**Web-based Ordering & Ingredient Estimating for Bakery Manufacturer**

Software Requirement Specification

By

**Miss. Nontra Nonsee 542115026**

**Mr. Parinya Panyanak 542115034**

Department of Software Engineering

College of Arts, Media and Technology

Chiang Mai University

Project Advisor

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Mr.Phudinan Singkhamfu**

|  |  |  |
| --- | --- | --- |
| **Software Requirement Specification** | | |
| Cross Ref. VSE-29110 | **Coverage Level:** | **Version:** |
|  | V.3.0 | V.3.0 |

|  |  |
| --- | --- |
| **Process Ownership** | **Approving Authority** |
| Web-based Ordering & Ingredient Estimating for Bakery Manufacturer | Aj.Phudinan Singkhamfu |
| **Scope** | **Approved Date** |
| Senior Project | 30 June 2015 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version Number** | **Record Date** | **Prepared/**  **Modified By** | **Reviewed By** | **Change Details** |
| **Documents History** | | | | |
| V.1.0 | 06-05-2015 | NN, PP | NN, PP | - Add Chapter 1  - Add Chapter 2  - Add Chapter 3  - Add Chapter 4  - Add Chapter 5  - Fix inappropriate words |
| V.2.0 | 07-05-2015 | NN, PP | NN, PP | - Adjust Chapter 2  - Adjust Chapter 3  - Fix inappropriate words |
| V.2.4 | 09-05-2015 | NN, PP | NN, PP | - Adjust Chapter 4  - Adjust Chapter 5  - Fix inappropriate words |
| V.2.6 | 12-05-2015 | NN, PP | NN, PP | - Adjust Product Features  - Fix inappropriate words |
| V.2.9 | 24-05-2015 | NN, PP | NN, PP | - Adjust Chapter 5  - Fix inappropriate words |
| V.3.0 | 27-05-2015 | NN, PP | NN, PP | - Adjust Product Features  - Adjust Chapter 3  - Adjust Chapter 4  - Adjust Chapter 5  - Fix inappropriate words |

**\*NN = Nontra Nonsee**

**\*PP = Parinya Panyanak**

**\*PS = Phudinan Singkhamfu**

**Table of contents**

[**Chapter One | Introduction** 4](#_Toc423340632)

[1.1 Objective 4](#_Toc423340633)

[1.2 Intended Audience and Reading Suggestions 4](#_Toc423340634)

[1.3 Project Scope 4](#_Toc423340635)

[**Chapter Two | Overall Description** 5](#_Toc423340636)

[2.1 Product Perspective 5](#_Toc423340637)

[2.2 Product Features 5](#_Toc423340638)

[2.3 User Classes and Characteristics 6](#_Toc423340639)

[2.4 Operation Environment 6](#_Toc423340640)

[2.5 Design and Implementation Constraints 6](#_Toc423340641)

[**Chapter Three | User Requirements** 7](#_Toc423340642)

[**Chapter Four | System Features** 9](#_Toc423340643)

[4.1 Software Requirement Specification 9](#_Toc423340644)

[**Chapter Five | Specific Requirement** 25](#_Toc423340645)

[5.1 Use Case Scenarios 25](#_Toc423340646)

[5.2 Use Case Description 28](#_Toc423340647)

[5.3 Use Case Specification 49](#_Toc423340648)

# **Chapter One | Introduction**

## 1.1 Objective

* To help manufacturer estimate ingredients usage and control manufacturing standard.
* To support the collaboration within the manufacturer.
* To control the manufacturer document easily.
* To support the ordering process can work conveniently.

## 1.2 Intended Audience and Reading Suggestions

This Software Requirement Specification was created for everyone that involves with the Web-based Ordering & Ingredient Estimating for Bakery Manufacturer. It will make benefits for people as following

**Development Team**

**-** Convenient to make strategies and planning process.

**-** Control and guarantee qualities of the system to make it right according to the standard.

**-** Check the customer requirement for makes sure that everyone has the same understand and avoid the fault of the work.

**-** In testing system, tester will verify the system work correctly without any problem.

**Customer**

**-** Help customer to understand about quality and limitation of this system.

**-** Checking the requirement of the customer and the system are the same.

## 1.3 Project Scope

The Web-based Ordering & Ingredient Estimating for Bakery Manufacturer is a management system that helps the admin can manage manufacture’s information efficiency. Then the system helps the wholesaler to manage their order easier, and the worker also can apply ingredient usage’s information with a traditional workflows. The project focuses on implementation of the general order system and supports the process of the manufacturer system.

# **Chapter Two | Overall Description**

## Product Perspective

Web-based Ordering & Ingredient Estimating for Bakery Manufacturer has three main sections which are Admin section, Wholesaler section, and Worker section. This system can help them for convenient to order bakery product, check manufacturer’s information and apply accurate documentation with workflows. The system also can help for keeping ordered history, calculate ingredient usage in each manufacturing and make a report are related to system.

## Product Features

Main features of the Web-based Ordering & Ingredients Estimating for Bakery Manufacturer will be following

**Feature 1: User Management System**

* User registration feature
* Member authentication feature
* Member information page
* Manage member information

**Feature 2: Product System**

* Product information page
* Manage product information

**Feature 3: Order System**

* Order feature
* Order information page
* Search order information
* Mark progressive of order status while the order in the stage
* Progress order status information while the order in each stage
* Search ordered history information
* Ordered history information page

**Feature 4: Ingredient Management System**

* Manage ingredient information
* Search ingredient information
* Ingredient information page

**Feature 5: Ingredient Estimation**

* Menu recipe management
* Menu recipe information page
* Search menu recipe information
* Ingredient estimation feature
* Search previous ingredient estimation information
* Ingredient estimation information page

**Feature 6: Report System**

* Report feature
* Report result page
* Chart feature
* Chart data page

## 2.3 User Classes and Characteristics

Intended users for this project are wholesaler who make an order to bakery manufacturer, worker that involve knowing information and admin who control primary information in the system.

## 2.4 Operation Environment

Firefox Version 32.0 or Later

## 2.5 Design and Implementation Constraints

**Design**

Provide web application for easy to use. Design for easy to understand. The user who not have skill able to order bakery product and useful all function from the website.

**Constraints**

- Only member can access the web application.

- Require internet connection.

- The web application can only use with the Phungnoi bakery workflows.

# **Chapter Three | User Requirements**

* **URS-01:** Admin can register to activate member to access the application by create

new member information; username, password, first name, last name, phone number, address, role name, and profile picture into the database.

* **URS-02:** Member authenticate the web application account.
  + **URS-03:** Admin can manage all member information; username, password,

first name, last name, phone number, address, role name, and profile picture in the database.

* **URS-04:** Member can update own personal information; username, password, first

name, last name, phone number, address, and profile picture in the database.

* **URS-05:** Member can view own personal information; member ID, username,

password, first name, last name, phone number, address, role name, create date, last sign in date, and profile picture in the database.

* **URS-06:** Admin can view all member’s information; member ID, username,

password, first name, last name, phone number, address, role name, create date, last sign in date, and profile picture in the database.

* **URS-07:** Admin can manage all product information; product name, price,

category, information, and product picture in the database.

* **URS-08:** Member and Visitor can view all bakery product information by search

filter keywords of the product ID, product name, price, and category in the database.

* **URS-09:** Wholesaler can order bakery products from the existing bakery product

in the system. An order can select by input product name, product quantity, and delivery date.

* **URS-10:** Wholesaler can manage own ordered information; product name, product

quantity, and delivery date. (According to time constraint of 8:00 am – 12:00 am, same day with current order created)

* **URS-11:** Wholesaler can view their current order information; order ID, first name,

last name, product ID, product name, product quantity, price, total price, discount, order date, delivery date, and order status.

* **URS-12:** Worker can view current order information of all wholesalers by search

filter keywords of the order ID, member ID, first name, last name, order status, and delivery date in the database.

* **URS-13:** Worker can view the summary of total bakery product are ordered in

daily.

* **URS-14:** Worker can mark progressive of all order status while the order in the

stage.

* **URS-15:** Wholesaler can view their ordered history by choosing a period of the

day, month, and year to searching the previous ordered in the database.

* **URS-16:** Worker can view all ordered history by choosing a period of the day,

month, and year with another filter keywords. For an example member ID, username, delivery date, and order status to searching the result in the database.

* **URS-17:** Admin can manage all ingredient information; ingredient name, category,

and ingredient picture in the database.

* **URS-18:** Admin can view all ingredient information by search filter keywords of

The ingredient ID, ingredient name, and category in the database.

* **URS-19:** Admin can manage all menu recipe information; product name,

ingredient name, ingredient quantity, and units in the database.

* **URS-20:** Admin can view all menu recipe information by search filter keywords of

the product ID, product name, and ingredient name in the database.

* **URS-21:** Worker can view the current ingredient estimation result; product ID,

product name, ingredient name, and ingredient quantity.

* **URS-22:** Worker can view previous ingredient estimation by choosing a period

of the day, month, and year.

* **URS-23:** Wholesaler and Worker can make a different report type according to

their member’s role.

* **URS-24:** Wholesaler and Worker can make a graph result from the ordered

statistical data.

# **Chapter Four | System Features**

## Software Requirement Specification

**URS-01:** Admin can register to activate member to access the application by create new member information; username, password, first name, last name, phone number, address, role name, and profile picture into the database.

**SRS-01:** The system can connect to the phungnoi database and access Member and webpages\_Roles tables.

**SRS-02:** The system shall provide a link for the admin registering new member information in member management page.

**SRS-03:** The system shall provide a registration form with a creating new member page for the admin input username, password, first name, last name, phone number, address, role name, and profile picture.

