Project Title: Waste Management System

Supervised By

Dr. Imran Mahmud

Professor & Head

Department of Software Engineering

Daffodil International University

Submitted By

Towhidul Islam Tunon

ID: 212-35-751

Department of Software Engineering

Daffodil International University

This project report has been submitted in fulfilment of the requirements for the Degree of Bachelor of Science in Software Engineering.

APPROVAL

This thesis titled on "**Waste Management System**", submitted by **Towhidul Islam Tunon (ID: 212-35-751)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS

Chairman

Dr. Imran Mahmud
Professor & Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 1

Md Shohel Arman
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 2

Md. Rajib Mia
Lecturer (Senior Scale)
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

External Examiner

Md Habibur Rahman
Associate Professor
Department of Computer Science and Engineering
Islamic University, Bangladesh

DECLARATION

I hereby declare that; this project has been done by me under the supervision of **Dr. Imran Mahmud, Professor & Head Department of Software Engineering, Faculty of Science and Information Technology, Daffodil International University.** I also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Towhidul Islam Tunon

Towhidul Islam Tunon
ID: 212-35-751
Department of Software Engineering
Daffodil International University

Certified by

Dr. Imran Mahmud
Professor & Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

ACKNOWLEDGEMENT

I would like to begin by expressing my sincere gratitude to Almighty Allah. I am grateful for His blessings, which gave me the strength, focus, and determination to complete this project.

I am truly thankful to my parents for their constant encouragement and support. Their faith in my abilities has motivated me throughout my academic journey and this achievement.

I want to extend my heartfelt appreciation to **Prof. Dr. Imran Mahmud, Head of the Department of Software Engineering at Daffodil International University**. His guidance and ongoing motivation were essential during this project. His valuable insights, consistent support, and careful supervision were crucial in shaping the final result.

I appreciate all the faculty members and lab assistants in the Department of Software Engineering for their help and technical support throughout the development process.

Finally, I am grateful to my classmates and friends for their collaboration, suggestions, and encouragement during this journey. Their support, whether big or small, has meant a lot to me.

Abstract

Our cities have a big problem with trash, and the old ways of dealing with it don't work anymore. This project shows how to use new technologies to handle trash in a better way. We have little sensors inside trash cans that tell us when they are full. Our software looks at this information at a central centre to discover the best ways to collect it. We don't send trucks out on a set schedule; we simply send them out when and when they are needed. We've also made a mobile app that lets customers report bins that are full and get information on when their trash will be picked up. We are saving money, cutting down on pollution, and using less fuel by doing this. This technique ultimately helps keep our cities cleaner and healthier, and it also makes garbage handling more efficient and better for the environment

Contents

Chapter 1: Introduction

1.1 Project Overview

The Waste Management System is a computerized tool that wants to make it easier and better to collect and manage trash in cities. It helps in planning, tracking, and reporting trash collection efforts, and it also promotes sustainability and public participation. This system gives people, city authorities, and waste collection workers the tools they need to make things run more smoothly, be more open, and make cities cleaner and greener.

Benefits:

More efficient waste collection

The solution lets you keep track of and handle garbage collection requests in real time. This cuts down on wait times and makes sure that service is on schedule. Waste collectors can work more efficiently by optimizing routes and keeping track of performance.

Centralized Reporting and Data

A central database has up-to-date information on the categories of garbage, how much there is, and how it is collected. This helps authorities plan, recognize trends, and use their resources more effectively.

Transparency and user engagement

The dashboard makes it easy for residents to ask for services, check on the status of their requests, and give comments.

Sustainability of the environment

The system aids the environment by encouraging recycling and cutting down on trash in landfills.

Scalability and Adaptability

The system is made to be adaptable and can develop to meet the needs of bigger cities. It can also include features like reward programs, IoT sensor integration, or AI-based insights.

1.2 Project Purpose

The goal of the Waste Management System is to build a smart, easy-to-use platform that automates and improves the collection and monitoring of trash. It fixes problems with traditional ways of dealing with waste, such as inefficiency, lack of transparency, and lack of public participation. The system gives service providers and the community a digital solution that helps both.

The goal of this system is to:

- Make it easier for residents, collection staff, and city officials to pick up trash.
- Make sure that trash is picked up on time and without fail by answering requests quickly and tracking them in real time.
- Tracking the work and actions of collecting staff will help hold them accountable.
- Help the environment by recycling more and planning your waste based on statistics.
- Help people make decisions by giving them easy-to-understand reports and access to all facts in one place.

The system's goal is to make places cleaner, motivate people to be responsible citizens, and help cities learn how to manage their resources in a wise and sustainable way.

1.3 Proposed System

The planned Waste Management System is a digital platform that aims to make it easier and better for both homes and businesses to collect and manage their waste. There are different parts of the system for users, drivers (who pick up trash), and administrators. This framework makes it easier to coordinate, deliver services, and keep an eye on things.

Important Parts of the Proposed System:

User Module:

1. Send in requests for trash pickup.
2. Keep an eye on the status of requests in real time.
3. Look at the service history and give feedback.
4. Get updates and reminders regarding collection schedules.

Driver Module:

1. See the assignments you have been given to collect.
2. You can accept or decline tasks based on your schedule.
3. Change the status of a task, like "collected" or "delayed."
4. Keep an eye on your own work and schedules.

Admin Module:

1. Take care of driver and user accounts.
2. Give out collection jobs and keep an eye on them.
3. Look at system stats and make reports.
4. Make sure you follow the rules on waste in your area.

Features of the system:

1. A centralized database for keeping track of all user, request, and performance data.
2. Tracking in real time for planning and keeping an eye on things.
3. Role-based access control protects the system and keeps data private.
4. A responsive web interface lets you access it from any device.
5. Tools for data analytics to help cities plan for waste and work toward sustainability.

What the Proposed System Wants to Achieve:

1. Make rubbish collection services more reliable and efficient.
2. Cut down on complaints, missing pickups, and delays.
3. Get more citizens involved and give feedback.
4. Help the environment by promoting recycling and wise trash planning.

1.4 Technical Feasibility Analysis

The technical feasibility of the Waste Management System examines whether the required technology, tools, and expertise are available and sufficient to implement and maintain the project successfully.

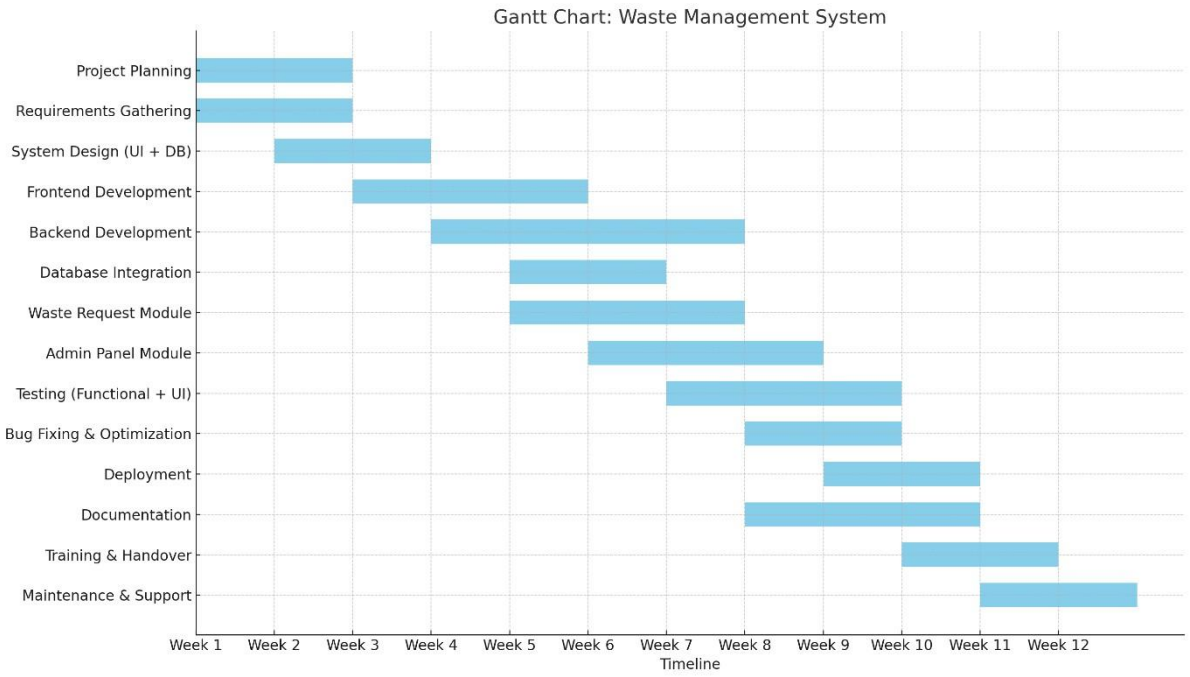
1. Technology Stack Availability

The technologies selected for the project are well-established and widely used in the software industry. These include:

- **Frontend**: HTML, CSS, and JavaScript—for creating responsive and user-friendly interfaces.
- **Backend**: Node.js with Express—for scalable and fast server-side logic.
- **Database**: MySQL for data storage
- **Authentication**: Firebase Authentication—for secure user login and registration.
- **Hosting**: Firebase Hosting—for fast deployment with minimal downtime.
- **Admin Dashboard**: Tailored using React and integrated with role-based access control.

All these tools are open-source or have free tiers, making them cost-effective and easily accessible.

1.5 Project Scheduling



Project Duration: 12 Weeks

Team Members: Project Manager, Developer, Designer, Tester, Admin

Chapter 2: Software Requirements Specification (SRS)

2.1 Introduction:

- **Purpose:** The purpose of this document is to provide a detailed specification of the requirements for the development of the Waste Management System.

- **Scope:** The system will encompass waste management and report functionalities.

- **Definitions, Acronyms, and Abbreviations:** Provide a list of technical terms, acronyms, and abbreviations used throughout the document.

