Journal of Engineering Design and Analysis
Volume 8, Issue 2 - 2025, Pg. No. 1-10
Peer Reviewed Journal

Research Article

Development of an IoT-Enabled RFID Attendance System for Educational Institutions: Addressing Proxy Attendance Challenges

Spandan Parikh[1], Harikrishna Parikh[2]

[1]Student, Department of Computer Engineering, V V P Engineering College, Rajkot, Gujarat, India
[2]Student, Department of Electronics, Saurashtra University, Rajkot, Gujarat, India
DOI: https://doi.org/10.24321/2582.5607.202501

I N F O

E-mail Id:
spandyparikh@gmail.com
Orcid Id:
https://orcid.org/0009-0007-7410-5058
How to cite this article:
Parikh S, Parikh H. Development of an IoT-Enabled RFID Attendance System for Educational Institutions: Addressing Proxy Attendance Challenges . *J Engr Desg Anal* 2025*;* 8(2): 1-10.

Date of Submission: 2025-08-12
Date of Acceptance: 2025-08-28

A B S T R A C T

Managing student attendance efficiently and reliably is a long-standing challenge in educational institutions, especially in the face of widespread proxy attendance practices. This study presents the design and development of a low-cost, IoT-enabled, RFID-based attendance system aimed at eliminating proxy attendance and improving academic accountability. Utilising ESP32 microcontroller-based embedded systems interfaced with MFRC522 RFID modules to collect attendance. The same is stored in a database hosted on a web server using a MySQL-PHP backend. The system records attendance in real-time and displays it on a web-based dashboard accessible to authorised users (faculty and administrators). Although formal field testing has not been conducted, the system was successfully demonstrated in a simulated environment using mock data (student, faculty and administrator). The system shows potential for reliability, cost-effectiveness, and scalability for broader institutional adoption.

Keywords: Attendance Management, RFID Technology, Internet of Things, Educational Institutions, Proxy Detection, Academic Integrity

Introduction

Accurate student attendance tracking plays a vital role in maintaining discipline, measuring academic performance, and fulfilling institutional and regulatory requirements in educational environments. In India and across the globe, attendance is often a factor contributing to eligibility for examinations and scholarships, as well as a reflection and benchmark for assessment of student engagement and participation in the classes. Despite its importance, traditional attendance tracking methods—such as manual roll calls and physical sign-in sheets—suffer from several drawbacks. These include being time-consuming, prone to human error, susceptible to manipulation, and limited in terms of real-time accessibility.

One of the most common and concerning issues in educational institutions is the act of proxy attendance. Students frequently exploit and hack the traditional systems by having their friends mark their presence when they are physically absent. In large classrooms, it becomes practically impossible for teachers to identify each student visually during roll calls. Proxy attendance compromises academic integrity, affects student discipline, and distorts records used in critical decisions like internal assessments and scholarship disbursement. This challenge has prompted the exploration of automated solutions that can ensure authenticity and accuracy in attendance tracking.

Recent advancements in embedded systems, wireless communication, and cloud computing have paved the way for the development of smart attendance systems. Integration of Radio Frequency Identification (RFID) with the Internet of Things (IoT) devices has gained significant attraction amongst them. RFID technology offers a non-contact and rapid method of identifying individuals using unique tag identifiers. When enhanced with IoT capabilities, these systems can log, store, analyse, and present data remotely in real time, enabling smarter administration and improved data-driven decision-making.

An IoT-RFID-based attendance system leverages RFID tags embedded in student ID cards (in other words, student ID cards printed over an RFID card), RFID readers installed at entry points, and microcontroller platforms such as ESP32 with built-in Wi-Fi capabilities to transmit attendance data. This data can be sent to a server where it is recorded, analysed, and displayed through a web dashboard for authorised users such as teachers and administrators. Compared to manual attendance, this approach provides benefits like time-saving, accuracy, reduced human intervention, and most importantly, the prevention of proxy entries.

The motivation behind this project stems from the growing need for a practical, cost-effective, and secure attendance monitoring system that suits the infrastructure of both schools and colleges. Many commercially available biometric or camera-based attendance systems are expensive and complex to install and maintain. Furthermore, they raise concerns about hygiene, privacy, and network dependency. In contrast, RFID-based systems can be designed to be low-cost, scalable, user-friendly, and maintainable, making them ideal for educational environments with tight budgets and resource constraints.

