Online Pizza Ordering System

**<Use Case Model>**

소프트웨어공학 02분반

20171105 이민욱

20173875 정용준

20170223 신원준

20172609 여일구

20174438 정종민

20173156 김준기

목차

[Online Pizza Ordering System 1](#_Toc70187119)

[**<Use Case Model>** 1](#_Toc70187120)

[**1.** **UseCases** 3](#_Toc70187121)

[1.1 **SubgroupA(PizzaOrder)** 3](#_Toc70187122)

[**1.2 SubgroupB(Management)** 3](#_Toc70187123)

[**1.3 SubgroupC(OrderStatus)** 4](#_Toc70187124)

[**2.** **Actors and UseCases** 4](#_Toc70187125)

[**2.1 SubgroupA(PizzaOrder)** 4](#_Toc70187126)

[**2.2 SubgroupB(Management)** 5](#_Toc70187127)

[**2.3 SubgroupC(OrderStatus)** 5](#_Toc70187128)

[**3.** **UseCase Details** 6](#_Toc70187129)

[**3.1 SubgroupA(PizzaOrder)** 6](#_Toc70187130)

[**3.2 SubgroupB(Management)** 10](#_Toc70187131)

[**3.3 SubgroupC(OrderStatus)** 18](#_Toc70187132)

[4. UseCase UML 24](#_Toc70187133)

[4.1 SubgroupA(PizzaOrder) 24](#_Toc70187134)

[4.2 SubgroupB(Management) 25](#_Toc70187135)

[4.3 SubgroupC(OrderStatus) 26](#_Toc70187136)

[5. Traceability Matrix 27](#_Toc70187137)

[**5.1 SubgroupA(PizzaOrder)** 27](#_Toc70187138)

[**5.2 SubgroupB(Management)** 28](#_Toc70187139)

[**5.3 SubgroupC(OrderStatus)** 28](#_Toc70187140)

# **UseCases**

* 1. **SubgroupA(PizzaOrder)**

A-UC1 : Add Menu To Cart

A-UC2 : Manage Cart - generalize UC3, UC4

A-UC3 : Modify Cart Item Quantity

A-UC4 : Remove Cart Item

A-UC5 : Order

A-UC6 : View Order History

A-UC7 : Cancel Order - include UC6

A-UC8 : Pay Online

**1.2 SubgroupB(Management)**

B-UC1: AuthenticateUser

B-UC2:ManageMenus extends UC4~UC6 includes UC2

B-UC3:AddPizza

B-UC4:UpdatePizza

B-UC5:DeletePizza

B-UC6:ManageUsers extends UC8~UC10 includes UC2

B-UC7:AddUser

B-UC8:UpdateUser

B-UC9:DeleteUser

B-UC10:DisplaySales extends UC12~UC13 includes UC2

B-UC11: DisplayByPizza

B-UC12: DisplayByPeriod

B-UC13: SwitchOpening

**1.3 SubgroupC(OrderStatus)**

C-UC1: Accept order

C-UC2: Complete Cook

C-UC3: Handle emergency

C-UC4: Start Delivery

C-UC5: Complete Delivery

C-UC6: Track order

C-UC7: Logout

C-UC8: Sign up

C-UC9: Display user's info page

C-UC10: Display Orders

C-UC11: Login User

# **Actors and UseCases**

**2.1 SubgroupA(PizzaOrder)**

|  |  |  |
| --- | --- | --- |
| Actor | Actor's Goal(what the actor intends to accomplish) | Use Case Name |
| Orderer | To add a menu item with options to cart | A-UC1 |
| Orderer | To view and edit/remove items in the cart | A-UC2 |
| Orderer | Modify the quantity of the item in cart | A-UC3 |
| Orderer | Remove items in the cart | A-UC4 |
| Orderer | Order pizza | A-UC5 |
| Orderer | See the own order history | A-UC6 |
| Orderer | See the detail of one order | A-UC7 |
| Orderer | cancel the order | A-UC8 |

**2.2 SubgroupB(Management)**

|  |  |  |
| --- | --- | --- |
| Actor | Actor’s Goal(what the actor intends to accomplish) | Use Case Name |
| User | To Authenticate role. | B-UC1 |
| Manager | To manage menus, add/remove/edit list. | B-UC2 |
| Manager | To register new menu to menu list. | B-UC3 |
| Manager | To edit data of menu item. | B-UC4 |
| Manager | To delete items in menu list. | B-UC5 |
| Manager | To manage users, ‘granting privilege, add/remove/edit list’ | B-UC6 |
| Manager | To Add a new employee’s account who has particular authorities for the system | B-UC7 |
| Manager | -To edit a user’s information.  -To endow privilege or role of the user | B-UC8 |
| Manager | To delete a user account | B-UC9 |
| Manager | To view sales of the store by some options. | B-UC10 |
| Manager | To view sales list of particular pizza. | B-UC11 |
| Manager | To view total sales list by period. | B-UC12 |
| Manager | To open/close store by switch. | B-UC13 |

**2.3 SubgroupC(OrderStatus)**

|  |  |  |
| --- | --- | --- |
| Actor | Actor’s Goal(what the actor intends to accomplish) | Use Case Name |
| Clerk | Clerk decides whether receiving order or not | C-UC1 |
| Clerk | Clerk changes the status of order from “order accepted” to “cook completed”. | C-UC2 |
| Clerk | In emergency senario, clerk refuses the order. | C-UC3 |
| Delivery man | After “cook completed”, delivery man receives it and start delivery or serving. | C-UC4 |
| Delivery man | When delivery is completed, delivery man changes the status from ”delivering” to “delivery completed”. | C-UC5 |
| Order Tracker | Older Tracker keeps updating the last status of order. | C-UC6 |
| Orderer | Remove user’s data from the device.  Or Prepare to login with another user. | C-UC7 |
| Orderer | Signup the member. | C-UC8 |
| Orderer | Orderer wants to look user info page. | C-UC9 |
| Screen | Display order list with each condition. | C-UC10 |
| User | The user wants to authenticate by the system. | C-UC11 |

