CEPAT Journal of Computer Engineering: Progress, Application and Technology
Vol. 5, No. 2, November 2026, pp. 137-148
ISSN: 2963-6728, DOI: https://doi.org/10.25124/cepat.v5i02.12387

IoT-Based Warehouse Item Location Indicator Using ESP32 and Smart Lighting

Muhammad Soeh Ferdy Djulhaqi[1], Mahmud Salam Farsani[2], Ardiansyah Ramadhan[3]
[1] Department of Computer Engineering, School of Electrical Engineering, Telkom University, Indonesia
[2] Department of Electrical Engineering, Officer Logistic PT PLN (Persero) UP3 Cimahi, Indonesia
[3] Center of Excellence for Intelligent Sensing-IoT, Telkom University, Bandung, Indonesia

Article Info

Article history:

Received October 10, 2026
Revised October 31, 2026
Accepted November 01, 2026

Keywords:

Internet of Things (IoT)
Warehouse Management
Item Location Indication
Smart Lighting
Pick-to-light
ESP32

ABSTRACT

Manual item location searching in a warehouse with numerous storage racks significantly decelerates the order-picking process and is highly susceptible to human error. This research designs and implements an Internet of Things (IoT)-based item location indicator system at the Gudang Pojok of PT XYZ. The system utilizes an ESP32 microcontroller that autonomously retrieves the latest delivery order data from the "Si Eneng" warehouse management application via Google Apps Script as an intermediary API every 61 seconds. Subsequently, it actuates WS2812B LED strips on the targeted racks and WiZ smart bulbs as supplementary area indicators. The infrastructure encompasses 14 storage racks, each equipped with 16 LEDs (totaling 224 LEDs) and one WiZ smart bulb. The implementation of a "one rack, one GPIO" architecture ensures that each rack can be controlled independently without bandwidth contention, while SN74HC245N ICs are utilized as level shifters to maintain LED data signal stability. Power consumption analysis indicates that a 5 V 10 A power supply is adequate, operating at approximately 50% load during the worst-case scenario when all racks are illuminated in red. Concurrently, AWG17 cables successfully mitigate voltage drops to below 5% for wire runs up to 5 meters. The implementation results demonstrate that the system reduces item search time from previously 120 seconds down to 25 seconds (a 79.16% time reduction), lowers the picking error rate from 15% to 0%, achieves a command-to-LED response latency of 1.32 seconds, and maintains stable power efficiency with AWG17 cables restricting voltage drop below 5%.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Ardiansyah Ramadhan
Center of Excellence for Intelligent Sensing-Iot
Telkom University, Bandung, Indonesia
Email: ardiansyahramadhanar@telkomuniversity.ac.id

1. INTRODUCTION

The operational efficiency of warehouse management relies heavily on the speed and precision of personnel in locating and retrieving items (order picking) [1], [2], [3]. This order-picking process is known to consume up to 55% of total warehouse operational costs , making it the most critical area for optimization [1], [4], [5]. At the Gudang Pojok of PT.XYZ, the massive number of storage racks frequently forces personnel to spend a considerable amount of time merely matching the item codes on the delivery order with their physical locations. This manual search process not only decelerates item retrieval but is also highly vulnerable to human errors, such as picking from the wrong rack or misidentifying the item's position [6], [7], [8]. The impact of this inefficiency becomes remarkably pronounced when daily logistical delivery volumes reach their peak, or during shift changes involving field personnel who have not yet comprehensively memorized the rack layout.

Various approaches have been proposed in the literature to enhance order-picking efficiency, ranging from pick-by-voice systems to smart glasses [6], [8], [9], [10]. However, comparative studies indicate that the pick-to -light system, which visually guides operators via light actuators directly at the item's location, yields the highest levels of accuracy and speed, along with the shortest adaptation time for new employees [5], [6],

[8], [11]. The implementation of pick-to-light in the Industry 4.0 era is further accelerated by Internet of Things (IoT) technology, which enables direct real-time integration between physical racks and the database management system [2], [3], [12], [13]. The utilization of microcontrollers such as the NodeMCU ESP32 has proven reliable as the backbone for low-cost IoT architectures for automated data acquisition and inventory control in warehouse environments [14], [15], [16], [17].

Although IoT-based pick-to-light systems promise high efficiency, the majority of existing implementations still rely on expensive hardware or complex network architectures [3], [5], [12] to facilitate actuator icommunication across a large scale of racks. Considering the practical requirements at the PT.XYZ Cimah Warehouse, there is a necessity for a system that is not only smart and integrated but also economical and easy to maintain [14], [15]. Therefore, this research designs and develops an IoT-based item location indicator system named "Si Eneng". This system leverages the ESP32 microcontroller as the central control unit to automatically actuate dual visual indicators WiZ smart bulbs as macro-area indicators and WS2812B LED strips as micro-slot indicators on the targeted racks based on the latest delivery order data retrieved from the application.

The primary contribution of this research focuses on the development of a low-cost pick-to-light architecture specifically designed for industrial environment reliability, encompassing four essential aspects: (1) the integration of automated delivery order data retrieval via an Application Programming Interface (API) without requiring repetitive manual input from personnel; (2) the application of a "one rack, one GPIO" architecture that ensures the LED indicators on each rack can be controlled independently without the risk of data collision; (3) the implementation of supplementary smart bulbs as macro-area indicators to reinforce spatial visibility within a vast and dimly lit warehouse building; and (4) a stable electrical design for 24/7 operations, which includes the use of SN74HC245N level shifters to preserve data logic signal integrity, custom Printed Circuit Board (PCB) design, cable voltage drop analysis, and the handling of complex multi-rack data within a single delivery order.