**SRS-04:** The system shall generate a new member ID by sort all member ID indefinitely.

**SRS-05:** The system shall provide a button for the admin saving new member in the database.

**SRS-06:** The system shall provide a service to validation input information; username, password, first name, last name, phone number, address, role name, and profile picture.

**SRS-07:** The system shall provide a message to display an error message if the input information has wrong.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-09:** The system shall provide a service to upload new member information to display in the member profile interface.

**URS-02:** Member authenticate the web application account.

**SRS-01:** The system can connect to the phungnoi database and access Member and webpages\_Roles tables.

**SRS-10:** The system shall provide a link for the member logging in to the web application on the authentication page.

**SRS-11:** The system shall provide a login form in an authenticate interface for the admin, worker, and wholesaler login members as a different role.

**SRS-12:** The system can hide the characters in the password field.

**SRS-13:** The system shall provide a button for the member logging in to the system.

**SRS-14:** The system shall provide a service to validation input information; username and password.

**SRS-07:** The system shall provide a message to display an error message if the input information has wrong.

**SRS-15:** The system can take the wholesaler, worker or admin to the main layout page of their roles.

**SRS-16:** The system shall provide a service to keep the last sign in date and time of the member.

**SRS-17:** The system shall provide a link for the member logging out from the web application.

**SRS-18:** The system shall provide a confirmation dialog box after logout link has chosen.

**SRS-19:** The system can take the wholesaler, worker or admin exit from the program.

**URS-03:** Admin can manage all member information; username, password, first name, last name, phone number, address, role name, and profile picture in the database.

**SRS-01:** The system can connect to the phungnoi database and access Member and webpages\_Roles tables.

**SRS-20:** The system shall provide a link for updating member information in the member management interface.

**SRS-21:** The system shall provide the existing member information in member management page for update new member information; username, password, first name, last name, phone number, address, role name, and profile picture.

**SRS-05:** The system shall provide a button for the admin saving new member in the database.

**SRS-06:** The system shall provide a service to validation input information; username, password, first name, last name, phone number, address, role name, and profile picture.

**SRS-07:** The system shall provide a message to display an error message if the input information has wrong.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-09:** The system shall provide a service to upload new member information to display in the member profile interface.

**SRS-22:** The system shall provide a link for the admin deleting member information in the member management interface.

**SRS-23:** The system shall provide a confirmation dialog box after delete link has chosen.

**SRS-24:** The system shall provide a service to remove member information that chosen from the database.

**URS-04:** Member can update own personal information; username, password, first name, last name, phone number, address, and profile picture in the database.

**SRS-01:** The system can connect to the phungnoi database and access Member and webpages\_Roles tables.

**SRS-20:** The system shall provide a link for updating member information in the member management interface.

**SRS-21:** The system shall provide the existing member information in member management page for update new member information; username, password, first name, last name, phone number, address, role name, and profile picture.

**SRS-05:** The system shall provide a button for the admin saving new member in the database.

**SRS-06:** The system shall provide a service to validation input information; username, password, first name, last name, phone number, address, role name, and profile picture.

**SRS-07:** The system shall provide a message to display an error message if the input information has wrong.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-09:** The system shall provide a service to upload new member information to display in the member profile interface.

**URS-05:** Member can view own personal information; member ID, username, password, first name, last name, phone number, address, role name, create date, last sign in date, and profile picture in the database.

**SRS-01:** The system can connect to the phungnoi database and access Member and webpages\_Roles tables.

**SRS-26:** The system shall provide a link for viewing member information in the member profile interface.

**SRS-27:** The system shall provide a user interface to display the existing member information which are the member ID, username, password, first name, last name, phone number, address, role name, create date, last sign in date, and profile picture on member profile page.

**URS-06:** Admin can view all member’s information; member ID, username, first name, last name, phone number, address, role name, create date, last sign in date, and profile picture in the database.

**SRS-01:** The system can connect to the phungnoi database and access Member and webpages\_Roles tables.

**SRS-27:** The system shall provide a textbox in member management interface to search filter keywords of the member information; member ID, username, role name, and last sign in date in the database.

**SRS-28:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-29:** The system shall provide a service to matching input information; member ID, username, first name, last name, phone number, address, role name or last sign in date with the existing database.

**SRS-30:** The system shall provide a user interface to display all member list. In each label, shall display the member ID, username text of member information that related to filter keywords.

**SRS-26:** The system shall provide a link for viewing member information in the member profile interface.

**SRS-27:** The system shall provide a user interface to display the existing member information which are the member ID, username, password, first name, last name, phone number, address, role name, create date, last sign in date, and profile picture on member profile page.

**URS-07:** Admin can manage all product information; product name, price, category, information, and product picture in the database.

**SRS-31:** The system can connect to the phungnoi database and access webpages\_Roles and Product tables.

**SRS-32:** The system shall provide a link for the admin creating new bakery product information in the product management interface.

**SRS-33:** The system shall provide a bakery product form with a creating new product page for the admin input product name, price, category, information, and product picture.

**SRS-34:** The system shall generate new product ID by sort all product ID indefinitely.

**SRS-35:** The system shall provide a link for the admin updating bakery product information in the product management interface.

**SRS-36:** The system shall provide the existing bakery product information in product management page for the admin update new bakery product information; product name, price, category, information, and product picture into the database.

**SRS-37:** The system shall provide a button for the admin saving new bakery product in the database.

**SRS-38:** The system shall provide a service to validation input information; product name, price, category, information, and product picture.

**SRS-39:** The system shall provide a message to display a notification after the bakery product form has saved.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-40:** The system shall provide a service to upload new bakery product information to display in the product interface.

**SRS-41:** The system shall provide a link for the admin deleting bakery product information from the product interface.

**SRS-25:** The system shall provide a confirmation dialog box after delete link has chosen.

**SRS-42:** The system shall provide a service to remove bakery product information which chosen from the database.

**SRS-43:** The system shall provide a message to display a notification after the bakery product information has removed.

**URS-08:** Member and visitor can view all bakery product information by search filter keywords of the product ID, product name, price, and category in the database.

**SRS-44:** The system can connect to the phungnoi database and access Product tables.

**SRS-45:** The system shall provide a textbox in product interface to search filter keywords of the bakery product information; product ID, product name, price, and category in the database.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-46:** The system shall provide a service to match input information; product ID, product name, price, and category with the existing database.

**SRS-47:** The system shall provide a user interface to display all list of bakery product. In each label, shall display the product ID, product name text, and price that related to filter keywords.

**SRS-48:** The system shall provide a link for viewing bakery product information in the product interface.

**SRS-49:** The system shall provide a user interface to display the existing bakery product information which are the product ID, product name, price, category, information, and product picture on the product page.

**URS-09:** Wholesaler can order bakery products from the existing bakery product in the system. An order can select by input product name, product quantity, and delivery date.

**SRS-50:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Product, Order, OrderDetail, and Status tables.

**SRS-51:** The system shall provide a link for the wholesaler ordering in the order interface within 8:00 am – 12:00 pm of daily.

**SRS-52:** The system shall provide a product cart in order page for the wholesaler select product name, product quantity, and delivery date.

**SRS-53:** The system shall generate the other information from product name field consist of the product ID and price in each label of bakery product list.

**SRS-54:** The system shall provide a button for the wholesaler repeating previous ordered information from the ordered history interface.

**SRS-55:** The system shall provide a user interface to display all ordered history list. In each label, shall display the delivery date, order ID, order status text and total price which related to filter keywords.

**SRS-56:** The system shall provide a link for the wholesaler requesting to use previously ordered list in the order interface.

**SRS-57:** The system shall provide a confirmation dialog box after previously ordered link has chosen.

**SRS-58:** The system shall provide a service to reuse all bakery product name and quantity information from the chosen previously ordered.

**SRS-59:** The system shall provide a calculating function to compute the total price and discount of each order.

**SRS-60:** The system shall provide a user interface to display the result of the total price and discount from all bakery product list.

**SRS-62:** The system shall provide a button for the wholesaler sending finished product cart.

**SRS-63:** The system shall provide a service to validation incoming order follow the time condition. (According to time constraint of 8:00 pm – 12:00 pm in daily)

**SRS-64:** The system shall provide a message to display a notification after the product cart has sent.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-65:** The system shall provide a service to upload new ordered information to display in the order information interface.

**SRS-66:** The system shall provide a button for the wholesaler saving unfinished product cart in their account.

**URS-10:** Wholesaler can manage own ordered information; product name, product quantity, and delivery date. (According to time constraint of 8:00 pm – 12:00 pm, same day with current order created)

**SRS-50:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Product, Order, OrderDetail, and Status tables.

**SRS-67:** The system shall provide a link for the wholesaler updating current order information in the order interface.

**SRS-68:** The system shall provide a user interface to display current order information in the order page for the wholesaler update order information; product name, product quantity, and delivery date into the database.

**SRS-53:** The system shall generate the other information from product name field consist of the product ID and price in each label of bakery product list.

**SRS-59:** The system shall provide a calculating function to compute the total price and discount of each order.

**SRS-60:** The system shall provide a user interface to display the result of the total price and discount from all bakery product list.

**SRS-62:** The system shall provide a button for the wholesaler sending finished product cart.

**SRS-63:** The system shall provide a service to validation incoming order follow the time condition. (According to time constraint of 8:00 pm – 12:00 pm in daily)

**SRS-64:** The system shall provide a message to display a notification after the product cart has sent.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-65:** The system shall provide a service to upload new ordered information to display in the order information interface.

**SRS-69:** The system shall provide a cancel link for the wholesaler canceling current order information in the order interface.

**SRS-70:** The system shall provide a confirmation dialog box after cancel link has chosen.

**SRS-71:** The system shall provide a service to remove current order information from the database.

**SRS-72:** The system shall provide a message to display a notification after the current order information has canceled.

