

Smart Waste Tracker: A Web-Based Waste Complaint and Pickup System for Smart Cities

Asst. Prof. Vaibhav C. Patil¹, Mansi Kiran Jadhav²

¹Assistant Professor, Master of Computer Applications

²Students, Master of Computer Applications

Sant Gajanan Maharaj College of Engineering, Mahagaon, Shivaji University, Kolhapur, India.

vaibhavpatil8743@gmail.com¹, mansijadhav72003@gmail.com²

Abstract: *Waste management is one of the key challenges for urban municipalities in India. Citizens lack a proper digital channel to report waste issues, and administrators do not have any centralized system to manage complaints and track resolution. This paper presents Smart Waste Tracker, a web-based platform that enables citizens to register waste complaints, schedule garbage pickup, and track complaint status in real time. The system is developed using Angular for the frontend and ASP.NET Core Web API for the backend, with Microsoft SQL Server as the database. JSON Web Token (JWT) is used for secure authentication and role-based access control. Key functionalities include complaint registration with photo upload, pickup scheduling with time slot selection, real-time complaint status tracking through a three-stage workflow, automated email notifications via Gmail SMTP at every stage, worker assignment by the administrator, and an analytics reports module with CSV export. The platform was tested across all modules and all core features performed correctly. SmartWaste Tracker provides a practical, scalable, and cost-effective digital solution for urban waste management that can be directly deployed in smart city environments. Keywords: Waste Management, Smart City, Angular, ASP.NET Core Web API, SQL Server, JWT Authentication, Email Notification, Complaint Tracking, Pickup Scheduling*

Keywords: *Waste management*

I. INTRODUCTION

Rapid urbanization in Indian cities has created serious urban waste management challenges. As city populations grow, the volume of municipal solid waste increases considerably, putting pressure on civic authorities to manage collection, disposal, and citizen complaints efficiently. In many Indian municipalities, citizens continue to report waste issues through informal channels such as phone calls, in-person visits to ward offices, or social media posts. These channels are unreliable, untracked, and often fail to produce timely responses.

The Government of India has launched the Smart Cities Mission to promote technology-driven urban governance. Digitizing waste management is one of the core pillars of this mission. A dedicated digital platform for waste complaint management can bring transparency, accountability, and efficiency to an area of civic service that directly affects the daily lives of citizens.

Existing approaches to digital waste management often focus on specific sub-problems such as route optimization for collection vehicles or IoT-based bin monitoring. Few systems provide an integrated citizen-facing platform that covers the complete workflow from complaint submission through resolution, while also supporting pickup scheduling and administrator management.

Smart Waste Tracker addresses this gap. It is a full-stack web application that allows citizens to report waste issues, schedule garbage pickups, and monitor complaint status. Administrators can manage all complaints, assign field



workers, update statuses, and view analytics. The system sends automated email notifications at every key workflow stage, keeping citizens informed without requiring them to follow up manually.

This paper proposes Smart Waste Tracker as a complete, deployable solution for digitizing urban waste complaint and pickup management in the Indian smart city context.

II. RELATED WORK

Several researchers and organizations have explored digital approaches to urban waste management and civic complaint systems, each providing valuable insights for the design of Smart Waste Tracker.

Singh and Kumar [1] studied mobile-based complaint management systems for municipal services across Indian cities. Their study found that digital complaint submission increased complaint volume by 40 percent and reduced average resolution time by approximately one-third, indicating strong citizen demand for accessible digital reporting channels.

Patel et al. [2] developed an IoT-based smart bin monitoring system that triggers collection alerts when bins reach capacity. While effective for pre-emptive collection scheduling, this approach requires hardware sensor installation at every bin location, making it expensive and difficult to scale. SmartWaste Tracker achieves similar transparency through a software-only approach accessible to any citizen with a web browser.

Sharma and Verma [3] proposed a GIS-integrated

III. LITERATURE REVIEW

1. Mobile-Based Municipal Complaint System

Authors: R. Singh, A. Kumar

Explanation:

This study examined how mobile-based complaint systems improved municipal service delivery.

