IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

Smart Parcel Receiving System: A Review of IoT-Based Secure Unattended Delivery

A. Yashwitha[1], Amrutha K[2], Ashwini N. R.[3], K. Bindu[4], Jayasubha J.[5]

Final-year students, Department of Computer Science and Engineering (Batch A7), K.S. School of Engineering and Management, Bengaluru, Karnataka, India[1,2,3,4]

Associate Professor (Guide), Department of Computer Science and Engineering, K.S. School of Engineering and Management, Bengaluru, Karnataka, India[5]

Abstract: This structured narrative review examines IoT-based parcel receiving boxes and lockers, focusing on platforms, parcel detection, access authentication, notifications, and evaluation. Thirteen sources are discussed and eight compared. One study reports measured timings and another uses simulation, limiting cross-system comparison. The review identifies gaps in quantitative evaluation, order-system integration, credential separation, connectivity, and security analysis. The related project has implemented its software layer; physical hardware integration and performance evaluation remain planned.

Keywords: Internet of Things, Last-mile delivery, One-time password, Smart parcel box, Unattended delivery.

I. INTRODUCTION

Online retail depends on a last step that is still physical: a courier must hand a parcel to someone or leave it somewhere. When the recipient is absent, the courier either returns later or leaves the parcel unattended. Several of the studies reviewed here start from this observation. Soemphol et al. link the problem to the growth of online shopping during the COVID-19 pandemic [1], Mokhsin et al. describe rising parcel volumes in Malaysia [2], Zainuddin et al. frame their work around unattended delivery, parcel loss and failed delivery attempts [6], and Cacapit et al. note that a parcel left at the doorstep is not secure [7].

The project that motivates this review began from informal interactions with online shoppers and delivery personnel on 1 April 2026. These pointed to five recurring problems: missed deliveries, theft or damage of parcels left outside, repeated delivery attempts, reliance on neighbors, and the absence of reliable proof of delivery. These are anecdotal observations by the project team, not a systematic survey, and are used only to motivate the scope of the review.

Problem statement and objectives: Parcels delivered to absent recipients are exposed to missed deliveries, repeat attempts and theft. An unattended box can reduce these risks only if it detects the parcel, controls access and informs the recipient reliably. This review addresses four questions: (1) which hardware and software architectures have been used for IoT parcel receiving boxes; (2) how these systems authenticate the courier and the recipient; (3) how they have been evaluated and what evidence of performance exists; and (4) what open problems remain and where an order-integrated design fits.

Scope and contributions: The scope is deliberately narrow: single-box or small-locker systems for homes and small buildings, built with low-cost microcontrollers. Large commercial locker networks, drone delivery and logistics optimisation are outside the scope. The contributions are a structured comparison of the reviewed prototypes (Section IV-A), a set of evidence-based research gaps (Section IV-B), and a transparent mapping of an ongoing project onto those gaps that separates implemented components from planned ones (Sections III-D and IV-C). Section II surveys the literature, Section III describes the review method and the project approach, Section IV presents the results and discussion together with limitations and future scope, and Section V concludes.

II. LITERATURE SURVEY

A. Technical Background

Most parcel receiving boxes follow a three-layer IoT structure. A perception layer holds the sensors that detect parcel presence or door state. A network layer carries data from the microcontroller to a cloud service or backend, typically over Wi-Fi. An application layer provides the mobile or web interface through which the recipient is notified. An actuator, usually an electronic lock such as a solenoid, is driven by the microcontroller. The reviewed designs use four platform families: the ESP8266-based NodeMCU [2], the ESP32 and its camera variant ESP32-CAM [1], [5], the Raspberry Pi [4], and Arduino boards [8]. The ESP8266 and ESP32 integrate Wi-Fi on the chip. A Raspberry Pi runs a full operating system, which suits camera processing but is generally heavier and more power-hungry. Arduino boards are simple to program but usually need an add-on module for network access. These are general characteristics; none of the reviewed papers is known to the authors to compare platforms directly.