This research paper presents the architecture and implementation of a prototype RFID-based attendance system enhanced with IoT features. It focuses on developing a local and networked solution using ESP32 microcontrollers, MFRC522 RFID readers, and a PHP-MySQL backend system. The solution allows for automated attendance marking, real-time data synchronisation, and report generation through an interactive dashboard. Although the system has not yet been deployed for formal live testing, it has been successfully demonstrated in a controlled, simulated environment to validate its core functionalities.

Sections in the line paper describe the system design and methodology, including the hardware configuration, software architecture, data flow, and dashboard capabilities. It then presents the expected performance outcomes based on simulated tests, followed by a discussion of the benefits and potential applications of the system. Finally, the paper concludes with future directions, including plans for real-world deployment and potential enhancements such as mobile integration and AI-based analytics.

Literature Review

Advancements in embedded systems and the rise of the Internet of Things and Radio Frequency Identification (RFID) technology have enabled the automation of attendance systems in educational institutions, which has been widely implemented in the last decades. Traditional attendance marking, including manual roll calls and paper registers is, time-consuming, error-prone, and can be manipulated. The integration of IoT with RFID is considered a viable alternative to counter these limitations. Such systems offer numerous advantages, like real-time data logging, transparency, and reduction of proxy entries, apart from saving valuable time of a teacher being consumed in manual roll call.

Radio Frequency Identification (RFID) is a wireless communication technology that uses electromagnetic fields to automatically identify and track tags attached to objects. RFID systems typically consist of tags (transponders), readers (interrogators), and backend databases. Passive RFID tags, which rely on the reader's energy, are widely used in student ID cards due to their cost effectiveness and small form factor.[1] MFRC522 is a commonly used 13.56 MHz RFID reader module, well-suited for embedded systems like the ESP32 due to its SPI and I²C interface support.[2] Compared to barcode and magnetic stripe cards, RFID offers faster read speeds, higher durability, and contactless operation). Its ability to read multiple tags simultaneously enables efficient crowd management. However, RFID systems also face challenges such as tag collisions, limited read range (5–10 cm for HF tags), and susceptibility to cloning and relay attacks.[4,5] Advances in cryptographic protocols and reader authentication mechanisms have improved the security posture of RFID systems.[6]

RFID has been widely adopted in smart campuses due to its ability to identify users uniquely and efficiently. RFID tags embedded in student ID cards enable quick scanning and automated logging. Shrivastava et al. developed an RFID-based attendance system using an Arduino ESP8266 & Adafruit.io, highlighting improved accuracy and reduced time in student tracking.[7]

The advent of IoT has multiplied the utility of RFID systems. Embedded microcontrollers such as the ESP32, equipped with Wi-Fi and Bluetooth capabilities, facilitate seamless data transfer to cloud servers and dashboards.[8,9] Koppikar et al. (2020) proposed an IoT-based RFID attendance model that sends real-time data to a web interface.[10] This enables

ISSN: 2582-5607
DOI: https://doi.org/10.24321/2582.5607.202501

administrators to monitor logs remotely and generate analytics, offering a transformative shift from isolated systems to connected infrastructures.

One of the critical challenges in RFID-based systems is proxy attendance, where students give their cards to others for unauthorised scanning. RFID systems without biometric integration are highly vulnerable to impersonation.[11] The researchers found that while RFID improves efficiency, it must be combined with secondary verification methods to ensure authenticity.[12]

Hybrid models are thus gaining attention. For example, Sanath et al. implemented a dual-layered system using RFID and facial recognition.[13] Their findings showed a significant drop in proxy cases, but at the cost of increased processing requirements and infrastructure complexity.[13] Such systems may not be feasible for budget-constrained institutions.

The literature strongly supports the integration of RFID and IoT for building scalable and efficient attendance systems. However, issues of proxy attendance and data security necessitate additional layers of verification, intelligent analytics, and robust backend infrastructures. The proposed system in this paper contributes to this domain by offering a low-cost, real-time, and modular solution that can be readily deployed in academic settings.

