

---

# **Software Requirements Specification**

**for**

# **FoodHero**

**Version 1.0 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>1</b>
<b>1.3 Intended Audience and Reading Suggestions</b>	<b>2</b>
<b>1.4 Product Scope</b>	<b>2</b>
<b>1.5. References</b>	<b>3</b>
<b>2. Overall Description</b>	<b>3</b>
2.1 Product Perspective	3
<b>2.2 Product Functions</b>	<b>4</b>
<b>2.3 User Classes and Characteristics</b>	<b>5</b>
<b>2.4 Operating Environment</b>	<b>5</b>
<b>2.5 Design and Implementation Constraints</b>	<b>6</b>
<b>2.6 User Documentation</b>	<b>6</b>
<b>2.7 Assumptions and Dependencies</b>	<b>7</b>
<b>3. External Interface Requirements</b>	<b>8</b>
3.1 User Interfaces	8
<b>3.2 Hardware Interfaces</b>	<b>26</b>
<b>3.3 Software Interfaces</b>	<b>26</b>
<b>3.4 Communications Interfaces</b>	<b>27</b>
<b>4. Functional Requirements (Lecture Format)</b>	<b>28</b>
<b>5. Functional Requirements (SRS Format)</b>	<b>34</b>
5.1 Main Menu	34
5.2 User Registration	35
5.3 Login	35
5.4 Submissions	36
5.5 Dashboard	37
5.6 Donation Match Confirmation System	39
Route Optimization	41
5.8 Notifications and Alerts	41
5.9 Chatbot Functionality	42
5.10 Interaction with Other Systems	43
<b>7. Other Requirements</b>	<b>46</b>
<b>8. Data Dictionary</b>	<b>47</b>

## **Revision History**

<b>Name</b>	<b>Date</b>	<b>Reason For Changes</b>	<b>Version</b>

# 1. Introduction

## 1.1 Purpose

*The Food Hero app is a CRM web application designed to bridge the gap between food donors and beneficiaries in Singapore. It facilitates the efficient matching of food donations with suitable beneficiaries, ensuring optimal distribution and minimizing food wastage. This document outlines the Software Requirements Specification (SRS) for Food Hero app, version 1.0, detailing the features and functionalities planned for the initial release.*

*The SRS specifies the software requirements for the entire system, encompassing both the backend and frontend components. Key features include:*

- **Location-based filtering:** Leveraging datasets of supermarkets and hawker centres to allow users to filter by region for more accurate and efficient food donation matching.
- **Unified login/signup page:** A single interface for both donors and beneficiaries, simplifying access to the system.
- **Dashboard functionality:** Both donors and beneficiaries will have access to a personalized dashboard to view existing requests and donations.
- **Donation application and beneficiary request form:** Forms that facilitate easy submission and management of donations and requests.

*This document serves as a comprehensive guide for developers, testers, and other stakeholders involved in the Food Hero app project. It outlines the scope, system functionalities, and user interface requirements, ensuring all project participants are aligned with the technical and operational goals of version 1.0.*

## 1.2 Document Conventions

*This document follows specific conventions and standards to ensure clarity and consistency throughout. The following typographical conventions are used to highlight important elements:*

- **Bold Text:** Used to emphasize titles, headings, and key terms.
- *Italic Text:* Used for examples and to highlight special notes or clarifications.
- *Numbered Lists:* Used to indicate the sequence of steps in processes or procedures.
- *Bullet Points:* Used for itemizing general points, requirements, or features.

### 1.3 Intended Audience and Reading Suggestions

*This document is intended for various stakeholders involved in the development and evaluation of the Food Hero app as part of our project:*

- **Developers:** *To understand the functional and non-functional requirements for implementing the app.*
- **Users:** *To gain insights into the app's functionality and user interface.*
- **Testers:** *To design and execute test cases based on the specified requirements.*
- **Documentation Writers:** *To prepare user manuals and help guides.*
- **Teaching Assistant (TA):** *To evaluate the app's development process and its adherence to the specified requirements.*

*The document is organized into sections starting with an overview, followed by detailed descriptions of functional and non-functional requirements. It is recommended that readers begin with the Introduction and Purpose sections, and then proceed to the sections most relevant to their role.*

### 1.4 Product Scope

*The Food Hero app is designed to streamline food donation efforts in Singapore by connecting donors with beneficiaries in need, ensuring timely and efficient distribution. With features like location-based filtering and real-time notifications, the app enhances the process by enabling donors to find nearby beneficiaries, minimizing food wastage, and ensuring donations are utilized while still fresh. The app integrates government datasets for automated location detection, providing users with accurate matches based on proximity. Additionally, a unified login/signup page and an interactive user interface create a seamless user experience for both donors and beneficiaries. Aligned with broader project objectives to address food insecurity and promote community welfare, Food Hero supports sustainability efforts and contributes to social responsibility initiatives, helping foster stronger community connections and making a significant impact on food distribution efficiency.*

### 1.5. References

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

## 2. Overall Description

### 2.1 Product Perspective

The **Food Hero** app is a new, self-contained product designed to facilitate food donations in Singapore. It operates independently, without reliance on any previous systems or applications. The app's primary purpose is to connect food donors with beneficiaries, leveraging location-based filtering and real-time notifications to ensure efficient food distribution. The app is built with a modular architecture, allowing for easy integration with potential future systems that may expand its functionality.

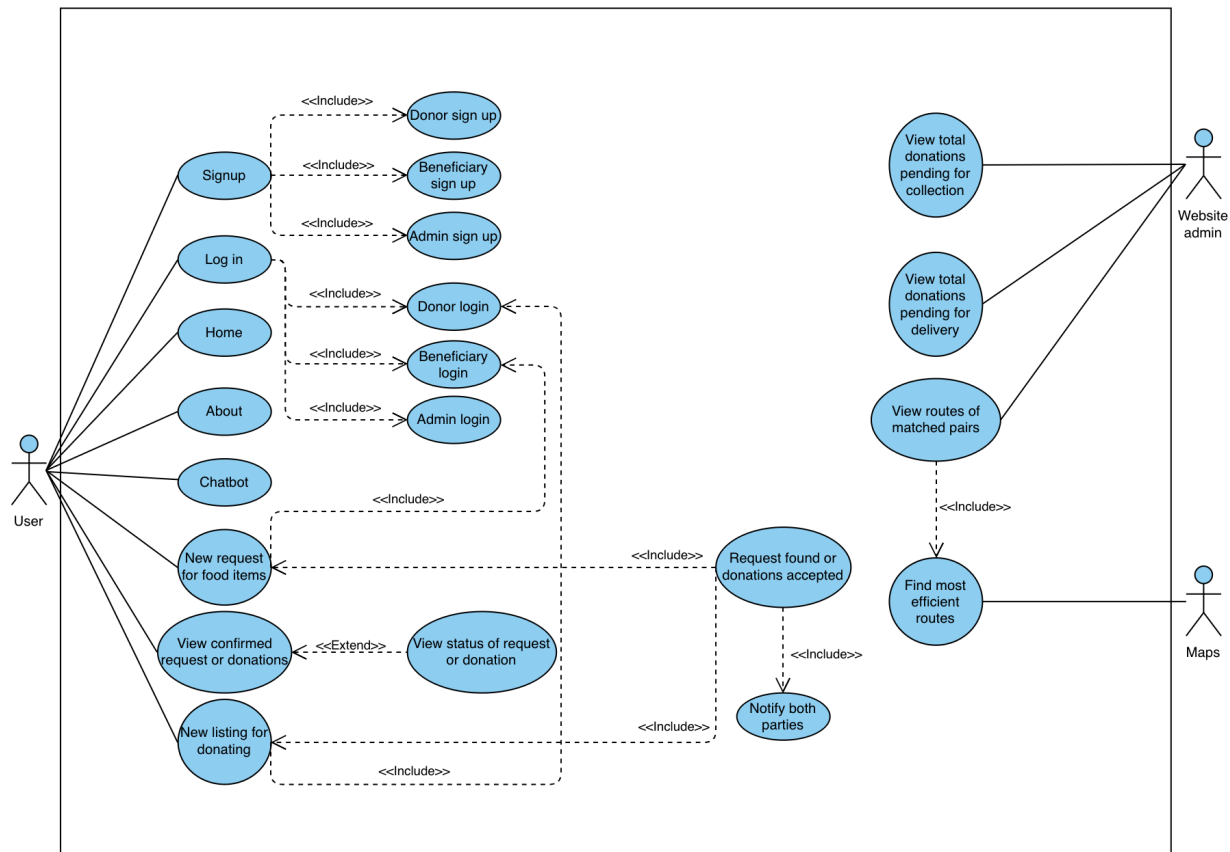
Major components include:

- **Frontend Interface:** A user-facing application is built using Next.js, where donors and beneficiaries interact with the system.
- **Backend System:** Manages data storage and processing, utilizing MongoDB for the database.
- **Location Services:** Integrates with external datasets to provide location-based filtering.
- **Notification System:** Utilizes AWS SNS for sending real-time notifications to users.
- **Chatbot:** A chatbot powered by the Gemini API, designed to assist users with any inquiries related to food donation processes, beneficiary matching, and navigating the app's features.

