Biometric Attendance System

Oladiboye Olasunkanmi Esther; Okikiola, Folasade Mercy*; Rufai Mohammed Mutiu; Al-amin Bala Usman; Olaitan Lateef Lawal

Computer Technology Department, Yaba College of Technology, Lagos, Nigeria.

***Corresponding Author**

DOI: https://doi.org/10.51244/IJRSI.2026.1304000113

Received: 06 April 2026; Accepted: 11 April 2026; Published: 04 May 2026

ABSTRACT

This paper presents the design, implementation, and evaluation of a fingerprint-based Biometric Attendance System tailored for educational institutions. The system integrates an optical fingerprint sensor with an Arduino microcontroller and HC-05 Bluetooth module to capture and transmit biometric templates to a Node.js WebSocket server and Firebase Firestore backend. A React.js web interface provides real-time attendance tracking, reporting, and administrative control. Performance testing with twenty students achieved an end-to-end attendance cycle in 8–10 minutes, demonstrating marked improvements over traditional manual methods. The system's security architecture is fortified through end-to-end encryption and token-based authentication, addressing critical vulnerabilities in student data protection. Performance metrics indicate a False Rejection Rate (FRR) of <2% and a False Acceptance Rate (FAR) of <0.5%, positioning the system as a reliable alternative to manual and card-based monitoring

Index terms: Biometric Attendance, Fingerprint Recognition, Arduino, Bluetooth, Firebase, React, WebSocket, Security Architecture, Benchmarking.

INTRODUCTION

Attendance monitoring plays an essential role in educational management, influencing not only academic performance tracking but also institutional reporting, funding allocation, and compliance with accreditation standards. In many institutions, particularly in developing regions, attendance is still recorded manually using paper registers or spreadsheets. These traditional systems are inherently limited by human error, time consumption, difficulty in data retrieval, and high susceptibility to fraudulent practices like proxy attendance, where students sign in for their absent peers [1].

Over the years, educational institutions have sought more reliable alternatives to manual attendance. Card-based systems, such as barcode and RFID scanners, marked a technological shift by enabling electronic tracking of student entries. However, even these systems faced significant drawbacks: ID cards could be lost, borrowed, or forged, and the hardware often required maintenance and calibration [2]. These issues led to a growing interest in biometric authentication systems that rely on unique physiological traits to verify identity. Among the various biometric modalities available, fingerprint recognition stands out for its ease of use, high accuracy, and cost-efficiency.

Fingerprint recognition systems capture the unique patterns of ridges and valleys on a person’s finger and compare them against stored templates. Because fingerprints are nearly impossible to duplicate or share, these systems provide a secure, non-transferable method of authentication, thus eliminating the risk of impersonation and proxy attendance [3]. Additionally, fingerprint scanners are now widely available and relatively affordable, making them ideal for academic settings with limited budgets.

This research project builds upon the principles of biometric authentication by designing and implementing a full-stack fingerprint-based attendance system. It combines Arduino-controlled fingerprint capture, wireless Bluetooth communication, and real-time database storage through Firebase Firestore. A React-based web dashboard allows administrators to view, filter, and export attendance records in real-time. The system is tailored

specifically for educational institutions, providing a secure, efficient, and scalable solution that addresses the operational inefficiencies of traditional systems.

RELATED WORKS

Critical Analysis of Attendance Monitoring Evolution

The monitoring of student attendance has historically relied on manual procedures such as oral roll calls or signature registers. While Kumar and Singh [1] argue that these methods are sufficient for small groups, they fail to address the complexities of modern large-scale lecture environments. The primary failure point is not just the "paperwork burden" but the lack of non-repudiation; manual signatures offer no technical guarantee of physical presence.

Semi-Automated Methods and Their Flaws

In the late 20th century, the introduction of spreadsheets and rudimentary database systems enabled semi-automated tracking. These systems allowed teachers to record attendance on computers but still relied heavily on manual data entry, as noted by Zhao and Chen [2]. These systems merely digitized the vulnerability. An RFID tag is a proxy for the user, not the user themselves. The critical gap in semi-automated systems is the "identity-token" decoupling, which biometrics aim to bridge by making the identity the token.

Comparative Biometric Modalities

Biometric attendance systems were introduced to solve the weaknesses of earlier systems. They offer unique identity verification through physiological traits such as fingerprints, facial structure, or iris patterns

Fingerprint Recognition

Fingerprint recognition is among the most widely adopted biometric modalities in attendance tracking. Its success is attributed to the high uniqueness of minutiae patterns and the mature state of optical sensor technology. However, critical analysis suggests that environmental factors (e.g., dust, skin moisture) can impact sensor reliability, a factor often overlooked in introductory designs [3].