System Design

The system design started with the basic challenge of realising an attendance management system with a constrained budget (maximum 2000 INR) per unit capable of recording attendance with utmost accuracy and desired precision. The specifications decided for the hardware system being developed are as follows:

- Maximum number of IDs =5000
- Maximum scanning time = 100ms
- In/Out detection and notification
- Recording of time and date
- Display support for ID verification
- Network connectivity (with Wi-Fi)

The system is basically a web-based database fed with attendance input from the hardware modules. This requires software running on an embedded system as well as on a server to facilitate data transaction between hardware modules and the web server.

A Block diagram of the system architecture is shown in fig. 1.

Hardware Design

As discussed above, the system mandated the development of embedded hardware to allow the collection of attendance by reading RFID tags/cards, mark whether the student is IN/OUT and display the ID on an appropriate display for verification.

After sufficient brainstorming, we decided to develop an embedded system around ESP32. ESP32 is a microcontroller from Espressif Systems, comprising dual Tensilica Xtensa LX6 cores, each of 32-bits running at a maximum of 240 MHz (14) and boasts 520 KiB RAM and 448 KiB ROM [14]. It offers 802.11 b/g/n (Wi-Fi) and v4.2 BR/EDR and BLE and has 34 programmable GPIOs, including support for touch sensors, 12-bit SAR ADCs, DAC, Delta-sigma modulated binary output, and PWM [14]. ESP32 supports diverse communication protocols, including SPI, I^2S, UART, Ethernet MAC interface and CAN bus 2.0.[14] In short, this is a miniature system capable of executing many complex tasks at the same time, providing connectivity with various networking protocols, making it a versatile option for such system implementation.

We have selected RFID tags for authentication, for simple reason that it carries the same impression and feeling as traditional ID cards. The looking-alike format marks attendance digitally instead of manually. When a student passes through the proximity of an RFID reader, the RFID tag is read to mark who has entered or left. The details of the scanned RFID tag and Entry/Exit are updated in the database.

To facilitate human interface, or to verify the tag details, the information saved in the tag is displayed on the alphanumeric display.

The schematic of the circuit is shown in fig. 2.

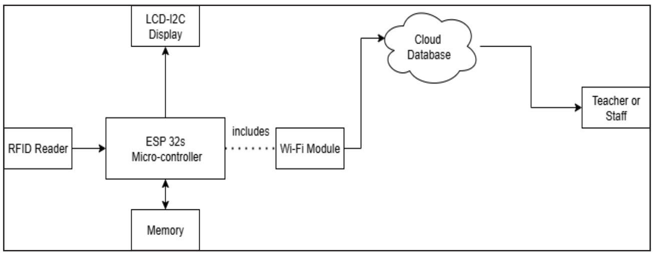

Figure 1.System Architecture

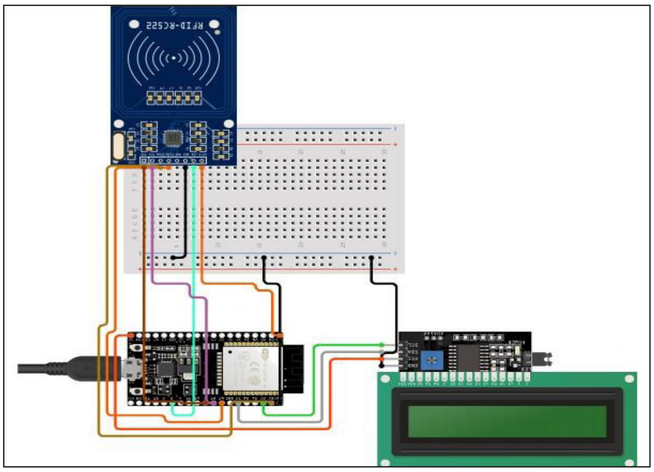

Figure 2.Schematic of Hardware Design

ISSN: 2582-5607
DOI: https://doi.org/10.24321/2582.5607.202501

Software Design

The system needs programs at two levels, one to run the embedded hardware and another to manage the web interface and database on the server. The programming of the embedded system is realised using Arduino IDE, which uses the embedded C language and ESP-IDF. The algorithm of codes running on the embedded system is as follows:

- Initialisation (with libraries WiFi, HTTP, SPI, RFID, LCD)
- Define hardware connections (RFID reader, LCD)
- Set up WiFi credentials and API endpoints
- Server, port, username, and key for Adafruit IO
- Declare global variables

Set up a Routine

- Initialise serial communication.
- LCD display.
- SPI interface and RFID reader.
- Display "Connecting..." on LCD.
- Connect to WiFi using the connectWiFi() function.
- Connect to Adafruit IO using the connectMQTT() function.
- Display system readiness on LCD (e.g., "Scan Card").
- Set lectureStartTime to the current time and mark lectureActive = true.

