**HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY**

**SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY**

**----------------------------**

****

**--------------------------------------------------------**

**PROJECT REPORT**

***Subject:* Design Analysis of Ecommerce website**

**Instructor:**

**Course:   
Class code:**

**Hanoi, 2024**

# Table of contents

[**Table of contents 1**](#_hlf058b6o0tk)

[**I. Overview 3**](#_i9gtauiuol5i)

[**II. Requirements 3**](#_ta9us4ex5srj)

[1. Functional requirements 3](#_h43kcut6ryqu)

[2. Non functional requirements 4](#_4ouki6mgbg37)

[3. Additional considerations 4](#_k48yb0ve2xef)

[**III. Functional modelling 5**](#_mfrafh3jn4m6)

[General Use case diagram 5](#_19hlrdf4xrbs)

[1. Account management 6](#_m5nm99g60omr)

[a, Use case diagram 6](#_tn93l4l51vp)

[b, Use case description 7](#_v4rezm23byzp)

[c, Activity diagram 13](#_stgqoh5put9)

[2. Shopping - Product search & view 17](#_nejarl7btj2z)

[a, Use case diagram 17](#_sfo5pixo7yif)

[b, Use case description 18](#_rqafhyvc8lbj)

[c, Activity diagram 19](#_s6acta4nk6zv)

[3, Shopping - Cart & Payment 20](#_c2tjkcqbq04h)

[a, Use case diagram 20](#_203iut24m86u)

[b, Use case description 21](#_s975gom4c5v9)

[c, Activity diagram 26](#_gtv9jktk1bwz)

[4. Product management for Admin 29](#_mudzj74dp0vw)

[a, Use case diagram 29](#_llkdc5q3b6k6)

[b, Activity diagram 29](#_cs7o7s1krk23)

[c, Use case description 30](#_r57607wmvgta)

[5. Product management for third party seller 33](#_ji4l4mpceipu)

[a, Use case diagram 33](#_1697z8gwn19t)

[b, Use case description 34](#_nxfd13avpoq4)

[c, Activity diagram 38](#_lr6oqbr2vsqr)

[6. Offer management for seller 40](#_2teebzvshxvq)

[a, Use case diagram 40](#_cnn8s75g0ji7)

[b, Use case description 41](#_1e75gau1gu4f)

[c, Activity diagram 42](#_kudg76kqwsfa)

[7. Customer Services 49](#_c8pd4xmmez4l)

[a, Use case diagram 49](#_ir7a9e1zo0jd)

[b, Activity diagram 49](#_m1ro6jxjdu0n)

[c, Use case description 50](#_m2u33274sq2h)

[**IV. Structural & Behavioral modelling 51**](#_7fqxe0joav7v)

[1. Use case “Create account/Sign up” 51](#_96o8g9sxsn5)

[a, Robustness diagram 51](#_volhhs5po0q)

[b, Sequence diagram 51](#_a2f3b5kfzg88)

[c, Communication diagram 52](#_kcsdb9hiry9v)

[d, State machine diagram - customer account 52](#_1bz93d9em2h6)

[2. Use case “Place Order” 56](#_trkqj0bjqo5q)

[a, Robustness diagram 56](#_hj95e2rjjs0f)

[b, Sequence diagram 57](#_h3kv73ud6oeb)

[c, Communication diagram 58](#_nzkwm9aiih5l)

[d, State machine diagram 59](#_a6rmzbt8pmae)

[3. Use case “Customer Service” 60](#_vrcf8as98r4o)

[a, Robustness diagram 60](#_8bgn5ko3csc2)

[b, Sequence diagram 61](#_hspgze70woj8)

[c, Communication diagram 62](#_ilvjtku6cqpu)

[d, State machine diagram 62](#_z5srym85yjw5)

# 

# I. Overview

# II. Requirements

## 1. Functional requirements

1.1. User accounts and preferences:

a, Users must be able to create accounts to save their personal information and purchase history.

b, Users must be able to log into their account at any time.

c, Users should be able to delete their account with confirmation.

d, Users should be able to recover their password using their registered email address.

e, Users should be able to create and manage wish lists for future purchases.

1.2. Product Browsing and Search

a, Users must be able to browse products by category, brand, price range, and other relevant filters.

b, Users must be able to search for products by keyword, product name, or product code.

c, Product detail pages should display high-quality images, descriptions, specifications, and availability information.

1.3. Shopping Cart and Checkout

a, Users must be able to add and remove items from their shopping cart.

b, Users must be able to edit quantities and apply coupons or promotions.

c, Checkout should be a simple and secure process with multiple payment options.

d, Order confirmation and tracking information should be readily available

1.4. Order Management:

a, Users should be able to track the status of their orders and view past orders.

b, Users should be able to cancel or modify orders before they are shipped.

c, Users should be able to easily initiate returns or exchanges.

d, Admins should have privilege to confirm changes, while being able to do everything a normal user can do.

1.5. Content Management:

a, Content editors should be able to easily add, edit, and publish blogs, product information, images, and descriptions.

b, Content editors should be able to manage promotions, coupons, and other marketing materials.

1.6. Third party sellers

a, Third party sellers can upload their product to be a part of website’s catalogues.

b, Partners should be able to change their stocks.

1.7. Customer’s support

a, Customers have the rights to write tickets and contact hotline for support

b, Customers should be able to see related information about website e.g. stores, contact addresses

c, The support officer should be able to see the responses by customers.

## 2. Nonfunctional requirements

2.1. Performance

a, The website should be able to handle high traffic volumes with minimal response times.

b, Page loading times should be optimized for a smooth user experience.

c, The website should be available 24/7 with minimal downtime (<0.1%)

2.2. Security

a, All user data, including payment information, should be encrypted and protected against unauthorized access.

b, Secure online payment gateways should be used for all transactions.

c, Regular security audits and penetration testing should be conducted.

2.3. Scalability

a,The website should be able to accommodate future growth in terms of users, products, and orders.

b,The system architecture should be flexible and modular to allow for easy expansion and adaptation.

2.4. Accessibility

The website should be able to support multiple languages and currencies to cater to a global audience.

## 3. Additional considerations

3.1. Integration with JYSK's existing store inventory and sales systems.

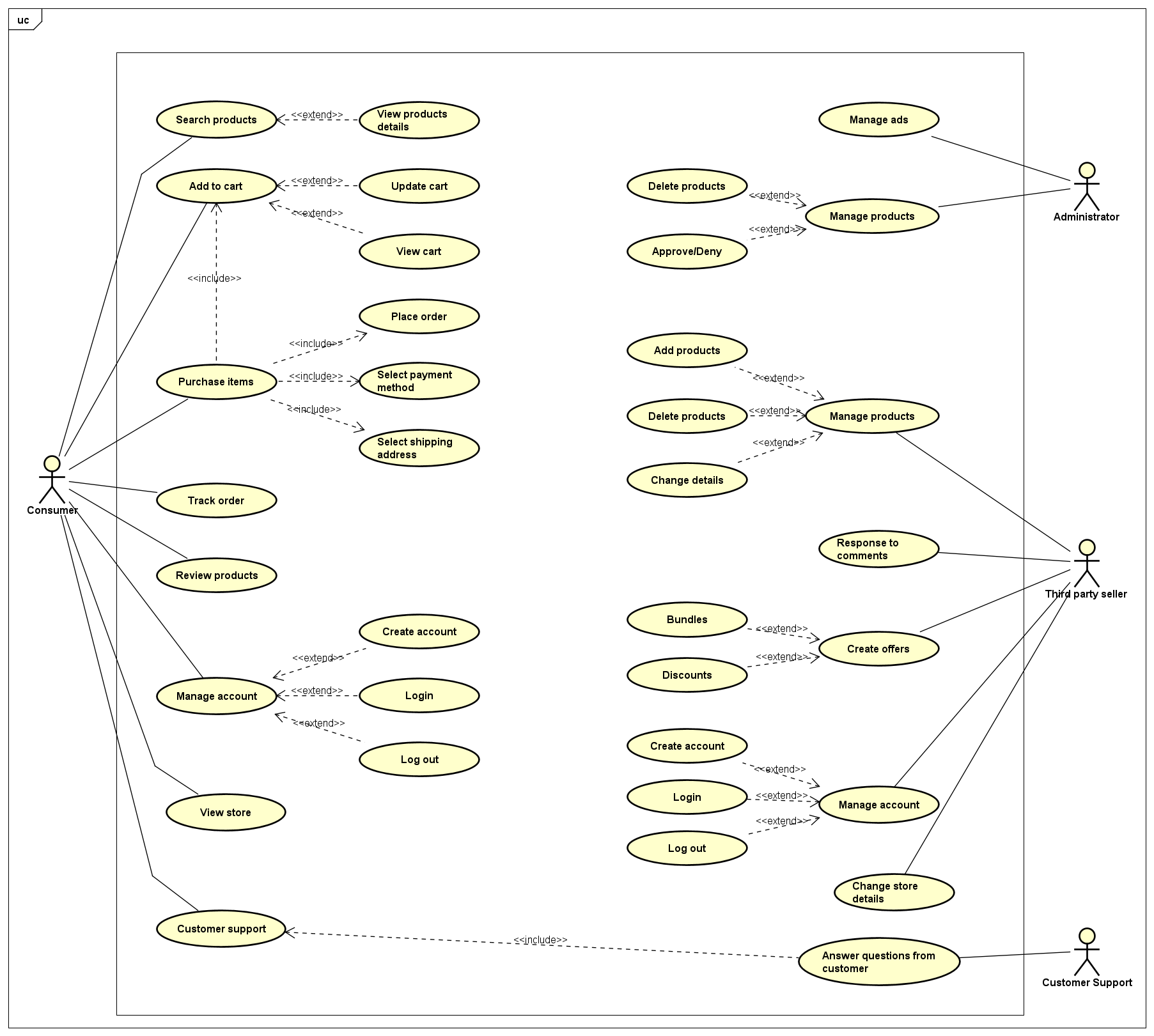
3.2. Implementation of a customer relationship management (CRM) system to track customer interactions and preferences.

3.3. Development of a comprehensive data analytics platform to gain insights into customer behavior and website performance

## 

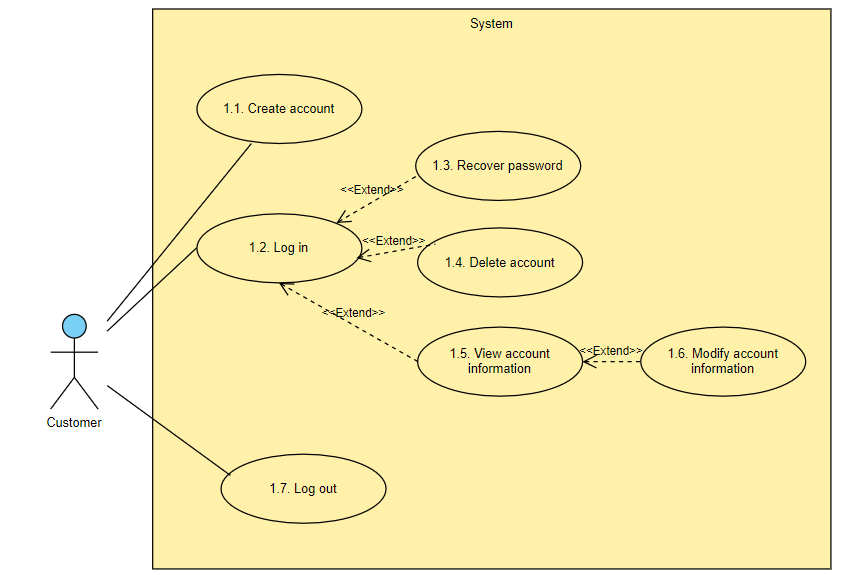
# III. Functional modelling

## General Use case diagram



## 1. Account management (Customer & Seller)

### a, Use case diagram

****

**Fig 2.1. Account management group use case diagram**

### b, Use case description

|  |  |
| --- | --- |
| ID | UC 1.1 |
| Use case name | Create account / sign up |
| Priority | Must have |
| Primary actor | Customer |
| Stakeholders & Interests | Customer - wants to create an account which they can use to buy products and get news and promotions  Website - wants to get information of customers in order to enhance experience and ensure a low rate of return purchases |
| Description | Customers create a new account on the website |
| Pre-conditions | Customer has an Internet connection and an email address |
| Trigger | Customer selects sign up / register |
| Post-conditions | Switch to login screen  Account registered in the system |
| Relationships |  |
| Normal flow | 1. The customer selects Log In and then Create Account  2. The customer enters basic information: Name, Email, Password  3. The customer selects “I accept Terms and Conditions” and click on Create Account  4. The system verifies if the entered email address is valid and send a verification email  5. The customer verifies the account via email  6. The account is registered in the system, the customer can now log into their account |
| Alternate/  Exceptional flow | 2a. The customer selects to sign up for newsletter  2a.1. The customer reads a newsletter preview  *Continues at step 3*  2b. The customer selects to read Terms and Conditions  *Continues at step 3*  4c. The customer entered an used/invalid email address  *Stall at step 4 until the email address is valid*  5d. The verification email is timed out  *Stop use case* |

