Multitek Indonesia: Jurnal Ilmiah
Volume: 19 No. 2 Desember 2025, Hal. 157-170
ISSN: 1907-6223 (print)
ISSN: 2579-3497 (Online)
http://journal.umpo.ac.id/index.php/multitek
https://doi.org/10.24269/mtkind.v19i2.12405

DEVELOPMENT OF AIR PACKIT: AN IOT-BASED PACKAGE RECEIVING BOX WITH GM66 BARCODE SCANNER

Noval Prakoso[1*], Ariep Jaenul[1], Mauludi Manfaluthy[1]
[1] Faculty of Engineering and Computer Science, Jakarta Global University
Jl. Boulevard Grand Depok City, Tirtajaya, Kec. Sukmajaya, Kota Depok, Jawa Barat
Email : novalprakoso@student.jgu.ac.id, ariep@jgu.ac.id, mauludi@jgu.ac.id
*Alamat Korespondensi: novalprakoso@student.jgu.ac.id

Dikimkan: 10 September 2025 Direvisi: 6 Desember 2025 Diterima: 18 Desember 2025

Abstract

The rapid growth of e-commerce services demands secure and efficient solutions for receiving packages without direct interaction between couriers and users. This study develops an IoT-based automatic package receiving box using an ESP32 microcontroller, designed through a prototyping approach consisting of identification, design, implementation, and testing stages. The system integrates a GM66 barcode scanner for validating tracking numbers, a load cell sensor for detecting package placement, and Telegram as a real-time communication interface. Experimental testing shows that the GM66 sensor performs consistently within a 10–30 cm reading range, with a response time of 1.33–2.71 seconds, while the complete verification and locker access process requires an average of 48.2 seconds. The load cell provides stable performance with a measurement error of 0.05 percent, and the system maintains reliable operation under normal Wi-Fi conditions. These results demonstrate that the prototype functions effectively and offers strong potential to enhance package security, improve delivery efficiency, and support independent package reception in modern e-commerce environments.

Keywords: E-commerce, ESP32, Prototyping, Load Cell Sensor, Telegram

INTRODUCTION

The increasing use of internet-based services has significantly influenced consumer behavior, particularly in the rapid growth of e-commerce transactions. In Indonesia, internet penetration reached 215.62 million users in 2022–2023, supporting the expansion of digital platforms and online purchasing activities [1], [2]. As a result, the logistics and delivery sector is experiencing a continuous increase in parcel volume, driven by the rising number of e-commerce users, which is projected to reach 220 million by 2025 [3]. This growth underscores the importance of efficient, accurate, and secure delivery processes for end users.

In practice, shipping goods through the freight industry does not always run smoothly. Common problems include damaged packages, delayed delivery, and lost packages [4]. This can occur because consumers are not at home when the package arrives, so the courier leaves the package in an arbitrary place, resulting in damage or loss. In addition, the use of cash on delivery systems can also cause delays in package delivery [5]. Several cases have been reported, such as packages being left in the ventilation system of a house when the recipient was not at home [6]. According to the source pluginongkoskirim.com, if the recipient is not at the destination address when the courier arrives, the customer can contact the JNE call center to reschedule the delivery for a time when the recipient is available [7].

This highlights the need for a goods storage system as a solution for secure package delivery. Kevin Ashton, a British man, first introduced the concept of the Internet of Things (IoT) in 1999. The term “Things” in the Internet of Things refers to various devices and tools that can be connected through a wide internet network [8]. The Internet of Things (IoT) is a technological concept that allows devices around us to connect and communicate with each other through the internet network [9]. Previous research has designed an Internet of Things-based parcel delivery box using Wemos D1 R2 with a barcode scanner and Telegram application for notifications, but it does not yet have a parcel weight detection system [10].

The scientific novelty of this research lies in the development of an Internet of Things-based parcel delivery box that adds a load cell sensor to detect parcel weight. This system will automatically open the locker door if the package matches the data in the database and send a notification to the user, as well as provide information on whether the locker is full or empty. The problem formulated in this study is how to design and build an Internet of Things-based package receiving box system that is capable of validating packages with a barcode scanner, detecting package weight with a load cell sensor, and sending notifications to users via the Telegram application in real time to support the package receiving process in e-commerce services. This research aims to produce an automatic parcel locker as a security solution for e-commerce service users in receiving parcels remotely. This system has five lockers called Air Packit based on the Internet of Things.

RESEARCH METHOD

The method used in this study is prototyping. A prototype is the first iteration in software system development that serves as an initial representation of an idea [11]. The following are the stages of the prototyping approach [12].

