Use Cases

for

Mall-E

Version 1.0 approved

Prepared by Chua Chong Wei Melvin, Chua Chong Hung Kelvin

30/01/2021

Revision History

Name	Date	Reason For Changes	Version
Melvin, Kelvin	30/1/2021	Creation of Use Cases	1.0

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.

Use Case List

Primary Actor	***Use Cases***
User	Crowd Density
User	Waiting Time at Restaurants
User	Covid News for Shopping Malls
User	Account Login
User	Forgot Password
User	Create Account

Use Case 1: Crowd Density

Use Case Description:

Use Case ID:	1		
Use Case Name:	Crowd Density		
Created By:	Melvin Chua, Kelvin Chua	Last Updated By:	Melvin Chua, Kelvin Chua
Date Created:	30/1/2021	Date Last Updated:	30/1/2021

Actors:	User, Best Time API
Description:	The latest Crowd Density data of Shopping Malls will be pull from the Best Time API
Trigger:	After the user have successfully login
Preconditions:	1. The application must have a stable connection with the internet 2. Best Time API must be up and running
Postconditions:	1. When the user are able to see the latest crowd density of shopping malls
Normal Flow:	1.1 The application will send a request to the Best Time API. 1.2 The application will receive a response of Crowd Density Data in Shopping malls from the Best Time API
Alternative Flows:	NIL
Exceptions:	1.1.E.1 If a request to the API cannot be sent, the system will inform the user 1.2.E.1 If the response from the API cannot be received, the system will inform the user
Includes:	NIL
Priority:	1
Frequency of Use:	1 time every minute
Business Rules:	NIL
Special Requirements:	The crowd density is indicated by their colours 1. Red (>80%) 2. Yellow (>50% & <80%) 3. Green (<50%)
Assumptions:	API consists of the most updated Crowd Density Data in Shopping Malls
Notes and Issues:	NIL

Use Case 2: Waiting Time of Restaurants

Use Case Description:

Use Case ID:	2		
Use Case Name:	Waiting Time of Restaurant		
Created By:	Melvin Chua, Kelvin Chua	Last Updated By:	Melvin Chua, Kelvin Chua
Date Created:	30/1/2021	Date Last Updated:	30/1/2021

Actors:	User, Waiting Time API
Description:	The latest Waiting Time of Restaurants data of Shopping Malls will be pull from the Waiting Time API
Trigger:	After the user have successfully login
Preconditions:	1. The application must have a stable connection with the internet 2. The Waiting Time API must be up and running
Postconditions:	1. When users are able to see the latest waiting time of restaurants
Normal Flow:	2.1 The application will send a request to the Waiting Time API 2.2 The application will receive a response of Waiting Time Data in Shopping malls from the Waiting Time API
Alternative Flows:	NIL
Exceptions:	2.1.E.1 If a request to the API cannot be sent, the system will inform the user 2.2.E.1 If the response from the API cannot be received, the system will inform the user
Includes:	NIL
Priority:	2
Frequency of Use:	1 time every minute
Business Rules:	NIL
Special Requirements:	NIL
Assumptions:	API consists of the most updated Restaurant Waiting Time Data in Shopping Malls
Notes and Issues:	NIL

Use Case 3: Covid News for Shopping Malls

Use Case Description:

Use Case ID:	3		
Use Case Name:	Covid News for Shopping Malls		
Created By:	Melvin Chua, Kelvin Chua	Last Updated By:	Melvin Chua, Kelvin Chua
Date Created:	30/1/2021	Date Last Updated:	30/1/2021

Actors:	User, Google News API
Description:	The latest news Articles relating to Covid 19 is pulled from Google News API
Trigger:	When the user presses the tab to show the recent news articles
Preconditions:	1. The application must have a stable connection with the internet 2. Google News API must be up and running
Postconditions:	1. A list of latest News articles relating to Covid 19 in shopping malls is shown to the user
Normal Flow:	3.1 The application will send a request to Google News API. 3.2 The application will receive a response of News Articles from the API
Alternative Flows:	NIL
Exceptions:	3.1.E.1 If a request to the API cannot be sent, the system will inform the user 3.2.E.1 If the response from the API cannot be received, the system will inform the user
Includes:	NIL
Priority:	3
Frequency of Use:	1 time every 2 minutes
Business Rules:	NIL
Special Requirements:	Users must be able to read the news headlines easily
Assumptions:	API consists of the most updated News Articles relating to Covid 19 at Shopping Malls
Notes and Issues:	NIL

