
Group 1

**Foodaholic
Use-Case Specification**

Version 1.0

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

Revision History

Date	Version	Description	Author
10/Jun/22	1.0	Initial Draft	Phan Tuấn Khải Lê Đăng Khoa Nguyễn Thiện Nhân Lê Trọng Anh Tú

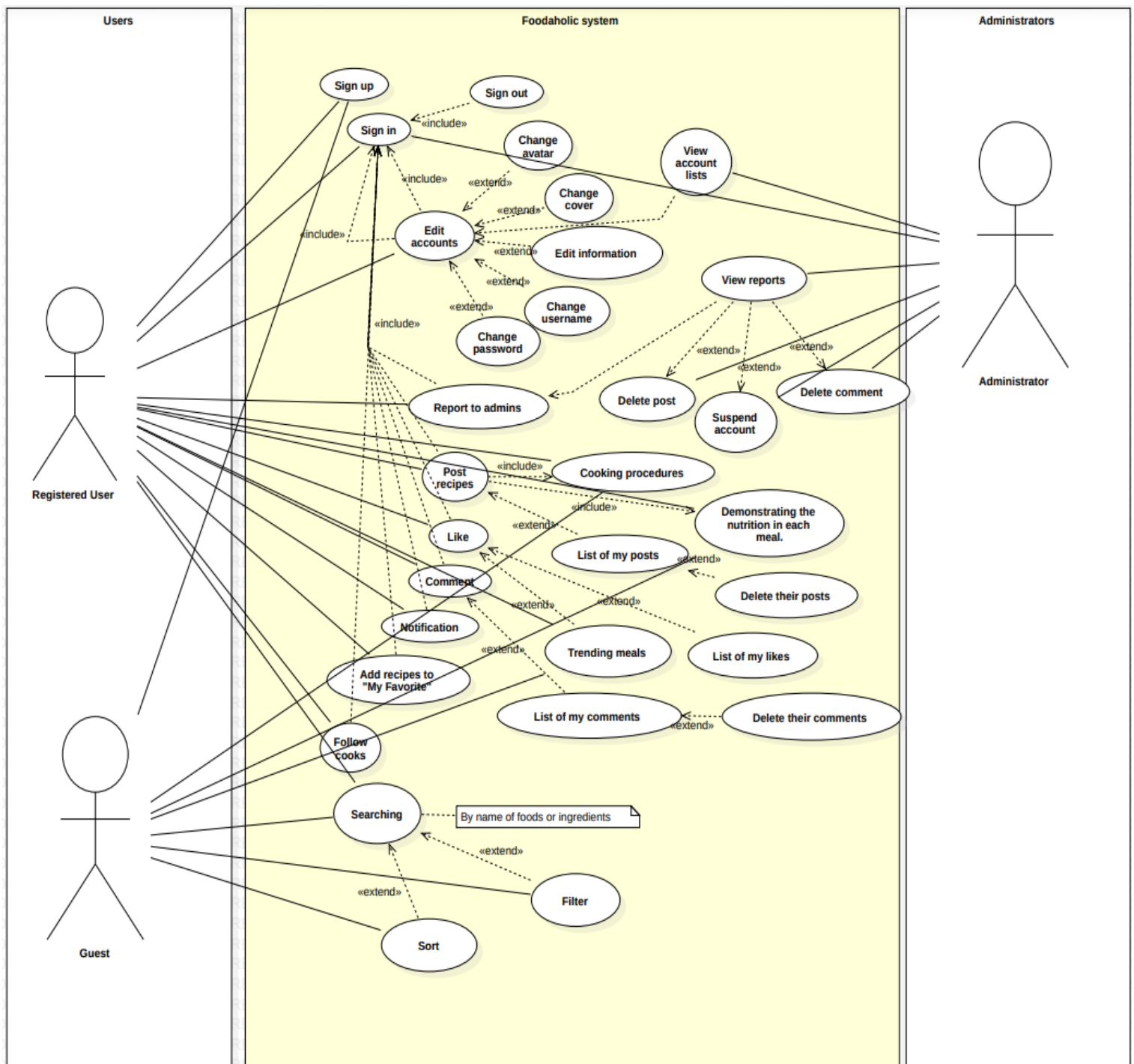
Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

Table of Contents

Use-case Model	4
Use-case Specifications	5
Use-case: Sign in	5
Use-case: Sign up (Register)	5
Use-case: Edit information	6
Use-case: Change avatar (Profile picture)	7
Use-case: Change password	8
Use-case: Change username	8
Use-case: Post recipes	9
Use-case: Detailed description of cooking steps	10
Use-case: Like post recipes	10
Use-case: List of user's like	11
Use-case: Comment on post recipes	11
Use-case: Administrator deletes the reported post	12
Use-case: Administrator deletes the reported comment	13
Use-case: Administrator suspends the reported account	13
Use-case: Registered user deletes their comment	14
Use-case: Administrator views reports	14
Use-case: Administrator views account list	15
Use-case: Report posts	15
Use-case: Searching (by name keywords or ingredients)	16
Use-case: Filter input requirements	17
Use-case: Filter searched results	17
Use-case: Sort searched results	18
Use-case: Show nutrition in each type of food	18
Use-case: Trending now	19
Use-case: Sign out	19
Use-case: Change cover	20
Use-case: Follow cooks	20
Use-case: Delete their posts	20
Use-case: List of my posts	21
Use-case: List of my comments	21
Use-case: Notification	22
Use-case: Add recipes to "My Favorite"	22