- **System Overview:** Describe the system's high-level architecture and components. Identify the key stakeholders and their roles in using the system.

2.2 Functional Requirements:

Requirement ID	Requirement Description	Stakeholders
FR - 01	allow users to register using email and password.	Resident Users
FR - 02	enable users to log in with valid credentials.	Resident Users
FR - 03	allow users to request waste collection and track the status.	Resident Users
FR - 04	allow drivers to accept or reject waste collection tasks.	Waste collector
FR - 05	able to assign tasks to available drivers.	Admin
FR - 06	generate collection history reports.	System
FR - 07	allow users to update profile information.	Resident Users
FR - 08	log all actions for audit purposes.	System

2.3 Non-Functional Requirements

Requirement ID	Requirement Description	Stakeholders
NFR - 01	respond to user actions within 2 seconds under normal load conditions.	System
NFR - 02	available 99.9% of the time throughout the year.	System
NFR - 03	support up to 1000 concurrent users without performance degradation.	System
NFR - 04	implement secure login with encryption for user credentials.	System
NFR - 05	compatible with modern web browsers (Chrome, Firefox, Edge).	System
NFR - 06	allow regular backups to ensure data recovery in case of failure.	Admin
NFR - 07	follow accessibility standards to support users with disabilities.	System
NFR - 08	maintain logs for all critical actions for auditing purposes.	System

2.4 Hardware Requirements

Server-side Requirements

Component	Minimum Specification	Recommended Specification
Processor (CPU)	Intel Core i5 / AMD Ryzen 5	Intel Xeon / AMD EPYC (Multi-core)
RAM	8 GB	16–32 GB
Storage	256 GB SSD	512 GB – 1 TB SSD (RAID preferred)
Network	100 Mbps Ethernet	1 Gbps Ethernet with static IP
Operating System	Ubuntu Server 20.04 / Windows Server 2019	Latest LTS version of Linux / Windows Server
Database Server	PostgreSQL / MySQL hosting supported	Dedicated database server or managed DB
Power Backup	UPS	UPS + Generator

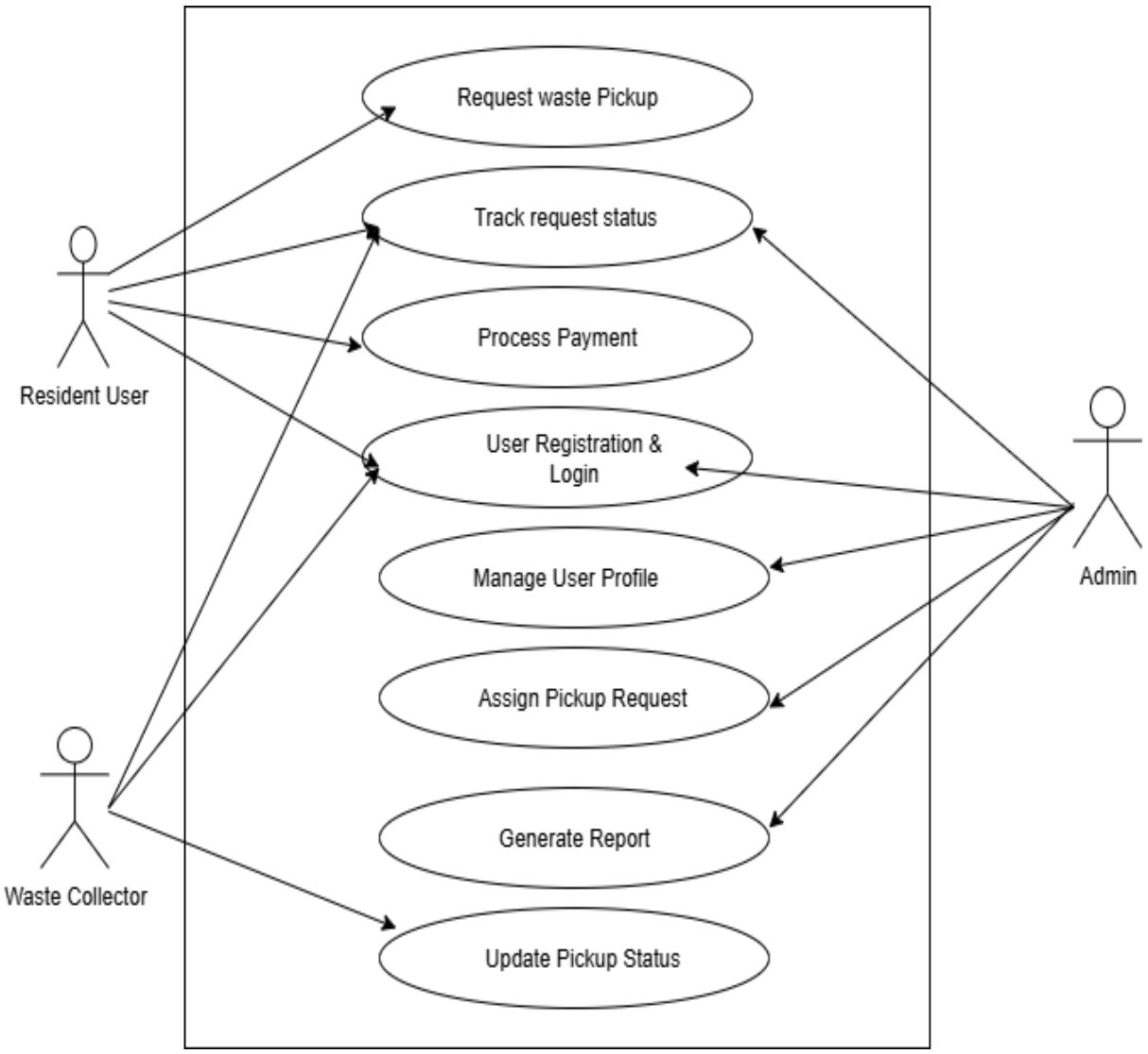
Chapter 3: Design and Implementation

Development Model:

For the development of the Waste Management System, the Incremental Development Model is chosen for its flexibility, faster delivery, and structured phase-by-phase development approach.

Phase	Description
Phase 1: Requirements Analysis	Gather system requirements through interviews, surveys, and observation of current practices.
Phase 2: System Design	Create system architecture, including use case diagrams, ER diagrams, and data flow.
Phase 3: Module Implementation	Develop individual modules such as User, Admin, and Driver modules incrementally.
Phase 4: Integration & Testing	Integrate all modules and perform functional, security, and performance testing.
Phase 5: Deployment	Deploy the system for real-world use and conduct final system validation.
Phase 6: Maintenance	Perform updates, bug fixes, and feature enhancements based on user feedback.

3.1 Use Case Diagram



3.2 Use Case Descriptions

3.2.1 User Registration & Login

Use Case Name	Login
Goal	Allow users to register and log in to the system.
Preconditions	User has a valid email address and access to the system.
Primary Actor	Resident User
Secondary Actor	System
Trigger	User navigates to the login or registration page.
Description/Main Success Scenario	User provides required credentials to register or log in. The system authenticates and grants access.
Post Condition	User is logged in and redirected to their dashboard.
Alternative Flow	If login fails, an error message is displayed.

3.2.2 Manage User Profile

Use Case Name	Manage User Profile
Goal	Allow users to update their personal and contact information.
Preconditions	User is logged in
Primary Actor	Resident User
Secondary Actor	System
Trigger	User selects 'Edit Profile' from the dashboard.
Description/Main Success Scenario	User updates profile details and saves changes. System validates and updates the information.
Post Condition	Profile information is updated successfully.
Alternative Flow	Validation error occurs if invalid data is entered.

3.3.3 Request Waste Pickup

Use Case Name	Request Waste Pickup
Goal	Allow users to request a waste pickup from their location.
Preconditions	User is logged in and location is set.
Primary Actor	Resident User
Secondary Actor	System
Trigger	User clicks on 'Request Pickup'.
Description/Main Success Scenario	User fills in pickup details and submits the request. System stores and queues the request.
Post Condition	Pickup request is saved and assigned a request ID.
Alternative Flow	System shows an error if mandatory fields are missing.

3.3.4 Track Request Status

Use Case Name	Track Request Status
Goal	Resident User views the current status of their pickup request.
Preconditions	A pickup request must have been submitted.
Primary Actor	Resident User
Secondary Actor	Waste Collector (for updating status), Admin (for oversight)
Trigger	Resident selects the option to track their request.
Description	System retrieves and displays the current status of the user's requests.
Post Condition	User sees real-time status updates.
Alternative Flow	If request ID is not found, display a 'No request found' message.

3.3.5 Assign Pickup Request

Use Case Name	Assign Pickup Request
Goal	Admin assigns the pickup task to an available Waste Collector.
Preconditions	There must be pending pickup requests in the system.
Primary Actor	Admin
Secondary Actor	Waste Collector
Trigger	Admin views and selects a request to assign.
Description	Admin assigns request to a collector. System updates status and notifies the collector.
Post Condition	Request is assigned and marked as 'In Progress'.
Alternative Flow	If no collector is available, request remains in the queue.

3.3.6 Process Payment

Use Case Name	Process Payment
Goal	Resident User pays for the waste pickup service.
Preconditions	Pickup request must be submitted and payment option must be enabled.
Primary Actor	Resident User
Secondary Actor	Admin (monitors transactions)
Trigger	Resident selects payment option during or after requesting pickup.
Description	User selects payment method and completes transaction. System verifies and updates status.
Post Condition	Payment is completed and receipt is generated.
Alternative Flow	If payment fails, user is notified and prompted to retry.

