

---

# **Software Requirements Specification**

**for**  
**FoodHero**

**Version 2.0 approved**

**Prepared by SCSD Group 47**

**NTU**

**10/09/2024**

# Table of Contents

<b>1. Introduction</b>	<b>1</b>
1.1 Purpose	1
<b>1.2 Document Conventions</b>	<b>2</b>
<b>1.3 Intended Audience and Reading Suggestions</b>	<b>2</b>
1.4 Product Scope	3
1.5. References	3
<b>2. Overall Description</b>	<b>4</b>
2.1 Product Perspective	4
<b>2.2 Product Functions</b>	<b>5</b>
2.3 User Classes and Characteristics	7
2.4 Operating Environment	7
2.5 Design and Implementation Constraints	8
2.6 User Documentation	9
2.7 Assumptions and Dependencies	9
<b>3. External Interface Requirements</b>	<b>11</b>
3.1 User Interfaces	12
<b>3.2 Hardware Interfaces</b>	<b>39</b>
<b>3.3 Software Interfaces</b>	<b>40</b>
3.4 Communications Interfaces	41
<b>4. Functional Requirements (Lecture Format)</b>	<b>42</b>
<b>5. Functional Requirements (SRS Format)</b>	<b>49</b>
5.1 Main Menu	49
5.2 User Registration	50
5.3 Login	50
5.4 Submissions	51
5.5 Dashboard	52
5.6 Donation Match Confirmation System	54
5.7 Route Optimization	56
5.8 Notifications and Alerts	57
5.9 Chatbot Functionality	57
5.10 Interaction with Other Systems	58
<b>7. Other Requirements</b>	<b>63</b>
<b>8. Data Dictionary</b>	<b>65</b>
<b>9. System Architecture Diagram</b>	<b>66</b>
<b>10. Class Diagram</b>	<b>67</b>
<b>11. Sequence Diagrams</b>	<b>68</b>
<b>12. Use Case Diagram</b>	<b>85</b>
<b>13. Black Box Testing</b>	<b>86</b>
<b>14. White Box Testing</b>	<b>96</b>

<b>15. API Documentation</b>	<b>100</b>
<b>16. Dialog Map</b>	<b>103</b>

## Revision History

Name	Date	Reason For Changes	Version
Liu Cong	2/11/24	Changed Functional Requirements	2.0
Katyayani	4/11/24	Changed Application Descriptions	3.0
James	6/11/24	Final submission with application updates	4.0

# 1. Introduction

## 1.1 Purpose

The Food Hero app is a CRM web application designed to connect food donors, primarily supermarkets, with beneficiaries in Singapore to facilitate efficient food distribution while minimizing food waste. This Software Requirements Specification (SRS) document for version 1.0 outlines the core features and functionalities of the app, covering both backend and frontend aspects to guide development and implementation.

**Objective:** The main goal of the Food Hero app is to optimize food donation processes to ensure timely and efficient distribution of available food to beneficiaries. By streamlining interactions between donors and beneficiaries, the app contributes to reducing food waste and supporting community well-being.

### Key Functionalities:

- **Location-Based Filtering:** Incorporates public datasets of supermarkets to enhance precision in donation matching, allowing users to filter options by region for efficient food distribution.
- **Unified Login/Signup Page:** Provides a streamlined interface for both donors and beneficiaries, simplifying access and user account management.
- **Dashboard Functionality:** Offers personalized dashboards for donors and beneficiaries, empowering them to manage, track, and review current and past donations and requests.
- **Donation Application and Beneficiary Request Forms:** Includes user-friendly forms for the submission and management of food donations and requests, ensuring an intuitive experience.
- **AI-Powered Chatbot:** Integrates a chatbot powered by the Gemini API to assist users with inquiries and provide guidance on how to effectively navigate the platform.
- **Route Optimization for Deliveries:** Leverages Google Maps API to optimize delivery routes, ensuring efficient and timely distribution of food.
- **Notification System:** Delivers real-time notifications to users, keeping them informed about the status of donations and deliveries, fostering transparent communication.

**Scope of the Document:** This SRS document acts as a comprehensive roadmap for developers, testers, and other project stakeholders. It offers detailed insights into the system's functionalities, ensuring a shared understanding that aligns development efforts toward the successful implementation and deployment of version 1.0 of the Food Hero app.

## 1.2 Document Conventions

This document adheres to specific conventions and standards to maintain clarity, readability, and consistency. The following typographical conventions are employed to emphasize critical elements and facilitate easy navigation:

- **Bold Text:** Utilized for titles, headings, and key terms to draw attention to important sections and concepts.
- *Italic Text:* Applied for examples, special notes, and clarifications to distinguish them from the main content.
- **Numbered Lists:** Used to indicate the sequence of steps in processes or procedures, ensuring logical flow and ease of understanding.
- **Bullet Points:** Employed for listing general points, requirements, features, and other non-sequential items, aiding in the quick scanning of information.

These conventions help stakeholders quickly identify and comprehend different elements of the document, enhancing overall usability and ensuring that all parties can effectively engage with the content.

## 1.3 Intended Audience and Reading Suggestions

This Software Requirements Specification (SRS) document serves as a comprehensive guide for a diverse group of stakeholders involved in the Food Hero app project. Each stakeholder group will use this document in distinct ways to support the app's development and evaluation processes:

- **Developers:** To gain a thorough understanding of both the functional and non-functional requirements necessary for coding, implementation, and ensuring alignment with project goals.
- **Users:** To become acquainted with the app's functionality, user interface design, and expected user experience, providing clarity on what the app offers.
- **Testers:** To develop and execute test cases that validate the app's compliance with its specified requirements, ensuring it functions as intended and meets quality standards.
- **Documentation Writers:** To produce user manuals, FAQs, and help guides that accurately reflect the app's features, providing essential support for end-users.
- **Teaching Assistant (TA):** To evaluate the app's development lifecycle, ensuring adherence to the outlined requirements and assessing its functionality, usability, and user-centric design.

This SRS document is methodically structured to facilitate a clear understanding of the project. It begins with an overview that sets the stage for a high-level understanding, followed by detailed sections that cover the app's functional and non-functional requirements, system architecture, and

user interface specifications. This systematic approach ensures that stakeholders can easily find relevant information and apply it to their specific roles within the project.

## 1.4 Product Scope

The Food Hero app is designed to revolutionize food donation efforts in Singapore by creating a seamless, efficient, and direct channel between food donors and beneficiaries. With a focus on minimizing food waste and promoting timely distribution, the app supports community well-being and aligns with broader sustainability goals.

### Key Features and Core Functionalities:

- **Location-Based Filtering:** Utilizes public datasets and advanced mapping services to facilitate accurate matches between donors and beneficiaries, ensuring donors can quickly find nearby recipients and streamline the donation logistics.
- **Automated Location Detection:** Simplifies the process for users by auto-detecting locations, reducing manual input and enhancing user convenience.
- **Real-Time Notifications:** Keeps users informed with timely updates about donation and request statuses. This feature enhances responsiveness, allowing users to manage their activities efficiently and stay engaged throughout the process.
- **User-Friendly Interface:** Built with a focus on simplicity and accessibility, the app's interface ensures smooth interactions for users with varying levels of technical expertise.
- **Route Optimization with Mapping Technology:** Integrates route planning tools to help facilitate faster, more sustainable deliveries. This feature supports optimal travel routes, reducing delivery time and contributing to the app's eco-conscious mission.

**Broader Social and Community Objectives:** The design of Food Hero reflects its commitment to social responsibility. By promoting the efficient use of food resources, it supports food security initiatives and encourages active community participation. This app acts as a catalyst for sustainable practices, reducing food waste while fostering a culture of sharing and support.

In essence, Food Hero serves not just as a digital platform but as a community-oriented tool that leverages technology to create meaningful social impact, ensuring that surplus food reaches those who need it most in a timely and effective manner.

## 1.5. References

**Supermarket Dataset (Singapore Government, 2024)**

[https://data.gov.sg/datasets/d\\_1bf762ee1d6d7fb61192cb442fb2f5b4/view](https://data.gov.sg/datasets/d_1bf762ee1d6d7fb61192cb442fb2f5b4/view)

## 2. Overall Description

### 2.1 Product Perspective

The Food Hero app is an innovative, standalone solution designed to streamline and optimize food donation processes in Singapore. It functions independently and is not built on any pre-existing platforms, ensuring a self-contained ecosystem tailored specifically for the needs of donors, beneficiaries, and administrators. The core aim of the app is to create seamless connections between food donors and those in need, enhancing the overall food distribution network through smart features like location-based filtering, real-time notifications, and automated assistance.

**Modular Architecture:** The app's modular architecture supports easy maintenance, scalability, and the potential for future integration with additional services, ensuring adaptability and long-term growth.

#### Key Components of the Food Hero App:

##### 1. Frontend Interface:

- Built with Next.js, the platform offers a responsive and intuitive user experience. It enables seamless interactions for donors, beneficiaries, and administrators, catering to varying levels of technical proficiency.

##### 2. Backend System:

- The backend, supported by MongoDB, handles data processing and operations with efficiency and reliability. This database structure ensures scalable data storage and swift retrieval, accommodating growing user demands.

##### 3. Location Services:

- Integrates public datasets and utilizes mapping services to enable location-based filtering. This feature helps users find and connect with nearby matches for donations, enhancing the effectiveness of food distribution.

##### 4. Web-Based Notification System:

- Provides users with real-time updates embedded directly within the app interface. The notification system keeps donors, beneficiaries, and administrators informed about matches, requests, and delivery statuses, ensuring timely communication.

##### 5. Chatbot Integration:

- An AI-powered chatbot, supported by the Gemini API, is available to guide users through various app functionalities. This includes answering common inquiries, assisting with donation submissions, and helping users navigate features effectively.

**Future-Ready Design:** The modular design of Food Hero ensures that it can continuously evolve, incorporating new functionalities and integrations as needed. This adaptability aligns with the app's mission to support efficient food donation and enhance community well-being in Singapore.

## 2.2 Product Functions

The Food Hero app offers a comprehensive set of major functions designed to streamline food donation and request processes while enhancing user experience. These functions include:

### 1. User Registration and Authentication:

- Secure user registration and login processes for donors, beneficiaries, and admins, ensuring validated and authenticated access to the app.
- Enhanced security measures, including real-time validation of email and password credentials.

### 2. Donation Management:

- **Donation Application:** Donors can easily submit details of their food donations, specifying the type, quantity, location. The system validates mandatory fields before submission.
- **Manage Donations:** Donors can view and manage their active and completed donations, with real-time status updates and the ability to edit or withdraw pending donations.

### 3. Beneficiary Request Management:

- **Request Submission:** Beneficiaries can create new requests for food, including detailed specifications such as food type, quantity, delivery location, and any special instructions.
- **Manage Requests:** Beneficiaries can view their past and current requests, including statuses (e.g., new, matched, in warehouse, awaiting delivery, delivered).

### 4. Real-Time Notifications:

- Users receive timely alerts on matched donations and requests, delivery status, and updates. Notifications are seamlessly integrated into the app's interface, enhancing communication between donors, beneficiaries, and admins.

### 5. Location-Based Filtering:

- The app uses geographical data and integrates with mapping services to connect donors with nearby beneficiaries, optimizing donation logistics and increasing the likelihood of successful matches.

## 6. Route and Delivery Management:

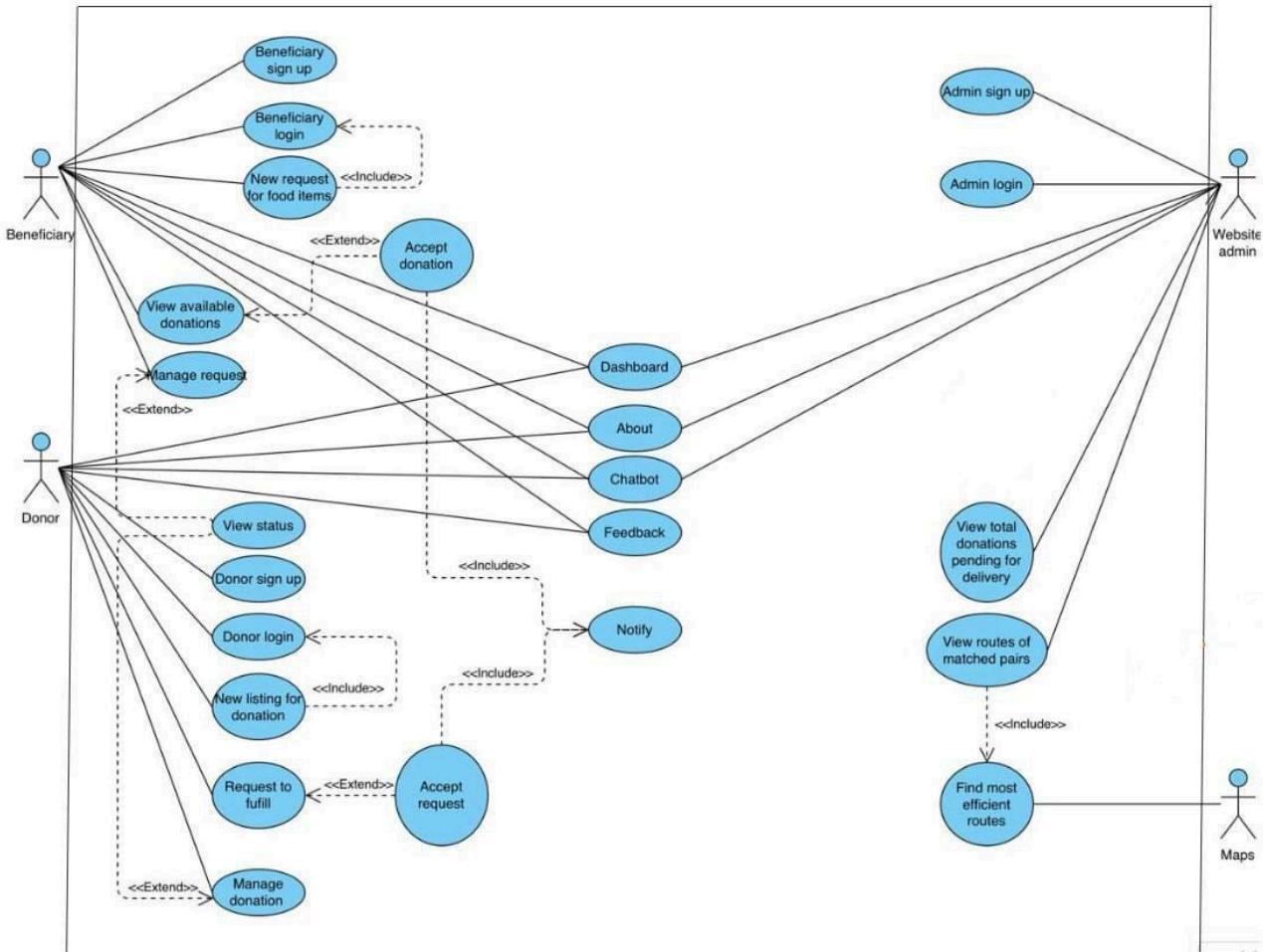
- Admins can view and manage the most efficient routes for delivery using Google Maps API integration. This includes displaying the fastest routes with traffic and distance considerations to ensure timely and efficient delivery.

## 7. AI-Powered Chatbot Assistance:

- A built-in chatbot powered by the Gemini API offers real-time assistance, guiding users through the app's features, providing information about the donation process, and helping troubleshoot common issues. Users can interact with the chatbot for quick support or escalate to human assistance if necessary.

## 8. Feedback Mechanism:

- Users can submit feedback regarding their experiences or any issues encountered. The system validates the feedback form to ensure complete information before submission, storing it for review and response by the admin team.



## 2.3 User Classes and Characteristics

The Food Hero app is designed to cater to a diverse set of users, each with unique roles and technical capabilities:

### 1. Donors:

- **Profile:** Individuals or organizations contributing food donations.
- **Technical Skills:** Expected to have basic to moderate technical proficiency.
- **Primary Functions:** Utilize the dashboard to list, manage, and monitor food donations; receive real-time notifications; and access the AI-powered chatbot for guidance.

### 2. Beneficiaries:

- **Profile:** Individuals or organizations seeking food assistance.
- **Technical Skills:** May have varying levels of technical expertise, ranging from minimal to advanced.
- **Primary Functions:** Submit and manage food requests, view available donations, receive real-time notifications on matched donations, and engage with the chatbot for support and information.

### 3. Administrators:

- **Profile:** Users with higher privilege levels responsible for overseeing and managing the system.
- **Technical Skills:** Advanced technical proficiency and familiarity with system management tools.
- **Primary Functions:** Monitor and manage food donations and requests, validate and approve matches, oversee delivery routes, maintain user data, respond to feedback, and ensure overall system functionality.

## 2.4 Operating Environment

The Food Hero app is designed to function seamlessly across a range of hardware and software environments, ensuring optimal accessibility and performance for all users:

### 1. Hardware:

- The app is compatible with various devices capable of running modern web browsers, including desktops, laptops, tablets, and smartphones.

### 2. Operating System Compatibility:

- **Desktop Platforms:** Fully compatible with major operating systems such as Windows, macOS, and Linux.

### 3. Software Dependencies:

- **Web Technologies:** Utilizes Next.js for the frontend, enabling a responsive and dynamic user interface that adapts to different screen sizes and devices.
- **Backend Infrastructure:** Powered by MongoDB to provide scalable, reliable data management and storage capabilities, essential for handling real-time data interactions.
- **Chatbot Functionality:** Integrated with the Gemini API to offer AI-powered support, guiding users through the app's features and addressing their queries.
- **Mapping and Route Services:** Employs the Google Maps API for enhanced location-based filtering and optimized route management, ensuring effective and efficient delivery logistics.
- **Notification System:** Real-time notifications are integrated directly within the app's interface, providing timely updates to users on their donation and request statuses.

This diverse and robust operating environment ensures that the Food Hero app remains reliable and efficient, catering to the needs of various stakeholders and supporting future scalability and enhancements.

## 2.5 Design and Implementation Constraints

The development of the Food Hero app must adhere to the following design and implementation constraints to ensure a secure, scalable, and high-performing solution:

### 1. Technology Stack:

- **Frontend and Backend:** Developed using Next.js to deliver a cohesive development framework that supports a responsive, dynamic user experience with server-side rendering for enhanced performance.
- **Database:** Utilizes MongoDB for scalable and efficient data storage and retrieval, supporting real-time data handling.
- **AI Chatbot:** Integrated with the Gemini API to provide interactive user assistance, guiding users through the app's features and addressing their queries effectively.
- **Mapping and Route Services:** Employs the Google Maps API for precise location-based filtering and optimized delivery route management.

### 2. Security Requirements:

- The app implements secure authentication protocols and encryption for data during transmission and at rest to protect user data and maintain trust.

- Regular security assessments and updates will be conducted to identify and mitigate vulnerabilities, ensuring adherence to best practices.

### 3. Regulatory Compliance:

- The app must comply with data protection laws, including Singapore's Personal Data Protection Act (PDPA), to safeguard user privacy.
- Food safety regulations must be observed to ensure that donated food is handled and distributed in accordance with health and safety standards.

### 4. Performance Requirements:

- The app should be capable of supporting a high volume of concurrent users without compromising on speed or reliability, ensuring a seamless user experience.
- The architecture should be designed for scalability to accommodate future growth, with the ability to manage increased user activity and data load efficiently.

These design and implementation constraints are essential for guiding the development process, ensuring that the Food Hero app meets both user expectations and technical standards for secure, efficient, and compliant operation.

## 2.6 User Documentation

The Food Hero app is designed to be user-friendly and intuitive, minimizing the need for extensive documentation. To further support users, an AI-powered chatbot is integrated into the app, assisting users in navigating its features and providing real-time support. This makes understanding and utilizing the app's capabilities simple and efficient.

All documentation will be available in digital format and accessible directly through the app's interface, ensuring users can find the assistance they need quickly and conveniently.

## 2.7 Assumptions and Dependencies

The development and operation of the Food Hero app rely on the following assumptions and dependencies:

### 1. Third-Party Services:

- **Dependencies:** The app depends on third-party services such as the Gemini API for chatbot functionality and the Google Maps API for location-based filtering and route optimization. The project assumes these services will remain available, functional, and reliable throughout the app's lifecycle.

## **2. Internet Connectivity:**

- **Assumption:** Users will have access to a stable internet connection to fully utilize the app's features. The app's performance and real-time capabilities depend on consistent connectivity.

## **3. Data Integrity:**

- **Assumption:** It is assumed that user-provided data and external datasets (e.g., public location data) are accurate and trustworthy. The app's functionality relies on the integrity and correctness of these data inputs for effective operation.

## **4. Scalability:**

- **Assumption:** The current infrastructure, built using Next.js and MongoDB, is assumed to be scalable and capable of supporting growth in user base and transaction volume without significant changes to the system architecture.

These assumptions and dependencies guide the design and development of the app, ensuring that it meets performance expectations and aligns with its operational goals.

