

Use Case Diagram and Use Case Descriptions

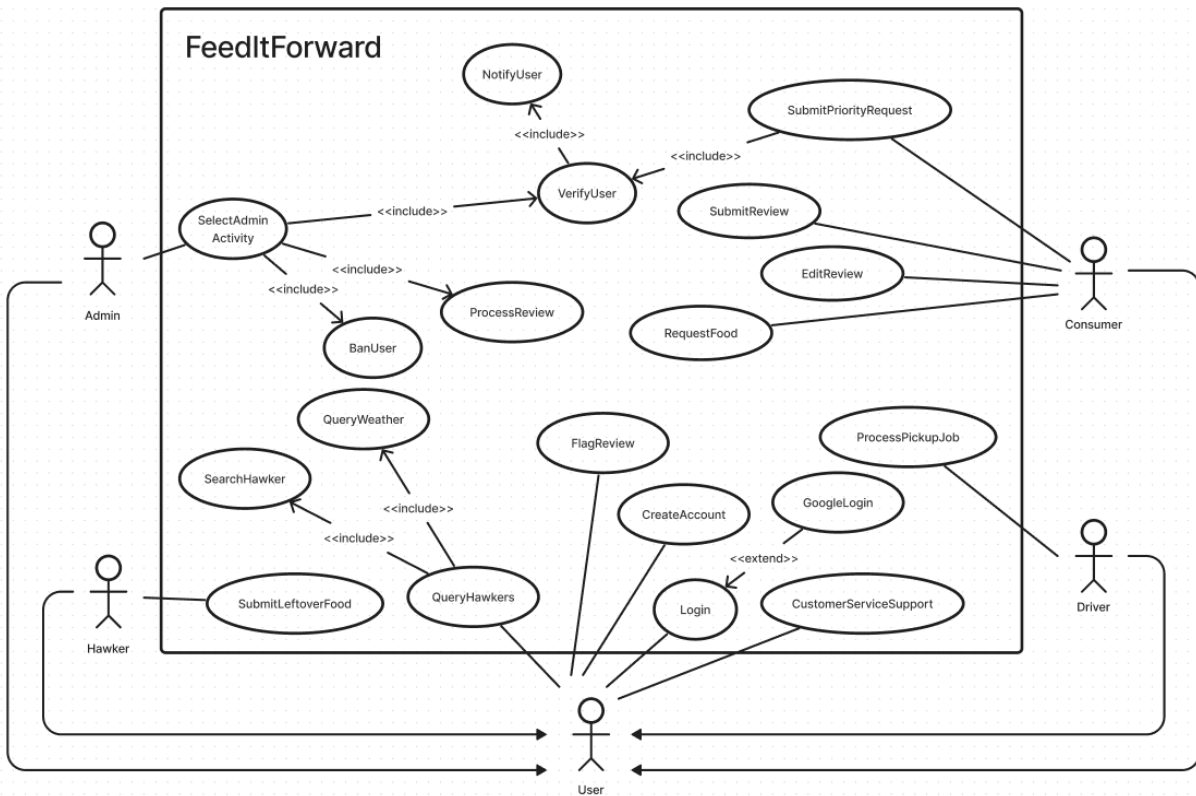
Table of Contents

Use Case Model and Use Case Description	3
A. Use Case Diagram	3
B. Use Case Descriptions	4
I. For Functional Requirement #1 (Admin)	4
I.I SelectAdminActivity	4
I.II BanUser	5
I.III VerifyUser	6
I.IV ProcessReview	8
I.V NotifyUser	9
II. For Functional Requirement #2 (Hawker)	10
II.I SubmitLeftoverFood	10
III. For Functional Requirement #3 (Consumer)	12
III.I SubmitReview	12
III.II EditReview	13
III.III RequestFood	15
III.IV SubmitPriorityRequest	17
IV. For Functional Requirement #4 (Driver)	18
IV.I ProcessPickupJob	18
V. For Functional Requirement #5	20
V.I CustomerServiceSupport	20
VI. For Functional Requirement #6 (User)	21
VI.I QueryHawkers	21
VI.II SearchHawkers	23
VI.III QueryWeather	24
VI.IV FlagReview	25
VII. For Functional Requirement #7 (Authentication)	26
VII.I CreateAccount	26
VII.II Login	27
VII.III GoogleLogin	28

Use Case Model and Use Case Description

A. Use Case Diagram

If the image is unclear, please refer to the raw png file that is uploaded together with this document.



