ZEROhunger! - Food Redistribution and Donation Platform

A PROJECT REPORT

Submitted by

PIYUSH MISHRA [RA2211027010038]
ADITHYAN R [RA2211027010040]
SRIKAR KUMAR [RA2211027010052]

Under the Guidance of

Dr. S SRIVIDHYA

(Assistant Professor, DSBS)

in partial fulfillment of the requirements for the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING
with specialization in Big Data Analytics

DEPARTMENT OF
DATA SCIENCE AND BUSINESS SYSTEMS
SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR- 603 203

MAY 2025

Department of Data Science and Business Systems

SRM Institute of Science & Technology

Own Work Declaration Form

To be completed by the student for all assessments

Degree/ Course	**: BTech / CSE with specialization in Big Data Analytics**
Student Name	**: Piyush Mishra** **Adithyan R** **Srikar Kumar**
Registration Number	**: RA2211027010038** **RA2211027010040** **RA2211027010052**
Title of Work	**: ZEROhunger! - Food Redistribution and Donation Platform**

I / We hereby certify that this assessment compiles with the University's Rules and Regulations relating to Academic misconduct and plagiarism**, as listed in the University Website, Regulations, and the Education Committee guidelines.

I / We confirm that all the work contained in this assessment is my / our own except where indicated, and that I / We have met the following conditions:

- Clearly referenced / listed all sources as appropriate
- Referenced and put in inverted commas all quoted text (from books, web, etc)
- Given the sources of all pictures, data etc. that are not my own
- Not made any use of the report(s) or essay(s) of any other student(s) either past or present
- Acknowledged in appropriate places any help that I have received from others (e.g. fellow students, technicians, statisticians, external sources)
- Compiled with any other plagiarism criteria specified in the Course handbook / University website

I understand that any false claim for this work will be penalized in accordance with the University policies and regulations.

DECLARATION:

I am aware of and understand the University's policy on Academic misconduct and plagiarism and I certify that this assessment is my / our own work, except where indicated by referring, and that I have followed the good academic practices noted above.

If you are working in a group, please write your registration numbers and sign with the date for every student in your group.

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR – 603 203

BONAFIDE CERTIFICATE

Certified that 21CSP302L - Project report titled “**ZEROhunger! - Food Redistribution and Donation Platform**” is the bonafide work of “**PIYUSH MISHRA [RA2211027010038], ADITHYAN R [RA2211027010040], SRIKAR KUMAR [RA2211027010052]”** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr. S SRIVIDHYA

SUPERVISOR
ASSISTANT PROFESSOR
DEPARTMENT OF
DATA SCIENCE AND
BUSINESS SYSTEMS

SIGNATURE

Dr. V. KAVITHA

PROFESSOR &HEAD
DEPARTMENT OF
DATA SCIENCE AND
BUSINESS SYSTEMS

ACKNOWLEDGEMENTS

We express our humble gratitude to **Dr. C. Muthamizhchelvan**, Vice-Chancellor, SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support.

We extend our sincere thanks to **Dr. Leenus Jesu Martin M,** Dean-CET, SRM Institute of Science and Technology, for his invaluable support.

We wish to thank **Dr. Revathi Venkataraman**, Professor and Chairperson, School of Computing, SRM Institute of Science and Technology, for her support throughout the project work.

We encompass our sincere thanks to, **Dr. M. Pushpalatha**, Professor and Associate Chairperson - CS, School of Computing and **Dr. Lakshmi,** Professor and Associate Chairperson -AI, School of Computing, SRM Institute of Science and Technology, for their invaluable support.

We are incredibly grateful to our Head of the Department Dr. V. Kavitha, Professor and Head, Data Science and Business Systems, SRM Institute of Science and Technology, for her suggestions and encouragement at all the stages of the project work.

We want to convey our thanks to our Project Coordinators, Panel Head, and Panel Members Department of Data Science and Business Systems, SRM Institute of Science and Technology, for their inputs during the project reviews and support.

We register our immeasurable thanks to our Faculty Advisor, Dr. K Panimalar, Department of Data Science and Business Systems, SRM Institute of Science and Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to our guide, Dr. S Srividhya, Department of Data Science and Business Systems, SRM Institute of Science and Technology, for providing us with an opportunity to pursue our project under her mentorship. She provided us with the freedom and support to explore the research topics of our interest. Her passion for solving problems and making a difference in the world has always been inspiring.

We sincerely thank all the staff members of Data Science and Business Systems, School of Computing, S.R.M Institute of Science and Technology, for their help during our project. Finally, we would like to thank our parents, family members, and friends for their unconditional love, constant support and encouragement

Authors

ABSTRACT

ZEROhunger is a purpose-driven platform dedicated to eradicating hunger by connecting restaurants with NGOs to ensure surplus food reaches those in need. By facilitating real-time communication and streamlined logistics, the platform bridges the gap between food donors and beneficiaries. ZEROhunger incorporates features like geolocation-based matching, safety compliance, and impact tracking to ensure efficient and safe food redistribution. By fostering collaboration between food producers and nonprofit organizations, ZEROhunger plays a pivotal role in reducing food waste and contributing to the fight against hunger. This initiative aligns with global sustainability efforts and aspires to create a world where excess food is redirected to create a positive impact.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

AI	Artificial Intelligence
API	Application Programming Interface
CNN	Convolutional Neural Network
CSS	Cascading Style Sheets
2FA	Two-Factor Authentication
HTML	HyperText Markup Language
IDE	Integrated Development Environment
JSON	JavaScript Object Notation
LSTM	Long Short-Term Memory
NWC	Networking and Communication
REST	Representational State Transfer
RNN	Recurrent Neural Network
S	Microsoft
SDG	Sustainable Development Goal
SQS	Simple Queue Service (Amazon Web Services)
UI	User Interface
UX	User Experience

CHAPTER 1
INTRODUCTION

1.1 Introduction to ZEROhunger! - Food Redistribution and Donation Platform:

ZEROhunger is an innovative platform designed to tackle two critical global challenges: food waste and hunger. With millions of people experiencing food insecurity and tons of edible food being discarded daily, ZEROhunger aims to bridge the gap between surplus food and those in need. The platform fosters partnerships by directly connecting restaurants with NGOs, creating an efficient food redistribution network. By transforming excess food into meaningful resources for underserved communities, ZEROhunger strives to contribute to the global movement toward sustainability and social equity. Through technology-driven solutions, real-time geolocation tools, and a commitment to safety standards, ZEROhunger provides a transparent and impactful way to reduce food waste while ensuring equitable access to nourishment. This initiative aligns with the United Nations' Sustainable Development Goal of achieving Zero Hunger, paving the way for a hunger-free world.

1.2 Motivation

The motivation behind ZEROhunger stems from the alarming paradox of widespread hunger coexisting with vast quantities of food waste. Every day, millions of people struggle to access nutritious meals, while surplus food from restaurants and other establishments goes to waste. This pressing challenge has inspired the creation of ZEROhunger - a platform that seeks to bridge this gap by utilizing technology to connect surplus food providers with NGOs that distribute it to those in need.

Additionally, ZEROhunger is driven by a vision of sustainability, social responsibility, and community empowerment. By preventing edible food from being wasted and redirecting it to those who lack access, the platform not only combats hunger but also reduces the environmental impact of food waste. The initiative reflects a commitment to global efforts aimed at achieving food security and sustainability, embodying a practical solution to a critical issue.

1.3 Sustainable Development Goal of the Project

ZEROhunger aligns closely with the United Nations' Sustainable Development Goal (SDG) 2: Zero Hunger. This SDG aims to end hunger, achieve food security, improve nutrition, and promote sustainable agriculture by 2030. The project addresses this global challenge by creating a platform that redistributes surplus food from restaurants to NGOs, ensuring that food reaches underserved communities instead of being wasted.

Additionally, ZEROhunger contributes to SDG 12: Responsible Consumption and Production, by reducing food waste—a critical component of sustainable development. The platform exemplifies how technology and collaboration can address systemic issues, fostering a more equitable and sustainable food distribution system. By supporting these interconnected SDGs, ZEROhunger advances the vision of a world where no individual goes hungry, and resources are utilized responsibly for the benefit of both people and the planet

1.4 Product Vision Statement

1.4.1 Audience:

- **Primary Audience:** Restaurants and food service providers with surplus food looking to donate responsibly.
- **Secondary Audience:** NGOs, food banks, and community organizations striving to feed the hungry and reduce food insecurity.

1.4.2 Needs:

- **Primary Needs:**
  - A seamless platform for restaurants to donate surplus food to NGOs.
  - Real-time geolocation-based matching to connect donors with recipients efficiently.
  - Assurance of food safety standards to ensure the safe redistribution of food.
- **Secondary Needs:**
  - Tracking tools to monitor the impact and quantity of redistributed food.
  - Analytics and reporting features for measuring contributions and outcomes.
  - Community-building tools to foster collaboration among donors, NGOs, and volunteers.

1.4.3 Products:

- **Core Product:**
  A web-based platform facilitating real-time food redistribution between restaurants and NGOs.
- **Additional Features:**
  - Real-time logistics tracking for efficient food delivery and pickup.
  - A built-in dashboard to track donations and measure community impact.
  - AI-driven features to optimize matches based on proximity, food type, and need.
  - Feedback mechanisms for continuous improvement and collaboration.

1.4.4 Values:

- **Core Values:**
  - **Sustainability:** Reducing food waste and its environmental footprint.
  - **Empathy:** Addressing hunger and supporting underprivileged communities.
  - **Collaboration:** Building a network of restaurants, NGOs, and volunteers for collective impact.
- **Differentiators:**
  - **Geolocation-Based Matching:** Ensuring quick and efficient food redistribution.
  - **Impact Metrics:** Providing tangible data on contributions and community benefits.
  - **Safety Assurance:** Maintaining high food safety standards to protect recipients.

1.5 Product Goal

The primary goal of ZEROhunger is to create a sustainable and efficient system for redistributing surplus food from restaurants to NGOs, thereby addressing hunger and reducing food waste. The platform aims to:

1. **Reduce Hunger:** Bridge the gap between surplus food and underserved communities by ensuring that nutritious meals reach those in need.
2. **Minimize Food Waste:** Prevent edible food from being discarded by redirecting it to meaningful causes, thereby contributing to environmental sustainability.
3. **Enhance Collaboration:** Build a connected ecosystem of restaurants, NGOs, and volunteers to streamline food redistribution efforts and maximize impact.

4. **Leverage Technology:** Use innovative tools like real-time geolocation, data analytics, and AI-driven matching to optimize the donation process and ensure efficiency.
5. **Raise Awareness:** Promote social responsibility among food producers and the wider community, inspiring active participation in the fight against hunger and waste.
6. **Achieve Sustainability Goals:** Align with global efforts, such as the United Nations' Sustainable Development Goals, particularly focusing on Zero Hunger (SDG 2) and Responsible Consumption and Production (SDG 12).

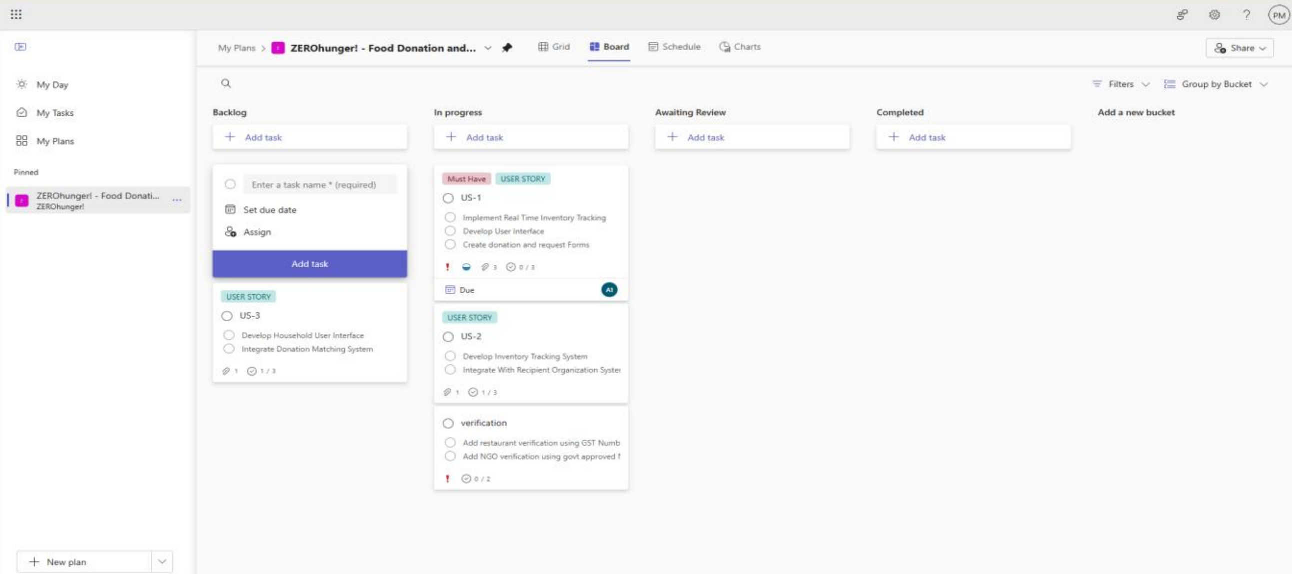
ZEROhunger aspires to be a transformative solution, fostering a world where surplus food is never wasted but instead serves as a lifeline for those facing food insecurity.

1.6 Product Backlog

S.No	**User Stories of ZEROhunger - Food Redistribution Platform**
#US 1	As a local business owner, I want to easily donate my surplus food to nearby recipient organizations so that I can contribute to reducing food waste and help those in need.
#US 2	As a volunteer, I want to receive real-time notifications about food collection and distribution schedules so that I can efficiently plan my volunteer activities and ensure timely delivery of surplus food to those in need.
#US 3	As a recipient organization manager, I want to easily track incoming food donations and inventory levels so that I can efficiently manage and distribute food to those in need.
#US 4	As a household member with surplus food, I want to easily list my excess food items on the ZEROhunger! platform so that I can ensure they are redistributed to those in need instead of going to waste.

The product backlog of ZEROhunger! was configured using the MS planner Agile Board which is represented in the following Figure 1.1. The Product Backlog consists of the complete user stories of Ai based E-learning Application

Each user story consists of necessary parameters like MoSCoW prioritization, Functional and non functional parameters, detailed acceptance criteria with linked tasks.



Backlog

+ Add task

○ Enter a task name * (required)

Set due date

Assign

Add task

USER STORY

○ US-3

- ○ Develop Household User Interface
- ○ Integrate Donation Matching System

1 1 / 3

In progress

+ Add task

Must Have USER STORY

○ US-1

- ○ Implement Real Time Inventory Tracking
- ○ Develop User Interface
- ○ Create donation and request Forms

! 3 0 / 3

Due AI

USER STORY

○ US-2

- ○ Develop Inventory Tracking System
- ○ Integrate With Recipient Organization Syster

1 1 / 3

○ verification

- ○ Add restaurant verification using GST Numb
- ○ Add NGO verification using govt approved f

! 0 / 2

Awaiting Review

+ Add task

Completed

+ Add task

Add a new bucket

Figure 1.1 MS Planner Board of ZEROhunger - Food Redistribution Platform

1.7 Product Release Plan

The following Figure 1.2 depicts the release plan of the project

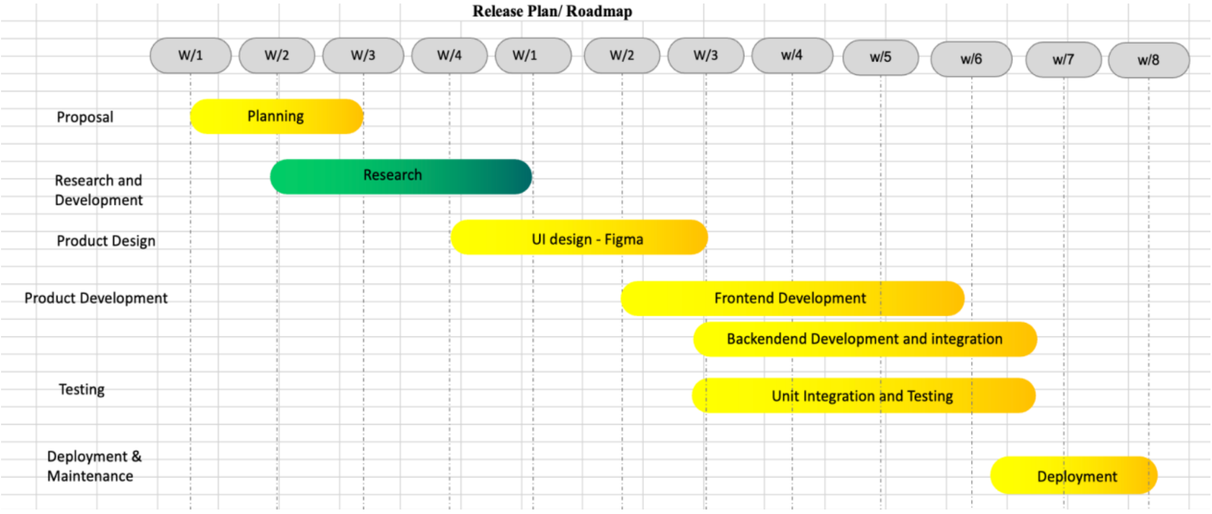


Figure 1.2 Release plan of AI E-learning Application

CHAPTER 2

SPRINT PLANNING AND EXECUTION

2.1 Sprint 1

2.1.1 Sprint Goal with User Stories of Sprint 1

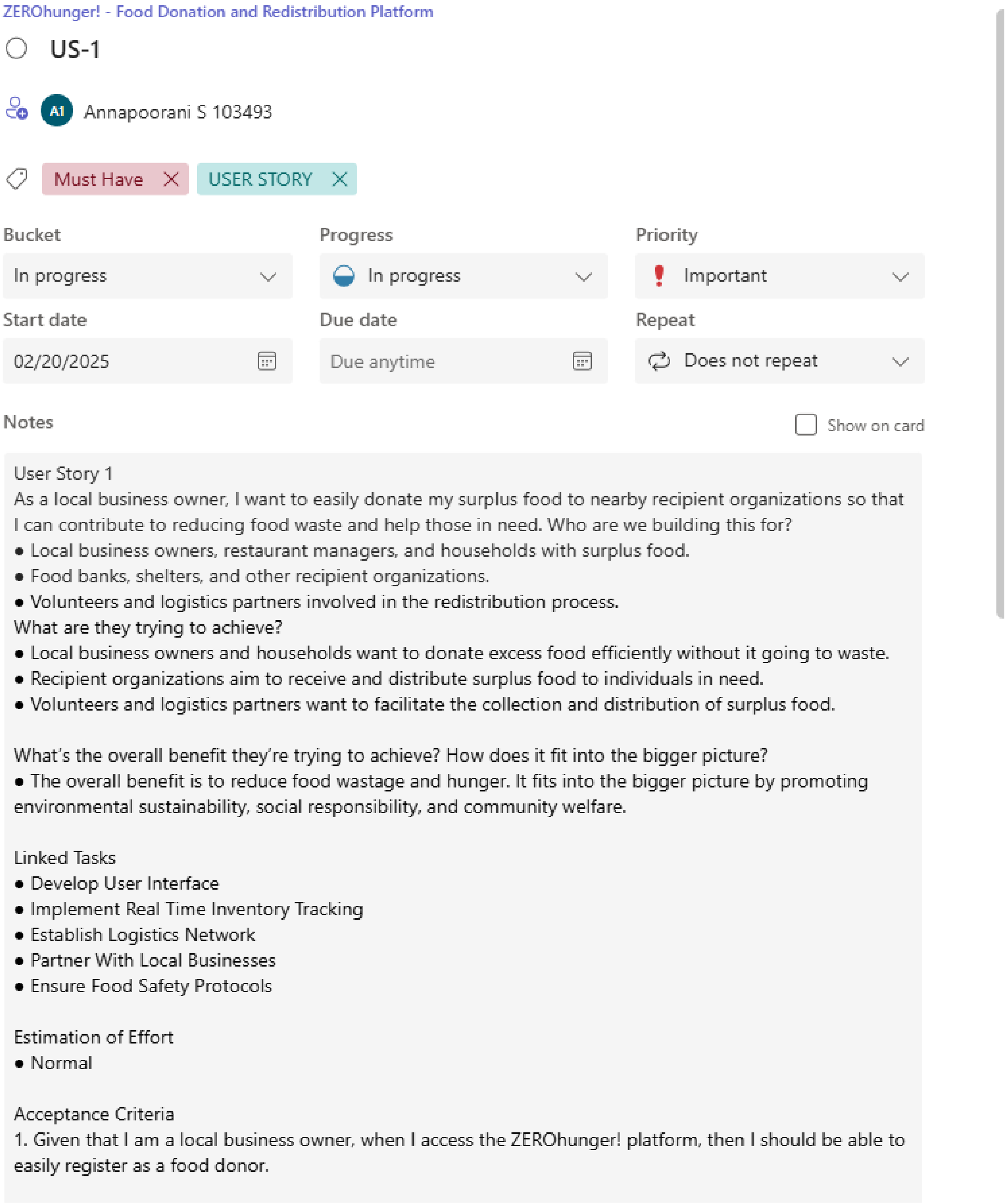
As a local business owner, I want to easily donate my surplus food to nearby recipient organizations so that I can contribute to reducing food waste and help those in need.

