E-commerce Web Application

Version <1.0>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 15/11/2020 | 1.0 |  | Huỳnh Nhật Nam, Nguyễn Phúc Thịnh |
| 8/12/2020 | 2.0 | Edit usecase (from go shopping to view product, merge manage product to global use-case model), add view order history, edit profile and add specification for Edited usecase | Huỳnh Nhật Nam, Nguyễn Phúc Thịnh |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Use-case Model 4

2. Use-case Specifications 5

2.1 Use-case: Sign in 5

2.2 Use-case: Create an account 6

2.3 Use-case: Make purchase 6

2.4 Use-case: Search product 7

2.5 Use-case: Browse product 7

2.6 Use-case: View cart 8

2.7 Use-case: Edit product in cart 8

2.8 Use-case: Remove product from cart 8

2.9 Use-case: Add a product to cart 8

2.10 Use-case: Payment 9

2.11 Use-case: Payment on delivery 9

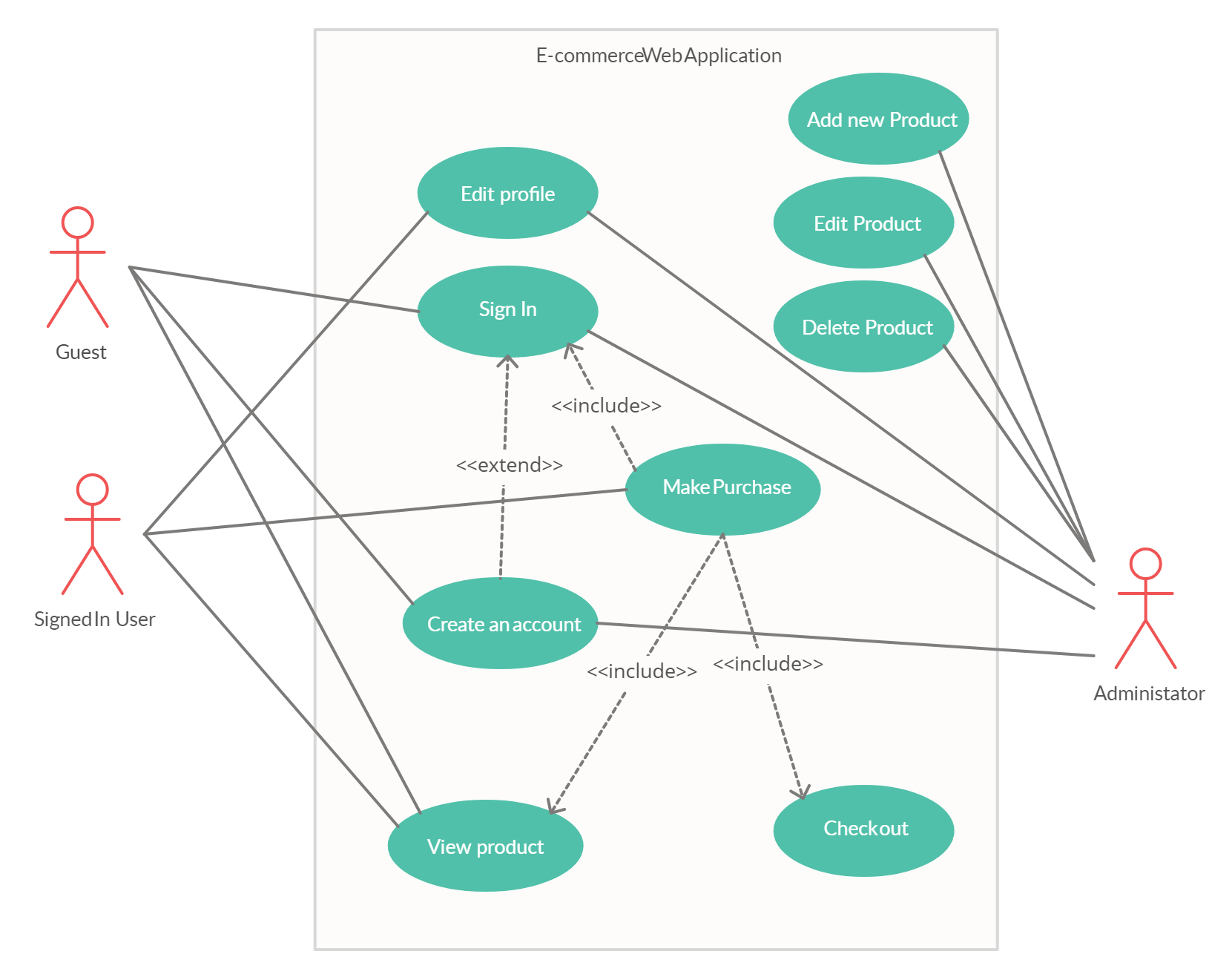
2.12 Use-case: Payment online 10

2.13 Use-case: Add new product 11

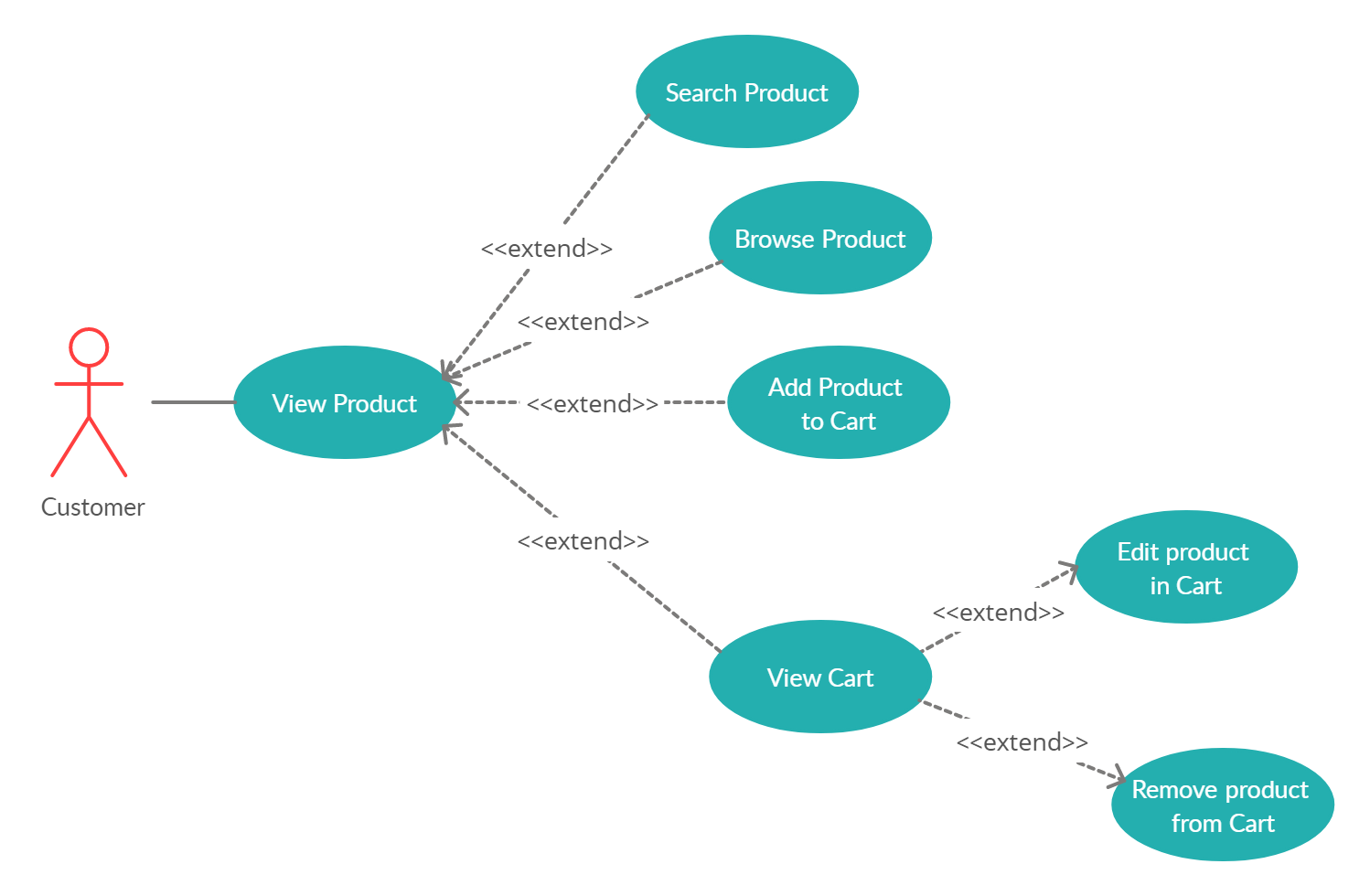
2.14 Use-case: Edit product 11

2.15 Use-case: Delete product 11

# Use-case Model



* View product model:



* Check Out model:

# Use-case Specifications

## Use-case: Sign in

|  |  |
| --- | --- |
| Use case Name | Sign in |
| Brief description | This use case describes how a user can sign in |
| Actors | Guest, Administrator |
| Basic Flow | 1. User enters username and password 2. Return to homepage if user signed in successfully |
| Alternative Flows | **Alternative flow 1: User enters wrong password**   1. System displays an error message 2. Continue step #1 of basic flow   **Alternative flow 2: User enters non-existent username**   1. System displays an error message 2. Continue step #1 of basic flow |
| Pre-conditions | User goes to login page through 1 of 3 ways:  - Click on sign in button at the top right of the home page  - User is not signed in and clicks on view cart button at the top right of the home page  - User is not signed in and adds a product to cart |
| Post-conditions | Guest successfully signs in and return to homepage |