1. Problem Identification

There are several problems that occur when receiving packages, including delays in receiving COD (Cash on Delivery) packages because the recipient is not at the delivery address, packages being left in random places where they could potentially be damaged, and packages being lost with a lengthy claim process.

2. Prototype Design

The following are several stages of prototype design:

a. Hardware Requirements Analysis

The hardware required includes:

Table 1. Hardware Requirements Analysis

No.	Hardware	Function
1.	ESP32 Devkit V1	Main microcontroller for system control
2.	GM66 Barcode Scanner	Reads package tracking barcode
3.	Load Cell + HX711	Measures package weight
4.	16x2 I2C LCD	Displays system information
5.	12V Power Supply	Power distribution
6.	Relay	Controls solenoid door lock
7.	Solenoid Door Lock	Locker locking mechanism
8.	PC / Laptop	Used for programming and monitoring

b. Software Requirements Analysis

The software required includes:

Table 2. Software Requirements Analysis

No.	Software	Function
1.	Arduino IDE ver. 2.3.4	Programming ESP32 Devkit V1
2.	Fritzing ver. 0.9.3	Creating schematic visualization
3.	Firebase CLI 13.29.1	Creating NoSQL database
4.	Telegram	Sending and Receiving Messages
5.	CorelDRAW ver. 21.2.0.706	Creating design of Air Packit
6.	Windows OS	The OS system used by researchers

c. Flowchart System

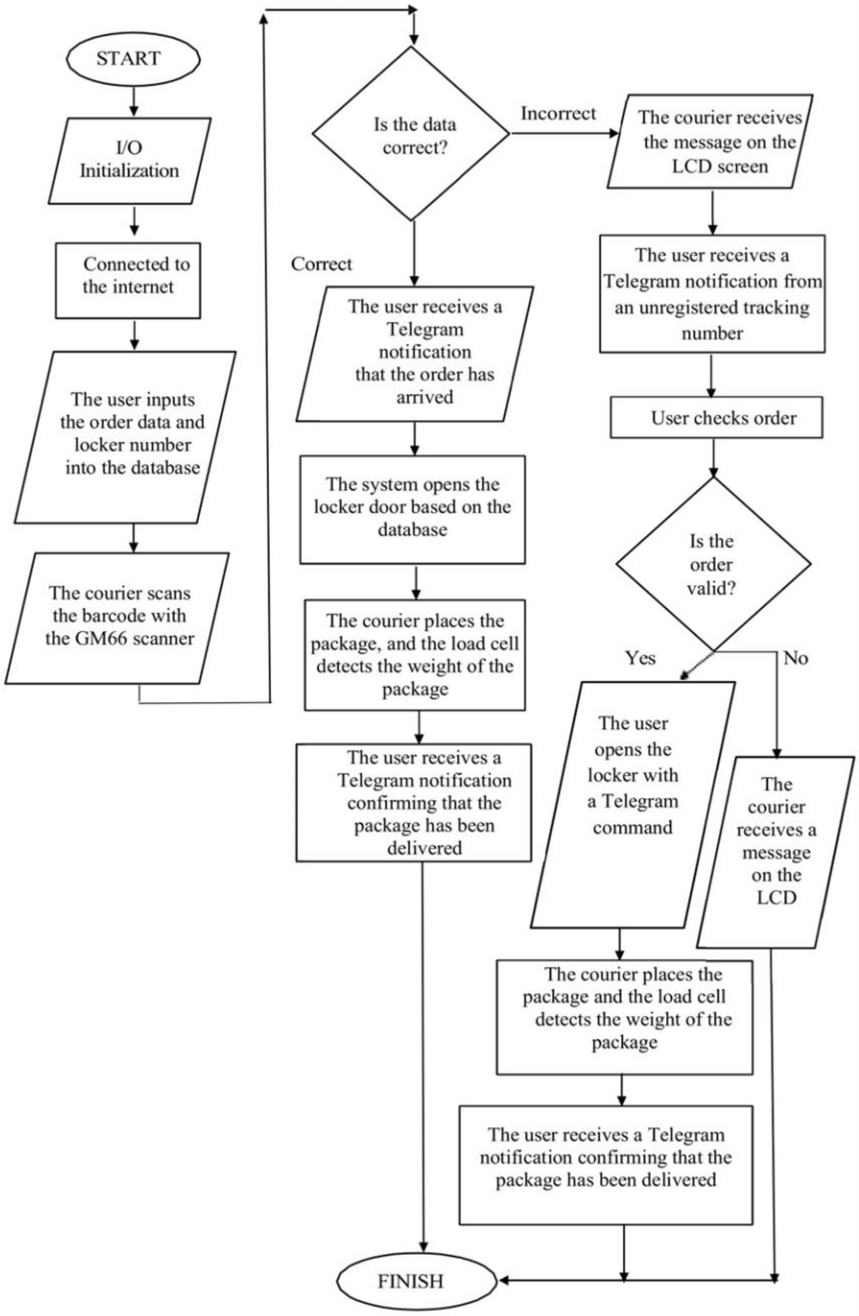

Figure 1. Flowchart System

Figure 1 provides a detailed overview of the system workflow, beginning with the initialization of input/output components and the establishment of an internet connection. After the user enters the order data and assigns a locker number in the database, the courier scans the barcode using the GM66 scanner. The system then validates the scanned data against the database.