**URS-11:** Wholesaler can view their current order information; order ID, first name, last name, product ID, product name, product quantity, price, total price, discount, order date, delivery date, and order status.

**SRS-73:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Status, and OrderDetail tables.

**SRS-74:** The system shall provide a link for viewing current order information in the order information interface.

**SRS-75:** The system shall provide a user interface to display current order information which are the order ID, member ID, first name, last name, product ID, product name, product quantity, order status, delivery date, order date, total price and discount on the current order information page.

**URS-12:** Worker can view current order information of all wholesalers by search filter keywords of order ID, member ID, first name, last name, order status, and delivery date in the database.

**SRS-73:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Status, and OrderDetail tables.

**SRS-76:** The system shall provide a textbox in order interface to search the order information; order ID, member ID, first name, last name, and order status.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-77:** The system shall provide a service to match input information; order ID, member ID, first name, last name, and order status with the existing current order information in the database.

**SRS-78:** The system shall provide a user interface to display all current order list. In each label, shall display the order ID, first name text, last name text, total price, and order status text which related to filter keywords.

**SRS-74:** The system shall provide a link for viewing current order information in the order information interface.

**SRS-75:** The system shall provide a user interface to display current order information which are the order ID, member ID, first name, last name, product ID, product name, product quantity, order status, delivery date, order date, total price and discount on the current order information page.

**URS-13:** Worker can view the summary of total bakery product are ordered in daily.

**SRS-79:** The system can connect to the phungnoi database and access webpages\_Roles and OrderDetail tables.

**SRS-80:** The system shall provide a user interface to display the current total ordered information which are the product ID, product name, total number, and summary price on the total ordered page.

**SRS-81:** The system shall provide a textbox in the total ordered interface to search the filter keywords of the day, month, and year of the previously total ordered result.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-82:** The system shall provide a service to match the period of the day, month, and year in the field textbox with the previously total ordered information in the database.

**SRS-83:** The system shall provide a user interface to display all total ordered list. In each label, shall display the delivery date and summary price which related to filter keywords.

**SRS-84:** The system shall provide a link for the worker viewing the total number of all bakery product and summary price in the total ordered interface.

**SRS-85:** The system shall provide a user interface to display the total ordered information which chosen by the worker.

**URS-14:** Worker can mark progressive of all order status while the order in the stage.

**SRS-73:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Status, and OrderDetail tables.

**SRS-75:** The system shall provide a user interface to display current order information which are the order ID, member ID, first name, last name, product ID, product name, product quantity, order status, delivery date, order date, total price and discount on the current order information page.

**SRS-86:** The system shall provide a drop down list for the worker selecting progressive of order status in the order management interface.

**SRS-87:** The system shall provide a button for the worker saving changed order status in the order management interface.

**SRS-88:** The system shall provide a message to display a notification after the order status in the preparation stage has changed.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-89:** The system shall provide a service to upload new progressive of order status to display in the order information interface.

**URS-15:** Wholesaler can view their ordered history by choosing a period of the day, month, and year to searching the previous ordered in the database.

**SRS-73:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Status, and OrderDetail tables.

**SRS-90:** The system shall provide a textbox in order interface to search the filter keywords of the day, month, and year of the delivery date.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-91:** The system shall provide a service to match the period of the day, month, and year in field textbox with the ordered history information in the database.

**SRS-55:** The system shall provide a user interface to display all ordered history list. In each label, shall display the delivery date, order ID, order status text and total price which related to filter keywords.

**SRS-92:** The system shall provide a link for viewing the ordered history information in the ordered history interface.

**SRS-93:** The system shall provide a user interface to display the ordered history information which are the member ID, first name, last name, order ID, product ID, product name, product quantity, total price, discount, order status and delivery date on the ordered history page.

**URS-16:** Worker can view all ordered history by choosing a period of the day, month, and year with another filter keywords. For an example member ID, first name, last name, delivery date, order status, and order ID to searching the result in the database.

**SRS-73:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Status, and OrderDetail tables.

**SRS-94:** The system shall provide the first textbox in the order interface to search the filter keywords of the day, month, and year. And the second textbox to search ordered history information; member ID, first name, last name, delivery date, order status, and order ID.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-95:** The system shall provide a service to match the filter keywords in the first textbox and second textbox with the ordered history information in the database.

**SRS-55:** The system shall provide a user interface to display all ordered history list. In each label, shall display the delivery date, order ID, order status text and total price which related to the filter keywords.

**SRS-92:** The system shall provide a link for viewing the ordered history information in the ordered history interface.

**SRS-93:** The system shall provide a user interface to display the ordered history information which are the member ID, first name, last name, order ID, product ID, product name, product quantity, total price, discount, order status and delivery date on the ordered history page.

**URS-17:** Admin can manage all ingredient information; ingredient name, category, and ingredient picture in the database.

**SRS-96:** The system can connect to the phungnoi database and access webpages\_Roles and Ingredient tables.

**SRS-97:** The system shall provide a link for the admin creating new ingredient information in the ingredient management interface.

**SRS-98:** The system shall provide an ingredient form in the ingredient management page for the admin input ingredient information; ingredient name, category, and ingredient picture.

**SRS-99:** The system shall provide a link for the admin updating ingredient information in the ingredient interface.

**SRS-100:** The system shall provide a user interface to display the existing ingredient information in the ingredient management page for the admin update new ingredient information; ingredient name, category, and ingredient picture.

**SRS-101:** The system shall generate the other information from the ingredient name field consist of the ingredient ID in each label of the ingredient list.

**SRS-102:** The system shall provide a button for the admin saving new ingredient in the database.

**SRS-103:** The system shall provide a service to validation input information; ingredient name, category, and ingredient picture.

**SRS-104:** The system shall provide a message to display a notification after the ingredient form has saved.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-105:** The system shall provide a link for the admin deleting ingredient information in the ingredient information interface.

**SRS-106:** The system shall provide a confirmation dialog box after delete link has chosen.

**SRS-107:** The system shall provide a service to remove ingredient information which chosen from the database.

**SRS-108:** The system shall provide a message to display a notification after the ingredient information has removed.

**URS-18:** Admin can view all ingredient information by search filter keywords of the ingredient ID, ingredient name, and category in the database.

**SRS-96:** The system can connect to the phungnoi database and access webpages\_Roles and Ingredient tables.

**SRS-109:** The system shall provide a textbox in the ingredient interface to search the filter keywords of the ingredient information; ingredient ID, ingredient name, and category.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-110:** The system shall provide a service to match the input information; ingredient ID, ingredient name, category, and ingredient picture with the existing ingredient information in the database.

**SRS-111:** The system shall provide a user interface to display all ingredient list. In each label, shall display the ingredient name text and ingredient ID which related to the filter keywords.

**SRS-112:** The system shall provide a user interface to display the existing ingredient information which are the ingredient ID, ingredient name, category, and ingredient picture on the ingredient information page.

**URS-19:** Admin can manage all menu recipe information; product name, ingredient name, ingredient quantity, and units in the database.

**SRS-113:** The system can connect to the phungnoi database and access webpages\_Roles, Product, Ingredient, Recipe, and Units tables.

**SRS-114:** The system shall provide a link for the admin creating menu recipe information in the menu recipe interface.

**SRS-115:** The system shall provide a menu recipe form in the menu recipe page for the admin input menu recipe information; product name, ingredient name, ingredient quantity, and units.

**SRS-116:** The system shall provide a link for the admin updating menu recipe information in the menu recipe interface.

**SRS-117:** The system shall provide a user interface to display the existing menu recipe information in the menu recipe page for the admin update new menu recipe information; product name, ingredient name, ingredient quantity, and units.

**SRS-118:** The system shall generate the other information from the product name and ingredient name field consist of the product ID, information, category and ingredient ID in each menu recipe.

**SRS-119:** The system shall provide a button for the admin saving new menu recipe in the database.

**SRS-120:** The system shall provide a service to validation input information; product name, ingredient name, ingredient quantity, and units.

**SRS-121:** The system shall provide a message to display a notification after menu recipe form has saved.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-122:** The system shall provide a link for deleting menu recipe information in the menu recipe interface.

**SRS-123:** The system shall provide a confirmation dialog box after delete link has chosen.

**SRS-124:** The system shall provide a service to remove menu recipe information which chosen from the database.

**SRS-125:** The system shall provide a message to display a notification after the menu recipe information has removed.

**URS-20:** Admin can view all menu recipe information by search filter keywords of the recipe ID, product ID, product name, ingredient ID, and ingredient name in the database**.**

**SRS-113:** The system can connect to the phungnoi database and access webpages\_Roles, Product, Ingredient, Recipe, and Units tables.

**SRS-126:** The system shall provide a textbox in the menu recipe interface to search filter keywords of the menu recipe information; recipe ID, product ID, product name, ingredient ID, and ingredient name.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-127:** The system shall provide service to match the input information; recipe ID, product ID, product name, ingredient ID, and ingredient name.

**SRS-128:** The system shall provide a user interface to display all menu recipe list. In each label, shall display the product name text and recipe ID on menu recipe list.

**SRS-129:** The system shall provide a user interface to display the existing menu recipe information which are the recipe ID, product ID, product name, product picture, information, category, ingredient ID, ingredient name, ingredient quantity, and units on the menu recipe information page.

**URS-21:** Worker can view the current ingredient estimation result; product ID, product name, ingredient ID, ingredient name, ingredient quantity, and units.

**SRS-130:** The system can connect to the phungnoi database and access webpages\_Roles, OrderDetail, Recipe and Units tables.

**SRS-131:** The system shall provide a button for the admin viewing the ingredient usage in the ingredient estimation interface.

**SRS-132:** The system shall provide a calculating function to compute the ingredient usage by using information from the menu recipe and total ordered.