### 2.2 Product Functions

The **Food Hero** app provides the following major functions:

- **User Registration and Authentication:** Allows donors and beneficiaries to create accounts and securely log in.
- **Donation Application:** Enables donors to submit details of food donations, including type, quantity, location and photographic evidence of the food.
- **Beneficiary Request Form:** Allows beneficiaries to submit requests for food donations.
- **Real-Time Notifications:** Sends alerts to donors and beneficiaries about successful matches and updates using AWS SNS.
- **Location-Based Filtering:** Uses geographical data to match donors with nearby beneficiaries.
- **Chatbot Assistance:** Provides real-time support to users through a chatbot powered by the Gemini API, offering guidance on the donation process and providing detailed information on how to utilize the app's various features effectively.



## 2.3 User Classes and Characteristics

The **Food Hero** app is designed for the following user classes:

- **Donors:** Individuals or organizations donating food. They are expected to have basic technical proficiency and will primarily use the donation application/dashboard function.
- **Beneficiaries:** Individuals or organizations in need of food donations. They may have varying levels of technical expertise and will use the request form/dashboard, receive notifications, and interact with the chatbot.
- **Administrators:** Users with higher privilege levels who manage the system, oversee matches, and ensure the smooth operation of the app.

## 2.4 Operating Environment

The **Food Hero** app is designed to operate in the following environment:

- **Hardware:** The app is expected to run on any device capable of running a modern web browser, including desktops, laptops, tablets, and smartphones.
- **Operating System:** The app will be compatible with major operating systems, including Windows, macOS, Linux, Android, and iOS.
- **Software Dependencies:** The app will run on a web server with a backend database using MongoDB and use standard web technologies like Next.js for the frontend. AWS SNS will be used for notification services, and the Gemini API will power the chatbot.

## 2.5 Design and Implementation Constraints

The following constraints apply to our project:

- **Technology Stack:** The app will be developed using Next.js for the frontend and backend, MongoDB for the database, AWS SNS for real-time notifications, and Gemini API for the chatbot.
- **Security Requirements:** The app must comply with standard security practices, including secure authentication and data encryption.
- **Regulatory Compliance:** The app must adhere to relevant regulations regarding data protection and food safety.
- **Performance Requirements:** The app must handle concurrent users efficiently, ensuring a responsive user experience.

## 2.6 User Documentation

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>

The **Food Hero** app is designed to be user-friendly and intuitive, minimizing the need for extensive documentation. Additionally, a chatbot is integrated into the app to assist users in navigating its features and provide real-time support, making it even easier to understand and utilize the app's capabilities. All documentation will be available in digital format, accessible directly through the app's interface.



## 2.7 Assumptions and Dependencies

*The following assumptions and dependencies are considered for our project:*

- **Third-Party Services:** *The app will depend on third-party services like AWS SNS for notifications, location-based filtering, and the Gemini API for the chatbot. It assumes these services will be available and functional.*
- **Internet Connectivity:** *The app assumes that users have access to a stable internet connection.*
- **Data Integrity:** *It is assumed that the data provided by users and external datasets is accurate and reliable.*
- **Scalability:** *The app assumes that the current infrastructure can be scaled to accommodate increasing numbers of users and transactions.*

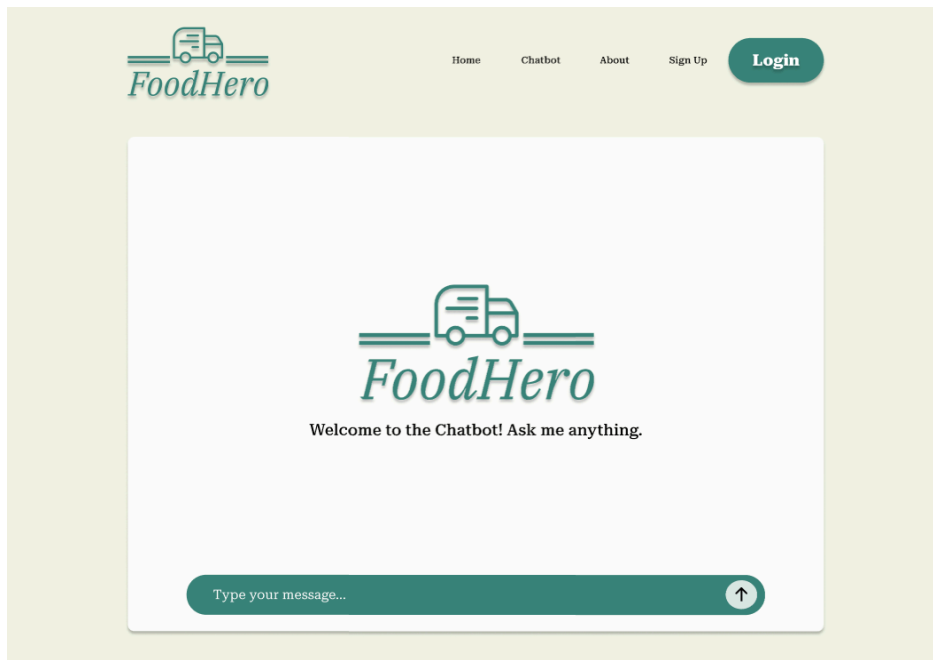
## 3. External Interface Requirements

### 3.1 User Interfaces

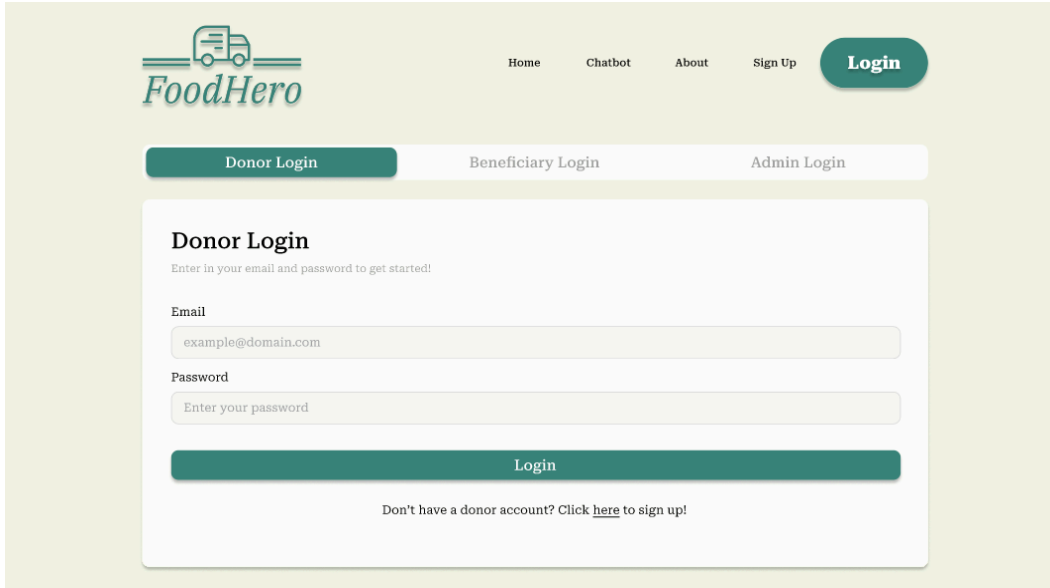
#### A. Home Page



#### B. Chatbot



### C. Login Page



The screenshot displays the FoodHero website's login interface. At the top left is the FoodHero logo, featuring a green truck icon and the brand name. To the right is a navigation menu with links for Home, Chatbot, About, Sign Up, and a prominent green Login button. Below the navigation is a horizontal bar with three tabs: Donor Login (active), Beneficiary Login, and Admin Login. The Donor Login section contains a title, a prompt to enter email and password, input fields for email (with a placeholder example) and password, a Login button, and a link for users without an account.

**FoodHero**

Home Chatbot About Sign Up **Login**

Donor Login Beneficiary Login Admin Login

**Donor Login**

Enter in your email and password to get started!

