
Software Requirements Specification

for
<NTUFoodie>

Version 1.3 approved

Prepared by Team 49
**<CHUA YONG TAI ANTHONY
HO KING HEI
KHOO QIAN YEE
MUHAMMAD WISNU DARMAWAN
RACHEL LIM LIN>**

15/11/2024

Table of Contents

1. Introduction	1
1.1 Purpose	1
1.2 Document Conventions	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Product Scope	1
2. Overall Description	2
2.1 Product Perspective	2
2.2 Product Functions	2
2.3 User Classes and Characteristics	2
2.4 Operating Environment	2
2.5 Design and Implementation Constraints	3
2.6 User Documentation	3
2.7 Assumptions and Dependencies	3
3. External Interface Requirements	4
3.1 User Interfaces	4
3.1.1 Style Guide	4
3.1.2 UI Mockups	5
3.2 Hardware Interfaces	16
3.2.1 Device Type Specification	16
3.2.2 Data and Control Interactions	16
3.2.3 Communication Protocols	16
3.3 Software Interfaces	16
3.3.1 Operating System	16
3.3.2 Web Browsers	16
3.3.3 Databases	16
3.3.4 Backend Tools	16
3.3.5 Frontend Tools	17
3.3.6 Mapbox APIs	17
3.3.7 Data Sharing	17
3.4 Communications Interfaces	17
4. System Features	19
4.1 User Features	19
4.1.1 Sign Up	19
4.1.2 Login	19
4.1.3 Logout	20
4.1.4 Update Profile	21
4.1.5 View Notifications	21
4.1.6 Delete Notifications	22
4.2 Consumer Features	23

4.2.1 View Map	23
4.2.2 Get Route	24
4.2.3 View Canteen	24
4.2.4 View Store	25
4.2.5 View Food	26
4.2.6 Search Food	27
4.2.7 Filter Food	27
4.2.8 Submit Food Review	28
4.2.9 Update Food Review	29
4.2.10 Delete Food Review	30
4.2.11 Flag Food Review	31
4.3 Store Owner Features	31
4.3.1 Submit Food Listing	31
4.3.2 Update Food Listing	32
4.3.3 Delete Food Listing	33
4.4 Admin Features	33
4.4.1 Process Review	33
4.4.2 Ban User	35
4.4.3 Notify User	36
5. Other Nonfunctional Requirements	37
5.1 Performance Requirements	37
5.2 Safety Requirements	37
5.3 Security Requirements	37
5.4 Software Quality Attributes	37
5.5 Business Rules	38
6. Other Requirements	39

Revision History

Name	Date	Reason For Changes	Version
NTUFoodies SRS	19/10/24	Initialisation of SRS Document	1.0
NTUFoodiesSRS	3/11/24	Upload of UI Mockups	1.1
NTUFoodie_SRS	4/11/24	Upload of diagrams	1.2
NTUFoodie_SRS	15/11/24	Content writeup	1.3

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document specifies the requirements for the NTUFoodie system. The application is designed to help NTU students and staff find, review, and share information about food options within the NTU campus due to inflating prices, all while reducing food wastage. This document covers the entire scope of the application, including user account management, food hunting features, and admin management functionalities.

1.2 Document Conventions

- Requirements are organized hierarchically and numbered for easy reference.
- "Shall" is used to indicate mandatory requirements.
- "Should" is used to indicate recommended but not mandatory features.
- Use case diagrams and descriptions are used to illustrate system functionality.
- UI mockups are provided to visualize the user interface design.

1.3 Intended Audience and Reading Suggestions

This document is intended for:

- Development Team: To understand the full scope of the system to be built
- Project Managers: To plan the development phases and allocate resources
- Testers: To develop test plans and cases based on the requirements
- NTU Stakeholders: To review and approve the proposed system features

1.4 Product Scope

The NTUFoodies application aims to simplify the process of finding and reviewing food options within the NTU campus. Key features include:

- User account management for consumers and store owners
- A map-based interface for locating food establishments
- Food search and filter functionality
- User-generated reviews and ratings
- Food listing management for store owners
- Administrative tools for content moderation by admin

The application will benefit the NTU community by providing a centralised platform for food-related information, promoting local eateries, providing affordable food information to NTU community and improving the overall dining experience on campus.

2. Overall Description

2.1 Product Perspective

The NTUFoodies application is a new, self-contained web application designed specifically for the NTU community. The application will consist of three main components:

1. User Account Management System
2. Food Exploration and Review System
3. Admin Management System

These components will work together to provide a comprehensive food discovery and review platform for the NTU community.

2.2 Product Functions

The main functions of the NTUFoodies application include:

- User registration and profile management
- Interactive campus map with food locations
- Food search and filtering
- Review submission and management
- Food listing creation and management (for store owners)
- Content moderation and user management (for admins)

2.3 User Classes and Characteristics

The application caters to three main user classes:

1. Students/Staff (**Consumers**): Regular users who will search for food, read, write and flag reviews.
2. **Store Owners**: Owners or managers of food establishments, such as canteen stall owner who can manage their food listings.
3. System Administrators (**Admins**): Responsible for content moderation (sending notification to users) and user management.

2.4 Operating Environment

The application shall operate in the following environment:

- Mobile devices running iOS 14+ or Android 10+
- Web browsers including Chrome, Firefox, Safari, and Edge (latest versions)
- Server infrastructure capable of handling up to 10,000 concurrent users

2.5 Design and Implementation Constraints

- The application must comply with NTU's data protection and privacy policies as well as Singapore's Personal Data Protection Act (PDPA).
- The application should be developed using cross-platform technologies to reduce maintenance overhead.
- All communication between client and server must be encrypted using HTTPS.

2.6 User Documentation

The following user documentation will be provided:

- User manual for consumers, store owners, and administrators
- Video tutorials, demonstrating the key features

2.7 Assumptions and Dependencies

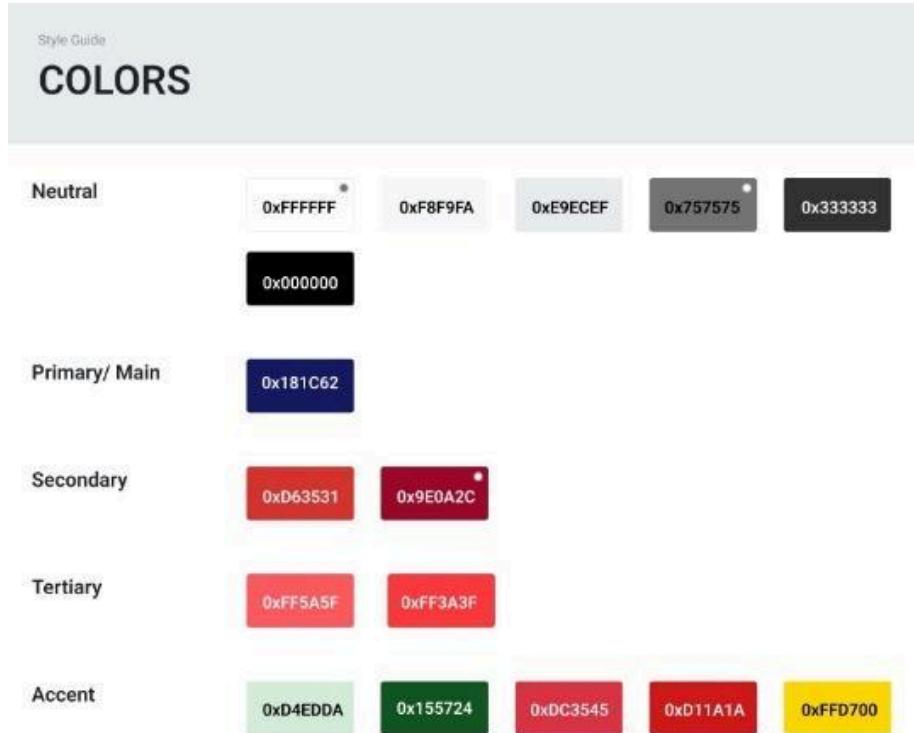
- Users have access to smartphones or computers with internet connectivity and location access.
- Food establishments on campus are willing to participate and keep their information updated.
- The application relies on user-generated content for reviews and ratings.

3. External Interface Requirements

3.1 User Interfaces

3.1.1 Style Guide

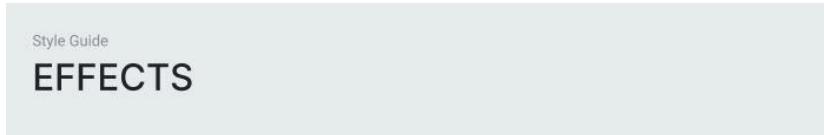
3.1.1.1 Theme Colours



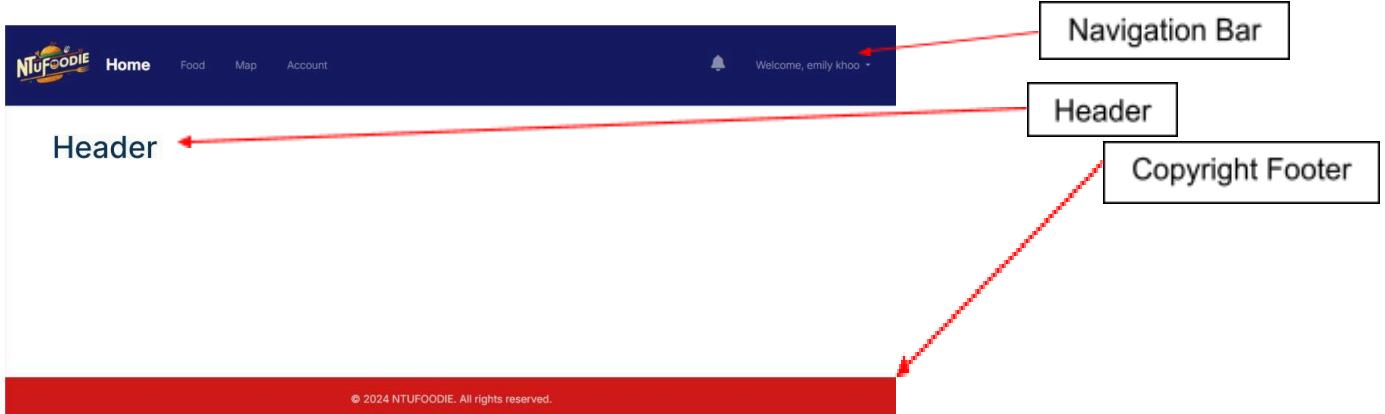
3.1.1.2 Typography

TYPOGRAPHY			
Heading	48px	Inter	San Serif
SubHeading	40px	Inter	San Serif
Text (Large)	36px	Inter	San Serif
Text (Medium)	32px	Inter	San Serif
Text (Small)	24px	Inter	San Serif
Text (Extra Small)	20px	Inter	San Serif

3.1.1.3 UI Effects



3.1.1.4 Template



3.1.2 UI Mockups

3.1.2.1 About page

The image displays two screenshots of the NTUFoodie website:

- Left Screenshot (Homepage):** Shows a banner with three women eating food. Below the banner are three main buttons: 'Search for Food' (magnifying glass icon), 'Locate Food Stalls' (location pin icon), and 'Rate and Review' (star icon).
- Right Screenshot (About Page):** Shows a dark header with the NTUFoodie logo and a search bar. Below the header are three sections: 'Search for Food', 'Locate Food Stalls', and 'Rate and Review'. Each section has a brief description and a quote from a student. At the bottom is a large blue button labeled 'Hungry?' with the text 'Find the perfect meal on NTU campus.' and a 'Explore Now' button.

3.1.2.2 Authentication (Login + Signup)

3.1.2.3 Signup Successful + User Banned Message

3.1.2.4 Wrong Username/ Password Message

3.1.2.5 Consumer UIs

Home

Available Food Listings

Image	Name	Details
	Salted Egg Chicken Rice	Rating: 4.2 ★ Non-Halal Chinese SGD 4.00 - SGD 4.40 Salted Egg Chicken Rice
	Gong Pao Chicken Rice	Rating: NA Non-Halal Chinese SGD 5.00 Delicious Gong Pao Chicken
	Bibimbap	Rating: 4.0 ★ Non-Halal Korean SGD 4.00 Delicious Bibimbap @ Korean Store
	Healthy Bowl (Salmon)	Rating: 4.0 ★ Vegetarian Western SGD 4.00 - SGD 7.50 Healthy Bowl (Salmon)

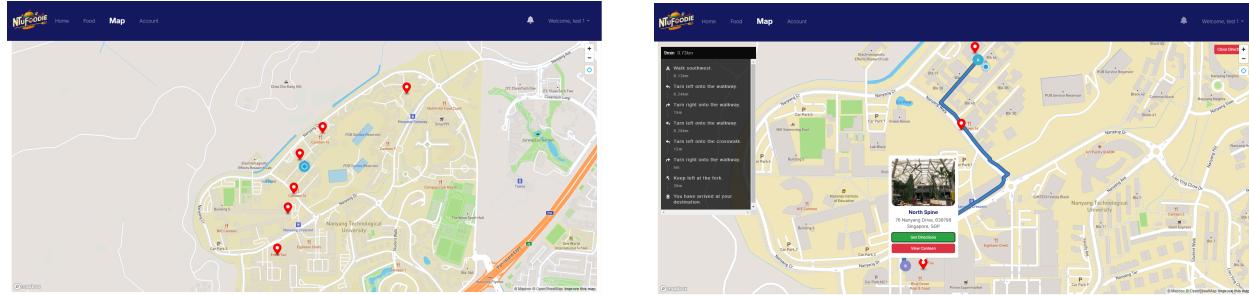
Food Filter + Food Search

Food Filter Result + Food Search Result

Image	Name	Details
	Healthy Bowl (Salmon)	Rating: 4.0 ★ Vegetarian Western SGD 4.00 - SGD 7.50 Healthy Bowl

Image	Name	Details
	Bibimbap	Rating: 4.0 ★ Non-Halal Korean SGD 4.00 - SGD 6.50 Delicious Bibimbap @ Korean Store
	Bulgogi	Rating: 4.0 ★ Non-Halal Korean SGD 4.00 - SGD 6.50 Sweet Bulgogi

Map (With User Geolocation and Canteen Markers) + Get Route



Canteen Details + Store Details

[Back to Map](#)

Quad Cafe



Stores in Quad Cafe



[View All](#)

[Back to Map](#)

[Back to Quad Cafe](#)

Korean Store



Available Food Items

 Bibimbap Price: \$5.5	 Bulgogi Price: \$6.5
--	---

Notifications in Navigation Bar (Active) + Notifications dropdown

[Home](#) [Food](#) [Map](#) [Account](#)

Welcome, test 1

Latest Reviews

 Test 1 5.0

What they had: Salted Egg Chicken Rice
Hall: Katsu Nasi Wok
Location: North Spine
Description: Solid



[View all](#)

Discounted Food



Healthy Bowl (Salmon)

4.0

Type: Vegetarian
Calories: 400



Salted Egg Chicken Rice

4.3

Type: Vegetarian
Calories: 450

[Home](#) [Food](#) [Map](#) [Account](#)

Welcome, test 1

Identifications

Welcome to NTUFOODIE!
Download our native app now
signing in

Test notifications
Message from test 1
This notification
is just a test.