|  |  |
| --- | --- |
| ID | UC 1.2 |
| Use case name | Log in |
| Priority | Must have |
| Primary actor | Customer |
| Stakeholders & Interests | Customer - wants to log into the account they created, where they store their information that makes their shopping experience convenient |
| Description | Customers log into their account |
| Pre-conditions | Customer must have an account |
| Trigger | Customer selects log in |
| Post-conditions | Switch to account screen  Log in action recorded in the system |
| Relationships |  |
| Normal flow | 1. The customer selects Log In  2. The customer enters registered email address and password  3. The system verifies if the entered email address and password matches the database entry  4. The system allows the user in, record a log in action in the system  5. The “My account” screen appears to the customer |
| Alternate/  Exceptional flow | 1a. The customer selects to remember their entry  *Step 2 will be done once, then skipped until their cache is cleared*  3b.1. The entered email address and password doesn’t match  *Repeat from step 2* |

|  |  |
| --- | --- |
| ID | UC 1.3 |
| Use case name | Recover Password |
| Priority | High priority |
| Primary actor | Customer |
| Stakeholders & Interests | Customer - recover their account with stored information and wishlist  Website - want to keep customer information updated and reduce storage usage for customer account |
| Description | Customers recover their password for accounts |
| Pre-conditions | Customer must have an account |
| Trigger | Customer selects forgot password button and input exact email or username |
| Post-conditions | Login the account with the new password  Switch to the main screen |
| Relationships | Extend: UC 1.2 |
| Normal flow | 1. User visits the website and click on “Forgot Password” link on the login page  2. The user is redirected to the “Forgot Password” page.  3. On the “Forgot Password” page, user is prompted to enter registered email address or username  4. The user enters their email address or username and submit the form  5. The website validates the entered information to ensure it matches a registered user  6. If the entered information is valid, the website generates a unique password reset link/token and sends it to the user’s registered email address  7. The user receives an email containing the password reset link/token.  8. The user clicks on the password reset link in the email.  9. The user is redirected to a "Reset Password" page on the website.  10. On the "Reset Password" page, the user is prompted to enter a new password and confirm it.  11. The user enters a new password and confirms it by re-entering it.  12. The website displays a success message indicating that the password has been reset.  13. The user can now use their new password to log in to the website. |
| Alternate/  Exceptional flow | 6a. The email entered is not valid  *Go back to step 5*  7b. The user cannot receive an email, they may ask for a resend  *Repeat step 7* |

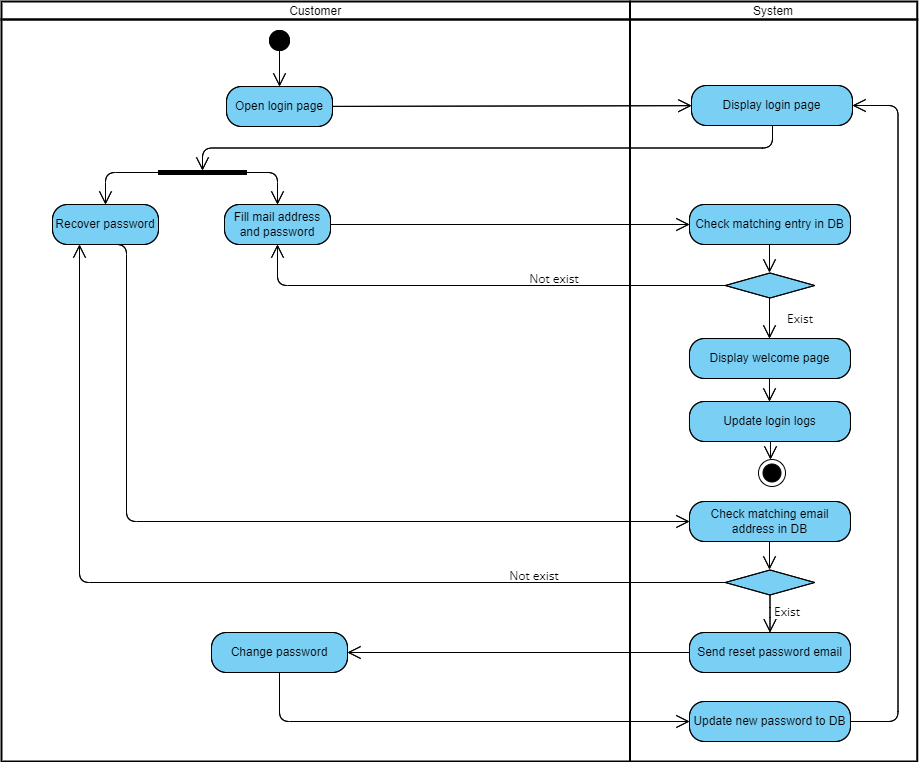
|  |  |
| --- | --- |
| ID | UC 1.4 |
| Use case name | Delete Account |
| Priority | Low priority |
| Primary actor | Customer |
| Stakeholders & Interests | Customer - wants delete an account that will not be used in the future  Website - delete the abundant account for saving memory in the database |
| Description | Customers delete their own accounts |
| Pre-conditions | Customer must have an account |
| Trigger | Customers choose to delete account |
| Post-conditions | The account has been deleted from the database  The user is navigated to the main website without login into any accounts |
| Relationships | Extend: UC 1.2 |
| Normal flow | 1. Users log in to their account on the website.  2. Once logged in, the user navigates to their account settings or profile page.  3. On the account settings or profile page, the user locates the "Delete Account" option.  4. The user clicks on the "Delete Account" option.  5. The website presents a confirmation dialog or a dedicated "Delete Account" page.  The confirmation dialog or page displays a message informing the user about the consequences of deleting their account, such as data loss and the inability to recover the account.  6. If the user decides to proceed with deleting their account, they confirm their intention by clicking on a "Delete Account" button or a similar confirmation action.  7. The system sends the customer a confirmation email to delete their account  8. The user receives an email containing the confirmation button (link).  9. The user clicks on the delete account link in the email.  10. During the account deletion process, the website may perform additional actions such as: Deleting the user's personal information and associated data from the database. Removing the user's account-related data from any connected services or platforms. Revoking any access tokens or authorization granted to the user's account.  11. Once the account deletion process is complete, the website displays a confirmation message indicating that the account has been successfully deleted.  12. The user is logged out of their account and redirected to the website's homepage. |
| Alternate/  Exceptional flow | 9a. The customer does nothing with the email  *Stop use case*  8b. The confirmation email expired  9b. The customer clicks on the button (link) after expiration  10b. The customer is navigated back to their profile page or log in page, with a dialog saying that the deletion link has expired.  *Stop use case* |

|  |  |
| --- | --- |
| ID | UC 1.5 and UC 1.6 |
| Use case name | View and modify account information |
| Priority | High priority |
| Primary actor | Customer |
| Stakeholders & Interests | Customer - view and update their account information  Website - want to keep a customer account updated |
| Description | Customers view and modify their account details |
| Pre-conditions | Customer must have an account |
| Trigger | Customer navigate to “My Profile” |
| Post-conditions | Customer can log into the account with the new information at any later time  All changes are updated in the system’s database |
| Relationships | Extend: UC 1.2 |
| Normal flow | 1. Users log in to their account on the website.  2. Once logged in, the user navigates to their account settings or profile page.  3. On the account settings or profile page, the user can view their current account information, such as name, email address, username, and profile picture, etc.  4. The user can choose to modify their account information by clicking on an "Edit" or "Modify" button or a similar action.  5. The website presents a form or a series of input fields where the user can make changes to their account information.  6. The user updates the relevant fields with the desired changes (refer to *subflows*)  6.1. Personal information  6.2. Password  6.3. Subscription status  7. The website may perform validation on the entered information to ensure it meets the required criteria (e.g., valid email format, username availability).  8. The website updates the user's account information in the database with the newly entered values.  9. The website displays a success message indicating that the account information has been successfully updated. |
| Subflows | S6.1. The customer updates relevant fields with their data  S6.2.1. The customer fills their old, new password and reconfirm the new password  S6.2.2. The system sends the customer an email to confirm if they wish to change their password  S6.2.3. The customer verifies the change by clicking the link attached to the email  S6.3.1. The customer select/unselect the checkbox to change their subscription status  S6.3.2. If the customer subscribes for the first time, the system sends them a coupon via email |
| Alternate/  Exceptional flow | 7a. Validation fails  *Repeat step 6 in required fields*  Other exceptional flows: Refer to *UC 1.6* |

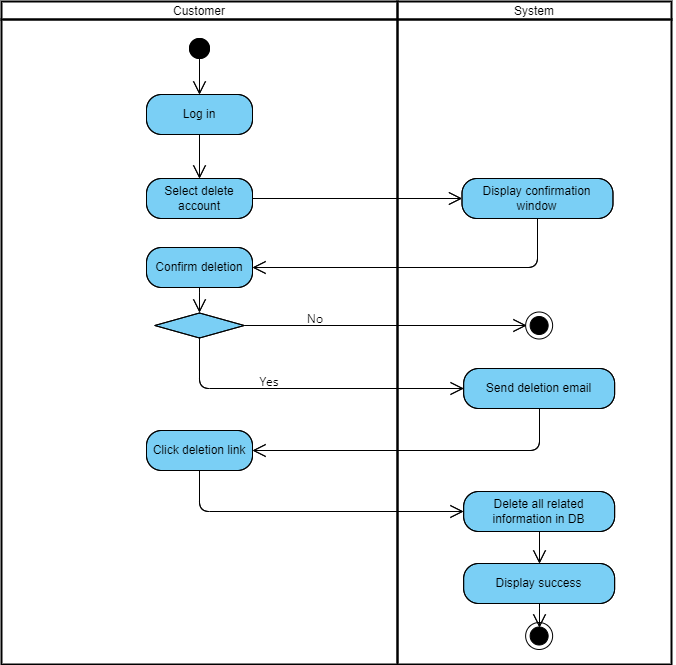
|  |  |
| --- | --- |
| ID | UC 1.7 |
| Use case name | Log out |
| Priority | Must have |
| Primary actor | Customer |
| Stakeholders & Interests | Customer - keep their account secure  Website - want to keep the number of customer logged in minimised due to server quota |
| Description | Customers log out from the website |
| Pre-conditions | Customer must have an account logged in |
| Trigger | Customer selects log out button |
| Post-conditions | Logout of the account  Switch to the main screen  Customer appear as an unknown web user |
| Relationships |  |
| Normal flow | 1. Users are currently logged in to their account on the website.  2. The user navigates to a designated "Logout" page or clicks on a "Logout" button or link.  3. The website presents a confirmation dialog or page to confirm the user's intention to log out.  4.The confirmation dialog or page displays a message informing the user about the consequences of logging out, such as ending their session and requiring them to log in again to access their account.  5. The user decides to proceed with logging out, they confirm their intention by clicking on a "Logout" button or a similar confirmation action.  6. The website performs the following actions:  Invalidates the user's session, terminating their current login session.  Clears any session-related data or tokens associated with the user's account.  Updates the user interface to reflect the logged-out state.  Displays a confirmation message indicating that the user has been successfully logged out.  7. The user is redirected to the login screen |
| Alternate/  Exceptional flow | a. The account have no new actions in a timeframe  *Automatically complete the use case*  5b. The user declines to log out  *Stop use case* |

