IoT-Based Package Drop Box System Using Arduino Uno and ESP32-CAM

Frengki Simatupang
Faculty of Vocational
Institut Teknologi Del
Laguboti, Indonesia
frengki.simatupang@del.ac.id

Marojahan M.T Sigiro
Faculty of Vocational
Institut Teknologi Del
Laguboti, Indonesia
marojahan@del.ac.id

Eka Stephani Sinambela
Faculty of Vocational
Institut Teknologi Del
Laguboti, Indonesia
eka@del.ac.id

Istas Pratomo Manalu
Faculty of Vocational
Institut Teknologi Del
Laguboti, Indonesia
istas@del.ac.id

Abstract—The rapid growth of e-commerce activity in Indonesia has led to a surge in package delivery volumes, resulting in increased workloads for couriers and a higher risk of delivery failures when recipients are not present at the delivery location. This issue demands an innovative solution that can address logistical challenges in an automated, secure, and efficient manner. This study aims to design and implement an IoT-Based Package Drop Box system that integrates Arduino Uno, Wemos D1 Mini, ESP32-CAM, and the GM66 barcode scanner, along with a web interface and WhatsApp notification service. The methodology follows a prototype engineering approach consisting of need analysis, system design, implementation, and hardware-software testing phases. The test results demonstrate that the system can successfully verify package tracking numbers via barcode scanning at an optimal distance of 10–19 cm, automatically unlock the box, capture images inside the drop box using ESP32-CAM, and send real-time delivery notifications to users via WhatsApp. The system is also capable of storing data locally when the internet connection is lost and synchronizing it once the connection is restored. The findings conclude that the integrated system provides a practical and reliable solution to common delivery issues and has the potential to be further developed for smart home environments or broader Internet of Things-based logistics systems.

Keywords— Arduino Uno, ESP32-CAM, Internet of Things, Package Drop Box, WhatsApp Notification.

I. INTRODUCTION

The rapid growth of e-commerce in Indonesia, driven by advances in digital platforms, mobile applications, and electronic payment systems, has significantly transformed consumer purchasing behavior and logistics services [1][2][3][4]. As online shopping becomes increasingly prevalent, parcel delivery volumes have risen sharply, placing greater operational demands on courier and logistics providers [5]. Couriers are now required to handle higher workloads while meeting expectations for fast and reliable delivery[6].

A common operational challenge arises when couriers arrive at delivery locations but recipients are unavailable [7]. In such cases, packages are often rescheduled, left with neighbors, or placed in unsecured locations such as porches or gates, increasing the risk of loss or damage [8]. These practices negatively affect both consumer satisfaction and courier efficiency, resulting in delivery delays, repeated visits, and customer complaints. Consequently, there is a need for an automated delivery solution that enables secure package reception without requiring direct interaction between couriers and recipients [9][10][11][12].

Internet of Things (IoT) technology offers a promising approach to addressing these challenges through automated, remotely monitored delivery systems [13][14][15]. One widely explored solution is the Smart Drop Box, an intelligent parcel storage unit designed to receive packages securely and autonomously [16]. Existing implementations typically employ microcontrollers such as Arduino, ESP32, or Raspberry Pi [17][18], along with cameras and notification services, to support remote monitoring and access control . However, many current systems rely heavily on continuous internet connectivity and lack robust mechanisms for delivery verification and data reliability [19].

Several related studies have proposed smart parcel reception systems with varying capabilities. Azrin et al. [20] developed a Raspberry Pi–based smart box with camera monitoring and Telegram notifications but did not include receipt validation. Faza et al. [21] implemented an ESP32-CAM–based system with web monitoring but depended entirely on internet connectivity without offline data storage. Junaidi et al. [22] integrated weight sensors and cameras but lacked an interactive user interface, while Valupi et al. [23] proposed an automatic compartment system without comprehensive barcode validation or integrated web services. Saputra & Zainul Abidin [24] designed an IoT-based automatic parcel reception system using an ESP8266, barcode scanner, and mobile app notifications to verify and secure incoming packages, demonstrating real-time verification and remote control functions. Febriyanti et al. [25] also implemented an IoT-based parcel reception box controlled via Telegram, emphasizing remote control and notification features in delivery handling.

These limitations highlight gaps in secure package verification and system robustness under unstable network conditions. Table 1 presents a comparison between previous research and the current study, focusing specifically on the Smart Drop Box Packet system.

