iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.80594

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

AI-Powered Smart Platform for Surplus Food Waste Management toReduce Hunger and Promote Sustainability

Dr. Sheryl Radley[1], Sanjay D[2], Saravanan P[3], Srinath Bala P K[4], Sriram N[5]
[1]*Department of Electronics and Communication Engineering,* [2, 3, 4, 5]*Department of Information Technology, Meenakshi College of Engineering, West KK Nagar, Chennai-87*

Abstract: ***Food insecurity and food wastage exist side by side in India, creating a paradox that has proved difficult to resolve through informal means alone. Current estimates suggest close to 40 per cent of food produced in the country never reaches a consumer, while over 190 million people go without adequate nutrition. The absence of any organized mechanism connecting those with surplus food to those who can distribute it is at the heart of the problem. This paper describes Plate-bridge, a web-based platform developed to bridge this gap by connecting food donors with verified (Non-Governmental organization) NGO through a centralized, real-time digital system. Donors can list surplus food using a simple form, and NGO can browse available donations, reserve items, and track pickups through a dedicated portal. An integrated AI chat-bot named Plate-bot, built on the Llama 3.3 70 B language model through the Grok API, assists users across the platform. The system is built on the MERN stack, uses MongoDB Geo-spatial indexing for proximity-based search, and enforces security through JWT based authentication and role-specific access controls. Testing confirmed that all modules performed reliably and that the platform can serve as a practical tool for reducing food wastage at a community level.***

Keywords: ***Food donation, surplus food management, AI chat-bot, NGO coordination, MERN stack, Geo-spatial query, real-time tracking, food waste reduction, LLM integration, MongoDB.***

I. INTRODUCTION

India's food system generates waste and hunger simultaneously, and the scale of both is considerable. Rough estimates put the share of food that is produced but never consumed at somewhere close to 40 per cent, while nearly one in five people in the country faces some form of food insecurity. The food that goes to waste is not simply food that has spoiled in transit — a significant portion of it is edible, cooked, and available for redistribution at the moment it is discarded. Restaurants discard leftovers after the last service of the day. Households throw away food prepared in excess. Wedding receptions and corporate events conclude with large quantities of untouched food that end up in landfills within hours.

What prevents this food from reaching people who need it is not a lack of willingness on either side. Donors are often ready to give; NGO are ready to collect. The problem is coordination. In the absence of a shared platform, donations are arranged through phone calls, personal contacts, and informal networks, all of which are slow, unreliable, and impossible to scale. By the time a donor finds a willing NGO, the food may have already crossed the line between safe and unsafe.

Plate-bridge is a web-based platform built to solve this coordination problem. It gives donors a simple, low-friction way to post surplus food listings, whether or not they have a registered account. It gives verified NGO a real-time view of what is available nearby, along with the tools to reserve, collect, and track their donations. It gives administrators the oversight tools they need to verify organization, check donation authenticity, and monitor overall platform activity. And through an embedded AI chat-bot, it gives every user a responsive assistant that can answer questions about the platform or about anything else they want to ask.

A. OBJECTIVES

The following objectives guided the development of Plate-bridge:

- Provide a single platform accessible to donors, NGO, volunteers, and administrators, each with clearly defined capabilities
- Allow both registered and unregistered donors to list surplus food with minimal effort
- Enable NGO to discover available donations based on proximity, using MongoDB Geo-spatial indexing
- Implement a two-stage NGO verification workflow so that only trusted organization can reserve donations

- Automate the removal of expired listings so that the platform always reflects accurate, current availability
- Integrate an AI chat-bot capable of assisting users with both platform-specific and general queries
- Secure all sensitive routes using JWT based authentication and role-based access control

B. SCOPE

The platform covers the full life-cycle of a food donation from initial listing through to collection and verification. Within this scope, the system handles:

- Guest and registered donor workflows, with separate handling of contact information for each
- NGO registration and administrator-controlled verification before reservation access is granted
- Geo-spatial donation discovery with configurable search radius
- Admin analytic covering active listings, meals saved, food type ratios, and pending verification
- An AI chat-bot embedded across all platform pages

II. LITERATURE SURVEY

Research on digital food donation systems has grown steadily over the past several years, producing a range of mobile and web-based solutions that address different aspects of the coordination problem. Reviewing this work helps identify both what has been achieved and where gaps remain.

Shubham, Shruthi, and Vinay (2019) built a mobile-based food donation system that used ERP integration to handle real-time upload and request matching through rule-based logic. The system worked well in small, well-defined deployments and showed that mobile technology could reduce response times in donation workflows. It did not, however, address the question of geographical proximity, and its matching logic was too rigid to adapt to the varied conditions of a real deployment.Mohammed Shafi, Anusha, and Deepika (2020) approached the problem from a data management angle, designing a web-based application with structured user registration, food upload modules, and an administrative dashboard. Their contribution lay in demonstrating how a well-organized database could improve tracking and reduce duplication. What the system lacked was any capability for intelligent matching or location-based discovery, which limited its practical utility for NGO operating across a city.

