SisInfo
e-ISSN: 2655-867X
p-ISSN: 2655-8661
Volume 8 No. 2 August 2026

Implementation of an Internet of Things (IoT) Based Digital Residential Gate Security System

Shalina Kalifa Putri, Fahmi Arnes
Information Technology, Universitas Binaniaga Indonesia
Email: shalinakalifa27@gmail.com; fahmi.arnes@unbin.ac.id.

Accepted:	Accepted After Revision:	Published:
30 July 2026	25 August 2026	28 August 2026

Abstract

This study aims to implement an Internet of Things (IoT)-based digital security system for residential gates to improve access management efficiency and real-time monitoring. The system integrates an ESP32 microcontroller, DC gearbox motor, limit switch, Network Time Protocol (NTP), Telegram Bot, Node.js server, MySQL database, and web dashboard to support PIN-based access verification, automatic gate control, real-time status notifications, and digital access activity logging. The main contribution of this study is the practical integration of hardware and software components to support more efficient and manageable residential access security. System testing resulted in 100% functional feasibility and 100% accuracy, while usability testing using the System Usability Scale (SUS) produced an average score of 80, categorized as Grade B (Good). These results indicate that the system performs its main functions according to the specified requirements. However, the evaluation was conducted on a limited prototype scale with a limited number of SUS respondents, so the usability results cannot yet be generalized broadly. The evaluation also did not include quantitative measurements of communication latency and network reliability under short-term and long-term usage scenarios. Future research should involve a larger number of users and evaluate communication latency and network reliability under various operating conditions.

Keywords: ESP32, Internet of Things, gate security system, Telegram Bot, access control, system integration, limit switch.

1 INTRODUCTION

Residential gate security systems commonly used today still rely on conventional methods such as physical keys, remote controls, or simple push-button mechanisms. These methods present several limitations, including the risk of lost keys, inconvenience when residents are away from home, dependence on security personnel, and the absence of a digital system for recording entry and exit activities.

According to the 2024 Criminal Statistics published by the Indonesian Central Bureau of Statistics (BPS) [1], crimes against property in Indonesia reached 140,076 cases in 2023, with West Java recording the third-highest number of cases (14,443 incidents). These statistics highlight the urgent need for a more intelligent, integrated, and real-time residential gate security system.

The Internet of Things (IoT) offers significant opportunities to address these challenges. IoT is a network of interconnected electronic devices capable of transmitting and receiving data over the Internet [2]. The IoT concept is divided into three perspectives [3]: Thing-Oriented Vision, emphasizing physical devices; Internet-Oriented Vision, focusing on network connectivity; and Semantic-Oriented Vision, concerning meaningful data processing [3].

The concept of an IoT-based fence security control system [4] is increasingly relevant in residential, boarding house, and corporate environments, given the need for more efficient and secure access management solutions. This system reduces reliance on inefficient manual methods for managing entry permits. Through a Telegram Chatbot [5] as an interface for the permit system, users can easily submit access requests through a digital platform, which will then be automatically verified by the system.

A security control system prototype was successfully designed and developed using an Arduino Uno [6]. This system can automatically open and close a gate using a DC motor controlled by an Arduino. Test results showed that the system can move objects weighing up to 69.02 kg, although the prototype was only tested up to 50 kg.

Several other studies have been carried out by integrating ESP32, DC motor gearboxes, limit switches, Node.js servers, and Telegram Bots [5], [7], [8], [9], [10], [11], [12], [13], [14], resulting in a gate security system capable of verifying identity, automatically recording access activity, and providing real-time notifications to boarding house owners or guards.

2 LITERATURE REVIEW

2.1 Research and Development (R&D)

Research and Development (R&D) is a systematic methodology used to develop products while simultaneously evaluating their effectiveness. The R&D process consists of several stages, including problem analysis, system design, prototype development, implementation, testing, evaluation, and product refinement to produce an innovation ready for practical application [15].

2.2 Internet of Things (IoT)

The Internet of Things (IoT) enables physical devices to communicate with one another through the Internet, allowing automatic data exchange with minimal human intervention [2]. The IoT paradigm consists of three primary perspectives [3]: Thing-Oriented Vision, focusing on hardware devices; Internet-Oriented Vision, emphasizing communication infrastructure and networking; and Semantic-Oriented Vision, concentrating on intelligent information processing and interpretation.