The following table 2.1 represents the detailed user stories of the sprint 1

Table 2.1 Detailed User Stories of sprint 1

S.NO	Detailed User Stories
US #1	As a restaurant owner, I must be able to register and login to the platform and access the restaurant dashboard

Planner Board representation of user story is mentioned below figure 2.1



ZEROhunger! - Food Donation and Redistribution Platform

○ US-1

Annapoorani S 103493

Must Have ✕ USER STORY ✕

Bucket	Progress	Priority
In progress	In progress	Important
Start date	Due date	Repeat
02/20/2025	Due anytime	Does not repeat

Notes □ Show on card

User Story 1
As a local business owner, I want to easily donate my surplus food to nearby recipient organizations so that I can contribute to reducing food waste and help those in need. Who are we building this for?

- Local business owners, restaurant managers, and households with surplus food.
- Food banks, shelters, and other recipient organizations.
- Volunteers and logistics partners involved in the redistribution process.

What are they trying to achieve?

- Local business owners and households want to donate excess food efficiently without it going to waste.
- Recipient organizations aim to receive and distribute surplus food to individuals in need.
- Volunteers and logistics partners want to facilitate the collection and distribution of surplus food.

What's the overall benefit they're trying to achieve? How does it fit into the bigger picture?

- The overall benefit is to reduce food wastage and hunger. It fits into the bigger picture by promoting environmental sustainability, social responsibility, and community welfare.

Linked Tasks

- Develop User Interface
- Implement Real Time Inventory Tracking
- Establish Logistics Network
- Partner With Local Businesses
- Ensure Food Safety Protocols

Estimation of Effort

- Normal

Acceptance Criteria
1. Given that I am a local business owner, when I access the ZEROhunger! platform, then I should be able to easily register as a food donor.

Figure 2.1 User story as a local business owner

2.1.2 Functional Document

2.1.2.1. Introduction

To create a platform called ZEROhunger! that facilitates the donation of surplus food from local businesses, restaurants, and households to recipient organizations such as food banks and shelters. The platform aims to reduce food waste and help those in need.

2.1.2.2. Product Goal

- Creating a user interface for ZEROhunger involves developing an intuitive platform where users can seamlessly interact (User Interface).
- This interface will include forms for both donations and requests, allowing users to either contribute resources or ask for help efficiently.
- Implementing real-time inventory tracking is crucial for maintaining an up-to-date overview of available resources, ensuring transparency and prompt response to needs.

2.1.2.3. Demography (Users, Location)

- Users: Local business owners, restaurant managers, households with surplus food, food banks, shelters, volunteers, and logistics partners.
- Location: Initially focusing on Chengalpattu, Tamil Nadu, India, with potential for expansion to other regions.

2.1.2.4. Business Processes

1. Donation Registration:

○ Local businesses and households register as food donors on the platform by providing their details and creating an account.

2. Surplus Food Entry:

○ Donors enter the details of the surplus food available for donation, including the type, quantity, and best-before date.

3. Matching Process:

○ The platform matches the surplus food donations with nearby recipient organizations based on criteria such as type of food needed, proximity, and capacity.

2.1.2.5. Features

Feature #1: User Interface for Food Donors

5.1.Description: A user-friendly interface for food donors to register and enter surplus food details, as well as for recipient organizations to accept donations and schedule distributions.

5.2.User Story: As a local business owner, I want to easily donate my surplus food to nearby recipient organizations so that I can contribute to reducing food waste and help those in need.

2.1.2.6. Authorization Matrix

Table 2.2 Access level Authorization Matrix

Role	Access Level
Local Business Owner	Register as food donor
Households	Register as food donor
Recipient Organization	Accept donations and schedule distribution
Logistics Partner	coordinate pickup and delivery

2.1.2.7. Assumptions

- All users have internet access and can use the platform via web or mobile devices.
- Recipient organizations have the capacity to store and distribute surplus food.
- Logistics partners are available to facilitate the collection and distribution process.

2.1.3 Architecture Document

2.1.3.1. Application

Microservices:

The ZEROhunger platform adopts a microservices architecture to ensure scalability, flexibility, and ease of deployment. Each major functionality is encapsulated within its own service. Key services include:

- AuthenticationService: Handles user registration, login, role-based authentication (Restaurant or NGO), and password recovery. May include two-factor authentication for enhanced security.

- Food Donation Service: Manages the creation, listing, and lifecycle of food donations. Restaurants can post surplus food details, which NGOs can view and accept.
- User Role Management Service: Manages user roles and permissions, ensuring restaurants and NGOs have access to appropriate functionality and UI views.

2.1.3.2 System Architecture-

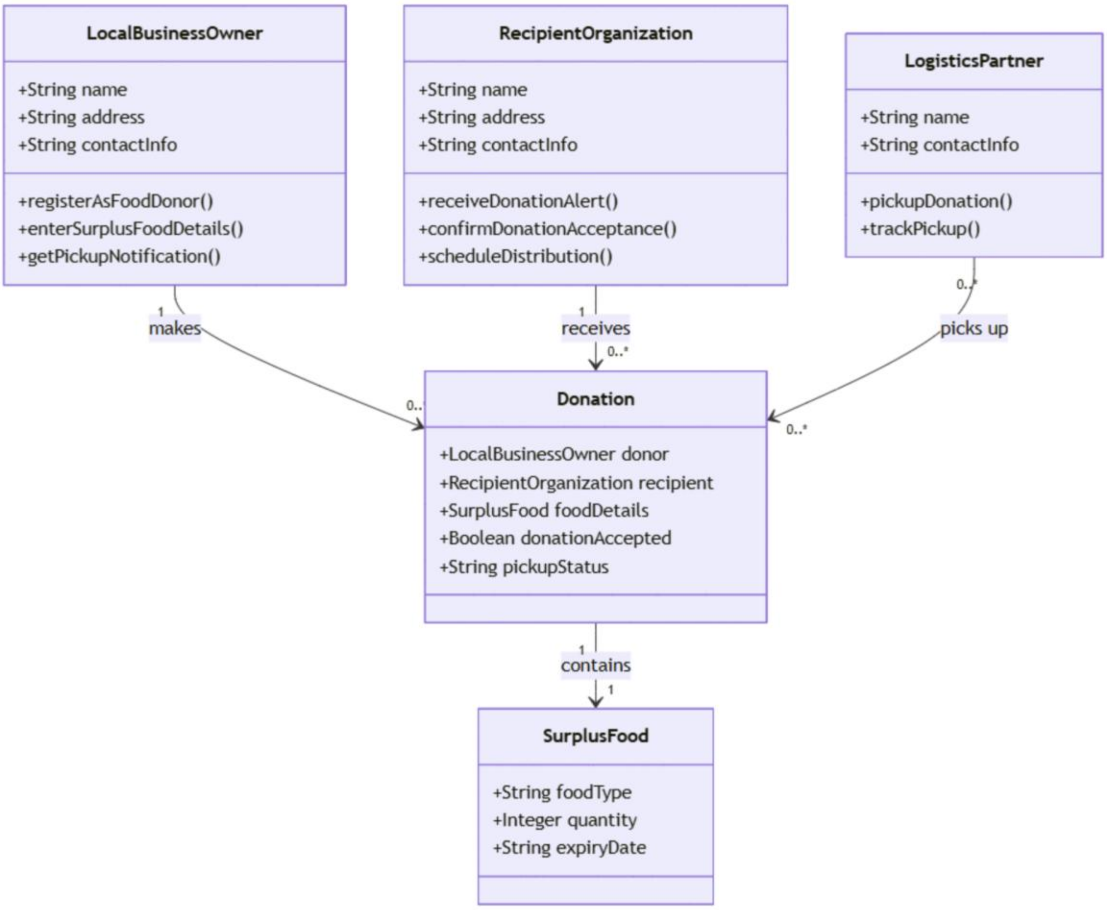


Figure 2.1.3.2 System Architecture Diagram

2.1.3.3. Data Exchange Contract:

Frequency of Data Exchanges:

Real-Time Exchanges:

- User Authentication: Login and registration operations occur in real-time using secure RESTful APIs.

- Donation and Matching: When a restaurant posts a donation, NGOs see updates immediately. Acceptances and rejections are handled in real-time.

Periodic Syncs:

- Analytics Data: Summary reports of donation history, NGO activities, and system health metrics are synced periodically (daily or weekly).

- Logs and Audit Trails: User activity logs are batched and uploaded at regular intervals.

Data Sets:

User Data:

- Includes names, email addresses, contact details, login credentials, and roles (Restaurant/NGO).

- Exchanged during login, profile updates, and chatbot interactions.

Donation Data:

- Contains food item details (type, quantity, expiry), images, and pickup location.

- Shared during creation, update, and confirmation stages of the donation lifecycle.

Match and Request Data:

- Includes request timestamps, acceptance statuses, NGO pickup confirmations.

- Used for transaction logs and matching algorithms.

Mode of Exchanges (API, File, Queue, etc.) :

API (RESTful): Used for synchronous, real-time exchanges between front-end and microservices (e.g., login, create donation, request food).

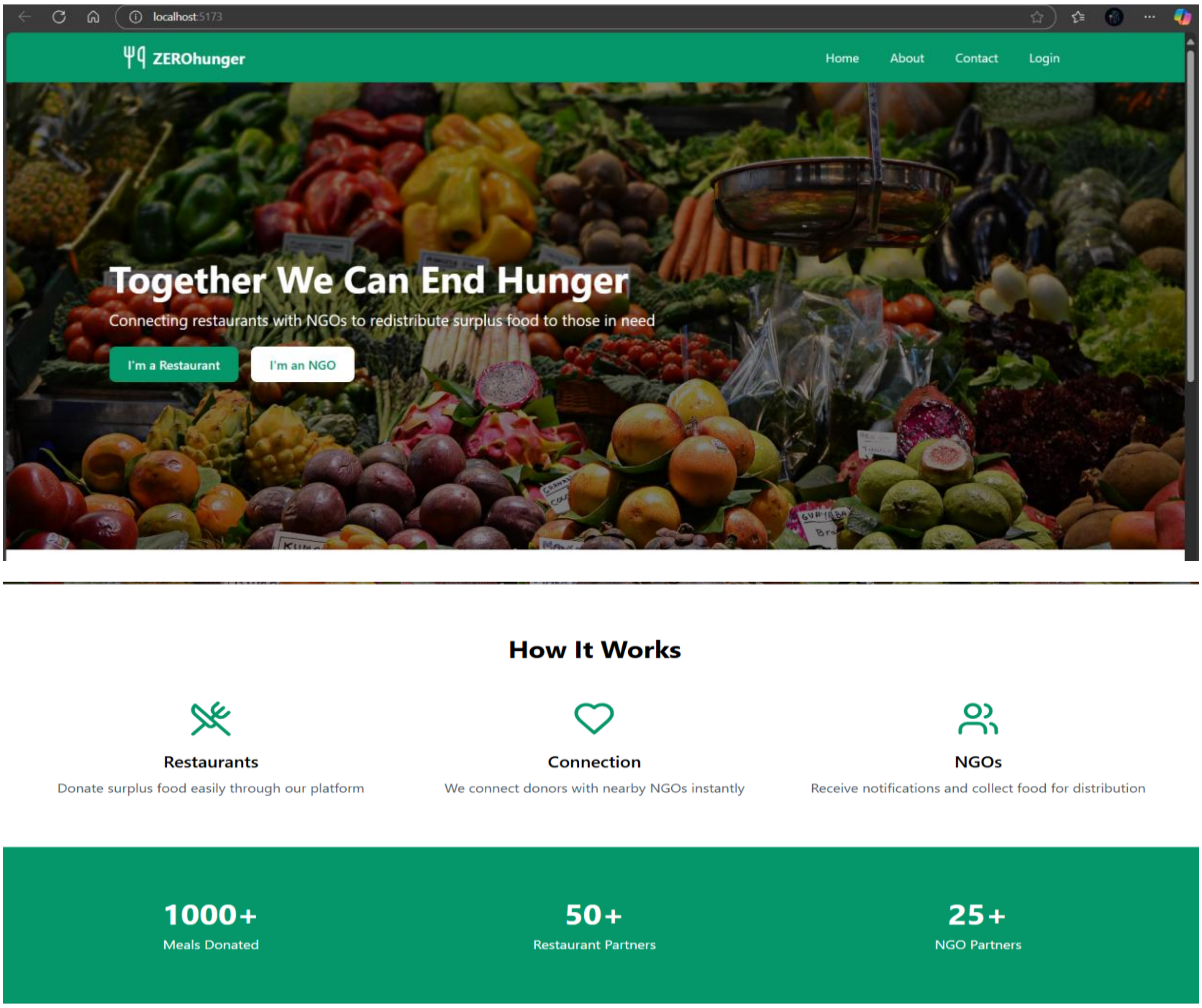
Message Queues: Services like RabbitMQ or AWS SQS manage asynchronous tasks such as:

- Sending confirmation messages
- Triggering chatbot responses
- Logging NGO pickup completions

File-Based Exchanges: Large uploads like:

- Donation photos (stored on AWS S3 or similar)
- Bulk NGO onboarding files (CSV)

2.1.4 UI DESIGN



Together We Can End Hunger

Connecting restaurants with NGOs to redistribute surplus food to those in need

How It Works

Restaurants

Donate surplus food easily through our platform

Connection

We connect donors with nearby NGOs instantly

NGOs

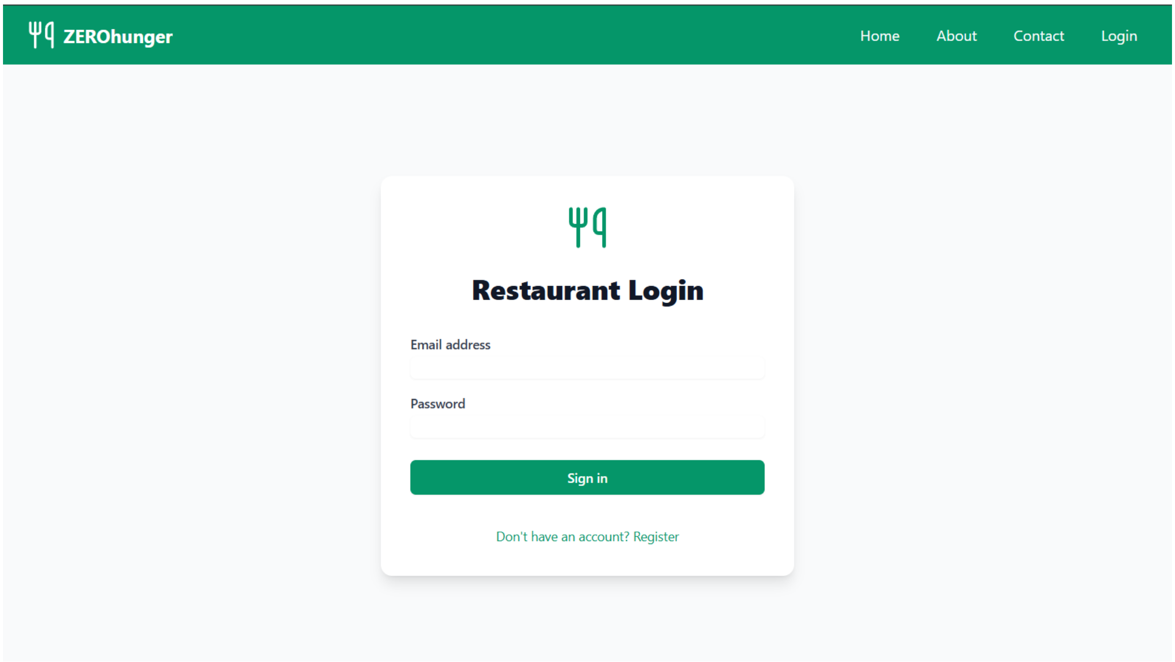
Receive notifications and collect food for distribution

1000+
Meals Donated

50+
Restaurant Partners

25+
NGO Partners

fig 2.1.4.1 Home page



ZEROhunger

Home About Contact Login

Restaurant Login

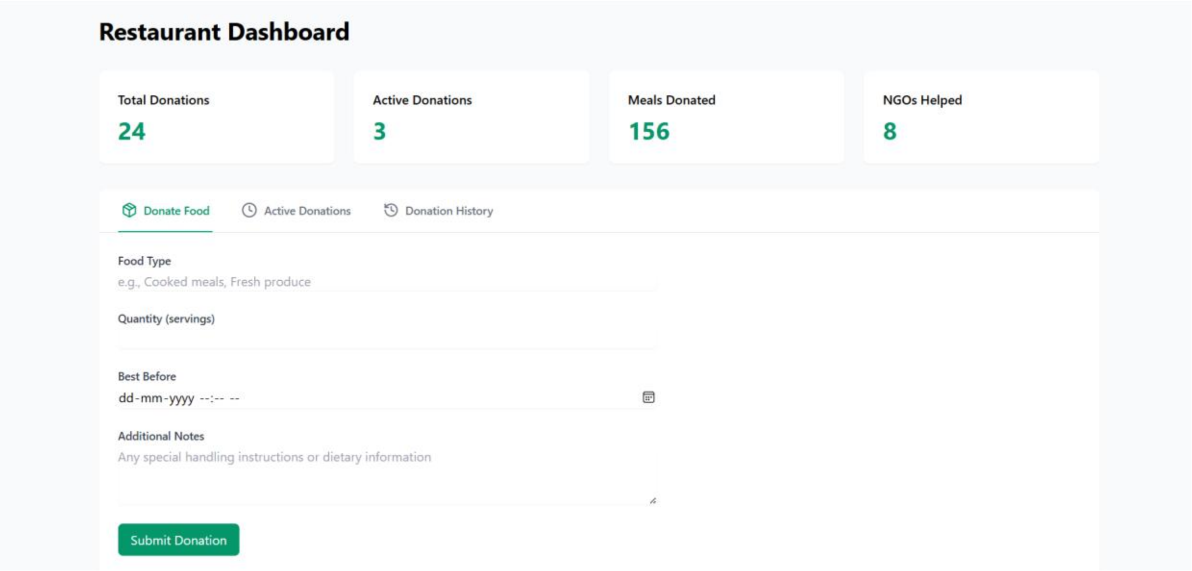
Email address

Password

Sign in

Don't have an account? Register

fig 2.1.4.2 Restaurant login page



Restaurant Dashboard

Total Donations	Active Donations	Meals Donated	NGOs Helped
24	3	156	8

Donate Food | Active Donations | Donation History

Food Type

e.g., Cooked meals, Fresh produce

Quantity (servings)

