HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

School of Information and communications technology

Software Requirement Specification

Version 2.0

eStoreManager

Store management solution

Subject: ITSS Software Development

Group 01

Nguyễn Sỹ An

Nguyễn Việt Anh

Theo Mercurio

Hanoi, 12/2018

1. Table of contents

1 Introduction

1.1 Objective

This document presents the detailed description for eStore Manager – a store management system user group and their usable function at run time. This document also describes the objectives and features of the system, interfaces and constraints of the system in response to external action.

This document is for stakeholders and related software developers.

We are trying to implement a desktop application, which:

+ Have multiple language. Firstly we support English and French.

+ Have clean structure for ease in development and maintainent.

+ Have a good UI design using EJS Templating.

1.2 Scope

In reality, any software needs to have the function of managing users or group of users, and dynamically granting permissions for usage of certain functions in the system.

This software will be built as a management tool for a small store, with only 1 branch.

The software's goal includes creating following components:

- Interface to make bills and sell products.

- Interface to manage suppliers, import products.

- Interface to manage employees.

- Interface to manage customer list.

- Interface to manage store finance.

Each function may be utilised by many roles of users in: Admin, cashier, warehouse manager.

1.3 Glossary

- ***Admin:*** *administrator in this software often is the store owner. This user can interact with the system in full permission.*

- ***Cashier:*** *This user is a role of shop keeper: make bills and sell products. Cashier also can have access to history of bills, remove a bill in the past (to modify some information in the bill).*

- ***Warehouse:*** *is a place where goods may be stored before their export or distribution for sale.*

- ***Warehouse manager:*** *is a user who can manage suppliers, add import bills and be responsible to manage goods in warehouse.*

- ***Supplier:*** *a vendor who provide product for the store at whole sale price. The store imports goods there to sell to to their customers.*

1.4 References

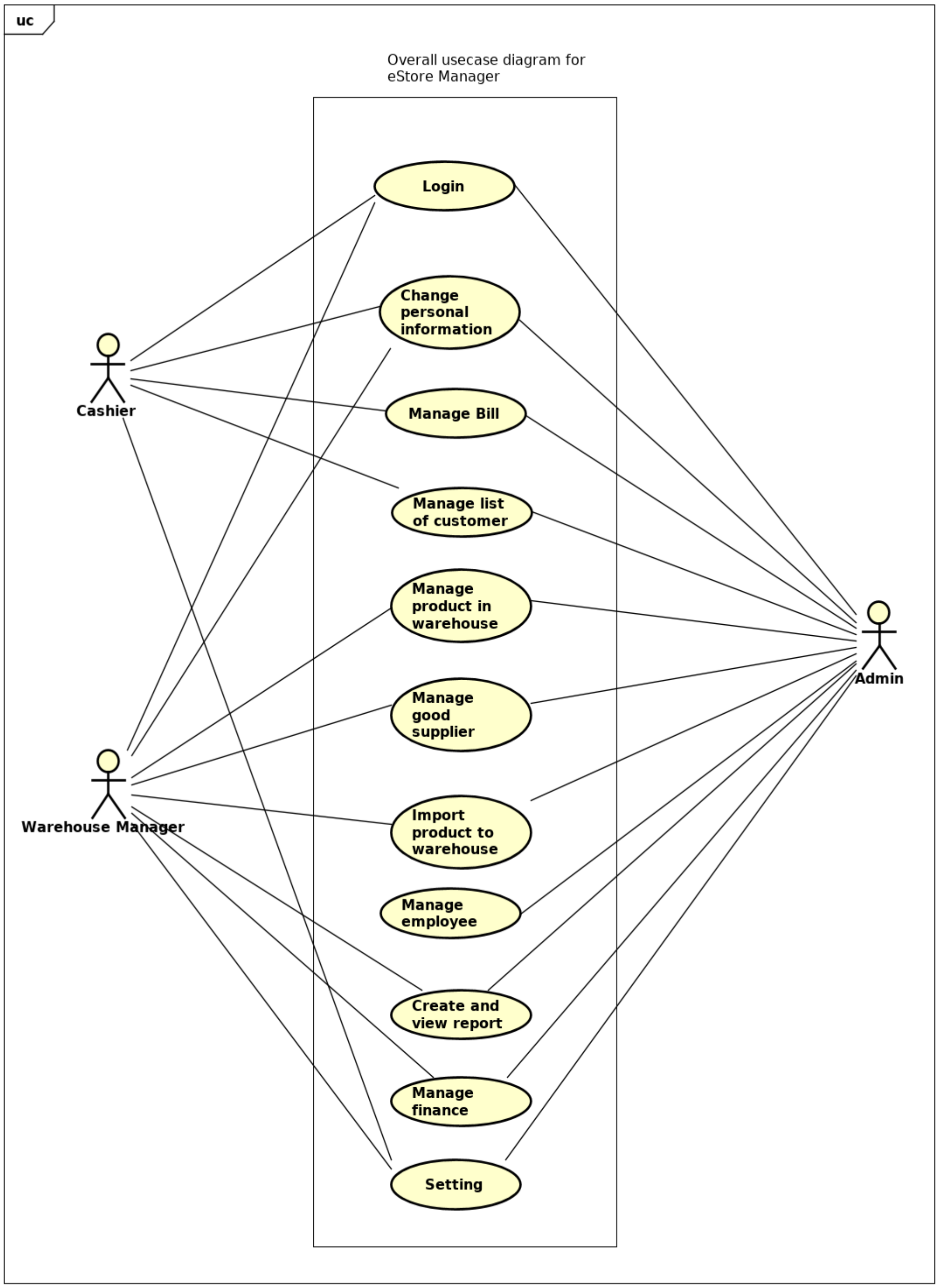
- *An online store management system: Kiot Viet: https://kiotviet.vn*

