* **Module – Computer project**
* **Project name- Smart door delivery authentication system**
* **Team number – 2**
* **Team members**

**Banuka Abeysundera – Team leader  
Gihan Atupolage  
Umair Mohommeth  
Senuri Minhari**

Table of Contents

[**Smart Door Delivery Authentication System** 4](#_Toc195182320)

[1. Introduction 4](#_Toc195182321)

[2. Background 4](#_Toc195182322)

[3. Problem Statement 4](#_Toc195182323)

[4. Literature Review 5](#_Toc195182324)

[5. Proposed Solution: Smart door delivery authentication system 8](#_Toc195182325)

[5.1. Project Aim 8](#_Toc195182326)

[5.2. Project Objectives 9](#_Toc195182327)

[5.3. Methodology and Technology 11](#_Toc195182328)

[5.4. Design Overview and Justification 15](#_Toc195182329)

[5.5. Project Scope 16](#_Toc195182330)

[5.6. Limitations of the Project 19](#_Toc195182331)

[5.7. Project Risk and Contingency Plan 20](#_Toc195182332)

[6. Project Deliverables & Milestones 22](#_Toc195182333)

[7. Schedule and Budget Summary 25](#_Toc195182334)

[8. References 28](#_Toc195182335)

[9. Definitions 29](#_Toc195182336)

# **Smart Door Delivery Authentication System**

## Introduction

With the surge in online shopping, unattended doorstep parcel deliveries have become commonplace. However, they also introduce risks such as parcel theft, mis delivery, and unauthorized pickups. Traditional CCTV systems lack intelligent detection and alerting mechanisms. This project introduces an intelligent IoT-based smart parcel box with cybersecurity features to secure doorstep deliveries through access control, tamper detection, and automated real-time monitoring.

## Background

Smart homes and IoT-based automation systems are evolving rapidly. However, secure last-mile delivery is still underdeveloped, especially in terms of autonomous parcel security. This project aims to fill this gap by combining hardware (ESP32-CAM, RFID, load cells, servo locks) with software (Telegram bots, SD logging) to ensure parcels are delivered and picked up securely. The system ensures authorized access only and creates a forensic log of all events.

## Problem Statement

With the explosive growth in e-commerce, doorstep delivery has become the vulnerability — unsecured parcels left unattended at doorsteps. These packages are highly susceptible to theft, tampering, or mistaken pickup, especially in urban or shared partial mitigation. They act as passive observers, not active preventers, and often fail to capture sufficient evidence or alert the recipient in real-time.

Moreover, in many cases:

* Delivery personnel do not verify recipient identity.
* There is no guarantee that the person who picks up the parcel is the intended recipient.
* No log or trail exists to verify when, by whom, and under what circumstances the parcel was picked up.
* Victims of parcel theft typically have no recourse or clear evidence to prove malicious activity.

This gap in security is further widened by the lack of affordable, intelligent systems that can:

* Provide parcel access control.
* Authenticate parcel pickup.
* Detect theft and generate real-time alerts.
* Operate on minimal hardware and internet bandwidth.

Cybersecurity in IoT-based delivery systems is often neglected or optional, yet it is crucial. A system that merely monitors but does not enforce authentication or create a verifiable log of actions cannot be trusted in critical use cases.

Hence, there is an urgent need for a compact, intelligent, and cost-effective system that not only records but also actively secures the last-mile delivery — a system that integrates cybersecurity, authentication, physical detection, and smart alerting to protect deliveries at the doorstep.

## Literature Review

With the rise of e-commerce and contactless delivery, the challenge of securely handling unattended parcel delivery has become increasingly relevant. Several researchers and companies have explored aspects of smart home integration, parcel security, IoT-based surveillance, and cybersecurity. However, a holistic system that combines hardware-based authentication, tamper detection, and secure offline and online logging is still lacking.

1. IoT-Based Smart Lockers for Delivery Authentication

* According to Singh & Rani (2020), IoT-based smart lockers with RFID and OTP-based access have been explored to reduce missed deliveries and enhance package security. Their system uses a centralized server and Android app to communicate with lockers. However, it heavily depends on cloud infrastructure and does not include physical tamper detection like load sensors or fail-safes for network outages.

Reference: Singh, A., & Rani, R. (2020). “IoT based Smart Locker System for Secure Parcel Delivery.” International Journal of Engineering Research & Technology (IJERT), Vol. 9, Issue 4.

2. Video Surveillance vs. Smart Sensing

* Traditional CCTV systems are widely used, but they lack intelligence or context awareness. As per Chen et al. (2019), intelligent video systems integrated with motion detection can identify unusual activity but are often expensive and require significant power and storage. They also raise privacy concerns.

Reference: Chen, Y., Li, X., & Zhang, H. (2019). “Intelligent Surveillance Systems: Challenges and Trends.” Journal of Sensors, Hindawi.