2.3 ESP32 Microcontroller

The ESP32 is a high-performance System-on-Chip (SoC) featuring built-in Wi-Fi and Bluetooth Low Energy (BLE) connectivity. It supports WebSocket communication and can be programmed using the Arduino IDE with the C++ programming language, making it highly suitable for IoT applications [16].

2.4 DC Gearbox Motor and L298N Motor Driver

The DC gearbox motor functions as the mechanical actuator responsible for opening and closing the residential gate [17]. The L298N motor driver serves as an interface between the ESP32's low-power digital control signals and the higher current requirements of the DC motor, enabling precise control of motor direction and rotational speed [18].

2.5 Limit Switch

A limit switch is an electromechanical sensor activated through physical contact [7]. In this system, two limit switches are installed at the fully opened and fully closed positions of the gate. These switches provide feedback signals to the ESP32, allowing the motor to stop automatically once the gate reaches a safe operating position.

2.6 Telegram Bot API and Node.js

The Telegram Bot API enables the development of automated bots capable of interacting with users [5]. A Node.js server built with Express.js functions as the backend system, coordinating communication among the ESP32, Telegram Bot, and the web-based dashboard through the WebSocket protocol to achieve real-time data exchange [19].

3 RESEARCH METHODS

3.1 Research Method

This study employed the Research and Development (R&D) methodology [15]. The research process consisted of six stages: 1) Problem identification through observation and interviews; 2) Data collection and literature review; 3) System design; 4) Prototype development; 5) System testing and evaluation; and 6) Final product revision.

3.2 Proposed Model

The proposed system was developed based on three complementary models: 1) Theoretical Model (flowcharts system); 2) Conceptual Model, developed according to the three IoT visions [3]; and 3) Procedural Model (the stages of system development).

3.2.1 Conceptual Model

The conceptual model is based on the three IoT perspectives [3]: 1) Thing-Oriented Vision, including: ESP32 microcontroller, DC gearbox motor, L298N motor driver, and Limit switches; 2) Internet-Oriented Vision, consists of: ESP32 Wi-Fi connectivity, Node.js server, WebSocket protocol, and Telegram Bot API; 3) Semantic-Oriented Vision, including: Automatic PIN verification, MySQL access log management, Web-based monitoring dashboard, and Telegram notification services.

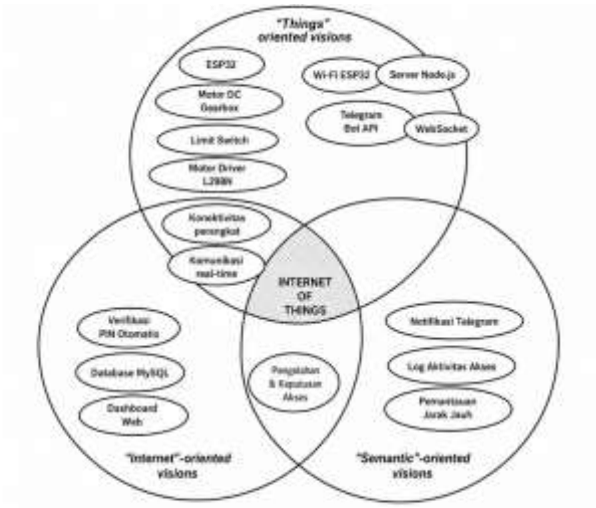

Figure 1. Conceptual Model Based on the Three IoT Perspectives [3]

3.2.2 Procedural Model

The procedural model illustrates the stages followed throughout the system development process: Requirements identification → Prototype development → System testing → Evaluation → Final product validation.

3.3 Product Testing

3.3.1 Data Types

This study utilized quantitative data, included Black-box Testing scores, System Usability Scale (SUS) scores, and System accuracy test results.

3.3.2 Data Collection Instruments

Three instruments were employed in this research: Black-box Testing questionnaire, using the Guttman scale (Valid = 1; Invalid = 0); System Usability Scale (SUS) questionnaire, consisting of ten statements evaluated using a five-point Likert scale; System accuracy testing sheet, used to assess the operational accuracy of the developed system. The product feasibility percentage was calculated using the following formula:

$$FeasibilityPercentage(\%) = \left(\frac{Expected\ Score}{Observed\ Score}\right) x\ 100\%$$

Table 1. System Usability Scale (SUS) Classification

Skor SUS	Grade	Predikat
>80,3	A	Excellent
68-80,3	B	Good
51-68	C	Okay
<51	D/F	Poor/Awful

Source: J. Brooke [20]

3.3.3 Data Analysis Techniques

A descriptive quantitative analysis approach was employed. The collected data were analyzed as follows: Black-box Testing results were evaluated using the product feasibility percentage (0–100%). System accuracy was calculated using the following equation [21]:

$$Accuracy(\%) = \left(\frac{Number\ of\ Accurate\ Results}{Total\ Number\ of\ Tests}\right) x\ 100\%$$

The analysis determined the system's functional validity, usability, and operational reliability.

