**Yingqiao Xiong, Scott Woods, Nathalie Langlois, Yue Zhou**

Version 2.5 | 3/15/16

Use Cases for Restaurant Recommender

Use cases

# Use Case List

|  |  |  |
| --- | --- | --- |
| ID | Primary Actor | Use Case Title |
| 1 | User | Log in |
| 2 | User | Create a restaurant category |
| 3 | User | Generate a restaurant recommendation |
| 4 | User | Update restaurant history |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-1 | | |
| Use Case Name: | Log in | | |
| Created By: | Nathalie Langlois | **Last Updated By:** | Nathalie Langlois |
| Date Created: | 3/13/16 | **Date Last Updated:** | 3/13/16 |

|  |  |
| --- | --- |
| Actors: | App User |
| Description: | When app is started, user must log in using existing credentials. |
| Trigger: | User opens app |
| Pre-Conditions: | None |
| Post-Conditions: | User has logged in and been authenticated. |
| Normal Flow: | UC-1.0 Log in   1. User enters username 2. User enters password 3. User selects “Log in” 4. System authenticates username and password combination    1. System sends information to data store    2. System validates the input 5. System displays main page with user’s personalization |
| Alternate Flows: | UC-1.1 User must create account   1. User selects “Create Account” 2. System prompts user for account details    1. Enter email    2. Enter username    3. Enter password twice 3. System validates input against requirements    1. If invalid, highlights incorrect fields and return to step 2 4. System displays confirmation message 5. System stores user information in data store 6. System displays main page   UC-1.2 User enters invalid log in information  *Begin after step 4.1*   1. System does not find username and password combination 2. System alerts user that the information is incorrect 3. System prompts user to try again 4. Return to normal flow step 1 |
| Exceptions: | None |
| Includes: | None |
| Priority: | High |
| Frequency of Use: | 1x per use of app |
| Business Rules: | User may only have one account |
| Special Requirements: | All database calls must take under 1 second.  Initial load time must be under 1 second. |
| Assumptions: | None |
| Notes and Issues: | Connect log in to another service (e.g. Facebook, Google)  Username vs. email  Password requirements  Forgot password button  Number of login attempts |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-2 | | |
| Use Case Name: | Create a restaurant category | | |
| Created By: | Nathalie Langlois | **Last Updated By:** | Nathalie Langlois |
| Date Created: | 3/13/16 | **Date Last Updated:** | 3/13/16 |

|  |  |
| --- | --- |
| Actors: | App User |
| Description: | User creates restaurant category (“Food Mood”) based off of inputted attributes. |
| Trigger: | User clicks button to create a food mood |
| Pre-Conditions: | User has logged in and been authenticated. |
| Post-Conditions: | User has created a Food Mood.  Food mood is stored as part of user’s account configuration.  Food mood is displayed on main page. |
| Normal Flow: | UC-2.0 Create a restaurant category   1. User is prompted to enter a name for the food mood 2. User is prompted to select if desired option filter is cuisine, price, or dining type 3. System displays elements for user within the selected option 4. User selects an element 5. System prompts user to enter more options    1. Return to step 1 (Repeat until at most 5 elements selected) 6. User selects “Done” 7. System stores food mood name and elements 8. System displays main page with food mood list updated |
| Alternate Flows: | UC-2.1 User does not complete food mood  *Begin at any time*   1. User clicks to exit food mood creator 2. System prompts user to finish creating the food mood    1. User clicks to continue, resume normal flow 3. User clicks to close 4. System displays main page without storing food mood |
| Exceptions: | None |
| Includes: | UC-1 Log in process |
| Priority: | High |
| Frequency of Use: | 3x per use of the app |
| Business Rules: | A restaurant category can have up to 5 elements. |
| Special Requirements: | All database calls must take under 1 second.  Initial load time must be under 1 second. |
| Assumptions: | None |
| Notes and Issues: | Other attribute types: speed, location/distance, takeout vs. delivery vs. eat in, stars  Review entire normal flow  Should user be able to submit/create new elements for food mood (e.g. “Don’t like your options? Tell us what we are missing”) |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-3 | | |
| Use Case Name: | Generate restaurant recommendation | | |
| Created By: | Nathalie Langlois | **Last Updated By:** | Nathalie Langlois |
| Date Created: | 2/16/16 | **Date Last Updated:** | 3/13/16 |

