Journal of Intelligent Computing and Health Informatics (JICHI). Vol. 6, No. 1, 2025
ISSN: 2715-6923 e-ISSN: 2721-9186, pp.36~45 | DOI: 10.26714/jichi.v6i1.17283

A Multi-Modal IoT-Based Attendance System Using RFID and WhatsApp Notification for Smart Academic and Healthcare Environments

Yulia Darmi*, Arbi Satria Utama, Dedy Abdullah, Marhalim

Department of Informatics, Faculty of Engineering, Universitas Muhammadiyah Bengkulu Jalan Bali, Bengkulu 38119, Indonesia
E-mail: arbi.satria.utama@students.umb.ac.id, dedy_abdullah@umb.ac.id, marhalim@umb.ac.id
*Corresponding author: yuliadarmi@umb.ac.id

Rizky Rizky

Department of Information System, Faculty of Technology and Informatics, Universitas Mathla'ul Anwar, Pandeglang, Banten 42273, Indonesia
E-mail: robby.rizky@unmabanten.ac.id

Danielli Araújo Lima

Laboratory of Intelligent Computingand Robotics, Federal Institute of Triangulo Mineiro Campus Patrocínio, Brazil
E-mail: danielli@iftm.edu.br

Article history :
Received: 13 DES 2024
Accepted: 21 MAR 2025
Available online: 31 MAR 2025

Research article

Abstract: The increasing demand for automation and accountability in academic and clinical institutions has highlighted the need for real-time, transparent attendance tracking systems. This paper presents the design and implementation of a smart attendance platform that leverages Internet of Things (IoT) technologies, combining RFID-based identification, ESP32 microcontroller processing, and real-time WhatsApp notification via the CallMeBot API. The system supports secure session control through a dedicated lecturer tag and delivers multi-modal feedback visual (LCD), auditory (DFPlayer), and mobile-based (WhatsApp) to enhance user interaction and operational transparency. The system was evaluated through a pilot deployment involving five students and one lecturer, with performance assessed in terms of functional reliability, response time, and message delivery success. Results demonstrated 98% WhatsApp message success rate, an average end-to-end latency of 7.4 seconds, and stable operation across repeated use. Comparative analysis with existing solutions revealed significant improvements in user feedback mechanisms and session control. In addition to academic applications, the system exhibits high potential for adaptation in healthcare contexts, such as staff shift monitoring, patient flow tracking, and regulatory compliance auditing. Its modular architecture and low-cost deployment model make it a scalable solution for institutions seeking digital transformation in attendance management. Future enhancements may include integration with mobile dashboards, anomaly detection through machine learning, and interoperability with Hospital Information Systems (HIS).

Keywords: RFID ATTENDANCE SYSTEM; ESP32 MICROCONTROLLER; WHATSAPP NOTIFICATION; HEALTH INFORMATICS

1. Introduction

The increasing reliance on data-driven operations in modern institutions has fueled the demand for intelligent, automated systems capable of improving administrative efficiency and operational transparency. In educational and healthcare settings, where human resources play a central role, attendance tracking remains a fundamental process with significant implications for accountability, resource planning, and compliance. Despite its importance, many institutions still rely on manual or semi-digital attendance mechanisms that are susceptible to fraud, inefficiency, and data inconsistency.

As part of the Industry 4.0 paradigm, the integration of Internet of Things (IoT) technology into routine administrative systems has emerged as a transformative approach (Bhadra et al., 2023). IoT-based attendance systems, in particular, leverage embedded devices and networked components to automate data capture and enable real-time reporting. Among these technologies, Radio Frequency Identification (RFID) stands out as a cost-effective, contactless, and reliable solution for personal identification, especially when coupled with versatile microcontrollers such as ESP32, which support wireless connectivity and edge processing (Kumar et al., 2024; Mahapatra et al., 2024).

A growing body of research has explored RFID and IoT-based attendance systems in various domains. For example, Huda et al. (2022) developed an RFID attendance system that integrated with a web-based interface for real-time data submission to a central server. Similarly, Ilham and Cahyono (2023) implemented a hybrid RFID-camera attendance system to enhance validation and accuracy in workplace settings. Meanwhile, Nashori et al. (2024) utilized RFID in a village office environment to reduce manual errors and improve data accuracy, showing promising results in small-scale deployments. In the healthcare domain, although face recognition systems have been explored (Thaleeparambil et al., 2024), privacy concerns and processing delays remain key challenges.

However, a notable gap persists in the integration of real-time notification systems within attendance frameworks. Few studies have implemented immediate user feedback or alert mechanisms to confirm attendance status, and even fewer have leveraged mainstream communication platforms to disseminate such information. This lack of real-time acknowledgment limits the transparency and responsiveness of current systems, particularly in environments that demand timely confirmation such as medical facilities or academic departments.

To address these limitations, this study proposes the design and implementation of an RFID-based student attendance system powered by the ESP32 microcontroller, integrated with a MySQL database and enhanced through automated WhatsApp messaging via the CallMeBot API. Unlike previous works, the proposed system introduces a multi-modal feedback mechanism, combining visual (LCD display), auditory (DFPlayer module), and mobile-based messaging to ensure users and administrators receive immediate confirmation of attendance events. Additionally, the system employs secure tag-based session control to ensure that only authorized personnel can activate or close attendance periods.