## Use-case: Create an account

|  |  |
| --- | --- |
| Use case Name | Create an account |
| Brief description | This use case describes how a user can create an account |
| Actors | Guest, administrator |
| Basic Flow | 1. At sign in page, user clicks on sign up button. 2. User goes to sign up page. 3. User enters account’s name 4. User enters password 5. User enters password again 6. User clicks sign up 7. Goes back to sign in page |
| Alternative Flows | **Alternative flow 1: User enters username that’s already taken**   1. From #6 of the basic flow, system displays error message 2. Continue from step #3.   **Alternative flow 2: User re-enters wrong password**   1. From #6 of the basic flow, system displays error message 2. Continue from step #5. |
| Pre-conditions | User goes to login page and clicks sign up button |
| Post-conditions | User successfully signs up and returns to sign in page |

## Use-case: Make purchase

|  |  |
| --- | --- |
| Use case Name | Make purchase |
| Brief description | This use case describes how a user can make a purchase |
| Actors | Signed in customer |
| Basic Flow | 1. At homepage, user goes to their cart by clicking on cart button 2. User click on purchase button 3. User click on checkout button. 4. User check their information again 5. User click on confirm button |
| Alternative Flows | **Alternative flow 1: User’s shopping cart is empty**   1. From #3 of the basic flow, systerm display error message 2. Systerm goes to homepage   **Alternative flow 2: User information is wrong**   1. From #4 of the basic flow, user re-enter their information 2. Continue step #5 |
| Pre-conditions | User goes to homepage  User has product in shopping-cart  User has correctly information in profile |
| Post-conditions | User successfully make a purchase and waiting for an email to confirm |

## Use-case: Search product

|  |  |
| --- | --- |
| Use case Name | Search product |
| Brief description | This use case describes how a user can search a product by search bar |
| Actors | Guest, administrator, signed in user |
| Basic Flow | 1. User goes to homepage 2. User clicks on search bar 3. User enter a text 4. System displays list of products according to the searching result 5. User clicks on a product 6. User goes to a page which displays that product’s information |
| Alternative Flows | **Alternative flow 1: User can’t not find the product**   1. From #4 of the basic flow, system displays no product is found 2. Continue step #2 |
| Pre-conditions | User goes to homepage |
| Post-conditions | User view a product’s information |

## Use-case: Browse product

|  |  |
| --- | --- |
| Use case Name | Browse product |
| Brief description | This use case describes how a user can browse product |
| Actors | Guest, administrator, signed in user |
| Basic Flow | 1. User goes to homepage 2. Using navigation bar, user can see types of product 3. User clicks on a type on navigation bar 4. System displays a list of products belongs to that specific type 5. User clicks on a product 6. System displays a page of that product’s information |
| Pre-conditions | User goes to homepage |
| Post-conditions | User views a product’s information |

## Use-case: View cart

|  |  |
| --- | --- |
| Use case Name | View cart |
| Brief description | This use case describes how a user can view his/her cart |
| Actors | Signed in user |
| Basic Flow | 1. User clicks on icon of the cart 2. System displays user’s cart (a page) |
| Pre-conditions | - User goes to homepage  - User has already signed in |
| Post-conditions | User views cart |

## Use-case: Edit product in cart

|  |  |
| --- | --- |
| Use case Name | Edit product in cart |
| Brief description | This use case describes how a user can edit product in cart |
| Actors | Signed in user |
| Basic Flow | 1. User views cart 2. User changes quantity of products 3. User adds requirements to products |
| Pre-conditions | - User goes to homepage  - User has already signed in. |
| Post-conditions | User changes product’s quantity in cart or add requirements |

## Use-case: Remove product from cart

|  |  |
| --- | --- |
| Use case Name | Remove product from cart |
| Brief description | This use case describes how a user can remove product from cart |
| Actors | Signed in user |
| Basic Flow | 1. User views cart 2. User clicks on remove button of a product |
| Pre-conditions | - User goes to homepage  - User has already signed in. |
| Post-conditions | A product is removed from cart |