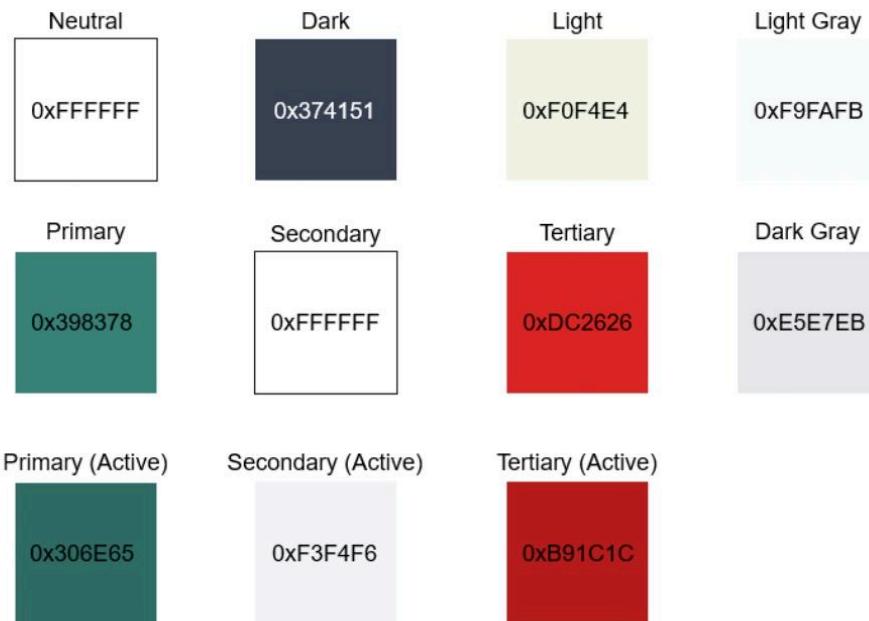
### **3. External Interface Requirements**

## 3.1 User Interfaces

### User Interfaces

#### 1 Style Guides

##### 1.1 Theme Colors



##### 1.2 Typography

Title	72px	Inter	Sans
Heading	36px	Inter	Sans
Subheading	30px	Inter	Sans
Text (Large)	20px	Inter	Sans
Text (Medium)	18px	Inter	Sans
Text (Small)	16px	Inter	Sans
Text (XS)	14px	Inter	Sans

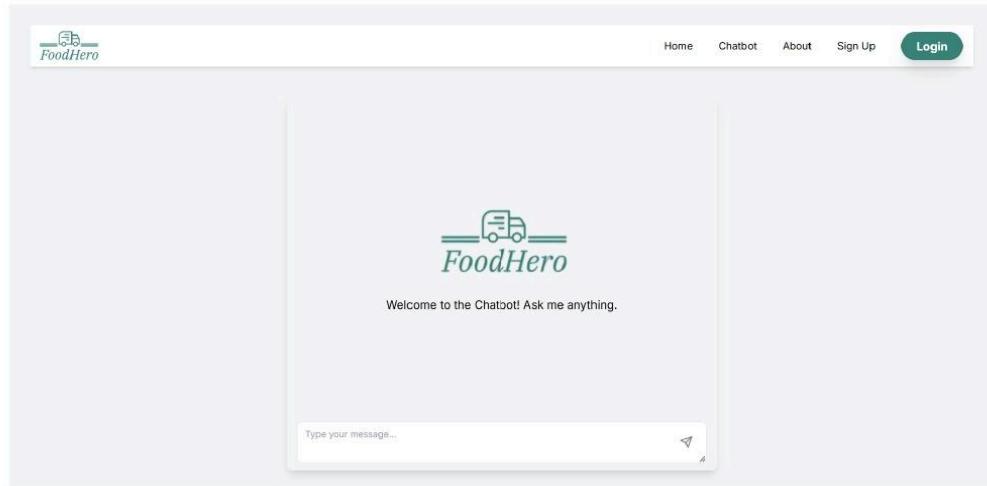
## 2 UI Mockups

### 2.1 Home Page (before login)

The screenshot displays the home page of the Food Hero website before logging in. At the top, there's a navigation bar with the Food Hero logo, Home, Chatbot, About, Sign Up, and a green Login button. The main title "Food Donation" is prominently displayed with a subtext about the mission to eliminate food wastage. Below this, there are three main sections: "Our Mission", "Our Values", and "What We Do". Each section includes a brief description and a "Learn More", "Explore Values", or "Discover More" button. The "Our Impact" section shows statistics: 5,000+ Meals Distributed, 200+ Businesses Engaged, and 100+ Charities Supported. A "Testimonials" section contains a quote from a partner organization. The footer provides links to "About Us" (Our Story, Team, Contact), "Get Involved" (Donate Food, Partner with Us), and "Address" (218 Pandan Loop, Level 6, Singapore 128408) along with operating hours.

Section	Description
<b>Our Mission</b>	We strive to eliminate food wastage by redistributing excess food to those in need.
<b>Our Values</b>	We value sustainability, compassion, and community support.
<b>What We Do</b>	We collect surplus food from businesses and distribute it to charities and families in need.
<b>Our Impact</b>	5,000+ Meals Distributed, 200+ Businesses Engaged, 100+ Charities Supported
<b>Testimonials</b>	"Food Hero has made a real impact in our community. The redistribution efforts have provided countless meals to those in need." - Partner Organization
<b>About Us</b>	Our Story, Team, Contact
<b>Get Involved</b>	Donate Food, Partner with Us
<b>Address</b>	218 Pandan Loop, Level 6, Singapore 128408 Operating Hours: Monday to Friday 9.30am - 6pm Saturday 10am - 5pm

## **2.2 Chatbot**



### 2.3 About Pages – Our Mission + Our Values + What We Do

#### 2.3.1 Our Mission Page

The screenshot shows the 'Our Mission' page of the FoodHero website. At the top, there is a navigation bar with links for Home, Chatbot, About, Sign Up, and a green 'Login' button. Below the navigation bar, the title 'Our Mission' is centered above a photograph of two people standing on a rocky outcrop, high-fiving against a backdrop of mountains and a cloudy sky. A descriptive paragraph follows, stating the mission to eradicate hunger and minimize food waste by ensuring excess food reaches those in need. It emphasizes a sustainable future where no one in Singapore goes hungry, driven by innovation, collaboration, and dedication to building a community where food is shared with dignity, care, and compassion. Two callout boxes are present: 'Empowering Communities' (describing work with local communities, volunteers, and businesses) and 'Innovative Solutions' (describing leveraging technology and strategies for food distribution). At the bottom, a large green box contains the heading 'Join Our Mission' and a message inviting users to make Singapore hunger-free, supported by individuals, businesses, or organizations. A 'Get Involved' button is located at the bottom of this box.

FoodHero

Home Chatbot About Sign Up Login

## Our Mission

Our mission is to eradicate hunger and minimize food waste by ensuring that excess food reaches those who need it most. We believe in a sustainable future where no one in Singapore goes hungry. Through innovation, collaboration, and dedication, we are committed to building a community where food is shared with dignity, care, and compassion.

**Empowering Communities**

We work closely with local communities, volunteers, and businesses to create a network that supports those in need. Our programs are designed to provide nutritious food to underprivileged families while fostering a sense of community and mutual support.

**Innovative Solutions**

By leveraging technology and innovative strategies, we aim to optimize food distribution, reduce waste, and create sustainable practices. Our goal is to make food security a reality for all through continuous improvement and creative problem-solving.

## Join Our Mission

We invite you to join us in our mission to make Singapore hunger-free. Whether you are an individual, business, or organization, your support can make a difference. Together, we can ensure that everyone has access to the food they need to thrive.

Get Involved

### 2.3.2 Our Values Page

The screenshot shows the 'Our Values' section of the FoodHero website. At the top, there is a navigation bar with links for Home, Chatbot, About, Sign Up, and Login. The main content area has a light gray background. The title 'Our Values' is centered at the top of the content area. Below the title, there are three sections: 'Sustainability', 'Compassion', and 'Community'. Each section contains a brief description and a small icon. Underneath these sections is a large green callout box titled 'How We Live Our Values' containing a bulleted list of actions. A 'Contact Us' button is located at the bottom of the callout box.

**Sustainability**  
We are committed to creating sustainable solutions that reduce food waste and promote environmental stewardship. We believe that through responsible practices, we can protect our planet for future generations.

**Compassion**  
Compassion is at the heart of everything we do. We strive to serve our community with kindness, respect, and empathy, ensuring that everyone is treated with dignity and care.

**Community**  
We believe in the power of community. By fostering strong relationships and working together, we can achieve more and create lasting change for those in need.

**How We Live Our Values**

- We operate with transparency and accountability to maintain trust and integrity.
- We prioritize partnerships that align with our values and mission.
- We empower individuals and communities to take action against hunger and food waste.

Contact Us

### 2.3.3 What We Do Page

The screenshot shows the 'What We Do' section of the FoodHero website. At the top, there is a navigation bar with links for Home, Chatbot, About, Sign Up, and Login. The main content area has a title 'What We Do' and a brief description of the organization's mission to bridge the gap between food surplus and scarcity. Below this, there are four sections: 'Food Collection', 'Distribution', 'Community Engagement', and 'Impact Measurement'. Each section contains a brief description and a small icon. At the bottom of this section, there is a large green box titled 'Our Impact' containing three statistics: '500,000+' meals distributed, '1,200+' volunteers engaged, and '300+' partners & donors. A 'Join Us' button is located at the bottom right of this box.

**What We Do**

At Singapore Food Bank, we bridge the gap between food surplus and food scarcity. Our efforts are dedicated to collecting, sorting, and redistributing surplus food to those who need it most. Through our programs, we work tirelessly to ensure that no one in our community goes hungry.

**Food Collection**

We partner with local businesses, supermarkets, and farms to collect surplus food. Our volunteers ensure that food is handled with care and delivered promptly to our distribution centers.

**Distribution**

Our distribution network spans across the island, reaching out to charities, shelters, and community centers. We focus on getting food to those who need it most, with a focus on nutritious and balanced meals.

**Community Engagement**

Beyond food distribution, we educate and engage the community on the importance of food security, sustainability, and reducing food waste. Through workshops and outreach programs, we empower others to take part in our mission.

**Impact Measurement**

We are committed to transparency and accountability. We regularly measure the impact of our programs and share our successes and challenges with the community.

**Our Impact**

**500,000+**  
Meals Distributed

**1,200+**  
Volunteers Engaged

**300+**  
Partners & Donors

[Join Us](#)

## 2.4 Sign Up Page – Donor + Beneficiary + Admin

### 2.4.1 Donor Sign Up Page

The screenshot shows the 'FoodHero' sign-up interface. At the top, there's a navigation bar with links for Home, Chatbot, About, Sign Up, and a prominent green 'Login' button. Below the navigation, three tabs are visible: 'Donor Sign Up' (which is selected and highlighted in green), 'Beneficiary Sign Up', and 'Admin Sign Up'. The main content area is titled 'Donor Sign Up' and includes a sub-instruction: 'Fill in your personal details and some information about you to get started!'. The form is divided into several sections:

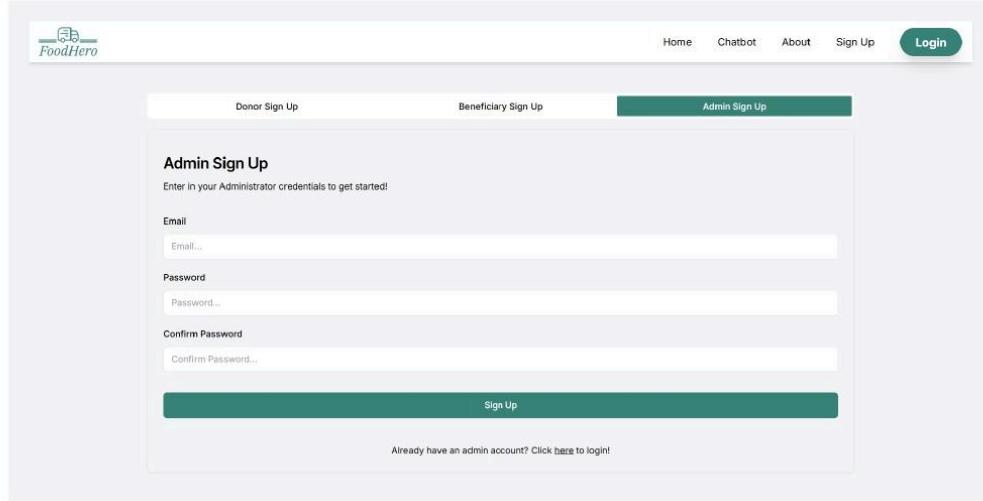
- Details**: Fields for Organisation Name (dropdown menu), Premise Address (dropdown menu), UEN Number (text input), Point of Contact Name (text input), and Point of Contact Phone Number (text input).
- Certification Status**: A note stating "Select if your organisation is Halal certified." followed by a sub-note: "If you declare that your organisation is Halal certified, random checks may be done to ensure that your organisation is Halal certified." A dropdown menu for 'Organisation's hygiene rating' is shown.
- Account Details**: Fields for Email (text input) and Password (text input).
- Confirm Password**: A field for Confirm Password (text input).

At the bottom right of the form is a large green 'Sign Up' button. Below the button, a small note reads: "Already have an account? Click [here](#) to login!"

### 2.4.2 Beneficiary Sign Up Page

The screenshot shows the 'Beneficiary Sign Up' page of the FoodHero application. At the top, there is a navigation bar with links for Home, Chatbot, About, Sign Up, and a prominent 'Login' button. Below the navigation, there are three tabs: 'Donor Sign Up', 'Beneficiary Sign Up' (which is highlighted in green), and 'Admin Sign Up'. The main content area is titled 'Beneficiary Sign Up' and contains instructions: 'Enter in your email and password to your Beneficiary account to get started!'. It is divided into two sections: 'Contact Information' and 'Account Details'. The 'Contact Information' section includes fields for 'Agency Name' (with placeholder 'Agency Name...'), 'Point of Contact Name' (placeholder 'Name...'), and 'Point of Contact Phone Number' (placeholder 'Phone Number...'). The 'Account Details' section includes fields for 'Email' (placeholder 'Email...'), 'Password' (placeholder 'Password...'), and 'Confirm Password' (placeholder 'Confirm Password...'). A large green 'Sign Up' button is located at the bottom of the form. A small note at the bottom right says 'Already have an account? Click [here](#) to login!'

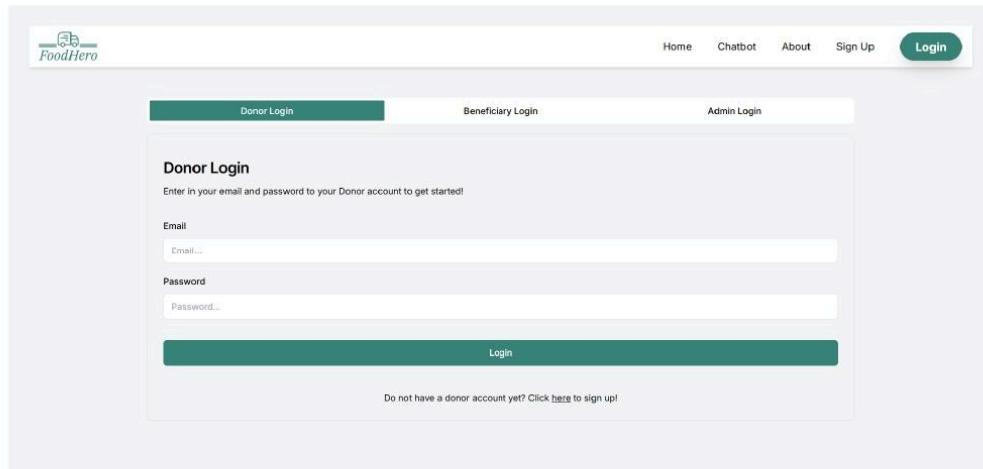
### 2.4.3 Admin Sign Up Page



The screenshot shows the 'Admin Sign Up' page of the FoodHero application. At the top, there is a navigation bar with links for Home, Chatbot, About, Sign Up, and a prominent green 'Login' button. Below the navigation bar, there are three tabs: 'Donor Sign Up', 'Beneficiary Sign Up', and the active 'Admin Sign Up' tab, which is highlighted with a dark teal background. The main content area is titled 'Admin Sign Up' and contains a sub-instruction: 'Enter in your Administrator credentials to get started!'. It features four input fields: 'Email' (placeholder 'Email...'), 'Password' (placeholder 'Password...'), 'Confirm Password' (placeholder 'Confirm Password...'), and a 'Sign Up' button at the bottom. A small note at the bottom of the form says: 'Already have an admin account? Click [here](#) to login!'

## 2.5 Login Page – Donor + Beneficiary + Admin

### 2.5.1 Donor Login Page



The screenshot shows the 'Donor Login' page of the FoodHero application. At the top, there is a navigation bar with links for Home, Chatbot, About, Sign Up, and a green 'Login' button. Below the navigation bar, there are three tabs: 'Donor Login' (which is active and highlighted in dark teal), 'Beneficiary Login', and 'Admin Login'. The main content area is titled 'Donor Login' and contains a sub-instruction: 'Enter in your email and password to your Donor account to get started!'. It features two input fields: 'Email' (placeholder 'Email...') and 'Password' (placeholder 'Password...'), and a 'Login' button at the bottom. A small note at the bottom of the form says: 'Do not have a donor account yet? Click [here](#) to sign up!'

### 2.5.2 Beneficiary Login Page

The screenshot shows the FoodHero Beneficiary Login page. At the top, there is a navigation bar with links for Home, Chatbot, About, Sign Up, and a prominent green 'Login' button. Below the navigation bar, there are three login options: 'Donor Login', 'Beneficiary Login' (which is highlighted in green), and 'Admin Login'. The 'Beneficiary Login' section contains the following fields: 'Email' (placeholder 'Email...'), 'Password' (placeholder 'Password...'), and a large green 'Login' button. Below these fields, a small note says 'Do not have a beneficiary account yet? Click [here](#) to sign up!'

### 2.5.3 Admin Login Page

The screenshot shows the FoodHero Admin Login page, which has a similar layout to the Beneficiary Login page. It features a navigation bar with Home, Chatbot, About, Sign Up, and a green 'Login' button. The 'Admin Login' section includes 'Donor Login', 'Beneficiary Login' (disabled), and 'Admin Login' (highlighted in green). It has 'Email' and 'Password' fields with their respective placeholders, and a large green 'Login' button. A note at the bottom encourages users to sign up if they don't have an admin account.

## 2.6 Feedback + Submit Pop-up

### 2.6.1 Feedback Form

The screenshot shows the 'Feedback Form' page of the FoodHero application. At the top, there is a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile, Logout, and a notification icon. The main content area has a title 'Feedback Form' and four input fields: Name, Email, Subject, and Message, each with a placeholder text. Below these fields is a large text area for the message. A green 'Submit' button is at the bottom left, and a success message 'Thank you for your feedback!' is at the bottom right.

### 2.6.2 Submit Pop-up

This screenshot is similar to the previous one but includes a modal dialog box centered over the form. The dialog box has a green header 'Feedback Submitted Successfully!' and a message 'Thank you for your feedback.' It contains a green 'Go Back to Dashboard' button. The background form fields show sample data: Name (testing), Email (testing@gmail.com), Subject (test123), and Message (test v good). The 'Submit' button and the 'Thank you for your feedback!' message are also visible.

## 2.7 Profile Page – Donor + Beneficiary + Admin

### 2.7.1 Donor Profile Page

The screenshot shows the FoodHero application interface for a donor profile. At the top, there is a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile (which is highlighted), and Logout. Below the navigation bar is a circular user icon. The main content area displays the profile information for "SHENG SIONG SUPERMARKET" (Donor). The fields shown include:

Address	UEN Number
122 ANG MO KIO AVENUE 3 #01-1753,#01-1757,#01-	1
Point of Contact Name	Point of Contact Phone
donor 1	22222222
Halal Certification	Hygiene Certification
No	B

At the bottom of the profile section are two buttons: "Edit Profile" and "Delete Profile".

### 2.7.2 Beneficiary Profile Page

The screenshot shows the FoodHero application interface for a beneficiary profile. At the top, there is a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile (which is highlighted), and Logout. Below the navigation bar is a circular user icon. The main content area displays the profile information for "Hudzaifah" (Beneficiary). The fields shown include:

Email	Point of Contact Name
beneficiary1@gmail.com	Hud
Phone	Agency
12341234	Hudzaifah
Halal Certification	Hygiene Certification
No	D
Role	Account Created At
beneficiary	10/29/2024, 4:08:06 PM

At the bottom of the profile section are two buttons: "Edit Profile" and "Delete Profile".

### 2.7.3 Admin Profile Page

The screenshot shows the 'Approved Admin Users' section of the FoodHero application. At the top, it displays a greeting message: 'Hello admin@gmail.com'. Below this, there is a table titled 'Approved Admin Users' with the following data:

No.	Email	Action
1	admin@gmail.com	Delete
2	kb@gmail.com	Delete
3	wee@gmail.com	Delete
4	admin1@gmail.com	Delete
5	admin2@gmail.com	Delete
6	asdqjiowei@gmail.com	Delete

At the bottom of the table, there is a green button labeled 'Add New Admin'.

## 2.8 Profile Functions – Edit + Delete + Add New Admin

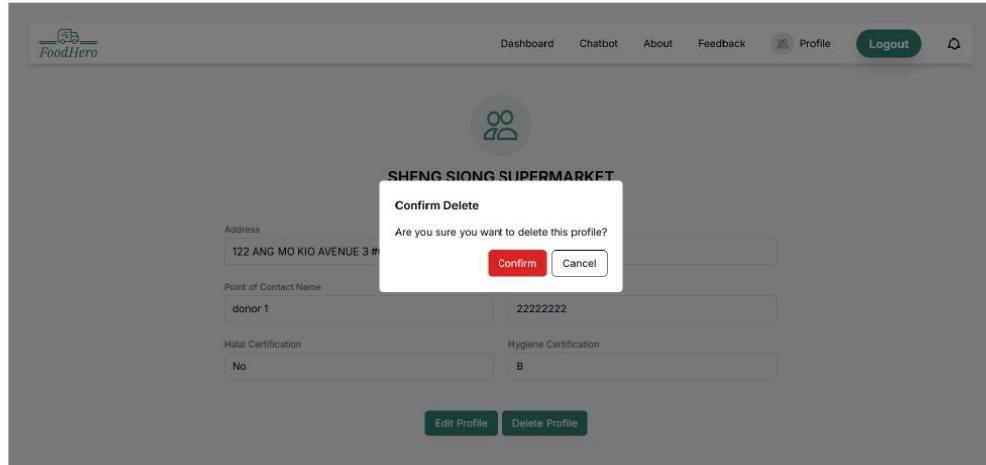
### 2.8.1 Edit Profile

The screenshot shows the 'Edit Profile' page for 'SHENG SIONG SUPERMARKET' (Donor). The page includes the following fields:

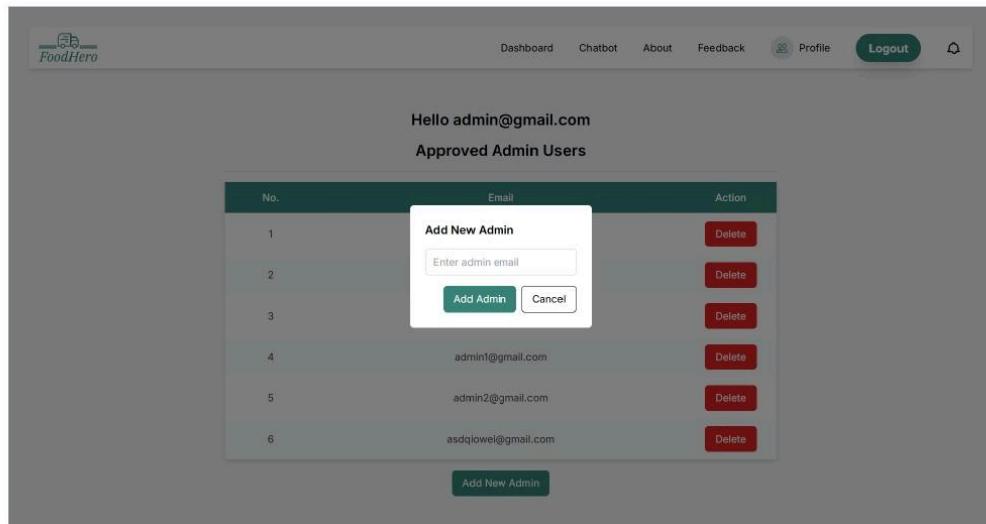
- Address:** 122 ANG MO KIO AVENUE 3 #01-1753,#01-1757,#01-
- UEN Number:** 1
- Point of Contact Name:** donor 1
- Point of Contact Phone:** 22222222
- Halal Certification:** No
- Hygiene Certification:** B

At the bottom of the form, there is a green button labeled 'Save Profile'.