### c, Activity diagram

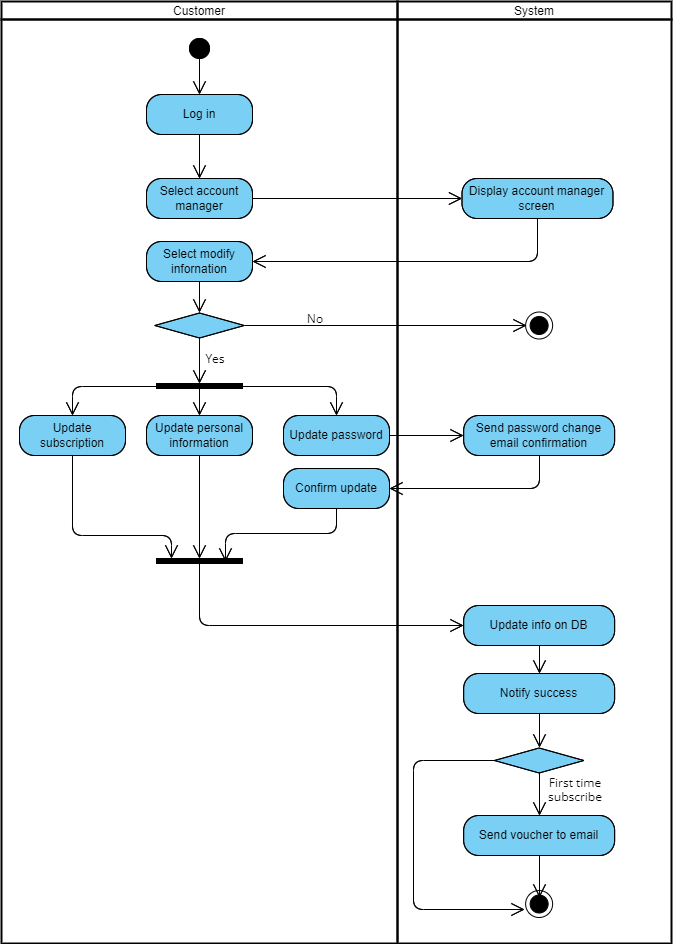
### 

AD 1.1. Signup/Create account****

AD 1.3. Recover password

****

AD 1.4. Delete account

****

AD 1.5 1.6. View & update account information

## 2. Shopping - Product search & view

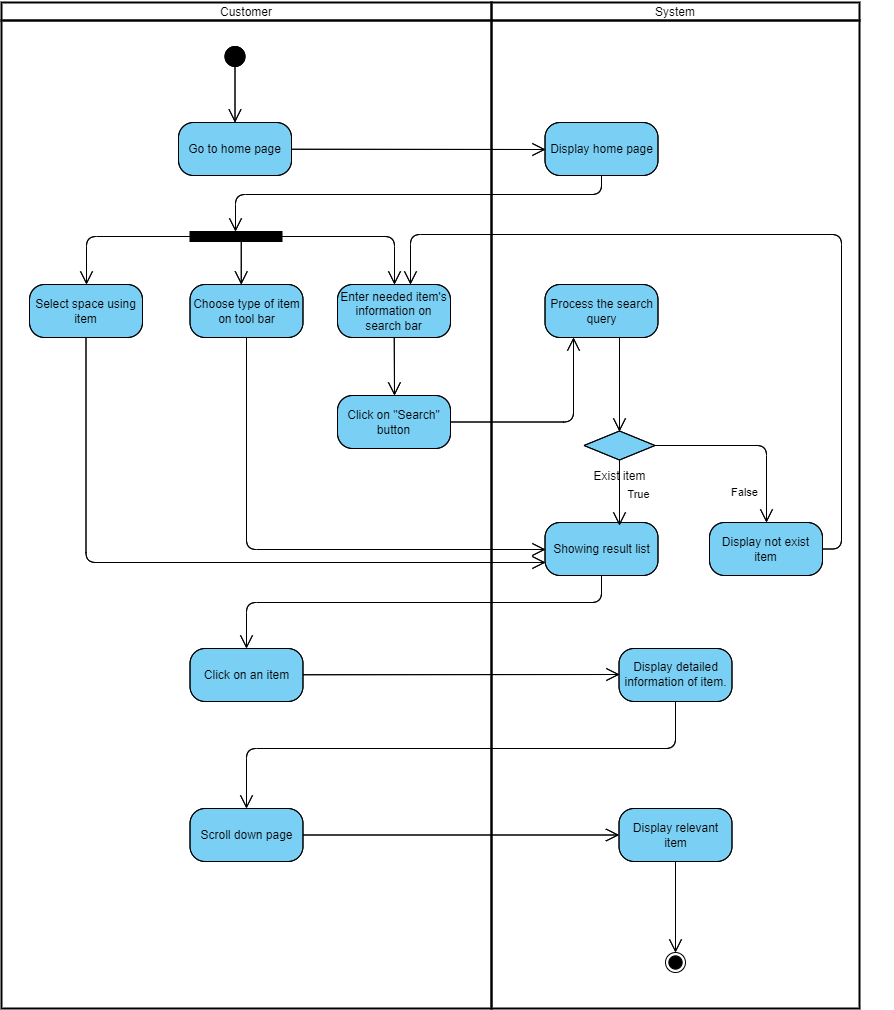
### a, Use case diagram

### 

### b, Use case description

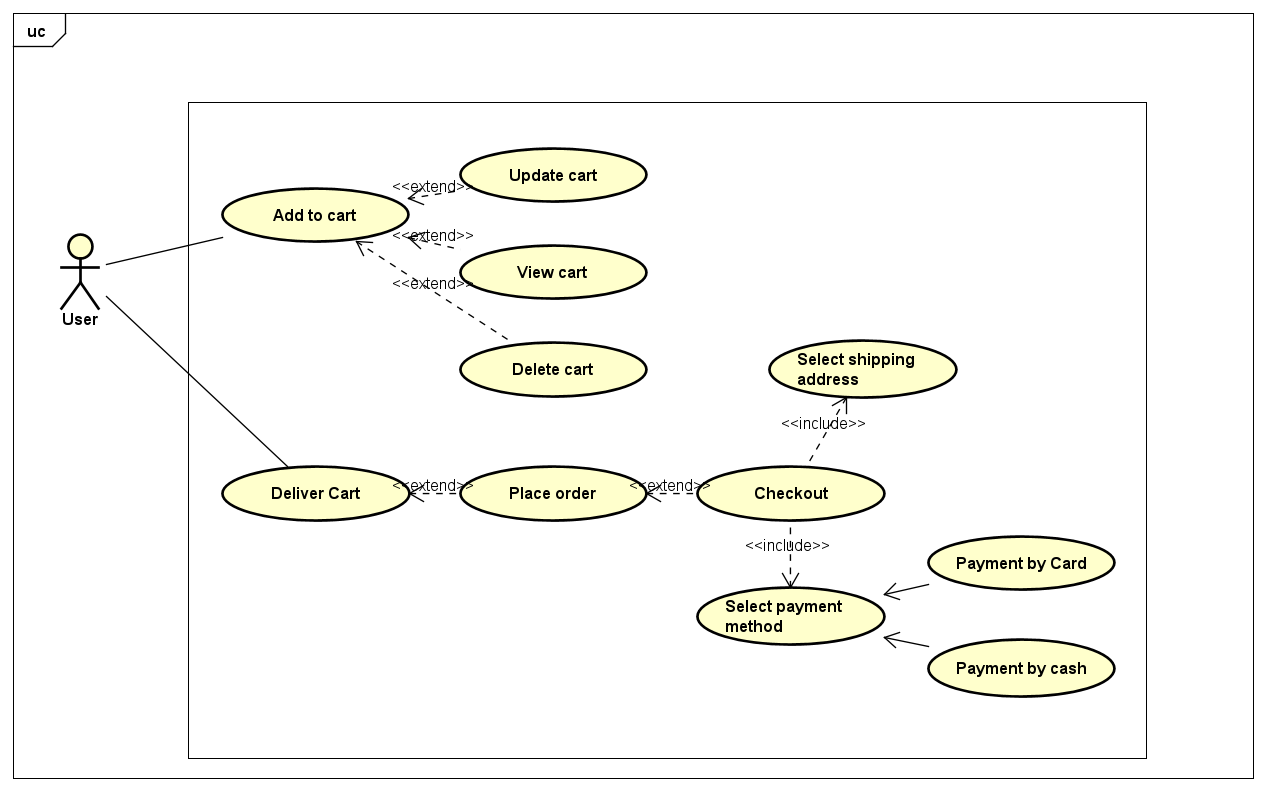
|  |  |
| --- | --- |
| ID | UC 2.1 |
| Use case name | Search products and view its detail |
| Priority | Must have |
| Primary actor | Customer |
| Stakeholders & Interests | Customer - search for a product and view its information  Website - want to display the information to the user |
| Description | Customers search for a product and view information |
| Pre-conditions | Customer has an Internet connection |
| Trigger | Customer input search query in the box or choose an icon of a product |
| Post-conditions | The information of relevant products of chosen products are displayed. |
| Relationships |  |
| Normal flow | 1. User opens the website and lands on the homepage or a designated search page. 2. The user enters a search query in the search bar, specifying the product they are looking for or relevant keywords. 3. The user submits the search query by clicking on a "Search" button or pressing Enter. 4. The website processes the search query and retrieves matching products from the database or product catalogue. 5. The website displays a search results page, listing the products that match the search query. 6. The user can browse through the search results and view summary information for each product, such as product name, image, price, and average rating. 7. The user selects a specific product from the search results by clicking on its name, image, or a "View Details" button. 8. The website navigates to a product details page dedicated to the selected product. 9. The product details page displays comprehensive information about the selected product, including detailed descriptions, specifications, customer reviews, and additional product images. 10. The user can scroll through the product details page to view all the available information and get a better understanding of the product. 11. Optionally, the website may provide related or recommended products on the product details page to offer additional options to the user. 12. The user can choose to add the product to their cart, save it for later, or proceed with the purchase by clicking on relevant buttons or links provided on the product details page. 13. If the user decides to add the product to their cart or make a purchase, the website may prompt them to log in or create an account if they haven't already done so. 14. The user completes the desired actions, such as adding the product to the cart, saving it for later, or initiating the checkout process. 15. The user can navigate back to the search results page or continue exploring other products on the website. 16. Optionally, the user can refine their search query by using search filters or sorting options provided on the search results page to further narrow down the displayed products based on their preferences. |
| Alternate/  Exceptional flow |  |

### c, Activity diagram



## 3, Shopping - Cart & Payment

### a, Use case diagram



### 

### 

### 

### b, Use case description

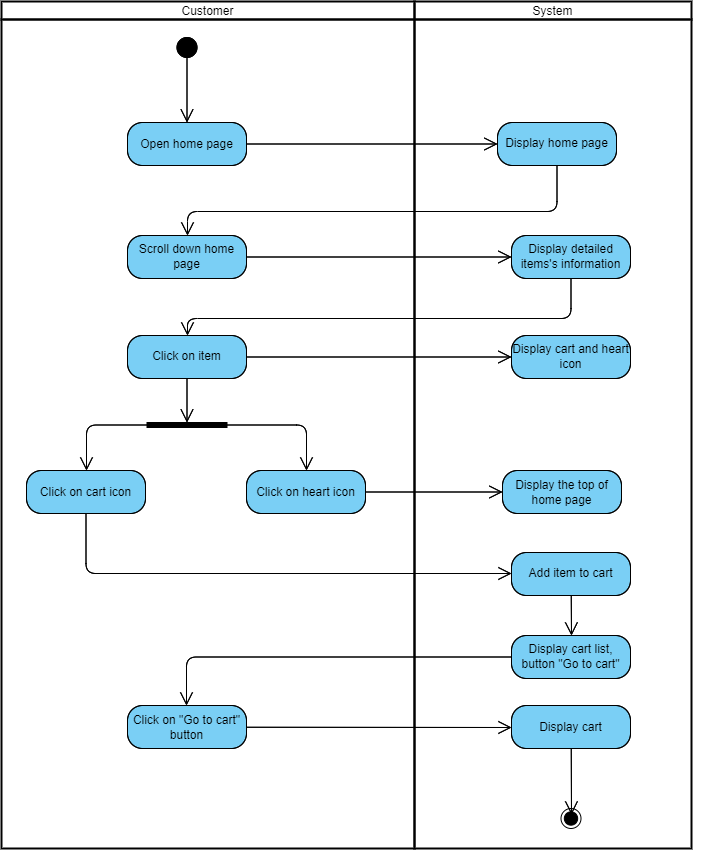
|  |  |
| --- | --- |
| ID | UC 3.1 |
| Use case name | Add item & View Cart |
| Priority | Must have |
| Primary actor | Customer |
| Stakeholders & Interests | Customer - view the information of a cart containing all products chosen  Website - want to display the information of a cart to the user |
| Description | Customers added some items into a cart and want to view its information |
| Pre-conditions | Customer has an Internet connection |
| Trigger | Customer chooses view cart button in the interface |
| Post-conditions | The information of the cart is displayed. |
| Relationships |  |
| Normal flow | 1. Users add products to their cart by clicking on an "Add to Cart" button or a similar action on the product details page. 2. The website updates the user's cart, adding the selected products and adjusting the quantities if necessary. 3. The user decides to view their cart by clicking on a "View Cart" or "Cart" button, typically located in the website's header or a prominent position. 4. The website navigates to the cart page, displaying an overview of the user's current cart contents. 5. The cart page shows a list of the products added to the cart, including their names, quantities, prices, and subtotal amounts. 6. The user can review the products in their cart, ensuring that the correct items and quantities are present. 7. The user has the option to modify the cart contents, such as updating the quantities, removing specific products, or applying coupon codes or discounts. 8. If the user wants to update the quantities, they can adjust the quantity field next to each product or use increment/decrement buttons. 9. If the user decides to remove a product from the cart, they can click on a "Remove" or "Delete" button associated with that product. 10. The website dynamically recalculates the cart totals based on any changes made by the user (e.g., quantity updates, product removals). 11. The cart page displays the updated cart totals, including the subtotal, any applied discounts, taxes, shipping costs, and the final grand total. 12. The user can review the cart totals to ensure accuracy and transparency in pricing. 13. The user has the option to proceed to checkout by clicking on a "Checkout" or "Proceed to Checkout" button if they are ready to make a purchase. 14. If the user wants to continue shopping, they can click on a "Continue Shopping" button or navigate back to the website's product catalogue or search pages. 15. The user can navigate back to the cart page at any time by clicking on the "View Cart" or "Cart" button in the website's header. 16. Optionally, the website may provide additional features on the cart page, such as the ability to save the cart for later, share the cart with others, or apply additional promotions or discounts. |
| Alternate/  Exceptional flow | 5a. Exceptional Flow: Insufficient Stock   * The website checks the stock availability for the products in the user's cart. * If any of the products in the cart have insufficient stock, the website displays an error message indicating the unavailable quantity. * The user has the option to update the quantities, remove the out-of-stock products, or proceed with the available items. * The website may provide suggestions for alternative products if available.   12a. Exceptional Flow: Cart Empty   * If the user has removed all products from the cart, the website displays a message indicating that the cart is empty. * The user may be prompted to continue shopping or return to the product catalogue to add items to the cart. |

