

Development of an Advanced Smart Delivery Robot for Contactless and Secure Parcel Management

Santhosh Kumar T C¹, S. Manoj², Vinay. G³, Chiranthan M Gowda⁴

¹Assistant Professor, Department of Mechanical Engineering, BGSIT, ACU Mandya

^{2,3,4}Student, Department of Mechanical Engineering, BGSIT, ACU Mandya

Abstract

This document presents the design, development, and implementation of an advanced smart delivery robot engineered to facilitate contactless, secure, and efficient parcel management in controlled environments such as hospitals, university campuses, and gated communities. The system addresses critical vulnerabilities in traditional delivery logistics, specifically the lack of recipient authentication, absence of real-time monitoring, and dependency on human operators. Driven by an ESP32 WROOM-32 microcontroller, the proposed robotic architecture integrates Internet of Things (IoT) cloud communication via Firebase, real-time video streaming through an IP camera, and an ultrasonic obstacle avoidance mechanism. A ReactJS-based web dashboard is developed for remote administration, allowing for both semi-autonomous navigation and manual control. To ensure delivery security, a One-Time Password (OTP) authentication system is incorporated, mechanically coupled to an SG90 servo motor for compartment access. Experimental evaluations demonstrate a robust command latency of 350 to 600 milliseconds, an obstacle detection accuracy triggering consistently at a 30-centimeter threshold, and a 100% success rate for secure OTP verification. The resulting platform offers a scalable, cost-effective, and highly reliable solution for decentralized last-mile delivery operations.

Keywords: Smart Delivery Robot, Internet of Things, Secure Authentication, Contactless Logistics, ESP32 Microcontroller

1. Introduction

In controlled environments such as healthcare facilities, university campuses, and corporate parks, the necessity for sterile, contactless, and decentralized parcel management has grown substantially. Traditional delivery systems heavily rely on human couriers, a dependency that is increasingly associated with logistical inefficiencies, heightened operational costs, and security vulnerabilities. Unauthorized access to delivered parcels and the lack of real-time tracking present critical challenges in last-mile delivery logistics. Furthermore, the absence of verification mechanisms often results in misplacements and compromises the integrity of sensitive deliveries, such as medical supplies or confidential documents.

The rise of automation in logistics has spurred significant interest in robotics to mitigate these challenges. Autonomous delivery robots offer a transformative solution by eliminating the need for repeated human trips for short-range deliveries. However, existing commercial solutions often lack comprehensive security features or require highly expensive infrastructure to operate effectively. Therefore, the primary

motivation for this research is the integration of emerging technologies—specifically IoT, cloud computing, embedded systems, and reactive web development—into a singular, cost-effective, and practically deployable delivery robot.

The proposed system addresses the aforementioned gaps by introducing an advanced smart delivery robot equipped with an OTP-based secure access mechanism, real-time obstacle detection, and live IP camera streaming. It is hypothesized that by utilizing a dual-core ESP32 microcontroller as the central processing unit and leveraging Google Firebase for real-time data synchronization, a highly reliable and semi-autonomous delivery platform can be achieved. This system aims to ensure that only authenticated recipients can access the delivered parcels, while providing system administrators with comprehensive, real-time visibility into the robot's operational status and environmental surroundings.

2. Literature Survey

The development of automated and IoT-enabled delivery systems has been extensively explored in recent literature. Previous studies have demonstrated the efficacy of autonomous navigation; for instance, Mehmet E., et al. (2019) presented a ROS-based autonomous robot navigation system utilizing ultrasonic and infrared sensors for obstacle detection, which provides a foundational understanding for autonomous movement architectures [1]. In the domain of cloud integration, Priya N. and Meghna S. (2020) highlighted the capabilities of IoT-based smart parcel delivery systems using Firebase and ESP8266 microcontrollers, validating the use of real-time databases for parcel tracking [2].