 This work is licensed under a Creative Commons Attribution 4.0 International License

Detection approaches reported in the reviewed sources include infrared sensing [2], proximity sensing [1], load-cell or weight sensing [5], image sensing [6], and ultrasonic ranging in a non-peer-reviewed project report [9]. A one-time password (OTP) is a credential valid for a single use or short time window. A QR code or barcode printed on a shipping label encodes an identifier such as a tracking number; it is a lookup key rather than a secret, because anyone handling the parcel can see it. Two-step authentication combines two checks, for example a label scan followed by a second verification.

B. Platform Choices and System Architectures

ParcelRestBox uses a NodeMCU V3 ESP8266, an infrared sensor and an Android application to detect each delivery and alert the household [2]. Soemphol et al. build a solar-powered IoT box that sends parcel status alerts and images through LINE Notify, motivated by the cost of SMS and email alerts [1]; according to the project's review summary it uses an ESP32-CAM, proximity sensors and a keypad lock. Valli et al. are summarised as using an ESP32-CAM, a load cell, a solenoid lock and cloud alerts [5]. Teoh et al. are summarised as using a Raspberry Pi with a mobile application and two-step authentication in ResiBox [4], and Cheng and Mohd describe an Arduino design with an RFID reader, a 4×4 membrane keypad, indicator LEDs, a relay and a solenoid lock [8].

The platform trend is toward Wi-Fi-capable microcontrollers for low-cost boxes, with single-board computers used where camera handling is central. Because none of these works benchmarks one platform against another, the choice appears to reflect cost, familiarity and camera requirements rather than demonstrated superiority. Four further sources identified at abstract or catalogue level only (‡) show that the Arduino-based pattern is common in student and regional work: a Blynk-based parcel box evaluated by a function test and a user survey [10], an RFID-card-unlocked smart box [11], the earlier ADDSMART smart mailbox [12], and an Arduino Uno mailbox that emails the user when a parcel barcode is scanned and names a locking mechanism only as future work [13]. Because only abstracts or catalogue records were consulted, these are not used for the comparative claims in Section IV.

C. Parcel Detection and Delivery Evidence

Detection is the point at which a physical event becomes a digital record, so its reliability determines the credibility of the notification. Soemphol et al. tested four parcel sizes and report 100% detection across all of them, with mean detection-and-notification times between about 8.4 s and 10.1 s depending on size [1]. Because this is one prototype tested with four parcel types, the result shows feasibility, not general reliability across shapes, materials or placements. Zainuddin et al. propose a parcel safe box combining weight and load sensors, image sensors and light sensors, and evaluate it in the MARS simulator, reporting a significant improvement in delivery performance [6]; this is a modelled estimate, not a measurement of a physical box. Cacapit et al. keep an image of the parcel in the box so that actual delivery can be evidenced in later disputes [7]. A non-peer-reviewed project report, SmartDrop, uses an ultrasonic sensor to confirm placement and requires the courier to upload a photo of the parcel in the box [9]; as an unreviewed web source it indicates a design direction, not validated evidence. Delivery evidence is increasingly image-based, but none of the available texts reports how image quality or tampering is handled.

D. Access Authentication

Access control is where the reviewed designs diverge most. DroParcel opens the box after scanning the QR code or barcode that carries the tracking number printed on the shipping label [3]. ResiBox is summarised as combining barcode verification with a second step, followed by camera recording and automatic locking [4]. Cheng and Mohd use an RFID tag per authorised user together with a keypad, which their abstract describes as dual-factor authentication [8]; the full text also mentions a choice between RFID and PIN entry, so whether the two are combined or offered as alternatives should be confirmed by the authors against the full text. Cacapit et al. include an OTP request when a courier wishes to open the box [7], and SmartDrop combines OTP authentication with face verification [9]. ParcelRestBox, as described in its abstract, focuses on detection and notification and does not describe an authentication scheme there [2].

Two observations follow. First, label-scanning schemes [3], [4] tie authorisation to an identifier visible to people who handle the parcel; whether this is sufficient depends on a threat model that the available texts do not discuss. This is the authors' interpretation, not a finding of those papers. Second, OTP-based schemes [7], [9] bind access to a short-lived secret, but the available descriptions do not say who generates the OTP, how it reaches the courier, or whether the recipient's own credential is exposed to the courier. Static RFID tags and PINs [8] are reusable credentials, so resistance to replay or cloning requires analysis rather than assumption.