Digital submission channels reduced paperwork and ensured complaints were logged and tracked automatically. The study highlighted the importance of automated status updates for citizen satisfaction.

Additional Issues: The study did not address garbage pickup scheduling or worker assignment workflows.

2. IoT Smart Bin Monitoring for Urban Municipalities

Authors: S. Patel, R. Mehta, P. Joshi

Explanation:

This paper presented an IoT-based system for monitoring garbage bin fill levels and triggering collection alerts. The system improved collection efficiency but required costly sensor hardware at each bin location and did not provide a citizen-facing complaint interface. waste tracking and route optimization platform for Indian urban local bodies. Their work demonstrated the value of location-based data in improving collection efficiency. Smart Waste Tracker captures location data through manual address entry, making it accessible without requiring GPS hardware.

Gupta [4] reviewed e-governance implementations across Indian smart cities and concluded that successful citizen adoption depends on interface simplicity, reliable notification mechanisms, and trust-building feedback loops. These findings directly influenced the design priorities of Smart Waste Tracker, particularly the multi-stage email notification system and the visual complaint status timeline.

Reddy and Nair [5] developed a web-based civic issue reporting platform and observed that email confirmation at submission significantly increased user trust and return engagement. This finding motivated the complaint acknowledgement email that Smart Waste Tracker sends immediately after each complaint is submitted.

Additional Issues: High hardware cost and absence of citizen complaint features limit applicability in low-budget municipalities.



3. GIS-Integrated Waste Tracking Platform Authors: D. Sharma, P. Verma Explanation:

The authors proposed a GIS-based platform for tracking garbage collection routes and optimizing vehicle deployment. While effective for back-end operations, the platform lacked a citizen portal for complaint submission or status tracking.

Additional Issues: Requires GIS infrastructure and does not address citizen engagement or real-time notifications.

4. E-Governance and Citizen Service Delivery

Authors: N. Gupta

Explanation:

This review analyzed e-governance implementations in Indian smart cities and identified the key factors behind successful citizen adoption. Simplicity of interface, multilingual support, and reliable notifications were identified as the most critical success factors.

Additional Issues: The review was limited to existing implementations and did not propose a new system design.

IV. PROBLEM STATEMENT

Urban waste management in Indian cities suffers from several persistent operational problems.

Citizens currently have no standardized digital channel to report waste issues. Complaints submitted through phone or informal means are frequently unacknowledged, leaving residents with no confirmation that their concern has been received.

Once a complaint is submitted, there is no mechanism for citizens to track its progress. This lack of visibility reduces trust in civic authorities and leads to repeated follow-up calls that burden already stretched municipal support teams.

Garbage pickup scheduling does not exist in most municipalities outside large metropolitan areas.

Residents with large or unusual volumes of waste have no way to arrange a specific pickup without visiting a ward office in person.

Administrators lack consolidated data on complaint volumes, resolution rates, and recurring problem locations. Without this data, evidence-based decisions about resource deployment and infrastructure investment are not possible.

Field worker assignment is informal and unstructured in most municipal setups, making accountability for complaint resolution difficult to establish or audit.

SmartWaste Tracker addresses all five of these gaps through a unified web-based platform that provides complaint registration, real-time status tracking, pickup scheduling, worker assignment, and analytics reporting within a single application.

V. PROPOSED SYSTEM OVERVIEW

SmartWaste Tracker is a web-based waste complaint and pickup management system that connects citizens and administrators on a single integrated platform. The system supports two user roles: Citizen and Administrator, each with a dedicated dashboard and set of functionalities.

A citizen can create an account, log in securely, submit a waste complaint with title, waste type, location, priority, description, and an optional photo, schedule a garbage pickup with preferred date and time slot, monitor complaint and pickup status through a real-time tracking dashboard, and receive automated email notifications at each stage of the workflow. An administrator can log in to a dedicated management dashboard, view and manage all citizen complaints and pickup requests, update complaint status through the Pending, In Progress, and Resolved workflow stages, assign field workers to specific complaints, view analytics reports with complaint volume charts and resolution statistics, and export complaint data to CSV for offline reporting.

