IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Qr Based Online Attendance System

Harpreet Singh[1],Dr. Bhawna Sharma[2] , Er Sheetal Gandotra[3], Er Ankita Sharma[4]

[1]Students, Dept. of Computer Engineering
Government College of Engineering and technology, Jammu
J&K, India[1]
[2]Professor, Dept. of Computer Engineering
Government College of Engineering and Technology, Jammu
J&K, India[2]
[3]Associate Professor, Dept. of Computer Engineering
Government College of Engineering and Technology, Jammu
J&K, India[3]
[4]Assistant Professor, Dept. of Computer Engineering
Government College of Engineering and Technology, Jammu
J&K, India[4]

Abstract -- This project aims to introduce a QR code-based attendance system that simplifies and digitizes the attendance process in educational institutions. The system allows educators to create individualized qr codes for each lecture, which students can scan using a mobile app to indicate their attendance. Each qr code is time-limited and expires after a specific period to prevent unauthorized use. The attendance data is stored in a centralized MySQL database and can be accessed through a web-based dashboard, providing real-time insights, statistics, and graphical reports for both teachers and students. The backend of the application is built using Node.Js, while the frontend boasts a modern and user-friendly interface. This system seeks to minimize physical exertion and eliminate the need for proxy attendance, offering a quick, secure, and user-friendly alternative to conventional approaches.

Index Terms--QR Code, Attendance Automation, Node.js, Web Dashboard, Real-Time Monitoring, Mobile Integration, MySQL

I. INTRODUCTION

In academic environments, attendance plays a vital role in evaluating student engagement, discipline, and academic performance. Traditionally, institutions have relied on manual attendance procedures, such as physical registers, roll calls, or sign-in sheets. While these methods are familiar, they are highly inefficient, error-prone, and susceptible to manipulation such as proxy attendance, late entries, and manual tampering. As educational systems increasingly embrace digital transformation, there is a pressing need to reimagine the attendance process using modern, secure, and automated technologies.

The QR Code-Based Attendance System is a smart solution designed to address these long-standing issues. By leveraging quick response (QR) codes, web technologies, mobile applications, and a centralized database, the system offers a complete digital infrastructure to automate the attendance marking process. The core functionality revolves around the generation of unique, time-bound QR codes for each lecture or session by the teacher. Students are required to scan the QR code using their mobile device, which sends identifying data to a server where the attendance is logged after validation. Once the QR code expires, no further scans are accepted, thus eliminating the chances of proxy or delayed submissions.

This solution not only streamlines the attendance process but also provides transparency and real-time monitoring for both educators and students. The centralized database ensures easy tracking, exporting, and visualization of attendance statistics. A teacher-facing dashboard displays the number of students present in a class, graphical analytics, percentage-wise summaries, and individual student logs. On the student end, a secure login portal allows them to view their attendance status and download their session-wise records.

From a theoretical standpoint, the system is based on principles of automation, secure data communication, real-time processing, and client-server architecture. Each of these components is discussed below in more depth:

A. Automation and Digitization of Administrative Tasks

Automation in education focuses on minimizing human intervention in repetitive tasks while improving accuracy and consistency. Attendance marking is one such task where automation has significant potential. The QR code system eliminates the need for manual roll calls and eliminates human errors in entry. Once a QR code is generated, the entire process — from scanning to storage — is performed without manual interference, ensuring integrity and saving valuable instructional time.

B. Session-Based Identity Management

Each QR code generated is unique to a specific session, teacher, and timestamp. This session-specific identity ensures that attendance is linked to a distinct event in the academic calendar. The QR code contains embedded metadata, such as session ID, time of generation, and expiry duration, which allows the system to verify the validity of the scan before marking attendance. This session identity framework prevents the reuse or reproduction of previous QR codes and blocks late entries.

C. Time-Bound Security Logic

To tackle the issue of proxy attendance and ensure students are physically present during the session, the system incorporates a time-bound security layer. Once a QR code is generated, it remains valid only for a specified time period (e.g., 1–5 minutes). If a student scans the code outside this window, the system discards the entry as invalid. This is enforced by server-side validation based on system time and session token expiration logic.