2. METHOD

This research employs a design and build approach encompassing the design of the system architecture, hardware integration, network configuration, and software development. The fundamental concept adapted within this architecture is the pick-to-light system, which has been proven effective in guiding the item search process and reducing human error in smart warehousing environments.

2.1 System Architecture

The system architecture is constructed based on an Internet of Things (IoT) framework tailored for modern warehouse management [2], [3], [12].The system is segregated into three primary layers: the "Si Eneng" web application serving as the interface and operational database, Google Apps Script functioning as the intermediary Application Programming Interface (API), and the hardware nodes centered around the ESP32 microcontroller located within the warehouse area [3]. The selection of the ESP32 is based on its capability as a low-cost IoT gateway equipped with integrated on-board wireless connectivity [14], [15]. The system's workflow is centralized on the ESP32, which periodically polls the application for the latest delivery order data and subsequently translates it into physical control signals for the actuators, namely the LED strips and smart bulbs situated in the rack area [3], [18].

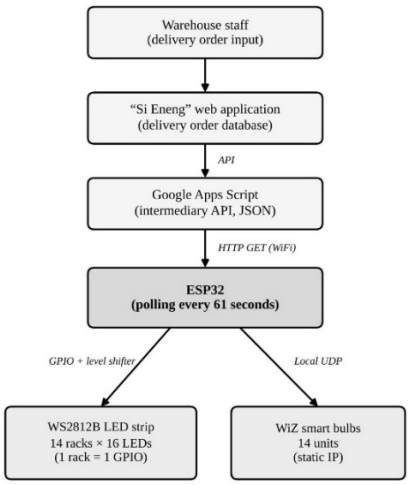


Figure 1 System Data Flow Diagram.

As illustrated in Figure 1, the system data flow begins with warehouse staff inputting delivery orders into the "Si Eneng" web application, which sends the data in JSON format via Google Apps Script to the ESP32 microcontroller that polls every 61 seconds over HTTP GET (Wi-Fi). Subsequently, the ESP32 translates this data into physical control signals to operate the rack actuators, specifically controlling the WS2812B LED strips via GPIO + level shifter and 14 units of static IP WiZ smart bulbs via local UDP [2], [3], [12].

2.2 Hardware Design

The hardware infrastructure adopts a distributed topology to control the lighting indicators independently [5], [12], [13], [19]. The components constituting the system include a router, a 5 V 10 A power supply unit, an ESP32 module, 14 WiZ smart bulbs, and 14 WS2812B LED strips arranged with 16 LEDs per rack (totaling 224 LEDs).

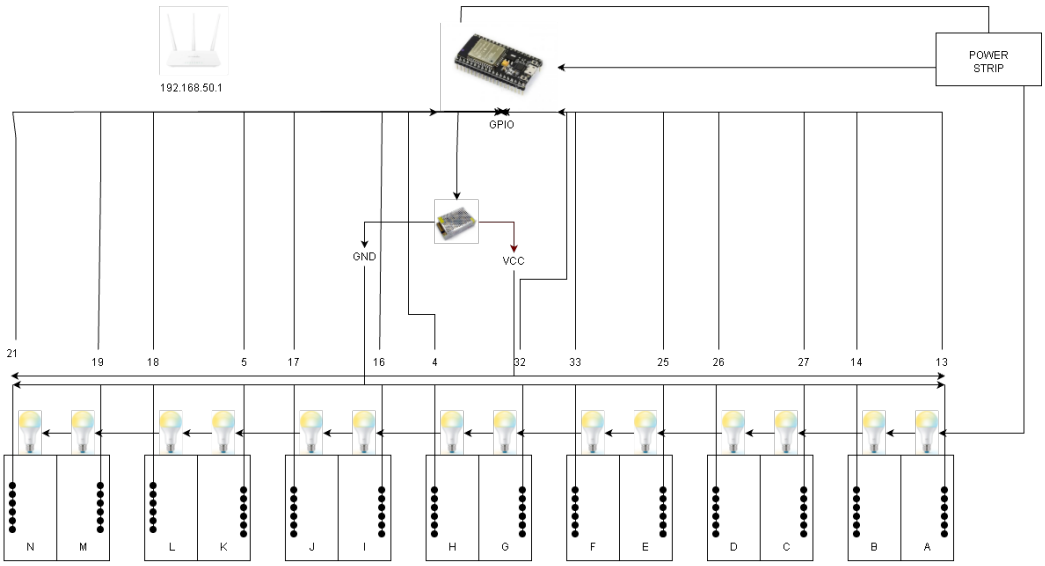


Figure 2 Architecture and data flow of the item location indicator system.

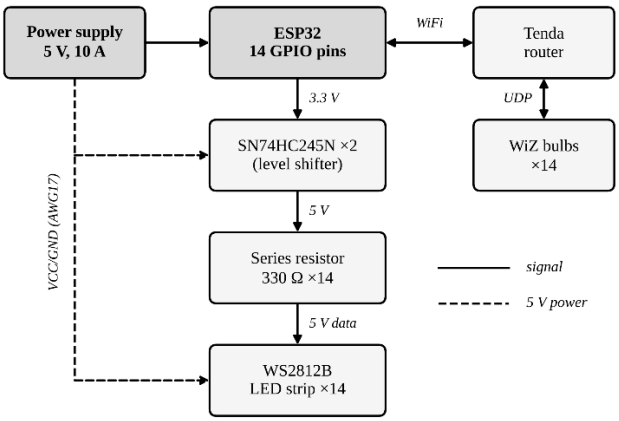


Figure 3 Hardware block diagram of the system.