E. Notification, Energy and Evaluation Practice

Soemphol et al. send alerts through LINE Notify [1]. ParcelRestBox sends notifications to an Android application [2], while Valli et al. are summarised as using cloud alerts [5]. These approaches depend on connectivity; ResiBox is summarised as internet-dependent [4]. The project review slides also describe the system of [1] as internet-dependent, although that detail is not stated in its abstract. Soemphol et al. use solar power for outdoor operation, but the abstract gives no power-consumption figures, so energy performance cannot be compared quantitatively [1].

 This work is licensed under a Creative Commons Attribution 4.0 International License

Evaluation methods are heterogeneous. One study uses timed trials across four parcel sizes [1]; one uses simulation [6]; DroParcel reports that its findings show the system to be easy to use and robust, but the available record does not show the underlying measurements [3]; ParcelRestBox is presented as a prototype design and development effort [2]. The remaining sources describe prototype construction without evaluation figures in the text available to the authors. This heterogeneity is the main reason a numerical comparison across systems is not possible.

III. METHODOLOGY / APPROACH

A. Review Method and Limits

This paper uses a structured narrative review rather than a systematic review. The available project materials indicate that sources were located through web searches for parcel box, parcel locker, and unattended delivery systems, supplemented by references listed in the project review presentations. The materials available for this revision do not record the search engines or databases, exact search strings, search dates, inclusion and exclusion criteria, or counts of records screened and retained. These details are therefore not reported, and the review should not be interpreted as exhaustive. The selection process cannot be independently reproduced from the records currently available.

Four further sources [10]–[13] were identified at abstract or catalogue level only and are not included in Table II. Source [9] is a non-peer-reviewed project page and is used only as an indication of design direction.

B. Proposed System Architecture

The project, titled Smart Parcel Receiving System, aims at secure unattended delivery through OTP-based locker access, camera-based delivery proof and real-time notification. The documented design is as follows. A customer places an order in an e-commerce application, and the backend generates an order-specific collection OTP that is shared with the customer's Smart Parcel app. A delivery person selects an assigned order in a delivery application, and the backend authorises access through a temporary delivery access code that does not expose the customer's OTP. The box is intended to detect the parcel, lock, and trigger a delivery confirmation to the customer.

C. Hardware and Software Requirements

Planned hardware: an ESP32 controller, electronic lock, keypad or touchscreen, camera, weight or infrared sensor, and an optional solar panel and battery. Software components reported as implemented: an e-commerce application (React) with customer, delivery and admin modules, a backend with a SQLite database for orders, parcels and delivery data, and the customer Smart Parcel app and delivery application.

D. Implementation Status

The project's third review reports a software-first implementation, with hardware integration deferred. Table I separates what is implemented from what is planned, so that no planned feature is read as an achieved result.

TABLE I STATUS OF PROJECT COMPONENTS

Component	Status	Relation to review
E-commerce application (React), customer/delivery/admin modules	Implemented (reported by team)	Addresses order-system integration gap (G2) at software level
Backend and SQLite database for orders, parcels and delivery data	Implemented (reported)	Supports order-system integration (G2)
Order-specific collection OTP generated by backend and shared with customer app	Implemented (reported)	OTP concept relates to [7], [9]; order-bound generation is a design response to G2
Temporary delivery access code separate from customer OTP	Implemented at software level; box communication not tested	Design response to credential separation (G3)
Notifications and status display in customer app	Implemented in app; box-triggered notifications depend on hardware	Relates to notification approaches in [1], [2]
ESP32 controller, electronic lock, keypad and parcel sensor	Planned	Hardware integration remains future work
Camera-based delivery proof	Planned	Relates to proof-of-delivery discussion [1], [7], [9]
Hardware–software integration and end-to-end testing	Planned	Required to evaluate reliability, connectivity and security

 This work is licensed under a Creative Commons Attribution 4.0 International License

E. Prototype Snapshots

Figs. 1 and 2 show the front panel and the interior of the prototype enclosure.



Fig. 1 Prototype enclosure: front panel with 4×4 keypad, LCD display, status LED and access door.



Fig. 2 Prototype enclosure interior: servo-actuated latch, IR proximity sensor, relay module and wiring.

IV. RESULTS AND DISCUSSION

No performance results of the authors' project are reported, because the physical box has not been built or measured. The results below are the findings of the review.