[View all](#)

Latest Reviews

 Test 1 5.0

What they had: Salted Egg Chicken Rice
Hall: Katsu Nasi Wok
Location: North Spine
Description: Solid



[View all](#)

Discounted Food



Healthy Bowl (Salmon)

4.0

Type: Vegetarian
Calories: 400

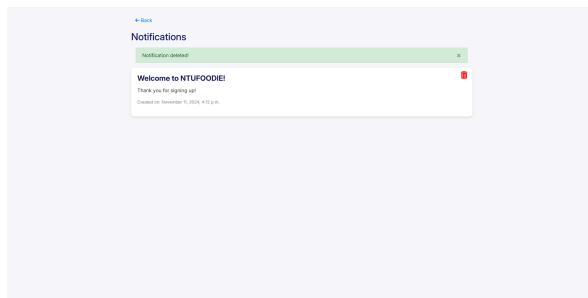
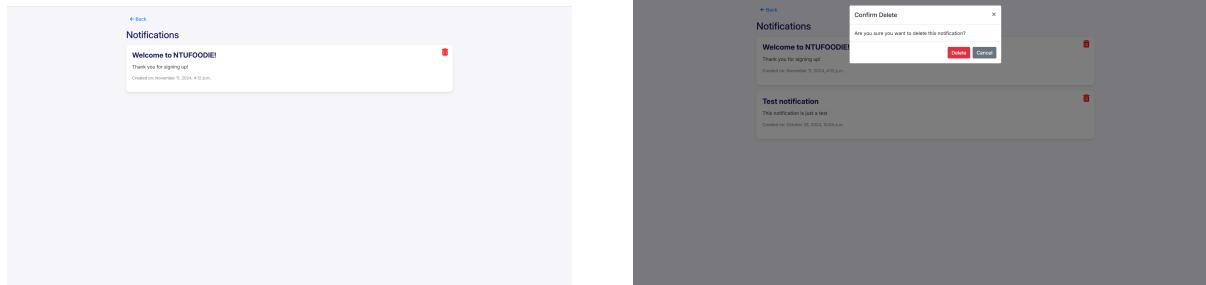


Salted Egg Chicken Rice

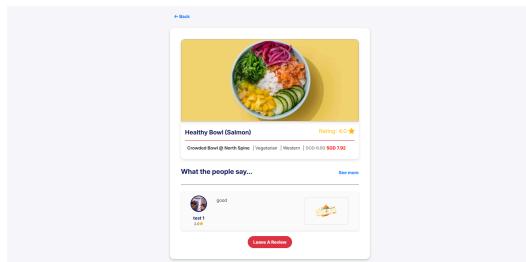
4.3

Type: Vegetarian
Calories: 450

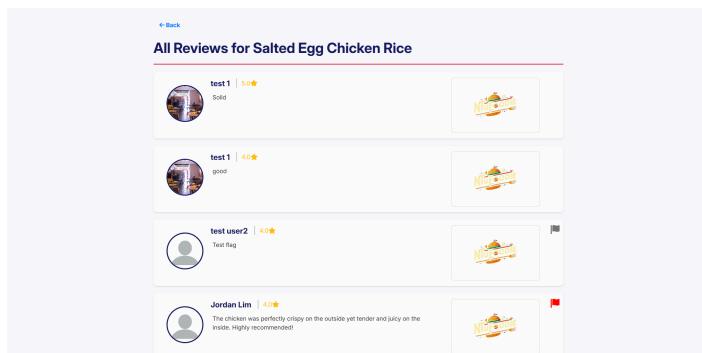
Notifications + Delete Notifications Confirmation + Success Message



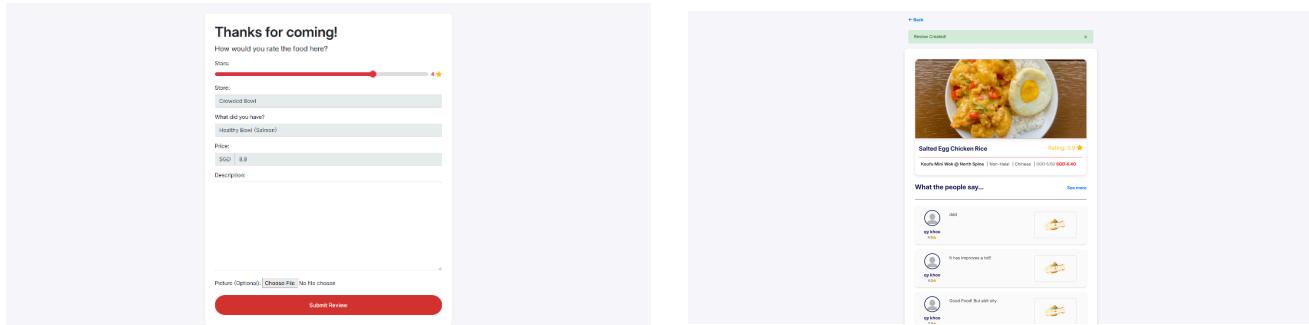
Food Details



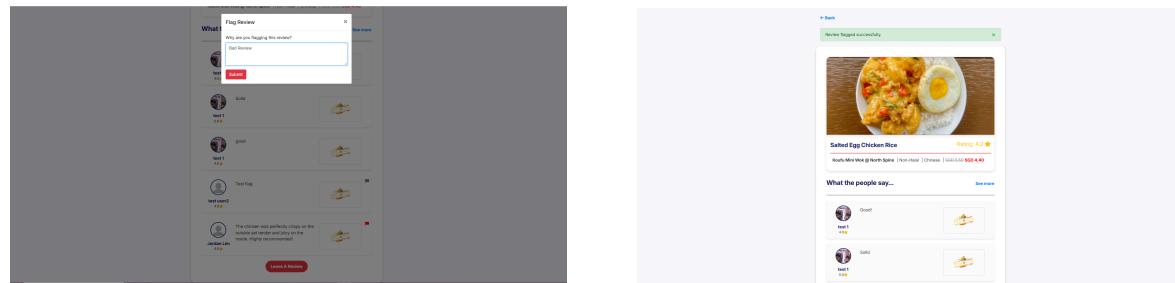
All Food Reviews (For Target Food Listing)



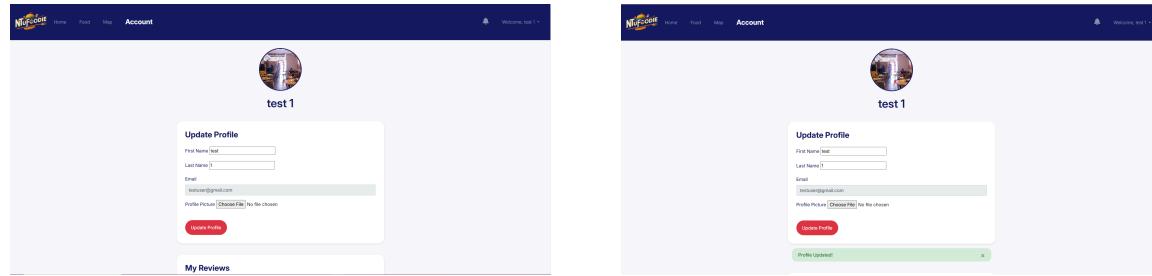
Food Review Form + Success Message



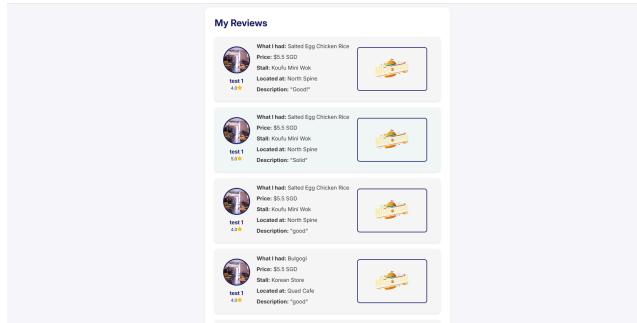
Flag Food Review + Success Message



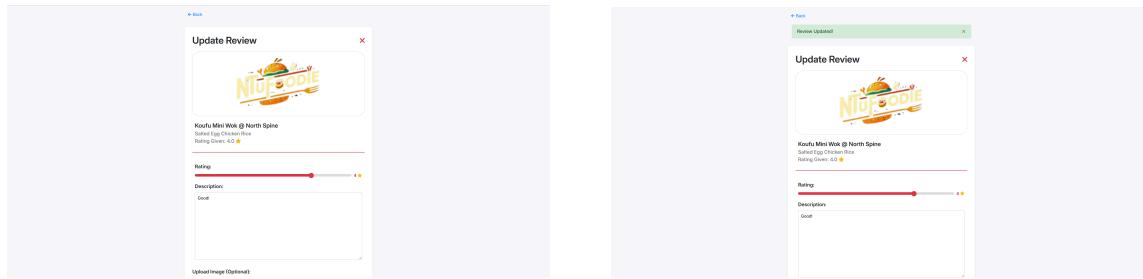
Consumer Account + Account Update Success Message



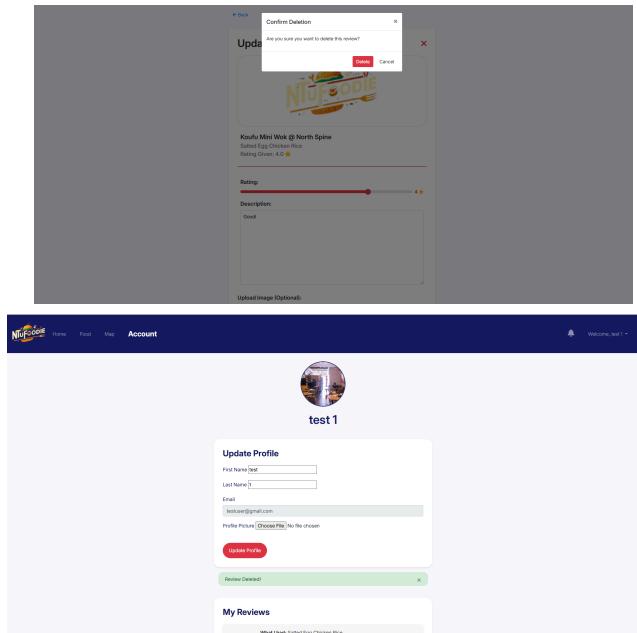
Consumer Past Reviews



Update Food Review + Success Message

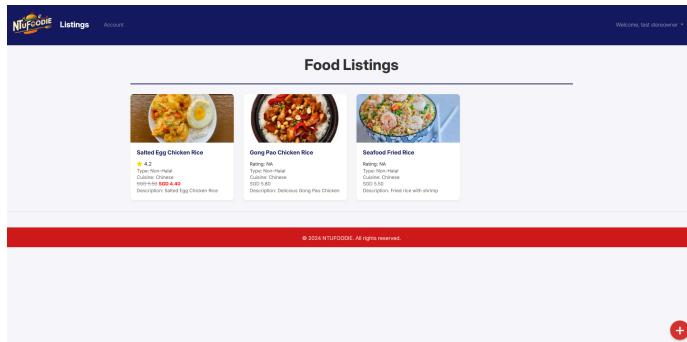


Delete Food Review + Success Message

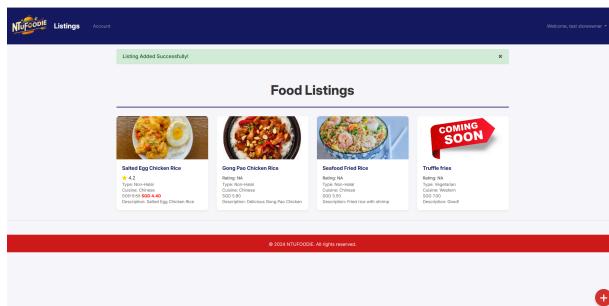
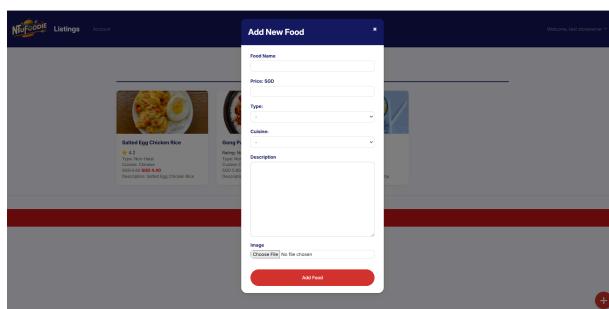


3.1.2.6 Store Owner UIs

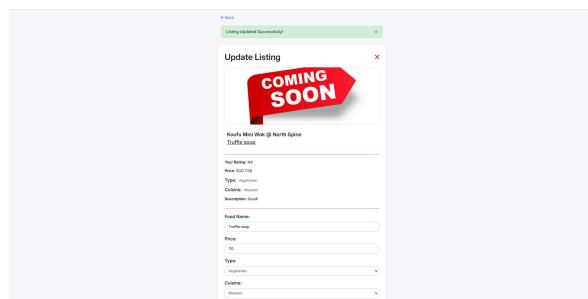
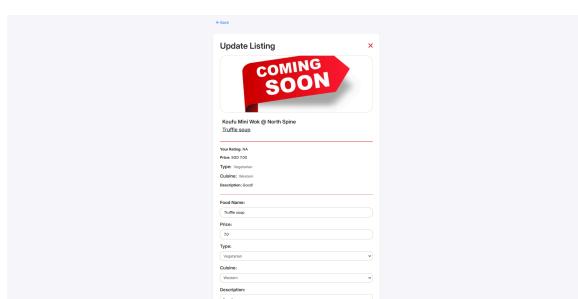
Food Listing(s) Posted



Add Food Listing Form + Success Message



Update Food Listing Information + Success Message



Delete Food Listing + Success Message

Store Owner Account + Update Success Message

3.1.2.7 Admin UIs

Admin Dashboard

Admin Home Page

System Notifications

The screenshot shows a list of notifications under the heading "Notifications". Each notification is a card with a red "X" icon in the top right corner. The notifications are:

- Review Flagged**: A review for Salted Egg Chicken Rice by Jordann has been flagged for: bad. Created on: November 2, 2024, 2:40 p.m.
- Review Flagged**: A review for Salted Egg Chicken Rice by emily has been flagged for: BAD. Created on: November 1, 2024, 8:14 a.m.
- Review Flagged**: A review for Salted Egg Chicken Rice by testuser2 has been flagged for: Very dumb comment here. Created on: November 1, 2024, 4:40 a.m.
- Review Flagged**: A review for Salted Egg Chicken Rice by testuser2 has been flagged for: irrelevant. Created on: October 28, 2024, 9:23 a.m.
- Review Flagged**: A review for Salted Egg Chicken Rice by Jordann has been flagged for: fake review. Created on: November 1, 2024, 4:40 a.m.

Delete Notification Confirmation + Success Message

The first screenshot shows a "Confirm Delete" dialog box asking "Are you sure you want to delete this notification?". The second screenshot shows the notifications list with one item removed, indicating a successful deletion.

Left Screenshot (Delete Confirmation):

Confirm Delete
Are you sure you want to delete this notification?
Yes **Cancel**

Right Screenshot (Updated Notifications):

Notifications
Notification deleted

- Review Flagged**: A review for Salted Egg Chicken Rice by emily has been flagged for: BAD. Created on: November 1, 2024, 8:14 a.m.
- Review Flagged**: A review for Salted Egg Chicken Rice by testuser2 has been flagged for: bad. Created on: November 1, 2024, 4:40 a.m.
- Review Flagged**: A review for Salted Egg Chicken Rice by testuser2 has been flagged for: irrelevant. Created on: October 28, 2024, 9:23 a.m.
- Review Flagged**: A review for Salted Egg Chicken Rice by Jordann has been flagged for: fake review. Created on: November 1, 2024, 4:40 a.m.

Flagged Reviews List

The screenshot shows a list of flagged reviews for "Jordan Lim, Salted Egg Chicken Rice from Koufu Mini Wok". The review description is: "The chicken was perfectly crispy on the outside yet tender and juicy on the inside. Highly recommended!". Consumer flagging is listed as "emily". The reason for flagging is "bad". The review was flagged on November 2, 2024, at 2:40 p.m. There are two buttons at the bottom: "Mark as Processed" (green) and "Delete Review" (red).