3.3.7 Update Pickup Status

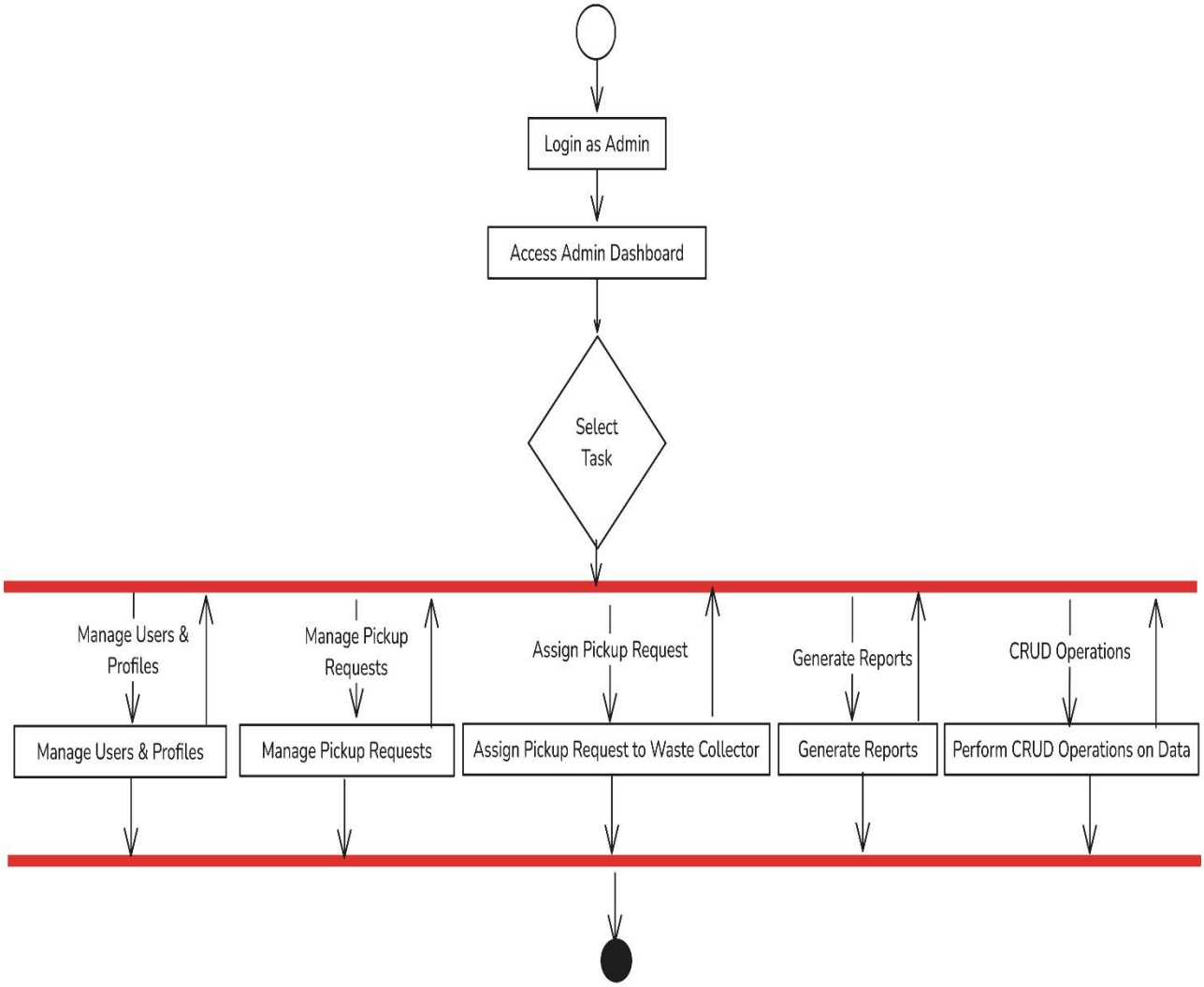
Use Case Name	Update Pickup Status
Goal	Waste Collector updates the status of the assigned pickup.
Preconditions	A pickup request must be assigned to the Waste Collector.
Primary Actor	Waste Collector
Secondary Actor	Resident User (receives status updates)
Trigger	Waste Collector completes or updates pickup process.
Description	Collector updates the request status via the dashboard. System logs the status change.
Post Condition	Status is updated and reflected in user view.
Alternative Flow	If update fails, the collector is notified.

3.3.8 Generate Report

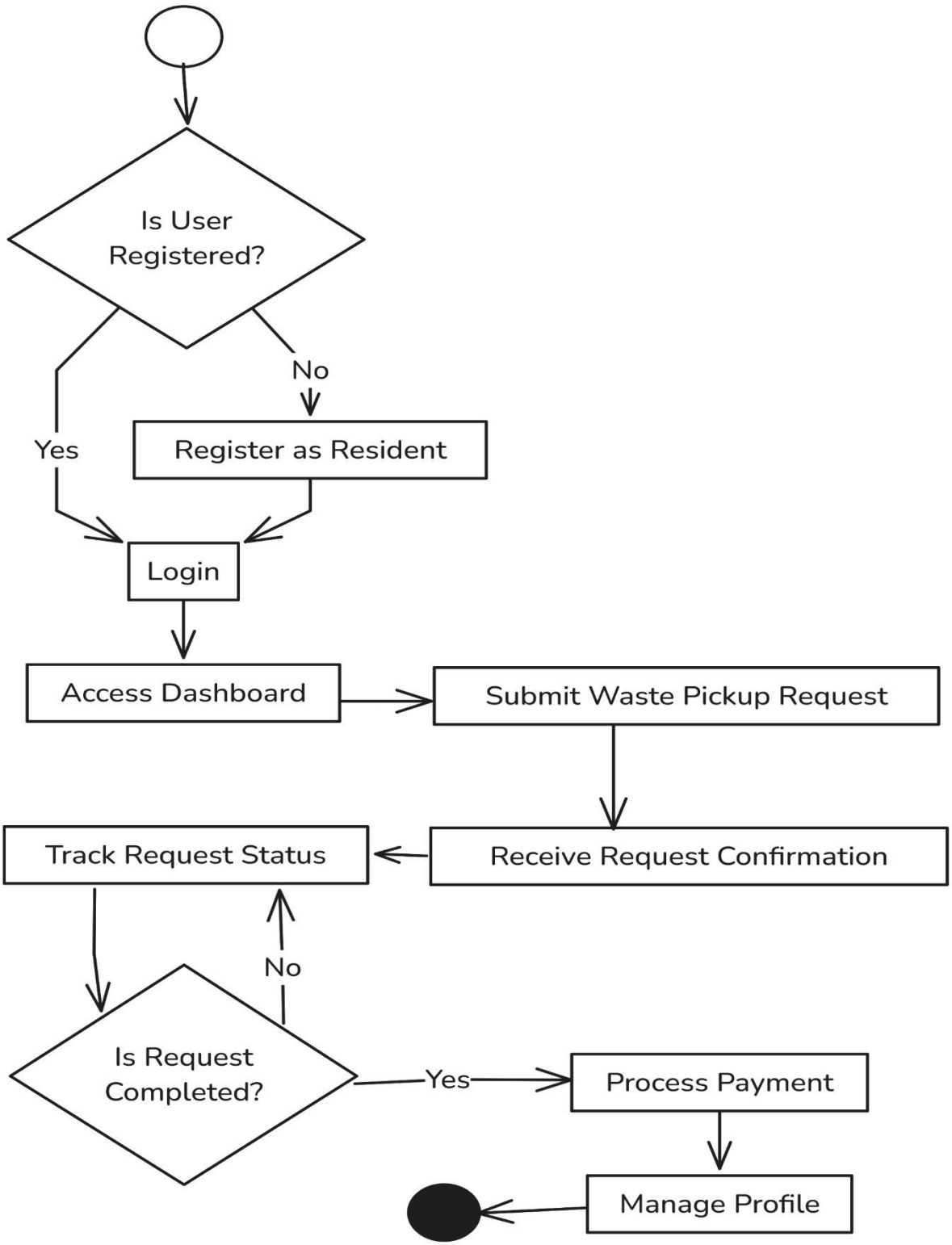
Use Case Name	Generate Report
Goal	Admin generates reports related to requests, payments, or system usage.
Preconditions	Admin must be logged in.
Primary Actor	Admin
Secondary Actor	None
Trigger	Admin selects the "Generate Report" function.
Description	Admin selects parameters and generates the report. System compiles and displays data.
Post Condition	Report is displayed/downloaded successfully.
Alternative Flow	If no data is found, system shows an empty report message.