3. Parcel Theft Detection through Smart Drop Boxes

* Gordon et al. (2021) presented a conceptual model using microcontrollers and motion sensors to detect theft from drop boxes. While effective in terms of basic alerts, their system did not integrate proper recipient authentication (like RFID), nor did it provide dual-camera vision or integration with messaging platforms such as Telegram.

Reference: Gordon, D., Myers, L., & Shah, T. (2021). “Design of a Smart Parcel DropBox Using IoT.” Proceedings of the 2021 IEEE SmartTech Conference.

4. Cybersecurity in IoT Home Devices

* A study by Alrawais et al. (2017) emphasized the critical importance of cybersecurity in IoT-based home devices, pointing out that systems should not rely solely on online databases and must log sensitive actions locally. This proposal aligns well with this recommendation by incorporating SD card logging as a backup and including RFID-based physical access control.

Reference: Alrawais, A., Alhothaily, A., Hu, C., & Cheng, X. (2017). “Fog Computing for the Internet of Things: Security and Privacy Issues.” IEEE Internet Computing.

5. Telegram as a Secure Messaging Platform for IoT Alerts

* Kumar & Srinivasan (2022) demonstrated the usage of Telegram Bot API as a secure and cost-effective way to notify users from IoT systems, offering better encryption and delivery speed than traditional email or SMS. The use of Telegram in your system aligns with this modern and practical communication strategy.

Reference: Kumar, V., & Srinivasan, R. (2022). “IoT-Based Alert System using Telegram Bot.” International Journal of Computer Applications, Vol. 184, Issue 17.

6. Multi-Factor Physical Authentication

* In recent work, Pradeep & Khan (2023) studied the effectiveness of RFID combined with weight sensing for verifying actions in smart inventory systems. Their findings support the effectiveness of multi-sensor fusion (e.g., RFID + load cells) in identifying valid users and interactions.

Reference: Pradeep, M., & Khan, R. (2023). “Smart Inventory Tracking with Multi-Factor Authentication using IoT.” International Journal of Emerging Trends in Engineering Research.

In summary, while individual solutions exist—such as smart lockers, video surveillance, or RFID-based systems—the integration of dual-camera monitoring, tamper-proof logging, real-time alerting via Telegram, and fallback cybersecurity protections (like SD-based logging) in a single, low-cost DIY solution is unique. This project fills a clear research and application gap by combining hardware-based physical verification and software-driven intelligent alerting in one holistic smart delivery solution.

## Proposed Solution: Smart door delivery authentication system

### Project Aim

The growing issue of parcel theft and unauthorized pickups from doorsteps has become a significant problem in the current e-commerce landscape. Current systems rely on passive solutions like CCTV, which do not actively prevent theft or verify delivery and pickup actions.

To address this, the project proposes the Smart Doorstep Delivery Authentication system. This solution integrates RFID-based access control, dual camera surveillance, weight sensors, and load cell technology, allowing the owner to authenticate the delivery and pickup process while providing real-time alerts.

This system ensures that:

* The delivery person is authenticated and logged via a combination of IR sensors, RFID, and image capture.
* The parcel is securely stored in a smart container with an automated lock mechanism.
* The owner is alerted in real-time about delivery, pickup, or theft via Telegram notifications.
* All actions are logged, including the images of the person and the parcel, and stored on an SD card, ensuring data is available even in the event of network failure.

By using this system, owners are assured of secure, verified deliveries and pickups, with enhanced cybersecurity features to ensure unauthorized access is immediately detected and mitigated.

### Project Objectives

#### 1. Planning and Research

**Objective:** Research existing technologies and systems, gather necessary hardware and software tools, and understand the problem space thoroughly.

**Tasks:**

* Research similar systems and identify existing solutions.
* Define the project requirements and specifications.
* Decide on the technologies and hardware (e.g., ESP32-CAM, RFID, IR sensors, etc.).
* Gather the necessary resources for development (both hardware and software).
* Prepare the project proposal.
* Team coordination and finalizing project scope.

#### 2. Hardware & Software Development Phase

**Objective:** Develop the core functionality for the system by setting up hardware components and software integration.

**Tasks:**

* Set up ESP32-CAM module and integrate both cameras (one built-in, the other connected wirelessly).
* Connect and configure sensors (IR, RFID, weight sensor, etc.).
* Develop the necessary software to control the hardware (camera snapshots, RFID scanning, weight detection, etc.).
* Program the logic to handle parcel delivery and pick-up.
* Implement the system for logging delivery and pick-up actions with images stored locally.

#### 3. Integration and Logic Design

**Objective:** Integrate the hardware components and ensure they work together seamlessly with the software.

**Tasks:**

* Integrate all hardware components into a single, functional system.
* Develop communication protocols between the hardware (e.g., using Wi-Fi for ESP32-CAM).
* Implement the system to capture pictures during the delivery, pickup, and theft events.
* Develop system logic for detecting valid RFID scans and triggering actions based on sensor data (e.g., lock/unlock container).