Delete Review Confirmation + Success Message

The left screenshot shows a 'Flagged Reviews' page with a modal titled 'Confirm Delete'. It asks 'Are you sure you want to delete the review?' with 'Yes' and 'Cancel' buttons. Below the modal, three flagged reviews are listed: 'QY khoo, Salted Egg Chick', 'QY khoo, Salted Egg Chicken Rice from Koufu Mini Wok', and 'Jordan Lim, Salted Egg Chicken Rice from Koufu Mini Wok'. Each review has a 'Mark as Processed' button and a 'Delete Review' button.

The right screenshot shows the same 'Flagged Reviews' page after the deletion. A green success message at the top says 'Review deleted and user notified.' The three reviews remain listed, each with a 'Mark as Processed' button and a 'Delete Review' button.

Mark Flagged Review Success Message

A screenshot of a 'Flagged Reviews' page. A green success message at the top says 'Flagged review marked as processed.' Below it, a single flagged review by 'Jordan Lim' is shown. The review details are: 'Review Description: The chicken was perfectly crispy on the outside yet tender and juicy on the inside. Highly recommended!', 'Consumer flagging: emoji', 'Reason for Flagging: bad', and 'Flagged on: November 2, 2024, 8:09 a.m.'. At the bottom are 'Mark as Processed' and 'Delete Review' buttons.

Ban User Page

A screenshot of a 'User List' page. It displays four user profiles: 'testuser' (joined on October 22, 2024, 8:07 a.m.), 'Jordann' (joined on October 22, 2024, 1:47 p.m.), 'testuser1' (joined on October 26, 2024, 8:50 a.m.), and 'testuser2' (joined on October 27, 2024, 8:55 a.m.). Each profile includes a red 'Ban' link next to the user's name.

Ban Confirmation + Success Message & New User List After Ban

The left screenshot shows a 'User List' page with a modal titled 'Confirm Ban'. It asks 'Are you sure you want to ban this user?' with 'Yes' and 'Cancel' buttons. Below the modal, four users are listed: 'testuser', 'Jordann', 'testuser1', and 'testuser2'. Each user has a red 'Ban' link next to their profile.

The right screenshot shows the same 'User List' page after banning 'testuser'. A green success message at the top says 'User banned!'. The user 'testuser' is now listed as 'User banned!' with the same profile information. The other three users ('Jordann', 'testuser1', and 'testuser2') remain in the list.

3.2 Hardware Interfaces

NTUFoodie is designed to be primarily used as a web application, hence minimizing the need for direct hardware interfaces. The following will be outlining the logical and physical characteristics of interfaces between the software and hardware:

3.2.1 Device Type Specification

The software is designed to be compatible with various devices, including desktop computers, laptops, smartphones and tablets that have GPS receivers in mobile devices for location services.

3.2.2 Data and Control Interactions

The user's device interacts with the software through standard input/output hardware such as keyboards, touchscreens and mice; for cursor control, input data, displays such as monitors or smartphone screens for output, cameras for uploading food and profile pictures and network interface cards for internet connectivity to use the web app.

3.2.3 Communication Protocols

The platform utilizes HTTPS to ensure secure web communication. APIs, WebSocket connections, and services will operate through standard web ports, specifically port 80 for HTTP and port 443 for HTTPS.

3.3 Software Interfaces

3.3.1 Operating System

Any OS (ex. Linux, Windows, MacOS, iOS, Android) that can run a modern web browser.

3.3.2 Web Browsers

Latest versions of Chrome, Firefox, Safari, Edge, and others which are fully capable of supporting HTML5, CSS3, and JavaScript ES6.

3.3.3 Databases

SQL-based database systems, accessed through backend services.

3.3.4 Backend Tools

1. SQLite (embedded, file-based RDBMS)
2. Django Web Framework

3.3.5 Frontend Tools

1. Google Fonts for the font 'Inter'.
2. Bootstrap4 (version 4.5.2) for templates.
3. Font Awesome (version 6.0.0) for icons used throughout the template
4. jQuery (version 3.5.1) for handling user interactions and DOM manipulation.
5. Popper.js for managing the positioning of elements.
6. Prettier extension to tidy codebase.

3.3.6 Mapbox APIs

Mapbox API provides interactive maps and geolocation services with ready-made styles and real-time data for web-based applications. It features an advanced routing which is ideal for university students to find canteen locations easily. Integrated within the app for seamless front-end-backend communication, Mapbox offers thorough documentation, detailing request and response formats to streamline data handling and enhance user experience

3.3.7 Data Sharing

JSON format is used for all data exchange between frontend and backend. Data persistence is maintained through the SQL database with specific defined data models.

The application shall interface with:

- Map API: Mapbox
- Database management system
- Push notification services for user alerts

3.4 Communications Interfaces

For NTUFoodie, various communication interfaces are integral to ensuring a seamless and responsive user experience:

- **HTTP/HTTPS:** All interactions with the web application will occur over HTTP/HTTPS protocols. This includes API calls, data submissions, and retrieval of sensitive information, which ensures compatibility with modern web browsers and guarantees secure data transmission for all user interactions and transactions, protecting sensitive information.
- **API Communication:** Mapbox APIs will be integrated to provide interactive map functionalities, enhancing the overall user experience by allowing users to locate nearby canteens and navigate effectively.

- **Geolocation API:** Utilizes the device's GPS capabilities to provide location-based services. Integrates with mapping services to display user location and nearby food options.
- **Django API:** The backend will provide views to handle client-server communication, managing CRUD operations for food listings, reviews, and user profiles. These views can return JSON responses for client-side JavaScript handling or render HTML templates as needed.
- **Push Notification Services:** Notifications are created through a custom middleware in the backend. When an admin creates a notification for a specific user within the Django dashboard, the middleware automatically handles and queues the notification for delivery.
- **Electronic Forms:** Forms submitted through the NTUFoodie platform will utilize secure HTTPS transactions to protect user-submitted data, ensuring privacy and security during data transmission.
- **Network Protocols:** Standard network protocols such as TCP/IP will be employed for network communication, ensuring adherence to established internet communication standards and facilitating reliable data transfer.
- **Security and Encryption:** TLS (Transport Layer Security) will be enforced for all communications to safeguard data integrity and user privacy.

The application shall be designed to gracefully handle communication failures and provide appropriate error messages to users in case of connectivity issues within the NTUFoodie web app, aligned with the aim of allowing consumers to cut monthly costs together with food surpluses.

4. System Features

4.1 User Features

4.1.1 Sign Up

4.1.1.1 Description and Priority

Description:

- Users can sign up for a Consumer or Store Owner account in NTUFoodie

Priority:

- High

4.1.1.2 Stimulus/Response Sequences

1. The user enters their first name, last name, username, email address, password and their role
2. The user submits their first name, last name, username, email address, password and their role
3. The system validates and verifies the submitted first name, last name, username, email address, password and their role
4. The system displays a message saying "Account was created for <username>" when the sign-up is successful
5. The system will redirect the user to the login page

4.1.1.3 Functional Requirements

REQ-1: The system shall provide input fields for the user to enter their first name, last name, username, email address, password, and role on the sign-up page.

REQ-2: The system shall have a "Submit" button on the sign-up page that allows the user to submit the entered data (first name, last name, username, email address, password, and role) for account creation.

REQ-3: The system shall validate each field.

REQ-4: The system shall display a confirmation message, "Account was created for <username>," upon successful account creation, where <username> reflects the submitted username.

REQ-5: The system shall automatically redirect the user to the login page upon successful account creation.

4.1.2 Login

4.1.2.1 Description and Priority

Description:

- Users can login to the app and start using it
- Priority:
- High

4.1.2.2 Stimulus/Response Sequences

1. The user enters their username and password
2. The user submits their username and password
3. The system validates and verifies the submitted username and password
4. The system will display the user's homepage

4.1.2.3 Functional Requirements

- REQ-1: The system shall provide input fields for the user to enter their username and password on the login page.
- REQ-2: The system shall have a "Submit" button on the sign-up page that allows the user to submit the entered data (username and password).
- REQ-3: The system shall validate each field.
- REQ-4: The system shall automatically redirect the user to the respective user's home page upon successful login.

4.1.3 Logout

4.1.3.1 Description and Priority

- Description:
- Users can logout of the app
- Priority:
- Medium

4.1.3.2 Stimulus/Response Sequences

1. The user clicks on the name/username drop down on the top right of the page
2. The system displays a "Logout" button
3. The user clicks on "Logout"
4. The system redirects the user back to the Login page

4.1.3.3 Functional Requirements

- REQ-1: The system shall display a logout button (dropdown) at the top right corner of the page. This menu shall be accessible when the user clicks on their name or username.
- REQ-2: The system shall display a "Logout" button within the user profile drop-down menu after the user clicks on their name/username. The "Logout" button shall be clearly labeled and available for interaction.
- REQ-3: The system shall register the user's action when they click the "Logout" button.
- REQ-4: The system shall redirect the user to the login page after they click the "Logout" button. This shall log the user out of the current session and return them to the login screen.

4.1.4 Update Profile

4.1.4.1 Description and Priority

Description:

- Users can update their profiles like first name, last name and profile picture.

Priority:

- Medium

4.1.4.2 Stimulus/Response Sequences

1. The user clicks on "Profile" on the navigation bar
2. The system will display the user's profile page
3. The user enters the new input on the relevant fields
4. The user clicks on the "Update"
5. The system displays a message saying "Profile Updated"
6. The system redirects the user back to the Profile page

4.1.4.3 Functional Requirements

REQ-1: The system shall provide a "Profile" link in the navigation bar, which is visible and accessible to the user.

REQ-2: The system shall display the user's profile page when the "Profile" link is clicked, showing relevant details like name, email, and other user information.

REQ-3: The system shall allow the user to enter or update information in relevant fields on the profile page.

REQ-4: The system shall provide an "Update" button on the profile page that the user can click to submit changes.

REQ-5: The system shall display a confirmation message, "Profile Updated," when the user successfully updates their profile information.

REQ-6: The system shall redirect the user back to the Profile page after the update is successful

4.1.5 View Notifications

4.1.5.1 Description and Priority

Description:

- Consumers and Admins can view their notifications

Priority:

- Medium

4.1.5.2 Stimulus/Response Sequences

1. The user is a consumer and is logged in

- a. The user clicks on the Bell button at the top of their home page
 - b. The system displays the 5 most recent notifications the user has
 - c. The user clicks on the "View All" button
 - d. The system displays the Notification page with all the notifications for the user
2. The user is an admin and is logged in
- a. The system admin clicks on the "Notifications" button
 - b. The system displays all the notifications the system admin has

4.1.5.3 Functional Requirements

REQ-1: The system shall recognize and authenticate the user as a consumer who is logged in.

REQ-1.1: The system shall display a Bell icon button at the top of the consumer's home page, indicating the presence of notifications.

REQ-1.2: The system shall display the 5 most recent notifications when the user clicks on the Bell button.

REQ-1.3: The system shall provide a "View All" button on the notifications panel.

REQ-1.4: The system shall display the Notification page with all the notifications when the user clicks on the "View All" button.

REQ-2: The system shall recognize and authenticate the user as an admin who is logged in.

REQ-2.1: The system shall display a "Notifications" button in the admin's interface, accessible in the navigation menu or relevant section.

REQ-2.2: The system shall display all the notifications for the admin when the "Notifications" button is clicked.

4.1.6 Delete Notifications

4.1.6.1 Description and Priority

Description:

- Consumers and Admins can delete their notifications

Priority:

- Medium

4.1.6.2 Stimulus/Response Sequences

1. The consumer clicks on the Bell button at the top of their home page
 - a. The system displays the 5 most recent notifications the consumer has
 - b. The consumer clicks on the "View All" button
 - c. The system displays the Notification page with all the notifications for the consumer
 - d. The consumer clicks on the bin icon next to the notification
 - e. The system displays a pop-up modal
 - f. The consumer clicks on the "Delete" button
 - g. The notification is deleted and the system displays a message saying "Notification deleted!"
2. The system admin clicks on the "Notifications" button

- a. The system displays all the notifications the system admin has
- b. The system admin clicks on the bin icon next to the notification
- c. The system displays a pop-up modal
- d. The system admin clicks on the "Delete" button
- e. The notification is deleted and the system displays a message saying "Notification deleted!"

4.1.6.3 Functional Requirements

REQ-1: The system shall recognize and authenticate the user as a consumer who is logged in.

REQ-1.1: The system shall display a Bell icon button at the top of the consumer's home page, indicating the presence of notifications.

REQ-1.2: The system shall display the 5 most recent notifications when the consumer clicks on the Bell button.

REQ-1.3: The system shall provide a "View All" button in the notifications panel.

REQ-1.4: The system shall display the Notification page with all notifications when the consumer clicks on the "View All" button.

REQ-1.5: The system shall display a bin icon next to each notification on the Notification page, allowing the consumer to delete a notification.

REQ-1.6: The system shall display a pop-up modal when the consumer clicks the bin icon next to a notification.

REQ-1.7: The system shall provide a "Delete" button in the pop-up modal for deleting the notification.

REQ-1.8: The system shall delete the notification and display a message saying "Notification deleted!" after the consumer clicks the "Delete" button.

REQ-2: The system shall recognize and authenticate the user as an admin who is logged in.

REQ-2.1: The system shall display a "Notifications" button in the admin's interface, accessible in the navigation menu or relevant section.

REQ-2.2: The system shall display all notifications for the admin when the admin clicks the "Notifications" button.

REQ-2.3: The system shall display a bin icon next to each notification on the Notification page, allowing the admin to delete a notification.

REQ-2.4: The system shall display a pop-up modal when the admin clicks the bin icon next to a notification.

REQ-2.5: The system shall provide a "Delete" button in the pop-up modal for deleting the notification.

REQ-2.6: The system shall delete the notification and display a message saying "Notification deleted!" after the admin clicks the "Delete" button.

4.2 Consumer Features

4.2.1 View Map

4.2.1.1 Description and Priority

Description:

- Consumers can view the map that shows the food areas in NTU

Priority:

- High

4.2.1.2 Stimulus/Response Sequences

1. The consumer clicks on "Map" in the navigation bar
2. The system will display a map that shows the food areas in NTU

4.2.1.3 Functional Requirements

REQ-1: The system shall display a "Map" link in the navigation bar, which is visible and accessible to the consumer.

REQ-2: The system shall display an interactive map showing the food areas within NTU when the consumer clicks on the "Map" link in the navigation bar. The map shall be centered on NTU with clearly marked locations or regions representing food areas.

4.2.2 Get Route

4.2.2.1 Description and Priority

Description:

- Consumers who has clicked on a store location on the Map can get directions from their location to the store

Priority:

- High

4.2.2.2 Stimulus/Response Sequences

1. The consumer clicks on "Map" in the navigation bar
2. The system will display a map that shows the food areas in NTU
3. The consumer clicks on a marker displayed on the map and clicks the "Get Directions" button
4. The system displays the walking path from the consumer's location to the destination

4.2.2.3 Functional Requirements

REQ-1: The system shall display a "Map" link in the navigation bar, which is visible and accessible to the consumer.

REQ-2: The system shall display an interactive map showing the food areas within NTU when the consumer clicks on the "Map" link in the navigation bar. The map shall be centered on NTU with clearly marked markers indicating food area locations.

REQ-3: The system shall allow the consumer to click on any marker displayed on the map to select a food area.

REQ-4: The system shall display a "Get Directions" button after the consumer clicks on a marker.

REQ-5: The system shall provide a walking path from the consumer's current location to the selected food area when the consumer clicks the "Get Directions" button. The walking path shall be displayed on the map with clear directions.

