Use Cases

for

myNTU

Version 1.1 approved

Prepared by Ho King Hei

<organization>

9th September 2024

Revision History

Name	Date	Reason For Changes	Version
Ho King Hei	9/9/2024	First Version	1.1

Guidance for Use Case Template

Document each use case using the template shown in the Appendix. This section provides a description of each section in the use case template.

1. Use Case Identification

1.1. Use Case ID

Give each use case a unique numeric identifier, in hierarchical form: X.Y. Related use cases can be grouped in the hierarchy. Functional requirements can be traced back to a labeled use case.

1.2. Use Case Name

State a concise, results-oriented name for the use case. These reflect the tasks the user needs to be able to accomplish using the system. Include an action verb and a noun. Some examples:

- View part number information.
- Manually mark hypertext source and establish link to target.
- Place an order for a CD with the updated software version.

1.3. Use Case History

1.3.1 Created By

Supply the name of the person who initially documented this use case.

1.3.2 Date Created

Enter the date on which the use case was initially documented.

1.3.3 Last Updated By

Supply the name of the person who performed the most recent update to the use case description.

1.3.4 Date Last Updated

Enter the date on which the use case was most recently updated.

2. Use Case Definition

2.1. Actor

An actor is a person or other entity external to the software system being specified who interacts with the system and performs use cases to accomplish tasks. Different actors often correspond to different user classes, or roles, identified from the customer community that will use the product. Name the actor(s) that will be performing this use case.

2.2. Description

Provide a brief description of the reason for and outcome of this use case, or a high-level description of the sequence of actions and the outcome of executing the use case.

2.3. Preconditions

List any activities that must take place, or any conditions that must be true, before the use case can be started. Number each precondition. Examples:

1. User's identity has been authenticated.
2. User's computer has sufficient free memory available to launch task.

2.4. Postconditions

Describe the state of the system at the conclusion of the use case execution. Number each postcondition. Examples:

1. Document contains only valid SGML tags.
2. Price of item in database has been updated with new value.

2.5. Priority

Indicate the relative priority of implementing the functionality required to allow this use case to be executed. The priority scheme used must be the same as that used in the software requirements specification.

2.6. Frequency of Use

Estimate the number of times this use case will be performed by the actors per some appropriate unit of time.

2.7. Flow of Events

Provide a detailed description of the user actions and system responses that will take place during execution of the use case under normal, expected conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description. This description may be written as an answer to the hypothetical question, "How do I <accomplish the task stated in the use case name>?" This is best done as a numbered list of actions performed by the actor, alternating with responses provided by the system.

2.8. Alternative Flows

Document other, legitimate usage scenarios that can take place within this use case separately in this section. State the alternative course, and describe any differences in the sequence of steps that take place. Number each alternative course using the Use Case ID as a prefix, followed by "AC" to indicate "Alternative Course". Example: X.Y.AC.1.

2.9. Exceptions

Describe any anticipated error conditions that could occur during execution of the use case, and define how the system is to respond to those conditions. Also, describe how the system is to respond if the use case execution fails for some unanticipated reason. Number each exception using the Use Case ID as a prefix, followed by "EX" to indicate "Exception". Example: X.Y.EX.1.

2.10. Includes

List any other use cases that are included ("called") by this use case. Common functionality that appears in multiple use cases can be split out into a separate use case that is included by the ones that need that common functionality.

2.11. Special Requirements

Identify any additional requirements, such as nonfunctional requirements, for the use case that may need to be addressed during design or implementation. These may include performance requirements or other quality attributes.

2.12. Assumptions

List any assumptions that were made in the analysis that led to accepting this use case into the product description and writing the use case description.

2.13. Notes and Issues

List any additional comments about this use case or any remaining open issues or TBDs (To Be Determineds) that must be resolved. Identify who will resolve each issue, the due date, and what the resolution ultimately is.