If the barcode matches the stored data, the system automatically opens the corresponding locker, allowing the courier to place the package. The load cell detects the weight of the newly stored package, and the system immediately sends a confirmation notification to the user via Telegram.

If the barcode does not match, the courier receives an error message on the LCD, and a Telegram notification is sent to the user indicating that an unregistered tracking number has been scanned. The user can then verify the order data and, if valid, remotely open the locker through a Telegram command.

After the courier places the package and the load cell confirms the placement, the system again notifies the user that the package has been successfully delivered.

d. Schematic Circuit

A schematic circuit is used to visualize the layout of electronic components and the paths between components in an electronic system [13]. The following is the Air Packit schematic circuit.

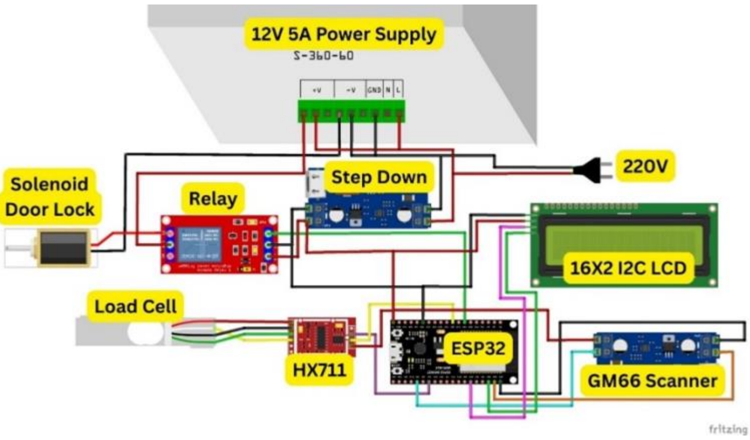

Figure 2. Schematic Circuit

e. Design Draft

The design draft was created based on the results of the schematic series. The following is the design draft that the author designed using CorelDRAW software.

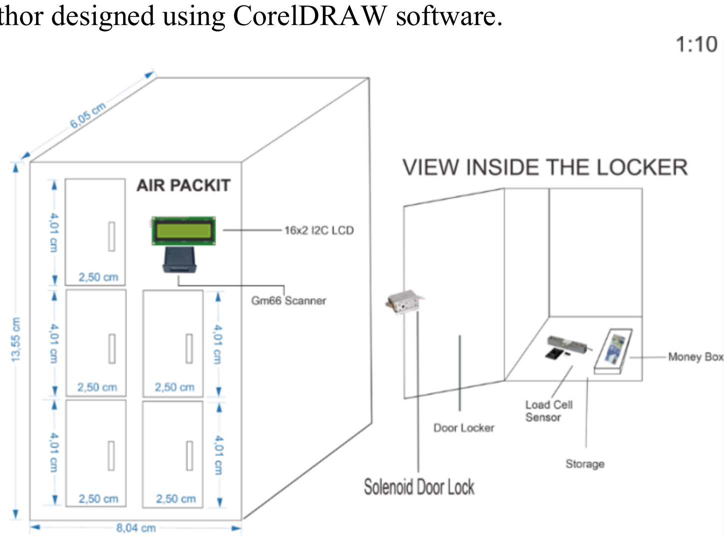

Figure 3. Design Draft

RESULTS AND DISCUSSION

1. Prototype Implementation

From the design results, the author implemented it into a physical prototype. The materials used were acrylic and hollow iron with thicknesses of 15 mm and 8 mm, respectively.

Figure 4. Air Packit Display

From Figure 4, it can be seen that Air Packit has five lockers to meet the needs of e-commerce users for one week. The author has created a PCB (Printed Circuit Board) for the designed schematic circuit.