B. Use Case Descriptions

I. For Functional Requirement #1 (Admin)

I.I SelectAdminActivity

Use Case ID:	#1-1		
Use Case Name:	SelectAdminActivity		
Created By:	Toh Jing Hua	Last Updated By:	Toh Jing Hua
Date Created:	10th September 2023	Date Last Updated:	10th September 2023

Actor:	Admin
Description:	Allows Admin to select actions to take
Preconditions:	Admin is logged in and is authenticated
Postconditions:	None
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. The system uses the included use case Login to verify the Admin.2. On successful login, the system prompts the Admin to select the desired activity: BAN, VERIFYUSER, REVIEW, or QUIT.3. If the Admin selects the activity BAN, then the Admin uses the included use case BanUser to ban a user.4. If the Admin selects the activity VERIFYUSER, then the Admin uses the included use case VerifyUser to approve or reject a user's application for priority.5. If the Admin selects the activity REVIEW, then the Admin uses the included use case ProcessReview to decide whether a review is to be removed.6. If the Admin selects the activity QUIT, the system returns to the login screen.
Alternative Flows:	None
Exceptions:	None
Includes:	<ol style="list-style-type: none">1. BanUser2. VerifyUser3. ProcessReview
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

I.II BanUser

Use Case ID:	#1-2		
Use Case Name:	BanUser		
Created By:	Toh Jing Hua	Last Updated By:	Toh Jing Hua
Date Created:	10th September 2023	Date Last Updated:	10th September 2023

Actor:	Admin
Description:	Allows Admin to ban a User by their user ID for a specified duration
Preconditions:	Admin is logged in and is authenticated
Postconditions:	User is either banned or not banned by the admin
Priority:	High
Frequency of Use:	Low
Flow of Events:	<ol style="list-style-type: none"> Admin inputs the user ID of the user that the Admin wants to ban. Admin inputs the duration of the ban. Admin confirms the ban by selecting CONFIRM.
Alternative Flows:	AF-S3: If Admin selects the CANCEL button <ol style="list-style-type: none"> The system will return to the SelectAdminActivity.
Exceptions:	<ol style="list-style-type: none"> If the user ID is not found, the system shall display an error message and prompt the Admin again.
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

I.III VerifyUser

Use Case ID:	#1-3		
Use Case Name:	VerifyUser		
Created By:	Toh Jing Hua	Last Updated By:	Toh Jing Qiang
Date Created:	10th September 2023	Date Last Updated:	8th November 2023

Actor:	Admin
Description:	Allows Admin to verify a User by their user ID.
Preconditions:	Admin is logged in and is authenticated
Postconditions:	The user's application is either approved, pending or rejected.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. FeedItForward lists the Consumers that have submitted an application for priority food 2. Admin selects the Consumer that the Admin wants to review. 3. FeedItForward displays all of the Consumer's application information as listed in the SubmitPriorityRequest use case, and prompts the Admin to either APPROVE, REQUEST, REJECT, or CANCEL. 4. Admin approves the application by selecting APPROVE. FeedItForward will approve the Consumer's application and the Consumer will now be a priority Consumer 5. The Consumer will be notified using the included use case NotifyUser.
Alternative Flows:	<p>AF-S4: If Admin selects REQUEST</p> <ol style="list-style-type: none"> 1. If the Admin selects REQUEST, the system will put the Consumer's application under pending and the Consumer will be notified to submit more documents using the included use case NotifyUser. <p>AF-S4: If Admin selects REJECT</p> <ol style="list-style-type: none"> 1. If the Admin selects REJECT, the system will reject the Consumer's application and the Consumer will be notified of the rejection using the included use case NotifyUser. <p>AF-S4: If Admin selects CANCEL</p> <ol style="list-style-type: none"> 1. Admin can select CANCEL at any time to cancel the operation and the system will return to the SelectAdminActivity.