# **UseCase Details**

**3.1 SubgroupA(PizzaOrder)**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case A-UC1 | | | AddMenuToCart |
| Related Requirements | | | REQ-1, REQ-2, REQ-3, REQ-4, REQ-5, REQ-7, REQ-13 |
| Initiating Actor | | | Orderer |
| Actor’s Goal | | | To add a menu item with options to cart |
| Participating Actors | | | Cart, Authenticator |
| Preconditions | | | At least there is more than one menu in the system database. |
| Postconditions | | | Selected item is added to cart |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Orderer accesses main page | |
| <- | 2. | System shows the list of menus to orderer. | |
| -> | 3. | Orderer selects menu in the list. | |
| <- | 4. | System redirects to detail page of the selected menu. | |
| -> | 5. | Orderer selects size, toppings and other options. | |
| <- | 6. | System recalculates and displays the menu price according to the options selected. | |
| -> | 7. | Orderer clicks “Add to cart” button | |
| <- | 8. | System asks if the orderer is logged in to Authenticator | |
| <- | 9. | System requests to Cart to add an item to cart from data source according to the menu and options selected | |
| <- | 10. | System asks the orderer if they want to go to cart | |
| -> | 11. | Orderer responds with “Go to cart” | |
| <- | 12. | System redirects to cart page | |
| Flow of Events for Extensions (Alternate Scenarios): | | | |
| 11a. Orderer responds with “Do not go to cart” | | | |
| <- | 1. | System keeps the current page | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case A-UC2 | | | | ManageCart |
| Related Requirements | | | | REQ-3, REQ-8, REQ-9, REQ-14, REQ-16, REQ-17, REQ-18, REQ-21, REQ-26, REQ-27, REQ-28 |
| Initiating Actor | | | | Orderer |
| Actor’s Goal | | | | To View and edit/remove items in the cart |
| Participating Actors | | | | Cart, Authenticator |
| Preconditions | | | | - |
| Postconditions | | | | Apply operations for items, and move pages if needed |
| Flow of Events for Main Success Scenario: | | | | |
| -> | 1. | Orderer accesses cart page | | |
| <- | 2. | System asks if the orderer is logged in to Authenticator | | |
| <- | 3. | System requests to Cart to get list of items in orderer’s cart from data source | | |
| <- | 4. | System displays all items, total price, “Order” button | | |
| -> | 5. | Orderer does some actions for (a)viewing detail information of the item, (b)modifying quantity of the item, (c)removing the item from list, (d) deciding to order | | |
| Flow of Events for Extensions(Alternate Scenarios): | | | | |
| 4a. There is no item in the cart | | | | |
| <- | 1. | System displays message that there is no item in the cart  and disables order action | | |
| 4b. It is outside of opening hours | | | | |
| <- | 1. | System shows message that order is not available now, informs opening hours,  and disable order action | | |
| 5a. Orderer’s action is for viewing details of the item | | | | |
| <- | 1. | System redirects to detail page of the selected menu. | | |
| 5b. Orderer’s action is for modifying quantity of the item | | | | |
|  | 1. | Include Edit ModifyCartItemQuantity (A-UC3) | | |
| 5c. Orderer’s action is for removing the item from list | | | | |
|  | 1. | Include RemoveCartItem (A-UC4) | | |
| 5d. Orderer’s action is for deciding to order | | | | |
|  | 1. | Include Order (A-UC5) | | |
|  | 1a. Orderer is not logged in | | | |
|  | <- | 1. | System notifies some messages about continuing as a non-member | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case A-UC3 | | | ModifyCartItemQuantity |
| Related Requirements, | | | REQ-26, REQ-27 |
| Initiating Actor | | | Orderer |
| Actor’s Goal | | | Modify the quantity of the item in cart |
| Participating Actors | | | Cart, Authenticator |
| Preconditions | | | There is at least one item in cart  and displaying list of cart items with editable quantity |
| Postconditions | | | The modified quantity of the item is stored into the data source |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Orderer changes quantity value of an item | |
| <- | 2. | System asks if the orderer is logged in to Authenticator | |
| <- | 3. | System requests to Cart to update the quantity of the item in data source | |
| Flow of Events for Extensions(Alternate Scenarios): | | | |
| 1a. Quantity is given as zero or negative | | | |
| <- | 1. | System sets the quantity value as 1 | |
|  | 2. | Same from step 2 | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case A-UC4 | | | RemoveCartItem |
| Related Requirements | | | REQ-8, REQ-27 |
| Initiating Actor | | | Orderer |
| Actor’s Goal | | | Remove items in the cart |
| Participating Actors | | | Cart, Authenticator |
| Preconditions | | | There is at least one item in cart  and displaying list of cart items with remove button |
| Postconditions | | | Removed item is deleted from data source |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Orderer clicks remove button of an item | |
| <- | 2. | System asks if the orderer is logged in to Authenticator | |
| -> | 3. | System requests to Cart to delete the item from data source | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case A-UC5 | | | Order |
| Related Requirements | | | REQ-6, REQ-9, REQ10, REQ-11,REQ-12, REQ-15, REQ-16, REQ-17, REQ-19, REQ-21, REQ-22, REQ-23, REQ-24, REQ-28 |
| Initiating Actor | | | Orderer |
| Actor’s Goal | | | To order pizza |
| Participating Actors | | | System, Order List |
| Preconditions | | | There must be pizza in cart (A-UC1 – Add Menu To Cart) |
| Postconditions | | | Orderer get order id |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Orderer clicks “move to order page” button from cart page | |
| <- | 2. | System shows the order page to orderer, if store is open | |
| -> | 3. | Orderer select receiving option (REQ-10) | |
|  | 4. | If Orderer select delivery option , Orderer fill in the address entry box | |
|  | 5.. | Orderer type extra requirements (REQ-6) | |
|  | 6. | Orderer type payment information (REQ-11, REQ-12) | |
|  | 7. | Orderer click “order complete” | |
|  | 8. | The system put the order in Order List | |
| <- | 9. | The system give order Id to orderer. (REQ-19) | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case A-UC6 | | | View Order History |
| Related Requirements | | | REQ-20, REQ-49 |
| Initiating Actor | | | Orderer |
| Actor’s Goal | | | To see the own order history |
| Participating Actors | | | System, Order List |
| Preconditions | | | Login |
| Postconditions | | |  |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Orderer click order history view button | |
| <- | 2. | The system displays the orderer’s order history to orderer | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case A-UC7 | | | View Order Detail |
| Related Requirements | | | REQ-20, REQ-49 |
| Initiating Actor | | | Orderer |
| Actor’s Goal | | | To see the detail of one order |
| Participating Actors | | | System |
| Preconditions | | |  |
| Postconditions | | |  |
| Flow of Events for Main Success Scenario: | | | |
|  | 1. | Orderer click “order detail” button | |
|  | 2. | Authenticate | |
| <- | 3. | The system show the order detail page | |
| Extends | | | |
| 2.1 | If Orderer is logged in | | |
| <- | 2.1.1 | The system displays order list of orderer | |
| -> | 2.1.2 | Orderer select an order | |
| 2.2 | If Orderer is not logged in | | |
| <- | 2.2.1 | The system displays the box where orderer type order id | |
| -> | 2.2.2 | Orderer type order id | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case A-UC8 | | | Cancle Order |
| Related Requirements | | | REQ-20, REQ-25, REQ-49 |
| Initiating Actor | | | Orderer |
| Actor’s Goal | | | To cancle order |
| Participating Actors | | | System, Order list |
| Preconditions | | | A-UC6, Before Clerk accept order |
| Postconditions | | |  |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Orderer move to order detail page | |
| <- | 2. | The system show the detail page to orderer | |
| -> | 3. | Orderer click “cancle” button | |
|  | 4. | If order is not accepted, The system cancle the order | |
| <- | 5. | The system displays the result of cancling order | |