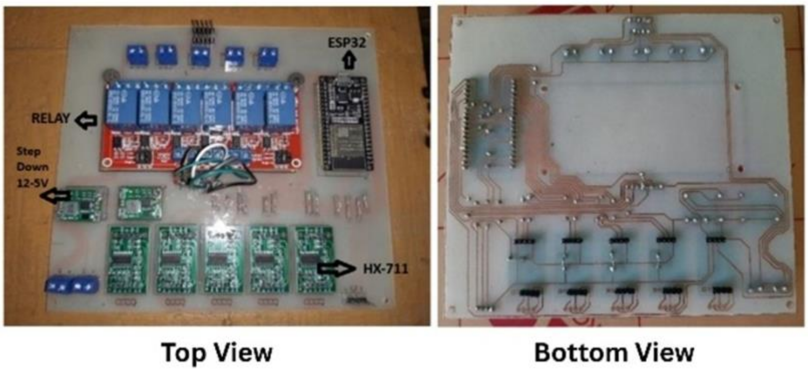

Figure 5. Printed Circuit Board

Figure 5 shows the top and bottom views of the PCB. On the top, there are five HX711 modules. Meanwhile, on the bottom side, there are components that the author has soldered to attach the microcontroller pins, two 5V 3A step-down converters, five HX711 modules, three male pin headers, and seven terminal blocks.

Before testing the function, there are several steps that need to be taken:

- Ensure that the PCB is soldered and the components are connected as shown in Figure 5.
- Upload the C language program to the Air Packit system using a USB cable and Arduino IDE software.
- Once the upload is complete, disconnect the USB cable. Then, connect the power supply to the plug and 220V outlet to turn on the system.

2. Function Testing

After the system was turned on, the author calibrated the load cell sensor, then tested the components. The components to be tested included the ESP32 microcontroller control center, GM66 sensor, 12C 16x2 LCD, and solenoid door lock. However, the calibration of the load cell sensor was only to determine the availability of lockers, not as an important parameter. When a load cell sensor is subjected to force or load, it will undergo elastic deformation in the form of strain changes, namely compression and pressure on the strain gauge. A strain gauge is a zigzag-shaped conductor on a membrane, whose resistance will increase when the membrane is stretched [14]. This pattern can be seen in the image below.

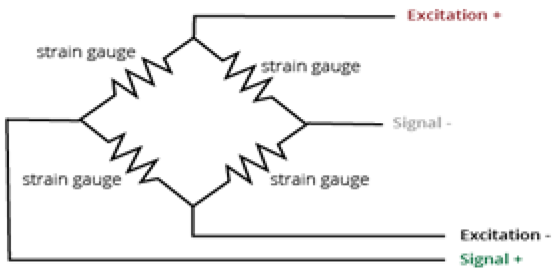

Figure 6. Wheatstone Bridge
Source: www.randomnerdtutorials.com

Figure 6 shows a Wheatstone bridge circuit. A Wheatstone bridge is an electrical circuit consisting of four resistors arranged in a bridge configuration. There are four points, consisting of two points used for positive and negative inputs, and two points used for measuring the output voltage. Changes in strain on the strain gauge cause changes in electrical resistance, which results in an imbalance in the Wheatstone bridge and produces an electrical signal proportional to the applied force. This signal is then processed to display the measured weight or force value [15].

The first step in performing calibration is to find the offset calibration factor or correction value by comparing the mass value of the measuring instrument (scale) with the mass of the test instrument (load cell sensor) [16], [17]. Once both data sets have been obtained, the correction value and relative error are calculated using the equation below.

$$K = m_{std} - m_{load\ cell} \qquad (1)$$

Description:
K = Correction Value (gr)
m_{std} = Standard Mass Value (gr)
$m_{load\ cell}$ = Load Cell Mass Value (gr)

After calculating the above equation, the calibration correction value, commonly known as the offset value, is obtained. After determining the correction value, the author adds the correction value to the load cell reading value. This gives the value after calibration. Next, to calculate the error comparison between conventional scales and load cells, the following formula is used.

$$Error = \frac{\text{mload cell} - \text{mstd}}{\text{mstd}} \qquad (2)$$

Figure 7. Load Cell Sensor Calibration

Figure 7 presents the calibration results for an object with a reference mass of 600 grams. Prior to calibration, the measurement deviation reached 36.5 grams, while the post-calibration error decreased significantly to 0.05%. This error level is considered highly accurate because typical low-cost load cell systems using HX711 amplifiers commonly exhibit accuracy deviations within the 1–5% range due to noise, non-linearity, and environmental fluctuations [18]. In contrast, an error of 0.05% corresponds to approximately 0.3 grams, which is substantially below the expected tolerance and therefore indicates strong calibration performance.

Additional analysis was conducted across all five load cells in the system. The results showed that the sum of correction values did not match perfectly with the difference between pre-calibration and post-calibration readings. This discrepancy aligns with the load cell datasheet specification, which reports an intrinsic error tolerance of 0.02% [19]. However, experimental findings revealed that the actual error values varied between 0.05% and 0.12% across the five sensors. Although the datasheet does not explain the cause of this variation, these differences are technically plausible and can be attributed to several engineering factors.