Vignesh, Harish, and Divya (2021) introduced a centralized donation portal that added automated alerts to the basic listing and request functionality seen in earlier work. NGO received notifications when new donations matching their needs were posted, which reduced the lag between listing and collection. The system still fell short in its inability to reason about distance or predict where demand would arise, and it offered no AI-assisted features of any kind.Wan Abdul Rahim et al. (2022) developed Food. Rescue, a mobile application that brought location-aware food sharing to the field using the Mobile Application Development Life Cycle methodology. The application enabled real-time updates and helped NGO identify nearby donations, making it one of the more practically useful systems in the reviewed literature. It was limited to mobile platforms, however, and lacked any administrative oversight layer or NGO verification mechanism.

Sarvasri Sowmya Lakshmi et al. (2024) presented Share and Care, a web application built on Python Flask and MySQL with map integration and real-time updates. The system was developed using Agile methodology, which gave the team flexibility to respond to changing requirements during development. The platform was effective within its scope but did not include AI-assisted functionality, Geo-spatial indexing, or any form of token-based authentication that would have made it suitable for deployment beyond a controlled environment.Taken as a whole, this body of work traces a progression from basic listing systems toward more responsive and location-aware tools. Plate-Bridge builds on this progression by combining full life-cycle donation management, administrator-controlled verification, Geo-spatial proximity search, and an (Large Language Model) LLM powered chat-bot in a single platform — a combination that none of the reviewed systems provides.

III. SYSTEM ANALYSIS

Before designing a new system, it is useful to understand where existing approaches fall short. This section examines the limitations of current food donation practices and describes how the proposed system addresses them.

A. EXISTING SYSTEM

Most food donation efforts today rely on informal coordination. Donors contact NGO they already know through calls or messages. When those relationships do not exist, food often goes to waste because the donor has no way of finding a collector quickly enough. The following problems characterize existing approaches:

- No shared platform exists to connect all parties — donors, NGO, volunteers, and administrators — in one place
- Donors frequently cannot identify nearby organization that can collect food before its window of safe consumption closes
- NGO cannot see what food is available in their area in real time, making planning impossible
- Food quality and authenticity cannot be verified before collection, which creates safety risks for recipients
- Expired or uncollected listings remain in circulation through informal channels, eroding trust
- There is no record of what was donated, who collected it, or whether it reached those it was intended for

B. PROPOSED SYSTEM

Plate-Bridge provides a centralized web-based platform that manages the full life-cycle of a food donation. The system allows donors to create listings in minutes, whether or not they hold an account. Verified NGO can see all active listings in real time, use proximity-based filters to find nearby food, and reserve items through a single click. Administrators have a dedicated panel for approving NGO registrations, verifying individual donations, and reviewing platform statistics.

The platform also includes an automated expiry mechanism that removes stale listings every thirty minutes, ensuring the NGO portal always reflects genuinely available food. Plate-bot, an AI chat-bot powered by the Grok API, assists users across every page with questions about the platform or on general topics. Security is enforced through JWT based authentication and role-specific middleware that restricts actions to those appropriate for each user type.

The key improvements over existing approaches are:

- A single coordinated platform that all parties use, eliminating the fragmentation of informal channels
- Proximity-based donation discovery that enables NGO to find relevant listings without manual searching
- An administrator verification layer that establishes accountability and builds trust among donors
- An AI chat-bot that reduces the learning curve for new users and makes the platform more accessible
- A complete donation history that supports reporting, planning, and accountability