Table 1. Comparison Between This Research and Previous Studies

Researcher and Year	Microcontroller Type	Receipt Verification Feature	Monitoring Camera	Notification Medium	Offline Storage	Interactive Website
Azrin et al. (2022)	Raspberry Pi + Arduino	Not Available	Yes (Camera)	Telegram	Not Available	Not Provided
Faza et al. (2023)	ESP-32 CAM	Not Available	Yes	Website & Telegram	Not Available	Package Monitoring Only
Mee et al. (2023)	ESP-32 CAM	Not Available	Yes	Telegram	Not Available	Not Available
Junaidi et al. (2024)	Arduino + Raspberry Pi	Not Available	Yes	Website	Not Available	Yes
Valupi et al. (2025)	ESP32-Wroom	Not Available	Yes	Telegram	Not Available	Not Available
This Study (2025)	Arduino Uno, Wemos D1R1 + ESP32-CAM	Barcode Scanner GM66	Yes (Live Monitoring)	Website and WhatsApp	Yes (Local MySQL)	Package Registration, Status Tracking, and Package Monitoring

This study addresses these gaps by designing and implementing an IoT-Based Package Drop Box System using Arduino Uno and ESP32-CAM, integrated with a web-based platform and WhatsApp notification service. Users can register package tracking numbers in advance, monitor delivery status, and view real-time images of the drop box interior. Couriers verify deliveries by scanning package barcodes using a GM66 barcode scanner, ensuring that only registered and authorized packages are accepted.

The primary novelty of this work lies in a two-way barcode-based verification mechanism combined with offline data synchronization. The system cross-verifies scanned barcodes against pre-registered tracking numbers stored in the database, preventing unauthorized package insertion. In addition, the system incorporates a local MySQL-based data redundancy mechanism, allowing all transaction data to be stored locally during internet outages and automatically synchronized once connectivity is restored. This feature significantly enhances system reliability compared to existing smart drop box solutions that depend solely on continuous internet access.

By integrating barcode-based authentication, real-time visual monitoring, WhatsApp notifications, and offline data resilience into a unified system, this research provides a more secure and robust solution for automated package reception. The modular system architecture also supports future expansion and integration into smart home and IoT-based logistics ecosystems, promoting the practical and sustainable adoption of IoT technology in everyday delivery services.

II. RESEARCH METHODOLOGY

A. Research Method

This research adopts a prototype engineering approach to design and implement an IoT-based Smart Package Drop Box System. The system is designed to enable automatic package reception through the integration of Arduino Uno, WeMOS D1 R1, and ESP32-CAM microcontrollers.

The prototype engineering method combines both hardware and software components through an iterative development cycle, allowing continuous refinement based on real-world testing and evaluation. This study is categorized as applied experimental research, in which the developed system is tested under conditions that simulate actual delivery scenarios, including barcode scanning, automatic locking, live camera monitoring, and web- and WhatsApp-based notifications.

B. Research Procedure

The research procedure was systematically structured into several stages, as illustrated in Figure 1, including planning, requirements analysis, system design, system implementation, and system testing. This structured approach ensures that system development aligns with the intended functional objectives and performance criteria.

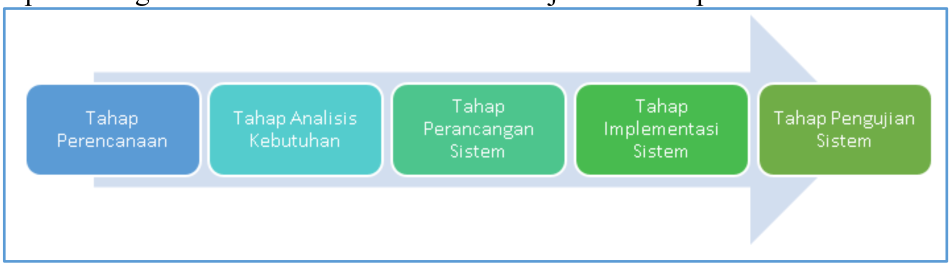

Figure 1. Research Implementation Stages

The initial stage involves planning, which defines the system requirements and scope. The researcher identified key challenges in conventional parcel delivery processes, such as the absence of recipients at home, package security issues, and limited control over delivery verification. A literature review was also conducted to analyze previous studies related to smart drop boxes and IoT-based delivery systems.

Following the planning phase, a requirements analysis was conducted for both hardware and software components. The results of this analysis established that the system requires: Arduino Uno for core system logic processing, WeMOS D1 R1 for Wi-Fi and server connectivity [22], GM66 barcode scanner as the receipt number reader, ESP32-CAM as the internal monitoring camera, and additional actuators such as a relay module, solenoid door lock, and LCD display. On the software side, the system includes a web-based user interface, a MySQL database for data storage, and WhatsApp API integration for automated notifications.

The system design stage covers both hardware and software architecture. A flowchart was developed to illustrate the operational workflow—from package registration and courier barcode scanning to the generation of notifications. Microcontroller programming was carried out using the Arduino IDE, while the website was developed using PHP and MySQL. The next phase was system implementation and assembly. All components were integrated into a prototype drop box, constructed from wood with dimensions of height = 80 cm and width = 60 cm. The Smart Drop Box Packet was designed to accommodate packages up to 30 cm wide and 50 cm high. After full assembly, preliminary testing was performed to ensure all modules operated according to the defined workflow.

The final stage, system testing, was conducted under several scenarios, including: Registration and verification of tracking numbers, Barcode scanning (for both registered and unregistered codes), Validation of receipt numbers, Automatic locking and unlocking functions, Camera data retrieval via ESP32-CAM, and notification delivery via WhatsApp. Additional testing was carried out under unstable internet conditions to evaluate the system's local data storage capability using MySQL and its automatic synchronization mechanism once connectivity was restored.