One contributing factor is the length of the load cell wiring. Longer cables increase electrical resistance and susceptibility to electromagnetic interference, both of which can introduce small deviations in signal amplification. Another factor is the surface area and mechanical stability of the load cell mounting plate. Inconsistent surface contact or uneven force distribution on the load cell can cause minor non-linear responses, leading to variation in output accuracy. Furthermore, manufacturing tolerances between individual load cells naturally result in slight differences in sensitivity and zero offset. that the overall system performance is robust and suitable for real-time package verification.

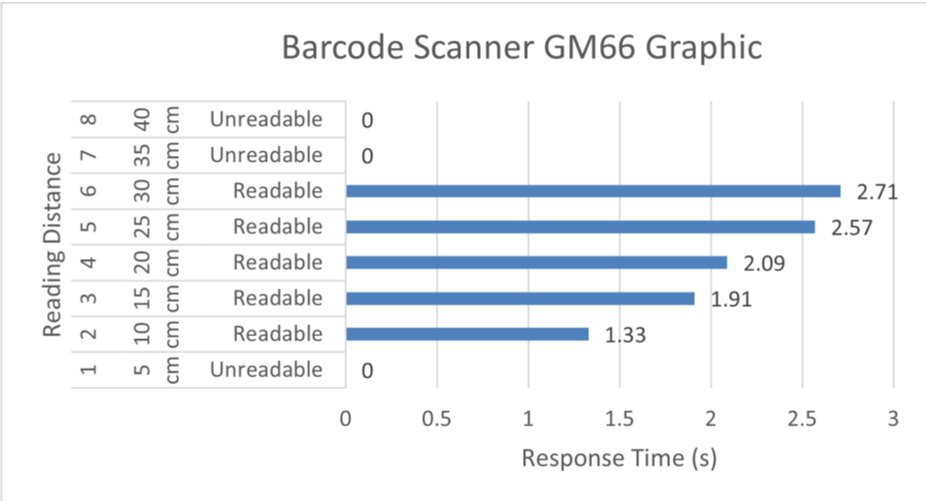

Figure 8. Graphic Barcode Scanner GM66

The results in Figure 8 showed that the GM66 barcode scanner has a limited effective reading range. Reliable detection occurs only between 10 cm and 30 cm, while distances shorter than 10 cm or greater than 30 cm result in unreadable barcodes. The response time also increases with distance, starting from 1.33 s at 10 cm and reaching 2.71 s at 30 cm. This behavior aligns with the characteristics of LED-aimer–based scanners, where signal strength decreases as distance increases, reducing the reflected light intensity received by the sensor [20]. This effect is consistent with the Inverse Square Law, which states that light intensity decreases proportionally to the square of the distance. At longer distances, increased visual noise and weaker reflections force the GM66's decoding algorithm to perform multiple attempts before successfully identifying the barcode pattern, resulting in longer response times. Overall, despite these limitations, the GM66 performs within the 10–30 cm reading distance specified in its datasheet.

Table 3. System Processing Duration Test

No.	Tracking No.	Status (Found / Not Found)	GM66 (Readable / Unreadable)	Solenoid Door Lock (ON / OFF)	Telegram Notification Time (s)
1.	TKP01 5C1CQG9Y	Found	Readable	ON	41
2.	4381442610	Found	Readable	ON	31
3.	SPXID047585494456	Found	Readable	ON	32
4.	SPXID042952105518	Found	Readable	ON	89
5.	SPXID04384875926C	Found	Readable	ON	48
6.	TKP01-H1UPLWC4	Not Found	Readable	OFF	-
7.	SPXID044105863666	Not Found	Readable	OFF	-
8.	JX3283029581	Not Found	Readable	OFF	-

In Table 3, the GM66 barcode scanner can read the barcode on the package and open the door via a solenoid door lock. This occurs because the tracking number is registered in the Firebase database. The reception duration varies, depending on how quickly the researcher responds to incoming messages by sending a telegram command and placing the package in the locker. In this case, the fastest reception time was in the second locker, which was 31 (thirty-one) seconds. Meanwhile, the slowest reception was in the fourth locker, which was 89 (eighty-nine) seconds. For registered packages, showing in the following figure.

Figure 9. Telegram Notification For Registered Tracking Number

In addition, the courier will receive messages on the LCD screen. That message shown "Welcome!" Then, after scanning, the courier will be asked to wait.

Default display **After scanning the barcode**

Figure 10. LCD Messages

For numbers 6-10, which are unregistered tracking numbers (not found), the GM66 barcode scanner can still read the barcode on the package. However, the solenoid door lock remains in the OFF position and does not open the door. This happens because the tracking number is not registered in the Firebase database.