Exceptions:	None
Includes:	NotifyUser
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

I.IV ProcessReview

Use Case ID:	#1-4		
Use Case Name:	ProcessReview		
Created By:	Toh Jing Hua	Last Updated By:	Toh Jing Hua
Date Created:	10th September 2023	Date Last Updated:	10th September 2023

Actor:	Admin
Description:	Allows Admin to process a review for deletion or for keeping.
Preconditions:	Admin is logged in and is authenticated
Postconditions:	Flagged review is deleted or ignored.
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. The FeedItFoward lists the reviews that have been flagged by it or the Users. 2. Admin selects the review that the Admin wants to process. 3. The system displays the content of the review and prompts the Admin to DELETE, IGNORE, or CANCEL. 4. If the Admin selects DELETE, the system will delete the flagged review.
Alternative Flows:	<p>AF-S4: If Admin selects IGNORE</p> <ol style="list-style-type: none"> 1. If the Admin selects IGNORE, the system will ignore the flagged review and remove it from the list of flagged reviews. <p>AF-S4: If Admin selects CANCEL</p> <ol style="list-style-type: none"> 1. Admin can select CANCEL at any time to cancel the operation and the system will return to the SelectAdminActivity.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

I.V NotifyUser

Use Case ID:	#1-5		
Use Case Name:	NotifyUser		
Created By:	Toh Jing Hua	Last Updated By:	Toh Jing Hua
Date Created:	10th September 2023	Date Last Updated:	10th September 2023

Actor:	Admin
Description:	Allows Admin to send notification to a user
Preconditions:	Admin is logged in and is authenticated
Postconditions:	Notification is sent.
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. Admin enters the content of the notification, which includes the title and description. 2. Admin enters the recipient User ID. 3. FeedItFoward then prompts the Admin to select CONFIRM or CANCEL. 4. If the Admin selects CONFIRM, the system will send the content to the specified User.
Alternative Flows:	AF-S4: If Admin selects CANCEL <ol style="list-style-type: none"> 1. If the Admin selects CANCEL, the system will cancel the operation and return the Admin back to the Admin's original page.
Exceptions:	<ol style="list-style-type: none"> 1. If the User ID is not found, FeedItFoward shall display an error message and prompt the User to reenter the User ID.
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

II. For Functional Requirement #2 (Hawker)

II.I SubmitLeftoverFood

Use Case ID:	#2-1		
Use Case Name:	SubmitLeftoverFood		
Created By:	Denise Tay	Last Updated By:	Denise Tay
Date Created:	9th September 2023	Date Last Updated:	1st November 2023

Actor:	Hawker
Description:	Allows hawkers to submit the available leftover food for pickup onto FeedItForward
Preconditions:	<ol style="list-style-type: none">1. Hawker must be logged in and authenticated.2. Hawker has leftover food available for pickup.
Postconditions:	Leftover food is successfully declared/submitted for pickup.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. Hawker selects the "Submit Leftover Food" option.2. Hawker indicates the amount of leftover food, unit of measurement (of the leftover food), name, amount of time passed (in hours) since the food was cooked, and uploads a picture of the leftover food.3. Hawker confirms the submission by pressing the "Confirm" button.
Alternative Flows:	None
Exceptions:	<ol style="list-style-type: none">1. If the Hawker is not logged in, they are prompted to log in before submitting leftover food.2. If any required information is missing or invalid (see below for more detailed examples of invalid information for this use case), an error message is displayed.
Includes:	None
Special Requirements:	System needs to validate user input data
Assumptions:	None
Notes and Issues:	None

SubmitLeftOverFood Invalid Information:

a) Amount of Leftover Food:

- i) Negative value: Entering a negative value for the amount of leftover food (in kilograms) is invalid.

- ii) Special symbols: Using special symbols in the input instead of alphanumeric characters would be invalid.
- a) Amount of Time Since Cooking:
 - i) Negative value: Entering a negative value for the time since the food was cooked (in hours) is invalid.
 - ii) Non-numeric characters: Using alphabetic characters or special symbols in the input instead of numbers is invalid.
- b) Picture Upload:
 - i) Uploading a file that is not an image: If the system expects an image file and the user uploads a non-image file format (e.g., a PDF or text document), it would be invalid.
 - ii) Corrupted or unreadable image file: Uploading an image file that is corrupted or unreadable would also be invalid.