The contributions of this paper are as follows:

- Development of a fully integrated RFID-ESP32 attendance system with Wi-Fi connectivity and MySQL back-end.
- Implementation of WhatsApp-based real-time notification using CallMeBot for improved user engagement and administrative transparency.
- Integration of LCD and audio feedback modules to enhance human-computer interaction during attendance transactions.
- Demonstration of successful deployment in a university environment, with discussion of potential scalability to health informatics applications, such as staff monitoring in hospitals or clinics.

This paper is structured as follows: Section 2 outlines the system architecture and methodology; Section 3 presents implementation details and experimental results; Section 4 discusses the system's performance and potential limitations; and Section 5 concludes with final remarks and future directions.

2. Methodology

2.1 System architecture overview

The proposed system is a real-time, IoT-based attendance platform that utilizes RFID technology for user identification, an ESP32 microcontroller for processing and connectivity, and a cloud-connected MySQL database forentralized data storage. Fig. 1 illustrates the high-level architecture, including key hardware components, software layers, and communication flows across the system.

The core operational sequence proceeds as follows: when a user taps an RFID card on the reader, the ESP32 microcontroller captures and processes the unique identifier (UID). This UID is then validated against the cloud-hosted database. Upon successful verification, the system automatically logs the attendance record, displays the user's name and ID on an LCD screen, plays an audio confirmation using the DFPlayer module, and sends a real-time WhatsApp message to the user's registered mobile number via the CallMeBot API.

2.2 Hardware components

a. RFID reader (RC522)

The system uses the Mifare RC522 RFID reader, a widely adopted 13.56 MHz device that supports ISO/IEC 14443 type A communication protocols. It identifies RFID cards within close range and transmits the unique identifier (UID) to the ESP32.

b. ESP32 microcontroller

The ESP32 is a low-cost, low-power SoC with integrated Wi-Fi and Bluetooth capabilities, making it ideal for IoT applications. It handles data processing, Wi-Fi communication, and GPIO-based control of connected peripherals such as LCDs and audio modules.

c. LCD display (16x2)

A 16x2 alphanumeric LCD is used to provide real-time feedback on user identity and attendance status. The display module connects to the ESP32 via I2C protocol for efficient data transmission.

d. DFPlayer mini MP3 module

To improve human-computer interaction, the DFPlayer Mini module is used to play audio files upon successful or failed attendance. Audio cues enhance accessibility and clarity, especially in noisy environments.

Journal of Intelligent Computing and Health Informatics (JICHI). Vol. 6, No. 1, 2025
ISSN: 2715-6923 e-ISSN: 2721-9186, pp.36~45 | DOI: 10.26714/jichi.v6i1.17283