All data is persisted in a Microsoft SQL Server database. The backend is implemented as a RESTful API using ASP.NET Core 8.0. The frontend is built with Angular 17 as a Single Page Application. JWT tokens issued at login are



validated on every API request through an HTTP interceptor, ensuring secure and role-appropriate access throughout each session.

VI. SYSTEM ARCHITECTURE

The Smart Waste Tracker system follows a simple three-tier architecture, which includes the presentation tier, application tier, and data tier. Each layer has its own role and works together smoothly.

The presentation tier is developed using Angular 17. It manages user interaction through standalone components, while reactive forms are used for input validation. RxJS helps in handling asynchronous communication with the backend, and Chart.js is used to display data visually in the form of bar and doughnut charts on the dashboard.

The application tier is built using ASP.NET Core 8.0 Web API. It handles all client requests, applies business logic, and communicates with the database using Entity Framework Core. It also includes an EmailService that sends HTML-based notifications at important stages, and CORS is enabled to connect the Angular frontend with the backend. The data tier uses Microsoft SQL Server to store system data in tables such as Users, Complaints, PickupRequests, and Workers. Entity Framework Core manages database operations and schema through a structured ApplicationDbContext. Overall, this architecture ensures that the system is organized, efficient, and easy to maintain.

1. Modules

Smart Waste Tracker is organized into seven main modules. Each module is designed to handle specific functionalities and ensure smooth operation of the complete system.

1. Authentication Module

This module handles citizen registration and login. JWT tokens are issued on successful authentication and validated on every protected API request. Role-based routing directs citizens to the user dashboard and administrators to the admin dashboard. A Forgot Password endpoint delivers the account password to the registered email address.

2. Complaint Registration Module Citizens submit waste complaints through a structured form. Eight waste type categories are available for selection. On submission, an acknowledgement email is sent to the citizen and a new complaint notification is sent to the administrator. The complaint is stored with an initial Pending status.

3. Pickup Request Module

Citizens can schedule garbage pickup by selecting waste type, entering location, choosing a preferred date with minimum one-day advance booking enforced, and selecting a preferred time slot from five available options. Pickup requests are stored in the Pickup Requests table and visible to the administrator.

4. Complaint Tracking Module

The tracking page displays all complaints submitted by the logged-in citizen with their current status shown on a visual three-stage timeline. Filter tabs allow citizens to view complaints by status. A summary panel shows total complaints, resolution rate, pickup count, and average response time.

5. Admin Dashboard and Worker Assignment Module

The administrator dashboard provides statistics cards, bar and doughnut charts, and tabbed complaint management tables. Administrators can update complaint status and assign field workers from the Workers table. Worker assignment triggers an email notification to the affected citizen.

6. Email Notification Module

The Email Service class sends HTML-formatted emails at five trigger points: registration, complaint submission, administrator new complaint alert, worker assignment, and status resolution. Emails are delivered through Gmail SMTP on port 587 with SSL encryption.

7. Reports Module

The reports page displays complaint analytics through Chart.js bar and doughnut charts, a detailed complaint table with all fields, and a pickup requests summary table. Administrators can export complaint data to CSV and print the report page.



Summary statistics include total complaints, resolved count, pending count, resolution rate, and registered citizens.

2. Backend Architecture (ASP.NET Core + SQL Server)

The system uses ASP.NET Core Web API services to manage authentication, data storage, and communication:

- JWT Authentication: Ensures secure login and role-based identity verification for Citizens and Administrators.
- SQL Server Database: Stores structured data related to users, complaints, pickup requests, workers, and status history.
- Gmail SMTP Email Service: Sends HTML-formatted email notifications for complaint submission, worker assignment, status updates, and password reset.

