Use Case Description

Table of content

[1. Use case: Level 1 2](#_Toc326308853)

[1.1. Level 2 - System Management 3](#_Toc326308854)

[1.2. Level 2 - Customer Management 4](#_Toc326308855)

[1.3. Level 2 - Category Management 4](#_Toc326308856)

[1.4. Level 2 - Product Management 5](#_Toc326308857)

[1.5. Level 2 - Record Management 5](#_Toc326308858)

[1.6. Level 2 - Retail Stores Management 6](#_Toc326308859)

[1.7. Level 2 – User Computer Management 6](#_Toc326308860)

[2. Use case Descriptions 7](#_Toc326308861)

[2.1. System Management 7](#_Toc326308862)

[2.2. Customer Management 10](#_Toc326308863)

[2.3. Category Management 14](#_Toc326308864)

[2.4. Product Management 15](#_Toc326308865)

[2.5. Record Management 17](#_Toc326308866)

[2.6. Retail Stores Management 19](#_Toc326308867)

[2.7. User Computer Management 21](#_Toc326308868)

# 1. Use case: Level 1



## Level 2 - System Management



## Level 2 - Customer Management



## Level 2 - Category Management



## Level 2 - Product Management



## Level 2 - Record Management



## Level 2 - Retail Stores Management



## Level 2 – User Computer Management



# Use case Descriptions

## System Management

|  |  |
| --- | --- |
| **Use Case Title: Add new user** | **Use Case ID: S01** |
| **General Use Case Description:** This function allows admin create new user. | |
| **Entities Involved: Admin** | |
| **Preconditions:**   1. Admin chooses System Management. 2. Admin chooses “add new user” function, add new user UI ready for admin to interact. | |
| **Primary Use Case Flow of Events:**   1. Admin fill the required information about user. 2. Admin choose user authorization :  * Staff * Cashier * Manager  1. Admin click button Create 2. System show notification that Create a new user successfully 3. System show new created user in user list. 4. End of create new user | |
| **Primary Use Case Post Conditions:**  Successful: show notification that Create a new user successfully, show new created user in user list.  Fail: show notification that Create a new user unsuccessfully. | |
| **Alternate Use Case #1 Flow of Events:** | |
| **Alternate Use Case #1 Post Events:** | |

|  |  |
| --- | --- |
| **Use Case Title: Search/ View user list** | **Use Case ID: S02** |
| **General Use Case Description:** This function allow admin Search and View user list | |
| **Entities Involved: Admin** | |
| **Preconditions:**   1. Admin chooses System Management. 2. User list UI available and ready for admin to interact. | |
| **Primary Use Case Flow of Events:**   1. System show the user list of all user 2. Admin choose search condition:  * Username * User ID * Store name  1. Admin input key word to textbox: 2. System show the user list which satisfy the key word 3. End search/ view user list. | |
| **Primary Use Case Post Conditions:**  Search successful, user list which satisfy the key word show up in UI. | |
| **Alternate Use Case #1 Flow of Events:** | |
| **Alternate Use Case #1 Post Events:** | |

|  |  |
| --- | --- |
| **Use Case Title: View user detail Information** | **Use Case ID: S03** |
| **General Use Case Description:** This function allows admin View user detail information | |
| **Entities Involved: Admin** | |
| **Preconditions:**  Admin is viewing the user list | |
| **Primary Use Case Flow of Events:**   1. Admin double click to user need to view detail information. 2. System shows the user detail information. 3. End of View user detail Information | |
| **Primary Use Case Post Conditions:**  System shows the user detail information. | |
| **Alternate Use Case #1 Flow of Events:** | |
| **Alternate Use Case #1 Post Events:** | |

|  |  |
| --- | --- |
| **Use Case Title: Update User Information, Assign authorize** | **Use Case ID: S04** |
| **General Use Case Description:** This function allows admin Update User Information and Assign authorize | |
| **Entities Involved: Admin** | |
| **Preconditions:**  Admin is viewing user detail information | |
| **Primary Use Case Flow of Events:**   1. Admin changes user authorize if needed. 2. Admin change or update user information. 3. Admin click Update | |
| **Primary Use Case Post Conditions:**  System show notification that update user information successful. | |
| **Alternate Use Case #1 Flow of Events:** | |
| **Alternate Use Case #1 Post Events:** | |