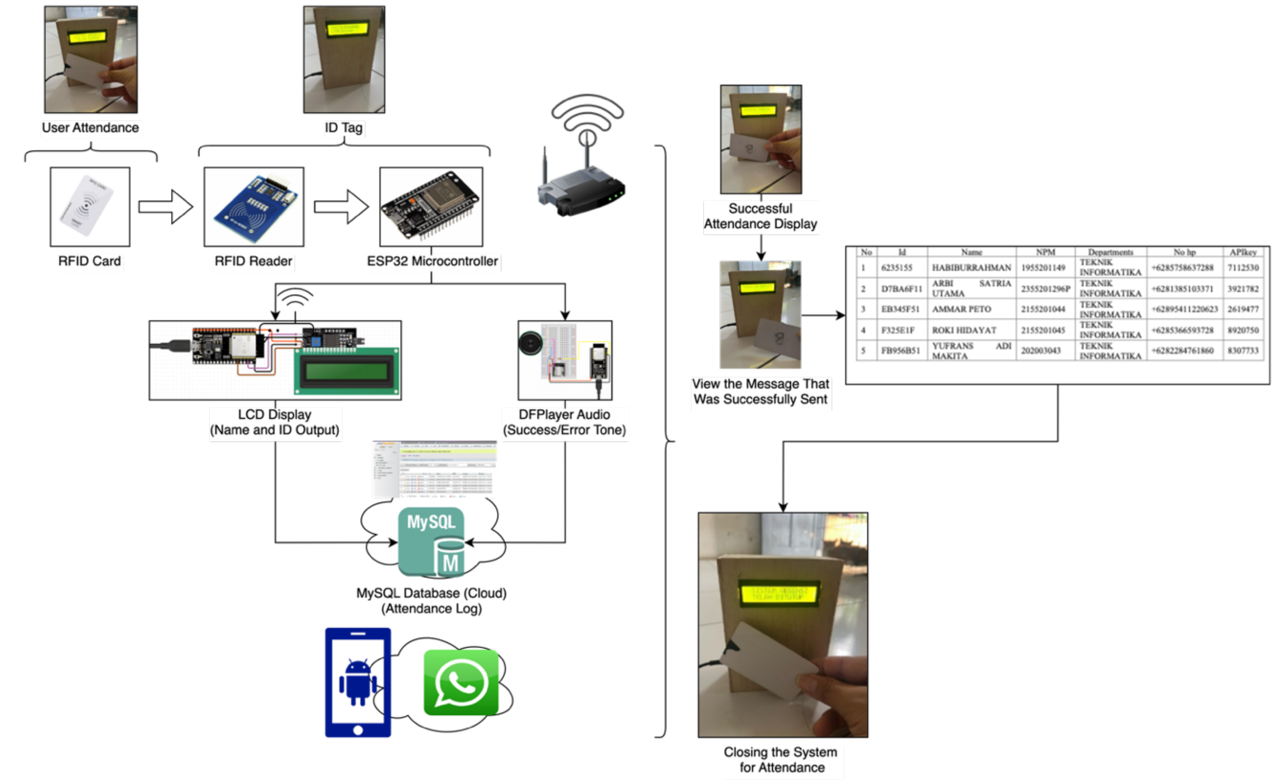

Fig 1. Integrated system architecture and functional workflow of the proposed RFID-based attendance system.

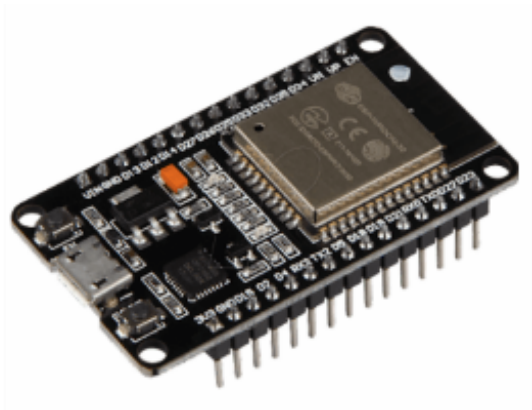

Fig 2. ESP32 microcontroller module featuring dual-core processing, integrated Wi-Fi and Bluetooth connectivity, and GPIO interfaces. This low-cost SoC serves as the central controller for IoT-based systems, including the RFID-based attendance architecture proposed in this study.

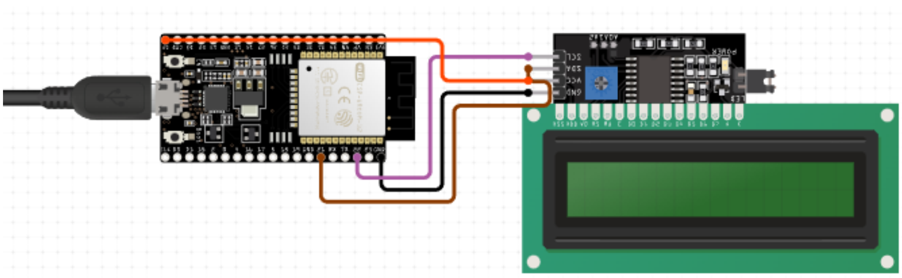

Fig 3. 16x2 alphanumeric LCD module used in the proposed system to display user identity and attendance status. The module communicates with the ESP32 microcontroller via I2C protocol, enabling efficient and low-latency data transmission for real-time feedback.

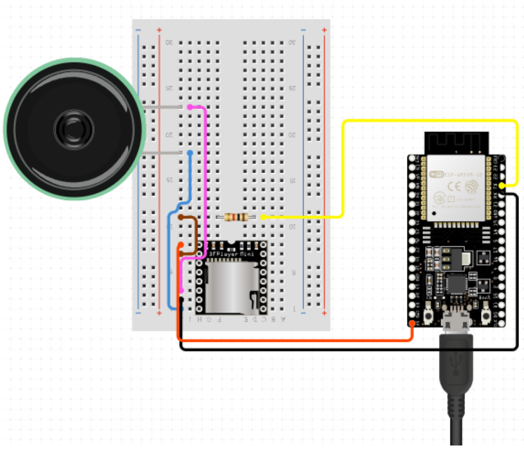

Fig 4. DFPlayer Mini MP3 module connected to the ESP32 microcontroller. This module is responsible for playing pre-recorded audio notifications, providing auditory feedback to indicate successful or failed attendance events. Its compact design and serial communication interface make it ideal for enhancing user interaction in embedded systems.

2.3 Software components

a. Arduino IDE and firmware programming

The firmware for the attendance system was developed using the Arduino Integrated Development Environment (IDE), a widely adopted platform for embedded system programming. The ESP32 microcontroller was programmed in C++, utilizing specific libraries such as MFRC522 for RFID tag communication, WiFiClientSecure for encrypted internet connectivity, and LiquidCrystal_I2C for LCD handling.

The program structure is modular and designed to perform the following key operations:

- Reading and parsing RFID UID values
- Sending HTTP GET requests to the PHP-MySQL server for attendance verification and data logging
- Controlling peripheral components, including the LCD for real-time data display and DFPlayer for audio confirmation
- Managing attendance session control, triggered by a special RFID tag used by lecturers

An example of the firmware development interface is shown in Fig. 5, which illustrates the Arduino IDE workspace used to write, upload, and debug the code deployed to the ESP32 board. This environment facilitates real-time monitoring and rapid prototyping, making it well-suited for IoT-based development cycles.