Main Loop (Repeat continuously)

- Detect RFID card
- Read UID (unique card ID) and convert UID to string. Debounce repeated scans within short intervals.
- Update lastScannedTag and lastScanTime.
- Display scanned UID on LCD
- Send data to local server
- Make an HTTP POST request to rfid_api.php with the scanned UID.
- Publish UID to Adafruit IO via MQTT
- If scanning is outside a specific time limit: Mark lectureActive = false.
- If the lecture is inactive and the email not sent
- Trigger email alert via HTTP POST to send_email.php. Set emailSent = true. k. Retry until successful.

The backend system of the IoT RFID-based Attendance management system is used to manage and store the daily attendance log of the institute.

The algorithm of codes running on the back-end system (web server) is as follows:

Module 1: RFID Data Handling - rfid_api.php:

Receive HTTP POST request from ESP32

Inputs: RFID, status (IN or OUT)

Sanitise and validate the received data

Ensure both RFID and status are present and valid

- Query the students table to find the student matching the RFID
- If student not found → return error response
- Check for duplicate scans within the last 5 seconds
- If a recent record with the same RFID and status exists → ignore the request
- Insert the attendance record into the attendance table
- Fields: RFID, name, department, status, timestamp
- Send a success response (e.g., JSON or plain text) · Module 2: Attendance Report Email - send_email.php:
- Connect to the database and select records (all attendance entries from today's date)
- Format retrieved records into an HTML table (name, department, status, timestamp)
- Initialise PHPMailer with SMTP config
- Set sender and recipient email addresses
- Compose and send the HTML email containing the report
- If success → show confirmation message
- If failure → display error info
- Module 3: Admin Dashboard - dashboard.php:
- Connect to database
- Handle filters (filter by date or department from dropdown or input)
- Fetch and display filtered records (data shown in an HTML table sorted by column headers)
- Display visual summary using Chart.js (bar chart or pie chart of IN vs OUT by department or day)
- Enable export options
- Export visible data as PDF (DOMPDF)
- Export as CSV via PHP script
- Module 4: Student Management via CSV - upload_csv.php
- Receive uploaded .csv file
- Parse CSV and validate each row to ensure required fields (name, RFID, department) are present
- Show preview table for admin confirmation
- Upon confirmation, insert valid student entries into the student's table
- Provide success/failure feedback
- Module 5: Database Configuration - db.php
- Connect to MySQL server using credentials (hostname, Username, Password, Database name)
- Check connection success
- If failed → exit with error message
- Else → make connection available globally

Data Flow Diagram

The Data Flow Diagram (DFD) in Figure 3 illustrates the logical flow of data in the proposed IoT RFID -based Attendance Monitoring System. The system is composed of external entities, internal processes, and data stores.

.

Entities: o Student: Represents the user carrying an RFID tag.

- **Local Server:** Receives, validates, and stores scanned data.
- **Adafruit IO Cloud:** Acts as a third-party MQTT broker for live data streaming and visualisation.

Processes

- **P1:** RFID Scanning – The ESP32 reads the RFID UID from the tag. o P2: Data Transmission – The UID is sent via HTTP POST to the local server and via MQTT to Adafruit IO.
- **P3:** Attendance Logging – The server script (rfid_api.php) stores UID in the database. o P4: Alert System – If a lecture ends and some users haven't scanned, an alert is triggered (send_email.php).

Data Stores

- **D1:** Attendance Database – Stores UID with timestamp and student data. o D2: Adafruit MQTT Feed – Stores scanned data for cloud visualisation.

Communication Protocols

The proposed prototype employs multiple standard communication protocols to ensure reliable data exchange between hardware components, the local server, and cloud-based platforms. The following subsections describe the protocols used and how different components interact within the system (table 1).