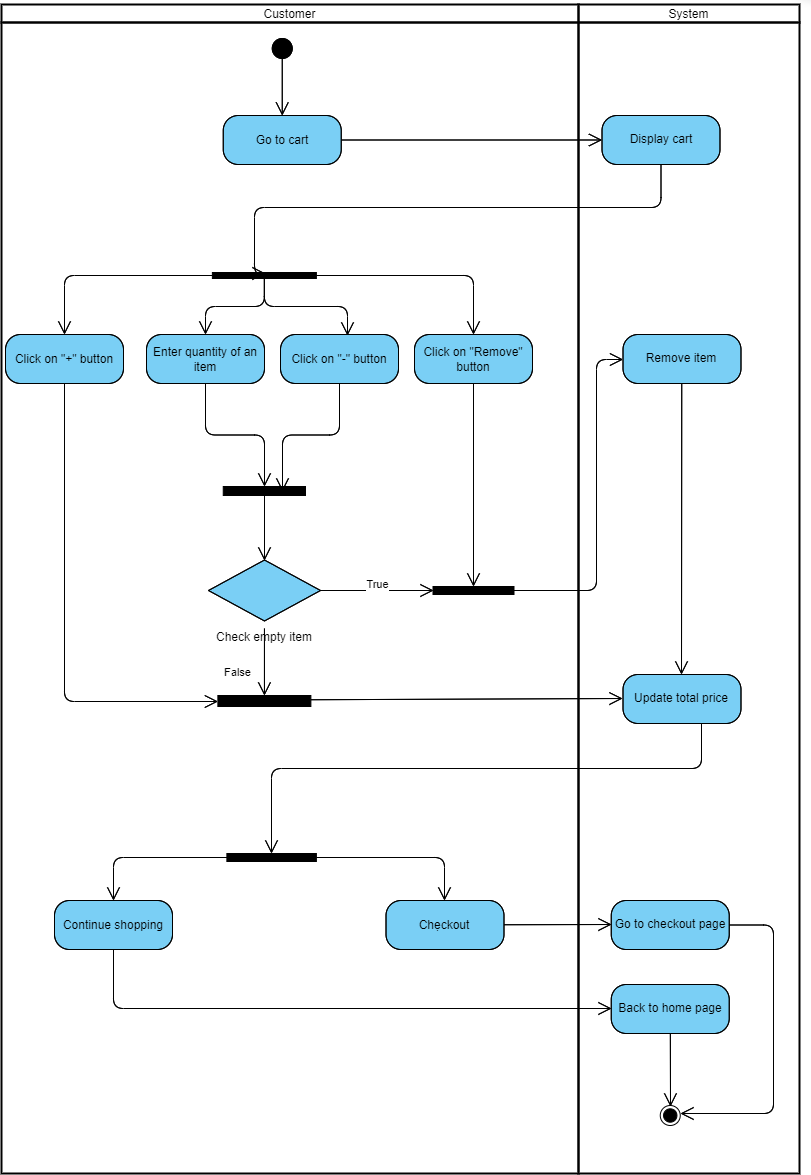
|  |  |
| --- | --- |
| ID | UC 3.2 |
| Use case name | Update Cart |
| Priority | Must have |
| Primary actor | Customer |
| Stakeholders & Interests | Customer - update the information of a cart containing all products chosen  Website - want to display the information of a cart to the user |
| Description | Customers update the information of a cart |
| Pre-conditions | Customer has an Internet connection |
| Trigger | Customer chooses some buttons for modifying cart in the interface |
| Post-conditions | The information of the cart is displayed. |
| Relationships |  |
| Normal flow | 1. User adds products to their cart by clicking on an "Add to Cart" button or a similar action on the product details page. 2. The website updates the user's cart, adding the selected products and adjusting the quantities if necessary. 3. The user decides to view their cart by clicking on a "View Cart" or "Cart" button, typically located in the website's header or a prominent position. 4. The website navigates to the cart page, displaying an overview of the user's current cart contents. 5. The cart page shows a list of the products added to the cart, including their names, quantities, prices, and subtotal amounts. 6. The user can review the products in their cart, ensuring that the correct items and quantities are present. 7. The user has the option to update the cart contents by modifying the quantities of specific products. 8. The user adjusts the quantity field next to each product or uses increment/decrement buttons to update the quantities. 9. The website dynamically recalculates the cart totals based on the updated quantities. 10. The cart page displays the updated cart totals, including the subtotal, any applied discounts, taxes, shipping costs, and the final grand total. 11. The user can review the updated cart totals to ensure accuracy and transparency in pricing. 12. The user has the option to continue modifying the cart by updating quantities, removing products, or applying coupon codes or discounts if necessary. 13. If the user wants to remove a product from the cart, they can click on a "Remove" or "Delete" button associated with that product. 14. The website dynamically recalculates the cart totals after any modifications made by the user. 15. The user can review the updated cart totals and continue making further adjustments if needed. 16. Optionally, the website may provide additional features on the cart page, such as the ability to save the cart for later, share the cart with others, or apply additional promotions or discounts. 17. The user can choose to proceed to checkout by clicking on a "Checkout" or "Proceed to Checkout" button if they are ready to make a purchase. 18. If the user wants to continue shopping, they can click on a "Continue Shopping" button or navigate back to the website's product catalog or search pages. 19. The user can navigate back to the cart page at any time by clicking on the "View Cart" or "Cart" button in the website's header. |
| Alternate/  Exceptional flow | Exceptional Flow: Insufficient Stock   * If the product's available quantity is lower than the requested quantity, the system displays an error message indicating the insufficient stock. * The customer has the option to update the quantity to a lower value or remove the product from their cart.   12a. Exceptional Flow: Product Unavailable   * The user adjusts the quantity of a specific product in their cart. * The website checks the availability of that product. * If the product becomes unavailable (e.g., out of stock, discontinued), the website displays an error message indicating that the product is no longer available. * The user has the option to remove the unavailable product from the cart or explore alternative options. |

|  |  |
| --- | --- |
| ID | UC 3.3 and UC 3.4 |
| Use case name | Deliver a Cart and Place Order |
| Priority | Must have |
| Primary actor | Customer |
| Stakeholders & Interests | Customer - place order.  Website - want to display the information of items that the user wants to buy on the screen. |
| Description | Customers want to buy items and place orders. |
| Pre-conditions | Customers have an Internet connection and at least an item that they want to buy. |
| Trigger | Customer chooses ”Place order” button in the interface |
| Post-conditions | The information of the cart is displayed. |
| Relationships |  |
| Normal flow | 1. User adds products to their cart by clicking on an "Add to Cart" button or a similar action on the product details page. 2. The website updates the user's cart, adding the selected products and adjusting the quantities if necessary. 3. The user decides to view their cart by clicking on a "View Cart" or "Cart" button, typically located in the website's header or a prominent position. 4. The website navigates to the cart page, displaying an overview of the user's current cart contents. 5. The cart page shows a list of the products added to the cart, including their names, quantities, prices, and subtotal amounts. 6. The user can review the products in their cart, ensuring that the correct items and quantities are present. 7. The user has the option to update the cart contents by modifying the quantities of specific products. 8. The user adjusts the quantity field next to each product or uses increment/decrement buttons to update the quantities. 9. The website dynamically recalculates the cart totals based on the updated quantities. 10. The cart page displays the updated cart totals, including the subtotal, any applied discounts, taxes, shipping costs, and the final grand total. 11. The user can review the updated cart totals to ensure accuracy and transparency in pricing. 12. The user chooses “Progress Payment” to go to the checkout phase. |
| Alternate/  Exceptional flow | Exceptional Flow: Insufficient Stock   * If the product's available quantity is lower than the requested quantity, the system displays an error message indicating the insufficient stock. * The customer has the option to update the quantity to a lower value or remove the product from their cart.   12a. Exceptional Flow: Product Unavailable   * The user adjusts the quantity of a specific product in their cart. * The website checks the availability of that product. * If the product becomes unavailable (e.g., out of stock, discontinued), the website displays an error message indicating that the product is no longer available. * The user has the option to remove the unavailable product from the cart or explore alternative options. |

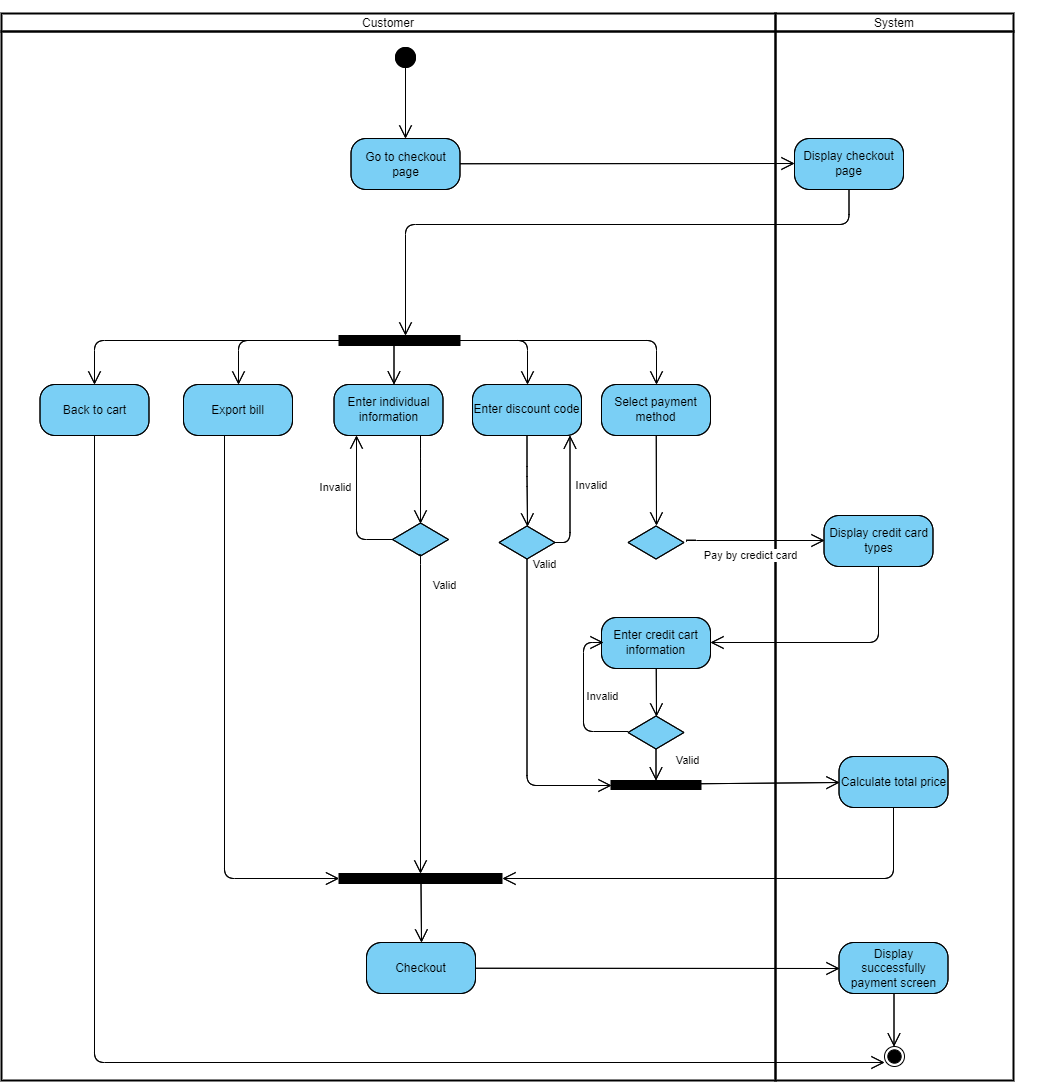
|  |  |
| --- | --- |
| ID | UC 3.5 |
| Use case name | Checkout |
| Priority | Must have |
| Primary actor | Customer |
| Stakeholders & Interests | Customer - view detailed pricing information for your customer's order.  Website - want to display the detailed pricing information of items that the user wants to buy on the screen. |
| Description | Customers want to see detailed pricing information about their order. |
| Pre-conditions | Customers have an Internet connection and have placed orders. |
| Trigger | Customer chooses ”Place order” button in the interface |
| Post-conditions | The information of the cart is displayed. |
| Relationships |  |
| Normal flow | 1. After the user chooses “Progress Payment”, the checkout screen has been displayed. 2. The screen will display again the order you have selected, the unit price of each item and additional costs incurred. 3. The user has to fill out a delivery form with essential information such as full name, phone, location, email and include an invoice request if desired. 4. The user has to choose the appropriate payment method. If the user chooses to pay by credit card then the user will have to fill in information related to your card. 5. The user can apply coupon code for discount if necessary in the discount/gift box. 6. The user can confirm the order by clicking on the “Progress Payment” button or clicking on the “Back” button to back the previous page. |
| Alternate/  Exceptional flow | 3a. Exceptional Flow: Individual information is incorrect   * The user enters individual information on the delivery form . * The website checks the correctness of that information. * If the information is incorrect, the website displays an error message indicating that the individual information is not correct. * The user has to fill out the delivery form again.   4a. Exceptional Flow: Information about credit card is incorrect.   * The user enters information about the credit card . * The website checks the correctness of that information. * If the information is incorrect, the website displays an error message indicating that the credit card’s information is not correct. * The user has to fill out the credit card’s information again or choose to pay by cash..   5a. Exceptional Flow: Coupon code or discount is incorrect.   * The user enters the coupon code or discount in the coupon code/gift box. * The website checks the correction of that coupon code. * If the coupon code is incorrect, the website displays an error message indicating that the coupon code is not existing or not correct. * The user has the option to remove the current coupon code from the box and not have a discount or enter another coupon code. |