#### 4. Testing and Debugging

**Objective:** Ensure the system works as intended, identify and resolve any issues.

**Tasks:**

* Conduct unit tests for each hardware component.
* Test software functionality in real-life scenarios (parcel delivery, pickup, theft detection).
* Identify bugs and resolve issues (both hardware and software).
* Validate the accuracy of images captured and logs stored on the SD card.
* Test the Telegram bot integration for notification functionality.

#### 5. Documentation and Finalizing

**Objective:** Finalize the project by documenting the development process, testing results, and system design.

**Tasks:**

* Document the hardware setup, software architecture, and integration process.
* Prepare user manuals and technical documentation.
* Finalize the code and hardware setup, ensuring everything is functioning properly.
* Create a final project report, including the project outcomes and any future recommendations.
* Conduct a final review with stakeholders and make necessary adjustments before submission.

### Methodology and Technology

#### **SDLC Model Selection**

For the development of the Smart Doorstep Delivery Authentication system the Incremental SDLC Model has been selected.

#### Why Incremental Model?

The system is composed of multiple independently functioning modules (e.g., RFID authentication, camera image capture, load cell weight detection, SD logging, Telegram bot messaging, servo locking). Developing all features in one go would be risky, hard to debug, and inflexible to changes. The incremental model allows each module to be built and tested one at a time. This approach ensures:

* Faster initial delivery of core functionalities (e.g., delivery detection and logging).
* Continuous integration and feedback on each module.
* Easier testing and debugging per module.
* Flexibility in incorporating additional features (like adding facial recognition in future versions).

#### Why Not Other Models?

1. Waterfall Model:

* Not flexible; any change later (e.g., upgrading the alert mechanism) would break the sequence.
* Not suitable for iterative sensor testing and hardware tuning.
* No partial working system until the very end.

1. Agile:

* Agile is more team-intensive with rapid sprints, meetings, and constant feedback loops.
* This project, being a small academic prototype with limited team and short timeline, doesn’t require full agile overhead.

1. Spiral Model:

* Risk-driven model suitable for very large or high-risk systems.
* Our project is low-budget and doesn't require constant risk assessment and iteration.

**Summary:** The Incremental model offers the right balance of modularity, flexibility, and ease of testing for this project. It enables rapid prototyping, validation of each sensor/component, and smooth integration—all of which are essential in hardware-software hybrid systems like this smart delivery security setup.

#### Technologies with Justification

|  |  |  |
| --- | --- | --- |
| Component | Technology | Justification |
| Controller | Esp32-CAM | Low-cost, Wi-Fi, Camera built-in |
| Secondary Cam | Ip Cam (ESP32 or Wi-Fi) | Covers Parcel side separately |
| Communication | Telegram Bot API | Free, reliable, real-time messaging |
| Storage | SD Card Module | Offline image and log backup |
| Auth | RFID RC522 | Secure and controlees access |
| Detection | Load Cell + HX711 | Detects presence/absence of parcel |
| Trigger | IR sensor | Detects person approaching |
| Lock | Servo/Solenoid | Secure physical access control |
| Programming | Arduino C++ | Lightweight and hardware compatible |

#### Team Structure

|  |  |  |
| --- | --- | --- |
| **Member** | **Role** | **Responsibility** |
| **Gihan** | **Hardware Engineer** | **Wiring, assembling sensors and servo/lock** |
| **Senuri** | **Software Engineer** | **Esp32 code, telegram bot integration** |
| **Umair** | **System Designer** | **Camera handling, logging, Sd & alert flow** |
| **Banuka** | **Cybersecurity Lead** | **Access control logic, failure scenarios, testing** |

### Design Overview and Justification

The proposed system integrates several components to ensure secure parcel delivery:

* ESP32-CAM (Cam 1): Captures images of the person during parcel delivery and pickup for accountability.
* Secondary Camera (Cam 2): Records the parcel's condition during delivery, pickup, and theft.
* RFID Reader: Scans the recipient's RFID tag to unlock the container, ensuring authorized access.
* Weight Sensor: Detects when a parcel is placed or removed, confirming valid interactions and detecting theft.
* Locking Mechanism: Keeps the container locked until authorized access is granted.
* MicroSD Card: Stores logs and images locally for reliability during internet outages.
* Telegram Bot: Sends real-time alerts to the owner, including delivery, pickup, and theft notifications.

Justification:

This design ensures security, user privacy, and reliability with low-cost components. It uses multi-factor validation (RFID + weight sensing) and captures visual evidence for full accountability. Offline data storage and Telegram alerts enhance cybersecurity and tamper-proof logging, making it suitable for a wide range of users and environments.

