
Use Cases

for

Jiak!

Version 5.0

Prepared by

**Lim Xin Wei (U2240093B),
Seah Kah Yen (U2240401D),
Le Yanzhi (U2240481K),
Chia Jia En (U2221939E),
Tay Wei Yang (U2221303F),**

Nanyang Technological University, MACS2 (group 3) The Buggers

20/4/2024

Revision History

Name	Date	Reason For Changes	Version
Yan Zhi	30/1/2024	Initial documentation specifying software requirements	1.0
Yan Zhi	25/2/2024	Refinements for lab 2 submission	2.0
Yan Zhi	1/4/2024	Refinements for lab 3 submission	3.0
Kah Yen	15/4/2024	Refinements for lab 4 submission	4.0
Jia en	20/4/2024	Refinements for lab 5 submission	5.0

Use Case Hierarchy

- 1.0 AccountRegistration
 - 1.1 RegistrationValidation
 - 2.0 Login
 - 2.1 SignInWithEmailandPassword
 - 3.0 ViewProfile
 - 3.1 ViewReferralCode
 - 3.2 SignOut
 - 4.0 EditProfile
 - 4.1 CuisinePreferencesSetting
 - 4.2 RatingSetting
 - 4.3 OperationStatusSetting
 - 4.4 ProximitySetting
 - 5.0 RetrieveInformation
 - 5.1 NotificationToggle
 - 5.1.1 ViewRestaurants
 - 5.1.1.1 ViewReviews
 - 5.1.1.2 Jiak!AtRestaurant
 - 5.1.1.2.1 UseReferralCode
 - 5.1.1.2.2 ReviewRestaurant
- 6.0 DiningHistory
 - 6.1 EditReview
 - 6.2 DeleteReview
- 7.0 Social
 - 7.1 RedeemPoints
 - 7.2 SplitCost
 - 7.3 Friends
 - 7.3.1 AddFriends
 - 7.3.2 AddtoParty
 - 7.3.3 ViewFriends
 - 7.3.4 ConfirmParty
 - 7.3.5 ViewOwedAmount

7.4 Parties

7.4.1 LeaveParty

7.4.2 ViewParties

8.0 UserFeedBack

8.1 SubmitFeedBack

9.0 WeatherPage

Use Case ID	1.0		
Use Case Name	AccountRegistration		
Created by	Le Yanzhi	Last Updated by	Le Yanzhi
Date Created	4/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	<p>First time Diners must register for an Account. Diners must enter their Username, Email, Password and confirm their Password to register for an account. Upon successful creation, Diners would then proceed to Use Case EditProfile (3.1).</p>
Preconditions	<ol style="list-style-type: none"> 1. Diner must have an Email. 2. The Email should not have been registered with another account previously.
Postconditions	<ol style="list-style-type: none"> 1. Diner must have an Account 2. Diner must have entered their food preferences.
Priority	High
Frequency of Use	Once per Diner
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on “New to Jiak? Sign up here.” link to register for an account. 2. System prompts user to input Username, Email, Password, Confirm Password. 3. Diner inputs Username, Email, Password, Confirm Password. 4. Diner clicks “Sign Up” button. 5. System validates the information entered using included Use Case RegistrationValidation (1.1). 6. If RegistrationValidation returns True, system creates the Account 7. System updates account information. 8. System displays “Account Created Successfully.” 9. System directs Diner to Use Case EditProfile (3.1). 10. System directs Diner to Home Screen.
Alternative Flows	<p>1.0.AC.1.0: If RegistrationValidation returns UsernameTakenException</p> <ol style="list-style-type: none"> 1. System displays “Username already exists, please choose a different username or login with an existing account.” 2. System returns to Step 2

	<p>1.0.AC.1.1: If RegistrationValidation returns EmailTakenException</p> <ol style="list-style-type: none"> 1. System displays “Email already exists, please choose a different Email or login with an existing account.” 2. System returns to Step 2 <p>1.0.AC.1.2: If RegistrationValidation returns PasswordInvalidException</p> <ol style="list-style-type: none"> 1. System displays “Password must be at least 8 characters long and include at least one uppercase character, one lowercase character, one number, and one special character.” 2. System returns to Step 2 <p>1.0.AC.2.0: If Diner clicks on “Have an account? Log in instead.” button</p> <ol style="list-style-type: none"> 1. System returns to Login Page <p>1.0.AC.3.0: If Diner did not fill up every information (Username, Email, Password, Confirm Password) when clicking “Sign Up” button.</p> <ol style="list-style-type: none"> 1. System displays “Please fill in all fields”. 2. System returns to Step 2. <p>1.0.AC.4.0: If Password and confirm Password does not match</p> <ol style="list-style-type: none"> 1. System displays “Password does not match” 2. System clears Password and confirm Password text box 3. System returns to Step 2.
Exceptions	NIL
Includes	1.1 RegistrationValidation 4.0 EditProfile
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	1.1		
Use Case Name	RegistrationValidation		
Created by	Le Yanzhi	Last Updated by	
Date Created	4/2/2024	Date Last Updated	