4.2.3 View Canteen

4.2.3.1 Description and Priority

Description:

- Consumers who have clicked on Map and has clicked on the "View Canteen" button on the pop-up of a food location can view the different stores in the canteen

Priority:

- Medium

4.2.3.2 Stimulus/Response Sequences

1. The consumer clicks on "Map" in the navigation bar
2. The system will display a map that shows the food areas in NTU
3. The consumer clicks on one of the markers
4. The system will display a pop-up card that contains the canteen details
5. The consumer clicks on the "View Canteen" button
6. The system displays the canteen page

4.2.3.3 Functional Requirements

REQ-1: The system shall display a "Map" link in the navigation bar, which is visible and accessible to the consumer.

REQ-2: The system shall display an interactive map showing the food areas within NTU when the consumer clicks on the "Map" link in the navigation bar. The map shall be centered on NTU with clearly marked markers indicating food area locations.

REQ-3: The system shall allow the consumer to click on any of the markers displayed on the map to select a food area.

REQ-4: The system shall display a pop-up card containing detailed information about the selected canteen when the consumer clicks on one of the markers.

REQ-5: The system shall provide a "View Canteen" button within the pop-up card for the consumer to click.

REQ-6: The system shall display the canteen's detailed page when the consumer clicks the "View Canteen" button.

4.2.4 View Store

4.2.4.1 Description and Priority

Description:

- Consumers can view the food listing of a store after clicking on it from the canteen page

Priority:

- Medium

4.2.4.2 Stimulus/Response Sequences

1. The consumer clicks on "Map" in the navigation bar
2. The system will display a map that shows the food areas in NTU
3. The consumer clicks on one of the markers
4. The system will display a pop-up card that contains the canteen details
5. The consumer clicks on the "View Canteen" button
6. The system displays the canteen page

7. The consumer clicks on the specific store that he/she wants to browse
8. The system displays the store page that displays all the food listings the store has

4.2.4.3 Functional Requirements

REQ-1: The system shall display a "Map" link in the navigation bar, which is visible and accessible to the consumer.

REQ-2: The system shall display an interactive map showing the food areas within NTU when the consumer clicks on the "Map" link in the navigation bar. The map shall be centered on NTU with clearly marked markers indicating food area locations.

REQ-3: The system shall allow the consumer to click on any of the markers displayed on the map to select a food area.

REQ-4: The system shall display a pop-up card containing detailed information about the selected canteen when the consumer clicks on one of the markers.

REQ-5: The system shall provide a "View Canteen" button within the pop-up card for the consumer to click.

REQ-6: The system shall display the canteen's detailed page when the consumer clicks the "View Canteen" button.

REQ-7: The system shall allow the consumer to click on a specific store listed within the canteen to browse that store.

REQ-8: The system shall display the store's detailed page showing all available food listings when the consumer clicks on the specific store.

4.2.5 View Food

4.2.5.1 Description and Priority

Description:

- Consumers can view the available food in NTU when they click on "Food" in the navigation bar

Priority:

- High

4.2.5.2 Stimulus/Response Sequences

1. The consumer clicks on "Food" in the navigation bar
2. The system displays the available food from the various stalls in NTU

4.2.5.3 Functional Requirements

REQ-1: The system shall display a "Food" link in the navigation bar, which is visible and accessible to the consumer.

REQ-2: The system shall display a list of available food items from various stalls in NTU when the consumer clicks on the "Food" link in the navigation bar, with clear details for each food item.

4.2.6 Search Food

4.2.6.1 Description and Priority

Description:

- Consumers can search for food by name or location of its respective canteen, and the relevant results shall be displayed

Priority:

- High

4.2.6.2 Stimulus/Response Sequences

1. The consumer clicks on "Food" in the navigation bar
2. The consumer clicks on the Search bar within the "Food" page
 - a. The consumer types in a food name
 - i. The system will show all stores that sell that particular food or food with matching names
 - b. The consumer types in the location of one of the food areas in NTU
 - i. The system will show the available food located in that particular food area

4.2.6.3 Functional Requirements

REQ-1: The system shall display a "Food" link in the navigation bar, which is visible and accessible to the consumer.

REQ-2: The system shall allow the consumer to click on the Search bar within the "Food" page to begin searching for food items.

REQ-2.1: The system shall allow the consumer to type in a food name in the Search bar on the "Food" page.

REQ-2.1.1: The system shall display all food items in NTU with matching names as the consumer types in the Search bar.

REQ-2.2: The system shall allow the consumer to type in the location of a food area in NTU in the Search bar.

REQ-2.2.1: The system shall display the available food items located in the specified food area when the consumer types the location in the Search bar.

4.2.7 Filter Food

4.2.7.1 Description and Priority

Description:

- Consumers can filter food by Cuisine, Type, Price or whether the food is on discount

Priority:

- High

4.2.7.2 Stimulus/Response Sequences

1. The consumer clicks on the "Food" button
2. The consumer clicks on the filter button
3. The consumer selects one of the filters or a combination of them

- a. If the consumer chooses to filter by Price, the system will display the food within the range that the consumer input
- b. If the user selects Type, the system will display the food of that specific type
- c. If the consumer selects a Cuisine, the system will display the food of that specific cuisine
- d. If the consumer selects Discount, the system will display all the food that are listed on a discount

4.2.7.3 Functional Requirements

REQ-1: The system shall display a "Food" button in the navigation bar, which is visible and accessible to the consumer.

REQ-2: The system shall allow the consumer to click on the "Filter" button on the "Food" page to access filtering options.

REQ-3: The system shall allow the consumer to select one or more filters from a list of available filtering options (e.g., Price, Type, Cuisine, Discount).

REQ-3.1: If the consumer selects the "Price" filter, the system shall allow the consumer to input a price range and display food items within that specified range.

REQ-3.2: If the consumer selects the "Type" filter, the system shall display food items of the selected type (e.g., vegetarian, non-vegetarian, vegan, etc.).

REQ-3.3: If the consumer selects the "Cuisine" filter, the system shall display food items of the selected cuisine (e.g., Italian, Chinese, Indian, etc.).

REQ-3.4: If the consumer selects the "Discount" filter, the system shall display all food items that are currently listed with a discount.

4.2.8 Submit Food Review

4.2.8.1 Description and Priority

Description:

- Consumers can submit their food review on the stall page

Priority:

- Medium

4.2.8.2 Stimulus/Response Sequences

1. The consumer clicks the "Leave a review" button
2. The system displays a form to be filled up
3. The user fills up relevant portions of the form and clicks "Submit Review"
4. The system displays a message saying "Review created!"
5. The system redirects the user to the relevant food page

4.2.8.3 Functional Requirements

REQ-1: The system shall display a "Leave a review" button on the relevant food page, which is visible and accessible to the consumer.

REQ-2: The system shall display a review form when the consumer clicks the "Leave a review" button. The form shall include fields for rating, comments, and any other relevant information.

REQ-3: The system shall allow the consumer to fill in the relevant portions of the form, including a rating and optional text comments, and then click the "Submit Review" button.

REQ-4: The system shall display a message saying "Review created!" after the consumer submits the review successfully.

REQ-5: The system shall redirect the consumer back to the relevant food page after the review is submitted successfully.

4.2.9 Update Food Review

4.2.9.1 Description and Priority

Description:

- Consumers can edit a review that they have previously submitted.

Priority:

- Medium

4.2.9.2 Stimulus/Response Sequences

1. The consumer clicks on "Account" in the navigation bar
2. The system displays the Account page
3. The consumer clicks on the specific review that he/she wants to update
4. The system displays a form with details of their submitted review
5. The consumer edits relevant portions of the form and clicks the "Update Review" button
6. The system displays a message saying "Review Updated"
7. The system redirects the user to the Reviews page

4.2.9.3 Functional Requirements

REQ-1: The system shall display an "Account" link in the navigation bar, which is visible and accessible to the consumer.

REQ-2: The system shall display the Account page when the consumer clicks on the "Account" link in the navigation bar.

REQ-3: The system shall allow the consumer to click on a specific review they want to update from their list of reviews on the Account page.

REQ-4: The system shall display a form pre-filled with the details of the selected review when the consumer clicks on the specific review to update.

REQ-5: The system shall allow the consumer to edit the relevant portions of the form, such as the rating or text comments, and click the "Update Review" button to submit the changes.

REQ-6: The system shall display a message saying "Review Updated" after the consumer successfully updates the review.

REQ-7: The system shall redirect the consumer to the Reviews page after the review is successfully updated.

4.2.10 Delete Food Review

4.2.10.1 Description and Priority

Description:

- Consumers are able to delete a review they have submitted. Admins are able to delete any review that has been flagged

Priority:

- Medium

4.2.10.2 Stimulus/Response Sequences

1. The consumer clicks on "Account" in the navigation bar
2. The system displays the Account page
3. The consumer clicks on the specific review that he/she wants to delete
4. The system displays a form with details of their submitted review
5. The consumer clicks on the "X" button
6. The system displays a confirmation message saying "Are you sure you want to this review?"
7. The consumer clicks on the "Delete" button
8. The system redirects the consumer back to the Account page and displays a message saying "Review Deleted!"

4.2.10.3 Functional Requirements

REQ-1: The system shall display an "Account" link in the navigation bar, which is visible and accessible to the consumer.

REQ-2: The system shall display the Account page when the consumer clicks on the "Account" link in the navigation bar.

REQ-3: The system shall allow the consumer to click on a specific review they want to delete from their list of reviews on the Account page.

REQ-4: The system shall display a form pre-filled with the details of the selected review when the consumer clicks on the specific review to delete.

REQ-5: The system shall display an "X" button next to the review, which the consumer can click to initiate the deletion process.

REQ-6: The system shall display a confirmation message saying "Are you sure you want to delete this review?" when the consumer clicks the "X" button.

REQ-7: The system shall allow the consumer to click the "Delete" button in the confirmation message to proceed with deleting the review.

REQ-8: The system shall redirect the consumer back to the Account page after the review is deleted and display a message saying "Review Deleted!"

4.2.11 Flag Food Review

4.2.11.1 Description and Priority

Description:

- Consumers can flag a food review that they deem inappropriate, highlighting it for the admin to review

Priority:

- Medium

4.2.11.2 Stimulus/Response Sequences

1. The consumer clicks on the "Flag" button on a previously submitted review by another consumer
2. The system displays a form that shows that ask the consumer for the reason for flagging the review
3. The consumer fills up the relevant portion of the form and clicks the "Submit" button
4. The system sends a notification to the system admin

4.2.11.3 Functional Requirements

REQ-1: The system shall display a "Flag" button next to each previously submitted review by another consumer, which is visible and accessible to the consumer.

REQ-2: The system shall display a form when the consumer clicks the "Flag" button, asking the consumer to provide a reason for flagging the review (e.g., inappropriate content, misleading information, etc.).

REQ-3: The system shall allow the consumer to fill in the relevant portion of the form with their reason for flagging the review.

REQ-4: The system shall provide a "Submit" button for the consumer to click after filling in the form.

REQ-5: The system shall send a notification to the system admin once the consumer submits the flagging form, alerting them of the flagged review for further investigation.

4.3 Store Owner Features

4.3.1 Submit Food Listing

4.3.1.1 Description and Priority

Description:

- Store Owners can submit food listings onto the app for consumers to view.

Priority:

- High

4.3.1.2 Stimulus/Response Sequences

1. The store owner clicks on the "+" button
2. The store owner enters the details (name, price, type, cuisine, description, image) of the food
3. The store owner clicks on the "Add Food" button
4. The system displays a message saying "Listing Added Successfully!"

4.3.1.3 Functional Requirements

REQ-1: The system shall display a "+" button, visible and accessible to the store owner, which allows them to add a new food listing.

REQ-2: The system shall allow the store owner to enter the details of the food item, including the name, price, type, cuisine, description, and an image, in the provided input fields.

REQ-3: The system shall provide an "Add Food" button that the store owner can click to submit the food listing.

REQ-4: The system shall display a message saying "Listing Added Successfully!" after the store owner successfully submits the food listing.

4.3.2 Update Food Listing

4.3.2.1 Description and Priority

Description:

- Store Owners can update their submitted food listings on the app, changing details such as name, price, type, cuisine, description, picture and discount (if any).

Priority:

- High

4.3.2.2 Stimulus/Response Sequences

1. The store owner clicks on the specific food listing to update
2. The system displays the food listing page
3. The store owner enters the new input on the relevant fields
4. The store owner clicks on the "Update Listing" button
5. The system displays a message saying "Listing Updated Successfully"

4.3.2.3 Functional Requirements

REQ-1: The system shall allow the store owner to click on a specific food listing to update it from the list of existing food items.

REQ-2: The system shall display the food listing page with the details of the selected food item, allowing the store owner to view and edit its information.

REQ-3: The system shall allow the store owner to edit the relevant fields (e.g., name, price, type, cuisine, description, image) of the food listing.

REQ-4: The system shall provide an "Update Listing" button for the store owner to click after making the necessary changes to the food listing.

REQ-5: The system shall display a message saying "Listing Updated Successfully" after the store owner successfully updates the food listing.

4.3.3 Delete Food Listing

4.3.3.1 Description and Priority

Description:

- Store Owners are able to delete their submitted food listings from the system.

Priority:

- Medium

4.3.3.2 Stimulus/Response Sequences

1. The store owner clicks on the specific food listing to delete
2. The system displays the food listing page
3. The store owner clicks on "X" button
4. The system displays a message saying "Are you sure you want to delete this listing?"
5. The store owner clicks on the "Delete" button
6. The system displays a message saying "Listing Deleted!"

4.3.3.3 Functional Requirements

REQ-1: The system shall allow the store owner to click on a specific food listing to delete it from the list of existing food items.

REQ-2: The system shall display the food listing page with the details of the selected food item when the store owner clicks on it.

REQ-3: The system shall provide an "X" button on the food listing page, which the store owner can click to initiate the deletion process.

REQ-4: The system shall display a confirmation message saying "Are you sure you want to delete this listing?" after the store owner clicks the "X" button.

REQ-5: The system shall allow the store owner to click the "Delete" button in the confirmation message to proceed with deleting the food listing.

REQ-6: The system shall display a message saying "Listing Deleted!" after the store owner successfully deletes the food listing.

4.4 Admin Features

4.4.1 Process Review

4.4.1.1 Description and Priority

Description:

- Admins are able to process the food reviews flagged by consumers, deleting them or marking them as processed accordingly.

Priority:

- Medium

4.4.1.2 Stimulus/Response Sequences

1. The system admin clicks on the "Process Reviews" button
2. The system displays all the flagged reviews with the most recent ones at the top
 - a. The system admin clicks on "Delete Review" button
 - i. The system displays a message saying "Are you sure you want to delete the review?"
 - ii. The system admin clicks the "Delete" button
 - iii. The system displays a message saying "Review deleted and user notified"
 - iv. The system sends a notification to the consumer saying that their review has been deleted
 - b. The system admin clicks on "Mark as Processed" button
 - i. The system displays a message saying "Flagged review marked as processed."
 - ii. The system removes the flagged review from the list and sends a notification to the consumer that flagged the review, saying that the review that he/she flagged has been processed

4.4.1.3 Functional Requirements

REQ-1: The system shall display a "Process Reviews" button, visible and accessible to the system admin.

REQ-2: The system shall display all flagged reviews to the system admin when the "Process Reviews" button is clicked, with the most recent flagged reviews at the top of the list.

REQ-3.1: The system shall allow the system admin to click on a "Delete Review" button next to a flagged review to initiate the deletion process.

REQ-3.1.1: The system shall display a confirmation message saying "Are you sure you want to delete the review?" after the system admin clicks the "Delete Review" button.

REQ-3.1.2: The system shall allow the system admin to click the "Delete" button in the confirmation message to proceed with deleting the review.

REQ-3.1.3: The system shall display a message saying "Review deleted and user notified" after the review is successfully deleted.

REQ-3.1.4: The system shall send a notification to the consumer who submitted the deleted review, informing them that their review has been deleted.