**SRS-133:** The system shall provide a user interface to display the ingredient estimation information which are the product ID, product name, ingredient ID, ingredient name, ingredient quantity, and units on the ingredient estimation page.

**URS-22:** Worker can view previous ingredient estimation by choosing a period of the day, month, and year.

**SRS-130:** The system can connect to the phungnoi database and access webpages\_Roles, OrderDetail, Recipe and Units tables.

**SRS-134:** The system shall provide a textbox in the ingredient estimation interface to search the filter keywords of the day, month, and year of the previous ingredient estimation result.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-135:** The system shall provide a service to match the period of the day, month, and year in field textbox with the previous ingredient estimation result in the database.

**SRS-136:** The system shall provide a user interface to display the previous ingredient estimation list. In each label, shall display the delivery date which related to the filter keywords.

**SRS-137:** The system shall provide a button for the admin viewing the ingredient usage in the ingredient estimation interface.

**SRS-138:** The system shall provide a calculating function to compute the ingredient usage by using information from the menu recipe and total ordered.

**SRS-139:** The system shall provide a user interface to display the ingredient estimation information which are the product ID, product name, ingredient ID, ingredient name, ingredient quantity, and units on the ingredient estimation page.

**URS-23:** Wholesaler and Worker can make a different report type according to their member’s role.

**SRS-140:** The system can connect to the phungnoi database and access Member, webpages\_Roles, OrderDetail, and Recipe tables.

**SRS-141:** The system shall provide a button for making a report.

**SRS-142:** The system shall provide a report function to retrieve the summary information from the database.

**SRS-143:** The system shall provide a user interface to display the summary report contents on the report page.

**SRS-144:** The system shall provide a button for saving a summary report as a PDF file format from the report interface.

**SRS-145:** The system shall provide a service to export a summary report document out.

**SRS-146:** The system shall provide a button for printing a summary report as a documented paper.

**URS-24:** Wholesaler and Worker can make a graph result from the ordered statistical data.

**SRS-73:** The system can connect to the phungnoi database and access Member, webpages\_Roles, and OrderDetail tables.

**SRS-147:** The system shall provide a button for converting ordered information to be a statistical graph.

**SRS-148:** The system shall provide a service to process statistical data became to a graph image.

**SRS-149:** The system shall provide a user interface to display the graph image and statistical portion on the graph page.

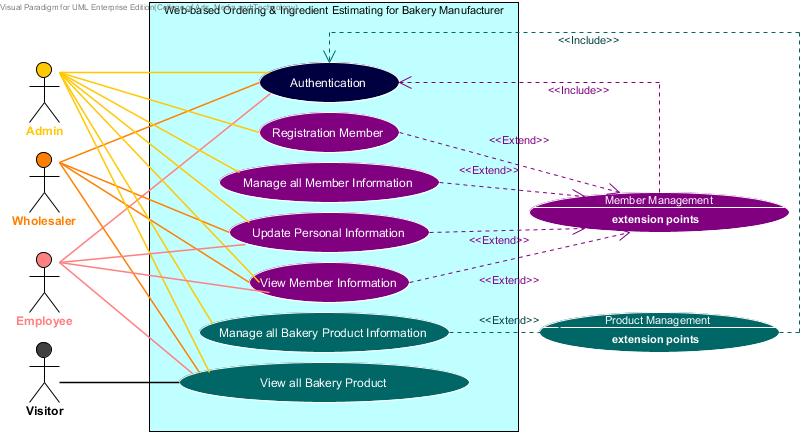
**SRS-150:** The system shall provide a button for saving a graph as an image file format from the graph interface.

**SRS-151:** The system shall provide a service to export a graph image out.

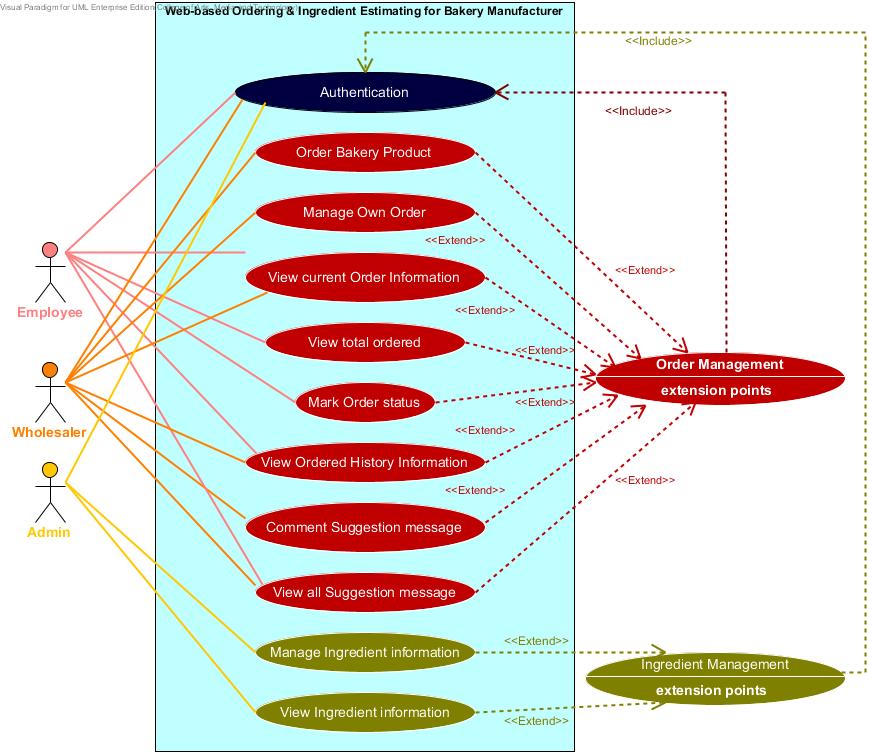
**SRS-152:** The system shall provide a button for printing a graph image as a documented paper.

# **Chapter Five | Specific Requirement**

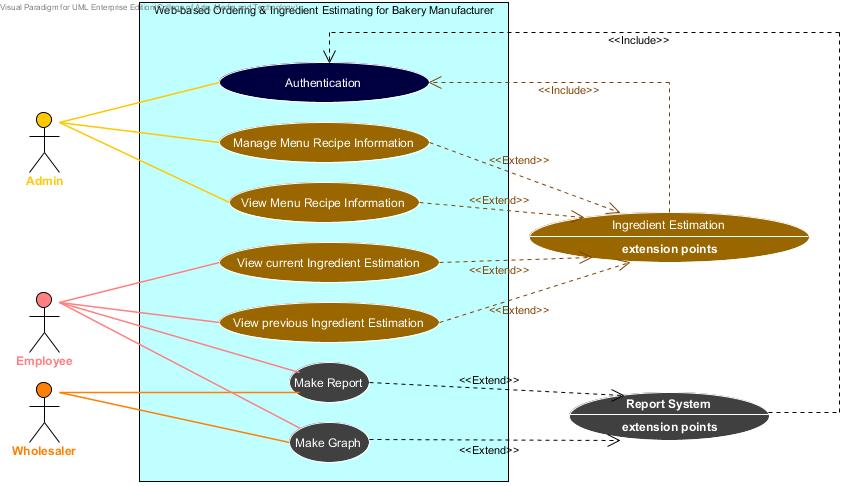
## 5.1 Use Case Scenarios



**Figure 1: Use case diagram that show activity of Progress report I**

****

**Figure 2: Use case diagram that show activity of Progress report II**



**Figure 3: Use case diagram that show activity of Final Progress report**

## 5.2 Use Case Description

**User Case:** Registration member

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 01 |
| **Use Case Name:** | Registration member |
| **Actors:** | Admin |
| **Short Description:** | This use case allows admin can activate member by register them to the website. |
| **Preconditions:** | Admin have login to the website. |
| **Post conditions:** | Member will use username and password to login to the website. |
| **Normal Flow:** | 1. Request a registration link in the member management page.  2. Input username, password, first name, last name, phone number, address, role name, and profile picture into a registration form.  3. System shall generate member ID for a new member.  4. Click “Save” button.  5. System shall checks validation from the input information.  6. System shall display a successful message of registration to the admin.  7. System shall collect information of the new member to the database. (member ID, username, password, first name, last name, phone number, address, role name, and profile picture)  8. System shall display a new member information to the member profile page. |
| **Alternative and Exceptional:** | 5a. In step 5 of the normal flow.  If the input invalid information type.  5.1 System displays the error message to the admin.  5.2 System shall request the command from the admin input again.  5b. In step 5 of the normal flow.  If the input valid information type.  5.1 Use case resumes on step 6 of normal flow. |

**User Case:** Authentication

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 02 |
| **Use Case Name:** | Authentication |
| **Actors:** | Member (Admin, Worker, Wholesaler) |
| **Short Description:** | This use case allows all member login to the system and logout from the system. |
| **Preconditions:** | Members have username and password. |
| **Post conditions:** | Member can login, use a service and logout from the website. |
| **Normal Flow:** | 1. System shall display a login form.  2. Input correct username and password.  3. System shall hide the password characters.  4. Click “Login” button.  5. System shall verify username and password from the database.  6. System shall collect sign in date and time information to the database.  7. System shall display a successful message to the member.  8. Then the member enters into the web application and can use the service of the application.  9. Click “Logout” button.  10. System shall back to the login page. |
| **Alternative and Exceptional:** | 5a. In step 5 of the normal flow.  If the input valid username and password.  5.1 Use case resumes on step 6 of normal flow.  5b. In step 5 of the normal flow.  If the input invalid username or password.  5.1 System displays the error message to the member.  5.2 System shall request the command from the member input again. |