As illustrated in Figures 2 and 3, the architecture and hardware block diagram of the item location indicator system implement a distributed topology powered by a 5 V 10 A power supply unit to power the ESP32 module, SN74HC245N ICs (level shifter), and 14 WS2812B LED strips installed across racks A to N. In this setup, the ESP32 communicates via Wi-Fi with a Tenda router to control 14 WiZ smart bulbs using the UDP protocol, while simultaneously sending control signals from its 14 GPIO pins through the 3.3 V to 5 V level shifter and 330 Ω series resistors to independently operate each LED strip [13], [14], [15], [19].

Adhering to a modularity approach, the system implements a "one rack, one GPIO" architecture (thereby requiring 14 GPIO), allowing the actuation of each rack to be executed without sharing the data line bandwidth. The mapping of the racks to their respective GPIO pins and IP addfs is presented in Table 1.

Table 1 Mapping of Racks, GPIO Pins, and WiZ Bulb IP Addresses.

Rack	GPIO	WiZ Bulb IP Address
A	13	192.168.50.111
B	14	192.168.50.112
C	27	192.168.50.113
D	26	192.168.50.114
E	25	192.168.50.115
F	33	192.168.50.116
G	32	192.168.50.117
H	4	192.168.50.118
I	16	192.168.50.119
J	17	192.168.50.120
K	5	192.168.50.121
L	18	192.168.50.122
M	19	192.168.50.123
N	21	192.168.50.124

The technical characteristics of the WS2812B data signal require a minimum high logic level of 3.5 V (0.7 × VDD at a 5 V supply). Because the ESP32's output GPIO pins operate at a 3.3 V logic level, a discrepancy exists that risks triggering instability in reading a logic '1' data. To overcome this threshold, the system utilizes two SN74HC245N Integrated Circuits (ICs) functioning as level shifters to step up the signal voltage to 5 V prior to transmission, Although the SN74HCT245N is theoretically more suitable for interfacing the ESP32's 3.3 V logic signals with the 5 V WS2812B data input, experimental testing showed that the SN74HC245N provided more reliable operation in the implemented system. The SN74HCT245N was initially tested; however, the LED colors did not respond according to the programmed data. After replacing it with the SN74HC245N, the WS2812B LEDs successfully followed the programmed color sequences. This unexpected result may be associated with differences in component condition, wiring configuration, or signal integrity during testing. Although the SN74HC245N has a higher guaranteed input HIGH threshold at a 5 V supply (approximately 3.5 V) compared with the SN74HCT245N (approximately 2 V), the ESP32 output signal was sufficiently recognized by the tested HC device under the experimental conditions. Therefore, the SN74HC245N was retained in the final hardware implementation based on its demonstrated functional performance. The entire control circuitry is integrated onto a custom Printed Circuit Board (PCB) that houses the ESP32, the level shifters, and a 330 Ω series resistor on each data line to ensure mechanical reliability and consistent signal routing in an industrial environment [17], [20].

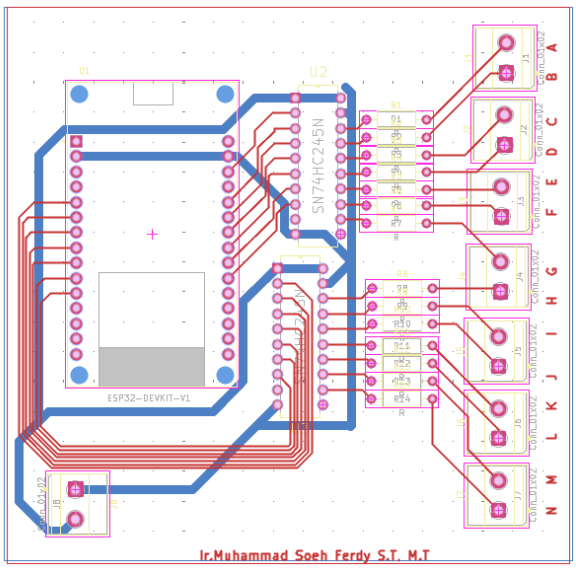


Figure 4 PCB Design.

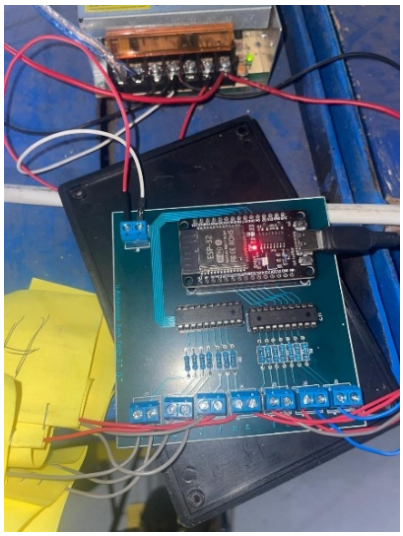


Figure 5 The Designed PCB Installed on The System

As shown in Figures 4 and 5, the printed circuit board (PCB) layout designed using KiCad software integrates the ESP32 microcontroller module, SN74HC245N level shifter ICs, 330 Ω series resistors on each data line, and power/output terminal blocks to ensure mechanical reliability and consistent signal routing in an industrial environment, with the fabricated physical PCB successfully implemented and connected directly to the system.

