Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 11, CONFLUENCE - 2026

An IoT-Enabled Fingerprint Attendance System with Real-Time Security Monitoring using ESP32-CAM

Sanjana U
Student, Dept. of MCA
Community Institute of Management Studies
Dr. Manmohan Singh Bengaluru City University
Bengaluru, India

Swati Zavar
Assistant Professor, Dept. of MCA
Community Institute of Management Studies
Dr. Manmohan Singh Bengaluru City University
Bengaluru, India

Dr. Manjula P. S.
Professor and Head, Dept. of MCA
Community Institute of Management Studies
Dr. Manmohan Singh Bengaluru City University
Bengaluru, India

***Abstract*—Manual registers and conventional identification methods are still common in attendance tracking, and they invite inaccurate records, proxy attendance and weak control over who gains entry. This paper describes an IoT-based biometric system for attendance and security monitoring that pairs fingerprint authentication with an ESP32 microcontroller to automate attendance for staff. Registered users are verified by fingerprint, and the system stores each login and logout together with its date and time. For added security, an ESP32-CAM takes a photograph each time an unauthorized attempt is detected. Users get instant feedback from an LCD, LEDs and a buzzer, and the administrator is kept informed of attendance and security events through Telegram messages. Records are saved in an Excel-compatible format so the administrator can open and manage them easily. Combining biometric verification, wireless connectivity, image capture, live notifications and automatic record keeping, the proposed design offers a practical way to cut manual work while making attendance monitoring more secure and dependable.**

Keywords*—Internet of Things (IoT), Biometric Authentication, Fingerprint Recognition, Attendance Management, ESP32, ESP32-CAM, Security Monitoring, Telegram Notification, Automated Attendance.

I. INTRODUCTION

Keeping attendance records both accurate and secure is a routine yet important administrative duty for any organization. Paper registers and card-swipe systems are vulnerable to human mistakes, proxy attendance ("buddy punching"), tampering with data and slow record handling. As workplaces move toward smart, connected technology, demand is rising for attendance solutions that are precise, automated, secure and able to monitor events in real time [1], [4].

Biometric authentication helps overcome many of these shortcomings, since biometric traits belong to one person and are hard to lend or copy. Of the available modalities, fingerprint recognition is among the most widely adopted because it is accurate, inexpensive, simple to integrate and quick to respond [2], [3]. When a fingerprint sensor is paired with a Wi-Fi-capable microcontroller such as the ESP32 [6], it can serve as the heart of an affordable Internet of Things (IoT) attendance terminal.

This paper introduces *NexAttend*, an IoT-based fingerprint attendance and security system built at the Department of MCA, Community Institute of Management Studies. Its main contributions are:

- a low-cost ESP32-based terminal that verifies faculty and students by fingerprint and records their login and logout date and time automatically;
- an intruder-response feature in which an unrecognized fingerprint prompts an ESP32-CAM to take a photograph of the person and send it to the administrator via a Telegram bot [5];
- local feedback through several channels (16×2 LCD, green and red LEDs, and a buzzer), along with live Telegram notifications and automatic logging to a spreadsheet that only the administrator can open; and
- an analysis of the attendance log generated while the prototype was in live use (35 records, 12 distinct days, six enrolled users).

The rest of the paper is structured as follows. Section II surveys related work, Section III presents the architecture, Section IV explains the implementation, Section V reports the results, Section VI covers advantages and applications, and Section VII closes with conclusions and future work.

II. RELATED WORK

A. Biometric Attendance Systems

Early attendance management depended on handwritten registers and ID cards, which were error-prone and simple to misuse. Fingerprint-based systems emerged to address this by linking every record to a unique identifier that is difficult to forge, which curbs fraudulent attendance and raises accuracy [2], [3]. Fingerprint matching generally extracts minutiae (ridge endings and bifurcations) and compares them against enrolled templates using a similarity threshold [2].