Security remains a paramount concern in physical IoT systems. Jian long T., et al. (2021) proposed robust OTP-based authentication frameworks for IoT access control systems, a methodology directly applicable to secure parcel retrieval mechanisms [3]. Furthermore, the performance of environmental sensing has been evaluated by Arun K. and Sujith R. (2020), who confirmed the reliability of HC-SR04 sensors in mobile robot platforms, validating a threshold-based halting strategy for collision avoidance [4]. The suitability of the ESP32 as a central IoT controller for multi-sensor systems has been established by Deepak B., et al. (2020) [5], while the development of reactive user interfaces for such systems was explored by Li W. and Chen F. (2021) using React and Firebase [6]. Additional research into battery management by Nishant V., et al. (2020) [7] and wireless video streaming via IP cameras by Asha R. and Murali K. (2019) [8] further supports the architectural decisions made in the development of the proposed delivery robot. Collectively, these studies underscore the viability of integrating localized embedded processing with cloud-based administration for robust last-mile logistics [9, 10].

3. Objectives of the Research

The primary objectives of this research project are defined as follows:

- To develop a semi-autonomous delivery robot capable of providing contactless delivery with real-time video streaming and system status updates.
- To design a secure parcel retrieval mechanism utilizing a dynamic One-Time Password (OTP) authentication system.
- To implement a reliable obstacle detection and avoidance system for enhanced operational safety in shared environments.
- To construct a centralized web-based admin dashboard using ReactJS and Firebase for remote monitoring, command dispatching, and historical event logging.

4. Proposed Methodology

The development of the smart delivery robot was executed through a structured, iterative methodology divided into four distinct phases.

Phase 1 involved system design and component selection. The system architecture was defined, and hardware components were selected based on performance, cost, and availability. A 3D-printed chassis was modeled using CAD software to appropriately accommodate the differential drive motors, battery compartment, electronics bay, delivery box, and camera mount. Open-source, widely documented modules compatible with the ESP32 WROOM ecosystem were prioritized.

Phase 2 focused on hardware integration and testing. Each hardware module was independently verified prior to full integration. Motor control logic was tested using the L293D drivers; the SG90 servo was calibrated for accurate locking mechanisms; the HC-SR04 sensor was validated against physical distance measurements; and the IP camera was configured for MJPEG streaming over the local Wi-Fi network.

Phase 3 encompassed the firmware development. The ESP32 firmware was written in C/C++ using the Arduino IDE environment. The code was structured into modular functions managing Wi-Fi setup, Firebase synchronization, motor control, obstacle detection, OTP verification, and battery monitoring. The dual-core architecture of the ESP32 was leveraged to ensure the main loop could concurrently poll the database and process sensor inputs without blocking.

Phase 4 involved the web dashboard development and integration. A ReactJS-based interface was engineered with functional components and state management hooks. The Firebase SDK for JavaScript was utilized to subscribe to Realtime Database events. Finally, the system underwent rigorous end-to-end integration testing, ensuring seamless communication between the hardware node and the web interface.

5. System Design and Implementation

The system architecture operates in two primary phases: the Delivery Dispatch Phase and the Recipient Authentication Phase. The entire workflow is coordinated through a Firebase Realtime Database, which acts as the communication backbone connecting the ESP32 firmware, the IP camera stream, and the ReactJS web dashboard.

5.1 Hardware Architecture The central processing unit of the delivery robot is the ESP32 WROOM-32 microcontroller, selected for its dual-core processing capabilities and integrated Wi-Fi. The mechanical drive system comprises a 4-wheel differential drive configuration powered by DC gear motors, controlled via L293D dual H-bridge motor drivers. A comprehensive list of the selected hardware components is detailed in Table 1.