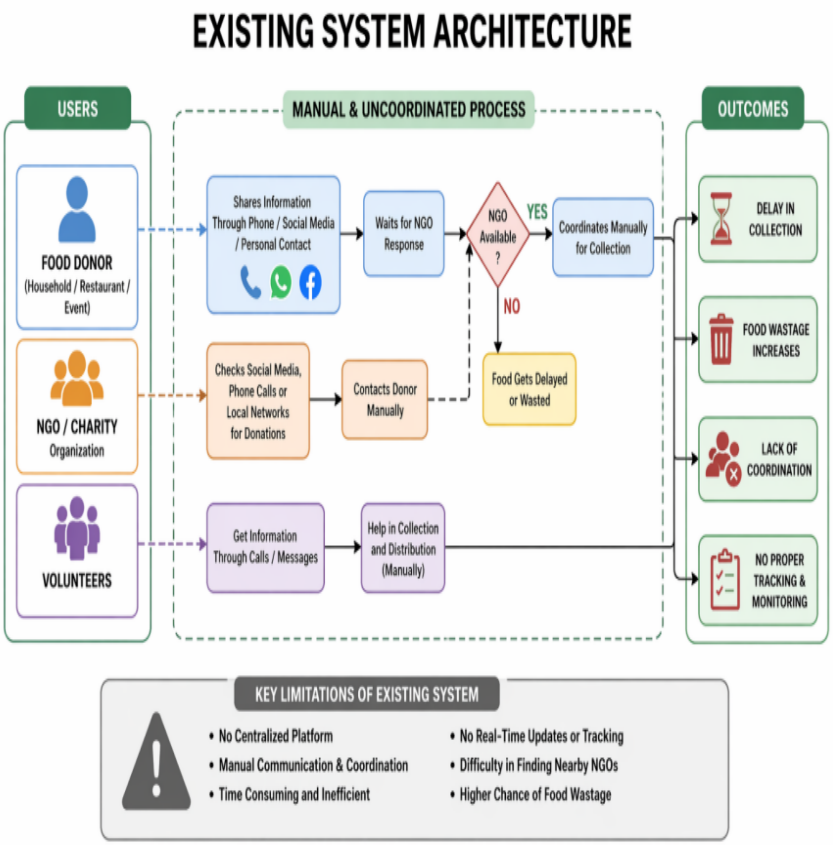


Fig 3.1 Existing System Architecture Diagram

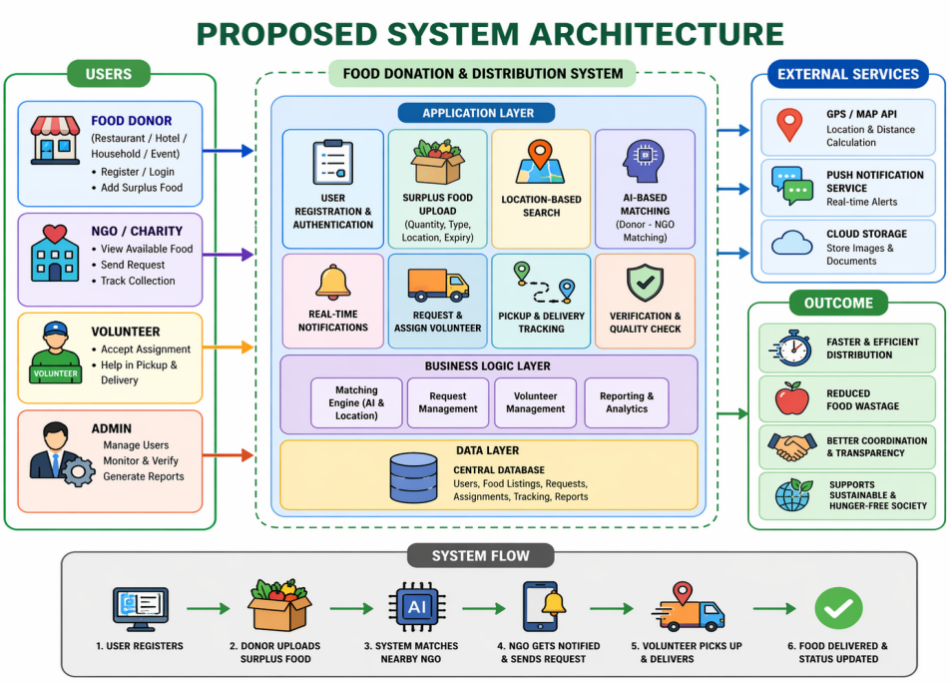


Fig 3.2 Proposed System Architecture Diagram

IV. SYSTEM SPECIFICATION

Running Plate-bridge requires a relatively modest hardware setup. A machine with an Intel Core i5 processor or equivalent, at least 8 GB of RAM, and a 256 GB SSD is adequate for development and local testing. Production deployment on a cloud platform relaxes most of these constraints. All users need a stable internet connection and access to a modern web browser — no client-side installation is required.

On the software side, the platform uses the MERN stack throughout. React with Type-script handles the front-end, styled using Tailwind CSS and Shad CN components. Node.js and Express.js power the back-end, which is structured around the MVC pattern. MongoDB Atlas provides cloud-hosted data storage, with a 2dsphere index on donation coordinates enabling the Geo-spatial queries used by the NGO portal. Passwords are hashed with bcrypt.js and sessions are governed by JWT tokens. The AI chat-bot connects to the Grok API using Node's built-in https module, keeping the API key securely on the server side.

A. SYSTEM DESIGN

The system follows a three-tier architecture. The React front-end communicates with the Express back-end through RESTfull API endpoints. The back-end handles all business logic, communicates with MongoDB Atlas for data operations, and proxies chat-bot requests to the Grok API. JWT middleware inspects all incoming requests to protected routes and verifies both the token and the user's role before permitting any action. A background job on the back-end checks for expired donations every thirty minutes and updates their status automatically.

B. ARCHITECTURAL DIAGRAM

The architecture connects five layers: the React front-end, the Express REST API, the JWT authentication middleware layer, the MongoDB Atlas database, and the Grok AI API. Every donor, NGO, and admin request passes through the Express back-end, which enforces access controls before responding. The AI chat-bot route is the only one that calls an external API, and it does so only after verifying the Grok API key is present in the server environment.