### c, Activity diagram

****

AD 3.1. View cart****

AD 3.2. Update cart

****

AD 3.3-3.5.Place order, Checkout & Payment

## 4. Product management for Admin

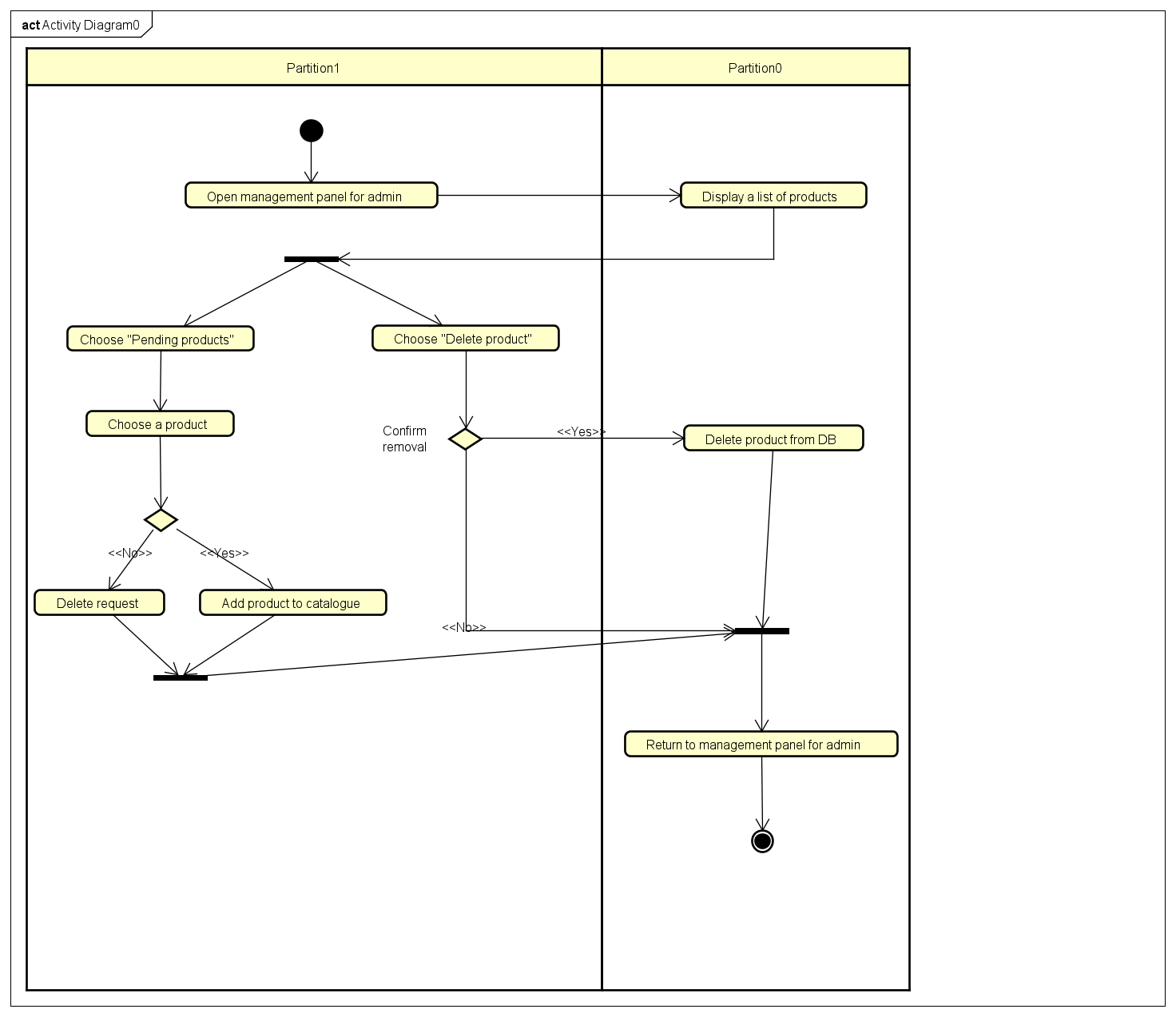
### a, Use case diagram

A diagram of a product

Description automatically generated

### 

### b, Activity diagram

****

### c, Use case description

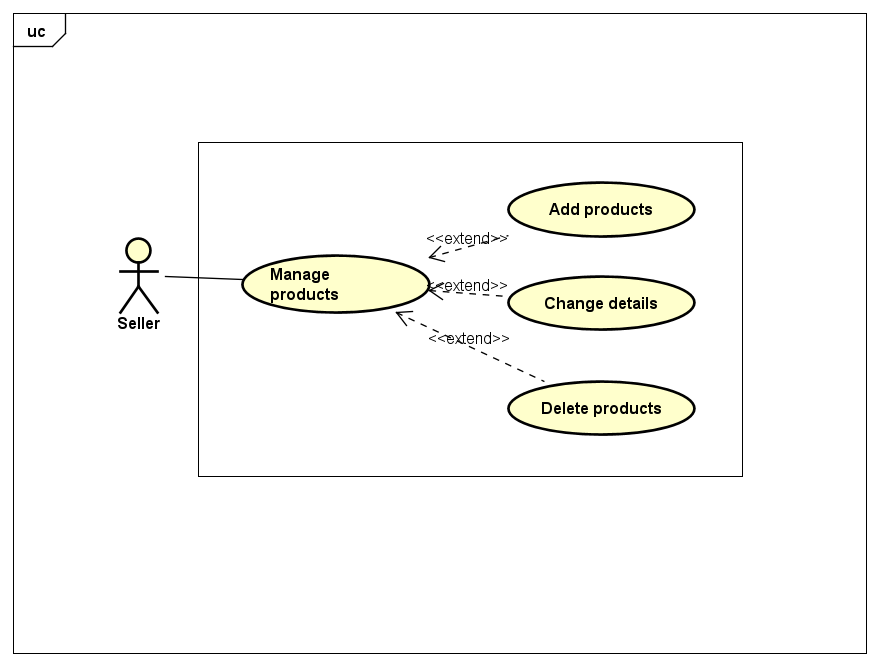
|  |  |
| --- | --- |
| ID | UC 4.1 |
| Use case name | Admin accept/deny product request |
| Priority | Must have |
| Primary actor | Admin |
| Stakeholders & Interests | Sellers - wants to add a product to the online market with detailed information about it.  Admin - wants to authenticate that the product has fitted requirements, after that accepts it to the online market. The user can search the product’s information. |
| Description | Admin adds/deny a product to the online market. |
| Pre-conditions | Actor is Admin and logged in |
| Trigger |  |
| Post-conditions | Switch to the product management screen.  Product is added into the system |
| Relationships |  |
| Normal flow | 1. The system authenticates the role of admin  2. Admin selects product management panel  3. Admin select a product request  4. Admin approves the request  5. The product is added into the system |
| Alternate/  Exceptional flow | 2a. The system authenticates the role of the actor.  2a.1. The actor has no privilege to add a product.  2a.2. The system rejects the actor's request.  *Stop use case*  3b. Admin deny the product request  3b.1. The system deletes the request |
| Non-functional requirement | NFR-3.1: Detailed information about the product includes name, price, description, origin, number of available items, … |

|  |  |
| --- | --- |
| ID | UC 4.2 |
| Use case name | Admin modifies product information |
| Priority | Must have |
| Primary actor | Admin |
| Stakeholders & Interests | Admin - wants to modify some information about the product.  JYSK - wants to authenticate that the Actor is truly the admin, after that it gets information about the product and modifies this data in the system. |
| Description | Admin modifies some detailed information of a product |
| Pre-conditions | Actor is Admin and logged in |
| Trigger |  |
| Post-conditions | Switch to the product management screen.  Product is updated in the system |
| Relationships |  |
| Normal flow | 1. Admin views a product and selects Modify Product  2. The system authenticates the role of admin  3. Admin updates detailed information about the product  4. The system retrieves the product’s detailed information  5. The product is updated in the system. |
| Alternate/  Exceptional flow | 2a. The system authenticates the role of the actor.  2a.1. The actor has no privilege to update the product.  2a.2. The system rejects the actor's request.  *Stop use case*  3b. Admin enters detailed information about the product  3b.1. Admin omits some required information  3b.2. The system warns and requests to refill the blank fields  *Go back to step 3* |
| Non-functional requirement | *Go to NFR-3.1* |

|  |  |
| --- | --- |
| ID | UC 4.2 |
| Use case name | Admin deletes product |
| Priority | Must have |
| Primary actor | Admin |
| Stakeholders & Interests | Admin - wants to delete the product.  Website - wants to authenticate that the Actor is truly the admin, then removes the product out of the system. |
| Description | Admin deletes a product |
| Pre-conditions | Actor is Admin and logged in |
| Trigger |  |
| Post-conditions | Switch to the product management screen.  Product is removed out of the system. |
| Relationships |  |
| Normal flow | 1. The system authenticates the role of admin.  2. Admin views a product and selects Delete Product.  3. The system accepts the request.  4. The product is removed out of the system. |
| Alternate/  Exceptional flow | 2a. The system authenticates the role of the actor.  2a.1. The actor has no privilege to delete the product.  2a.2. The system rejects the actor's request.  *Stop use case* |