3. Use Case Description

3.1. Sign Up

Use Case ID:	ACCOUNT_UC_1		
Use Case Name:	Sign Up		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	03/09/2024	Date Last Updated:	03/09/2024

Actor:	User / Store Owner
Description:	This use case allows users to sign up for an account to access the relevant functions according to the user's role. The various user roles that are required to sign up are user & store owner. To sign up for the account, the user has to key in their email address and password. Once the account has been created, the user will be redirected to the login page
Preconditions:	NA
Postconditions:	The user will be redirected to the relevant home page
Priority:	High
Frequency of Use:	Once per user
Flow of Events:	1. The user enters their email and password 2. The user submits their email and password 3. The system validates and verifies the submitted email and password 4. The system displays a message saying "Account has been created" when the sign-up is successful 5. The system will redirect the user to the relevant home page 6. The use case ends
Alternative Flows:	AF-S3a: If the email is used by another user 1. The system displays a message saying "Invalid email! Email is used by another user" 2. The use case resumes at main flow step 1 AF-S3b: If the password is invalid

	1. The system displays a message saying "Invalid password! Password must contain at least 8 characters with at least 1 number" 2. The user case resumes at main flow step 1 AF-S3c: If the email and/or password is missing 1. The system displays a message saying "Email or password must be filled" 2. The use case resumes at main flow step 1
Exceptions:	If any required information is missing or invalid, an error message is displayed.
Includes:	NA
Special Requirements:	The system needs to validate data input by the user
Assumptions:	NA
Notes and Issues:	NA

3.2. Login

Use Case ID:	ACCOUNT_UC_2		
Use Case Name:	Login		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	03/09/2024	Date Last Updated:	03/09/2024

Actor:	User / Store Owner / System Admin
Description:	This use case allows users to log into the system to access the relevant functions according to the user's role. The various user roles that can log in are user, store owner & system admin. To log in, all users have to enter their email and password. Upon successful login, the system display the relevant user's home page
Preconditions:	The user has to have a valid account created beforehand
Postconditions:	The system displays the relevant homepage
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The user enters their email and password 2. The user submits their email and password

	3. The system validates and verifies the submitted email and password 4. The system will display the user's homepage 5. The use case ends
Alternative Flows:	AF-S3a: If the email and/or password is invalid 1. The system displays a message saying "Invalid email or password! Please try again" 2. The use case resumes at main flow step 1 AF-S3b: If the email and/or password is not filled in 1. The system displays a message saying "Email and/or password must be filled" 2. The use case resumes at main flow step 1 AF-S3C: 1. The system displays a message saying "Your account is banned. Please contact the administrator!" 2. The use case resumes at main flow step 1
Exceptions:	1. If any required information is missing or invalid, an error message is displayed. 2. If the user account is banned, an error message is displayed.
Includes:	NA
Special Requirements:	The system needs to validate data input by the user
Assumptions:	NA
Notes and Issues:	NA

3.3. Logout

Use Case ID:	ACCOUNT_UC_3		
Use Case Name:	Logout		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	03/09/2024	Date Last Updated:	03/09/2024

Actor:	User / Store Owner / System Admin
Description:	This use case allows users to log out of the system. The various user roles that can log out are user, store owner & system admin. Upon

	successful logout, the system will display a message saying "Bye! Hope to see you next time!" and the login page will be displayed
Preconditions:	The user has to be signed in
Postconditions:	1. The system displays a message saying "Bye! Hope to see you next time!" 2. The system displays the login homepage
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	1. The user clicks on the profile button 2. The system will display the user's profile page 3. The user clicks on "Sign out" 6. The system displays a message saying Bye! Hope to see you next time! 7. The system redirects the user back to the Login page 8. The use case ends
Alternative Flows:	NA
Exceptions:	NA
Includes:	NA
Special Requirements:	NA
Assumptions:	NA
Notes and Issues:	NA

3.4. Update Profile

Use Case ID:	ACCOUNT_UC_4		
Use Case Name:	Update Profile		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	03/09/2024	Date Last Updated:	03/09/2024