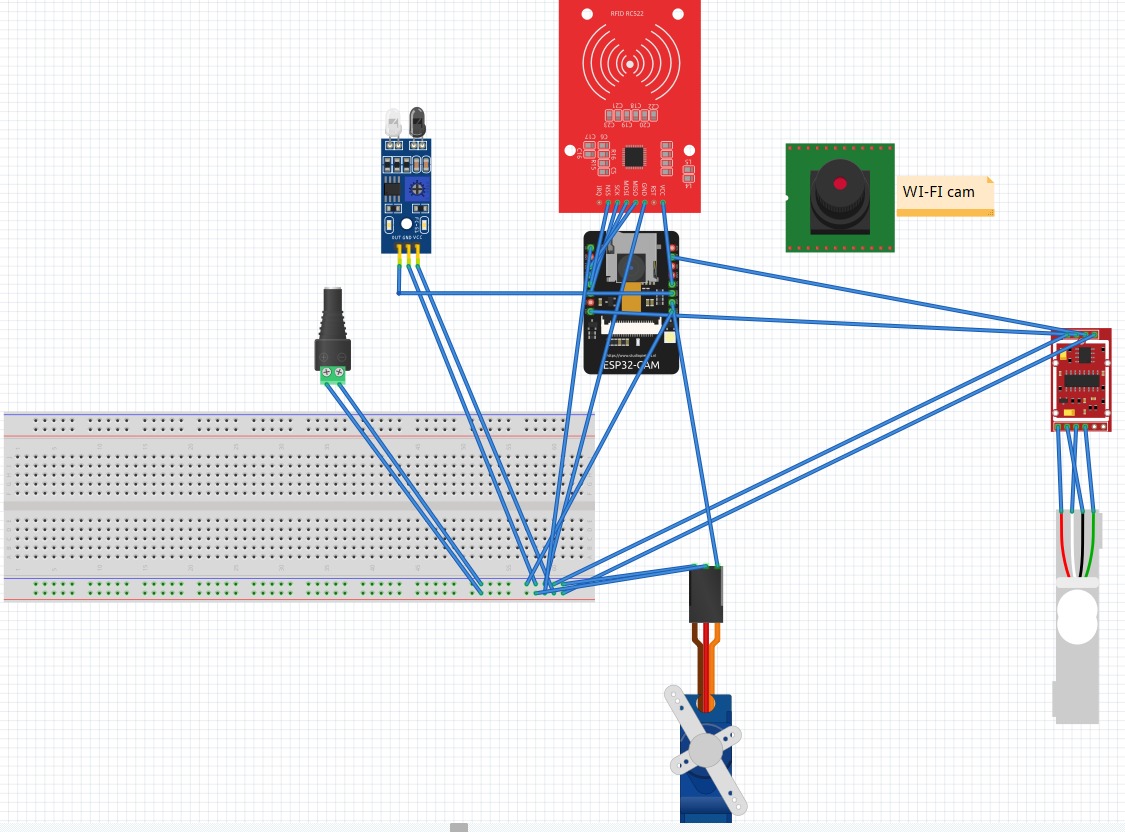


Figure 1 system design

### Project Scope

The project scope outlines the detailed features, system components, and deliverables for the **Smart Doorstep Delivery Authentication System**. The aim is to build a reliable system capable of securing unattended parcel deliveries and pickups.

System Functionalities

1. **User Authentication (RFID-based)**:
   * The system ensures only authorized individuals (parcel owners) can retrieve the parcel by utilizing RFID tags for authentication.
   * **Process**: The user scans their RFID tag to unlock the compartment after confirming the parcel placement, ensuring the right person retrieves the parcel.
2. **Parcel Delivery (Parcel Placement Detection)**:
   * The system uses an **IR sensor** to detect the delivery person and unlocks the parcel compartment.
   * **Process**: The delivery person places the parcel inside the compartment, and the lock mechanism is automatically triggered to secure it after placement. The **load cell** is used to confirm the presence of the parcel.
3. **Parcel Pickup (Theft Detection and Confirmation)**:
   * After the parcel is picked up by the authorized user, the **load cell** detects the removal of the parcel. A picture of the person and parcel is captured using the cameras for verification.
   * **Theft Detection**: If the **load cell** does not register a decrease in weight or if an unauthorized user attempts to pick up the parcel, the system will trigger a theft alert. Images are captured of the person and the parcel during the theft attempt, and an alert is sent via **Telegram**.
4. **Real-time Notifications**:
   * The system sends **real-time notifications via Telegram** to the parcel owner about the delivery and pickup process, including images for visual confirmation.
   * Notifications include details about the delivery time, person’s identity, and the status of the parcel (delivered, picked up, or stolen).
5. **Image Capture for Verification**:
   * **Camera 1** (inbuilt on ESP32): Captures a picture of the person at the time of delivery and pickup.
   * **Camera 2** (external, connected wirelessly): Captures a picture of the parcel when it is placed inside the compartment, and also when it is picked up or stolen.
6. **Tamper Detection**:
   * If any tampering (like attempting to break open the compartment) is detected, the system alerts the owner and logs the event.
   * **IR Sensor** helps monitor unauthorized interactions with the parcel box.
7. **Logs & Data Storage**:
   * All activities (parcel delivery, pickup, and theft) are logged and stored on an **SD card**. This data includes the pictures, timestamps, and RFID tag IDs of the involved individuals.
   * The system can function **offline** without relying on internet connectivity, with the Telegram bot serving as an alert system when the internet is available.
8. **Container Design & Lock Mechanism**:
   * The parcel box is designed as a container that remains unlocked until the delivery person places the parcel inside.
   * After the parcel is placed inside, the system locks the box with a **servo motor** mechanism.
   * The **RFID** system ensures that only the parcel owner can unlock the box by scanning their RFID tag.