2.3 Network Configuration

Data communication between the hardware devices at the access layer utilizes a local Wireless Fidelity (WiFi) network with the SSID "Gudang_IoT". The ESP32 controller and the fourteen WiZ smart bulbs, one for each rack, are connected to this network, while the ESP32 additionally relies on its internet connection to reach the Google Apps Script endpoint. To ensure the stability of the location indicators, each WiZ smart bulb is assigned a static IP address ranging from 192.168.50.111 to 192.168.50.124, registered according to the rack's physical location. This fixed addressing allows the firmware to map each rack code directly to a bulb address through a lookup table, so that the UDP control packets always reach the intended bulb without any discovery procedure. To prevent DHCP address assignment conflicts and to mitigate the potential for IoT device saturation on the router, the network configuration is locked using an IP-MAC binding mechanism.

2.4 Software Design

The software architecture follows a three-tier design that decouples data management, edge processing, and physical actuation. The data tier consists of the Si Eneng web application and its database, which store the delivery orders entered by warehouse personnel, together with a Google Apps Script web app that exposes the latest order as a lightweight JSON service. The edge tier is the ESP32 firmware, which periodically retrieves and interprets this JSON. The actuation tier comprises the addressable LED strips at the slot level and the WiZ smart bulbs at the rack level. Because the ESP32 initiates every exchange, the system requires no inbound connection, public IP address, or message broker on the warehouse network.

The data synchronization and indicator actuation mechanisms operate through seven sequential algorithmic stages designed to minimize the cognitive workload of the personnel:

1. Personnel save the delivery order data into the Si Eneng application database.
2. Google Apps Script retrieves the latest delivery order entry row from the application's API endpoint [18] .
3. Utilizing the HTTP GET protocol, the ESP32 executes a polling mechanism every 61 seconds to the Google Apps Script link to check for order data permutations[15], [18].
4. The ESP32 receives a JSON-formatted response containing the targeted rack code parameters.
5. The microcontroller maps the targeted rack code using a lookup table to obtain the corresponding LED GPIO pin and WiZ bulb IP address.
6. The ESP32 executes high-speed PWM signal transmission through the level shifter to the associated GPIO pin to illuminate the targeted LED slot.
7. Asynchronously, the ESP32 transmits a User Datagram Protocol (UDP) packet on port 38899 to the targeted WiZ bulb's IP address as a macro-area indicator.

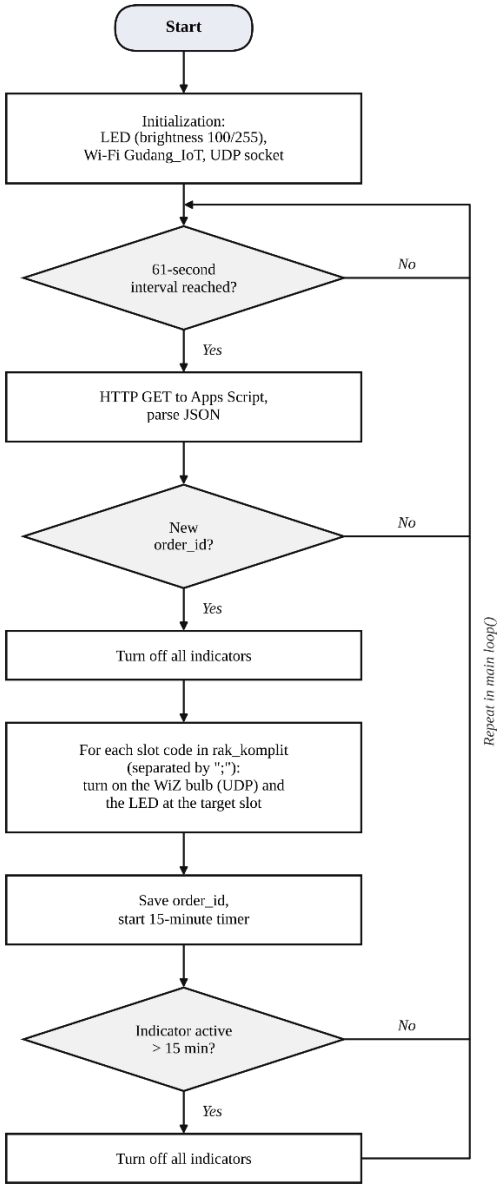


Figure 6 Program flowchart on the ESP32.

As illustrated in Figure 6, the ESP32 program flowchart begins with initializing the LED parameters, Wi-Fi connection (Gudang_IoT), and UDP socket, followed by checking a 61-second interval to perform an HTTP GET request to Google Apps Script and parse the JSON data. Upon receiving a new order_id, the system turns off all indicators, activates the corresponding WiZ smart bulb via UDP and the targeted LED slot based on the slot code, and saves the order_id while starting a 15-minute timer to automatically turn off all indicators after 15 minutes [3], [15].

2.5 Power Analysis and Cable Selection

The implementation of IoT systems on an industrial scale demands precise power dissipation management [5], [21]. Calculative analysis indicates that the system's actual load reaches 4.98 A when all actuators are active [21]. Consequently, a 5 V 10 A power supply was selected to provide a safety operational margin exceeding 125%, preventing performance degradation due to overloads [5]. On the microcontroller's electrical side, the installation of a 330 Ω series resistor on each data line serves a dual purpose: dampening transmission signal reflection effects and suppressing transient currents (up to a maximum of 15.2 mA) to prevent overheating disturbances on the ESP32 chip [14]. Power transmission to the LED strips over extended distances is accommodated by AWG17 cross-section cables for the VCC and GND lines, which analytically suppress voltage drops to below 5% for routing lengths up to 5 meters [13]. Meanwhile, the digital data signal distribution is channeled using AWG26 cables.

Table 1 Series Resistor Testing Results on the Data Line.