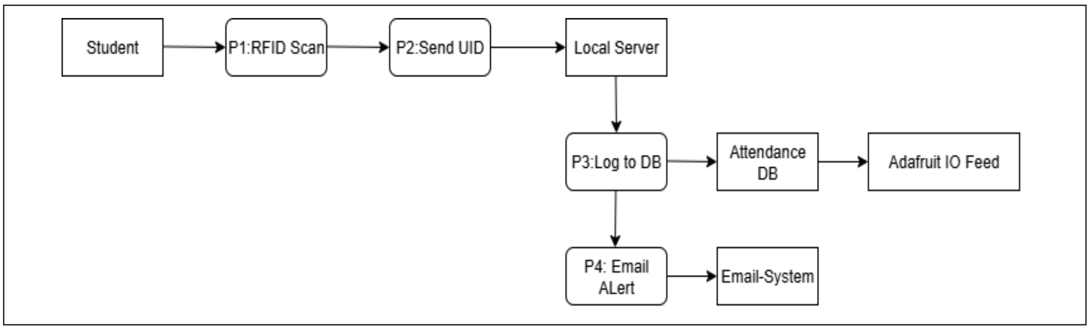

Figure 3.Data Flow Diagram

Table 1.Protocols Used

Table 1: Protocols Used	Purpose	Application in SAMS
SPI (Serial Peripheral Interface)	High-speed synchronous communication	Used for communication between ESP32 and the MFRC522 RFID reader
I^2C (Inter-Integrated Circuit)	Low-speed synchronous protocol for peripherals	Used to interface the I2C-based LCD display with the ESP32
UART (Universal Asynchronous Receiver/Transmitter)	Serial communication protocol	Used internally for serial debugging and monitoring via the Arduino Serial Monitor
HTTP (Hypertext Transfer Protocol)	Client-server communication over TCP/IP	Used to send RFID UID from the ESP32 to a local PHP-based API (rfid_api.php) and trigger email alerts (send_email.php)
MQTT (Message Queuing Telemetry Transport)	Lightweight publish-subscribe protocol over TCP/IP	Used to publish UID data to the Adafruit IO cloud dashboard in real time

ISSN: 2582-5607
DOI: https://doi.org/10.24321/2582.5607.202501

Protocols Used

Justification of Protocols o SPI is chosen for RFID due to its fast and reliable data transfer needs.

o I^2C reduces pin usage and supports multi-device communication on a shared bus. o HTTP is suitable for RESTful interaction with backend APIs in local networks. o MQTT is ideal for constrained devices and real-time messaging to the cloud. o UART is essential for on-device monitoring and debugging during development.

System Implementation

The development of the mentioned Smart Attendance Management System followed an iterative and modular approach, enabling parallel hardware-software integration and continuous improvement through testing. This section outlines the step-by-step methodology adopted for system implementation. Phases for the development of the system are elaborated as follows.

Phase 1: Requirement Analysis and Planning o Identified core requirements: contactless attendance, real-time monitoring, and cloud backup.

Selected components: ESP32 microcontroller, MFRC522 RFID reader, I^2C LCD, and local server with PHP and MySQL.

- Chose communication protocols: SPI, I2C, HTTP, and MQTT.

Phase 2: Hardware Prototyping o Interfaced MFRC522 RFID reader with ESP32 via SPI. o Connected 16x2 LCD display over I2C for status output. o Powered and tested components on a breadboard setup.

- Verified correct tag reading and LCD output independently

Phase 3: Firmware Development o Programmed ESP32 using Arduino IDE.

- Implemented functions for: o Wi-Fi connection setup o Reading RFID tags
- HTTP POST to local server (rfid_api.php) o MQTT publish to Adafruit IO o LCD display control o Added time tracking for lecture duration and email alert triggering.

Phase 4: Server-Side API Development o Created PHP script (rfid_api.php) to log data to a MySQL database. o Designed another API (send_email.php) to send automated email alerts for non-compliant attendance.

- Local Apache server (XAMPP) hosted the backend APIs and database.

Phase 5: Cloud Integration o Registered and configured an Adafruit IO account. o Created MQTT feed (rfid_data) for real-time UID publishing.

- Embedded Adafruit dashboard for visual attendance tracking.

Phase 6: Testing and Iteration o Conducted unit testing for each module (RFID, LCD, Wi-Fi, HTTP, MQTT). o Performed integration testing to ensure smooth interaction between components.

- Tested with multiple RFID tags to evaluate:
- Tag recognition speed o Data accuracy and consistency o Network resilience
- Debugged issues using UART-based serial monitoring.
- Made firmware adjustments based on scan timing, debounce interval, and cloud latency.