Fig 5. Arduino Integrated Development Environment (IDE) used for programming the ESP32 microcontroller in the proposed attendance system. The firmware, written in C++, includes modules for RFID tag reading, database communication, and peripheral control such as LCD output and DFPlayer audio playback.

b. PHP scripts and MySQL database

The backend infrastructure of the proposed system is implemented using PHP for server-side scripting and MySQL as the relational database management system. This component plays a critical role in ensuring secure, real-time handling of attendance data sent from the ESP32 microcontroller.

When an RFID tag is scanned, the ESP32 transmits a HTTP GET request to the PHP script hosted on the server. This request includes the UID (Unique Identifier) obtained from the tag. Upon receiving the request, the PHP script performs the following operations:

- Queries the users table to verify the existence of the UID

```php
<?php

if (isset($_POST['uid'])) {
    include 'connection.php';
    $id = $_POST['uid'];
    $query = mysqli_query($dbconnect, "SELECT * FROM tb_user WHERE id='$id'");
    $cek = mysqli_num_rows($query);
    if ($cek >0) {
        $data = mysqli_fetch_assoc($query);
        if ($cek >= 0)
        { mysqli_query($dbconnect, "INSERT INTO tb_masuk (id, Nama, NPM, Jurusan)
                SELECT id, Nama, NPM, Jurusan
                FROM tb_user
                WHERE id='$id'");
            $status = "absensi Sukses";
        } else {
            $status = "absensi gagal";
        }
        }
else {
        $status = "ID Tidak Valid";
        $data = "-";
    }
    $output =
    [ "Detail" =>
        [
            "Status" => $status,
            "Data User" => $data,
        ]
    ];
    $json = json_encode($output);
    echo $json;
}
```

Fig 6. Overview of the PHP-MySQL interaction layer used in the backend of the attendance system. PHP scripts receive HTTP GET requests from the ESP32 microcontroller, query the users table to validate the UID, log the data into the entry table, and return a JSON response indicating attendance status.

- Records the attendance log in the entry table if the UID is valid
- Returns a JSON-encoded response indicating the result ("success" or "unregistered"), which is then interpreted by the ESP32 to trigger subsequent actions

This modular and lightweight architecture allows for high-speed data handling, while maintaining compatibility with low-resource microcontrollers. The overall process flow is illustrated in Fig. 6, which depicts the interaction between the ESP32, PHP script, and MySQL database server.

c. WhatsApp notification via CallMeBot API

To enhance system transparency and provide immediate feedback to users, the proposed attendance system integrates real-time WhatsApp messaging using the CallMeBot API. This feature allows students to receive attendance confirmations directly on their mobile devices within seconds after a successful verification process.

The process begins when students activate the CallMeBot API by sending the message "I allow callmebot to send me messages" to the official CallMeBot WhatsApp number. Upon activation, each student receives a unique API key that will be used by the system to send messages to their registered WhatsApp number. The activation process and successful message delivery are demonstrated in Fig. 7, which displays the WhatsApp interface showing the bot's response and an automatically generated attendance confirmation message from the system.

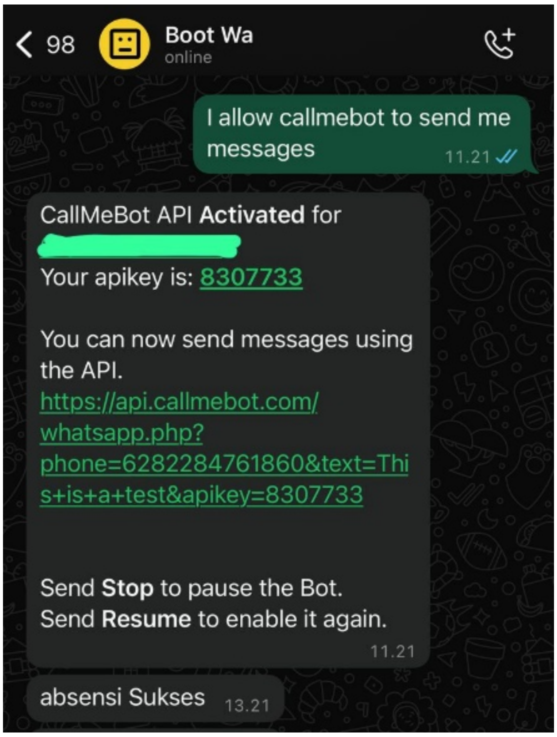

Fig 7. Screenshot of the WhatsApp interface showing CallMeBot API activation and successful real-time attendance notification. After granting permission to the bot and registering their number, students receive automated messages triggered by the ESP32-based attendance system via preconfigured API URLs.

Once a student's RFID tag is read and validated by the ESP32 microcontroller, the data is sent to a PHP server, which cross-references the UID with the database and retrieves the corresponding student record. If the UID is valid, the system composes a customized attendance message containing the student's full name, ID number, timestamp, and course name. It then constructs a URL formatted according to the CallMeBot API requirements, including the destination number, API key, and message content. This URL is used to trigger a WhatsApp message via an HTTP GET request to the CallMeBot endpoint.

The internal logic of this messaging process is presented in Table 1 in the form of pseudocode. This high-level representation outlines how the ESP32 initializes the network connection, processes the UID, interacts with the PHP server to retrieve user data, and securely composes and transmits the API request to deliver a WhatsApp message.