Email

example@domain.com

Password

Enter your password

**Login**

Don't have a donor account? Click [here](#) to sign up!

Donor Login

Beneficiary Login

Admin Login

### Beneficiary Login

Enter in your email and password to get started!

Email

Password

Login

Don't have a beneficiary account? Click [here](#) to sign up!

Donor Login

Beneficiary Login

Admin Login

### Admin Login

Enter in your Administrator credentials to get started!

Email

Password

Login

Don't have an admin account? Click [here](#) to sign up!

D. Sign Up Page  
a. Donor Sign-Up Page (Supermarket)

The screenshot displays the 'Donor Sign Up' page for FoodHero. The page has a light beige background. At the top, there is a navigation bar with the FoodHero logo (a green truck icon and the text 'FoodHero') on the left, and links for 'Home', 'Chatbot', 'About', 'Sign Up', and a green 'Login' button on the right. Below the navigation bar, there are three tabs: 'Donor Sign Up' (active, highlighted in green), 'Beneficiary Sign Up', and 'Admin Sign Up'. The main content area is titled 'Donor Sign Up' and includes a sub-header 'Fill in your personal details and some information about you to get started!'. The form is divided into several sections: 1. 'Select User Type\*' with radio buttons for 'Supermarket' (selected) and 'Individual'. 2. 'Details' section with fields for 'Organisation Name\*' (dropdown), 'Address\*' (dropdown), 'UEN Number' (text input with placeholder 'Enter your UEN number (if applicable)'), 'Point of Contact Name\*' (text input with placeholder 'Enter your full name'), and 'Point of Contact Phone Number\*' (text input with placeholder 'Enter your phone number'). 3. 'Certification Status' section with a checkbox for 'Select if your organisation is Halal certified.' and a note: 'If you declare that your organisation is Halal certified, random checks may be done to ensure that your organisation is Halal certified.' Below this is a dropdown for 'Select your organisation's hygiene rating\*'. 4. 'Account Details' section with fields for 'Email\*' (text input with placeholder 'example@domain.com'), 'Password\*' (text input with placeholder 'Enter your password'), and 'Confirm Password\*' (text input with placeholder 'Confirm your password'). At the bottom of the form is a large green 'Sign Up' button. Below the button, there is a link: 'Already have an account? Click [here](#) to login!'.

**FoodHero**

Home Chatbot About Sign Up **Login**

Donor Sign Up Beneficiary Sign Up Admin Sign Up

### Donor Sign Up

Fill in your personal details and some information about you to get started!

**Select User Type\***

☒ Supermarket ☐ Individual

**Details**

**Organisation Name\***

Select your organisation

**Address\***

Select your address

**UEN Number**

Enter your UEN number (if applicable)

**Point of Contact Name\***

Enter your full name

**Point of Contact Phone Number\***

Enter your phone number

**Certification Status**

Select if your organisation is Halal certified.

If you declare that your organisation is Halal certified, random checks may be done to ensure that your organisation is Halal certified.

☐

**Select your organisation's hygiene rating\***

Select a hygiene rating

**Account Details**

**Email\***

example@domain.com

**Password\***

Enter your password

**Confirm Password\***

Confirm your password

**Sign Up**

Already have an account? Click [here](#) to login!

- b. Donor Sign-Up Page (Supermarket) (Organisation Name + hygiene rating field dropdown)

The screenshot displays the 'Donor Sign Up' page for a supermarket. The page has a header with three tabs: 'Donor Sign Up' (active), 'Beneficiary Sign Up', and 'Admin Sign Up'. Below the header, the page title 'Donor Sign Up' is followed by a subtitle 'Fill in your personal details and some information about you to get started!'. The 'Select User Type\*' section shows 'Supermarket' selected with a radio button. The 'Details' section includes an 'Organisation Name\*' dropdown menu with a list of options: 'Fairprice', 'Giant', 'Sheng Siong', 'Cold Storage', and 'Hao Mart'. Below this is a text field for 'Enter your UEN number (if applicable)'. The 'Point of Contact Name\*' section has a text field for 'Enter your full name'. The 'Point of Contact Phone Number\*' section has a text field for 'Enter your phone number'. The 'Certification Status' section includes a checkbox for 'Select if your organisation is Halal certified.' and a note: 'If you declare that your organisation is Halal certified, random checks may be done to ensure that your organisation is Halal certified.' Below this is a 'Select your organisation's hygiene rating\*' dropdown menu with options 'A', 'B', 'C', and 'D'. The 'Password\*' section has a text field for 'Enter your password'. The 'Confirm Password\*' section has a text field for 'Confirm your password'. At the bottom, there is a green 'Sign Up' button and a link: 'Already have an account? Click [here](#) to login!'.

- c. Donor Sign-Up Page (Supermarket) (Address field dropdown)

The screenshot displays the 'Donor Sign Up' page for a supermarket, focusing on the address field. The page has a header with three tabs: 'Donor Sign Up' (active), 'Beneficiary Sign Up', and 'Admin Sign Up'. Below the header, the page title 'Donor Sign Up' is followed by a subtitle 'Fill in your personal details and some information about you to get started!'. The 'Select User Type\*' section shows 'Supermarket' selected with a radio button. The 'Details' section includes an 'Organisation Name\*' dropdown menu with the option 'Fairprice' selected. Below this is an 'Address\*' dropdown menu with a list of options: '1 JELEBU ROAD #02-20, #01-15, BUKIT PANJANG PLAZA S(677743)', '30 SEMBAWANG DRIVE #B1-01-03, SUN PLAZA S(757713)', '71 WOODLANDS AVENUE 3 #01-01, MARSILING MRT STATION S(739044)', '888 WOODLANDS DRIVE 50 #01-757, 888 PLAZA S(730888)', and '414 YISHUN RING ROAD #01-1853, S(760414)'. Below the address dropdown is a text field for 'Enter your full name'.

## d. Donor Sign Up Page (Individual)

The screenshot displays the 'Donor Sign Up' page for FoodHero. The page has a light beige background. At the top left is the FoodHero logo, which includes a green truck icon and the text 'FoodHero'. To the right of the logo are navigation links: 'Home', 'Chatbot', 'About', 'Sign Up', and a green 'Login' button. Below the navigation bar is a horizontal menu with three tabs: 'Donor Sign Up' (which is active and highlighted in green), 'Beneficiary Sign Up', and 'Admin Sign Up'. The main content area is a white box with the title 'Donor Sign Up' and a subtitle 'Fill in your personal details and some information about you to get started!'. The form consists of several sections: 'Select User Type\*' with radio buttons for 'Supermarket' and 'Individual' (the latter is selected); 'Contact Information' with fields for 'Agency Name' (placeholder: 'Enter your agency name (if applicable)'), 'Point of Contact Name\*' (placeholder: 'Enter your full name'), and 'Point of Contact Phone Number\*' (placeholder: 'Enter your phone number'); and 'Account Details' with fields for 'Email\*' (placeholder: 'example@domain.com'), 'Password\*' (placeholder: 'Enter your password'), and 'Confirm Password\*' (placeholder: 'Confirm your password'). At the bottom of the form is a large green 'Sign Up' button. Below the button is a link: 'Already have an account? Click [here](#) to login!'.

**FoodHero**

Home Chatbot About Sign Up **Login**

Donor Sign Up Beneficiary Sign Up Admin Sign Up

### Donor Sign Up

Fill in your personal details and some information about you to get started!

