**Sri Lanka Institute of Information Technology**

A blue and orange logo

AI-generated content may be incorrect.

**Case Studies in Software Engineering (SE3070)  
Prorata-02**

**IT21355028**

**Marasinghe M.M.C**

**(Individual Group)**

**0773717193**

**Smart Waste Management System for Urban Areas**

# **Introduction**

**UrbanEcoFlow** is a smart waste management platform designed to modernize and optimize urban waste collection, monitoring, and billing processes. It integrates sensor based tracking, intelligent scheduling, flexible billing models, and user friendly digital services to improve efficiency, transparency, and environmental sustainability in city waste management.

At its core, UrbanEcoFlow ensures that **waste bins are equipped with sensors or tags** that track fill levels, weights, and collection events. Collection staff use the **Collection Truck System** to scan these bins, log pickups, capture geo location and timestamps, and receive instant feedback. This data is processed to prevent duplicate pickups, detect faulty sensors, and generate maintenance requests automatically, ensuring reliable and accurate operations.

For **residents and businesses**, system provides a secure digital portal and mobile app. Users can request new **sensors/tags** for their bins, place **normal or special collection requests** (such as bulky items, e-waste, or hazardous materials), upload supporting images, and monitor their collection history. The system validates requests, suggests available slots, handles urgent requests with premium options, and keeps users updated with real-time **notifications** about their service status.

The system incorporates **smart route optimization**, powered by predictive analytics and realtime sensor data. This helps operations planners and collection trucks reduce travel distances, minimize fuel usage, and optimize manpower allocation. Additionally, environmental authorities can access **reports and sustainability dashboards** to monitor waste patterns, high waste areas, and carbon footprint impacts.

Billing in UrbanEcoFlow is flexible and citizen friendly. The platform supports **pay-as-you-throw (weight-based) models** as well as **flat-rate billing**, depending on municipal policies. It also integrates **paybacks and incentives** for recyclable materials such as e-waste, encouraging citizens to participate in circular economic initiatives. Payments are processed securely through **multiple payment options** (credit/debit cards, subscriptions), while users can view detailed invoices, track balances, and receive reminders for pending payments.

UrbanEcoFlow also ensures backend coordination: **Maintenance Crews** receive automatic alerts when sensors are faulty or bins are damaged, while **Collection Teams** are assigned daily routes that include both routine and special requests. The system’s integrated **Notification Service** ensures that all stakeholders residents, staff, and planners are informed promptly about relevant updates.

Through these features, UrbanEcoFlow bridges technology, sustainability, and user convenience, providing cities with an efficient and transparent platform for managing waste while empowering citizens to contribute to a cleaner and greener urban environment.

# **Proposed Use Cases of the System**

**1. Resident/Business Account Management, Sensor Purchase, and Collection Requests**

This use case enables residents and businesses to register and manage their accounts, request and purchase bin sensors/tags, and place waste collection requests. Users can initiate both normal and special waste collection requests, upload supporting images, and schedule convenient time slots. The system validates all requests, processes payments where required, and provides real-time updates through notifications, ensuring a seamless and transparent service experience.

**2. Waste Collection by Teams and Trucks**

This use case covers the operational side of UrbanEcoFlow where collection staff and trucks identify bins through sensors, validate and log collection events, and capture GPS and timestamp data at each location. The system ensures accurate weight measurements, prevents duplicate pickups, and provides instant audio/visual feedback to staff. Faulty bins or anomalies are automatically flagged for maintenance, while collected data supports optimized route planning and service reliability.