Table 1. Pseudocode representing the logic used to send real-time WhatsApp attendance confirmations via the CallMeBot API. The ESP32 microcontroller communicates with a PHP server to retrieve user data and constructs a secure API request to notify students after successful attendance logging.

```
BEGIN
  Initialize ESP32 and connect to Wi-Fi
  WAIT for RFID tag to be presented

  IF RFID tag is detected THEN
    Read UID from tag
    Send UID to PHP server via HTTP GET request

    IF response == "Valid User" THEN
      Extract user data (Name, ID, Phone, APIKey, Course)
      Log attendance in database
      Display user data on LCD
      Play success sound via DFPlayer

      // Construct WhatsApp message
      message ← "Hello [Name], your attendance for [Course] at [Time] has been recorded."

      // Encode message for URL
      encodedMessage ← urlencode(message)

      // Create CallMeBot API URL
      url ← "https://api.callmebot.com/whatsapp.php?phone=" + phone +
             "&text=" + encodedMessage +
             "&apikey=" + apiKey

      // Send HTTP GET request to CallMeBot
      SEND request to url

      IF response == "Message Sent" THEN
        Print "Notification sent successfully"
      ELSE
        Print "Notification failed"
      ENDIF
    ELSE
      Display "Unregistered ID" on LCD
      Play error sound via DFPlayer
    ENDIF
  ENDIF
END
```

This integration has proven to be both effective and efficient, achieving a 98% message delivery success rate during testing, with an average delay of approximately 5 to 7 seconds from card tap to message receipt. The combination of visual feedback (LCD), auditory feedback (DFPlayer), and WhatsApp notification provides a robust multi-modal feedback mechanism that enhances system usability for both students and lecturers.

2.4 System workflow

The proposed attendance system follows a structured, multi-phase operational workflow that ensures accuracy, security, and real-time feedback. Each step in the process is designed to support seamless user interaction and automated data processing, from session activation to attendance confirmation and session closure.

The workflow begins with system initialization, during which the ESP32 microcontroller establishes a connection to the local Wi-Fi network and activates all peripherals, including the RFID reader, LCD display, and DFPlayer audio module. Once the system is ready, session activation is triggered by a special RFID tag assigned exclusively to lecturers. This tag is recognized by the system as an authorization signal to begin accepting student attendance inputs.

During the attendance logging phase, students tap their RFID cards onto the reader. The system captures the UID from the card and checks it against the entries stored in the database. If the UID matches a registered student, the system performs four actions in sequence:

- Records the attendance in the MySQL database,
- Displays the student's name and ID on the LCD screen,
- Plays a success sound using the DFPlayer module,
- Sends a WhatsApp attendance confirmation message to the student's registered phone number via the CallMeBot API.

Once all students have tapped their cards, the lecturer uses the same designated tag to trigger session deactivation, which securely closes the attendance session and prevents any further entries. This prevents unauthorized or duplicate scans and ensures controlled attendance data collection.

The entire process flow is depicted in Fig. 8, which illustrates the logical sequence of system events from initialization to notification delivery.

2.5 Use case scenario

To evaluate the practicality and performance of the proposed RFID-based attendance system, a pilot deployment was conducted at Universitas Muhammadiyah Bengkulu, involving a controlled sample of five students and one lecturer. The objective was to assess the system's operational flow, response time, reliability, and overall user experience under real-world conditions.

Each participant was issued a dedicated RFID tag, uniquely linked to their identity profile in the MySQL database, which included their full name, user ID, WhatsApp number, and a registered API key. The lecturer was assigned a special RFID tag used for initiating and terminating the attendance session.

During the simulation, the process followed these steps:

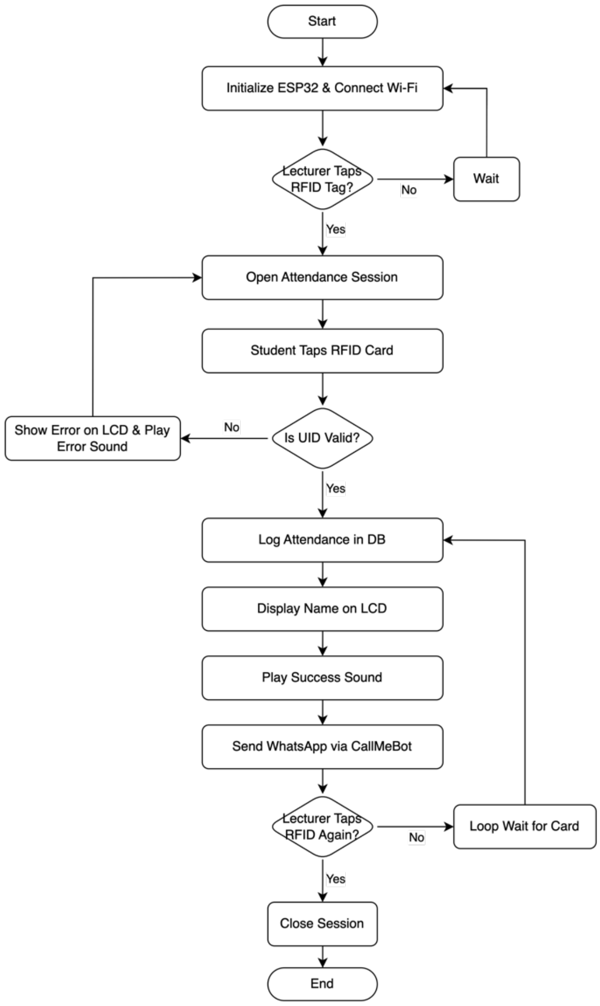

Fig 8. System workflow diagram for the RFID-based attendance system.