Key Functional Components

* **IR Sensor**: Detects the presence of a person (delivery or pickup).
* **Load Cell + HX711**: Detects whether the parcel is placed in or removed from the box (helps to track when the parcel is picked up).
* **RFID Module**: Ensures that only the authorized parcel owner can open the box and pick up their parcel.
* **Camera 1 (ESP32-CAM)**: Captures a picture of the person during delivery and pickup for verification.
* **Camera 2 (Secondary Camera)**: Captures an image of the parcel for verification during delivery and pickup.
* **Servo Motor**: Mechanism to lock/unlock the parcel box.
* **Telegram Bot**: Sends real-time notifications to the parcel owner with images of the process.

System Capabilities

* **Real-time Monitoring and Control**: System can detect and log every interaction with the parcel box.
* **Scalable & Customizable**: The system can be extended to support multiple boxes or integrated with other smart home systems.
* **Offline Functionality**: The system works even when the internet is down (logs to SD card, Telegram notifications when online).
* **Security**: Only authorized users with valid RFID tags can retrieve the parcel. Images and logs are stored for post-event analysis.
* **Event Detection & Reporting**: The system can detect theft attempts, unauthorized access, and tampering. Immediate notifications are sent to the owner, and image evidence is stored.

Expanded Functional Requirements

* **Automated Box Locking Mechanism**: Automatic locking after a parcel is placed ensures security at all times.
* **Cybersecurity**: Protection of user data (RFID tags and transaction logs) via encryption (if applicable) and secure Telegram bot messaging.
* **Backup Data**: In case of system failure, the SD card stores all transaction logs and images, ensuring no data is lost.
* **Physical Security**: The physical construction of the box should be tamper-resistant, including reinforced locking mechanisms to prevent theft by breaking or forcing open.

### Limitations of the Project

While the Smart Doorstep Delivery Authentication System (SDDACS) provides an effective and innovative solution for securing home parcel deliveries, a few practical limitations are recognized, primarily due to cost and hardware constraints:

1. Internet Dependency for Alerts  
 Real-time image and alert notifications are sent via Telegram, which requires a stable Wi-Fi connection. In the event of internet failure, the system continues to operate and store data locally, but remote notifications will be delayed or unavailable.

2. Single Parcel Handling  
 The container is designed to secure and manage one parcel at a time. Support for multiple concurrent deliveries or queueing is not implemented, which makes it best suited for individual residential deliveries.

3. External Power Requirement  
 The prototype relies on an external power source. While it is energy-efficient and can be adapted to battery or solar power in future iterations, the current design assumes a constant power supply.

4. Fixed Field of View  
 With two fixed-position cameras, the system covers key angles (person and parcel) effectively for typical scenarios. However, if a person stands outside the expected zone, image clarity or framing might be suboptimal.

5.Basic Authentication Model  
 The system uses RFID tags for authentication, which is secure for basic use but may be vulnerable if a tag is lost or duplicated. This can be addressed in future versions with encryption, tag revocation, or multi-factor authentication.

### Project Risk and Contingency Plan

Risk Factors:

* 1. Hardware Malfunctions:
* Risk: Components like the ESP32, RFID readers, or cameras may fail.
* Impact: This can affect the parcel detection or authentication process.
* Mitigation: Use durable components, perform regular testing, and keep spare parts available for quick replacements.
  1. Connectivity Issues:
* Risk: Wi-Fi connection may be unstable, leading to failed communication with the Telegram bot.
* Impact: Failure in sending notifications to the user.
* Mitigation: Store logs locally on the SD card and implement retry mechanisms to resend messages once connectivity is restored.
  1. Incorrect RFID Scanning:
* Risk: RFID tags may fail to register or may be misread.
* Impact: Incorrect parcel pickup validation, leading to potential theft or errors.
* Mitigation: Use high-quality RFID tags, test thoroughly, and include a manual override option in case of failure.
  1. Power Failures:
* Risk: Power outages could disable the system temporarily.
* Impact: The system will not operate, affecting parcel management.
* Mitigation: Integrate a UPS (Uninterruptible Power Supply) to keep the system operational during short outages.