**User Case:** Manage all member information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 03 |
| **Use Case Name:** | Manage all member information |
| **Actors:** | Admin |
| **Short Description:** | This use case allows the admin can manage all member information. The admin can update all member information and remove member in the system. |
| **Preconditions:** | Admin have login to the website and member information has created. |
| **Post conditions:** | Admin can update member information or remove a member from the database. |
| **Normal Flow:** | 1. Request the member management page.  2. System shall display current member information.  3. Select update information or delete member functions.  4. Input update information (username, password, first name, last name, phone number, address, role name, and profile picture).  5. Click “Save” button to save the updated information.  6. Click “OK” button to remove the member.  7. System shall display a successful message to the admin.  8. System shall collect updated member information to the database. (username, password, first name, last name, phone number, address, role name, and profile picture)  9. System shall display an update member information to the member profile page. |
| **Alternative and Exceptional:** | 3a. In step 3 of the normal flow.  If the admin click update information.  3.1 System validates the input information before update in the database.  3.2 Use case resume on step 7 of normal flow.  3b. In step 3 of the normal flow.  If the admin click delete member.  3.1 Use case resume on step 6 of normal flow. |

**User Case:** Update personal information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 04 |
| **Use Case Name:** | Update personal information |
| **Actors:** | Member (Admin, Worker, Wholesaler) |
| **Short Description:** | This use case allows all member update own information in the system. |
| **Preconditions:** | Members have login to the website and member information has created. |
| **Post conditions:** | Member can update own information and save new information into the database. |
| **Normal Flow:** | 1. Request the member profile page.  2. Click “Update information” button.  3. Input update personal information (username, password, first name, last name, phone number, address, role name, and profile picture).  4. Click “Save” button.  5. System shall checks validation the input information.  6. System shall display a successful message in updated information to the member.  7. System shall collect updated member information to the database. (username, password, first name, last name, phone number, address, role name, and profile picture)  8. System shall display a new member information to the member profile page. |
| **Alternative and Exceptional:** | 5a. In step 5 of the normal flow.  If the input invalid information type.  5.1 System displays the error message to the member.  5.2 System shall request the command from the member input again.  5b. In step 5 of the normal flow.  If the input valid information type.  5.1 Use case resume on step 6 of normal flow. |

**User Case:** View member information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 05 |
| **Use Case Name:** | View member information |
| **Actors:** | Member (Admin, Worker, Wholesaler) |
| **Short Description:** | This use case allows all member view their personal information and the admin can view all member information in the system. |
| **Preconditions:** | Members have login to the website. |
| **Post conditions:** | Member can know member information into the database. |
| **Normal Flow:** | 1. Request the member profile page.  2. Click “View Profile” button.  3. System shall request member information from the database.  4. System shall display a member information to the member profile page. |
| **Alternative and Exceptional:** | - |

**User Case:** Manage all bakery product information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 06 |
| **Use Case Name:** | Manage all bakery product information |
| **Actors:** | Admin |
| **Short Description:** | This use case allows the admin can manage bakery product information. The admin can add, update, and delete bakery product information in the system. |
| **Preconditions:** | Admin have login to the website. |
| **Post conditions:** | Admin can manage bakery product information and save new information into the database. |
| **Normal Flow:** | 1. Request the product management page.  2. Select “create”, “update” or “delete” bakery product functions.  3. Input bakery product information (product name, product picture, price, category, and information) in the bakery product form.  4. Click “Save” button to save the updated information.  5. System shall checks validation the input information.  6. System shall display a successful message to the admin.  7. System shall collect bakery product information to the database. (product ID, product name, category, information, price, and product picture)  8. System shall display bakery product information to the product page. |
| **Alternative and Exceptional:** | 2a. In step 2 of the normal flow.  If the admin click create new bakery product information.  2.1 System validates the input information before add in the database.  2.2 Use case resume on step 3 of normal flow.  2b. In step 2 of the normal flow.  If the admin click update bakery product information.  2.1 System validates the input information before add in the database.  2.2 Use case resume on step 3 of normal flow.  2c. In step 2 of the normal flow.  If the admin click delete bakery product.  2.1 Use case resume on step 4 of normal flow. |

**User Case:** View all bakery product information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 07 |
| **Use Case Name:** | View all bakery product information |
| **Actors:** | Member (Admin, Worker, Wholesaler) |
| **Short Description:** | This use case allows member and visitor view all bakery product information in the system. |
| **Preconditions:** | Users have enters to the website. |
| **Post conditions:** | User can know the bakery product information into the database. |
| **Normal Flow:** | 1. Request the product page.  2. Search the bakery product information in the web application.  3. Click “Search” button.  4. System shall verify the filter keywords from the database.  5. System shall request a bakery product list from the database.  6. Select a label from all bakery product list.  7. System shall request bakery product information from the database.  8. System shall display a bakery product information to the product page. |
| **Alternative and Exceptional:** | 4a. In step 4 of the normal flow.  If the input valid with the existing information in the database.  4.1 Use case resume on step 5 of normal flow.  4b. In step 4 of the normal flow.  If the input invalid with the existing information in the database.  4.1 System displays the not found message to the user.  4.2 System shall request the command from the user input again. |

**User Case:** Order bakery product

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 08 |
| **Use Case Name:** | Order bakery product |
| **Actors:** | Wholesaler |
| **Short Description:** | This use case allows the wholesaler can order bakery products by adding, updating, and removing product name, product quantity, and delivery date to their product cart. |
| **Preconditions:** | Wholesaler has login to the website. |
| **Postconditions:** | Wholesaler can send their product cart to the database. |
| **Normal Flow:** | 1. Request an order link in the order page.  2. Select “New order” or “Repeat order information” button functions.  3. Input new order information (product name, product quantity, and delivery date of the order) into the product cart.  4. System shall generate order ID, product ID, and order date for the product cart.  5. System shall display a calculation function for summary total price and discount of all bakery product.  6. Select “Save unfinished” or “Send order” button functions.  7. System shall collect order information (order ID, product name, product quantity, order date and delivery date of order) to the database.  8. System shall display a successful message to the wholesaler. |
| **Alternative and Exceptional:** | 2a. In step 2 of the normal flow.  If the wholesaler selects, a new order  2.1 Use case resume on step 3 of normal flow.  2b. In step 2 of the normal flow.  If the wholesaler selects, repeat order information.  2.1 System shall request all previous order list from the system.  2.2 Select one label from previous delivery date.  2.3 Use case resume on step 4 of normal flow.  6a. In step 6 of the normal flow.  If the wholesaler selects, save unfinished product cart.  6.1 System shall keep incomplete product cart in their account.  6.2 Use case resume on step 8 of normal flow.  6b. In step 6 of the normal flow.  If the wholesaler selects, send order information.  6.1 System shall validates conditional of time before midday.  6.2 Decide to send order information again.  6.3 Use case resume on step 7 of normal flow. |

**User Case:** Manage own order information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 09 |
| **Use Case Name:** | Manage own order information |
| **Actors:** | Wholesaler |
| **Short Description:** | This use case allows the wholesaler manage own ordered information by updating and deleting ordered information in the system. (According to time constraint of 8:00 am – 12:00 pm, same day with current order created) |
| **Preconditions:** | Wholesaler has login to the website and has current ordered information into the database. |
| **Postconditions:** | Wholesaler can update and cancel own order information within the time condition. |
| **Normal Flow:** | 1. Request the order page.  2. System shall display current order information; order ID, product name, product quantity, total price, discount, and delivery date of the order.  3. Select “Update” or “Cancel” ordered information functions.  4. Input update ordered information (product name, product quantity, and delivery date of the order) into the cart.  5. System shall display update calculation of the discount, and total price in the order page.  6. Click “Send order” button function.  7. System shall send update order information to the database.  8. System shall display a successful message updated to the wholesaler. |
| **Alternative and Exceptional:** | 3a. In step 3 of the normal flow.  If the wholesaler selects, update own order information.  3.1 Use case resume on step 4 of normal flow.  3b. In step 3 of the normal flow.  If the wholesaler selects, cancel order information.  3.1 System shall validates conditional of time before midday.  3.2 Accept conditional to cannot cancel order information.  3.3 Use case resume on step 7 of normal flow.  6a. In step 6 of the normal flow.  If the updated order invalid with the conditional of time before midday.  6.1 System displays the conditional message to the wholesaler.  6.2 System shall request confirmation from the wholesaler.  6b. In step 6 of the normal flow.  If the updated order valid with the conditional of time before midday.  6.1 Use case resume on step 7 of normal flow. |

**User Case:** View current order information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 10 |
| **Use Case Name:** | View current order information |
| **Actors:** | Worker and Wholesaler |
| **Short Description:** | This use case allows the worker and wholesaler can view current order information in the system. |
| **Preconditions:** | Worker and Wholesaler have login to the website. |
| **Postconditions:** | Worker and Wholesaler can know current order information from the database. |
| **Normal Flow:** | 1. Request the order information page.  2. Click “View current order” button.  3. System shall request the current order information from the database.  4. System shall display current order information; order ID, product name, product quantity, total price, discount, and delivery date of the order. |
| **Alternative and Exceptional:** | 2a. In step 2 of the normal flow.  If the database have current order information.  2.1 Use case resume on step 3 of normal flow.  2b. In step 2 of the normal flow.  If the database no have current order information.  2.1 System displays the not found message to the screen. |

**User Case:** View total ordered information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 11 |
| **Use Case Name:** | View total ordered information |
| **Actors:** | Worker |
| **Short Description:** | This use case allows the worker can view the total ordered information in the system. |
| **Preconditions:** | Worker has login to the website. |
| **Postconditions:** | Worker can know the total ordered information from the database. |
| **Normal Flow:** | 1. Request the order information page.  2. Search the previous total ordered information in the textbox.  3. Click “Search” button.  4. System shall verify the period from the database.  5. System shall request the delivery date list from the database.  6. Select a label from all delivery date list.  7. Click “View total ordered” button.  8. System shall display the total ordered information to the order information page. |
| **Alternative and Exceptional:** | 4a. In step 4 of the normal flow.  If the input valid with the existing information in the database.  4.1 Use case resume on step 5 of normal flow.  4b. In step 4 of the normal flow.  If the input invalid with the existing information in the database.  4.1 System displays the not found message to the screen. |