3.4 Activity Diagram

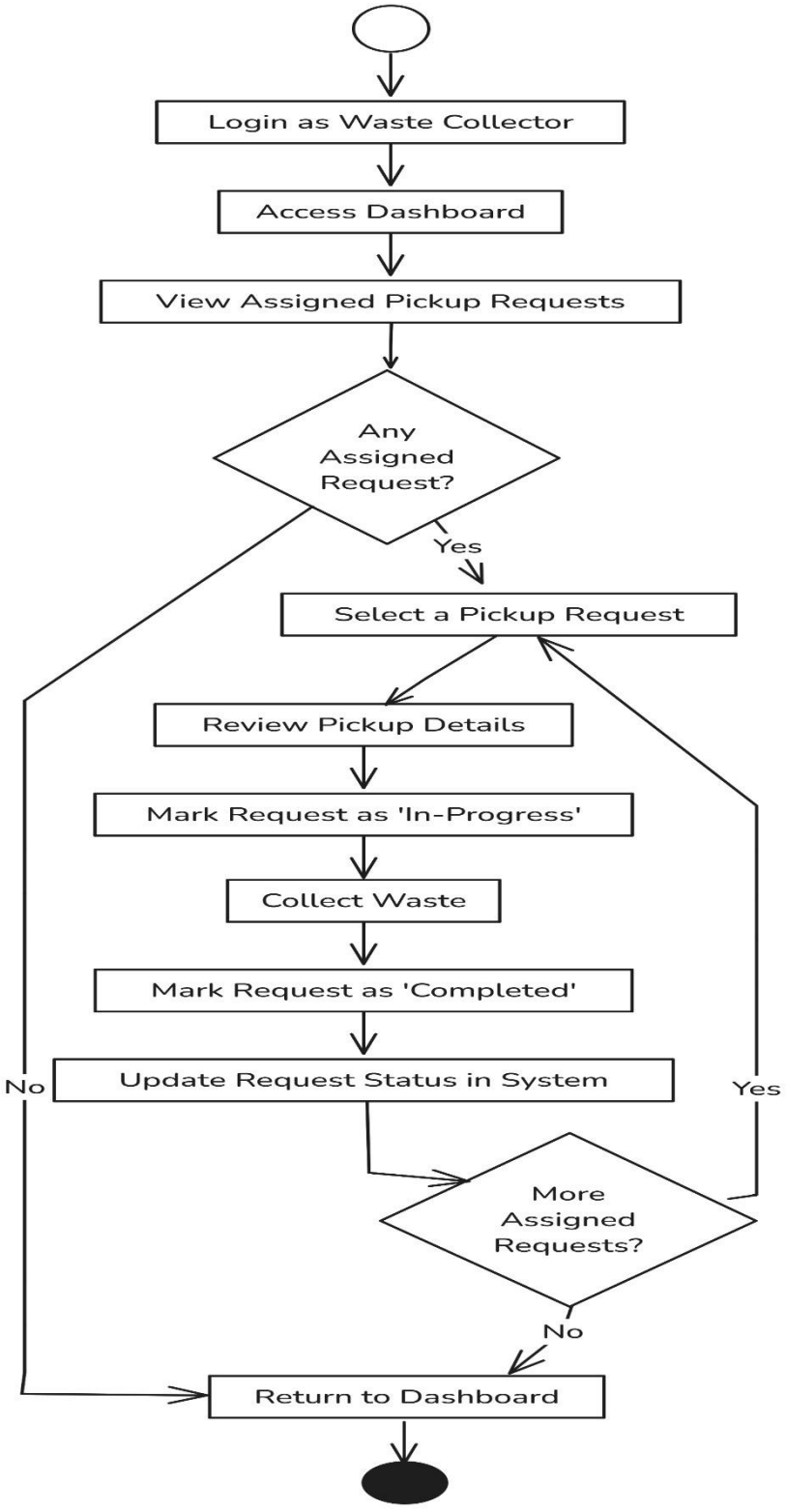
3.4.1 Admin



3.4.2 Resident User

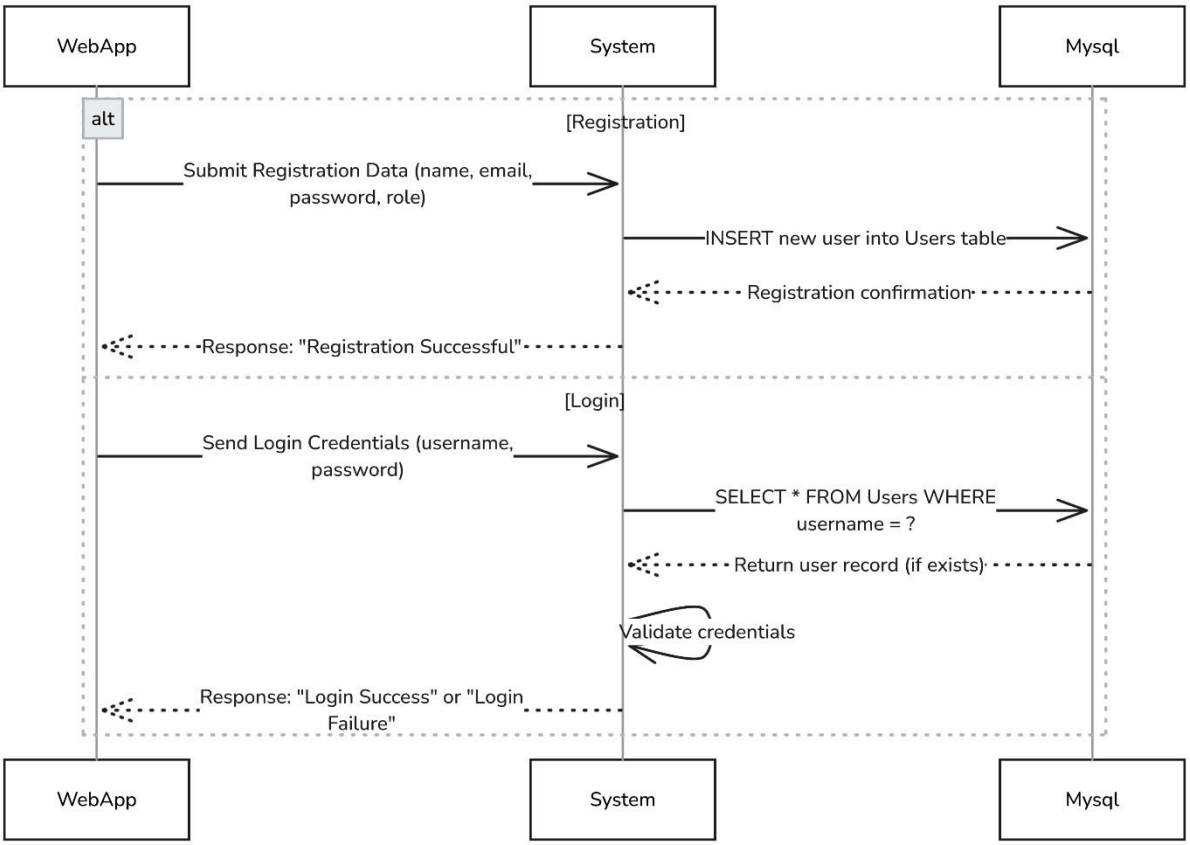


3.4.3 Waste Collector

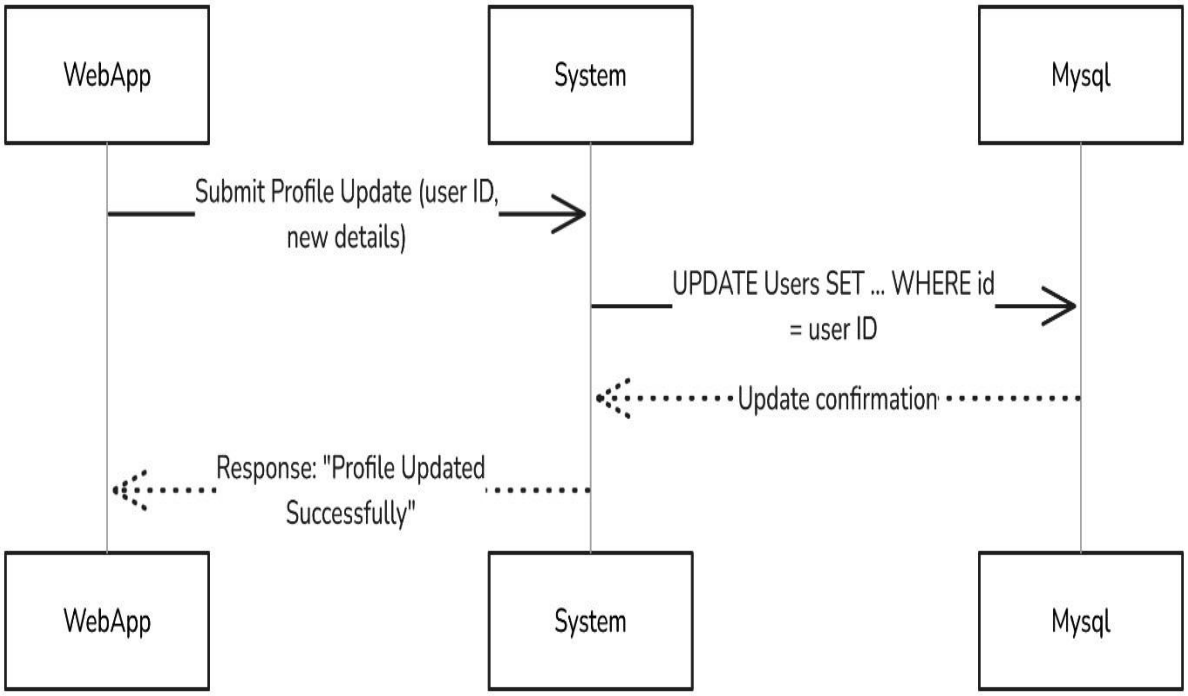

Login as Waste Collector
Access Dashboard
View Assigned Pickup Requests
Any Assigned Request?
Yes
No
Select a Pickup Request
Review Pickup Details
Mark Request as 'In-Progress'
Collect Waste
Mark Request as 'Completed'
Update Request Status in System
More Assigned Requests?
Yes
No
Return to Dashboard