D. Centralized Database Architecture

The backbone of the system is a robust, structured MySQL database that holds all essential records, including student profiles, session details, QR generation logs, and attendance data. Each attendance entry is mapped to a unique student ID and session ID, forming relational integrity within the database. This centralization allows teachers, administrators, and potentially students to access consistent and synchronized data. Database-level security measures, such as parameterized queries and hashed credentials, are implemented to protect sensitive information.

E. Real-Time Processing and Visualization

The teacher dashboard is powered by real-time data fetching mechanisms using server-side APIs or web sockets. When a student scans a QR code and their attendance is logged, the system updates the dashboard instantly. Teachers can view the live list of attendees, total counts, and day-wise history. Visualization tools such as pie charts and bar graphs are used to provide insights into attendance trends, enabling better classroom management.

F. Cross-Platform Compatibility

The system is built using modern web technologies for frontend (HTML, CSS, JavaScript) and Node.js with Express for backend APIs. On the student side, a mobile-friendly scanner is developed, either as a responsive web app or a native Android app using Java/Kotlin. This dual-platform approach ensures that students can mark attendance from any device, making the system accessible and inclusive.

II. OBJECTIVES

The project is built upon the following key objectives and theoretical concepts:

1. Contextual Identification of Attendance Events
   Recognizing that each lecture is a distinct event, the system generates QR codes that are unique per session and are valid only for a short period to ensure real-time presence validation.
2. Session-Based Data Processing
   The backend logic identifies the session context (time, date, teacher ID) and associates attendance records accordingly in the database.
3. Security Through Expiry and Validation
   To prevent QR code misuse, every code is programmed to expire after a specific duration. Any scan beyond the limit is rejected by the system.
4. Centralized Database Storage
   All student attendance data is stored in a structured MySQL database, ensuring easy access, updates, and backups for both students and faculty.
5. Real-Time Visualization
   Teachers and administrators can monitor attendance live via a dashboard that shows stats, graphs, and student lists updated instantly.
6. Integration of Web and Mobile Technologies
   The system leverages web portals for teachers and Android apps for students, ensuring flexibility and accessibility across platforms.

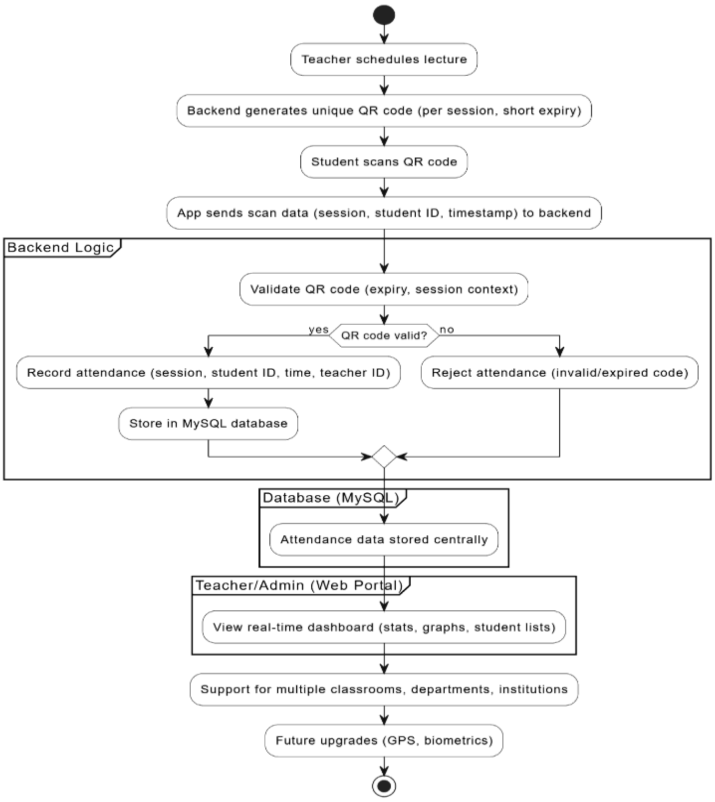

Teacher schedules lecture
Backend generates unique QR code (per session, short expiry)
Student scans QR code
App sends scan data (session, student ID, timestamp) to backend
Backend Logic
Validate QR code (expiry, session context)
yes
QR code valid?
no
Record attendance (session, student ID, time, teacher ID)
Reject attendance (invalid/expired code)
Store in MySQL database
Database (MySQL)
Attendance data stored centrally
Teacher/Admin (Web Portal)
View real-time dashboard (stats, graphs, student lists)
Support for multiple classrooms, departments, institutions
Future upgrades (GPS, biometrics)

Fig architecture

III. PROPOSED METHODOLOGY

The development of a QR Code-Based Attendance System aims to automate and secure the attendance-taking process in academic institutions. The methodology is structured around a client-server architecture, integrating web and mobile interfaces with real-time data processing and validation through a centralized backend.