Actor	Diner
Description	System checks that the input Username, Email, and Passwords are valid.
Preconditions	1. Diner must have filled up Username, Email, and Password before clicking on “Register” button during AccountRegistration use case
Postconditions	2. All information has been verified
Priority	High
Frequency of Use	Once per Diner
Flow of Events	<ol style="list-style-type: none"> 1. System check the Username has not been taken (not case sensitive) 2. System checks that Password is at least 8 characters long, has at least 1 special character, 1 uppercase letter, 1 lowercase letter and 1 number. 3. System checks that Email is not linked to an existing account. 4. System returns True to AccountRegistration
Alternative Flows	NIL
Exceptions	<p>1.1.EX.1: If Username has been taken</p> <ol style="list-style-type: none"> 1. Throws UsernameTakenException to AccountRegistration <p>1.1.EX.2: If Password is less than 8 characters, or does not contain at least 1 special character, 1 upper letter, 1 lowercase letter and 1 number.</p> <ol style="list-style-type: none"> 1. Throws PasswordInvalidException to AccountRegistration <p>1.1.EX.3: If Email has already been linked to an existing account</p> <ol style="list-style-type: none"> 1. Throws EmailTakenException to AccountRegistration
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	2.0		
Use Case Name	Login		
Created by	Le Yanzhi	Last Updated by	Le Yanzhi
Date Created	4/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner who already has an account will be able to log in to the system and access the functionalities in the Landing Page
Preconditions	1. Diner should have an account created from AccountRegistration Use Case
Postconditions	2. Diner's login information (Email and Password) has been verified and is successfully logged in to the account
Priority	High
Frequency of Use	Once per diner
Flow of Events	<ol style="list-style-type: none"> 1. Diner press "Log In" Button 2. System prompts Diner to input Email and Password 3. Diner input his Email and Password 4. System checks that the Email and Password matches in Database 5. System displays "Log in Successfully." 6. System directs the Diner to the Home Screen.
Alternative Flows	<p>2.0.AC.1.0: If Email does not match any entries in Database</p> <ol style="list-style-type: none"> 1. System displays "Theres no user exists with that email" 2. System returns to Step 2. <p>2.0.AC.2.0: If Diner did not fill up every information (Email, Password) when clicking "Log In" button.</p> <ol style="list-style-type: none"> 1. System displays "Please fill in all fields." 2. System returns to Step 2. <p>2.0.AC.3.0: If Diner entered an invalid email address format</p> <ol style="list-style-type: none"> 1. System displays "The email address is invalid." 2. System returns to Step 2. <p>2.0.AC.4.0: If Email and Password does not match any entires in Database.</p> <ol style="list-style-type: none"> 1. System displays "The password is invalid." 2. System returns to Step 2. <p>2.0.AC.5.0: If Diner has tried to log in too many times.</p>

	<ol style="list-style-type: none"> 1. System display “Access to this account has been temporarily disabled due to many failed login attempts. Try again later.” 2. System returns to Step 2.
Exceptions	NIL
Includes	2.1 SignInWithEmailAndPassword
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	2.1		
Use Case Name	SignInWithEmailAndPassword		
Created by	Le Yanzhi	Last Updated by	Le Yanzhi
Date Created	4/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner who already has an account will be able to log in to the system
Preconditions	1. Diner should have an account created from AccountRegistration Use Case
Postconditions	2. Diner's login information (Email and Password) has been verified and is successfully logged in to the account
Priority	High
Frequency of Use	Once per diner
Flow of Events	<ol style="list-style-type: none"> 1. Diner press "Log In" Button 2. System prompts Diner to input Email and Password 3. Diner input his Email and Password 4. System checks that the Email and Password matches in Database 5. System displays "Log in Successfully." 6. System directs the Diner to the Home Screen.
Alternative Flows	<p>2.0.AC.1.0: If Email does not match any entries in Database</p> <ol style="list-style-type: none"> 1. System displays "Theres no user exists with that email" 2. System returns to Step 2. <p>2.0.AC.2.0: If Diner did not fill up every information (Email, Password) when clicking "Log In" button.</p> <ol style="list-style-type: none"> 1. System displays "Please fill in all fields." 2. System returns to Step 2. <p>2.0.AC.3.0: If Diner entered an invalid email address format</p> <ol style="list-style-type: none"> 1. System displays "The email address is invalid." 2. System returns to Step 2. <p>2.0.AC.4.0: If Email and Password does not match any entires in Database.</p> <ol style="list-style-type: none"> 1. System displays "The password is invalid." 2. System returns to Step 2.