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

1. Use-case Model



Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

2. Use-case Specifications

2.1 Use-case: Sign in

Use case Name	Sign in
Brief description	This use case describes how to sign in to the system. Therefore you can like, comment in any posted recipes
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> 1. Successfully access to our website (The domain of website will be announced later) 2. In the homepage, user clicks 'Sign in' 3. System shows the sign in page 4. User inputs username and password 5. After that, user clicks 'Sign in' in sign in page 6. System checks if it's an available account 7. System notices that 'Sign in successfully', moves to homepage with this account
Alternative Flows	<p>Alternative flow 1: '404 Error'</p> <ol style="list-style-type: none"> 1. In #1, this user can't access our website (Our website is maintained, this domain is not available...) <p>Alternative flow 2:</p> <ol style="list-style-type: none"> 1. In #5, user can press 'Enter' in user's keyboard instead click 'Sign in' in sign in page <p>Alternative flow 3: Sign in unsuccessfully</p> <ol style="list-style-type: none"> 1. In #6, if the account user input isn't exist in the system, system will notice 'Sign in unsuccessfully because of wrong username or password' 2. The system will move to sign in page for user to input again <p>Alternative flow 4: Avoid spamming password</p> <ol style="list-style-type: none"> 1. In #5 and #6, if the user signs in and fails greater than or equal to 5 times, the system will show captcha to avoid spamming password to hack this account. <p>Alternative flow 5: No information in username or password (or both)</p> <ol style="list-style-type: none"> 1. System will notice "Please fill in this field"
Pre-conditions	<ul style="list-style-type: none"> - User must access successfully to the system (No 404 ERROR) - User must have an account in the system
Post-conditions	The user successfully signs in and can like, comment in any post. User can also post recipes to everyone

2.2 Use-case: Sign up (Register)

Use case Name	Sign up
Brief description	This use case describes how to register to the system. Then the user will then have full access to all of the registered user's features.
Actors	Guest user
Basic Flow	<ol style="list-style-type: none"> 1. Successfully access to our website (The domain of website will be announced later) 2. In the homepage, user click 'Sign in' 3. Rather than logging in, the user presses the "Create new account" button. 4. System shows the 'Sign up' screen

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

	<ol style="list-style-type: none"> 5. User inputs Surname, First name, username, password and Re-enter password (password is a combination of at least six numbers, letters and punctuation marks such as ! and &). 6. After that, user click 'Sign up' button in page 7. System checks if this account is valid (all five fields are valid). 8. System notices that 'Sign up successfully', move to homepage with this account
Alternative Flows	<p>Alternative flow 1: 'One of the five fields is blank'</p> <ol style="list-style-type: none"> 1. If missing Surname or First name, system will display notifications "What's your name?" 2. If missing username, system will display notifications "You'll use this when you log in and if you ever need to reset your password." 3. If missing password or Re-enter password, system will display notifications "Enter a combination of at least six numbers, letters and punctuation marks (such as ! and &)." <p>Alternative flow 2: "Invalid character in Surname or First name"</p> <ol style="list-style-type: none"> 1. Invalid character such as number, punctuation marks or a character that is not alphabetic 2. Then system will display notifications "Invalid username" <p>Alternative flow 3: "Invalid character in username"</p> <ol style="list-style-type: none"> 1. Invalid character in username is a character that is not an alphabetic or numeric character. 2. The system will move to sign in page for user to input again <p>Alternative flow 4: "Username already exists"</p> <ol style="list-style-type: none"> 1. In #5, if username is already used by another account, system will display notifications "Username already exists, try another username" <p>Alternative flow 5: "Password and Re-enter password field does not match"</p> <ol style="list-style-type: none"> 1. System will display notifications "The entered passwords do not match. Try again." <p>Alternative flow 6: "Press Enter"</p> <ol style="list-style-type: none"> 1. In #6, user can press 'Enter' in user's keyboard instead click 'Sign up' button
Pre-conditions	<ul style="list-style-type: none"> • User must access successfully to the system (No 404 ERROR)
Post-conditions	Users will then have full access to all of the registered user's features.

2.3 Use-case: Edit information

Use case Name	Edit information
Brief description	This use case describes how to edit information in your account. Therefore you can update your information and leave the contact methods such as email, phone number...
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> 1. Sign in successfully to the system (Use-case specification about 'Sign in' is described in section 2.1) 2. In the homepage, click the avatar of this account in the upper right corner 3. The system will show the list of account's features like sign out, change information, list of my posts...

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

	<ol style="list-style-type: none"> User click 'Edit information' System show the edit information page User input some information that need to update/change or create User click 'Completed' to complete edit information process System notice 'Edit information successfully', update information of this account and move to homepage
Alternative Flows	<p>Alternative flow 1: Unique information</p> <ol style="list-style-type: none"> In #6, user inputs information that matches with other users like email, phone number... So the system can't edit information and notice users like "Email is used. Please type another email" <p>Alternative flow 2:</p> <ol style="list-style-type: none"> In #7, user can press 'Enter' instead of click 'Completed' to complete edit information process <p>Alternative flow 3:</p> <ol style="list-style-type: none"> In #7, if the user suddenly disconnect from the Internet, the old version of information will be kept
Pre-conditions	User must sign in successfully to the website
Post-conditions	Update user's information based on what user inputs in change information page