A. Comparative Analysis

Table II summarises what the sources report. Three patterns emerge. First, the systems mostly use low-cost microcontrollers, although [4] uses a Raspberry Pi; none of the sources reports a bill of materials, so cost claims cannot be compared. Second, a notification channel is reported for seven of the eight entries, whereas authentication is reported for five. Third, quantitative results are present for [1] only, and [6] relies on simulation. The table therefore cannot support a numerical ranking of systems.

TABLE II COMPARISON OF REVIEWED PARCEL RECEIVING SYSTEMS

Ref.	Platform/sensing	Authentication	Notification	Evaluation	Reported findings	Reported limitations
[1] Soemphol et al., 2026	ESP32-CAM; proximity sensors; solar power	Keypad lock (reported in review slides)	LINE Notify with images	Timed trials; four parcel sizes	100% detection; mean detection/notification 8.4–10.1 s	Placement, image clarity and connectivity limitations noted
[2] Mokhsin et al., 2021	NodeMCU V3 ESP8266; infrared sensor	Not reported	Android app; Firebase keyword	Prototype design/development	Delivery detection and app alert described	Not reported in available text
[3] Alghfeli et al., 2022	Smart parcel box and mobile app	QR/barcode scan of label tracking number	Mobile app	Not reported	Described as easy to use and robust	Underlying measurements not available
[4] Teoh et al., 2025	Raspberry Pi; sensors; camera	Barcode verification plus second step	Mobile app	Not reported	Secure unattended delivery/monitoring described	Details rely partly on project review slides
[5] Valli et al., 2025	ESP32-CAM; load cell; solenoid lock	Not reported	Cloud alerts (reported in review slides)	Not reported	Real-time monitoring and theft prevention described	Full text not accessed for this review

 This work is licensed under a Creative Commons Attribution 4.0 International License

Ref.	Platform/sensing	Authentication	Notification	Evaluation	Reported findings	Reported limitations
[6] Zainuddin et al., 2023	Weight/load, image and light sensors	Not reported	Alert system	MARS simulation	Improved delivery performance reported in simulation	Simulation only
[7] Cacapit et al., 2023	IoT box with parcel-presence detection	OTP request to open box	Mobile app	Not reported	Parcel image retained for disputes	Not reported in available text
[8] Cheng and Mohd, 2026	Arduino; RC522 RFID; keypad; relay; solenoid lock	RFID tag and keypad (details require full-text interpretation)	Not reported	Not reported	Low-cost/scalable design claimed by authors	Not reported in available text

B. Research Gaps and Open Challenges

The following gaps apply to the sources reviewed here, a small set that is not a census of the field. Each is labelled as established within this set or as an opportunity.

G1. Limited quantitative evaluation (established within this set): Among sources with accessible text, only [1] reports measured detection and notification performance, and [6] relies on simulation. False-detection rate, lock-actuation reliability, end-to-end latency under varying network conditions and power consumption over time are not reported across the set.

G2. Weak coupling to the order system (established within this set): The reviewed designs centre on the box and a companion app. DroParcel ties access to the tracking number on the label [3]; the other available descriptions do not describe credentials generated by the shopping platform for each order.

G3. Courier and recipient credential separation (opportunity): The OTP-based descriptions [7], [9] do not state whether the courier receives a credential distinct from the recipient's collection credential. This is a design consideration inferred from the literature, not a gap that any reviewed paper demonstrates.

G4. Connectivity and scalability (established within this set, as summarised): ResiBox is summarised as dependent on the internet, limited in capacity and difficult to scale. Offline or degraded-network operation is not reported in the available texts.

G5. Security analysis (opportunity): The available texts do not report threat models, attack testing, or analysis of label-based credential visibility, although several designs unlock a physical container.

C. Relation of the Project to the Gaps

The project partially addresses G2 and G3 at the level of software design (Table I) and leaves G1, G4 and G5 for future work, because they require the physical box and its evaluation. It does not yet provide evidence that G2 and G3 are solved, because the physical box, the lock actuation path and the end-to-end flow have not been integrated or measured. The appropriate claim is therefore that the project is a proposed design direction that responds to gaps identified in the literature; whether it improves security, reliability or user experience will require the evaluation outlined below.