Figure 11. Telegram Notification For Unregistered Tracking Number

In Figure 10 showed a scenario for checking locker availability. When the courier is asked to wait for the user to check whether or not they want to order the package. After confirming the order, the user can monitor the locker availability status via Telegram messages by sending a Telegram command to perform a check. The load cell sensor then detects the weight in each locker and sends a Telegram message back with a list of occupied or available lockers. Then, if the package is not registered, it will display the following message.

LCD message if the barcode does not match/is not registered in the database. "Sorry, barcode does not match"

Figure 12. Barcode does not match

Below is the real-time database that the author uses with Firebase. Firebase Real-time Database is a data storage platform. As the name Real-time Database suggests, all data is stored directly via cloud media and can be accessed immediately. Firebase includes a NoSQL database in the form of a JSON tree (data is stored in a large, hierarchically structured JSON format). Here is the program that the author uses.

```
{
    "orders": {
      "TKP01-5C1CQG9Y": {
        "itemName":
        "Nama Barang",
        "trackingNumber":
        "No. Resi",
        "lockerNumber":
        "No. Loker"
      }
    }
}
```

In the item name section, enter the name of the item; in the tracking number section, enter the tracking number; and in the locker number section, enter the number of the locker where we want to place the item. Here is the real-time database display.

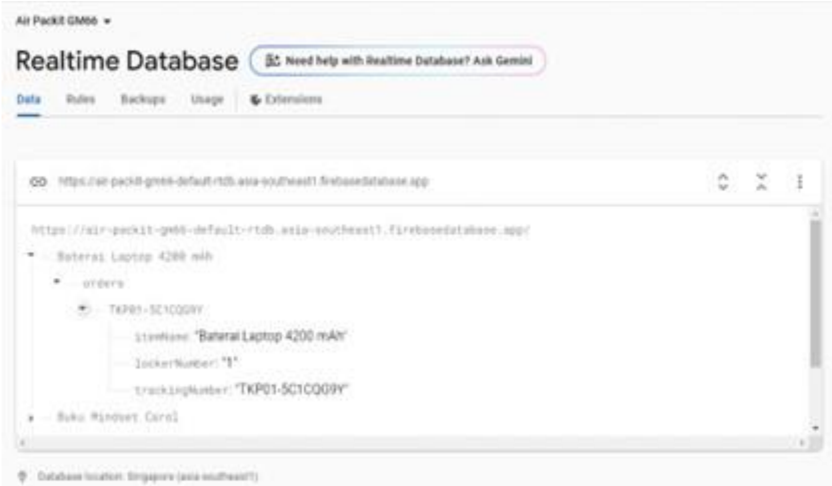

Figure 13. Realtime Database

For clarification, here is a system evaluation table for performing test procedures on several components.

Table 4. System Evaluation

Components	Procedure Test	Status
ESP32	As a control center controlling systems such as sensors and actuators	Success
GM66 Barcode Scanner	QR Code Scanner on the package	Success
LCD	Displaying text messages	Success
Solenoid Door Lock	Opening and closing locker doors	Success
Telegram	Sending notifications.	Success

Based on the test results, the ESP32 successfully operated all input and output components, including the GM66 barcode scanner, load cell sensors, 16×2 I2C LCD, solenoid door locks, and Telegram notifications. The GM66 accurately detected registered QR codes, allowing packages to be placed in the appropriate locker, while unreadable or damaged codes prompted users to verify the order database. The LCD displayed the corresponding system messages, and the solenoid door locks consistently activated and deactivated as expected. Telegram also functioned properly by delivering notifications for both successful and unsuccessful scans.

This study did not specifically evaluate system performance under unstable or intermittent internet conditions. However, based on the system architecture—which depends on continuous data transmission between the ESP32 microcontroller, Firebase Database, and Telegram Bot—unstable connectivity would theoretically affect response time, message delivery, and synchronization of stored data.

When the connection becomes unstable, the ESP32 may experience delayed HTTP requests, failure to upload data, or repeated retries triggered by timeout events. This condition could result in temporary loss of real-time functionality, such as delayed barcode verification or postponed notification delivery through Telegram [21].

With the introduction of Air Packit as an automatic package receiving box, users can more easily manage deliveries without direct interaction with couriers, improving convenience, security, and logistical efficiency in homes, dormitories, and high-traffic residential areas.

3. Comparison with Previous Research

The following is a comparison table between previous research [10] and current research.