REQ-3.2: The system shall allow the system admin to click on a "Mark as Processed" button next to a flagged review to mark it as processed.

REQ-3.2.1: The system shall display a message saying "Flagged review marked as processed" after the system admin clicks the "Mark as Processed" button.

REQ-3.2.2: The system shall remove the flagged review from the list of reviews and send a notification to the consumer who flagged the review, informing them that the review they flagged has been processed.

4.4.2 Ban User

4.4.2.1 Description and Priority

Description:

- Admins can ban users such that they are not able to login to the app.

Priority:

- Medium

4.4.2.2 Stimulus/Response Sequences

1. The system admin clicks on the "Ban User" button
2. The system displays the user list (excluding store owners) for the system admin to ban
3. The system admin clicks on "Ban" button of next to the target account
4. The system displays a message saying "Are you sure you want to ban this user?"
5. The system admin clicks on the "Ban" button
6. The system displays a message saying "User banned!"
7. The system redirects the system admin to the page that displays the user list

4.4.2.3 Functional Requirements

REQ-1: The system shall display a "Ban User" button, visible and accessible to the system admin.

REQ-2: The system shall display a list of users (excluding store owners) when the system admin clicks the "Ban User" button. The user list shall include relevant details such as usernames and account types.

REQ-3: The system shall allow the system admin to click on a "Ban" button next to a target user's account in the user list.

REQ-4: The system shall display a confirmation message saying "Are you sure you want to ban this user?" after the system admin clicks the "Ban" button.

REQ-5: The system shall allow the system admin to click the "Ban" button in the confirmation message to proceed with banning the user.

REQ-6: The system shall display a message saying "User banned!" after the system admin successfully bans the user.

REQ-7: The system shall redirect the system admin to the user list page after the user is banned.

4.4.3 Notify User

4.4.3.1 Description and Priority

Description:

- Admins can send notifications to users directly.

Priority:

- Medium

4.4.3.2 Stimulus/Response Sequences

1. The system admin clicks on "Notifications" table in the dashboard
2. The system displays the Notification page
3. The system admin clicks on the "Add Notification +" button
4. The system displays a form for the system admin to field in with fields such as title, description, and the target user
5. The system admin fills in the relevant fields on the Notification form
6. The system admin clicks the "Save" button
7. The system displays a message saying "The notification was added successfully"
8. The system sends a notification to the target user
9. The system redirects the system admin back to the Notification page

4.4.3.3 Functional Requirements

REQ-1: The system shall display a "Notifications" table on the dashboard, which is visible and accessible to the system admin.

REQ-2: The system shall display the Notification page when the system admin clicks on the "Notifications" table in the dashboard.

REQ-3: The system shall display an "Add Notification +" button on the Notification page, visible and accessible to the system admin.

REQ-4: The system shall display a form when the system admin clicks the "Add Notification +" button. The form shall include fields such as title, description, and target user.

REQ-5: The system shall allow the system admin to fill in the relevant fields on the Notification form, including the title, description, and the target user.

REQ-6: The system shall provide a "Save" button for the system admin to click after filling in the Notification form.

REQ-7: The system shall display a message saying "The notification was added successfully" after the system admin successfully submits the Notification form.

REQ-8: The system shall send a notification to the target user as specified in the Notification form after the system admin adds the notification.

REQ-9: The system shall redirect the system admin back to the Notification page after the notification is successfully added.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- **Loading Time:** Each page should load in no more than 4 seconds, including logging in and authentication.
- **Response Time:** The app should respond to user actions within 2 seconds
- **Scalability:** The app should be scalable enough to support at least 10,000 concurrent users without performance degradation.
- **Throughput:** The system should process up to 100 actions per second.
- **Availability:** The app should have an uptime of 99.9%.
- **Backup:** Data should be backed up every 24 hours.
- **Recovery:** The system should recover from failures within 1 hour.
- **Error Rate:** The total error rate of store owners submitting food listings, consumers submitting food reviews, and admin actions, should be kept below 1%. Also, errors that could arise from synchronization issues where, for example, two or more consumers flag the same inappropriate review left on a specific food listing should be kept to a maximum of no more than 200 times per month.

5.2 Safety Requirements

- The application shall not interfere with other device functions, such as emergency calls.
- User location data shall only be used when explicitly granted permission by the user.

5.3 Security Requirements

- Authentication: Users must authenticate using a secure method.
- Data Encryption: All sensitive data should be encrypted both in transit and at rest.
- Access Control: Different user roles should have appropriate access controls.
- Data Protection: The app must comply with relevant data protection regulations, e.g. Singapore PDPA.
- Audit Logs: All user actions should be logged for auditing purposes.

5.4 Software Quality Attributes

- User Interface: The UI should be intuitive and easy to navigate for all users.
- Accessibility: The app should be accessible to users with disabilities, following WCAG 2.1 guidelines.
- Localization: The app should support the English language.
- Code Quality: The codebase should follow best practices and be well-documented to facilitate maintenance.

- Modularity: The system should be modular to allow for easy updates and feature additions.
- Data Access Standards: The app should expose data endpoints using Django views and URL patterns, enabling compatibility with other systems. Although not fully RESTful, the structured endpoints will allow for data retrieval, creation, updating, and deletion as needed.
- Data Formats: The app should support common data formats like JSON and XML for data exchange.
- Supported Platforms: The app should be compatible with major operating systems (iOS, Android, Windows, macOS).
- Browser Compatibility: The web version should support major browsers (Chrome, Firefox, Safari, Edge).

5.5 Business Rules

- Users must agree to terms of service and practice proper conduct before submitting reviews or food listings.

6. Other Requirements

Appendix A: Glossary

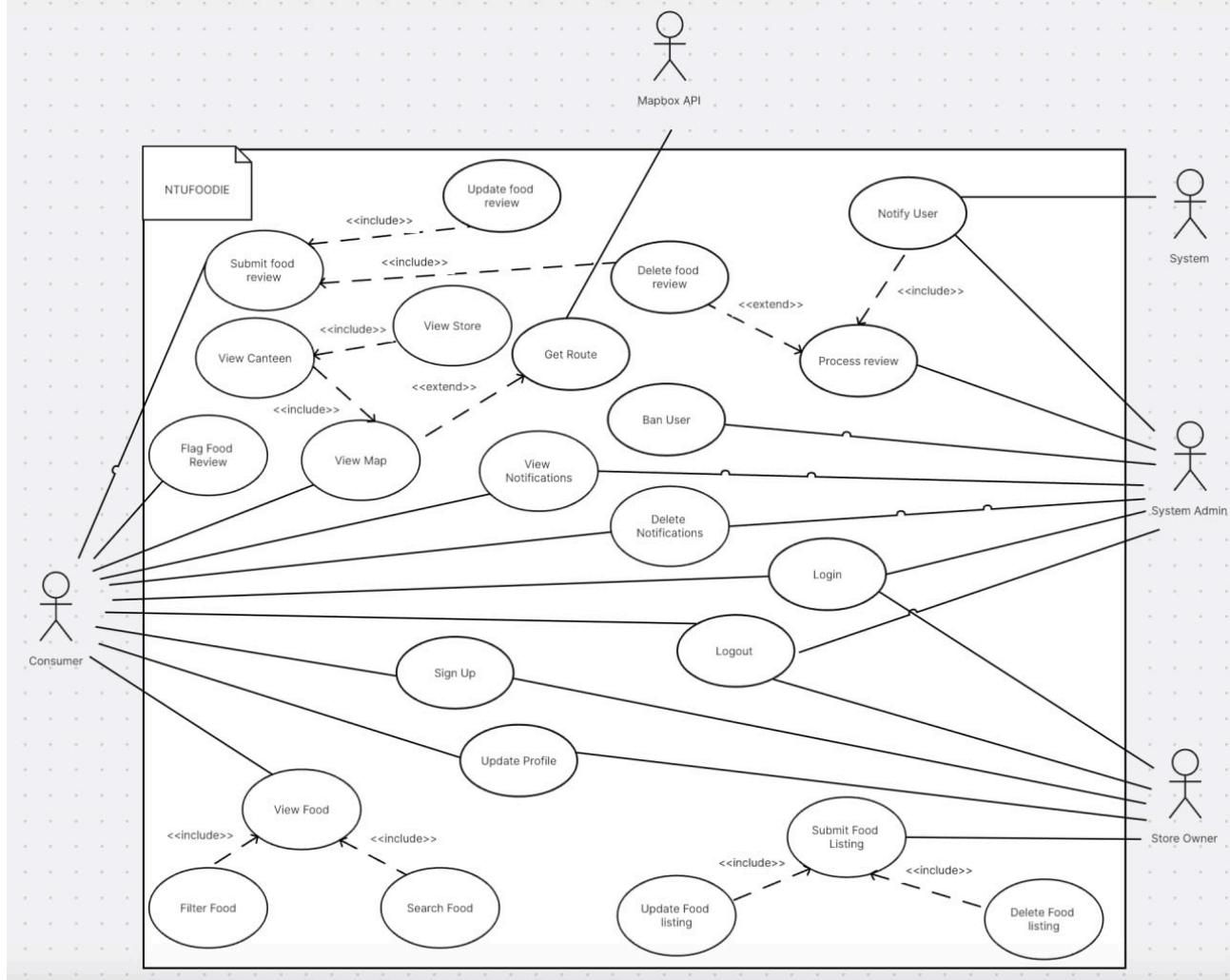
Admin	A system administrator responsible for managing user reviews, banning inappropriate users, and approving food reviews for publication.
Ban	A temporary suspension of a user's account due to inappropriate reviews. Users will not be able to access the application for the duration of the ban.
Canteen	A food court/a collection of some stalls in a specific place in NTU, e.g. Canteen 11, Canteen 2, etc.
Consumer	User account that belongs to a generic user (not store owner or admin) to perform food-related queries
Display Name	A name chosen by the user for public display within the application. If not set, the system uses the user's registered name by default.
Email Address	A unique identifier that users must provide during account registration. The system uses this for user authentication and communication.
Food Cuisines	A category of food based on its country or region of origin. This will enable the user to filter the food based on what they want at the time of using our application.
Food Filter	A feature that allows users to filter food options based on price, dietary restrictions, or cuisines.
Food Item	A dish or meal option available for search or review within the NTU campus.
Food Listing	A feature that allows hawkers to list food items at a discount for students to purchase within a set timeframe.
Food Review	User-submitted evaluation of a food item, including a rating (1 to 5, up to 1 decimal place), optional photo, and description. Reviews are processed and verified by an admin before posting.
Food Types	A filter option to categorise food based on specific dietary needs, such as Halal, Vegetarian, or Vegan.
Inappropriate Review	A review flagged by the admin for inappropriate content, such as offensive language or unreasonable ratings, which may result in banning the user.

Password	A string of characters required for user authentication. It must contain at least 8 characters, including at least 1 number.
Profile Picture	An optional image file uploaded by the user to personalise their account. If no image is uploaded, a default profile picture is displayed.
Review Description	A textual user input detailing the user's experience with a specific food item, including taste, quality, and service.
Review Process	A system feature where an admin checks and either approves or revokes reviews before they are published on the platform.
Review Rating	A numerical score from 1 to 5 representing the overall satisfaction of a food item, store, or service.
Stall	A specific F&B establishment that sells food, where the User can give a review on the food that the stall sells. The owner of this Stall is Store Owner. Stall is also commonly known as Hawker.
Store Owner	User account that belongs to the owner of a food stall.
User Account	A registered entity in the system that allows users to access the app. Users must provide an email, password, and name to create an account.

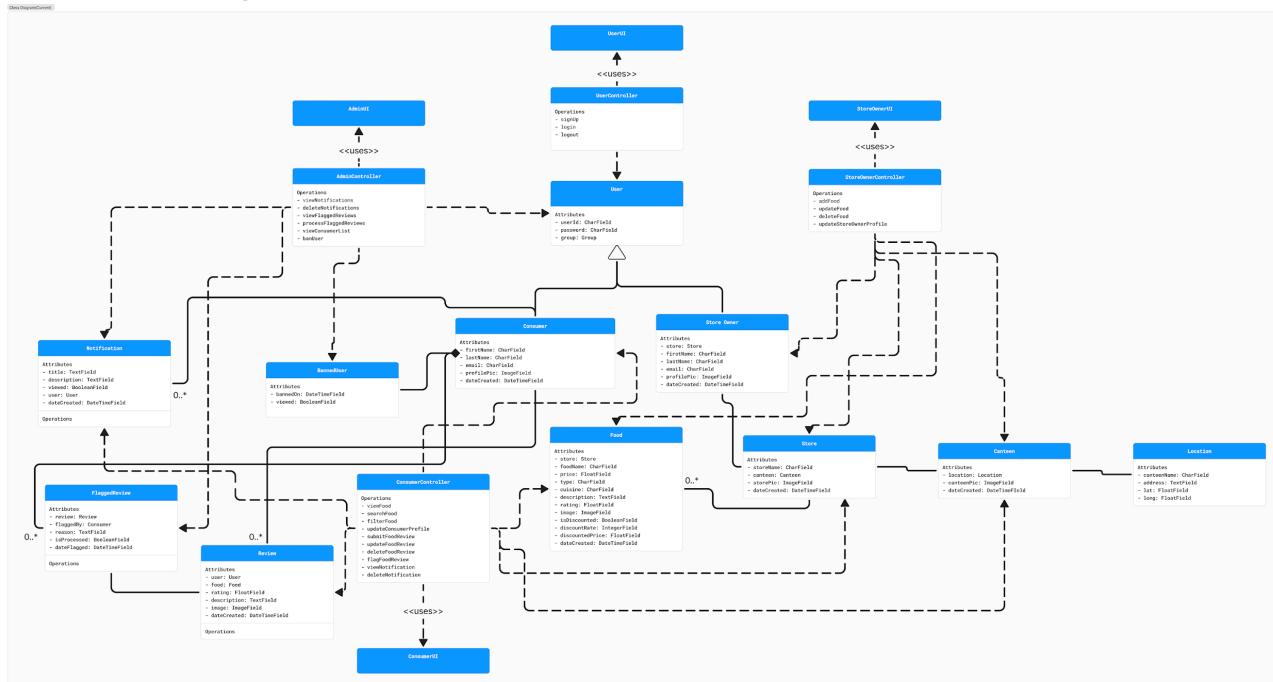
Appendix B: Analysis Models

*If an image is unclear, please refer to the GitHub repository, under 2006-TDBB-49/Lab 5 Deliverables/Supporting Documents and Diagrams