Select User Type\*

☐ Supermarket ☒ Individual

**Contact Information**

Agency Name

Enter your agency name (if applicable)

Point of Contact Name\*

Enter your full name

Point of Contact Phone Number\*

Enter your phone number

**Account Details**

Email\*

example@domain.com

Password\*

Enter your password

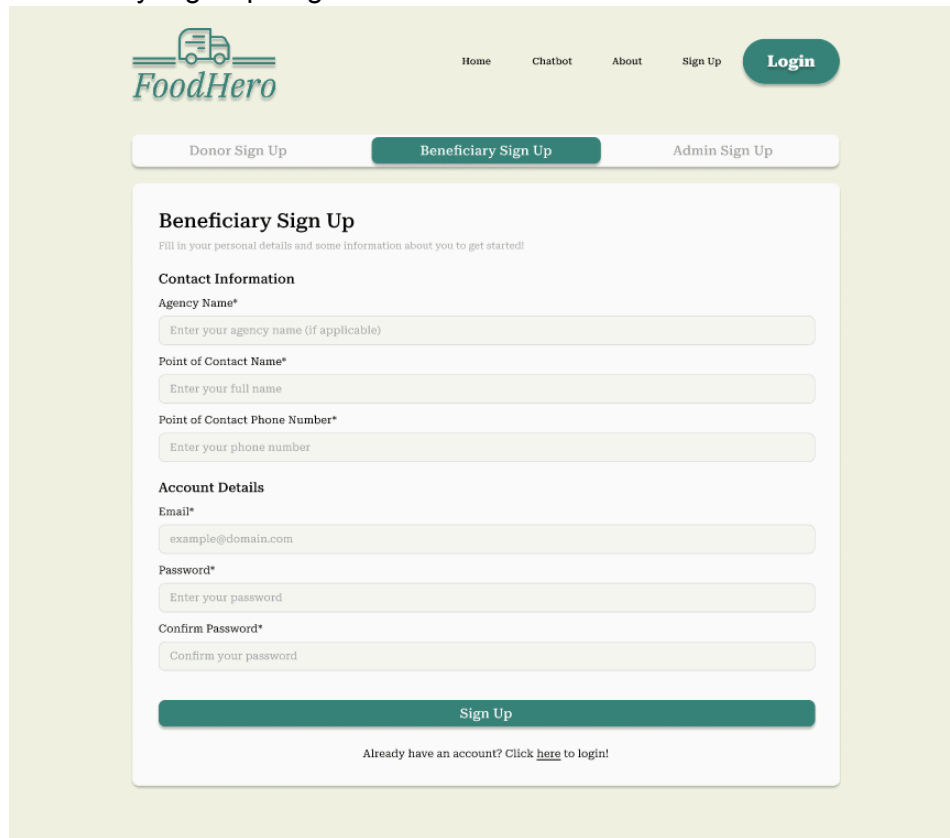
Confirm Password\*

Confirm your password

**Sign Up**

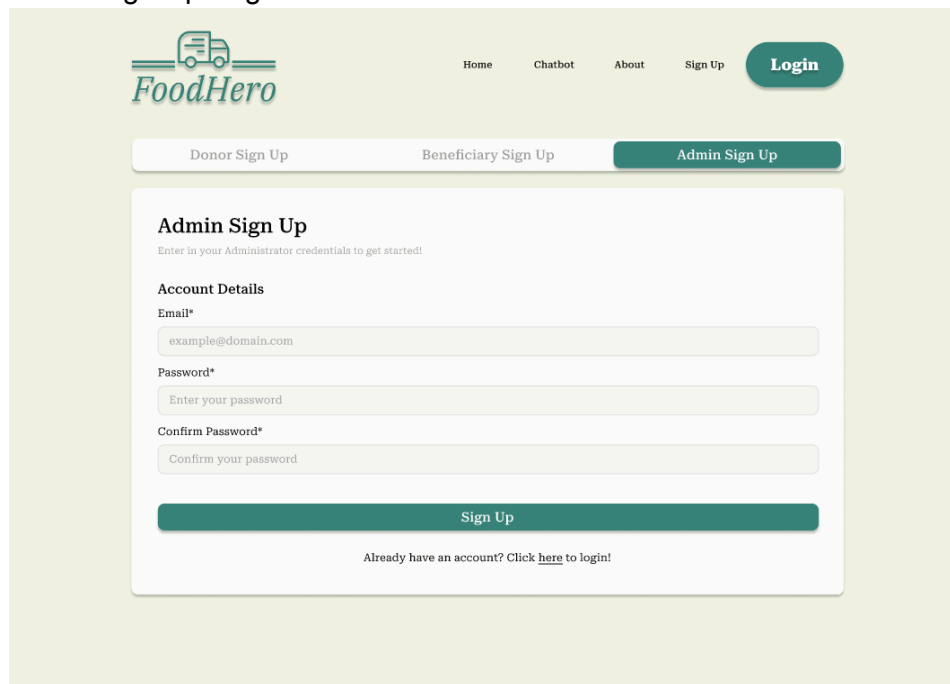
Already have an account? Click [here](#) to login!

## e. Beneficiary Sign Up Page



The Beneficiary Sign Up page features a light beige background. At the top left is the FoodHero logo, which includes a green truck icon and the text "FoodHero". To the right of the logo is a navigation bar with links for "Home", "Chatbot", "About", "Sign Up", and a prominent green "Login" button. Below the navigation bar is a horizontal menu with three tabs: "Donor Sign Up", "Beneficiary Sign Up" (which is highlighted in green), and "Admin Sign Up". The main content area is a white card with the title "Beneficiary Sign Up" and a subtitle "Fill in your personal details and some information about you to get started!". The form is divided into two sections: "Contact Information" and "Account Details". The "Contact Information" section includes three input fields: "Agency Name\*" with placeholder text "Enter your agency name (if applicable)", "Point of Contact Name\*" with placeholder text "Enter your full name", and "Point of Contact Phone Number\*" with placeholder text "Enter your phone number". The "Account Details" section includes three input fields: "Email\*" with placeholder text "example@domain.com", "Password\*" with placeholder text "Enter your password", and "Confirm Password\*" with placeholder text "Confirm your password". At the bottom of the form is a large green "Sign Up" button. Below the button is a link that says "Already have an account? Click [here](#) to login!".