	<p>2.0.AC.5.0: If Diner has tried to log in too many times.</p> <ol style="list-style-type: none"> 1. System display “Access to this account has been temporarily disabled due to many failed login attempts. Try again later.” <p>2.System returns to Step 2.</p>
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	3.0		
Use Case Name	ViewProfile		
Created by	Le Yanzhi	Last Updated by	Le Yanzhi
Date Created	5/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner is allowed to view his profile after successfully login.
Preconditions	1. Diner must have successful login from Use Case Login (2.0)
Postconditions	2. System displays account information to Diner
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner selects option to view profile in Home Screen 2. System displays Username, Cuisine Preferences, Rating, Operation Status, Proximity, Referral Code and a Signout Button.
Alternative Flows	<p>3.0.AC.1.0: If Diner selects the “Edit Profile” button in View Profile Page.</p> <ol style="list-style-type: none"> 1. System directs Diner to Edit Profile Page using included Use Case EditProfile (3.1) 2. System returns to Step 2.
Exceptions	NIL
Includes	3.1 EditProfile
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	3.1		
Use Case Name	ViewReferralCode		
Created by	Le Yanzhi	Last Updated by	Le Yanzhi
Date Created	5/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner is allowed to view his referral code on their profile after successfully login.
Preconditions	1. Diner must have successful login from Use Case Login (2.0)
Postconditions	2. System displays unique referral code to Diner
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner selects option to view profile in Home Screen 2. System displays Referral code and other food preference settings
Alternative Flows	3.1.AC.1.0: If Diner selects the “Edit Profile” button in View Profile Page. 1. System directs Diner to Edit Profile Page using included Use Case EditProfile (3.1) 2. System returns to Step 2.
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	3.2		
Use Case Name	SignOut		
Created by	Le Yanzhi	Last Updated by	Le Yanzhi
Date Created	5/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner is allowed to view his profile to sign out after successfully login.
Preconditions	3. Diner must have successful login from Use Case Login (2.0)
Postconditions	4. System allows diner to sign out
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner selects option to view profile in Home Screen 2. System displays button for diner to sign out 3. Diner signs out of their account and brings them to the log in page
Alternative Flows	3.2.AC.1.0: If Diner selects the “Edit Profile” button in View Profile Page. 1. System directs Diner to Edit Profile Page using included Use Case EditProfile (3.1) 2. System returns to Step 2.
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	4.0		
Use Case Name	EditProfile		
Created by	Le Yanzhi	Last Updated by	Le Yanzhi
Date Created	5/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner is allowed to edit his profile after successful login.
Preconditions	1. Diner must have successful login from Login Use Case
Postconditions	2. Diner's Account has been successfully updated in Database
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner selects option to edit profile details in View Profile Page 2. System directs Diner to Preferece Page 1, which includes CuisinePreferencesSetting Use Case (3.1.1) 3. System directs Diner to Preference Page 2, which includes RatingSetting Use Case (3.1.2), OperationStatusSetting Use Case (3.1.3), ProximitySetting Use Case (3.1.4). 4. System update account information. 5. System displays "Preferences Saved". 6. System directs Diner to Home Screen.
Alternative Flows	NIL
Exceptions	NIL
Includes	3.1.1 CuisinePreferencesSetting 3.1.2 RatingSetting 3.1.3 OperationStatusSetting 3.1.4 ProximitySetting
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	4.1		
Use Case Name	CuisinePreferenceSetting		
Created by	Le Yanzhi	Last Updated by	Le Yanzhi
Date Created	6/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner is allowed to choose the cuisines of the restaurants that he wants to be recommended.
Preconditions	1. Diner must have successful login from Login Use Case
Postconditions	1. Diner's Account has been successfully updated in Database
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. System shows the checklist of all the cuisines, such as Chinese, Japanese, Korean, Western. 2. Diner chooses the cuisines that he prefers by clicking on the cuisine icons. 3. Diner clicks "Next" button 4. System redirects Diner to Preference Page 2
Alternative Flows	3.1.1.AC.1.0 If Diner selected less than 3 cuisines. <ol style="list-style-type: none"> 1. System displays "Please select at least 3 interests" 2. System returns to Step 2.
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	4.2		
Use Case Name	RatingSetting		
Created by	Le Yanzhi	Last Updated by	Le Yanzhi
Date Created	6/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner is allowed to choose the range of Rating for the restaurants that will be recommended to him.
Preconditions	1. Diner must have successful login from Login Use Case
Postconditions	1. Diner's Account has been successfully updated in Database
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. System prompt Diner to select the minimum rating of restaurant that will be recommended to him. 2. Diner selects the rating (ranged from 1 to 5). 3. Diner clicks "Save" button 4. System updates account information 5. System redirects Diner to Home Screen
Alternative Flows	NIL
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	4.3		
Use Case Name	OperationStatusSetting		
Created by	Le Yanzhi	Last Updated by	Le Yanzhi
Date Created	6/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner is allowed to choose whether he also wishes to be recommended the nearby restaurants that are currently closed.
Preconditions	1. Diner must have successful login from Login Use Case
Postconditions	1. Diner's Account has been successfully updated in Database
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. System shows a button that can be toggled "Recommend me only currently open Restaurants" and "Recommend me all Restaurants". 2. Diner chooses to toggle the button. 3. Diner clicks "Save" button. 4. System updates account information 5. System redirects Diner to Home Screen.
Alternative Flows	NIL
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	4.4		
Use Case Name	ProximitySetting		
Created by	Le Yanzhi	Last Updated by	Le Yanzhi
Date Created	6/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner is allowed to choose the maximum distance of restaurants that will be recommended to him.
Preconditions	1. Diner must have successful login from Login Use Case
Postconditions	1. Diner's Account has been successfully updated in Database
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. System shows a draggable bar that Diner can drag to select the maximum distances. 2. Diner chooses to drag the bar. 3. Diner clicks "Save" button 4. System updates account information 5. System redirects Diner to Home Screen,
Alternative Flows	NIL
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	4.5		
Use Case Name	NotificationSetting		
Created by	Le Yanzhi	Last Updated by	Le Yanzhi
Date Created	6/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner is allowed to choose whether he wish to receive pop up notifications in home screen for food recommendations
Preconditions	1. Diner must have successful login from Login Use Case
Postconditions	2. Diner's Account has been successfully updated in Database
Priority	NIL
Frequency of Use	NIL
Flow of Events	<ol style="list-style-type: none"> 1. System displays a button in home screen that can be toggled "Notitication On" and "Notification Off" 2. Diner chooses to toggle the button. 3. System updates account information 4. Diner then sees all the recommended restaurants that pops out according to their food recommendation
Alternative Flows	NIL
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	5.0		
Use Case Name	RetrieveInformation		
Created by	Seah Kah Yen	Last Updated by	Le Yanzhi
Date Created	9/2/2024	Date Last Updated	27/3/2024

