**Jurnal Studi Multidisiplin Ilmu**
*ISSN: 3026-7412, Vol 2, No 3, 2024, 49-64* https://doi.org/10.35912/jasmi.v2i3.6912

# SIMOSA: Design and Implementation of a Waste Monitoring Application for Integrated Urban Waste Management

**Zhafran Rafi Al Rasyid[1*], M. Said Hasibuan[2]**
Institut Informatika dan Bisnis Darmajaya, Bandar Lampung, Indonesia[1,2]
zhafranrafiar.1911010046@mail.darmajaya.ac.id[1*], msaid@darmajaya.ac.id[2]

**Article History**
Received on 8 July 2024
1st Revision on 26 July 2024
2nd Revision on 10 August 2024
3rd Revision on 26 August 2024
Accepted on 4 September 2024

**Abstract**
**Purpose**: Urban waste accumulation poses a persistent environmental governance challenge across Indonesian cities, driven by rapid population growth, insufficient collection infrastructure, and the absence of demand-responsive coordination mechanisms linking waste generators to available collection services.
**Methodology**: This study applies the Prototype development methodology to design and implement *Sistem Monitoring Sampah* (SIMOSA), an Android-based mobile application developed using Android Studio with Java, Firebase Authentication and Realtime Database, and Google Maps SDK. Development stages encompassed problem identification, data collection through observation and interviews, stakeholder communication, quick planning, interface modeling, prototype construction, and Black Box functional testing.
**Results**: SIMOSA was successfully implemented as a dual-role Android application supporting Customer (waste generator) and Driver (waste collector) roles within a single installation. Black Box testing confirmed 100% pass rates across all six core features: Login, Registration, Menu Selection, Map/Pickup Location Input, Driver Matching, and Order Completion.
**Conclusions**: SIMOSA provides a functional, accessible technological solution for urban waste pickup coordination, reducing informal disposal by enabling on-demand requests through a straightforward mobile interface with Google Maps and Firebase real-time synchronization.
**Limitations**: The application is currently limited to Android; iOS deployment is absent. Real-time driver location tracking during transit is incomplete, and formal security penetration testing has not been conducted.
**Contributions**: SIMOSA contributes a replicable Android, Firebase, and Google Maps architecture for community-level waste pickup coordination adaptable to diverse urban waste management contexts across Indonesian cities.

**Keywords:** *Android Application, Firebase, Mobile Application, SIMOSA, Waste Management*

**How to Cite:** Rasyid, Z. R. A., & Hasibuan, M. S. (2024). SIMOSA: Design and Implementation of a Waste Monitoring Application for Integrated Urban Waste Management. *Jurnal Studi Multidisiplin Ilmu*, 2(3), 49-64.

## 1. Introduction

Urban waste management remains one of the most pressing and consequential environmental governance challenges facing Indonesian municipalities today. The World Bank reports that solid waste production from urban areas reaches approximately 2.01 billion tones per year globally, and this volume is projected to increase substantially as an estimated 68 to 75 percent of the world's population will reside in urban areas by 2050 (Rapii, 2021). In the Indonesian context, the consequences of insufficient waste management infrastructure are already visible: informal dumping sites proliferate in residential neighborhoods, drainage channels become clogged with discarded materials, and communities experience increased health risks from decomposing organic waste accumulating near residential zones (Elamin, Ilmi, Arifin, & Siswanto, 2018). The sheer scale of this challenge demands not only improvements in physical collection infrastructure but also fundamental changes in the coordination mechanisms that connect waste generators with collection services.

Household sources contribute approximately 49.37% of total waste volume in Indonesia, making residential communities the single largest waste category and the most critical target for improved collection coordination (Hasibuan, Febriani, & Subing, 2023). Traditional municipal collection services in Indonesian cities typically operate on fixed schedules that cannot respond to the heterogeneous, unpredictable waste generation rhythms of individual households. This temporal mismatch between supply and demand creates gaps during which waste accumulates in informal locations, encouraging behaviors such as burning, dumping in drainage channels, or stockpiling waste in public spaces (Budiarto, 2021). Addressing this coordination failure requires not only policy intervention but technological innovation that makes responsible waste disposal as convenient and accessible as informal alternatives.

The proliferation of Android smartphones across Indonesian urban communities has created an unprecedented technological opportunity to bridge this coordination gap. As of recent years, Indonesia ranks among the largest Android user markets in Southeast Asia, with smartphone penetration reaching across income groups in urban centers (Febriana, Susilowati, & Pramono, 2022). This connectivity infrastructure, when combined with cloud-based real-time database services and geolocation capabilities, enables the development of demand-responsive waste coordination platforms that were not feasible even a decade ago. The model is well-established in urban transportation: ride-hailing platforms such as Gojek and Grab demonstrated that mobile applications can effectively coordinate supply and demand for on-demand services across large, complex urban environments. Extending this coordination architecture to waste collection represents a natural and technically feasible application of proven technology (Marzuki, Purnomo, & Hidayat, 2024).

Several prior Android-based civic applications have explored technology-mediated approaches to community waste management and demonstrated encouraging preliminary results. The E-Pulung application developed by Febriana et al. (2022) for waste bank digitalisation in Magelang confirmed that mobile interfaces can significantly increase community participation in responsible waste disposal when they reduce the friction of engaging with formal collection services. Marzuki et al. (2024) demonstrated through a web-based campus waste bank application that digital monitoring enables real-time waste flow tracking and eliminates manual paper-based recording overhead. These precedents validate the architectural premise of SIMOSA while also identifying gaps: existing applications tend to focus either on waste bank deposit management or on static information delivery, rather than on dynamic, real-time coordination of on-demand pickup between individual waste generators and available drivers.

International literature similarly underscores the transformative potential of mobile technology in waste management. Studies from multiple countries document that digitally enabled waste management systems consistently outperform traditional scheduled-collection models in terms of community engagement, collection efficiency, and environmental outcomes, particularly when applications are designed with intuitive interfaces that minimize the behavioral effort required to participate (Saud et al., 2022; Yadav et al., 2021; Kumar & Samadder, 2017). The integration of real-time geolocation services into waste management applications represents a particularly significant advancement, enabling dynamic matching of pickup requests with available drivers rather than relying on pre-defined collection zones and fixed schedules (Pardini et al., 2020; Chowdhury et al., 2023).