Use Case 4: Account Login

Use Case Description:

Use Case ID:	4		
Use Case Name:	Account Login		
Created By:	Melvin Chua, Kelvin Chua	Last Updated By:	Melvin Chua, Kelvin Chua
Date Created:	30/1/2021	Date Last Updated:	30/1/2021

Actors:	User
Description:	The User can login their account that they have created
Trigger:	When the user opens the application
Preconditions:	1.The application must have a stable connection with the internet 2.The database (MongoDB) must be up and running
Postconditions:	1.Success message showing that the login was successful.
Normal Flow:	4.1 User will key in they Username and Password 4.2 User presses on the “Login” button 4.3 The application will send a GET query to database and check if the user credentials are correct
Alternative Flows:	Wrong Password 4.3.1 Query result in Wrong password 4.3.2 Go back to Step 4.1 Username does not exists 4.3.1 Query result in account not existing 4.3.2 Application will inform the user to Create a account 4.3.3 Go back to Step 4.1
Exceptions:	3.1.E.1 If a query to the database cannot be sent, the system will inform the user. 3.2.E.1 if the user log in and fail more than 10 times, The application will inform that account will be locked and have to contact user account support
Includes:	1. Forgot Password 2. Create Account
Priority:	4
Frequency of Use:	1 time every 30 minutes
Business Rules:	NIL
Special Requirements:	NIL
Assumptions:	NIL
Notes and Issues:	NIL

Use Case 5: Forgot Password

Use Case Description:

Use Case ID:	5		
Use Case Name:	Forgot Password		
Created By:	Melvin Chua, Kelvin Chua	Last Updated By:	Melvin Chua, Kelvin Chua
Date Created:	30/1/2021	Date Last Updated:	30/1/2021

Actors:	User
Description:	For the user to reset their password if they have forgotten their password
Trigger:	When the "forgot password" button is pressed
Preconditions:	1. The application must have a stable connection with the internet 2. The database (MongoDB) must be up and running
Postconditions:	1. The user will be able to reset their password 2. The user's password is updated
Normal Flow:	5.1 User key in their email 5.2 User presses "confirm" button 5.3 Application will send an email to the user containing a temporary password. 5.4 The user will login to the application using the temporary password 5.5 User enter their new password 5.6 User presses "Confirm Password" 5.6 The application will sent an UPDATE query to the database
Alternative Flows:	Unable to send email 5.3.1 If the email to the user cannot be sent, allow user to press "Send email again" Button after a certain time out (3 minutes) 5.3.2 Go back to step 5.3 Wrong Password 5.4.1 Query result in Wrong password 5.4.2 Go back to Step 5.4
Exceptions:	NIL
Includes:	NIL
Priority:	5
Frequency of Use:	1 time every 30 minutes
Business Rules:	NIL
Special Requirements:	User enter password 2 times in 2 fields to ensure that the password is matching
Assumptions:	NIL
Notes and Issues:	NIL

Use Case 6: Create Account

Use Case Description:

Use Case ID:	5		
Use Case Name:	Create Account		
Created By:	Melvin Chua, Kelvin Chua	Last Updated By:	Melvin Chua, Kelvin Chua
Date Created:	30/1/2021	Date Last Updated:	30/1/2021

Actors:	User
Description:	For the user to create their account in order to use the application
Trigger:	When the “create account” button is pressed
Preconditions:	1. The application must have a stable connection with the internet 2. The database (MongoDB) must be up and running
Postconditions:	1. The user would have their own account created 2. There is a new entry in the database
Normal Flow:	5.1 User key in their email 5.2 User key in their desired password 5.3 User presses “confirm” button 5.4 The application will sent an POST query to the database
Alternative Flows:	Wrong email format 5.3.1 If the email has the incorrect format, it would inform the user to correct the email 5.3.2 Go back to Step 5.1 Not matching passwords 5.4.1 Query result in Wrong password 5.4.2 Go back to Step 5.4
Exceptions:	NIL
Includes:	NIL
Priority:	5
Frequency of Use:	1 time every 30 minutes
Business Rules:	NIL
Special Requirements:	User enter password 2 times in 2 fields to ensure that the password is matching
Assumptions:	NIL
Notes and Issues:	NIL