Actor	Google Maps API
Description	Call for API to obtain data of all restaurants in 1km proximity to the diner, such as locations of user and restaurant, and user reviews from the Google Maps database
Preconditions	1. Diners have successfully logged onto the application.
Postconditions	1. Diners will be able to see the restaurants in 1km proximity to them based from the Google Maps API database. 2. Diners can choose to click and view any restaurants in details.
Priority	High
Frequency of Use	High
Flow of Events	<ol style="list-style-type: none"> 1. System directs Diner to Home Screen with the Google Map Geolocation. 2. System prompts Diner to grant permission to application to access location. 3. Diner grants application to access location. 4. Diner sees all the restaurants in 1km proximity to them on the Google Map interface.
Alternative Flows	<p>5.0.AC.1.0: If Diner clicks on a restaurant icon on the map.</p> <ol style="list-style-type: none"> 1. System directs Diner to Use Case ViewRestaurants (5.1) 2. System returns to Step 4. <p>5.0.AC.2.0: If Diner clicks on the dining history button</p> <ol style="list-style-type: none"> 1. System directs Diner to Use Case DiningHistory (6.0) 2. System returns to Step 4. <p>5.0.AC.3.0: If Diner click on his own profile.</p> <ol style="list-style-type: none"> 1. System directs Diner to Use Case ViewProfile (3.0) 2. System returns to Step 4. <p>5.0.AC.4.0: If Diner click on weather button.</p> <ol style="list-style-type: none"> 3. System directs Diner to Use Case WeatherPage(9.0) 4. System returns to Step 4. <p>5.0.AC.5.0: If Diner click on friends button.</p>

	<p>5. System directs Diner to Use Case Social (7.0)</p> <p>6. System returns to Step 4.</p> <p>5.0.AC.6.0: If Diner click on user feedback button.</p> <p>7. System directs Diner to Use Case UserFeedBack8.0)</p> <p>8. System returns to Step 4.</p> <p>5.0.AC.7.0: If Diner choose to not grant permission to application to access location.</p> <p>1. System displays “Permission to access location was denied”.</p>
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	5.1		
Use Case Name	ViewRestaurant		
Created by	Seah Kah Yen	Last Updated by	Le Yanzhi
Date Created	9/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Call for API to obtain data of a specific restaurant, such as locations of user and restaurant, and user reviews from the Google Maps database.
Preconditions	1. Diner chose to click and view a specific restaurant in 1km proximity to themselves, in details.
Postconditions	1. Diners will be able to see the details of that specific restaurant viewed from the Google Maps API database.
Priority	High
Frequency of Use	High
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on any restaurant on the Home Screen they would want to view in detail, obtained from the API database. 2. System displays the following to Diner in a pop up. <ol style="list-style-type: none"> a. Diners can see the average ratings of that restaurant b. Diners can see the distance to that restaurant c. Diners can see the cuisine of that restaurant d. Diners can see the address of that restaurant.
Alternative Flows	<p>5.1.AC.1.0: If Diner clicks on “Jiak!” Button</p> <ol style="list-style-type: none"> 1. System directs Diner to Use Case Jiak!AtRestaurant (5.1.2) 2. System returns to Step 1. <p>5.1.AC.2.0: If Diner clicks on view reviews button.</p> <ol style="list-style-type: none"> 1. System directs Diner to Use Case ViewReviews (5.1.1) 2. System returns to Step 1.
Exceptions	NIL
Includes	<p>5.1.1 ViewReviews</p> <p>5.1.2 Jiak!AtRestaurant</p>
Special Requirements	NIL
Assumptions	NIL