2.4 Use-case: Change avatar (Profile picture)

Use case Name	Change avatar
Brief description	This use case describes how users can change their profile picture.
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> Sign in successfully to the system (Use-case specification about 'Sign in' is described in section 2.1) In the homepage, click the avatar of this account in the upper right corner The system will show the list of account's features like sign out, change information, list of my posts... Users click 'Edit information' System shows the edit information page User clicks on the avatar again and selects "change profile picture" The popup for Avatar Update appears. The user can either upload photos or choose from the system's existing images. The user will be shown a demonstration by the system. If the users accept, they can click "save avatar", which will be replaced in a few minutes.
Alternative Flows	<p>Alternative flow 1: 'User chooses a file with the wrong extension or too big'</p> <ol style="list-style-type: none"> Valid extensions include JPG, PNG, JPEG, GIF Maximum file size is 4MB If user chooses a file with the wrong extension, system will display notifications "I'm unable to upload photographs. Photo must be in JPG, PNG, GIF, JPEG format and it's size must be less than 4MB" <p>Alternative flow 2: "User click 'X' button"</p> <ol style="list-style-type: none"> Before #8, if user clicks X button in the upper right corner. It will close the Avatar Update popup without display notifications. After #8, system will display notifications "Are you sure you want to discard the changes? The avatar you just selected will not be saved"

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

	<ul style="list-style-type: none"> • If user click “Discard”, Avatar update popup will be closed and profile picture will not be saved • If user click “Cancel”, notification will be closed, continue step #9
Pre-conditions	<ul style="list-style-type: none"> • User must access successfully to the system (No 404 ERROR) • User must sign in successfully to the website
Post-conditions	Your avatar will be uploaded and everyone will see your avatar when you post or click into your profile (including Guest - who doesn't have an account).

2.5 Use-case: Change password

Use case Name	Change password
Brief description	This use case describes how to change the password of your account. Therefore you can protect your account, avoid leaking your own password
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> 1. Sign in successfully to the system (Use-case specification about ‘Sign in’ is described in section 2.1) 2. In the homepage, click the avatar of this account in the upper right corner 3. The system will show the list of account's features like sign out, change information, change password, list of my posts... 4. User click ‘Change password’ 5. System show the change password page 6. User enter old password to verify again to do this feature 7. User enter new password and re-enter new password 8. User click ‘Completed’ 9. System will update new password for user's account 10. System log out all accounts which is active and turn back to Sign in page
Alternative Flows	<p>Alternative flow 1:</p> <ol style="list-style-type: none"> 1. In #6, if the old password which is entered by user doesn't match with current password of this account, system will notice that ‘Wrong password’ and let the user type again 2. If the user fail on verifying old password greater than or equal to 5 times, the system will disable this feature in 1 day and send email to notice that someone is trying to change your password <p>Alternative flow 2:</p> <ol style="list-style-type: none"> 1. In #7, if the new password match with the old password, system will notice and let the user to enter your new password again with constraint that new password and old password mustn't match <p>Alternative flow 3:</p> <ol style="list-style-type: none"> 1. With the device which is active on this system (different from the device which is changing the password of this account), the system will show a box to notice that ‘This account has just changed the password. Please sign in again’
Pre-conditions	User must sign in successfully to the website
Post-conditions	Update user's password based on what user inputs in the change password page. System will log out all accounts which is active in the system and take the user sign in again

2.6 Use-case: Change username

Use case Name	Change username
Brief description	This use case describes how to change username, which is required to login
Actors	Registered user

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

Basic Flow	<ol style="list-style-type: none"> 1. Sign in successfully to the system (Use-case specification of 'Sign in' is described in section 2.1) 2. In the homepage, click the avatar of the account in the upper right corner 3. The system will show the list of account's features like sign out, change information, change password, list of my posts... 4. Users click 'Change username' 5. System shows the 'change username' page 6. Users enter password to verify again to do this feature 7. Then click 'Verify' button, system will check the password in database 8. Users type new username and click "Complete" 9. System will update new username for user's account if possible 10. System logs out this account and turn back to Sign in page
Alternative Flows	<p>Alternative flow 1: "Verify password failure"</p> <ol style="list-style-type: none"> 11. In #6, if the old password entered by user doesn't match with current password of this account, system will notice that 'Wrong password' and let the user type again 12. If the user fails on verifying old password greater than or equal to 5 times, the system will disable this feature in 1 day and send email to notice that someone is trying to change your password <p>Alternative flow 2: "Same username"</p> <ol style="list-style-type: none"> 13. In #8, if the new username matches with the old username, system will notice and let the user enter your new username again with a constraint that new username and old username mustn't match. <p>Alternative flow 3: "Invalid character in username"</p> <ol style="list-style-type: none"> 1. Invalid character is a character that is not an alphabetic or numeric character. 2. The system will display notification "Invalid character in username" and let user input again <p>Alternative flow 4: "Username already exists"</p> <ol style="list-style-type: none"> 1. In #8, if username is already used by another account, system will display notifications "Username already exists, try another username" <p>Alternative flow 5: "Multiple login tabs"</p> <ol style="list-style-type: none"> 1. If you have many browsers or tabs login with the same account, the system will show a box to notice that "This account has just changed the username. Please sign in again"
Pre-conditions	<ul style="list-style-type: none"> • User must access successfully to the system (No 404 ERROR) • User must sign in successfully to the website
Post-conditions	Update user's username based on what user inputs in the 'change username' page. System will log out all tabs which are active with the account and ask the user sign in again. Changing user's username can help keep user's account safe and secure, preventing it from being exposed or being lost.