## 5. Product management for third party seller

### a, Use case diagram



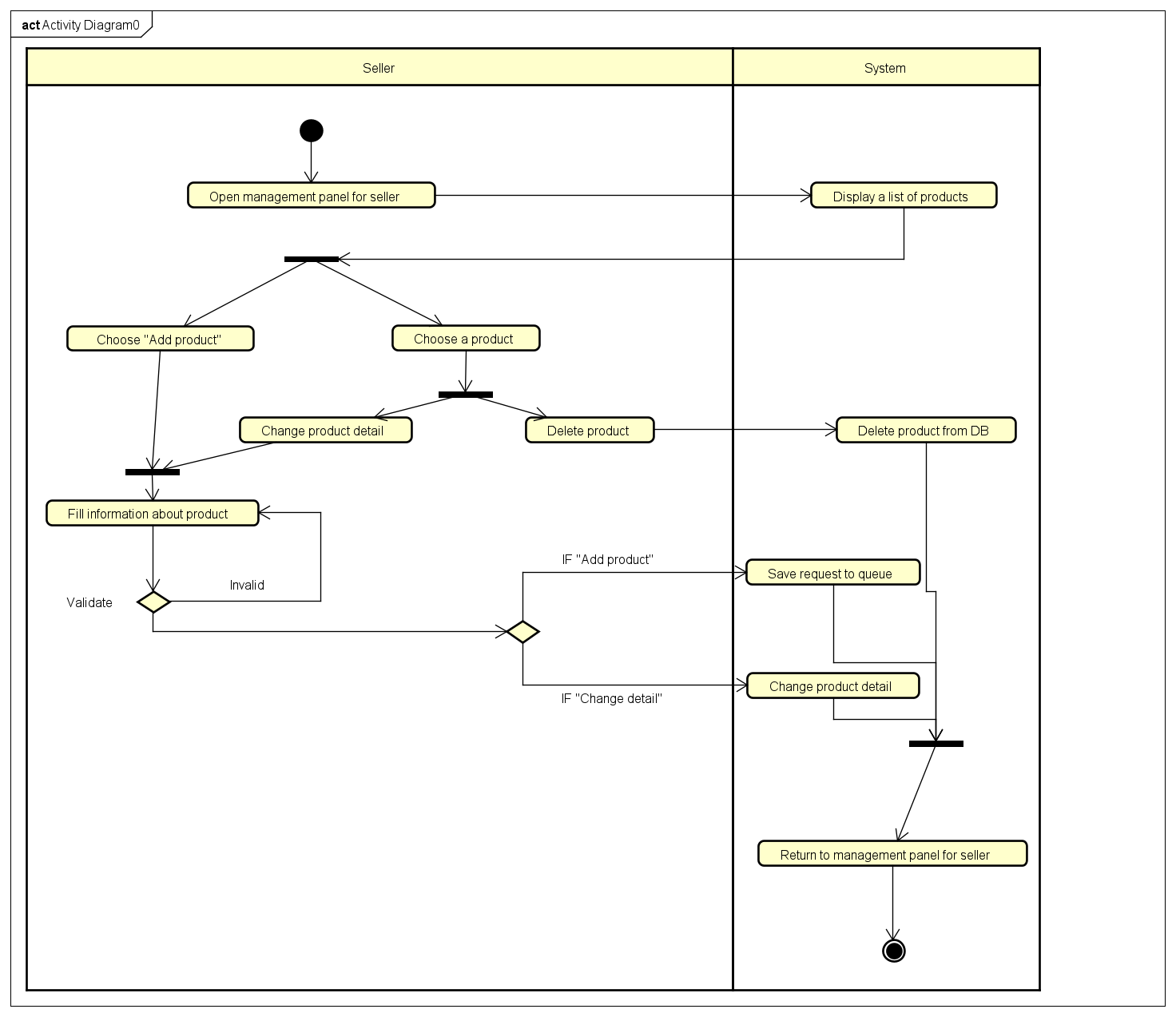
### b, Use case description

|  |  |
| --- | --- |
| ID | UC 5.1 |
| Use case name | Third party seller requests to add product |
| Priority | Must have |
| Primary actor | Third party seller |
| Stakeholders & Interests | Seller - requests to add a product.  Website - wants to authenticate the role of the seller, then adds the requests into the pending queue in the system. |
| Description | Third party seller requests to add product |
| Pre-conditions | Actor is logged in |
| Trigger |  |
| Post-conditions | Switch to the product management screen.  The request is added into the system. |
| Relationships | Request must be approved by the admin: UC 5.4. |
| Normal flow | 1. Third party seller selects Request to Add Product.  2. The system authenticates the role of the actor.  3. The seller enters detailed information about the product.  4. The system retrieves the product’s detailed information.  5. The request is added into the pending queue in the system. |
| Alternate/  Exceptional flow | 2a. The system authenticates the role of the actor.  2a.1. The actor has no privilege to request to add the product  2a.2. The system rejects the actor's request.  *Stop use case*  3b. Seller enters detailed information about the product.  3b.1. Seller omits some required information.  3b.2. The system warns and requests to refill the required fields.  *Go back to step 3* |
| Non-functional requirement | *Go to NFR-3.1* |

|  |  |
| --- | --- |
| ID | UC 5.2 |
| Use case name | Third party modify product information |
| Priority | Must have |
| Primary actor | Third party seller |
| Stakeholders & Interests | Seller - requests to modify some information about the product.  Website - wants to authenticate the role of the seller and ensures that the ownership of the product belongs to the seller |
| Description | Third party seller modify some detailed information about the product |
| Pre-conditions | Actor is logged in |
| Trigger |  |
| Post-conditions | Switch to the product management screen.  The request is added into the system. |
| Relationships | Request must be approved by the admin: UC 5.4. |
| Normal flow | 1. Third party seller selects Request to Add Product.  2. The system authenticates the role of the actor.  3. The system verifies that the ownership of the product belongs to the actor.  4. The seller enters detailed information about the product.  5. The system retrieves the product’s detailed information.  6. The product’s information is changed |
| Alternate/  Exceptional flow | 2a. The system authenticates the role of the actor.  2a.1. The actor has no privilege to request to modify the product.  2a.2. The system rejects the actor's request.  *Stop use case*  3b. The system verifies that the ownership of the product belongs to the actor  3b.1. The product does not belong to the seller  3b.2. The system rejects the actor's request.  *Stop use case*  4c. Seller enters detailed information about the product.  4c.1. Seller omits some required information.  4c.2. The system warns and requests to refill the required fields.  *Go back to step 3* |
| Non-functional requirement | *Go to NFR-3.1* |

|  |  |
| --- | --- |
| ID | UC 5.3 |
| Use case name | Third party seller deletes product |
| Priority | Must have |
| Primary actor | Third party seller |
| Stakeholders & Interests | Seller- wants to delete the product.  Website - wants to authenticate that the Actor is truly the admin, then removes the product out of the system. |
| Description | Third party seller deletes a product |
| Pre-conditions | Actor is Admin and logged in |
| Trigger |  |
| Post-conditions | Switch to the product management screen.  Product is removed out of the system. |
| Relationships |  |
| Normal flow | 1. Admin views a product and selects Delete Product.  2. The system authenticates the role of admin.  3. The system accepts the request.  4. The product is removed out of the system. |
| Alternate/  Exceptional flow | 2a. The system authenticates the role of the actor.  2a.1. The actor has no privilege to delete the product.  2a.2. The system rejects the actor's request.  *Stop use case*  3b. The system verifies that the ownership of the product belongs to the actor  3b.1. The product does not belong to the seller  3b.2. The system rejects the actor's request.  *Stop use case* |

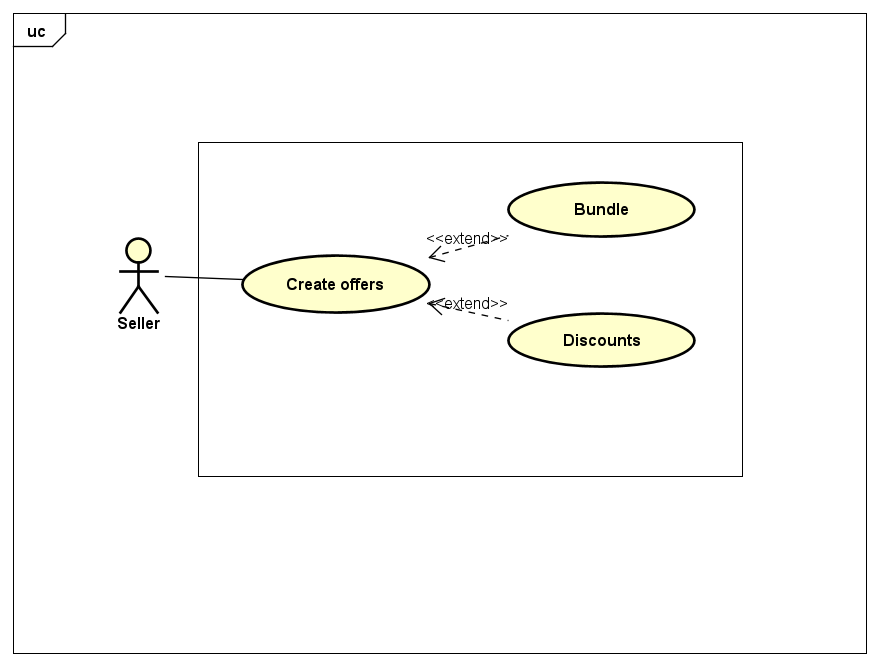
### c, Activity diagram

****

AD 5.1 - 5.3. Third party seller uses

## 6. Offer management for seller

### a, Use case diagram

****

### 

### b, Use case description

|  |  |
| --- | --- |
| ID | UC 6.1 |
| Use case name | Seller creates bundle |
| Priority | Must have |
| Primary actor | Seller |
| Stakeholders & Interests | Seller - creates bundle for many products  Website - updates the bundle |
| Description | Seller creates bundle for many products |
| Pre-conditions | Actor is seller and is logged in |
| Trigger |  |
| Post-conditions | Bundle is updated in the system |
| Relationships |  |
| Normal flow | 1. Seller identifies the specific products they want to create a bundle  2. Seller enter details for the bundle  3. Seller publish the offer  4. The system updated the bundle |
| Alternate/  Exceptional flow | 1. The system authenticates the role of the actor   * The actor has no privilege to make the offer * The system rejects the actor's request |

|  |  |
| --- | --- |
| ID | UC 6.2 |
| Use case name | Seller creates discounts |
| Priority | Must have |
| Primary actor | Seller |
| Stakeholders & Interests | Seller - creates discounts for customer  Website - updates the personalized offer for the corresponding customer |
| Description | Seller creates discounts for customer |
| Pre-conditions | Actor is seller and is logged in |
| Trigger |  |
| Post-conditions | Discounts is updated in the system |
| Relationships |  |
| Normal flow | 1. Seller identifies discount generic or for the specific product they want  2. Seller enter details for the discount  3. Seller enter the eligible conditions for the discount  4. Seller publish the offer  5. The system updated the offer for the customer |
| Alternate/  Exceptional flow | 1. The system authenticates the role of the actor   * The actor has no privilege to make the offer * The system rejects the actor's request   2. The system authenticates the role of the consumer   * The consumer is not eligible for the offer * The system rejects the usage of the offer |

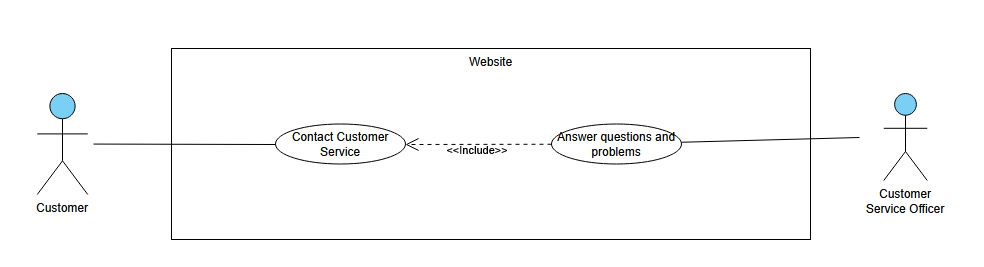
### c, Activity diagram

### 

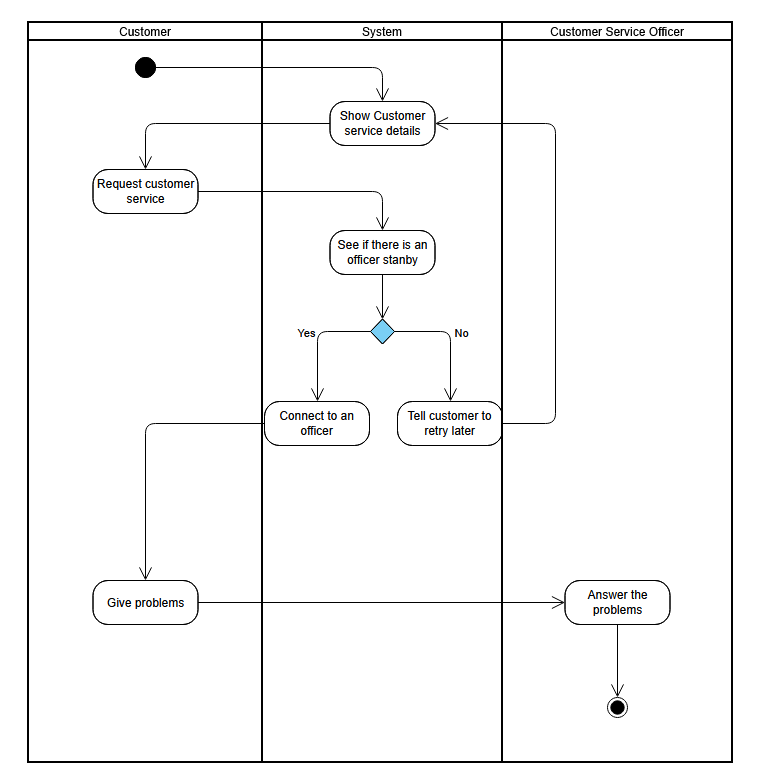
AD 6. Create offers

## 7. Customer Services

### a, Use case diagram

****

### b, Activity diagram

****

### c, Use case description

|  |  |
| --- | --- |
| ID | 9.1 |
| Use case name | Answer Customers’ questions |
| Priority | Should have |
| Primary actor | Customer service officer (CSO) & Customer |
| Stakeholders & Interests | CSO: Interested in providing assistance and resolving customer inquiries or issues promptly.  Customers: Interested in receiving timely and effective customer service to address their concerns or problems.  Website Administrator: Interested in maintaining a positive customer experience and addressing any potential issues or complaints. |
| Description | This use case describes the process of a customer service officer handling customer inquiries or issues in JYSK shopping system |
| Pre-conditions | The customer service officer must have access to the customer service tools or interface of the online shopping system. |
| Trigger | A customer contacts customer service with a question, concern, or issue related to their shopping experience. |
| Post-conditions | A new order is created and added to the system. The customer receives an order confirmation with a unique order number. |
| Relationships |  |
| Normal flow | 1. The customer service officer receives a customer inquiry or issue through a communication channel (e.g., phone, email, live chat). 2. The officer verifies the customer's identity and gathers relevant information about the inquiry or issue. 3. The officer listens attentively and empathetically to the customer's concerns, providing a supportive and professional approach. 4. The officer accesses the online shopping system's resources, such as the customer's order history, product information, or policies, to address the inquiry or issue effectively. 5. The officer provides accurate and comprehensive information, answers questions, and resolves the customer's concern to the best of their ability. 6. If necessary, the officer may need to consult with other departments or escalate the issue to higher-level support to ensure a satisfactory resolution. 7. The officer documents the customer interaction, including the details of the inquiry or issue, actions taken, and any resolutions provided. 8. The officer follows up with the customer if additional steps or information is required, ensuring open lines of communication until the issue is fully resolved. |
| Alternate/  Exceptional flow | 1. Exceptional Flow before main flow: Technical Difficulties  * If the online shopping system experiences technical difficulties or outages, the officer informs the customer about the situation and provides alternative solutions * The officer may need to apologize for the inconvenience caused and assure the customer that the issue will be resolved as soon as possible.  1. Exceptional Flow after flow : Escalation or Complaint  * If the customer's inquiry or issue cannot be resolved immediately or requires further investigation, the officer assures the customer that their concern will be escalated to the appropriate department or team. * The officer provides the customer with a clear timeline for when they can expect a resolution or follow-up communication. * If the customer expresses dissatisfaction or raises a complaint, the officer ensures that the complaint is properly recorded and initiates appropriate measures to address the complaint promptly. |