Database / Data Management and Database Structure

The backend system uses a MySQL relational database named ‘student db’ to manage student information and attendance records. It contains the following key tables 3-4:

Students

Attendance

Data Ingestion

When a student taps his/her RFID card on the reader, corresponding UID is transmitted to the web server via HTTP POST. If valid, the server logs the attendance with a timestamp and IN or OUT status. The system is designed to neglect multiple rapid scans using a minimum delay of 5 seconds to avoid possible duplicate entries.

Data Retrieval and Usage

The admin dashboard fetches data using PHP and MySQL queries based on Date, Department and

Attendance status (IN, OUT). Chart.js is used to visualise this data in real-time as Bar charts (daily IN/OUT trends) or Pie charts (department-wise attendance). Export options allow exporting data to CSV or PDF formats.

System Workflow

The system provides a seamless, contactless, and automated way to record attendance using RFID and IoT technologies. This section outlines the real-time workflow from a user’s perspective and describes the internal process flow. The description is followed by the work flow diagram in fig. 4

ISSN: 2582-5607
DOI: https://doi.org/10.24321/2582.5607.202501

Table 2.Database structure (student table)

Field	Type	Description
Id	INT (PK, AI)*	Unique student ID
Name	VARCHAR(100)	Full name of the student
roll_number	VARCHAR(20)	Unique institutional roll number
Department	VARCHAR(50)	Academic department (e.g., CSE)
Email	VARCHAR(100)	Email address for notifications
Rfid	VARCHAR(50)	Unique RFID card UID

Table 3.Database structure (attendance table)

Field	Type	Description
Id	INT (PK, AI)*	Unique attendance record ID
rfid	VARCHAR(50)	Foreign key linked to students. rfid
name	VARCHAR(100)	Redundant storage for quick lookup
department	VARCHAR(50)	Department for filtering
status	VARCHAR(10)	IN or OUT status
timestamp	DATETIME	Date and time of scan

*AI: Auto Increment, PK: Primary Key

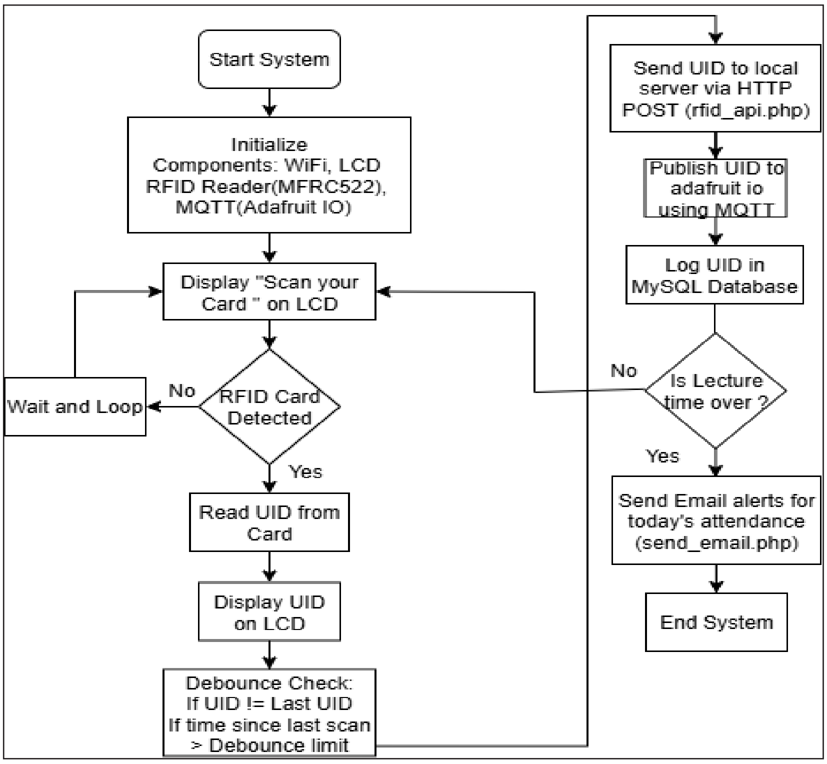

Figure 4.Work Flow Diagram

User Interaction Workflow

- **System Start-up:** This includes initialisation of Wi-Fi, LCD, RFID, and MQTT connections and concludes with displaying “Scan Your Card” on the LCD.
- **RFID Card Scanning:** When a student places an RFID card near the reader, the UID is read and displayed on the LCD.
- **Data Logging:** UID is sent to a local server (rfid_api.php) via HTTP. UID is simultaneously published to Adafruit IO using MQTT.