III. For Functional Requirement #3 (Consumer)

III.I SubmitReview

Use Case ID:	#3-1		
Use Case Name:	SubmitReview		
Created By:	Toh Jing Qiang	Last Updated By:	Toh Jing Qiang
Date Created:	9th September 2023	Date Last Updated:	9th September 2023

Actor:	Consumer
Description:	Consumers can submit a review on a Hawker by uploading pictures and writing a review text.
Preconditions:	The Consumer must be logged in and authenticated.
Postconditions:	The Review is successfully submitted.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none">1. The system displays a "Review" button for Consumers to submit a review on a particular Hawker.2. When a Consumer selects the "Review" button, the system will prompt the Consumer to upload pictures, enter review text about a specific Hawker and its food, and give a rating from 0 to 5.3. The system will then prompt the user to confirm his review via a "Submit Review" Button.4. If the Consumer selects the "Submit Review" Button, the system will save the review (pictures, text, and rating) in the database and display that the review is submitted.
Alternative Flows:	None
Exceptions:	<ol style="list-style-type: none">1. If any required information is missing or invalid (e.g.: uploaded photo format is not of picture format), an error message is displayed.
Includes:	None
Special Requirements:	System needs to validate user input data.
Assumptions:	None
Notes and Issues:	None

III.II EditReview

Use Case ID:	#3-2		
Use Case Name:	EditReview		
Created By:	Toh Jing Qiang	Last Updated By:	Toh Jing Qiang
Date Created:	9th September 2023	Date Last Updated:	9th September 2023

Actor:	Consumer
Description:	Consumers can edit his/her review on a Hawker by uploading pictures, removing uploaded pictures or editing the review text.
Preconditions:	The Consumer must be logged in and authenticated.
Postconditions:	The Review is successfully edited.
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. If a Consumer has already made a review on a particular Hawker, the system will display an "Edit" button beside his/her review. 2. When the Consumer selects the "Edit" button, the system will prompt the Consumer to upload new pictures, remove uploaded pictures, modify the review text, and edit the rating. 3. The system will display a "Save" button to allow the Consumer to save the changes made, and a "Cancel" button to allow the Consumer to abandon the changes made. 4. When the "Save" button is selected, the system will save the changes made.
Alternative Flows:	<p>AF-S1: If Consumer has not make a review on a particular Hawker</p> <ol style="list-style-type: none"> 1. The system will not display an "Edit" button. 2. The system will only display the "Review" button for Consumers to submit a review on the Hawker. <p>AF-S4: If the "Cancel" button is selected</p> <ol style="list-style-type: none"> 1. The system will not save the changes made. 2. The system will redirect the Consumer away from the Edit Hawker Screen.
Exceptions:	<ol style="list-style-type: none"> 1. If any required information is missing or invalid, an error message is displayed.
Includes:	None
Special Requirements:	System needs to validate user input data.

Assumptions:	None
Notes and Issues:	None

III.III RequestFood

Use Case ID:	#3-3		
Use Case Name:	RequestFood		
Created By:	Toh Jing Qiang	Last Updated By:	Toh Jing Qiang
Date Created:	9th September 2023	Date Last Updated:	9th September 2023

Actor:	Consumer
Description:	Consumer chooses from a list of food available for request and submits a request for chosen food
Preconditions:	<ol style="list-style-type: none"> 1. The Consumer must be logged in and authenticated. 2. There must be food available for request.
Postconditions:	<ol style="list-style-type: none"> 1. The Request is successfully submitted. 2. The Hawker is alerted of the Consumer's food request.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> 1. The system will display a list of food available for request via a list interface. 2. For each food available for request, the system will display the amount of food, the unit of measurement of the food, picture of food, amount of time (in hours) that has passed, and a "Request for Food" button. 3. When the "Request for Food" button is pressed, the system will prompt the Consumer to enter the amount of food that the Consumer wants, choose between the options of having the food delivered to them by a Driver or having the food collected by themselves, and confirmation via a "Confirm Request" button. 4. When the Consumer inputs all the required information in 3 and selects the "Confirm Request" button, the system will notify the associated Hawker that a food request has been made. 5. If the Consumer chooses the food delivery option, the system will assign a driver to pick up the food and notify the Hawker and Consumer about the driver's estimated time of arrival at the Hawker and Consumer respectively.
Alternative Flows:	<p>AF-S5: If the Consumer chooses the self collection option</p> <ol style="list-style-type: none"> 1. The system will notify the Hawker that the Consumer is collecting the food him/herself.