**3. Operational Planning, Maintenance, and Governance**

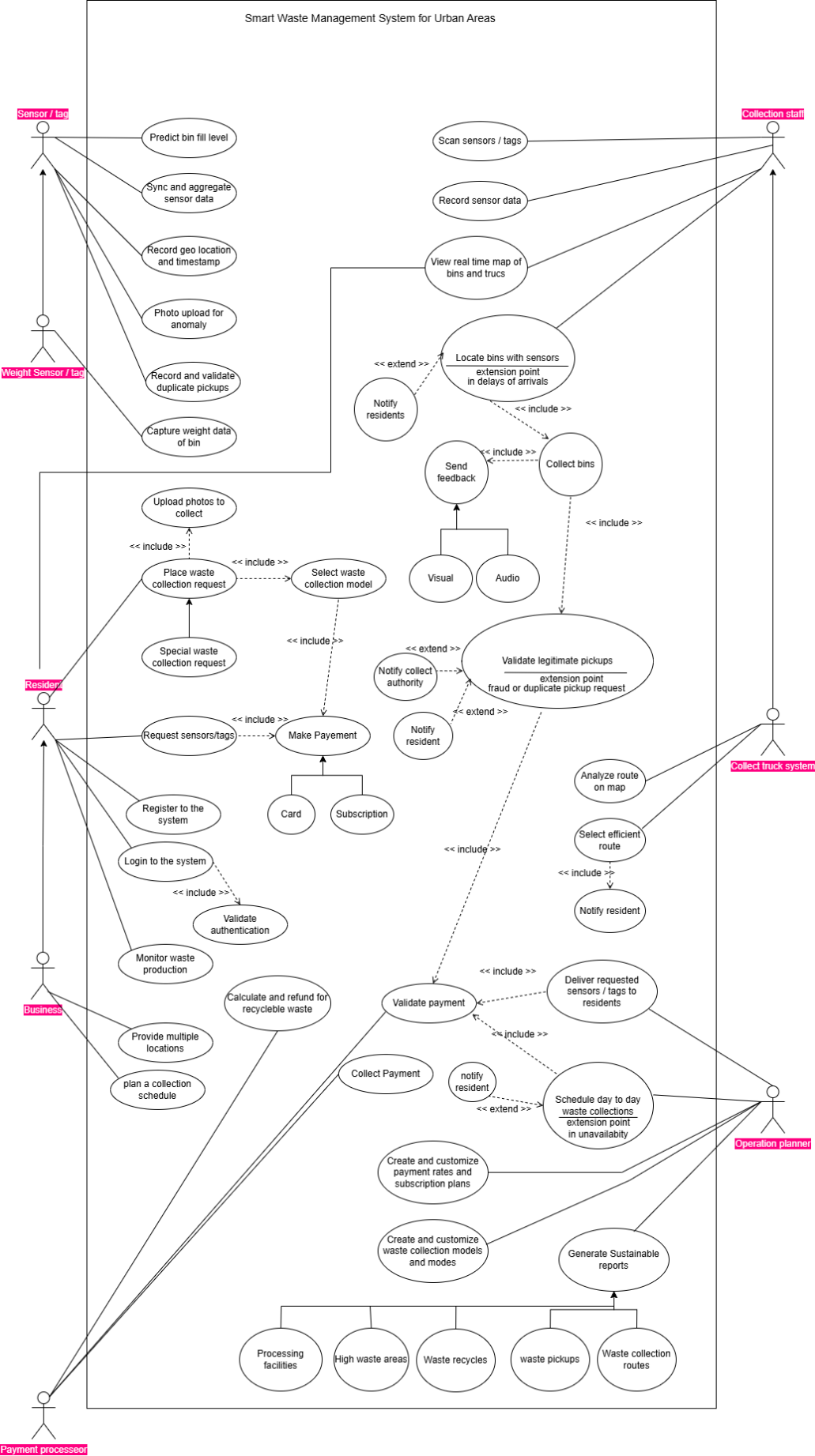
This use case focuses on the back-end functions managed by planners, maintenance crews, and governing authorities. It includes scheduling collections, assigning routes, and monitoring sensor performance. Maintenance teams are notified of faulty sensors or damaged bins, while planners leverage predictive analytics for route optimization. Environmental authorities receive detailed reports on waste generation, recycling rates, and carbon impacts, enabling data-driven decisions and promoting sustainable waste governance.

**4. User Dashboard & Waste Analytics**

Residents and businesses with a **personalized dashboard** to monitor their waste production, collection history, payment status, and sensor performance. The dashboard displays analytics such as waste generation trends, recyclable credits earned, and upcoming collection schedules. Users can track the status of their placed collection requests in real time and receive alerts about anomalies or pending actions

Note: Use case diagram, Sequence diagram and class diagram source files and source images are accessible from below hyperlink if the images provided in this document are difficult to see or zoom in.  
[see source files](https://drive.google.com/drive/folders/1aNxbnAYu8_roEFHag-_wLsd_9bLyBxyF?usp=sharing)

# **High-level use case diagram (for all purposed use cases)**



5

3

4

3

2

1

**Design Alterations**

In this section the design changes and enhancement suggestions are proposed.  
  
**1.** Should be added <<include>> relation from “Place Collection Request” to “Make Payment” for shows payment is mandatory.

**2.** Should be added <<include>> relation from “Scan sensor tags” to “Record sensor data” for shows sensor data will be recorded automatically.