# IV. Structural & Behavioral modelling

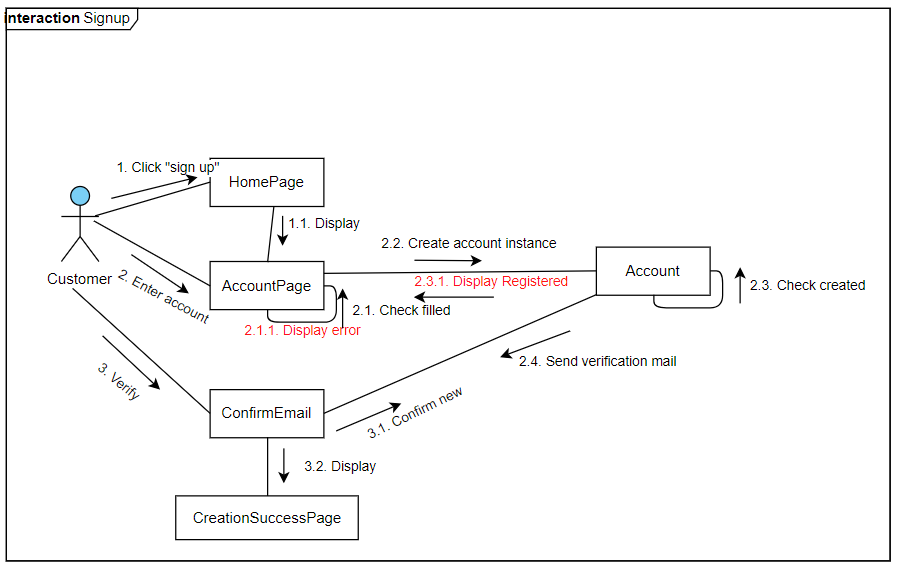
## 1. Use case “Create account/Sign up”