## Customer Management

|  |  |
| --- | --- |
| **Use Case Title: Add new customer** | **Use Case ID: S2.1** |
| **General Use Case Description:** This function allows staff create new customer | |
| **Entities Involved:** Staff | |
| **Preconditions:**   1. Staff choose Customer management 2. Staff Add new customer function, Add new customer UI ready for staff to interact | |
| **Primary Use Case Flow of Events:**   1. Staff fill required information of customer:  * Name (required) * Birthday (required) * Address (optional) * ID number (required) * Gender (required)  1. Staff click button Create 2. System shows the notification that Add new customer successfully 3. System show new created customer in user list. 4. End of create new customer | |
| **Primary Use Case Post Conditions:**  Successful: Add new customer successfully. System show new created customer in user list.  Fail: show notification that Create a new customer unsuccessfully. | |
| **Alternate Use Case #1 Flow of Events:**   1. If required information was blank 2. System shows the notification that some information needs to be filled to create new customer. 3. Staff fill required information 4. Staff click create again. 5. Continue step 3 of primary flow   **Alternate Use Case #2 Flow of Events:**   1. If ID number was duplicated. 2. System shows the notification that ID number was duplicated 3. Staff check the ID number 4. Staff click create again. 5. Continue step 3 of primary flow | |
| **Alternate Use Case #1 Post Events:** | |

|  |  |
| --- | --- |
| **Use Case Title: Search/ View customer list** | **Use Case ID: S02.2** |
| **General Use Case Description:** This function allows staff create new customer Search/ View customer list | |
| **Entities Involved: Staff** | |
| **Preconditions:**   1. Staff chooses Customer Management. 2. Customer list UI available and ready for admin to interact. | |
| **Primary Use Case Flow of Events:**   1. System show the customer list 2. Staff choose search condition:  * Customer name * Customer ID number * Store name  1. Staff input key word to textbox: 2. System show the user list which satisfy the key word 3. End search/ view user list. | |
| **Primary Use Case Post Conditions:**  Search successful, customer list which satisfy the key word show up in UI. | |
| **Alternate Use Case #1 Flow of Events:** | |
| **Alternate Use Case #1 Post Events:** | |

## Category Management

|  |  |
| --- | --- |
| **Use Case Title: Add new category** | **Use Case ID: S2.31** |
| **General Use Case Description:** This function allows Staff create new category. | |
| **Entities Involved: Staff** | |
| **Preconditions:**   1. Staff chooses category management 2. Create new category UI available and ready for staff to interact. | |
| **Primary Use Case Flow of Events:**   1. Staff input category name 2. Staff click Create button 3. System show the notification that create new category successfully 4. End of create new category | |
| **Primary Use Case Post Conditions:**  Successful: create new category successfully, new category appear in category list  Fail: System show the notification that create new category unsuccessfully | |
| **Alternate Use Case #1 Flow of Events:**   1. Category name was duplicated 2. System show the notification that Category name was duplicated 3. Staff input another category name 4. Staff click Create 5. Continue step 3 in primary flow | |
| **Alternate Use Case #1 Post Events:** | |

## Product Management

|  |  |
| --- | --- |
| **Use Case Title: Add new product** | **Use Case ID: S2.31** |
| **General Use Case Description:** This function allows Staff create new product. | |
| **Entities Involved: Staff** | |
| **Preconditions:**   1. Staff chooses product management 2. Create new product UI available and ready for staff to interact. | |
| **Primary Use Case Flow of Events:**   1. Staff input product name 2. Staff chooses category 3. Staff input product price 4. Staff click Create button 5. System show the notification that create new product successfully 6. End of create new product | |
| **Primary Use Case Post Conditions:**  Successful: create new product successfully, new product appear in product list.  Fail: System show the notification that create new product unsuccessfully | |
| **Alternate Use Case #1 Flow of Events:**   1. Product name was duplicated 2. System show the notification that Product name was duplicated 3. Staff input another product name 4. Staff click Create 5. Continue step 5 in primary flow | |
| **Alternate Use Case #1 Post Events:** | |