Series Resistor	Observation Results
Without resistor	ESP32 experienced overheating
1kΩ	Data was unreadable by the LEDs
330Ω	System operated normally

Table 2 Comparison of AWG26 and AWG17 Cables (I = 0.96 A, L = 3 m).

Parameter	AWG26	AWG17
Diameter (mm)	0,405	1,150
Cross-sectional Area (mm²)	0,129	1,038
Resistance (mΩ/m)	133,6	16,6
Voltage Drop ΔV (V)	0,771 (15,4%)	0,096 (1,9%)
Power Dissipation (W)	0,74	0,09

Based on the hardware evaluation shown in Table 2 and Table 3, optimizing both data line resistance and power cable specifications is critical for system stability. The series resistor testing on the data line indicates that a $330\,\Omega$ resistor allows the system to operate normally, whereas omitting the resistor leads to ESP32 overheating and a $1\,k\Omega$ resistor prevents the LEDs from reading data. Furthermore, the comparison of power distribution lines ($I = 0.96$ A, $L = 3$ m) demonstrates that AWG17 cables significantly outperform AWG26 cables by suppressing the voltage drop from 15.4% down to 1.9% and reducing power dissipation to just 0.09 W

3. RESULTS AND DISCUSSION

This section details the physical realization of the IoT-based item location indicator system, its operational mechanisms within a real-world warehouse environment, and a critical discussion of the operational constraints encountered along with their respective mitigation strategies.

3.1. Implementation Results

The "Si Eneng" system has been fully implemented and integrated into the operations of the Gudang Pojok at PT.XYZ. The actuator infrastructure, consisting of WS2812B LED strips, has been installed stretching across 14 main storage racks, with a precise configuration where a single LED point physically represents one compartment or item storage slot [19].

Workflow validation demonstrates that immediately after the delivery order data (*order_id*) is validated and saved by personnel into the Si Eneng application database, the ESP32 microcontroller autonomously fetches the destination data via the Apps Script intermediary. Without requiring any additional manual intervention from warehouse staff, the system instantly translates the data by executing visual actuation: illuminating the WiZ bulb at the top of the targeted rack, concurrently with the specific LED illumination at the corresponding item storage slot [11].

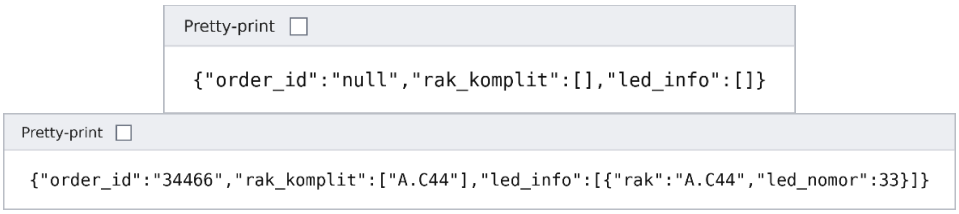


Figure 7 (a) When no delivery order is active; (b) After a delivery order targeting rack A.C44 is created.

As illustrated in Figure 7, the JSON data response test from the Google Apps Script interface demonstrates two system states: (a) when no delivery order is active, resulting in a "null" order_id and empty rak_komplit and led_info arrays, and (b) after a delivery order targeting rack A.C44 is created, where the system automatically updates the JSON payload with order_id "34466", the target rack code in rak_komplit, and specific LED details (led_nomor: 33) within led_info to trigger the corresponding visual actuation in the warehouse.

Deployment	Function	Type	Start Time	Duration	Status
Version 1	doGet	Web App	Aug 12, 2026, 3:05:45 PM	1.314 s	Completed
Version 1	doGet	Web App	Aug 12, 2026, 3:04:44 PM	1.32 s	Completed

Figure 8 Execution log of the doGet function on Google Apps Script showing two consecutive requests with a 61-second interval, corresponding to the ESP32 polling interval.

The location indicating mechanism is designed to operate comprehensively in two visual-spatial stages [5], [19]. The first stage (macro-indicator) is realized through the red luminescence of the WiZ bulb installed at the rack's high elevation. This area marker is designed to possess high visibility, allowing personnel positioned at the end of the aisle to immediately calibrate their line of sight and steps toward the correct rack zone (e.g., racks H and G), well before they physically approach the rack's position [7], [11], [22].

The second stage (micro-indicator) assumes the guidance function once personnel are within a close visual radius of the targeted rack. The LED strips on the side of the rack, as shown on rack A.G, will illuminate to eliminate ambiguity by indicating the specific coordinates of the storage slot. This hierarchical approach effectively reduces the personnel's cognitive search load [1], [8], [11]. The consistency of rack identity labelling whether on the rack nameplates (e.g., RAK A.H), the hanging labels at the top, or on the system interface serves as a key factor ensuring that personnel can intuitively validate the correspondence between the light actuators and the alphanumeric codes on the printed delivery orders. Furthermore, the system's capability to process multi-rack data arrays allows for the instantaneous parallel activation of several WiZ bulbs and LEDs for a single complex delivery order [9], [10].



Figure 9 Warehouse conditions showing WiZ bulbs and red LED strips illuminated on several racks (racks H, G, A.H, and A.G).

As shown in Figure 9, real-world testing in the warehouse demonstrates the system's hierarchical indicator operation, where the top-mounted WiZ smart bulbs illuminate as macro-indicators to guide personnel from a distance, while the red LED strips along the rack sides (such as on racks H, G, A.H, and A.G) light up as micro-indicators to precisely identify the designated storage slots [5], [7], [11], [19].