4 RESULTS AND DISCUSSION

4.1 Result

4.1.1 Requirements Analysis

The requirements analysis was conducted through direct observations and interviews with boarding-house residents and the property caretaker. The findings revealed that gate access was still managed manually using physical keys, with a curfew enforced at 10:00 PM. Residents returning after curfew were required to wait for the caretaker to unlock the gate. Moreover, the existing system lacked an access logging mechanism, allowing anyone possessing a key to enter without identity verification.

To address these shortcomings, an IoT-based digital gate security system was proposed. The comparison between the existing and the proposed business processes is presented below.

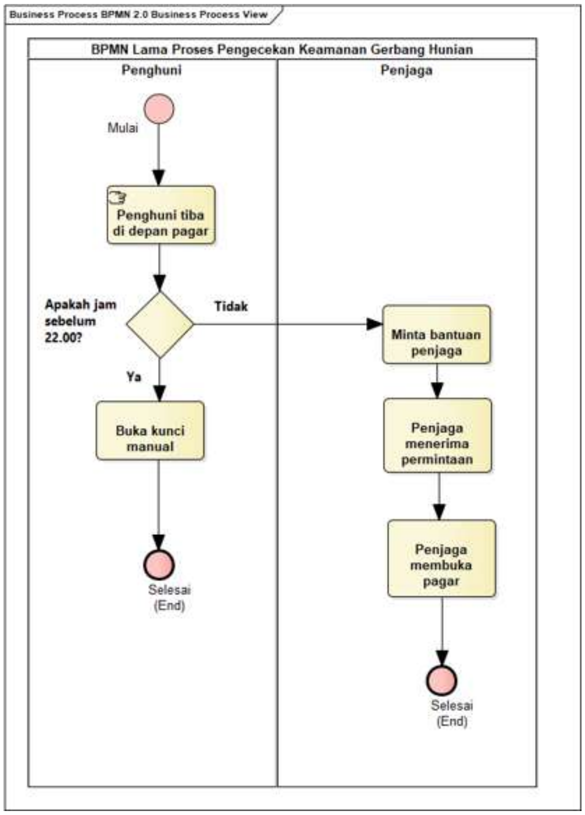

Figure 2. Existing Business Process of Residential Gate Access (Source: Authors).

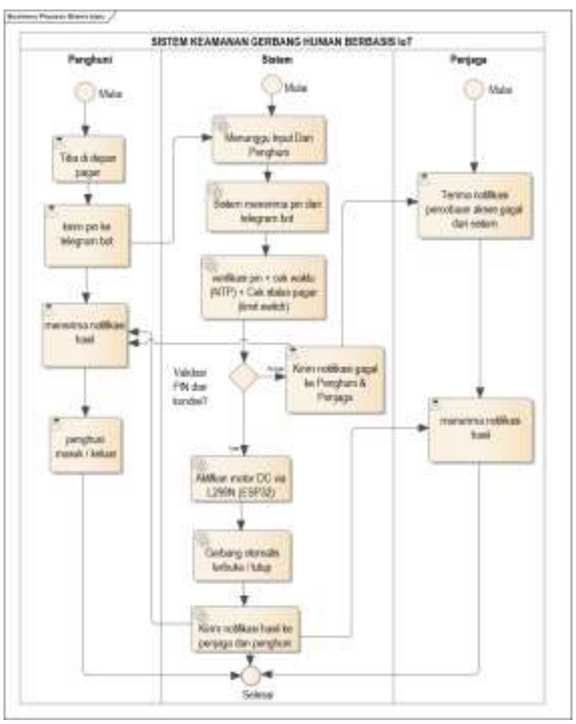

Figure 3. Proposed Business Process of the IoT-Based Digital Gate Security System (Source: Authors).

Compared with the conventional process, the proposed system introduces identity verification through a Telegram Bot, automated gate operation, real-time notifications, and digital access logging, thereby improving both security and operational efficiency.