**3.2 SubgroupB(Management)**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC1 | | | Authenticate User |
| Related Requirements | | | [REQ61], [REQ67], [REQ45] |
| Initiating Actor | | | User |
| Actor’s Goal | | | The orderer wants to access without privileges.  The authenticator wants to block the orderer. |
| Participating Actors | | | Authenticator |
| Preconditions | | | The User is loginned in |
| Postconditions | | |  |
| Flow of Events for Main Succes Scenario: | | | |
| -> | 1. | A user tries to pages with user information. | |
| <- | 2. | The Authenticator returns the role of the user | |
| <- | 3. | The system checks whether the User matches with the requiring role. | |
|  | 3-1. | Allow the access. | |
|  | 3-2. | Deny the access. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC2 | | | ManageMenus |
| Related Requirements | | | REQ-30, REQ-32, REQ-35 |
| Initiating Actor | | | Manager |
| Actor’s Goal | | | To manage menus, add/remove/edit list. |
| Participating Actors | | | Menulist |
| Preconditions | | | System has a user interface that can add, edit, and remove menu list item. |
| Postconditions | | | * The modified data is stored into the Menulist * Redirect or Refresh to menu list page |
| Flow of Events for Main Success Scenario: | | | |
|  | 1. | AuthenticateUser: include B-UC1 | |
| -> | 2. | Manager clicks the link “Menu Management”. | |
| <- | 3. | System requests Menulist to get list of the menus. | |
| -> | 4. | Menulist returns the requested data. | |
| <- | 5. | System displays list of menus(including their information) to the Manager. | |
| -> | 6. | Manager selects appropriate option for managing menus. | |
| <- | 7. | System requests Menulist to (a) store or (b) update or (c) delete the menu data. | |
| Flow of Events for Extensions (Alternate Scenarios): | | | |
| 6a. | | Selected button entails adding new menu item: Include AddMenu(B-UC3) | |
| 6b. | | Selected button entails editing menu item: Include UpdateMenu(B-UC4) | |
| 6c. | | Selected button entails removing menu item: Include DeleteMenu(B-UC5) | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC3 | | | AddMenu |
| Related Requirements | | | REQ-35 |
| Initiating Actor | | | Manager |
| Actor’s Goal | | | To register new menu to menu list. |
| Participating Actors | | | Menulist |
| Preconditions | | | * ‘AddMenu’ button or icon presents in the page. |
| Postconditions | | | * Redirect to menu list page. |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Manager clicks the ‘AddMenu’ button. | |
| <- | 2. | System displays a page with text blanks to fill in with essential information(name, type, price, allergy, and short description). | |
| -> | 3. | Manager fills in each blanks and press “complete button”. | |
| <- | 4. | System requests Menulist to store the new data. | |
| Flow of Events for Extensions (Alternate Scenarios): | | | |
| 3a. | | Manager fills in the blanks with invalid type of data. | |
| <- | | System (a) displays error message over the blank and (b) disable “complete button”. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC4 | | | UpdateMenu |
| Related Requirements | | | REQ-35 |
| Initiating Actor | | | Manager |
| Actor’s Goal | | | To edit data of menu item. |
| Participating Actors | | | Menulist |
| Preconditions | | | -‘Edit Menu’ button or icon presents in each line of the menu item.  -More than one menu be in list. |
| Postconditions | | | * The modified data is stored into the Menulist. * System redirects to menu list page. |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Manager clicks the “edit button” located on each item of the menu list. | |
| <- | 2. | System requests for the clicked menu’s data to Menulist. | |
| -> | 3. | Menulist returns the requested menu data. | |
| <- | 4. | System shows input list for essential information(name, type, price, allergy, short description) filled with returned menu data. | |
| -> | 5. | Manager edits the menu data and submit to the system. | |
| <- | 6. | System requests Menulist to store editted information. | |
| Flow of Events for Extensions (Alternate Scenarios): | | | |
| 3a. | | Manager fills in the blanks with invalid type of data. | |
| <- | | System (a) displays error message over the blank and (b) disable “complete button”. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC5 | | | DeleteMenu |
| Related Requirements | | | REQ-35 |
| Initiating Actor | | | Manager |
| Actor’s Goal | | | To delete items in menu list. |
| Participating Actors | | | Menulist |
| Preconditions | | | -‘DeleteMenu’ button or icon presents in each line of the menu list.  -More than one menus be in list. |
| Postconditions | | | * Menulist deletes the menu. * System refreshes current page. |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Manager clicks the ‘DeleteMenu’ button next to the particular line of the menu list. | |
| <- | 2. | System requests Menulist to delete the clicked menu. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC6 | | | ManageUsers |
| Related Requirements | | | [REQ33], [REQ34], [REQ37], [REQ38] |
| Initiating Actor | | | Manager |
| Actor’s Goal | | | To manage users, ‘granting privilege, add/remove/edit list’ |
| Participating Actors | | | User(Orderer, Clerk, Delivery man, Manger) |
| Preconditions | | | * More than one user be in list * Users has their own appropriate authorities. * System has a user interface that can edit or remove each user account. * System has a user interface that can add an user account |
| Postconditions | | | * The modified data is stored into the User. * Redirect or Refresh to user list page |
| Flow of Events for Main Success Scenario: | | | |
|  | 1. | AuthenticateUser: include B-UC1 | |
| -> | 2. | Manager selects clicks the link “User Management” | |
| <- | 3. | System requests User to get list of the users. | |
| -> | 4. | User returns the requested data. | |
| <- | 5. | System displays (a)list of users(including their information) to the Manager and (b)user interfaces | |
| -> | 6. | Manager selects an appropriate option for managing users. | |
| <- | 7. | System requests User to (a)store or (b)update or (c)delete the data. | |
| Flow of Events for Extensions (Alternative Scenarios): | | | |
| 6a. | Selected activity entails adding new users: Include AddUser(B-UC7) | | |
| 6b. | Selected activity entails updating new users: Include UpdateUser(B-UC8) | | |
| 6c. | Selected activity entails deleting new users: Include DeleteUser(B-UC9) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC7 | | | AddUser |
| Related Requirements | | | [REQ37], [REQ68] |
| Initiating Actor | | | Manager |
| Actor’s Goal | | | * To Add a new employee’s account who has particular authorities for the system |
| Participating Actors | | | User(Orderer, Clerk, Delivery man, Manger) |
| Preconditions | | | * ‘AddUser’ button or icon presents in the page * New user’s information is not redundant. * New user has his/her own special role(Orderer, Clerk, Delivery man, Delivery man) for the system |
| Postconditions | | | * The new user data is stored into the User. * Redirect to user list page |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Manager clicks ‘AddUser’ button | |
| <- | 2. | System displays a page with text blanks to fill in with essential information(id, password, name, address, email, Role, phone number) | |