3.5 Sequence Diagram

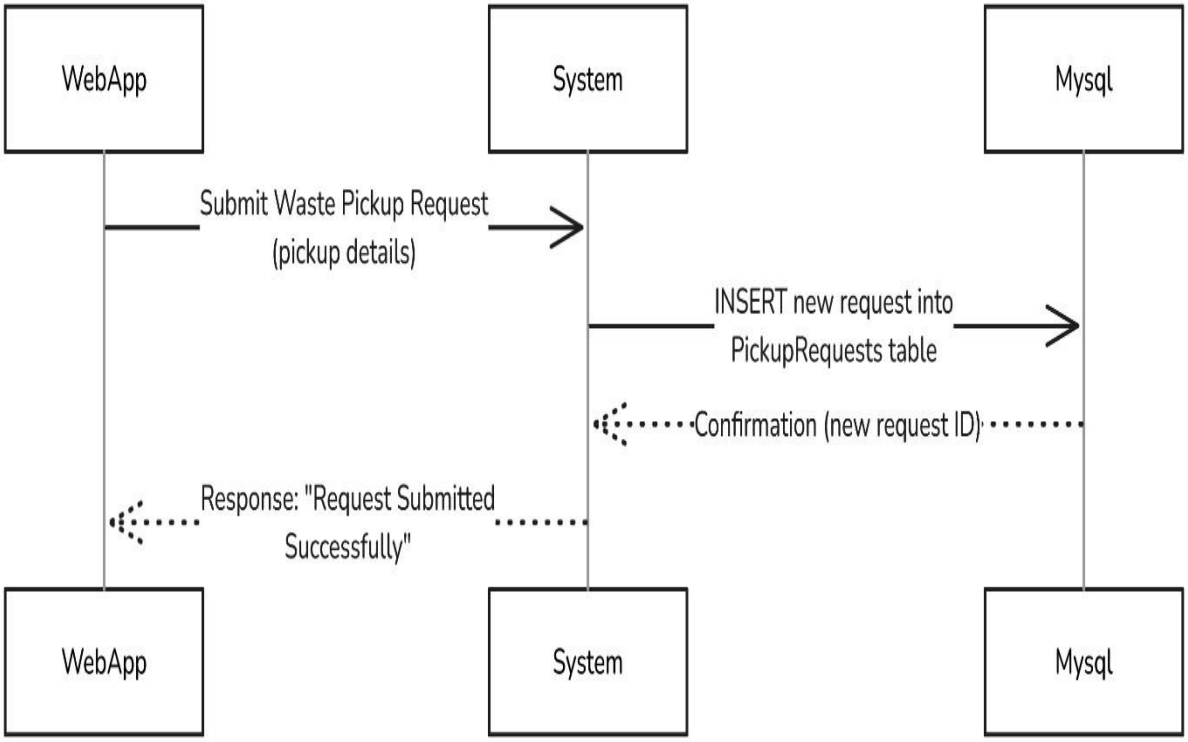
3.5.1. Registration & Login



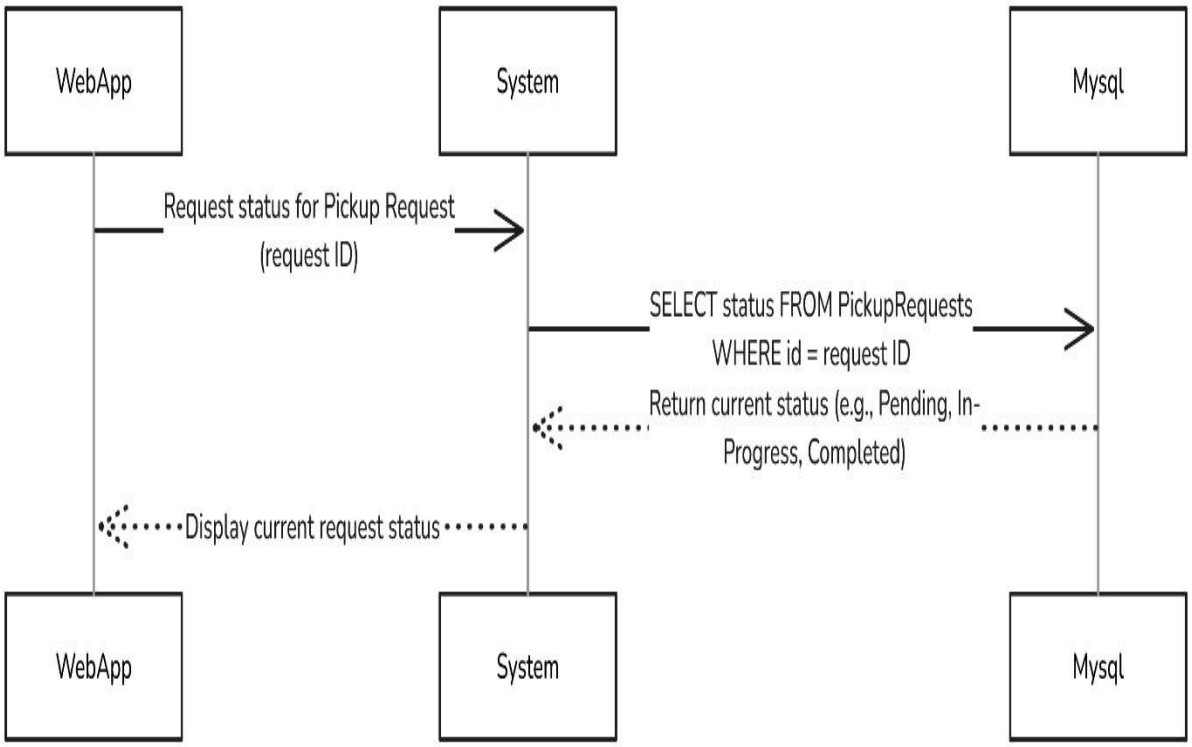
3.5.2 Manage User Profile



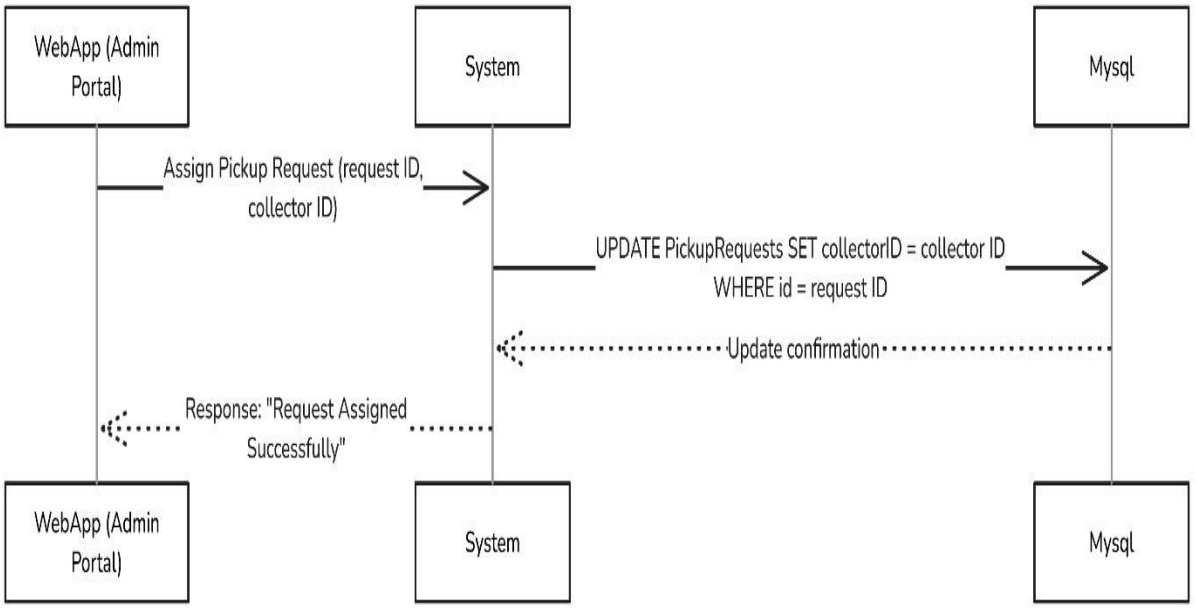
3.5.3 Request Waste Pickup



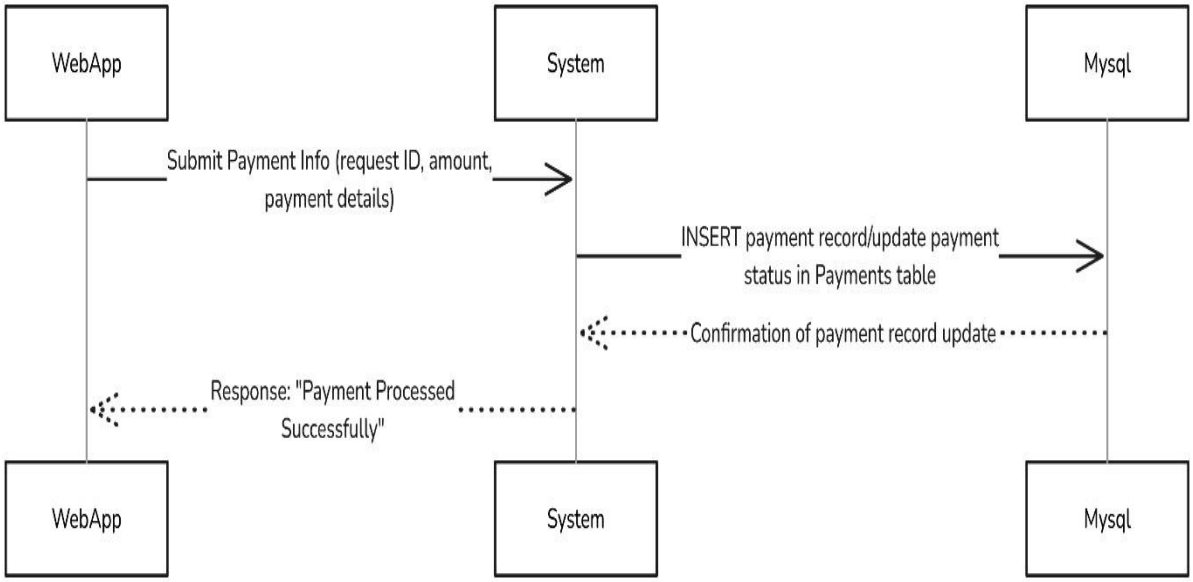
3.5.4 Track Request Status



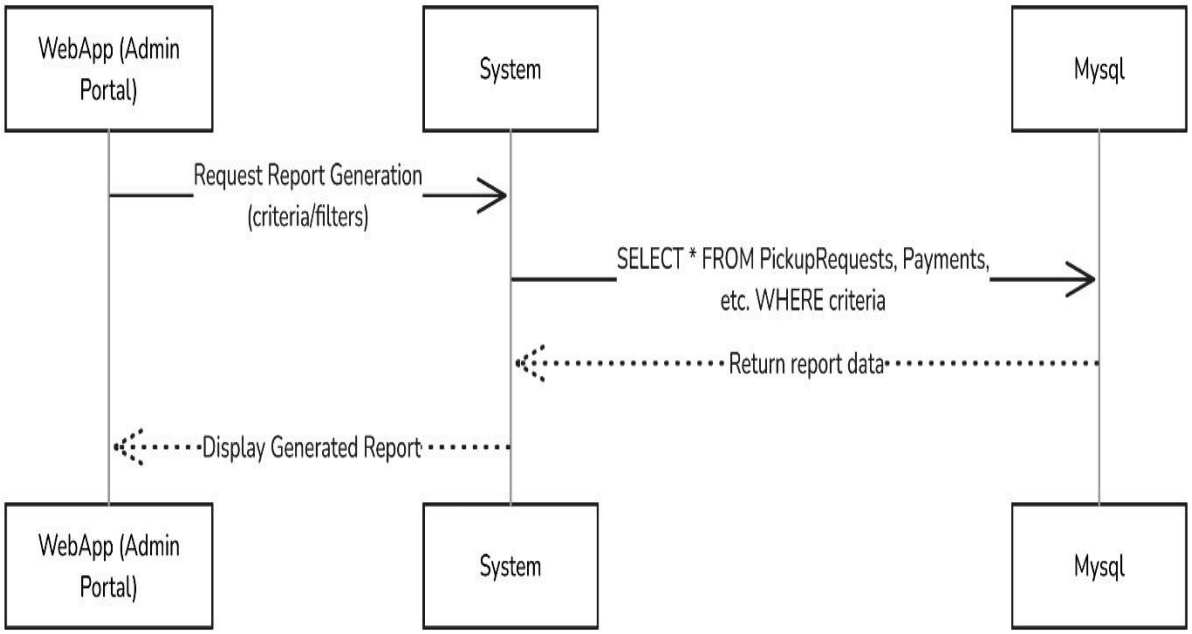
3.5.5 Assign Pickup Request



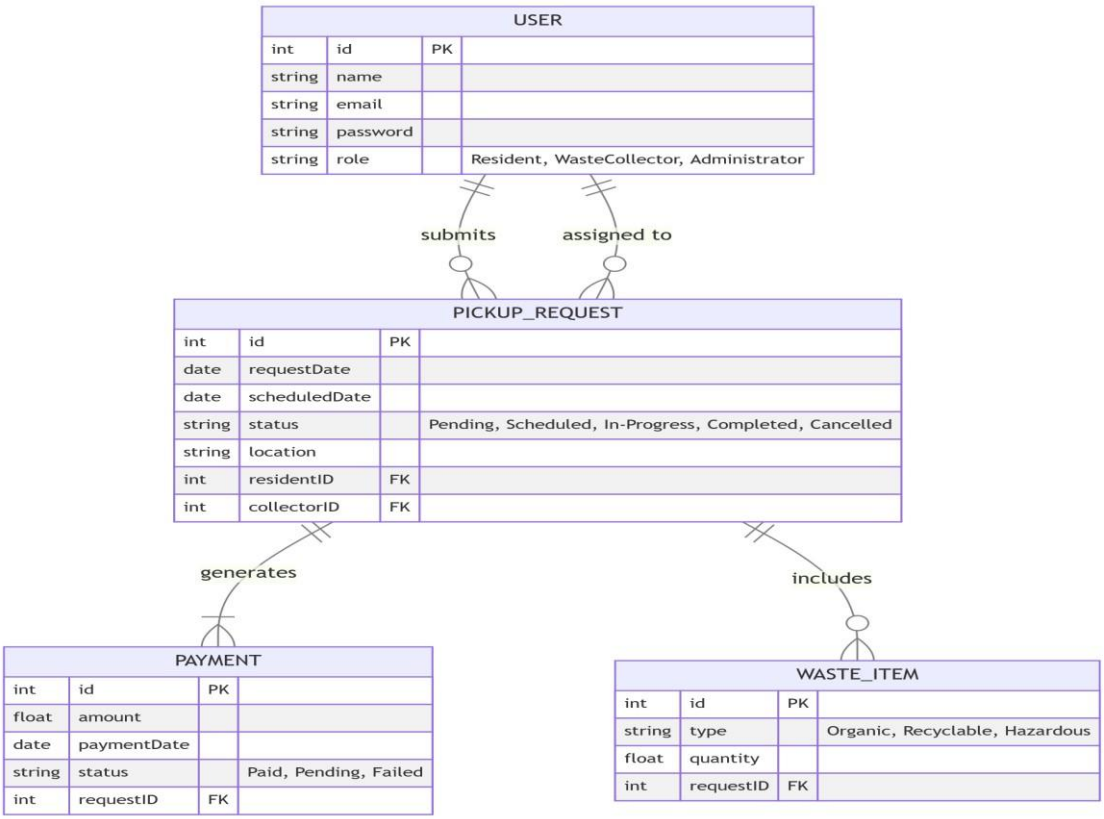
3.5.6 Process Payment



3.5.7 Generate Report



3.6 Entity Relationship Diagram



Chapter 4: Software Testing

4.1 Testing Feature:

Feature	Description
User Authentication & Authorization Testing	Ensures that only valid users (e.g., admin, operators, citizens) can access the system with appropriate roles.
Scheduling Feature Testing	Verifies that waste collection schedules are generated correctly and updated as per inputs.
Route Optimization Testing	Confirms that the system provides the most efficient route for waste collection vehicles.
Real-Time GPS Tracking Testing	Ensures live tracking of vehicles is accurate and up-to-date.
Notification System Testing	Tests if SMS/Email/App notifications are sent for collection reminders, delays, etc.
Waste Volume Calculation Testing	Verifies accuracy in volume/weight calculation from input data or sensors.
Reporting & Analytics Testing	Ensures correct and complete generation of reports like daily collection, missed pickups, fuel usage, etc.
Billing & Payment Testing	Validates invoice generation, payment status, and transactions (if integrated).
System Load Testing	Tests system behaviour under peak usage
Mobile App Testing	Ensures the companion mobile app works correctly on various devices and OS versions.
Data Sync Testing	Ensures cloud and local data stay consistent and synchronized.
Security & Access Control Testing	Verifies protection against unauthorized access, data breaches, and role misuse.

4.2 Testing Strategies:

1. Manual Testing
    - o Used for exploratory testing, UI verification, and small module validations.
2. Black Box Testing
    - o Focuses on testing the system's functionality without knowing internal code.
3. **White Box Testing**
    - o Used in unit testing and integration testing.
4. **Regression Testing**
    - o Ensures that new code changes don't break existing functionalities.
5. **Performance Testing**
    - o Tests system response time, load capacity, and scalability.
6. **Security Testing**
    - o Identifies vulnerabilities in authentication, access control, and data handling.
7. **Usability Testing**
    - o Ensures the system is intuitive and user-friendly for all stakeholders.
8. **Acceptance Testing (UAT)**

- Conducted before final approval and deployment.

4.3 System Testing

Test Case ID	Feature	Objective	Preconditions	Test Steps	Expected Result	Actual Result