### 2.8.2 Delete Profile



### 2.8.3 Add New Admin



## 2.9 Notifications

The screenshot shows a sidebar navigation with links to Dashboard, Chatbot, About, and Feedback. The Profile icon is highlighted. A Logout button is at the top right. The main content area is titled "Notifications". It lists four items:

- Pear: Request • Quantity: 10 (Status: matched)
- Potato: Request • Quantity: 10 (Status: new)
- Nasi lemak: Donation • Quantity: 610 (Status: inwarehouse) - Details: 10/29/2024, 11:36:17 PM
- Flour: Donation • Quantity: 6 (Status: new) - Details: 10/29/2024, 11:35:42 PM

Below the sidebar, there is some placeholder text: "les]" followed by the current date and location.

3, 2024 at 11:23 PM  
0 Boon Lay Pl, Block 210, Singapore 640210

## 2.10 Donor Dashboard – My Donations + Requests to Fulfill + Accept Request

### 2.10.1 My Donations Tab

The screenshot shows a header with the FoodHero logo and navigation links for Dashboard, Chatbot, About, Feedback, Profile, and Logout. A "New Donation" button is in the top right. The main area is titled "Donor Dashboard" and features a "My Donations" section. On the left, there's a welcome message for the donor: "Welcome Back Donor! Manage your donations or fulfill requests." Below this is a search bar and a filter dropdown set to "All".

The "My Donations" section displays two items:

- Carrot [Vegetables]**: Status: matched. Details: Quantity: 10, Beneficiary needs delivery by: November 2, 2024 at 11:28 PM. Please deliver to our warehouse by: November 1, 2024. Consume by: November 6, 2024 at 11:17 PM. Special Request: keep in dry place. Beneficiary name: Hud, Phone Number: 12341234. A "Mark as Delivered" button is present.
- Nasi Lemak**: Status: new. Details: Number of servings: 4, Consume by: November 5, 2024 at 11:20 PM. A "Mark as Delivered" button is present.

### 2.10.2 Requests to Fulfill Tab

The screenshot shows the FoodHero Donor Dashboard. At the top, there is a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile, Logout, and a bell icon. Below the navigation bar is a search bar labeled "Search by food name...". On the left, there is a welcome message: "Welcome Back Donor! Manage your donations or fulfill requests." In the center, there is a section titled "Requests to Fulfill" containing two items:

- Potato [Vegetables]**  
Quantity: 10  
Need by: November 8, 2024 at 11:23 PM  
**Donate**
- Fried Rice**  
Number of servings: 5  
Need by: November 3, 2024 at 11:25 PM  
Special Request: no pork  
**Donate**

### 2.10.3 Accept Request

The screenshot shows the FoodHero Donor Dashboard with the "Accept Request" dialog open. The dialog is titled "Enter Consume By Details" and contains a date input field "Consume By mm/dd/yyyy" with a calendar icon, and "Confirm" and "Cancel" buttons. The background shows the same "Requests to Fulfill" section as the previous screenshot, with the "Fried Rice" item visible.

## 2.11 Donor New Donation – Non-Cooked Food + Cooked Food

### 2.11.1 Non-Cooked Food Form

The screenshot shows the 'Donate Food' form for non-cooked food. At the top, there's a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile, Logout, and a bell icon. Below the navigation is a header with a back arrow and the text 'Donate Food'. A sub-header says 'Help us make a difference, one meal at a time. Share your food today!'. The main form area has two tabs: 'Non-Cooked Food' (selected) and 'Cooked Food'. The fields include:

- Food Name: An input field.
- Food Category: A dropdown menu labeled 'Select Food Type'.
- Quantity of Items: An input field containing '0'.
- Consume By: A date input field with the placeholder 'mm/dd/yyyy ...;... ...'.
- Food Images: A file upload field labeled 'Choose Files' with the message 'No file chosen'.
- Special Storage/Handling Requirements: A text input field.

A 'Submit' button is located at the bottom right of the form.

### 2.11.2 Cooked Food Form

The screenshot shows a web application interface for 'FoodHero'. At the top, there is a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile, Logout, and a bell icon. The main content area has a title 'Donate Food' and a subtitle 'Help us make a difference, one meal at a time. Share your food today!'. Below this, there are two tabs: 'Non-Cooked Food' (unselected) and 'Cooked Food' (selected, highlighted in green). The form fields include:

- Food Name:** An input field.
- Consume By Timing:** A date input field with a placeholder 'mm/dd/yyyy' and a calendar icon.
- Number of Servings:** An input field containing the value '0'.
- Note:** A small note below the serving input: 'This does not have to be accurate. An estimate will do!'
- Food Images:** A file input field labeled 'Choose Files' with a placeholder 'No file chosen'.
- Submit:** A green 'Submit' button at the bottom right of the form.

## 2.12 Beneficiary Dashboard – My Requests + Available Donations + Accept Donation

### 2.12.1 My Requests Tab

The screenshot shows the FoodHero Beneficiary Dashboard. At the top, there is a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile, Logout, and a bell icon. Below the navigation bar, there is a search bar with placeholder text "Search by food name..." and a dropdown menu for "Filter by status: All". On the left, there is a sidebar with tabs for "My Requests" (which is selected) and "Available Donations". A large image on the left side of the dashboard features two people and the text "Welcome Back Beneficiary! Manage your requests or view available donations.".

**My Requests**

- Potato [Vegetables]**
  - # Quantity: 10
  - Need by: November 8, 2024 at 11:23 PM
  - 📍 Delivery Location: 210 Boon Lay Pl, Block 210, Singapore 640210
  - 📍 Unit Number: 6-120

[Edit Details](#) [Withdraw](#) new
- Fried Rice**
  - # Number of servings: 5
  - Need by: November 3, 2024 at 11:25 PM
  - ☆ Special Request: no pork
  - 📍 Delivery Location: Blk 832 Woodlands Street 83, #01-65 S, Singapore 730832

new

### 2.12.2 Available Donations Tab

The screenshot shows the FoodHero Beneficiary Dashboard. At the top, there is a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile, Logout, and a notification bell icon. Below the navigation bar, a search bar says "Search by food name...". A "New Request" button is located in the top right corner. On the left, there is a sidebar with a "Welcome Back Beneficiary!" message and a "Manage your requests or view available donations." link. The main content area is titled "Available Donations" and lists two items:

- Nasi Lemak**  
Number of servings: 4  
Consume By: November 5, 2024 at 11:20 PM  
**Accept** button
- Chicken Soup [Canned Food]**  
Quantity: 15  
Consume By: December 10, 2024 at 11:21 PM  
**Accept** button

### 2.12.3 Accept Donation

The screenshot shows the FoodHero Beneficiary Dashboard with a modal window open for accepting a donation. The modal is titled "Enter Delivery Details" and contains fields for "Delivery Location" (with placeholder "Enter a location") and "Floor Number" (with placeholder "If applicable"). It also includes a "Delivery Date" field set to "11/dd/2024" and a date picker icon. At the bottom of the modal are "Confirm" and "Cancel" buttons. The background shows the same "Available Donations" list as the previous screenshot.

## 2.13 Beneficiary New Request – Non-Cooked Food + Cooked Food

### 2.13.1 Non-Cooked Food

The screenshot shows a web application interface for food requests. At the top, there is a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile, and Logout. Below the navigation bar is a title 'Request for Food' with a subtitle 'Complete the form below to request food donations from our generous donors.' A 'Non-Cooked Food' tab is selected, indicated by a green background. The form fields include:

- Food Name: An input field.
- Food Category: A dropdown menu labeled 'Select Food Category'.
- Quantity of Items: An input field containing the value '0'.
- Special Requests (e.g. dietary restrictions): An input field labeled 'If applicable'.
- Request Food By: A date input field with the placeholder 'mm/dd/yyyy'.
- Delivery Location: An input field labeled 'Enter a location'.
- Floor Number: An input field labeled 'If applicable'.

A 'Submit' button is located at the bottom right of the form area.

### 2.13.2 Cooked Food

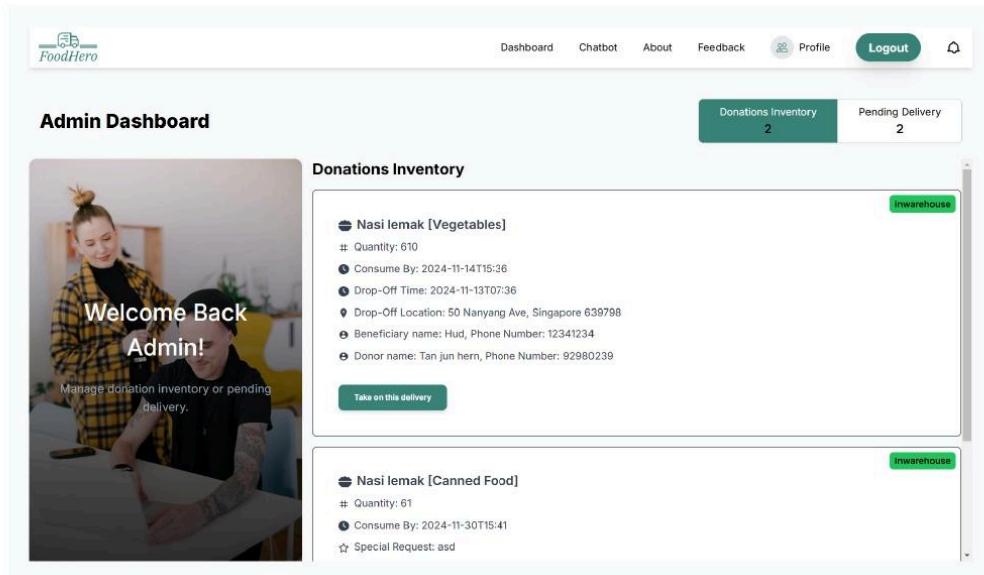
The screenshot shows a web application interface for 'FoodHero'. At the top, there's a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile, Logout, and a notification icon. Below the navigation is a title 'Request for Food' with a back button. A note says 'Complete the form below to request food donations from our generous donors.' There are two tabs: 'Non-Cooked Food' (disabled) and 'Cooked Food' (selected). The form fields include:

- Food Name: Input field.
- Number of Servings: Input field with value '0'.
- Special Requests (e.g. dietary restrictions): Input field with placeholder 'If applicable'.
- Request Food By: Input field for date (mm/dd/yyyy) with a calendar icon.
- Delivery Location: Input field with placeholder 'Enter a location'.
- Floor Number: Input field with placeholder 'If applicable'.

A green 'Submit' button is located at the bottom right of the form area.

## 2.14 Admin Dashboard – Donations Inventory + Pending Delivery + Display Map

### 2.14.1 Donations Inventory

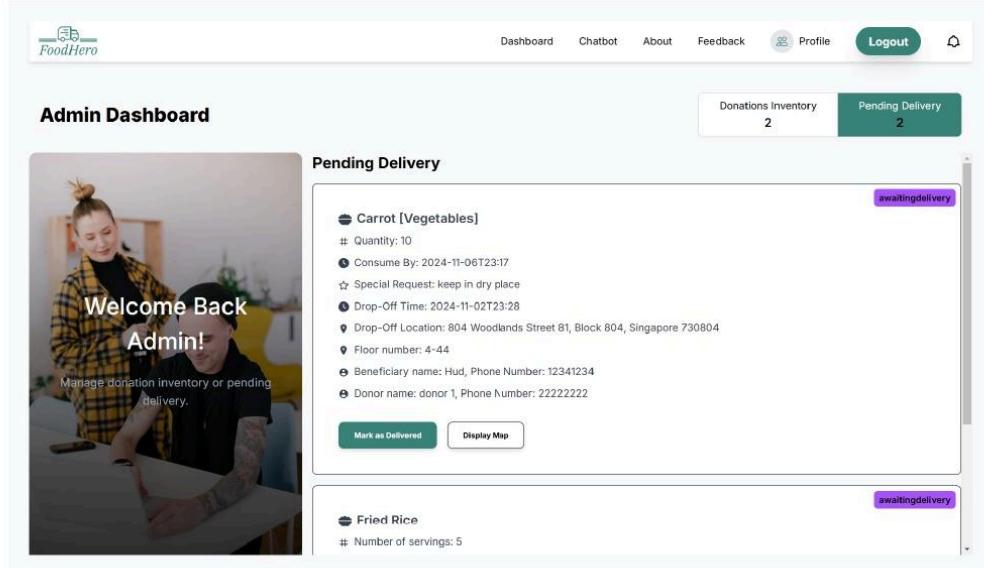


The screenshot shows the FoodHero Admin Dashboard. At the top, there's a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile, and Logout. Below the navigation is a header with the FoodHero logo and two buttons: "Donations Inventory" (2) and "Pending Delivery" (2). The main area is titled "Admin Dashboard" and features a "Welcome Back Admin!" message with a photo of two people. Below this, there are two sections for "Donations Inventory".

- Nasi lemak [Vegetables]**
  - # Quantity: 610
  - Consume By: 2024-11-14T15:36
  - Drop-Off Time: 2024-11-13T07:36
  - 📍 Drop-Off Location: 50 Nanyang Ave, Singapore 639798
  - ✉ Beneficiary name: Hud, Phone Number: 12341234
  - ✉ Donor name: Tia jun hern, Phone Number: 92980239

[Take on this delivery](#)
- Nasi lemak [Canned Food]**
  - # Quantity: 61
  - Consume By: 2024-11-30T15:41
  - ☆ Special Request: asd

### 2.14.2 Pending Delivery



The screenshot shows the FoodHero Admin Dashboard. At the top, there's a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile, and Logout. Below the navigation is a header with the FoodHero logo and two buttons: "Donations Inventory" (2) and "Pending Delivery" (2). The main area is titled "Admin Dashboard" and features a "Welcome Back Admin!" message with a photo of two people. Below this, there are two sections for "Pending Delivery".

- Carrot [Vegetables]**
  - # Quantity: 10
  - Consume By: 2024-11-06T23:17
  - ☆ Special Request: keep in dry place
  - Drop-Off Time: 2024-11-02T23:28
  - 📍 Drop-Off Location: 804 Woodlands Street 81, Block 804, Singapore 730804
  - 📍 Floor number: 4-44
  - ✉ Beneficiary name: Hud, Phone Number: 12341234
  - ✉ Donor name: donor 1, Phone Number: 22222222

[Mark as Delivered](#) [Display Map](#)
- Fried Rice**
  - # Number of servings: 5

### 2.14.3 Display Map

The screenshot shows the FoodHero Admin Dashboard. At the top, there is a navigation bar with links for Dashboard, Chatbot, About, Feedback, Profile, and Logout. Below the navigation bar, there are two green buttons: 'Donations Inventory' (value: 2) and 'Pending Delivery' (value: 2). The main area is titled 'Admin Dashboard' and features a large image of two people working at a counter with the text 'Welcome Back Admin!' overlaid. Below the image, there is a message: 'Manage donation inventory or pending delivery.' To the right of the image is a map titled 'Map Route with ETA'. The map shows a route from Johor Bahru to Singapore, with point A marked in red and point B marked in blue. The route is labeled 'Distance: 21.1 km Estimated Time: 27 mins'. There are also buttons for 'Mark as Delivered' and 'Display Map'.

## 3 Information Card

### 3.1 Donor Cards with Status

#### 3.1.1 New

The screenshot shows a donor card for 'Nasi Lemak'. The card has a 'new' status indicator in the top right corner. The card details are as follows:

- Item:** Nasi Lemak
- Number of servings:** 4
- Consume by:** November 5, 2024 at 11:20 PM

At the bottom of the card are two buttons: 'Edit Details' and 'Withdraw'.

### 3.1.2 Matched

● Carrot [Vegetables]

# Quantity: 10

● Beneficiary needs the delivery by: November 2, 2024 at 11:28 PM. **Please deliver to our warehouse by: November 1, 2024.**

● Consume by: November 6, 2024 at 11:17 PM

☆ Special Request: keep in dry place

● Beneficiary name: Hud, Phone Number: 12341234

**Mark as Delivered**

### 3.1.3 In Warehouse

● Carrot [Vegetables]

# Quantity: 10

● Beneficiary needs the delivery by: November 2, 2024 at 11:28 PM. **Please deliver to our warehouse by: November 1, 2024.**

● Consume by: November 6, 2024 at 11:17 PM

☆ Special Request: keep in dry place

● Beneficiary name: Hud, Phone Number: 12341234

### 3.1.4 Awaiting Delivery

● Carrot [Vegetables]

# Quantity: 10

● Beneficiary needs the delivery by: November 2, 2024 at 11:28 PM. **Please deliver to our warehouse by: November 1, 2024.**

● Consume by: November 6, 2024 at 11:17 PM

☆ Special Request: keep in dry place

● Beneficiary name: Hud, Phone Number: 12341234

### 3.1.5 Delivered

Carrot [Vegetables] delivered

# Quantity: 10

● Beneficiary needs the delivery by: November 2, 2024 at 11:28 PM. **Please deliver to our warehouse by: November 1, 2024.**

● Consume by: November 6, 2024 at 11:17 PM

☆ Special Request: keep in dry place

● Beneficiary name: Hud, Phone Number: 12341234

## 3.2 Beneficiary Cards with Status

### 3.2.1 New

Potato [Vegetables] new

# Quantity: 10

● Need by: November 8, 2024 at 11:23 PM

📍 Delivery Location: 210 Boon Lay Pl, Block 210, Singapore 640210

📍 Unit Number: 6-120

Edit Details Withdraw

### 3.2.2 Matched

Carrot [Vegetables] matched

# Quantity: 10

● Need By: November 2, 2024 at 11:28 PM

● Consume By: November 6, 2024 at 11:17 PM

☆ Special Request: keep in dry place

📍 Delivery Location: 804 Woodlands Street 81, Block 804, Singapore 730804

📍 Unit Number: 4-44

● Donor name: donor 1, Phone Number: 22222222

### 3.2.3 In Warehouse

 **Carrot [Vegetables]** inwarehouse

# Quantity: 10

⌚ Need By: November 2, 2024 at 11:28 PM

⌚ Consume By: November 6, 2024 at 11:17 PM

☆ Special Request: keep in dry place

📍 Delivery Location: 804 Woodlands Street 81, Block 804, Singapore 730804

📍 Unit Number: 4-44

👤 Donor name: donor 1, Phone Number: 22222222

### 3.2.4 Awaiting Delivery

 **Carrot [Vegetables]** awaitingdelivery

# Quantity: 10

⌚ Need By: November 2, 2024 at 11:28 PM

⌚ Consume By: November 6, 2024 at 11:17 PM

☆ Special Request: keep in dry place

📍 Delivery Location: 804 Woodlands Street 81, Block 804, Singapore 730804

📍 Unit Number: 4-44

👤 Donor name: donor 1, Phone Number: 22222222

### 3.2.5 Delivered

 **Carrot [Vegetables]** delivered

# Quantity: 10

⌚ Need By: November 2, 2024 at 11:28 PM

⌚ Consume By: November 6, 2024 at 11:17 PM

☆ Special Request: keep in dry place

📍 Delivery Location: 804 Woodlands Street 81, Block 804, Singapore 730804

📍 Unit Number: 4-44

👤 Donor name: donor 1, Phone Number: 22222222

## 4 Common Components

### 4.1 Misc



Navigation Bar (before login)



Navigation Bar (after login)

## 3.2 Hardware Interfaces

The Food Hero app is designed to interface with various hardware components using standard web technologies to ensure accessibility and compatibility across a wide range of devices. The logical and physical characteristics of these interfaces are as follows:

### 1. Supported Devices:

- **Device-Agnostic Design:** The app functions seamlessly across multiple types of devices, including desktops, laptops, tablets, and smartphones. This flexibility allows users to access the app on their preferred device without limitations.
- **Input Methods:** Supports touch inputs on mobile and tablet devices for ease of use and standard keyboard/mouse interactions on desktops and laptops, accommodating different user preferences.

### 2. Data and Control Interactions:

- **Browser-Based Interaction:** The app handles user interactions through a web browser, including touch, keyboard, and mouse events. The app is optimized for low-latency response times, ensuring a smooth and responsive user experience.
- **Device Input Handling:** Designed to manage a variety of input types effectively, allowing for fluid navigation and functionality across touch and non-touch interfaces.

### 3. Communication Protocols:

- **Secure Data Transmission:** Utilizes HTTP/HTTPS protocols for communication between client devices and the server, ensuring that data transmitted is encrypted and secure across all devices. This protocol selection supports high security and reliability in data exchange.