3. Approval and Notification Complaint Workflow:

- Citizen submits a waste complaint through the web portal.
- System stores complaint with Pending status and sends acknowledgement email to citizen.
- Administrator receives new complaint notification email.
- Administrator reviews complaint and assigns a field worker.
- System updates status to In Progress and notifies citizen.
- Worker resolves the issue; Admin marks complaint Resolved.
- System sends resolution notification email to citizen.
- Status updates are reflected in the citizen's tracking dashboard in real time.

VII. IMPLEMENTATION DETAILS

The implementation of Smart Waste Tracker consists of four main stages: User Registration and Authentication, Complaint and Pickup Submission, Administrator Management and Worker Assignment, and Notification and Reporting.

User Registration and Authentication

Citizens register by providing their full name, email address, mobile number, address, and password. The registration endpoint stores the user record in the SQL Server Users table and sends a welcome email. The login endpoint validates credentials against the stored data and returns a JWT token along with the user profile. The token is stored in the browser's local storage and attached to all subsequent API requests through an Angular HTTP interceptor. An Auth Guard component protects all routes, redirecting unauthenticated users to the login page.

Complaint and Pickup Submission

Citizens submit waste complaints through a structured form capturing a title, waste type from eight categories, pickup location, priority level, description, and an optional photo attachment.

On submission, an acknowledgement email is sent to the citizen and a new complaint notification email is sent to the administrator. Pickup requests capture waste type, location, preferred date with a minimum one-day lead time enforced, and preferred time slot from five available options.

Administrator Management and Worker Assignment

The administrator dashboard presents statistics cards, complaint and status charts, and tabbed data tables for complaints, pickup requests, and citizen management. For pending complaints, the administrator selects an available field worker from a dropdown listing all active workers.

Assigning a worker updates the complaint status to In Progress, records the worker name, and triggers an assignment notification email to the citizen.

Notification and Reporting

The Email Service component sends HTML-formatted emails at five workflow events: welcome on registration, acknowledgement on complaint submission, new complaint notification to the administrator, worker assignment



notification to the citizen, and resolution notification when a complaint is marked Resolved. The reports module renders Chart.js visualizations for complaint status distribution and waste type breakdown. Administrators can export all complaint data to CSV and print reports using the browser print API.