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 11, CONFLUENCE - 2026

B. Microcontroller-Based and IoT Attendance

As embedded platforms such as Arduino and the ESP family became widespread, biometric sensors were connected to microcontrollers so that login and logout times could be captured without human involvement, easing administrative workload [8]. IoT connectivity then allowed these terminals to sync with cloud services and remote servers, giving administrators remote access to records and centralized data management [1], [4].

C. Notification and Surveillance

Messaging channels such as Telegram, e-mail and mobile alerts have been added to attendance terminals so that administrators learn about attendance and access events immediately [5]. Cameras have also been paired with biometric checks to capture images of unknown people attempting entry, which strengthens access control. Cloud-hosted spreadsheets and databases are now a common choice for automatic record storage and report generation, with access limited to authorized administrators.

D. Positioning of This Work

The shift from manual procedures to intelligent, IoT-enabled biometric systems brings fingerprint authentication, wireless communication, cloud storage and surveillance together on a single platform. NexAttend continues in this direction using one ESP32 and ESP32-CAM pair; it logs attendance for both faculty and students and sends an automatic intruder-photo alert, all with inexpensive off-the-shelf components.

III. SYSTEM ARCHITECTURE

A. Overview

Fig. 1 presents the block diagram. The ESP32 acts as the central controller: it receives authentication results from the fingerprint sensor over UART, drives the LCD via I²C, controls the LEDs and buzzer through GPIO, and uses its integrated Wi-Fi to connect to the Telegram Bot API, the spreadsheet web service and the ESP32-CAM. The ESP32-CAM remains idle until an unauthorized attempt triggers it.

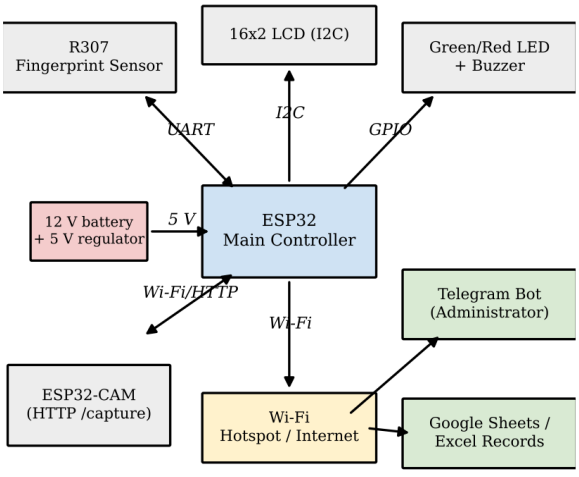


Fig. 1. Block diagram of the NexAttend system.

B. Hardware Components

The hardware is listed in Table I. The R307 optical module carries out image capture, feature extraction and template matching on its own and reports back only the matched template ID or a no-match result, so no additional signal-conditioning circuitry is required. The ESP32 combines a dual-core 32-bit processor with IEEE 802.11 b/g/n Wi-Fi and Bluetooth [6], while the ESP32-CAM packs a camera, Wi-Fi and a TF-card slot onto a small board.

TABLE I. HARDWARE COMPONENTS AND INTERFACES

Component	Role	Interface to ESP32
ESP32 board	Central controller	—
R307 fingerprint sensor	Enroll, match templates	UART2, 57600 baud
ESP32-CAM	Intruder image capture	Wi-Fi, HTTP /capture
16×2 LCD (addr. 0x27)	User/status display	I²C (SDA, SCL)
Green LED, red LED	Success / alert indicator	GPIO 19, GPIO 2
Buzzer	Audible feedback	GPIO 18
12 V battery, 5 V regulator	Power supply	VIN / 5 V rail

C. Power Supply

A 12 V battery, connected through a switch, powers the prototype. A DC–DC buck converter or a 7805 regulator reduces this to 5 V, which supplies the ESP32 VIN pin, the fingerprint sensor, the ESP32-CAM, the LCD and the buzzer. The ESP32's on-board regulator then produces 3.3 V for the processor and GPIO. All modules are tied to a common ground, and the LEDs are driven from GPIO pins via current-limiting resistors.