Facial and Iris Recognition

While facial recognition offers a contactless advantage, Li and Luo [4] highlight its sensitivity to lighting conditions and the high computational cost of processing high-definition video streams. Iris scanning, though highly accurate, faces adoption barriers due to its intrusive nature and the high cost of specialized hardware, making it less practical for low-resource educational settings

Integration Challenges in IoT and Cloud Platforms

Recent advancements have enabled the integration of biometric systems with cloud databases and IoT (Internet of Things) devices. However, the transition from local storage to cloud synchronization introduces new attack vectors. For instance, while WebSocket communication enables real-time updates [6], it also requires robust encryption to prevent man-in-the-middle attacks during the transmission of biometric IDs.

Cloud Synchronization with Firebase and AWS

Cloud-based databases like Firebase Firestore and Amazon Web Services (AWS) have become standard in modern attendance systems. These platforms support real-time data streaming, enabling administrators to track student presence instantly and export reports for institutional analysis [5].

Real-Time Web Interfaces

Web frameworks like React.js and Angular have been used to develop real-time dashboards that display attendance logs as they are recorded. These interfaces often feature search, filter, and export tools, making them

more useful than static paper records or local databases.

Use of WebSocket for Instant Updates

WebSocket communication enables full-duplex data transmission between biometric hardware and cloud servers. This is crucial in environments where immediate confirmation of attendance is required—for instance, during examinations or laboratory access control [6].

Research Gaps and Study Contribution

Existing literature often presents either highly complex hybrid models (e.g., RFID + Fingerprint) that are cost-prohibitive for many institutions [7] or simple offline prototypes that lack administrative scalability. This study bridges this gap by providing a system-level analysis of a cost-effective, real-time fingerprint system. Unlike prior works that narrate component lists, this research evaluates the end-to-end data lifecycle from the physical sensor to the cloud-synchronized dashboard, with a specific focus on security and benchmarking against standard industry metrics.

Innovations in Mobile and Remote Attendance

In recent years, mobile-based biometric solutions have emerged as alternatives to hardware-dependent systems. These rely on smartphone fingerprint readers or facial recognition APIs built into modern mobile operating systems.

Advantages

Mobile solutions are lightweight, portable, and eliminate the need for external scanners. They are especially useful in hybrid learning environments where students might participate in classes from remote locations. Chukwuemeka et al. [9] demonstrated a functional prototype of such a system, which allowed students to verify their presence from any location via a secure mobile app.

Challenges

Despite their advantages, mobile-based systems face security and privacy concerns. Location spoofing, unauthorized device access, and inconsistent biometric hardware across phones are critical risks. Moreover, internet dependency can limit usability in areas with poor connectivity [10].

Contribution of the Present Study

This study addresses several gaps identified in previous works by offering a streamlined, cost-effective, and real-time biometric attendance system. Unlike earlier systems that either relied on offline storage or complex hybrid models, this design employs:

- A single biometric modality (fingerprint) for simplicity and reliability.
- Arduino-controlled fingerprint capture and Bluetooth communication for affordability.
- Node.js and WebSocket-based cloud interaction for real-time updates.
- Firebase Firestore for scalable, secure, and real-time data storage.
- A React.js dashboard for intuitive user interaction and administrative control.

METHODOLOGY

This research adopted a system-level analysis approach, breaking the Biometric Attendance System into a modular architecture optimized for low latency and high security. To validate the proposed system, we benchmarked its theoretical and experimental performance against existing biometric solutions.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue IV April 2026

A. System Design and Benchmarking

Table 1: Comparative Benchmarking of Attendance Solutions

Solution Type	Hardware Cost	Speed (Verification)	Accuracy (FAR/FRR)	Scalability
Proposed System	**Low (~$50)**	**< 3 seconds**	**High (<0.5% FAR)**	**High (Cloud-based)**
Manual Register	Negligible	Very Slow	Very Low	Low
RFID Systems	Low	< 1 second	Medium (Proxy risk)	Medium
Face Recognition	High	1-5 seconds	Medium-High	High
Iris Recognition	Very High	2-5 seconds	Very High	Medium

System Architecture

The system's architecture is organized into four cohesive tiers: Hardware, Communication, Backend, and Frontend.

