Use Cases

for

NomNom

Version 2.0

Prepared by Veena Dali, Erika Becerra De La Roca, Ziyan Huo, &
Timothy Bagby

Team 7

4/24/16

1. Guidance for Use Case Template

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

2. Use Case Identification

2.1. Use Case ID

Give each use case a unique integer sequence number identifier. Alternatively, use a hierarchical form: X.Y. Related use cases can be grouped in the hierarchy.

2.2. Use Case Name

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

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

2.3. Use Case History

2.3.1. Created By

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

2.3.2. Date Created

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

2.3.3. Last Updated By

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

2.3.4. Date Last Updated

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

3. Use Case Definition

3.1. Actors

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

3.2. Trigger

Identify the event that initiates the use case. This could be an external business event or system event that causes the use case to begin, or it could be the first step in the normal flow.

3.3. Description

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

3.4. Preconditions

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

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

3.5. Postconditions

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

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

3.6. Normal Flow

Provide a detailed description of the user actions and system responses that will take place during execution of the use case under normal, expected conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description. This description may be written as an answer to the hypothetical question, “How do I <accomplish the task stated in the use case name>?” This is best done as a numbered list of actions performed by the actor, alternating with responses provided by the system. The normal flow is numbered “X.0”, where “X” is the Use Case ID.

3.7. Alternative Flows

Document other, legitimate usage scenarios that can take place within this use case separately in this section. State the alternative flow, and describe any differences in the sequence of steps that take place. Number each alternative flow in the form “X.Y”, where “X” is the Use Case ID and Y is a sequence number for the alternative flow. For example, “5.3” would indicate the third alternative flow for use case number 5.

3.8. Exceptions

Describe any anticipated error conditions that could occur during execution of the use case, and define how the system is to respond to those conditions. Also, describe how the system is to respond if the use case execution fails for some unanticipated reason. If the use case results in a durable state change in a database or the outside world, state whether the change is rolled back, completed correctly, partially completed with a known state, or left in an undetermined state as a result of the exception. Number each alternative flow in the form “X.Y.E.Z”, where “X” is the Use Case ID, Y indicates the normal (0) or alternative (>0) flow during which this exception could take place, “E” indicates an exception, and “Z” is a sequence number for the exceptions. For example “5.0.E.2” would indicate the second exception for the normal flow for use case number 5.

3.9. Includes

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

3.10. Priority

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

3.11. Frequency of Use

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

3.12. Business Rules

List any business rules that influence this use case.

3.13. Special Requirements

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

3.14. Assumptions

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

3.15. Notes and Issues

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

Primary Use Case List

ID	*Primary Actor*	*Use Case Title*
1	Registered User	User signs onto site with unique account
2	Registered User	Search for restaurant with name, cuisine-type, friend, or map
3	Registered User	Create/update profile for registered user with reviews and recommendations
4	Registered User	Add Friends

Use Case 1

Use Case ID:	1		
Use Case Name:	User signs onto site with unique account		
Created By:	Tim B.	Last Updated By:	Tim B.
Date Created:	2/28/16	Date Last Updated:	5/2/16

Actors:	User
Description:	To access site
Trigger:	User wants to personal profile
Preconditions:	1. User access site
Postconditions:	1. User is able to access the rest of the site's functionality
Normal Flow:	1.0.1. User on main page 1.0.2. User clicks on login button 1.0.3. User enters registration info via Facebook 1.0.4. Registered user enter main profile page
Alternative Flows:	1.1.1. User initially presented with login screen 1.1.2. User enters credentials 1.1.3. User enters main page
Exceptions:	1.1.1.1. User has option to register 1.1.1.2. User has to pick unique credentials that are not currently on file 1.0.3.1. User has option to register 1.0.3.2. User has to pick unique credentials that are not currently on file
Includes:	N/A
Priority:	High
Frequency of Use:	Once to many per site visit/ cookie wipe
Business Rules:	Secure Login
Special Requirements:	1. Password is not displayed/ hidden when typed 2. Password storing to done securely

Assumptions:	User is registered, otherwise must register
Notes and Issues:	N/A

Use Case 2

Use Case ID:	2		
Use Case Name:	Search for restaurant with name, cuisine-type, friend, or map		
Created By:	Erika B.	Last Updated By:	Veena D.
Date Created:	2/19/16	Date Last Updated:	5/2/16

Actors:	Registered user
Description:	To find new restaurants to try
Trigger:	User does not know which restaurant they want to go to so they use the search option on NomNom to find one
Preconditions:	
Postconditions:	1. Depending on which option the user uses to search for the restaurant, display a list of restaurants depending on that 2. Find restaurant to go eat at
Normal Flow:	2.0.1. User enters keyword to search 2.0.2. User clicks “Search” or “enter” on keyboard 2.0.3. System lists the different options for the user depending on the keyword 2.0.4. User clicks on the element that they want to view in the list 2.0.5. Registered user with friends ‘Log In’ 2.0.6. Registered user views friend’s profiles or registered user finds friends
Alternative Flows:	2.1.1. Registered user clicks on search option and identifies how he/she wants to search for a restaurant 2.1.2. User enters keyword 2.1.3. System presents user with a list of options that relate to the keyword entered
Exceptions:	2.1.3.1. Suggests finding friends when the user does not have friends to search

Includes:	1, 4 (only if you want to search by friend)
Priority:	High
Frequency of Use:	Once to many per site visit
Business Rules:	Privacy Settings
Special Requirements:	1. Visually display the list of options
Assumptions:	The user is registered and knows how to access the search function
Notes and Issues:	The search by friend is does not currently working because right now it does not allow the user to add friends.