**3.** Recording different types of sensor data could be proposed through generalization.

**4.**Record and validate duplicate pickups could be extracted to separate two use cases and connect through the <<include>> relation

**5.** When Efficient route was selected Operation planner should get notified also

**Revised Use case diagram** A diagram of a diagram

AI-generated content may be incorrect.

# **Class diagram (for all purposed use cases)**

A diagram of a computer

AI-generated content may be incorrect.

2

3

2

1

**Design Alterations**

In this section the design changes and enhancement suggestions are proposed.  
  
**1.** Generating report responsibility should be moved to the Report class. It should not be the responsibility of the PaymentProcessor class.

**2.** Since sending notification could be treat as a major functionality it can be form up a separate class and it should be poses the sending notification responsibility.

**3.** CollectionStaff directly linked with Sensor → staff should interact through Bin  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
**Revised Class diagram** A diagram of a computer

AI-generated content may be incorrect.

# **Selected use case**

# Resident/Business Account Management, Sensor Purchase, and Collection Requests

This use case enables residents and businesses to register and manage their accounts, request and purchase bin sensors/tags, and place waste collection requests. Users can initiate both normal and special waste collection requests, upload supporting images, and schedule convenient time slots. The system validates all requests, processes payments where required, and provides real-time updates through notifications, ensuring a seamless and transparent service experience.

# **Detailed use case scenario (for selected use case)**

|  |  |  |
| --- | --- | --- |
| **Name** | Request Sensors and Place Waste Collection Request | |
| **Description** | This use case allows Residents or Businesses to request bin sensors/tags, make payments, place normal or special waste collection requests, and optionally upload supporting images. The system validates requests, assigns schedules, and notifies users of confirmations. | |
| **Preconditions** | * Resident/Business is registered and logged into the system. * At least one active account and address exist. * Sensors are required in order to place within bins. * Collection schedules and teams are already configured by the Operations Planner. * Payment account is valid (if charges apply). | |
| **Main flow** | **Step** | **Action** |
| 1 | **Resident/Business** logs into the system portal/app |
| 2 | User selects **“Request Sensor/Tag”** if their bin lacks one |
| 3 | Make payment for the requested sensors |
| 4 | User selects **“Place Waste Collection Request.”** |
| 5 | User specifies request type |
| 6 | User upload images for special waste collection request |
| 7 | User make payment for the request and place request |
| 8 | User view the placed collection request in **My Collection Orders** |
| 9 | User monitors his/her usual waste production in their dashboard |
| **Alternate Flows** | **Step** | **Action** |
| 2a | If sensor/tag cannot be delivered (e.g., stock unavailable), system notifies user and reschedules |
| 3a | System notifies user that account funds are insufficient |
| 5a | If Special Collection: system prompts to **upload supporting images** (optional for normal collection). |
| 5b | User can either confirm new date or mark request as urgent (extra fee) |
| 7a | System validates request details (bin ID, account, waste type). |
| 7b | System confirms the request and sends notification (SMS/email/push). |
| **Exception Flows** | **Step** | **Action** |
| 3a | System populate user current balance to the user |
| 5a | If No Slots Available for Special RequestSystem suggests alternate dates |
| 6a | If file format/size is unsupported, system prompts re-upload |
| 7a | If request is linked to an unregistered bin, system rejects it and suggests requesting a new sensor/tag first. |
| 7b | If user submits hazardous waste as normal, system flags and redirects to **Special Collection workflow**. |
| 7c | If payment for special collection fails, system cancels the request and prompts retry |
| **Postconditions** | * Sensor request is logged, scheduled, and tracked. * Waste Collection Request is recorded, scheduled, and either completed or rescheduled. * Notifications are sent to the user confirming request status. * Any applicable fees are processed or retried. * Collection team receives request details in their daily route plan. | |

**Design Alterations**

In this section the design changes and enhancement suggestions are proposed.  
  
**1.** In **Preconditions**, “User must have an active account with valid credentials.” Should be included

**2.** In **Postconditions**, “User dashboard updated with request and waste trend analytics.” Should be included.

# **Detailed sequence diagram (for selected use case)**

A black and white document

AI-generated content may be incorrect.