**User Case:** Mark order status

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 12 |
| **Use Case Name:** | Mark order Status |
| **Actors:** | Worker |
| **Short Description:** | This use case allows the worker can mark all progressive of order status while the order in the stage. |
| **Preconditions:** | Worker has login to the website. |
| **Postconditions:** | Worker can mark order status into the database. |
| **Normal Flow:** | 1. Request the check order page.  2. System shall display current order list.  3. Select a label from all current order list.  4. System shall display current order information; order ID, product name, product quantity, total price, discount, and delivery date of the order.  5. Select an order status from the drop down list.  6. System shall send updated order status information to the database.  7. System shall display new order status to the current order information. |
| **Alternative and Exceptional:** | - |

**User Case:** View ordered history information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 13 |
| **Use Case Name:** | View ordered history information |
| **Actors:** | Worker and Wholesaler |
| **Short Description:** | This use case allows the worker and wholesaler can view ordered history information in the system. |
| **Preconditions:** | Worker and Wholesaler have login to the website. |
| **Postconditions:** | Worker and Wholesaler can know previous ordered information from the database. |
| **Normal Flow:** | 1. Request the check order page.  2. Select the period of start and end date.  3. Click “Search” button.  4. System shall verify the period with the existing information.  5. System shall request a bakery product list from the database.  6. System shall display previously ordered list.  7. Select a label from all ordered list.  8. System shall request previously ordered information from the database.  9 . System shall display ordered history information to the check order page. |
| **Alternative and Exceptional:** | 4a. In step 4 of the normal flow.  If the input valid with the existing information in the database.  4.1 Use case resume on step 5 of normal flow.  4b. In step 4 of the normal flow.  If the input invalid with the existing information in the database.  4.1 System displays the not found message to the screen. |

**User Case:** Manage Ingredient Information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 14 |
| **Use Case Name:** | Manage Ingredient Information |
| **Actors:** | Admin |
| **Short Description:** | This use case allows the admin can manage all ingredient information. The admin can add, update and delete ingredient information in the system. |
| **Preconditions:** | Admin has login to the website. |
| **Postconditions:** | Admin can manage all ingredient information and save new information into the database. |
| **Normal Flow:** | 1. Request the ingredient management page.  2. Select “create”, “update” or “delete” ingredient information functions.  3. Input ingredient information (Ingredient ID, ingredient name, category, and ingredient picture) into the ingredient form.  4. Click “Save” button.  5. System shall validation the input information.  6. System shall collect new information to the database.  7. System shall display a successful message to the admin. |
| **Alternative and Exceptional:** | 2a. In step 2 of the normal flow.  If the admin clicks, create new ingredient information.  2.1 Use case resume on step 3 of normal flow.  2b. In step 2 of the normal flow.  If the admin clicks, update ingredient information.  2.1 Use case resume on step 3 of normal flow.  2c. In step 2 of the normal flow.  If the admin clicks, delete ingredient.  2.1 Use case resume on step 5 of normal flow.  5a. In step 5 of the normal flow.  If the admin input invalid information type.  5.1 System displays an error message to the admin.  5.2 System shall request the command from the admin to input again.  5b. In step 5 of the normal flow.  If the admin input valid information type.  5.1 Use case resumes on step 6 of normal flow. |

**User Case:** View Ingredient Information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 15 |
| **Use Case Name:** | View Ingredient Information |
| **Actors:** | Admin |
| **Short Description:** | This use case allows the admin can view all ingredient information in the system. |
| **Preconditions:** | Admin has login to the website. |
| **Postconditions:** | Admin can know all ingredient information into the database. |
| **Normal Flow:** | 1. Request the ingredient information page.  2. Search the ingredient information in the web application.  3. Click “Search” button.  4. System shall verify the filter keyword from the database.  5. System shall request the existing ingredient list from the database.  6. Select a label from all ingredient list.  7. System shall request ingredient information from the database. |
| **Alternative and Exceptional:** | 4a. In step 4 of the normal flow.  If the input valid with the existing information in the database.  4.1 Use case resume on step 5 of normal flow.  4b. In step 4 of the normal flow.  If the input invalid with the existing information in the database.  4.1 System displays the not found message to the admin.  4.2 System shall request the command from the admin to input again. |

**User Case:** Manage menu recipe Information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 16 |
| **Use Case Name:** | Manage menu recipe Information |
| **Actors:** | Admin |
| **Short Description:** | This use case allows the admin can manage all menu recipe information. The admin can add, update and delete menu recipe information in the system. |
| **Preconditions:** | Admin has login to the website. |
| **Postconditions:** | Admin can manage all menu recipe information and save new menu recipe information into the database. |
| **Normal Flow:** | 1. Request the menu recipe page.  2. Select “create”, “update” or “delete” menu recipe information functions.  3. Input menu recipe information (product name, ingredient name, and ingredient quantity) into the menu recipe form.  4. Click “Save” button.  5. System shall validation the input information.  6. System shall collect new information to the database.  7. System shall display a successful message to the admin. |
| **Alternative and Exceptional:** | 2a. In step 2 of the normal flow.  If the admin clicks, create new menu recipe information.  2.1 Use case resume on step 3 of normal flow.  2b. In step 2 of the normal flow.  If the admin clicks, update menu recipe information.  2.1 Use case resume on step 3 of normal flow.  2c. In step 2 of the normal flow.  If the admin clicks, delete menu recipe.  2.1 Use case resume on step 4 of normal flow.  5a. In step 5 of the normal flow.  If the admin input invalid information type.  5.1 System displays an error message to the admin.  5.2 System shall request the command from the admin to input again.  5b. In step 5 of the normal flow.  If the admin input valid information type.  5.1 Use case resumes on step 6 of normal flow. |

**User Case:** View menu recipe Information

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 17 |
| **Use Case Name:** | View menu recipe Information |
| **Actors:** | Admin |
| **Short Description:** | This use case allows the admin can view all menu recipe information in the system. |
| **Preconditions:** | Admin has login to the website. |
| **Postconditions:** | Admin can know all menu recipe information into the database. |
| **Normal Flow:** | 1. Request the menu recipe page.  2. Search the menu recipe information in the web application.  3. Click “Search” button.  4. System shall verify the filter keyword from the database.  5. System shall request the existing menu recipe list from the database.  6. Select a label from all menu recipe list.  7. System shall request menu recipe information from the database. |
| **Alternative and Exceptional:** | 4a. In step 4 of the normal flow.  If the input valid with the existing information in the database.  4.1 Use case resume on step 5 of normal flow.  4b. In step 4 of the normal flow.  If the input invalid with the existing information in the database.  4.1 System displays the not found message to the admin.  4.2 System shall request the command from the admin to input again. |

**User Case:** View current ingredient estimation

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 18 |
| **Use Case Name:** | View current ingredient estimation |
| **Actors:** | Worker |
| **Short Description:** | This use case allows the worker can view current ingredient estimation information in the system. |
| **Preconditions:** | Worker has login to the website. |
| **Postconditions:** | Worker can know current ingredient estimation information from the database. |
| **Normal Flow:** | 1. Request the ingredient estimation page.  2. Click “View ingredient estimation” button.  3. System shall request current ingredient estimation information from the database.  4. System shall display current ingredient estimation information to the ingredient estimation page. |
| **Alternative and Exceptional:** | 2a. In step 2 of the normal flow.  If the database have current ingredient estimation information.  2.1 Use case resume on step 3 of normal flow.  2b. In step 2 of the normal flow.  If the database no have current ingredient estimation information.  2.1 System displays the not found message to the screen. |

**User Case:** View previous ingredient estimation

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 19 |
| **Use Case Name:** | View previous ingredient estimation |
| **Actors:** | Worker |
| **Short Description:** | This use case allows the worker can view previous ingredient estimation information in the system. |
| **Preconditions:** | Worker has login to the website. |
| **Postconditions:** | Worker can know previous ingredient estimation information from the database. |
| **Normal Flow:** | 1. Request the ingredient estimation page.  2. Search the previous ingredient estimation information with the existing information in the database.  3. Click “Search” button.  4. System shall verify the period from the database.  5. System shall request previous ingredient estimation list from the database.  6. Select a label from all ingredient estimation list.  7. System shall request ingredient estimation information from the database.  8. System shall display previous ingredient estimation information to the ingredient estimation page. |
| **Alternative and Exceptional:** | 4a. In step 4 of the normal flow.  If the input valid with the existing information in the database.  4.1 Use case resume on step 5 of normal flow.  4b. In step 4 of the normal flow.  If the input invalid with the existing information in the database.  4.1 System displays the not found message to the screen. |

**User Case:** Make a report

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 20 |
| **Use Case Name:** | Make a report |
| **Actors:** | Wholesaler and Worker |
| **Short Description:** | This use case allows the wholesaler and worker can make a report in the report page. |
| **Preconditions:** | Wholesaler and Worker have login to the website. |
| **Postconditions:** | Wholesaler and Worker can save and print a report as a PDF file format. |
| **Normal Flow:** | 1. Request the report page.  2. Click “Make a report” button.  3. System shall retrieve the existing data and summary to the report format.  4. System shall display a report content to the report page.  5. Click “Save” or “Print” a report as a PDF file format. |
| **Alternative and Exceptional:** | 5a. In step 5 of the normal flow.  If the member clicks, save a report.  5.1 System shall request a report file export.  5b. In step 5 of the normal flow.  If the member clicks, print a report.  5.1 System shall request a report print out as a paper document. |