2 Overall Description

2.1 Actors

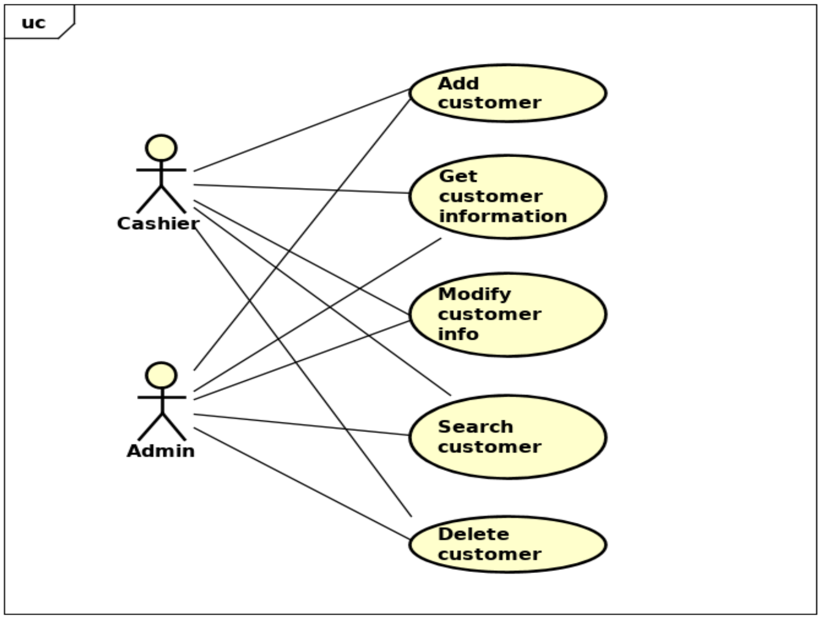
There are 3 main actors in the system: *Admin*, *Cashier*, *Warehouse manager*.

2.2 Overall use case diagram

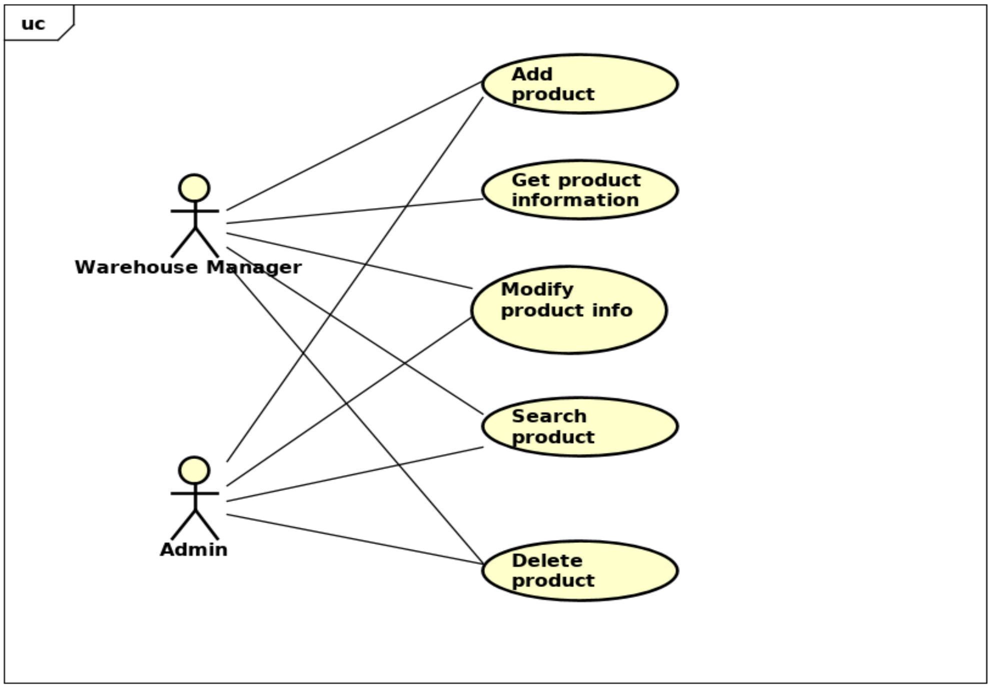


2.3 *Decomposition of Use cases*

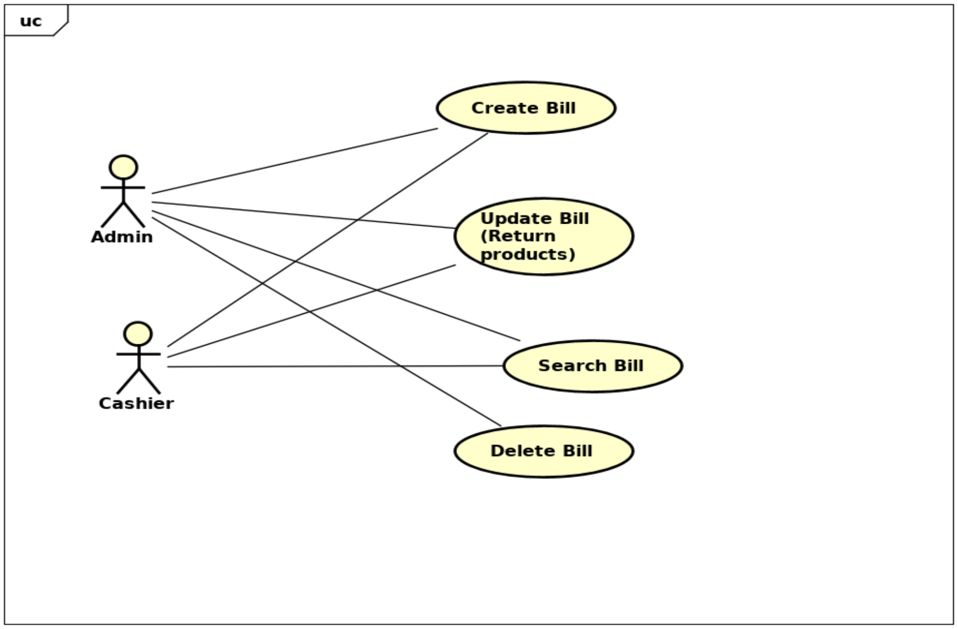
2.3.1 Use case diagram for “Manage customer list”