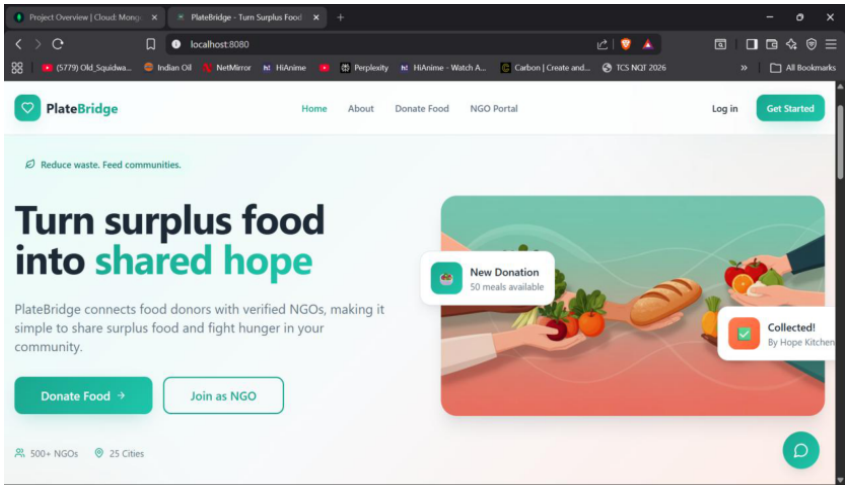


Fig 4.1 Landing Page / Home Page

V. SYSTEM DESCRIPTION

Plate-bridge is a real-time food donation coordination system that connects three types of users through a single interface. Donors can post surplus food immediately. NGOs that have been approved by an administrator can search for available food, reserve items, and confirm collection. Administrators oversee the platform, verifying new registrations, auditing donations, and reviewing statistics.

VI. SYSTEM IMPLEMENTATION

Development followed an Agile process with iterative sprints. Each sprint delivered a working increment covering authentication, donation management, the NGO portal, the admin panel, Geo-spatial querying, and AI chat-bot integration in turn. End-of-sprint reviews shaped the priorities of subsequent iterations.

A. Authentication

Users register through the /api/auth/register endpoint, which hashes their password with bcrypt.js before storing the user document. NGO accounts are created with a verification. Status of pending and an is Verified flag set to false. Login at /api/auth/login compares the submitted password against the stored hash and, on success, returns a JWT token signed with the server's secret key. This token must accompany every subsequent request to a protected route, where the protect middleware decodes and validates it before attaching the user object to the request.

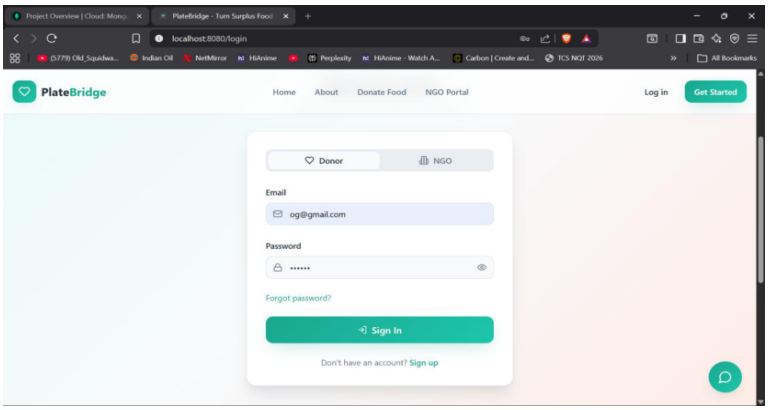


Fig 6.1 Donor Login Page

When an NGO registers, their account sits in the pending state until an administrator approves it through the admin panel. Only at that point does is Verified change to true and the NGO gain access to the reservation features. This two-stage flow prevents un-vetted organization from interacting with donations before their credentials have been confirmed.

B. Donation Management

The /api/donations endpoint accepts POST requests from both authenticated users and guests. If a valid JWT is present, the donation is linked to the logged-in donor's account. If no token is found, the request is treated as a guest submission and the contact details in the form body are stored in a separate guest Donor object within the donation record. Location data is stored as a Geo JSON Point with coordinates in the order MongoDB requires: longitude first, then latitude. This ensures compatibility with the 2dsphere index and the $near operator.

C. Reservation Flow

When a verified NGO reserves a donation, the API checks that the donation is still in Listed status and that its expiry date has not passed. A new Reservation document is created linking the NGO user ID to the donation ID, and the donation's status is updated to Reserved in the same operation. A unique index on the donation field in the Reservations collection prevents two simultaneous reservation attempts from both succeeding. If the NGO later collects the food, they mark the reservation as collected, which sets both the Reservation and Donation statuses to Collected.