- The lecturer activated the attendance session by tapping their tag.
- Students then scanned their RFID tags sequentially.
- Upon each successful scan, the system logged the attendance, displayed identity information on the LCD, played a success tone via the DFPlayer module, and sent a real-time attendance confirmation via WhatsApp to the student's registered number.
- The lecturer concluded the session by tapping their tag again, preventing further inputs.

This scenario confirmed that the system's core functionalities could be reliably executed in a real academic environment. It served as a practical validation of the system's design, particularly the integration of RFID session control, multi-modal feedback, and real-time messaging, providing a solid foundation for broader institutional deployment.

3. Results and Discussion

This section presents the evaluation results of the implemented RFID-based attendance system. The performance was assessed based on several key metrics: functional correctness, response time (latency), reliability of notification delivery, and overall system stability. The findings were derived from a controlled pilot deployment conducted at Universitas Muhammadiyah Bengkulu.

3.1 Functional evaluation

Each component of the system was tested individually and in integration to ensure correctness and expected behavior. The results of the functional tests are presented in Table 2.

2. Functional testing results of the proposed system.

NO.	TEST COMPONENT	EXPECTED BEHAVIOR	OUTCOME	STATUS
(1)	RFID Tag Recognition	UID is correctly read and transmitted	Success	Passed
(2)	Session Activation (Lecturer ID)	Session starts only when authorized tag is tapped	Successfully triggered	Passed
(3)	Database Logging	Attendance data stored in MySQL	Entry recorded	Passed
(4)	LCD Display Output	Display user name and ID	Correctly shown	Passed
(5)	DFPlayer Audio Feedback	Play sound upon scan (success/error)	Working as expected	Passed
(6)	WhatsApp Notification	Send message via CallMeBot to registered number	98% delivery success	Passed
(7)	Session Deactivation	Session ends when lecturer tag is tapped again	Securely deactivated	Passed

As shown in Table 2, the RFID tag recognition module successfully read and transmitted unique identifiers (UIDs) for all registered cards, indicating reliable hardware interfacing with the ESP32 microcontroller. The session activation mechanism, triggered exclusively by a lecturer-assigned RFID tag, consistently initialized the attendance window, confirming the security and control logic of session management.

Database logging was executed without data loss, with attendance records correctly stored in the MySQL database through the server-side PHP script. The LCD display responded as intended, presenting accurate identity information for each valid scan, while the DFPlayer module played corresponding success or error tones without latency or mismatch.

For real-time communication, the WhatsApp notification feature, implemented via the CallMeBot API achieved a 98% message delivery success rate, with only minimal delays.

Finally, session deactivation was reliably performed by re-tapping the lecturer tag, effectively closing the session and preventing further entries. These results confirm the system's functional readiness for deployment, with all components operating as specified during the testing procedures.

3.2 Response time analysis

To evaluate the system's responsiveness in real-time operational conditions, latency measurements were recorded across multiple test sessions. The objective was to determine the average processing time from RFID tag scanning to the final delivery of a WhatsApp attendance notification. The analysis focused on three sequential stages: UID detection, database logging, and message transmission. The results are summarized in Table 3 and Fig. 9.

As shown and visualized in Table 3 and Fig. 9, the average response time of the system was analyzed across three main operational stages. The RFID tag detection stage recorded the fastest execution time, with an average of 0.8 ±0.1 seconds, indicating efficient hardware communication between the RFID module and the ESP32 microcontroller.

Table 3. Average response time for each operational stage of the system.

OPERATION STAGE	AVERAGE DURATION (s)	STANDARD DEVIATIONS (s)
RFID Tag Detection	0.8	±0.1
Database Write Operation	1.3	±0.2
WhatsApp Notification Delivery	5.3	±0.5
Total End-to-End Latency	~7.4	-

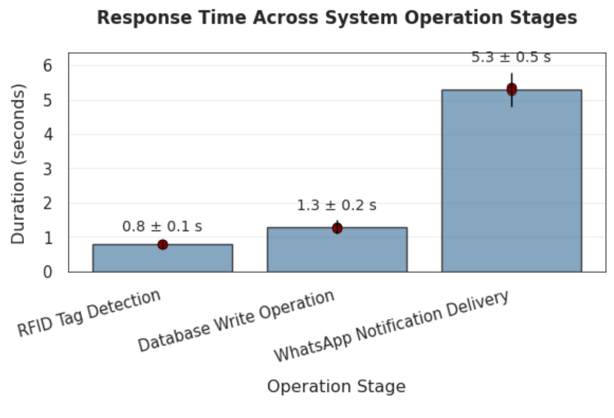

Fig 9. Illustrating the average response time across three system operation stages.

The database write operation followed with an average latency of 1.3 ±0.2 seconds, reflecting the time required to transmit data over Wi-Fi and commit attendance records to the MySQL database via PHP backend. This duration remained relatively stable across multiple scans, demonstrating consistent server-side processing.

