Version <1.0>

[Note: The following template is provided for use with the Rational Unified Process. Text enclosed in square brackets and displayed in blue italics (style=InfoBlue) is included to provide guidance to the author and should be deleted before publishing the document. A paragraph entered following this style will automatically be set to normal (style=Body Text).]

[To customize automatic fields in Microsoft Word (which display a gray background when selected), select File>Properties and replace the Title, Subject and Company fields with the appropriate information for this document. After closing the dialog, automatic fields may be updated throughout the document by selecting Edit>Select All (or Ctrl-A) and pressing F9, or simply click on the field and press F9. This must be done separately for Headers and Footers. Alt-F9 will toggle between displaying the field names and the field contents. See Word help for more information on working with fields.]

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 14/11/2019 | 1.0 | Use-case specification feature: Search product, Add a bill, Add a product to bill, Export a bill, Search a member, Statistics by product, Statistics by product type, Statistics product many buyer, Add a staff, Delete a staff, Search a staff. | Nguyễn Lý Nhật Phương |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Use-case Model 4

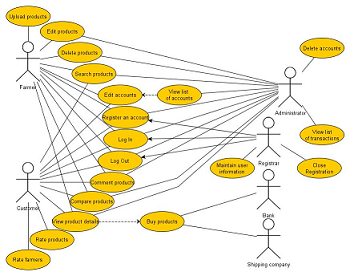
2. Use-case Specifications 4

2.1 Use-case: Add a product to cart 4

2.2 Use-case: Create a new account 5

# Use-case Model

[*Put an image of the use-case diagram modeling all use-cases in this section*]



# Use-case Specifications

## Use-case: Search product in the store

|  |  |
| --- | --- |
| Use case Name | Search product in the store. |
| Brief description | This use case describes how the store employees can see the information of a product in the store. |
| Actors | Store employees. |
| Basic Flow | 1. At the main menu, user clicks on ‘Check product in store’ button to open search product window. 2. System displays the search product window. 3. User enters the ID of the product on the ‘Search’ field. 4. User clicks on the ‘Search’ button to start searching a product. 5. System displays the products found. |
| Alternative Flows | **Alternative flow 1: User cannot find products searched**   1. From #3 of the basic flow, user enters another term. 2. Continue step #4 in the basic flow.   **Alternative flow 2: User leaves the ‘Search’ field blank**   1. System displays information of all the product in the store. |
| Pre-conditions | User goes to main menu. |
| Post-conditions | The user sucessfully sees information of the searched product or sees information of all the product in the store. |

## Use-case: Search product in the storage

|  |  |
| --- | --- |
| Use case Name | Search product in the storage. |
| Brief description | This use case describes how the storage staffs can see the information of a product in the storage. |
| Actors | Storage employees. |
| Basic Flow | 1. At the main menu, user clicks on ‘Check inventory in storage’ button to open search product window. 2. System displays the search product window. 3. User enters the ID of the product on the ‘Search’ field. 4. User clicks on the ‘Search’ button to start searching a product. 5. System displays the products found. |
| Alternative Flows | **Alternative flow 1: User cannot find products searched**   1. From #3 of the basic flow, user enters another term. 2. Continue step #4 in the basic flow.   **Alternative flow 2: User leaves the ‘Search’ field blank**   1. System displays information of all the product in the storage. |
| Pre-conditions | User goes to main menu. |
| Post-conditions | The user sucessfully sees information of the searched product or sees information of all the product in the store. |

## Use-case: Search product in all the system

|  |  |
| --- | --- |
| Use case Name | Search product in the storage. |
| Brief description | This use case describes how the manager can see the information of a product in both the store and the storage. |
| Actors | Store manager. |
| Basic Flow | 1. At the main menu, user clicks on ‘Check product’ button to open search product window. 2. System displays the search product window . 3. User enters the ID of the product on the ‘Search’ field. 4. User can check in ‘In store’ button to search for product in the store, check in ‘In storage’ button to search for product in the storage. 5. User clicks on the ‘Search’ button to start searching a product. 6. System displays the product found. |
| Alternative Flows | **Alternative flow 1: User cannot find products searched**   1. From #3 of the basic flow, user enters another term. 2. Continue step #4 in the basic flow.   **Alternative flow 2: User leaves the ‘Search’ field blank**   1. System displays information of all the product.   **Alternative flow 3: User doesn’t check ‘In store’ or ‘In storage’**   1. System displays information of the product with quantity is sum of the product quantity in the store and in the storage. |
| Pre-conditions | User goes to main menu. |
| Post-conditions | The user successfully sees information of the searched product or sees information of all the product in the store or in the storage or both. |