### a, Robustness diagram

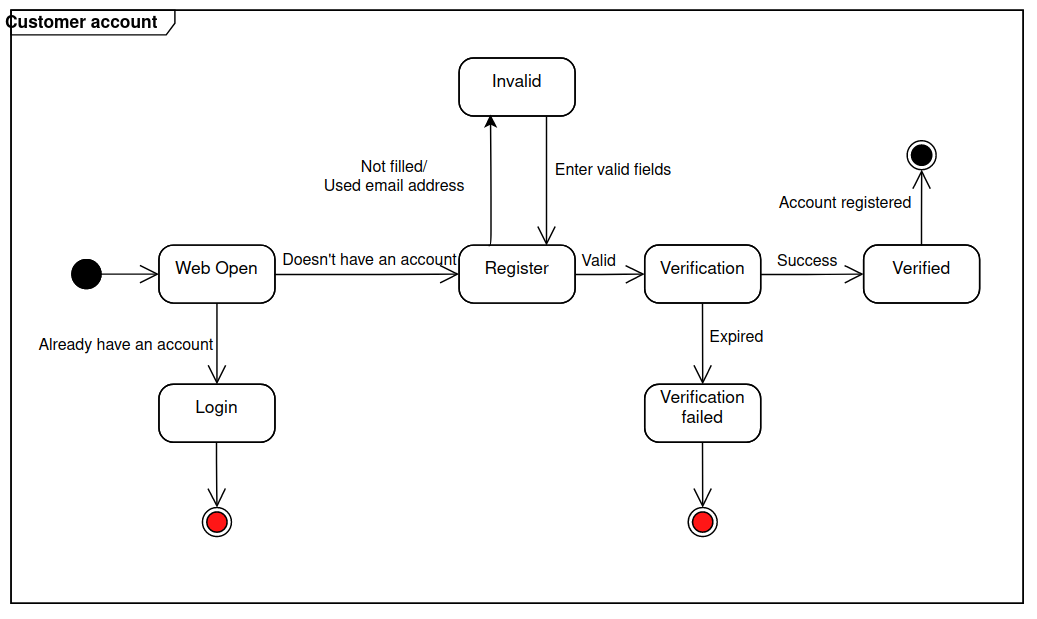
### 

### b, Sequence diagram

### c, Communication diagram

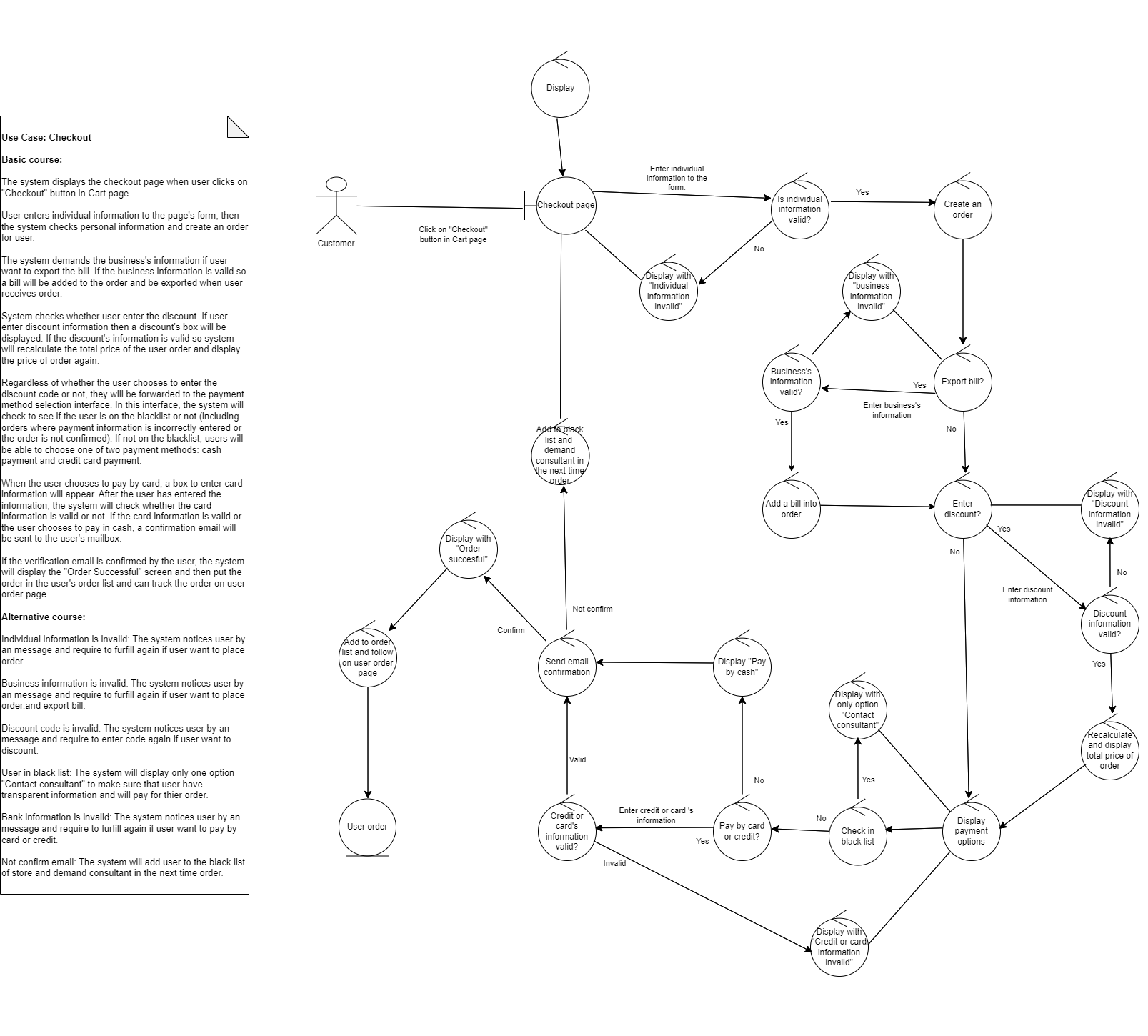


### d, State machine diagram - customer account

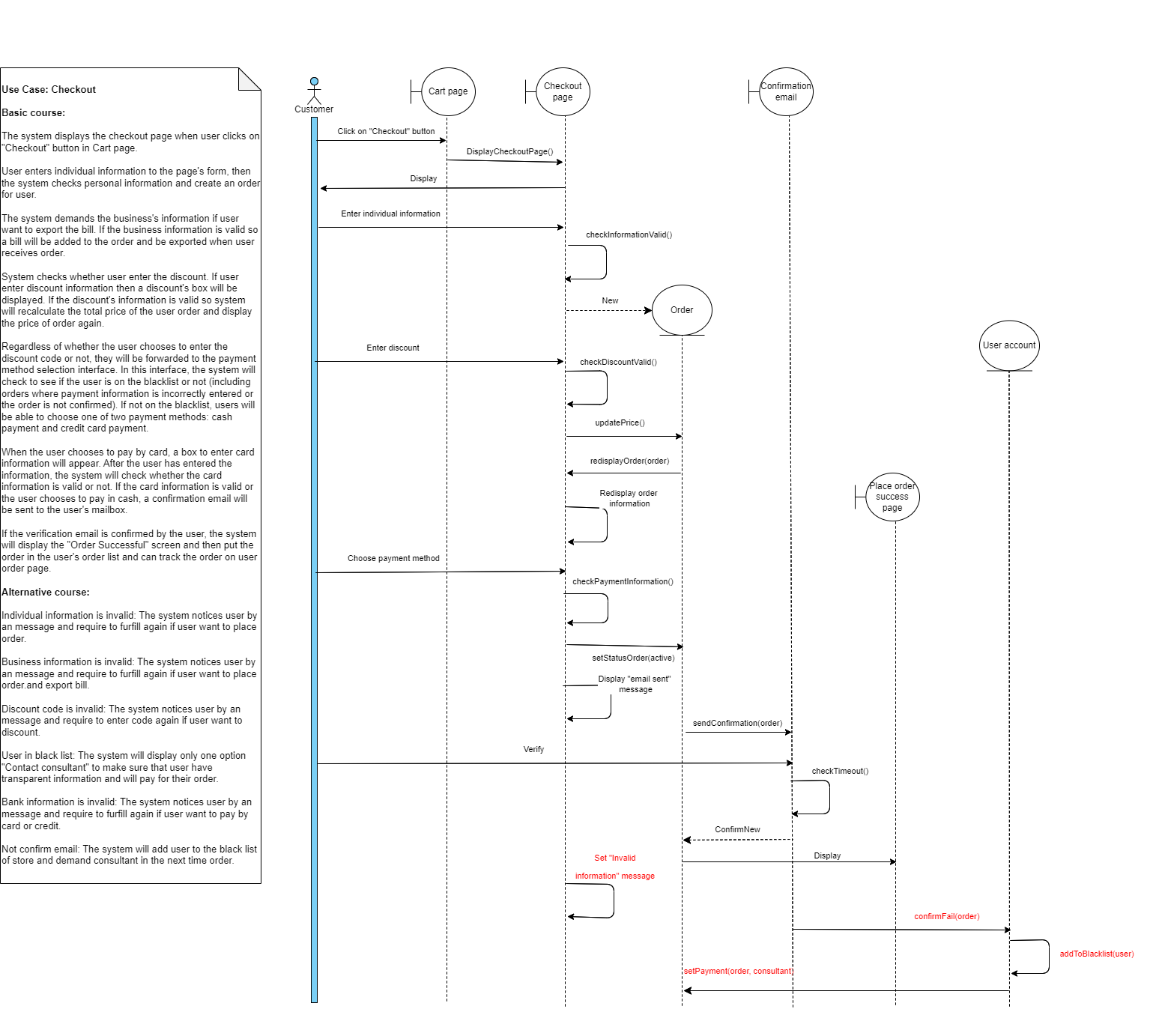


## 2. Use case “Place Order”

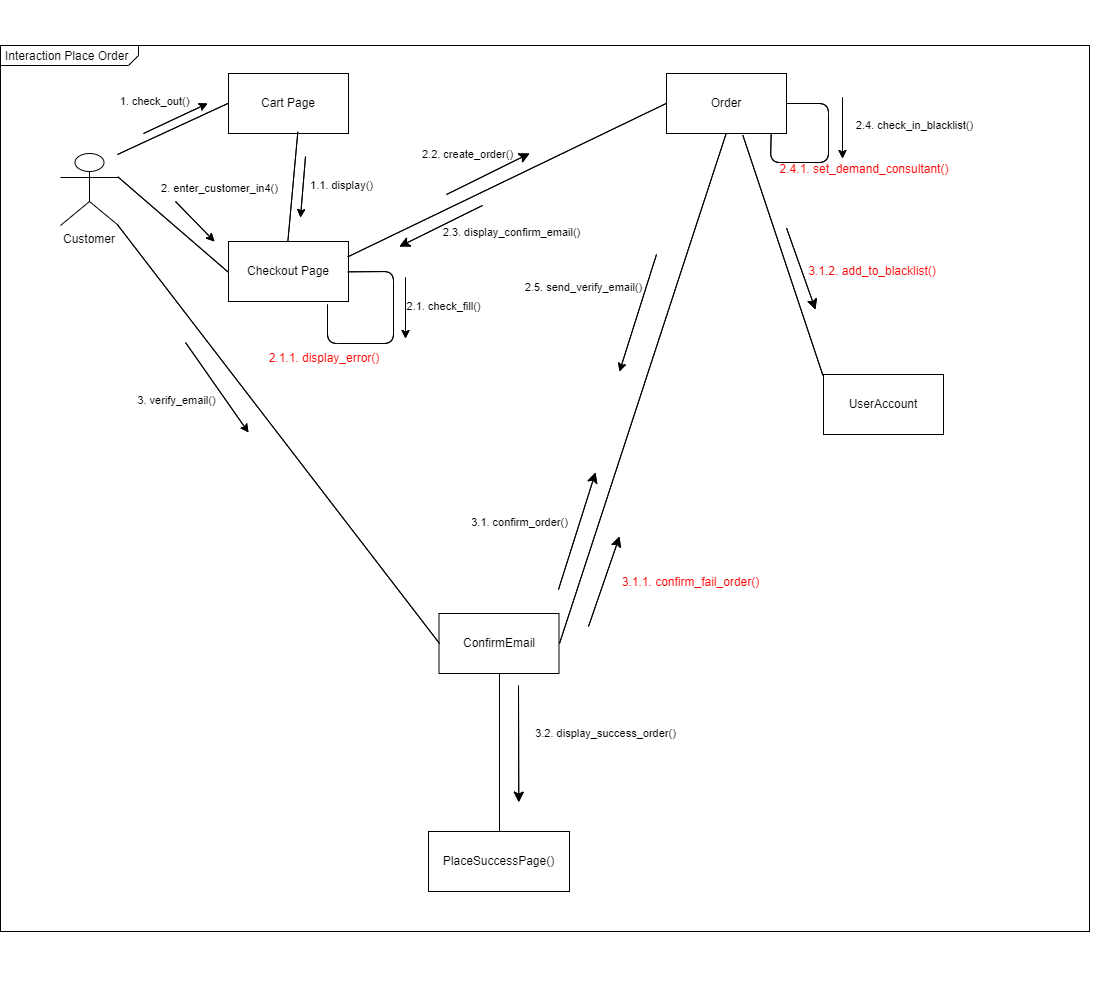
### a, Robustness diagram



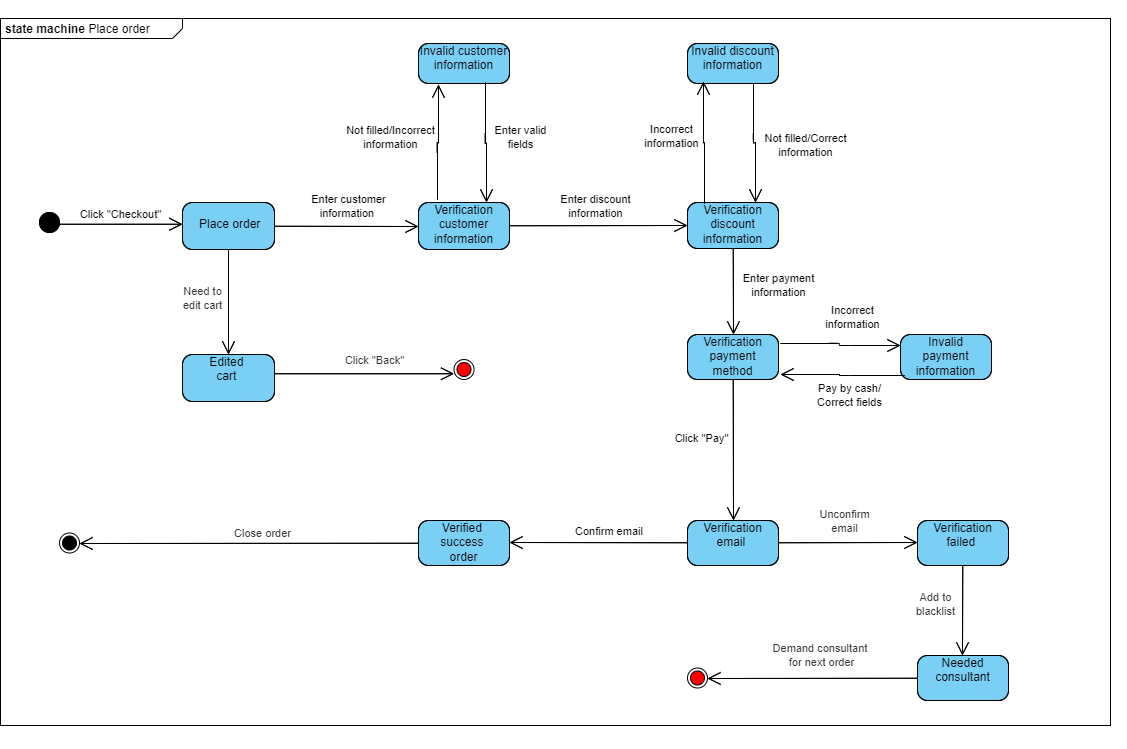
### b, Sequence diagram



### c, Communication diagram

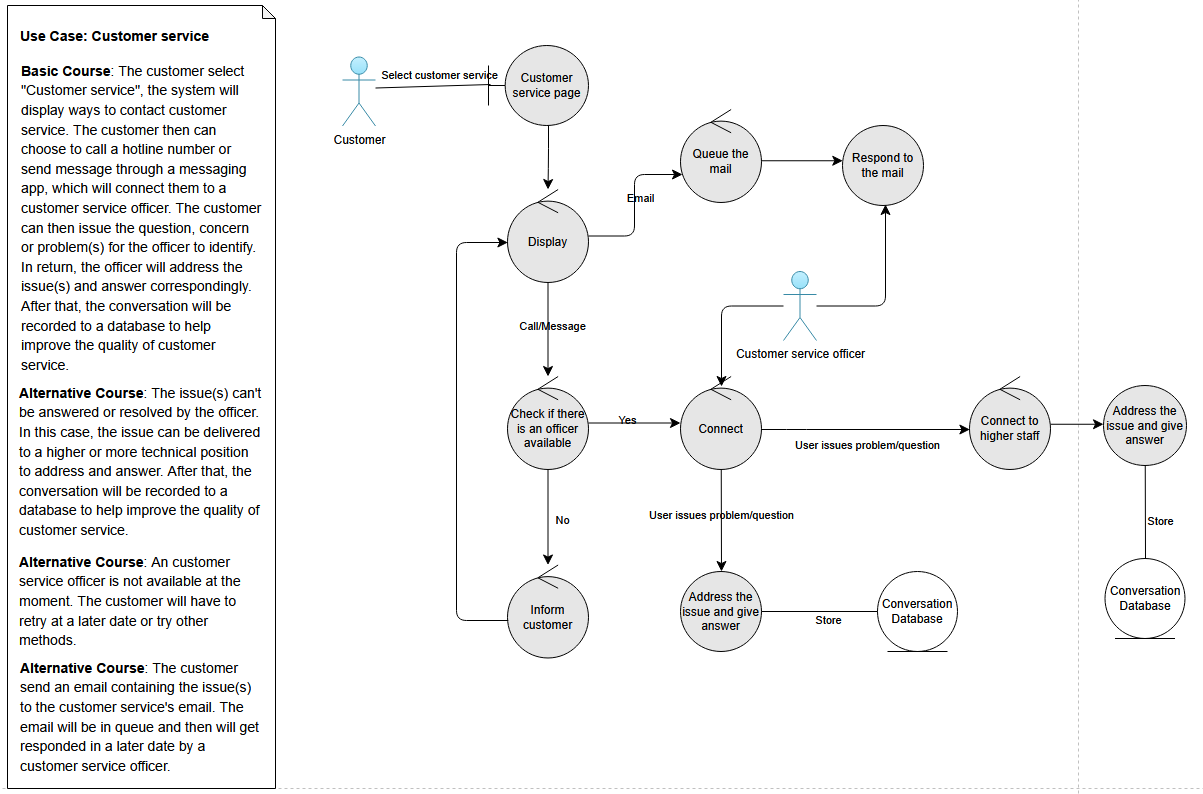


### d, State machine diagram

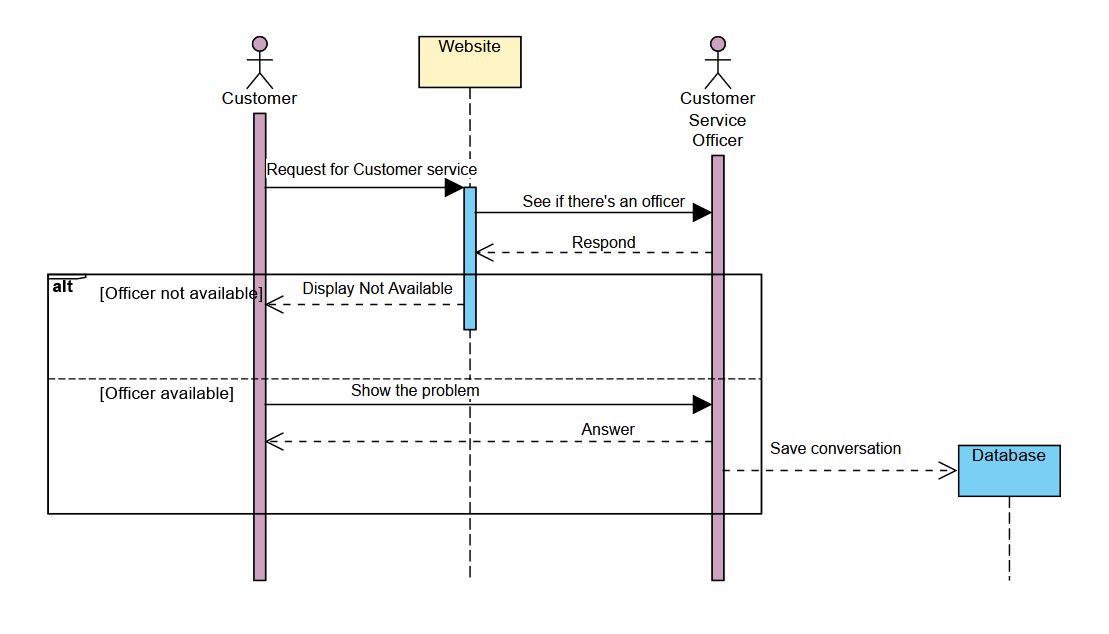


## 3. Use case “Customer Service”

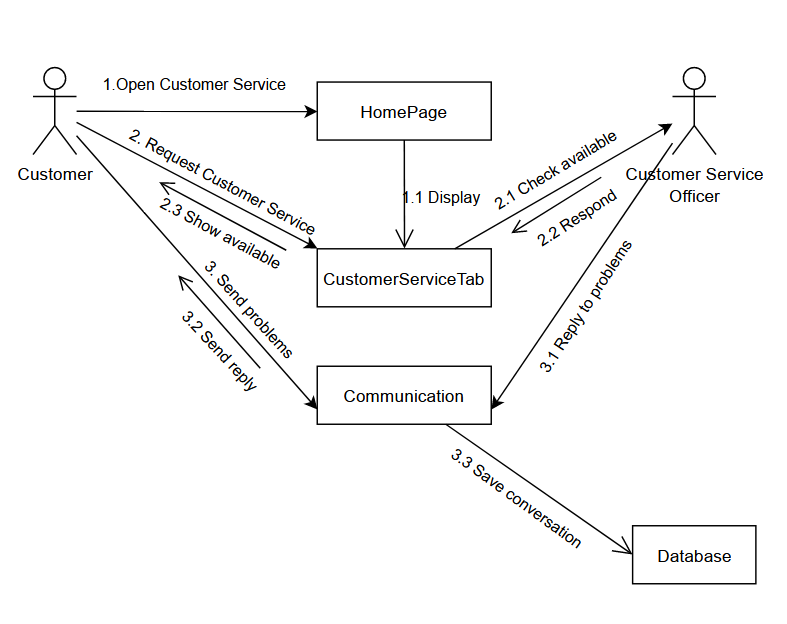
### a, Robustness diagram



### b, Sequence diagram



### c, Communication diagram



### d, State machine diagram