B1: Use Case Diagram

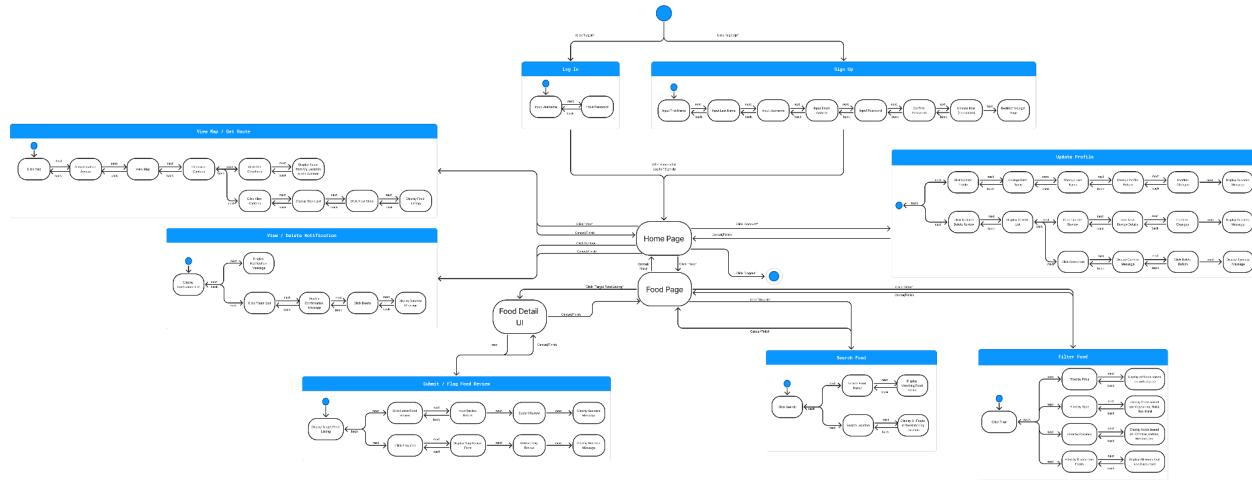


B2: Class Diagram

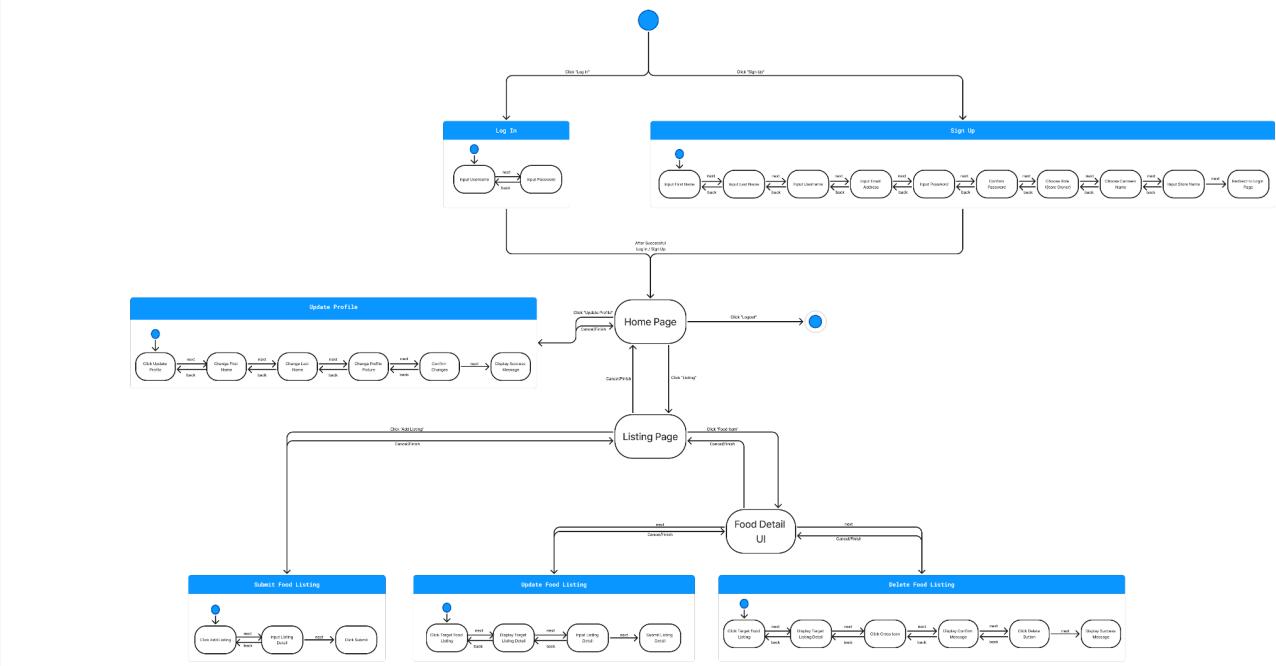


B3: Dialog Map

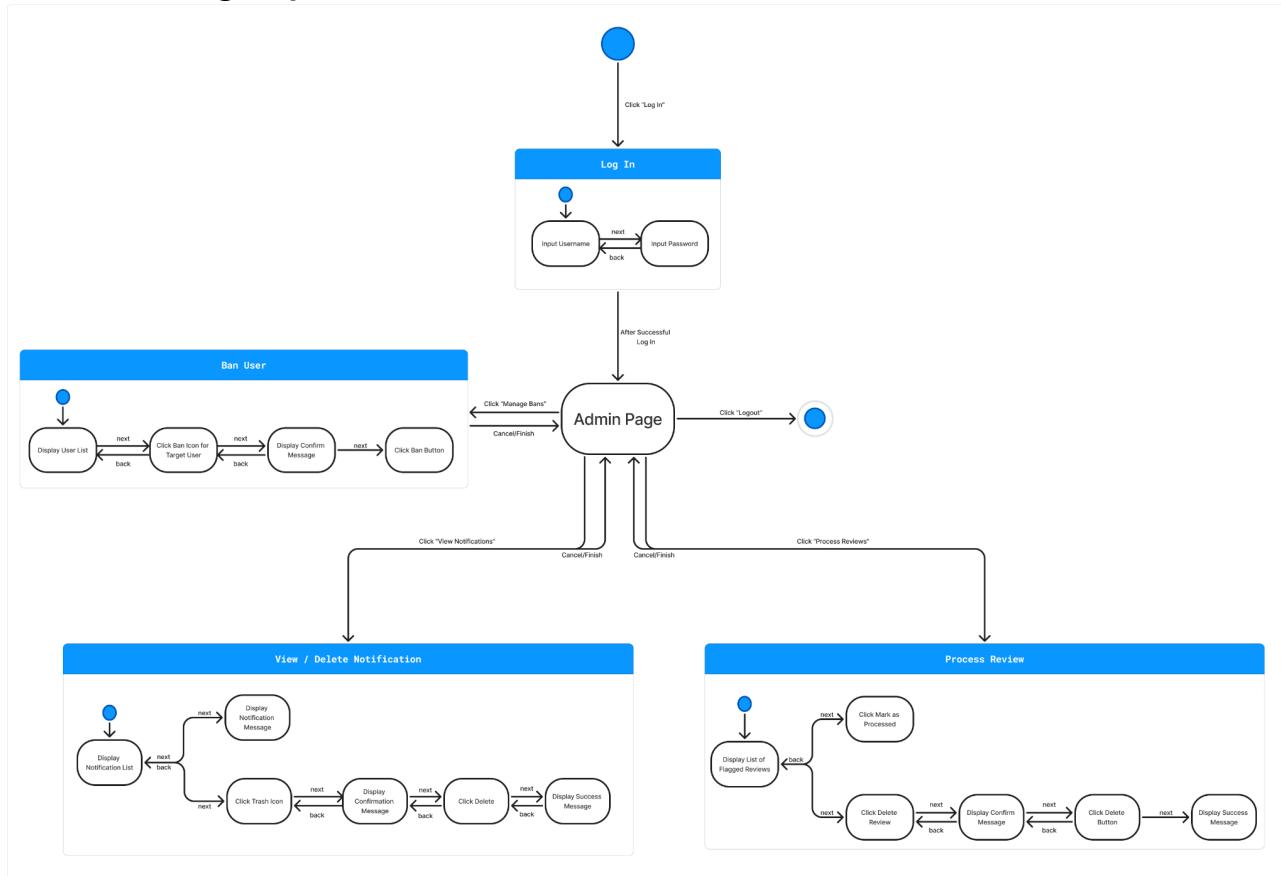
Consumer's Dialog Map



Store Owner's Dialog Map

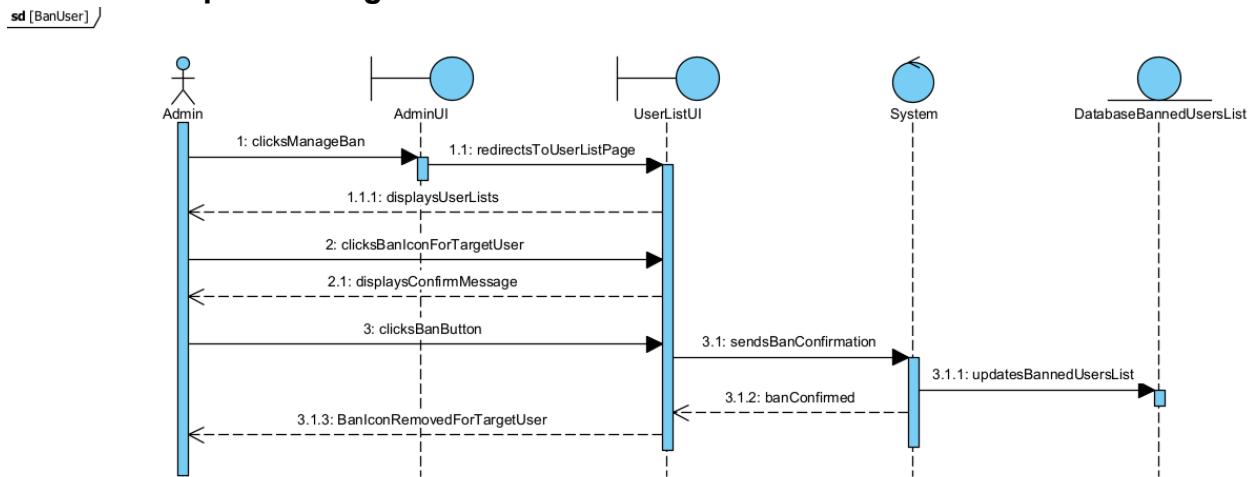


Admin's Dialog Map

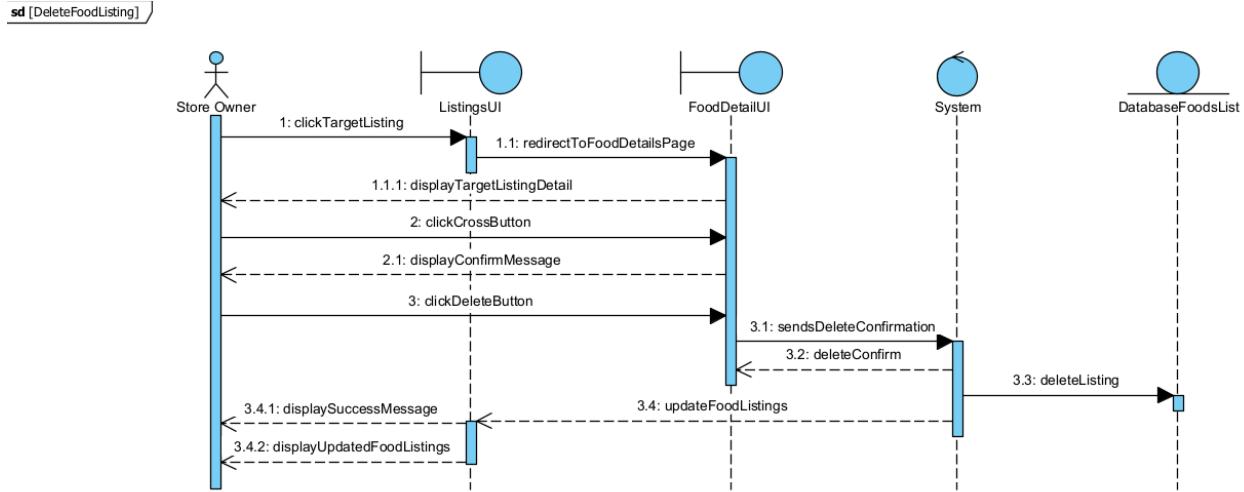


B4: Sequence Diagrams

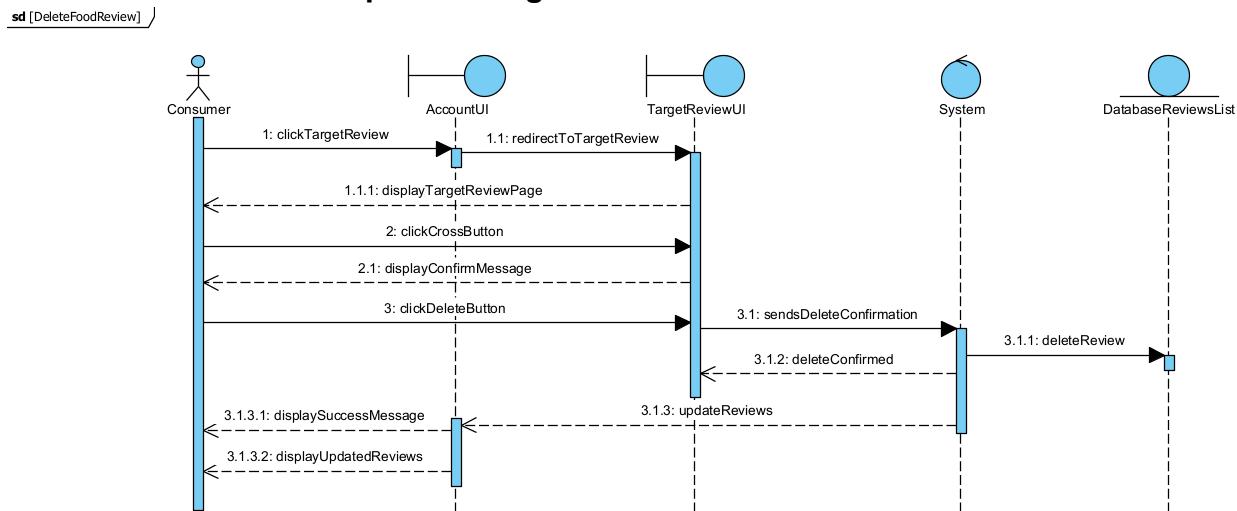
Ban User Sequence Diagram



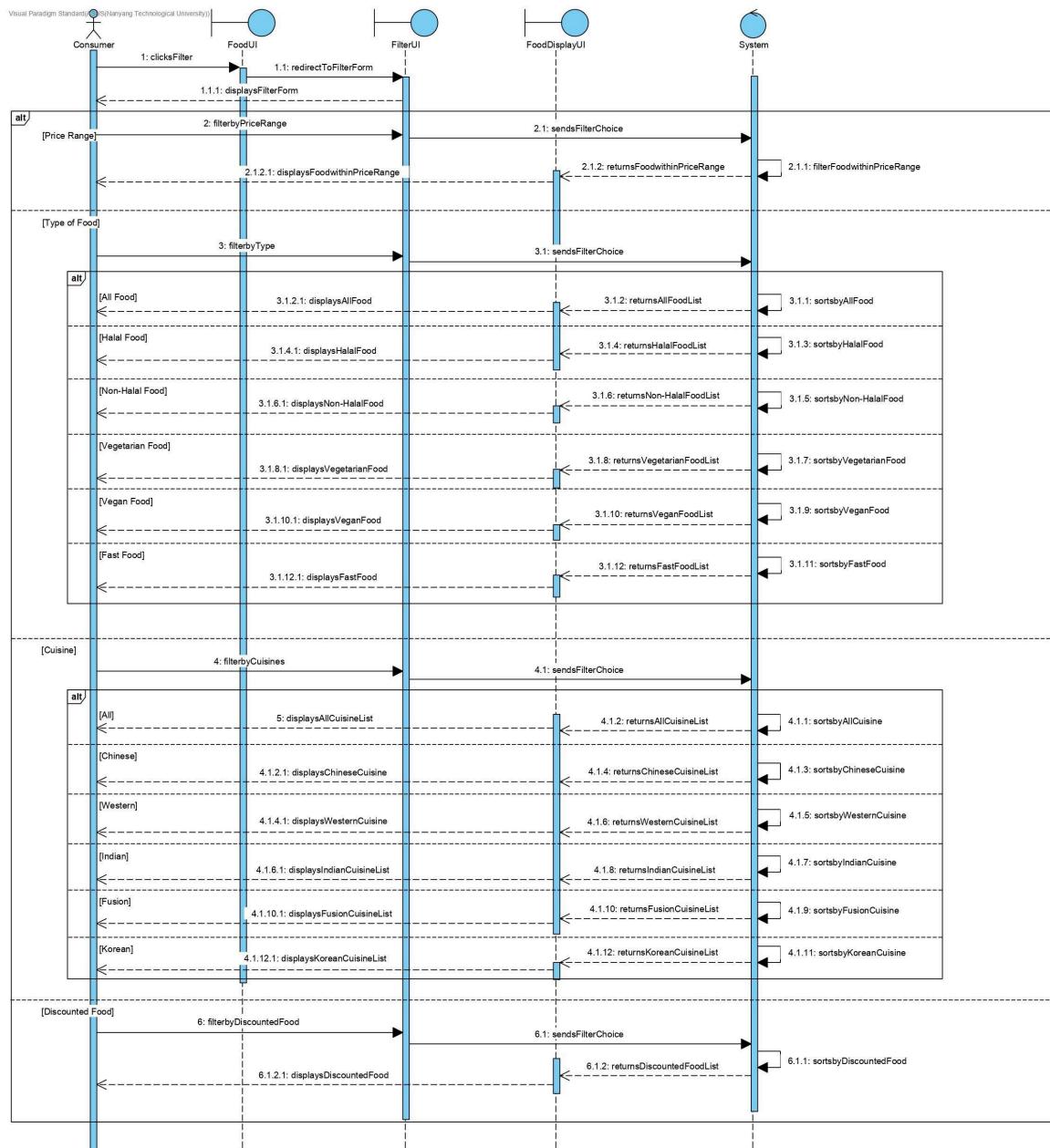
Delete Food Listing Sequence Diagram