All these procedures aim to ensure that the system operates automatically, securely, and remotely, fulfilling user needs while addressing logistical challenges in modern digital parcel delivery systems.

C. System Architecture and Hardware Design

The proposed IoT-Based Package Drop Box System is designed using a modular hardware architecture to ensure flexibility, reliability, and ease of maintenance. The system integrates multiple microcontrollers and peripheral devices, each responsible for specific operational tasks. Arduino Uno functions as the main controller responsible for barcode verification, relay activation, and lock control. The ESP32-CAM module is used for real-time image capture inside the drop box, while the WeMOS D1 R1 handles Wi-Fi connectivity and communication between hardware modules and the web server. A GM66 barcode scanner is employed to verify package tracking numbers, ensuring that only authorized parcels are accepted by the system. Table 2 summarizes the hardware components and their respective functions, while Table 3 presents the hardware connection configuration to reduce implementation verbosity.

Table 2. Hardware Components Description

No	Component	Function
1	Arduino Uno	Barcode processing and lock control
2	ESP32-CAM	Real-time image capture
3	WeMOS D1 R1	Wi-Fi communication
4	GM66 Barcode Scanner	Tracking number verification
5	Relay Module	Solenoid lock activation
6	Solenoid Lock	Physical locking mechanism
7	LCD Display	System status information

Table 3. Summary of Hardware Connections

No	Component	Interface	Connected To	Function
1	Arduino Uno	UART	GM66 Scanner	Barcode data reading
2	Arduino Uno	Digital I/O	Relay Module	Lock control
3	Arduino Uno	I2C	LCD Display	Status output
4	WeMOS D1 R1	UART	ESP32-CAM	Camera communication
5	Power Supply	12V–5V	All Modules	System power

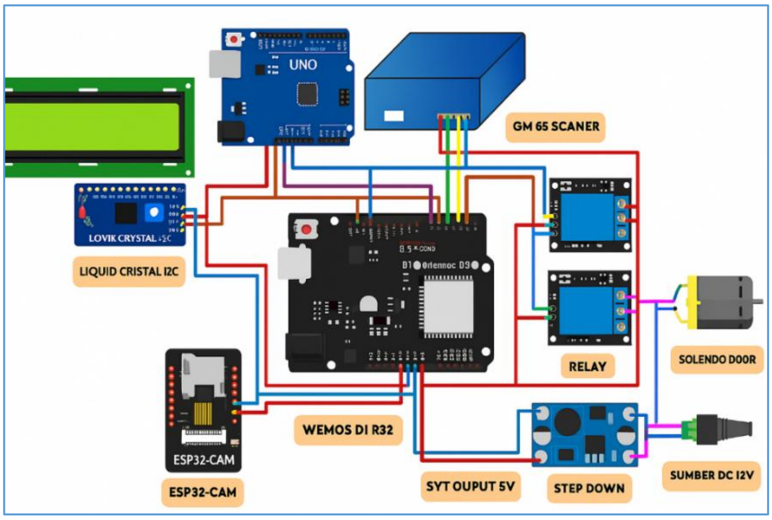

Figure 2. Schematic of the Smart Drop Box Package Circuit

The schematic design ensures efficient interconnection among all components, enabling the Smart Drop Box system to perform core functionalities such as package verification, door lock control, live monitoring, and system notifications.

D. Software Design and Implementation

The software architecture of the system follows a layered design consisting of firmware-level control, web-based management, and notification services. The system firmware, developed using the Arduino IDE, manages barcode scanning, solenoid lock operation, and camera triggering based on verification results. A web-based application developed using PHP and MySQL allows users to register package tracking numbers, monitor delivery status, and access real-time camera feeds. WhatsApp notifications are integrated to provide instant alerts upon successful package delivery. To enhance system reliability, an offline data redundancy mechanism is implemented. During internet connectivity loss, transaction data are stored locally in a MySQL database and automatically synchronized with the central server once connectivity is restored. Table 4. summarizes the software modules used in the system.

Table 4. Software Module Summary

No	Module	Platform	Description
1	Firmware Control	Arduino IDE	Barcode verification and lock control
2	Camera Module	ESP32-CAM	Image capture and streaming
3	Web Application	PHP	Tracking registration and monitoring
4	Database System	MySQL	Data storage and synchronization
5	Notification Service	WhatsApp API	Real-time user alerts
6	Offline Sync Module	Local MySQL	Temporary data storage

The system implementation consists of two main components: hardware implementation, which involves assembling and configuring all electronic and mechanical elements of the Smart Package Dropbox prototype, and software implementation, which includes microcontroller programming, web-based system development using PHP and MySQL, and integration of WhatsApp API for user notifications.