4.1.2 Product Design

4.1.2.1 System Architecture

The proposed architecture employs the ESP32 as the central controller responsible for coordinating all hardware components and communication services. The ESP32 performs the following tasks: connects to the Internet through a Wi-Fi network; communicates with the Node.js server using the WebSocket protocol; receives authentication commands from the Telegram Bot; controls the DC gearbox motor through the L298N motor driver; and receives position feedback from the limit switches.

Meanwhile, the Node.js server manages communication among the ESP32, the Telegram Bot, the web dashboard, and the MySQL database. Every access event is recorded in real time and displayed through the monitoring dashboard.

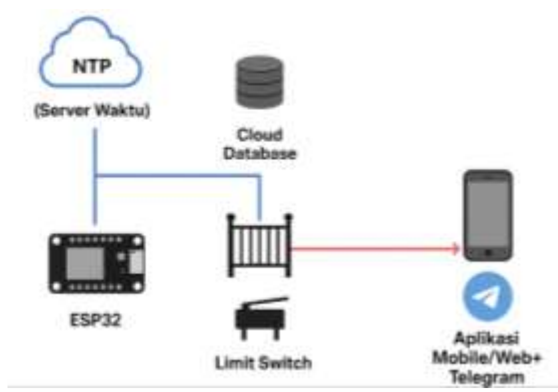

Figure 4. Architecture of the IoT-Based Digital Gate Security System (Source: Authors).

4.1.2.2 Use Case Diagram

The system involves two primary actors:

Resident, can: submit their personal PIN through the Telegram Bot; request the gate to open or close; receive access confirmation notifications; and obtain information regarding gate status.

Boarding-House Caretaker is responsible for: monitoring all gate activities through the web dashboard; reviewing access history; managing registered residents; and granting special access permissions when necessary.

This separation of responsibilities improves both operational management and system security.

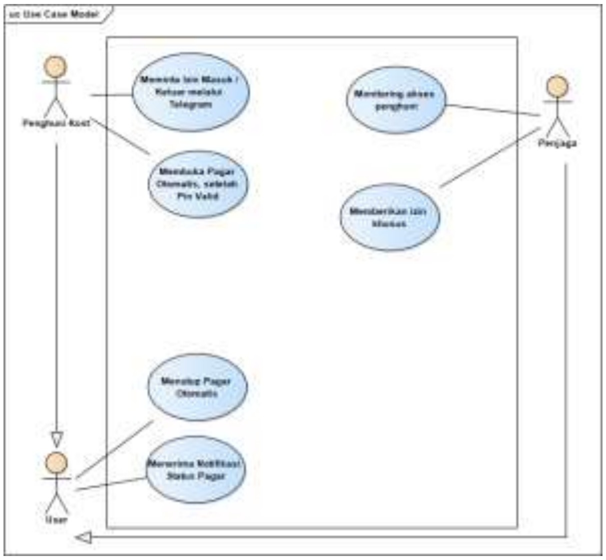

Figure 5. Use Case Diagram of the Digital Gate Security System (Source: Authors).

Table 2. Hardware Components of the Proposed System

No.	Component	Function
1	ESP32 Development Board	Main microcontroller responsible for PIN verification, motor control, and real-time Wi-Fi communication.
2	12V DC Gearbox Motor	Mechanical actuator used to open and close the residential gate automatically.
3	L298N Motor Driver	Interfaces the ESP32 digital control signals with the higher current required by the DC motor while controlling motor direction and speed.
4	Limit Switches (2 units)	Detect the fully opened and fully closed positions of the gate automatically.
5	5V–12V Power Adapter	Supplies electrical power to all hardware components of the system.

4.1.2.3 Internet of Things Implementation

The Internet of Things implementation integrates hardware devices, communication protocols, cloud services, and monitoring applications into a unified platform. Within the proposed system, the ESP32 functions as the primary controller responsible for: processing PIN authentication requests received from the Telegram Bot; controlling the DC gearbox motor via the L298N motor driver; monitoring gate position through limit switches; and transmitting operational data to the Node.js server.

Communication between the ESP32 and the server is established through the WebSocket protocol, enabling low-latency, bidirectional communication. This architecture allows immediate synchronization between hardware devices and the monitoring dashboard.

All access activities—including user authentication, gate opening and closing events, timestamps, and system status—are automatically recorded in the MySQL database. Simultaneously, Telegram notifications are delivered instantly to authorized users whenever the gate status changes.