2.7 Use-case: Post recipes

Use case Name	Post recipes
Brief description	This use case describes how to post recipes to the system and shows them to everyone who access this website
Actors	Registered user

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

Basic Flow	<ol style="list-style-type: none"> 1. Sign in successfully to the system (Use-case specification of 'Sign in' is described in section 2.1) 2. Click the box which show 'Post something here' in the middle of the homepage 3. Enter the recipe you want to post into this box 4. When you finish your post, click 'Post' to upload your recipe to the system 5. The system will handle your post and take a short time to show this in the website 6. When the system completed, it will show you the message to alert you uploaded your post successfully and the system will return to the homepage 7. You can see your post in 'List of my posts' (which will be mentioned later)
Alternative Flows	Alternative flow 1: <ol style="list-style-type: none"> 1. In #5, if your account is being locked on the 'Post' function because of many reports in your previous post, you can't post your recipe until there is no more block.
Pre-conditions	User must sign in successfully to the website
Post-conditions	Your post is uploaded and everyone will see your post (including Guest - who doesn't have an account). You can see the number of likes, comments in the below of the post

2.8 Use-case: Detailed description of cooking steps

Use case Name	Cooking procedures
Brief description	This use case describes how to cook 1 dish with detailed step-by-step
Actors	Registered user, Guest
Basic Flow	<ol style="list-style-type: none"> 1. Access to our website (The domain of website will be announced later) 2. The system will show the homepage 3. Click any recipe you see in the homepage 4. System will move you to this recipe page 5. Scroll until you see the detailed description of cooking steps
Alternative Flows	Alternative flow 1: <ol style="list-style-type: none"> 1. In #1, users/guests can't access our website (Our website is maintained, this domain is not available...) if a problem occurs Alternative flow 2: <ol style="list-style-type: none"> 1. In #3, if you can't see any recipe, it means that there is no recipe in the system. You can wait for the first recipe in the system uploaded by a registered user. Alternative flow 3: <ol style="list-style-type: none"> 1. If you want to ask for more detailed or want to give feedback about any recipe, you must be a registered user (have an account and sign in to the system)
Pre-conditions	User must access to the website and click to any recipe which is posted by Registered user
Post-conditions	You can see the detailed of cooking steps and after that, maybe if you do it (step-by-step) perfectly, you can make a delicious meal

2.9 Use-case: Like post recipes

Use case Name	Like post recipes
Brief description	This use case describes how a user likes another user's post recipes or their own, which will affect the trending
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> 1. Sign in successfully to the system (Use-case specification about 'Sign in' is described in section 2.1)

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

	<ol style="list-style-type: none"> In the lower left corner of post, it will contain emoji expressions include User clicks one of those emojis to express their feelings for the post System will notify user who post a post you liked their post User can see this post in “List of my like” (which will be mentioned later)
Alternative Flows	<p>Alternative flow 1: “Double like” or “Unlike”</p> <ol style="list-style-type: none"> If user likes the post and chose that emoji again, the user will cancel the emoticon for that post <p>Alternative flow 2: “Choose other emoji”</p> <ol style="list-style-type: none"> If user likes the post and chose other emoji, the selected emoticon will be replaced by the new chosen emoji
Pre-conditions	<ul style="list-style-type: none"> User must access successfully to the system (No 404 ERROR) User must sign in successfully to the website It is necessary to have website’s posts available.
Post-conditions	Below the post will show the people who liked the post. If the number of likes is in the top of the posts with the most likes, that post will be placed in the trending section of the homepage.

2.10 Use-case: List of user’s like

Use case Name	List of user’s like
Brief description	This use case describes how a users see posts they've liked.
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> Sign in successfully to the system (Use-case specification about ‘Sign in’ is described in section 2.1) In the homepage, click the avatar of this account in the upper right corner The system will show the list of account’s features like sign out, change information, list of my posts... Users click ‘List of my like’ Users will be directed to the “List of my like” page containing the posts they have liked
Alternative Flows	<p>Alternative flow 1: “Unlike in List of my like page”</p> <ol style="list-style-type: none"> If the user cancels the emoticon for that post, this post will disappear after user refreshes page or comes back to the page next time. <p>Alternative flow 2: “User don’t like anything yet”</p> <ol style="list-style-type: none"> System will notification “Haven’t you liked a post yet? Go pick some of your favorite foods and "Like" them; they'll show here.”
Pre-conditions	<ul style="list-style-type: none"> User must access successfully to the system (No 404 ERROR) User must sign in successfully to the website
Post-conditions	“List of my like” help users review useful posts, review recipes that have been shared in that post

2.11 Use-case: Comment on post recipes

Use case Name	Comment on post recipes
Brief description	This use case describes how a user comments on another user's post recipes or their own.
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> Sign in successfully to the system (Use-case specification about ‘Sign in’ is described in section 2.1)

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

	<ol style="list-style-type: none"> Below the post, it will contain a textbox with message “Add your comment here” User adds their comment in that box and clicks send System will notify user who post a post you liked their post User can see this comment right above the comment box User can see this post in “List of my comments” (which will be mentioned later)
Alternative Flows	<p>Alternative flow 1: “Limit the number of characters”</p> <ol style="list-style-type: none"> Number of characters will be limited to 200 characters to avoid spam If the number of characters reached the limit, the system will not allow the user to enter more <p>Alternative flow 2: “Enter press”</p> <ol style="list-style-type: none"> In #3, instead click send to send comment, users can press enter on keyboard to send their comment
Pre-conditions	<ul style="list-style-type: none"> User must access successfully to the system (No 404 ERROR) User must sign in successfully to the website It is necessary to have website’s posts available.
Post-conditions	Below the post will show the people who commented on the post. Users who post this post can like this comment.