The software implementation of the IoT-Based Package Dropbox System, which integrates microcontroller control, web-based management, and WhatsApp notification services to enable automated and real-time operation. The system firmware, developed using the Arduino IDE, handles barcode scanning, tracking number verification, relay-based door control, and image capture via the ESP32-CAM. A web application built with PHP and MySQL allows users to register tracking numbers, monitor delivery status, and access captured images, with data synchronized through HTTP communication. In addition, WhatsApp API integration provides automatic delivery notifications directly to users. To ensure reliability, the system supports local data storage during internet outages and automatically synchronizes data once connectivity is restored.

Figure 7 depicts the hardware implementation of the IoT-Based Package Drop Box System, where all electronic components are integrated into a wooden storage box to enable automated package reception and monitoring. The system employs an Arduino Uno as the main controller, a WeMOS D1 R1 for Wi-Fi communication, an ESP32-CAM for internal image capture, and a GM66 barcode scanner for tracking number verification. Additional components include a relay-controlled solenoid lock for automated access control and an I2C LCD for status display. The hardware is organized into separate storage and control compartments to ensure reliable operation. Upon successful barcode verification, the door unlocks automatically, an image is captured, and a WhatsApp notification with the captured image is sent to the user, demonstrating a fully autonomous and remotely monitored delivery system.

Figure 3. Implementation of Drop Box Packet System

III. RESULTS AND DISCUSSION

A. Tracking Number Registration Test

The tracking number registration test was conducted using both pre-generated barcode labels and actual shipping receipts from existing packages. Table 5 presents the dataset of registered tracking numbers for packages that have been ordered and are scheduled for delivery by couriers.

Table 5. Registered Package Tracking Numbers

No	Receipt Number	Courier	Sender Contact	Recipient Name	Recipient Contact	Shipping Date
1	LP1884HB439	SiCepat	+62822875 53621	Jusepril	+62878109 21776	10-08- 2024
2	JDGE7493902	JNT	+62822550 86846	Jusepril	+62878109 21776	07-08- 2024

3	JD0214RJOO44	JNE	+62852627 18940	Jusepril	+62878109 21776	10-05- 2024
4	GW839400 572	POS	+62896202 99076	Jusepril	+62878109 21776	28-02- 2024
5	SKWQ752 089	SPX	+62878602 08606	Jusepril	+62878109 21776	03-08- 2024

Table 6 demonstrates that the tracking number registration system on the website functions correctly. New tracking numbers with valid formats are successfully stored in the system, while previously registered tracking numbers are properly detected and rejected to prevent data duplication. In addition, the system is able to identify and reject tracking numbers with invalid formats. These results confirm that the validation and duplicate detection mechanisms operate as intended.

Table 6. Results of Registration Testing of Receipt Number on The Website

No	Receipt Barcode	System Status	Status	Description
1	LP1884HB439	Success	Success	Save the receipt number data to the system
2	JD0214RJOO44	Success	Success	Save the receipt number data to the system
3	GW839400572	Success	Success	Save the receipt number data to the system
4	SKWQ752089	Success	Success	Save the receipt number data to the system
5	JDGE7493902	Success	Success	Save the receipt number data to the system
6	JDGE7493902	Error	Success	Receipt Number data already exists in the system
7	SKWQ752089	Error	Success	Receipt Number data already exists in the system
8	GW839400572	Error	Success	Receipt Number data already exists in the system
9	JD0214RJOO44	Error	Success	Receipt Number data already exists in the system
10	LP1884HB439	Error	Success	Receipt Number data already exists in the system
11	1212112121212	Error	Success	Data is not a combination of receipt numbers
12	AVASSWCDS	Error	Success	Data is not a combination of receipt numbers
13	198023948123	Error	Success	Data is not a combination of receipt numbers

B. Package Barcode Scanning Test Results

The barcode scanning test was conducted to evaluate the ability of the Smart Dropbox system to recognize registered tracking numbers and generate appropriate responses. In this test, the courier scanned the barcode associated with tracking number LP1884HB439, after which the system automatically verified the data against the database. When a valid and registered tracking number was detected, the system activated the solenoid to unlock the drop box and displayed the messages “Tracking Number Registered” and “Insert Package” on the LCD. This result confirms that the system performs real-time verification and provides automated mechanical and visual feedback.

Table 7 summarizes the scanning test results using ten tracking numbers, consisting of five registered and five unregistered entries. The system correctly identified the registered tracking number LP1884HB439. For other inputs labeled as registered but not found in the database, the system appropriately displayed a “not registered” notification, indicating correct validation behavior. Similarly, all unregistered tracking numbers were accurately detected and rejected. Overall, the test results demonstrate that the Smart Dropbox system reliably distinguishes between registered and unregistered tracking numbers, ensuring consistent and accurate verification performance.