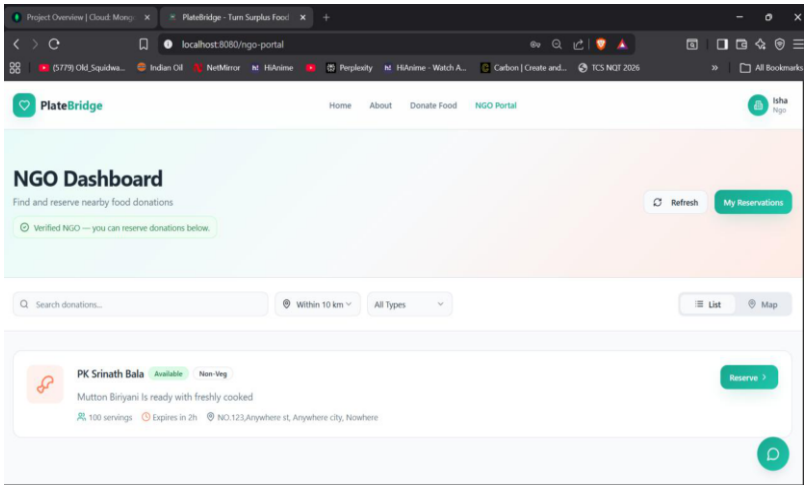


Fig 6.2 NGO Dashboard

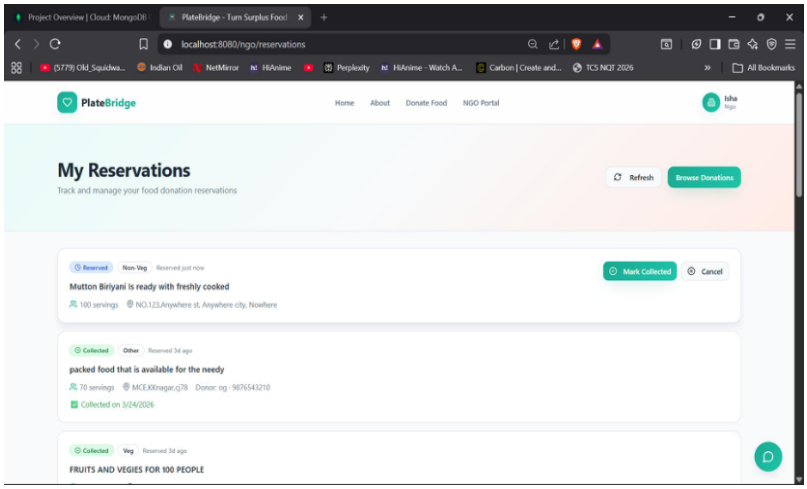


Fig 6.3 My Reservations Page

D. Admin Verification

The admin dashboard aggregates twelve statistics from MongoDB using Promise.all, running all queries in parallel rather than sequentially to keep response times low. The meals saved figure is produced by an aggregation pipeline that sums the quantity field across all donations with a Collected status. Donation verification adds a record of the verifying administrator's ID, the timestamp, and a note to the donation document. This information is visible in the Donations tab and provides a lightweight audit trail.