## Use-case: Add a bill

|  |  |
| --- | --- |
| Use case Name | Add a bill. |
| Brief description | This use case describes how the store employees and manager can add it to the database. |
| Actors | Store employees, store manager. |
| Basic Flow | 1. At the main menu, the user clicks ‘Check bill’ button. 2. System displays the bill window. 3. User clicks ‘Add a bill’ button. 4. System displays the add bill window. 5. User adds all product the customer bought to the bill. 6. User clicks ‘Add bill’ button. 7. System adds the bill to the database. 8. System displays all the information of the bill added. |
| Alternative Flows | **Alternative flow 1: The customer is a member**   1. After step #4 of the basic flow, user enters the member id of the customer. 2. Continue step #5 in the basic flow.   **Alternative flow 1: Add bill failed**   1. Do again from step #5 of the basic flow. |
| Pre-conditions | User goes to bill menu. |
| Post-conditions | The user successfully adds the bill to the database. |

## Use-case: Add a product to bill

|  |  |
| --- | --- |
| Use case Name | Add an item to bill. |
| Brief description | This use case describes how the store employees and manager make a bill. |
| Actors | Store employees, store manager. |
| Basic Flow | 1. At the add bill window, the user clicks on the ‘Add product’ button. 2. System displays the add product window. 3. User writes the product id on the ‘Product ID’ field. 4. User writes the quantity that customer buy of the product. 5. User clicks ‘Add’ button. 6. System adds a new product to bill. 7. System displays the quantity of all products in the bill. |
| Alternative Flows | **Alternative flow 1: User leaves the ‘Quantity’ field blank**   1. System default the quantity is one.   **Alternative flow 2: Bill has already had the product**   1. From #6 of the basic flow, system increases the quantity of the product. 2. Continue step #7 in the basic flow. |
| Pre-conditions | User goes to the add bill menu. |
| Post-conditions | The user successfully adds new product to the bill or increases quantity of the existing product in the bill. |

## Use-case: Export a bill

|  |  |
| --- | --- |
| Use case Name | Export a bill. |
| Brief description | This use case describes how the store employees and manager can export a bill information. |
| Actors | Store employees, store manager. |
| Basic Flow | 1. At the main menu, the user clicks ‘Check bill’ button. 2. System displays the bill window. 3. User searchs and selects the bill to export. 4. User clicks ‘Export a bill’ button. 5. System displays the window to choose the output directory and file type. 6. User select output directory and file type. 7. User clicks ‘OK’ button. 8. System displays the confirm window. 9. User clicks ‘Yes’ button. 10. System export the bill to the selected directory. 11. System displays all information of the exported bill. |
| Alternative Flows | **Alternative flow 1: User cannot find the bill**   1. From #2 of the basic flow, user searchs and selects another bill. 2. Continue step #7 in the basic flow. |
| Pre-conditions | User goes to bill window. |
| Post-conditions | The user sucessfully exports the bill. |

## Use-case: Search a staff

|  |  |
| --- | --- |
| Use case Name | Search a staff. |
| Brief description | This use case describes how the manager can search for a staff who work for the store. |
| Actors | Store manager. |
| Basic Flow | 1. At the main menu, user clicks on ‘Staffs’ button to open staffs window. 2. System displays the staffs window . 3. User enters the ID of the staff on the ‘Search’ field. 4. User clicks on the ‘Search’ button to start searching the staff. 5. System displays the staff found. |
| Alternative Flows | **Alternative flow 1: User cannot find the staff searched**   1. From #3 of the basic flow, user enters another term. 2. Continue step #4 in the basic flow. |
| Pre-conditions | User goes to main menu. |
| Post-conditions | The user sucessfully sees information of the searched staff. |

## Use-case: Add a staff

|  |  |
| --- | --- |
| Use case Name | Add a staff. |
| Brief description | This use case describes how the store manager can add a new staff or employee when a person apply for the job. |
| Actors | Store manager. |
| Basic Flow | 1. At the main menu, the user clicks the ‘Staffs’ button. 2. System displays the staffs window. 3. User clicks ‘Add a staff’ button. 4. User writes down the information of the new staff. 5. User clicks ‘Add’ button. 6. System adds the new staff to the database. 7. System displays the information of the new added staff. |
| Alternative Flows | **Alternative flow 1: Add a new staff failed**   1. Do again from step #4 of the basic flow. |
| Pre-conditions | User goes to main menu. |
| Post-conditions | The user successfully adds a new staff to the database. |