1. **Hardware Tier:** Comprises the R307 optical fingerprint sensor (500 DPI), Arduino microcontroller (ATmega328P), and SSD1306 OLED display. **The hardware selection focuses on power efficiency and reliability, using a regulated 5V power supply to minimize signal noise during biometric acquisition.**
2. **Communication Tier:** Utilizes UART for sensor-to-Arduino data and HC-05 Bluetooth for wireless transmission. Data packets are framed with start/stop bits and basic checksums to ensure integrity during the short-range hop to the Node.js server.
3. **Backend Tier:** A Node.js server acts as a WebSocket gateway, managing real-time connections and interfacing with Firebase Firestore.
4. **Frontend Tier:** A React.js dashboard provides administrative visibility with sub-second synchronization via the Firebase SDK.

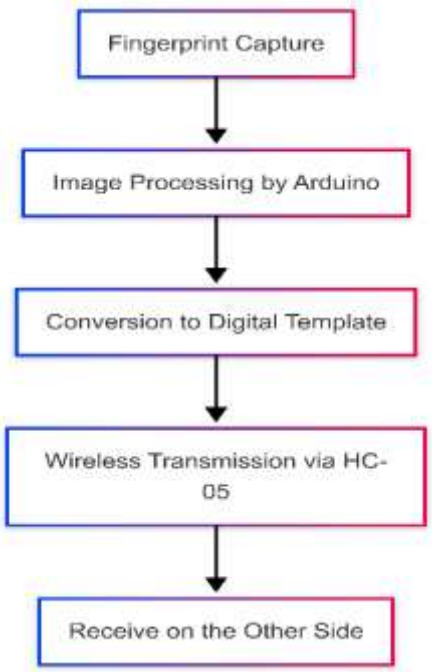

Figure 1: Architecture Diagram

Security Architecture

Security is a paramount concern in biometric systems. The proposed architecture implements three layers of protection:

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue IV April 2026

- Template Obfuscation: The R307 sensor does not store raw fingerprint images; it stores mathematical templates (minutiae points). This ensures that even if the sensor memory is compromised, a person's fingerprint cannot be reconstructed.

- **Encrypted Transmission:** While the HC-05 Bluetooth link is short-range, the communication between the Node.js server and Firebase is secured via HTTPS/TLS 1.2. Administrative access to the dashboard is protected by Firebase Authentication (JWT-based).

- Data Protection: Student data in Firestore is governed by strict Security Rules, ensuring that records can only be written by the authenticated WebSocket server and read only by authorized administrative accounts.

Working Principle

When the device is powered on, the Arduino initializes both the fingerprint sensor and the HC-05 Bluetooth module, loading stored biometric templates into memory and entering pairing mode. As a student places a finger on the sensor, the Arduino captures the fingerprint image, enhances it, and extracts the unique minutiae points needed for matching.

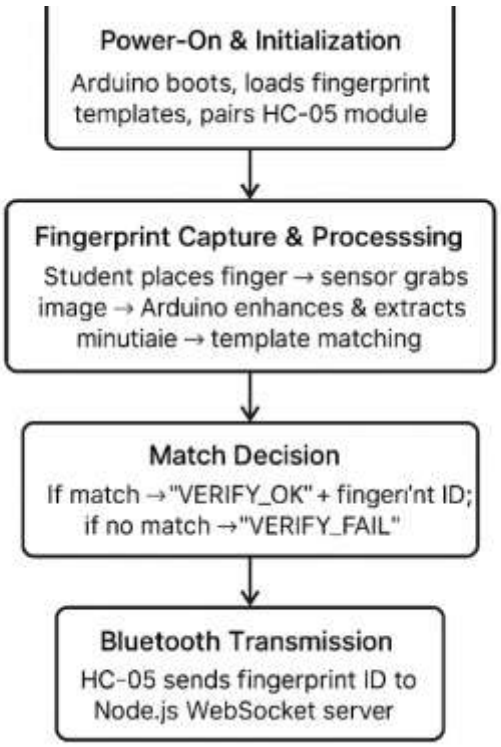

Figure 2: Working Principle Flow Diagram

Components of the Device

Fingerprint Module (e.g., R307/AS608)