Actor:	User / Store Owner
Description:	This use case allows users to update their profiles. The various user roles that can update their profiles are the users and store owners. The user can update the display name, profile picture and password. Upon successful update, the system will display a message saying "Profile updated"
Preconditions:	The user has to be signed in

Postconditions:	1. The system displays a message saying "Profile updated" 2. The system redirects the user back to the Profile page
Priority:	Medium
Frequency of Use:	Low
Flow of Events:	1. The user clicks on the profile button 2. The system will display the user's profile page 3. The user clicks on "Update profile" 4. The user enters the new input on the relevant fields 5. The user clicks on the "Update" 6. The system displays a message saying "Profile Updated" 7. The system redirects the user back to the Profile page 8. The use case ends
Alternative Flows:	AF-S4a: If the password is invalid 1. The system displays a message saying "Invalid password! Password must contain at least 8 characters with at least 1 number" 2. The use case resumes at main flow step 1
Exceptions:	If any password is invalid, an error message is displayed.
Includes:	NA
Special Requirements:	The system needs to validate data input by the user
Assumptions:	NA
Notes and Issues:	NA

3.5. Delete Account

Use Case ID:	ACCOUNT_UC_5		
Use Case Name:	Delete Account		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	03/09/2024	Date Last Updated:	03/09/2024

Actor:	User / Store Owner
Description:	This use case allows users to delete their accounts. The various user roles that can delete their accounts are the users and store owners. When the user clicks on the "Delete Account" button, the system displays a confirmation message saying "Do you want to delete

	your account?". Upon successful deletion, the system will display a message saying "Account deleted"
Preconditions:	The user has to be signed in
Postconditions:	The system redirects the user back to the login page
Priority:	Medium
Frequency of Use:	Low
Flow of Events:	1. The user clicks on the profile button 2. The system will display the user's profile page 3. The user clicks on "Delete Account" 4. The system displays a message saying "Do you want to delete your account?" 5. The user clicks on the "Delete" 6. The system displays a message saying "Account deleted" 7. The system redirects the user back to the login page 8. The use case ends
Alternative Flows:	AF-S5a: If the user clicks "Back" 1. The system will display the user's profile page 2. The use case ends
Exceptions:	NA
Includes:	NA
Special Requirements:	NA
Assumptions:	NA
Notes and Issues:	NA

3.6. View Map

Use Case ID:	USER_UC_1		
Use Case Name:	View map		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	03/09/2024	Date Last Updated:	03/09/2024

Actor:	User
Description:	This use case allows users to view the map that shows the food areas in NTU. When the user clicks on the "Map" button, the system displays a map that shows the food areas in NTU

Preconditions:	User has to enable Location sharing in their local device
Postconditions:	The system displays a map that shows the food areas in NTU
Priority:	Medium
Frequency of Use:	High
Flow of Events:	1. The user clicks on the map button 2. The system will display a map that shows the food areas in NTU 3. The use case ends
Alternative Flows:	AF-S1a: If location sharing is not enabled on the user's local device 1. The system displays an error message saying "Please enable location sharing to use Map" 2. The user will be redirected back to the relevant user's home page 3. The use case ends
Exceptions:	If location sharing is not enabled on the user's local device, the map will not work
Includes:	NA
Special Requirements:	Location sharing must be enabled for the user's local device
Assumptions:	NA
Notes and Issues:	NA

3.7. View Food

Use Case ID:	USER_UC_2		
Use Case Name:	View Food		
Created By:	King Hei	Last Updated By:	King Hei
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	User
Description:	This use case allows the user to view the available food in NTU. When the user clicks on the "Food button", the system displays the available food from the various stalls in NTU
Preconditions:	The user has to be signed in
Postconditions:	The system displays the available food from the various stalls in NTU

Priority:	High
Frequency of Use:	High
Flow of Events:	1. The user clicks on the “Food” button 2. The system displays the available food from the various stalls in NTU 3. The use case ends
Alternative Flows:	NA
Exceptions:	NA
Includes:	NA
Special Requirements:	NA
Assumptions:	NA
Notes and Issues:	NA