Table 1: Hardware Component Specifications

S.No	Component	Specification	Quantity	Voltage	Purpose
1	ESP32 WROOM-32	240 MHz, Wi-Fi, BT	1	3.3V	Central controller
2	L293D Motor Driver	600 mA/ch, 36V max	2	5–36V	Motor control
3	SG90 Servo Motor	180°, 9g, 2.5 kg·cm	1	5V	Door lock/unlock
4	IP Camera	720p/1080p, MJPEG	1	5V/12V	Live video stream

S.No	Component	Specification	Quantity	Voltage	Purpose
1	ESP32 WROOM-32	240 MHz, Wi-Fi, BT	1	3.3V	Central controller
5	HC-SR04 Ultrasonic	2–400 cm, ±3mm	1	5V	Obstacle detection
6	4×4 Keypad	16-key matrix	1	3.3–5V	OTP input
7	18650 3S Battery	11.1V, 6000 mAh	1 pack	11.1–12.6V	Power supply
8	DC Gear Motors	6–12V, 150 RPM	4	6–12V	Wheel drive
9	Active Buzzer	5V, 85 dB	1	5V	Audio alerts
10	Buck Converter	5V/3A output	1	Input: 12V	Voltage regulation

To provide an adequate power supply, an 18650 3S Lithium-Ion battery pack is utilized, yielding a nominal voltage of 11.1V. A DC-DC step-down buck converter is implemented to supply a stable 5V to the ESP32, sensors, and IP camera, while the motor drivers draw power directly from the 12V bus.

5.2 Software Architecture The software framework is bifurcated into the embedded firmware and the cloud-hosted web application. The ESP32 firmware is developed utilizing standard C/C++ libraries, prominently including WiFi. h, Firebase ESP32.h, Keypad. h, ESP32Servo.h, and New Ping. h. The Firebase Realtime Database utilizes a JSON tree structure to store motor direction commands, operating modes, battery status, obstacle flags, and the active OTP logs.

The admin interface is constructed using ReactJS (v18+). It features distinct functional components: a Video Stream component for embedding the IP camera MJPEG feed, a Contro lPanel for issuing directional commands, an OTP Manager for cryptographic key generation, and an Alert Log for timestamped historical tracking.

5.3 Circuit and Pin Configuration Precise pin mapping is critical for the concurrent operation of the connected peripherals. The ESP32 manages multiple Pulse Width Modulation (PWM) channels, digital input/outputs, and analog-to-digital conversions (ADC). The detailed pin configuration is illustrated in Table 2.

Table 2: ESP32 Pin Mapping and Circuit Configuration

S.No	ESP32 GPIO Pin	Connected To	Signal Type	Description
1	GPIO 14	L293D – ENA (Motor 1 Enable)	PWM Output	Speed control of Front-Left motor
2	GPIO 12	L293D – IN1 (Motor 1 Dir A)	Digital Output	Direction bit A for Front-Left
3	GPIO 13	L293D – IN2 (Motor 1 Dir B)	Digital Output	Direction bit B for Front-Left

4	GPIO 26	L293D – ENB (Motor 2 Enable)	PWM Output	Speed control of Front-Right motor
5	GPIO 27	L293D – IN3 (Motor 2 Dir A)	Digital Output	Direction bit A for Front-Right
6	GPIO 25	L293D – IN4 (Motor 2 Dir B)	Digital Output	Direction bit B for Front-Right
7	GPIO 33	L293D #2 – ENA (Motor 3 Enable)	PWM Output	Speed control of Rear-Left motor
8	GPIO 32	L293D #2 – IN1 (Motor 3 Dir A)	Digital Output	Direction bit A for Rear-Left
9	GPIO 35	L293D #2 – IN2 (Motor 3 Dir B)	Digital Output	Direction bit B for Rear-Left
10	GPIO 4	L293D #2 – ENB (Motor 4 Enable)	PWM Output	Speed control of Rear-Right motor
11	GPIO 16	L293D #2 – IN3 (Motor 4 Dir A)	Digital Output	Direction bit A for Rear-Right
12	GPIO 17	L293D #2 – IN4 (Motor 4 Dir B)	Digital Output	Direction bit B for Rear-Right
13	GPIO 18	SG90 Servo – Signal Wire	PWM Output	50 Hz PWM for door lock/unlock
14	GPIO 5	HC-SR04 – Trigger	Digital Output	10 μ s pulse to trigger ultrasonic burst
15	GPIO 19	HC-SR04 – Echo (via divider)	Digital Input	Echo pulse width \rightarrow distance calc
16	GPIO 21	Keypad – Row 1	Digital I/O	Keypad matrix row scanning
17	GPIO 22	Keypad – Row 2	Digital I/O	Keypad matrix row scanning
18	GPIO 23	Keypad – Row 3	Digital I/O	Keypad matrix row scanning
19	GPIO 15	Keypad – Row 4	Digital I/O	Keypad matrix row scanning