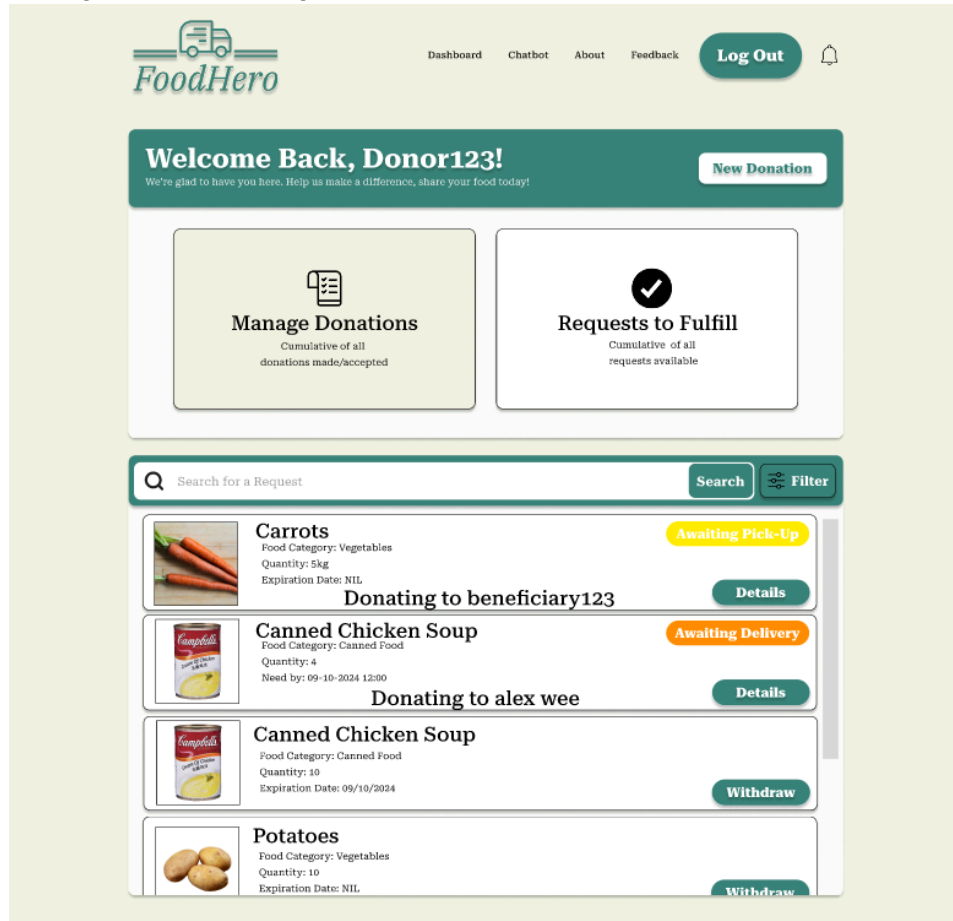
## f. Admin Sign Up Page



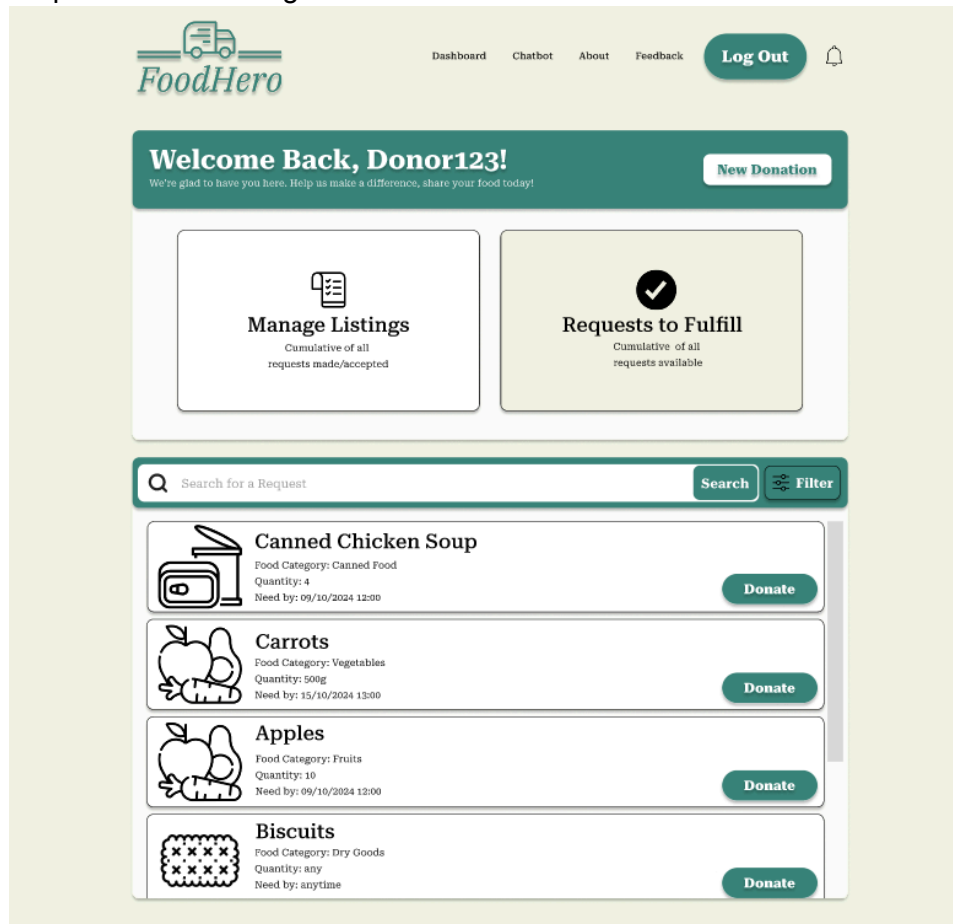
The Admin Sign Up page features a light beige background. At the top left is the FoodHero logo, which includes a green truck icon and the text "FoodHero". To the right of the logo is a navigation bar with links for "Home", "Chatbot", "About", "Sign Up", and a prominent green "Login" button. Below the navigation bar is a horizontal menu with three tabs: "Donor Sign Up", "Beneficiary Sign Up", and "Admin Sign Up" (which is highlighted in green). The main content area is a white card with the title "Admin Sign Up" and a subtitle "Enter in your Administrator credentials to get started!". The form is divided into one section: "Account Details". This section includes three input fields: "Email\*" with placeholder text "example@domain.com", "Password\*" with placeholder text "Enter your password", and "Confirm Password\*" with placeholder text "Confirm your password". At the bottom of the form is a large green "Sign Up" button. Below the button is a link that says "Already have an account? Click [here](#) to login!".



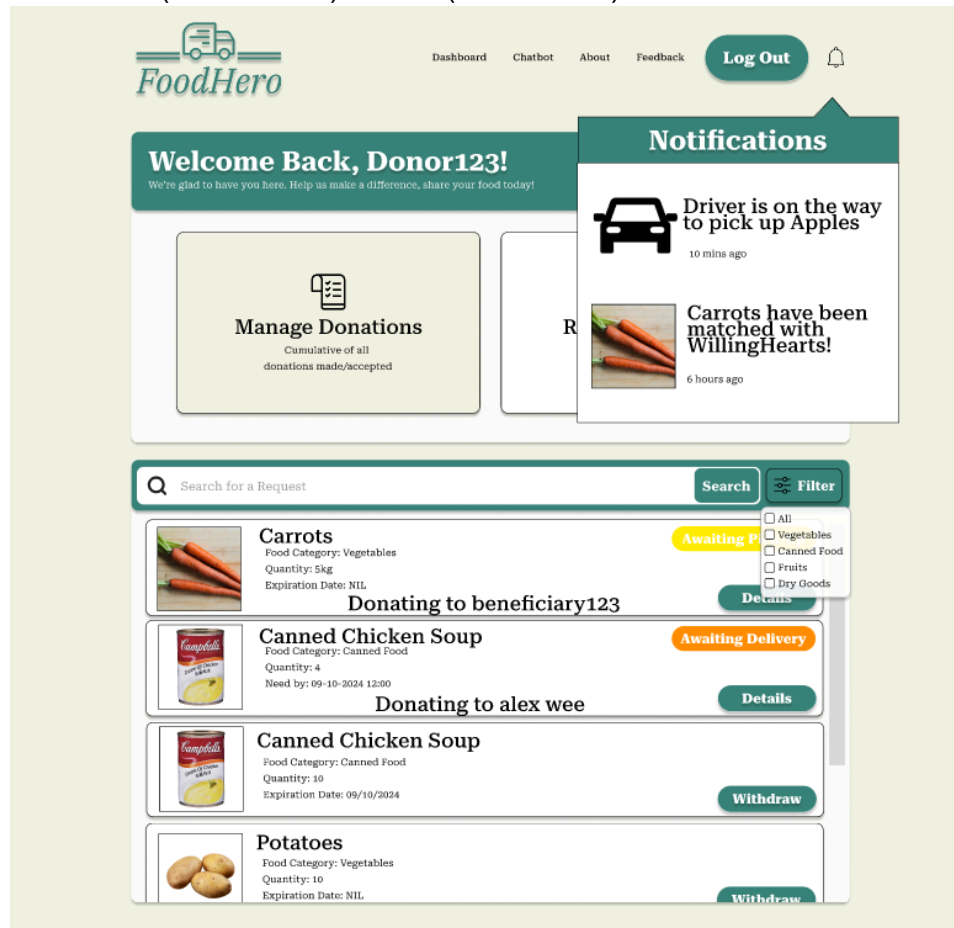
E. Donor Dashboard  
a. Manage Donations Page



## b. Requests to Fulfill Page



## c. Notification (Bell function) + Filter (Filter button)



## d. Delivery details (Details button)

The screenshot displays the FoodHero dashboard for a user named Donor123. The dashboard includes a navigation bar with links to Dashboard, Chatbot, About, Feedback, and a Log Out button. A welcome message is shown at the top, followed by two main action buttons: 'Manage Donations' and 'Requests to Fulfill'. Below these is a search bar and a list of requests. The first request is for Carrots, with a delivery date of 09-10-2024 12:00. A message from the driver is displayed: 'Our driver will contact you on the day of collection, please pack it nicely. Thank you.' The second request is for Canned Chicken Soup, with a quantity of 4 and a need by date of 09-10-2024 12:00. The third request is for Potatoes, with a quantity of 10 and an expiration date of NIL. Each request has a 'Details' button, and the last two have a 'Withdraw' button.