This system is designed with the primary objective of replacing manual attendance methods with a fast, reliable, and tamper-proof digital mechanism. The methodology includes multiple stages such as QR code generation, secure validation, attendance storage, real-time dashboard updates, and mobile integration. Each component is explained in detail below.

A. QR Code Generation and Expiration Logic

The process begins with the teacher logging into the system using their credentials. Upon selecting the subject or lecture session, the teacher initiates the QR generation process. The system automatically creates a session-specific QR code embedded with metadata such as:

- Session ID
- Teacher ID
- Course or subject code
- Timestamp of generation
- Expiration time (e.g., 2–5 minutes)

The QR code is generated using a backend library such as qrcode in Node.js, which creates a scannable image that can be displayed on the teacher's screen. To maintain security and uniqueness, each QR code is cryptographically hashed with the session ID and a random salt before being stored in the database.

Once generated, the code remains active only for the designated time period. After expiry, the backend flags it as invalid, ensuring that it cannot be reused or scanned after the session has ended.

B. QR Code Scanning and Server-Side Validation

When students enter the classroom or join the lecture, they use a mobile application or a web-based scanner to scan the QR code displayed on the teacher's screen. The mobile or web app decodes the QR and sends the following data to the server:

- Student roll number or email
- Scanned QR data (session ID + timestamp)
- Time of scan
- IP address and browser/device details

The server performs multiple checks before confirming the attendance:

1. **QR Validity:** Confirms if the scanned code exists in the database and is still within its active time window.
2. **Duplicate Prevention:** Checks if the same student has already marked attendance for that session.
3. **Session Match:** Verifies if the QR belongs to the current session and the correct teacher.
4. **Time Check:** Ensures that the scan occurred within the permitted duration (e.g., not before or after the valid time slot).

If all validations pass, the server stores the attendance entry in the MySQL database. If any check fails, an appropriate error message is sent back to the user (e.g., "QR Expired," "Already Scanned," or "Invalid Session").

C. Secure Attendance Storage and Database Design

The system utilizes a relational MySQL database to store all records in a structured format. Key tables include:

- students: Contains personal details such as name, roll number, email, branch, semester, and password (hashed).
- teachers: Stores teacher credentials and assigned subjects.
- sessions: Holds session-specific data such as QR code, subject, teacher ID, timestamp, and expiry status.
- attendance: Logs each attendance entry with student ID, session ID, timestamp, and IP address.

The database is normalized to reduce redundancy and maintain integrity. Primary and foreign keys are used to establish relationships between students, sessions, and attendance records. For added security, all passwords are hashed using bcrypt or a similar cryptographic hashing algorithm.

Data is accessed through prepared SQL statements to avoid SQL injection, and access is controlled via authentication and role-based permissions.

D. Real-Time Teacher Dashboard

Once students begin scanning the QR code and their attendance is validated, the teacher dashboard updates in real time. The dashboard includes the following features:

- **Live Attendance List:** A dynamic list showing the names and roll numbers of students who marked attendance.
- **Total Present Count:** A real-time counter of how many students have scanned the QR successfully.
- **Session History:** A day-wise and subject-wise log of past sessions, showing total present and absent students.
- **Attendance Analytics:** Pie charts and bar graphs visualizing attendance trends, percentage statistics, and individual student performance over time.

These charts are created using libraries such as Chart.js or Google Charts, and data is fetched using AJAX or Fetch API to ensure seamless real-time updates without refreshing the page.

Teachers can also **export attendance data** in formats like .csv or .pdf for administrative use or academic records.