Contingency Plan:

* Fallback Mode: In case of system failures (hardware, software, or connectivity), a local logging system will capture all events (including images) on the SD card. The system will attempt to resend data once connectivity is restored.
* Maintenance Schedule: Regular checks and system updates will be conducted to ensure hardware and software are up to date.
* Alternative Communication: If the Telegram bot fails, users will be notified by email or SMS as a backup, using low-cost solutions like Twilio.

This plan ensures that the system remains functional, even under less-than-ideal conditions.

## Project Deliverables & Milestones

After the successful completion of the Smart Doorstep Delivery Authentication System , the following key deliverables will be provided:

1. Functional Prototype Unit
   * The fully assembled hardware including:
     + ESP32-CAM board with inbuilt camera
     + Secondary Wi-Fi camera module
     + RFID scanner module with registered tag
     + IR sensor, load cell with HX711, and servo-lock system
     + SD card module with storage
     + Power system (battery/adapter)
     + Packaged inside a basic container for demonstration
2. Image & Event Logging System
   * All parcel delivery, pickup, theft attempt images captured and stored in SD card
   * Timestamped logs in text/CSV format showing:
     + Person arrival
     + Parcel placed
     + Parcel picked (via RFID)
     + Unauthorized access attempts (e.g., pickup without RFID)
3. Real-Time Alert System
   * Working Telegram bot sending real-time messages with image attachments for:
     + Parcel deliveries
     + Valid pickups
     + Theft or unauthorized access alerts
4. Technical Documentation
   * Detailed system architecture and component diagram
   * Wiring schematics and circuit layout
   * Source code with comments for ESP32, Telegram bot, and sensor integration
   * Component datasheets and configuration notes
5. Testing Documents
   * Unit testing results for each sensor and module (load cell, RFID, IR, servo)
   * Integration testing outcomes with test scenarios (e.g., theft simulation, delivery without pickup)
   * Verification logs and photos from test runs
6. User Manual
   * Step-by-step guide for operating the smart parcel box
   * Instructions for adding new RFID users
   * Setup and configuration instructions (Wi-Fi, Telegram bot token, SD card formatting)
7. Maintenance Guide
   * How to update firmware
   * Troubleshooting common issues (e.g., camera failure, servo jam)
   * Guidelines for battery or power supply maintenance
8. Final Report & Presentation Slides
   * Complete academic project report (proposal, implementation, testing, conclusion)
   * Ready-to-present PowerPoint slides with visuals and working demo summary