Exceptions:	1. If any required information is missing or invalid (e.g.: amount of food is not in decimal), an error message is displayed.
Includes:	None
Special Requirements:	System needs to validate user input data.
Assumptions:	There are Drivers to be assigned to food requests to pick up the food.
Notes and Issues:	None

III.IV SubmitPriorityRequest

Use Case ID:	#3-4		
Use Case Name:	SubmitPriorityRequest		
Created By:	Toh Jing Qiang	Last Updated By:	Toh Jing Qiang
Date Created:	9th September 2023	Date Last Updated:	9th September 2023

Actor:	Consumer
Description:	Consumers can submit a request to gain priority for food request acceptance.
Preconditions:	The Consumer must be logged in and authenticated.
Postconditions:	The Request is successfully submitted.
Priority:	High
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. The system will allow the Consumer to input their personal information like household income, number of residents in household, occupation, and house category (3-room, 4-room, 5-room, executive condo, condo, landed, or penthouse). 2. After the Consumers upload their personal information, the system will submit a request for priority in food request acceptance to the Admins to verify the information using the included use case VerifyUser.
Alternative Flows:	None
Exceptions:	<ol style="list-style-type: none"> 1. If any required information is missing or invalid, an error message is displayed.
Includes:	VerifyUser
Special Requirements:	System needs to validate user input data.
Assumptions:	None
Notes and Issues:	None

IV. For Functional Requirement #4 (Driver)

IV.I ProcessPickupJob

Use Case ID:	#4-1		
Use Case Name:	ProcessPickupJob		
Created By:	Denise Tay	Last Updated By:	Denise Tay
Date Created:	9th September 2023	Date Last Updated:	9th September 2023

Actor:	Driver
Description:	Allows drivers to select whether to accept or ignore a pickup job allocated to them.
Preconditions:	<ol style="list-style-type: none">1. Driver is logged in to the FeedItForward application.2. There is leftover food available for pickup.
Postconditions:	<ol style="list-style-type: none">1. If the Driver chooses to "Accept" the pickup job, they are directed to complete the job's required steps and responsibilities.2. If the Driver selects to "Ignore" the pickup job, it remains unassigned, and the Driver returns to the main task management screen.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. Driver is logged in to the FeedItForward application.2. The FeedItForward system continuously monitors for leftover food available for pickup.3. When leftover food becomes available, the system automatically allocates one pickup job to the Driver.4. A pop-up modal appears on the Driver's screen, notifying them of the allocated pickup job.5. The modal provides information about the job, including the pickup location, estimated delivery time, and any specific instructions or notes from the Hawker.6. FeedItForward present the Driver with two options – ACCEPT or IGNORE7. If the Driver selects ACCEPT:<ol style="list-style-type: none">7.1. The system proceeds to display the steps required to complete the job.7.2. Driver collects food at pickup location<ol style="list-style-type: none">7.2.1. Driver submits a picture of the food at the pickup location.7.2.2. Driver submits a picture of the Hawker at the pickup location.

	<p>7.3. Driver follows the optimal route determined by FeedItForward, from pickup location to delivery location.</p> <p>7.4. Driver delivers food at delivery location</p> <p>7.4.1. Driver submits a picture of food at the delivery location.</p> <p>7.4.2. Driver submits a picture of the consumer at the delivery location.</p> <p>8. Job is marked as completed.</p>
Alternative Flows:	<p>AF-S7: If the Driver selects IGNORE:</p> <p>1. The pickup job remains unassigned, and the Driver returns to the main screen.</p>
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	FeedItForward system allocates one pickup job to the Driver when leftover food is available for pickup.
Notes and Issues:	None

V. For Functional Requirement #5

V.I CustomerServiceSupport

Use Case ID:	#5-1		
Use Case Name:	CustomerServiceSupport		
Created By:	Teh Min Ze	Last Updated By:	Toh Jing Qiang
Date Created:	8th September 2023	Date Last Updated:	12th September 2023

Actor:	User
Description:	Communicate via text between a Consumer/Hawker/Driver and an Admin respectively, and vice versa.
Preconditions:	Users must be logged in and authenticated.
Postconditions:	None.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. Consumer/Hawker/Driver navigates to the Customer Service Support screen.2. Consumer/Hawker/Driver selects to send a text message to Admin.3. Admin sends a message back to reply to Consumer/Hawker/Driver's message.
Alternative Flows:	
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