**FoodHero**

Dashboard Chatbot About Feedback **Log Out**

**Welcome Back, Donor123!**  
We're glad to have you here. Help us make a difference, share your food today!

**New Donation**

**Manage Donations**  
Cumulative of all requests made/accepted

**Requests to Fulfill**  
Cumulative of all requests available

Search for a Request **Search** **Filter**

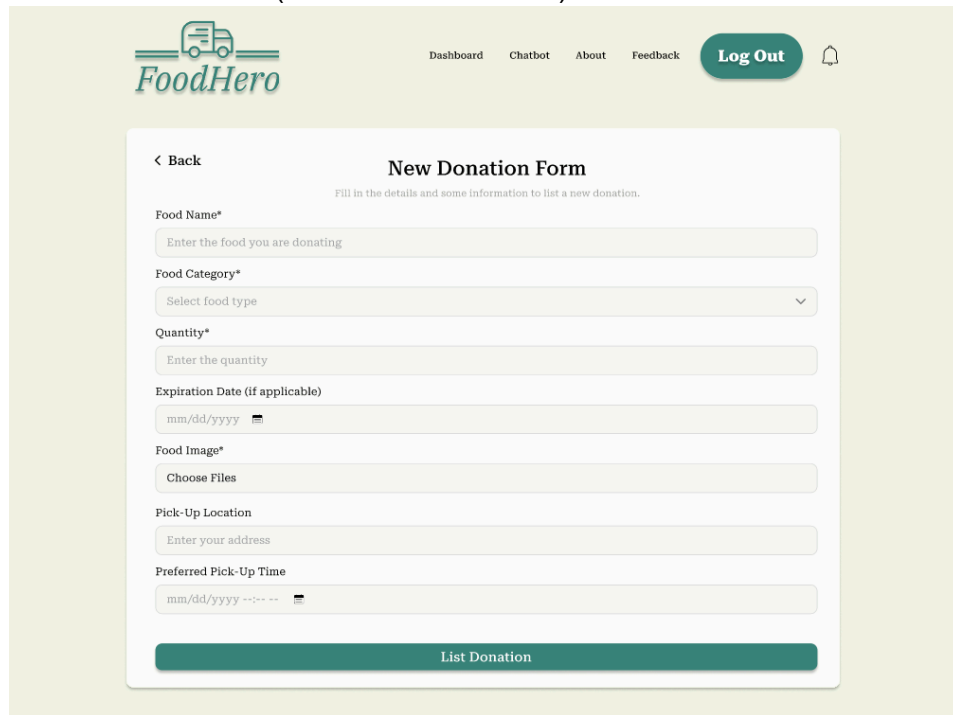
**Carrots**  
Food Category: Vegetables  
Quantity: 5kg  
Expiration Date: 09-10-2024 12:00  
**DELIVERY ON 09-10-2024 12:00**  
Our driver will contact you on the day of collection, please pack it nicely. Thank you.  
**Waiting Pick-Up**  
**Details**

**Canned Chicken Soup**  
Food Category: Canned Food  
Quantity: 4  
Need by: 09-10-2024 12:00  
**Donating to alex wee**  
**Waiting Delivery**  
**Details**

**Canned Chicken Soup**  
Food Category: Canned Food  
Quantity: 10  
Expiration Date: 09/10/2024  
**Withdraw**

**Potatoes**  
Food Category: Vegetables  
Quantity: 10  
Expiration Date: NIL  
**Withdraw**

## e. New Donation Form (New Donation button)

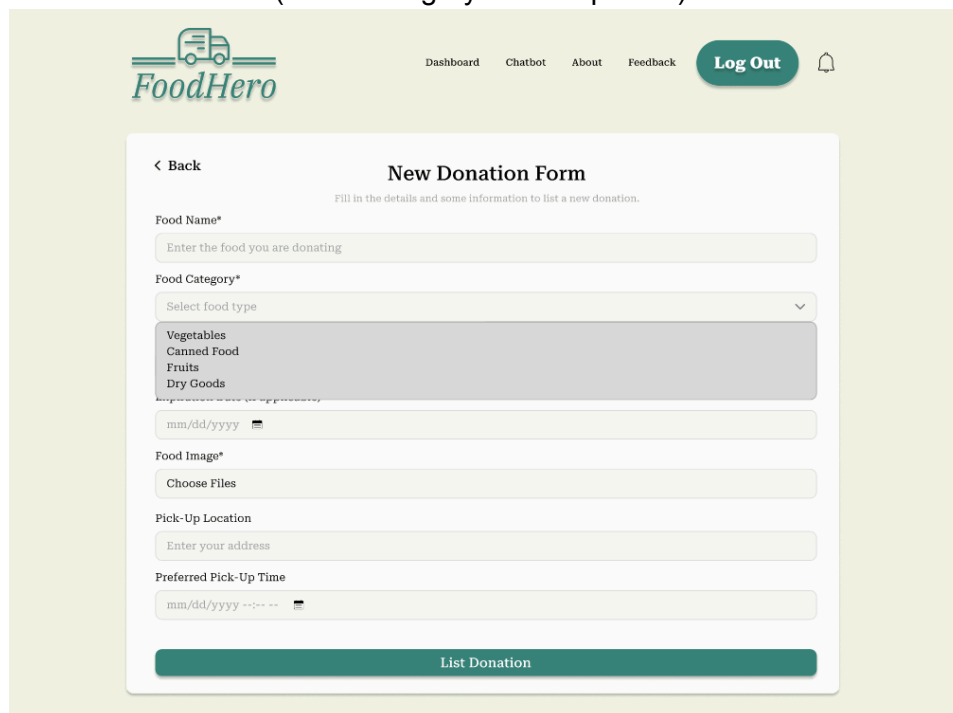


The screenshot shows the 'New Donation Form' in the FoodHero application. The form is titled 'New Donation Form' and includes a subtitle 'Fill in the details and some information to list a new donation.' The form fields are as follows:

- Food Name\***: A text input field with the placeholder 'Enter the food you are donating'.
- Food Category\***: A dropdown menu with the placeholder 'Select food type'.
- Quantity\***: A text input field with the placeholder 'Enter the quantity'.
- Expiration Date (if applicable)**: A date input field with the placeholder 'mm/dd/yyyy' and a calendar icon.
- Food Image\***: A file upload button labeled 'Choose Files'.
- Pick-Up Location**: A text input field with the placeholder 'Enter your address'.
- Preferred Pick-Up Time**: A date and time input field with the placeholder 'mm/dd/yyyy --:-- --' and a calendar icon.

A green button labeled 'List Donation' is located at the bottom of the form.

## f. New Donation Form (Food Category field dropdown)

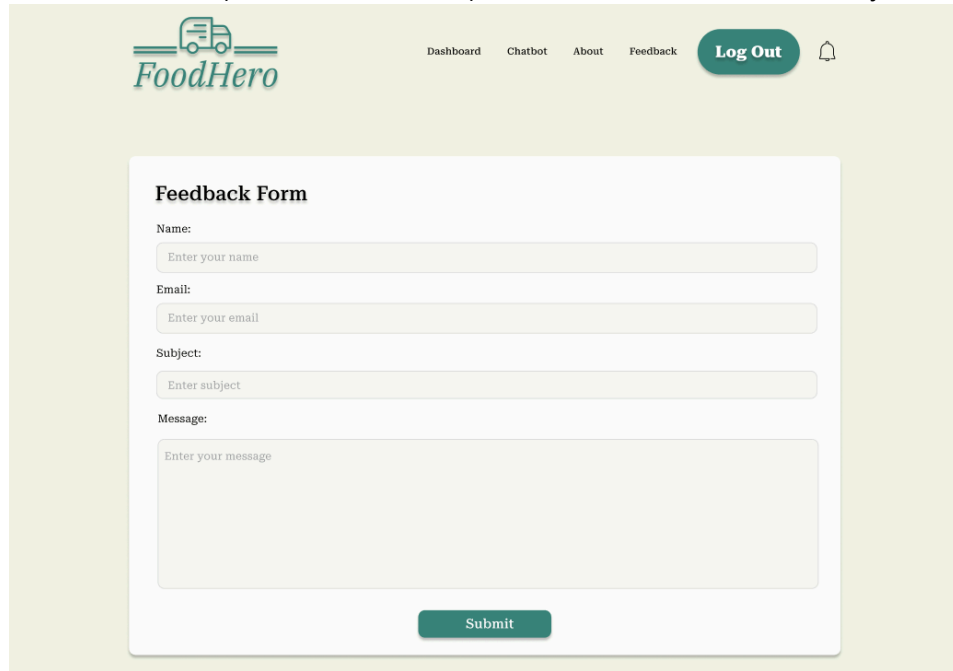


The screenshot shows the 'New Donation Form' in the FoodHero application, with the 'Food Category\*' dropdown menu open. The dropdown menu displays the following options:

- Vegetables
- Canned Food
- Fruits
- Dry Goods

The form fields and the 'List Donation' button are the same as in the previous screenshot.

g. Feedback Form (Feedback function) for both Donor and Beneficiary



The screenshot displays the 'Feedback Form' on the FoodHero website. The form is centered on a light beige background. At the top left is the FoodHero logo, which includes a green truck icon and the text 'FoodHero'. To the right of the logo is a navigation bar with links for 'Dashboard', 'Chatbot', 'About', and 'Feedback'. Further right is a green 'Log Out' button and a bell icon. The feedback form itself is a white box with a green border. It has a title 'Feedback Form' in bold. Below the title are four input fields: 'Name:' with a placeholder 'Enter your name', 'Email:' with a placeholder 'Enter your email', 'Subject:' with a placeholder 'Enter subject', and 'Message:' with a placeholder 'Enter your message'. The message field is a larger text area. At the bottom of the form is a green 'Submit' button.