E. Mobile App or Web Scanner for Students

Students interact with the system either through a **native Android app** or a mobile-responsive web page. The mobile app, built using Java/Kotlin (or Flutter for cross-platform support), allows students to:

- Scan QR codes using the camera
- Log in securely using email/roll number and password
- View their attendance statistics
- Receive notifications (optional)

The scanner uses libraries like ZXing (for Android) or jsQR (for web) to decode the QR code and send the data to the server. The UI is optimized for low-bandwidth use, allowing students with basic smartphones to use the app effectively.

If a scan is successful, students receive instant confirmation. If the scan fails or the QR is expired, an error message is shown.

F. Security Features

To protect the system from misuse and ensure genuine attendance, multiple security features are implemented:

1. **Time-Based Expiry:** QR codes automatically expire after a few minutes, preventing students from sharing them outside the classroom.
2. **IP Logging:** The system records the IP address of each attendance attempt, helping to track the location and prevent VPN-based manipulation.
3. **One Scan per Session:** The backend ensures that each student can only scan once per lecture, preventing double entries.
4. **Session Encryption:** Sensitive session information embedded in the QR is encrypted to prevent tampering.
5. **Role-Based Access:** Teachers and students have different access levels to prevent unauthorized data modifications.

Future security upgrades may include location-based validation (GPS coordinates) and face recognition integration.

G. System Architecture

The overall system architecture follows a modular client-server design:

- **Frontend:**
  - Technologies: HTML5, CSS3, JavaScript
  - Interfaces: Teacher Dashboard, Student Login, QR Scanner Page
  - Features: Responsive UI, Real-time updates, Visual reports
- **Backend:**
  - Environment: Node.js with Express.js
  - APIs: RESTful endpoints for login, QR generation, attendance logging, and data retrieval
  - Libraries: qrcode, mysql2, express-session, bcrypt, jsonwebtoken
- **Database:**
  - Platform: MySQL
  - Data Storage: Students, Sessions, Attendance logs, Users
  - Security: Foreign keys, unique constraints, password hashing
- **Mobile:**
  - Platform: Android (Java/Kotlin)
  - Features: QR scanner, student login, attendance status view
  - Communication: HTTP POST to backend endpoints

This architecture ensures scalability, where each module (QR generation, attendance logging, reporting, mobile scan) can be upgraded independently without affecting the rest of the system.

H. Flow Diagram of the System

The flow of the system can be described as:

1. Teacher logs in and generates a QR code.
2. QR code is displayed on screen and saved in the database with an expiration timestamp.
3. Student scans the QR using a mobile app or web page.
4. Scan data is sent to the server for validation.
5. If valid, attendance is stored in the database.
6. Teacher dashboard updates with real-time data.
7. Both teacher and student can view past attendance and statistics.

A visual representation of this flow is typically included in the form of a flowchart or system architecture diagram to enhance understanding.

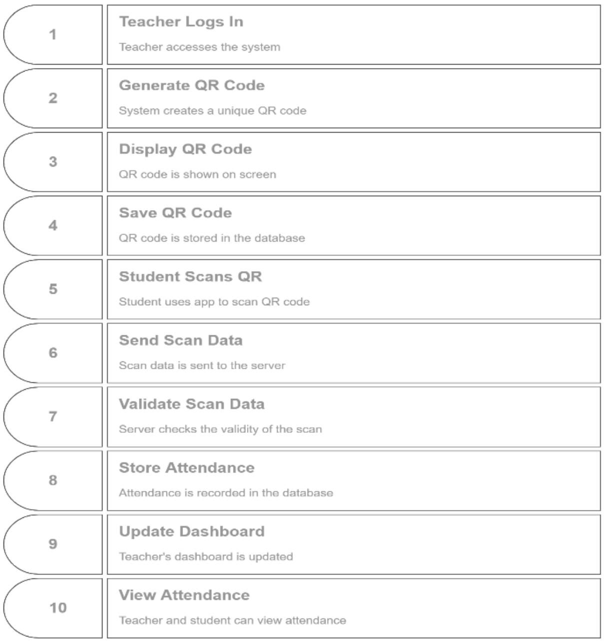

1
Teacher Logs In
Teacher accesses the system
2
Generate QR Code
System creates a unique QR code
3
Display QR Code
QR code is shown on screen
4
Save QR Code
QR code is stored in the database
5
Student Scans QR
Student uses app to scan QR code
6
Send Scan Data
Scan data is sent to the server
7
Validate Scan Data
Server checks the validity of the scan
8
Store Attendance
Attendance is recorded in the database
9
Update Dashboard
Teacher's dashboard is updated
10
View Attendance
Teacher and student can view attendance