These hardware interface characteristics ensure that the Food Hero app provides a consistent, secure, and efficient user experience across different devices and input methods.

### 3.3 Software Interfaces

The Food Hero app integrates with various software components to provide its full suite of functionalities. The key software interfaces include:

#### 1. Database:

- **MongoDB:** The app utilizes MongoDB for data storage and retrieval, with custom APIs designed to interact with the database for efficient handling of user information, donation records, and beneficiary requests. This ensures scalable and reliable data management.

#### 2. Operating Systems:

- **Compatibility:** The app is compatible with all major operating systems, including Windows, macOS, Linux, Android, and iOS. It operates within standard web browsers on these platforms, ensuring broad accessibility across desktop and mobile devices.

#### 3. Third-Party Services:

- **Gemini API:** Integrated for chatbot functionality, providing users with real-time support and assistance. Communication between the app and the Gemini API occurs via RESTful API calls, ensuring efficient and responsive interaction.
- **Google Maps API:** Used for location-based filtering and route optimization to facilitate the matching process between donors and beneficiaries, as well as delivery planning.

#### 4. Data Flow:

- **Frontend and Backend Communication:** Data flows between the frontend (developed using Next.js) and the backend (Node.js integrated with MongoDB) through RESTful API calls. The backend processes user inputs, interacts with the database, and communicates with external services such as the Gemini API and Google Maps API.
- **Real-Time Updates:** Notifications and real-time updates are integrated directly within the app's interface, ensuring continuous communication between users and the app.

These software interfaces ensure that the Food Hero app provides seamless, secure, and scalable interactions, enabling users to experience smooth data handling and feature integration across various platforms.

### 3.4 Communications Interfaces

The Food Hero app employs several communication interfaces to support seamless and secure interactions:

#### 1. Network Protocols:

- **HTTP/HTTPS:** The app uses HTTP/HTTPS for all network communications, ensuring secure data transfer between client devices and the server. All data transmissions are encrypted to protect user information and maintain data integrity.

#### 2. In-App Notifications:

- **Real-Time Updates:** Notifications are integrated within the app's interface, providing users with timely updates about donations, matches, and system changes. This ensures continuous engagement and effective communication between the app and its users without external services.

#### 3. Synchronization Mechanisms:

- **Real-Time Data Sync:** The app features real-time synchronization between the frontend (Next.js) and backend (Node.js with MongoDB), ensuring users have the most current information. This is particularly crucial for updates related to donation matches and status notifications.

#### 4. Security:

- **Encryption Protocols:** All communications are secured with standard encryption protocols to safeguard sensitive user data, including personal information and donation details.

These communication interfaces support reliable, secure, and real-time interactions, providing users with an efficient and safe experience while using the Food Hero app.

## 4. Functional Requirements (Lecture Format)

### 4.1 Main Menu

- 4.1.1. The system must allow the user to register their account.
- 4.1.2. The system must allow the user to log in to their account.
- 4.1.3. The system must allow the donor to send food donation details.
- 4.1.4. The system must allow the beneficiary to submit food requests.
- 4.1.5. The system must include a chatbot to assist users.

### 4.2 User Registration

- 4.2.1. The system must allow the user to register a new account.
- 4.2.2. The system must allow users to input specific registration requirements for each type of user, such as admin, beneficiary, and donor.
- 4.2.3. The system must validate that all required fields are filled out before allowing account creation.
- 4.2.4. The system must check the availability of the email address before completing the registration.
- 4.2.5. The system must display appropriate error messages to guide the user for correction if any required fields are missing or incorrectly filled.
- 4.2.6. The system must allow the user to specify their role (donor, beneficiary, admin) during registration.
- 4.2.7. The system must allow the approved admin user to sign up,

### 4.3. Login

- 4.3.1. The system must allow the user to log in using their registered email and password.
- 4.3.2. The system must validate that the email address and password are correct.
- 4.3.3. The system must display an error message if the login information is incorrect.

4.3.4. The system must redirect the user to their respective dashboard upon successful login.

## 4.4 Submissions

### 4.4.1. Food Donation Submission

4.4.1.1. The system must allow donors to submit food donation details.

4.4.1.2. The system must require the following details in the food donation form:

#### 4.4.1.2.1 Non-Cooked Food

- Food Name
- Food Category
- Quantity of Items
- Consume By
- Special Storage/Handling Requirements

#### 4.4.1.2.2 Cooked Food

- Food Name
- Consume By Timing
- Number of Servings

4.4.1.3. The system must validate that all fields in the food donation form are filled out before submission.

### 4.4.2. Food Request Submission

4.4.2.1. The system must allow beneficiaries to submit food requests.

4.4.2.2. The system must require the following details in the food request form:

#### 4.4.2.2.1 Non-Cooked Food

- Food Name
- Food Category
- Quantity of items
- Special Requests (If applicable)
- Request food by

- Delivery Location
- Floor Number (if applicable)

#### 4.2.2.2 Cooked Food

- Food Name
- Number of Servings
- Special Requests (if applicable)
- Request Food By
- Delivery Location
- Floor Number (if applicable)

4.4.3. The system must validate that all fields in the food request form are filled out before submission.

### 4.5. Dashboard

#### 4.5.1. Donor Dashboard

4.5.1.1. The donor must be able to view the past and current donation listings and display its details. They are able to do so by searching by food listing names or filter by status.

4.5.1.2. The donor must be able to browse through requests from beneficiaries to find new donation opportunities. They are able to do so by searching by food listing names.

4.5.1.3. The donor must have the option to donate to the beneficiary if the request falls within their means. Upon donating, the donor should key in the consume by details.

4.5.1.4. The donor must be able to put up new listings for donations that are open for beneficiaries to accept.

4.5.1.5. The donor must be able to edit their existing listings if they are not matched yet.

4.5.1.5. The donor must be able to withdraw their existing listings if they are not matched yet.

4.5.1.6. The donor must be able to submit feedback at any point of time.

4.5.1.7. The donor must be able to submit feedback after a donation is successfully delivered or received by the beneficiary.

#### 4.5.2. Beneficiary Dashboard

4.5.2.1. The beneficiary must be able to view the existing listings and display its details. They are able to do so by searching by food listing names or filter by status.

4.5.2.2. The beneficiary must be able to view available donations and request for donations based on their requirements. They are able to do so by searching by food listing names

4.5.2.3. The beneficiary must be able to accept donations from donors. Upon accepting, the beneficiary must key in their location details and delivery date.

4.5.2.4. The beneficiary must be able to put up listings to request for new donations.

4.5.2.5. The beneficiary must be able to withdraw their listings.

4.5.2.6. The beneficiary must be able to submit feedback at any point of time.

4.5.2.7. The beneficiary must be able to view the status of the listings.

#### 4.5.3. Admin Dashboard

4.5.3.1. The admin must be able to view the total number of donations pending collection and deliveries after successful matches between beneficiaries and donors.

4.5.3.2. The admin must be able to view the details of each respective listing that is ready to be collected/delivered.

4.5.3.3. The admin must be able to view the fastest route generated from the Route Optimization System.

4.5.3.4. The admin must be able update the status of the delivery to “Delivered” via a button click.

### 4.6. Donation Match Confirmation System

#### 4.6.1. Donation Match:

4.6.1.1. The system must detect and respond to any changes made by users, such as withdrawals or acceptance of matches, and must update accordingly to work seamlessly with the dashboard system and notification system.

#### 4.6.2. Status Updates

##### 4.6.2.1. New

4.6.2.1.1. The system must link up with the dashboard system to display the status "new" indicating that the item has been newly listed.

##### 4.6.2.2. Matched

4.6.2.2.1. The system must link with the dashboard system to display the status "matched" indicating that the donation/request has been matched between donor and beneficiary.

##### 4.6.2.3. In Warehouse

4.6.2.3.1. The system must link with the dashboard system to display the status "inwarehouse," indicating that the donor has successfully delivered the item to our warehouse.

##### 4.6.2.4. Awaiting Delivery

4.6.2.4.1. The system must link up with the dashboard system to display the status "awaitingdelivery" indicating that the admin is on the way to deliver the donation to the beneficiary.

##### 4.6.2.5. Delivered

4.6.2.5.1. The system must link with the dashboard system to display the status "delivered," indicating that the donation has been successfully sent to the beneficiary.

### 4.7. Route Optimization

#### 4.7.1. Default Scheduled Delivery

4.7.1.1. The system must be restricted to only accept scheduled delivery managed by admins and must not allow any other delivery options.

#### 4.7.2. Route Planning

4.7.2.1. The system must calculate the most efficient path from headquarters to the beneficiary locations once admin decides to pick up/deliver.

4.7.2.2. The system must use the Google Maps API to generate an optimized route that is within a 10% margin of the shortest possible distance.

### 4.8. Notifications and Alerts

4.8.1. The system must provide status updates for the listing, visible to both the beneficiary and donor, reflecting any changes in the listing's progress.

4.8.2. The system must allow the beneficiary and donor to view detailed information about the match, including:

4.8.2.1. Food name

4.8.2.2. Quantity

4.8.2.3. Date added

### 4.9. Chatbot Functionality

4.9.1. The system must provide a chatbot that process the query and provides relevant response.

4.9.2. The chatbot must be accessible from the main menu and available to all user roles (donor, beneficiary, admin, or potential users).

### 4.10. Interaction with Other Systems

4.10.1. The system must retrieve supermarket location data from Data.gov.sg.

4.10.2. The system must use the Google Maps API to find the shortest possible route for food deliveries.



## 5. Functional Requirements (SRS Format)

### 5.1 Main Menu

#### 5.1.1 Description and Priority

The main menu allows users to navigate the system, including account registration, logging in, food donation submissions, food requests, approval of matches, and chatbot assistance. This feature is of High priority as it is the central hub for all user actions and interactions within the system.

#### 5.1.2 Stimulus/Response Sequences

**Stimulus:** User accesses the main menu.

**Response:** System displays options for registration, login, food donation, food request, match approvals, and chatbot assistance.

**Stimulus:** User selects "Register" from the menu.

**Response:** System directs the user to the registration page.

**Stimulus:** User selects "Login."

**Response:** System presents the login form for the user to enter credentials.

**Stimulus:** Donor selects "Submit Food Donation."

**Response:** System displays the donation submission form.

#### 5.1.3 Functional Requirements

**REQ 1:** The system must allow the user to register their account.

**REQ 2:** The system must allow the user to log in to their account.

**REQ 3:** The system must allow the donor to send food donation details.

**REQ 4:** The system must allow the beneficiary to submit food requests.

**REQ 5:** The system must include a chatbot to assist users.

## 5.2 User Registration

### 5.2.1 Description and Priority

User registration allows new users (donors, beneficiaries, admins) to create accounts. **High** priority as it enables access to the platform.

### 5.2.2 Stimulus/Response Sequences

**Stimulus:** User fills out the registration form.

**Response:** System checks for missing or invalid fields and prompts corrections.

**Stimulus:** User submits the form.

**Response:** System verifies information and creates an account if successful.

### 5.2.3 Functional Requirements

**REQ 1:** The system must allow the user to register a new account.

**REQ 2:** The system must allow users to input specific registration requirements for each type of user, such as admin, beneficiary, and donor.

**REQ 3:** The system must validate that all required fields are filled out before allowing account creation.

**REQ 4:** The system must check the availability of the email address before completing the registration.

**REQ 5:** The system must display appropriate error messages to guide the user for correction if any required fields are missing or incorrectly filled.

**REQ 6:** The system must allow the user to specify their role (donor, beneficiary, admin) during registration.

**REQ 7:** The system must allow approved admin to register.

## 5.3 Login

### 5.3.1 Description and Priority

This feature allows users to securely access their accounts using email and password. **High** priority as it is essential for system access.

### 5.3.2 Stimulus/Response Sequences

**Stimulus:** User submits login form with email and password.

**Response:** System verifies credentials and grants access.

**Stimulus:** User enters incorrect login details.

**Response:** System displays an error message and requests re-entry.

### 5.3.3 Functional Requirements

**REQ 1: The system must allow the user to log in using their registered email and password.**

**REQ 2: The system must validate that the email address and password are correct.**

**REQ 3: The system must display an error message if the login information is incorrect.**

**REQ 4: The system must redirect the user to their respective dashboard upon successful login.**

## 5.4 Submissions

### 5.4.1 Description and Priority

This feature allows donors to submit food donations and beneficiaries to request food. High priority as it facilitates core system operations.

### 5.4.2 Stimulus/Response Sequences

**Stimulus:** Donor fills in the food donation form.

**Response:** System verifies inputs and submits the donation details.

**Stimulus:** Beneficiary submits a food request form.

**Response:** System validates the request and records it in the system.

### 5.4.3 Functional Requirements

**REQ 1: Food Donation Submission**

1.1. The system must allow donors to submit food donation details.

1.2. The system must require the following details in the food donation form:

- Food Name
- Food Category
- Quantity
- Consue By
- Special Storage/Handling Requirements

1.3. The system must validate that all fields in the food donation form are filled out before submission.

**REQ 2: Food Request Submission**

- 2.1. The system must allow beneficiaries to submit food requests.
- 2.2. The system must require the following details in the food request form:

- Food Name
- Food Category
- Quantity
- Special Requests
- Request Food By
- Delivery Location
- Floor Number

**REQ 3: The system must validate that all fields in the food request form are filled out before submission.**

## 5.5 Dashboard

### 5.5.1 Description and Priority

The dashboard provides an overview and management tools for donors, beneficiaries, and admins. High priority as it centralizes user interactions.

### 5.5.2 Stimulus/Response Sequences

**Donor's Perspective:**

**Stimulus:** Donor logs in and views the dashboard.

**Response:** The system displays all current donations, their statuses (new, matched, inwarehouse, awaitingdelivery and delivered), new opportunities for donation, and any feedback prompts after completed donations.

**Stimulus:** Donor completes a donation.

**Response:** The system prompts the donor to provide feedback about the donation process and the beneficiary.

**Beneficiary's Perspective:**

**Stimulus:** Beneficiary logs in and views available food donations.

**Response:** The system shows donation listings that match the beneficiary's criteria (food type, quantity, etc.) and displays their submitted requests and their statuses.

**Stimulus:** Beneficiary receives a donation.

**Response:** The system prompts the beneficiary to provide feedback on the donor and the quality of the food received.

#### **Admin's Perspective:**

**Stimulus:** Admin logs in and views the dashboard.

**Response:** The system displays all successful matches, including the number of donations in the warehouse and waiting for delivery, and provides route optimization for deliveries.

**Stimulus:** Admin accepts the delivery.

**Response:** The system displays the shortest route generated by the Google Maps API, along with the shortest distance to the beneficiary.

**Stimulus:** Admin updates the status of a completed delivery.

**Response:** The system notifies both donor and beneficiary of the status and prompts them for feedback.

### 5.5.3 Functional Requirements

#### **5.4.3.1. Donor Dashboard**

**REQ 1:** The donor must be able to view the existing listings and display its details.

**REQ 2:** The donor must be able to browse through requests from beneficiaries to find new donation opportunities.

**REQ 3:** The donor must have the option to donate to the beneficiary if the request falls within their means.

**REQ 4:** The donor must be able to put up new listings for donations that are open for beneficiaries to accept.

**REQ 5:** The donor must be able to withdraw their existing listings.

**REQ 6:** The donor must be able to view the status of the listings.

**REQ 7:** The donor must be able to submit feedback at any point of time.

#### **5.5.3.2. Beneficiary Dashboard**

**REQ 1:** The beneficiary must be able to view the existing listings and display its details.

**REQ 2:** The beneficiary must be able to view available donations and request for donations based on their requirements.

- REQ 3:** The beneficiary must be able to accept donations from donors.
- REQ 4:** The beneficiary must be able to put up listings to request for new donations.
- REQ 5:** The beneficiary must be able to withdraw their listings.
- REQ 6:** The beneficiary must be able to view the status of the listings.
- REQ 7:** The beneficiary must be able to submit feedback at any point of time.

#### 5.5.3.3. Admin Dashboard

- REQ 1:** The admin must be able to view the total number of donations pending collection and deliveries after successful matches between beneficiaries and donors.
- REQ 2:** The admin must be able to view the details of each respective listing that is ready to be collected/delivered.
- REQ 3:** The admin must be able to view the fastest route generated from the Route Optimization System.
- REQ 4:** The admin must be able update the status of the delivery to “Completed” via a button click.

#### 5.5.3.4. Search and Filter

- REQ 1:** Both donors and beneficiaries should have search and filter functionality to quickly locate specific items or requests. They should be able to search by name or apply filters based on the item's status.

### 5.6 Donation Match Confirmation System

#### 5.6.1 Description and Priority

This feature facilitates the confirmation process of donation matches between donors and beneficiaries. Once both parties mutually accept the match, the

system updates the status accordingly. High priority as it is essential for ensuring successful transactions.

#### 5.6.2 Stimulus/Response Sequences

**Stimulus:** Donor offers a donation, and the beneficiary views it.  
**Response:** Beneficiary accepts the donation.

**Stimulus:** Beneficiary sends a request, and the donor views it.  
**Response:** donor accepts the requests.

**Stimulus:** Both donor and beneficiary mutually accept the donation.  
**Response:** The system confirms the match and updates the status for both parties in their respective dashboards.

#### 5.6.3 Functional Requirements

##### 5.6.3.1. Donation Match:

**REQ 1:** The system must detect and respond to any changes made by users, such as withdrawals or acceptance of matches, and must update accordingly to work seamlessly with the dashboard system and notification system.

##### 5.6.3.2. Status Updates

###### 5.6.3.2.1. New

**REQ 2:** The system must link up with the dashboard system to display the status "new" indicating that the item has been newly listed.

###### 5.6.3.2.2. Matched

**REQ 3:** The system must link with the dashboard system to display the status "matched" indicating that the donation/request has been matched between donor and beneficiary.

###### 5.6.3.2.3. In Warehouse

**REQ 4:** The system must link with the dashboard system to display the status "inwarehouse," indicating that the donor has successfully delivered the item to our warehouse.

#### 5.6.3.2.4. Awaiting Delivery

**REQ 5:** The system must link up with the dashboard system to display the status "awaitingdelivery" indicating that the admin is on the way to deliver the donation to the beneficiary.

#### 5.6.3.2.5. Delivered

**REQ 6:** The system must link with the dashboard system to display the status "delivered," indicating that the donation has been successfully sent to the beneficiary.

### 5.7 Route Optimization

#### 5.7.1 Description and Priority

This feature helps determine the most efficient delivery routes for picking up and delivering food. Medium priority as it enhances delivery efficiency but is not critical for basic system functionality.

#### 5.7.2 Stimulus/Response Sequences

**Stimulus:** Admin selects a donation ready for delivery.

**Response:** System calculates the optimal route for pickup and delivery using Google Maps API.

**Stimulus:** Admin confirms the route.

**Response:** System displays the generated route and estimated time of delivery.

#### 5.7.3 Functional Requirements

**REQ-1:** The system must be restricted to only accept scheduled delivery managed by admins and must not allow any other delivery options.

**REQ-2:** The system must calculate the most efficient path from headquarters to both donor and beneficiary locations once admin decides to pick up/deliver.

**REQ-3:** The system must use the Google Maps API to generate an optimized route that is within a 10% margin of the shortest possible distance.

**REQ-4:** The system must allow admin to mark the listing as delivered once they have delivered to the beneficiary.

## 5.8 Notifications and Alerts

### 5.8.1 Description and Priority

This feature ensures that users are kept informed about their donation or request statuses through notifications. High priority as it keeps users engaged and informed about key activities.

### 5.8.2 Stimulus/Response Sequences

**Stimulus:** Admin updates the status of a delivery.

**Response:** The system must provide status updates for the listing, visible to both the beneficiary and donor, reflecting any changes in the listing's progress.

### 5.8.3 Functional Requirements

**REQ-1:** The system must notify the beneficiary and donor when their match is mutually accepted through email.

**REQ-2:** The system must allow the beneficiary and donor to view detailed information about the match, including:

- Food name
- Quantity
- Date added

## 5.9 Chatbot Functionality

### 5.9.1 Description and Priority

The chatbot provides real-time assistance to users regarding platform usage. Medium priority as it improves user experience but is not essential for core operations.

#### 5.9.2 Stimulus/Response Sequences

**Stimulus:** User accesses the chatbot from the main menu.

**Response:** The chatbot provides options for account registration, login, and submission guidance.

**Stimulus:** User asks a question about how to submit a donation.

**Response:** The chatbot guides the user step-by-step through the donation submission process.

#### 5.9.3 Functional Requirements

**REQ-1:** The system must provide a chatbot that processes the query and provides relevant responses.

**REQ-2:** The chatbot must be accessible from the main menu and available to all user roles (donor, beneficiary, admin, or potential users).

### 5.10 Interaction with Other Systems

#### 5.10.1 Description and Priority

This feature handles interactions between the system and external services, such as retrieving supermarket data and route planning. High priority as it allows the system to function seamlessly with external data sources.

#### 5.10.2 Stimulus/Response Sequences

**Stimulus:** Donor registers and selects a location for food drop-off.

**Response:** System retrieves the nearest supermarket location from Data.gov.sg and displays it.

**Stimulus:** Admin requests route optimization for a delivery. **Response:** System uses Google Maps API to calculate the shortest route.

#### 5.10.3 Functional Requirements

**REQ-1:** The system must retrieve supermarket location data from Data.gov.sg.

**REQ-2: The system must use the Google Maps API to find the shortest possible route for food deliveries.**

## 6. Non-functional Requirements

### 6.1. Performance

#### 6.1.1. Response Time

6.1.1.1. The system must respond to user interactions (such as form submissions) within 3 seconds under normal load conditions.

6.1.1.2. The system must process food donation and request submissions within 5 seconds.

#### 6.1.2. Scalability

6.1.2.1. The system must support up to 100 simultaneous users without performance degradation.

6.1.2.2. The system must handle up to 50 food donations and requests per hour during peak times.

#### 6.1.3. Availability

6.1.3.1. The system must be available 99.9% of the time per month.

6.1.3.2. The system must have a disaster recovery plan to restore service within 1 hour in case of major failure.

### 6.2. Safety

#### 6.2.1. Data Integrity

6.2.1.1. The system must ensure that data entered by users is accurate and consistent across all components of the application.