## A screenshot of a computer program AI-generated content may be incorrect.Schedule and Budget Summary

Figure 2- Gantt chart

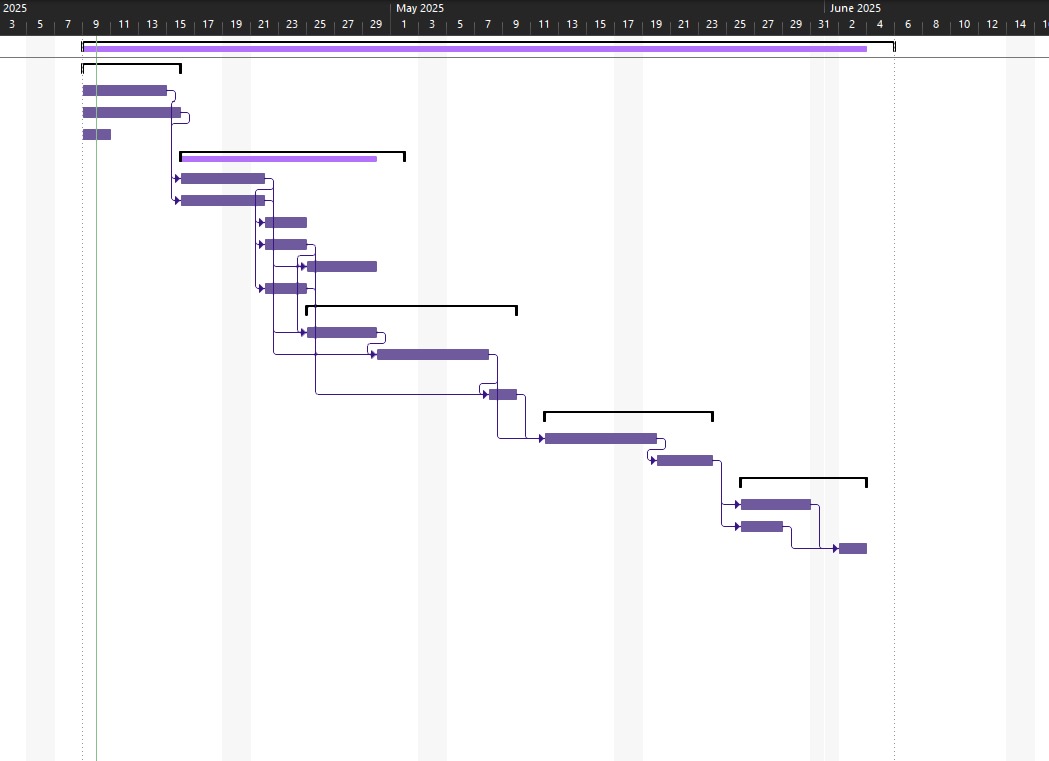


Figure 3- Gantt chart diagram

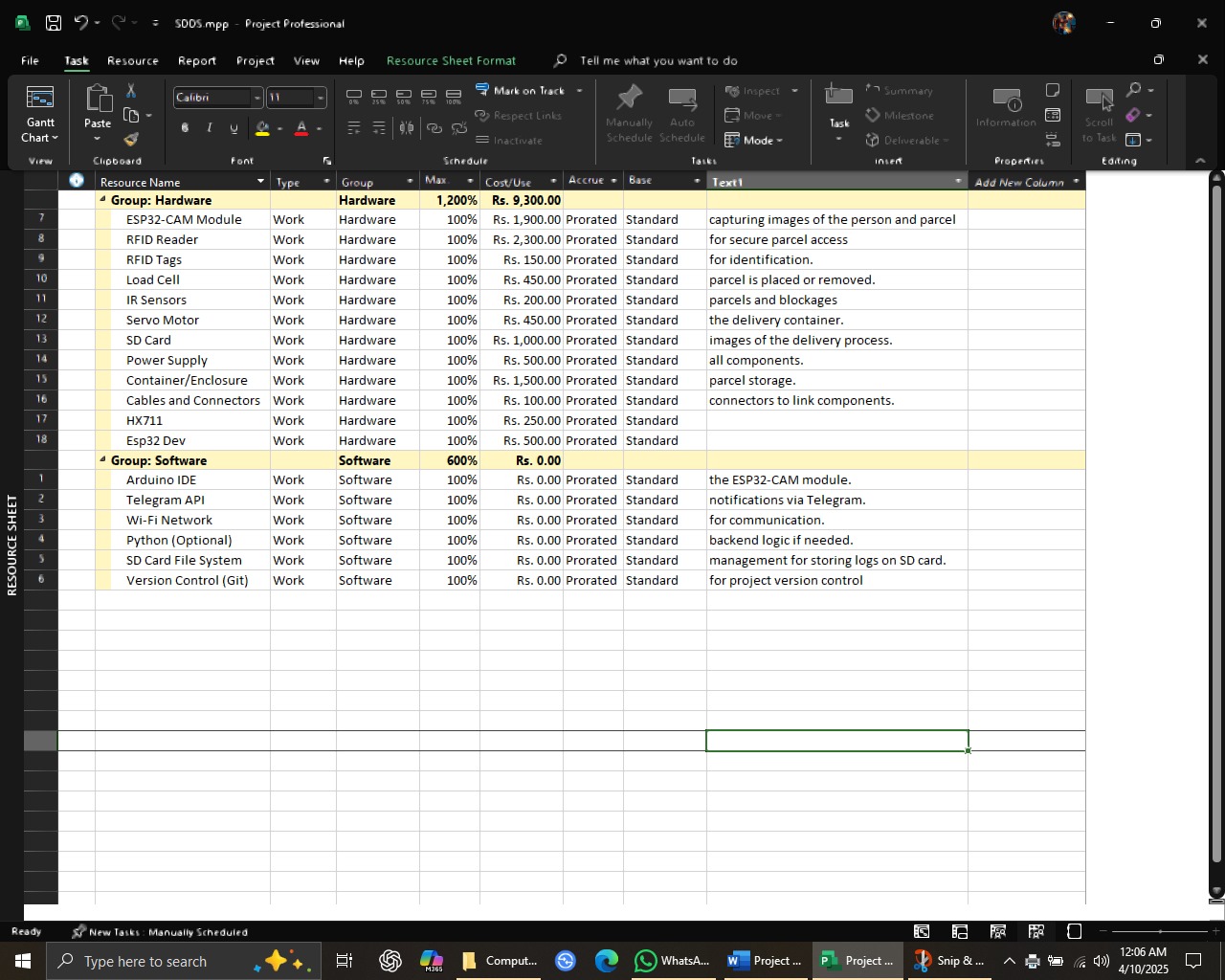


Figure 4 - Resource sheet

The project is structured to be completed within 8 weeks, with key stages outlined in the Gantt chart (included in the proposal). These stages cover hardware setup, software development, system integration, testing, and final deployment. Regular milestones and progress reviews will ensure the project stays on track.