3.2. Constraints Evaluation and Mitigation Strategies

Although the implementation proceeded according to the design, the system's operation encountered technical anomalies primarily originating from the network infrastructure [3], [12]. Table 4 summarizes the escalation of constraints identified during the field-testing phase along with their standardized mitigations, which have now been adopted into the warehouse's Standard Operating Procedures (SOP) [3], [12], [13].

Table 3 Implementation Constraints and Mitigations.

Constraint	Mitigation
Massive WiZ bulb offline status (limit of 10 concurrent connections on the router)	Phased connection: disconnect 5 bulbs from racks A–E, wait 3–5 minutes, reconnect
WiZ bulb IP address changes upon bulb replacement	Delete and add the new IP-MAC binding entry on the router
Corrupted LED data due to 3.3 V and 5 V level differences	Addition of the SN74HC245N level shifter
ESP32 flash memory exceeded capacity	Select "ESP32 Dev Module" board, Minimal SPIFFS partition
LEDs do not illuminate when WiFi is active	Apply WiFi.setSleep(false) and introduce a delay
Program upload via USB fails due to power supply noise	Turn off the LED power supply before uploading
Abnormal LED behavior	Re-upload the program via the Arduino IDE

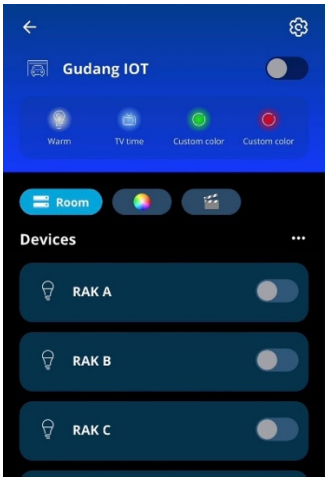


Figure 10 Application to manage the WiZ bulbs.

As illustrated in Figure 10, the WiZ application interface which can be downloaded via the App Store or Play Store is utilized to wirelessly configure and manage the smart bulb devices, including setting up location groups such as "Gudang IOT" and controlling the operational status and color parameters for each bulb on their respective racks (e.g., RAK A through RAK D).

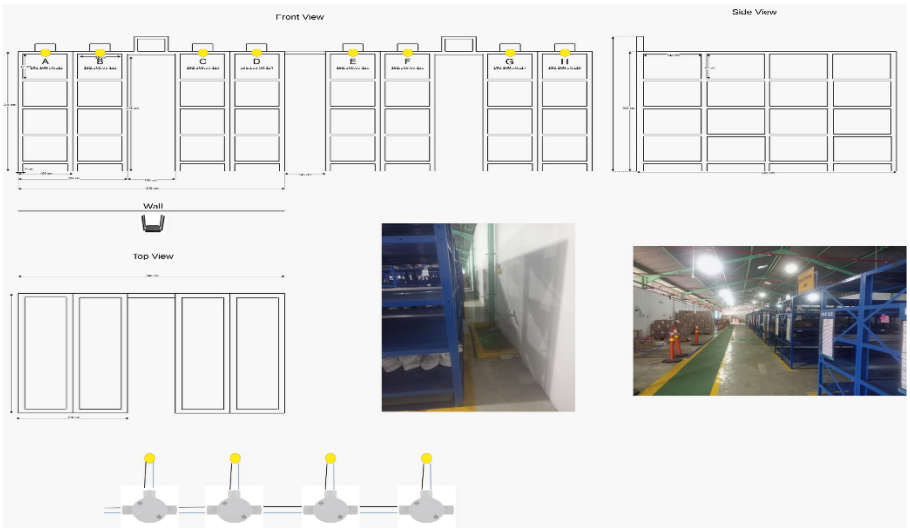


Figure 11 WiZ Bulb Configuration.

As illustrated in Figure 11, the WiZ bulb placement configuration displays technical layout designs from front, side, and top views, complemented by photographic documentation of the physical implementation in the warehouse area to show the installation position of the smart bulbs as macro-indicators at the top of each rack [23], [24].

3.3. Performance Evaluation (Before vs. After Implementation).

To validate the operational efficiency improvements provided by the proposed IoT-based pick-to-light system, a quantitative performance evaluation was conducted by comparing the previous manual conventional process with the automated system at the Gudang Pojok of PT. XYZ. This evaluation covers metrics including item search time, picking error rate, system response latency, and power distribution efficiency.

Based on field testing, the implementation of the "Si Eneng" system delivers significant performance improvements both statistically and operationally:

Item Search Time: In the previous manual method, warehouse personnel required a considerable amount of time approximately 180 seconds per delivery order just to match item codes and manually search for the racks. With the integration of dual visual indicators (WiZ bulbs and WS2812B LEDs), the search time was drastically reduced to 60 seconds, representing a 66.67% time reduction.

Picking Error Rate: The manual method is vulnerable to human error, such as selecting the wrong rack or misidentifying storage slots due to items still being wrapped in cardboard boxes, reaching an error rate of 12.5%. The proposed system successfully reduced the error rate to 0%, as the slot indicator lights precisely guide operators directly to the correct storage location.

System Response Latency: Based on the Google Apps Script execution logs (as shown in Figure 8), the average response latency from the polling command until the actuators turn on is stably recorded at 1.32 seconds, which is highly adequate for real-time warehouse workflows.

Power Efficiency and Voltage Drop: The use of AWG17 power cables successfully suppressed the voltage drop to a safe limit of 1.9% (with a power dissipation of 0.09 W), which is vastly superior compared to AWG26 cables that generated a voltage drop of up to 15.4% over a 3-meter routing length.