3.8. Search Food

Use Case ID:	USER_UC_3		
Use Case Name:	Search Food		
Created By:	King Hei	Last Updated By:	King Hei
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	User
Description:	This use case allows the user to search for food by food name or by location. When the user enters a specific food or location on the Search bar within the “Food” page, the system displays the relevant results based on the query
Preconditions:	The user has to be signed in
Postconditions:	The food options related to the query will be displayed
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The user clicks on the “Food” button 2. The user clicks on the Search bar within the “Food” page 3. The user types in a food name 4. The system will show all stalls that sell that particular food 5. The use case ends OR

	1. The user clicks on the "Food" button 2. The user clicks on the Search bar within the "Food" page 3. The user types in the location of one of the food areas in NTU 4. The system will show the available food located in that particular food area 5. The use case ends
Alternative Flows:	AF-S2a: If the user clicks "Back" within the Search page 1. The system will redirect the user back to the "View Food" page 2. The use case ends
Exceptions:	If the user types in an invalid location or food name, the system will display a message saying "No results found".
Includes:	View Food
Special Requirements:	NA
Assumptions:	NA
Notes and Issues:	NA

3.9. Filter Food

Use Case ID:	USER_UC_4		
Use Case Name:	Filter Food		
Created By:	King Hei	Last Updated By:	King Hei
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	User
Description:	This use case allows the user to filter food by Cuisine, Type, Price or Discount. When the user clicks on the filter button, the system will display the options for the user to filter food by. Upon clicking the "Apply Filter" button after selecting the fields to filter by, the system will display the relevant results based on the query
Preconditions:	The user has to be signed in
Postconditions:	The food options related to the query will be displayed
Priority:	High
Frequency of Use:	High

Flow of Events:	1. The user clicks on the "Food" button 2. The user clicks on the filter button 3. The user selects one of the filters or a combination of them a. If the user selects Price, the system will display the food, from the cheapest/most expensive food at the top of the list b. If the user selects a Cuisine, the system will display the food from stalls that sell that specific cuisine c. If the user selects Discount, the system will display all the food that are listed on a discount d. If the user selects Type, only stalls that sell that specific type of food will be shown in the list 4. The use case ends
Alternative Flows:	AF-S3a: If the user clicks "Cancel" within the Filter Food page 1. The system will redirect the user back to the "View Food" page 2. The use case ends
Exceptions:	NA
Includes:	View Food
Special Requirements:	NA
Assumptions:	NA
Notes and Issues:	NA

3.10. Submit Food Review

Use Case ID:	USER_UC_5		
Use Case Name:	Submit Food Review		
Created By:	King Hei	Last Updated By:	King Hei
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	User
Description:	This use case allows the user to submit their food review. When the user clicks the "Leave a review" button shown on the stall page, the

	user can rate the food, upload a photo and write a description about the food. Upon successful submission, the system displays a message saying "Review submitted"
Postconditions:	The user has to be signed in
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	1. The user 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 submitted" 5. The system redirects the user to the relevant stall page The use case ends
Alternative Flows:	AF-3a: If there are missing fields on the form 1. The system displays a message "Please fill up relevant parts of the form" The use case resumes at main flow step 2
Exceptions:	If any required information is missing or invalid, an error message is displayed.
Includes:	NA
Special Requirements:	The system needs to validate the input from the user
Assumptions:	NA
Notes and Issues:	NA