2.12 Use-case: Administrator deletes the reported post

Use case Name	Administrator deletes the reported post
Brief description	This use case allows an administrator to delete a reported post
Actors	Administrator
Basic Flow	<ol style="list-style-type: none"> An administrator signs in to their account They can view the “Report List” box After clicking, they view a list of the posts reported by the user(s) Choosing the “Delete” button to delete a post Send notification to the user
Alternative Flows	<p>Alternative flow 1:</p> <ol style="list-style-type: none"> In #4, if the administrator chooses the “Deny” button, the reported post will not be eliminated and the system will send notification to the reporter (instead of the user having the reported post in #6) to announce that the post does not violate the rules <p>Alternative flow 2:</p> <ol style="list-style-type: none"> In #5, if the system fails to delete the post after clicking “Delete” button, it will send an alert to the administrator and nothing further will be done until the deletion is successfully redone or ignored (not deleting the post) <p>Alternative flow 3:</p> <ol style="list-style-type: none"> If the user deletes their post before being censored, report of this post will be deleted
Pre-conditions	Administrator must sign in to their account and the list of reported posts must not be empty.
Post-conditions	The reported post will be deleted or kept based on if it violates our conditions and national laws or not

2.13 Use-case: Administrator deletes the reported comment

Use case Name	Administrator deletes the reported comment
---------------	--

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

Brief description	This use case allows an administrator to delete a reported comment
Actors	Administrator
Basic Flow	<ol style="list-style-type: none"> 1. An administrator signs in to their account 2. They can view the “Report List” box 3. After clicking, they view a list of the comments reported by the user(s) 4. Choosing the “Delete” button to delete a comment 5. Send notification to the user
Alternative Flows	<p>Alternative flow 1:</p> <ol style="list-style-type: none"> 1. In #4, if the administrator chooses the “Deny” button, the reported comment will not be eliminated and the system will send notification to the reporter (instead of the user having the reported comment in #6) to announce that the comment does not violate the rules <p>Alternative flow 2:</p> <ol style="list-style-type: none"> 1. In #5, if the system fails to delete the comment after clicking “Delete” button, it will send an alert to the administrator and nothing further will be done until the deletion is successfully redone or ignored (not deleting the comment) <p>Alternative flow 3:</p> <ol style="list-style-type: none"> 1. If the user deletes their comment before being censored by administrator, report of this comment will be deleted
Pre-conditions	Administrator must sign in to their account and the list of reported comments must not be empty.
Post-conditions	The reported comment will be deleted or kept based on if it violates our conditions and national laws or not

2.14 Use-case: Administrator suspends the reported account

Use case Name	Administrator suspends the reported account
Brief description	This use case allows an administrator to suspend an user account
Actors	Administrator
Basic Flow	<ol style="list-style-type: none"> 1. An administrator signs in to their account 2. They can view the “Report List” box 3. After clicking, they view a list of the accounts reported by the user(s) 4. Choosing the “Suspend” button to suspend an account 5. Announce suspending successfully to the administrator 6. Send notification to the user
Alternative Flows	<p>Alternative flow 1:</p> <ol style="list-style-type: none"> 1. In #4, if the administrator chooses the “Deny” button, the reported account will not be suspended and the system will send notification to the reporter (instead of the reported user in #6) to announce that the account does not violate the rules <p>Alternative flow 2:</p> <ol style="list-style-type: none"> 1. In #5, if the system fails to suspend the account after clicking “Suspend” button, it will send an alert to the administrator and nothing further will be done until the suspension is successfully redone or ignore (not suspending the account) <p>Alternative flow 3:</p> <ol style="list-style-type: none"> 1. If this account were suspended before, the suspending time would be incremented
Pre-conditions	Administrator must sign in to their account and the list of reported accounts must not be empty.
Post-conditions	The reported account will be suspended or remained based on if it violates our conditions or national laws or not. If the

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

2.15 Use-case: Registered user deletes their comment

Use case Name	Registered user deletes their comment
Brief description	This use case allows the user to delete their comment in case they do not want to post it anymore
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> 1. The user signs in to their account 2. They find the comment they want to delete 3. Click the “...” button to open the setting box of the post 4. Choose “Delete comment” 5. Alert to confirm deleting the comment 6. Choose “Confirm” to delete the comment
Alternative Flows	<p>Alternative flow 1:</p> <ol style="list-style-type: none"> 1. In #1, if the user is signing in, they can find the post to delete the comment without logging in again 2. If the user chooses to delete their comment without signing in, they will be asked to login to the account <p>Alternative flow 2:</p> <ol style="list-style-type: none"> 1. In #2: <ol style="list-style-type: none"> a. If the comment was posted on their post, the user can open their list of posts in their wall to find the needed post then delete the wanted comment. b. If the comment was posted on other users’ post, the user have to find the account, find the needed post then delete the wanted comment. c. In both cases a and b above, the user can open “Settings” instead to find the wanted comment in their list of comments <p>Alternative flow 3:</p> <ol style="list-style-type: none"> 1. In #6, if the user chooses “Cancel” button instead of “Confirm” button, the alert will be closed and no further actions are done <p>Alternative flow 4:</p> <ol style="list-style-type: none"> 1. In #6, if the system fails to delete their comment after clicking “Confirm” button, it will send an alert to the user and nothing further will be done. Then, the user must redoes the deletion if they still want to delete it. <p>Alternative flow 5:</p> <ol style="list-style-type: none"> 1. If this comment is deleted by administrator, this comment will not show again or the system will notice that “This comment is deleted”
Pre-conditions	Correct registered user account must be signed in and the comment must be found to be deleted
Post-conditions	The comment will be deleted after confirming unless the user changes their mind to cancel the process

2.16 Use-case: Administrator views reports

Use case Name	Administrator views reports
Brief description	This use case allows the administrator to view lists of reported posts, comments, and accounts
Actors	Administrator
Basic Flow	<ol style="list-style-type: none"> 1. The administrator signs in to their account 2. Click the “Report List” button on the left side 3. The system will show three columns with boxes relatively showing reports of posts, comments, and accounts