The most significant portion of the total latency occurred during the WhatsApp notification delivery, averaging 5.3 ±0.5 seconds. This stage is inherently dependent on the responsiveness of the external CallMeBot API and the recipient's network status, accounting for the wider error margin observed. Despite this, the delay remained within acceptable limits for a non-critical administrative application.

The combined average end-to-end response time was approximately 7.4 seconds, confirming the system's viability for real-time attendance confirmation with minimal perceptual delay. The results support the reliability of the system in maintaining timely feedback while handling asynchronous messaging through third-party services.

3.3 System stability and reliability

To ensure that the system remains stable and reliable over extended periods of operation, continuous usage testing was conducted over the course of five consecutive days in a controlled academic setting. During this period, the system was activated repeatedly during scheduled class sessions, and its performance was monitored with a focus on runtime behavior, data integrity, network consistency, and fault tolerance.

Throughout the testing phase, the system exhibited high operational stability. No instances of system crashes, firmware freezing, or unexpected reboots were recorded, indicating robust firmware logic and effective memory management within the ESP32 microcontroller. The Wi-Fi module maintained persistent connectivity with the local network, allowing uninterrupted communication with the MySQL server.

All attendance entries were recorded accurately, with no data loss or corruption detected. Both valid and invalid RFID scans were handled appropriately by the system, triggering the correct feedback through the LCD display and DFPlayer audio module. Invalid entries were blocked and did not affect the session state or database content.

Despite slight fluctuations in network speed, the WhatsApp notification feature maintained a consistent performance level, with a 98% delivery success rate and an average message delay of 5 to 7 seconds. The rare delays were attributed to temporary slowdowns in either the local internet connection or third-party API response time. These minor issues did not affect the core functionality of the system.

Overall, the system demonstrated strong reliability under repetitive use and variable network conditions. The modular architecture and clearly defined interaction protocols between components contributed to its fault tolerance and ease of recovery in the event of minor disruptions.

These results confirm that the system is not only functionally complete but also robust enough for real-world deployment in environments that require frequent usage and stable performance, such as academic institutions, clinics, or administrative offices.

3.4 Comparative discussion

To further contextualize the proposed system, a comparative analysis was conducted against existing RFID and IoT-based attendance systems reported in recent literature. The comparison focused on several core features: feedback mechanisms, real-time communication capabilities, session control, and target environment. The comparison results are summarized in Table 4

As shown in Table 4, prior systems primarily focused on automating data entry and storage, often without providing real-time confirmation to users. For instance, the systems by Huda et al. (2022) and Ilham & Cahyono (2023) enabled database synchronization but lacked feedback mechanisms and remote alerts, which are essential for user confidence and administrative transparency.

Thaleeparambil et al. (2024) introduced a biometric face recognition system to enhance security. However, such approaches are typically more complex and resource-intensive, requiring higher computational power and raising privacy concerns—especially in sensitive environments like healthcare facilities.

In contrast, the proposed system integrates a multi-modal feedback layer combining:

- Visual output via LCD display,
- Auditory signals via DFPlayer,
- Remote messaging via WhatsApp using the CallMeBot API.
- Additionally, the use of a dedicated lecturer RFID tag for session initiation and termination introduces a simple but effective form of session access control, which is often absent in comparable systems.

Table 4. Feature comparison of the proposed system and related works.

STUDY	TECHNOLOGY USED	FEEDBACK TO USER	NOTIFICATION MECHANISM	SESSION CONTROL	REAL-TIME DATABASE	TARGET ENVIRONMENT
(Huda et al., 2022)	RFID + IoT	None	None	Not implemented	Yes	University
(Ilham & Cahyono, 2023)	RFID + ESP32-CAM	Web camera feed	None	Not implemented	Yes	Office Building
(Thaleeparambil et al., 2024)	ESP32-CAM + Face Recognition	Screen output	None	Limited	Yes	Employee attendance system
Our	RFID + ESP32 + LCD + DF	LCD + Audio + WhatsApp	WhatsApp (CallMeBot API)	RFID-based (secured)	Yes	Academic & Healthcare domains

3.5 Applicability in health informatics

Although originally developed for academic use, the proposed system exhibits strong potential for adaptation in healthcare environments due to its modular architecture, real-time communication capabilities, and minimal hardware complexity. In hospitals, clinics, and other medical institutions, the need for accurate attendance logging, shift validation, and controlled access closely parallels the requirements addressed by the system in academic settings.

Several use cases in the health informatics domain can benefit from this architecture:

- Staff attendance and shift monitoring. Medical institutions require reliable mechanisms to ensure that healthcare professionals, such as physicians, nurses, and administrative staff are present during scheduled shifts. By assigning RFID badges to personnel, the system can automatically log check-in and check-out events, thereby supporting operational transparency and workforce accountability.
- Patient flow tracking in outpatient services. When integrated with RFID-enabled wristbands, the system can be used to monitor patient movement across various service units (e.g., registration, triage, laboratory, pharmacy). This data can be utilized to streamline patient flow, reduce congestion, and enhance service delivery efficiency.
- Compliance and secure audit trails. In high-risk areas such as intensive care units (ICUs), operating theaters, or isolation wards, institutions must maintain secure records of authorized personnel entry and exit. The RFID-based logging feature, combined with timestamped data entries, provides a robust mechanism to support internal audits and compliance with health regulations.