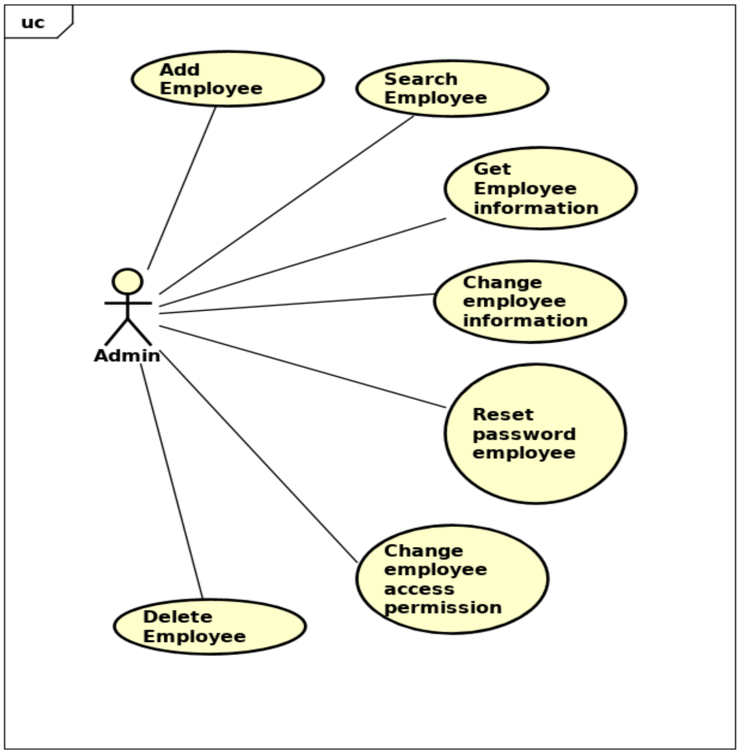
2.3.2 Use case diagram for “Manage products”



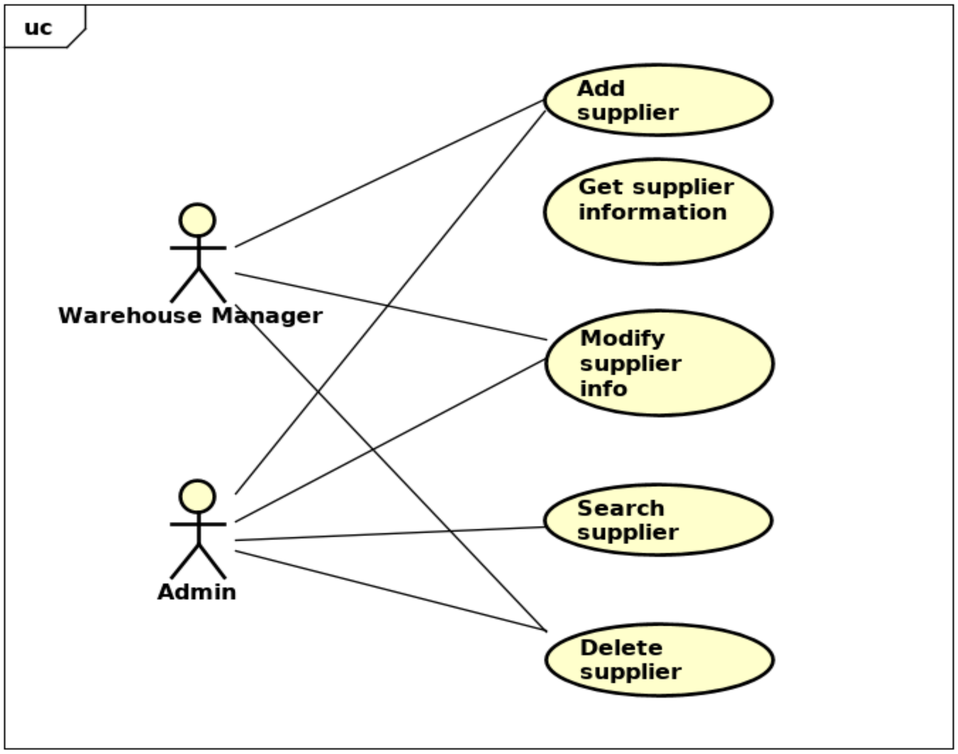
2.3.3 Use case diagram for “Manage Bills”



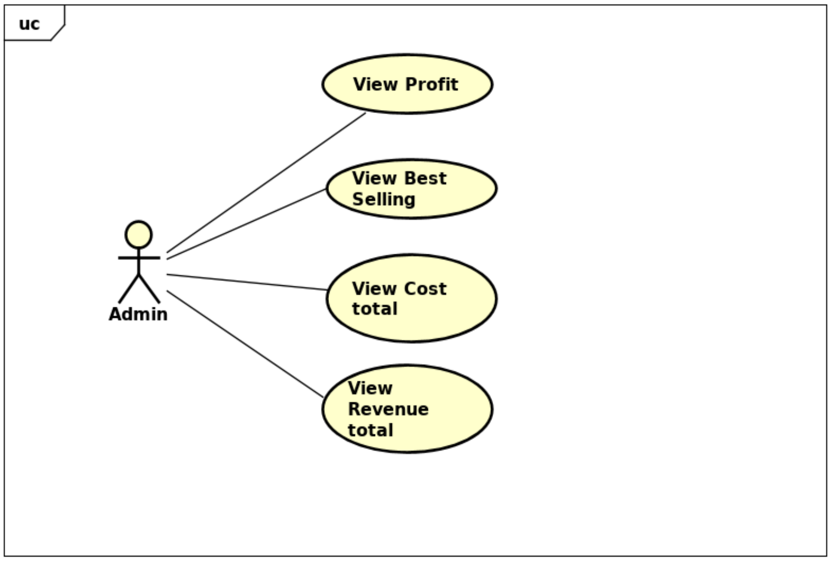
2.3.4 Use case diagram for “Manage Employees”