ISSN: 2582-5607
DOI: https://doi.org/10.24321/2582.5607.202501

- **Database Entry:** The PHP backend stores the UID with a timestamp and student info in a MySQL database.
- **Cloud Visualisation:** UID appears on the Adafruit dashboard in real time.
- **Lecture Timer:** System tracks the duration of the lecture. After the time threshold, if a UID is not recorded, send_email.php is called to notify absentees.

Results and Discussion

The implementation and testing of the proposed Attendance Monitoring System produced encouraging results in terms of accuracy, response time, and usability. This section presents the key outcomes observed during prototype testing in a real-world classroom simulation (Table 5-6).

Functional Validation

The system was evaluated based on its ability to perform core tasks reliably:

Performance Metrics

The system displayed the message on power on "Scan your Card" and upon scanning the ID, displayed the details of the ID as "Last Scanned: UID". The screenhot of the server log is shown in fig. 5. The Adafruit Dashboard: showed live entries of UIDs as they were scanned, is shown in figure 6.

Limitations Observed

- Requires stable Wi-Fi connectivity for real-time sync.
- Initial power-up time for ESP32 was ~5 seconds.
- The range of used RFID reader module is limited to ~5-6 cm, which is acceptable for controlled access.

Final Outcome

The successful implementation of the Attendance Monitoring System demonstrated the feasibility of using RFID and IoT technologies for seamless, real-time attendance tracking. The system met all functional objectives, including accurate user identification, secure data logging, and synchronised cloud integration.

Through comprehensive testing, the prototype exhibited high reliability, low latency, and user-friendly interaction. Key achievements include:

- Real-time attendance monitoring with <1 second response time per scan.
- Dual logging system: local database (MySQL) and remote cloud (Adafruit IO).
- Automated email notifications triggered after lecture timeout.
- Contactless and efficient operation, ensuring minimal classroom disruption.
- Scalable architecture for future integration with biometric or facial recognition modules. Overall, the system provides a cost-effective, scalable, and smart solution for educational institutions seeking to automate and digitise attendance management. Its modular design allows easy customisation and expansion for diverse deployment environments.

Table 5.Functional validation

Functionality	Expected Behavior	Observed Result
RFID card detection	Detect and read tag UID	Successfully detected in <1 second
UID display on LCD	Show scanned ID or error message	Displayed UID clearly on 16x2 LCD
HTTP request to local server	Send UID and receive acknowledgment	Consistent success (status 200)
Data logging in MySQL	Save UID, timestamp, and metadata	All records inserted correctly
MQTT publishing to Adafruit IO	Push UID data to cloud dashboard	Appeared in real-time (<1.5s latency)
Email notification trigger	If UID not scanned after timeout	Successfully sent through PHP script

Table 6.Performance Metrics

Metric	Result
Average UID read time	~800 ms
HTTP transmission latency	~200–300 ms (local network)
MQTT cloud update latency	~1–1.5 seconds
Attendance match accuracy	100% (in test group of 10 tags)
Email delivery delay	<5 seconds after timeout

ISSN: 2582-5607
DOI: https://doi.org/10.24321/2582.5607.202501

id	rfid	roll_number	name	department	status	timestamp
1	42152B2	23CE184	Riya Shah	CE	IN	2025-07-18 22:15:41
2	42152B2	23CE184	Riya Shah	CE	OUT	2025-07-18 22:15:47
3	E3D32BE2	23CE198	Amit Patel	CE	IN	2025-07-18 22:18:10
4	42152B2	23CE184	Riya Shah	CE	IN	2025-07-18 22:18:17
5	E3D32BE2	23CE198	Amit Patel	CE	OUT	2025-07-18 22:18:20
6	42152B2	23CE184	Riya Shah	CE	OUT	2025-07-18 22:18:25