Table 5. Comparison with Previous Research

No.	Aspect	Previous Research [10]	Air Packit
1.	Microcontroller	Wemos D1 R1 (ESP8266-based), single-core, lower processing speed	ESP32, dual-core 240 MHz, higher performance, supports multitasking
2.	Connectivity	Wi-Fi only	Wi-Fi + Bluetooth, more flexible for IoT integration
3.	Primary Function	Barcode-based package verification using GM66 scanner	Barcode verification + load cell-based weight detection
4.	Locker Management	Single locker configuration	Five independent lockers with automated empty-locker selection
5.	Sensor Used	GM66 barcode scanner	GM66 barcode scanner + Load Cell HX711 module

Table 4 presents a comparative analysis between the proposed Air Packit system and the previous work described in [10]. The earlier study implemented an IoT-based parcel receiving box using the Wemos D1 R1 microcontroller, which is based on the ESP8266 architecture and supports only basic Wi-Fi connectivity. In contrast, the system developed in this research utilizes the ESP32, which offers significantly higher computational performance due to its dual-core CPU, larger memory resources, and integrated Bluetooth capability. These hardware improvements enable more efficient processing of IoT tasks and support the integration of multiple sensors and actuators.

In terms of system functionality, the study in [10] focused solely on barcode-based package validation using the GM66 scanner and was limited to a single-storage compartment. Meanwhile, the Air Packit prototype introduces a load cell module for real-time weight measurement, allowing enhanced package verification and automated identification of empty lockers. This study also expands the storage capacity to five independent lockers, enabling multi-package management within a single system. Overall, the enhancements implemented in this research demonstrate improvements in verification accuracy, automation, and system scalability compared to the work presented in [10].

CONCLUSION

The development of this prototype demonstrates that an IoT-based package receiving system can improve the security and efficiency of remote package delivery. The integration of the GM66 barcode sensor, LCD module, Telegram communication, and load cell sensor enables accurate package validation, real-time notifications, and reliable weight measurement with an error rate of 0.05%. System testing also shows that the barcode reading response time ranges from 1.33 to 2.71 seconds within the optimal scanning distance, and the overall package reception process—from barcode scanning to package placement—requires an average of 48.2 seconds.

Despite the successful implementation, this prototype still has limitations that open opportunities for further development. Future research is recommended to enhance system security by implementing stronger protection mechanisms, such as improved access control, data encryption, and integration with camera modules for visual verification. Additionally, expanding the system through a dedicated mobile application could improve user experience by providing a more seamless interface for monitoring package status, receiving alerts, and managing delivery histories. These enhancements are expected to increase the system's reliability, robustness, and applicability for broader real-world use.

ACKNOWLEDGEMNETS

The author would like to express his sincere gratitude to the supervisor for the valuable guidance, advice, and continuous support throughout the research process. The author also wishes to extend appreciation to the Multitek Indonesia team for providing article templates, which greatly facilitated the systematic writing and compilation of this scientific article.

REFERENCES

[1] Okolo, C. V., Susaeta, A., & Okorie, D. I. (2025). On the relationship between internet usage and technological innovation: Insight from world panel data. *Sustainable Futures*, *9*, 100403. https://doi.org/https://doi.org/10.1016/j.sftr.2024.100403

[2] Khalil, M., Saleh, A., Ginting, B., Sekretariat Badan, S., Pangan, K., Pertanian, K., Program, D., Komunikasi, S., Pertanian, P., & Pedesaan, D. (2018). PEMANFAATAN INTERNET SEBAGAI MEDIA KOMUNIKASI OLEH PENGELOLA SEKRETARIAT DEWAN KETAHANAN PANGAN PROVINSI. *Jurnal Komunikasi Pembangunan*, *15*(2). https://doi.org/10.46937/15201722800

[3] Affifa. (2024). *Indonesia E-commerce Market: Growth Insights, Challenges, and Prospect*. https://cube.asia/read/indonesia-e-commerce-market-growth-insights-challenges-and-prospect/

[4] Chopra, A., Arora, M., & Pandey, S. (2021). *Delivery Issues Identification from Customer Feedback Data*. https://doi.org/https://doi.org/10.48550/arXiv.2112.13372

[5] Rizki Damanik, R., & Grata, F. (2021). PERANCANGAN APLIKASI PENYEDIA JASA PENGIRIMAN PAKET BERBASIS QR CODE. *Journal Information System Development (ISD)*, *Vol. 6* (No. 1). https://ejournal-medan.uph.edu/index.php/isd/article/view/431/0

[6] Azrin, U., Ziad, I., & Suroso, S. (2022). Rancang Bangun Smart Box Penerima Paket Berbasis IoT Menggunakan Raspberry Pi. *Emitor: Jurnal Teknik Elektro*, *22*(2), 118–125. https://doi.org/10.23917/emitor.v22i2.19405

