware	Not required; browser-based deployment.
Real-time faculty dashboard	Yes; live attendance panel.
Monthly/defaulter export	Yes; spreadsheet downloads.

IX. LIMITATIONS, COST AND REQUIREMENT TRACEABILITY

IX-A. CURRENT LIMITATIONS

- Geo-fencing by GPS exists at the architectural level but has not yet been integrated into the active flow of students via scanning. The current pipeline uses QR + Face recognition.
- Face recognition is impacted by bad lighting, as noted during some of the test sessions. While the Teacher's ability to manually override the system mitigates such situations, the dependency on lighting conditions still exists.
- The system has been tested only for the AIML department with a small number of concurrently logging students. Multi-department, high concurrency behavior was not load tested yet.
- Bulk upload of student records using a spreadsheet is not currently available; each student has to be added individually using the Admin form.

IX-B. COST-BENEFIT OBSERVATION

According to the project report, it takes around 3–5 minutes to take the manual roll call for each lecture period of around 60 students in a Final Year AIML batch having five to six lectures daily. The total manual calculation and defaulter generation at the end of each month would take an additional 1–2 hours per subject every month. However, with the developed software, attendance registration coincides with the start of the lecture and monthly/defaulter reports can be generated in one go. The hosting fee of the developed system is around ₹3,000 per year.

IX-C. REQUIREMENT TRACEABILITY

Requirement	Implemented in	Evidence
FR-1.1 Role-based login	Authentication module	TC-01, login screen
FR-2.1 Dynamic QR generation	Session start API	TC-02, Teacher QR
FR-3.1 Mobile QR scanning	Student flow	TC-03, mobile scan
FR-4.1 Face verification + liveness	Face verification service	TC-04, face screen
FR-5.1 Attendance after verification	Attendance API/table	TC-05, live attendance
FR-5.2 Manual Mark Present	Override API	TC-07, session attendance
FR-6.1 Add student	Admin module	TC-08, Add Student
FR-6.2 Filter student list	Student API/Admin list	TC-09, filtered list
FR-7.1 Monthly/Defaulter reports	Reports engine	TC-11/12, exported sheets

IX-D. DEPLOYMENT AND COMPATIBILITY

Students' tests were conducted using the Android version of Chrome with permission to use the device's camera and screen size 360px or greater. Testing of the admin and teachers' dashboard was done using Chromium-based desktop browser, with 1366px or more preferred for the dashboard table.

Tests were carried out using a reliable mobile data or Wi-Fi connection. The project also provides details of how students, faculties, admin, QRSession and Attendance data are backed up for re-deployment of the system.

X. CONCLUSION AND FUTURE WORK

Proxy Resistant Real-time QR-Based Smart Attendance System was developed and tested as browser based application. This includes Role Based Authentication, Session Bound QR code generation, Mobile Scan, Live Face Detection with Liveness check, Attendance Management, Dashboards and Report Generation in spreadsheets. Thus the project provides end-to-end solution to proxy attendance without need of mobile application.

Future Work includes integration of GPS Geo-fencing, Anomaly detection for attendance, Push notifications for students who cross the threshold, Large scale pilot test in multiple departments, High concurrency validation and integration with the institute ERP or academic management system. Other improvements suggested by feedback include Bulk Import of students and Countdown timer in session.

Other than technical output, this project provided practical experience of requirements gathering, SRS Preparation, Design using UML/DFD, Full Stack implementation in React and Node.js/Express along with integration of AI Service and Testing, Deployment and Collaboration using Git.

X-A. FUTURE WORK

- Incorporate the use of GPS geo-fencing into the real-time scanning process to make location an extra condition for verification purposes.
- Include anomaly detection to identify any strange attendance behavior, allowing for early analysis of the questionable attendance records.
- Implement the use of push notifications in case a student has less than the stated level of attendance.
- Conduct validation through pilots in larger departments and sessions with high concurrency.
- Incorporate the use of bulk student importation, a count down on sessions, and integration into an institute's ERP or academic management system.
- Continue with face-verification improvement depending on lighting conditions and camera.

REFERENCES

- [1] S. Kumari and A. Kumar, "QR-Based Attendance System with Geo-Location Verification," *International Journal of Advanced Research in Information Technology (IJARIIT)*, vol. 10, no. 2, 2022.
- [2] R. Patil, P. Deshmukh, and S. Jadhav, "Real-Time Attendance System Using QR Code and Face Recognition," *The International Journal of Engineering Research (TIJER)*, vol. 11, no. 3, 2023.
- [3] A. Shinde, V. Pawar, and S. Kulkarni, "Smart Attendance Control System Using GPS and QR Code," *IEEE International Conference on Smart Computing Technologies*, pp. 1–7, 2023.
- [4] M. Rathod and D. Patil, "Review of QR Code and Biometric Based Attendance Systems," *IJARSCCT*, vol. 3, no. 2, pp. 112–118, 2023.
- [5] N. Gupta, R. Verma, and A. Singh, "QR Code Based Attendance System: Challenges and Solutions," *IJERT*, vol. 11, no. 5, pp. 234–239, 2022.
- [6] S. Kadam, P. Chavan, and A. More, "Contactless Smart Attendance System Using Facial Recognition and QR Code," *IJIRT*, vol. 10, no. 4, pp. 428–433, 2023.
- [7] H. Patel, R. Shah, and K. Mehta, "Mobile Based Student Attendance System Using Geo-Fencing and Face Recognition," *International Journal of Computer Applications*, vol. 174, no. 8, 2021.
- [8] H. S. Al-Khalifa, "QR Code-Based Reliable Attendance Management System," *Engineering Faculty Research Journal*, vol. 1, no. 4, pp. 45–52, 2022.
- [9] K. Sharma and P. Yadav, "Smart Attendance System Using Face Recognition Techniques," *IJISRT*, vol. 7, no. 3, pp. 601–606, 2022.
- [10] R. Soni and A. Malhotra, "QR-Based Attendance Management System with Location Tracking," *IJIRCCE*, vol. 11, no. 8, pp. 4561–4568, 2023.
- [11] P. Kumar, S. Singh, and R. Jain, "Automated Attendance System Using Face Recognition and GPS Localization," *International Journal of Intelligent Engineering and Systems*, vol. 15, no. 2, 2022.
- [12] A. Raut, V. Patil, and S. Desai, "Dynamic QR Code-Based Attendance System for Educational Institutions," *HBRP Journal of Advances in Science and Technology*, vol. 6, no. 2, pp. 44–50, 2023.
- [13] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," *IEEE CVPR*, pp. 815–823, 2015.
- [14] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint Face Detection and Alignment Using Multi-task Cascaded Convolutional Networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.
- [15] P. Reddy and S. Rao, "Survey on Automatic Attendance Systems Using Biometrics and QR Codes,"

- Journal of Advanced Research in Engineering and Technology, vol. 11, no. 4, pp. 120–128, 2020.
- [16] H. Li, J. Xu, and T. Chen, “Deep Learning Based Facial Recognition for Classroom Attendance,” Journal of Intelligent Learning Systems and Applications, vol. 15, no. 7, pp. 51–60, 2023.
- [17] A. Singh and R. Kumar, “Real-Time Location Verification Techniques for Mobile Attendance Systems,” International Journal of Mobile Computing and Multimedia Communications, vol. 12, no. 3, pp. 21–34, 2021.
- [18] F. Ahmed and S. Rahman, “Anomaly Detection in Automated Attendance Systems Using Machine Learning,” International Journal of Computer Applications, vol. 182, no. 12, pp. 27–34, 2018.