D. Software Platform

The firmware is developed in Embedded C/C++ with the Arduino IDE and uses the Adafruit Fingerprint, LiquidCrystal I²C, WiFi, HTTPClient, WiFiClientSecure, UniversalTelegramBot and Preferences libraries. Timestamps come from an NTP server using a UTC+05:30 offset. Notifications are delivered through Telegram, and attendance rows are written to a spreadsheet that is kept and exported in Excel format.

IV. IMPLEMENTATION

A. Enrollment and Identity Mapping

Fingerprints are enrolled in the sensor's flash memory, with each template given a numeric ID. In the prototype, IDs 1–4 correspond to four faculty members (a Professor and Head, a Professor and two Assistant Professors) and IDs 5–6 to two MCA students. The firmware stores each ID's name, designation and department or course. Since matching takes place inside the sensor, the ESP32 never handles raw fingerprint images.

B. Login/Logout Logic

Fig. 2 outlines the control flow. After start-up checks (LCD, Wi-Fi, NTP and sensor), the main loop keeps polling the sensor. When a match occurs, the firmware flips a per-user IN/OUT flag so that consecutive scans alternate between login and logout. These flags are saved in the ESP32's non-volatile Preferences memory, so the state is preserved across a reboot or power failure. The system then (i) pulses the green LED and buzzer, (ii)

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 11, CONFLUENCE - 2026

displays the user's name, a welcome or thank-you message, the status and the date/time on the LCD, (iii) sends a Telegram message with the name, designation, department or course, date, time and status, and (iv) makes an HTTP GET request to a web-service endpoint that adds the record to the employee or student sheet.

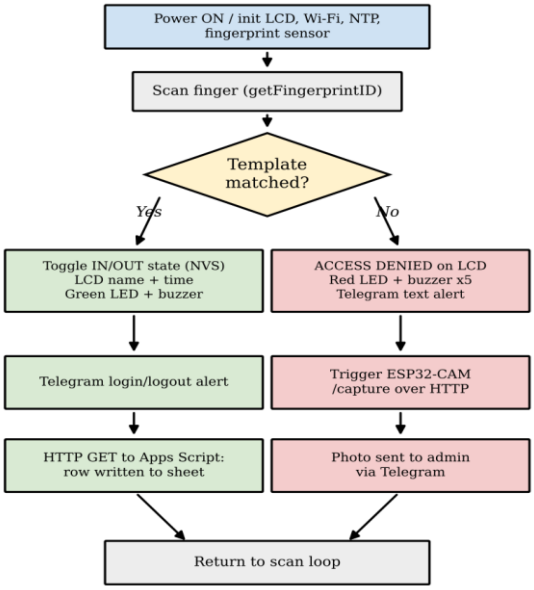


Fig. 2. Control flow of the firmware for authorized and unauthorized scans.

C. Unauthorized Access Handling

When the sensor reports a valid scan that matches no stored template, the ESP32 flashes the red LED and sounds the buzzer five times, shows "ACCESS DENIED – TRY AGAIN" on the display, and sends a Telegram text alert. It then sends an HTTP request to the ESP32-CAM's capture endpoint (30 s time-out). The camera takes a frame and forwards the photograph to the administrator's Telegram chat; on success the camera responds "Photo Sent" and the LCD displays "PHOTO SENT TO ADMIN". If the camera is unreachable, the LCD shows "CAMERA NOT REACHABLE", and an upload failure appears as "TELEGRAM FAILED".

D. Record Keeping and Access Control

Every record contains the name, date, in-time, out-time and either the department (faculty) or the course (students). Faculty and student records are stored on separate sheets in a single workbook ("EmployeeAttendance" and "StudentAttendance"). Only the administrator can access the data, through the sharing permissions of the underlying spreadsheet account.