[7] Fauzan, Y. (2020). *KOTAK PENERIMA PAKET BERBASIS IoT MENGGUNAKAN MODUL ESP32-CAM*. https://repository.uinjkt.ac.id/dspace/bitstream/123456789/56069/1/YUSUF%20FAUZAN-FST.pdf

[8] Subni, G., Putra, A., Nabila, A., Pulungan, A. B., Negeri, U., Jl, P., & Air Tawar, H. (2020). Power Supply Variabel Berbasis Arduino. *JTEIN: Jurnal Teknik Elektro Indonesia*, *1*(2). https://doi.org/10.24036/jtein.v1i2.53

[9] Duhin Mukin, Y. (2023). Simulasi Jaringan Smart Home dengan Sistem Berbasis IoT. In *Jurnal Komunikasi Sains dan Teknologi* (Vol. 2, Issue 1). https://journal.proletargroup.org/index.php/JKST

[10] Rismayana, A. H., Mustopa, M. S., & Rohmayani, D. (2022). Rancang Bangun Kotak Penerima Paket Menggunakan Barcode Berbasis Internet of Things (IoT). *Journal Informatics and Electronics Engineering*, *02*(02), 35–40. https://ejournal.poltektedc.ac.id/index.php/jiee/article/view/677

[11] Nurhanifah, D., Sriyeni, Y., Studi Sistem Informasi Program Sarjana, P., & Teknologi dan Bisnis PalComTech, I. (2023). PERANCANGAN APLIKASI JASA SALON MENGGUNAKAN MODEL PROTOTIPE. *Https://Jurnal.Mdp.Ac.Id/*, *Vol. 2 No. 1*(Vol 2 No 1 (2023): The 2nd MDP Student Conference 2023). https://doi.org/https://doi.org/10.35957/mdp-sc.v2i1.4495

[12] Syarifudin, A. (2019). Perancangan Sistem Informasi Pengajuan dan Pelaporan Pembayaran Tunjangan Kinerja Kementerian Keuangan Menggunakan Metode Prototype. *Jurnal Sisfokom (Sistem Informasi Dan Komputer)*, *8*(2), 149–158. https://doi.org/10.32736/sisfokom.v8i2.641

[13] *Schematic Drawing - an overview | ScienceDirect Topics*. (n.d.). Retrieved September 2, 2025, from https://www.sciencedirect.com/topics/computer-science/schematic-drawing

[14] Kamble, V., shinde, V., Kittur, J., Scholar, R., & Profesoor. (2021). Overview of Load Cells. www.matjournals.com, *Vol.* *6*(Issue-3), 1–8. https://www.researchgate.net/publication/355370308_Overview_of_Load_Cells

[15] Ştefănescu, D. (2011). Strain gauges and Wheatstone bridges — Basic instrumentation and new applications for electrical measurement of non-electrical quantities. *International Multi-Conference on Systems, Signals and Devices, SSD'11 - Summary Proceedings*, 1–5. https://doi.org/10.1109/SSD.2011.5767428

[16] Al-Dahiree, O. S., Tokhi, M. O., Hadi, N. H., Hmoad, N. R., Ghazilla, R. A. R., Yap, H. J., & Albaadani, E. A. (2022). Design and Shape Optimization of Strain Gauge Load Cell for Axial Force Measurement for Test Benches. *Sensors*, *22*(19). https://doi.org/10.3390/s22197508

[17] Machado, P., Fachini, L., Machado, V., Szmoski, R., & Antonini Alves, T. (2019). Load cells calibration with a low cost data acquisition system. *Revista Brasileira de Física Tecnológica Aplicada*, *6*. https://doi.org/10.3895/rbfta.v6n1.10125

[18] Mukhammad, Y., Santika, A., Haryuni, S., & Artikel, A. W. (2022). Analisis Akurasi Modul Amplifier HX711 untuk Timbangan Bayi INFO ARTIKEL ABSTRAK. *Journal of Robotics and Control (JRC)*. https://doi.org/10.18196/mt.v4i

[19] Joy-IT. (2025). *Load Cell Datasheet*. https://www.joy-it.net/en/products/SEN-HX711-05

[20] Deng, X., Linnartz, J. P. M. G., Long, X., & Zhou, G. (2019). Reading Analysis for Barcode Scanner with Interference from LED-Based Lighting. *IEEE Access*, *7*, 96787–96798. https://doi.org/10.1109/ACCESS.2019.2930328

[21] Ardita, M., & Orisa, M. (2021). Wi-Fi-Based Internet of Things (IoT) Data Communication Performance in Dense Wireless Network Traffic Conditions. In *Journal of Electrical Engineering, Mechatronic and Computer Science* (Vol. 4, Issue 1). http://jurnal.unmer.ac.id/index.php/jeemecs