SYSTEM AECHEITURE

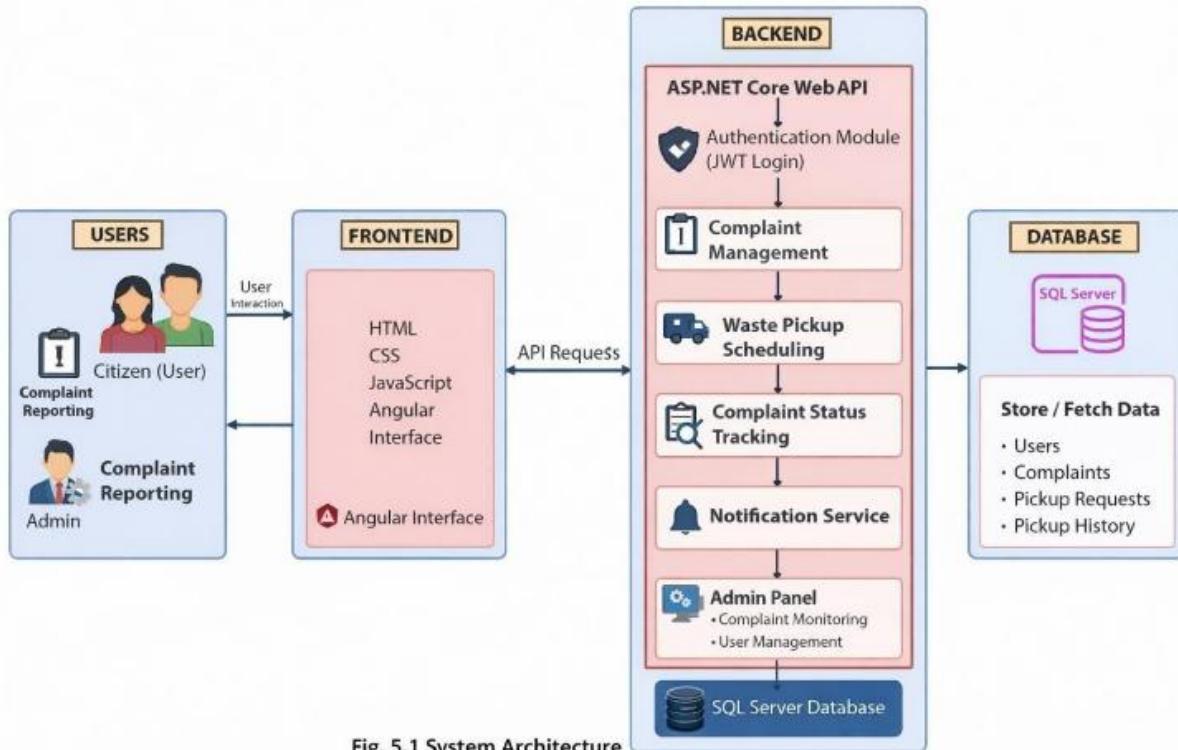


Fig. 5.1 System Architecture

Figure 1: System architecture

VIII. PROPOSED SYSTEM

System Architecture:

The system architecture of Smart Waste Tracker is shown in Figure 1.

Our proposed system works in the following steps:

Step 1: User Registration

Citizens create an account by entering their name, email address, mobile number, and home address. Once registered, they can log in and use all features of the platform.

Step 2: Login and Authentication

The system checks the entered credentials and verifies the user through JWT-based secure authentication. After login, the user is taken to their personal dashboard based on their role.



Step 3: Complaint Submission

Citizens fill out a complaint form with details like waste type, location, priority, and description. They can also attach a photo of the waste issue. Once submitted, an email is sent to both the citizen and the administrator.

Step 4: Pickup Scheduling

Citizens can request a garbage pickup by selecting the waste type, entering their address, choosing a preferred date, and selecting a time slot. The request is saved and shown to the administrator.

Step 5: Admin Review and Worker Assignment

The administrator opens the complaint from the dashboard, reads the details, and selects a suitable worker from the list. Once assigned, the complaint status changes to In Progress and the citizen gets an email notification.

Step 6: Status Update and Resolution

After the field worker completes the work, the administrator updates the complaint status to Resolved. The citizen receives a final email notification confirming that their complaint has been resolved.

IX. ANALYSIS OF PROPOSED SYSTEM

1. Transparency and Accountability

Every complaint in Smart Waste Tracker gets its own unique ID as soon as it is submitted. The complaint then passes through three clear stages —

Pending, In Progress, and Resolved. At each stage, the citizen gets an email update, so they always know what is happening with their complaint. This kind of clear record is not possible when complaints are handled over the phone.

2. Easy to Expand

Smart Waste Tracker is built in separate modules, and each module works on its own. If we want to add new features in the future, like sending SMS alerts or showing complaints on a map, we can do that without changing the existing parts of the system. This makes it easy to grow the platform step by step as the city's needs increase.

3. Saves Time for Administrators

Everything the administrator needs is available in one dashboard. They can manage complaints, check pickup requests, assign workers, view citizen details, and see reports — all from the same screen. They do not need to use separate tools or maintain paper records. Worker assignments are also saved in the database, so there is a clear record of who handled each complaint.

4. Helps in Better Decision Making

The reports section shows charts and statistics about complaints, including how many are resolved, how many are still pending, and which type of waste problems occur most often.