**Feedback Form**

Name:  
Enter your name

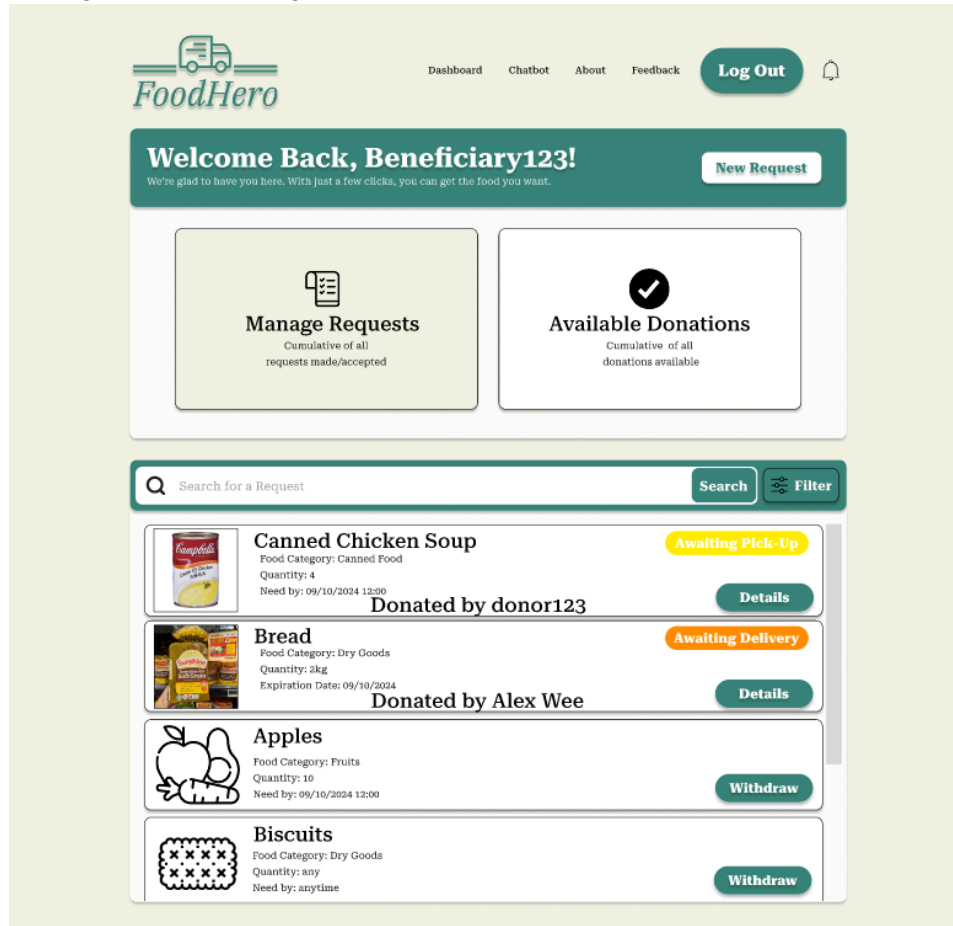
Email:  
Enter your email

Subject:  
Enter subject

Message:  
Enter your message

Submit

F. Beneficiary Dashboard  
a. Manage Requests Page



## b. Available Donations Page

The screenshot shows the 'Available Donations' page of the FoodHero application. The header includes the FoodHero logo, navigation links (Dashboard, Chatbot, About, Feedback), a 'Log Out' button, and a notification bell. A green banner at the top says 'Welcome Back, Beneficiary123!' with a 'New Donation' button. Below this are two main cards: 'Manage Requests' (Cumulative of all requests made/accepted) and 'Available Donations' (Cumulative of all donations available). A search bar with 'Search for a Request' and 'Search'/'Filter' buttons is present. The main content area lists four donation items, each with an image, title, category, quantity, expiration date, and an 'Accept' button:

Item	Food Category	Quantity	Expiration Date
Carrots	Vegetables	5	NIL
Bread	Dry Goods	3	09/10/2024
Canned Chicken Soup	Canned Food	10	09/10/2024
Potatoes	Vegetables	10	NIL

## c. New Request Form (New Request button)

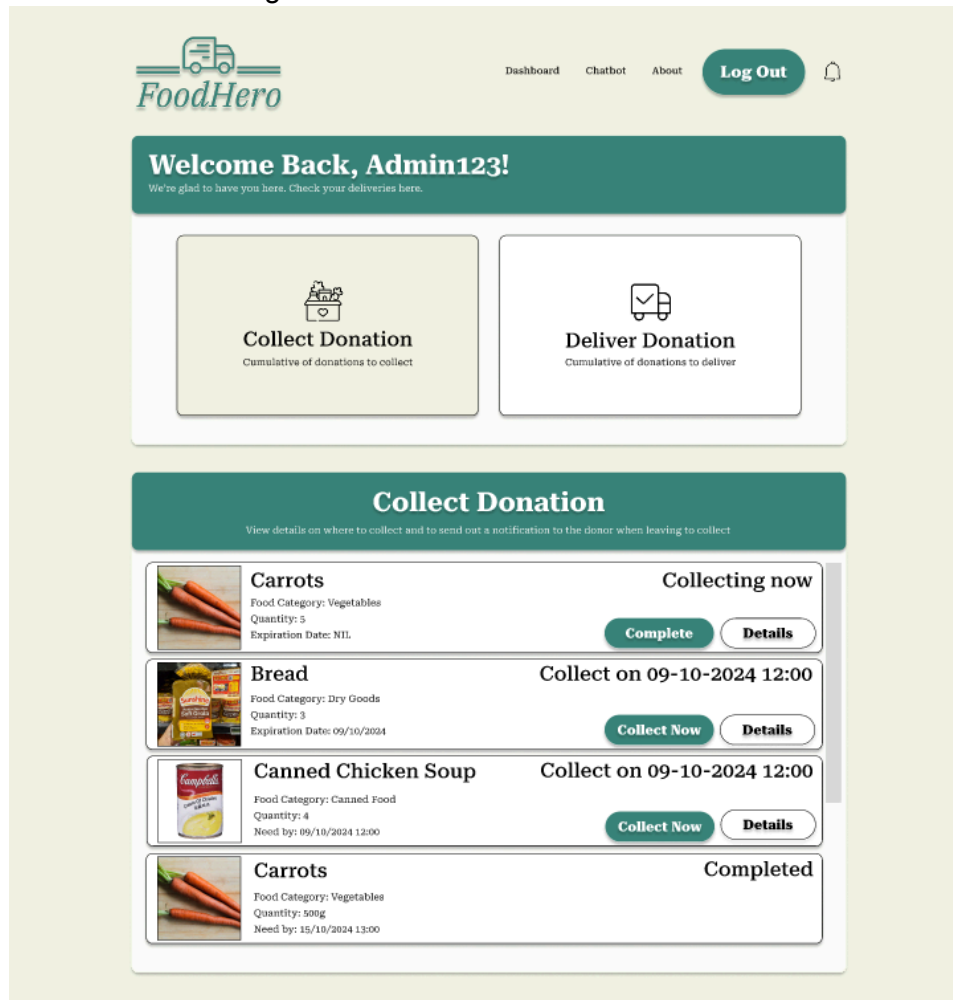
The screenshot shows the 'New Request Form' in the FoodHero application. The header is identical to the previous page. A '< Back' button is in the top left of the form. The title 'New Request Form' is centered, with a subtitle 'Fill in the details and some information to list a new request.' Below the title are several input fields:

- Food Name\***: Text input with placeholder 'Enter the food you are requesting'.
- Food Category\***: Dropdown menu with placeholder 'Select food type'.
- Quantity\***: Text input with placeholder 'Enter the quantity'.
- Need by (if applicable)**: Date input with placeholder 'mm/dd/yyyy --:-- --' and a calendar icon.
- Delivery Location**: Text input with placeholder 'Enter your address'.
- Preferred Delivery Time**: Date input with placeholder 'mm/dd/yyyy --:-- --' and a calendar icon.


A green 'List Request' button is at the bottom of the form.



G. Admin Dashboard  
a. Collect Donation Page




## b. Deliver Donation Page



DashboardChatbotAboutLog Out


Welcome Back, Admin123!

We're glad to have you here. Check your deliveries here.