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

Alternative Flows	Alternative flow 1: <ol style="list-style-type: none"> In #3, in each column, if there is not any report, it will show “No reports” to the administrator Alternative flow 2: <ol style="list-style-type: none"> If there are too many reports, the list will be split into every 20 reports shown in per page
Pre-conditions	Administrator must sign in to access this feature
Post-conditions	The reports will be presented to the administrator whenever they click the “Report List” button

2.17 Use-case: Administrator views account list

Use case Name	Administrator views account list
Brief description	This use case allows the administrator to view the list of account
Actors	Administrator
Basic Flow	<ol style="list-style-type: none"> The administrator signs in to their account Click the “Account List” button on the left side The system will show a table of all accounts (present username, hidden password, created date, the status “Active”, “Offline”, or “Suspended”, and a note)
Alternative Flows	Alternative flow 1: <ol style="list-style-type: none"> In #3, in the table, if there is not any account created, it will show “No accounts” to the administrator Alternative flow 2: <ol style="list-style-type: none"> In #3, in the status column, if an account is “Active” or “Offline”, the system will show an empty “Note” On the other hand, in #3, in the status column, if an account is “Suspended”, it will show the reasons in the “Note” Alternative flow 3: <ol style="list-style-type: none"> If there are too many accounts, the list will be split into every 20 accounts shown in per page
Pre-conditions	Administrator must sign in to access this feature
Post-conditions	The account list will be presented to the administrator whenever they click the “Account List” button

2.18 Use-case: Report posts

Use case Name	Report posts
Brief description	This use case allows the user (or more detailed, allows the registered user) to report a post which is not suitable to appear on this website
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> After signing in successfully, you will be moved to the homepage Registered user click ‘...’ in the upper right corner of the post which you want to report The system show some features about this post Click ‘Report this post to administrator’ The system will display a list of activities that are considered inappropriate to choose and a box below to detailed about what the post violates The user enters the content to be reported After completed, click ‘Send’

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

	8. System will send this report to administrator and notice to administrator to handling
Alternative Flows	<p>Alternative flow 1:</p> <ol style="list-style-type: none"> In #7, you must choose at least 1 activity or there is some information in the box below. If there isn't such its content, the report will notice you to fill in and send it again <p>Alternative flow 2:</p> <ol style="list-style-type: none"> In #7, if the post which you reported is deleted by the owner of the post or by the administrator, you can't report it anymore due to the post is deleted from this system
Pre-conditions	Registered user account must be signed in
Post-conditions	The report will be sent to administrator and they have time to check whether there are any violations in the post being reported

2.19 Use-case: Searching (by name keywords or ingredients)

Use case Name	Search
Brief description	This use case describes how user searches a post of receipts by name of meal or ingredients of foods
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> Sign in successfully to the system (Use-case specification about 'Sign in' is described in section 2.1) In the homepage, there is a text box with title "Search by dish title or ingredients" In the search field, the user types the search word. Then, user click "Search" button The system will look for posts that contain the keywords supplied by the user and return results.
Alternative Flows	<p>Alternative flow 1: "Couldn't find any posts"</p> <ol style="list-style-type: none"> System will display the message "Don't see what you're looking for? To assist others, be the first to share how to create it!" <p>Alternative flow 2: "User input is too long"</p> <ol style="list-style-type: none"> Maximum number of search characters is 128. If user input is longer than that number, system will break 128 characters automatically and search with that amount of characters. <p>Alternative flow 3:</p> <ol style="list-style-type: none"> At #4, user can press Enter from keyboard instead of clicking "Search" button
Pre-conditions	<ul style="list-style-type: none"> User must access successfully to the system (No 404 ERROR) User must sign in successfully to the website
Post-conditions	Posts that match the keywords will appear at the top. Make it simpler for users to find what they're looking for.

2.20 Use-case: Filter input requirements

Use case Name	Filter input requirements
Brief description	This use case allows the user (registered or guest user) to filter input requirements to search for some recipes
Actors	Registered user, guest
Basic Flow	<ol style="list-style-type: none"> After signing in successfully, the user will be moved to the homepage

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

	<ol style="list-style-type: none"> Click the button “Category” next to the search bar to filter The user chooses the category they want to search for (breakfast, lunch, dinner, meat, vegetables, etc) Fill content in the search bar Click the search button or press Enter to search with filtered input The user then sees the results of their input requirements
Alternative Flows	<p>Alternative flow 1:</p> <ol style="list-style-type: none"> In #1, if the user is not signing in, it is still possible to filter and search as a guest user <p>Alternative flow 2:</p> <ol style="list-style-type: none"> In #2 and #3, if the user does not click the button “Category”, they will be showed the found results without any filter In #3, if the user click the button “Category” but then does not choose any category, they will be showed the found results without any filter <p>Alternative flow 3:</p> <ol style="list-style-type: none"> In #4, if the user does not fill any content in the search bar but in #3, they have chosen a category, the system will return all recipes in the selected category In #4, if the user does not fill any content in the search bar and in #3, they have not chosen any category, the system will return all recipes in all categories <p>Alternative flow 4:</p> <ol style="list-style-type: none"> In #6, if there is not any fitted results, the system will show “No results for ...”
Pre-conditions	Registered user or guest must click the search button or press Enter to have this function work
Post-conditions	The recipes will be shown to the user after searching with filtered requirements successfully. If there is not any appropriate recipe, the system will return the announcement “No results for ...”