Best Before

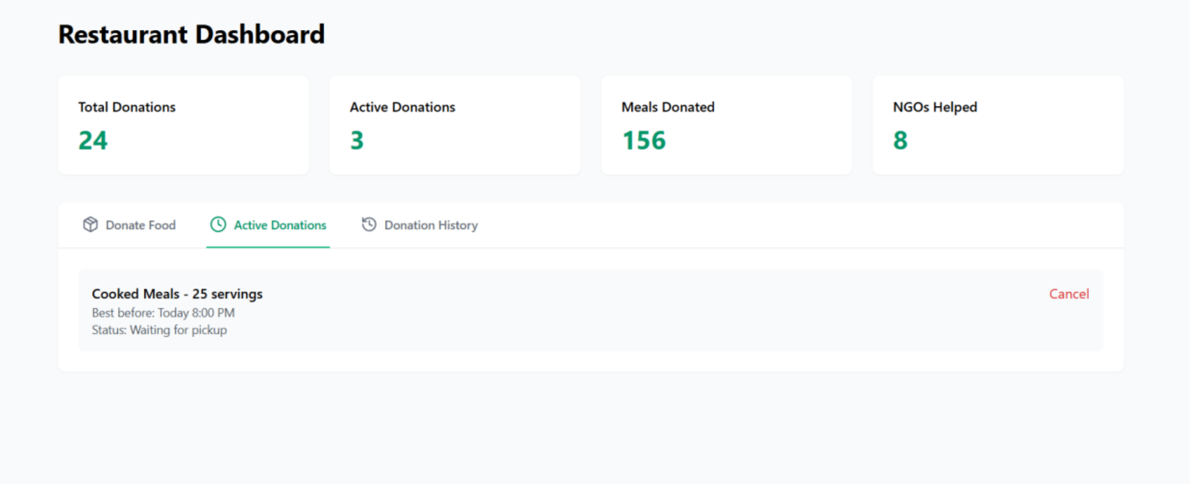
dd-mm-yyyy --:-- --

Additional Notes

Any special handling instructions or dietary information

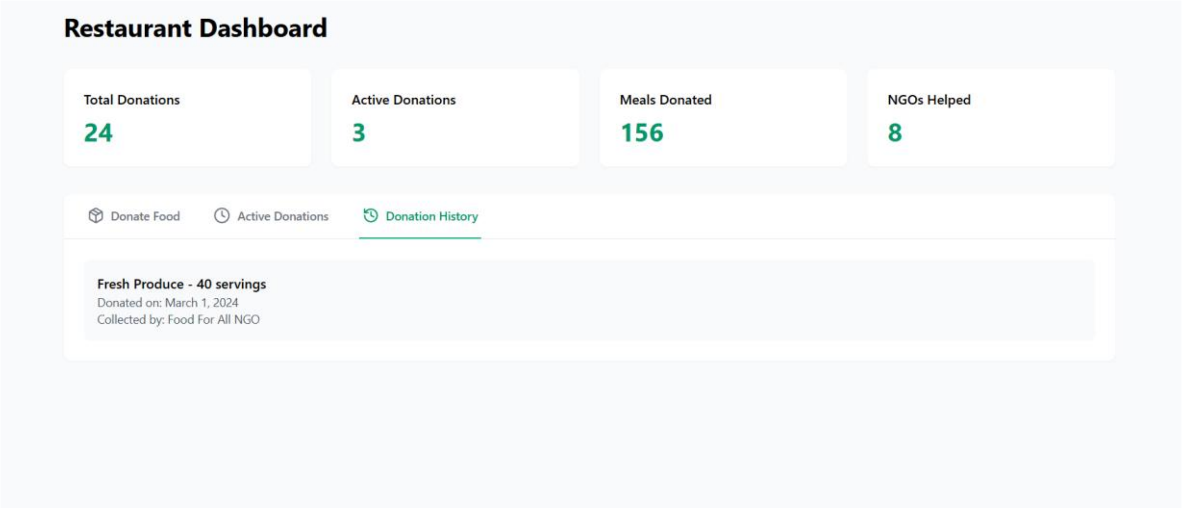
Submit Donation

fig 2.1.4.3 Donation form


Restaurant Dashboard

Cooked Meals - 25 servings
Best before: Today 8:00 PM
Status: Waiting for pickup

fig 2.1.4.4 Active donations

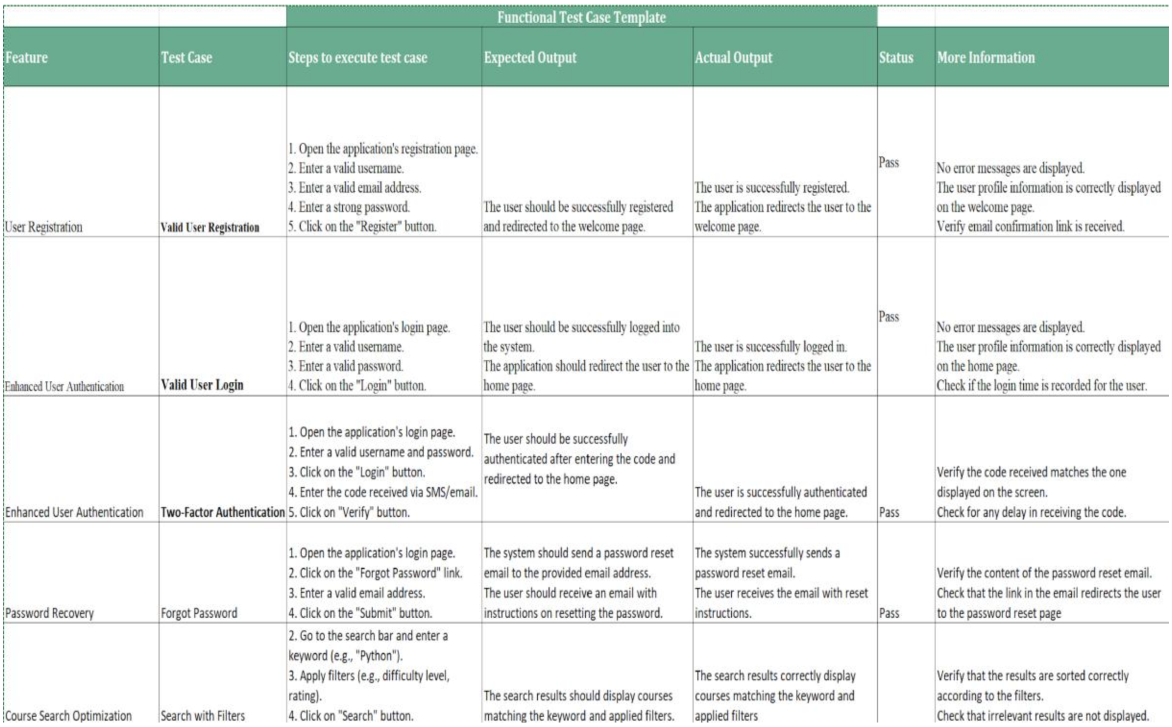

Restaurant Dashboard

Fresh Produce - 40 servings
Donated on: March 1, 2024
Collected by: Food For All NGO

fig 2.1.4.5 Donation history

2.1.5 Functional Test Cases

Table 2.3 Detailed Functional Test Case



		Functional Test Case Template				
Feature	Test Case	Steps to execute test case	Expected Output	Actual Output	Status	More Information
User Registration	**Valid User Registration**	1. Open the application's registration page. 2. Enter a valid username. 3. Enter a valid email address. 4. Enter a strong password. 5. Click on the "Register" button.	The user should be successfully registered and redirected to the welcome page.	The user is successfully registered. The application redirects the user to the welcome page.	Pass	No error messages are displayed. The user profile information is correctly displayed on the welcome page. Verify email confirmation link is received.
Enhanced User Authentication	**Valid User Login**	1. Open the application's login page. 2. Enter a valid username. 3. Enter a valid password. 4. Click on the "Login" button.	The user should be successfully logged into the system. The application should redirect the user to the home page.	The user is successfully logged in. The application redirects the user to the home page.	Pass	No error messages are displayed. The user profile information is correctly displayed on the home page. Check if the login time is recorded for the user.
Enhanced User Authentication	**Two-Factor Authentication**	1. Open the application's login page. 2. Enter a valid username and password. 3. Click on the "Login" button. 4. Enter the code received via SMS/email. 5. Click on "Verify" button.	The user should be successfully authenticated after entering the code and redirected to the home page.	The user is successfully authenticated and redirected to the home page.	Pass	Verify the code received matches the one displayed on the screen. Check for any delay in receiving the code.
Password Recovery	Forgot Password	1. Open the application's login page. 2. Click on the "Forgot Password" link. 3. Enter a valid email address. 4. Click on the "Submit" button.	The system should send a password reset email to the provided email address. The user should receive an email with instructions on resetting the password.	The system successfully sends a password reset email. The user receives the email with reset instructions.	Pass	Verify the content of the password reset email. Check that the link in the email redirects the user to the password reset page
Course Search Optimization	Search with Filters	2. Go to the search bar and enter a keyword (e.g., "Python"). 3. Apply filters (e.g., difficulty level, rating). 4. Click on "Search" button.	The search results should display courses matching the keyword and applied filters.	The search results correctly display courses matching the keyword and applied filters		Verify that the results are sorted correctly according to the filters. Check that irrelevant results are not displayed.

2.1.6 Committed Vs Completed User Stories

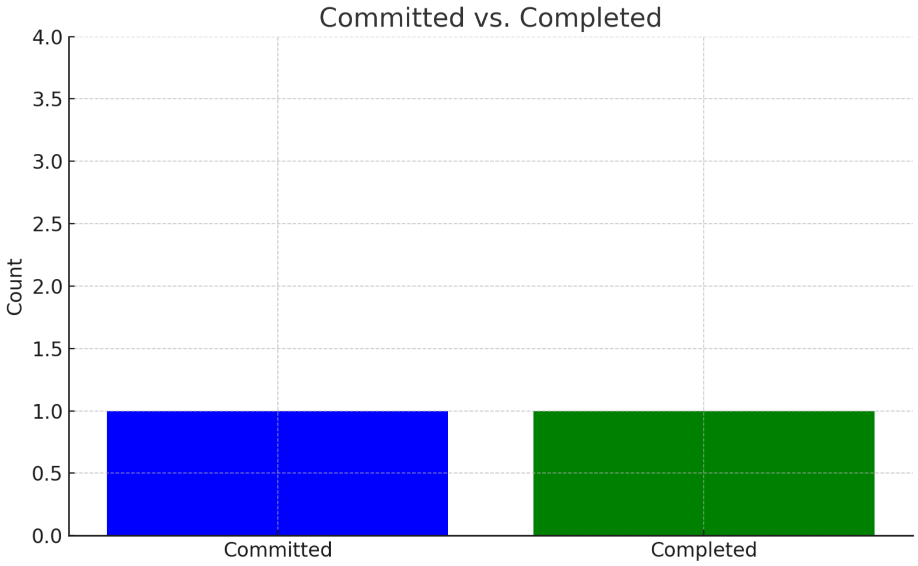
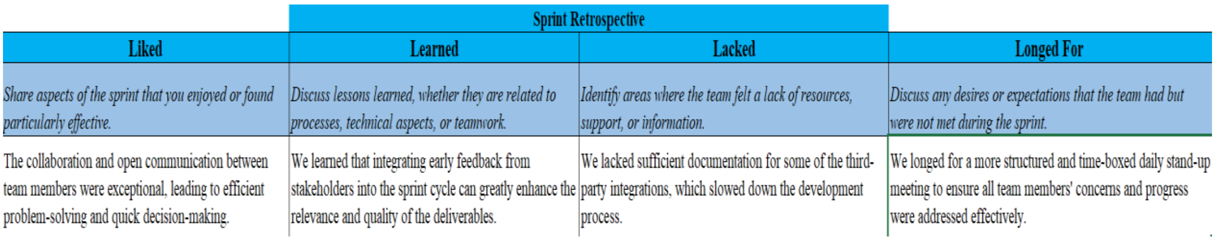


Figure 2.1.6 Bar graph for Committed Vs Completed User Stories

2.1.8 Sprint Retrospective



	Sprint Retrospective		
Liked	**Learned**	**Lacked**	**Longed For**
Share aspects of the sprint that you enjoyed or found particularly effective.	*Discuss lessons learned, whether they are related to processes, technical aspects, or teamwork.*	*Identify areas where the team felt a lack of resources, support, or information.*	*Discuss any desires or expectations that the team had but were not met during the sprint.*
The collaboration and open communication between team members were exceptional, leading to efficient problem-solving and quick decision-making.	We learned that integrating early feedback from stakeholders into the sprint cycle can greatly enhance the relevance and quality of the deliverables.	We lacked sufficient documentation for some of the third-party integrations, which slowed down the development process.	We longed for a more structured and time-boxed daily stand-up meeting to ensure all team members' concerns and progress were addressed effectively.

Figure 2.1.8 Sprint Retrospective for the Sprint 1

2.2 SPRINT 2

As a recipient organization manager, I want to easily track incoming food donations and inventory levels so that I can efficiently manage and distribute food to those in need.

2.2.1 Sprint Goal with User Stories of Sprint 2

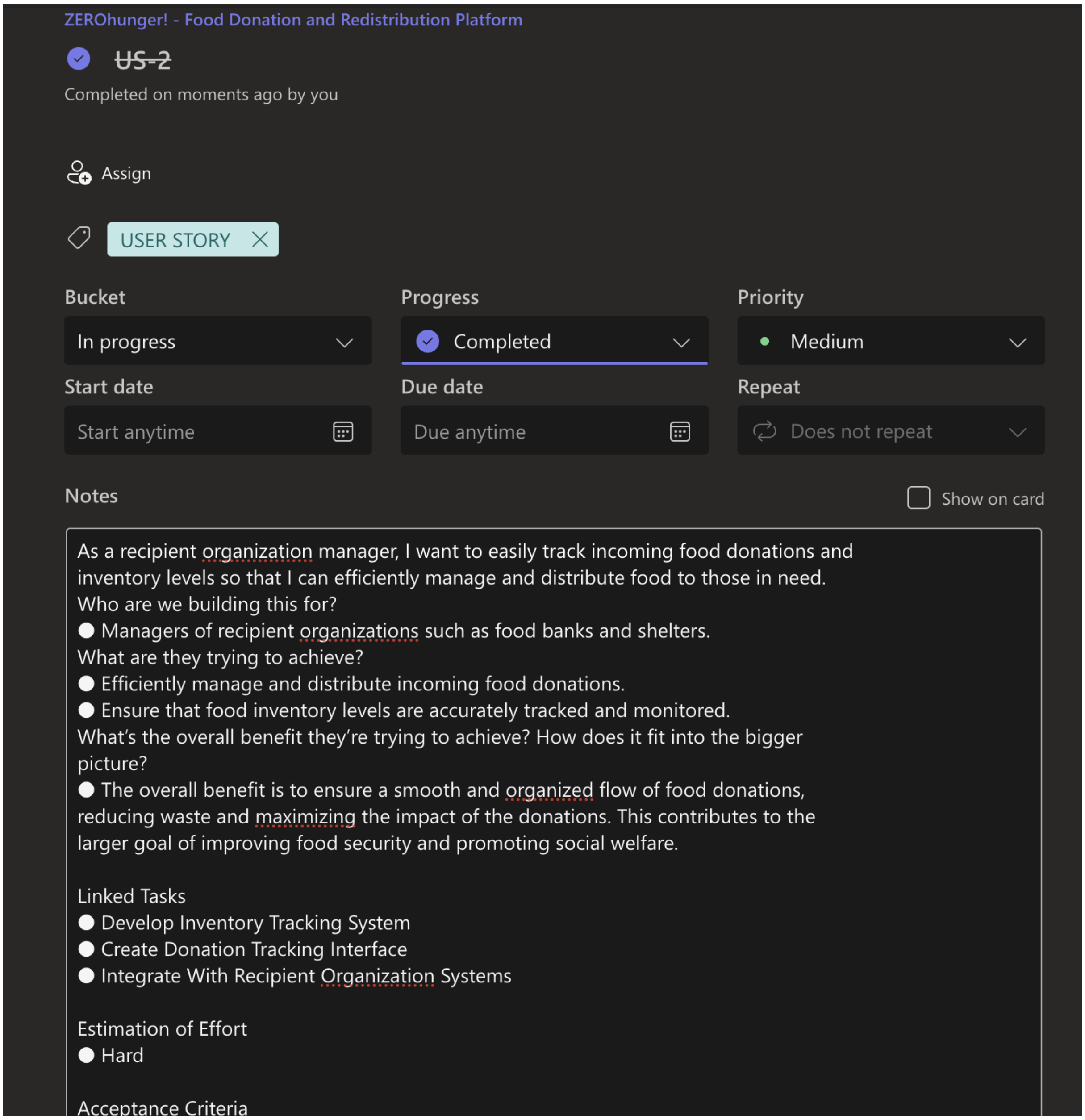
The Goal of the second sprint is to enable recipient organization managers to efficiently track incoming food donations and manage inventory for effective food distribution.

The following Table 2.2.1 represents the detailed user stories of the sprint 2

Table 2.2.1 Detailed User Stories of sprint 2

S No	Detailed User Stories
US #2	As a recipient organization manager, I want to easily track incoming food donations and inventory levels so that I can efficiently manage and distribute food to those in need.

Planner Board representation of user stories are mentioned below figures 2.2.1



ZEROhunger! - Food Donation and Redistribution Platform

~~US-2~~

Completed on moments ago by you

Bucket: In progress

Progress: Completed

Priority: Medium

Start date: Start anytime

Due date: Due anytime

Repeat: Does not repeat

Notes

□ Show on card

As a recipient organization manager, I want to easily track incoming food donations and inventory levels so that I can efficiently manage and distribute food to those in need.

Who are we building this for?

- Managers of recipient organizations such as food banks and shelters.

What are they trying to achieve?

- Efficiently manage and distribute incoming food donations.
- Ensure that food inventory levels are accurately tracked and monitored.

What's the overall benefit they're trying to achieve? How does it fit into the bigger picture?

- The overall benefit is to ensure a smooth and organized flow of food donations, reducing waste and maximizing the impact of the donations. This contributes to the larger goal of improving food security and promoting social welfare.

Linked Tasks

- Develop Inventory Tracking System
- Create Donation Tracking Interface
- Integrate With Recipient Organization Systems

Estimation of Effort

- Hard

Acceptance Criteria

Figure 2.2.1 MS Planner for US2

2.2.2 Functional Document

To extend the capabilities of the ZEROhunger! platform by introducing dedicated features for **recipient organization managers**, enabling them to track incoming food donations, manage inventory levels, and efficiently distribute food to those in need. This sprint ensures NGOs have the tools they need to maintain food flow and reduce waste.

2.2.2.1 Product Goal

To empower recipient organizations with a streamlined, real-time NGO dashboard that allows:

- Viewing and managing incoming food donations
- Real-time inventory tracking
- Generating reports and managing food distribution efficiently
- Submitting food requests when necessary

2.2.2.2 Demography (Users, Location)

Users:
Recipient organization managers from food banks, shelters, and non-profit kitchens.

Location:
Chengalpattu, Tamil Nadu, India (initial deployment); scalable for regional expansion across India.

2.2.2.3 Business Processes

NGO Registration:

Recipient organizations register by providing their name, contact person, address, and authentication credentials.

NGO Login:

Registered NGOs can log in using secure authentication to access their personalized dashboard.

Donation Tracking:

NGOs receive a list of available or incoming donations based on proximity, need, and food suitability. They can accept, reject, or schedule pickups.

2.2.2.4 Features

Feature 1: NGO Registration & Login

5.1.Description:
A secure system allowing NGOs to register and log in to access platform functionalities.

5.2. User Story:
As an NGO manager, I want to register and log into the platform so I can manage incoming food donations and inventory.

Feature 2: NGO Dashboard

5.1. Description:
An intuitive dashboard showing real-time data on donations, current inventory levels, pending pickups, and food request forms.

5.2. User Story:
As an NGO manager, I want to view and track all incoming food donations and inventory levels to efficiently manage food distribution.