20	GPIO 2	Keypad – Col 1	Digital I/O	Keypad matrix column scanning
21	GPIO 0	Keypad – Col 2	Digital I/O	Keypad matrix column scanning
22	GPIO 4	Keypad – Col 3	Digital I/O	Keypad matrix column scanning
23	GPIO 34	Battery Voltage (ADC via divider)	Analog Input	Battery % monitoring (0–3.3V range)
24	GPIO 1	Buzzer (via BC547 transistor)	Digital Output	Triggers buzzer for alerts
25	3.3V	ESP32 VCC	Power	3.3V regulated supply from buck converter
26	GND	Common Ground	Ground	Shared GND for all modules

A voltage divider network is employed to safely read the 12.6V maximum battery voltage at the 3.3V-tolerant ESP32 ADC pin, utilizing a 100kΩ and 33kΩ resistor pairing. Similarly, the 5V echo pulse from the ultrasonic sensor is stepped down to 3.3V to protect the microcontroller.

6. Results and Discussion

The fully assembled smart delivery robot was subjected to rigorous functional testing across all operational modules. The performance metrics and operational results are summarized in Table 3.

Table 3: System Testing Parameters and Results Test Parameter Target Specification Achieved Result

Test Parameter	Target Specification	Achieved Result	Status
Firestore command latency	< 1 second	350–600 ms	PASS
Obstacle detection range	30 cm threshold	28–32 cm trigger	PASS
OTP verification accuracy	100%	100% (40/40 tests)	PASS
Servo unlock response time	< 2 seconds	1.2 sec avg	PASS
Video stream frame rate	> 10 FPS	15–25 FPS	PASS
Battery monitoring accuracy	±5%	±3%	PASS
Continuous operation time	> 30 min	45–60 min	PASS

False obstacle detection	< 5%	4% (2/50 runs)	PASS
--------------------------	------	----------------	------

It is observed that the motor control response exhibited a latency of 350 to 600 milliseconds. This delay is primarily attributable to the HTTPS request round-trip time between the ESP32 and the Firebase cloud servers. While this latency is highly acceptable for standard indoor delivery tasks, it highlights a reliance on local internet bandwidth.

The HC-SR04 ultrasonic sensor accurately detected obstacles, reliably halting the robot within a 2-centimeter margin of error around the 30-centimeter threshold. False positives, initially noted due to environmental ultrasonic interference, were successfully mitigated to a negligible 4% by implementing an algorithmic average of three consecutive sensor readings in the firmware.

Furthermore, the OTP authentication mechanism demonstrated flawless reliability. Across 40 independent trials, the ESP32 accurately queried the Firebase database, verified the entered digits, and actuated the SG90 servo motor with an average response time of 1.2 seconds. Incorrect entries successfully triggered the active buzzer and logged the failed attempt to the administrator dashboard, thereby validating the system's security integrity.

7. Advantages and Applications

The implemented system offers significant advantages over conventional delivery logistics. The integration of an OTP-based lock ensures verifiable, tamper-proof parcel retrieval, while the real-time Firebase synchronization provides administrators with unparalleled visibility into the robot's status. The dual-mode operation allows for autonomous routing during standard tasks and manual override capabilities during complex navigation scenarios. Furthermore, the utilization of open-source technologies significantly reduces the overall implementation cost.