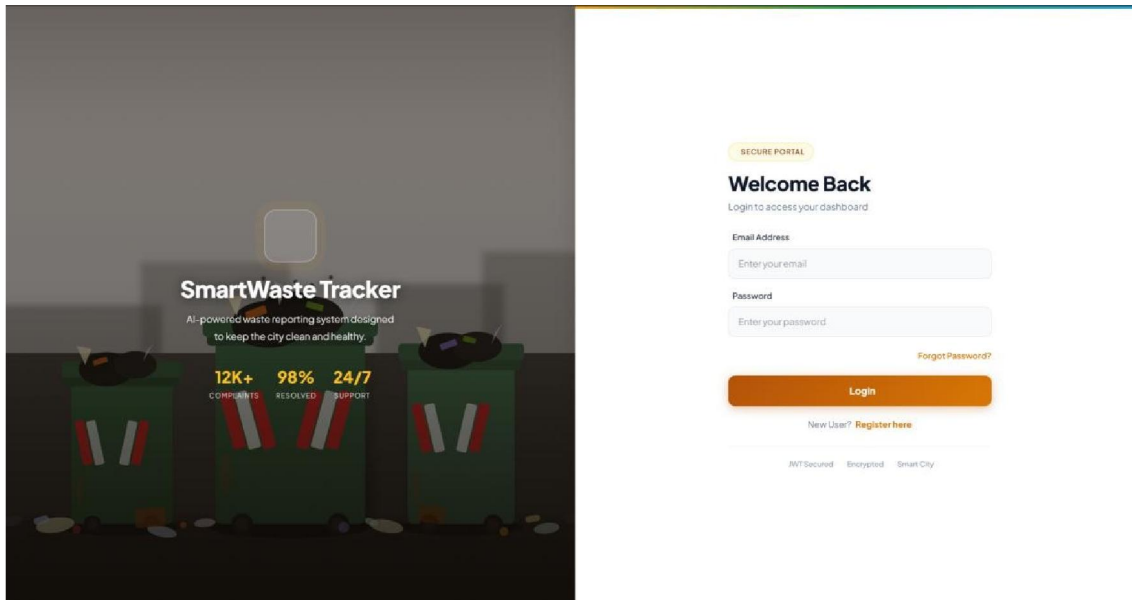
Administrators can use this information to decide where to send more workers and which areas need better waste collection services. The data can also be downloaded as a CSV file for further analysis.

5. Encourages Citizens to Participate

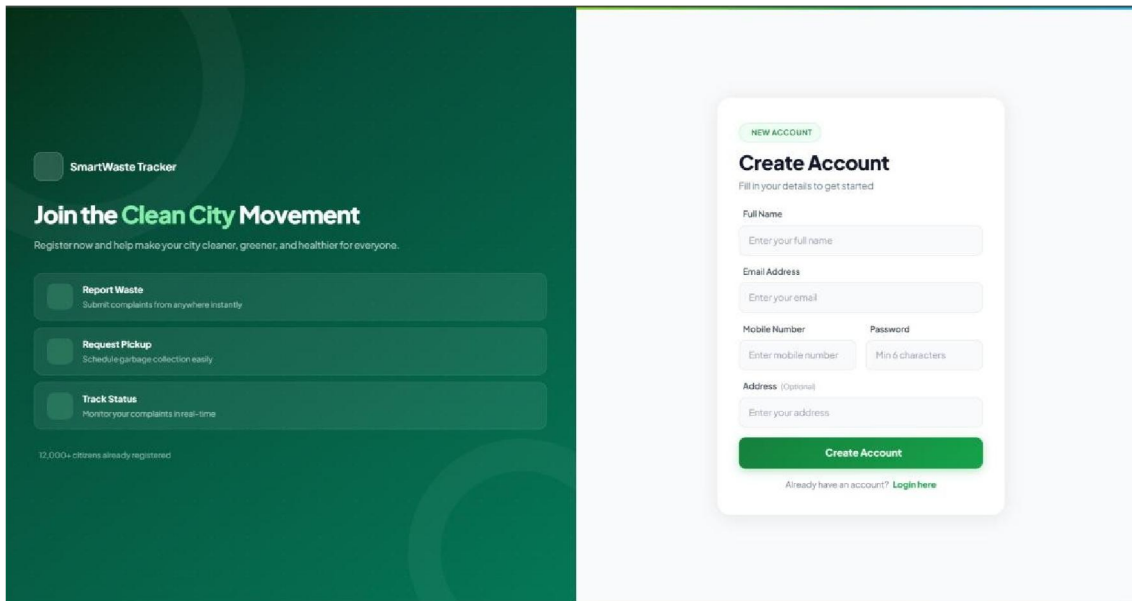
When citizens see that their complaint was received and acted upon, they feel confident in the system. Getting email updates at every step makes them feel informed and valued. This encourages more people to report waste issues regularly, which gives the city better data and helps keep the streets cleaner.