2.21 Use-case: Filter searched results

Use case Name	Filter searched results
Brief description	This use case allows the user (registered or guest user) to filter searched results
Actors	Registered user, guest
Basic Flow	<ol style="list-style-type: none"> After searching, the results will be shown on the webpage In the “Filter Result” box, there will be categories for the user to filter The user chooses the wanted category Click the “Apply Filter” button The webpage will be reloaded Finally, the initial results will be replaced by the filtered results
Alternative Flows	<p>Alternative flow 1:</p> <ol style="list-style-type: none"> In #3 and #4, if the user does not choose any category, then in #6 the initial results will not be replaced <p>Alternative flow 2:</p> <ol style="list-style-type: none"> In #1, if there are no searched results, then in #6 the system will present similar message to the user <p>Alternative flow 3:</p> <ol style="list-style-type: none"> In #6, if there is not any fitted result, the system will return the message “No results for ...”
Pre-conditions	Registered user or guest must click the “Apply Filter” button to have this function work
Post-conditions	The recipes will be shown to the user after filtering the results successfully. If there is not any fitted recipe, the system will return the announcement “No results for ...”

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

2.22 Use-case: Sort searched results

Use case Name	Sort searched results
Brief description	This use case allows the user (registered user or guest) to sort searched results to find the best recipe for their demand
Actors	Registered user, Guest
Basic Flow	<ol style="list-style-type: none"> 1. After searching, the results will be shown on the webpage (You can see the feature ‘Search and show recipes by keywords/ingredients for more detailed about this step) 2. Click ‘Sort by...’ button 3. After clicking, there is some choice for the user such as sorting in ascending/descending order by nutrition, by name... 4. User click one of them and click ‘Sort’ 5. System will do their algorithm to sort by request from user 6. After doing this, system will show the result after sorting
Alternative Flows	<p>Alternative flow 1:</p> <ol style="list-style-type: none"> 1. In #4, if user do not click any choice and start sorting, the result will not change from the original <p>Alternative flow 2:</p> <ol style="list-style-type: none"> 1. In #7, if there is no recipe after searching in #1, then this use-case will show nothing in the end of sorting although the user do step-by-step correctly
Pre-conditions	Registered user or guest must access this website successfully and click to the ‘Sort by ...’ button to have the feature work
Post-conditions	The recipes which satisfy the requirements of sorting from the user will be shown to the user. If there is not any appropriate recipe, the system will shown nothing in the page

2.23 Use-case: Show nutrition in each type of food

Use case Name	Show nutrition in each type of food
Brief description	This use case allows the user (registered or guest user) to visit a page showing nutrition in each type of food
Actors	Registered user, guest
Basic Flow	<ol style="list-style-type: none"> 1. After entering the home page, the system will show the “Nutrition” button in the navigation bar 2. The user clicks the “Nutrition” button 3. They will be directed to the nutrition page 4. The page will show many kinds of food and brief descriptions of their nutrition which show only how much/many calories, carbohydrate, fat, and protein 5. Click a kind of food to see details of all nutrition existed in that kind of food 6. The user will be directed to the detailed page (title: “Nutrition - ...”) of that food which presents how much/many calories, carbohydrate, fat, protein, and all other kinds of nutrition such as sodium, potassium, cholesterol, etc
Alternative Flows	<p>Alternative flow 1:</p> <ol style="list-style-type: none"> 1. In #4, if there is not any food existing in the database, the system will show “No available food” and no further steps beyond #4 can be done <p>Alternative flow 2:</p> <ol style="list-style-type: none"> 1. In #6, after directing to the detailed page, the user can click the “Back” button to come back to the main “Nutrition” page <p>Alternative flow 3:</p>

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

	1. In #1, if the user is not on the home page but the current page still has the navigation bar with the “Nutrition” button, it is still possible for them to access the “Nutrition” page by clicking it
Pre-conditions	Registered user or guest must click the “Nutrition” button to access the feature
Post-conditions	The nutrition in each type of food will be shown on the webpage to the user. If there is not any existing food, the system will return the message “No available food”

2.24 Use-case: Trending now

Use case Name	Trending meals
Brief description	This use case allows the user (registered or guest user) to see which foods, recipes or cooks are trending now based on the amount of likes, comments, followers...
Actors	Registered user, guest
Basic Flow	<ol style="list-style-type: none"> 1. User access to our homepage (The domain of website will be announced later) 2. Click ‘Trending now’ in the homepage 3. System will display top 3 or top 5 cooks, recipes... in a week
Alternative Flows	<p>Alternative flow 1: Inaccessible</p> <ol style="list-style-type: none"> 1. In #1, this user can’t access our website (Our website is maintained, this domain is not available now...) <p>Alternative flow 2: Not enough information</p> <ol style="list-style-type: none"> 1. If there is no information about this trending page, it means that there are no cooks, no recipes on our system to calculate who is in top trending now. You should wait until next week to see if there are any cooks or recipes in our system. (If there are, we will update to trending in the next week)
Pre-conditions	Registered user or guest must access successfully to the website
Post-conditions	You can see which foods, recipes or cooks are on trending now. So for example you can believe that the recipe in this trending is so popular and very reputation

2.25 Use-case: Sign out

Use case Name	Sign Out
Brief description	User click on “Log Out” and their session was terminated
Actors	Logged User
Basic Flow	<ol style="list-style-type: none"> 1. User is done web application 2. User click on button “Log out” 3. System log user out and terminated session 4. Web application back to default web page
Alternative Flows	<p>Alternative flow 1:</p> <ol style="list-style-type: none"> 1. Log out fail due to network will back to current session
Pre-conditions	<ol style="list-style-type: none"> 1. User was logged in 2. User no longer wanted to be login
Post-conditions	Logout successful