VI. For Functional Requirement #6 (User)

VI.I QueryHawkers

Use Case ID:	#6-1		
Use Case Name:	QueryHawkers		
Created By:	Teh Min Ze	Last Updated By:	Toh Jing Qiang
Date Created:	8th September 2023	Date Last Updated:	13th September 2023

Actor:	User
Description:	Query all Hawkers to show Hawkercs' detailed information and/or optimal route (taking into account weather condition) from User to Hawker. Users can search for a particular Hawker for query too.
Preconditions:	<ol style="list-style-type: none">1. The User must be logged in and authenticated.2. The User's device location service is turned on.
Postconditions:	<ol style="list-style-type: none">1. Successfully display detailed information of queried Hawker to the User.2. Successfully display optimal route from User's location to Hawker.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. The system allows the User to view Hawkercs by either through an interactive map or a list interface.2. User selects the list interface option.3. The System shall display a list of Hawkercs in Singapore.4. For each Hawker, the system will display detailed information about the Hawker such as business name, location, contact details, types of food sold, operating hours, reviews given by Consumers and whether the Hawker is registered with FeedItForward.5. If the User selects the "Weather" feature button, the system will return the current and forecasted weather condition using the included use case QueryWeather.6. If the User selects the "Search" feature button, the system will return a filtered list of Hawkercs using the included use case SearchHawker.
Alternative Flows:	AF-S3: User selects the interactive map option <ol style="list-style-type: none">1. The system shall display a real-time map that displays the locations of all hawkercs, with "Registered" tags beside Hawkercs registered with FeedItForward.

	<ol style="list-style-type: none"> 2. The system allows the User to zoom in, zoom out, and pan the map. 3. When the User selects a specific Hawker, a pop-up modal containing the Hawker information, such as business name, types of food sold, operating details (opening and closing hours), reviews given by Consumers, whether the Hawker is registered with FeedItForward, location, email, and contact number will be displayed. 4. Return to step 5.
Exceptions:	None
Includes:	<ol style="list-style-type: none"> 1. QueryWeather 2. SearchHawker
Special Requirements:	Phone Permission for location access is enabled.
Assumptions:	None
Notes and Issues:	None

VI.II SearchHawkers

Use Case ID:	#6-2		
Use Case Name:	SearchHawker		
Created By:	Teh Min Ze	Last Updated By:	Toh Jing Qiang
Date Created:	13th September 2023	Date Last Updated:	13th September 2023

Actor:	User
Description:	Allows users to search for Hawkers by name, food type or location via a search bar and get a filtered list of Hawkers.
Preconditions:	The User must be logged in and authenticated.
Postconditions:	A filtered list of Hawkers is returned.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> 1. The User can search for Hawkers according to their name, food type or location via a search bar. 2. The system will return a filtered list of Hawkers for the User.
Alternative Flows:	None
Exceptions:	<ol style="list-style-type: none"> 1. If no Hawker data matches the search criteria, the system shall display a message stating no Hawker is found.
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

VI.III QueryWeather

Use Case ID:	#6-4		
Use Case Name:	QueryWeather		
Created By:	Teh Min Ze	Last Updated By:	Toh Jing Qiang
Date Created:	13th September 2023	Date Last Updated:	13th September 2023

Actor:	User
Description:	The system to display the current weather and Singapore's weather forecasted 1 day into the future at a 6-hours interval, and 4 days into the future at a 1 day interval.
Preconditions:	<ol style="list-style-type: none"> 1. The User must be logged in and authenticated. 2. User's device location service is turned on. 3. User has selected the "Weather" feature button.
Postconditions:	<ol style="list-style-type: none"> 1. Successfully display the weather condition.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> 1. The system will display the current weather and Singapore's weather conditions forecasted 1 day into the future at a 6-hours interval, and 4 days into the future at a 1 day interval.
Alternative Flows:	None
Exceptions:	None
Includes:	None
Special Requirements:	Phone permission for location access is enabled.
Assumptions:	None
Notes and Issues:	None

VI.IV FlagReview

Use Case ID:	#6-4		
Use Case Name:	FlagReview		
Created By:	Toh Jing Hua	Last Updated By:	Toh Jing Hua
Date Created:	10th September 2023	Date Last Updated:	10th September 2023