Table 7. Test Results for Registered and Unregistered Package Tracking Numbers

No	Test Scenario	Input Tracking Number	System Response	**Status**
1	Scan of registered tracking number	LP1884HB439	Displays message: “Tracking Number ‘LP1884HB439’ Registered”	Successful
2	Scan of registered tracking number	JD0214RJOO44	Displays message: “Tracking Number ‘JD0214RJOO44’ Not Registered”	Successful

3	Scan of registered tracking number	GW839400572	Displays message: "Tracking Number 'GW839400572' Not Registered"	Successful
4	Scan of registered tracking number	SKWQ752089	Displays message: "Tracking Number 'SKWQ752089' Not Registered"	Successful
5	Scan of registered tracking number	JDGE7493902	Displays message: "Tracking Number 'JDGE7493902' Not Registered"	Successful
6	Scan of unregistered tracking number	ZX0012TEST99	Displays message: "Tracking Number 'ZX0012TEST99' Not Registered"	Successful
7	Scan of unregistered tracking number	TRX2024ABC01	Displays message: "Tracking Number 'TRX2024ABC01' Not Registered"	Successful
8	Scan of unregistered tracking number	PKG9912XZL88	Displays message: "Tracking Number 'PKG9912XZL88' Not Registered"	Successful
9	Scan of unregistered tracking number	SPX01TEST987	Displays message: "Tracking Number 'SPX01TEST987' Not Registered"	Successful
10	Scan of unregistered tracking number	JNT2244MNBV3	Displays message: "Tracking Number 'JNT2244MNBV3' Not Registered"	Successful

C. Results of Barcode Scanning Distance Test

The barcode scanning distance test was conducted to evaluate the effective detection range of the GM66 barcode sensor in reading tracking numbers accurately. The distance parameter plays a crucial role, as it directly influences the accuracy and responsiveness of barcode detection. Based on the experimental results, it was observed that the sensor failed to read barcodes properly when placed too close to the scanner, particularly at distances below 4 cm. This finding aligns with the manufacturer's technical documentation, which specifies a minimum reading distance of approximately 2.5 cm for the GM66 sensor. At distances shorter than this threshold, focus distortion and misreading tend to occur [23].

Furthermore, the test results indicate that the optimal reading range for the GM66 sensor lies between 10–19 cm, where the barcode can be detected with high accuracy and stable response time. However, when the barcode was positioned at distances between 20–28 cm, the sensor could no longer read the barcode effectively. This suggests that the maximum effective range of the GM66 scanner is around 25 cm, consistent with the manufacturer's specifications. To ensure stable and reliable system performance, the implemented Smart Dropbox design includes a distance marker that guides users to place packages within the optimal scanning range (10–19 cm) from the sensor. The detailed results of the distance-based scanning tests are presented in Table 8 below.

Table 8. Test Results of Barcode Scanning Distance for Package Tracking Numbers

No	Distance	Detection Status	Description
1	0 – 3 cm	Not Detected	Too close and out of focus
2	4 cm	Partially Detected	Blurry and inconsistent results; affected by lighting
3	5 cm	Partially Detected	Blurry and inconsistent results; affected by lighting
4	7 cm	Partially Detected	Blurry and inconsistent results; affected by lighting
5	8 cm	Partially Detected	Blurry and inconsistent results; affected by lighting
6	9 cm	Partially Detected	Blurry and inconsistent results; affected by lighting
7	10 cm	Clearly Detected	Optimal and stable reading
8	12 cm	Clearly Detected	Optimal and stable reading
9	15 cm	Clearly Detected	Optimal and stable reading
10	17 cm	Clearly Detected	Optimal and stable reading
11	19 cm	Clearly Detected	Optimal and stable reading
12	20 cm	Partially Detected	Blurry and inconsistent results; affected by lighting
13	23 cm	Partially Detected	Blurry and inconsistent results; affected by lighting
14	25 cm	Partially Detected	Blurry and inconsistent results; affected by lighting
15	>25 cm	Not Detected	Beyond maximum sensor range

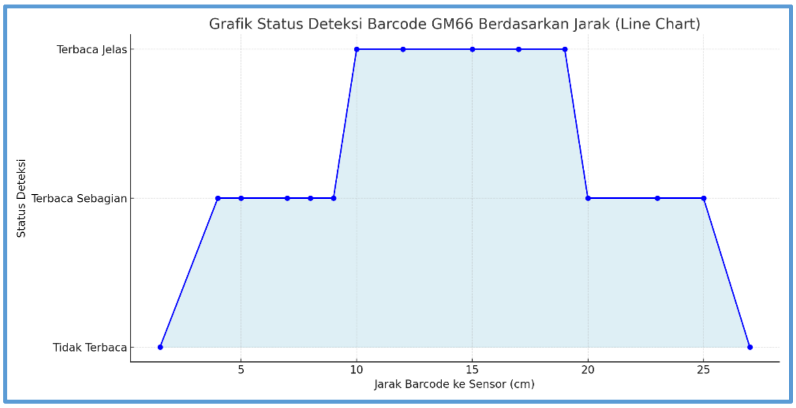

Figure 4. Line Chart graph of the relationship between barcode distance by GM66 sensor and detection status

Figure 4 presents a line chart illustrating the relationship between the distance of the barcode from the GM66 sensor and its detection status. The results indicate three distinct detection zones: Not Detected (0–3 cm and >25 cm), which occurs when the barcode is positioned too close or too far from the sensor; Partially Detected (4–9 cm and 20–25 cm), where detection is inconsistent and highly dependent on lighting conditions; and Clearly Detected (10–19 cm), representing the optimal and stable scanning range. This chart confirms that the 10–19 cm distance range provides the best performance for barcode detection using the GM66 sensor, making it the most suitable range for implementation in the Smart Dropbox system.