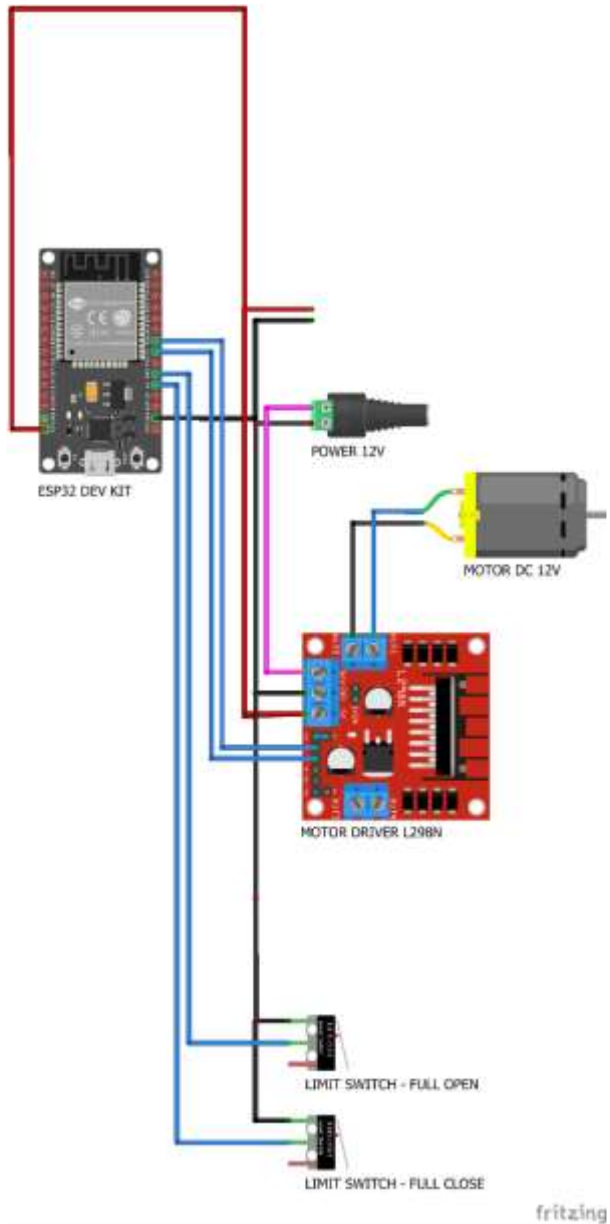

Figure 6. Internet of Things Implementation in the Residential Gate Security System (Source: Authors).

4.1.2.4 Hardware Prototype

A physical prototype was developed using all hardware components specified during the design stage.

The two limit switches were installed at the fully opened and fully closed positions of the gate to prevent over-travel and mechanical damage. Whenever either switch is activated, the ESP32 immediately stops the motor, ensuring safe and reliable gate operation.

The prototype successfully demonstrated stable real-time communication among the ESP32, the Node.js server, the Telegram Bot, and the web-based monitoring dashboard.

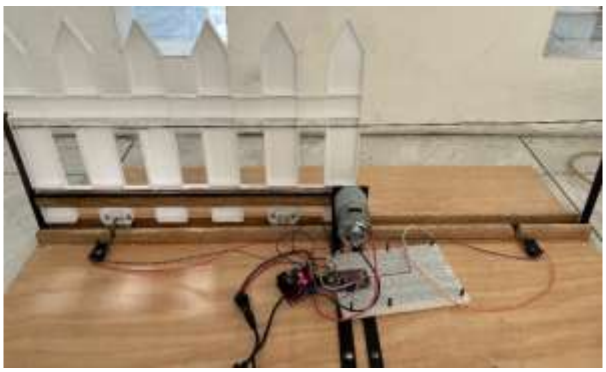

Figure 7. Prototype Mock-up of the Digital Gate Security System (Source: Authors).

Figure 8. Physical Prototype of the IoT-Based Digital Gate Security System (Source: Authors).

4.1.2.5 Website/Application Prototype

A web-based dashboard was developed using Node.js, Express.js, EJS, and Socket.IO to provide real-time system monitoring and management. The dashboard provides the following features: real-time gate status monitoring; access activity logs with Network Time Protocol (NTP)-based timestamps; resident data management by providing centralized monitoring and efficient administrative control.

a) Login Page

Figure 9. Login Interface of the Web-Based Monitoring Dashboard (Source: Authors).

b) Dashboard Interface

Figure 10. Real-Time Monitoring Dashboard (Source: Authors).