| -> | 3. | Manager fills in each blanks and press “complete button”. | |
| <- | 4. | System encrypts password and requests User to store information. | |
| Flow of Events for Extensions (Alternative Scenarios): | | | |
| 3a. | Manager fills in the blanks with invalid type of data or insufficient password. | | |
| <- | System (a) displays error message over the blank and (b) disable “complete button”. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC8 | | | UpdateUser |
| Related Requirements | | | [REQ37], [REQ68] |
| Initiating Actor | | | Manager |
| Actor’s Goal | | | * To edit a user’s information. * To endow privilege or role of the user. |
| Participating Actors | | | User(Orderer, Clerk, Delivery man, Manger) |
| Preconditions | | | * ‘Edit user’ button or icon presents in each line of the user information. * The user has his/her own special role(Orderer, Clerk, Delivery man) for the system. * Existing information is filled in the form * More than one user be in list. |
| Postconditions | | | * The modified data is stored in the User * Redirect to user list page. |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Manager clicks “edit button” located on each user of the user list. | |
| <- | 2. | System requests the user data stored in User. | |
| -> | 3. | User returns the existing user data of the user. | |
| <- | 4. | System shows input list for essential information(id, password, name, address, email, Role, phone number and existing information is filled. | |
| -> | 5. | Manager (a) edits information and submit to the system or (b) endows privilege or role of the user | |
| <- | 6. | System encrypts password and requests User to store editted information. | |
| Flow of Events for Extensions (Alternative Scenarios): | | | |
| 5a. | Manager fills in the blanks with invalid type of data or insufficient password. | | |
| <- | System (a) displays error message over the blank and (b) disable “complete button”. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC9 | | | DeleteUser |
| Related Requirements | | | [REQ37] |
| Initiating Actor | | | Manager |
| Actor’s Goal | | | To delete a user account |
| Participating Actors | | | User(Orderer, Clerk, Delivery man, Manger) |
| Preconditions | | | * ‘DeleteUser’ button or icon presents in each line of the user information. * The user has his/her own special role(Orderer, Clerk, Delivery man) for the system. * More than one user be in list. |
| Postconditions | | | * The user account is removed from User. * Refresh page |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Manager clicks DeleteUser button next to the particular line of the user list. | |
| <- | 2. | System requests User to delete the account. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC10 | | | DisplaySales |
| Related Requirements | | | REQ-29, REQ-30, REQ-36 |
| Initiating Actor | | | Manager |
| Actor’s Goal | | | To view sales of the store by some options. |
| Participating Actors | | | Orderlist |
| Preconditions | | | System has a user interface that can select criteria to view. |
| Postconditions | | | none |
| Flow of Events for Main Success Scenario: | | | |
|  | 1. | AuthenticateUser: UC-1 | |
| -> | 2. | Manager clicks the link “Display sales” | |
| <- | 3. | System requests for total sales list to Orderlist. | |
| -> | 4. | Orderlist returns the requested data. | |
| <- | 5. | System displays total sales list, total sales revenue, and view criteria(including drop-down menu “by pizza” and two blanks for start date, end date). | |
| -> | 6. | Manager specifies the criteria and clicks “Apply button”. | |
| <- | 7. | System requests for sales list that matches the specified criteria to Orderlist. | |
| -> | 8. | Orderlist returns the matching sales list. | |
| <- | 9. | System displays the sales list and revenue returned by Orderlist. | |
| Flow of Events for Extensions (Alternate Scenarios): | | | |
| 6a. | | Selected drop-down menu entails displaying new sales list, revenue by pizza: Include DisplayByPizza(B-UC11) | |
| 6b. | | Filled in blanks(start date, end date) entails displaying new sales list, revenue by period: Include DisplayByPeriod(B-UC12) | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC11 | | | DisplayByPizza |
| Related Requirements | | | REQ-29, REQ-36 |
| Initiating Actor | | | Manager |
| Actor’s Goal | | | To view sales list of particular pizza. |
| Participating Actors | | | Orderlist |
| Preconditions | | | System has drop-down menu consists of menu list. |
| Postconditions | | | none |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Manager clicks the drop-down menu “by pizza”. | |
| <- | 2. | System provide filtering options consists of menu list and “none”. | |
| -> | 3. | Manager selects one option from list and clicks “Apply button”. | |
| <- | 4. | System requests for sales list that matches the specified criteria to Orderlist. | |
| -> | 5. | Orderlist returns the matching sales list. | |
| <- | 6. | System displays the sales list and revenue returned by Orderlist. | |
| Flow of Events for Extensions (Alternate Scenarios): | | | |
| 3a. | | Manager selects “none” option. | |
| <- | 1. | System filters sales list only by period. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC12 | | | DisplayByPeriod |
| Related Requirements | | | REQ-29, REQ-36 |
| Initiating Actor | | | Manager |
| Actor’s Goal | | | To view total sales list by period. |
| Participating Actors | | | Orderlist |
| Preconditions | | | System has two blanks to fill in with date. |
| Postconditions | | | none |
| Flow of Events for Main Success Scenario: | | | |
| -> | 1. | Manager fills in the blanks(start date, end date) with date type text and clicks “Apply button”. | |
| <- | 2. | System requests for sales list that matches the specified criteria to Orderlist. | |
| -> | 3. | Orderlist returns the matching sales list. | |
| <- | 4. | System displays the sales list and revenue returned by Orderlist. | |
| Flow of Events for Extensions (Alternate Scenarios): | | | |
| 1a. | | Manager fills in the start date blank incompletely. | |
| <- | 1. | System fills in the start date with open date automatically. | |
| 1b. | | Manager fills in the end date blank incompletely. | |
| <- | 1. | System fills in the end date with current date automatically. | |
| 1c. | | Manager fills in the date blank with invalid type data. | |
| <- | 1. | System displays error message over the blank and disable “Apply button”. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case B-UC13 | | | SwitchOpening |
| Related Requirements | | | REQ-30, REQ-31 |
| Initiating Actor | | | Manager |
| Actor’s Goal | | | To open/close store by switch. |
| Participating Actors | | | none |
| Preconditions | | | System has a toggle switch on admin page. |
| Postconditions | | | none |
| Flow of Events for Main Success Scenario: | | | |
|  | 1. | AuthenticateUser: B-UC1 | |
| -> | 2. | Manager clicks the toggle switch on admin page to open/close store. | |
| <- | 3. | System changes state(ON->OFF, OFF->ON). | |
| Flow of Events for Extensions (Alternate Scenarios): | | | |
| 2a. | | The state of store is “Closed” | |
| <- | 1. | System changes state from “Closed” to “Opened”. | |
| 2b. | | The state of store is “Opened”. | |
| <- | 1. | System changes state from “Opened” to “Closed”. | |