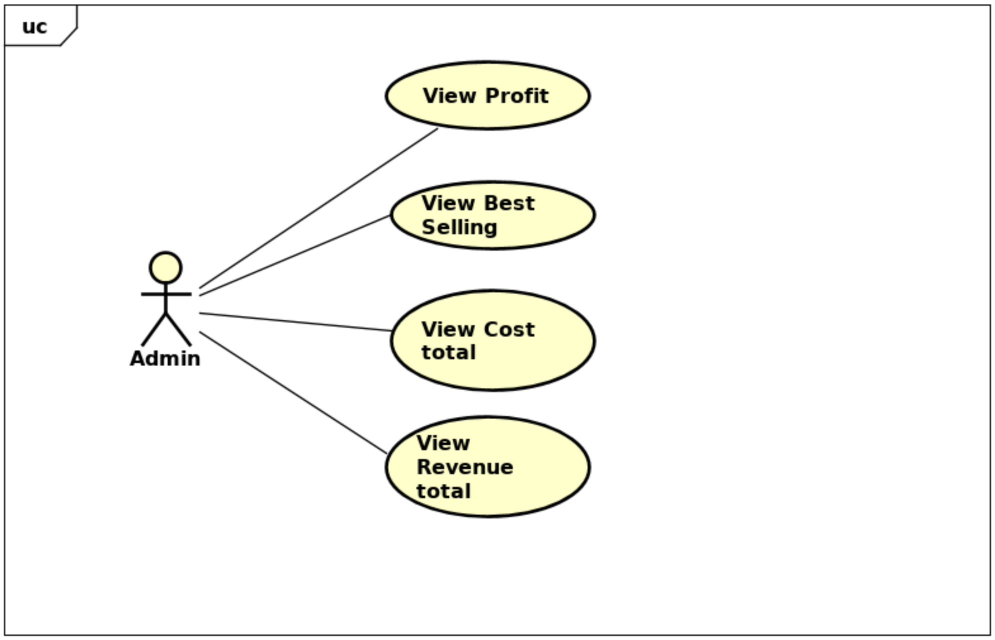
2.3.5 Use case diagram for “Manage suppliers”