4.1.2.6 System Testing Results (Black-Box Testing)

Functional testing was performed using the Black-box Testing method by two expert evaluators. Eight functional scenarios were evaluated using the Guttman scale (Valid = 1; Invalid = 0). Both evaluators concluded that all system functions operated correctly and satisfied the specified functional requirements.

Table 3. Black-Box Testing Results

No.	Tested Function	Result	Description
1	PIN verification via	Valid	Registered PINs are authenticated correctly, while invalid PINs are rejected.

	Telegram Bot		
2	Automatic gate opening via Telegram	Valid	The DC motor opens the gate completely in approximately five seconds.
3	Automatic gate closing via Telegram	Valid	The motor closes the gate successfully, while the limit switch responds in less than 0.5 seconds.
4	Gate position detection (Limit Switch)	Valid	The ESP32 accurately detects OPEN and CLOSED gate positions.
5	Real-time Telegram notifications	Valid	Notifications are delivered immediately whenever the gate status changes.
6	MySQL database logging	Valid	Access activities are automatically stored with accurate NTP timestamps.
7	Real-time web dashboard monitoring	Valid	Administrators can monitor gate status and access history through the web dashboard.
8	Access restriction outside operational hours	Valid	Unauthorized access attempts outside scheduled hours are rejected and appropriate notifications are sent.

Source: Research Data (2025).

The total score obtained was 24 out of 24, resulting in a feasibility percentage of: (24/24) × 100% = 100%. Accordingly, the proposed system was classified as Highly Feasible, with all functional requirements operating as designed.

4.1.2.7 User Evaluation Using the System Usability Scale (SUS)

Usability testing was conducted using the System Usability Scale (SUS). The SUS instrument consists of 10 statements rated on a scale of 1 to 5. The testing involved users who interact directly with the system: boarding house residents and the caretaker. In the testing design, the user respondents consisted of five boarding house residents and one caretaker, bringing the total number of participating users to six.

Table 4. System Usability Scale (SUS) Results

Respondent	SUS Score	Grade	Interpretation
Resident 1	80.0	B	Good
Resident 2	82.5	B	Good
Resident 3	77.5	B	Good
Resident 4	80.0	B	Good
Resident 5	82.5	B	Good
Caretaker	75.0	B	Good
Average	**80.0**	**B**	**Good**

Source: Research Data (2025).

Based on the calculations presented in the report, the system achieved an average SUS score of 80.0, placing it in the Grade B category with a "Good" rating. This indicates that the system's functions—including the use of the Telegram bot and gate status monitoring—are generally understandable and usable for the participants.

However, due to the limited number of respondents, this score does not fully capture the usability perceptions of the user group involved in the testing; consequently, it cannot be generalized to represent all potential users of the system.

4.1.2.8 System Accuracy Testing

System accuracy was evaluated by conducting ten trials for each primary functional parameter.

Table 5. System Accuracy Test Results

No.	Test Scenario	Trials	Successful Trials	Accuracy
1	Valid PIN authentication	10	10	100%
2	Invalid PIN rejection	10	10	100%
3	Automatic gate opening	10	10	100%
4	Automatic gate closing	10	10	100%
5	Gate position detection using limit switches	10	10	100%
6	Telegram notification delivery	10	10	100%

7	MySQL database logging	10	10	100%

Source: Research Data (2025).

All seven functional parameters achieved 100% accuracy across ten experimental trials, indicating that the proposed system consistently performed without functional failures. Based on the predefined evaluation criteria, the system is classified as Highly Accurate (Grade A).