Feature 3: Inventory Management System

5.1. Description:
Tracks food received and available stock, with filters for expiry dates, food types, and quantities.

5.2. User Story:
As an NGO manager, I want the system to automatically update inventory after accepting a donation and allow me to view stock levels.

Feature 4: Request Form

5.1. Description:
Allows NGOs to request specific types or quantities of food when current inventory is insufficient.

5.2. User Story:
As an NGO manager, I want to submit food requests to the platform when our stock is low.

2.2.2.4 Assumptions

- NGOs and other users have access to stable internet via web/mobile.
- Recipient organizations have sufficient storage and staff to manage and distribute food.
- Logistics partners are onboarded and available for pickups and deliveries.
- Donations are vetted and match basic food safety guidelines before being listed or accepted.

2.2.3 Architecture Document

2.2.3.1. Application

Microservices:

The ZEROhunger platform is built on a microservices architecture, where distinct functionalities are encapsulated within independent services. Key services for this sprint include:

- NGO Authentication Service: Manages registration, secure login, session management using JWT, and role-based authentication for recipient organizations.
- NGO Dashboard Service: Provides a real-time dashboard for recipient organizations to view donation activity, current inventory, and manage food distribution.
- Donation Tracker Service: Handles the processing of incoming donations from donors. Allows recipient organizations to accept or reject donations and schedule pickups.
- Inventory Management Service: Updates inventory in real-time once a donation is accepted. Tracks food items, quantities, expiration dates, and availability.
- Request Management Service: Enables recipient organizations to request specific food items based on need and generate reports on requests fulfilled.
- Notification Service: Sends real-time alerts and updates for donation status, request approvals, and low-stock warnings.

2.2.3.2. System Architecture

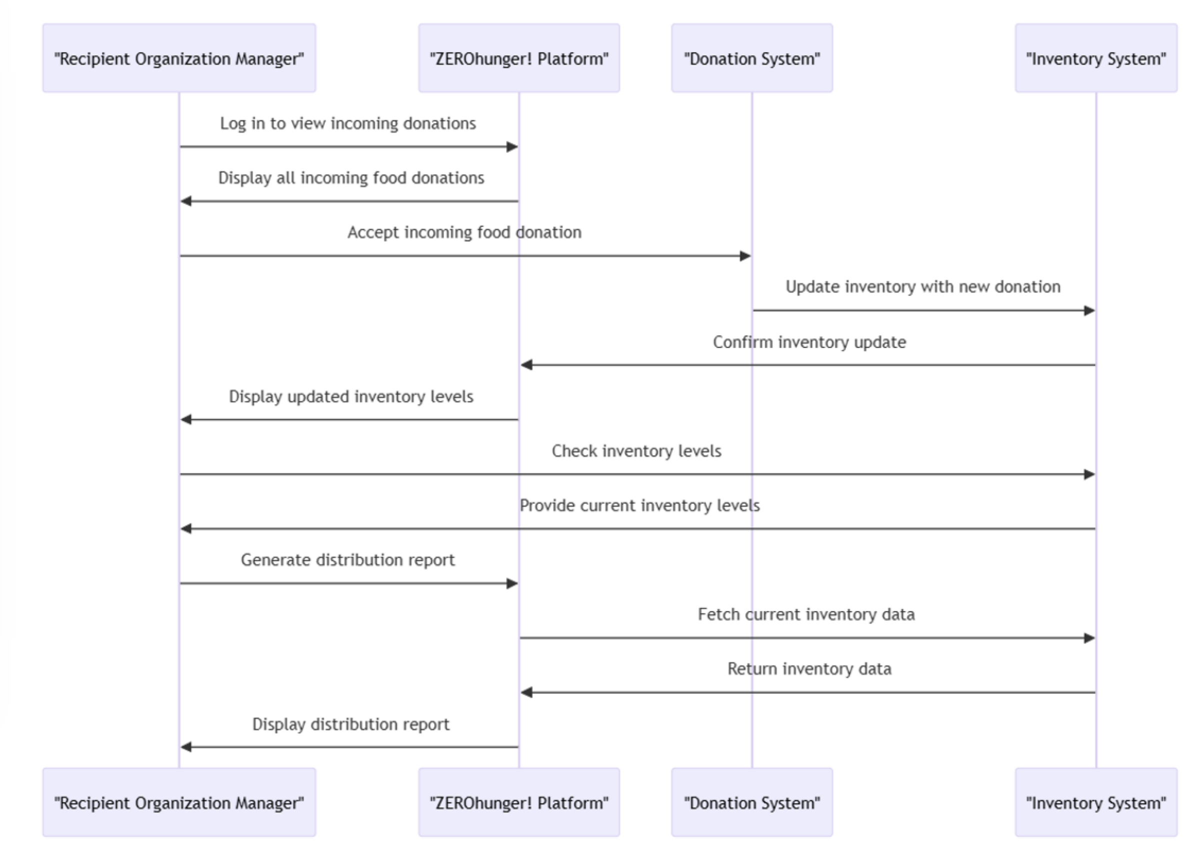

"Recipient Organization Manager"
"ZEROhunger! Platform"
"Donation System"
"Inventory System"
Log in to view incoming donations
Display all incoming food donations
Accept incoming food donation
Update inventory with new donation
Confirm inventory update
Display updated inventory levels
Check inventory levels
Provide current inventory levels
Generate distribution report
Fetch current inventory data
Return inventory data
Display distribution report
"Recipient Organization Manager"
"ZEROhunger! Platform"
"Donation System"
"Inventory System"

Figure 2.2.3.2 System Architecture

2.2.3.3. Data Exchange Contract

Frequency of Data Exchanges:

Data exchanges are managed with performance and efficiency in mind. Real-time exchanges occur for critical operations such as NGO login, donation acceptance, and inventory updates. These ensure the platform remains accurate and responsive. Periodic synchronizations handle non-critical updates like report generation, donation trends, and data archiving, typically executed on a daily or weekly basis.

Data Sets:

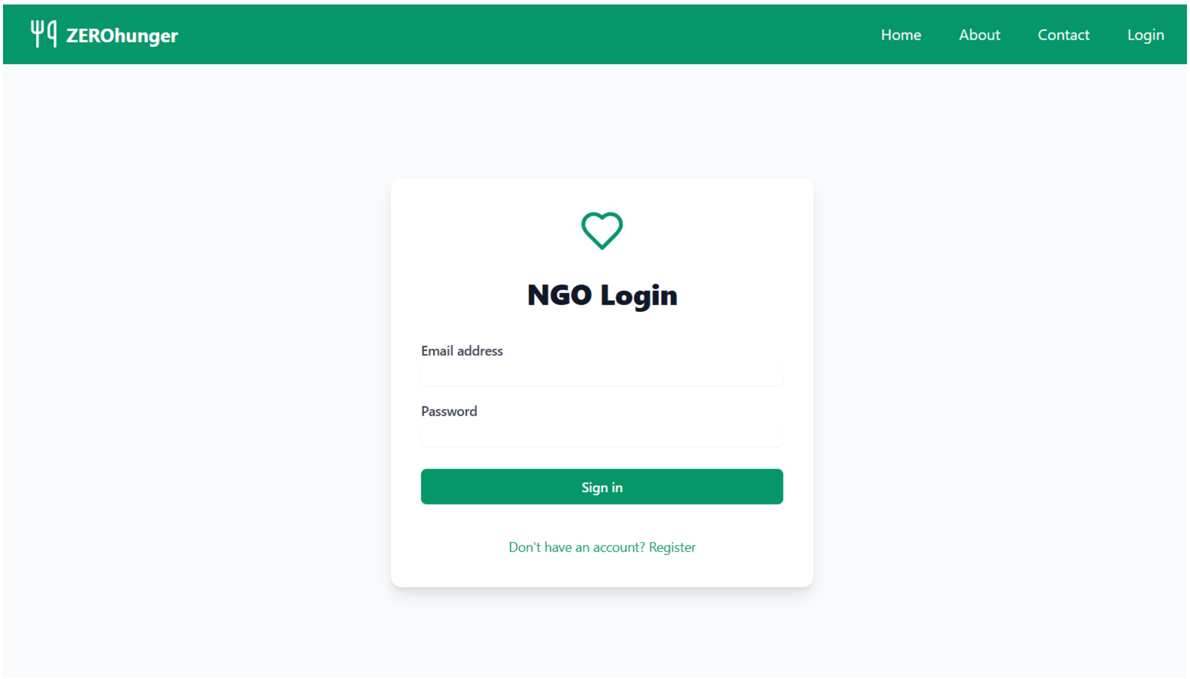
The platform processes several important types of data:

- User Data involves registration information, login credentials, role assignments, and organizational profiles. This data is exchanged during the NGO registration, login, and while accessing dashboards.
- Donation Data includes detailed records of food donations such as item type, quantity, location, expiry, and donor identity. It is exchanged when a donor submits a new entry and when NGOs interact with these donations.
- Inventory Data tracks the live status of accepted food items, including current stock levels, usage history, and expiry timelines. It is actively updated after each donation or food distribution.
- Request Data contains structured information on what recipient organizations need, such as type of food, quantity, urgency, and timestamps. This is used to facilitate matching with surplus donations and enable smarter distribution.

Mode of Exchanges (API, File, Queue, etc.):

- API (RESTful): Used for all front-end to back-end communication, such as form submissions, dashboard views, and real-time updates.
- Message Queues: Employed for asynchronous tasks like notifications, scheduled report generation, and data cleanup jobs.
- File-Based Exchanges: Used for bulk data uploads (e.g., donation histories or NGO request logs) and to generate downloadable reports stored in cloud storage platforms like AWS S3.

2.2.4 UI Design

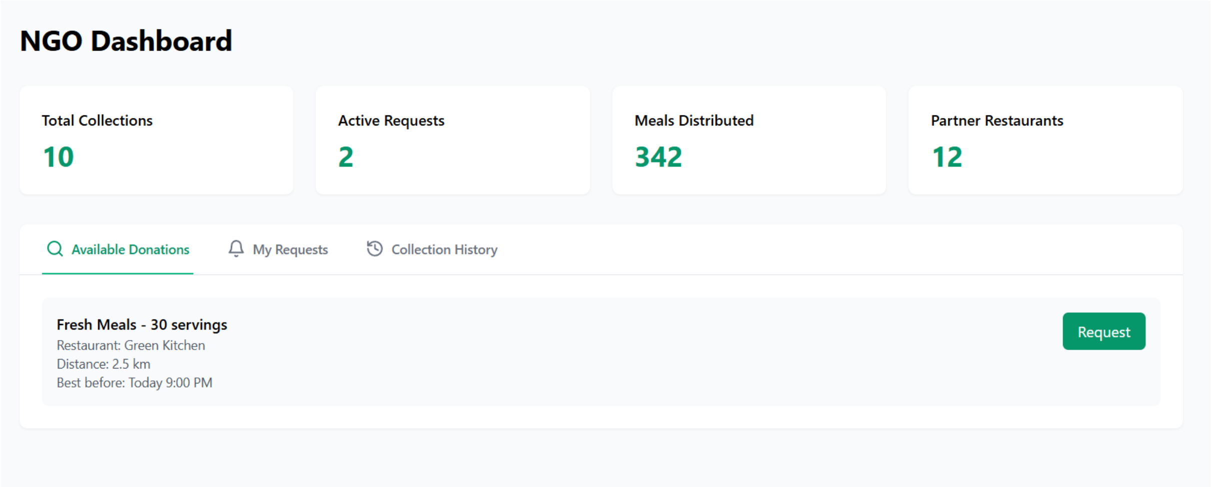


NGO Login

Email address

Password

fig 2.2.4.1 NGO login page



NGO Dashboard

Total Collections 10

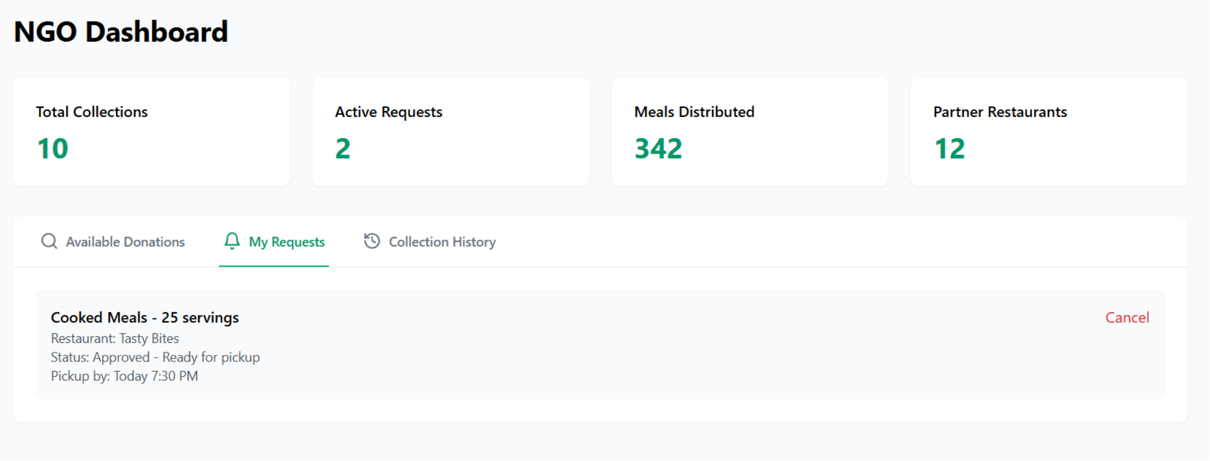
Active Requests 2

Meals Distributed 342

Partner Restaurants 12

Fresh Meals - 30 servings
Restaurant: Green Kitchen
Distance: 2.5 km
Best before: Today 9:00 PM

fig 2.2.4.2 Available Donations



NGO Dashboard

Total Collections
10

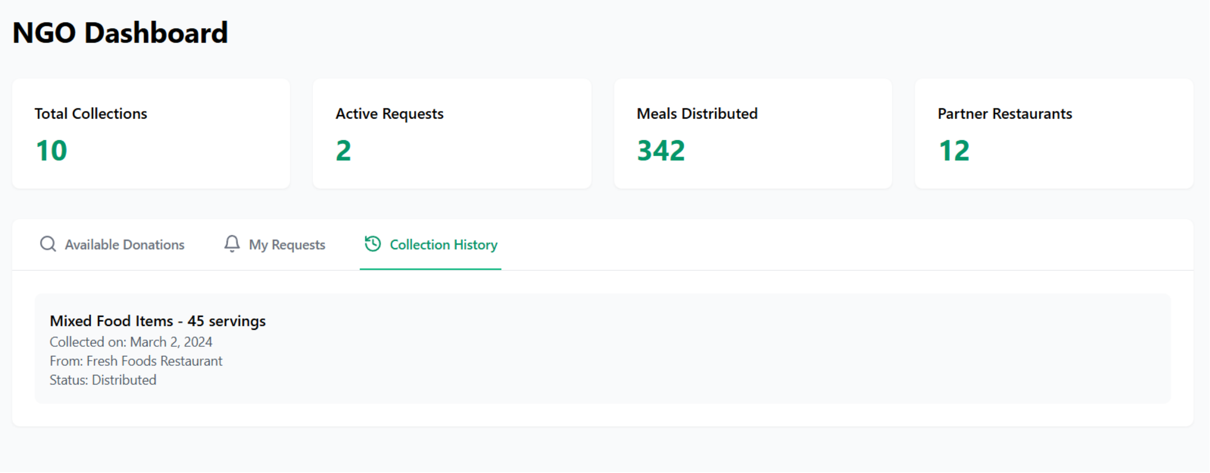
Active Requests
2

Meals Distributed
342

Partner Restaurants
12

Cooked Meals - 25 servings
Restaurant: Tasty Bites
Status: Approved - Ready for pickup
Pickup by: Today 7:30 PM

fig 2.2.4.3 My Requests



NGO Dashboard

Total Collections
10

Active Requests
2

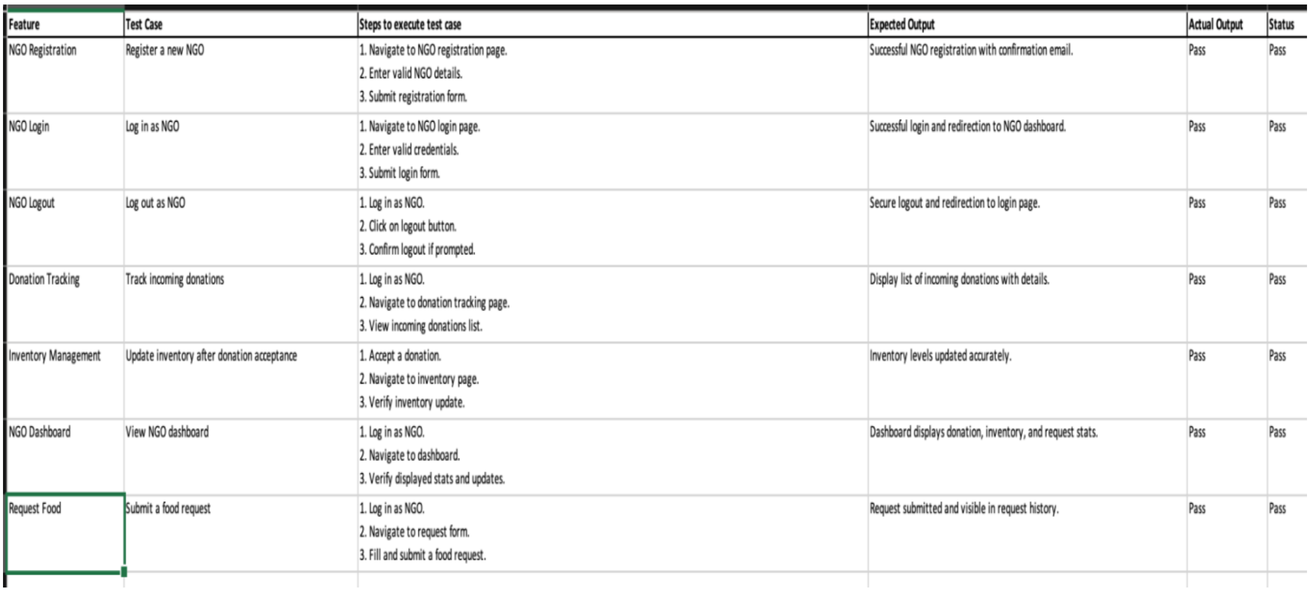
Meals Distributed
342

Partner Restaurants
12

Mixed Food Items - 45 servings
Collected on: March 2, 2024
From: Fresh Foods Restaurant
Status: Distributed

fig 2.2.4.4 Collection History

2.2.5 Functional Test Cases



Feature	Test Case	Steps to execute test case	Expected Output	Actual Output	Status
NGO Registration	Register a new NGO	1. Navigate to NGO registration page. 2. Enter valid NGO details. 3. Submit registration form.	Successful NGO registration with confirmation email.	Pass	Pass
NGO Login	Log in as NGO	1. Navigate to NGO login page. 2. Enter valid credentials. 3. Submit login form.	Successful login and redirection to NGO dashboard.	Pass	Pass
NGO Logout	Log out as NGO	1. Log in as NGO. 2. Click on logout button. 3. Confirm logout if prompted.	Secure logout and redirection to login page.	Pass	Pass
Donation Tracking	Track incoming donations	1. Log in as NGO. 2. Navigate to donation tracking page. 3. View incoming donations list.	Display list of incoming donations with details.	Pass	Pass
Inventory Management	Update inventory after donation acceptance	1. Accept a donation. 2. Navigate to inventory page. 3. Verify inventory update.	Inventory levels updated accurately.	Pass	Pass
NGO Dashboard	View NGO dashboard	1. Log in as NGO. 2. Navigate to dashboard. 3. Verify displayed stats and updates.	Dashboard displays donation, inventory, and request stats.	Pass	Pass
Request Food	Submit a food request	1. Log in as NGO. 2. Navigate to request form. 3. Fill and submit a food request.	Request submitted and visible in request history.	Pass	Pass