**3.3 SubgroupC(OrderStatus)**

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case C-UC1 | | | Accept order |
| Related Requirementxfs | | | [REQ54] The system should allow a clerk to change order's status from "ordered" to ("order accepted" or "order rejected"). |
| Initiating Actor | | | Clerk |
| Actor’s Goal | | | Clerk decides whether receiving order or not |
| Participating Actors | | | Order Tracker, Screen |
| Preconditions | | | None |
| Postconditions | | | C-UC2 |
| Flow of Events for Main Succes Scenario: | | | |
|  | 1 | User indentification(include Authenticate user C-UC8) | |
|  | 2 | The system takes orders from user and transfers it to Clerk. | |
| → | 3 | Clerk checks the order and updates its status from “ordered” to “ordered accepted” or “ordered rejected”. | |
| ← | 4 | Order Tracker updates order’s status. | |
| ← | 5 | Screen displays the Order Tracker. | |
| Use Case C-UC2 | | | Complete Cook |
| Related Requirements | | | [REQ55] The system should allow a clerk to change order's status from "order accepted" to "cook completed". |
| Initiating Actor | | | Clerk |
| Actor’s Goal | | | Clerk changes the status of order from “order accepted” to “cook completed”. |
| Participating Actors | | | Order Tracker, Screen |
| Preconditions | | | C-UC1 |
| Postconditions | | | C-UC4 |
| Flow of Events for Main Succes Scenario: | | | |
| → | 1 | When pizza is done, Clerk updates its status from “order accepted” to “cook completed”. | |
| ← | 2 | Order Tracker updates order’s status. | |
| ← | 3 | Screen displays the Order Tracker. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case C-UC3 | | | Handle emergency |
| Related Requirements | | | [REQ56] The system shall change the order status to "invalid" in an emergency by clerk. |
| Initiating Actor | | | Clerk |
| Actor’s Goal | | | In emergency senario, clerk refuses the order. |
| Participating Actors | | | Order Tracker, Screen |
| Preconditions | | | None |
| Postconditions | | | None |
| Flow of Events for Main Succes Scenario: | | | |
|  | 1 | The system takes orders from user and transfers it to Clerk. | |
| → | 2 | In emergency, Clerk updates order’s status from “ordered” to “invlid”. | |
| ← | 3 | Order Tracker updates order’s status. | |
| ← | 4 | Screen displays the Order Tracker. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case C-UC4 | | | Start Delievery |
| Related Requirements | | | [REQ57] The system should allow a delivery man and server to change order's status from "cook completed" to ("delivering" and "served"). |
| Initiating Actor | | | Delivery man |
| Actor’s Goal | | | After “cook completed”, delivery man receives it and start delivery or serving. |
| Participating Actors | | | Order Tracker, Screen |
| Preconditions | | | C-UC2 |
| Postconditions | | | C-UC4 |
| Flow of Events for Main Succes Scenario: | | | |
|  | 1 | User indentification(include Authenticate User C-UC8) | |
| → | 2 | Delivery man receives the pizzas that have to deliver to orderer. | |
| → | 3 | Delivery man updates its status from “cook completed” to “delivering” or “served”.. | |
| ← | 4 | Order Tracker updates order’s status. | |
| ← | 5 | Screen displays the Order Tracker. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case C-UC5 | | | Complete Delivery |
| Related Requirements | | | [REQ58] The system should allow a delivery man to change order's status from "delivering" to "delivery completed". |
| Initiating Actor | | | Delivery man |
| Actor’s Goal | | | When delivery is completed, delivery man changes the status from ”delivering” to “delivery completed”. |
| Participating Actors | | | Order Tracker, Screen |
| Preconditions | | | C-UC4 |
| Postconditions | | | None |
| Flow of Events for Main Succes Scenario: | | | |
| → | 1 | Delivery man delivers the pizzas to orderer. | |
| → | 2 | Delivery man updates its status from “delivering” to “delivery completed”. | |
| ← | 3 | Order Tracker updates order’s status. | |
| ← | 4 | Screen displays the Order Tracker. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case C-UC6 | | | Track order |
| Related Requirements | | | [REQ59] The system should record the last time when order's status has changed. |
| Initiating Actor | | | Order Tracker |
| Actor’s Goal | | | Older Tracker keeps updating the last status of order. |
| Participating Actors | | | Clerk, Delivery man, Screen |
| Preconditions | | | C-UC1, C-UC2, C-UC3, C-UC4, C-UC5 |
| Postconditions | | | None |
| Flow of Events for Main Succes Scenario: | | | |
| → | 1 | Clerk and Delivery man update status of orders | |
| ← | 2 | When the order status is changed, Order Tracker recognizes and reflects it. | |
| ← | 3 | Screen displays the Order Tracker with last changed time of order’s status. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case C-UC7 | | | Logout |
| Related Requirements | | | [REQ60] The system should return to the initial screen when the user logs out. |
| Initiating Actor | | | User(Orderer, Manager, Clerk, Delivery man) |
| Actor’s Goal | | | Remove user’s data from the device.  Or Prepare to login with another user. |
| Participating Actors | | | Authenticator |
| Preconditions | | | C-UC8(Authenticate User) |
| Postconditions | | |  |
| Flow of Events for Main Succes Scenario: | | | |
| → | 1. | An user click logout button. | |
|  | 2. | The authenticator remove user’s login state. | |
| ← | 3. | Redirect main page. | |