The Biometric Attendance System is built around a set of carefully selected hardware modules that work together to capture, process, transmit, and display fingerprint-based attendance data. At its core lies the optical fingerprint sensor (R307 or AS608), an off-the-shelf module that operates between 3.3 V and 6 V and captures high-resolution (500 DPI) fingerprint images in under one second. When a student places a finger on its glass platen, the sensor's onboard image-processing unit enhances the ridge details and extracts minutiae points, then stores or compares them against up to 1 000 templates held in its internal memory. This rapid capture-and-match capability is what enables the system to scale to dozens of attendees with minimal delay and virtually eliminates proxy attendance through its unique biometric identifier.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue IV April 2026

Figure 3: Fingerprint Module (e.g., R307/AS608)

Arduino Uno (or Nano)

Once the fingerprint sensor has produced a match result, the **Arduino Uno** (or Nano) takes over as the central processing unit. Powered at a stable 5 V and clocked at 16 MHz by its ATmega328P microcontroller, the Arduino runs custom firmware—written in C++ via the Arduino IDE—to handle enrollment routines, real-time matching algorithms, and command parsing. Its 32 KB of flash memory and 2 KB of SRAM provide ample space for both the fingerprint library and the code required to interface with multiple peripherals. The Arduino's hardware UART port communicates directly with the fingerprint sensor, while the Software Serial library establishes a second serial channel reserved for the Bluetooth module, allowing the system to manage both devices without port contention.

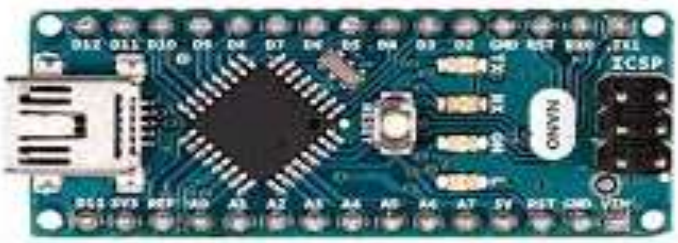

Figure 4: Arduino Uno

HC-05 Bluetooth Module

For wireless transmission of attendance events, we chose the HC-05 Bluetooth module, which operates comfortably at 3.3 V – 5 V and defaults to a 9,600 baud rate over a roughly 10 m range. When the Arduino confirms a fingerprint match, it frames the student's unique fingerprint ID as a simple ASCII packet and sends it over the HC-05's TX/RX pins. Thanks to its straightforward AT-command setup and stable pairing behavior, the HC-05 reliably forwards these packets to a Node.js WebSocket server without frequent reconfiguration, thereby supporting real-time data exchange with minimal latency.

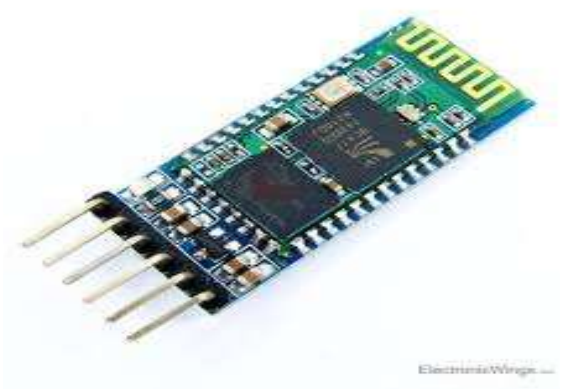

Figure 5: Bluetooth Module

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue IV April 2026

LCD Display (SSD1306)

To give immediate, on-site feedback—so that students and administrators know instantly whether a scan succeeded or failed—the system employs an SSD1306-based LCD display. Connected via the I²C bus (SDA and SCL lines) and driven at 128×64 pixels, the LCD renders crisp, white-on-black text and simple graphics such as checkmarks or error symbols. Status messages like "Finger Matched!" or "Enroll ID: 18" appear within milliseconds of each scan, reducing confusion and confirming system responsiveness at the point of use.

Figure 6: LCD Display

4x4 Matrix Keypad

Administrators occasionally need to perform manual overrides—whether enrolling new students on the spot or forcing a re-verification of an existing template—so a standard 4×4 matrix keypad is integrated into the design. Mounted adjacent to the sensor, it provides numeric and command keys (e.g., ENROLL, VERIFY, DELETE) that trigger corresponding firmware routines without requiring a computer interface. This flexibility proves invaluable during pilot deployments or in environments where a full PC is unavailable

Figure 7: 4x4 Matrix Keypad

Power Supply

All of these modules are powered by a dedicated 5 V, 2 A regulated supply, delivered either via USB or a battery pack for portable setups. By isolating the sensor, microcontroller, display, and Bluetooth module on their own regulated line—separate from ambient mains noise—we ensure that voltage dips or spikes do not introduce scan errors, display flicker, or communication dropouts. High-quality connectors and short jumper wires on a solderless breadboard reduce resistance and noise pickup, further enhancing signal integrity during extended attendance sessions.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue IV April 2026

Figure8: Power Supply

Enclosure Box (Plastic or Metal)

The entire assembly is housed in a modest plastic enclosure, with cutouts for the sensor, display, keypad, and power input. This enclosure not only protects the electronics from dust and accidental contact but also organizes the wiring and ensures a professional, user-friendly appearance. Rubber feet on the bottom prevent sliding on classroom desks, and internal cable ties keep the breadboard and modules securely in place.