D. Camera Monitoring Performance

The camera monitoring performance was evaluated to assess the reliability of the ESP32-CAM module in capturing package images after a successful delivery event. As shown in Table 9, the image capture process was tested over 50 repetitions, all of which resulted in successful image acquisition, yielding a success rate of 100%. In addition, the average image capture delay was measured at 1.2 seconds following the system trigger. These results demonstrate that the ESP32-CAM module operates in a stable and responsive manner, making it suitable for real-time visual monitoring and delivery verification within the Smart Dropbox system.

Table 9.Camera Capture Performance

No	Metric	Result
1	Test repetitions	50
2	Successful image captures	50
3	Success rate	100%
4	Average capture delay	1.2 s

E. WhatsApp Notification Latency

Notification performance was evaluated by measuring the delay between package insertion and message delivery to the user’s WhatsApp account.

Table 10. Notification Delivery Performance

No	Metric	Result
1	Test repetitions	50 x
2	Average notification delay	1.4 s
3	Minimum delay	1.1 s
4	Maximum delay	2.0 s
5	Success rate	100%

The WhatsApp notification latency was evaluated by measuring the time delay between package insertion into the Smart Dropbox and message delivery to the user's registered WhatsApp account. As summarized in Table 10, the notification delivery test was conducted over 10 repetitions, all of which were completed successfully, resulting in a success rate of 100%. The results indicate an average notification delay of 1.4 seconds, with the minimum and maximum delays recorded at 1.1 seconds and 2.0 seconds, respectively. These findings demonstrate that the notification system provides timely and reliable real-time feedback to users. By delivering instant notifications without requiring manual access to the web interface, the system enhances usability and ensures prompt awareness of package delivery events.

F. Performance Evaluation Results

The results indicate that the system successfully performs barcode-based verification and secure package reception with consistent accuracy. The offline synchronization mechanism effectively preserved transaction data during connectivity loss, ensuring system robustness and operational continuity.

Table 11. Performance Evaluation Results

Test Type	Sample Size	Repetitions	Success Rate (%)
Tracking Registration	50 codes	10x	100
Barcode Verification	50 packages	10x	100
Camera Capture	50 events	10x	100
WhatsApp Notification	50 events	10x	100
Offline Data Sync	50 events	10x	100

G. Offline Data Synchronization Result

Offline data synchronization testing was conducted to evaluate system reliability under unstable network conditions. In this test, internet connectivity was intentionally disconnected during package delivery operations to simulate real-world network disruptions. The objective was to verify whether transaction data could be stored locally and synchronized correctly once the network connection was restored. During the offline period, all delivery-related data—including tracking numbers, timestamps, barcode verification results, and image capture references—were stored in a local MySQL database. After internet connectivity was re-enabled, the system automatically synchronized the locally stored data with the central server without requiring manual intervention.

Table 12. Offline Data Synchronization Test Results

No	Parameter	Value
1	Total offline delivery events	50
2	Data successfully stored locally	50
3	Data loss during offline mode	0
4	Successful data synchronization	50
5	Synchronization success rate	100%

H. Failure Case Analysis and System Limitations

Although the system performed reliably overall, several failure cases were observed: Barcode detection failures occurred when scanning distances were below 10 cm or above 20 cm, as well as under low-light conditions. These issues are primarily related to optical focus limitations and environmental lighting. Notification delays increased during unstable internet connections, reaching up to 2.0 seconds. However, this latency did not affect data integrity due to the local data storage mechanism. Insufficient lighting affected both barcode readability and image clarity from the ESP32-CAM. Future improvements may include additional LED illumination and enhanced camera calibration. Despite these limitations, the system demonstrated strong operational robustness and maintained functional integrity across all test scenarios.

I. Comparative Analysis with Previous Studies

Based on the comparative analysis in Table 1, most previous studies focused primarily on basic package storage and notification functions. Azrin et al. developed a Raspberry Pi–based smart box with camera monitoring and Telegram notifications but did not implement receipt verification, allowing packages to be deposited without prior validation. Similarly, Faza et al. and Mee et al. employed ESP32-CAM for monitoring purposes but relied entirely on continuous internet connectivity and lacked local data storage, making their systems vulnerable to network instability. In contrast, the proposed system introduces a barcode-based two-way verification mechanism using a GM66 scanner to validate courier-delivered packages against tracking numbers pre-registered by recipients through a web interface. In addition, the system incorporates local MySQL-based data storage, enabling all transaction records and images to be retained during internet outages and automatically synchronized once connectivity is restored, thereby improving system robustness. From a communication perspective, the proposed system supports multi-platform notifications via both a web interface and WhatsApp, with experimental results showing notification delivery within 1–2 seconds. Furthermore, quantitative evaluation indicates that the barcode scanner performs optimally at a distance of 10–19 cm with consistent detection accuracy, a level of performance characterization often absent in prior studies. Overall, compared to existing approaches, this study provides greater analytical depth and practical applicability by combining secure package verification, offline data synchronization, real-time visual monitoring, and multi-channel notifications, resulting in a more reliable IoT-based package reception solution.