2.26 Use-case: Change cover

Use case Name	Change cover
Brief description	This use case describes how to change cover.
Actors	Registered user

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

Use case Name	Change cover
Brief description	This use case describes how to change cover.
Basic Flow	<ol style="list-style-type: none"> 1. Sign in successfully to the system (Use-case specification about 'Sign in' is described in section 2.1) 2. In the homepage, click the avatar of this account in the upper right corner 3. The system will show the list of account's features like sign out, change information, list of my posts... 4. Users click 'Edit information' 5. System shows the edit information page 6. User clicks on the cover again and selects "change cover picture" 7. The popup for Cover Update appears. 8. The user can either upload photos or choose from the system's existing images. 9. The user will be shown a demonstration by the system. 10. If the users accept, they can click "save cover", which will be replaced in a few minutes.
Alternative Flows	<ol style="list-style-type: none"> 1. When user click on button "Edit information" fail web will back to main page. 2. If user upload image error web will notify and return to previous state. 3. If error occur when user click "save cover", system will keep old cover.
Pre-conditions	<ol style="list-style-type: none"> 1. User must access successfully to the system (No 404 ERROR) 2. User must sign in successfully to the website
Post-conditions	Change cover successful

2.27 Use-case: Follow cooks

Use case Name	Follow Cook
Brief description	User click on "Follow" and cook will add to list of follower
Actors	Logged User
Basic Flow	<ol style="list-style-type: none"> 1. User click on button "Follow" 2. This Cook will be added to list follow of user
Alternative Flows	<ol style="list-style-type: none"> 1. If user click button "follow" but error occur, cook will not been added to list and system will notify error
Pre-conditions	<ol style="list-style-type: none"> 1. User was logged in
Post-conditions	Follow successful and Cook will be added to list follow

2.28 Use-case: Delete their posts

Use case Name	Delete Post
Brief description	User who want to delete their post
Actors	Logged User
Basic Flow	<ol style="list-style-type: none"> 1. After login web 2. User click on "All post" and list post will appears 3. Chose post and click into this post 4. When use click on this post and click on "setting" after this click into "delete post"
Alternative Flows	<ol style="list-style-type: none"> 1. If user delete by mistake, user can be restore post in "delete store"

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

Pre-conditions	<ol style="list-style-type: none"> 1. User logged in 2. User have post and no longer to post
Post-conditions	Delete post successful

2.29 Use-case: List of my posts

Use case Name	List of my post
Brief description	This use case describes how users see posts they've posted.
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> 1. Sign in successfully to the system (Use-case specification about 'Sign in' is described in section 2.1) 2. In the homepage, click the avatar of this account in the upper right corner 3. The system will show the list of account's features like sign out, change information, list of my posts... 4. Users click 'List of my like' 5. Users will be directed to the "List of my post" page containing the posts they have posted
Alternative Flows	Alternative flow 1: "User don't post anything yet" <ol style="list-style-type: none"> 1. System will notification "Haven't you posted a post yet? Go back and post some post; they'll show here."
Pre-conditions	<ul style="list-style-type: none"> • User must access successfully to the system (No 404 ERROR) • User must sign in successfully to the website
Post-conditions	"List of my post" help users review your post.

2.30 Use-case: List of my comments

Use case Name	List of my comment
Brief description	This use case describes how users see comment they've posted.
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> 1. Sign in successfully to the system (Use-case specification about 'Sign in' is described in section 2.1) 2. In the homepage, click the avatar of this account in the upper right corner 3. The system will show the list of account's features like sign out, change information, list of my comment... 4. Users click 'List of my comment' 5. Users will be directed to the "List of my comment" page containing the posts they have posted
Alternative Flows	Alternative flow 1: "User don't comment anything yet" <ol style="list-style-type: none"> 1. System will notification "Haven't you commented a post yet?"
Pre-conditions	<ul style="list-style-type: none"> • User must access successfully to the system (No 404 ERROR) • User must sign in successfully to the website
Post-conditions	"List of my comment" help users review your comment in all post.

2.31 Use-case: Notification

Use case Name	Notification
Brief description	This use case describes how users see any notification of web or action.

Foodaholic	Version: 1.0
Use-Case Specification	Date: 10/Jun/22
<document identifier>	

Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> 1. Sign in successfully to the system (Use-case specification about 'Sign in' is described in section 2.1) 2. In the homepage, see "notification button" near user avatar. 3. Click on button, web will show all notification.
Alternative Flows	<p>Alternative flow 1: "Dont have any notification"</p> <ol style="list-style-type: none"> 1. System will show: "Dont have any notification for you" <p>Alternative flow 2: "View notification"</p> <ol style="list-style-type: none"> 1. If user click on notification, this notification will mark read.
Pre-conditions	<ul style="list-style-type: none"> • User must access successfully to the system (No 404 ERROR) • User must sign in successfully to the website
Post-conditions	Notification will help user overview and remind notification

2.32 Use-case: Add recipes to "My Favorite"

Use case Name	Add recipes to "My favorites"
Brief description	This use case describes how users add recipes to favorites list.
Actors	Registered user
Basic Flow	<ol style="list-style-type: none"> 1. Sign in successfully to the system (Use-case specification about 'Sign in' is described in section 2.1) 2. In new feed or any user's wall have repices. 3. Click on the flag button to add to "My Favorites"
Alternative Flows	<p>Alternative flow 1: If user wrong add</p> <ol style="list-style-type: none"> 1. Click on like button again and recipes will un add
Pre-conditions	<ul style="list-style-type: none"> • User must access successfully to the system (No 404 ERROR) • User must sign in successfully to the website
Post-conditions	Add to my favorites