## Use-case: Add a product to cart

|  |  |
| --- | --- |
| Use case Name | Add a product to cart |
| Brief description | This use case describes how a user can add a product to cart |
| Actors | Signed in user |
| Basic Flow | 1. User browses or searchs a product 2. At product’s information page, user clicks on add to cart button 3. System adds product to user’s cart |
| Alternative Flows | **Alternative flow 1: Product is out of stock**   1. After step #2 of basic flow, system displays error message 2. Continue from step #1.   **Alternative flow 2: Product is already in cart**   1. From step #3 of basic flow, system increases product’s quantity in cart |
| Pre-conditions | User goes to homepage  User has already signed in |
| Post-conditions | User adds product to cart or increases product’s quantity |

## Use-case: Payment

|  |  |
| --- | --- |
| Use case Name | Add a product to cart |
| Brief description | This use case describes how a user can do a payment |
| Actors | Signed in custormer |
| Basic Flow | 1. User goes to shopping cart 2. User click on purchase button 3. User click on checkout button 4. User choose payment methods 5. User click on confirm button |
| Alternative Flows | **Alternative flow 1: User’s shopping cart is empty**   1. From #2 of the basic flow, systerm display error message 2. Systerm goes to homepage   **Alternative flow 2: Payment methods is not available**   1. From step #4 of basic flow, system display error message 2. Goes to #4 |
| Pre-conditions | User goes to shopping cart  User has already signed in  User has already product in shopping cart |
| Post-conditions | User successfully to pay for a purchase. |

## Use-case: Payment on delivery

|  |  |
| --- | --- |
| Use case Name | Payment on delivery |
| Brief description | This use case describes how a user can choose payment method is pay when receive product |
| Actors | Signed in custormer |
| Basic Flow | 1. User goes to shopping cart 2. User click on purchase button 3. User click on checkout button 4. User choose payment on delivery 5. User click on confirm button |
| Alternative Flows | **Alternative flow 1: User’s shopping cart is empty**   1. From #2 of the basic flow, systerm display error message 2. Systerm goes to homepage   **Alternative flow 2: Payment methods is not available**   1. From step #4 of basic flow, system display error message 2. Goes to #4 and choose other payment methods |
| Pre-conditions | User goes to shopping cart  User has already signed in  User has already product in shopping cart |
| Post-conditions | User successfully to pay for a purchase. |

## Use-case: Payment online

|  |  |
| --- | --- |
| Use case Name | Payment online |
| Brief description | This use case describes how a user can choose payment method is pay when receive product |
| Actors | Signed in custormer |
| Basic Flow | 1. User goes to shopping cart 2. User click on purchase button 3. User click on checkout button 4. User choose payment online 5. User click on confirm button |
| Alternative Flows | **Alternative flow 1: User’s shopping cart is empty**   1. From #2 of the basic flow, systerm display error message 2. Systerm goes to homepage   **Alternative flow 2: Payment methods is not available**   1. From step #4 of basic flow, system display error message 2. Goes to #4 and choose other payment methods   **Alternative flow 3: Credit card not available**   1. From step #4 of basic flow, systerm display error message 2. Goes to #4 and re-enter. |
| Pre-conditions | User goes to shopping cart  User has already signed in  User has already product in shopping cart |
| Post-conditions | User successfully to pay for a purchase. |

## Use-case: Add new product

|  |  |
| --- | --- |
| Use case Name | Add new product |
| Brief description | This use case describes how administrator can add new product |
| Actors | Administrator |
| Basic Flow | 1. At administrator page, admin clicks on add new product button 2. System displays a new page 3. Admin fills in product’s information 4. Admin clicks on add button 5. System adds product to database |
| Pre-conditions | Administrator signed in with admin account |
| Post-conditions | Admin adds new product to database |

## Use-case: Edit product

|  |  |
| --- | --- |
| Use case Name | Edit product |
| Brief description | This use case describes how administrator can edit product’s information |
| Actors | Administrator |
| Basic Flow | 1. At administrator page, admin clicks on search button 2. Admin enters products name 3. System displays product’s information 4. Admin clicks on a field 5. Admin changes its content 6. Admin clicks OK 7. System changes product’s content in dadabase |
| Alternative Flows | **Alternative flow 1: Product don’t exist in database**   1. From step #2 of basic flow, admin enters another term 2. Continue to step #6. |
| Pre-conditions | Administrator signed in with admin account |
| Post-conditions | Admin edits product’s information in database |

## Use-case: Delete product

|  |  |
| --- | --- |
| Use case Name | Delete product |
| Brief description | This use case describes how administrator can delete a product |
| Actors | Administrator |
| Basic Flow | 1. At administrator page, admin clicks on search button 2. Admin enters products name 3. System displays product’s information 4. Admin clicks on delete button 5. Admin confirms 6. Systems delete product in database |
| Alternative Flows | **Alternative flow 1: Product don’t exist in database**   1. From step #2 of basic flow, admin enters another term 2. Continue to step #6. |
| Pre-conditions | Administrator signed in with admin account |
| Post-conditions | Admin deletes a product in database |