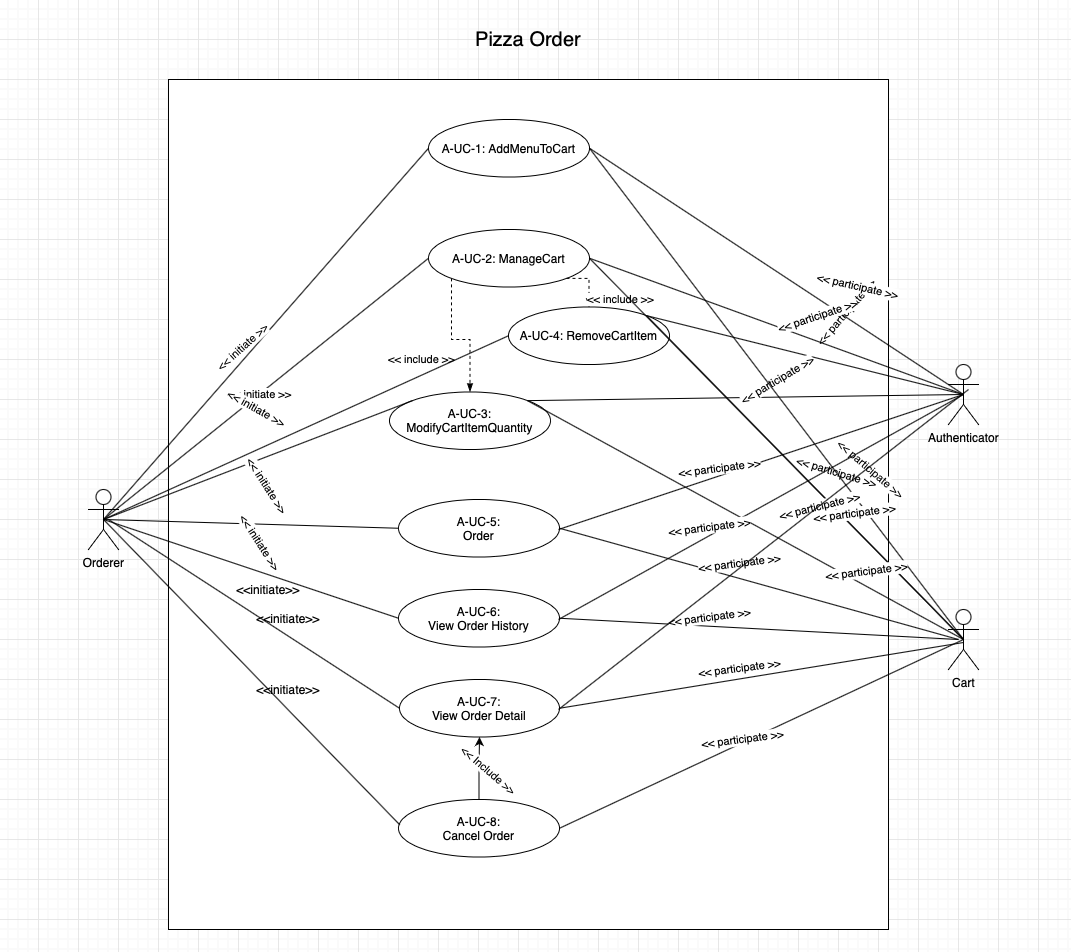
|  |  |  |  |
| --- | --- | --- | --- |
| Use Case C-UC8 | | | Sign up |
| Related Requirements | | | [REQ62] The system should check that the password length is more than 8 characters.  [REQ63] The system should check that the password contains at least one number, upper/lower case and special character.  [REQ68] The system shall encrypt and store orderer pw.  [REQ64] The system should maintain an unique user's ID and email. |
| Initiating Actor | | | User(Orderer, Clerk, Delivery man) |
| Actor’s Goal | | | Signup the member. |
| Participating Actors | | | Authenticator |
| Preconditions | | |  |
| Postconditions | | |  |
| Flow of Events for Main Succes Scenario: | | | |
| -> | 1. | An user wants to register the hompage as a member. | |
|  | 2. | The authenticator check that the user id and email are unique. | |
|  | 3. | The authenticator check that the length of password is more than 8 characters. | |
|  | 4. | The system should check that the password contains at least one number, upper/lower case and special character. | |
| <- | 5-1. | If any of scenario 2, 3, 4 are failed, reject the user’s signup. | |
| <- | 5-2. | Successfully signup. | |
| Use Case C-UC9 | | | Display user’s info page |
| Related Requirements | | | [REQ65] The system should provide user info page.  [REQ66] The system should get the password when the user try to enter user info page. |
| Initiating Actor | | | Orderer |
| Actor’s Goal | | | Orderer wants to look user info page. |
| Participating Actors | | | Authenticator |
| Preconditions | | | C-UC8(Authenticate User) |
| Postconditions | | |  |
| Flow of Events for Main Succes Scenario: | | | |
| -> | 1. | Request User info page with user password | |
|  | 2. | UC8 | |
| <- | 3. | Response User info. | |

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case C-UC10 | | | Display Orders |
| Related Requirements | | | [REQ53] The system shall allow an orderer to see status("ordered" as "cooking", "cook completed") on a simple page.  [REQ69] The system should provide the page that "ordered completed" and "cook completed" can't rewrite(Readonly). |
| Initiating Actor | | | Screen |
| Actor’s Goal | | | Display order list with each condition. |
| Participating Actors | | | Order List |
| Preconditions | | |  |
| Postconditions | | |  |
| Flow of Events for Main Succes Scenario: | | | |
| -> | 1. | Screen requests uncompleted order list. | |
|  | 2. | Order List response uncompleted order list. | |

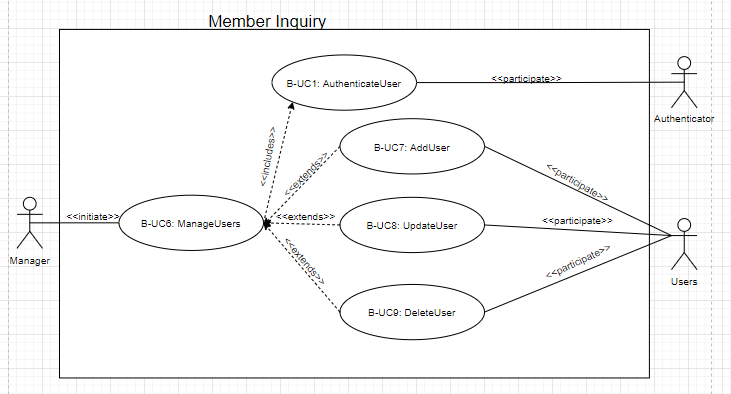
|  |  |  |  |
| --- | --- | --- | --- |
| Use Case C-UC11 | | | Login User |
| Related Requirements | | | [REQ3] The system should display "go to cart", "login"/"logout" buttons on every pages  [REQ52] The system shall allow an orderer to order new pizza both as a member or non-member. |
| Initiating Actor | | | User |
| Actor’s Goal | | | The user wants to authenticate by the system. |
| Participating Actors | | | Authenticator |
| Preconditions | | |  |
| Postconditions | | |  |
| Flow of Events for Main Succes Scenario: | | | |
| -> | 1. | An user inputs his/her ID and password. | |
|  | 2. | Validate ID and password. | |
| <- | 3-1. | When the information is valid, send a success message. | |
| <- | 3-2. | When the information is invalid, send a failure message. | |