2.3.6 Use case diagram for “Manage finance”

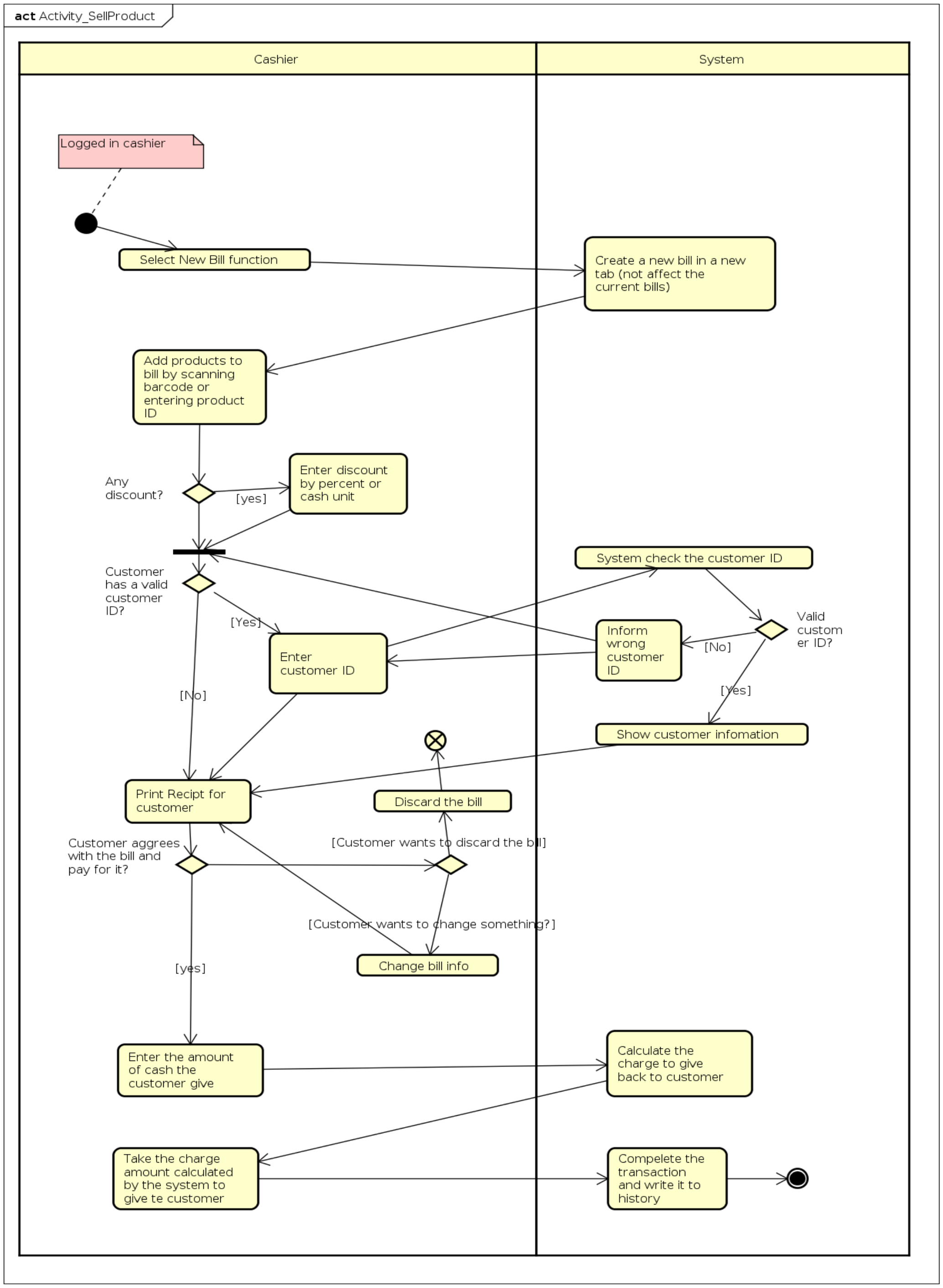


2.3.7 Use case diagram for “Create reports”

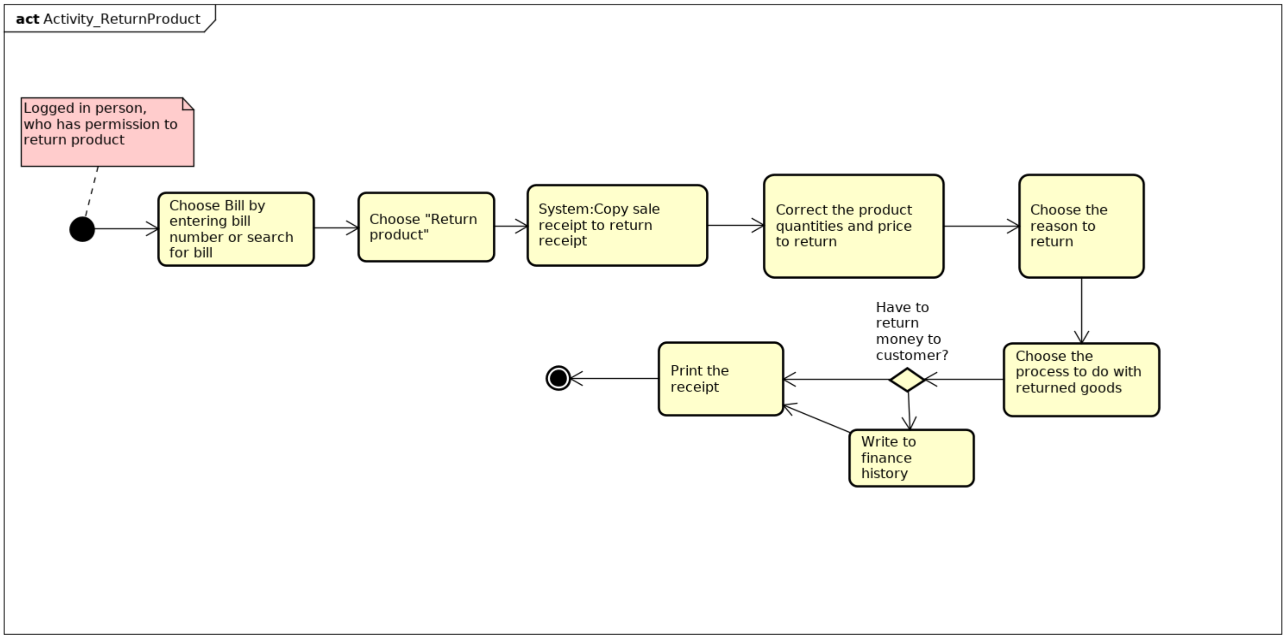


2.4 *Business processes*

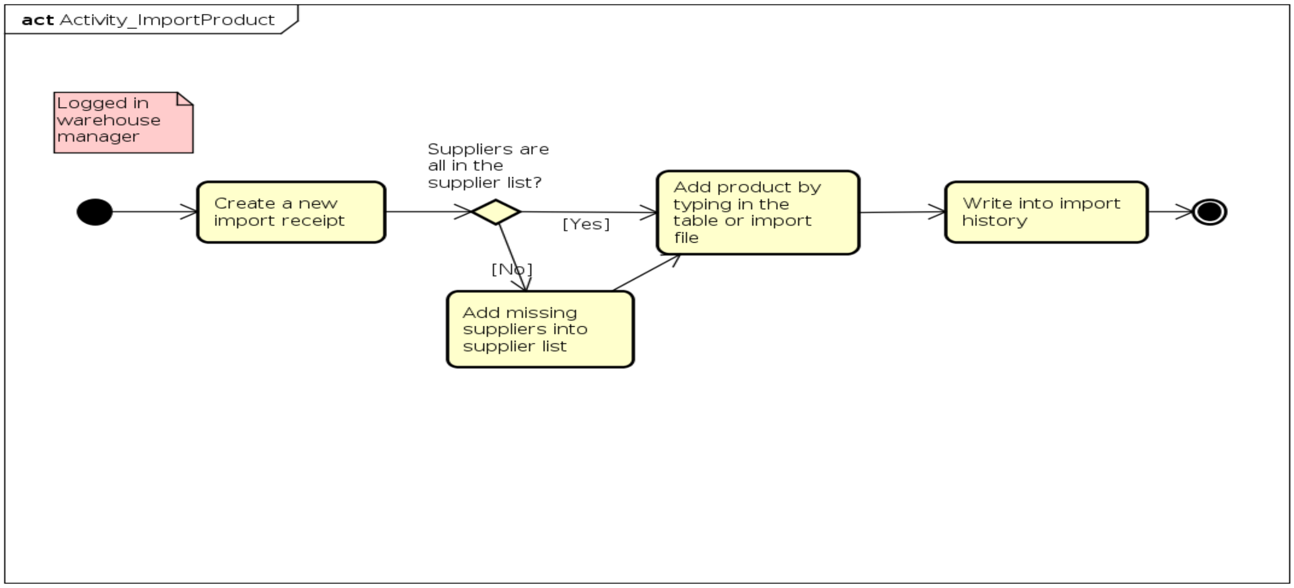
2.4.1 **Process of add bill**


act Activity_SellProduct
Cashier
System
Logged in cashier
Select New Bill function
Create a new bill in a new tab (not affect the current bills)
Add products to bill by scanning barcode or entering product ID
Any discount?
[yes]
Enter discount by percent or cash unit
Customer has a valid customer ID?
[Yes]
Enter customer ID
System check the customer ID
Valid custom er ID?
[No]
Inform wrong customer ID
[Yes]
Show customer infomation
[No]
Print Recipt for customer
Discard the bill
[Customer wants to discard the bill]
Customer aggrees with the bill and pay for it?
[Customer wants to change something?]
Change bill info
[yes]
Enter the amount of cash the customer give
Calculate the charge to give back to customer
Take the charge amount calculated by the system to give te customer
Compelete the transaction and write it to history