6.2.1.2. The system must provide error handling and validation mechanisms to prevent data corruption.

#### 6.2.2. Error Handling

6.2.2.1. The system must handle errors gracefully and provide meaningful error messages to users.

6.2.2.2. The system must log errors and exceptions for troubleshooting and auditing purposes.

## 6.3. Security

### 6.3.1. Authentication and Authorization

6.3.1.1. The system must use secure authentication methods (e.g., hashed passwords) to ensure user identity.

6.3.1.2. The system must implement role-based access control to ensure users can only access functionalities appropriate to their role.

### 6.3.2. Data Protection

6.3.2.1. The system must encrypt sensitive data (such as personal details and donation information) both in transit and at rest.

6.3.2.2. The system must comply with relevant data protection regulations (e.g., GDPR, PDPA).

### 6.3.3. Vulnerability Management

6.3.3.1. The system must be regularly tested for security vulnerabilities and updated with security patches.

6.3.3.2. The system must protect against common web security threats (e.g., SQL injection, cross-site scripting).

## 6.4. Software Quality Attributes

### 6.4.1. Usability

6.4.1.1. The system must have an intuitive and user-friendly interface that is easy to navigate.

6.4.1.2. The system must provide help and support resources (e.g., FAQs, chatbot) for users to assist with common tasks and issues.

### 6.4.2. Maintainability

6.4.2.1. The system must be designed for ease of maintenance and updates, with clear documentation for developers.

6.4.2.2. The system must use modular architecture to simplify updates and bug fixes.

### 6.4.3. Interoperability

6.4.3.1. The system must integrate seamlessly with external systems (e.g., government and Google Maps API) as required.

6.4.3.2. The system must be compatible with commonly used web browsers and devices.

## **6.5. Business Rules**

6.5.1. Role-Specific Functionality

6.5.1.1. Donors, beneficiaries must have access to functionalities appropriate to their role, as defined by the system's role-based access control.

6.5.2. Data Handling and Privacy

6.5.2.1. The system must handle and store data in compliance with business rules and legal requirements.

6.5.2.2. The system must provide data access and control features to users based on their role and permissions.

## 7. Other Requirements

The Food Hero app must meet the following additional requirements to ensure optimal functionality, compliance, and future scalability:

### 1. Database Requirements:

- **High Availability:** The MongoDB database must be configured for high availability, capable of handling large volumes of data efficiently and ensuring consistent performance under peak loads.
- **Backup and Recovery:** The database should include robust backup and recovery mechanisms to prevent data loss and support rapid data restoration in the event of an incident.

### 2. Legal Requirements:

- **Regulatory Compliance:** The app must comply with all applicable local and international laws and regulations, including those related to food safety, data protection, and electronic communications.
- **Data Protection:** Adherence to Singapore's Personal Data Protection Act (PDPA) is mandatory, ensuring that user data is managed securely and responsibly.
- **Food Safety Standards:** The app must align with food safety regulations to ensure that the handling, distribution, and management of food donations meet legal health and safety requirements.

### 3. Reuse Objectives:

- **Component Reusability:** The app's architecture should be modular and designed with reusability in mind. Key components, such as the matching algorithm and chatbot integration, should be adaptable for future projects or potential app enhancements. This design approach supports long-term development flexibility and cost-efficiency.

These requirements ensure that the Food Hero app is reliable, compliant, and scalable, meeting the needs of users while maintaining a foundation for future improvements and expansion.

## Appendix A: Glossary

**AI (Artificial Intelligence):** A branch of computer science that enables machines to simulate human intelligence and perform tasks such as problem-solving, learning, and pattern recognition.

**API (Application Programming Interface):** A set of protocols and tools that allow different software applications to communicate and share data with each other.

**CRM (Customer Relationship Management):** A system or software used to manage interactions and relationships with current and potential customers or, in the case of Food Hero, between donors and beneficiaries.

**Gemini API:** The specific API used for the AI-powered chatbot functionality in the Food Hero app, providing users with automated support and interaction.

**HTTPS (Hypertext Transfer Protocol Secure):** A secure version of HTTP that uses encryption to protect data transferred between the user's browser and the server.

**MongoDB:** A NoSQL database known for its scalability and flexibility in handling large volumes of unstructured data, used as the backend database for the Food Hero app.

**Next.js:** A React-based framework for building server-rendered applications and static websites, used for both the frontend and backend development of the Food Hero app.

**PDPA (Personal Data Protection Act):** A regulation in Singapore that governs the collection, use, and disclosure of personal data to protect individuals' privacy.

**RESTful API:** An API that conforms to the constraints of REST (Representational State Transfer) architecture, enabling communication between the app's frontend and backend.

**TLS (Transport Layer Security):** A cryptographic protocol that provides security for data transferred over a network, ensuring data integrity and privacy.

**UI (User Interface):** The part of the app that users interact with directly, designed to be intuitive and user-friendly in the Food Hero app.

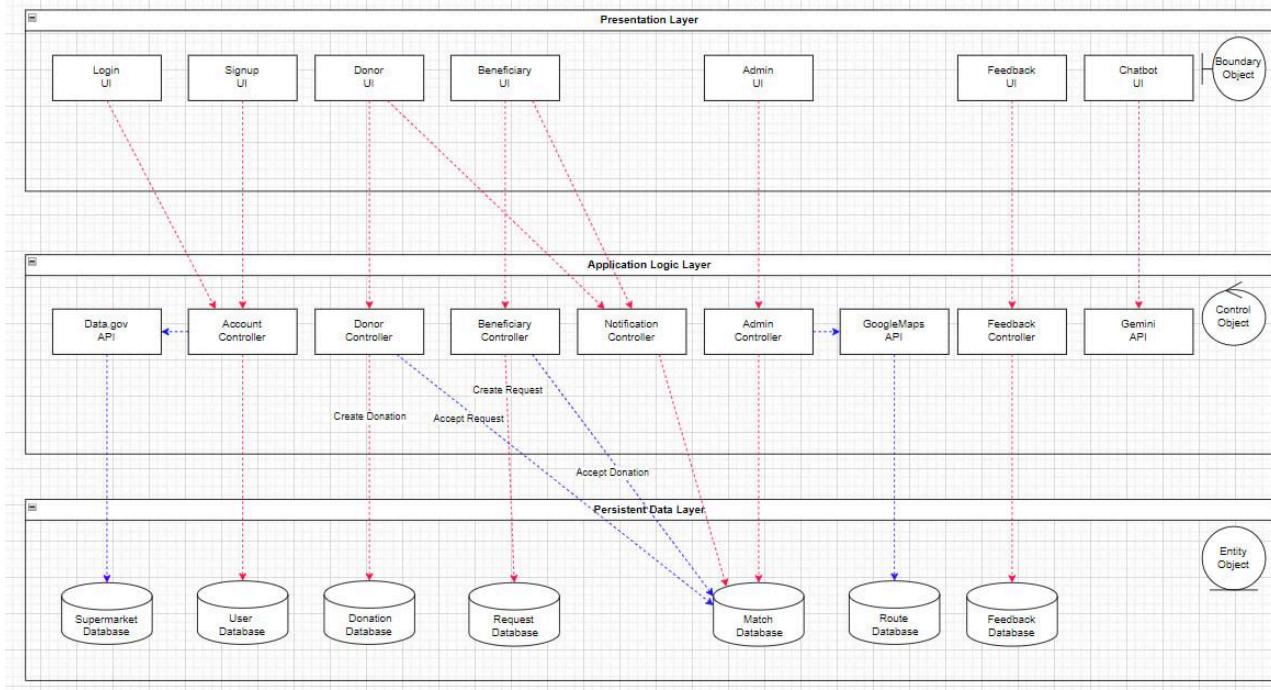
**User:** Refers to any individual who interacts with the Food Hero app, which includes donors, beneficiaries, and administrators.

**Web-Based Notification System:** The in-app system used to provide real-time updates and notifications to users regarding donation and request statuses.

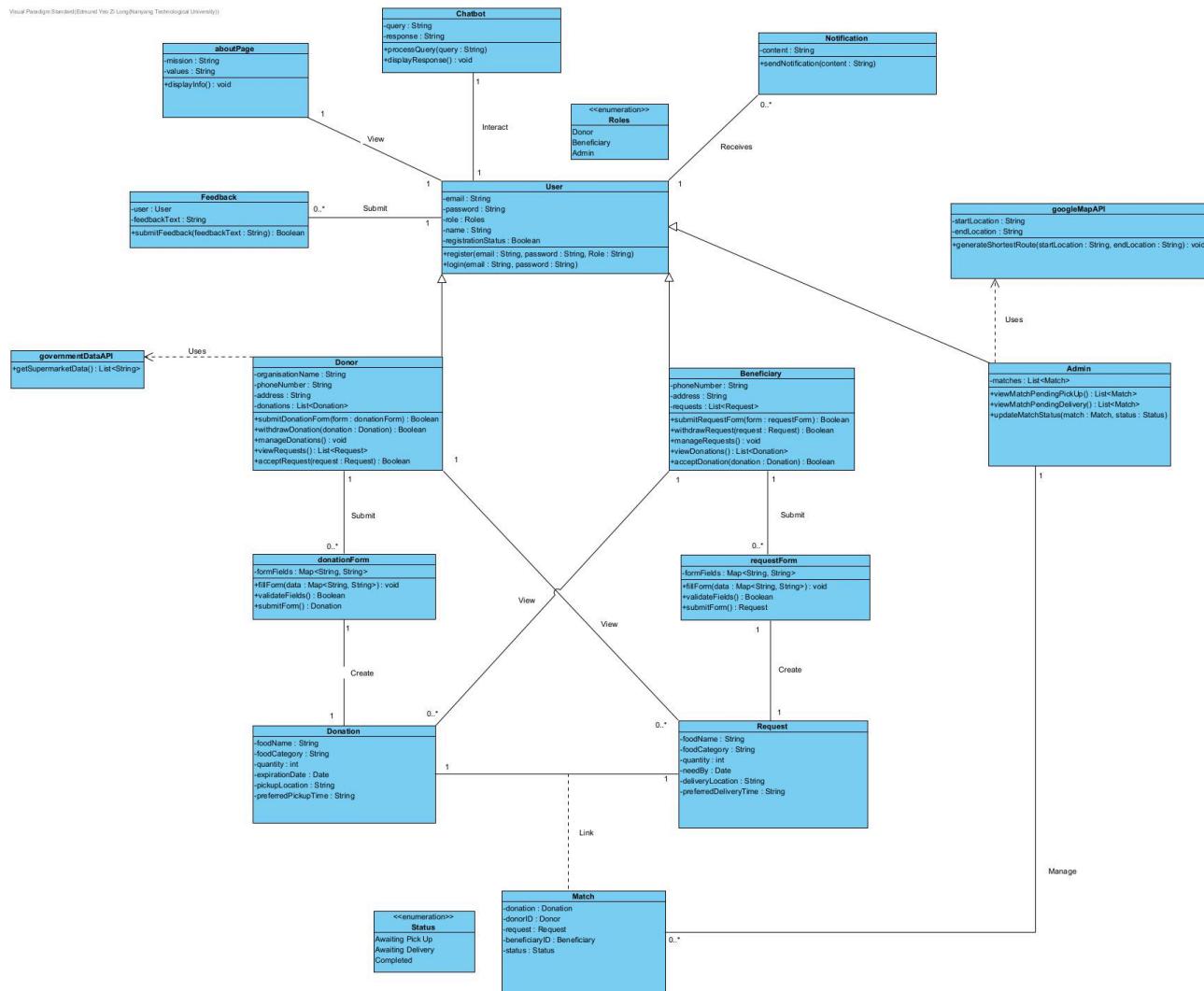
## 8. Data Dictionary

Term	Definition
User	An individual interacting with the system, categorized as donor, beneficiary or admin.
Donor	A user who provides food donations to the system
Beneficiary	A user who requests food from the system.
Admin	A user who manages and helps facilitate the deliveries of the donations between donors and beneficiaries
Dashboard	A user interface element that displays key information and allows users to manage their donations, requests, and account settings.
Food Donation	Food provided by a donor to the system
Food Request	A request made by a beneficiary for food.
Listing	A record posted by a beneficiary to request donations, detailing their specific needs for donor evaluation and response.
Status	Each status reflects a specific stage in the progression of a donation from acceptance to delivery. ie, "new," "matched," "inwarehouse", "awaitingdelivery", "delivered" etc.
Match	The pairing of a food donation with a food request based on mutual agreement between the donor and the beneficiary.
Feedback	Donors and Beneficiaries can send feedback based on their user experience.