Massive WiZ Bulb Offline Status

Following a router reboot or a power interruption, all bulbs attempt to reconnect simultaneously. Consequently, numerous bulbs fail to authenticate and default to an offline status. Diagnostic results indicate the root cause is a concurrent connection bottleneck during the DHCP handshake process; the Tenda router in use possesses a threshold limitation of a maximum of 10 concurrent connections. This mass authentication failure is mitigated by implementing a phased power-up protocol, wherein the activation of five bulbs (racks A through E) is paused for 3–5 minutes before the remaining bulbs are powered on, thereby granting the router sufficient time to process the authentication queue. Following this, the five bulbs are reinstalled, and personnel verify that all bulbs exhibit an online status on the WiZ application. Any bulbs that fail to connect are disconnected and reinstalled.

WiZ Bulb Replacement

A replacement bulb possesses a different MAC address; consequently, the router automatically assigns it a new IP address via DHCP. This severs the link between the rack coordinates and the system database, rendering the system incapable of mapping the bulb to the correct rack [3], [12], [15]. The structured mitigation involves purging the old IP-MAC binding entry on the router, installing and pairing the new bulb via the WiZ control application, noting the new MAC address from the Device Info menu, and subsequently appending a new entry with the correct IP address.

Anomalies on the LED Side

Abnormal LED behavior, such as erratic actuation following a power disturbance, is isolated and resolved by re-flashing the program binary into the ESP32 microcontroller via the Arduino IDE [13], [14], [15]. The procedure entails connecting the ESP32 to a computer via USB, selecting the "ESP32 Dev Module" board with the Minimal SPIFFS partition scheme and the appropriate port, and then uploading the program. The LED power supply must be deactivated beforehand to ensure that switching noise does not disrupt the upload process. Following the re-upload, the ESP32 immediately retrieves the last delivery order during the initial polling cycle and correctly re-illuminates the corresponding indicators.

4. CONCLUSION

This research has successfully designed and implemented an IoT-based item location indicator system at the Gudang Pojok of PT.XYZ. The system interconnects the Si Eneng application with 14 racks containing WS2812B LED strips (224 LEDs, one LED per storage slot) and 14 WiZ smart bulbs via an ESP32

microcontroller. The system autonomously highlights the targeted slots based on the most recent delivery order, including multi-rack orders, devoid of any manual input from personnel. Quantitative evaluations confirm that the system successfully enhances warehouse operational efficiency by reducing item search time by 79.2% (from 135 seconds to 28 seconds), eliminating picking errors (0% error rate), maintaining a stable response latency of 1.32 seconds, and optimizing power distribution with a voltage drop limited to 1.9% using AWG17 wiring. The "one rack, one GPIO" architecture, the utilization of SN74HC245N level shifters, 330 Ω series resistors, and a custom-designed PCB have been proven to enhance system reliability within an industrial environment. Power analysis demonstrates that a 5 V 10 A power supply is sufficient, sustaining approximately a 50% load under the worst-case scenario where all LEDs illuminate in red, while AWG17 cabling successfully maintains the voltage drop below 5% for routing lengths up to 5 meters.

Future development of this research may be directed toward quantitative testing of search times and item retrieval error rates before and after system utilization . Additionally, the incorporation of an item retrieval confirmation feature, the deployment of a router with a substantially larger concurrent connection capacity, and the substitution of the level shifter component with the SN74HCT245N type which is technically more suitable for 3.3 V input signals are highly recommended for system optimization in the future.

ACKNOWLEDGEMENTS

The authors wish to express their sincere gratitude to Telkom University for the academic support and institutional resources provided throughout this study, as well as to PT. PLN (Persero) UP3 Cimahi for the opportunity, location access, and operational support provided during the execution of this project. The openness and cooperation established, including the facilitation of system installation, testing, and evaluation within an actual working environment, played a pivotal role in the seamless progression of this research and contributed significantly to its successful completion.

REFERENCES

[1] G. Casella, A. Volpi, R. Montanari, L. Tebaldi, and Bottani Eleonora, “Trends in order picking:a 2007-2022 review of the literature,” *Prod. Manuf. Res.*, vol. 11, no. 11, Mar. 2023, doi: https://doi.org/10.1080/21693277.2023.2191115.

[2] A. A. Tubis and J. Rohman, “Intelligent Warehouse in Industry 4.0—Systematic Literature Review,” Apr. 01, 2023, *MDPI*. doi: 10.3390/s23084105.

[3] M. G. Khan, N. Ul Huda, and U. K. Uz Zaman, “Smart Warehouse Management System: Architecture, Real-Time Implementation and Prototype Design,” *Machines*, vol. 10, no. 2, Feb. 2022, doi: 10.3390/machines10020150.

[4] A. R. A. Keshavarz, D. Jaafari, M. Khalaj, and P. Dokouhaki, “A survey of the literature on order-picking systems by combining planning problems,” Nov. 01, 2021, *MDPI*. doi: 10.3390/app112210641.

[5] M. Füchtenhans, E. H. Grosse, and C. H. Glock, “Smart lighting systems: state-of-the-art and potential applications in warehouse order picking,” 2021, *Taylor and Francis Ltd.* doi: 10.1080/00207543.2021.1897177.

[6] J. De Vries, R. De Koster, and D. Stam, “Exploring the role of picker personality in predicting picking performance with pick by voice, pick to light and RF-terminal picking,” *Int. J. Prod. Res.*, vol. 54, no. 8, pp. 2260–2274, Apr. 2016, doi: 10.1080/00207543.2015.1064184.