X. RESULTS

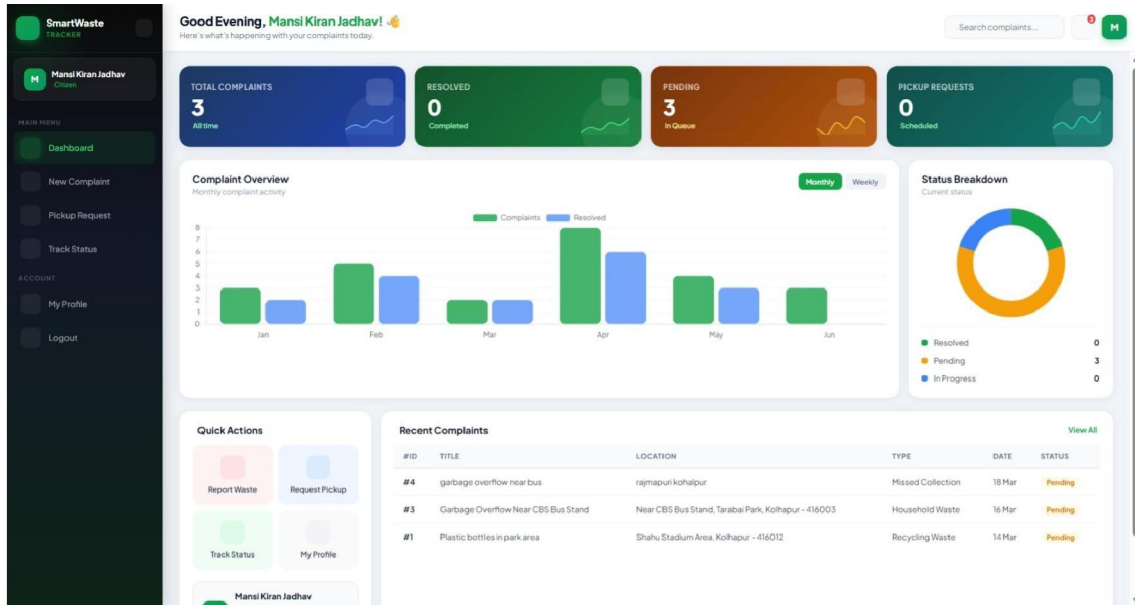


LOGIN PORTAL USER LOGINS & ADMIN LOGINS

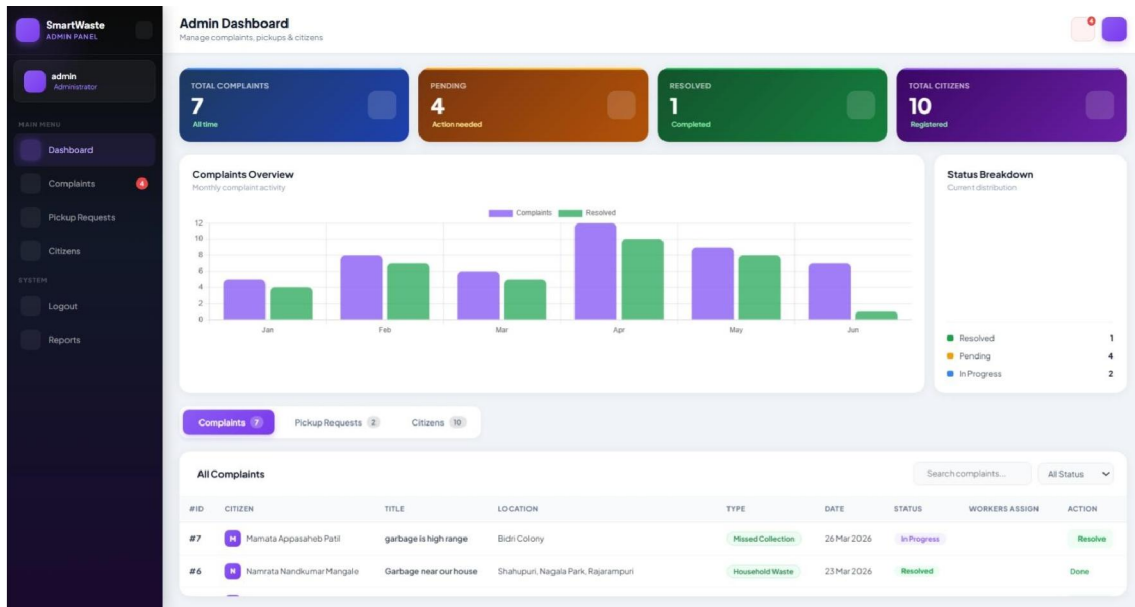


REGISTRATION PAGE





USER DASHBOARD



ADMIN DASHBOARD

X. CONCLUSION

Smart Waste Tracker demonstrates that a modern web application built using Angular and ASP.NET Core can effectively digitize the end-to-end workflow of urban waste complaint and pickup management. The platform eliminates the need for informal, phone-based complaint channels and replaces them with a transparent, accountable, and data-driven digital system.

The seven integrated modules address the five operational gaps identified in urban waste management: inaccessible reporting, lack of status visibility, absence of pickup scheduling, no analytics capability, and unstructured worker



assignment. End-to-end testing confirmed that all modules function correctly and that the Gmail SMTP email notification system delivers communications reliably at every workflow stage.

The JWT-based security model ensures that each user role accesses only the functionality appropriate to their position, providing a secure multi-user environment. The modular architecture allows new features to be added without affecting existing functionality.

Future work will focus on three areas: developing a companion mobile application for Android and iOS to enable complaint submission from any device, integrating geolocation-based mapping to visualize complaint hotspots on an interactive city map, and deploying the platform on Microsoft Azure to support multi-city operation at scale.