This study designs and implements SIMOSA (Sistem Monitoring Sampah), an Android application that enables community members to request waste pickup at a specified location, matches their request to available drivers in real time through Google Maps integration and Firebase database synchronization, and provides customers with tracking visibility of the approaching driver. The system architecture supports a dual-role design within a single Android application installation, enabling users to function as either Customers (waste generators requesting pickup) or Drivers (waste collectors accepting and fulfilling requests) depending on their current role and capacity. The development followed the Prototype methodology as described by Pressman (2012), enabling iterative refinement based on stakeholder feedback before final implementation.

This study makes three specific contributions. First, it demonstrates the design and implementation of a dual-role single-application architecture for waste pickup coordination on the Android platform. Second, it provides empirical evidence through Black Box testing of the functional correctness of the Google Maps SDK and Firebase Realtime Database integration for waste pickup coordination. Third, it offers a replicable architectural template for community-level waste management applications applicable to other Indonesian urban contexts facing similar coordination challenges.

## 2. Literature Review

### *2.1 Urban Waste Management: Problem Context and Technology Opportunities*

Waste is defined as residual material from both domestic and industrial production processes, encompassing solid or liquid materials that are no longer functionally useful to their original producer (Harapan, 2018). The relationship between population growth and waste volume is direct and well-documented across urban contexts worldwide: as cities grow, waste generation increases proportionally unless matched by equivalent expansion in collection capacity and management infrastructure (Alfath, 2018). In Indonesian urban settings, inadequate waste management generates cascading consequences including environmental contamination of water bodies and soil, increased disease vectors from rotting organic matter, aesthetic deterioration of public spaces, and erosion of social cohesion as communities disagree about informal disposal practices.

The primary management challenge in Indonesian urban waste is coordination rather than technology or resources alone. Municipal collection services operate on fixed routes and schedules that cannot respond to the heterogeneous, unpredictable waste generation patterns of individual households and businesses. Community members who engage in informal dumping typically do so not from disregard for environmental consequences, but because no practical, readily accessible collection service is available at the moment their waste accumulates beyond manageable household storage (Budiarto, 2021). This behavioral observation has important design implications: successful waste management technology must make responsible disposal more convenient and accessible than informal alternatives, rather than simply providing information or awareness without addressing the practical coordination barrier.

Research from urban planning and environmental management literature consistently identifies on-demand digital coordination as one of the most promising architectural approaches to this challenge. Pardini et al. (2020) demonstrated through a systematic literature review of smart waste management systems that mobile application-mediated coordination reduces informal dumping by an average of 34% in pilot urban contexts compared to scheduled-only collection. Chowdhury et al. (2023) further found that real-time geolocation integration is the single most important feature for driver efficiency, reducing average travel time to pickup locations by enabling direct navigation rather than fixed-route driving. These findings directly motivate the Google Maps SDK integration in SIMOSA's design.

IoT-based waste monitoring systems have also demonstrated significant promise in international literature. Sensor-equipped smart bins that communicate fill levels to central management platforms enable dynamic, data-driven collection routing that minimises unnecessary trips to under-capacity bins while ensuring overfull containers are addressed before they create health and aesthetic problems (Setiawan et al., 2023; Yadav et al., 2021). However, IoT implementations require significant hardware investment in sensor-equipped bins and supporting network infrastructure, creating adoption barriers that mobile application-only approaches do not face. SIMOSA's approach prioritises accessibility by requiring only a standard Android smartphone, eliminating hardware procurement requirements for community adoption.

### *2.2 Android Mobile Application Development*

Android is a Linux-based mobile operating system that integrates three functional layers, such as the core operating system, the middleware layer providing platform services, and the application layer where user-facing functionality is implemented. Android applications developed in Java or Kotlin can access device hardware capabilities including GPS, camera, network connectivity, and notification services through the Android SDK, providing a comprehensive development framework for complex civic applications (Holla & Katti, 2012). Android's open-source licensing model and global market penetration make it the dominant platform for civic technology applications in developing markets including Indonesia, where Android devices account for the majority of smartphone ownership across income groups (Sibuea, Sembiring, & Ginting, 2022).

Android Studio, Google's official Integrated Development Environment for Android development, provides a comprehensive toolset including the Gradle build system, a visual layout editor, a built-in emulator for testing without physical devices, code completion and refactoring tools, and direct integration with GitHub for version control (Sibuea et al., 2022). For SIMOSA's development, Android Studio's emulator capability was particularly important during the design and prototyping phases, enabling iterative testing of interface designs and feature implementations without requiring repeated deployment to physical devices. The IDE's integration with Firebase and Google Maps SDK through Gradle dependency management also significantly streamlined the backend service integration process, reducing configuration overhead relative to manual SDK integration approaches.

The selection of Java as SIMOSA's primary development language reflects both the project team's competency profile and the extensive documentation and community support available for Java-based Android development, which remains the most widely documented Android development language despite Kotlin's growing adoption as Google's preferred language for new Android projects. Kotlin's interoperability with Java would enable future migration of the SIMOSA codebase without complete redevelopment (Safitri, Putra, & Wulandari, 2022).

### *2.3 Firebase as Application Backend*

Firebase, developed and maintained by Google, provides a comprehensive backend-as-a-service infrastructure for mobile and web applications, enabling developers to implement complex backend functionality including user authentication, real-time data synchronisation, push notifications, and cloud storage without custom server development (Maulana, 2017). This backend-as-a-service model is particularly valuable for civic technology applications where development teams lack the resources or expertise to maintain dedicated server infrastructure while still requiring enterprise-grade reliability and scalability for community-facing services.

Firebase Authentication provides SIMOSA's user identity management through email and password credential verification, with support for account creation, password reset, and administrative account management through the Firebase console. The Authentication module's integration with Firebase Realtime Database enables role-based data access control, ensuring that customer pickup request data is accessible to drivers in the matching process while remaining protected from unauthorised access Maulana, 2017. This security architecture is fundamental to maintaining user trust in a civic application handling location data.

Firebase Realtime Database provides SIMOSA's core data synchronisation infrastructure through a cloud-hosted JSON database that enables all connected clients to receive data updates in real time without polling. When a Customer submits a pickup request, the request data is written to the Firebase Realtime Database, where it becomes immediately visible to all active Driver clients through a persistent connection, triggering the driver matching process without perceptible latency (Ichwan, Husni, & Rasyid, 2021). This real-time property is what distinguishes Firebase from conventional REST API database architectures for SIMOSA's use case: the waste pickup coordination workflow requires sub-second synchronisation between customer and driver interfaces to provide a smooth user experience that competes with the convenience of informal disposal alternatives.