TC001	User Login	Verify user can log in with valid credentials	User is registered in the system	1. Open login page 2. Enter valid username and password 3. Click login	User is redirected to the dashboard	User is redirected to the dashboard
TC002	Schedule Collection	Check if waste collection can be scheduled successfully	Admin is logged in	1. Navigate to scheduling page 2. Select date, time, location 3. Click submit	Collection is scheduled and confirmed message is displayed	Collection is scheduled and confirmed message is displayed
TC003	Route Optimization	Verify optimized route is generated	At least 3 pickups scheduled	1. Navigate to route planner 2. Click 'Optimize Route'	System displays optimized route on map	System displays optimized route on map
TC004	Notification System	Ensure notification is sent for scheduled pickup	Collection scheduled for today	1. Wait for scheduled time 2. Check email/app for notification	Notification received before scheduled time	Notification received before scheduled time
TC005	Billing Module	Verify invoice generation after collection	Collection is completed	1. Navigate to billing 2. View generated invoice	Invoice is generated with correct amount	Invoice generated with correct amount

Chapter 5: Deployment and Maintenance

5.1 Deployment

Deployment is the process of making the waste management software available for use in a live (production) environment.

Deployment Steps

- **Build Final Version**: Compile and package the application with tested and approved code.
- **Set Up Environment**: Configure servers (cloud or on-premise), databases, network, and storage.
- **Install Software**: Deploy backend, frontend, mobile apps, and APIs.
- **Data Migration**: Import existing data (users, routes, schedules, etc.) into the new system.
- **Configuration**: Set system parameters like location zones, user roles, notification settings.
- **Testing in Live Environment**: Perform smoke tests to verify everything works post-deployment.
- **Go Live**: Make the system available to users. •
- **User Training**: Provide training to municipal staff, drivers, and admin users.

Deployment Models

- **Cloud Deployment** -Scalable, cost-effective, and easy to maintain.
- **On-Premise Deployment** – Hosted on local servers; more control but higher maintenance.

5.2 Maintenance Activities

- **Bug Fixes**: Regular updates based on error reports or testing.
- **System Monitoring**: Track server uptime, API health, GPS data accuracy, etc.
- **Data Backup**: Schedule regular backups to avoid data loss.
- **Security Patches**: Apply updates to fix vulnerabilities.
- **User Support**: Offer helpdesk or ticketing system for issue tracking.
- **Performance Tuning**: Analyse and improve slow modules or processes.
- **Feedback Loop**: Collect user feedback and update system features accordingly.