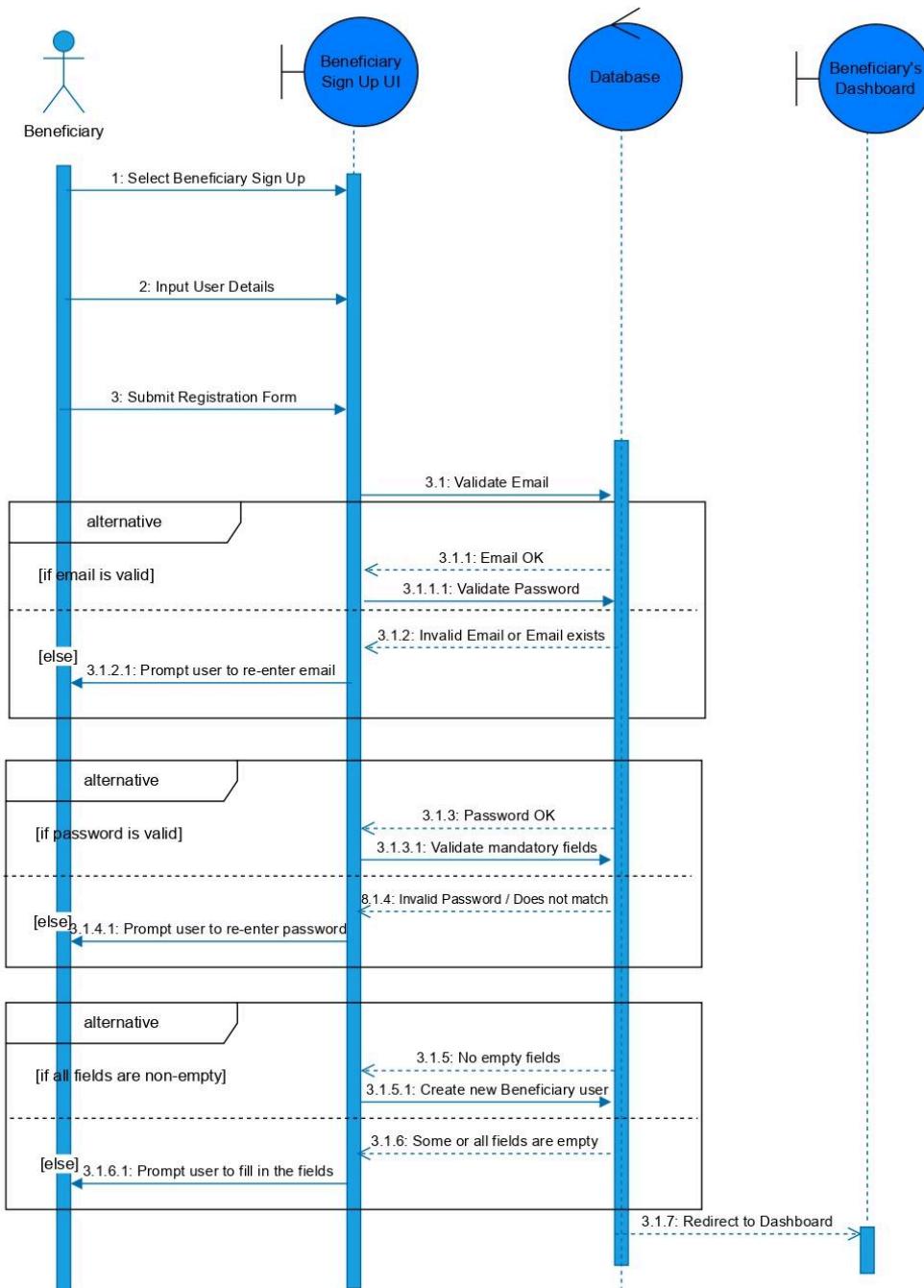
## 9. System Architecture Diagram

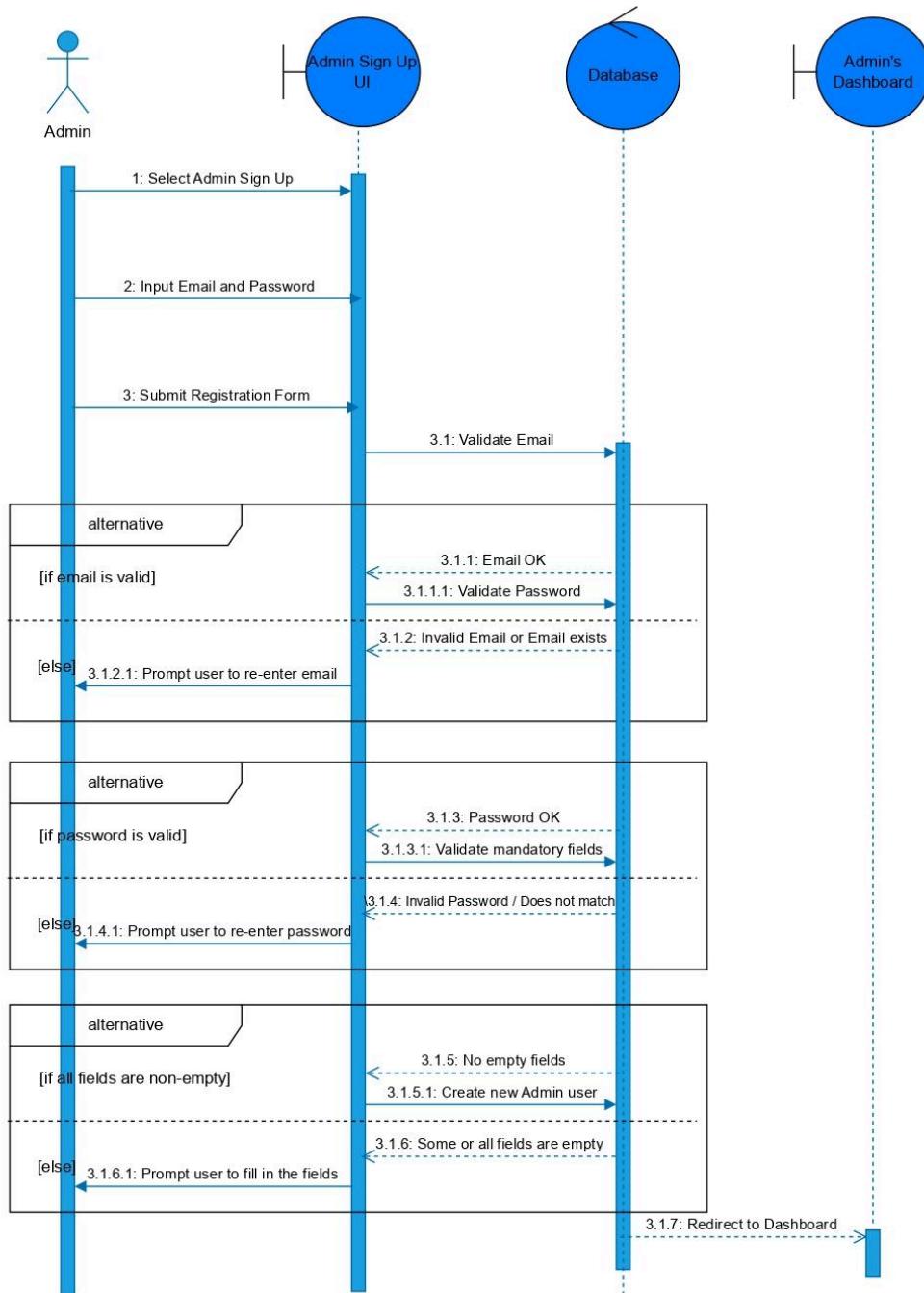


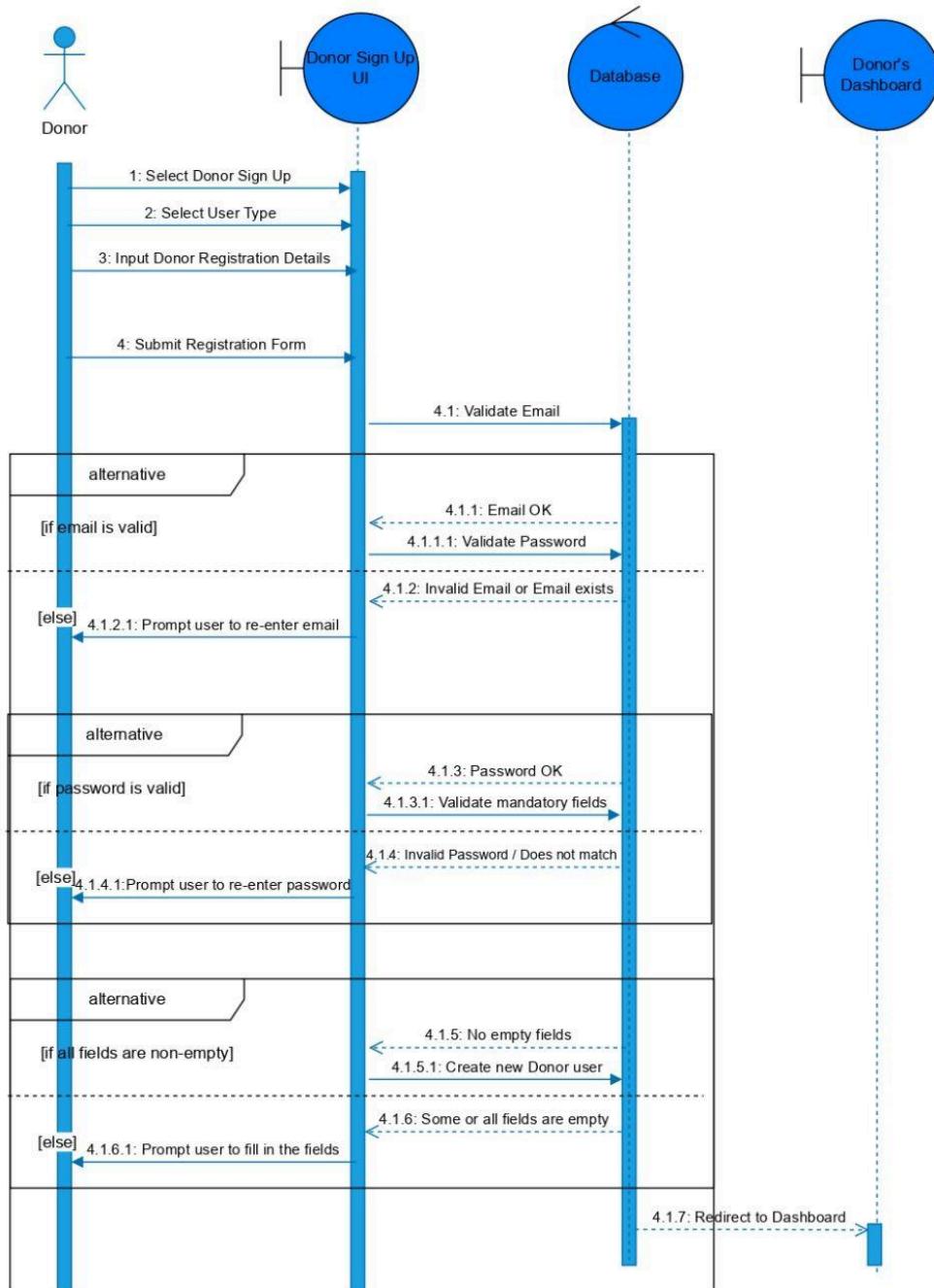
## 10. Class Diagram

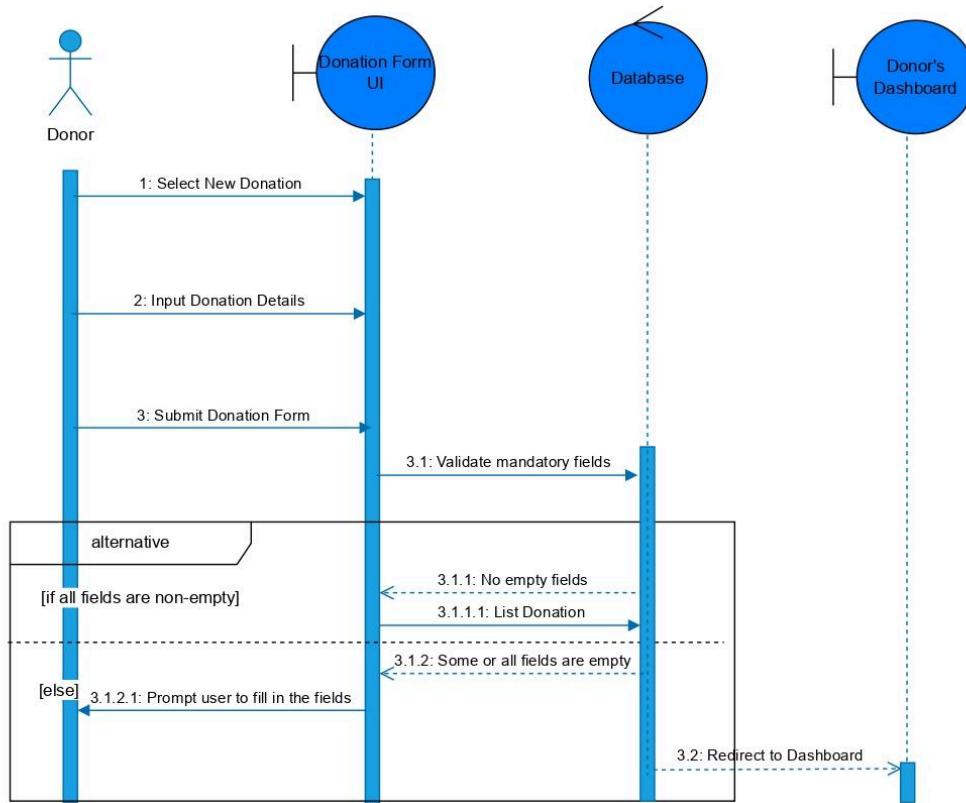


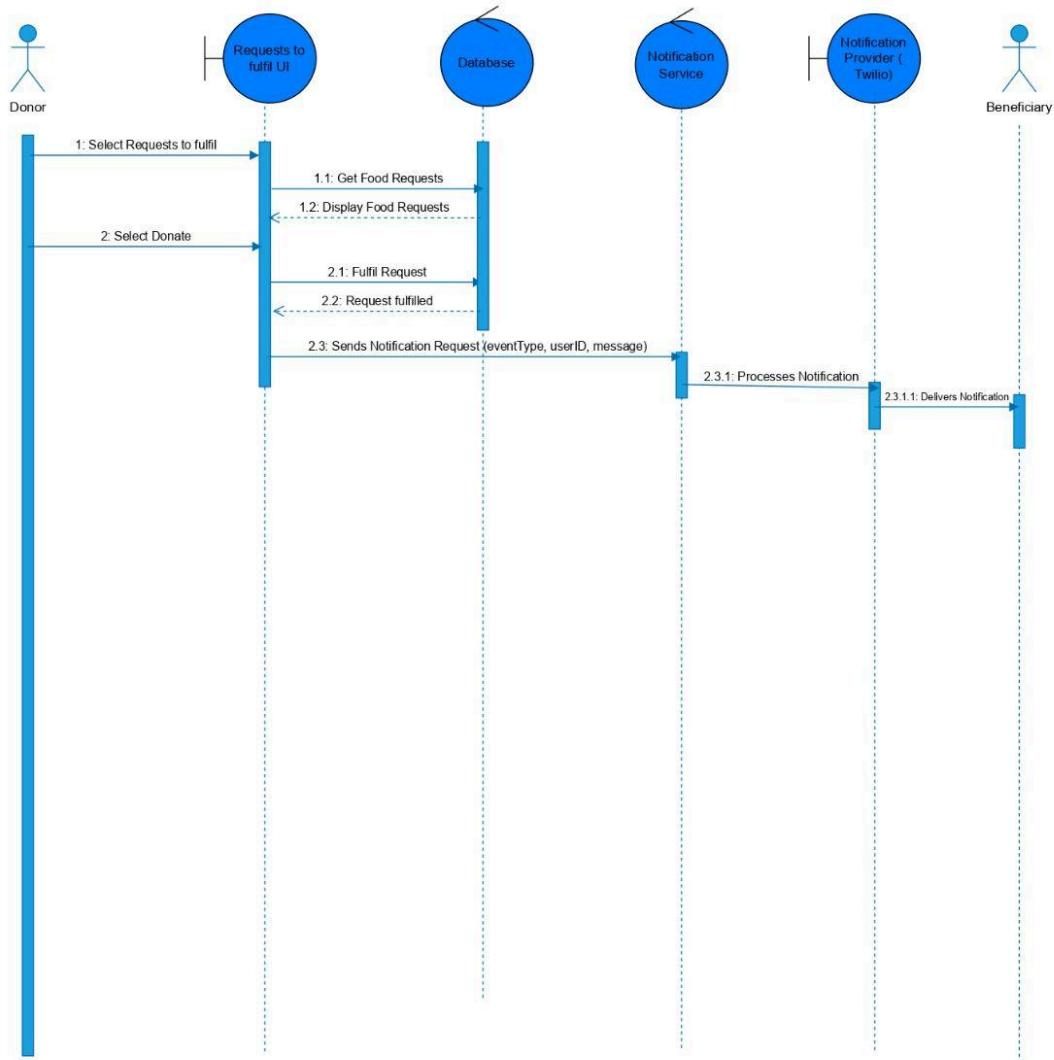
## **11. Sequence Diagrams**

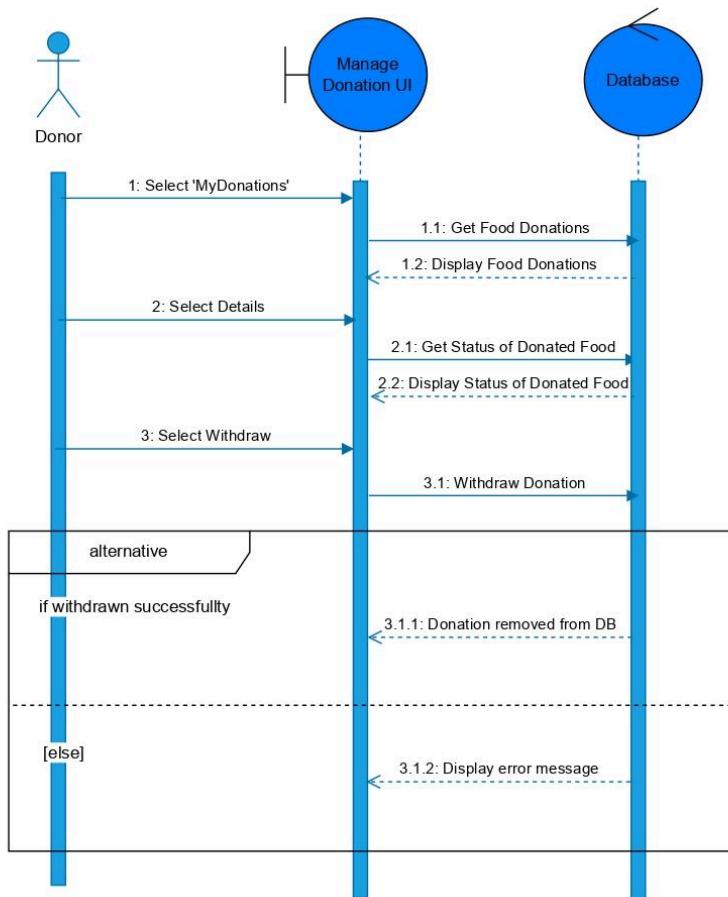


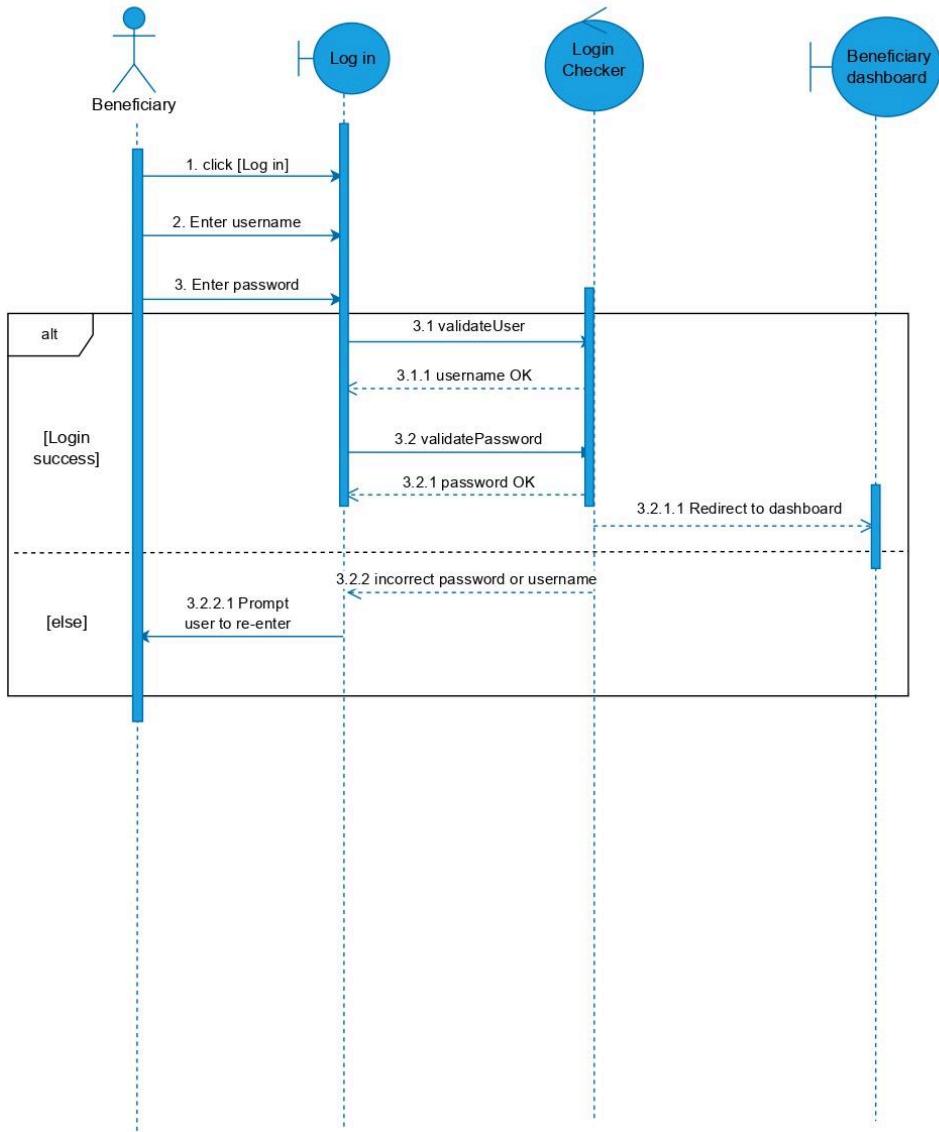


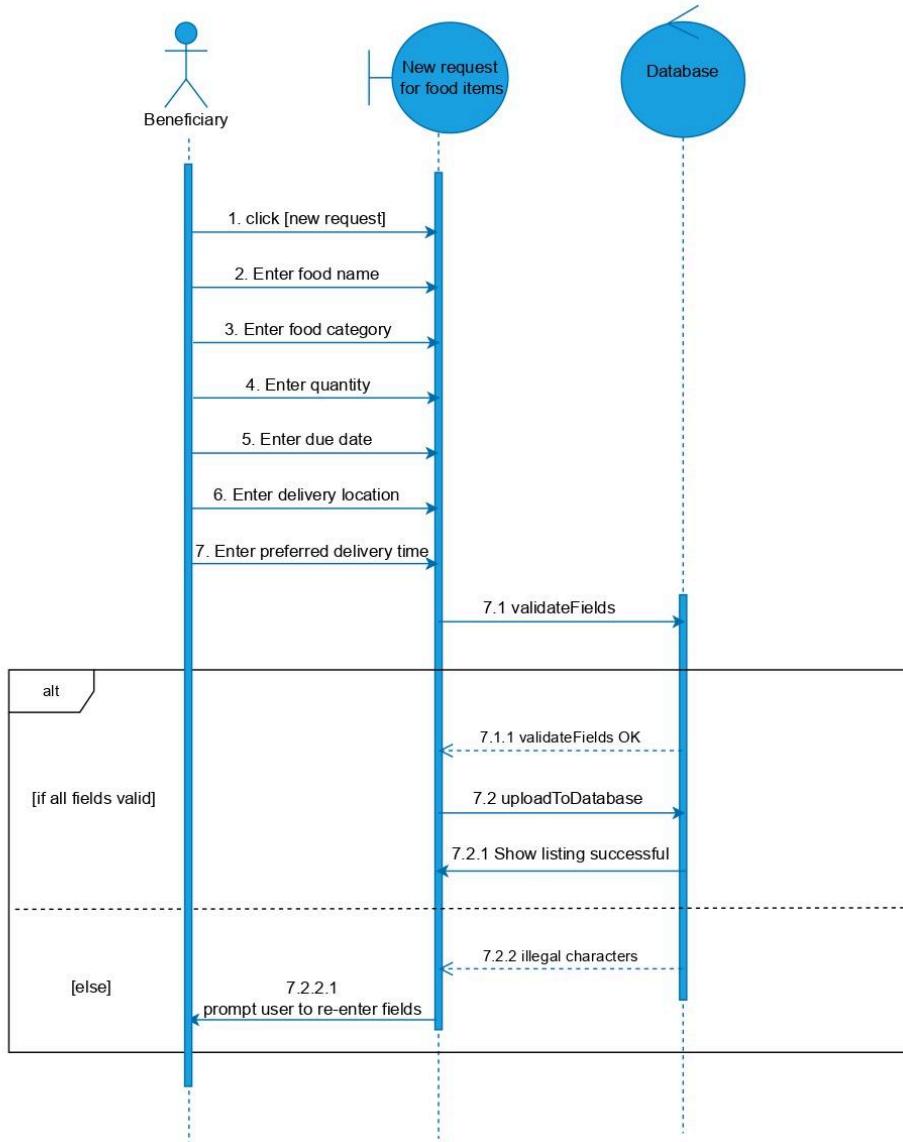


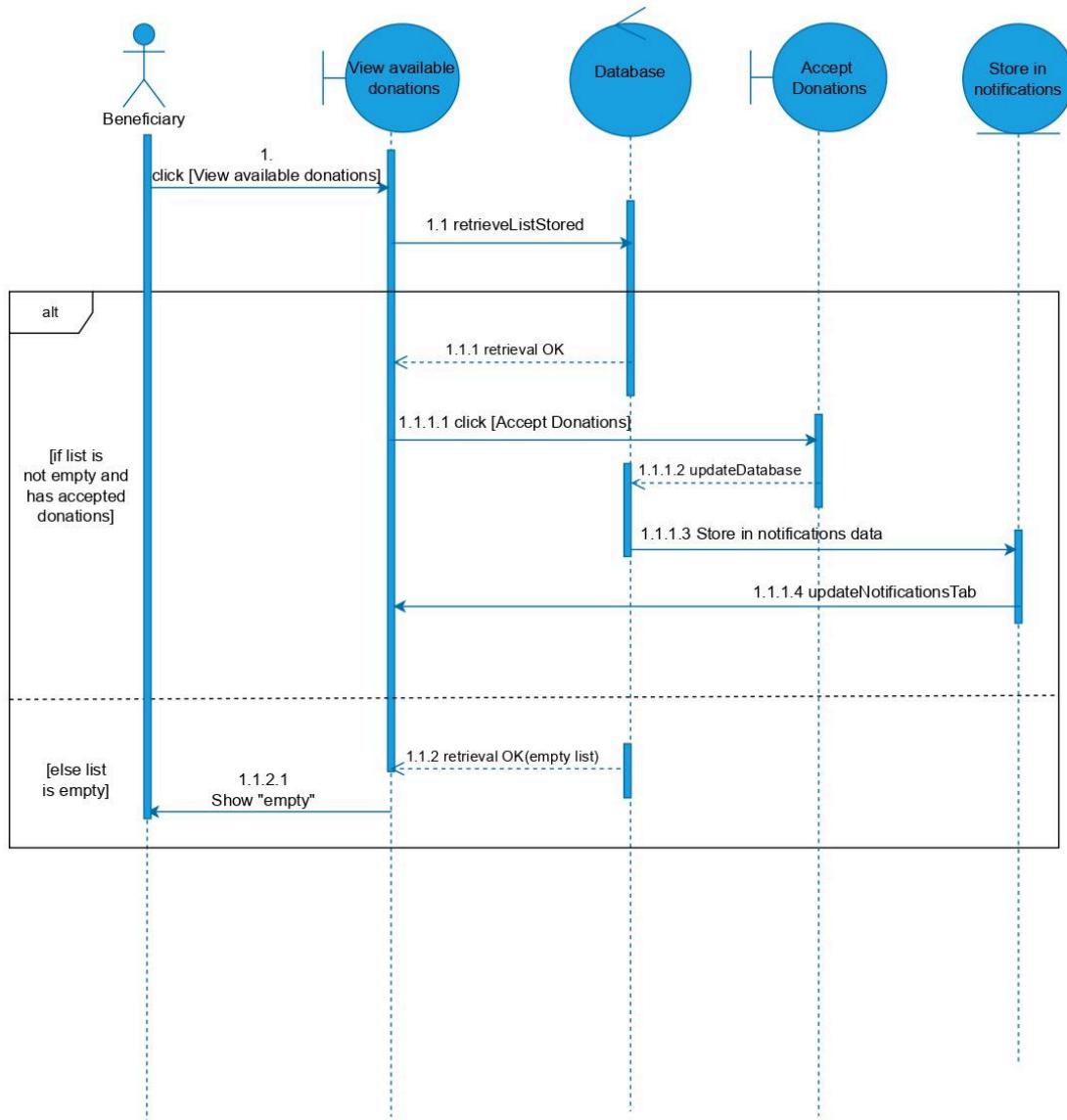


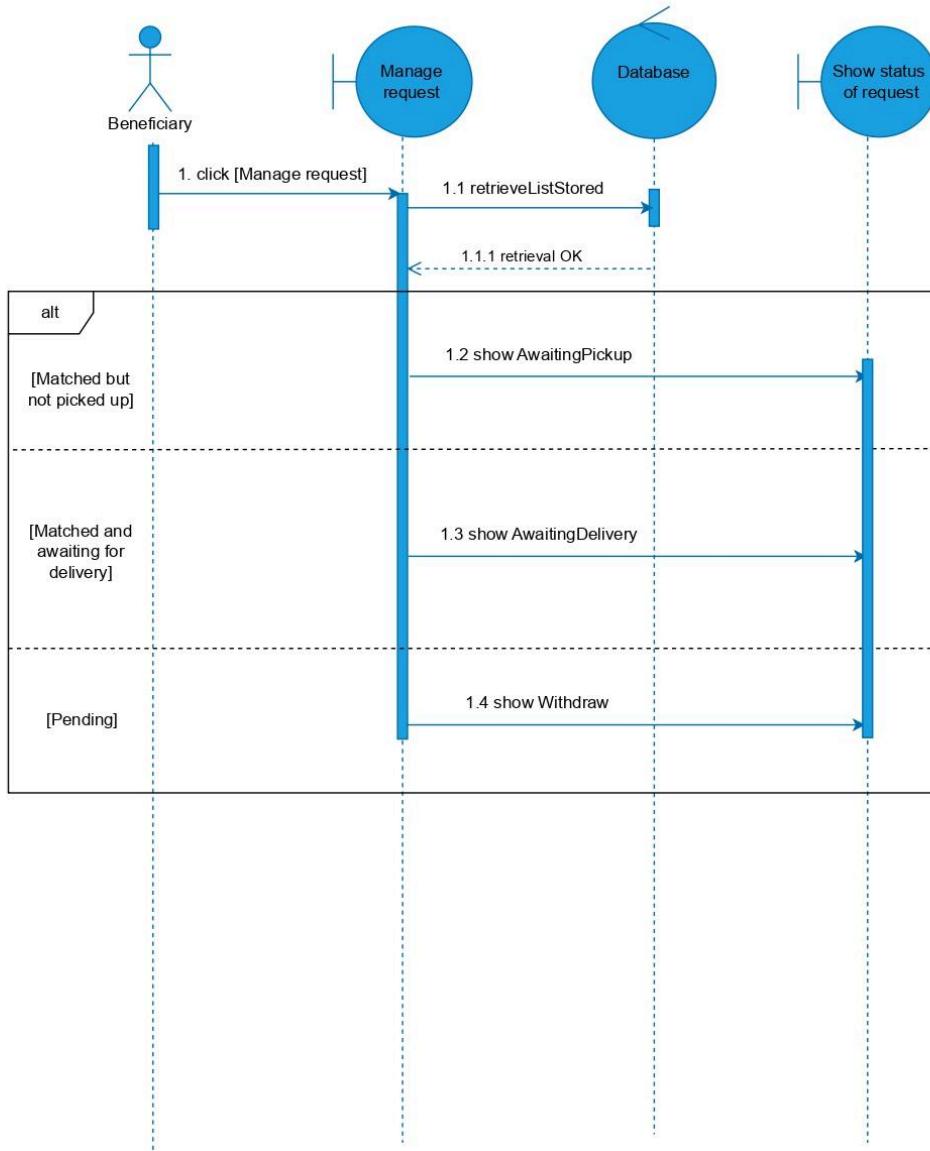


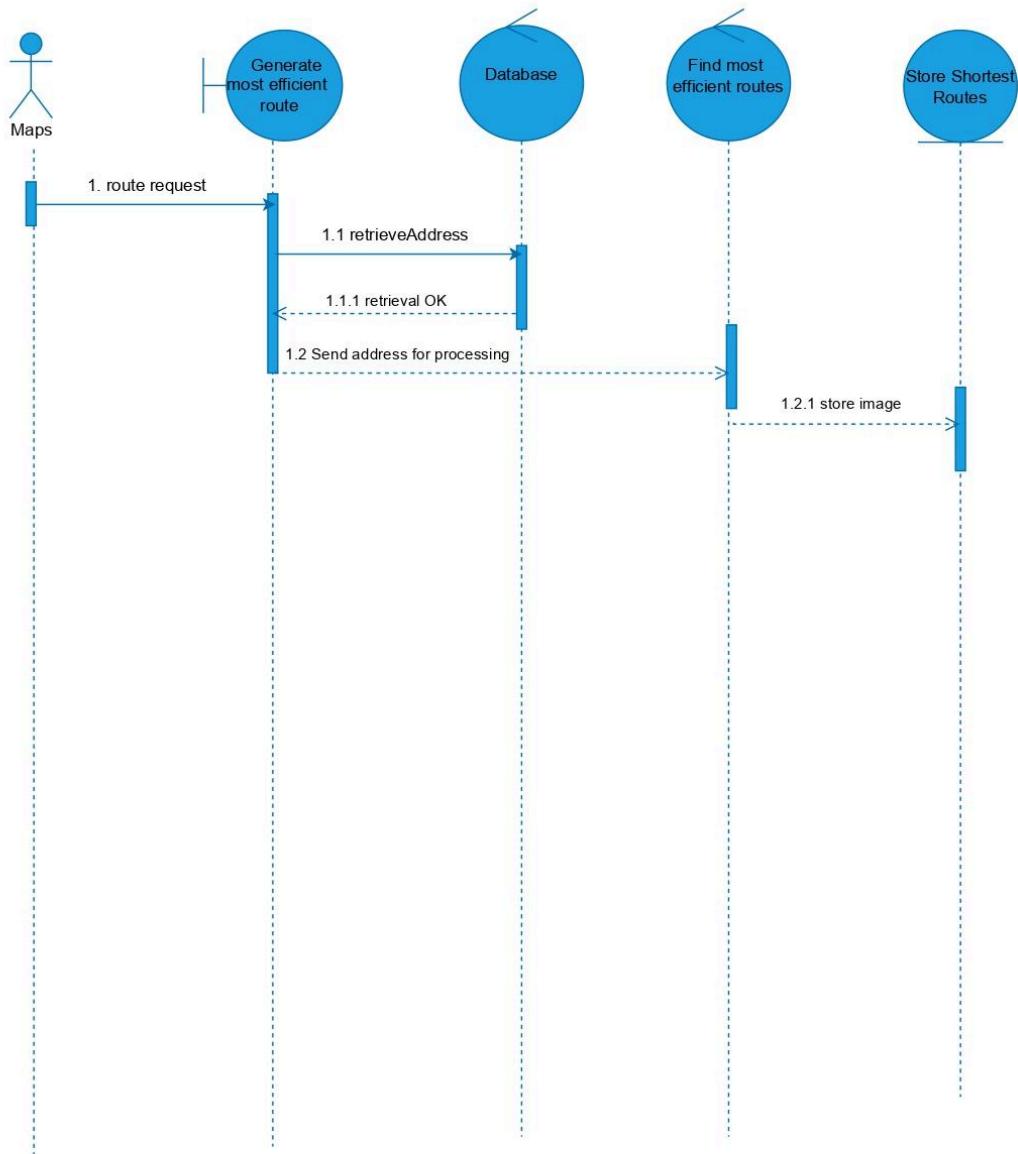


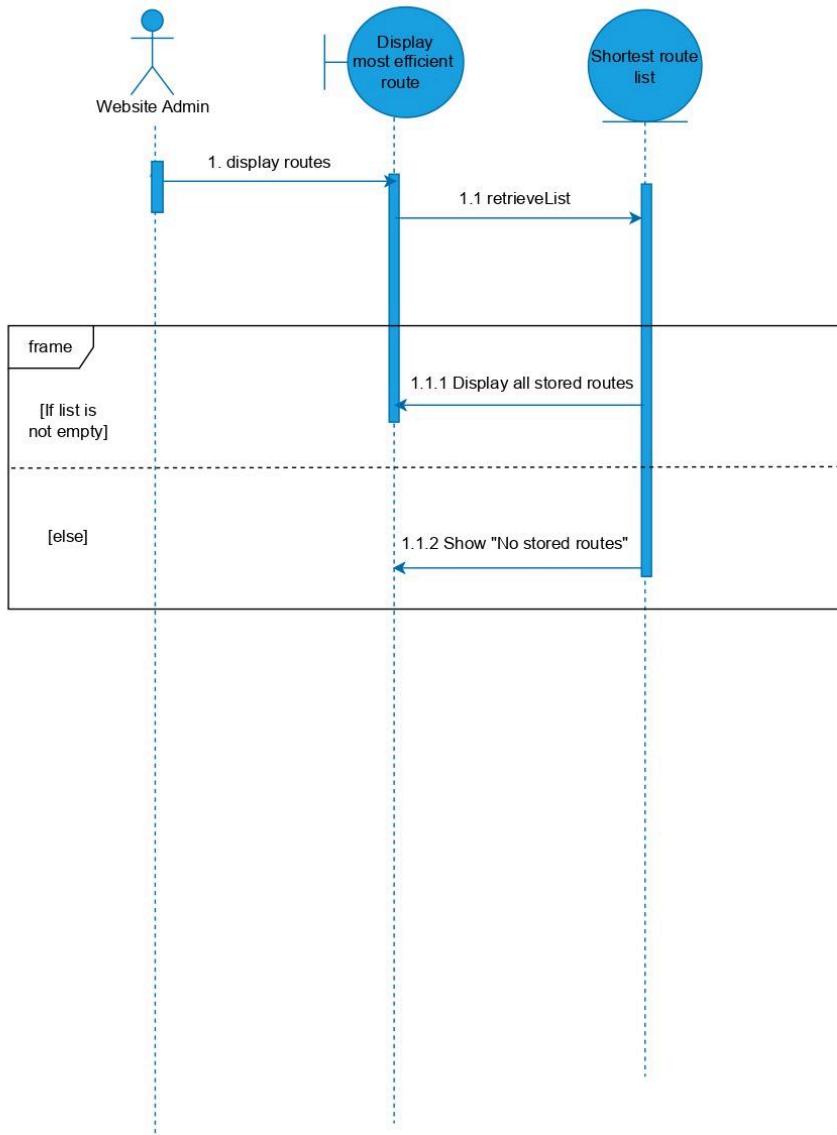


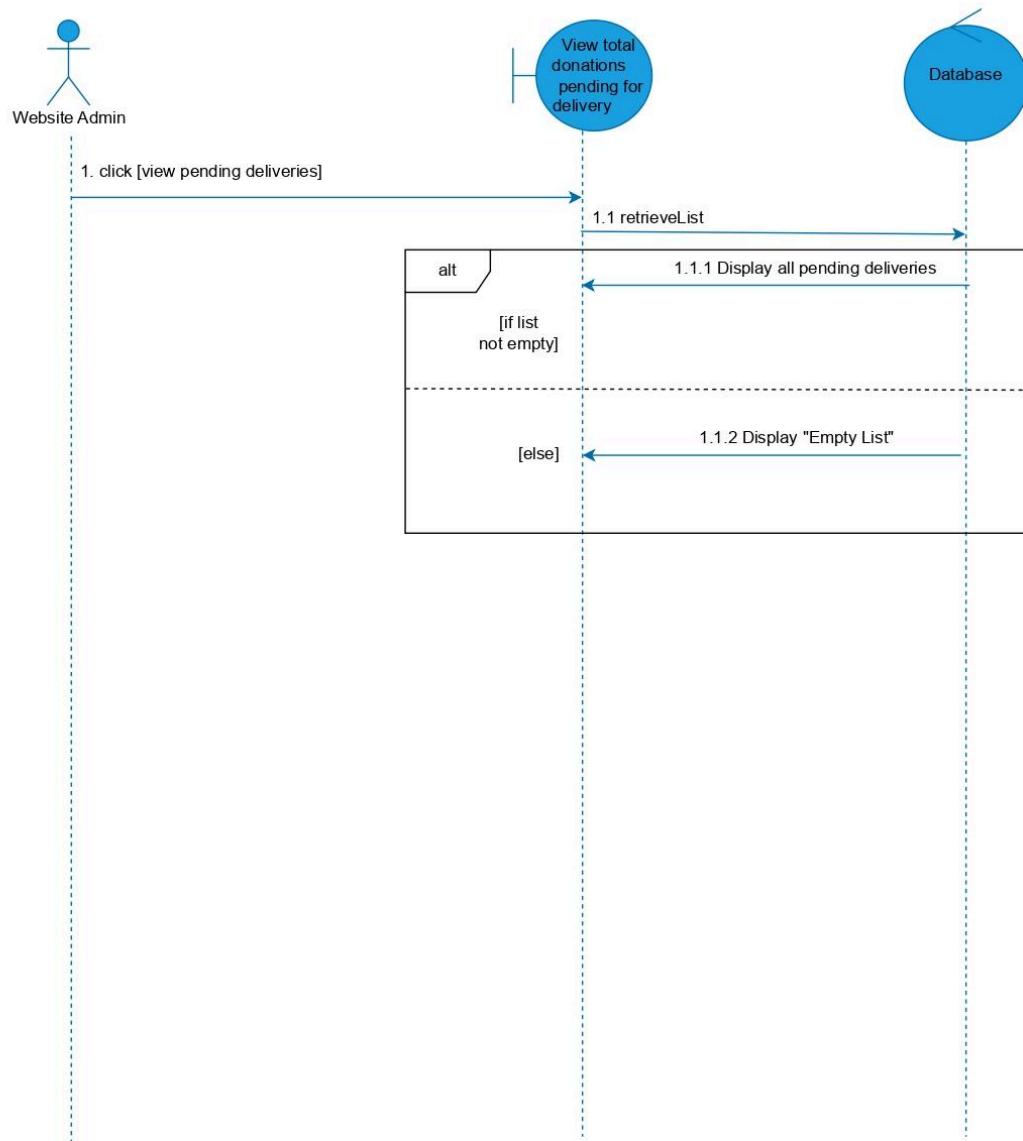


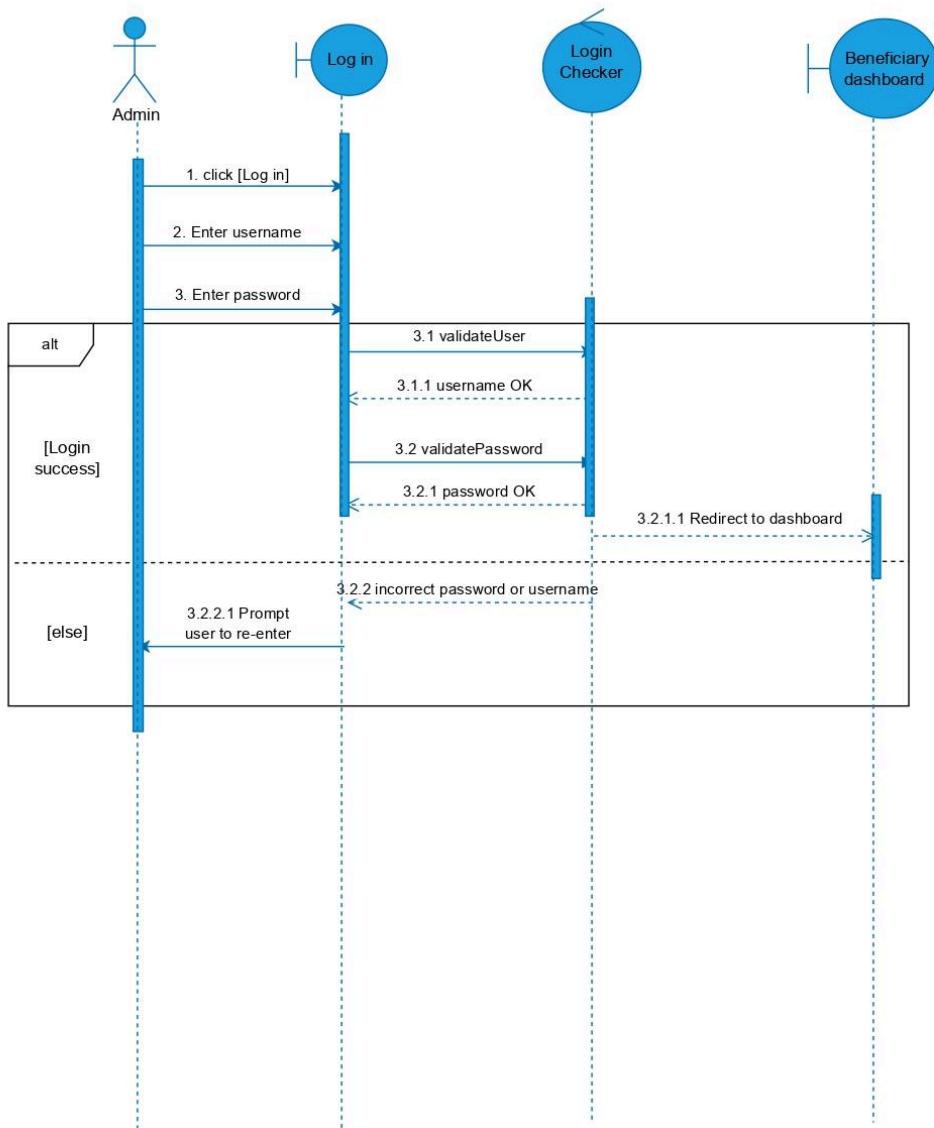


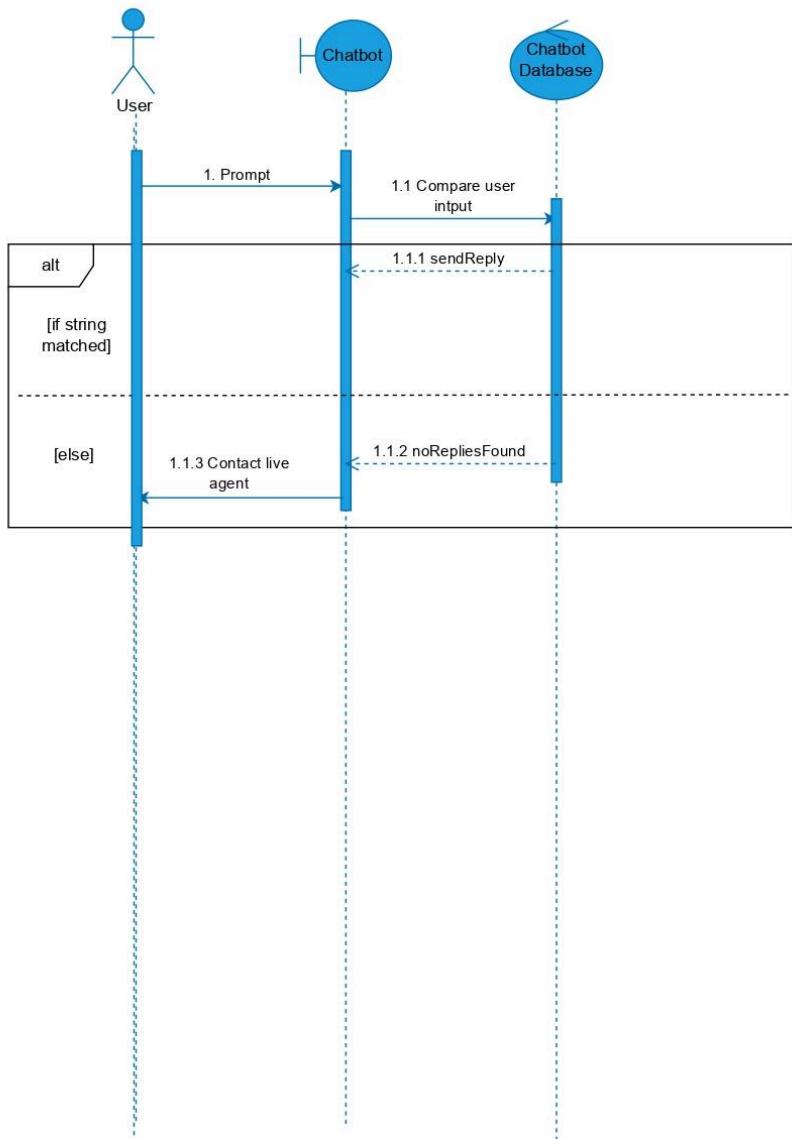


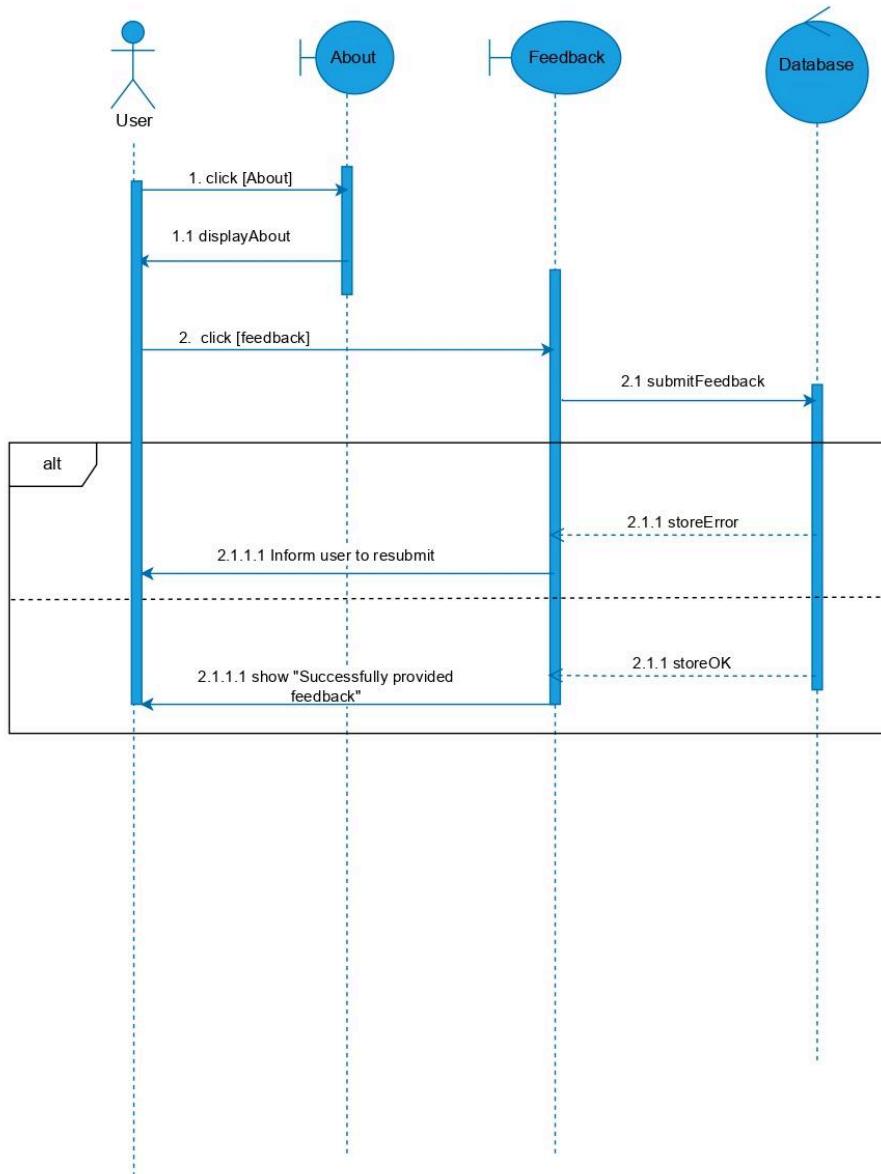




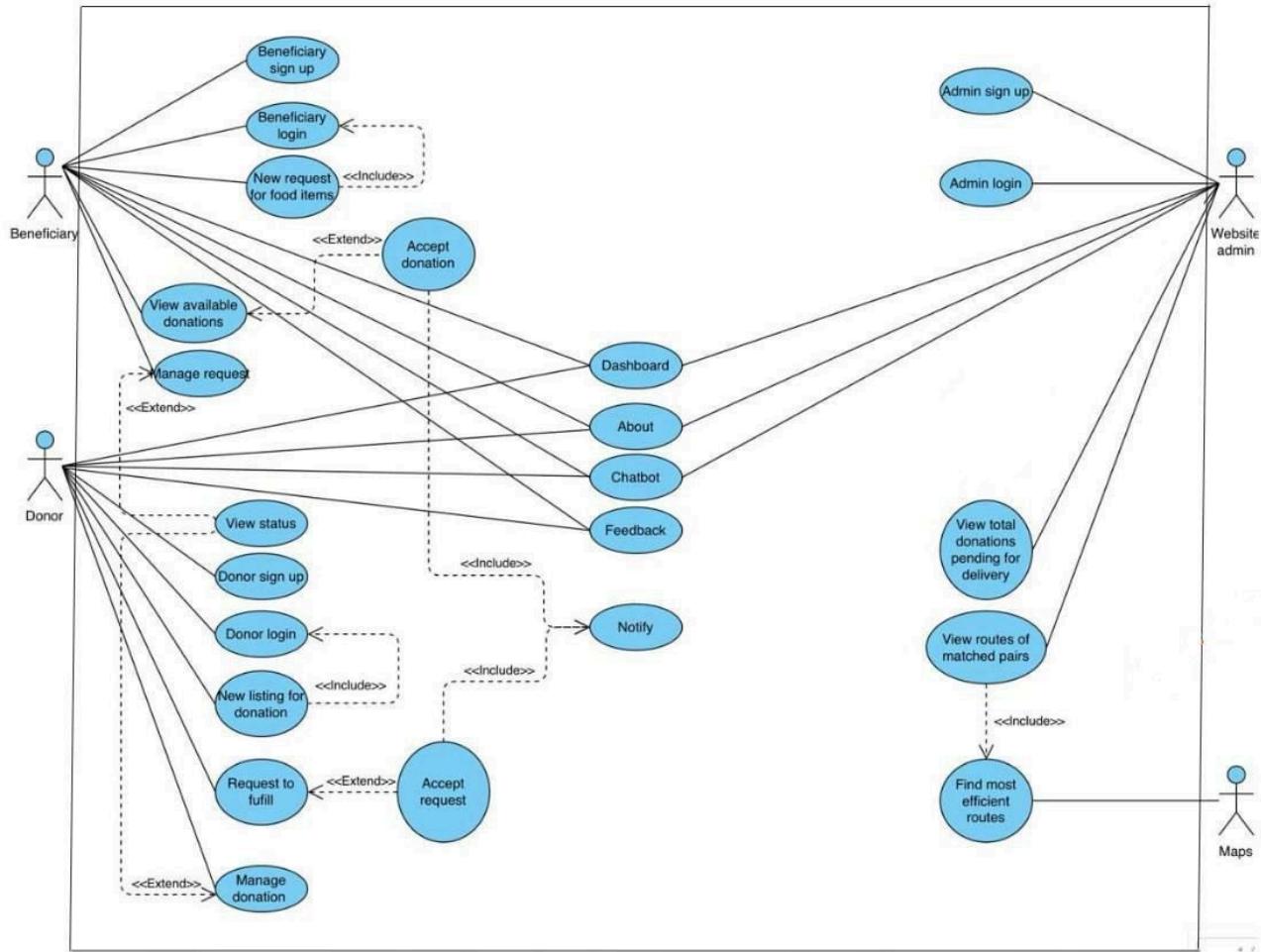








## 12. Use Case Diagram



## 13. Black Box Testing

### **AuthController**

- Manages user authentication such as user registration and user login

### **Equivalence Class and Boundary Value Testing**

#### **1. Sign Up Function**

##### **a. Donor Sign Up**

Valid Equivalence Classes: Organisation Name, Premise Address, UEN Number, Point of Contact Name, Point of Contact Phone Number, Email, Hygiene Rating, Password and Confirm Password with correct formats

Invalid Equivalence Classes: Organisation Name, Premise Address, UEN Number, Point of Contact Name, Point of Contact Phone Number, Hygiene Rating, Email, Password and Confirm Password with incorrect formats or empty mandatory fields

##### **b. Beneficiary Sign Up**

Valid Equivalence Classes: Agency Name, Point of Contact Name, Point of Contact Phone Number, Email, Password and Confirm Password with correct formats

Invalid Equivalence Classes: Agency Name, Point of Contact Name, Point of Contact Phone Number, Email, Password and Confirm Password with incorrect formats or empty mandatory fields

##### **c. Admin Sign Up**

Valid Equivalence Classes: Email, Password and Confirm Password with correct formats

Invalid Equivalence Classes: Email, Password and Confirm Password with incorrect formats or empty mandatory fields

## 2. Login Function

Valid Equivalence Classes: Email, Password and Confirm Password with correct formats

Invalid Equivalence Classes: Email, Password and Confirm Password with incorrect formats or empty mandatory fields

### Test Cases and Results

#### 1. Sign Up

##### a. Donor Sign Up

Test Case ID	Test Case Title	Test Input	Expected Output	Actual Output	Pass?

1	All valid inputs	<p>Organisation Name: “SHENG SHIONG SUPERMARKET”</p> <p>Premise Address: “845 YISHUN STREET 81 #01-184, S(760845)”</p> <p>UEN Number: “201023989Z”</p> <p>Point of Contact Name: “John”</p> <p>Point of Contact Phone Number: “80804968”</p> <p>Hygiene Rating: “A”</p> <p>Email: “<a href="mailto:lol@gmail.com">lol@gmail.com</a>”</p> <p>Password: “Lol123!”</p>	<p>System displays “Sign Up Success! Redirecting ...” and redirects user to Login Page</p>	<p>System displays “Sign Up Success! Redirecting ...” and redirects user to Login Page</p>	Yes
---	------------------	---	--	--	-----

		Confirm Password: “Lol123!”			
2	All valid inputs except Organisation Name	Organisation Name: “”	System displays “Donor Name cannot be empty”	System displays “Donor Name cannot be empty”	Yes

3	All valid inputs except Premise Address	Premise Address: “”	System displays “Premise Address cannot be empty”	System displays “Sign Up Success! Redirecting ...” and redirects user to Login Page	No
4	All valid inputs except UEN Number	UEN Number: “”	System displays “Sign Up Success! Redirecting ...” and redirects user to Login Page	System displays “Sign Up Success! Redirecting ...” and redirects user to Login Page	Yes
5	All valid inputs except Point of Contact Name	Point of Contact Name: “”	System displays “Point of Contact Name should not be empty”	System displays “Point of Contact Name should not be empty”	Yes
6	All valid inputs except Point of Contact Phone Number	Point of Contact Phone Number: “”	System displays “Phone number is required”	System displays “Phone number is required”	Yes
7	All valid	Hygiene Rating: “”	System	System did	No

	inputs except Hygiene Rating		displays “Hygiene Rating is required”	not display anything but sign up is unsuccessful	
8	All valid inputs except Email	Email: “”	System displays “Email is required”	System displays “Email is required”	Yes