|  |  |
| --- | --- |
| Actors: | App User |
| Description: | After either selecting an existing restaurant category (“Food Mood”) or generating a new one, the system will return a recommendation based on the Food Mood and the user’s history. |
| Trigger: | User clicks button to generate a new recommendation |
| Pre-Conditions: | User has logged in and been authenticated.  User needs location services enabled.  User has created a Food Mood |
| Post-Conditions: | User has selected a restaurant.  Restaurant has been added to the user’s history.  System is set to prompt user for a rating of their experience. |
| Normal Flow: | UC-3.0 Generate a restaurant recommendation   1. User selects existing Food Mood 2. System retrieves user preferences from data store 3. System retrieves user’s location    1. System retrieves options from Yelp    2. Selects restaurants that are open    3. Selects restaurants that match the food mood 4. Selects restaurants that are near the user’s location 5. System generates recommendation based on this information 6. System displays restaurant 7. User selects restaurant 8. System displays fastest route to the restaurant |
| Alternate Flows: | UC-3.1 User does not select recommended restaurant  *Begin after step 6*   1. User opts to see another recommendation 2. System generates a second recommendation 3. Resumes normal flow at step 7   UC-3.2 User does not select any restaurant   1. User exits the Food Mood |
| Exceptions: | UC-3.0.E.1 Cannot find any recommendations  *Begin at any point in the process*   1. System does not find any restaurants that match the given specifications 2. System displays error message 3. System prompts user to try a different food mood 4. Resume normal flow at step 1 |
| Includes: | UC-1 Log in process  UC-2 Create Food Mood |
| Priority: | High |
| Frequency of Use: | 2x per use of the app |
| Business Rules: | Restaurants are not displayed if they have less than 3 stars.  Restaurant must be within 10 miles of the user.  User can continue to see more recommendations until they have seen them all. |
| Special Requirements: | All database calls must take under 1 second.  Initial load time must be under 1 second. |
| Assumptions: | Yelp servers are always available. |
| Notes and Issues: | Options for Food Mood (location, cuisine, etc.)  Location radius (customizable? 10 miles?)  Criteria that system uses to recommend restaurants  Incorporate reservation booking (Open Table)  Potential business rule: only show restaurants that we can make money from  If user goes through every recommendation without selecting one, does it go to the exception or the alternate course?  If a user goes through a certain number of recommendations, should it suggest that they select a new Food Mood? (Possible alternate flow)  How many restaurants does the system grab at a time? |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | UC-4 | | |
| Use Case Name: | Update restaurant history | | |
| Created By: | Nathalie Langlois | **Last Updated By:** | Nathalie Langlois |
| Date Created: | 3/13/16 | **Date Last Updated:** | 3/13/16 |

|  |  |
| --- | --- |
| Actors: | App User |
| Description: | After selecting a restaurant, the user will be prompted to review the restaurant the next time they open the app. |
| Trigger: | User opens app |
| Pre-Conditions: | User has logged in and been authenticated.  User selected a restaurant during most recent use of app.  Restaurant has been added to the user’s history.  System is set to prompt user for a rating of their experience. |
| Post-Conditions: | User’s restaurant history has been updated with review.  Restaurant review has been added to data store. |
| Normal Flow: | UC-4.0 Update restaurant history   1. System prompts user to rate their most recent restaurant selection 2. User rates restaurant out of five stars 3. System sends rating to data store 4. System updates user’s personal restaurant history 5. System displays main page with user’s personalization |
| Alternate Flows: | None |
| Exceptions: | None |
| Includes: | UC-1 Log in process  UC-2 Create Food Mood  UC-3 Generate restaurant recommendation |
| Priority: | Medium |
| Frequency of Use: | 1x per use of the app |
| Business Rules: | None |
| Special Requirements: | All database calls must take under 1 second.  Initial load time must be under 1 second. |
| Assumptions: | None |
| Notes and Issues: | Are there any exceptions or business rules? This seems too simple |