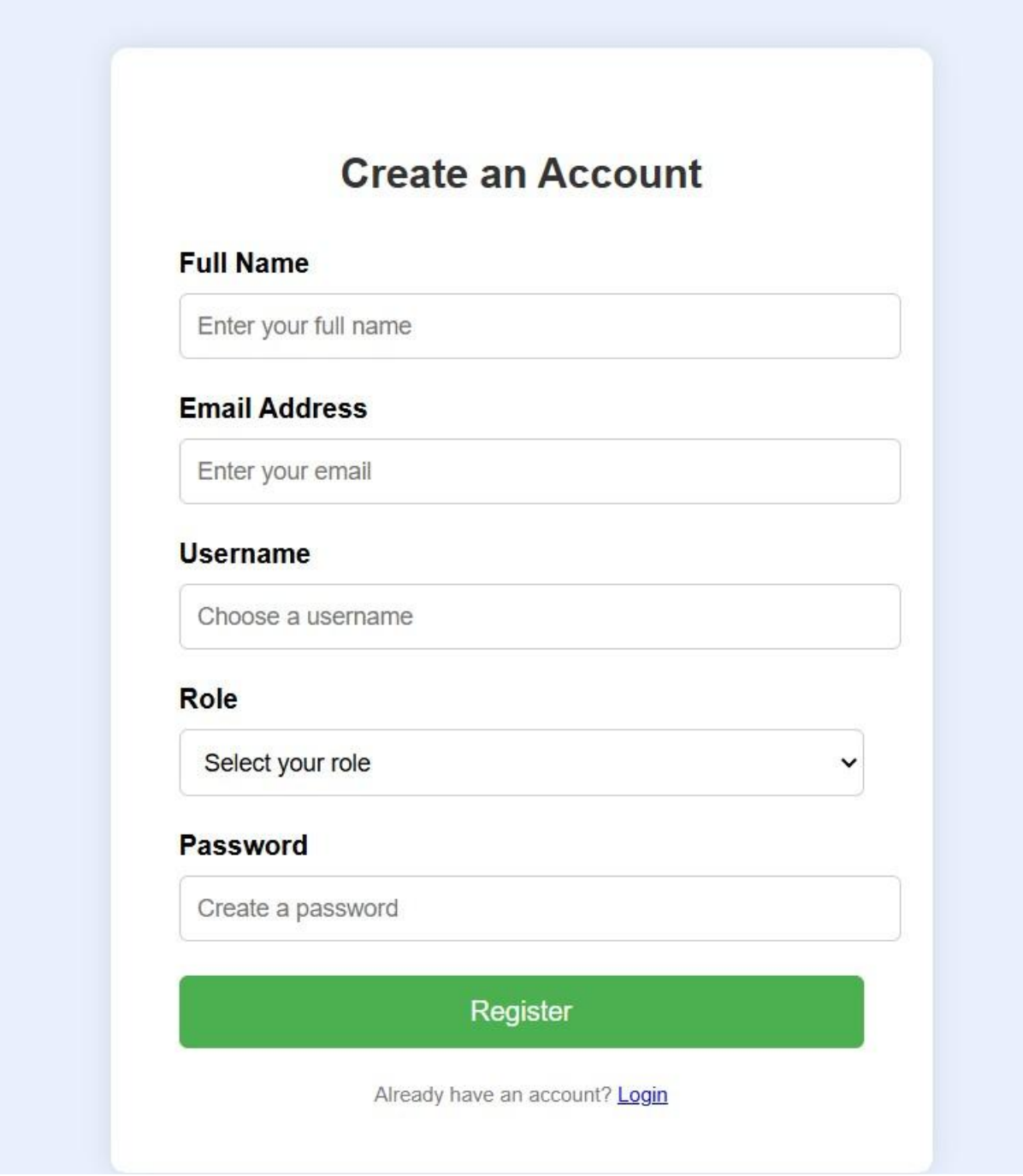
5.3 Maintenance Schedule

Task	Frequency
Server and Database Backup	Daily
Bug Fixes & Minor Updates	Weekly
Security Audits	Monthly
Feature Enhancements	Quarterly
System Performance Review	Bi-annually