**User Case:** Make a graph

|  |  |
| --- | --- |
| **Use Case ID:** | UC – 21 |
| **Use Case Name:** | Make a graph |
| **Actors:** | Wholesaler and Worker |
| **Short Description:** | This use case allows the wholesaler and worker can make a graph image in the graph page. |
| **Preconditions:** | Wholesaler and Worker have login to the website. |
| **Postconditions:** | Wholesaler and Worker can save and print a graph as a PDF file format. |
| **Normal Flow:** | 1. Request the graph page.  2. Search the products ordered history information with the existing information in the database.  3. Click “Search” button.  4. System shall verify the period from the database.  5. System shall request products ordered list from the database.  6. Select a label from all products ordered list.  7. System shall request products ordered information from the database.  8. System shall display previous products ordered information to the graph page.  9. Click “Make a graph” button.  10. System shall retrieve the statistical data and convert to be a graph.  11. System shall display a graph image to the graph page.  12. Click “Save” or “Print” a graph as a PDF file format. |
| **Alternative and Exceptional:** | 4a. In step 4 of the normal flow.  If the input valid with the existing information in the database.  4.1 Use case resume on step 5 of normal flow.  4b. In step 4 of the normal flow.  If the input invalid with the existing information in the database.  4.1 System displays the not found message to the screen.  12a. In step 12 of the normal flow.  If the member clicks, save a graph.  5.1 System shall request a graph image export.  5b. In step 5 of the normal flow.  If the member clicks, print a graph.  5.1 System shall request a graph print out as a paper document. |

## 5.3 Use Case Specification

**URS-01:** Admin can register to activate member to access the application by create

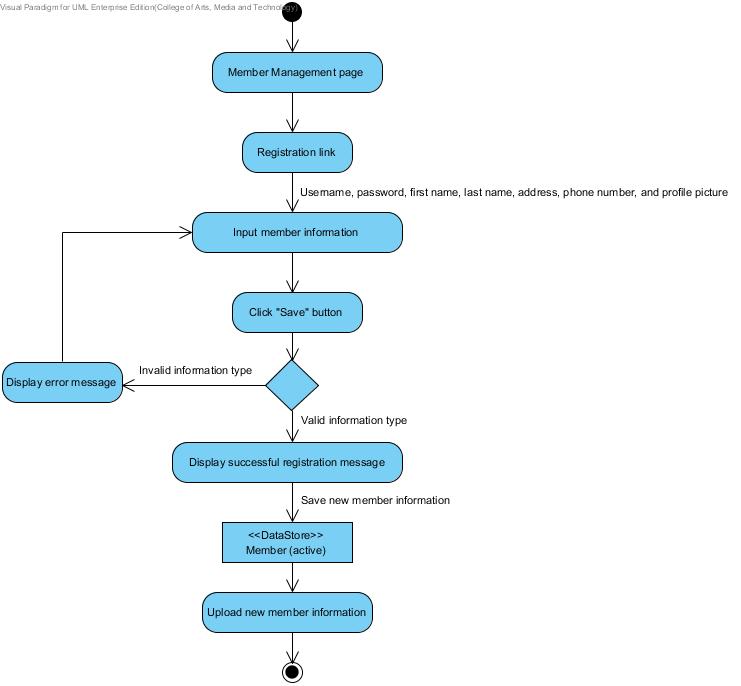
new member information; username, password, first name, last name, phone number, address, role name, and profile picture into the database.

**Precondition:** The admin must login to the system.

**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| First name | Only alphabet is allowed. No special character can used. The length should not over 20 characters. | “Nontra” |
| Last name | Only alphabet is allowed. No special character can used. The length should not over 20 characters. | “Mahachanont” |
| Username | Username should be the e-mail format. | “nontra@gmail.com” |
| Password | Password must be 4-20 characters and can be the small letter, capital letter, and number. | aWefk2343 |
| Phone number | The number which length is exact 10 digits. | “0812345678” |
| Address | Address unlimited characters and contain the small letter, capital letter, and number. | “House No.197 Chroenrat Road, Watgate sub-district, Mueng Chiang Mai district, Chiang Mai province, 5000, Thailand” |
| Profile picture | The profile picture of a member. The picture can be any image file. The size of a picture is unlimited. | vcm_s_kf_m160_155x160.jpg  Me.jpg |

**Action:**



**Figure 4: AD-01 Admin registration to active member**

**Output:** All information of a member will record to the database, and a member will get username and password for login to the website.

**System Requirement Specification:**

**SRS-01:** The system can connect to the phungnoi database and access Member and webpages\_Roles tables.

**SRS-02:** The system shall provide a link for the admin registering new member information in member management page.

**SRS-03:** The system shall provide a registration form with a creating new member page for the admin input username, password, first name, last name, phone number, address, role name, and profile picture.

**SRS-04:** The system shall generate a new member ID by sort all member ID indefinitely.

**SRS-05:** The system shall provide a button for the admin saving new member in the database.

**SRS-06:** The system shall provide a service to validation input information; username, password, first name, last name, phone number, address, role name, and profile picture.

**SRS-07:** The system shall provide a message to display a notification after the registration form has saved.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-09:** The system shall provide a service to upload new member information to display in the member profile interface.

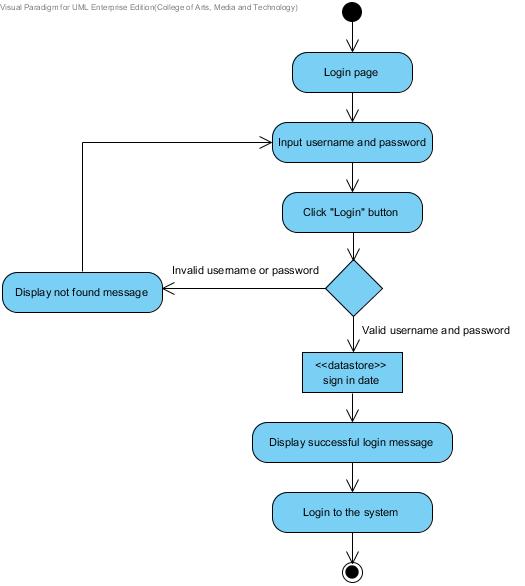
**URS-02:** Member authenticate the web application account.

**Precondition:** All of the members must have username and password before login to the system.

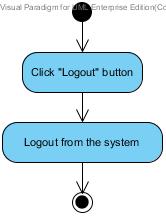
**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Username | Username should be the e-mail format. | “nontra@gmail.com” |
| Password | Password must be 4-20 characters and can be the small letter, capital letter, and number. | aWefk2343 |

**Action:**

****

**Figure 5: AD-02 Member login to the system**

****

**Figure 6: AD-03 Member logout from the system**

**Output:** Member can login to the website and logout from the website.

**System Requirement Specification:**

**SRS-01:** The system can connect to the phungnoi database and access Member and webpages\_Roles tables.

**SRS-10:** The system shall provide a link for the member logging in to the web application on the authentication page.

**SRS-11:** The system shall provide a login form in an authenticate interface for the admin, worker, and wholesaler login members as a different role.

**SRS-12:** The system can hide the characters in the password field.

**SRS-13:** The system shall provide a button for the member logging in to the system.

**SRS-14:** The system shall provide a service to validation input information; username and password.

**SRS-15:** The system shall provide a message to display a notification after the login form has authenticated.

**SRS-16:** The system can take the wholesaler, worker or admin to the main layout page of their roles.

**SRS-17:** The system shall provide a service to keep the last sign in date and time of the member.

**SRS-18:** The system shall provide a link for the member logging out from the web application.

**SRS-19:** The system shall provide a confirmation dialog box after logout link has chosen.

**SRS-20:** The system can take the wholesaler, worker or admin exit from the program.

**URS-03:** Admin can manage all member information; username, password,

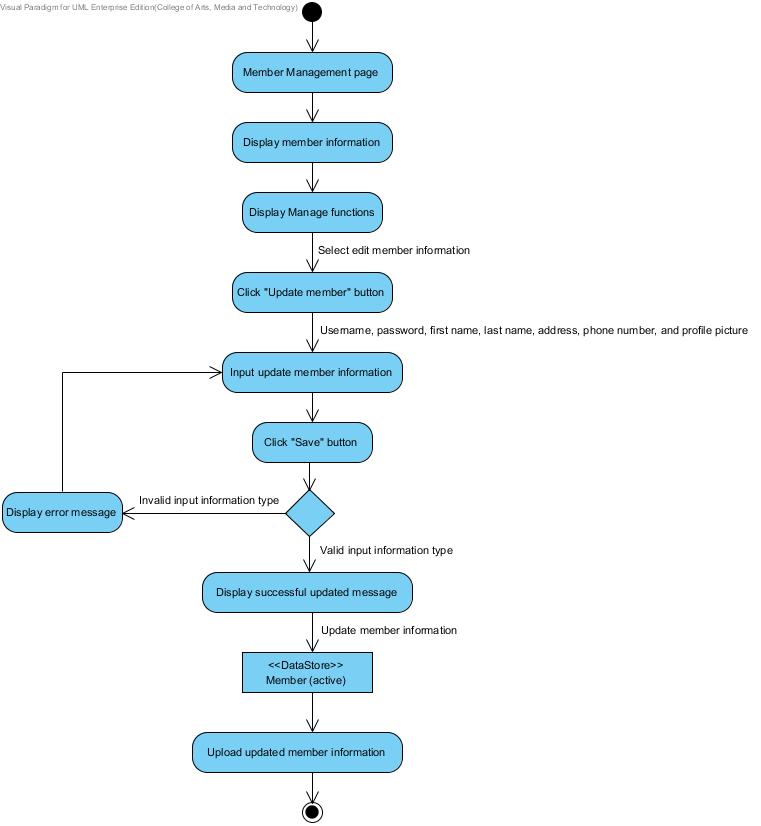
first name, last name, phone number, address, role name, and profile picture in the database.