Actor:	User
Description:	Allows User to flag a review for the Admin to process for deletion.
Preconditions:	User is logged in and is authenticated
Postconditions:	Review is flagged
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	<ol style="list-style-type: none"> 1. The system displays an option to flag a review for every review that the User has received. 2. If the User flags the review, the system will prompt the user to input the reason as to why they have flagged the review. 3. The system then prompts the User to select CONFIRM or CANCEL. 4. If the user selects CONFIRM, the system will flag the review and add it to the list of flagged reviews for the Admins to process.
Alternative Flows:	AF-S4: If the user selects CANCEL <ol style="list-style-type: none"> 1. If the user selects CANCEL, the system will cancel the operation and return the User to the review.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

VII. For Functional Requirement #7 (Authentication)

VII.I CreateAccount

Use Case ID:	#7-1		
Use Case Name:	CreateAccount		
Created By:	Pin Yang	Last Updated By:	Toh Jing Qiang
Date Created:	13th September 2023	Date Last Updated:	13th September 2023

Actor:	User
Description:	Allows a person to create an Account in the FeedItForward to become a User.
Preconditions:	None
Postconditions:	An Account with a specific role (Admin, Hawker, Consumer or Rider) is created for the User.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none">1. The system prompts the User to enter their name, address, password, email, contact number, profile picture, and role.2. The User enters the required information and selects the "Create Account" button to confirm his inputs.3. An Account with a specific role (Admin, Hawker, Consumer or Rider) is created for the User.
Alternative Flows:	
Exceptions:	<ol style="list-style-type: none">1. If any required information is missing or invalid (password is not at least 8 characters long), an error message is displayed.
Includes:	None
Special Requirements:	System needs to validate user input data.
Assumptions:	None
Notes and Issues:	None

VII.II Login

Use Case ID:	#7-2		
Use Case Name:	Login		
Created By:	Pin Yang	Last Updated By:	Denise
Date Created:	13th September 2023	Date Last Updated:	27th September 2023

Actor:	User
Description:	Allows User to log into his/her FeedItForward Account using his/her email and password.
Preconditions:	None
Postconditions:	User is logged into the FeedItForward application and is navigated to the home screen of the application.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> 1. The system allows the User to login with either (a) Email and password, or (b) Google. 2. The User chooses to login with email and password. 3. The User enters his/her email and password. The Password is masked as dots, but the User can choose to unmask it by clicking on the eye icon. 4. The User selects the "Login" button.
Alternative Flows:	AF-S2: User chooses to login with Google <ol style="list-style-type: none"> 1. The User selects the "Login with Google" button. 2. The User logs into the system using Singpass login credentials using the included use case GoogleLogin.
Exceptions:	<ol style="list-style-type: none"> 1. If any required information is missing, an error message is displayed. 2. If email and password do not match when the User tries to login in step 4, FeedItForward shall display "Email and password do not match" to the user.
Includes:	GoogleLogin
Special Requirements:	System needs to validate user input data.
Assumptions:	The User has an existing FeedItForward Account.
Notes and Issues:	None

VII.III GoogleLogin

Use Case ID:	#7-3		
Use Case Name:	GoogleLogin		
Created By:	Pin Yang	Last Updated By:	Toh Jing Qiang
Date Created:	13th September 2023	Date Last Updated:	1st November 2023

Actor:	User
Description:	Allows User to log into his/her FeedItForward Account using his/her Google account.
Preconditions:	Entry Condition: Called by the use case Login when the "Login with Google" button is selected.
Postconditions:	User is logged into the FeedItForward application and is navigated to the home screen of the application.
Priority:	High
Frequency of Use:	High
Flow of Events:	<ol style="list-style-type: none"> 1. The User is redirected to the Google Auth User Interface to provide verification. 2. If the Google Auth verification is successful, the system will direct the User to FeedItForward's home page.
Alternative Flows:	AF-S2: Singpass verification is unsuccessful <ol style="list-style-type: none"> 1. The system will redirect the User to the FeedItForward's Login screen and display an error message saying that login with Google was unsuccessful.
Exceptions:	None
Includes:	None
Special Requirements:	<ol style="list-style-type: none"> 1. Access and Usage of Google Auth API
Assumptions:	<ol style="list-style-type: none"> 1. The User has a Google Account.
Notes and Issues:	None