IV. CONCLUSION

This research aimed to design and implement an IoT-Based Package Drop Box system utilizing Arduino Uno and ESP32-CAM, integrated with a web-based platform and WhatsApp notification service, as a solution to the challenges faced in package delivery when recipients are not present at home. Based on the results of testing and analysis, the system successfully achieved its objectives, demonstrating effective integration between the GM66 barcode sensor, microcontroller units, camera module, web system, and WhatsApp notification service. The developed system is capable of registering and verifying tracking numbers (resi), automatically unlocking the drop box when valid data is detected, monitoring package conditions inside the box using the ESP32-CAM-based CCTV module, and delivering real-time status notifications to users via WhatsApp. All processes operated smoothly and synchronously, even under unstable network conditions, supported by a local data storage and resynchronization mechanism.

The system's effectiveness was demonstrated through several validation tests: verification of tracking number formats and registration status, barcode reading accuracy within the optimal range of 10–19 cm, reliability of camera-based monitoring, and accuracy of automated WhatsApp notifications. These results confirm that the proposed system provides a secure, automated, and remotely accessible solution for package reception management. For future research, the system can be enhanced with advanced features, such as courier facial authentication using AI-based image processing, voice recognition for secondary verification, and integration with smart home locks and digital wallet systems for authentication and automated payment. Furthermore, improving data security and system scalability should be prioritized to support wider implementation in smart home environments and office logistics systems.

REFERENCES

[1] H. KALKHA, A. Khiat, A. Bahnasse, and H. Ouajji, "The rising trends of smart e-commerce logistics," *IEEE Access*, vol. PP, p. 1, 2023, doi: https://doi.org/10.1109/ACCESS.2023.3252566.

[2] W. Weryani, L. Lisniawati, H. Hamsani, and R. Susi, "E-Commerce Trends in Indonesia : A Literature Review on Adaptation and Consumer Behavior," in *Proceedings of the 1st International Conference of Economics,*

Management, Accounting, and Business Digital (ICEMAB 2024), Atlantis Press International BV, 2024, pp. 35–41. doi: https://doi.org/10.2991/978-94-6463-614-7.

[3] A. Rejeb, S. Simske, K. Rejeb, and H. Treiblmaier, "Internet of Things Internet of Things research in supply chain management and logistics : A bibliometric analysis," *Internet of Things*, vol. 12, no. 100318, 2020, doi: https://doi.org/10.1016/j.iot.2020.100318.

[4] A. Wahyu, "Jejak Inovasi Teknologi E-Commerce Di Indonesia : Perkembangan Dan Tantangan Di Era Digital," *J. Ilmu Sains dan Teknol.*, vol. 1, no. 1, pp. 12–17, 2025, [Online]. Available: https://artmediapub.id/index.php/JIST/article/view/68

[5] N. Zhang, R. Bahsoon, N. Tziritas, and G. Theodoropoulos, "A Digital Twin Approach for Adaptive Compliance in Cyber-Physical Systems : Case of Smart Warehouse Logistics," *IEEE Trans. Syst. MAN, Cybern. Syst.*, vol., no., pp. 1–10, 2023.

[6] Q. Zhang and E. Demir, "Parcel locker solutions for last mile delivery : a systematic literature review and future research directions," *Front. Futur. Transp*, vol. 6, pp. 1–18, 2025, doi: https://doi.org/10.3389/ffutr.2025.1654621.

[7] J. Allen, M. Browne, A. Woodburn, and J. Leonardi, "The Role of Urban Consolidation Centres in Sustainable Freight Transport," *Transp. Rev.*, vol. 32, no. 4, pp. 473–490, 2012, doi: https://doi.org/10.1080/01441647.2012.688074.

[8] R. Mangiaracina, A. Perego, A. Seghezzi, and A. Tumino, "Innovative solutions to increase last-mile delivery efficiency in B2C e-commerce: a literature review," *Int. J. Phys. Distrib. Logist. Manag.*, vol. 49, no. 9, pp. 901–920, 2019, doi: https://doi.org/10.1108/IJPDLM-02-2019-0048.

[9] P. K. M. Gowda, H. K. Ruthvik, R. S. Nathanya, D. Shetty, and P. C. K. N, "Enhancing Parcel Management by IoT Enabled Smart Locker System," *Int. Adv. Res. J. Sci. Eng. Technol.*, vol. 12, no. 11, pp. 256–261, 2025, doi: https://doi.org/10.17148/IARJSET.2025.121145.