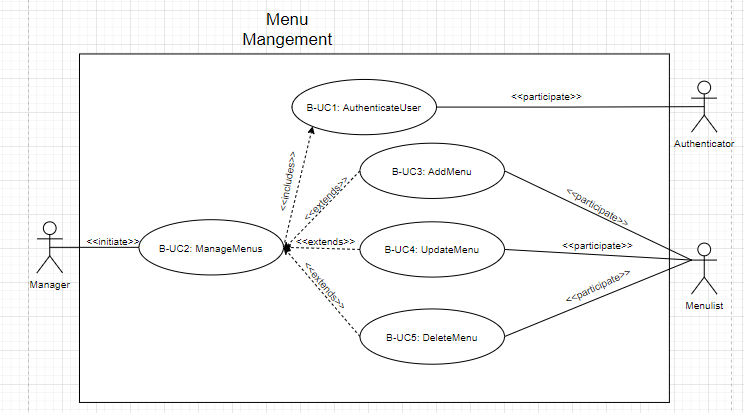
# UseCase UML

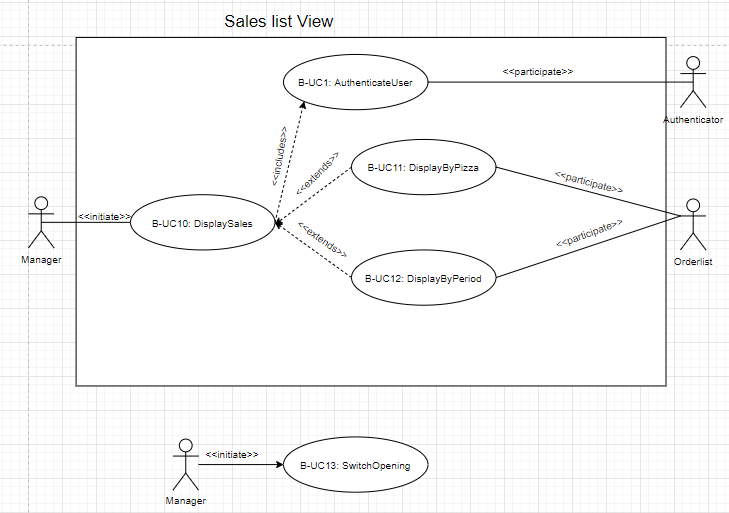
4.1 SubgroupA(PizzaOrder)



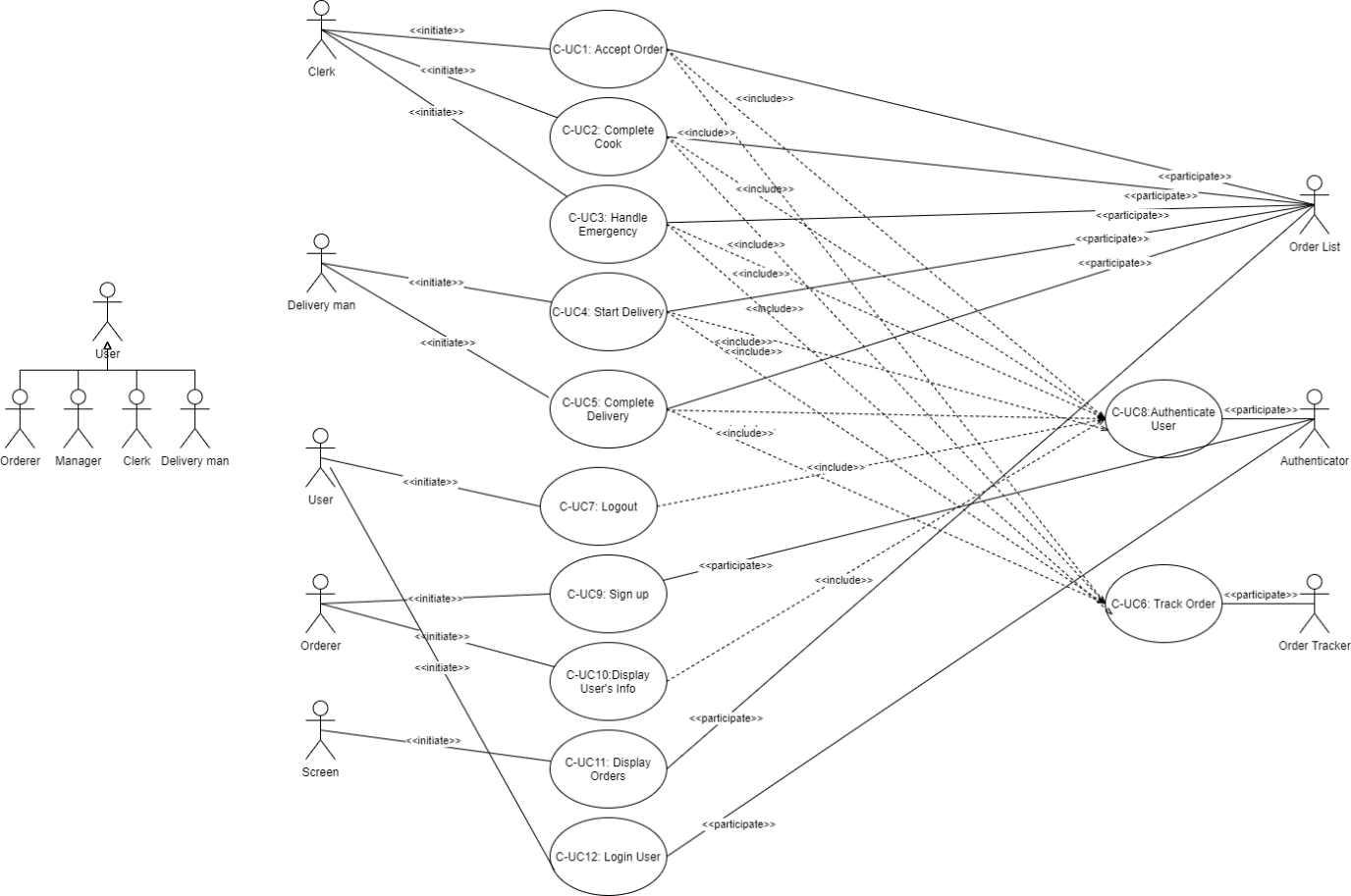
4.2 SubgroupB(Management)