4.2 Discussion

Overall, the research findings indicate that the IoT-based digital residential gate security system was successfully implemented and performs its primary functions as designed. Expert evaluation yielded a 100% feasibility rating, while accuracy testing demonstrated full operational compliance across seven test conditions. The geared DC motor demonstrated stable performance, while the limit switches responded effectively, ensuring accurate positioning and preventing mechanical overrun. Compared to prior research using the ESP8266 without identity verification[22], this proposed system offers enhanced security through PIN-based authentication and automated activity logging within a MySQL database. Similarly, compared to a previously developed system that relied on the commercial Blynk platform[11][23], this proposed solution utilizes the Telegram Bot API, providing a cost-effective, standalone communication platform free of subscription fees. Regarding usability, a System Usability Scale (SUS) score of 80 indicates a positive user perception; however, interpreting these results requires acknowledging the limited number of respondents. Consequently, this study is best characterized as an initial implementation and evaluation of system integration rather than a large-scale system validation. The study's practical contribution lies in the system's ability to link access verification (via PIN), gate control, status notifications, and activity logging into a single, integrated workflow. This integration lays the groundwork for a more digitized gate security system compared to manual access management mechanisms. Future research should involve a larger user base, varied network conditions, and observation over an extended period of use. Such steps would broaden user testing and allow for quantitative network performance evaluation, thereby providing more robust evidence regarding the system's usability, reliability, and performance.

5 CONCLUSION

Based on the results of the system development and experimental evaluation, the following conclusions can be drawn:

a. The resident identity verification system using PIN-based authentication through the Telegram Bot was successfully developed and achieved 100% functional validity according to the Black-box Testing results. The system accurately verifies registered users, rejects unauthorized access attempts, and provides reliable identity authentication.
b. The automatic gate control mechanism demonstrated stable performance, completing each opening and closing cycle in approximately five seconds, while the limit switches responded in less than 0.5 seconds. Furthermore, the system achieved 100% accuracy across seven primary functional parameters, classifying it as a Highly Accurate (Grade A) system.
c. The implementation of real-time access logging using a MySQL database, web-based monitoring dashboard, and Telegram notification service was successfully accomplished. The average System Usability Scale (SUS) score of 80 (Grade B – Good) indicates that the proposed system is practical, user-friendly, responsive, and suitable for deployment in real residential environments.
d. However, the research was limited to a prototype scale and did not measure communication latency or network reliability in short- or long-term usage scenarios. The limited number of respondents in the SUS testing also means the findings cannot be widely generalized. Therefore, future research should conduct testing with a larger number of users and measure latency and network reliability under various usage conditions.

REFERENCES

[1] BPS, *Laporan statistik kriminal*, vol. 15. Subdit. Statistik Politik dan Keamanan, Badan Pusat Statistik, 2004.

[2] Tri Rachmadi, *Mengenal Apa itu Internet of Things*. Bandar Lampung: TIGA Ebook, 2020.

[3] Yudho Yudhanto and Abdul Azis, *Pengantar Teknologi Internet of Things (IoT)*, 1st ed. Surakarta: UNS Press, 2019.

[4] M. Gupta, S. Bhatt, A. H. Alshehri, and R. Sandhu, *Access Control Models and Architectures For IoT and Cyber Physical Systems*. Cham: Springer International Publishing, 2022. doi: 10.1007/978-3-030-81089-4.

[5] R. Trishardian, A. Fadli, M. S. Aliim, R. Supriyanti, and Y. Ramadhani, "Aplikasi Bot Telegram Pada Sistem Presensi dan Pengukuran Suhu Tubuh Berbasis IoT," *Jurnal Teknik Elektro dan Komputer TRIAC*, vol. 9, no. 3, pp.

113–121, Nov. 2022, doi: 10.21107/triac.v9i3.15752.

[6] I. R. Muttaqin and D. B. Santoso, “Prototype Pagar Otomatis Berbasis Arduino Uno Dengan Sensor Ultrasonic Hc-SR04,” *JE-Unisla*, vol. 6, no. 2, p. 41, Sep. 2021, doi: 10.30736/je-unisla.v6i2.695.

[7] D. T. Laksono, “IMPLEMENTASI KENDALI BERURUTAN OTOMATIS BERBASIS LIMIT SWITCH PADA MODUL PRAKTIKUM PLC OMRON CP1E UNTUK PEMBELAJARAN OTOMASI INDUSTRI,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 14, no. 1, Jan. 2026, doi: 10.23960/jitet.v14i1.8326.

[8] A. Pingkan, I. Marasti, and B. Nixon, “Rancang Bangun Sistem Miniatur Smart Garden dengan Aplikasi Android Berbasis Internet of Things (IoT),” *Spektral*, vol. 4, no. 1, pp. 173–178, Apr. 2023, doi: 10.32722/spektral.v4i1.5645.

[9] S. Alam, F. Fauzi, G. Tjahjadi, and R. S. Sya’ban, “Rancang Bangun Sistem Kendali Pintu Pagar Otomatis Berbasis Pengolahan Citra Digital Pelat Nomor Kendaraan Menggunakan Metode Optical Character Recognition (OCR),” *Faktor Exacta*, vol. 15, no. 2, Aug. 2022, doi: 10.30998/faktorexacta.v15i2.12922.