## Record Management

|  |  |
| --- | --- |
| **Use Case Title: Add New Record** | **Use Case ID:** |
| **General Use Case Description:** This use case helps cashiers who work at retail stores make bill-paying for customers. Cashier can use barcode reader or directly enter the product code and product number by keyboard, then use the payment function to save to database and printed out bills for customers. | |
| **Entities Involved: Cashier** | |
| **Preconditions:**   1. User is assigned authorized to use this function 2. User has chosen Record Management function 3. The system is available. | |
| **Primary Use Case Flow of Events:**   1. User choose “Add new bill” button 2. Program displays Add new store interface 3. User fills in Bar Code text field 4. User choose “Add” button 5. The system load information from database to the “Product List” table on the interface. 6. User fills in Quantity text field 7. User scan or fill in customer Loyalty card ID 8. The system load information from database and show customer name with their point on the interface 9. If customer want to use their point to pay bill, User choose “Type to pay bill” 10. There are three ways to pay bill: 11. Pay by cash: User does not fill in “Point” text field. The system auto shows all the money customer has to pay in “Cash” text field. 12. Pay by point: User fills in “Point” text field if user has enough point to pay that bill, the system auto shows “0 VND” in “Cash” text field. 13. Pay by point and cash: User fills in “Point” text field; the system shows money that customer still has to pay with that bill in “Cash” text field. 14. User choose “Pay by Cash” 15. User choose Pay bill button 16. The system stores the bill’s information that has just been created and prints the bill. Then the program returns to bill management interface 17. End Use-case | |
| **Primary Use Case Post Conditions:**  Successful: Create new successful. The system stores the bill’s information that has just been created and prints the bill. Then the program returns to bill management interface  Fail: Can’t print the bill | |
| **Alternate Use Case #1 Flow of Events:**   1. The bar code is invalid (starting from step 5 of the main flow) 2. Program loads information from database and shows message “The bar code is invalid” 3. User clicks "OK" to confirm 4. Program allows user to retype the bar code 5. End Use-case | |
| **Alternate Use Case #1 Post Events:** | |

## Retail Stores Management

|  |  |
| --- | --- |
| **Use Case Title:** Add New Store | **Use Case ID:** |
| **General Use Case Description:** This use case help the manager add new sale store. This function can also split products of retail stores that are consumed, as well as regulate the product price. | |
| **Entities Involved:** Manager | |
| **Preconditions:**   1. User is assigned authorized to use this function 2. User has chosen Retail Stores Management function 3. The system is available | |
| **Primary Use Case Flow of Events:**   1. User choose “Add new store” button 2. Program displays Add new store interface 3. User fills in Store name text field 4. User fills in Address text field 5. User chooses a Category 6. Program displays Products which has in that Category 7. User chooses a Product 8. Program add the product which user just has chosen to the Product List with the standard Price 9. User can edit the Price base on how much the store want to sell that product. 10. User clicks "OK". 11. Program shows message "Are you sure you want to create new retail store’s information as above?" 12. User clicks "OK" to confirm 13. The system stores the retail store’s information that has just been created and returned to retail store management interface 14. End Use-case | |
| **Primary Use Case Post Conditions:**  Successful: Create new successful. The system stores the retail store’s information that has just been created and returned to retail store management interface  Fail: Failed to create new | |
| **Alternate Use Case #1 Flow of Events:**   1. There is a store with already name (starting from step 11 of the main flow) 2. Program shows message "Are you sure you want to create new retail store’s information as above?" 3. User clicks "OK" to confirm 4. Program check in database and found that it has a store with already name 5. Program display warning dialog “There is a store with already name. Please fill in again” 6. End Use-case | |
| **Alternate Use Case #1 Post Events:** | |

## User Computer Management

|  |  |
| --- | --- |
| **Use Case Title:** | **Use Case ID:** |
| **General Use Case Description:** | |
| **Entities Involved:** | |
| **Preconditions:** | |
| **Primary Use Case Flow of Events:** | |
| **Primary Use Case Post Conditions:**  Successful:  Fail: | |
| **Alternate Use Case #1 Flow of Events:** | |
| **Alternate Use Case #1 Post Events:** | |