## Use-case: Delete a staff

|  |  |
| --- | --- |
| Use case Name | Delete a staff. |
| Brief description | This use case describes how the store manager can remove a staff or employee when the person quit the job. |
| Actors | Store manager. |
| Basic Flow | 1. At the main menu, the user clicks the ‘Staffs’ button. 2. System displays the staffs window. 3. User searchs for the staff want to delete. 4. User clicks on the staff. 5. User clicks ‘Delete a staff’ button. 6. System displays the confirm window. 7. User clicks ‘Yes’ button. 8. System deletes the staff from the database. 9. System displays information of all people who work for the store. |
| Alternative Flows | **Alternative flow 1: Delete staff failed**   1. Do again from step #3 of the basic flow. |
| Pre-conditions | User goes to main menu. |
| Post-conditions | The user successfully deletes a staff from the database. |

## Use-case: Search a member

|  |  |
| --- | --- |
| Use case Name | Search a member. |
| Brief description | This use case describes how the store employees and manager can search for a member information. |
| Actors | Store employees, store manager. |
| Basic Flow | 1. At the main, the user click ‘Check member’ button. 2. System displays member window. 3. User enters the customer ID or phone number on the ‘Search’ field. 4. Users clicks on ‘Search’ button to start searching a customer. 5. System displays the customer found. |
| Alternative Flows | **Alternative flow 1: User cannot find customer searched**   1. From #3 of the basic flow, user enters another term 2. Continue step #4 in the basic flow |
| Pre-conditions | User goes to main menu. |
| Post-conditions | The user successfully sees the information of the member seached. |

## Use-case: Statistics by product

|  |  |
| --- | --- |
| Use case Name | Statistics by product. |
| Brief description | This use case describes how the store manager can make statistics of sale by product. |
| Actors | Store manager. |
| Basic Flow | 1. At the homepage, the user clicks ‘Statistic’ button. 2. System displays the statistic window. 3. User selects the time want to statistic. 4. User clicks ‘Statistic by product’ button. 5. System makes statistic by product in the selected time. 6. System displays the statistic result. |
| Alternative Flows | **Alternative flow 1: User doesn’t select the time**   1. From #5 of the basic flow, system make statistic by product from the beginning of the data to the current time. 2. Continue step #6 in the basic flow.   **Alternative flow 2: User just selects year and month**   1. From #5 of the basic flow, system make statistic by product in the selected month of the selected year. 2. Continue step #6 in the basic flow.   **Alternative flow 3: User just selects year**   1. From #5 of the basic flow, system make statistic by product in all the month of the selected year. 2. Continue step #6 in the basic flow. |
| Pre-conditions | User goes to main menu. |
| Post-conditions | The user sucessfully sees the statistic result. |

## Use-case: Statistics by product type

|  |  |
| --- | --- |
| Use case Name | Statistics by product type. |
| Brief description | This use case describes how the store manager can make statistics of sale by product type. |
| Actors | Store manager. |
| Basic Flow | 1. At the homepage, the user clicks ‘Statistic’ button. 2. System displays the statistic window. 3. User selects the time want to statistic. 4. User clicks ‘Statistic by product type’ button. 5. System makes statistic by product type in the selected time. 6. System displays the statistic result. |
| Alternative Flows | **Alternative flow 1: User doesn’t select the time**   1. From #5 of the basic flow, system make statistic by product type from the beginning of the data to the current time. 2. Continue step #6 in the basic flow.   **Alternative flow 2: User just selects year and month**   1. From #5 of the basic flow, system make statistic by product type in the selected month of the selected year. 2. Continue step #6 in the basic flow.   **Alternative flow 3: User just selects year**   1. From #5 of the basic flow, system make statistic by product type in all the month of the selected year. 2. Continue step #6 in the basic flow. |
| Pre-conditions | User goes to main menu. |
| Post-conditions | The user sucessfully sees the statistic result. |

## Use-case: Statistics product many buyer

|  |  |
| --- | --- |
| Use case Name | Statistics product many buyer. |
| Brief description | This use case describes how the store manager can make statistics for best sell products. |
| Actors | Store manager. |
| Basic Flow | 1. At the homepage, the user clicks ‘Statistic’ button. 2. System displays the statistic window. 3. User selects the time want to statistic. 4. User clicks ‘Statistic product many buyer’ button. 5. System makes statistic for best sell products of the store in the selected time. 6. System displays the statistic result. |
| Alternative Flows | **Alternative flow 1: User doesn’t select the time**   1. From #5 of the basic flow, system make statistic for best sell products of the store from the beginning of the data to the current time. 2. Continue step #6 in the basic flow.   **Alternative flow 2: User just selects year and month**   1. From #5 of the basic flow, system make statistic for best sell products of the store in the selected month of the selected year. 2. Continue step #6 in the basic flow.   **Alternative flow 3: User just selects year**   1. From #5 of the basic flow, system make statistic for best sell products of the store in all the month of the selected year. 2. Continue step #6 in the basic flow. |
| Pre-conditions | User goes to main menu. |
| Post-conditions | The user sucessfully sees the statistic result. |