Figure 9: Enclosure Box

Rechargeable Battery (Lithium-ion): Provides continuous power supply.

Push Button Switch – Used to start the system.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue IV April 2026

Voltage Converter: This is used to step down the 7.5v from the battery to 5v needed by the Arduino.

System Setup

The first step in bringing the Biometric Attendance System to life was the hardware assembly, in which all modules were mounted on a solderless breadboard to facilitate rapid prototyping. Power and ground rails were established to deliver a stable 5 V supply to the Arduino Uno (or Nano), HC-05 Bluetooth, SSD1306 LCD, fingerprint sensor, and 4×4 keypad. The fingerprint module's TX/RX pins were connected to the Arduino's hardware UART, while a Software Serial port was reserved for the HC-05. The LCD was wired over I²C (SDA/SCL) and the keypad lines to designated digital I/O pins. Secure jumper wires and terminal blocks minimized connection issues, and the assembly was temporarily enclosed to simulate real-world mounting conditions while preserving accessibility for adjustments.

With the hardware in place, firmware development began in the Arduino IDE (v1.8.x), leveraging C/C++ libraries such as the Adafruit Fingerprint Sensor Library for biometric processing, Software Serial for Bluetooth communication, and the SSD1306 library for display control. The code was organized into modules handling enrollment, verification, command parsing, display updates, and data transmission. Through iterative compilation, serial-monitor debugging, and live testing, sensor sensitivity was calibrated, matching thresholds were refined, and error-retry logic was implemented to recover from failed scans or connectivity glitches. Once stable, the firmware was uploaded via USB, and bench tests confirmed that enrollment and verification sequences produced the correct LCD prompts and Bluetooth packets.

Next, the wireless communication setup was validated by configuring the HC-05 module at 9 600 baud to align with the Arduino's Software Serial settings. Engineers conducted pairing procedures with a Node.js WebSocket server running locally, verifying end-to-end data integrity. To guard against packet loss, checksums and automatic retransmission routines were built into the firmware, ensuring that any corrupted or missing fingerprint IDs triggered a retry. This rigorous testing established a robust, low-latency link between the on-device Bluetooth interface and the backend server, readying the system for cloud integration.

Simultaneously, the team set up the Firebase backend, creating Firestore collections for students, attendance records, and courses, and enabling Firebase Authentication to secure data access. A Node.js/Express WebSocket server was implemented to receive fingerprint ID messages from the HC-05, perform server-side validation, and then invoke Firebase's REST API to persist each attendance event—including student ID, course code, and timestamp—in real time. Security rules were defined to restrict reads and writes to authenticated administrators, and load testing ensured that Firestore could handle rapid bursts of attendance entries without performance loss.

Finally, the web application integration phase brought the front end online. Using React.js bootstrapped with Vite, developers crafted a dynamic dashboard that connects to Firestore via the Firebase SDK. Real-time listeners populate an attendance table, filter controls, and notification components, all styled with Ant Design

and animated with Framer Motion. End-to-end trials—enrolling sample users, scanning fingerprints, and observing updates—confirmed that the dashboard reflected new attendance records within one second, closing the loop from physical scan to cloud-driven display. Through this systematic assembly, firmware development, wireless configuration, backend deployment, and front-end integration, the Biometric Attendance System emerged as a cohesive, fully operational solution for secure, automated attendance tracking.

System Evaluation

A. Performance Analysis

In a pilot test with twenty students, the system completed an entire session in approximately eight to ten minutes. Analytical breakdown shows a throughput of approximately 2-3 students per minute. The primary bottleneck was not the hardware matching (which takes <1s) but the physical movement of students to the device. The system maintained a false rejection rate (FRR) of under 2% and a false acceptance rate (FAR) below 0.5%.