Notes and Issues	NIL
------------------	-----

Use Case ID	5.1.1		
Use Case Name	ViewReviews		
Created by	Seah Kah Yen	Last Updated by	Le Yanzhi
Date Created	9/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Call for API to obtain the most recent user reviews of a specific restaurant from the Google Maps database.
Preconditions	1. Diner chose to click and view the reviews of a specific restaurant on Home Screen.
Postconditions	1. Diner is able to see the most recent reviews given by other patrons of the restaurant on Google Maps.
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on the view review button inside the restaurant pop up. 2. System displays the user reviews of that restaurant from Google Maps.
Alternative Flows	NIL
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	5.1.2		
Use Case Name	Jiak!AtRestaurant		
Created by	Seah Kah Yen	Last Updated by	Le Yanzhi
Date Created	9/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner is able to indicate in the application that they have chosen to eat at a restaurant (i.e. Jiak!), either alerted by the notification or through the Google Map Geolocation interface.
Preconditions	<ol style="list-style-type: none"> 1. Diner clicks on the icon of a specific restaurant on Home Screen and pressed the Jiak! Button, or 2. Diner is alerted by the pop-up notification of a recommended restaurant nearby, and pressed the Jiak! Button
Postconditions	<ol style="list-style-type: none"> 1. Diner has finished his meal at the restaurant and press the Review button to give review of the restaurant.
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on the “Jiak!” button to indicate in the application that they have chosen to eat at this restaurant. 2. System redirects Diner to Review Landing Page. 3. Diner clicks on “Review” button. 4. System redirects Diner to ReviewRestaurant Use Case (5.1.2.2)
Alternative Flows	<p>5.1.2.AC.1.0: If Diner clicks on “Have a referral code?” button in Review Landing Page</p> <ol style="list-style-type: none"> 1. System redirects Diner to UseReferralCode Use Case (5.1.2.1) 2. System returns to Step 2.
Exceptions	NIL
Includes	5.1.2.1 UseReferralCode
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	5.1.2.1		
Use Case Name	UseReferralCode		
Created by	Chia Jia En	Last Updated by	Le Yanzhi
Date Created	16/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner can use a referral code that they have obtained for a discount.
Preconditions	1. Diners have pressed the button “Use Referral Code” in Review Landing Page.
Postconditions	1. System returns a barcode for the restaurant to scan.
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on “Use Referral Code” button in Review Landing Page to be redirected to the Input Referral Page. 2. Diner input referral code. 3. Diner submit referral code by pressing the “Submit” button 4. Diners will receive a barcode <ol style="list-style-type: none"> a. Diners must present the barcode while paying b. Restaurants will scan the barcode c. The discount will be applied 5. System direct Diner back to Review Landing Page.
Alternative Flows	NIL
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	5.1.2.2		
Use Case Name	ReviewRestaurant		
Created by	Seah Kah Yen	Last Updated by	Le Yanzhi
Date Created	9/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diners must review the restaurant they have chosen and indicated to dine at in the application.
Preconditions	1. Diners have pressed the Review button in Review Landing Page.
Postconditions	1. System has saved the review in Database
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. After dining, Diners must press the button “Review” in Review Laning Page. 2. System directs Diner to Review Page. 3. Diner select the number of stars and write their reviews. 4. Diner press submit button 5. System saves the review the in Database. 6. System displays “Review Submitted!” 7. System directs Diner to Home Screen.
Alternative Flows	NIL
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	5.2		
Use Case Name	SendNotification		
Created by	Seah Kah Yen	Last Updated by	Le Yanzhi
Date Created	9/2/2024	Date Last Updated	27/3/2024