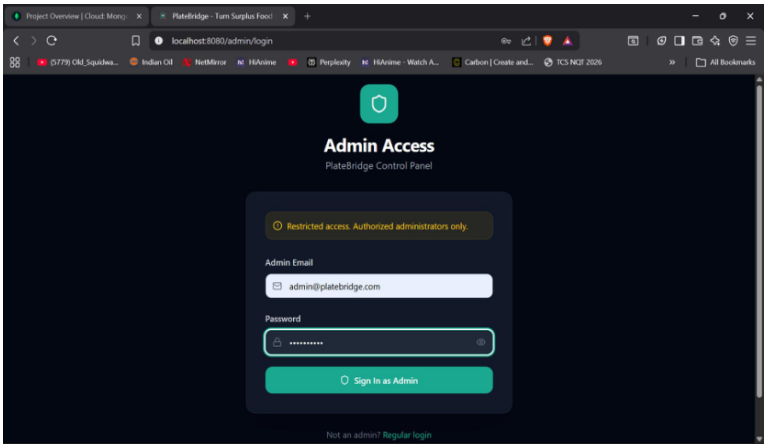


Fig 6.4 Admin Panel

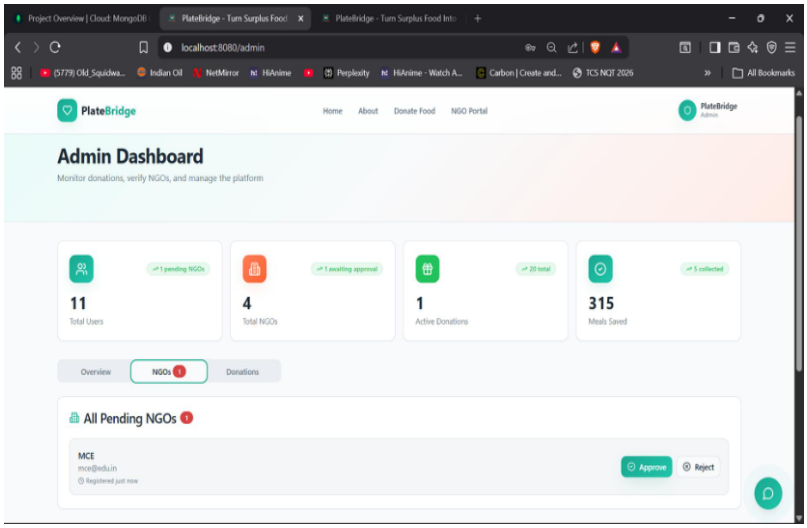


Fig 6.5 Admin Dashboard

E. Geo-spatial Query

The /api/donations/nearby endpoint takes latitude, longitude, and radius as query parameters. The radius value is converted from kilometre to metre and passed as the max-distance argument to MongoDB's $near operator. The query also filters for donations with a Listed status and an expiry date after the current time. Results are returned sorted by distance from the provided coordinates, so the closest available donations appear first. On test data, this query consistently returned results in under 200 milliseconds for a 10 km search radius.

F. Plate-bot Integration

The back-end chat route uses Node's built-in https module rather than fetch to communicate with the Grok API, which avoids compatibility issues across Node.js versions. The route constructs a system prompt that describes the platform and the current user's role, pretends it to the conversation history, and sends the complete message array to the API. When a response arrives, the route extracts the text from the choices array and returns it as a JSON response. The API key is read from an environment variable and is never exposed to the client.

VII. SYSTEM TESTING AND MAINTENANCE

Testing was conducted across all three user roles and covered both the functional behaviour of individual routes and the end-to-end performance of the platform under realistic conditions.

A. TESTING OVERVIEW

The testing programme combined black-box and white-box methods. Black-box testing was applied to all REST API endpoints, verifying that valid inputs produced the expected outputs and that invalid or unauthorized requests returned appropriate error responses. White-box testing examined the internal logic of the authentication middleware, the Geo-spatial query builder, and the expiry job to check that edge cases were handled correctly.

B. TESTING METHODOLOGIES

1) Black-Box Testing

Each endpoint was tested using Postman with a range of valid and invalid inputs. Authentication endpoints were tested with correct credentials, wrong passwords, missing fields, and requests carrying no token. Donation endpoints were tested as both authenticated and guest users, including scenarios with missing required fields, invalid date formats, and coordinates outside valid geographic ranges. Reservation endpoints were tested for duplicate prevention, expired donation handling, and attempts to access NGO routes using a donor token.

2) White-Box Testing

The protect middleware was verified with expired tokens, tokens signed with the wrong secret, and tokens referencing user IDs that had been removed from the database. The Geo-spatial query was tested against seeded data at various distances to confirm that the radius filter excluded items beyond the limit. The expiry job was verified by seeding donations with past expiry times and checking that the job updated their statuses within a single execution cycle.

3) Performance Testing

The nearby donations endpoint was benchmark against a data-set of 50 listings distributed across a simulated area. Average response times remained below 200 milliseconds for a 10 km radius, confirming the effectiveness of the 2dsphere index. The admin dashboard, which runs 12 parallel MongoDB queries, responded within 300 milliseconds under normal conditions. Chat-bot responses arrived within approximately 800 milliseconds to 1.5 seconds, which was considered acceptable for a conversational interface.

C. TEST CASES

1	Guest donation creation	Valid food details, no token	201 response, donation saved	Pass
2	Registered donor listing	Valid JWT + food details	Donation linked to user account	Pass
3	Missing required field	Empty description field	400 validation error returned	Pass
4	NGO reserve listed donation	Verified NGO token + valid ID	Reservation created, status Reserved	Pass
5	Duplicate reservation	Same donation ID reserved twice	400 error: Already reserved	Pass
6	Reserve expired donation	Expiry date in the past	400 error: Donation has expired	Pass
7	Donor on NGO route	Donor JWT on NGO endpoint	403 Forbidden returned	Pass
8	Unverified NGO reserve	Pending NGO token	403 Pending verification returned	Pass
9	Admin approve NGO	Admin approves pending account	isVerified set to true	Pass
10	Nearby donations query	Lat/lng within 10 km of data	Sorted donation list returned	Pass
11	Auto expiry job	Donations with past expiry	Status updated to Expired	Pass
12	PlateBot general query	"What is food waste?"	Relevant answer returned	Pass

VIII. RESULTS AND DISCUSSION

The Plate-Bridge platform was implemented and tested end to end across all three user roles. The outcomes confirmed that the system behaves as intended and that it can operate reliably under the conditions evaluated.

Donor functionality worked correctly in both guest and registered modes. Forms validated as expected, rejecting incomplete submissions and storing correctly structured records in MongoDB Atlas. Donors received clear confirmation after successful submission, and registered users could view their complete donation history along with the reservation status of each item.