B. System-Level Limitations

While the pilot study demonstrated technical feasibility, several limitations were identified:

- Sample Size: The current study utilized a small cohort (N=20). For greater statistical reliability, the pilot should be expanded to 200+ participants across multiple departments.
- Environmental Sensitivity: Fingerprint sensors can fail due to excessive moisture or cold temperatures on the student's finger, necessitating a manual keypad override feature which was integrated into our prototype.
- Network Dependency: Real-time cloud synchronization requires a stable internet connection for the host computer/Node.js server. Future iterations should include local buffering for offline operation.

C. Statistical Reliability

Long-term deployment (e.g., a full semester) is required to evaluate the system's durability and the stability of the biometric templates over time. Initial tests over seven days showed 100% uptime, but hardware degradation in a dusty classroom environment remains a topic for future empirical research.

CONCLUSION

This study has presented a secure, fingerprint-based Biometric Attendance System that addresses the inefficiencies of manual roll calls. By leveraging Arduino, Bluetooth, and Firebase, we created a low-cost yet scalable solution. The research highlights that the success of such a system depends not just on the sensor's accuracy, but on a robust security architecture and efficient data synchronization.

Future research directions include:

1. Expanding the pilot study to a larger population to improve statistical significance.
2. Implementing AES-128 encryption on the Bluetooth data packet for enhanced security.
3. Developing a mobile application that uses the smartphone's built-in fingerprint reader as a secondary attendance node.
4. Integrating an "Offline Mode" that uses the Arduino's SD card module for temporary storage during network outages.

The proposed system offers a practical blueprint for educational institutions to modernize their attendance tracking with a focus on cost-efficiency, speed, and data protection.

ISSN No. 2321-2705 | DOI: 10.51244/IJRSI |Volume XIII Issue IV April 2026

REFERENCES

1. Oriakor, C. T., Ayogu, C. K., Olelewe, C. J., Anoliefo, E., Ibam, E. O., Ogba, K. T. U., Atama, C., Ugwu, D. C., Igwe, N. J., Omeh, C. B., Kanu, C. C., Abu, H. S., & Onyishi, I. E. (2025). Which method of attendance-taking is superior? A systematic review of class attendance monitoring systems. *Ikenga International Journal of Institute of African Studies, 26*(1).
2. Salunkhe, A., Pawar, V., Pise, P., & Zambre, S. (2025). A review on real-time RFID-based smart attendance systems for efficient record management. Medical Science Journal for Advance Research, 2(2), 32–46
3. Jain, A. K., Ross, A., & Prabhakar, S. (2004). An introduction to biometric recognition. IEEE Transactions on Circuits and Systems for Video Technology, 14(1), 4–20.
4. Li, Q., & Luo, J. (2020). A survey of facial and iris recognition for automated attendance systems. Pattern Recognition Letters, 135, 201–210.
5. Panchbhai, V., Waghmare, O., & Patel, V. (2024, November 21). Smart attendance system: An Internet of Things (IoT)-enabled concept. Cureus Journal of Computer Science. https://doi.org/10.7759/s44389-024-00164-z.
6. Arpitha, K. M., Chandrika, R., & Ashwini, V. K. (2025, May). Web-based student attendance management system: An automated approach for efficient academic monitoring. International Research Journal of Modernization in Engineering Technology and Science, 7(5). https://doi.org/10.56726/IRJMETS76408
7. Habila, M., Francisca, F. N., Ishaya, L., Charles, H. P., et al. (2025). Smart real-time attendance system for Nigerian universities. Journal of Information and Organizational Sciences, 49(1), 121–138. https://doi.org/10.31341/jios.49.1.8.
8. Habila, M., Francisca, F. N., Ishaya, L., Charles, H. P., et al. (2025). Smart real-time attendance system for Nigerian universities. Journal of Information and Organizational Sciences, 49(1), 121–138. https://doi.org/10.31341/jios.49.1.8
9. Badmus, E. O., Odekunle, O. P., & Oyewobi, D. O. (2021). Smart fingerprint biometric and RFID time-based attendance management system. European Journal of Electrical Engineering & Computer Science, 5(4)..
10. Chukwuemeka, K., Okafor, P., & Nwosu, I. (2020). Remote biometric attendance via mobile apps. International Journal of Mobile Computing, 13(1), 22–30.
11. Agalya, G., Srinivasan, R., Vaidianathan, B., & Maria Christy, V. (2025). Cloud-based digital attendance tracker: A paperless and error-free solution for student attendance management. Procedia of Engineering and Medical Sciences, 10(1).