The versatility of the Advanced Smart Delivery Robot makes it highly applicable across numerous domains. In healthcare environments, it can autonomously transport medicines or biological samples between wards, minimizing cross-contamination risks. In gated residential communities and university campuses, it facilitates decentralized parcel distribution, ensuring secure last-mile delivery without granting external personnel access to private premises. Additionally, it serves as a highly efficient internal logistics tool for document routing in corporate office environments.

8. Future Scope

While the current system architecture performs robustly within its specified parameters, several enhancements can be integrated into future iterations. The open-loop timer-based navigation is susceptible to positional drift over extended periods; this can be rectified by incorporating wheel encoders and an Inertial Measurement Unit (IMU) for closed-loop PID position control. Environmental awareness can be significantly upgraded by replacing the single-axis ultrasonic sensor with a multi-directional LIDAR module, enabling Simultaneous Localization and Mapping (SLAM) for fully autonomous path planning. Additionally, integrating a Time-Based One-Time Password (TOTP) algorithm and an MQTT broker fallback would further bolster system security and offline reliability.

9. Conclusion

The development of the Advanced Smart Delivery Robot successfully demonstrates the practical and eff-

ective integration of IoT communication, embedded mechatronics, and reactive web technologies. By bridging physical robotics with cloud-based administration, the project provides a comprehensive solution to the challenges inherent in last-mile delivery logistics. The implementation of strict OTP authentication, paired with real-time video monitoring and autonomous obstacle avoidance, results in a highly secure, safe, and efficient delivery platform. The system remains highly modular and scalable, offering a strong foundation for future advancements in automated campus and healthcare logistics.

Acknowledgement

The authors wish to express their gratitude to the respective departments and organizations that provided the necessary resources, laboratory facilities, and guidance throughout the development and testing phases of this research project.

References

1. Mehmet E., Derya E., “Design of an Autonomous Mobile Robot Based on ROS”, 2019 International Conference on Electronics, Computer and Computation (ICECCO), IEEE, 2019, 1-5.
2. Priya N., Meghna S., “IoT-Based Smart Parcel Delivery System Using Firebase and ESP8266”, International Journal of Engineering Research & Technology (IJERT), May 2020, 9 (5), 112-117.
3. Jianlong T., Xiaoran S., Xuan L., “Secure OTP-Based Authentication for IoT Access Control Systems”, IEEE Access, 2021, 9, 43210–43221.
4. Arun K., Sujith R., “Obstacle Avoidance in Mobile Robots Using Ultrasonic Sensing”, International Journal of Computer Applications (IJCA), 2020, 120 (11), 22-28.
5. Deepak B., Vimal U., Pranshu B., “ESP32-Based IoT Platform for Real-Time Monitoring and Control”, 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), IEEE, 2020, 1-6.
6. Li W., Chen F., “React-Based Real-Time Dashboard for IoT Systems Using Firebase”, Journal of Web Engineering, 2021, 20 (4), 1147–1178.
7. Nishant V., Ajay Y., “Battery Management and Monitoring Systems for Mobile Robots”, International Journal of Engineering Development and Research (IJEDR), 2020, 8 (2), 45-51.
8. Asha R., Murali K., “Wireless Video Streaming Using IP Camera in Surveillance Applications”, International Journal of Recent Technology and Engineering (IJRTE), 2019, 8 (3), 199-204.
9. Arulmurugan R., Anitha R., “Last-Mile Delivery Automation: A Review of Robotic Systems and Technologies”, Procedia Computer Science, Elsevier, 2022, 215, 512–520.
10. Nagarajan M., Priya S., “4-Wheel Differential Drive Robot Control via Web Interface”, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering (IJAREEIE), 2020, 9 (3), 88-94.



Licensed under [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/)