Use Case 3

Use Case ID:	3		
Use Case Name:	Create/Update profile for registered user with reviews and recommendations		
Created By:	Veena Dali	Last Updated By:	Erika Becerra
Date Created:	3/11/16	Date Last Updated:	3/11/16

Actors:	Registered User
Description:	To create a profile for a registered user where they can review restaurants, add friends, and see which restaurants are recommended to them.
Trigger:	User would like to display recommendations and reviews
Preconditions:	1. Must be a registered user 2. Must be signed in
Postconditions:	1. The user's recommendations/reviews will appear on his or her profile 2. This info will be searchable
Normal Flow:	3.0.1 User will be asked to sign in 3.0.2 Update profile option will be available 3.0.3 Review a restaurant 3.0.4 Recommend a restaurant 3.0.5 Save to profile
Alternative Flows:	3.1.1 Search a restaurant to review 3.1.2 Click the review restaurant 3.2.1 On user's profile go to the top 5 recommendation section 3.2.2 Update recommendations button 3.2.3 Add restaurant to user's top 5 recommendation section 3.2.4 Save updated recommendation list
Exceptions:	
Includes:	1
Priority:	High

Frequency of Use:	Once to many per site
Business Rules:	
Special Requirements:	
Assumptions:	Has a unique account
Notes and Issues:	TBD

Use Case 4

Use Case ID:	4		
Use Case Name:	Add Friends		
Created By:	Tim B.	Last Updated By:	Tim B.
Date Created:	2/29/16	Date Last Updated:	2/29/16

Actors:	Registered User
Description:	Add friends
Trigger:	User wants to see reviews from people they know
Preconditions:	1. User has logged onto site 2. At least one other users exists on the site
Postconditions:	1. User is connected with other user, and sees their reviews of different restaurants
Normal Flow:	4.0.1. User is on homepage 4.0.2. User clicks on "Add Friend" button 4.0.3. User types in name of user 4.0.4. User is prompted with "add friend" button 4.0.5. User now has access to view friend's reviews
Alternative Flows:	4.1.1. User is on homepage 4.1.2. User enter in search field on the homepage a user's name 4.1.3. If the user is not already friends, user is prompted if they want to add 4.1.4. If the user accepts, user can access friend's reviews
Exceptions:	4.0.3.1 Error returns that user with such a name does not exist on the site. 4.1.2.1 Error returns that user with such a name does not exists on the site
Includes:	1
Priority:	High
Frequency of Use:	Medium to Low (depending on how many people the user wants to friend)

Business Rules:	Privacy Setting of user accounts (i.e.e whether they choose their account to be visible to other user or not)
Special Requirements:	1. Visual way to display search results for users 2. Prevent already friended user to be friended again
Assumptions:	At least two users are registered that want to be friends
Notes and Issues:	N/A

Revision History

Name	Date	Reason For Changes	Version
Veena	2/19/16	Specified Trigger for 2C use case (original: Want to be informed of what your friends suggest as restaurants)	1.0
Tim Bagby	2/29/16	Added primary case 3;	1.0
Veena	3/11/16	Updated use case 3 (create review for restaurant) to include use case 5 (On each profile it would have their food preferences (e.g. Vegan)); deleted use case 5	2.0
Veena	3/11/16	Consolidated 2A,2B,2C,2D into use case 2	2.0
Veena	3/11/16	Included use case 6 (Each user's most recent rated restaurants) and use case 7 (Recommended Restaurants) in use case 3	2.0
Veena	3/11/16	Changed use case 8 to use case 4	2.0
Veena	3/11/16	Deleted use case 9	2.0
Veena	3/11/16	Deleted use case 4 and 8 details (can still be found in version 1.0, if needed)	2.0
Veena	3/11/16	Updated use case 2 details to include name, cuisine-type, and map	2.0
Erika	3/12/16	Updated Use cases 3 up until alternative flows	2.0
Ziyan	3/12/16	Added use case 4 implemented by Tim	2.0
Veena	4/24/16	Deleted precondition for Use Case 2 'User has been registered' because user does not have to be registered to be able to search	2.0
Veena	4/24/16	Deleted first step in normal flow for Use Case 2 "Registered user is on home page" because of the changes that we made in the final product	2.0
Veena	4/24/16	Deleted exception in Use Case 2 "Unregistered User is asked to register"	2.0
Veena	4/24/16	Added to Normal Flow for Use Case 2	2.0

Veena	5/2/16	Updated 1.0.3 (in use case 3) to “User enters registration info via Facebook” because we have set-up the registration using facebook	2.0
Veena	5/2/16	Deleted Exception “2.1.3.2 Suggests providing location if it is not in the system already”	2.0