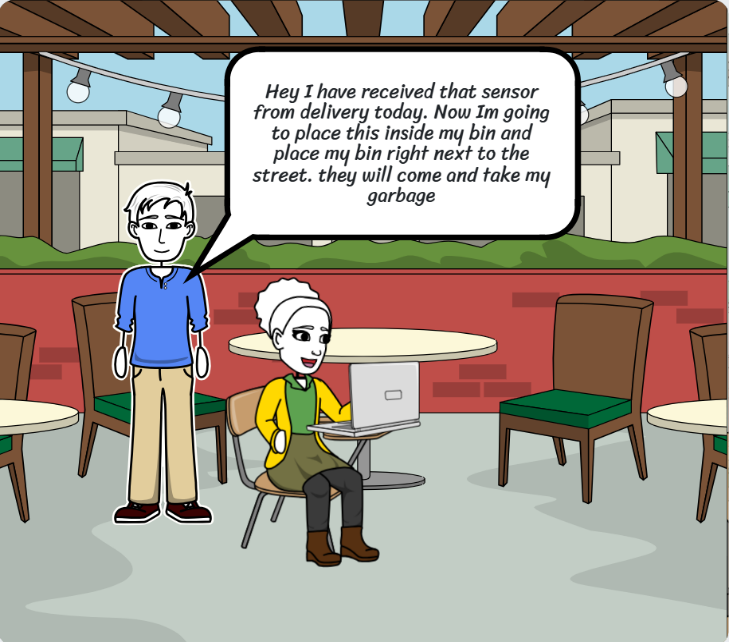
**Design Alterations**

In this section the design changes and enhancement suggestions are proposed.  
  
**1.** Adding **OPT fragment** for placing urgent waste collection request with additional fee.

**Revised Sequence diagram  
  
A white sheet of paper with lines and dots

AI-generated content may be incorrect.**

# **Storyboard (for selected use case)**



# **Low-fidelity wireframe (for selected use case)**

A screenshot of a register account

AI-generated content may be incorrect.A screenshot of a login form

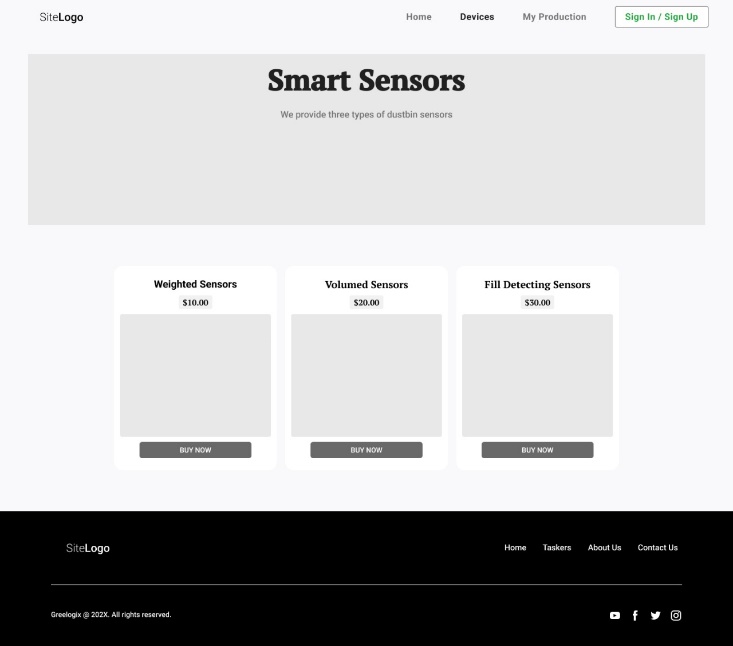
AI-generated content may be incorrect.

A screenshot of a login page

AI-generated content may be incorrect.A screenshot of a login page

AI-generated content may be incorrect.