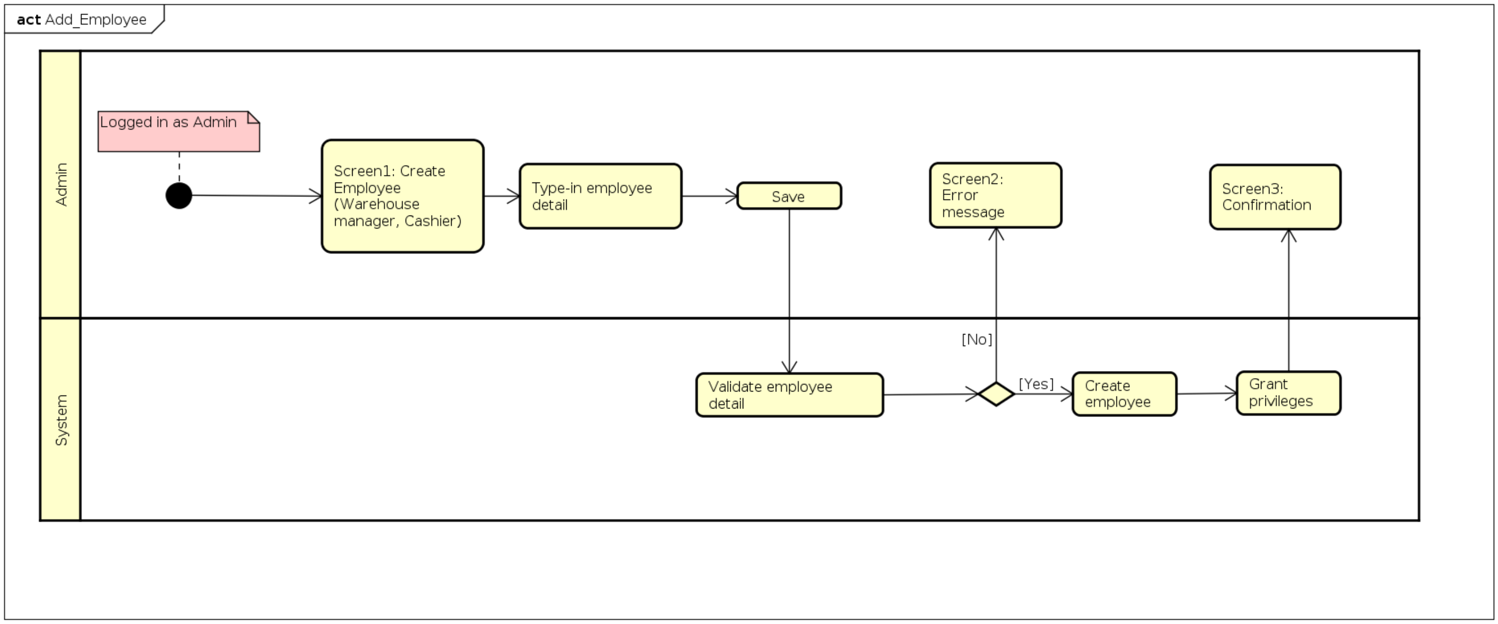
2.4.2 Process of return products


act Activity_ReturnProduct
Logged in person, who has permission to return product
Choose Bill by entering bill number or search for bill
Choose "Return product"
System:Copy sale receipt to return receipt
Correct the product quantities and price to return
Choose the reason to return
Choose the process to do with returned goods
Have to return money to customer?
Write to finance history
Print the receipt

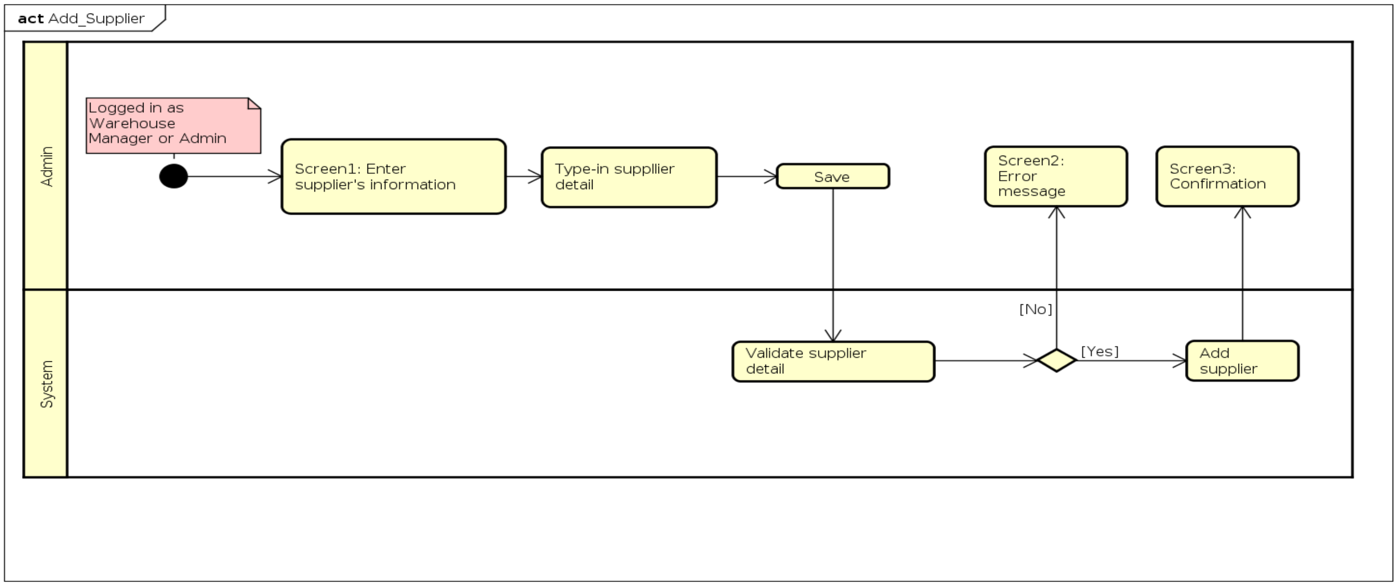
2.4.3 Process of import products


act Activity_ImportProduct
Logged in warehouse manager
Create a new import receipt
Suppliers are all in the supplier list?
[Yes]
[No]
Add missing suppliers into supplier list
Add product by typing in the table or import file
Write into import history