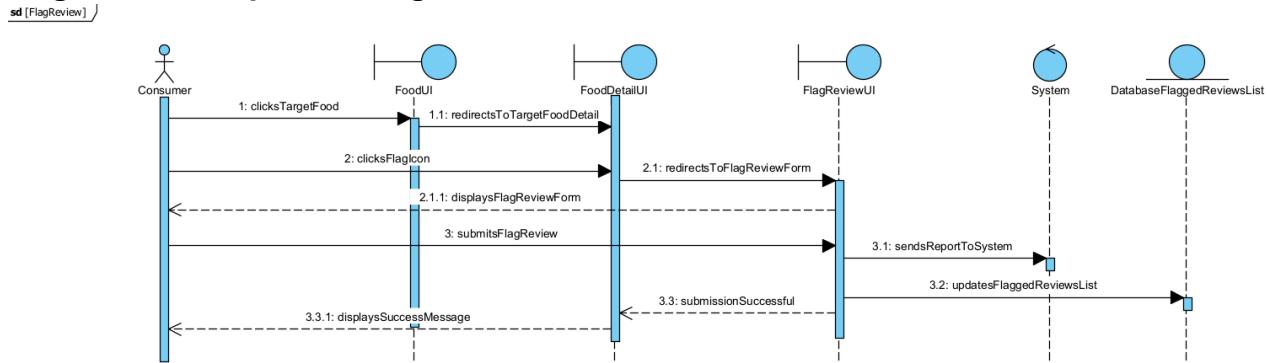
Delete Food Review Sequence Diagram



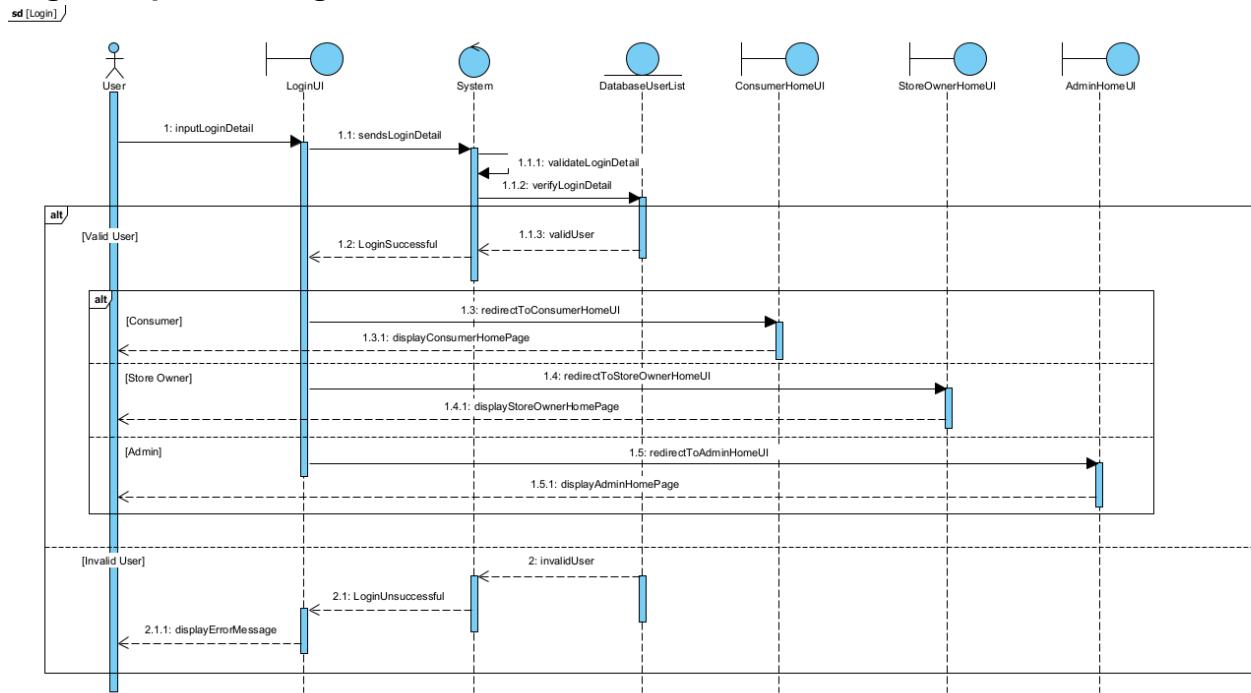
Filter Food Sequence Diagram



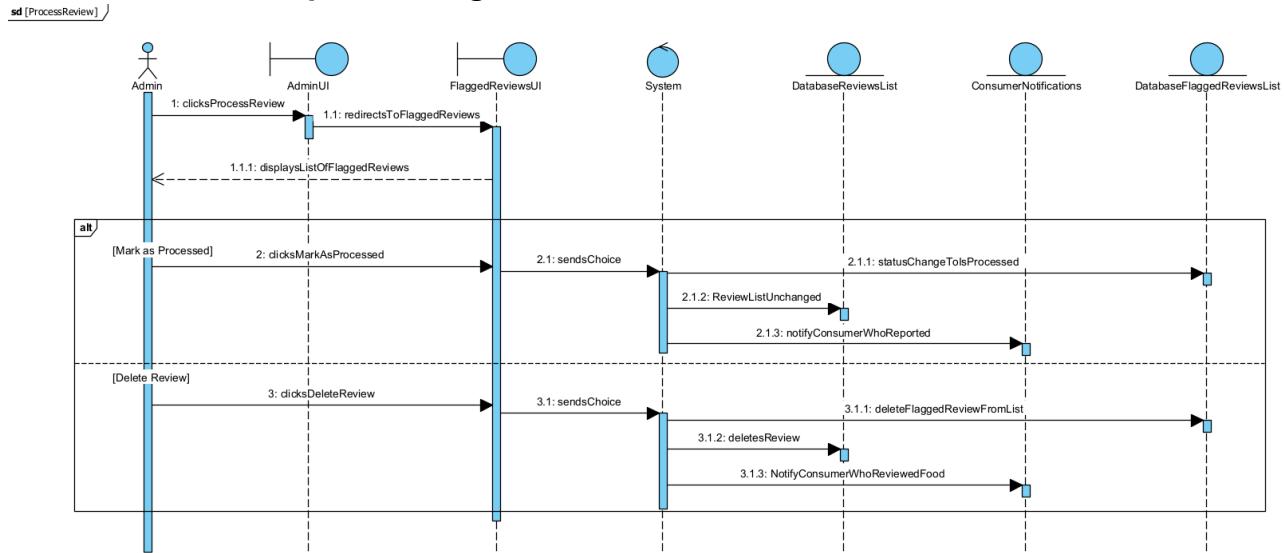
Flag Review Sequence Diagram



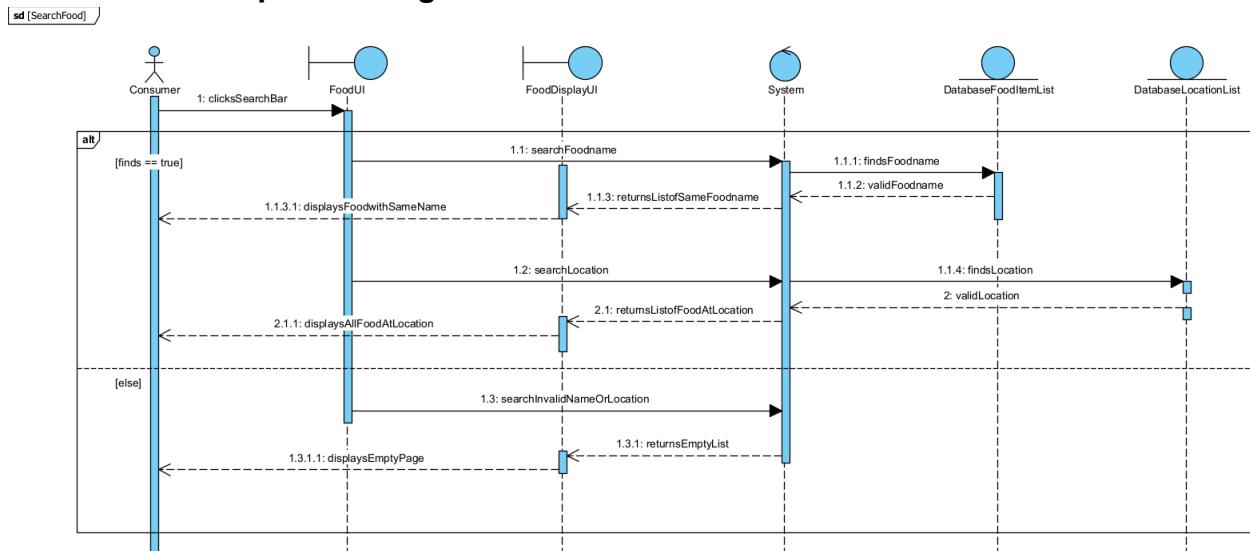
Login Sequence Diagram



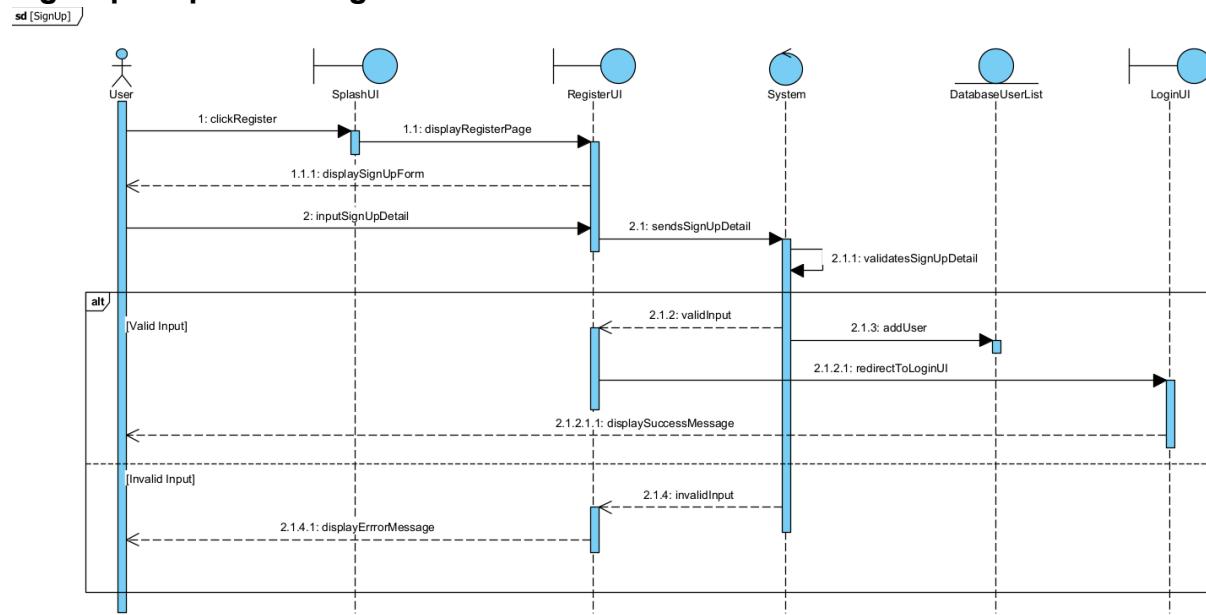
Process Review Sequence Diagram



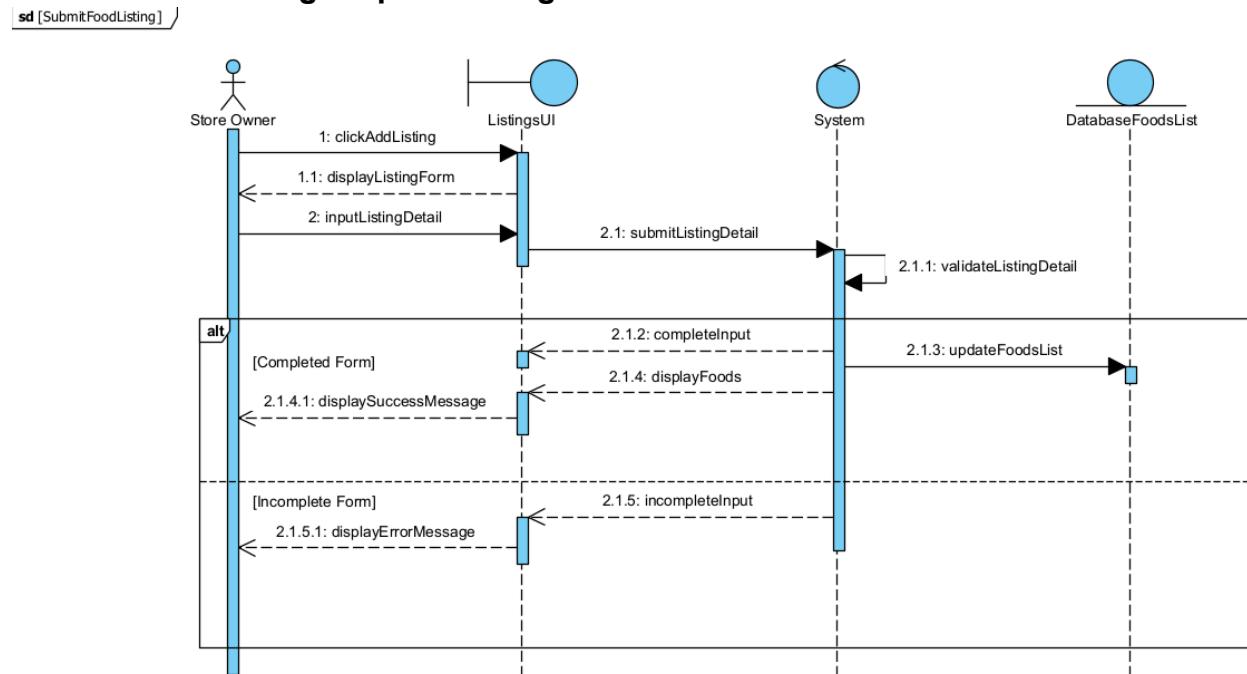
Search Food Sequence Diagram



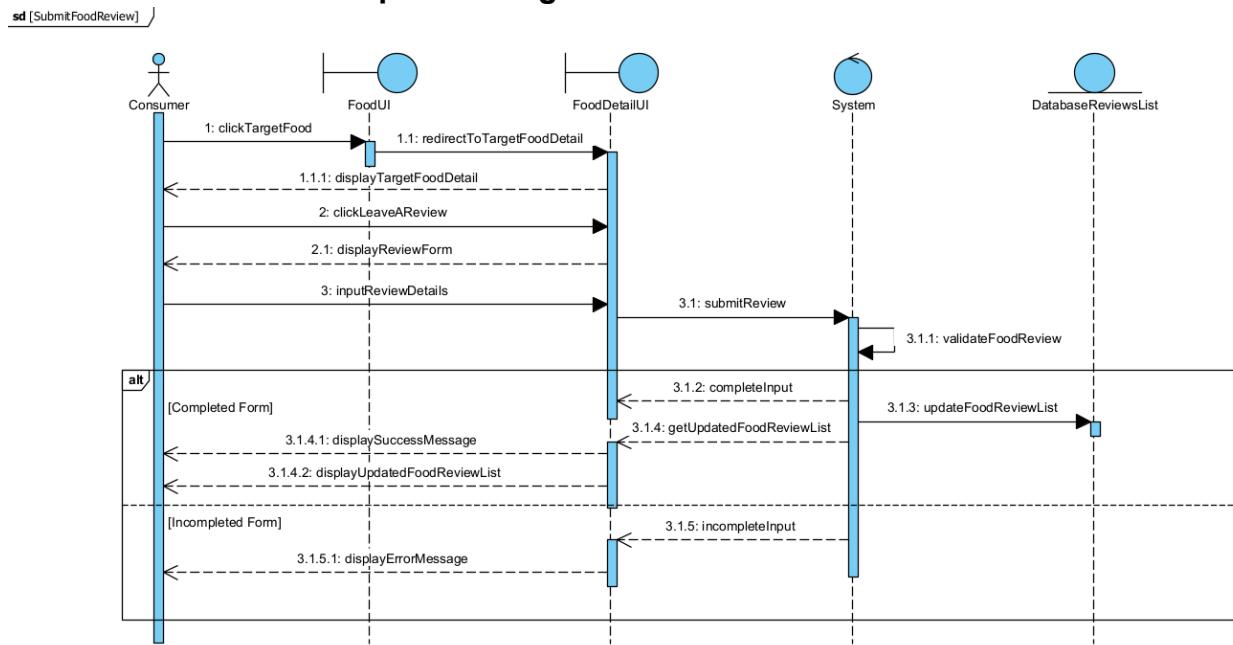
Sign Up Sequence Diagram