E. Firmware Timing

The delays hard-coded in the firmware determine the minimum time each interaction takes. For an authorized scan they add up to about 9 s: 0.6 s of LED/buzzer feedback, three LCD screens of 2 s each, 0.5 s before the web request and a closing 2 s display, plus network latency for the Telegram and spreadsheet calls. The unauthorized sequence includes 2 s of red-LED/buzzer alerting (five 0.4 s cycles) ahead of the camera request. These values are derived from the source code rather than from instrumented timing measurements.

V. RESULTS AND DISCUSSION

A. Functional Testing

The prototype was exercised with the enrolled users and with unregistered fingers. Table II summarizes how each function behaved, as documented in the project report.

TABLE II. FUNCTIONAL TEST SUMMARY

Function	Observed outcome
Fingerprint authentication	Enrolled users recognized; attendance marked without manual input
Attendance recording	Login/logout time, date and name stored automatically
LCD display	Name, status, date/time and error messages shown correctly
LED and buzzer	Green + short beep on success; red + repeated buzzer on denial
Intruder capture	ESP32-CAM triggered; photo delivered to administrator
Telegram alerts	Login/logout and unauthorized-access alerts received

All modules worked together without any communication problems during testing. The tests were functional only; false acceptance and false rejection rates, matching latency and camera image quality were not measured and are left for future evaluation.

B. Analysis of the Attendance Log

The workbook generated by the prototype between 17 August and 5 October 2026 was analyzed. It holds 35 records (20 faculty, 15 student) over 12 distinct dates, summarized in Table III. Of these 35 records, 20 (57.1%) include both an IN and an OUT time, whereas 15 (42.9%) have only an IN time, meaning that no logout scan was captured for those entries. Fig. 3 shows the per-user distribution.

TABLE III. ATTENDANCE LOG SUMMARY (17 AUG–5 OCT 2026)

Metric	Faculty	Students	Total
Enrolled users	4	2	6
Records logged	20	15	35
Distinct dates with activity	9	8	12*
Complete (IN + OUT) records	10	10	20
Open (IN only) records	10	5	15
Median IN→OUT interval (min)	12	4	—
Mean IN→OUT interval (min)	9.6	5.1	—

**Dates on which both groups were active overlap, so the total is not the sum.*

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14, Issue 11, CONFLUENCE - 2026

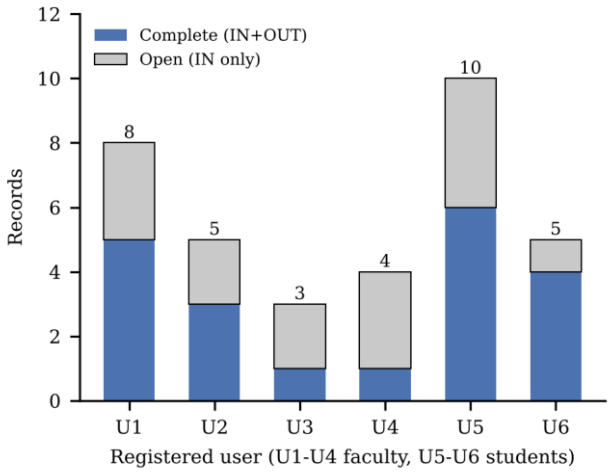


Fig. 3. Records per user, split into complete and open entries.

Several observations can be made. First, every enrolled user appears in the log and no entry has an unknown identity, which agrees with a correct mapping from templates to identities. Second, the IN→OUT intervals are short (up to 20 min, several of them 0–1 min); they come from demonstration and testing sessions rather than real working hours, so they should not be taken as workload data. Third, repeated sessions by the same user on one day (for instance two entries by one student on 24 August and two by one faculty member on 5 October) confirm that the toggle logic handles multiple IN/OUT cycles. Fig. 4 shows that logins took place both in the morning (10–11 h) and in the evening (17–22 h), matching the times when testing was carried out.