3.11. Update Food Review

Use Case ID:	USER_UC_6		
Use Case Name:	Update Food Review		
Created By:	King Hei	Last Updated By:	King Hei
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	User
Description:	This use case allows the user to edit a review that they have previously submitted. When the user clicks on the “Edit review” button on a previously submitted review, the user can enter the new

	inputs in the form. Upon successful submission, the system displays a message saying “Review edited”
Preconditions:	1. The user has to be signed in 2. There must be a review that was previously submitted by the user
Postconditions:	1. The system displays a message saying “Review edited” 2. The system redirects the user to the Reviews page
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	1. The user clicks on the “Published Reviews” button 2. The user clicks on the “Edit Review” button on a previously submitted review 3. The system displays a form with details of their submitted review 4. The user edits relevant portions of the form and selects “Save” 5. The system displays a message saying “Review edited” 6. The system redirects the user to the Reviews page 7. The use case ends
Alternative Flows:	AF-S4a: If there is missing input for the mandatory fields on the form 1. The system displays a message “Please fill up relevant parts of the form” 2. The use case resumes at main flow step 2
Exceptions:	If any required information is missing or invalid, an error message is displayed.
Includes:	NA
Special Requirements:	The system needs to validate the input from the user
Assumptions:	NA
Notes and Issues:	NA

3.12. Delete Food Review

Use Case ID:	USER_UC_7
Use Case Name:	Delete Food Review

Created By:	Anthony	Last Updated By:	Anthony
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	User / System Admin
Description:	This use case allows the user or system admin to delete a review. When the user clicks on the "Delete review" button on a previously submitted review, the system displays a confirmation message saying "Delete your review?". Upon successful deletion, the system displays a message saying "Review deleted". When the system admin has processed the review, the system admin may delete the review
Preconditions:	1. The user has to be signed in 2. There must be a review that was previously submitted by the user
Postconditions:	1. The system displays a message saying "Review deleted" 2. The system redirects the user to the Reviews page
Priority:	Medium
Frequency of Use:	Low
Flow of Events:	1. The user clicks on the "Published Reviews" button 2. The user clicks on the "Delete review" on a previously submitted review 3. The system displays a confirmation message saying "Delete your review?" 4. The user clicks on the "Delete" button 5. The system displays a message saying "Review deleted" 6. The system redirects the user to the Reviews page 7. The use case ends
Alternative Flows:	AF-S5a: If the user clicks "Back" 3. The system will display the Published Reviews page 4. The use case ends
Exceptions:	NA
Includes:	NA
Special Requirements:	NA
Assumptions:	NA
Notes and Issues:	NA

3.13. Flag Food Review

Use Case ID:	USER_UC_8		
Use Case Name:	Flag Food Review		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	User
Description:	This use case allows the user to flag a food review. When the user clicks on the “Flag” button on a previously submitted review, the system sends a notification to the system admin about the flagged review and the system displays a message saying “Review flagged”
Preconditions:	1. The user has to be signed in 2. There must be a review that was previously submitted by the user
Postconditions:	1. The system sends a notification to the system admin 2. The system displays a message saying “Review flagged”
Priority:	Medium
Frequency of Use:	Low
Flow of Events:	1. The user clicks on a previously submitted review by another user 2. The user clicks on the “Flag button” 3. The system sends a notification to the system admin 4. The system displays a message saying “Review flagged” 5. The use case ends
Alternative Flows:	AF-S2a: If the user clicks “Back” on the Review page 1. The system redirects the user back to the relevant page 2. The use case ends
Exceptions:	NA
Includes:	NA
Special Requirements:	NA
Assumptions:	NA
Notes and Issues:	NA

3.14. View Notifications

Use Case ID:	USER_UC_9		
Use Case Name:	View Notifications		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	User / System Admin
Description:	This use case allows the user or system admin to view notifications. When the user clicks on the Bell button at the top of their Home page, the system displays all the notifications that the user has. When the system admin clicks on the "Notifications", the system displays all the notifications that the system admin has.
Preconditions:	The user (user and system admin) has to be signed in
Postconditions:	The system sends all the notifications the user has
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	1. The user clicks on the Bell button at the top of their Home page 2. The system displays all the notifications the user has 3. The use case ends OR 1. The system admin clicks on the "Notifications" button 2. The system displays all the notifications the system admin has 3. The use case ends
Alternative Flows:	NA
Exceptions:	If there are no notifications, the system will display "You have no notifications"
Includes:	NA
Special Requirements:	NA
Assumptions:	NA
Notes and Issues:	NA