NGO functionality also performed as expected. The portal loaded current listings accurately, the Reserve button triggered the reservation flow without error, and duplicate reservation attempts were blocked at the database level. The My Reservations page allowed NGOs to update reservation status and provided contact information for the donor where applicable.

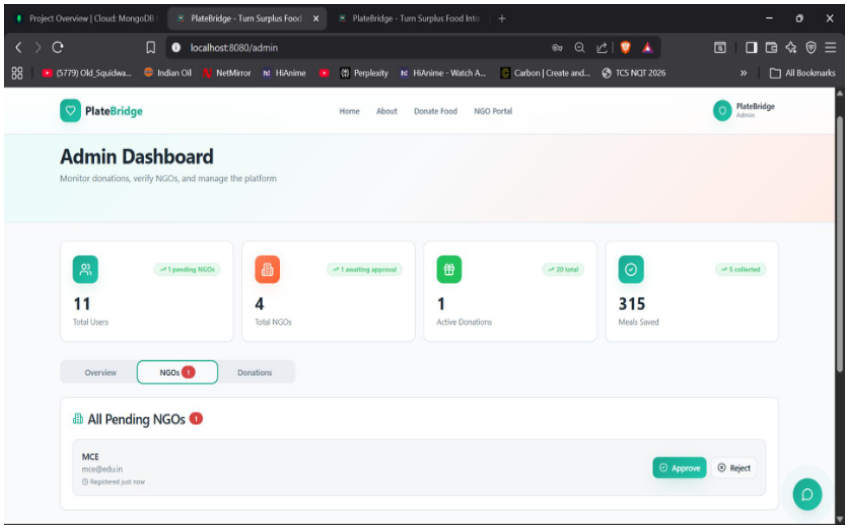


Fig 8.1 Admin Dashboard with Live Statistics

The admin dashboard displayed live data pulled from the database, including pending NGO applications that could be acted upon immediately. The donation verification feature added an audit-able record to each verified item, giving NGO additional confidence in the authenticity of what they were collecting. The dark-themed admin login page successfully rejected login attempts from non-admin accounts, confirming that the role check was functioning correctly.

Plate-bot responded accurately to questions about the platform, including questions about how to donate food, how NGO verification works, and how to cancel a reservation. It also handled general questions on topics such as food safety, nutrition, and general knowledge, confirming its utility beyond platform-specific guidance. Response times from the Grok API were consistently within one to two seconds, which maintained a fluid conversational experience.

Geo-spatial queries returned relevant results within the configured radius and sorted them correctly by proximity. The automated expiry job cleared stale listings on schedule, keeping the NGO portal clean without any manual intervention. Altogether, these results support the claim that Plate-Bridge can function as a reliable, practical food redistribution tool.

IX. CONCLUSION AND FUTURE ENHANCEMENT

Plate-bridge addresses a real and persistent coordination problem in food redistribution by providing a structured, transparent, and accessible digital platform. The system brings donors, NGO, and administrators together in one place, removes the delays associated with informal coordination, and ensures that only verified organizations can access the surplus food that donors list.