Actor	Google Map API
Description	Send notification of the recommended restaurant that is in proximity to the diner, based on the Google Map Geolocation.
Preconditions	<ol style="list-style-type: none"> 1. Diners have successfully logged onto the application 2. Diner is at Home Screen. 3. Diners have enabled notifications.
Postconditions	<ol style="list-style-type: none"> 1. Diner receive a notification alert on the restaurant that is in proximity.
Priority	High
Frequency of Use	High
Flow of Events	<ol style="list-style-type: none"> 1. Diner turns on the notification from Use Case NotificationSetting (4.0) 2. System send a notification to the diner of recommended restaurant in proximity. <ol style="list-style-type: none"> a. Notification pop up contains restaurant information such as rating, distance, location.
Alternative Flows	<p>5.2.AC.1.0: If Diner choose to dine at the recommended restaurant by pressing Jiak! Button.</p> <ol style="list-style-type: none"> 1. System directs to Use Case Jiak!AtRestaurant (5.1.2) 2. System returns to Step 1.
Exceptions	NIL
Includes	5.1.2 Jiak!AtRestaurant
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	6.0		
Use Case Name	DiningHistory		
Created by	Lim Xin Wei	Last Updated by	Lim Xin Wei
Date Created	8/2/2024	Date Last Updated	8/2/2024

Actor	Diner
Description	Diner is able to view a record of the restaurants that they have dined at in the past.
Preconditions	1. Diner must have successful login from Login Use Case.
Postconditions	1. System displays a chronological list of restaurants that the diner has eaten at in the past.
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on the dining history button. 2. Diner shall be able to click on their review of a restaurant to see what they have reviewed previously.
Alternative Flows	<p>6.0.AC.1.0 Diner shall able to edit their review of a restaurant they have dined at before. (EditReview 6.1)</p> <p>6.0.AC.2.0 Diner shall able to delete their review of a restaurant they have dined at before. (DeleteReview 6.2)</p>
Exceptions	NIL
Includes	<p>6.1 EditReview</p> <p>6.2 DeleteReview</p>
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	EditReview 6.1 and DeleteReview 6.2 can only be done if there already exist at least one review of a restaurant.

Use Case ID	6.1		
Use Case Name	EditReview		
Created by	Lim Xin Wei	Last Updated by	Le Yanzhi
Date Created	8/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner is able to edit a review of the restaurant that they have dined at in the past.
Preconditions	<ol style="list-style-type: none"> 1. Diner must have successful login from Login Use Case. 2. Diner must already have an existing review of the restaurant to be edited.
Postconditions	<ol style="list-style-type: none"> 1. Systems updated Database with the edited review.
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on the edit button to edit their review. 2. System deletes the review (DeleteReview 6.2) 3. Diner will review restaurant again. (ReviewRestaurant 5.1.2.2) 4. Diner has successfully updated their review of the restaurant. 5. Systems displays a “successfully edited review” message.
Alternative Flows	NIL
Exceptions	NIL
Includes	6.2 DeleteReview
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	6.2		
Use Case Name	DeleteReview		
Created by	Lim Xin Wei	Last Updated by	Le Yanzhi
Date Created	8/2/2024	Date Last Updated	27/3/2024