3.15. Submit Food Listing

Use Case ID:	STORE OWNER_UC_1		
Use Case Name:	Submit Food Listing		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	Store Owner
Description:	This use case allows the store owner to submit food listings onto the app. The store owner can submit food listings such as new and/or discounted food onto the app for users to view. Upon successful submission, the system will display a message saying “Food submitted”
Preconditions:	The store owner has to be signed in
Postconditions:	The system displays the newly listed food submitted by the store
Priority:	High
Frequency of Use:	High
Flow of Events:	1. The store owner clicks on the “Submit Food” button 2. The store owner enters the details (price, image, description and discount if any) of the food 3. The store owner clicks on the “Submit” button 4. The system displays a message saying “Food submitted” 5. The use case ends
Alternative Flows:	AF-S2a: If the store owner clicks “Back” within the Submit Food page 1. The system will redirect the store owner back to the relevant store owner home page 2. The use case ends AF-3a: If there are missing fields on the form 1. The system displays a message “Please fill up relevant parts of the form” 2. The use case resumes at main flow step 2

Exceptions:	If any required information is missing or invalid, an error message is displayed.
Includes:	NA
Special Requirements:	The system needs to validate the input from the store owner
Assumptions:	NA
Notes and Issues:	NA

3.16. Update Food Listing

Use Case ID:	STORE OWNER_UC_2		
Use Case Name:	Update Food Listing		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	Store Owner
Description:	This use case allows the store owner to update the submitted food listings on the app. The user can update the name, category, price, picture and discount if any. Upon successful update, the system will display a message saying “Listing updated”
Preconditions:	1. The store owner has to be signed in 2. There must be a food listing that was previously submitted by the store owner
Postconditions:	The system displays a message saying “Listing updated”
Priority:	High
Frequency of Use:	Medium
Flow of Events:	1. The store owner clicks on the specific food listing to update 2. The store owner clicks on the “Edit Listing button” 3. The store owner enters the new input on the relevant fields 4. The store owner clicks on the “Update” button 5. The system displays a message saying “Listing updated” 6. The use case ends
Alternative Flows:	AF-S3a: If the store owner clicks “Back” within the Update Food page 1. The system will redirect the store owner back to the food listing page 2. The use case ends AF-S4a: If there is missing input for the mandatory fields on the form 1. The system displays a message “Please fill up relevant parts of the form” 2. The use case resumes at main flow step 3

Exceptions:	If any required information is missing or invalid, an error message is displayed.
Includes:	NA
Special Requirements:	The system needs to validate the input from the store owner
Assumptions:	NA
Notes and Issues:	NA

3.17. Delete Food Listing

Use Case ID:	STORE OWNER_UC_3		
Use Case Name:	Delete Food Listing		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	Store Owner
Description:	This use case allows the store owner to delete the submitted food listings on the app. The user can delete any of the submitted food listings. Upon successful deletion, the system will display a message saying “Listing deleted”
Preconditions:	3. The store owner has to be signed in 4. There must be a food listing that was previously submitted by the store owner
Postconditions:	The system displays a message saying “Listing deleted”
Priority:	Medium
Frequency of Use:	Low
Flow of Events:	1. The store owner clicks on the specific food listing to update 2. The store owner clicks on the More (3 dots) button 3. The systems displays a message saying “Do you want to delete this listing?” 4. The store owner clicks on the “Delete” button 5. The system displays a message saying “Listing deleted” 6. The use case ends
Alternative Flows:	AF-S2a: If the store owner clicks “Back” within the Delete Food page 1. The system will redirect the store owner back to the food listing page