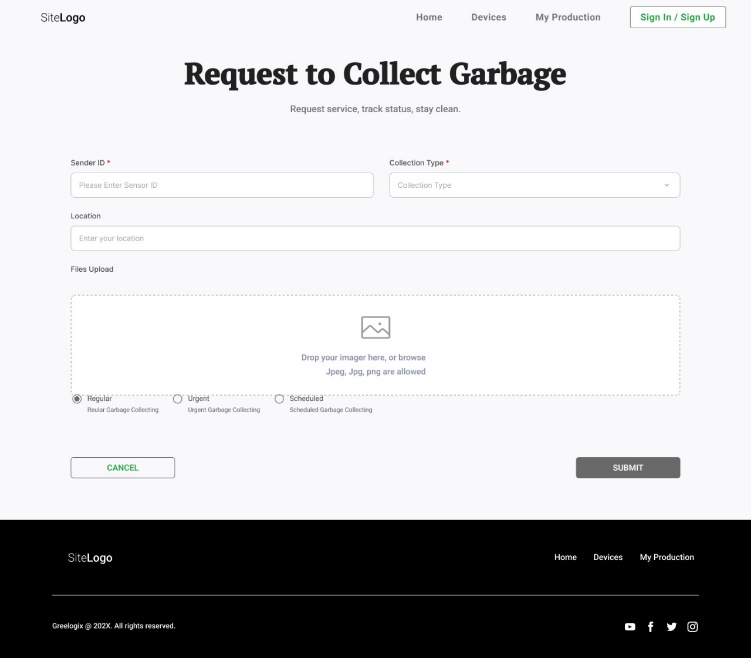
A screenshot of a computer

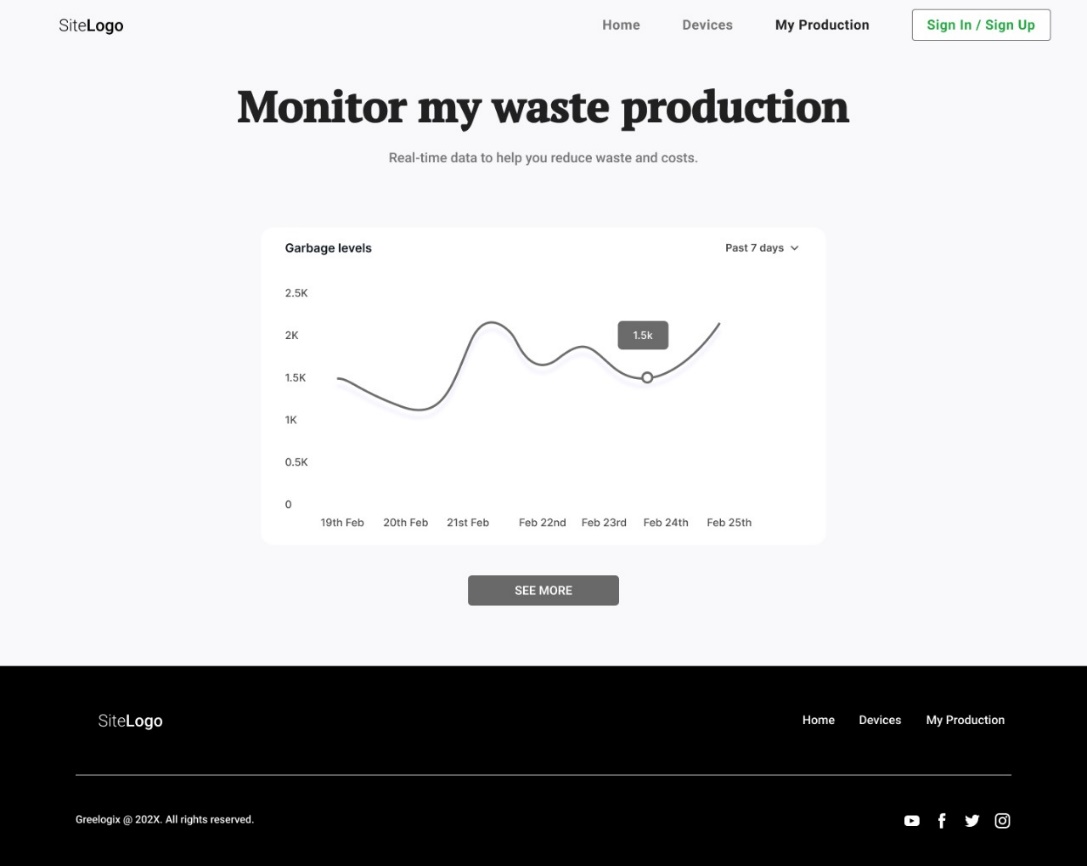
AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