Push notifications through Firebase Cloud Messaging (FCM) provide supplementary real-time communication for order status updates, alerting drivers to new nearby pickup requests and notifying customers when a driver accepts their request even when the application is not in the foreground. This background notification capability is essential for practical usability, as waste pickup requests may be submitted at times when drivers are not actively monitoring the application interface (Pramana, Widodo, & Santoso, 2022).

### *2.4 Google Maps SDK Integration*

The Google Maps SDK for Android provides SIMOSA with comprehensive geolocation functionality including interactive map display, GPS-based device location detection, custom map markers for pickup location pinning, and turn-by-turn navigation capability (Rahman, Setiawan, & Fathoni, 2022). For the waste pickup coordination use case, map integration serves three distinct functional purposes: enabling customers to visually specify their pickup location by pinning a map marker at their address or a nearby accessible point; enabling the system to display the geographic positions of available drivers relative to the customer's request; and enabling drivers to navigate directly to accepted pickup locations through Google's established navigation infrastructure.

The Maps SDK's Places API integration enables address autocomplete functionality in the customer location input, reducing input errors from manual address typing and ensuring that pickup location coordinates are accurate and geocodable for driver navigation. Location accuracy is critical for SIMOSA's effectiveness: inaccurate pickup location data would result in drivers being navigated to incorrect locations, creating service failures that would undermine community trust in the application (Nugroho, Prasetya, & Kusumaningrum, 2021; Sinaga, Napitupulu, & Situmorang, 2021).

### *2.5 Prototype Development Methodology*

The Prototype methodology, as described by Pressman (2012), is a system development approach that creates working functional models for stakeholder evaluation before complete system implementation. The methodology's iterative evaluation structure enables requirements to be refined and the application design to be improved based on user feedback at each prototype stage, substantially reducing the risk of building a complete system that fails to match actual user needs and workflows (Maulida, 2022). For civic technology applications such as SIMOSA that serve diverse community users with varying levels of technical literacy, this iterative refinement capability is particularly valuable: interface designs that seem intuitive to developers may present significant usability barriers to community users unfamiliar with mobile application conventions (Susanti, Wibowo, & Rahmawati, 2022).

The Prototype methodology comprises sequential stages that structure the development process while preserving flexibility for design iteration. Communication involves structured stakeholder interviews to identify functional requirements and usage scenarios. Quick Planning defines the technical architecture and resource requirements to support those requirements. Modeling Quick Design translates requirements into interface mockups and workflow diagrams for stakeholder review. Prototype Construction implements the design in functional software. Deployment and Feedback exposes the functional prototype to user evaluation, generating insights for refinement in subsequent cycles (Pressman, 2012; Kurniati, 2021). Fadillah et al. (2019) documented that Prototype methodology consistently produces information systems with higher user satisfaction scores than waterfall-only approaches when applied to user-facing civic applications, validating its selection for SIMOSA's development.

***2.6 Prior Empirical Studies***

Table 1. Summary of prior studies on Mobile Applications for waste management and monitoring

| Author(s) & Year | Application / Setting | Method / Platform | Key Finding Relevant to Waste Monitoring App Development |
|---|---|---|---|
| Maghfiroh et al. (2022) | E-Pulung Android app for waste bank digitalisation, Magelang | Android, prototype methodology | Android-based waste bank application digitalises waste collection and exchange; app mediation significantly reduces informal waste dumping; community participation increases when technology interface is intuitive. |
| Marzuki et al. (2024) | Campus waste bank web application | Website, prototyping | Web-based campus waste bank app integrates collection, recording, and reporting; digital monitoring enables real-time waste flow tracking; reduces manual paper-based recording overhead. |
| Hasibuan et al. (2023) | Integrated waste bank community empowerment | Community-based, information system | Household waste constitutes 49.37% of total waste; integrated waste management systems connecting communities with collection drivers reduce accumulation; digital coordination improves collection efficiency. |
| Alfath (2018) | Waste management IT, Palangka Raya | IT utilisation study | Information technology significantly improves waste management in urban contexts; digital systems enable more responsive collection scheduling and route optimisation. |
| Rapii (2021) | Community-based integrated waste management, Rumbuk Village | Community empowerment study | Integrated waste management combining technology and community participation most effectively reduces waste accumulation; technology bridges gap between waste generators and collection services. |
| Budiarto (2021) | Technology for economic-value waste management | Technology utilisation discussion | Technology adoption for waste management requires addressing community mindset; mobile apps can incentivise responsible waste disposal by making it more convenient than informal dumping. |
| Setiawan et al. (2023) | Smart waste monitoring IoT-based system | IoT, sensor-based, Android | IoT-based sensor integration with Android applications enables automated bin fill-level monitoring; significantly reduces unnecessary collection trips and optimises driver routing. |
| Kurniati (2021) | Document archiving system, Lais District Office | Prototype methodology | Prototype methodology effectively aligns system output with stakeholder needs through iterative evaluation; reduces post-deployment revision requirements significantly. |
| Present Study (2024) | SIMOSA app, waste monitoring for urban community, Android | Prototype methodology, Android Studio, Firebase, Google Maps SDK | SIMOSA enables Customer-Driver waste pickup coordination; 6 core features all pass Black Box testing; green UI design; Firebase real-time database; dual user-role (Customer/Driver) in single application. |

Table 1 summarizes prior empirical studies most directly relevant to SIMOSA's design rationale and technical approach. These studies collectively establish the evidence base for mobile application-mediated waste management coordination, validate the technical feasibility of the Firebase and Google Maps integration approach, and identify gaps in existing applications that SIMOSA's dual-role design addresses. The studies collectively confirm several design principles that shaped SIMOSA's development. First, mobile interface accessibility is the most critical determinant of community adoption: applications that reduce the behavioral effort of responsible waste disposal consistently achieve higher participation rates than those requiring significant user effort or technical skill. Second, real-time data synchronization is essential for coordination-type applications where both customer and driver experiences depend on immediate response to state changes. Third, the Prototype methodology enables superior requirement alignment for community-facing applications compared to waterfall development approaches. SIMOSA extends this evidence base by combining all three capabilities in a single dual-role application targeting Indonesian urban communities.

## 3. Research Methodology

### *3.1 Development Methodology: Prototype Approach*

The Prototype development methodology was employed for SIMOSA's design and implementation. This methodological choice was motivated by two primary considerations: the iterative refinement capability of the Prototype approach aligns with the diversity of the target user base (community members with varying mobile application experience), and the methodology's emphasis on functional prototype evaluation enables early identification of usability issues before full system completion. The following stages were implemented for the SIMOSA project.

Problem Identification documented the urban waste accumulation problem in Indonesian cities, specifically the temporal and logistical gap between household waste generation and scheduled municipal collection, and the resulting prevalence of informal disposal behaviors. Data Collection combined three complementary approaches: direct observation of waste disposal behavior and informal dumping patterns in residential neighborhoods; structured interviews with prospective users including community members and waste collection workers to understand workflow requirements and usage constraints; and systematic literature review of existing waste management technology solutions and Android application development approaches. Communication involved structured stakeholder interviews with both user segments to define functional requirements, role-specific workflows, and critical performance expectations for the pickup coordination features.

Quick Planning specified the software and hardware requirements needed to support the defined functional requirements, including the selection of Firebase over alternative backend services based on real-time synchronization capability, and Google Maps SDK over alternative mapping services based on navigation accuracy and local coverage in Indonesian urban areas. Modeling Quick Design developed interface mockups for all nine application screens, using a consistent green color palette selected to reinforce the application's environmental mission. Prototype Development implemented the designed application in Android Studio using Java, Firebase backend services, and Google Maps SDK integration. Deployment and Testing subjected the completed prototype to Black Box functional testing across all six core application features.

### *3.2 Technology Stack*

Table 2. SIMOSA application technology stack

| Component | Technology | Function in SIMOSA |
|---|---|---|
| Programming language | Java / Kotlin | Android application logic and user interface control. |
| Development IDE | Android Studio | App building, emulation, Gradle build management, GitHub integration. |
| Database / backend | Firebase (Realtime Database, Authentication) | Real-time user data storage; Firebase Authentication for email/password login; push notifications. |

| Component | Technology | Function in SIMOSA |
|---|---|---|
| Map / geolocation | Google Maps SDK | Pickup location pin; driver real-time tracking; route visualisation between customer and driver. |
| UI design tools | Canva | Application logo and icon design. |
| Target platform | Android (64-bit OS) | Mobile deployment; minimum spec: 8 GB RAM, Intel Gen 2 processor for development environment. |

Table 2 describes the complete technology stack selected for SIMOSA's implementation. Each technology component was selected to meet specific functional requirements identified during the Quick Planning stage, with selection decisions documented below the table. The SIMOSA's technology stack was composed of six integrated components covering every layer of the application architecture from development environment to deployment platform. Java was selected as the primary programming language given the development team's existing competency base and the extensive documentation available for Java-based Android development. Android Studio provides the complete development environment including the Gradle build system for dependency management, which was used to integrate Firebase and Google Maps SDK libraries without manual configuration. Firebase was selected as the backend service provider based on its free tier availability for development-scale projects, its native integration with Android Studio, and critically, its Realtime Database service that provides sub-second synchronisation required for the pickup coordination workflow. Google Maps SDK was selected as the geolocation provider based on its accuracy, coverage of Indonesian urban areas, and availability of the Places API for address autocomplete. Canva was used for UI asset creation including the application logo and icon, enabling professional-quality visual design without requiring dedicated graphic design resources (Ardiyansyah, Purwanti, & Setiawan, 2023).

### *3.3 Application Architecture: Dual-Role Design*

SIMOSA's core architectural decision is the dual-role design, which implements both the Customer and Driver functional workflows within a single Android application installation. This design choice eliminates the operational complexity of maintaining two separate applications while enabling role-specific interface views and functionality within the same codebase. After authentication through Firebase, the Menu screen presents authenticated users with a binary choice between the Customer and Driver roles, routing them to role-specific functional screens.

This architecture mirrors the design pattern used successfully by Indonesian ride-hailing platforms such as Gojek and Grab, which SIMOSA conceptually extends from transportation to waste collection coordination. The advantage of the unified application over a two-application architecture is particularly significant in the Indonesian urban context, where community members who generate waste during the week may also participate as collection drivers during weekends or off-hours, making role flexibility within a single application practically valuable rather than merely architecturally convenient.

The Customer workflow proceeds through the following sequence: the user authenticates via Login, selects the Customer role at the Menu screen, accesses the Map screen to enter pickup location and disposal destination, submits the pickup request through the PESAN PICK UP button which writes the request to Firebase Realtime Database, views the Driver Search screen showing an animated map while the system identifies an available driver, transitions to the real-time tracking screen when a driver accepts the request displaying the driver's current map position, and concludes at the Order Completed screen when the driver completes the pickup. The Driver workflow proceeds: the user authenticates via Login, selects the Driver role at the Menu screen, views available pickup requests from Customer users in the area, accepts a request which triggers navigation guidance to the pickup location via Google Maps, executes the physical pickup, and marks the order as complete. Firebase Realtime Database synchronises these two workflows in real time, ensuring that Customer requests are immediately visible to Drivers and that Driver acceptance is immediately reflected in the Customer's tracking interface (Yuliani, Prasetyo, & Hidayatullah, 2023).

### *3.4 Interface Design: Screen Specifications*

Nine interface screens were designed during the Modeling Quick Design stage, each corresponding to a functional step in the Customer or Driver workflow. The Welcome screen serves as the application entry point, featuring the SIMOSA logo and navigation animation depicting a masked driver using the application, with Login and Register navigation buttons. The Login screen provides email and password input fields with Firebase Authentication integration and error handling for invalid credentials. The Registration screen collects full name, email, password (minimum six characters), and password confirmation with Firebase account creation on successful submission. The Menu screen presents the Customer and Driver role selection buttons with clear visual differentiation to prevent accidental role selection. The Customer Map screen provides a full-screen Google Maps interface with address search input, pickup location pin functionality, destination input, and the PESAN PICK UP submission button. The Driver Search screen displays an animated map visual representing the geographic search for available drivers in the area. The Driver Matched screen shows the driver's real-time location on the map with distance and estimated arrival information. The Order Completed screen confirms successful pickup completion with a PESAN LAGI button enabling immediate new request submission. The Firebase Admin Web Database interface provides administrators with user account management capability outside the Android application itself (Wahyuningsih, Setiawan, & Prabowo, 2023).

All screens use a dominant green color palette, selected as a symbolic representation of environmental awareness and ecological cleanliness consistent with the application's waste management mission. This color-coding also provides a clear visual identity that distinguishes SIMOSA from general-purpose utility applications on a user's device home screen, reinforcing the application's environmental purpose at every point of interaction (Nasution, 2021).

### *3.5 Testing Methodology: Black Box Testing*

Black Box testing was employed to validate SIMOSA's functional correctness across all six core application features. Black Box testing evaluates system outputs against expected outputs for defined inputs without reference to the internal implementation logic, confirming that each functional feature behaves as specified from the user's perspective (Wijaya, 2021). This approach is well-suited to validation of Android application functional flows because it mirrors the actual user experience of navigating through application screens and performing actions, verifying both the correctness of individual features and the integrity of the transitions between them. Six test cases were defined, each corresponding to one of the six core features identified during the Communication and Quick Planning stages: Login, Registration, Menu Selection, Map/Pickup Location Input, Driver Matching, and Order Completion.

## 4. Results and Discussions

### *4.1 SIMOSA Application Implementation*

The SIMOSA was successfully implemented as a functional Android application using Android Studio, Firebase backend services, and Google Maps SDK for location services. The application's dominant green visual identity, applied consistently across all screens through the UI color scheme, reinforces the application's environmental mission and creates a distinctive brand identity appropriate to its waste management function. The application operates through internet connectivity, enabling real-time data synchronisation between Customer requests and Driver responses through Firebase Realtime Database. The implementation required approximately twelve weeks from initial design through prototype completion, following the Prototype methodology's sequential stages with iterative refinement of the interface design based on stakeholder feedback during the Quick Design review stage.

The Firebase Authentication module manages user identity through email and password credential management, with the Firebase administrator web console enabling administrative user management including manual account addition and deletion, login method configuration, and account status monitoring. This administrative capability ensures that the application operator can maintain control over platform access and respond to security or account issues without requiring in-app administrative functionality in the Android application itself. The separation of end-user functionality from administrative functionality in different interfaces reduces complexity in the mobile application while preserving necessary oversight capability for platform operators (Ichwan, Husni, & Rasyid, 2021).

The Google Maps SDK integration was the most technically complex component of the SIMOSA implementation, requiring configuration of API credentials through the Google Cloud Console, enabling the relevant Maps and Places APIs, and integrating the SDK into the Android Studio project through Gradle dependency management. The geolocation accuracy of the Maps SDK in the Bandar Lampung test environment was confirmed during prototype testing, with address search returning accurate results for residential addresses in the target deployment area. Firebase Realtime Database rules were configured to allow authenticated users to read and write pickup request data while preventing unauthenticated access, providing baseline data security for the coordination workflow (Rahman, Setiawan, & Fathoni, 2022).

### *4.2 Core Feature Implementation*

#### *4.2.1 Welcome, Login, and Registration*

The welcome screen serves as the application entry point with a navigation animation that functions as an ambient introduction to the application's purpose. Two clearly labeled navigation options, Login and Register, direct users to the appropriate subsequent screen based on their account status. The Login screen collects email and password inputs, with Firebase Authentication handling credential verification and routing authenticated users to the Menu screen. Invalid credential combinations trigger an inline error message without navigating away from the Login screen, enabling users to correct input errors without restarting the login flow. The Registration screen collects four required fields including full name, email address, password with a minimum six-character requirement, and password confirmation. Firebase creates a new user account upon successful form completion and assigns a unique user identifier that persists across sessions and links user profile data to their pickup request history in the Realtime Database. The registration flow includes client-side validation for password minimum length and password-confirmation matching before submission to Firebase, reducing unnecessary API calls for predictably invalid inputs and providing faster feedback to users during account creation. This validation approach reflects established Android form design best practices for civic applications where user onboarding friction must be minimized to encourage adoption (Safitri, Putra, & Wulandari, 2022).

#### *4.2.2 Pickup Request and Driver Coordination*

The Customer Map screen, powered by the Google Maps SDK, enables customers to specify their waste pickup location by searching an address using the Places API autocomplete feature or by directly pinning a location on the map interface. The map renders at street-level zoom for the user's current detected GPS location upon screen launch, providing immediate spatial context for pickup location selection. After specifying the pickup location and disposal destination, the customer submits the request through the *PESAN PICK UP* button, which writes the complete request record including location coordinates, timestamp, and customer user identifier to the Firebase Realtime Database.

Request submission triggers a transition to the Driver Search screen, which displays an animated map visual representing the geographic search process while Firebase queries for available drivers in the area. When a Driver accepts the request, the Customer's interface transitions automatically to the real-time tracking screen, which displays the Driver's current location as a moving marker on the Google Maps interface. The Driver's location updates in the tracking view as the Driver navigates toward the pickup location, providing the Customer with visibility into the approach progress. Upon the Driver marking the pickup as complete, both the Customer and Driver interfaces transition to the Order Completed screen, which confirms the successful transaction and presents the *PESAN LAGI* button for immediate new request submission (Nugroho, Prasetya, & Kusumaningrum, 2021).

The Driver-side workflow is designed to minimise the decision friction for waste collectors accepting requests. The available requests view presents incoming Customer pickup requests with key information including pickup location, approximate distance from the Driver's current position, and customer identity. One-tap request acceptance triggers immediate navigation guidance to the pickup location through Google Maps' established navigation infrastructure, eliminating the need for drivers to manually enter addresses or switch between applications during active pickup operations.

### *4.3 Black Box Testing Results*

Table 3 presents the complete Black Box testing results for SIMOSA's six core application features, showing the test procedure, expected outcome, and actual result for each tested feature. Testing was conducted at Institut Informatika dan Bisnis Darmajaya, Bandar Lampung, in 2024 using Android devices running current versions of the Android operating system. All six tested features achieved Pass results, confirming 100% functional specification compliance across the complete core application workflow. The Login and Registration test cases (Cases 1 and 2) validated Firebase Authentication's integration with the Android application, confirming that user credential management operates correctly in both directions: successful authentication routes users to the Menu screen, and successful registration creates Firebase accounts accessible for subsequent login. The Menu Selection test case (Case 3) confirmed that the dual-role architectural design routes users to the correct role-specific screen upon selection, validating the core navigation logic of the dual-role design.

The most operationally critical test cases are Cases 4 and 5, which together validate the Google Maps SDK integration and Firebase Realtime Database synchronization that constitute SIMOSA's unique value proposition as a waste pickup coordination application. Case 4 confirmed that the Customer Map screen successfully captures pickup location inputs and writes them to the Firebase Realtime Database in a format accessible to Driver clients. Case 5 confirmed that the Driver matching process completes successfully and that the Customer's interface transitions to the real-time tracking view upon match confirmation, validating the end-to-end coordination workflow that distinguishes SIMOSA from simpler information delivery applications. Case 6 confirmed that the Order Completion workflow operates correctly from the Driver's completion action through to the Customer's order confirmation screen and the PESAN LAGI re-request functionality.

### *4.4 Discussion*

The SIMOSA's architecture addresses a specific coordination failure in urban waste management that prior research has identified but not fully solved through mobile technology: the temporal and logistical gap between household waste generation and available collection services. By enabling on-demand pickup requests through a familiar mobile interface, the application converts waste disposal from a passive, schedule-dependent activity into an active, user-initiated service interaction. This model transformation mirrors the disruption that Gojek and Grab introduced to urban transportation in Indonesia, demonstrating that mobile-mediated on-demand coordination can reshape service delivery patterns in sectors traditionally governed by fixed schedules and geographic zones. Hasibuan, Febriani, and Subing (2023) finding that household waste constitutes 49.37% of total urban waste directly motivates SIMOSA's focus on residential community users as the primary customer segment, establishing that improvements in residential waste coordination can have disproportionately large impacts on total urban waste volumes.

The Firebase Realtime Database architecture provides the synchronization infrastructure required for SIMOSA's coordination function. When a customer submits a pickup request, the data is immediately available to all active Driver clients through Firebase's persistent connection architecture, enabling real-time market-clearing between waste generators and collectors without perceptible latency. This real-time property distinguishes SIMOSA from scheduled-appointment waste management applications and makes it appropriate for the irregular, demand-driven nature of household waste accumulation. Ichwan, Husni, and Rasyid (2021) documented that Firebase Realtime Database achieves synchronization latencies below 200 milliseconds under normal network conditions, confirming that the real-time user experience promised by SIMOSA's design is technically achievable within the Firebase architecture.

The dual-role single-application design presents both architectural advantages and practical design challenges that merit discussion. The primary advantage is operational simplicity: community members can function as either Customers or Drivers within the same application without managing multiple installations or switching between applications. This flexibility is practically valuable in Indonesian urban communities where informal waste collection is often performed by individuals who supplement other income sources with occasional waste collection work, rather than by dedicated full-time drivers. The design challenge is role-switching clarity: users who primarily function as Customers must clearly understand and navigate to the Driver view if they choose to participate as collectors, without the role-specific branding that dedicated driver applications typically provide. SIMOSA addresses this through the Menu screen's clear CUSTOMER and DRIVER button design, but user acceptance testing with actual community members would be needed to confirm whether this interface design successfully prevents role confusion in real-world usage contexts.

Comparing SIMOSA's results with prior studies in similar domains reveals consistent themes. Febriana, Susilowati, and Pramono (2022) reported that E-Pulung's Black Box testing similarly achieved 100% pass rates across all tested features, while noting that real-world community adoption required additional effort beyond functional correctness to address user onboarding and habituation challenges. Marzuki, Purnomo, and Hidayat (2024) reported that their web-based waste bank application achieved high functional test coverage but identified integration complexity between the web interface and physical waste bank operations as a significant implementation challenge. These experiences suggest that SIMOSA's functional testing success, while necessary, is not sufficient to ensure real-world effectiveness: the critical next step is user acceptance testing with actual community members in the target deployment area to identify interface usability gaps that Black Box testing cannot reveal (Saud, Salim, & Haris, 2022).

The green UI color palette selected for SIMOSA represents a deliberate alignment between visual design and application purpose. Environmental psychology research consistently documents that green color schemes are associated with ecological awareness and positive environmental attitudes, suggesting that SIMOSA's color design may have subtle but measurable effects on users' sense of environmental purpose during application use (Pardini, Rodrigues, Kozlov, Kumar, & Furtado, 2020). While this effect is difficult to quantify in the current study, the deliberate alignment of visual design with application mission represents a thoughtful design decision that future user acceptance research could investigate more rigorously.

The limitations identified in testing, particularly the incomplete real-time driver location tracking during transit, represent the most significant gap between SIMOSA's designed functionality and its current implementation. The Customer tracking screen currently displays the driver's matched location rather than the driver's continuously updating GPS position during navigation, meaning that customers cannot observe driver approach progress in the current version. This limitation is technically straightforward to address within the existing Firebase Realtime Database architecture through periodic GPS coordinate writes from the Driver's device but was not implemented in the current prototype due to scope constraints. Completing this feature would substantially improve the customer experience and bring SIMOSA's tracking capability in line with the ride-hailing applications that established community expectations for real-time location visibility in on-demand coordination services (Kumar, & Samadder, 2017; Chowdhury, Saifuddin, & Islam, 2023).

## 5. Conclusions

### *5.1 Conclusion*

This study successfully designed and implemented SIMOSA, an Android-based waste monitoring and pickup coordination application, using the Prototype development methodology with Android Studio, Firebase Authentication and Realtime Database, and Google Maps SDK. The application delivers a functional dual-role architecture supporting both Customer and Driver workflows within a single Android installation, enabling community members to request on-demand waste pickup at specified locations and drivers to receive, navigate to, and complete those pickups with map integration. Black Box testing confirmed 100% pass results across all six core features encompassing Login, Registration, Menu Selection, Map and Pickup Location Input, Driver Matching, and Order Completion. SIMOSA addresses the coordination gap between household waste generation and formal collection services by providing an accessible, smartphone-mediated platform that makes responsible waste disposal more convenient than informal dumping, directly targeting the community behavioral dimension of urban waste accumulation in Indonesian cities. The Firebase and Google Maps architecture demonstrated through SIMOSA provides a replicable template for community-level waste management applications that can be adapted to diverse urban contexts across Indonesia without requiring custom server infrastructure or significant per-deployment technical customization.

***5.2 Research Limitations***

Three limitations apply to the current iteration of this study. First, SIMOSA is implemented exclusively for the Android platform, excluding iOS users from the current version and limiting the application's potential community reach in urban areas where iPhone adoption is significant among higher-income demographic segments. Second, real-time continuous driver location tracking during active pickup transit was not fully implemented in the current prototype; the Customer tracking screen displays the driver's matched location at the time of acceptance but does not update the driver's position continuously as navigation proceeds, reducing the tracking experience relative to mature ride-hailing applications. Third, the application's security posture has not been evaluated through formal penetration testing or independent security audit; the Firebase Authentication configuration and Realtime Database security rules may require strengthening to withstand adversarial access attempts before public deployment at community scale.

***5.3 Suggestions and Directions for Future Study***

Four development and research directions are recommended for advancing SIMOSA beyond its current prototype stage. The highest-priority technical enhancement is completing real-time continuous driver location tracking during active pickup transit. This feature is technically feasible within the existing Firebase Realtime Database architecture by implementing periodic GPS coordinate writes from the Driver device to a dedicated location node in the database, with the Customer tracking screen subscribing to updates on that node to refresh the driver marker position at regular intervals. Completing this feature would substantially improve the customer experience and close the most significant gap between SIMOSA's current functionality and the tracking experience that community members expect based on their experience with ride-hailing applications.

Second, cross-platform development extending SIMOSA to iOS should be evaluated as a medium-term enhancement to maximise potential community reach. A React Native or Flutter rebuild of the existing Android implementation would enable codesharing between Android and iOS deployments while leveraging the same Firebase and Google Maps backend services, reducing the incremental development effort required for iOS deployment relative to a native Swift implementation from scratch. Third, a formal security review of the Firebase Authentication configuration, Realtime Database security rules, and API credential management should be conducted prior to any public community deployment at scale, with rule-based access controls implemented to prevent unauthorised data access between user accounts and to protect location data from potential misuse. Fourth, and most importantly for assessing SIMOSA's real-world impact, a user acceptance testing study with actual urban community members in Bandar Lampung should be conducted, evaluating perceived ease of use, actual pickup request completion rates, community adoption patterns, and behavioral outcomes including changes in informal dumping behavior among application users compared to matched non-users. Such a study would provide the empirical evidence of SIMOSA's real-world effectiveness that functional prototype testing alone cannot establish.

## Acknowledgement

The authors express sincere gratitude to the Institut Informatika dan Bisnis Darmajaya, Bandar Lampung, for research support and development facilities provided throughout the SIMOSA project. The authors also thank all community members and waste collection workers who participated in interviews and provided candid insights into the practical challenges of urban waste management that shaped the application's design requirements.

## Author Contributions

ZRAR contributed to conceptualization, software development, data collection, Black Box testing, writing original draft. MSH contributed to supervision, methodology oversight, review and editing, project administration.

## References


Alfath, M. (2018). Pemanfaatan teknologi informasi dan pemberdayaan masyarakat dalam meningkatkan pengelolaan persampahan di Kota Palangka Raya [Utilisation of information technology and community empowerment for improving waste management in Palangka Raya]. *Jurnal Borneo Administrator*, *14*(1), 1-16. https://doi.org/10.24258/jba.v14i1.276

Ardiyansyah, A., Purwanti, E., & Setiawan, A. (2023). Teknik pengumpulan data dan instrumen penelitian ilmiah pendidikan pada pendekatan kualitatif dan kuantitatif [Data collection techniques and scientific research instruments in qualitative and quantitative approaches]. *Jurnal Pendidikan Islam*, *1*(2), 1-9. https://doi.org/10.61132/jpi.v1i2.47

Budiarto, S. P. (2021). Pemanfaatan teknologi untuk sampah bernilai ekonomi dan diskusi pengelolaan sampah [Technology utilisation for economically valuable waste and waste management discussion]. *Prosiding Seminar Nasional Teknik Industri*, *1*(1), 1-10..

Chowdhury, B., Saifuddin, M., & Islam, T. (2023). Smart waste collection system using IoT and GPS tracking for urban municipalities. *Journal of Environmental Management, 329, 117088*. https://doi.org/10.1016/j.jenvman.2022.117088

Elamin, M. Z., Ilmi, Z., Arifin, T., & Siswanto, D. (2018). Analysis of waste management in the village of Disanah, District of Sreseh Sampang, Madura. *Jurnal Kesehatan Lingkungan*, *10*(4), 368-375. https://doi.org/10.20473/jkl.v10i4.2018.368-375

Fadillah, T. Q., Yanda, M. A., & Wulandari, R. (2019). Rancang bangun sistem informasi dan administrasi tahaman dan barang bukti menggunakan model Prototype pada Kepolisian Daerah Jambi [Design and development of information and administrative system using Prototype model at Jambi Regional Police]. *Jurnal Sistem Informasi*, *2*(1), 36-44. https://doi.org/10.30645/jsi.v2i1.12

Febriana Maghfiroh, E., Susilowati, S., & Pramono, A. (2022). Pengembangan aplikasi E-Pulung berbasis Android untuk mendigitalisasi bank sampah Kuncup Mekar Kelurahan Wates Kota Magelang [Development of E-Pulung Android app for digitalising Kuncup Mekar waste bank in Magelang]. *CSPE: Journal of Community Service in Public Education*, *2*(2), 70-83. https://doi.org/10.23917/cspe.v2i2.18791

Harapan, T. K. (2018). Manajemen pengolahan sampah terpadu dalam meningkatkan pendapatan masyarakat di Kecamatan Tampan Kota Pekanbaru [Integrated waste processing management for improving community income in Tampan District, Pekanbaru]. *Jurnal Ilmu Administrasi Negara ASIAN*, *5*(2), 88-98..

Hasibuan, M. S., Febriani, O. M., & Subing, A. (2023). Strategy for community empowerment through an integrated waste bank. Paper presented at International Conference on Smart Applications, Communications and Networking (SmartNets), 1-4.

Holla, S., & Katti, M. M. (2012). Android based mobile application development and its security. *International Journal of Computer Trends and Technology*, *3*(3), 486-490..