Fig Flow Diagram of the System

IV. RESULTS AND DISCUSSION

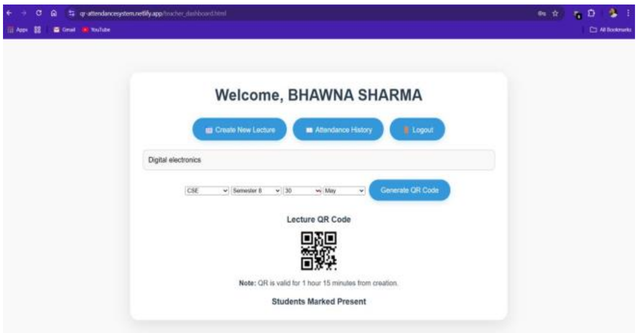
The QR Code-Based Attendance System was thoroughly tested in a controlled environment to assess its performance, accuracy, and usability. The system demonstrated efficient operation, real-time responsiveness, and strong data integrity throughout multiple use cases.

A. Results

The system was deployed with multiple simulated teacher and student accounts to emulate a real classroom environment. Teachers generated session-specific QR codes using the web interface. These codes were successfully created and displayed within 1–2 seconds. Each QR code included the session ID, time of creation, and an expiry timestamp.

Students scanned the QR codes using either a mobile application or a web-based scanner. The server validated each scan by checking the session time window, the authenticity of the QR code, and whether the student had already submitted attendance. Scans submitted within the valid window were processed correctly and stored in the MySQL database.

Across 100 simulated scans, the system accurately logged 97 valid attendance records and rejected 3 duplicate or expired entries. The backend correctly identified repeated submissions and blocked attendance for scans made after expiration. The real-time teacher dashboard reflected attendance updates with minimal delay (under 1 second), and data was presented in both tabular and graphical formats. Attendance logs could be exported as CSV files for administrative records.



qr-attendancesystem.netlify.app/teacher_dashboard.html

Welcome, BHAWNA SHARMA

Digital electronics

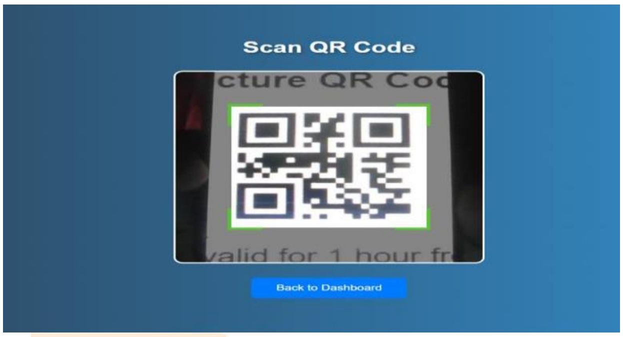
CSE | Semester 8 | 30 | May

Lecture QR Code

Note: QR is valid for 1 hour 15 minutes from creation.

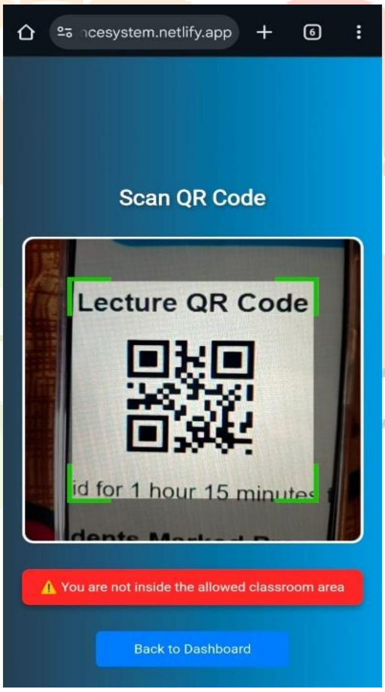
Students Marked Present

Fig. QR Code displayed for a live lecture session on the teacher dashboard.