4.3 SubgroupC(OrderStatus)



# Traceability Matrix

**5.1 SubgroupA(PizzaOrder)**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| REQ | PW | UC1 | UC2 | UC3 | UC4 | UC5 | UC6 | UC7 | UC8 |
| REQ1 | 1 | O |  |  |  |  |  |  |  |
| REQ2 | 3 | O |  |  |  |  |  |  |  |
| REQ3 | 1 | O | O |  |  |  |  |  |  |
| REQ4 | 5 | O |  |  |  |  |  |  |  |
| REQ5 | 5 | O |  |  |  |  |  |  |  |
| REQ6 | 3 |  |  |  |  | O |  |  |  |
| REQ7 | 5 | O |  |  |  |  |  |  |  |
| REQ8 | 4 |  | O |  | O |  |  |  |  |
| REQ9 | 5 |  | O |  |  | O |  |  |  |
| REQ10 | 4 |  |  |  |  | O |  |  |  |
| REQ11 | 4 |  |  |  |  | O |  |  |  |
| REQ12 | 4 |  |  |  |  | O |  |  |  |
| REQ13 | 1 | O |  |  |  |  |  |  |  |
| REQ14 | 3 |  | O |  |  |  |  |  |  |
| REQ15 | 3 |  |  |  |  | O |  |  |  |
| REQ16 | 3 |  | O |  |  | O |  |  |  |
| REQ17 | 2 |  | O |  |  | O |  |  |  |
| REQ18 | 1 |  | O |  |  |  |  |  |  |
| REQ19 | 2 |  |  |  |  | O |  |  |  |
| REQ20 | 1 |  |  |  |  |  | O | O | O |
| REQ21 | 5 |  | O |  |  | O |  |  |  |
| REQ22 | 1 |  |  |  |  | O |  |  |  |
| REQ23 | 1 |  |  |  |  | O |  |  |  |
| REQ24 | 4 |  |  |  |  | O |  |  |  |
| REQ25 | 1 |  |  |  |  |  |  |  | O |
| REQ26 | 1 |  | O | O |  |  |  |  |  |
| REQ27 | 4 |  | O | O | O |  |  |  |  |
| REQ28 | 3 |  | O |  |  | O |  |  |  |
| MAX PW | | 5 | 5 | 4 | 4 | 5 | 1 | 1 | 1 |
| Total PW | | 21 | 34 | 5 | 8 | 44 | 1 | 1 | 2 |

**5.2 SubgroupB(Management)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Req’t** | **PW** | B-UC1 | B-UC2 | B-UC3 | B-UC4 | B--UC5 | B-UC6 | B-UC7 | B-UC8 | B-UC9 | B-UC10 | B-UC11 | B-UC12 | B-UC13 |
| REQ29 | 1 |  |  |  |  |  |  |  |  |  | O | O | O |  |
| REQ30 | 5 |  | O |  |  |  |  |  |  |  | O |  |  | O |
| REQ31 | 5 |  |  |  |  |  |  |  |  |  |  |  |  | O |
| REQ32 | 2 |  | O |  |  |  |  |  |  |  |  |  |  |  |
| REQ33 | 2 |  |  |  |  |  | O |  |  |  |  |  |  |  |
| REQ34 | 1 |  |  |  |  |  | O |  |  |  |  |  |  |  |
| REQ35 | 3 |  | O | O | O | O |  |  |  |  |  |  |  |  |
| REQ36 | 3 |  |  |  |  |  |  |  |  |  | O | O | O |  |
| REQ37 | 3 |  |  |  |  |  | O | O | O | O |  |  |  |  |
| REQ38 | 2 |  |  |  |  |  | O |  |  |  |  |  |  |  |
| REQ61 | 5 | O |  |  |  |  |  |  |  |  |  |  |  |  |
| REQ67 | 4 | O |  |  |  |  |  |  |  |  |  |  |  |  |
| REQ45 | 2 | O |  |  |  |  |  |  |  |  |  |  |  |  |
| Max PW | | 5 | 5 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 5 | 3 | 3 | 5 |
| Total PW | | 11 | 10 | 3 | 3 | 3 | 8 | 3 | 3 | 3 | 9 | 4 | 4 | 10 |

**5.3 SubgroupC(OrderStatus)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Req’t** | **PW** | C-UC1 | C-UC2 | C-UC3 | C-UC4 | C-UC5 | C-UC6 | C-UC7 | C-UC8 | C-UC9 | C-UC10 | C-UC11 |
| REQ52 | 4 |  |  |  |  |  |  |  |  |  |  | O |
| REQ53 | 5 |  |  |  |  |  |  |  |  |  | O |  |
| REQ54 | 5 | O |  |  |  |  |  |  |  |  |  |  |
| REQ55 | 5 |  | O |  |  |  |  |  |  |  |  |  |
| REQ56 | 3 |  |  | O |  |  |  |  |  |  |  |  |
| REQ57 | 5 |  |  |  | O |  |  |  |  |  |  |  |
| REQ58 | 5 |  |  |  |  | O |  |  |  |  |  |  |
| REQ59 | 1 |  |  |  |  |  | O |  |  |  |  |  |
| REQ60 | 3 |  |  |  |  |  |  | O |  |  |  |  |
| REQ61 | 5 |  |  |  |  |  |  | O |  |  |  |  |
| REQ62 | 3 |  |  |  |  |  |  |  | O |  |  |  |
| REQ63 | 2 |  |  |  |  |  |  |  | O |  |  |  |
| REQ64 | 5 |  |  |  |  |  |  |  | O |  |  |  |
| REQ65 | 3 |  |  |  |  |  |  |  |  | O |  |  |
| REQ66 | 1 |  |  |  |  |  |  |  |  | O |  |  |
| REQ67 | 4 |  |  |  |  |  |  |  |  |  |  | O |
| REQ68 | 4 |  |  |  |  |  |  |  | O |  |  |  |
| REQ69 | 2 |  |  |  |  |  |  |  |  |  | O |  |
| Max PW | | 5 | 5 | 3 | 5 | 5 | 2 | 5 | 5 | 3 | 2 | 4 |
| Total PW | | 5 | 5 | 3 | 5 | 5 | 2 | 8 | 14 | 4 | 2 | 4 |