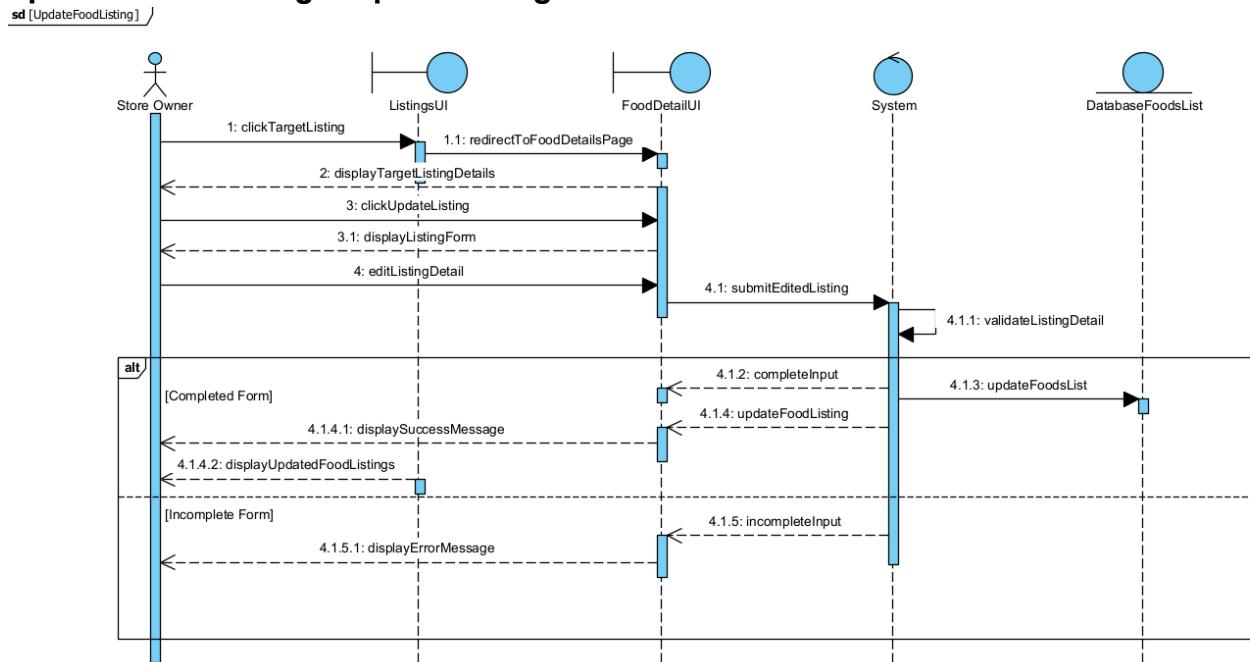
Submit Food Listing Sequence Diagram



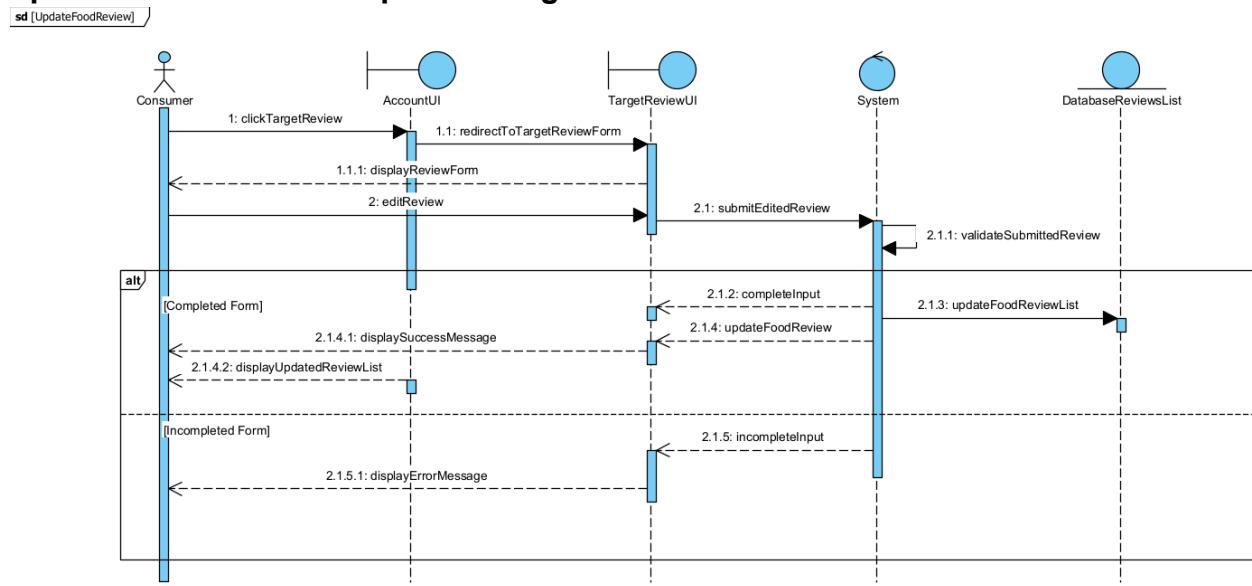
Submit Food Review Sequence Diagram



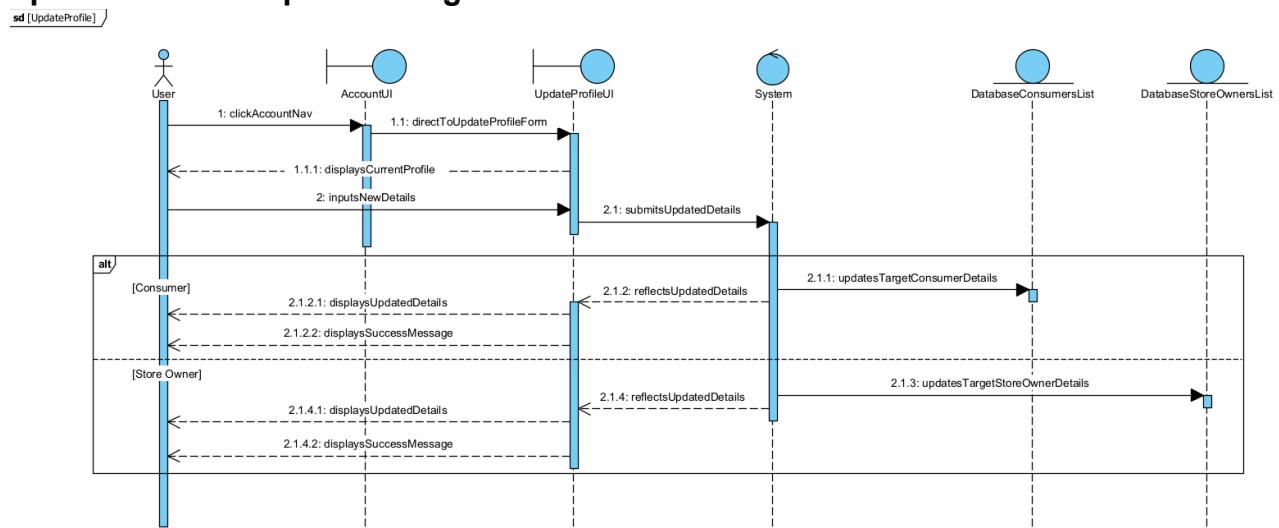
Update Food Listing Sequence Diagram



Update Food Review Sequence Diagram

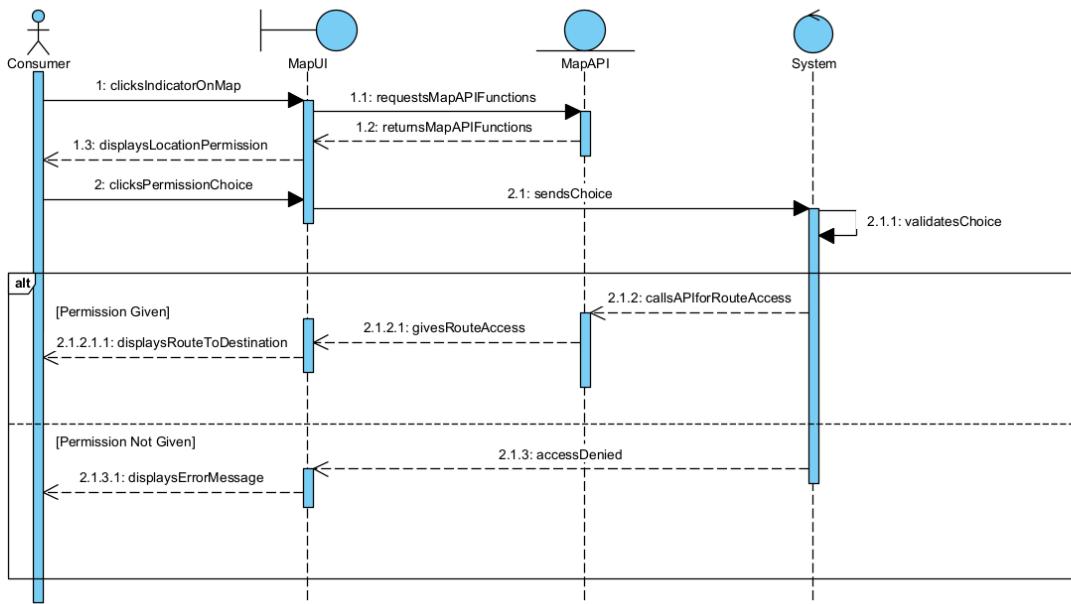


Update Profile Sequence Diagram

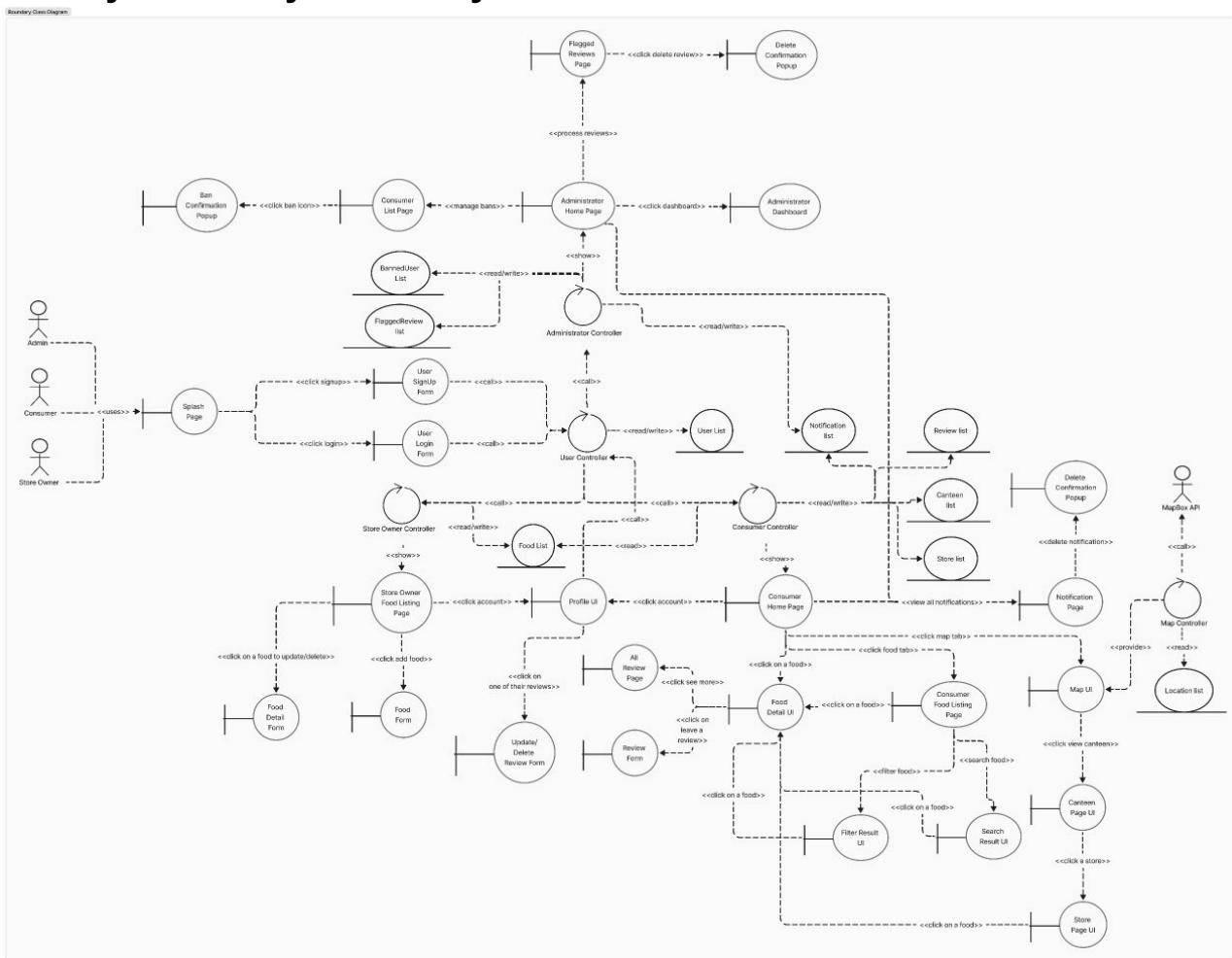


View Map Sequence Diagram

sd [ViewMap]



B5: Key Boundary and Entity Classes



B6: Architecture Diagram