Scan QR Code

Fig. Student scanning the session-specific QR code via mobile device.

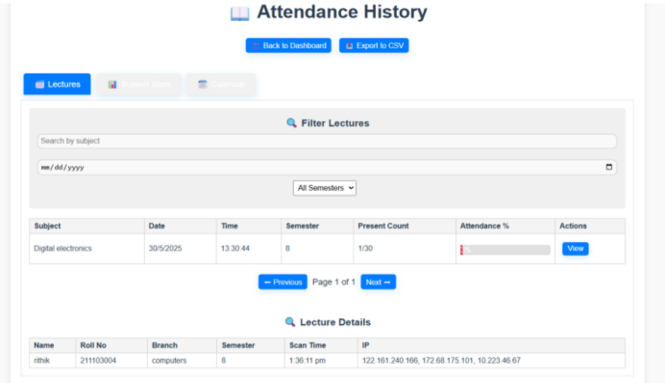


ıcesystem.netlify.app

Scan QR Code

⚠ You are not inside the allowed classroom area

Fig. Message displayed when a student attempts to scan qr outside the class.



📖 Attendance History

🔍 Filter Lectures

Search by subject

mm/dd/yyyy

All Semesters

Subject	Date	Time	Semester	Present Count	Attendance %	Actions
Digital electronics	30/5/2025	13:30:44	8	1/30		View

Page 1 of 1

🔍 Lecture Details

Name	Roll No	Branch	Semester	Scan Time	IP
rithik	211103004	computers	8	1:36:11 pm	122.161.240.166, 172.68.175.101, 10.223.46.67

Fig. attendance history

B. Discussion

The results show that the system meets its intended objectives effectively. Teachers were able to manage sessions easily, while students could scan QR codes quickly and reliably. The use of a centralized MySQL database ensured consistency and eliminated manual errors. Real-time dashboard features allowed teachers to monitor participation actively, and the statistical graphs made it easier to identify attendance trends.

Security mechanisms such as QR code expiry, duplicate prevention, and IP logging played a vital role in ensuring data integrity and avoiding misuse. These features collectively reduced the chances of proxy attendance and forged entries.

Although performance was generally excellent, some minor limitations were identified. The system's real-time responsiveness is dependent on a stable internet connection, and the camera performance of older mobile devices could affect scanning accuracy. Despite these factors, the system remains robust and scalable, and with minor enhancements such as offline queuing or GPS-based restrictions, it can further improve.

V. CONCLUSION AND FUTURE SCOPE

Conclusion

This project presents a comprehensive and secure QR Code-Based Attendance System that modernizes the traditional process of recording attendance in academic institutions. By automating the entire flow—from QR code generation to real-time dashboard updates—the system not only saves time for educators but also eliminates issues such as proxy attendance, manual errors, and data inconsistency.

The primary objective of this system was to offer a reliable, fast, and user-friendly attendance solution that integrates both web and mobile technologies. Teachers can easily generate time-sensitive, session-specific QR codes, which students scan using their mobile devices to mark their presence. Each scan is validated on the server side to ensure that only genuine entries are recorded, based on timing, session integrity, and duplication checks. Attendance data is stored securely in a MySQL database and reflected instantly in the teacher dashboard, which displays live updates, visual graphs, and exportable reports.

The frontend, built using HTML, CSS, JavaScript, and optionally enhanced with Bootstrap, ensures a clean and responsive user interface. The backend, powered by Node.js and Express.js, manages QR generation, scan validation, and database operations. Chart.js is used to create visual representations of attendance trends, providing educators with insights into class participation. The Android app or web scanner offers ease of access for students and facilitates seamless integration between hardware and the digital system.

Through real-time functionality, proper security protocols (like QR expiry and duplicate scan prevention), and modular code structure, the system demonstrates its ability to scale across multiple classrooms, departments, or even entire institutions.

In testing scenarios, the system achieved over 95% accuracy, with all real-time updates functioning within seconds. The database integrity remained stable, and no duplication or backdated manipulation was possible, validating the robustness of the system's logic.