D. Limitations and Future Scope

Technical constraints: An unattended outdoor box faces weather, power and connectivity limits. Soemphol et al. address power through solar operation [1]; the project lists solar and battery power as optional. Wi-Fi coverage may be unreliable, which makes fallback behaviour (for example a locally validated, time-limited code) a design question the reviewed literature does not resolve.

Physical and capacity limits: ResiBox is summarised as having limited capacity [4]. A single compartment cannot handle several simultaneous deliveries, whereas multi-compartment lockers add cost and wiring. The project's present plan concerns a single smart box.

Security and privacy: A locking device exposed to the street needs analysis of credential guessing, replay of an observed code, tampering with the enclosure and lock failure under power loss; none of these is evaluated in the reviewed texts. Cameras that record couriers or passers-by raise privacy questions; the reviewed sources emphasise the benefit of image proof [1] and do not discuss retention, access or consent.

Future scope: (i) Complete the ESP32 integration with lock, keypad and sensors; (ii) measure unlock latency, detection accuracy across parcel sizes (comparable with the protocol of [1]), notification delay and power consumption;

 This work is licensed under a Creative Commons Attribution 4.0 International License

(iii) test behaviour with degraded connectivity; (iv) perform a structured security review of the OTP and delivery-access flow; and (v) study usability with real delivery staff and residents. These are proposals and are not completed work.

F. Future Enhancements

The following enhancements are proposed extensions of the project. None is implemented or evaluated, and each is intended to address one of the gaps G1–G5 or a limitation discussed above.

Hardware realisation*:* integrate the ESP32 controller with the electronic lock, keypad or touchscreen and parcel sensor (weight or infrared), and connect the lock-actuation path to the backend so that a valid code opens the box and a closed door triggers the delivery confirmation (G1).

Camera-based delivery proof: capture an image at the moment of delivery and attach it to the order record as proof of delivery, with defined retention periods, access rules and signage to address the privacy concerns noted in Section IV-D.

Offline and degraded-network operation: allow the box to validate a locally stored, time-limited code when Wi-Fi is unavailable, and queue notifications for delivery once connectivity returns (G4).

Security hardening: add code expiry, attempt limits and lockout, protected transport between the apps, backend and box, tamper detection on the enclosure and a defined fail-secure or fail-safe behaviour on power loss, followed by a structured threat-model-based review (G3, G5).

Power management: add battery backup with low-power sleep between events, and optional solar charging as in [1], with measured power consumption over time (G1).

Capacity and scalability: extend the single compartment to a multi-compartment locker with per-compartment locks and sensors, so that several orders can be received at once.

Notifications and integration: add push, SMS or messaging-app alerts alongside in-app notifications, and expose the order-bound OTP flow through an interface that other e-commerce or courier platforms could adopt (G2).

Evaluation and usability: measure unlock latency, detection accuracy across parcel sizes, notification delay and false-detection rate, and conduct usability studies with delivery staff and residents.

V. CONCLUSION

The reviewed IoT parcel receiving systems converge on a simple architecture of a Wi-Fi-capable microcontroller, a sensor for arrival, an electronic lock and a smartphone alert, but differ in authentication and in the strength of their evaluation. Only one reviewed study reports measured timings and a detection rate [1], and another relies on simulation [6], which limits what can be concluded about relative merit. The most defensible gaps in this set are the shortage of quantitative evaluation, the weak connection between box credentials and the order system, the unaddressed separation of courier and recipient credentials, dependence on connectivity, and the absence of security analysis. The authors' project responds to the integration and credential-separation gaps through its software design, but its hardware and performance remain to be built and measured; conclusions about its effectiveness must wait for that evaluation.

REFERENCES

[1]. C. Soemphol, C. Photong, and P. Panmuang, "A smart parcel locker system with parcel status and image notifications via LINE application," *Bull. Electr. Eng. Inform.*, vol. 15, no. 1, pp. 161–170, Feb. 2026, doi: 10.11591/eei.v15i1.10683.

[2]. M. Mokhsin, M. Z. M. Ludin, A. I. H. Suhaimi, A. S. Zainol, M. H. Mohd Som, and H. A. Halim, "ParcelRestBox: IoT-based parcel receiving box system design for smart city in Malaysia," in *Proc. 2021 IEEE Int. Conf. Comput. (ICOCO)*, 2021, pp. 180–185, doi: 10.1109/ICOCO53166.2021.9673588.