Figure 2.2.5 Functional Test Cases

2.2.6 COMMITTED Vs COMPLETED USER STORIES

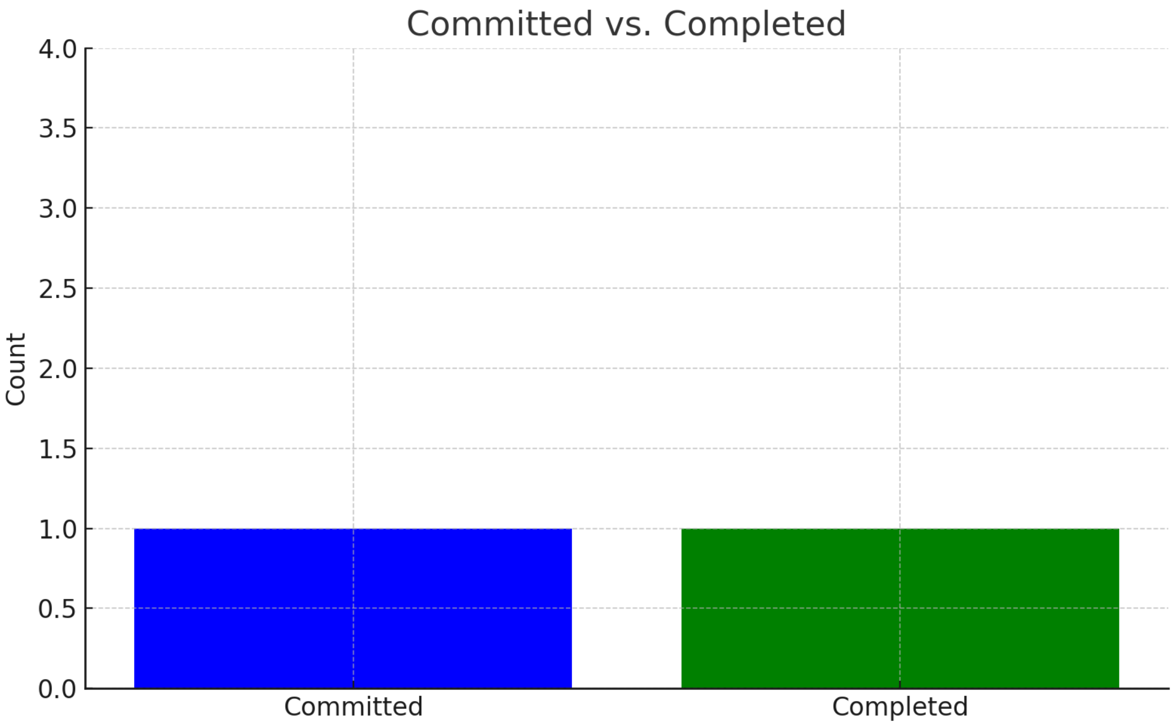
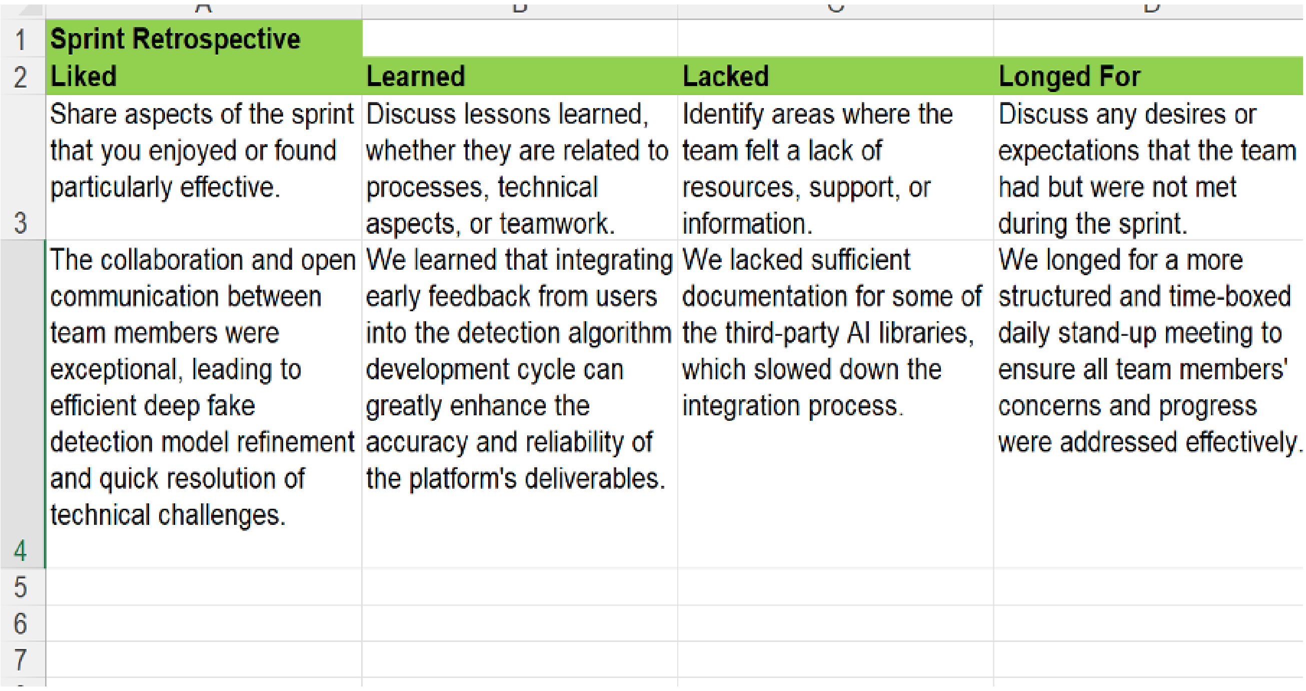


Figure 2.2.6 Committed vs Completed user stories

2.2.7 Sprint Retrospective



Sprint Retrospective

Liked	Learned	Lacked	Longed For
Share aspects of the sprint that you enjoyed or found particularly effective.	Discuss lessons learned, whether they are related to processes, technical aspects, or teamwork.	Identify areas where the team felt a lack of resources, support, or information.	Discuss any desires or expectations that the team had but were not met during the sprint.
The collaboration and open communication between team members were exceptional, leading to efficient deep fake detection model refinement and quick resolution of technical challenges.	We learned that integrating early feedback from users into the detection algorithm development cycle can greatly enhance the accuracy and reliability of the platform's deliverables.	We lacked sufficient documentation for some of the third-party AI libraries, which slowed down the integration process.	We longed for a more structured and time-boxed daily stand-up meeting to ensure all team members' concerns and progress were addressed effectively.

Figure 2.2.7 Sprint Retrospective

2.3 Sprint 3

Added GSTIN and document verification for Restaurants, NGO verification, Admin dashboard , geolocation based tracking for NGOs to detect nearby restaurant donations, map feature.

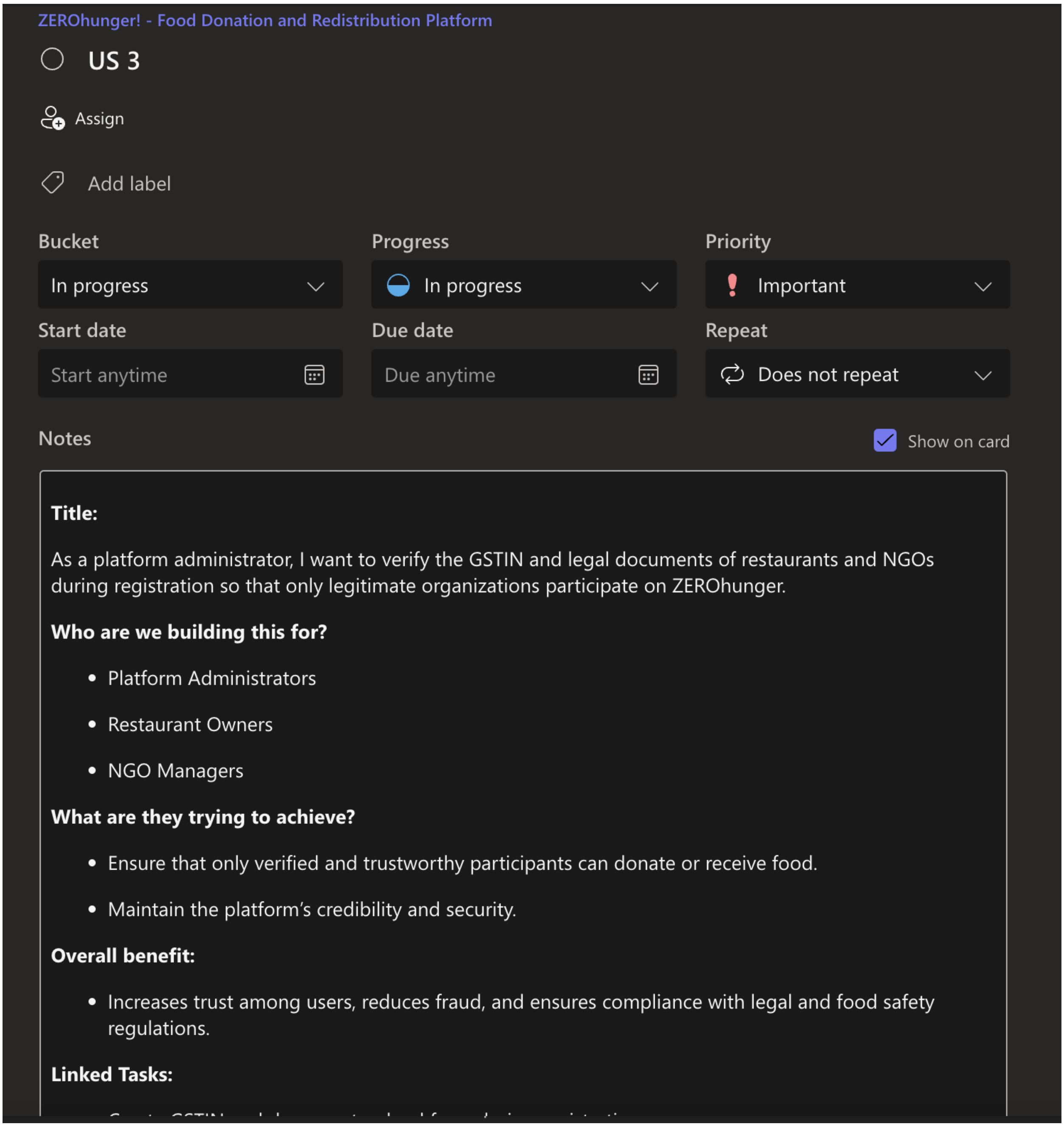
2.3.1 Sprint Goal with User Stories of Sprint 3

The Goal of this sprint is to Strengthen user verification and enhance operational efficiency on ZEROhunger! by implementing GSTIN/document verification, building an Admin dashboard, and adding geolocation-based tracking with map integration for NGOs.

Table 2.3.1 Detailed User Stories of sprint 3

S.NO	Detailed User Stories
US #3	As a platform administrator, I want to verify the GSTIN and legal documents of restaurants and NGOs during registration so that only legitimate organizations participate on ZEROhunger.
US#4	As an NGO manager, I want to view nearby restaurant donations on a map so that I can quickly claim and collect surplus food donations.

Planner Board representation of user stories are mentioned below figures 2.3.1 and 2.3.2



ZEROhunger! - Food Donation and Redistribution Platform

○ US 3

Bucket: In progress

Progress: In progress

Priority: ❗ Important

Start date: Start anytime

Due date: Due anytime

Repeat: Does not repeat

Notes

☑ Show on card

Title:

As a platform administrator, I want to verify the GSTIN and legal documents of restaurants and NGOs during registration so that only legitimate organizations participate on ZEROhunger.

Who are we building this for?

- Platform Administrators
- Restaurant Owners
- NGO Managers

What are they trying to achieve?

- Ensure that only verified and trustworthy participants can donate or receive food.
- Maintain the platform’s credibility and security.

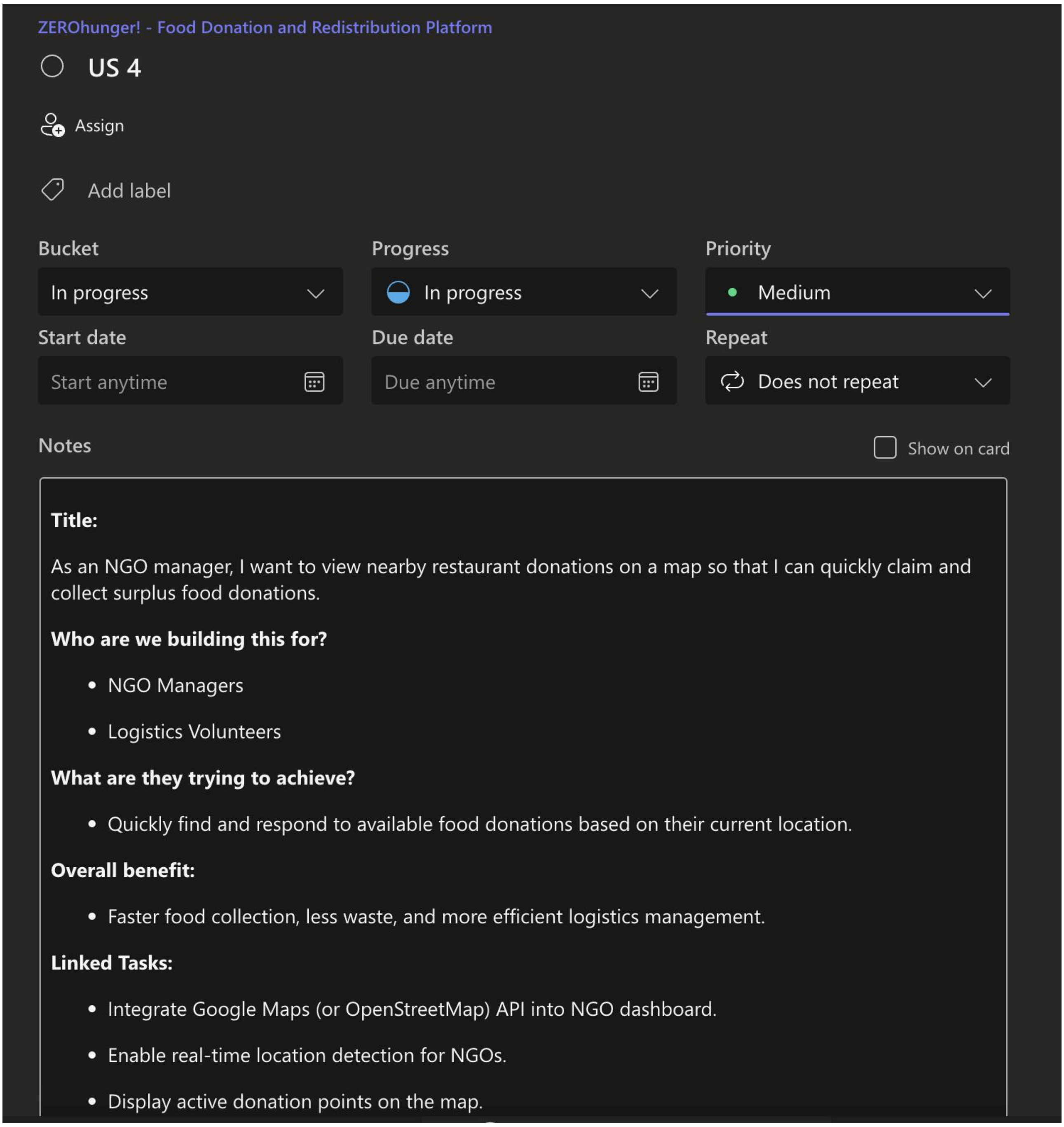
Overall benefit:

- Increases trust among users, reduces fraud, and ensures compliance with legal and food safety regulations.

Linked Tasks:

- Create GSTIN and document upload form during registration

Figure 2.3.1.1 User story 3


ZEROhunger! - Food Donation and Redistribution Platform

○ US 4

Assign

Add label

Bucket: In progress

Progress: In progress

Priority: ● Medium

Start date: Start anytime

Due date: Due anytime

Repeat: Does not repeat

Notes

□ Show on card

Title:

As an NGO manager, I want to view nearby restaurant donations on a map so that I can quickly claim and collect surplus food donations.

Who are we building this for?

- NGO Managers
- Logistics Volunteers

What are they trying to achieve?

- Quickly find and respond to available food donations based on their current location.

Overall benefit:

- Faster food collection, less waste, and more efficient logistics management.

Linked Tasks:

- Integrate Google Maps (or OpenStreetMap) API into NGO dashboard.
- Enable real-time location detection for NGOs.
- Display active donation points on the map.

Figure 2.3.1.2 User story 4

2.3.2 Functional Document

2.3.2.1 Introduction

This sprint focuses on strengthening the credibility and operational efficiency of the ZEROhunger! platform by verifying the identities of restaurants and NGOs through GSTIN/document verification, and enhancing the NGO experience through geolocation tracking and map-based donation discovery.

2.3.2.2 Product Goal

Enable secure onboarding through document verification and improve real-time visibility of nearby food donations for NGOs using geolocation and map integration.

2.3.2.3 Demography (Users, Location)

Users: Restaurant Managers, NGO Managers, Platform Administrators

Location: Initially focusing on Chengalpattu, Tamil Nadu, India, with scope for scaling across different cities.

2.3.2.4 Business Processes

1. GSTIN and Document Verification

- Registration Process: Restaurants and NGOs must upload relevant legal documents (GSTIN for restaurants, certifications for NGOs) during registration.
- Admin Verification: Uploaded documents are reviewed by the admin team. Verified users are allowed to access the platform fully. If rejected, users are notified to re-upload.
- Status Update: Users can track their verification status on their dashboards (Pending / Verified / Rejected).

2. Geolocation-Based Tracking and Map Feature

- NGO Dashboard: NGOs can access a map showing active restaurant donation listings based on their current location.
- Real-Time Tracking: NGOs will see nearby donations with distance filters and food type labels.
- Claiming Donations: NGOs can click on a map marker to view donation details and initiate a claim for pickup.

2.3.2.5 Features

Feature #1: GSTIN and Document Verification

1. Description

Allow restaurants and NGOs to upload necessary documents at the time of registration, and facilitate admin-driven document verification.

2. User Story

As a platform administrator, I want to verify the GSTIN and legal documents of restaurants and NGOs during registration so that only legitimate organizations participate.

Feature #2: Geolocation-Based Tracking and Map Integration

1. Description

Enable NGOs to view and claim nearby restaurant donations using an integrated map based on their current location.

2. User Story

As an NGO manager, I want to view nearby restaurant donations on a map so that I can quickly claim and collect surplus food donations.

2.3.2.6 Assumptions

- All users have access to a device with internet and location services enabled.
- Admins are available to review and verify documents within a standard SLA.
- Restaurants and NGOs agree to upload authentic and valid documentation.
- Google Maps or OpenStreetMap API is available for integration without major cost or license issues.

2.3.3 Architecture Document

2.3.3.1 Application

Microservices:

The ZEROhunger! platform continues to use a microservices-based architecture, where each functionality is independently developed, deployed, and scaled. Key microservices added in this sprint include:

- Authentication Service: Manages user registration, login, and token-based session management.
- Document Verification Service: Handles GSTIN verification for restaurants and legal document verification for NGOs. Includes status tracking (Pending, Verified, Rejected).
- Admin Dashboard Service: Provides an interface for admins to review uploaded documents, manage user approvals, and view platform statistics.
- Geolocation Service: Manages the capture and processing of NGO location data and donation location data.
- Map Service: Integrates external APIs (e.g., Google Maps or OpenStreetMap) to display restaurant donation locations and allow NGOs to claim nearby donations.
- Notification Service: Sends automated updates to users regarding document verification status and available nearby donations.