Actor	Diner
Description	Diner is able to delete a record of the restaurant that they have dined at in the past.
Preconditions	<ol style="list-style-type: none"> 1. Diner must have successful login from Login Use Case. 2. Diner must already have an existing review of the restaurant to be deleted.
Postconditions	<ol style="list-style-type: none"> 1. Systems deleted the review from Database.
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on the delete button to delete their review. 2. Diner confirms the deletion of their review 3. Diner has successfully deleted their review of the restaurant. 4. Systems displays a “Review Deleted!” message.
Alternative Flows	NIL
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	7.0		
Use Case Name	Social		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diner is able to redeem points, and choose to view and add friends, split bills and host or join parties
Preconditions	1. Diner must have successful login from Login Use Case.
Postconditions	2. System allows diners to redeem points, and choose to view and add friends, split bills and host or join parties
Priority	High
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on the Friends button in the homepage UI. 2. Diner should be able redeem point, choose to view and add friends, split bills and host or join parties
Alternative Flows	NIL
Exceptions	NIL
Includes	7.1 RedeemPoints 7.2 SplitCost 7.3 Friends 7.4 Parties
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	7.1		
Use Case Name	RedeemPoints		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diners can see how many points they currently have, and can redeem points from the main social page
Preconditions	1. Diner must have successful login from Login Use Case.
Postconditions	2. System allows diners to redeem points, and obtain discounts through the number of points they have redeemed
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on the Friends button in the homepage UI. 2. Diner inputs the value of points they want to redeem 3. Diner clicks on the redeem button 4. Diner receives confirmation on successful redemption, which can be shown to the cashier to receive discounts
Alternative Flows	<p>1.0.AC.1.0: If current points are less than what they input to redeem</p> <ol style="list-style-type: none"> 1. System displays “Redeem failed. You do not have enough point” <p>1.0.AC.1.1: If they provide an invalid input (special characters or decimal values)</p> <ol style="list-style-type: none"> 1. System displays “Redeem failed. Please enter a valid number of points to redeem”
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	7.2		
Use Case Name	SplitCosts		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diners can see how who is in their party, including the hosts and the guests. Diners can also split the cost of the bill equally among members.
Preconditions	<ol style="list-style-type: none"> 1. Diner must have successful login from Login Use Case. 2. Diner must have friends 3. Diner must be in a party.
Postconditions	<ol style="list-style-type: none"> 1. Diners know who is in their party 2. Diners able to split costs between members in their party
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on the Split Cost button in the Social Main Page 2. Diner inputs the total costs of the bill 3. Diner presses split cost 4. Diner receives confirmation on successful splitting of bill. 5. Diner will be able to see the divided amount in the Friends Page
Alternative Flows	<p>1.0.AC.1.0: If party is empty, of no members</p> <ol style="list-style-type: none"> 1. System displays “Empty Party, Add some friends to start the party” <p>1.0.AC.1.1: If they provide an invalid input (special characters or negative values)</p> <ol style="list-style-type: none"> 1. System displays “Error. Enter a valid cost to split to the party
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	7.3		
Use Case Name	Friends		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diners will be able to add friends. After adding friends, diners will be able to add their friends to a party, or remove friends from a party
Preconditions	1. Diner must have successful login from Login Use Case.
Postconditions	1. Diners will have newly added friends 2. Diners will have friends to add in a party 3. Diners will know how much is owed to their friends.
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on the Friends button in the Social Main Page 2. Diner inputs an existing and correct diner's code 3. Diner adds friend successfully 4. Diner is now able to add friend to a party, or remove them from a party 5. Diner will be able to see the amount owed to their friends
Alternative Flows	<p>1.0.AC.1.0: If they provide an invalid friend code (empty field or input non-existent in the user database.)</p> <ol style="list-style-type: none"> 1. System displays "Unsuccessful. Please enter a valid Eater's code"
Exceptions	NIL
Includes	7.3.3 ViewFriends
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	7.3.1		
Use Case Name	AddFriends		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diners will be able to add friends.
Preconditions	1. Diner must have successful login from Login Use Case.
Postconditions	1. Diners will have newly added friends 2. Diners will have friends to add in a party 3. Diners will know how much is owed to their friends.
Priority	Medium
Frequency of Use	Medium
Flow of Events	1. Diner clicks on the Friends button in the Social Main Page 2. Diner inputs an existing and correct diner's code 3. Diner adds friend successfully
Alternative Flows	1.0.AC.1.0: If they provide an invalid friend code (empty field or input non-existent in the user database.) 1. System displays "Unsuccessful. Please enter a valid Eater's code"
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	7.3.2		
Use Case Name	AddToParty		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diners will be able to add their friends to a party, or remove friends from a party
Preconditions	<ol style="list-style-type: none"> 1. Diner must have successful login from Login Use Case. 2. Diner must have successfully added at least 1 friend.
Postconditions	<ol style="list-style-type: none"> 1. Diner will be able to add friends to their party, or remove friends from their party.
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks “Add Friend To Party” button
Alternative Flows	<p>1.0.AC.1.0: Diner clicks "Remove From Party" button</p> <ol style="list-style-type: none"> 1. The friend will now be removed from the party
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	7.3.3		
Use Case Name	ViewFriends		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diners will be able to view their added friends.
Preconditions	1. Diner must have successful login from Login Use Case.
Postconditions	1. Diners can see their newly added friends 2. Diners will have friends to add in a party
Priority	Medium
Frequency of Use	Medium
Flow of Events	1. Diner clicks on the Friends button in the Social Main Page 2. Diner inputs an existing and correct diner's code 3. Diner adds friend successfully 4. Diner can view their friends that they have added