[10] M. Mokhsin *et al.*, "Intelligent Parcel Receiving Box System (ParcelRestBox) Towards the Future of Malaysia Smart City Development," *J. Electr. Syst.*, vol. 20, no. 10, pp. 4326–4336, 2024, [Online]. Available: https://journal.esrgroups.org/jes/article/view/6068

[11] R. D. Handayani, Z. Jamal, L. Rosmalia, N. H. Sudibyo, and M. Alkahfiansyah, "Intelligent and Secure Package Receiver System Utilizing Internet of Things (IoT) Technology," *Vokasi Unesa Bull. Eng. Technol. Appl. Sci.*, vol. 2, no. 3, 2025, doi: https://doi.org/10.26740/vubeta.v2i3.38254.

[12] K. A. Valli, B. Yuvaraju, G. A. Kumar, M. H. Girish, M. Mouli, and Y. Misra, "International Journal of Research Publication and Reviews Smart Parcel Delivery Receiving Box for Secure Parcel Drop-Off," *Int. J. Res. Publ. Rev.*, vol. 6, no. 10, pp. 5853–5860, 2025, doi: https://doi.org/10.55248/gengpi.06.1025.3789.

[13] A. S. Shuaibu, A. S. Mahmoud, and T. R. Sheltami, "A Review of Last-Mile Delivery Optimization : Strategies , Technologies , Drone Integration , and Future Trends," *Drones*, vol. 9, pp. 1–46, 2025, doi: https://doi.org/10.3390/drones9030158.

[14] H. Y. Song and H. Han, "A Design of a Parcel Delivery System for Point to Point Delivery with IoT Technology," *MDPI Futur. Internet*, vol. 12, no. 70, pp. 1–13, 2020, doi: https://doi.org/10.3390/fi12040070.

[15] A. Al-fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things : A Survey on Enabling Technologies , Protocols and Applications," *IEEE Commun. Surv. Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015, doi: https://doi.org/10.1109/COMST.2015.2444095.

[16] S. V Nthin, S. Nithin, S. N. Nithish, V. Nithishwara, and N. Jothy, "SMART AND SECURE PARCEL MANAGEMENT USING IOT," *Int. J. Res. Anal. Rev.*, vol. 12, no. 2, pp. 788–793, 2025, doi: https://doi.org/10.6084/m9.doi.one.IJRAR25B215.

[17] X. L. Y. M. Tang, K. Y. Chau, D. Xu, "Consumer perceptions to support IoT based smart parcel locker logistics in China," *J. Retail. Consum. Serv.*, vol. 62, 2021, doi: https://doi.org/10.1016/j.jretconser.2021.102659.

[18] M. Jonathan, D. Hendriana, and H. Nasution, "Development of Smart Locker System with Simultaneous User Access Capabilities for Warehouse Package Handovers," *J. Appl. Sci. Adv. Eng.*, vol. 3, no. 1, pp. 1–6, 2025, doi: https://doi.org/10.59097/jasae.v3i1.48.

[19] W. U. Khan and S. Arabia, "IoT-Based Smart Lockers: A Validation for the Saudi Arabian Market," *J. Bus. Manag. Stud.*, pp. 218–221, 2025, doi: https://doi.org/10.32996/jbms.

[20] U. Azrin, I. Ziad, and S. Suroso, "Rancang Bangun Smart Box Penerima Paket Berbasis IoT Menggunakan Raspberry Pi," *Emit. J. Tek. Elektro*, vol. 22, no. 2, pp. 118–125, 2022, doi: https://doi.org/10.23917/emitor.v22i2.19405.

[21] F. Dila Faza, D. Mardiyanti, E. Budihartono, and A. Winarso, "Smart Box Penerima Paket Berbasis Website Menggunakan Esp32-Cam Dan Notifikasi Telegram," *J. Manuf. Enterp. Inf. Syst.*, vol. 1, no. 2, pp. 103–115, 2023, doi: https://doi.org/10.52330/jmeis.v1i2.176.

[22] F. Junaidi, Jasmir, and W. Riyadi, "Perancangan Box Penerimaan Paket Berbasis IoT," *J. Inform. Dan Rekayasa Komput.*, vol. 4, no. 2, pp. 1032–1039, 2024, doi: https://doi.org/10.33998/jakakom.v4i2.

[23] C. Valupi, T. Ariyadi, N. Paramyta, and T. D. Purwanto, "Prototipe Sistem Smart Box untuk Keamanan Penerimaan Paket Online Shopping," *J. Ilm. Univ. Batanghari Jambi*, vol. 25, no. 1, pp. 964–971, 2025, doi: https://doi.org/10.33087/jiubj.v25i1.5675.

[24] Y. W. Saputra and R. Z. Abidin, "Perancangan Sistem Penerimaan Paket Otomatis Berbasis Internet of Things (Iot) Untuk Mendeteksi dan Mengamankan Paket Pengiriman," *J. Sains Student Res.*, vol. 3, no. 5, pp. 1094–1104, 2025, doi: https://doi.org/10.61722/jssr.v3i5.6437.

[25] E. D. Febriyanti, Alimuddin, and H. M. Putra, "Rancang Bangun Penerimaan Box Paket Berbasis Internet of Things (IoT)," vol. 2, no. 1, pp. 37–45, 2024, doi: https://doi.org/10.29408/jprinter.v2i1.23825.