2.3.3.2. System Architecture

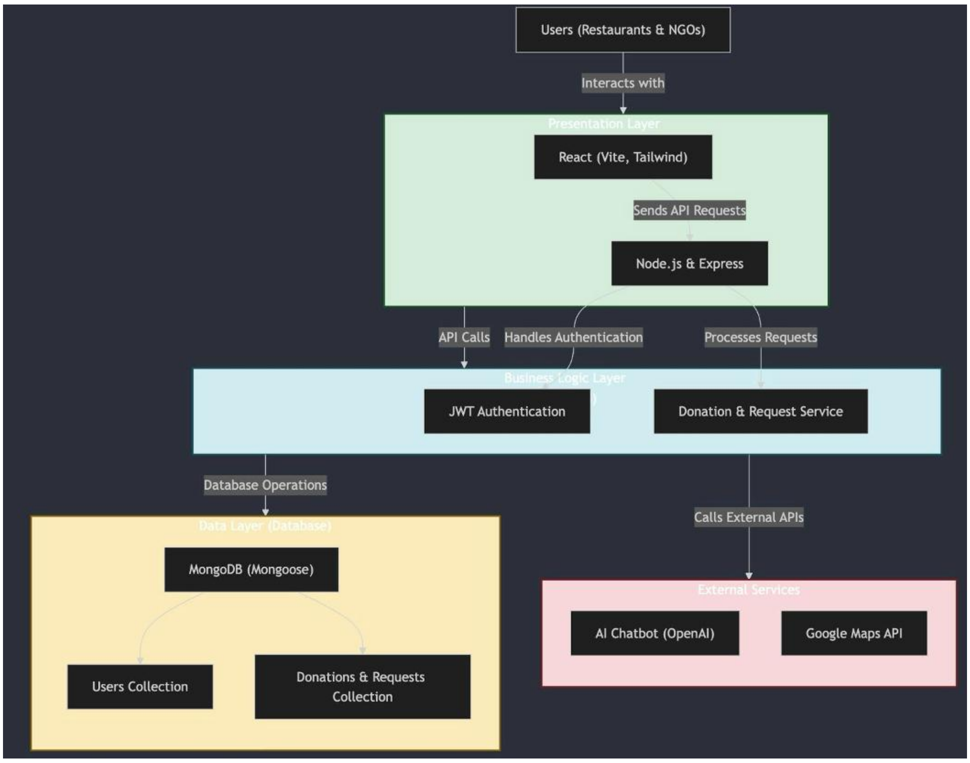


Figure 2.3.3.2 System Architecture

2.3.3.2. Data Exchange Contract

Frequency of Data Exchanges:

- Real-Time Exchanges:
  - User authentication and registration.
  - Document upload and verification status updates.
  - Real-time NGO location updates and restaurant donation mapping.
  - Notifications for donation claims.
- Periodic Syncs:
  - Admin dashboard analytics (e.g., daily/weekly summary reports).
  - Donation and user activity logs for audit purposes.

Data Sets:

- User Data:
  - Includes basic profiles, uploaded documents (GSTIN, NGO certifications), verification status.
- Donation Data:
  - Details of food donations by restaurants, including type, quantity, location, and expiry.

- Geolocation Data:
  - Current latitude/longitude of NGOs and location of restaurants offering donations.
- Verification Data:
  - Document metadata, verification timestamps, and admin remarks.

2.3.4 UI Design

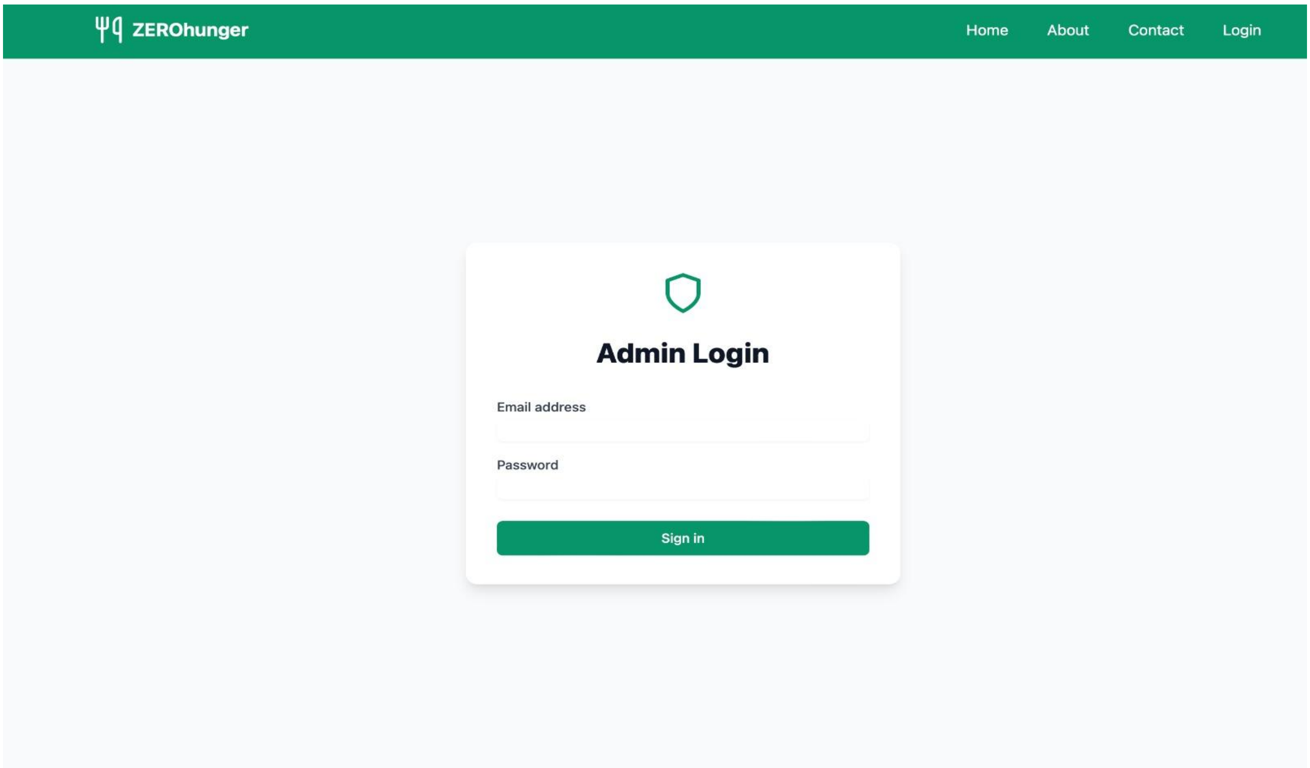
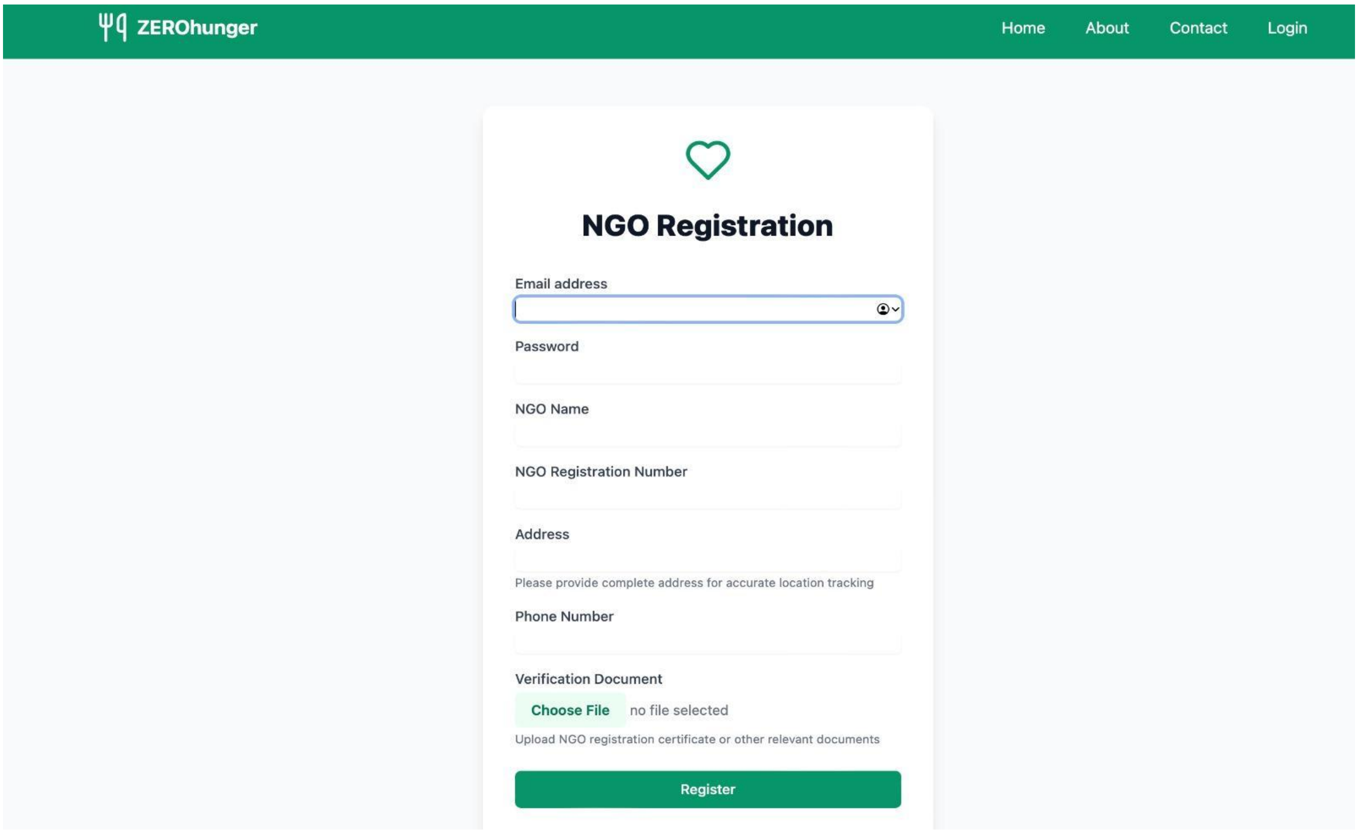


fig 2.3.4.1 Admin login page



NGO Registration

Email address

Password

NGO Name

NGO Registration Number

Address

Please provide complete address for accurate location tracking

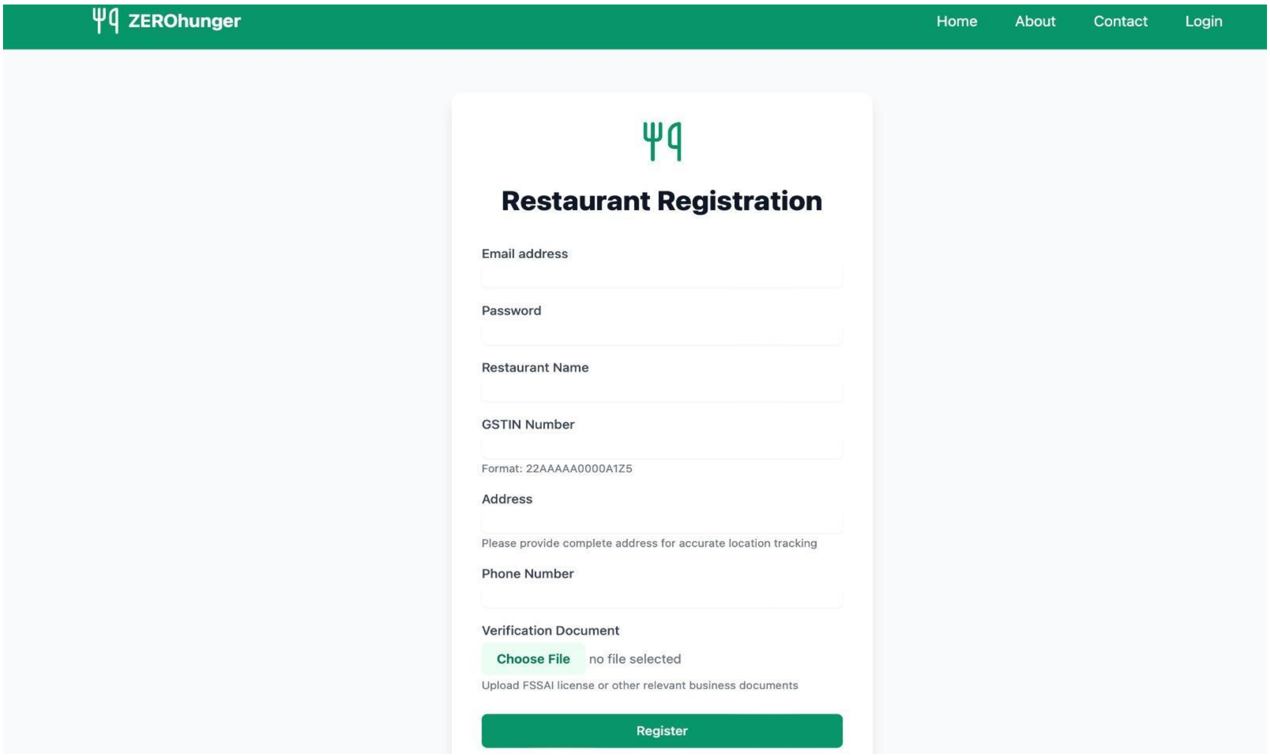
Phone Number

Verification Document

no file selected

Upload NGO registration certificate or other relevant documents

fig 2.3.4.2 NGO registration page



Restaurant Registration

Email address

Password

Restaurant Name

GSTIN Number

Format: 22AAAAA0000A1Z5

Address

Please provide complete address for accurate location tracking

Phone Number

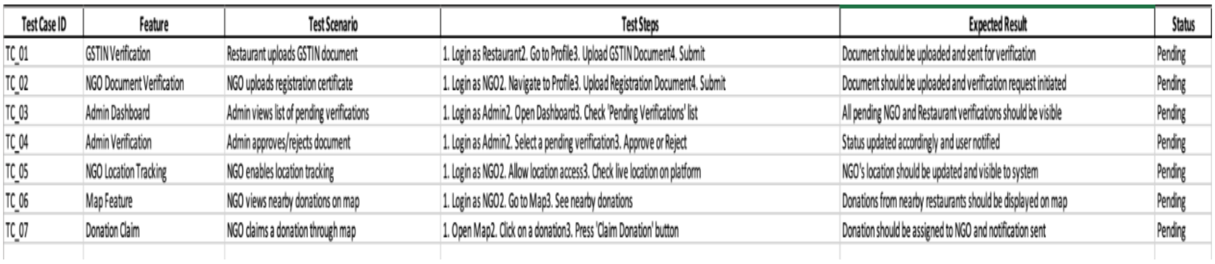
Verification Document

no file selected

Upload FSSAI license or other relevant business documents

fig 2.3.4.3 Restaurant registration page

2.3.5 Functional Test Cases



Test Case ID	Feature	Test Scenario	Test Steps	Expected Result	Status
TC_01	GSTIN Verification	Restaurant uploads GSTIN document	1. Login as Restaurant2. Go to Profile3. Upload GSTIN Document4. Submit	Document should be uploaded and sent for verification	Pending
TC_02	NGO Document Verification	NGO uploads registration certificate	1. Login as NGO2. Navigate to Profile3. Upload Registration Document4. Submit	Document should be uploaded and verification request initiated	Pending
TC_03	Admin Dashboard	Admin views list of pending verifications	1. Login as Admin2. Open Dashboard3. Check 'Pending Verifications' list	All pending NGO and Restaurant verifications should be visible	Pending
TC_04	Admin Verification	Admin approves/rejects document	1. Login as Admin2. Select a pending verification3. Approve or Reject	Status updated accordingly and user notified	Pending
TC_05	NGO Location Tracking	NGO enables location tracking	1. Login as NGO2. Allow location access3. Check live location on platform	NGO's location should be updated and visible to system	Pending
TC_06	Map Feature	NGO views nearby donations on map	1. Login as NGO2. Go to Map3. See nearby donations	Donations from nearby restaurants should be displayed on map	Pending
TC_07	Donation Claim	NGO claims a donation through map	1. Open Map2. Click on a donation3. Press 'Claim Donation' button	Donation should be assigned to NGO and notification sent	Pending

Figure 2.3.5 Functional Test Cases

2.3.6 Committed Vs Completed User Stories

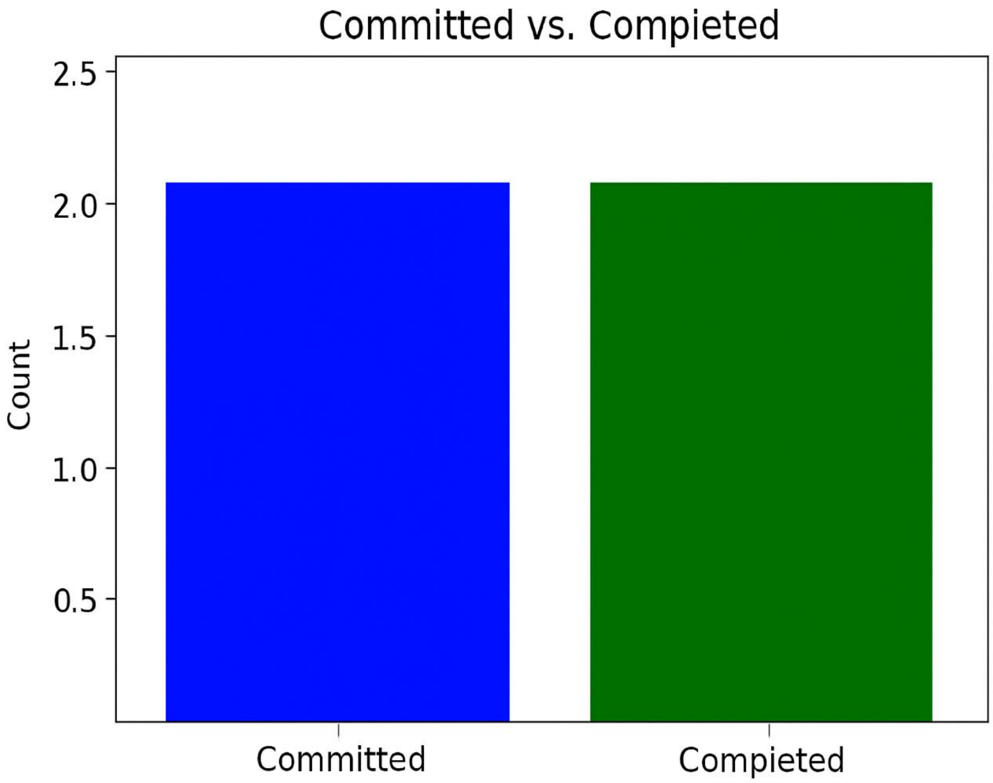
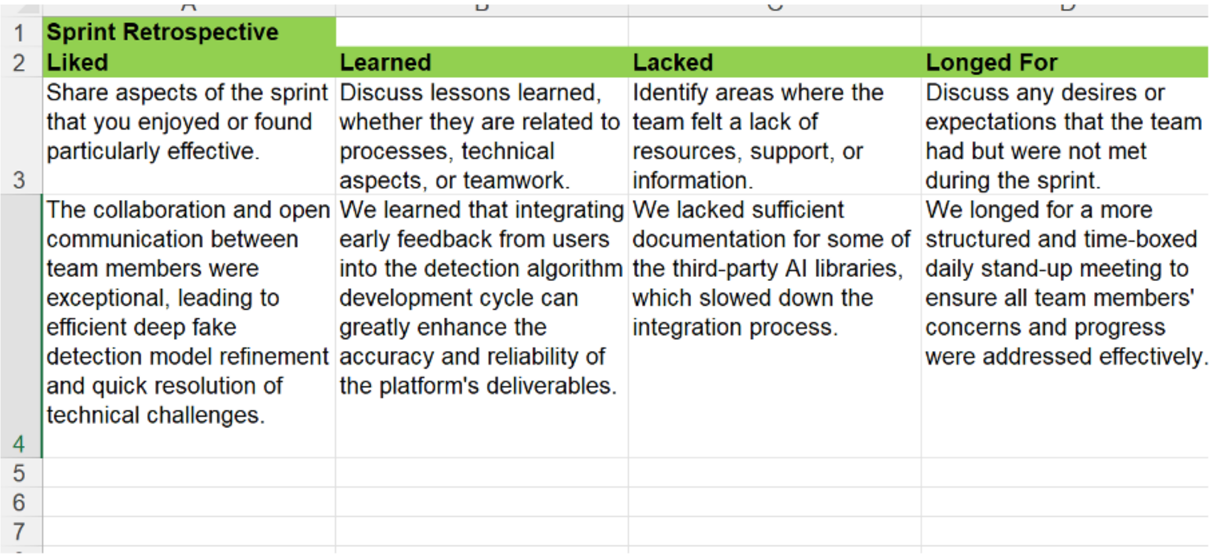


Figure 2.3.6 Committed Vs Completed User Stories

2.3.7 Sprint Retrospective



Sprint Retrospective			
Liked	**Learned**	**Lacked**	**Longed For**
Share aspects of the sprint that you enjoyed or found particularly effective.	Discuss lessons learned, whether they are related to processes, technical aspects, or teamwork.	Identify areas where the team felt a lack of resources, support, or information.	Discuss any desires or expectations that the team had but were not met during the sprint.
The collaboration and open communication between team members were exceptional, leading to efficient deep fake detection model refinement and quick resolution of technical challenges.	We learned that integrating early feedback from users into the detection algorithm development cycle can greatly enhance the accuracy and reliability of the platform's deliverables.	We lacked sufficient documentation for some of the third-party AI libraries, which slowed down the integration process.	We longed for a more structured and time-boxed daily stand-up meeting to ensure all team members' concerns and progress were addressed effectively.