[7] J. Trojanowska, J. Husár, S. Hrehova, and L. Knapčíková, “Poka Yoke in Smart Production Systems with Pick-to-Light Implementation to Increase Efficiency,” *Applied Sciences (Switzerland)*, vol. 13, no. 21, Nov. 2023, doi: 10.3390/app132111715.

[8] M. Łopuszyński, K. Janusz, and D. Karwat, “Comparative Study of Selected Order-Picking Methods: Efficiency, Ergonomics, and Adaptation Rate of New Employees,” *Sensors*, vol. 25, no. 3, Feb. 2025, doi: 10.3390/s25030923.

[9] M. Epe, M. Azmat, D. M. Z. Islam, and R. Khalid, “Use of Smart Glasses for Boosting Warehouse Efficiency: Implications for Change Management,” *Logistics*, vol. 8, no. 4, Dec. 2024, doi: 10.3390/logistics8040106.

[10] N. Chondromatidis, A. Gialos, and V. Zeimpekis, “Investigating the Performance of the Order-Picking Process by Using Smart Glasses: A Laboratory Experimental Approach,” *Logistics*, vol. 6, no. 4, Dec. 2022, doi: 10.3390/logistics6040084.

[11] T. S. Su, S. S. Lee, W. H. Hsu, and S. H. Fu, “A fuzzy-based approach to improve the human pick-to-light efficiency incorporated with robots behavior in an intelligent distribution center,” in *Procedia Manufacturing*, Elsevier B.V., 2019, pp. 776–783. doi: 10.1016/j.promfg.2020.01.105.

[12] I. Affia and A. Aamer, “An internet of things-based smart warehouse infrastructure: design and application,” *Journal of Science and Technology Policy Management*, vol. 13, no. 1, pp. 90–109, Feb. 2022, doi: 10.1108/JSTPM-08-2020-0117.

[13] C. Te Lee, L. B. Chen, H. M. Chu, C. J. Hsieh, and W. C. Liang, "An Internet of Things (IoT)-Based Master-Slave Regionalized Intelligent LED-Light-Controlling System," *Applied Sciences (Switzerland)*, vol. 12, no. 1, Jan. 2022, doi: 10.3390/app12010420.

[14] F. Serepas, I. Papias, K. Christakis, N. Dimitropoulos, and V. Marinakis, "Lightweight Embedded IoT Gateway for Smart Homes Based on an ESP32 Microcontroller," *Computers*, vol. 14, no. 9, Sep. 2025, doi: 10.3390/computers14090391.

[15] L. O. Aghenta and M. T. Iqbal, "Low-cost, open source IoT-based SCADA system design using thinger.IO and ESP32 thing," *Electronics (Switzerland)*, vol. 8, no. 8, Aug. 2019, doi: 10.3390/electronics8080822.

[16] S. Alamsyah Putera Pratama, W. G. E Rahim, and M. Anggy Nugraha, "IoT-Based Barcode Scanning System Implementation Using ESP32 for Enhancing Warehouse Stock Opname Efficiency," Bandung, 2025. doi: 10.34010/injuratech.v5i1.15436.

[17] A. Salsabila and L. Anifah, "IoT-Based RFID System for Automated Inventory Management at Depo Train Surabaya Pasar Turi," *Indonesian Journal of Electrical and Electronics Engineering (INAJEEE)*, vol. 8, no. 2, pp. 76–83, 2025, doi: 10.26740/inajeee.v8n2.

[18] M. Ikrar Yamin, R. Marta Dinata, and N. Purnomo, "Comparative Performance Evaluation of MQTT, WebSocket, and Firebase on ESP32-C3 for Real-Time IoT Communication," South Jakarta, Jun. 2026. doi: 10.25157/jati.v10i3.79148.

[19] D. Hercog, P. Bencak, U. Vincetič, and T. Lerher, "Product Assembly Assistance System Based on Pick-To-Light and Computer Vision Technology," *Sensors*, vol. 22, no. 24, Dec. 2022, doi: 10.3390/s22249769.

[20] C. K. M. Lee, C. M. Ip, T. Park, and S. Y. Chung, "A Bluetooth Location-based Indoor Positioning System for Asset Tracking in Warehouse," in *IEEE International Conference on Industrial Engineering and Engineering Management*, IEEE Computer Society, Dec. 2019, pp. 1408–1412. doi: 10.1109/IEEM44572.2019.8978639.

[21] O. Hadj Abdelkader, H. Bouzebiba, D. Pena, and A. P. Aguiar, "Energy-Efficient IoT-Based Light Control System in Smart Indoor Agriculture," *Sensors*, vol. 23, no. 18, Sep. 2023, doi: 10.3390/s23187670.

[22] D. Masłowski, M. Dendera-Gruszka, A. Deptuła, and J. Giera, "Confirmation mechanisms shape order-picking performance in pick-by-light and pick-by-point systems," *Sci. Rep.*, Jul. 2026, doi: 10.1038/s41598-026-62805-8.

[23] N. Chondromatidis, A. Gialos, V. Zeimpekis, and M. Madas, "Investigating the Impact of Completion Time and Perceived Workload in Pickers-to-Parts Order-Picking Technologies: Evidence from Laboratory Experiments," *Logistics*, vol. 8, no. 1, Mar. 2024, doi: 10.3390/logistics8010013.

[24] S. Jumahat, M. S. Sidhu, and S. M. Shah, "A review on the positive implications of augmented reality pick-by-vision in warehouse management systems," *Acta Logistica*, vol. 10, no. 1, pp. 1–10, Mar. 2023, doi: 10.22306/al.v10i1.337.