[3]. M. Alghfeli, M. Alnuaimi, N. Alsebaiha, S. Alnuaimi, B. Pradeep, and P. Kulkarni, "DroParcel: Smart system for secure parcel delivery," in *Proc. 2022 IEEE 12th Int. Conf. Consum. Electron. (ICCE-Berlin)*, Berlin, Germany, 2022, pp. 1–6, doi: 10.1109/ICCE-Berlin56473.2022.9937128.

[4]. S.-K. Teoh, K.-H. Tan, L.-Y. Tan, and J.-J. Chang, "A two-step authentication residence parcel dropbox (ResiBox) monitoring and controlling system," in *Proc. 2025 IEEE Int. Conf. Autom. Control Intell. Syst. (I2CACIS)*, Kuala Lumpur, Malaysia, 2025, pp. 185–189, doi: 10.1109/I2CACIS65476.2025.11100736.

[5]. K. Atchyuta Valli, B. Yuvaraju, G. Arun Kumar, M. Hema Girish, M. Mouli, and Y. Misra, "Smart parcel delivery receiving box for secure parcel drop-off," *Int. J. Res. Publ. Rev.*, vol. 6, no. 10, pp. 5853–5860, Oct. 2025, doi: 10.55248/gengpi.06.1025.3789.

[6]. A. Zainuddin, H. Mansor, N. I. Badrulhisham, N. N. Zulkifli, A. A. M. Ridzal, and N. Ghazalli, "Simulating the effectiveness of an IoT parcel alert system for enhancing delivery efficiency and safety during Covid-19," *Malays. J. Sci. Adv. Technol.*, vol. 3, no. 1, pp. 28–36, Mar. 2023.

 This work is licensed under a Creative Commons Attribution 4.0 International License

[7]. P. Cacapit, R. Garcia, S. Minor, G. Olan, and G. Cu, "Smart box delivery system," in *Proc. De La Salle Univ. Research Congress*, 2023, paper HCT-03. [Online]. Available: https://www.dlsu.edu.ph/wp-content/uploads/pdf/conferences/research-congress-proceedings/2023/HCT-03.pdf

[8]. J. T. C. Cheng and M. S. Mohd, "Design of parcel box using RFID technology with Arduino," *Multidiscip. Appl. Res. Innov.*, vol. 7, no. 2, pp. 49–54, 2026. [Online]. Available: https://periodical.uthm.edu.my/index.php/mari/article/view/18041

[9]. "SmartDrop: An IoT-driven secure parcel delivery and monitoring system," ElectronicWings project page (non-peer-reviewed). [Online]. Available: https://www.electronicwings.com/users/NischalNBangera/projects/6553/smartdrop---an-iot-driven-secure-parcel-delivery-and-monitoring-system

[10]. M. A. A. Mohd Noor *et al.*, "Mereka bentuk kotak parcel pintar menggunakan sistem IoT," *Prog. Eng. Appl. Technol.*, vol. 4, no. 2, pp. 648–657, Dec. 2023. [Online]. Available: https://periodical.uthm.edu.my/index.php/peat/article/view/10415

[11]. S. A. Rashid and R. Darman, "Sistem kotak pintar berasaskan IoT bagi peti pos" ["IoT-based smart box system for mailboxes"], *Appl. Inf. Technol. Comput. Sci.*, vol. 5, no. 1, pp. 691–708, 2024, doi: 10.30880/aitcs.2024.05.01.039.

[12]. J. R. Tew and L. Ray, "ADDSMART: Address digitization and smart mailbox with RFID technology," in *Proc. 2016 IEEE 7th Annu. Ubiquitous Comput., Electron. Mobile Commun. Conf. (UEMCON)*, 2016, pp. 1–6, doi: 10.1109/UEMCON.2016.7777879.

[13]. M. F. Mohamed Firdaus and K.-A. Kamarudin, "Development prototype of smart mailbox that send email to user when parcel barcode being scanned," *Res. Prog. Mech. Manuf. Eng.*, vol. 3, no. 1, pp. 839–847, 2022, doi: 10.30880/rpmme.2022.03.01.089.