Figure 2.3.7 Sprint Retrospective

CHAPTER 3

RESULTS AND DISCUSSION

3.1 Project Outcomes

Here are some of the project outcomes for ZEROhunger! app:

1. Increased Food Redistribution: Successfully facilitated the transfer of surplus food from restaurants to NGOs, helping to reduce food waste and feed those in need.
2. User Engagement: Achieved active participation from restaurants and NGOs through a seamless sign-up/login process, making it easy to donate or request food.
3. Real-Time Tracking and Transparency: Implemented a live donations and requests tracking system, ensuring transparency and building trust among users.
4. Efficient Communication: Integrated an AI chatbot to handle FAQs, common queries, and quick navigation, improving user experience and reducing manual support needs.
5. Enhanced Awareness and Outreach: Increased awareness about food waste and hunger issues through an informative 'About Us' page and easy accessibility features.
6. Platform Scalability: Built a scalable MERN stack architecture that can accommodate growing numbers of users and expand to new locations or types of food providers and recipients.
7. Social Impact Measurement: Tracked and documented metrics like the number of meals saved, food requests fulfilled, and NGOs supported, showcasing the project's tangible impact.
8. Community Building: Fostered a community spirit between food providers and NGOs, encouraging long-term collaborations and partnerships.

3.2 Committed Vs Completed User stories

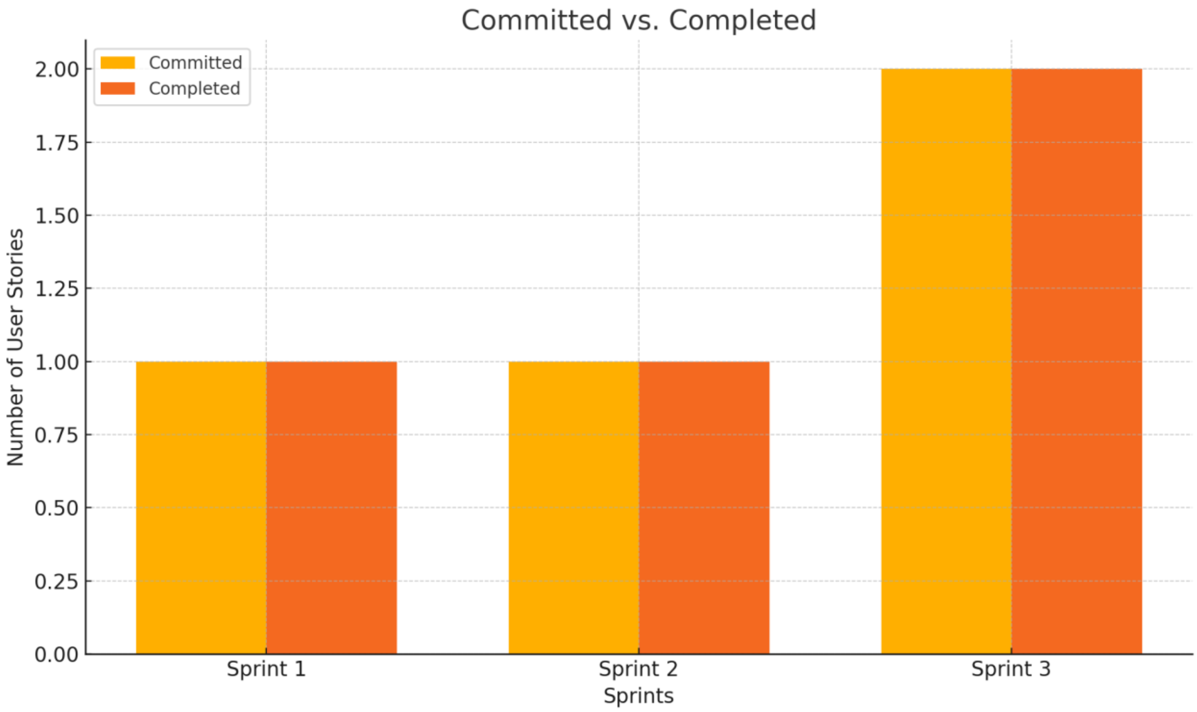


Figure 3.2 Committed Vs Completed User stories

CHAPTER 4

CONCLUSION & FUTURE ENHANCEMENTS\

ZEROhunger represents a powerful step forward in addressing the dual challenges of hunger and food waste. By leveraging technology to connect restaurants with NGOs, the platform creates a seamless and sustainable food redistribution system, making surplus food accessible to those who need it most. Through its efforts, ZEROhunger aligns with global sustainability goals and promotes social responsibility within the food industry. The project underscores the importance of collaboration and innovation in solving critical societal issues, contributing meaningfully to a world where food is never wasted, and hunger is eradicated.

Looking ahead, ZEROhunger envisions several future enhancements to amplify its impact:

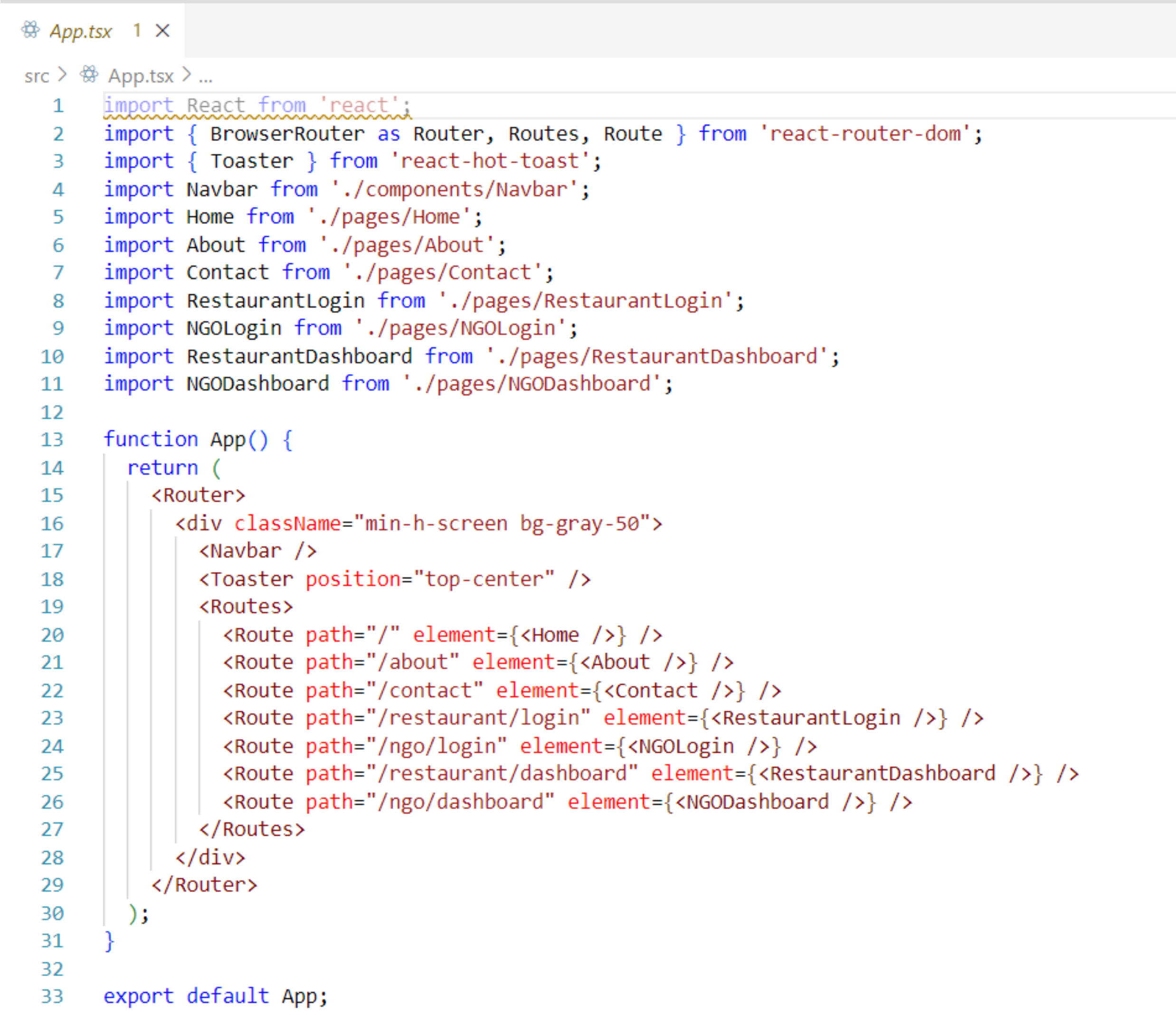
1. Expanded Partnerships: Scaling the platform to include more food donors, NGOs, and community organizations across diverse regions.
2. AI Optimization: Implementing advanced AI algorithms for more efficient matching and predictive analytics to anticipate needs and surpluses.
3. Integration with Policy Advocacy: Collaborating with governments and policymakers to incentivize food redistribution efforts and promote food safety regulations.
4. Enhanced User Experience: Adding multilingual support and accessibility features to accommodate a wider range of users.
5. Volunteer Network: Building a dedicated network of volunteers for logistics support, food delivery, and community outreach.
6. Educational Campaigns: Raising awareness about food waste and hunger through campaigns, workshops, and digital content to inspire community engagement.

With these enhancements, ZEROhunger aspires to reach greater heights, delivering innovative solutions to create a sustainable, hunger-free world.

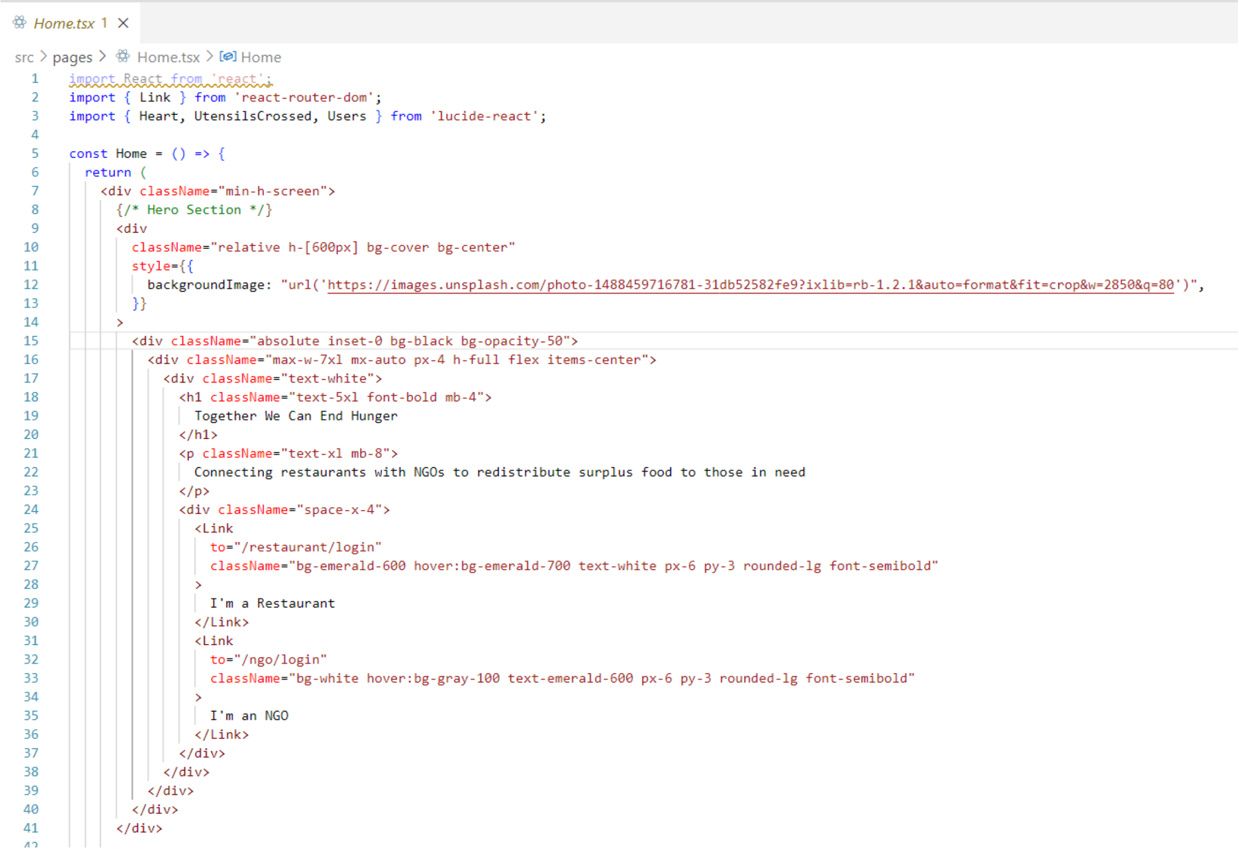
APPENDIX

A. PATENT DISCLOSURE FORM

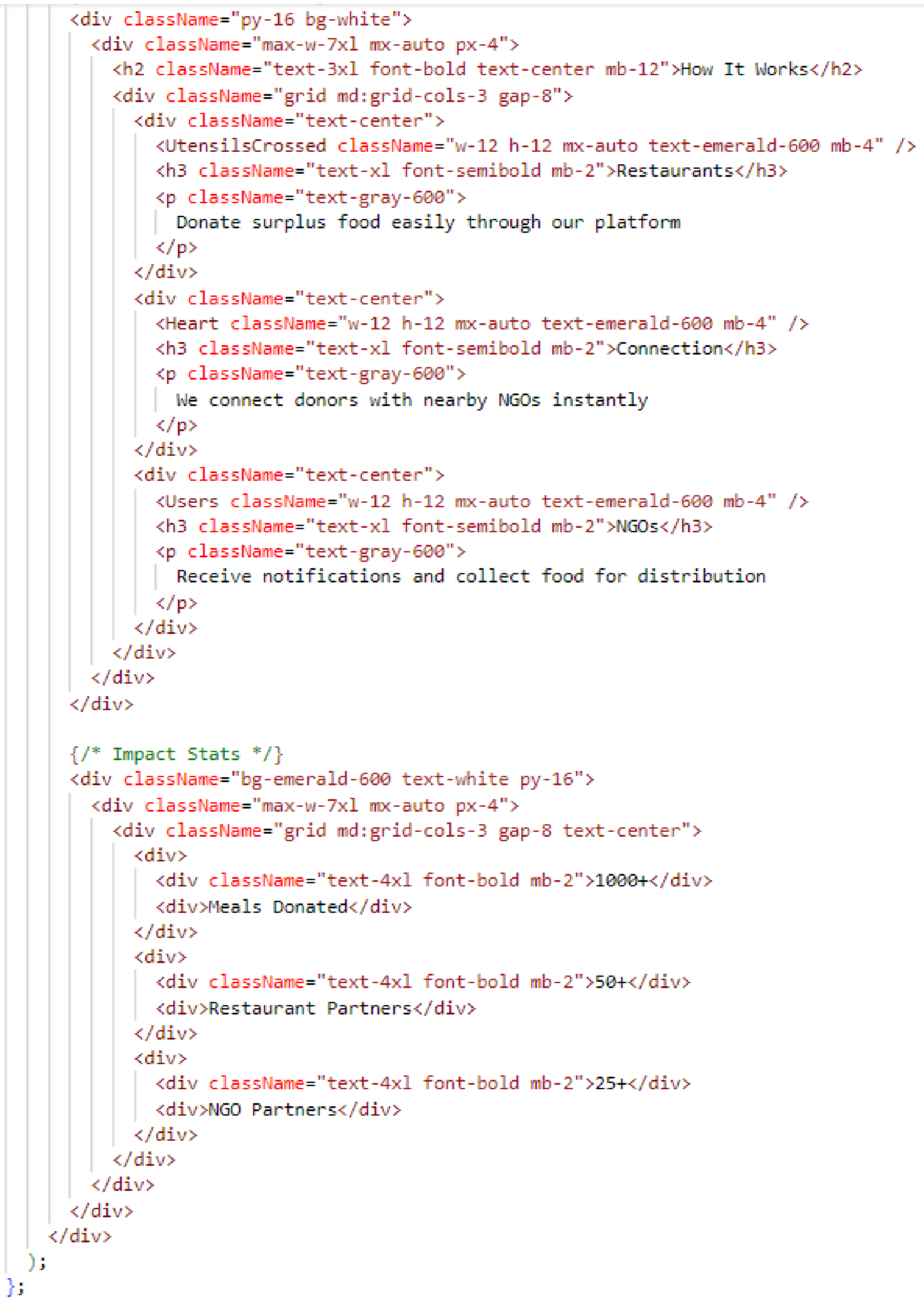
B. SAMPLE CODING



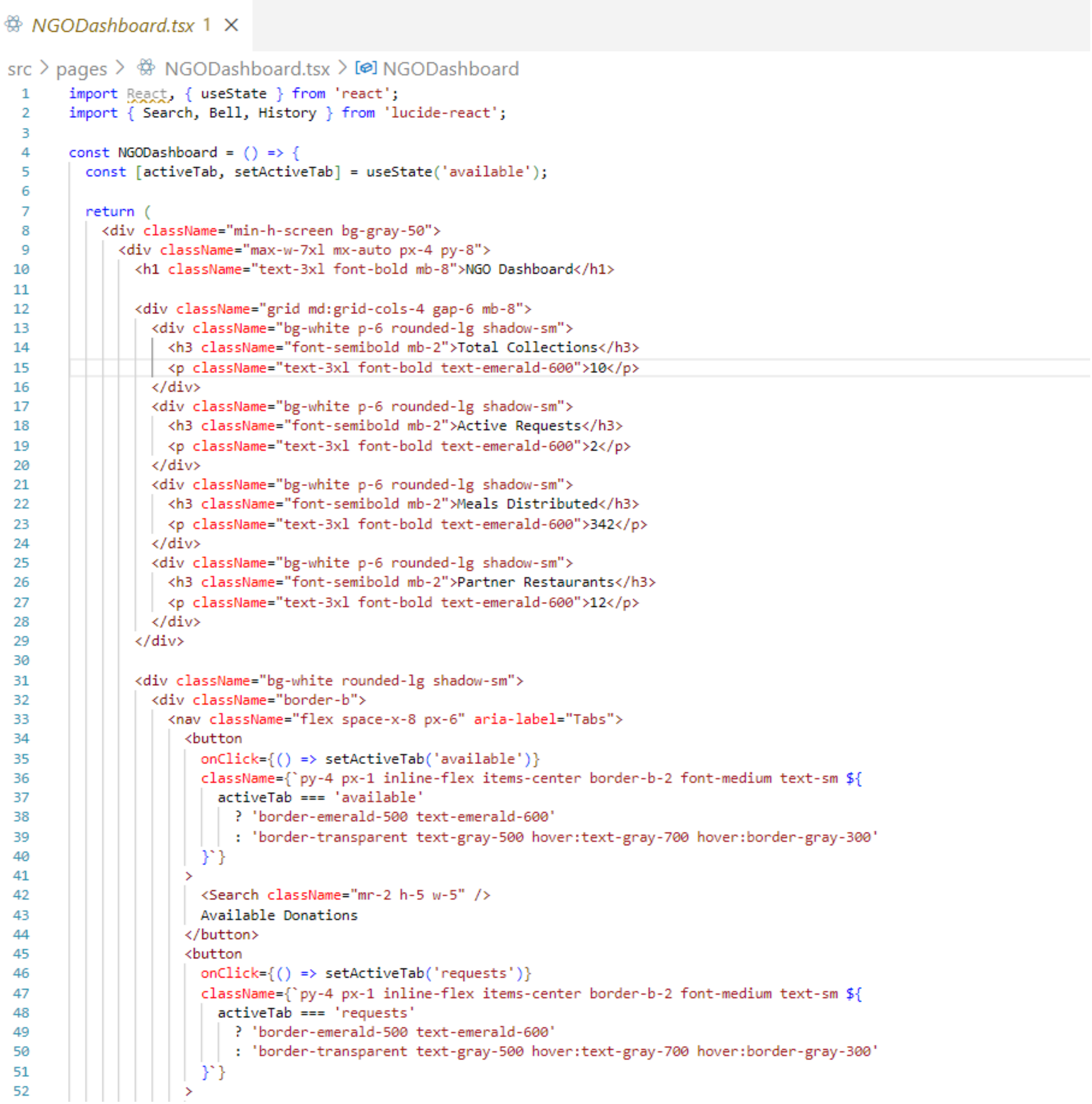
```tsx
import React from 'react';
import { BrowserRouter as Router, Routes, Route } from 'react-router-dom';
import { Toaster } from 'react-hot-toast';
import Navbar from './components/Navbar';
import Home from './pages/Home';
import About from './pages/About';
import Contact from './pages/Contact';
import RestaurantLogin from './pages/RestaurantLogin';
import NGOLogin from './pages/NGOLogin';
import RestaurantDashboard from './pages/RestaurantDashboard';
import NGODashboard from './pages/NGODashboard';

function App() {
  return (
    <Router>
      <div className="min-h-screen bg-gray-50">
        <Navbar />
        <Toaster position="top-center" />
        <Routes>
          <Route path="/" element={<Home />} />
          <Route path="/about" element={<About />} />
          <Route path="/contact" element={<Contact />} />
          <Route path="/restaurant/login" element={<RestaurantLogin />} />
          <Route path="/ngo/login" element={<NGOLogin />} />
          <Route path="/restaurant/dashboard" element={<RestaurantDashboard />} />
          <Route path="/ngo/dashboard" element={<NGODashboard />} />
        </Routes>
      </div>
    </Router>
  );
}

export default App;
```