The Gantt chart visually breaks down the timeline, showing the dependencies and durations of each phase, from initial planning to final testing and deployment. It is designed to ensure smooth collaboration across team members, with tasks appropriately distributed for efficient execution.

The budget focuses on acquiring essential hardware components such as ESP32 modules, cameras, RFID readers, sensors, and other necessary peripherals. Additional costs will be minimized through the use of free or low-cost software tools. The resource sheet (attached) outlines the hardware and software resources needed, along with a clear allocation of tasks and responsibilities among team members.

By closely following the schedule and budget, the project aims to remain cost-effective while delivering a high-quality and functional system that meets the project goals.

## References

Ai-Thinker, n.d. ESP32-CAM Product Specification. [pdf] Available at: https://loboris.eu/ESP32/ESP32-CAM%20Product%20Specification.pdf [Accessed 9 Apr. 2025].

Arduino, n.d. MFRC522. [online] Arduino Documentation. Available at: https://docs.arduino.cc/libraries/mfrc522/ [Accessed 9 Apr. 2025].

Necula, B., n.d. HX711 Arduino Library. [online] Arduino Libraries. Available at: https://docs.arduino.cc/libraries/hx711-arduino-library/ [Accessed 9 Apr. 2025].

Lough, B., n.d. Universal Telegram Bot Library. [online] Arduino Libraries. Available at: https://docs.arduino.cc/libraries/universaltelegrambot/ [Accessed 9 Apr. 2025].

Telegram, n.d. Telegram Bot API. [online] Available at: https://core.telegram.org/bots/api [Accessed 9 Apr. 2025].

Components101, n.d. ESP32-CAM Camera Module Pinout, Datasheet, Features and Specs. [online] Available at: https://components101.com/modules/esp32-cam-camera-module [Accessed 9 Apr. 2025].

Balboa, M., n.d. Arduino RFID Library for MFRC522. [online] GitHub. Available at: https://github.com/miguelbalboa/rfid [Accessed 9 Apr. 2025].

Tillaart, R., n.d. HX711 Arduino Library. [online] Arduino Libraries. Available at: https://docs.arduino.cc/libraries/hx711/ [Accessed 9 Apr. 2025].

Lough, B., n.d. Universal-Arduino-Telegram-Bot. [online] GitHub. Available at: https://github.com/witnessmenow/Universal-Arduino-Telegram-Bot [Accessed 9 Apr. 2025].

Telegram, n.d. Bots: An Introduction for Developers. [online] Available at: https://core.telegram.org/bots [Accessed 9 Apr. 2025].

## Definitions

* ESP32: A low-cost, low-power microcontroller with built-in Wi-Fi and Bluetooth capabilities, commonly used for IoT applications. In this project, it is used for controlling the cameras, RFID reader, and communication with other modules.
* ESP32-CAM Module: A development board based on the ESP32 microcontroller that includes a camera module, which is used in this project to capture images of the delivery person and the parcel.
* RFID (Radio Frequency Identification): A technology used for identification and data exchange between devices through radio waves. In this project, RFID is used to authenticate the parcel recipient before allowing access to the delivery container.
* Load Cell: A sensor that measures force or weight. In this project, it is used to detect the placement or removal of a parcel from the container by sensing changes in weight.
* MicroSD Card Module: A storage device that allows data to be written and read. In this project, it is used to store images, logs, and other data locally, ensuring no data loss even when the internet is unavailable.
* Telegram Bot: A software application that runs on the Telegram platform, allowing automated communication with users. In this project, it is used to send real-time alerts about the parcel delivery, pickup, and theft to the owner.
* Solenoid Lock: An electronically controlled locking mechanism that is used to lock and unlock the parcel container based on RFID authentication. It ensures that only the authorized person can open the container.
* System Integration: The process of combining various hardware and software components to work together as a cohesive system. In this project, it involves ensuring all sensors, cameras, and the locking mechanism function together seamlessly.
* SDLC (Software Development Life Cycle): A process for planning, creating, testing, and deploying an information system. This includes phases like planning, design, development, testing, and maintenance, which are used for the software development in this project.
* Tamper Detection: The identification of unauthorized actions or changes to a system. In this project, it involves detecting when a package is stolen or accessed without proper authorization, using a combination of cameras, weight sensors, and RFID.
* Cybersecurity Logging: The process of recording data related to user interactions, system activities, and potential security incidents. In this project, it involves storing logs of all actions (such as RFID scans and weight changes) to provide a traceable history of all parcel activities.
* IoT (Internet of Things): A network of interconnected devices that can communicate and exchange data. In this project, IoT technologies such as the ESP32 microcontroller and Telegram bot allow the system to monitor and manage deliveries remotely.
* Scalability: The ability of the system to handle increasing numbers of users, devices, or data. The system is designed to be scalable, allowing it to be adapted for use in multiple locations or larger-scale operations in the future.