Alternative Flows	1.0.AC.1.0: If they provide an invalid friend code (empty field or input non-existent in the user database.) 2. System displays "Unsuccessful. Please enter a valid Eater's code"
Exceptions	NIL
Includes	7.3.5 ViewOwedAmount
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	7.3.4		
Use Case Name	ConfirmParty		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diners will be able to add their friends to a party, or remove friends from a party
Preconditions	<ol style="list-style-type: none"> 1. Diner must have successful login from Login Use Case. 2. Diner must have successfully added at least 1 friend.
Postconditions	<ol style="list-style-type: none"> 1. Diner will be able to add friends to their party, or remove friends from their party.
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks “Add Friend To Party” button 2. Diner clicks ‘Confirm Party’ button 3. Diner’s friend will now be added into their party 2. Diner will be able to see the amount owed to their friends
Alternative Flows	<p>1.0.AC.1.0: Diner clicks "Remove From Party" button and then clicks ‘Confirm Party’ button</p> <ol style="list-style-type: none"> 1. The friend will now be removed from the party
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	7.3.5		
Use Case Name	ViewOwedAmount		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diners will be able to see the amount owed to their friends after splitting cost
Preconditions	<ol style="list-style-type: none"> 1. Diner must have successful login from Login Use Case. 2. Diner must have successfully added at least 1 friend 3. Diner must have successfully SplitCost to view owed amount, otherwise default value is 0
Postconditions	<ol style="list-style-type: none"> 1. Diner will be able to add friends to their party, or remove friends from their party.
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks “Friends” button on Social UI 2. Diner will be able to see the amount owed to their friends
Alternative Flows	NIL
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	7.4		
Use Case Name	Parties		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diners will be able to see which party they joined
Preconditions	<ol style="list-style-type: none"> 1. Diner must have successful login from Login Use Case. 2. Diner must have at least 1 friend 3. Diner must have joined a party 4. Diner must not be the host of the party 5. Diner must still be in the party
Postconditions	<ol style="list-style-type: none"> 1. Diner will be able to see the members of the party, including who is the hosts and the guests 2. Diner will have the option to leave the party
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks “Parties” button 2. Diner will be able the members of the party they joined, including who is the hosts and the guests
Alternative Flows	NIL
Exceptions	NIL
Includes	7.4.2 ViewParties
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	7.4.1		
Use Case Name	LeaveParty		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diners will be able to leave the party they joined
Preconditions	<ol style="list-style-type: none"> 1. Diner must have successful login from Login Use Case. 2. Diner must have at least 1 friend 3. Diner must have joined a party 4. Diner must not be the host of the party 5. Diner must still be in the party
Postconditions	<ol style="list-style-type: none"> 1. Diner have left the party 2. System displays “Success. You have left this party!”
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks “Leave this party” button 2. Diner have successfully leave the party
Alternative Flows	NIL
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	7.4.2		
Use Case Name	ViewParties		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diners will be able to see which party they joined
Preconditions	<ol style="list-style-type: none"> 1. Diner must have successful login from Login Use Case. 2. Diner must have joined a party 3. Diner must not be the host of the party 4. Diner must still be in the party
Postconditions	<ol style="list-style-type: none"> 1. Diner will be able to see the members of the party, including who is the hosts and the guests
Priority	Medium
Frequency of Use	Medium
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks “Parties” button 2. Diner will be able the members of the party they joined, including who is the hosts and the guests
Alternative Flows	NIL
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	8.0		
Use Case Name	UserFeedBack		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diner is able to input any opinion or feedback they have about our application
Preconditions	1. Diner must have successful login from Login Use Case.
Postconditions	1. Diner has successfully submitted their user feedback and it will be stored in the firebase such that we, the operators of the application can view.
Priority	Low
Frequency of Use	Low
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on the UserFeedBack button in the homepage UI. 2. Diner would be able to type in their feedback in the input, and submit it to be stored in the firebase.
Alternative Flows	<p>8.0.AC.1.0: If user leaves input field blank</p> <ol style="list-style-type: none"> 1. System displays: "Error. Please enter your feedback before submitting." 2. Goes back to step 2
Exceptions	NIL
Includes	8.1 SubmitFeedBack
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	8.1		
Use Case Name	SubmitFeedBack		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diner is able to submit their opinion or feedback they have about our application
Preconditions	<ol style="list-style-type: none"> 1. Diner must have successful login from Login Use Case. 2. Diner has input their opinion to be submitted
Postconditions	<ol style="list-style-type: none"> 1. Diner has successfully submitted their user feedback and it will be stored in the firebase such that we, the operators of the application can view.
Priority	Low
Frequency of Use	Low
Flow of Events	<ol style="list-style-type: none"> 1. Diner clicks on the UserFeedBack button in the homepage UI. 2. Diner would be able to type in their feedback in the input 3. Diner can submit it to be stored in the firebase.
Alternative Flows	<p>8.1.AC.1.0: If user leaves input field blank</p> <ol style="list-style-type: none"> 3. System displays: “Error. Please enter your feedback before submitting.” 4. Goes back to step 2
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL

Use Case ID	9.0		
Use Case Name	WeatherPage		
Created by	Seah Kah Yen	Last Updated by	Seah Kah Yen
Date Created	8/4/2024	Date Last Updated	8/4/2024

Actor	Diner
Description	Diner is able to view the weather page
Preconditions	1. Diner must have successful login from Login Use Case.
Postconditions	2. Diner would know the current weather in their area.
Priority	Low
Frequency of Use	Low
Flow of Events	1. Diner clicks on the weather button in the homepage UI. 2. Diner would know the current weather
Alternative Flows	NIL
Exceptions	NIL
Includes	NIL
Special Requirements	NIL
Assumptions	NIL
Notes and Issues	NIL