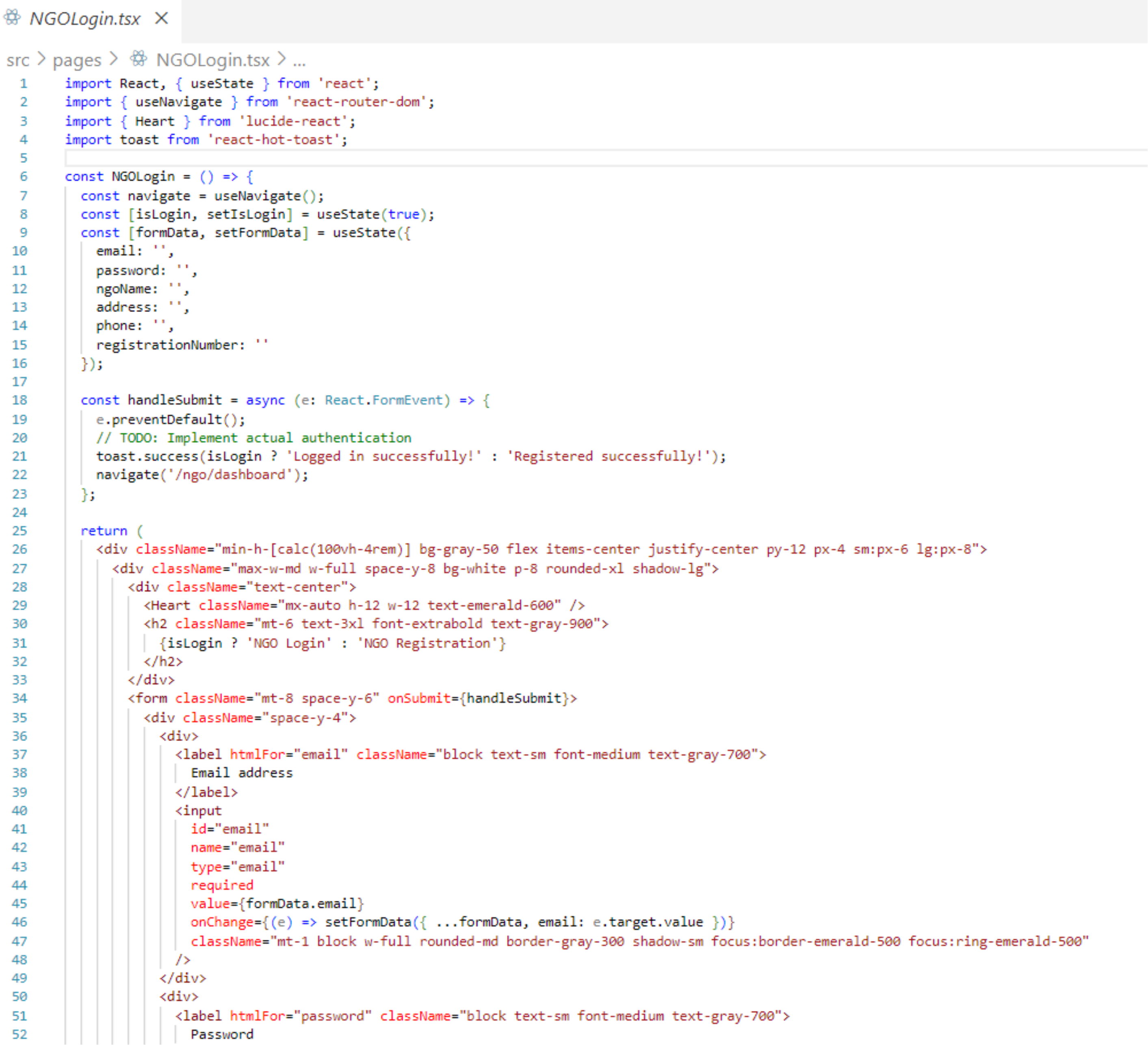
```tsx
import React from 'react';
import { Link } from 'react-router-dom';
import { Heart, UtensilsCrossed, Users } from 'lucide-react';

const Home = () => {
  return (
    <div className="min-h-screen">
      {/* Hero Section */}
      <div
        className="relative h-[600px] bg-cover bg-center"
        style={{
          backgroundImage: "url('https://images.unsplash.com/photo-1488459716781-31db52582fe9?ixlib=rb-1.2.1&auto=format&fit=crop&w=2850&q=80')",
        }}
      >
        <div className="absolute inset-0 bg-black bg-opacity-50">
          <div className="max-w-7xl mx-auto px-4 h-full flex items-center">
            <div className="text-white">
              <h1 className="text-5xl font-bold mb-4">
                Together We Can End Hunger
              </h1>
              <p className="text-xl mb-8">
                Connecting restaurants with NGOs to redistribute surplus food to those in need
              </p>
              <div className="space-x-4">
                <Link
                  to="/restaurant/login"
                  className="bg-emerald-600 hover:bg-emerald-700 text-white px-6 py-3 rounded-lg font-semibold"
                >
                  I'm a Restaurant
                </Link>
                <Link
                  to="/ngo/login"
                  className="bg-white hover:bg-gray-100 text-emerald-600 px-6 py-3 rounded-lg font-semibold"
                >
                  I'm an NGO
                </Link>
              </div>
            </div>
          </div>
        </div>
      </div>
```



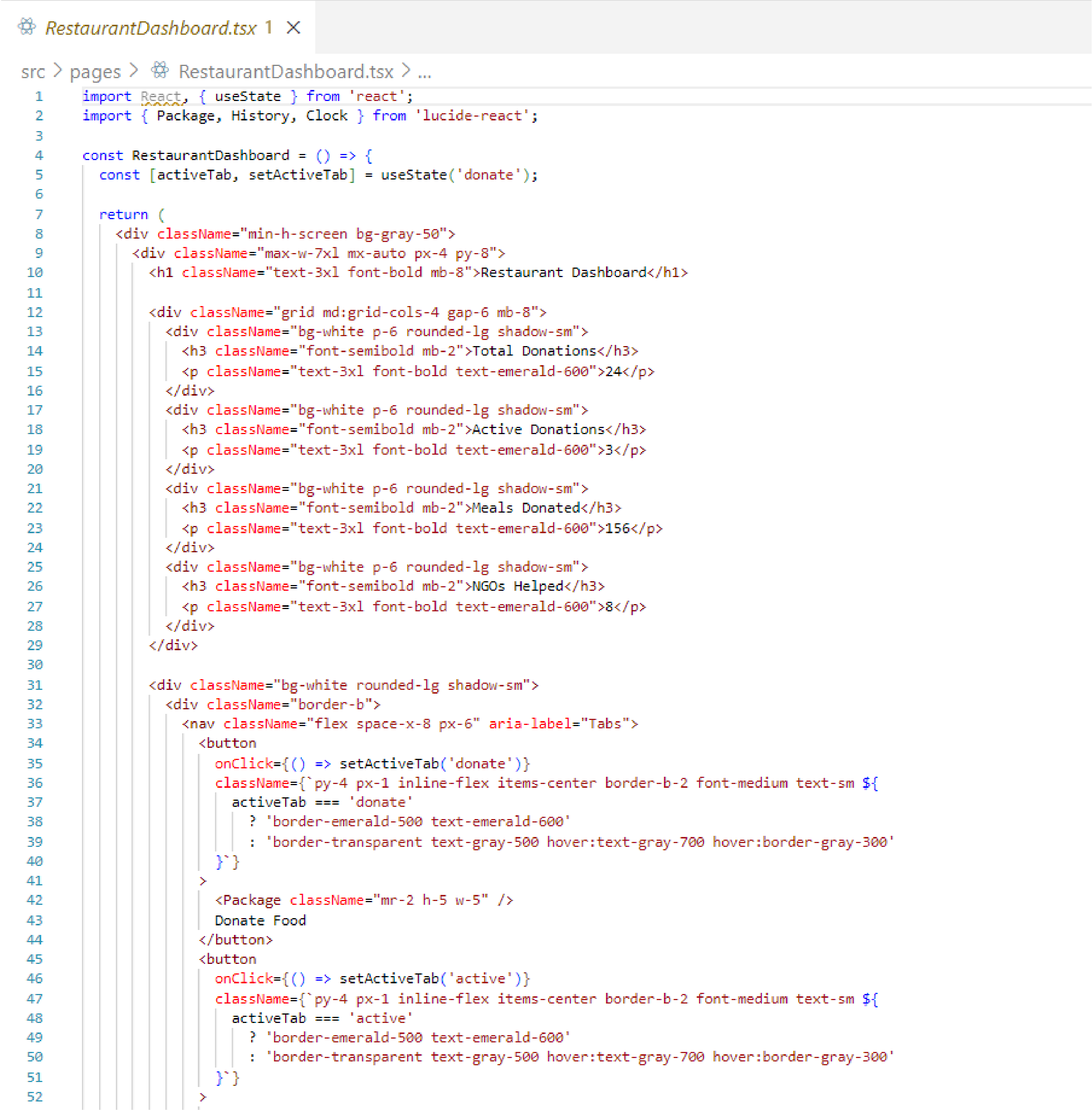
```
      <div className="py-16 bg-white">
        <div className="max-w-7xl mx-auto px-4">
          <h2 className="text-3xl font-bold text-center mb-12">How It Works</h2>
          <div className="grid md:grid-cols-3 gap-8">
            <div className="text-center">
              <UtensilsCrossed className="w-12 h-12 mx-auto text-emerald-600 mb-4" />
              <h3 className="text-xl font-semibold mb-2">Restaurants</h3>
              <p className="text-gray-600">
                Donate surplus food easily through our platform
              </p>
            </div>
            <div className="text-center">
              <Heart className="w-12 h-12 mx-auto text-emerald-600 mb-4" />
              <h3 className="text-xl font-semibold mb-2">Connection</h3>
              <p className="text-gray-600">
                We connect donors with nearby NGOs instantly
              </p>
            </div>
            <div className="text-center">
              <Users className="w-12 h-12 mx-auto text-emerald-600 mb-4" />
              <h3 className="text-xl font-semibold mb-2">NGOs</h3>
              <p className="text-gray-600">
                Receive notifications and collect food for distribution
              </p>
            </div>
          </div>
        </div>
      </div>

      {/* Impact Stats */}
      <div className="bg-emerald-600 text-white py-16">
        <div className="max-w-7xl mx-auto px-4">
          <div className="grid md:grid-cols-3 gap-8 text-center">
            <div>
              <div className="text-4xl font-bold mb-2">1000+</div>
              <div>Meals Donated</div>
            </div>
            <div>
              <div className="text-4xl font-bold mb-2">50+</div>
              <div>Restaurant Partners</div>
            </div>
            <div>
              <div className="text-4xl font-bold mb-2">25+</div>
              <div>NGO Partners</div>
            </div>
          </div>
        </div>
      </div>
    </div>
  );
};
```



```
import React, { useState } from 'react';
import { Search, Bell, History } from 'lucide-react';

const NGODashboard = () => {
  const [activeTab, setActiveTab] = useState('available');

  return (
    <div className="min-h-screen bg-gray-50">
      <div className="max-w-7xl mx-auto px-4 py-8">
        <h1 className="text-3xl font-bold mb-8">NGO Dashboard</h1>

        <div className="grid md:grid-cols-4 gap-6 mb-8">
          <div className="bg-white p-6 rounded-lg shadow-sm">
            <h3 className="font-semibold mb-2">Total Collections</h3>
            <p className="text-3xl font-bold text-emerald-600">10</p>
          </div>
          <div className="bg-white p-6 rounded-lg shadow-sm">
            <h3 className="font-semibold mb-2">Active Requests</h3>
            <p className="text-3xl font-bold text-emerald-600">2</p>
          </div>
          <div className="bg-white p-6 rounded-lg shadow-sm">
            <h3 className="font-semibold mb-2">Meals Distributed</h3>
            <p className="text-3xl font-bold text-emerald-600">342</p>
          </div>
          <div className="bg-white p-6 rounded-lg shadow-sm">
            <h3 className="font-semibold mb-2">Partner Restaurants</h3>
            <p className="text-3xl font-bold text-emerald-600">12</p>
          </div>
        </div>

        <div className="bg-white rounded-lg shadow-sm">
          <div className="border-b">
            <nav className="flex space-x-8 px-6" aria-label="Tabs">
              <button
                onClick={() => setActiveTab('available')}
                className={`py-4 px-1 inline-flex items-center border-b-2 font-medium text-sm ${
                  activeTab === 'available'
                    ? 'border-emerald-500 text-emerald-600'
                    : 'border-transparent text-gray-500 hover:text-gray-700 hover:border-gray-300'
                }`}
              >
                <Search className="mr-2 h-5 w-5" />
                Available Donations
              </button>
              <button
                onClick={() => setActiveTab('requests')}
                className={`py-4 px-1 inline-flex items-center border-b-2 font-medium text-sm ${
                  activeTab === 'requests'
                    ? 'border-emerald-500 text-emerald-600'
                    : 'border-transparent text-gray-500 hover:text-gray-700 hover:border-gray-300'
                }`}
              >
```



```tsx
import React, { useState } from 'react';
import { useNavigate } from 'react-router-dom';
import { Heart } from 'lucide-react';
import toast from 'react-hot-toast';

const NGOLogin = () => {
  const navigate = useNavigate();
  const [isLogin, setIsLogin] = useState(true);
  const [formData, setFormData] = useState({
    email: '',
    password: '',
    ngoName: '',
    address: '',
    phone: '',
    registrationNumber: ''
  });

  const handleSubmit = async (e: React.FormEvent) => {
    e.preventDefault();
    // TODO: Implement actual authentication
    toast.success(isLogin ? 'Logged in successfully!' : 'Registered successfully!');
    navigate('/ngo/dashboard');
  };

  return (
    <div className="min-h-[calc(100vh-4rem)] bg-gray-50 flex items-center justify-center py-12 px-4 sm:px-6 lg:px-8">
      <div className="max-w-md w-full space-y-8 bg-white p-8 rounded-xl shadow-lg">
        <div className="text-center">
          <Heart className="mx-auto h-12 w-12 text-emerald-600" />
          <h2 className="mt-6 text-3xl font-extrabold text-gray-900">
            {isLogin ? 'NGO Login' : 'NGO Registration'}
          </h2>
        </div>
        <form className="mt-8 space-y-6" onSubmit={handleSubmit}>
          <div className="space-y-4">
            <div>
              <label htmlFor="email" className="block text-sm font-medium text-gray-700">
                Email address
              </label>
              <input
                id="email"
                name="email"
                type="email"
                required
                value={formData.email}
                onChange={(e) => setFormData({ ...formData, email: e.target.value })}
                className="mt-1 block w-full rounded-md border-gray-300 shadow-sm focus:border-emerald-500 focus:ring-emerald-500"
              />
            </div>
            <div>
              <label htmlFor="password" className="block text-sm font-medium text-gray-700">
                Password
```


RestaurantDashboard.tsx 1 ×
src > pages > RestaurantDashboard.tsx > ...

```tsx
import React, { useState } from 'react';
import { Package, History, Clock } from 'lucide-react';

const RestaurantDashboard = () => {
  const [activeTab, setActiveTab] = useState('donate');

  return (
    <div className="min-h-screen bg-gray-50">
      <div className="max-w-7xl mx-auto px-4 py-8">
        <h1 className="text-3xl font-bold mb-8">Restaurant Dashboard</h1>

        <div className="grid md:grid-cols-4 gap-6 mb-8">
          <div className="bg-white p-6 rounded-lg shadow-sm">
            <h3 className="font-semibold mb-2">Total Donations</h3>
            <p className="text-3xl font-bold text-emerald-600">24</p>
          </div>
          <div className="bg-white p-6 rounded-lg shadow-sm">
            <h3 className="font-semibold mb-2">Active Donations</h3>
            <p className="text-3xl font-bold text-emerald-600">3</p>
          </div>
          <div className="bg-white p-6 rounded-lg shadow-sm">
            <h3 className="font-semibold mb-2">Meals Donated</h3>
            <p className="text-3xl font-bold text-emerald-600">156</p>
          </div>
          <div className="bg-white p-6 rounded-lg shadow-sm">
            <h3 className="font-semibold mb-2">NGOs Helped</h3>
            <p className="text-3xl font-bold text-emerald-600">8</p>
          </div>
        </div>

        <div className="bg-white rounded-lg shadow-sm">
          <div className="border-b">
            <nav className="flex space-x-8 px-6" aria-label="Tabs">
              <button
                onClick={() => setActiveTab('donate')}
                className={`py-4 px-1 inline-flex items-center border-b-2 font-medium text-sm ${
                  activeTab === 'donate'
                    ? 'border-emerald-500 text-emerald-600'
                    : 'border-transparent text-gray-500 hover:text-gray-700 hover:border-gray-300'
                }`}
              >
                <Package className="mr-2 h-5 w-5" />
                Donate Food
              </button>
              <button
                onClick={() => setActiveTab('active')}
                className={`py-4 px-1 inline-flex items-center border-b-2 font-medium text-sm ${
                  activeTab === 'active'
                    ? 'border-emerald-500 text-emerald-600'
                    : 'border-transparent text-gray-500 hover:text-gray-700 hover:border-gray-300'
                }`}
              >
```

C. PLAGIARISM REPORT