Collect Donation

Cumulative of donations to collect




Deliver Donation

Cumulative of donations to deliver

Deliver Donation

View details on where to deliver and to send out a notification to the beneficiary when leaving to deliver



Carrots


Food Category: Vegetables

Quantity: 5

Expiration Date: NIL

Delivering now

CompleteDetails



Bread


Food Category: Dry Goods

Quantity: 3

Expiration Date: 09/10/2024

Deliver on 09-10-2024 12:00

Deliver NowDetails



Canned Chicken Soup


Food Category: Canned Food

Quantity: 4

Need by: 09/10/2024 12:00

Deliver on 09-10-2024 12:00

Deliver NowDetails



Carrots

Food Category: Vegetables

Quantity: 500g

Need by: 15/10/2024 13:00

Completed

- c. Delivery details (Details button) + Fastest Route (Generate Fastest Route button)

The screenshot displays the FoodHero Admin Dashboard. At the top, the navigation bar includes 'Dashboard', 'Chatbot', 'About', and a 'Log Out' button. A welcome message for 'Admin123!' is shown. Below this are two main action buttons: 'Collect Donation' and 'Deliver Donation'. The 'Collect Donation' section lists four items: Carrots (Collecting now), Bread (Collect on 09-10-2024 12:00), Canned Chicken Soup (Collect on 09-10-2024 12:00), and another Carrots entry (Completed). Each item has a 'Details' button. A 'Generate Fastest Route' button is positioned over the Bread and Canned Chicken Soup entries. The 'Fastest Route' section at the bottom shows a map with a route from '218 Pandan Loop' to '24 Toa Payoh Drive'.

**FoodHero** Dashboard Chatbot About Log Out

Welcome Back, Admin123!  
We're glad to have you here. Check your deliveries here.

**Collect Donation**  
View details on where to collect and to send out a notification to the donor when heading to collect.

**Carrots** Collecting now  
Food Category: Vegetables  
Quantity: 1  
Expiration Date: 08/10/2024  
Collecting from: donor123  
Contact number: 98765432  
Address: 24 Toa Payoh Drive, Singapore 234345  
Generate Fastest Route Details

**Bread** Collect on 09-10-2024 12:00  
Food Category: Baking Ingredients  
Quantity: 3  
Expiration Date: 08/10/2024  
Collect Now Details

**Canned Chicken Soup** Collect on 09-10-2024 12:00  
Food Category: Canned Food  
Quantity: 4  
Need by: 09/10/2024 12:00  
Collect Now Details

**Carrots** Completed  
Food Category: Vegetables  
Quantity: 500g  
Need by: 10/10/2024 12:00

**Fastest Route**

Map showing the fastest route from 218 Pandan Loop (Singapore 118438) to 24 Toa Payoh Drive (Singapore 234345). The route passes through Pandan Road, Holland Road, and Toa Payoh Road.

## 3.2 Hardware Interfaces

The **Food Hero** app interfaces with a variety of hardware components, primarily through standard web technologies. The logical and physical characteristics of these interfaces include:

- **Supported Devices:** The app is designed to be device-agnostic, functioning seamlessly on desktops, laptops, tablets, and smartphones. It supports touch inputs on mobile devices and standard keyboard/mouse inputs on desktop systems.
- **Data and Control Interactions:** The app interacts with hardware primarily through the web browser, handling inputs such as touch, keyboard, and mouse events. The app is optimized for low-latency responses to user actions.
- **Communication Protocols:** The app uses HTTP/HTTPS protocols for communication between the client device and the server. This ensures secure data transmission across all devices.

## 3.3 Software Interfaces

The **Food Hero** app integrates with several software components to deliver its full functionality. Key software interfaces include:

- **Database:** The app uses MongoDB for data storage and retrieval, with APIs built to interact with the database for storing user information, donation details, and beneficiary requests.
- **Operating Systems:** The app is compatible with major operating systems, including Windows, macOS, Linux, Android, and iOS, running within standard web browsers on these platforms.
- **Third-Party Services:** The app integrates with AWS SNS for real-time notifications and the Gemini API for the chatbot feature. These services communicate with the app via RESTful APIs, ensuring smooth data exchange and real-time updates.
- **Data Flow:** Data flows between the frontend (Next.js) and the backend (Node.js with MongoDB) via RESTful API calls. The backend processes the data and interacts with the database, while also handling requests to external services like AWS SNS and the Gemini API.

### 3.4 Communications Interfaces

The **Food Hero** app relies on several communication interfaces to function effectively:

- **Network Protocols:** The app uses HTTP/HTTPS for all network communications, ensuring secure data transmission between clients and the server. All communications are encrypted using TLS to protect user data.
- **Email Notifications:** The app integrates with AWS SNS to send email notifications to users. The email messages are formatted in HTML and include relevant details about donations, matches, and system updates.
- **Synchronization Mechanisms:** The app uses real-time synchronization between the frontend and backend to provide users with up-to-date information, especially in relation to donation matches and notifications.
- **Security:** All communications are secured with standard encryption protocols to protect sensitive user data, including personal information and donation details.

## **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.

### **4.3. 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:

- Food Name
- Food Category
- Quantity
- Expiration Date (if applicable)
- Food Image
- Pickup Location
- Preferred Pick Up Time

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:

- Food Name
- Food Category
- Quantity
- Need By (If applicable)
- Delivery Location
- Preferred Delivery Time

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 existing listings and display its details.

4.5.1.2. The donor must be able to browse through requests from beneficiaries to find new donation opportunities.

4.5.1.3. The donor must have the option to donate to the beneficiary if the request falls within their means.

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 withdraw their existing listings.

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.

4.5.2.2. The beneficiary must be able to view available donations and request for donations based on their requirements.

4.5.2.3. The beneficiary must be able to accept donations from donors.

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 “Completed” via a button click.

#### 4.5.4. Search and Filter

4.5.4.1. Both donors and beneficiaries must be able to search and filter functionality to quickly locate specific items or requests by applying filters such as item type, date.

### **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. Awaiting Pick Up

4.6.2.1.1. The system must link up with the dashboard system to display the status "Awaiting Pick Up" indicating that the donation is ready for pick up upon a match.

##### 4.6.2.2. Awaiting Delivery

4.6.2.2.1. The system must link with the dashboard system to display the status "Awaiting Delivery" indicating that the donation is ready for delivery once it has been picked up.

##### 4.6.2.3. Completed

4.6.2.3.1. The system must link with the dashboard system to display the status "Completed," indicating that the donation has been successfully sent to the donors.

## **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 both donor and 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 notify the beneficiary and donor when their match is mutually accepted through email.

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

4.8.2.1. Food type

4.8.2.2. Amount available

4.8.2.3. Expiration date

4.8.2.4. Donor's name

4.8.3. The system must notify the donor and beneficiary when the admin has picked up the food or when the delivery is completed.

## **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.**

## 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
- Expiration Date (if applicable)
- Food Image
- Pickup Location
- Preferred Pick Up Time

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
- Need By (if applicable)
- Delivery Location
- Preferred Delivery Time

**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 (pending, accepted, completed), 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, location, 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 donations pending collection or delivery, and provides route optimization for deliveries. The admin can track the progress of each match and update statuses as needed (e.g., collection completed, delivery pending).

**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 must be able to search and filter functionality to quickly locate specific items or requests by applying filters such as item type, date.**

## **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. Awaiting Pick Up

**REQ 2: The system must link up with the dashboard system to display the status "Awaiting Pick Up" indicating that the donation is ready for pick up upon a match.**

##### 5.6.3.2.2. Awaiting Delivery

**REQ 3: The system must link with the dashboard system to display the status "Awaiting Delivery" indicating that the donation is ready for delivery once it has been picked up.**

##### 5.6.3.2.3. Completed

**REQ 4: The system must link with the dashboard system to display the status "Completed," indicating that the donation has been successfully sent to the donors.**

## 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.**

## 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: A match is confirmed between a donor and a beneficiary.**

**Response:** System sends an email to both parties with the details of the match.

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

**Response:** System notifies the donor and beneficiary of the updated status via email.

#### 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 type
- Amount available
- Expiration date
- Donor's name

**REQ-3:** The system must notify the donor and beneficiary when the admin has picked up the food or when the delivery is completed.

## 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 also includes the following requirements:

- **Database Requirements:** The app's database (MongoDB) must support high availability and be capable of handling large volumes of data. It should also include backup and recovery mechanisms to prevent data loss.
- **Internationalization:** The app should be designed to support multiple languages, starting with English and potentially expanding to include other languages spoken in Singapore.
- **Legal Requirements:** The app must comply with all relevant legal requirements, including those related to food safety, data protection, and electronic communications. This includes adherence to local laws and regulations in Singapore.
- **Reuse Objectives:** The app's architecture should be designed with reusability in mind, allowing components like the matching algorithm or chatbot to be adapted for future projects or enhancements.

## Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

## Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

## Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>

Source: [http://www.frontiernet.net/~kwiegers/process\\_assets/srs\\_template.doc](http://www.frontiernet.net/~kwiegers/process_assets/srs_template.doc)

## 8. Data Dictionary

Term	Definition
User	An individual interacting with the system, categorized as donor, beneficiary, admin, or volunteer.
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 approves matches between donors and beneficiaries
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, "Awaiting Pick Up," "Awaiting Delivery," and "Completed."
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.