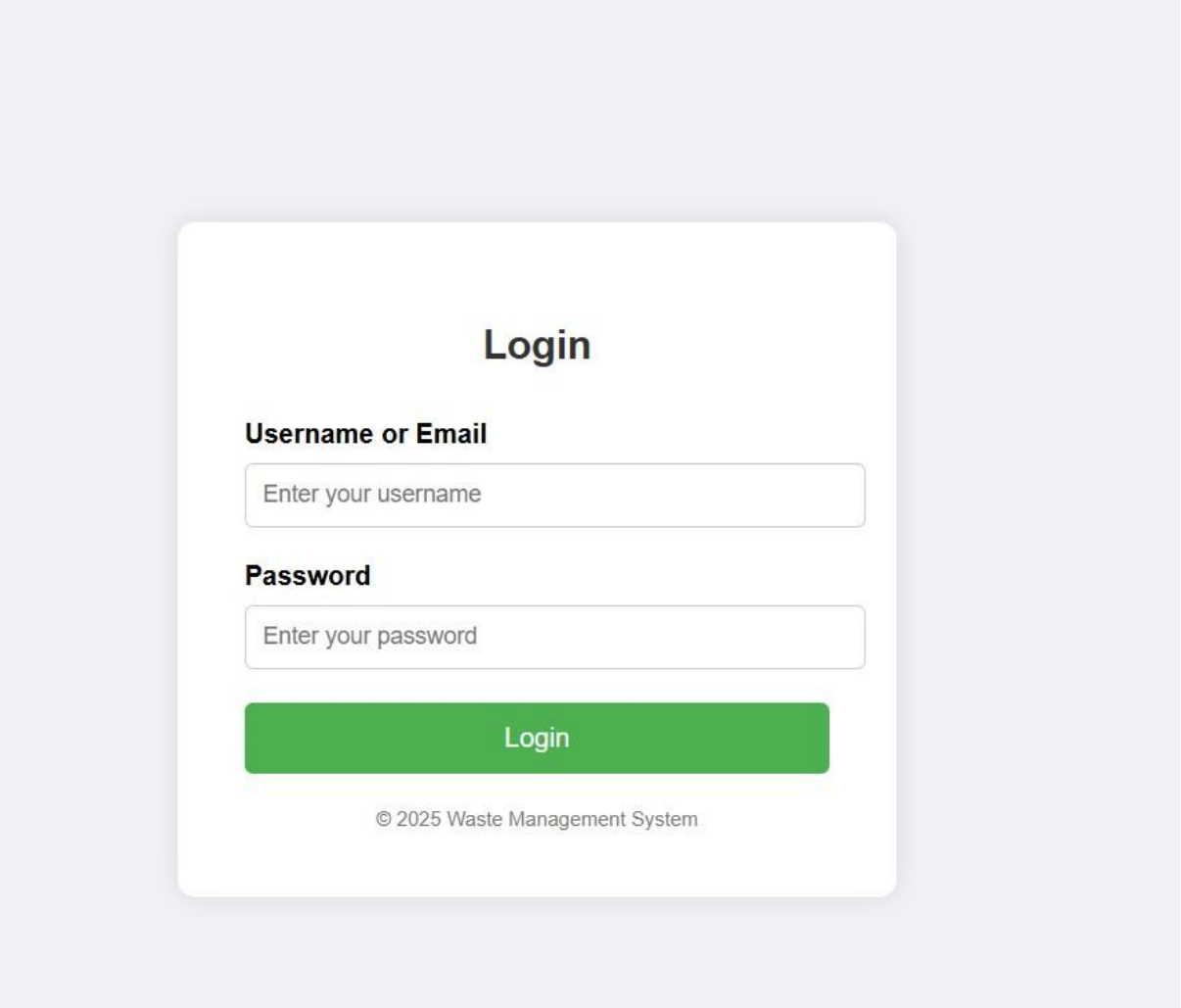
Chapter 6: User Manual

6.1 Home page

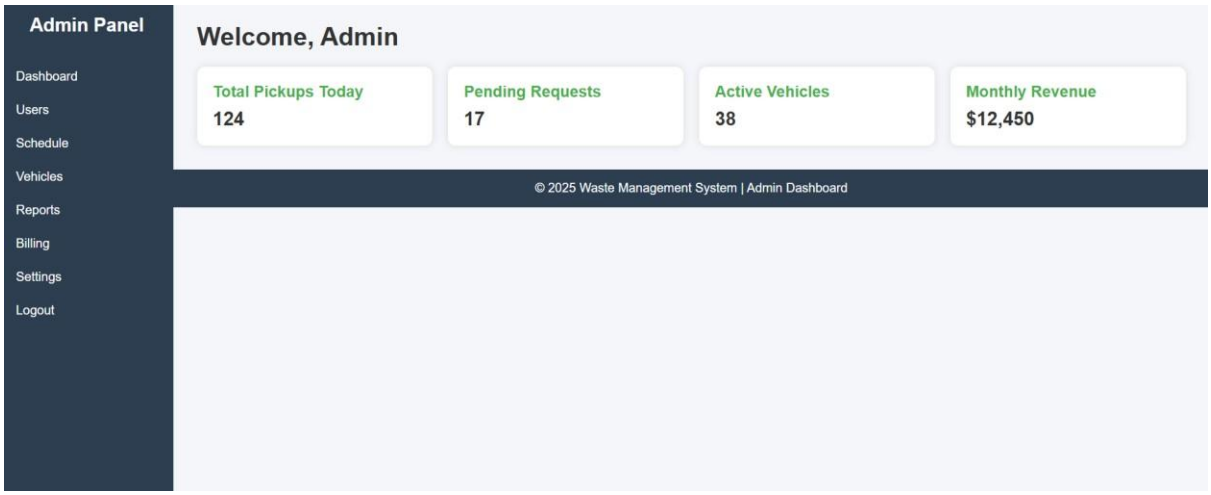
6.2 Registration



6.3 Login

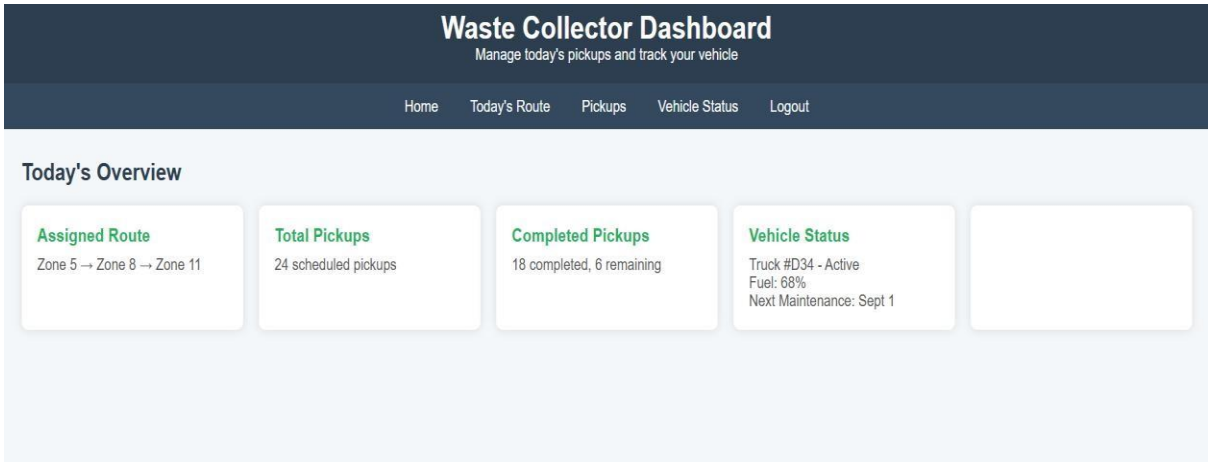

Login
Username or Email
Enter your username
Password
Enter your password
Login
© 2025 Waste Management System

6.4 Admin Dashboard

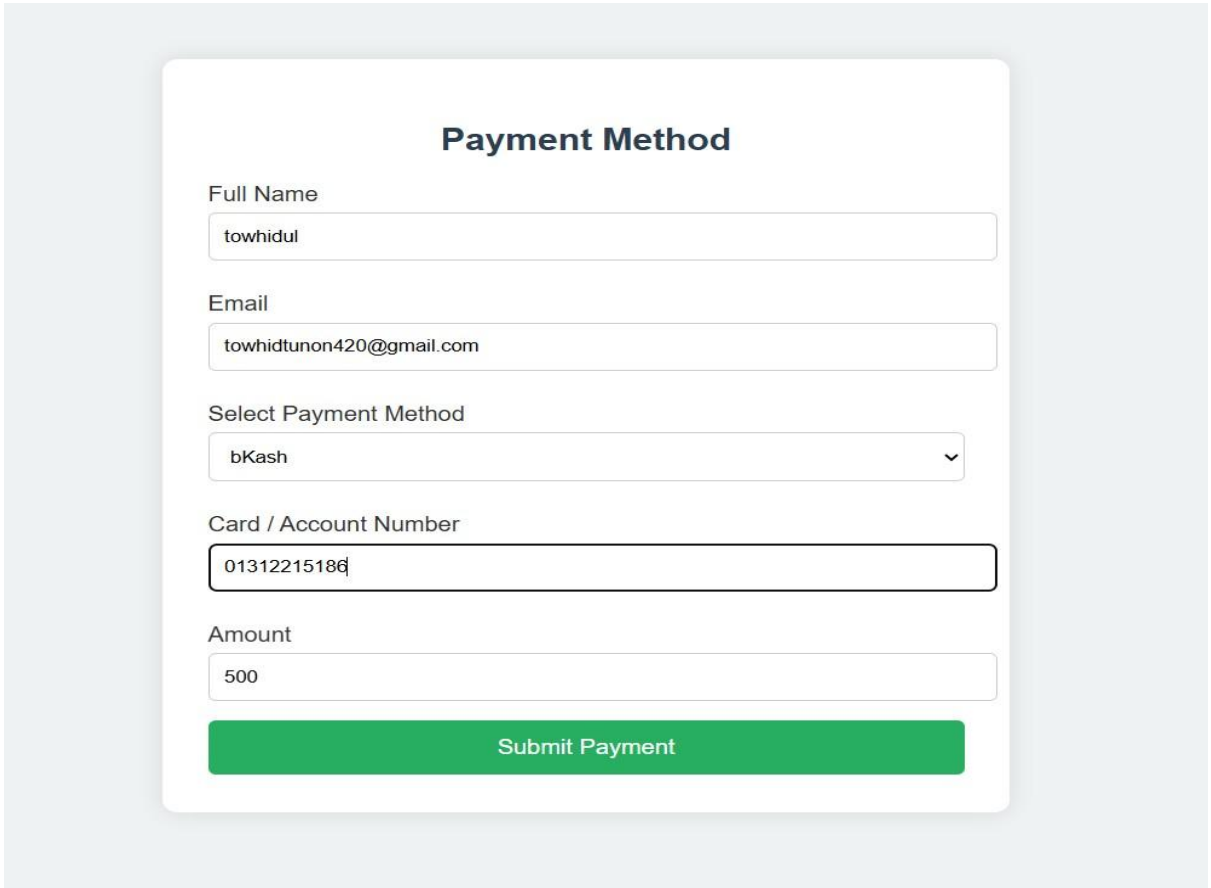


6.5 Resident User Dashboard

6.6 Waste Collector Dashboard



6.7 Payment



Chapter 7: Project Summary

7.1 Overview

The Waste Management System is a digital platform that's all about making the whole process of handling household and community trash easier and more efficient. it's trying to connect the dots between the locals, the folks picking up the trash, and the city officials with a one-stop-shop approach users can ask for pickups, check where they are, and pay for them; admins and collectors can handle jobs and keep an eye on how the system's doing. this system is all about keeping things neat, working well, and being kind to the planet when we deal with trash.

Key features include:

1.A user-friendly Waste Pickup Request module.

2.Real-time Request Tracking for residents.

3.Admin Dashboard for task assignment and monitoring.

4. Collector Panel for keeping tabs on what's picked up and managing tasks.

7.2 Achievements

The system does a great job at doing what was planned, like:

1.Signing up and logging in with different access levels (Resident, Admin, Collector).

2. A cool feature where you can set up when your trash gets picked up from your place.

3. keeping an eye on when residents get their trash picked up in real time.

4. A dashboard for keeping tabs on users, picking out collectors, and keeping an eye on what's going on.
5. Keep an eye on your tasks and get confirmation when they're picked up.

6.Setting up a database to smoothly store and pull-out user and request info.

7.3 Limitations

Despite its success, the system has a few limitations:

1.Roles are set in stone and can't be changed on the fly.

2.People who collect stuff are just given their tasks without thinking about the best way to get around.

3.Users don't get any email or text message alerts when there's new stuff or changes.

4.Admins can't make reports that you can download or see the nitty-gritty analytics details.

7.4 Future Enhancements

To make the system better and more user-friendly, here are some cool upgrades we should think about-

1.Dynamic Role Management: Let users make and tweak their own roles right from the admin dashboard.

2.Make waste pickup routes smarter by using mapping tools to plan the best paths.

3.Hey, just a heads up - your pickup's confirmed, or there's a delay/change. Keep an eye on your messages.

4. Add cool visuals, track how much people use it, and make it easy to get the admin stuff out.
5. Make a mobile app so people can use it and collect stuff more easily.

6.Make it easier for people who speak other languages to use our stuff in different places.

7. Set up a way to pay online for those special or planned pickups.

7.5 Conclusion

This waste management system is a smart and effective project that makes handling trash easier and more efficient through digital means. It sets up a clear way for folks living there, the ones collecting stuff, and the people running things to talk to each other. The system's easy-to-use design and smart handling of info help make our surroundings cleaner and our city more intelligent. even though there's room for improvement, this version is solid and sets us up for better stuff down the line and when we actually use it in the real world.

CHAPTER 8: REFERENCES

1. World Bank. (2023). What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050.

Retrieved from: https://datatopics.worldbank.org/what-a-waste/

2. Bangladesh Ministry of Environment and Forests. (2022). National 3R Strategy for Waste Management. Dhaka, Bangladesh.
3. Firebase Documentation. (2024). Firebase Authentication, Fire store, and Hosting. Retrieved from: https://firebase.google.com/docs
4. W3Schools. (2024). HTML, CSS, JavaScript, and Responsive Web Design Tutorials. Retrieved from: https://www.w3schools.com
5. OpenAI. (2024). Use of AI Tools for Code Generation and Report Writing. ChatGPT, OpenAI.
6. Bootstrap. (2024). Frontend Framework for Responsive Web Design. Retrieved from: https://getbootstrap.com/
7. Google Maps API. (2024). Location and Route Mapping API Documentation. Retrieved from: https://developers.google.com/maps/documentation
8. Stack Overflow. (2024). Community Answers on Web Development and Bug Fixes. Retrieved from: https://stackoverflow.com/https://developers.google.com/maps/documentation
9. Department of Urban Development, Bangladesh. (2021). Smart City Waste Management Guidelines. Dhaka, Bangladesh.

Plagiarism Report

212-35-751

ORIGINALITY REPORT

15%	13%	1%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	9%
2	Submitted to National Institute of Business Management Sri Lanka Student Paper	1%
3	Submitted to University of Greenwich Student Paper	1%
4	Submitted to Arab Open University Student Paper	1%
5	Submitted to ESoft Metro Campus, Sri Lanka Student Paper	1%
6	Submitted to RMIT University Student Paper	1%
7	Submitted to University of Wales Institute, Cardiff Student Paper	1%
8	Submitted to Dublin Business School Student Paper	<1%
9	Submitted to Kingston University Student Paper	<1%
10	www.webasha.com Internet Source	<1%
11	journal.50sea.com Internet Source	<1%
12	Submitted to HELP UNIVERSITY Student Paper	<1%

Account Clearance