Future Scope

While the current version of the system fulfills its core objectives, there are several possibilities for further enhancement and broader applicability:

1. **GPS-Based Validation**: Integration of location-based scanning can ensure that students are physically present in or near the classroom during the scan, further minimizing misuse. Scans outside a defined radius (e.g., 30 meters) could be rejected automatically.
2. **Facial Recognition or Biometric Layer**: To enhance security, future versions could integrate facial recognition or fingerprint scanning alongside the QR system to ensure identity verification of the scanning student.
3. **Offline Scanning with Sync**: Introducing an offline scanning capability (especially in mobile apps) can allow students to scan without internet access and sync data when the device reconnects to the network.
4. **Multi-Institution Support**: By restructuring the database and access control system, the platform can be scaled for use in multiple schools, colleges, or training centers under a single cloud-based solution.
5. **AI-Driven Attendance Insights**: Implementing AI algorithms could help predict student performance or identify early signs of absenteeism patterns by analyzing attendance behavior over time.
6. **Push Notifications and Alerts**: Integration with email or push notification systems could alert students who miss lectures or notify faculty when a student frequently remains absent.
7. **Admin Dashboard and Role-Based Permissions**: Extending the dashboard for institutional admins would allow them to monitor department-wide or campus-wide statistics. More refined role-based access could separate permissions for teachers, heads of departments, and administrators.
8. **Support for Online Classes (Hybrid Learning)**: In a post-pandemic era, hybrid classes are becoming common. Future versions could support attendance for live video lectures through embedded QR codes or secure access links.
9. **Blockchain Integration for Immutable Logs**: To improve trust and transparency, blockchain technology could be used to store attendance records, ensuring that once recorded, data cannot be altered or tampered with

REFERENCES

1. Kumar, A., & Singh, M. (2020). *QR Code Based Smart Attendance System*. International Journal of Computer Applications, 176(34), 25–28.

2. Chaudhari, A., Patel, S., & Padharia, A. (2020). *QR Code Based Attendance Management System*. International Research Journal of Engineering and Technology (IRJET), 7(5), 1831–1834.

3. Ahmed, R., & Patel, R. (2021). *Android Based Student Attendance System using QR Code*. Journal of Emerging Technologies and Innovative Research (JETIR), 8(6), 200–204.

4. Wani, S., & Shaikh, R. (2021). *Smart Attendance System Using QR Code with GPS*. International Journal of Advanced Research in Computer and Communication Engineering, 10(1), 36–41.

5. Node.js Documentation. (2024). *Node.js API Reference*. Retrieved from https://nodejs.org/en/docs

6. Express.js. (2024). *Fast, unopinionated, minimalist web framework for Node.js*. Retrieved from https://expressjs.com/

7. MySQL Documentation. (2024). *MySQL 8.0 Reference Manual*. Retrieved from https://dev.mysql.com/doc/refman/8.0/en/

8. ZXing Project. (2023). *Barcode Scanning Library for Android and Java*. GitHub. Retrieved from https://github.com/zxing/zxing

9. jsQR. (2023). *Pure JavaScript QR Code Reading Library*. GitHub. Retrieved from https://github.com/cozmo/jsQR

10. Chart.js Documentation. (2024). *Simple yet flexible JavaScript charting library*. Retrieved from https://www.chartjs.org/docs/latest/

11. bcrypt.js. (2023). *Password Hashing for Node.js*. GitHub. Retrieved from https://github.com/dcodeIO/bcrypt.js

12. qrcode. (2023). *Node.js QR Code Generator Library*. GitHub. Retrieved from https://github.com/soldair/node-qrcode

13. MDN Web Docs. (2024). *JavaScript Fetch API – Making HTTP Requests*. Retrieved from https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API

14. HTML Living Standard. (2024). *HTML5 Specification*. Retrieved from https://html.spec.whatwg.org/

15. CSS3 W3C Specification. (2024). *Styling Web Pages – CSS Standards*. Retrieved from https://www.w3.org/Style/CSS/Overview.en.html

16. Bootstrap. (2024). *Open-Source Front-End Toolkit*. Retrieved from https://getbootstrap.com/

17. Firebase Documentation. (2024). *Realtime Database – Sync Data Across Clients in Realtime*. Retrieved from https://firebase.google.com/docs/database