9	All valid inputs except Password	Password: “”	System displays “Password cannot be empty”	System displays “Password cannot be empty”	Yes
10	All valid inputs except Confirm Password	Confirm Password: “”	System displays “Confirm Password cannot be empty”	System displays “Confirm Password cannot be empty”	Yes
11	All valid inputs except Confirm Password	Password: “Lol123!” Confirm Password: “Lol1233!”	System displays “Passwords don't match!”	System displays “Passwords don't match!”	Yes

b. Beneficiary Sign Up

Test Cas	Test Case Title	Test Input	Expecte d Output	Actua l	Pass?

e ID				Outp ut	
1	All valid inputs	Agency Name: "Test"  Point of Contact Name: "Kelly"  Point of Contact Phone Number: "80801234"  Email: <a href="mailto:test@gmail.com">"test@gmail.com"</a>  Password: "Test123!"  Confirm Password:  "Test123!"	System displays "Sign Up Success! Redirecting ..." and redirects user to Login Page	System displays "Sign Up Success! Redirecting ..." and redirects user to Login Page	Yes
2	All valid inputs except Agency Name	Agency Name: ""	System displays "Required"	System displays "Required"	Yes
3	All valid inputs except Point of Contact Name	Point of Contact Name: ""	System displays "Point of Contact Name should not be empty"	System displays "Point of Contact Name should not be empty"	Yes

4	All valid inputs except Phone number	Phone number: “”	System displays “Phone number is required”	System displays “Phone number is required”	Yes
5	All valid inputs except Email	Email: “”	System displays “Email is required”	System displays “Email is required”	Yes

6	All valid inputs except Password	Password: “”	System displays “Password cannot be empty”	System displays “Password cannot be empty”	Yes
7	All valid inputs except Confirm Password	Confirm Password: “”	System displays “Confirm Password cannot be empty”	System displays “Confirm Password cannot be empty”	Yes
8	All valid inputs except Confirm Password	Password: “Test123!” Confirm Password: “Test1233!”	System displays “Passwords don't match!”	System displays “Passwords don't match!”	Yes

c. Admin Sign Up

Test Case ID	Test Case Title	Test Input	Expected Output	Actual Output	Pass?
1	All valid inputs	Email: “admin@gmail.com”  Password: “Admin123!”  Confirm Password: “Admin123!”	System displays “Sign Up Success! Redirecting ...” and redirects user to Login Page	System displays “Sign Up Success! Redirecting ...” and redirects user to Login Page	Yes
5	All valid inputs except Email	Email: “”	System displays “Email is required”	System displays “Email is required”	Yes
6	All valid inputs except Password	Password: “”	System displays “Password cannot be empty”	System displays “Password cannot be empty”	Yes
7	All valid inputs except Confirm Password	Confirm Password: “”	System displays “Confirm Password cannot be empty”	System displays “Confirm Password cannot be empty”	Yes
8	All valid inputs except Confirm Password	Password: “Admin123!” Confirm Password: “Admin1233!”	System displays “Passwords don't match!”	System displays “Passwords don't match!”	Yes

## 2. Login

Test Case ID	Test Case Title	Test Input	Expected Output	Actual Output	Pass?
1	All valid inputs	Email: “admin@gmail.com” Password: “Admin123!”	System displays “Login Success! Redirecting ...” and redirects user to Admin’s Dashboard	System displays “Login Success! Redirecting ...” and redirects user to Admin’s Dashboard	Yes
5	Invalid email and valid password	Email: “” Password: “Admin123!”	System displays “Email is required”	System displays “Email is required”	Yes
6	Invalid email and valid password	Email: “testing” Password: “Admin123!”	System displays “Email is required”	System displays “Email is required”	Yes

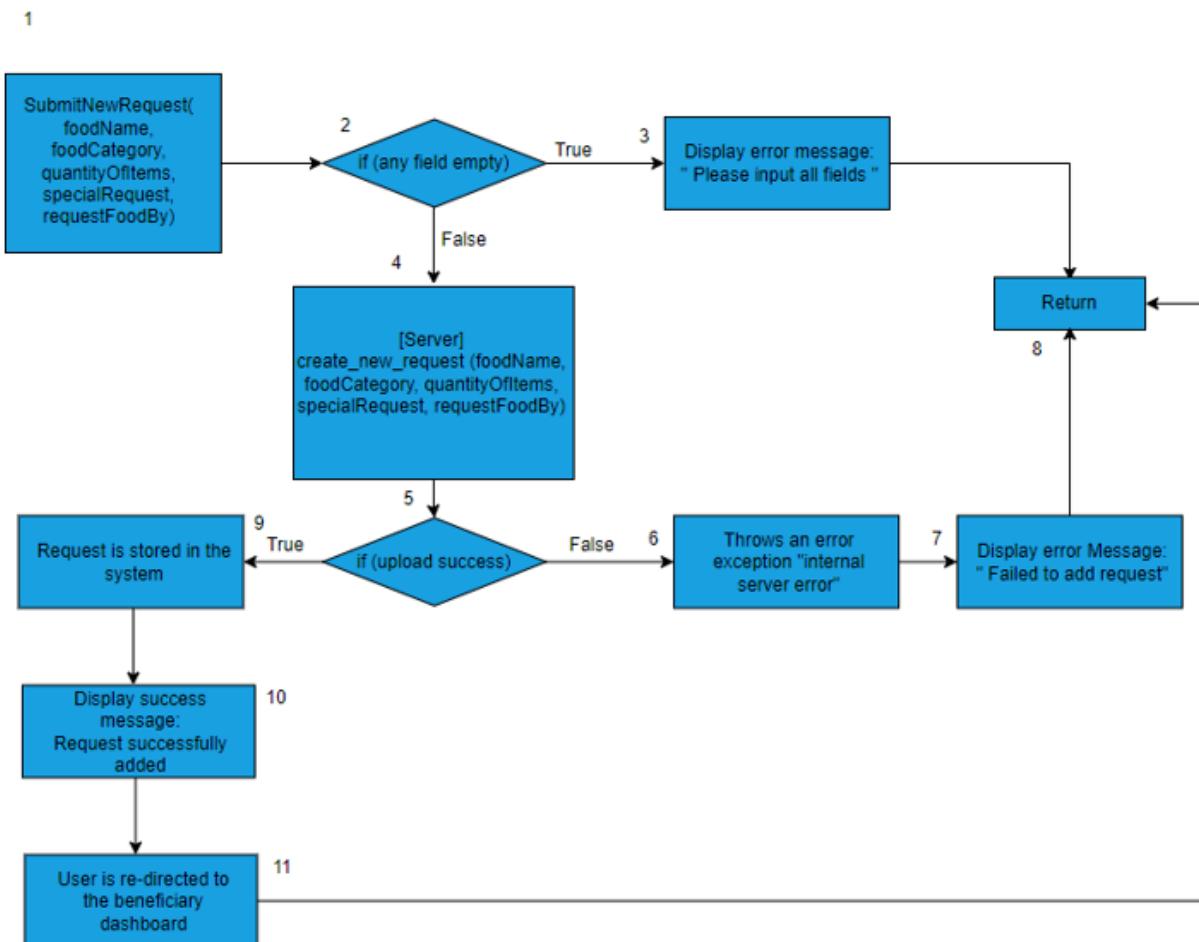
7	Valid email and Invalid password	Email: “admin@gmail.com”  Password: “”	System displays “The username/password you entered is incorrect. Please try again”	System displays “The username/password you entered is incorrect. Please try again”	Yes
8	Valid email and Invalid password	Email: “admin@gmail.com”  Password: “wrongpass”	System displays “The username/password you entered is	System displays “The username/password you entered is incorrect. Please	Yes
			incorrect. Please try again”	try again ”	

## 14. White Box Testing

The following features will be tested with the white box testing method:

- 1) Submit new request as beneficiary

- i) Control flow graph



## ii) Basic Path Testing

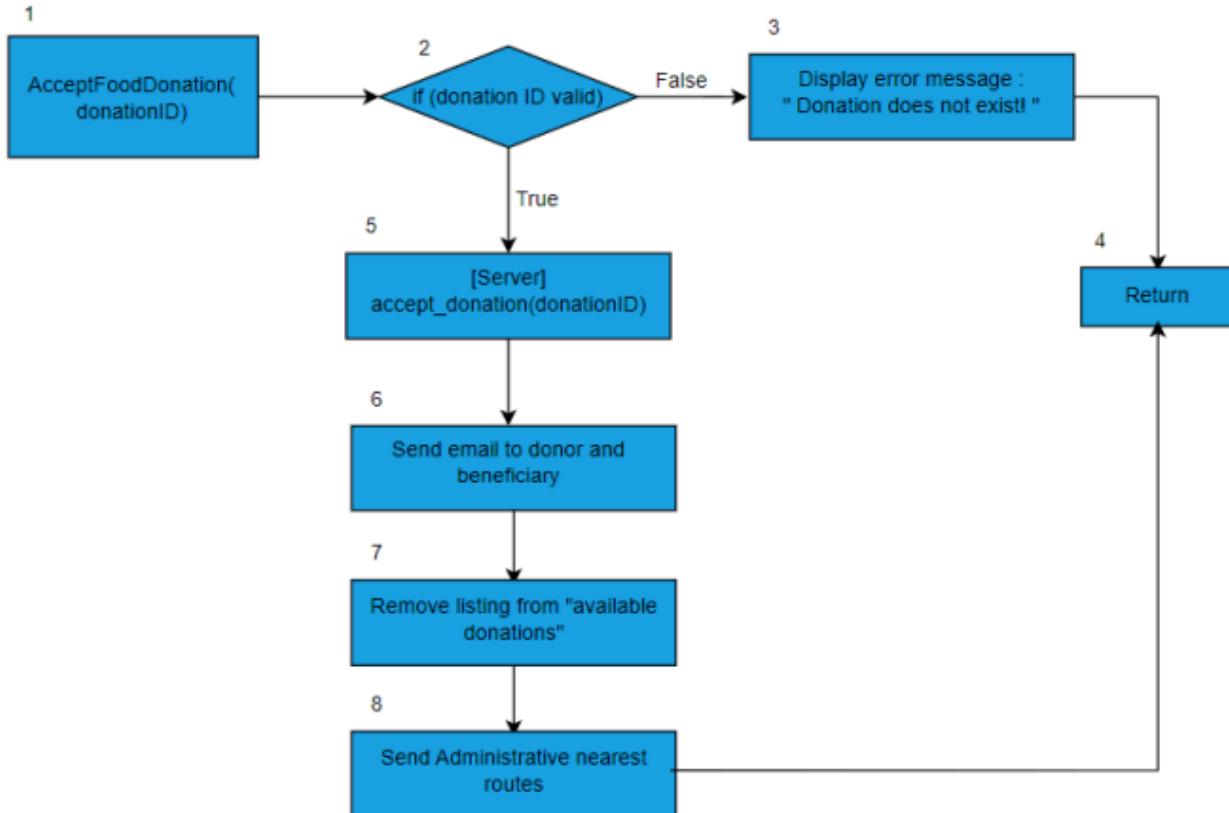
Cyclomatic complexity : | Decision points | + 1 = 2 + 1 = 3

### Basic paths:

- 1) Baseline path: 1,2,4,5,9,10,11
- 2) Basis path 2: 1,2,3,8
- 3) Basis path 3: 1,2,4,5,6,7,8

### iii)Test Cases And Results

No.	Test Input	Expected Output	Actual Output	Pass?
1	<b>All fields empty</b> (foodName, servings, specialRequest, date, deliveryMethod, etc.)	Error: "Please input all fields"	Please input all fields	Yes
2	<b>Some fields empty</b> (e.g., servings, deliveryMethod)	Error: "Please input all fields"	Please input all fields	Yes
3	<b>All fields filled, valid data</b> (e.g., valid foodName, servings, etc.)	Success: Request successfully added, redirect to dashboard	Request successfully added, redirect to dashboard	Yes
4	<b>All fields filled, but illegal fields</b> (e.g., invalid characters in foodName)	Error: "Internal server error"	Internal server error	Yes
5	<b>Invalid date format</b> (e.g., date in wrong format)	Error: "Invalid date format"	Invalid date format	Yes
6	<b>Server error during request creation</b> (simulating server failure)	Error: "Failed to add request"	Failed to add request	Yes

**2) Accepting new donations from donor as beneficiary**i) Control Flow Graph

ii) Basic Paths Testing:

Cyclomatic complexity: | Decision points | + 1 = 2

**Basic paths:**

- 1) Baseline path: 1,2,5,6,7,8,4
- 2) Basis path: 1,2,3,4

iii) Test case and Results

No.	Test Input	Expected Output	Actual Output	Pass?
1	Valid donation ID	Success: Donation accepted successfully.	Donation accepted successfully	Yes
2	Invalid donation ID	HTTP 400 exception thrown with message “Donation does not exist”.  Display error message: “Failed to add accept donation”	Failed to add accept donation	Yes

## 15. API Documentation

The following are documentations for the SC2006 project for group TDDB-47: FoodHero API.

### 1.adminDetails

GET	/app/api/adminDetails	get admin details based on current logged in admin email
-----	-----------------------	--

### 2.adminList

POST	/app/api/adminList	add approved admins email
GET	/app/api/adminList	get list of approved admins
DELETE	/app/api/adminList	delete approved admin account

### 3. beneficiaryDetails

POST	/app/api/beneficiaryDetails	update current logged in beneficiary profile
GET	/app/api/beneficiaryDetails	get current logged in beneficiary profile
DELETE	/app/api/beneficiaryDetails	delete current logged in beneficiary profile

### 4. donation

POST	/app/api/donation	add new donation
GET	/app/api/donation	get donation based on donation id
PUT	/app/api/donation	update donation based on donation id

### 5. donations

GET	/app/api/donations	get all donations form based on current logged in donor email
DELETE	/app/api/donations	delete donation by ID

## 6. donorDetails

POST	/app/api/donorDetails	update current logged in donor profile
GET	/app/api/donorDetails	get current logged in donor profile
DELETE	/app/api/donorDetails	delete current logged in donor profile

## 7. feedback

POST	/app/api/feedback	add new feedback form
------	-------------------	-----------------------

## 8. getAcceptedDeliveries

GET	/app/api/getAcceptedDeliveries	get list of accepted deliveries for current logged in admin
-----	--------------------------------	---

## 9. request

POST	/app/api/request	add new request
GET	/app/api/request	get request based on request id
PUT	/app/api/request	update request based on request id

## 10. requests

GET	/app/api/requests	get all requests based on current beneficiary email
DELETE	/app/api/requests	delete request by ID

## 11. statusUpdate

PUT	/app/api/statusUpdate	update status of delivery
-----	-----------------------	---------------------------

## 12. updateBeneficiary

PUT	/app/api/updateBeneficiary	update beneficiary email field of the listed donation based on the donor id
-----	----------------------------	---

## 13. updateDonor

PUT	/app/api/updateDonor	update donor email field of the listed request based on the request id
-----	----------------------	--

## 14. auth

GET	/app/api/auth	Retrieve session data
POST	/app/api/auth	Sign in, sign out

## 15. chatbot

POST	/app/api/chatbot	Process user messages
------	------------------	-----------------------

## **16. Dialog Map**