[10] David Fadlianda, Muhammad Fikry, and Nunsina, “Innovative IoT-Based Automatic Gate System with RFID and Electro-Magnetic Lock for Secure Access,” *Proceedings of Malikussaleh International Conference on Multidisciplinary Studies (MICoMS)*, vol. 4, p. 00005, Dec. 2024, doi: 10.29103/micoms.v4i.884.

[11] C. Widiasari, P. Abram Sianipar, and M. Diono, “Sistem Kontrol Otomatis Pagar Rumah Berbasis Internet of Things (IoT),” *Jurnal Elektro dan Mesin Terapan*, vol. 8, no. 2, pp. 162–174, Nov. 2022, doi: 10.35143/elementer.v8i2.5748.

[12] A. B. B. Wijanto and Y. M. Dinata, “Rancang Bangun Sistem Pintu Gerbang Otomatis Berbasis Internet of Things dengan Aplikasi Android,” *Jurnal Informatika dan Sistem Informasi*, vol. 10, no. 1, pp. 19–30, May 2024, doi: 10.37715/juisi.v10i1.4708.

[13] Benedict & Marta, “Rancang Bangun Sistem Pintu Gerbang Otomatis Berbasis Internet of Things dengan Aplikasi Android,” *Jurnal UC (JUISI)*, vol. 10, 2024.

[14] I. P. Dewi and R. Fikri, “Optimalisasi Keamanan Rumah dengan Implementasi Sistem Notifikasi Gerbang Cerdas Berbasis Internet of Things (IoT),” *Journal of Computer System and Informatics (JoSYC)*, vol. 4, no. 4, pp. 816–829, Aug. 2023, doi: 10.47065/josyc.v4i4.4004.

[15] Sugiyono, *Metode penelitian kuantitatif, kualitatif, dan R&D* , 3rd ed. Bandung: Alfabeta, 2021.

[16] ST. , MT. Rahmat Fauzi Siregar and ST. , MT. Affandi, *Sistem Mikrokontroler I*. Medan: UMSU Publisher, 2024.

[17] M. Z. U. Rahman *et al.*, “Design, Analysis, and Control of Biomedical Healthcare Modular Wheelchair with Posture Transformation,” *Complexity*, vol. 2023, pp. 1–13, Jun. 2023, doi: 10.1155/2023/7310265.

[18] P. Bhandari and P. Z. Csurcsia, “Digital implementation of the PID controller,” *Software Impacts*, vol. 13, p. 100306, Aug. 2022, doi: 10.1016/j.simpa.2022.100306.

[19] R. Hasan, F. A. Ashan, F. A. Antara, and J. Islam, “Growgreen: A Machine Learning-Based Crop Recommendation and Whatsapp-Controlled Smart IoT Greenhouse,” in *2026 IEEE 2nd International Conference on Quantum Photonics, Artificial Intelligence & Networking (QPAIN)*, IEEE, Apr. 2026, pp. 1–6. doi: 10.1109/QPAIN69676.2026.11545917.

[20] J. Brooke, “Sus: a “quick and dirty’usability,” *Usability evaluation in industry*, vol. 189, no. 3, pp. 189–194, 1996.

[21] I. Kononenko and M. Kukar, “Machine Learning Basics,” *Machine Learning and Data Mining*, pp. 59–105, 2007, doi: 10.1533/9780857099440.59.

[22] W. Raditya, A. Surahman, A. Budiawan, F. Amanda, N. Dwi Putri, and S. Yudha, “PENERAPAN SISTEM KEAMANAN GERBANG RUMAH BERBASIS TELEGRAM MENGGUNAKAN ESP8266,” *Jurnal Teknik dan Sistem Komputer (JTIKOM)*, vol. 3, no. 2, p. 2022.

[23] F. P. Syahrani, H. K. Saputra, S. Anori, W. Agustiarmi, F. T. Ayasrah, and P. Van Thanh, “IoT-Enabled Smart Fence: Remote Security and Monitoring Using NodeMCU ESP32 and Blynk,” *Journal of Hypermedia & Technology-Enhanced Learning*, vol. 3, no. 1, pp. 1–15, Jan. 2025, doi: 10.58536/j-hytel.158.