# **Detailed high-fidelity wireframe (for selected use case**

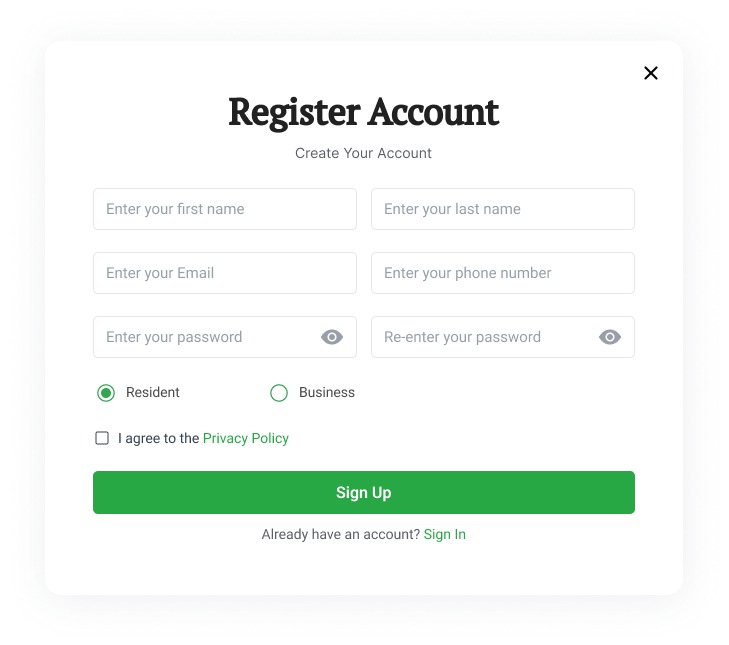




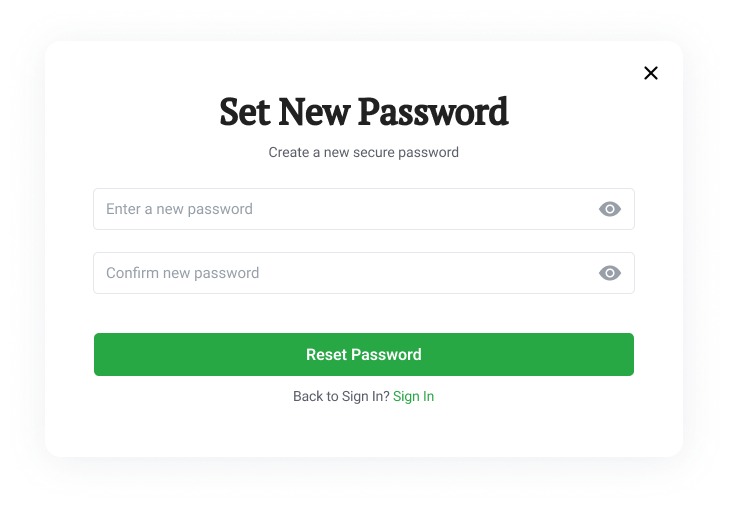
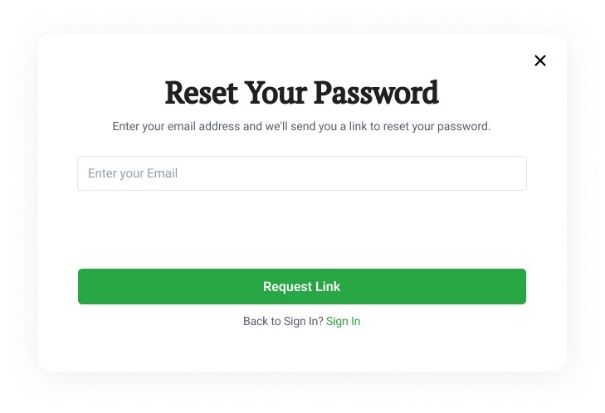
A screenshot of a computer

AI-generated content may be incorrect.

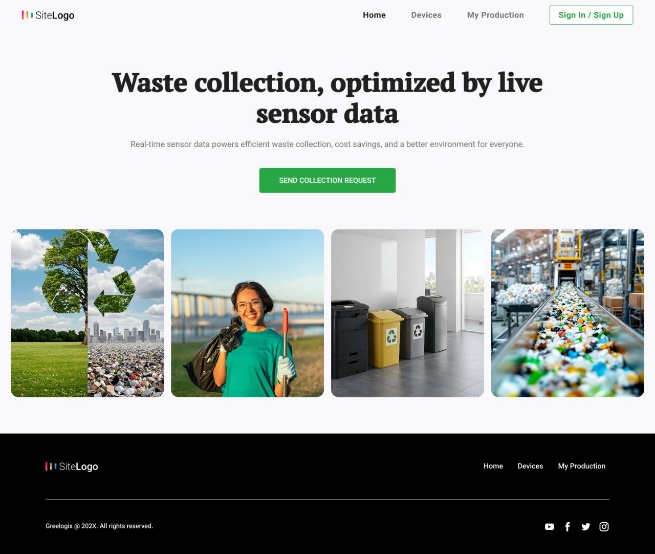
**Detailed high-fidelity wireframe (for selected use case**

A screenshot of a login form

AI-generated content may be incorrect.



A screenshot of a website

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a phone

AI-generated content may be incorrect.