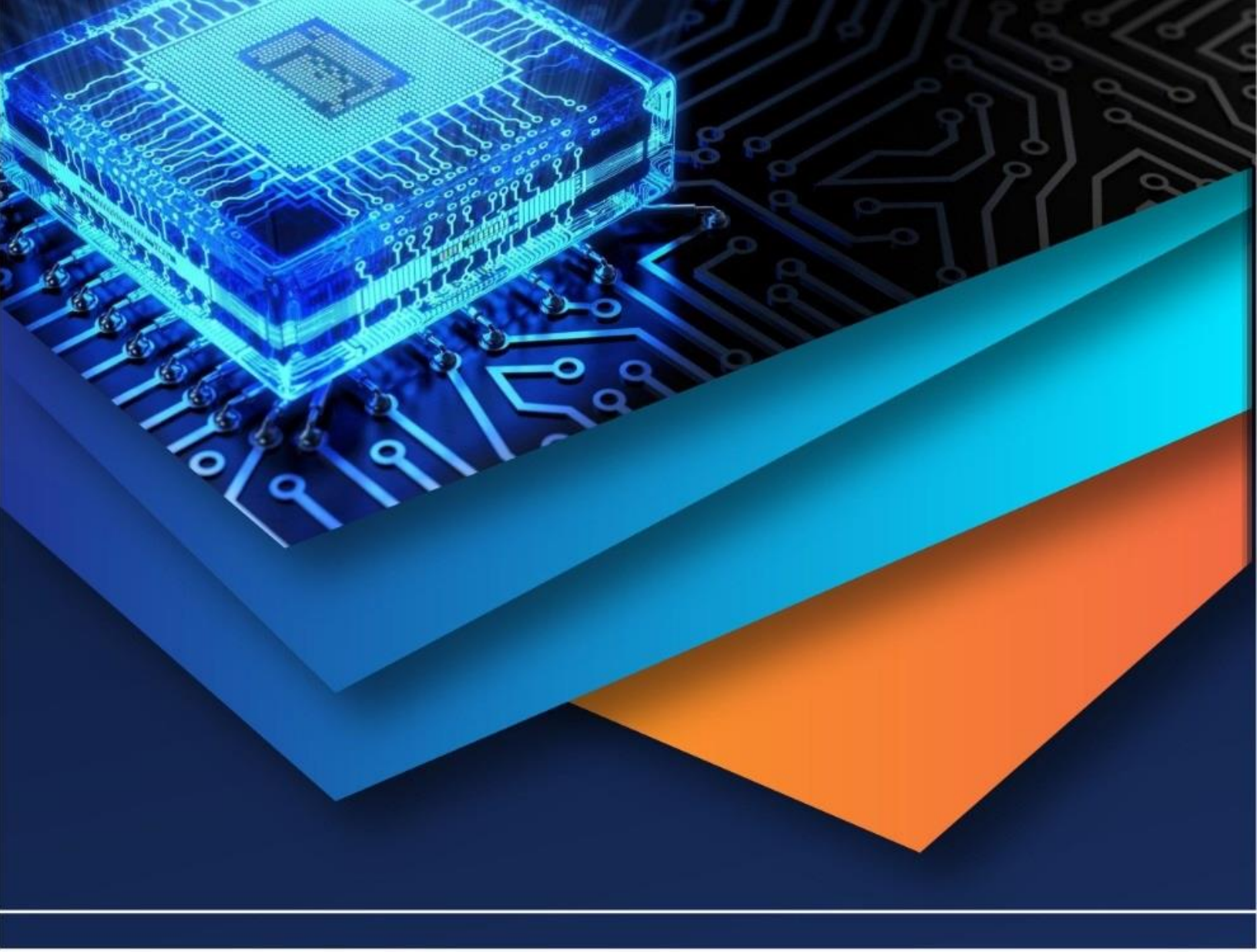
The technical choices — Geo-spatial indexing for proximity search, JWT authentication for secure access, MongoDB Atlas for scale-able cloud storage, and a Grok-powered chat-bot for user assistance — are well-suited to the platform's requirements. The automated expiry job and admin verification layer add operational integrity that informal systems cannot provide. Testing across all modules confirmed that the platform handles expected inputs correctly and responds appropriately to edge cases such as expired donations and duplicate reservation attempts.

The project contributes to the goals set out in UN Sustainable Development Goal 2 on Zero Hunger and SDG 12 on Responsible Consumption and Production. It demonstrates that a thoughtfully built web platform, developed with relatively modest resources, can make a meaningful difference in how surplus food reaches those who need it.

Several directions are available for extending the platform in subsequent work. A mobile application for Android and iOS would extend the platform's reach to users who primarily access services through their phones. Predictive models trained on historical donation data could help NGO plan collection schedules and anticipate availability patterns before listings are posted

REFERENCES

[1] United Nations, Sustainable Development Goal 2 – Zero Hunger Initiative, UN SDG, 2015. [Online]. Available: https://sdgs.un.org/goals/goal2
[2] S. Shubham, R. Shruthi and M. Vinay, A Smart Food Donation System Using Mobile Technology to Reduce Food Waste, International Journal of Engineering Research and Technology, vol. 8, no. 5, 2019.
[3] A. Mohammed Shafi, S. Anusha and R. Deepika, Web-Based Application for Food Waste Management, International Journal of Advanced Science and Technology, vol. 29, no. 7, 2020.
[4] P. Vignesh, K. Harish and T. Divya, Web-Based Food Donation and Redistribution System to Minimize Food Waste and Support NGOs, International Journal of Computer Applications, 2021.
[5] Wan Abdul Rahim Wan Mohd Isa, Ahmad Iqbal Hakim Suhaimi, Nurulhuda Noordin and Nurul Nadia Taslim, FoodRescue: Excess Food Donation Mobile Application, MADLC Methodology, 2022.
[6] V. Sarvasri Sowmya Lakshmi, M. N. V. Surekha, P. Sasi Sanjana, R. Jyothi Sai Durga and M. Charan Sri Sai, Share and Care: A Food Donation Web Application, International Journal of Innovative Science and Research Technology, vol. 9, no. 2, Feb. 2024, pp. 74–82.
[7] Meta AI, Llama 3.3: Open Foundation and Fine-Tuned Chat Models, Meta AI Research, 2024. [Online]. Available: https://ai.meta.com/llama
[8] MongoDB Inc., Geospatial Queries, MongoDB Atlas Documentation, 2024. [Online]. Available: https://www.mongodb.com/docs/manual/geospatial-queries
[9] Express.js Foundation, Express.js – Fast, Un-opinionated Web Framework for Node.js, 2024. [Online]. Available: https://expressjs.com
[10] Groq Inc., Groq API Documentation, 2024. [Online]. Available: https://console.groq.com/docs



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY