Zero Waste

A Real-Time Geo-Based Food Redistribution Platform

Project Documentation Report

Team Members:

23B81A1203

23B81A1212

23B81A1224

23B81A1231

23B81A1253

23B81A1255

Department of Information Technology

Academic Year: 2025-2026

Contents

1 Introduction **3**

2 Problem Statement **3**

3 Objectives of the Project **3**

4 Proposed Solution **4**

5 System Architecture **4**

5.1 Presentation Layer 4

5.2 Application Layer 4

5.3 Database Layer 4

5.4 Geo-Location Integration 5

6 Module Description **5**

6.1 Authentication Module 5

6.2 Host Module 5

6.3 NGO Module 5

6.4 Volunteer Module 5

7 Booking Logic and Workflow **5**

8 Technology Stack **6**

8.1 Frontend Technology: React 6

8.2 Backend Technology: Node.js 6

8.3 Database Technology: PostgreSQL 7

8.4 Integration and Workflow 7

9 Database Design **7**

10 Security and Privacy Considerations **8**

11 Impact and Benefits **8**

12 Future Enhancements **8**

13 Conclusion **8**

1 Introduction

Food wastage is one of the most pressing global challenges in the modern world. Every year, millions of tons of edible food are discarded from households, restaurants, weddings, corporate events, and public gatherings. This wastage not only results in economic loss but also contributes to environmental degradation through increased landfill usage and greenhouse gas emissions.

At the same time, a large segment of society, including orphanages, old-age homes, shelters, and underprivileged communities, struggles to secure daily meals. The imbalance between food surplus and food scarcity highlights a systemic coordination failure rather than a production shortage.

The Zero Waste project aims to solve this issue by introducing a real-time geo-based digital platform that connects event organizers, non-governmental organizations (NGOs), and volunteers. By leveraging modern web technologies and location-based services, the system ensures efficient redistribution of leftover food before it expires.

The primary objective of this project is to minimize food wastage, maximize social benefit, and promote community engagement using technological innovation.

2 Problem Statement

Despite technological advancements, there is currently no widely adopted real-time platform that efficiently connects surplus food providers with organizations in need. In many cases, leftover food from events is discarded due to lack of awareness, absence of communication channels, or logistical challenges.

Existing food donation systems often lack essential features such as live geo-location tracking, booking confirmation mechanisms, and volunteer coordination modules. Without a structured system, multiple organizations may attempt to claim the same donation, causing confusion and inefficiency.

Therefore, there is a critical need for a centralized, secure, and location-aware system that enables instant posting, booking, tracking, and optional delivery support to ensure timely redistribution of surplus food.

3 Objectives of the Project

The main objectives of the Zero Waste system are:

- To reduce food wastage from weddings and public events.

- To provide a real-time platform for NGOs to discover nearby food donations.
- To implement a secure booking mechanism that prevents duplicate claims.
- To enable volunteer-based delivery support for food transportation.
- To maintain transparency through structured status tracking.

These objectives collectively contribute to sustainability and social welfare improvement.

4 Proposed Solution

The proposed system is a web-based application that integrates geo-location services with a role-based user interface. Users are categorized into three primary roles: Host, NGO, and Volunteer.

Hosts can upload detailed information about leftover food, including quantity, food type, expiry time, and GPS location. NGOs can view nearby listings through a live map interface and book available food. Volunteers can assist in transportation if required.

The system ensures that once a food listing is booked, it becomes unavailable to others, preventing duplication. Automatic status updates provide transparency throughout the process.

5 System Architecture

The system follows a client-server architecture consisting of the following layers:

5.1 Presentation Layer

This layer handles the user interface. It provides dashboards tailored to different user roles and ensures responsive design for accessibility across devices.

5.2 Application Layer

The application layer processes business logic, including booking validation, status updates, and role-based access control. It acts as an intermediary between the frontend and the database.

5.3 Database Layer

The database securely stores user credentials, food listings, booking records, and volunteer assignments. Relational integrity ensures accurate data management.

5.4 Geo-Location Integration

The system uses browser GPS or mapping APIs to capture latitude and longitude coordinates. This enables radius-based filtering and live map visualization of food listings.

6 Module Description

6.1 Authentication Module

This module verifies user credentials and grants access based on assigned roles. Password encryption and session management enhance security.

6.2 Host Module

Hosts can post leftover food details such as food name, quantity, type, expiry time, and contact information. The system automatically captures location coordinates to ensure accurate mapping.

6.3 NGO Module

NGOs can browse available food listings based on proximity. Each listing displays essential details including distance, quantity, and remaining time before expiry. NGOs can confirm booking through a single-click action.

6.4 Volunteer Module

Volunteers can view pending delivery requests and accept assignments. Upon successful delivery, the system updates the status to completed.

7 Booking Logic and Workflow

The booking mechanism is designed to ensure fairness and efficiency. Once an NGO selects the booking option, the system updates the status from "Available" to "Booked". The listing is then removed from the public availability feed.

The workflow follows these steps:

1. Host logs into the system.
2. Host uploads leftover food details.
3. The listing appears on the live map.
4. NGO books the food.
5. Status changes to "Booked".

6. Volunteer accepts delivery request (if needed).
7. Status changes to "Completed" after successful delivery.

This structured process ensures transparency and prevents conflicts.

8 Technology Stack

The Zero Waste system is developed using a modern full-stack web development architecture. The selected technologies ensure scalability, performance, security, and maintainability of the application. The system uses React for frontend development, Node.js for backend services, and PostgreSQL as the relational database management system.

8.1 Frontend Technology: React

React is a popular JavaScript library used for building interactive and dynamic user interfaces. It follows a component-based architecture, which allows developers to create reusable UI components and maintain clean project structure.

In the Zero Waste system, React is used to build role-based dashboards for Hosts, NGOs, and Volunteers. The dynamic nature of React enables real-time updates of food listings, booking status changes, and live map rendering without requiring full page reloads.

React's Virtual DOM improves performance by updating only the necessary parts of the user interface when data changes. This ensures smooth user experience, especially when displaying multiple food listings and map-based content.

Additionally, React supports integration with APIs and third-party libraries such as mapping services, making it suitable for geo-location-based applications.

8.2 Backend Technology: Node.js

Node.js is a server-side JavaScript runtime environment that allows execution of JavaScript code outside the browser. It is built on the Chrome V8 engine and is known for its non-blocking, event-driven architecture.

In this project, Node.js is used to handle backend logic including user authentication, food posting, booking validation, status updates, and database communication. The asynchronous nature of Node.js allows the system to manage multiple user requests simultaneously without performance degradation.

The backend exposes RESTful APIs that connect the frontend React application with the PostgreSQL database. Middleware components handle authentication, authorization, and data validation to ensure secure transactions.

Node.js also enhances scalability, making it possible to extend the system in the future for larger user bases or mobile integration.

8.3 Database Technology: PostgreSQL

PostgreSQL is an advanced open-source relational database management system known for reliability, data integrity, and performance.

In the Zero Waste system, PostgreSQL is used to store user information, food listings, booking records, and volunteer assignments. It supports structured query language (SQL) for efficient data retrieval and manipulation.

PostgreSQL ensures relational integrity through primary keys and foreign key constraints. For example, each food listing is linked to a host ID, and each booking is associated with both food ID and NGO ID. This structured relational mapping prevents inconsistencies and ensures accurate data tracking.

The database also supports indexing, which improves query performance when filtering food listings based on location or status.

8.4 Integration and Workflow

The overall workflow of the system is achieved through seamless integration of these technologies. The React frontend sends HTTP requests to the Node.js backend using REST APIs. The backend processes the request, interacts with PostgreSQL to retrieve or store data, and returns the response to the frontend.

This three-tier architecture ensures separation of concerns, maintainability, and future scalability of the application.

9 Database Design

The database includes the following key entities:

- **User**: Stores user ID, role, name, and contact details.
- **FoodListing**: Stores food ID, host ID, quantity, expiry time, and GPS coordinates.
- **Booking**: Maintains booking ID, NGO ID, and status.
- **VolunteerAssignment**: Tracks delivery responsibilities.

Proper indexing and foreign key relationships ensure fast query performance and data consistency.

10 Security and Privacy Considerations

The system incorporates multiple security measures including secure login authentication, encrypted password storage, and role-based authorization. Sensitive contact details are only shared after booking confirmation.

Expiry validation mechanisms ensure that food past its safe consumption time is automatically removed from the system. This protects beneficiaries and maintains trust in the platform.

11 Impact and Benefits

The Zero Waste platform has significant social, environmental, and economic impact. By redistributing surplus food efficiently, the system reduces landfill waste and lowers environmental pollution.

The platform supports NGOs and shelters by providing timely access to food resources. It also encourages volunteerism and strengthens community engagement.

Overall, the project demonstrates how digital innovation can be used to address real-world societal challenges.

12 Future Enhancements

Future improvements may include mobile application development, push notification systems, AI-based demand prediction models, and integration with government welfare schemes.

Additional features such as rating systems and analytics dashboards can further enhance transparency and scalability.

13 Conclusion

Zero Waste is an innovative geo-based food redistribution platform that effectively connects food donors, NGOs, and volunteers. Through structured booking logic, live mapping, and real-time updates, the system ensures efficient and transparent surplus food management.