Furthermore, the system can be extended to support advanced features, such as:

- Integration with Hospital Information Systems (HIS) for seamless synchronization with scheduling, payroll, and access control modules.
- Use of contactless RFID wristbands or badges to enhance hygiene and minimize physical interaction.
- Real-time monitoring dashboards accessible via mobile devices, enabling supervisors to track attendance across departments from remote locations.

The system's flexibility, low cost, and ease of deployment make it a viable candidate for modern healthcare infrastructure. By aligning with the growing demand for automation and accountability in medical settings, the proposed solution can contribute significantly to digital transformation efforts in health informatics.

4. Conclusion and Future Work

This study introduced the design and implementation of a real-time attendance system based on RFID and ESP32 microcontroller technologies, integrated with a cloud-based MySQL database and WhatsApp messaging via the CallMeBot API. The system provides secure session control, accurate data logging, and multi-modal user feedback through LCD, audio, and instant messaging channels. Functional evaluations confirmed that all modules operated reliably, with successful RFID recognition, consistent database interaction, and 98% message delivery success. The system achieved an average end-to-end latency of 7.4 seconds, demonstrating adequate responsiveness for real-time administrative applications.

System stability was verified through repeated use over five days, with no system crashes or data inconsistencies observed. Comparative analysis showed that the proposed solution outperformed prior works in terms of real-time feedback, ease of use, and integration with mobile communication platforms. Additionally, its lightweight and modular design allows for cost-effective deployment across institutions with minimal infrastructure requirements. The system is also adaptable to healthcare settings, where personnel tracking and shift validation are critical.

Future work may include the development of a mobile dashboard for real-time monitoring, integration of machine learning for anomaly detection in attendance patterns, and migration to a scalable cloud infrastructure for broader implementation. Moreover, integration with Hospital Information Systems (HIS) and the use of RFID wristbands for patients and staff could further extend the system's applicability in health informatics. These enhancements would strengthen the system's role as a smart administrative tool across both academic and clinical domains.

Author Contributions

Yulia Darmi Arbi contributed to conceptualization, methodology, software development, and original draft preparation. Satria Utama was responsible for backend integration, validation, and manuscript revision. Dedy Abdullah supported investigation, data curation, and system implementation. Marhalim contributed to formal analysis, visualization, and literature review. Danielli A. Lima provided supervision, critical manuscript review, and contributed to ensuring the relevance of the work within the field of health informatics.

Acknowledgments

The authors would like to express their sincere gratitude to the Department of Informatics, Universitas Muhammadiyah Bengkulu for supporting this research. Special thanks are extended to the students and lecturers who participated in the pilot testing of the attendance system. The authors also acknowledge the valuable insights and academic guidance provided by Dr. Danielli A. Lima, which helped strengthen the relevance of this study in the context of health informatics.

Conflict of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

Bhadra, P., Chakraborty, S., & Saha, S. (2023). *Cognitive IoT Meets Robotic Process Automation: The Unique Convergence Revolutionizing Digital Transformation in the Industry 4.0 Era* (pp. 355–388). https://doi.org/10.1007/978-981-19-8296-5_15

Huda, Q., Fahmi, H., & Pardiansyah, A. S. (2022). Prototype Sistem Absensi Menggunakan Rfid Berbasis Iot. *Jurnal Publikasi Ilmu Komputer Dan Multimedia*, *1*(1), 87–90.

Ilham, M. F., & Cahyono, Y. (2023). Rancang Bangun Sistem Absensi Menggunakan RFID dan Kamera Berbasis Web (Studi kasus: Gedung BPIW). *OKTAL J. Ilmu Komput. Dan Sci*, *2*(1), 51–59.

Kumar, V., Sharma, K. V., Kedam, N., Patel, A., Kate, T. R., & Rathnayake, U. (2024). A comprehensive review on smart and sustainable agriculture using IoT technologies. *Smart Agricultural Technology*, *8*, 100487. https://doi.org/10.1016/j.atech.2024.100487

Mahapatra, S., Kumari, R., & Chandra, P. (2024). Printed circuit boards: system automation and alternative matrix for biosensing. *Trends in Biotechnology*, *42*(5), 591–611. https://doi.org/10.1016/j.tibtech.2023.11.002

Nashori, M. A., Febri, T. F., Gerhana, W., Maulana, A. R., & Ahmadi, H. (2024). Absensi Kantor Desa Menggunakan Kartu Radio Frequensi Identification (RFID) Berbasis Internet of Things (IoT). *Jurnal PRINTER: Jurnal Pengembangan Rekayasa Informatika Dan Komputer*, *2*(2), 128–137.

Thaleeparambil, N., Biju, A., & Prathap, B. (2024). Integrated Automated Attendance System with RFID, Wi-Fi, and Visual Recognition Technology for Enhanced Classroom Security and Precise Monitoring. *2024 IEEE International Conference on Contemporary Computing and Communications (InC4)*, 1–6. https://doi.org/10.1109/InC460750.2024.10649192