Figure 5.Server log output

2025/07/19 08:41:30AM	e3d32be2 - IN
2025/07/19 08:41:24AM	e3d32be2 - OUT
2025/07/19 08:41:18AM	e3d32be2 - IN
2025/07/19 08:41:10AM	e3d32be2 - OUT
2025/07/19 08:41:03AM	e3d32be2 - IN
2025/07/19 08:23:04AM	e3d32be2 - OUT

Figure 6.Adafruit.io Feeds for RFID with UID and Status

Conclusion

In this study, the Attendance Monitoring System was designed and implemented to address the inefficiencies and manual overhead associated with traditional attendance methods in academic environments. Leveraging RFID technology, IoT-based communication protocols, and real-time cloud synchronisation, the proposed system successfully automates the process of student identification and attendance tracking and management with minimal human intervention.

The system operates effectively by integrating multiple technologies such as RFID (for tag detection), HTTP and MQTT (for data transmission), and a local server with PHP and MySQL (for backend storage and processing). The inclusion of Adafruit IO enables real-time visualisation and monitoring of attendance data, while the automated email alert system provides timely notifications to enforce compliance.

Comprehensive testing of the prototype confirmed that the system performs with high accuracy and low latency, meeting the demands of a real-world classroom scenario. Additionally, the modular design of the system ensures that it can be scaled or upgraded with future enhancements such as biometric validation, facial recognition, or integration with institutional ERP systems.

In conclusion, the prototype demonstrates a robust, efficient, and scalable solution for smart attendance management. It contributes toward the growing need for digitisation in educational administration and serves as a stepping stone for more intelligent campus automation systems in the future.

References

1. Finkenzeller , Klaus. RFID handbook: fundamentals and applications in contactless smart cards, radio frequency identification and near-field communication: John wiley & sons; 2020.
2. Z, Y W, H Z. Design and Implementation of RFID Reader System Based on MFRC522. In Proceedings of the IEEE International Conference on Information and Automation.; 2017.
3. Feb WR.2, 5(1):25-33.. An introduction to RFID technology. IEEE pervasive computing. 2006 February; 5(1): 25-33.
4. Juels A. RFID security and privacy: a research survey. IEEE Journal on Selected Areas in Communications. 2006 February; 24(2): 381-394.

ISSN: 2582-5607
DOI: https://doi.org/10.24321/2582.5607.202501

5. Karygiannis T, Eydt B, Barber G, Bunn L, Phillips T. Guidelines for securing radio frequency identification (RFID) systems.. 2007 Apr 6;80:1-54. Special Publication. ; 2007.
6. G A, MA B, S K, C L, B M. A framework for analysing RFID distance bounding protocols. Journal of Computer Security. 2011; 19(2): 289-317.
7. Shrivastava A, Prasad SS, Yeruva A, Mani P, Nagpal P, Chaturvedi A. IoT Based RFID Attendance Monitoring System of Students using Arduino ESP8266 & Adafruit.io on Defined Area. Cybernetics and Systems. 2023; 56(1): 21–32.
8. Hercog D, Lerher T, Truntič M, Težak O. Design and Implementation of ESP32-Based IoT Devices. Sensors. 2023; 23(15): 6739.
9. Oner VO. Developing IoT Projects with ESP32: Packt Publishing; 2021.
10. Koppikar U, Hiremath S, Shiralkar A, Rajoor A, Baligar V. IoT based Smart Attendance Monitoring System using RFID. In 2019 1st International Conference on Advances in Information Technology (ICAIT); 2019; Chikmagalur. p. 193-197.
11. Xiao Q, Gibbons T, Lebrun H. RFID Technology, Security Vulnerabilities, and Countermeasures. In Supply Chain, The Way to Flat Organisation. Vienna, Austria; 2008. p. 357-382.
12. Huang Y, Fu B, Peng N, Ba Y, Liu X, Zhang S. RFID Authentication System Based on User Biometric Information. Applied Sciences. 2022; 12(24): 12865.
13. Sanath K, Meenakshi K, Rajan M, Balamurugan V, Harikumar M. RFID and Face Recognition based Smart Attendance System. In 2021 5th International Conference on Computing Methodologies and Communication (ICCMC); 2021; ERODE, INDIA. p. 492-499.
14. Espressif Systems. ESP32 Series Datasheet. [Online].; 2025 [cited 2025 July 27. Available from: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf.

ISSN: 2582-5607
DOI: https://doi.org/10.24321/2582.5607.202501