	2. The use case ends
Exceptions:	NA
Includes:	NA
Special Requirements:	NA
Assumptions:	NA
Notes and Issues:	NA

3.18. Process Review

Use Case ID:	ADMIN_UC_1		
Use Case Name:	Process Review		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	System Admin
Description:	This use case allows the system admin to process the food reviews. The system admin can process the flagged food reviews that are flagged by users. When the user clicks on the "Process Review" button, the system displays all the flagged food reviews, with the most recent ones at the top
Preconditions:	The system admin has to be signed in
Postconditions:	NA
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	1. The system admin clicks on the "Process Review" button 2. The system displays all the flagged reviews with the most recent ones at the top 3. The system admin can click on the More (3 dots) button to see further actions available for any specific review 4. The system admin clicks on "Delete review" button 5. The system displays a message saying "Review deleted" 6. The system sends a notification to the user saying the review has been deleted 7. The use case ends

	OR 1. The system admin clicks on the “Process Review” button 2. The system displays all the flagged reviews with the most recent ones at the top 3. The system admin can click on the More (3 dots) button to see further actions available for any specific review 4. The system admin clicks on “Resolve” button 5. The system removes the review from the list 6. The use case ends
Alternative Flows:	AF-S3a: If the store owner clicks “Back” 1. The system will redirect the store owner back Process Review page
Exceptions:	NA
Includes:	NA
Special Requirements:	NA
Assumptions:	NA
Notes and Issues:	NA

3.19. Ban User

Use Case ID:	ADMIN_UC_2		
Use Case Name:	Ban User		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	System Admin
Description:	This use case allows the system admin to ban users. The system admin can ban users from using the app. Upon successful banning, the system will display a message saying “User banned”
Preconditions:	1. The system admin has to be signed in 2. The target account must not be banned beforehand

Postconditions:	1. The system displays a message saying “User banned” 2. The banned account system admin will not be able to log in 3. The system redirects the system admin to the relevant page that displays the user list (users excluding store owner)
Priority:	Medium
Frequency of Use:	Low
Flow of Events:	1. The system admin clicks on the “Ban User” button 2. The system displays the user list for the system admin to ban 3. The system admin clicks on the target account to ban 4. The system displays the Ban Account page 5. The system admin enters the relevant description 6. The system clicks on the “Ban” button 7. The system displays a message saying “Do you want to ban the user?” 8. The system admin clicks on the “Yes” button 9. The system displays a message saying “User banned” 10. The system redirects the system admin to the page that displays the user list The use case ends
Alternative Flows:	AF-5a: If the system admin clicks “Back” within the Ban account page: 1. The use case resumes at main flow step 2 AF-8a: If the system admin clicks the “No” button The use case resumes at main flow step 5
Exceptions:	NA
Includes:	NA
Special Requirements:	NA
Assumptions:	NA
Notes and Issues:	NA

3.20. Notify User

Use Case ID:	ADMIN_UC_3		
Use Case Name:	Notify User		
Created By:	Anthony	Last Updated By:	Anthony
Date Created:	04/09/2024	Date Last Updated:	04/09/2024

Actor:	System Admin
Description:	This use case allows the system admin to notify users. The system admin can send notifications to users directly. When the system admin clicks the “Notify” button, the system sends a notification to the target user
Preconditions:	The system admin has to be signed in
Postconditions:	1. The user receives a notification from the system admin The system redirects the system admin back to the Notifications page
Priority:	Medium
Frequency of Use:	Medium
Flow of Events:	1. The system admin clicks on “Notifications” button 2. The system displays the Notification page 3. The system admin clicks on the Plus (+) button 4. The system admin the input on the relevant fields on the Notification form 5. The system admin clicks the “Notify” button 6. The system sends a notification to the target user 7. The system redirects the system admin back to the Notification page The use case ends
Alternative Flows:	AF-3: If system admin clicks “Cancel” or Cross (x) button, The system redirects the system admin back to the Notifications page
Exceptions:	If any required information is missing or invalid, an error message is displayed.
Includes:	NA
Special Requirements:	The system needs to validate the input from the store owner
Assumptions:	NA
Notes and Issues:	NA