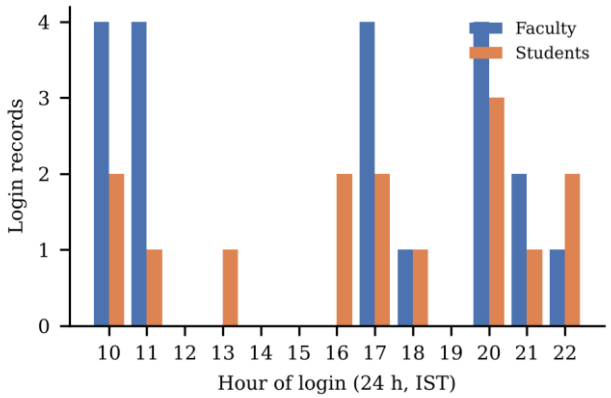


Fig. 4. Distribution of login records by hour of day.

C. Limitations

The large proportion of open records (42.9%) suggests that a logout scan is easy to miss. Since the system depends on strict alternation, a single missed scan can also reverse the meaning of that user's later scans. Further limitations are: (i) the evaluation covers only a small single-site prototype with six users; (ii) the system relies on a Wi-Fi hotspot, and a record is lost if the web request fails while offline; (iii) the Wi-Fi password and bot token are compiled into the firmware and the camera trigger uses plain HTTP on the local network, so credentials should be moved to secure storage and the link protected; and (iv) optical fingerprint sensors can be affected by dirty, wet or damaged fingers [2].

VI. ADVANTAGES AND APPLICATIONS

Relative to manual and card-based methods, NexAttend (a) binds each record to a unique biometric identity, eliminating proxy attendance; (b) automates logging, which saves time and avoids transcription errors; (c) gives users immediate audio-visual confirmation; (d) notifies the administrator remotely in real time; (e) provides photographic evidence of unauthorized attempts; and (f) relies on low-cost, easily available parts and built-in Wi-Fi, so no extra communication hardware is needed. Thanks to its modular design, it fits offices, educational institutions, laboratories, small and medium enterprises, hospitals and government offices that need staff attendance tracking and basic access monitoring.

VII. CONCLUSION AND FUTURE WORK

This paper described NexAttend, an IoT-enabled fingerprint attendance and security monitoring system built on an ESP32 and an ESP32-CAM. The prototype verifies registered faculty and students, records login and logout times automatically, provides local feedback through an LCD, LEDs and a buzzer, informs the administrator via Telegram, and sends a photograph of anyone whose fingerprint is not recognized. The log from live use (35 records over 12 days) demonstrates correct multi-user operation and also reveals that missed logout scans are the chief weakness of the toggle-based design.

Planned future work covers automatic pairing or time-out closure of open sessions, quantitative evaluation (false accept/reject rates and end-to-end latency), offline buffering of records, encrypted credential storage and HTTPS for the camera link, and a cloud database with a web dashboard and mobile app. Possible extensions include face recognition or RFID as a second factor, e-mail/SMS alerts, payroll integration, GPS-based verification, AI-based analysis of attendance patterns and multi-branch support. These are proposals and not features of the current prototype.

REFERENCES

[1] P. P. Ray, "A survey on Internet of Things architectures," *J. King Saud Univ.–Comput. Inf. Sci.*, vol. 30, no. 3, pp. 291–319, Jul. 2018.

[2] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*, 2nd ed. London, U.K.: Springer, 2009.

[3] N. Ratha and R. Bolle, Eds., *Automatic Fingerprint Recognition Systems*. New York, NY, USA: Springer, 2004.

[4] R. Kamal, *Internet of Things: Architecture and Design Principles*. New Delhi, India: McGraw-Hill Education, 2017.

[5] Telegram, "Telegram Bot API." [Online]. Available: https://core.telegram.org/bots/api

[6] Espressif Systems, "ESP32 series datasheet." [Online]. Available: https://www.espressif.com/en/support/documents/technical-documents

[7] Fingerprint Cards AB, "Fingerprint biometrics technology overview," Tech. White Paper, 2022.

[8] How2Electronics, "Fingerprint biometric attendance system using Arduino." [Online]. Available: https://how2electronics.com/fingerprint-biometric-attendance-system-arduino/

(This work is licensed under a Creative Commons Attribution 4.0 International License.)