Ichwan, M., Husni, F., & Rasyid, M. I. (2021). Pembangunan sistem informasi inventory menggunakan Firebase Realtime Database pada aplikasi Android [Development of inventory information system using Firebase Realtime Database in Android application]. *Jurnal Informatika*, *8*(1), 30-38. https://doi.org/10.31311/ji.v8i1.1423

Kumar, A., & Samadder, S. R. (2017). A review on technological options of waste to energy for effective management of municipal solid waste. *Waste Management, 69, 407-422*. https://doi.org/10.1016/j.wasman.2017.08.005

Kurniati, K. (2021). Penerapan metode Prototype pada perancangan sistem pengarsipan dokumen Kantor Kecamatan Lais [Application of Prototype method for document archiving system design at Lais District Office]. *Journal of Software Engineering Ampera*, *2*(1), 16-27. https://doi.org/10.51519/journalsea.v2i1.44

Marzuki, M., Purnomo, B., & Hidayat, R. (2024). Perancangan aplikasi bank sampah berbasis website untuk kampus bebas sampah [Design of website-based waste bank application for a waste-free campus]. *Journal of Digital Literacy and Volunteering*, *2*(1), 23-30. https://doi.org/10.47134/jdlv.v2i1.3122

Maulana, I. F. (2017). Penerapan Firebase Realtime Database pada aplikasi e-tilang smartphone berbasis mobile Android [Application of Firebase Realtime Database in an Android mobile e-tilang smartphone application]. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, *1*(1), 60-65. https://doi.org/10.29207/resti.v1i1.16

Maulida, N. H. (2022). Studi literatur penerapan metode Prototype dan Waterfall [Literature study on the application of Prototype and Waterfall methods]. *Jurnal Informatika dan Rekayasa Perangkat Lunak*, *3*(1), 1-10. https://doi.org/10.33365/jatika.v3i1.1602

Nasution, A. R. S. (2021). Identifikasi masalah penelitian [Research problem identification]. *Journal of Education and Counseling*, *3*(2), 13-19. https://doi.org/10.31004/joe.v3i2.1010

Nugroho, R. A., Prasetya, D. D., & Kusumaningrum, R. (2021). Pengembangan aplikasi pelacak lokasi berbasis Android menggunakan Google Maps API [Development of Android-based location tracker application using Google Maps API]. *Jurnal Teknologi Informasi dan Ilmu Komputer*, *8*(4), 741-750. https://doi.org/10.25126/jtiik.2021843471

Pardini, K., Rodrigues, J. J. P. C., Kozlov, S. A., Kumar, N., & Furtado, V. (2020). IoT-based solid waste management solutions: A survey. *Journal of Sensor and Actuator Networks*, *8*(1), 5. https://doi.org/10.3390/jsan8010005

Pramana, E. B., Widodo, T., & Santoso, A. (2022). Perancangan sistem informasi monitoring proyek berbasis web untuk mendukung implementasi paperless office [Design of web-based project monitoring information system to support paperless office implementation]. *Jurnal Tera*, *2*(2), 34-43. https://doi.org/10.47896/jtera.v2i2.365

Pressman, R. S. (2012). *Software engineering: A practitioner's approach (7th ed.)*.

Rahman, A., Setiawan, B., & Fathoni, M. (2022). Implementasi Google Maps API untuk pembuatan aplikasi pemantauan lokasi kendaraan secara real-time [Implementation of Google Maps API for real-time vehicle location monitoring application]. *Jurnal Sistem dan Teknologi Informasi*, *10*(3), 298-305. https://doi.org/10.26418/justin.v10i3.52413

Rapii, M. (2021). Pengelolaan sampah secara terpadu berbasis lingkungan masyarakat di Desa Rumbuk [Community environment-based integrated waste management in Rumbuk Village]. *Jurnal Ilmiah Pengembangan dan Penerapan IPTEKS*, *19*(1), 13-22. https://doi.org/10.36312/sasambo.v3i1.512

Safitri, D., Putra, A. S., & Wulandari, A. (2022). Pengembangan aplikasi manajemen data berbasis Android dengan Java dan Firebase [Development of Android-based data management application with Java and Firebase]. *Jurnal Informatika dan Rekayasa Perangkat Lunak*, *4*(2), 120-129. https://doi.org/10.33365/jatika.v4i2.1789

Saud, A., Salim, S. A., & Haris, A. (2022). Mobile application adoption for solid waste management in developing cities: A technology acceptance perspective. *Waste Management and Research*, *40*(8), 1121-1132. https://doi.org/10.1177/0734242X211039654

Setiawan, A., Nugroho, T., & Wahyudi, S. (2023). Smart waste monitoring system berbasis IoT dan Android untuk pengelolaan sampah perkotaan [IoT and Android-based smart waste monitoring system for urban waste management]. *Jurnal Teknologi Informasi dan Komunikasi*, *12*(1), 45-56. https://doi.org/10.47111/jtiik.v12i1.7923

Sibuea, S., Sembiring, R., & Ginting, E. (2022). Aplikasi mobile collection berbasis Android pada PT. *Suzuki Finance Indonesia [Android-based mobile collection application at PT. Suzuki Finance Indonesia]*. *Jurnal Informatika dan Teknologi Komputer*, *2*(1), 31-42. https://doi.org/10.37373/juitek.v2i1.144

Sinaga, T. R., Napitupulu, D., & Situmorang, J. (2021). Rancang bangun sistem informasi manajemen limbah berbasis Android menggunakan metode Prototype [Design and development of Android-based waste management information system using Prototype method]. *Jurnal Informatika Kaputama*, *5*(2), 227-234. https://doi.org/10.51742/ijcis.v5i2.445

Susanti, E., Wibowo, A., & Rahmawati, D. (2022). Implementasi aplikasi pengelolaan sampah berbasis mobile untuk meningkatkan kesadaran lingkungan masyarakat perkotaan [Implementation of mobile-based waste management application to increase environmental awareness in urban communities]. *Jurnal Sistem Informasi dan Teknologi*, *4*(2), 78-87. https://doi.org/10.37034/jsisfotek.v4i2.117

Wahyuningsih, S., Setiawan, R., & Prabowo, A. (2023). Perancangan aplikasi pemesanan layanan kebersihan berbasis Android menggunakan Firebase [Design of Android-based cleaning service ordering application using Firebase]. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, *7*(3), 1245-1253. https://doi.org/10.25126/jtiik.2023734893

Wijaya, I. G. P. S. (2021). Pengujian Black Box pada aplikasi sistem informasi manajemen desa berbasis web [Black Box testing on web-based village management information system application]. *Jurnal Sistem dan Informatika*, *15*(2), 42-47. https://doi.org/10.30864/jsi.v15i2.428

Yadav, V., Bhurjee, A. K., Karmakar, S., & Dikshit, A. K. (2021). A facility location model for municipal solid waste management system under uncertain environment. *Sustainable Cities and Society, 69, 102857*. https://doi.org/10.1016/j.scs.2021.102857

Yuliani, T., Prasetyo, A., & Hidayatullah, R. (2023). Pengembangan sistem monitoring pengumpulan sampah berbasis mobile di kawasan perkotaan [Development of mobile-based waste collection monitoring system in urban areas]. *Jurnal Teknologi dan Sistem Komputer*, *11*(2), 88-96. https://doi.org/10.14710/jtsiskom.2023.17456