REFERENCES

1. R. Singh and A. Kumar, "Mobile-Based Complaint Management for Municipal Services: A Study of Citizen Adoption and Operational Impact," *International Journal of E-Government Research*, vol. 17, no. 2, pp. 34–49, 2021.
3. S. Patel, R. Mehta, and P. Joshi, "IoT- Based Smart Waste Bin Monitoring System for Urban Municipalities," *International Journal of Internet of Things and Cloud Computing*, vol. 9, no. 1, pp. 12–22, 2022.
1. Microsoft ASP.NET Core Documentation — Provides complete information about ASP.NET Core Web API, middleware, Entity Framework Core, and JWT authentication. <https://learn.microsoft.com/en-us/aspnet/core>
2. Angular Developer Documentation — Official guide for building Angular applications, components, services, HTTP client, and routing. <https://angular.io/docs>
3. SQL Server Documentation — Complete reference for Microsoft SQL Server database management, queries, and configuration. <https://learn.microsoft.com/en-us/sql/sql-server>
4. Visual Studio Code Documentation — Editor guide for Angular development, debugging, and extensions. <https://code.visualstudio.com/docs>
5. A. Freeman — *Pro ASP.NET Core 8*, 9th Edition, Apress, 2023. This book explains ASP.NET Core Web API development, middleware pipeline, Entity Framework Core, and JWT authentication implementation.
6. N. Murray, F. Coury, A. Lerner, C. Taborda — *ng-book: The Complete Guide to Angular, Fullstack.io*, 2023. Covers Angular component architecture, reactive forms, HTTP client, routing, and service injection patterns.