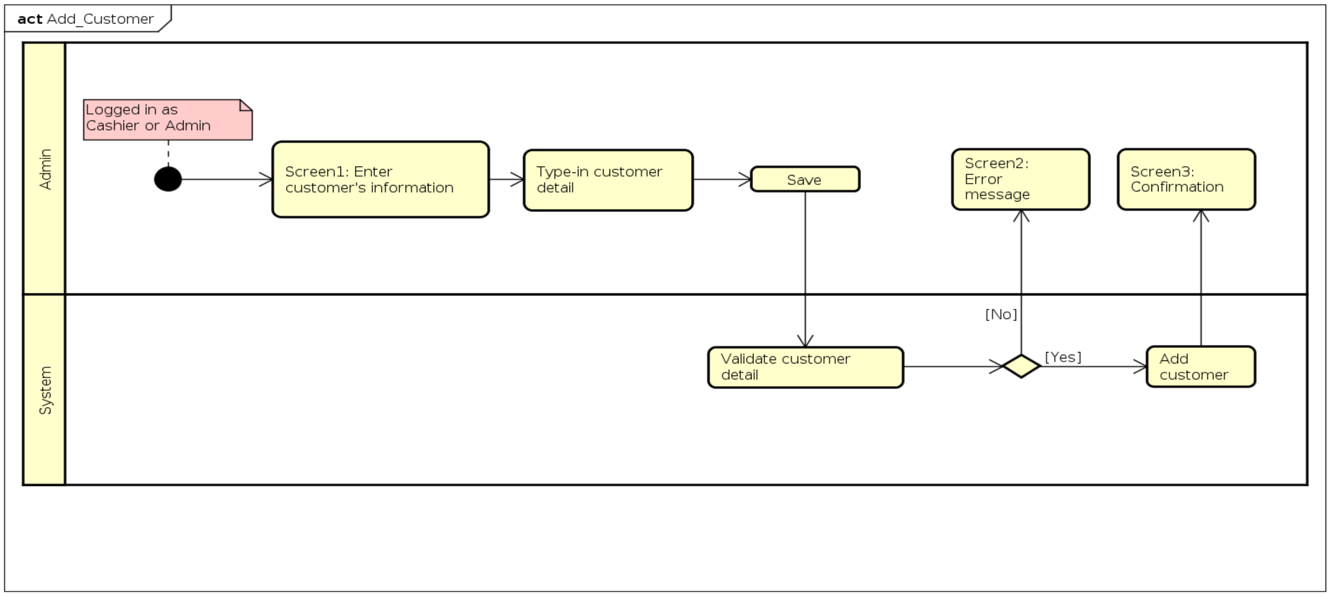
2.4.4 Process of add employee



2.4.5 Process of add supplier



2.4.6 Process of add customer



3 Specific requirements

Details of the use cases given in following sections are specified below.

3.1 *Specification of Use case UC001 - “Log in”*

<table>
<tr><td><b>Use case code</b></td><td>UC001</td><td><b>Use case name</b></td><td>Sell Product</td></tr>
<tr><td><b>Actor</b></td><td colspan="3">User (Cashier, Warehouse Manager, Admin)</td></tr>
<tr><td><b>Pre-condition</b></td><td colspan="3">None</td></tr>
<tr><td><b>Main flow of events (Success)</b></td><td colspan="3">
<table>
<tr><th>No</th><th>Actor</th><th>Action</th></tr>
<tr><td>1</td><td>User</td><td>Select Log in function</td></tr>
<tr><td>2</td><td>System</td><td>Display log in interface</td></tr>
<tr><td>3</td><td>User</td><td>Input user name and passwords</td></tr>
<tr><td>4</td><td>User</td><td>Request log in</td></tr>
<tr><td>5</td><td>System</td><td>Check whether user had input the required field</td></tr>
<tr><td>6</td><td>System</td><td>Validate the user name and password</td></tr>
<tr><td>7</td><td>System</td><td>Log user into the system</td></tr>
<tr><td>8</td><td>Sytem</td><td>Display the Main Form and the use case end</td></tr>
</table>
</td></tr>
<tr><td><b>Alternative flow of events</b></td><td colspan="3">
<table>
<tr><th>No</th><th>Actor</th><th>Action</th></tr>
<tr><td>6a</td><td>Cashier</td><td>Output error: need to put into the missing required field if had not done so</td></tr>
<tr><td>7a</td><td>System</td><td>Output error: If the user name and password is invalid, require user enter user name and password again</td></tr>
</table>
</td></tr>
<tr><td><b>Post-condition</b></td><td colspan="3">None</td></tr>
</table>

3.2 Specification of Use case UC002- “Change password”

Use case code	UC002	**Use case name**	Change password
Actor	User (Cashier, Warehouse Manager, Admin)		
Pre-condition	User logged in to system		
Main flow of events (Success)	**No**	**Actor**	**Action**
	1	User	Select change password function
	2	System	Display change password interface, require confirm old password and new password
	3	Cashier	Enter old password and new password
	4	Cashier	Check whether user had input the required field
	5	Cashier	Check the new password is secure enough or not
	6	Cashier	Check the old password is right or not
	7	System	Display change password successfully, save new password
Alternative flow of events	**No**	**Actor**	**Action**
	5a.	Cashier	Output error: need to put into the missing required field if had not done so
	6a.	System	Output error: the new password is not secure enough, back to step confirm password
	7a.	System	Output error: the old password is not right, back to step confirm password
Post-condition	None		

3.3 Specification of Use case UC003 - “Add Employee”

<table>
<tr><td><b>Use case code</b></td><td>UC003</td><td><b>Use case name</b></td><td>Add Employee</td></tr>
<tr><td><b>Actor</b></td><td colspan="3">Admin</td></tr>
<tr><td><b>Pre-condition</b></td><td colspan="3">Admin logged in to system</td></tr>
<tr><td><b>Main flow of events (Success)</b></td><td colspan="3">
<table>
<tr><th>No</th><th>Actor</th><th>Action</th></tr>
<tr><td>1</td><td>Admin</td><td>Select add employee function</td></tr>
<tr><td>2</td><td>System</td><td>Display form to enter employee information</td></tr>
<tr><td>3</td><td>Admin</td><td>Enter employee information then choose save</td></tr>
<tr><td>4</td><td>System</td><td>Validate employee information</td></tr>
<tr><td>5</td><td>System</td><td>Create new employee in system</td></tr>
<tr><td>6</td><td>System</td><td>Grant privileges for employee</td></tr>
<tr><td>7</td><td>System</td><td>Confirm save new employee successfully</td></tr>
</table>
</td></tr>
<tr><td><b>Alternative flow of events</b></td><td colspan="3">
<table>
<tr><th>No</th><th>Actor</th><th>Action</th></tr>
<tr><td>5a</td><td>System</td><td>Display error in validating employee information</td></tr>
</table>
</td></tr>
<tr><td><b>Post-condition</b></td><td colspan="3">None</td></tr>
</table>

3.4 Specification of Use case UC004- “Add Customer”

Use case code	UC004	**Use case name**	Add Customer
Actor	Cashier/Admin		
Pre-condition	Cashier/Admin logged in to system		

Main flow of events (Success)	**No**	**Actor**	**Action**
	1	Cashier/ Admin	Select add customer function
	2	System	Display form to enter customer information
	3	Cashier/ Admin	Enter customer information then choose save
	4	System	Validate customer information
	5	System	Create new customer in system
	6	System	Confirm save new employee successfully
Alternative flow of events	**No**	**Actor**	**Action**
	5a	System	Display error in validating employee information
Post-condition	None		

3.5 Specification of Use case UC005 - “Add supplier”

Use case code	UC005	**Use case name**	Add Supplier
Actor	Warehouse Manager/Admin		
Pre-condition	Warehouse Manager/Admin logged in to system		
Main flow of events (Success)	**No**	**Actor**	**Action**
	1	Warehouse Manager/Admin	Select add supplier function
	2	System	Display form to enter supplier information
	3	Warehouse Manager/Admin	Enter supplier information then choose save
	4	System	Validate supplier information

	5	System	Create new supplier in system
	6	System	Confirm save new supplier successfully
Alternative flow of events	**No**	**Actor**	**Action**
	5a	System	Display error in validating supplier information
Post-condition	None		

3.6 Specification of Use case UC006 - “Modify customer information”

Use case code	UC006	**Use case name**	Modify customer information
Actor	Cashier/Admin		
Pre-condition	Cashier/Admin logged in to system		
Main flow of events (Success)	**No**	**Actor**	**Action**
	1	Admin	Select modify customer information function
	2	System	Display form to enter customer information (with copy of old information)
	3	Admin	Enter new customer information then choose save
	4	System	Validate customer information
	6	System	Save change
	7	System	Confirm save change successfully
Alternative flow of events	**No**	**Actor**	**Action**
	5a	System	Display error in validating customer information
Post-condition	None		

3.7 Specification of Use case UC007 - “Add bill”

<table>
<tr><td><b>Use case code</b></td><td>UC007</td><td><b>Use case name</b></td><td>Add bill</td></tr>
<tr><td><b>Actor</b></td><td colspan="3">Cashier/Admin</td></tr>
<tr><td><b>Pre-condition</b></td><td colspan="3">Cashier/Admin has logged in</td></tr>
<tr><td><b>Main flow of events (Success)</b></td><td colspan="3">
<table>
<tr><th>No</th><th>Actor</th><th>Action</th></tr>
<tr><td>1</td><td>Cashier/ Admin</td><td>Select New Bill function</td></tr>
<tr><td>2</td><td>System</td><td>Create a new bill in a new tab (not affect the current bills)</td></tr>
<tr><td>3</td><td>Cashier/ Admin</td><td>Add products to bill by scanning barcode or entering product ID</td></tr>
<tr><td>4</td><td>Cashier/ Admin</td><td>Enter discount by percent or cash unit if there is any discount</td></tr>
<tr><td>5</td><td>CCashier/ Admin</td><td>Print receipt for customer</td></tr>
<tr><td>6</td><td>Cashier/ Admin</td><td>Receive cash from customer and Enter the amount of cash the customer give</td></tr>
<tr><td>7</td><td>System</td><td>Calculate the charge to give back to customer</td></tr>
<tr><td>8</td><td>Cashier/ Admin</td><td>Take the charge amount calculated by the system to give te customer</td></tr>
<tr><td>9</td><td>System</td><td>Compelete the transaction and write it to history</td></tr>
</table>
</td></tr>
<tr><td><b>Alternative flow of events</b></td><td colspan="3">
<table>
<tr><th>No</th><th>Actor</th><th>Action</th></tr>
<tr><td>5a1</td><td>Cashier/ Admin</td><td>Enter customer ID if customer has a valid id (may be from customer card)</td></tr>
<tr><td>5a2</td><td>System</td><td>Check the customer id</td></tr>
<tr><td>5a3</td><td>System</td><td>Display the customer information</td></tr>
</table>
</td></tr>
</table>

	7a	Cashier/ Admin	If customer does not agree with the bill, discard it
	7b	Cashier/ Admin	If customer wants to correct something, change the bill and go back to step 5.
Post-condition	None		

3.8 *Specification of Use case UC008- “Import products”*

Use case Code	UC008	**Use case Name**	Import products
Actor	Warehouse manager		
Pre-condition	Warehouse manager has logged in		
Main flow of events (Success)	**No**	**Actor**	**Action**
	1	Warehouse manager	Create a new import receipt
	2	Warehouse manager	Add product by typing in the table or import file
	3	System	Write into import history
Alternative flow of events	**No**	**Actor**	**Action**
	2a	Warehouse manager	If goods are importing from unlisted supplier, the ware house manager go to supplier list to add.
Post-conditions	None		

3.9 Specification of Use case UC009 - "Return products"

<table>
<tr><td><b>Use case code</b></td><td>UC009</td><td><b>Use case name</b></td><td>Return products</td></tr>
<tr><td><b>Actor</b></td><td colspan="3">Cashier/Admin</td></tr>
<tr><td><b>Pre-condition</b></td><td colspan="3">Cashier/Admin has logged in</td></tr>
<tr><td><b>Main flow of events (Success)</b></td><td colspan="3">
<table>
<tr><th>No</th><th>Actor</th><th>Action</th></tr>
<tr><td>1</td><td>Cashier/ Admin</td><td>Entering bill number or search for bill</td></tr>
<tr><td>2</td><td>System</td><td>Find bill match to searching then display to screen</td></tr>
<tr><td>3</td><td>Cashier/ Admin</td><td>Choose bill to edit, return product</td></tr>
<tr><td>4</td><td>System</td><td>Copy "sale receipt" to "return receipt"</td></tr>
<tr><td>5</td><td>Cashier/ Admin</td><td>Correct the product quantities and price to return</td></tr>
<tr><td>6</td><td>Cashier/ Admin</td><td>Choose the reason to return</td></tr>
<tr><td>7</td><td>Cashier/ Admin</td><td>Choose the process to do with return goods (return money to customer or not)</td></tr>
<tr><td>8</td><td>System</td><td>Save the new receipt</td></tr>
<tr><td>9</td><td>System</td><td>Print the receipt</td></tr>
</table>
</td></tr>
<tr><td><b>Alternative flow of events</b></td><td colspan="3">
<table>
<tr><th>No</th><th>Actor</th><th>Action</th></tr>
<tr><td>8</td><td>System</td><td>Write to finance history in case returning money to customer</td></tr>
</table>
</td></tr>
<tr><td><b>Post-condition</b></td><td colspan="3">None</td></tr>
</table>

4 Supplementary

- ***Functionality***

Multi machine – one database. You can use this app by client application and connect to server.

- ***Usability***

Train the employees and store owner to use the system

- ***Reliability***

Automatically restart the server after a system failure.

- ***Performance***

Acceptable performance. No lag when using.

- ***Supportability***

Host and maintain the system for the stores.

- ***Other requirements***

Nice and modern user interface.