**Precondition:** The admin must have login to the system and member information has created.

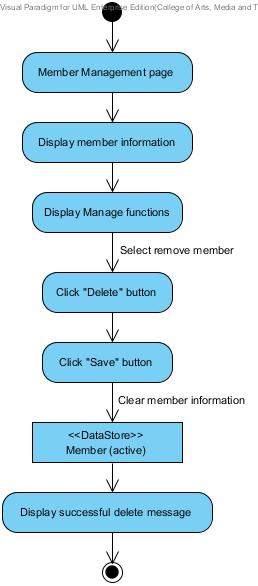
**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| First name | Only alphabet is allowed. No special character can used. The length should not over 20 characters. | “Nontra” |
| Last name | Only alphabet is allowed. No special character can used. The length should not over 20 characters. | “Mahachanont” |
| Username | Username should be the e-mail format. | “nontra@gmail.com” |
| Password | Password must be 4-20 characters and can be the small letter, capital letter, and number. | aWefk2343 |
| Phone number | The number that length is exact 10 digits. | “0812345678” |
| Address | Address unlimited characters and contain the small letter, capital letter, and number. | “House No.197 Chroenrat Road, Watgate sub-district, Mueng Chiang Mai district, Chiang Mai province, 5000, Thailand” |
| Profile picture | The profile picture of a member. The picture can be any image file. The size of a picture is unlimited. | vcm_s_kf_m160_155x160.jpg  Me.jpg |

**Action:**



**Figure 7: AD-04 Admin update member information**



**Figure 8: AD-05 Admin delete member information**

**Output:** All information of a member will record to the database.

**System Requirement Specification:**

**SRS-01:** The system can connect to the phungnoi database and access Member and webpages\_Roles tables.

**SRS-21:** The system shall provide a link for updating member information in the member management interface.

**SRS-22:** The system shall provide the existing member information in member management page for update new member information; username, password, first name, last name, phone number, address, role name, and profile picture.

**SRS-05:** The system shall provide a button for the admin saving new member in the database.

**SRS-06:** The system shall provide a service to validation input information; username, password, first name, last name, phone number, address, role name, and profile picture.

**SRS-23:** The system shall provide a message to display a notification after the update profile form has saved.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-09:** The system shall provide a service to upload new member information to display in the member profile interface.

**SRS-24:** The system shall provide a link for the admin deleting member information in the member management interface.

**SRS-25:** The system shall provide a confirmation dialog box after delete link has chosen.

**SRS-26:** The system shall provide a service to remove member information that chosen from the database.

**SRS-27:** The system shall provide a message to display a notification after the member information has removed.

**URS-04:** Member can update own personal information; username, password, first

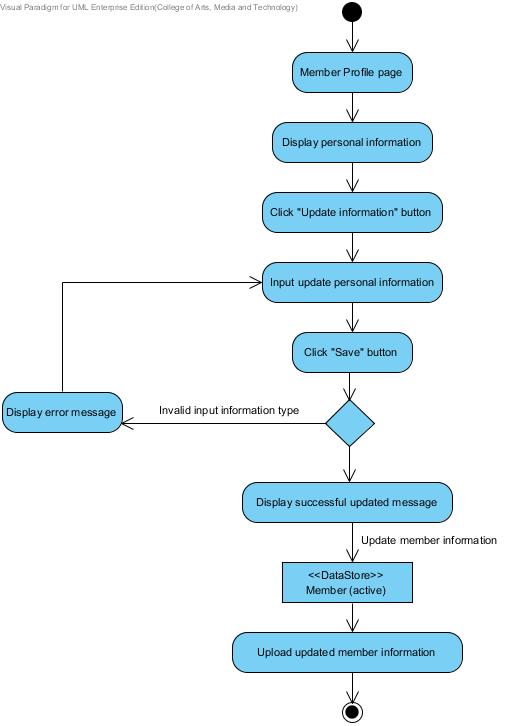
name, last name, phone number, address, and profile picture in the database.

**Precondition:** The member must have login to the system and a member information has created.

**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| First name | Only alphabet is allowed. No special character can used. The length should not over 20 characters. | “Nontra” |
| Last name | Only alphabet is allowed. No special character can used. The length should not over 20 characters. | “Mahachanont” |
| Username | Username should be the e-mail format. | “nontra@gmail.com” |
| Password | Password must be 4-20 characters and can be the small letter, capital letter, and number. | aWefk2343 |
| Phone number | The number that length is exact 10 digits. | “0812345678” |
| Address | Address unlimited characters and contain the small letter, capital letter, and number. | “House No.197 Chroenrat Road, Watgate sub-district, Mueng Chiang Mai district, Chiang Mai province, 5000, Thailand” |
| Profile picture | The profile picture of a member. The picture can be any image file. The size of a picture is unlimited. | vcm_s_kf_m160_155x160.jpg  Me.jpg |

**Action:**



**Figure 9: AD-06 Member update own information**

**Output:** All information of a member will record to the database.

**System Requirement Specification:**

**SRS-01:** The system can connect to the phungnoi database and access Member and webpages\_Roles tables.

**SRS-21:** The system shall provide a link for updating member information in the member management interface.

**SRS-22:** The system shall provide the existing member information in member management page for update new member information; username, password, first name, last name, phone number, address, role name, and profile picture.

**SRS-05:** The system shall provide a button for the admin saving new member in the database.

**SRS-06:** The system shall provide a service to validation input information; username, password, first name, last name, phone number, address, role name, and profile picture.

**SRS-23:** The system shall provide a message to display a notification after the update profile form has saved.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-09:** The system shall provide a service to upload new member information to display in the member profile interface.

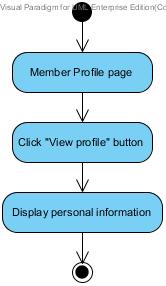
**URS-05:** Member can view own personal information; member ID, username,

password, first name, last name, phone number, address, role name, create date, last sign in date, and profile picture in the database.

**Precondition:** The member must have login to the system

**Input: -**

**Action:**



**Figure 10: AD-07 Member view personal information**

**Output:** The member’s information will display to the profile page.

**System Requirement Specification:**

**SRS-01:** The system can connect to the phungnoi database and access Member and webpages\_Roles tables.

**SRS-28:** The system shall provide a link for viewing member information in the member profile interface.

**SRS-29:** The system shall provide a user interface to display the existing member information which are the member ID, username, password, first name, last name, phone number, address, role name, create date, last sign in date, and profile picture on member profile page.

**URS-06:** Admin can view all member’s information; member ID, username,

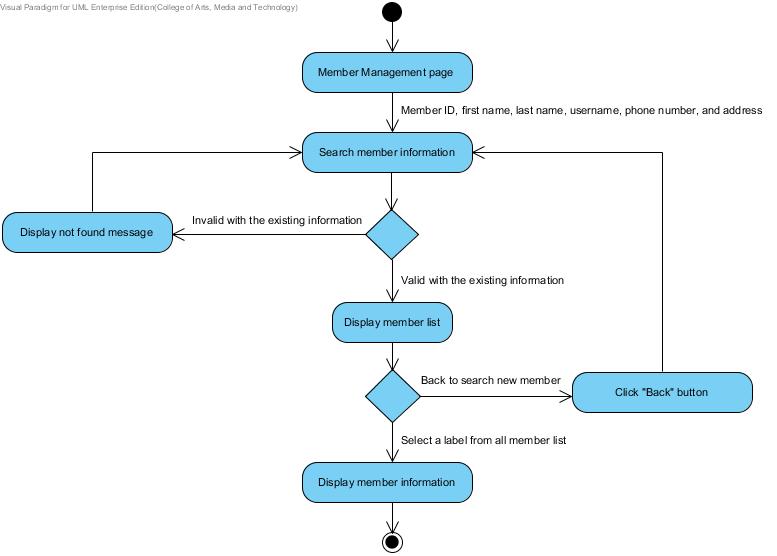
password, first name, last name, phone number, address, role name, create date, last sign in date, and profile picture in the database.

**Precondition:** The admin must have login to the system

**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Member ID | The digit number that identify the member. | 1234 |
| First name | Only alphabet is allowed. No special character can used. The length should not over 20 characters. | “Nontra” |
| Last name | Only alphabet is allowed. No special character can used. The length should not over 20 characters. | “Mahachanont” |
| Username | Username should be the e-mail format. | “nontra@gmail.com” |
| Phone number | The number that length is exact 10 digits. | “0812345678” |
| Address | Address unlimited characters and contain the small letter, capital letter, and number. | “House No.197 Chroenrat Road, Watgate sub-district, Mueng Chiang Mai district, Chiang Mai province, 5000, Thailand” |

**Action:**

****

**Figure 11: AD-08 Admin view all member information**

**Output:** The member’s information will display to the member information page.

**System Requirement Specification:**

**SRS-01:** The system can connect to the phungnoi database and access Member and webpages\_Roles tables.

**SRS-30:** The system shall provide a textbox in member management interface to search filter keywords of the member information; member ID, username, first name, last name, phone number, address, role name, and last sign in date in the database.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-32:** The system shall provide a service to matching input information; member ID, username, first name, last name, phone number, address, role name or last sign in date with the existing database.

**SRS-33:** The system shall provide a user interface to display all member list. In each label, shall display the member ID, username text, first name text, and last name text of member information that related to filter keywords.

**SRS-28:** The system shall provide a link for viewing member information in the member profile interface.

**SRS-29:** The system shall provide a user interface to display the existing member information which are the member ID, username, password, first name, last name, phone number, address, role name, create date, last sign in date, and profile picture on member profile page.

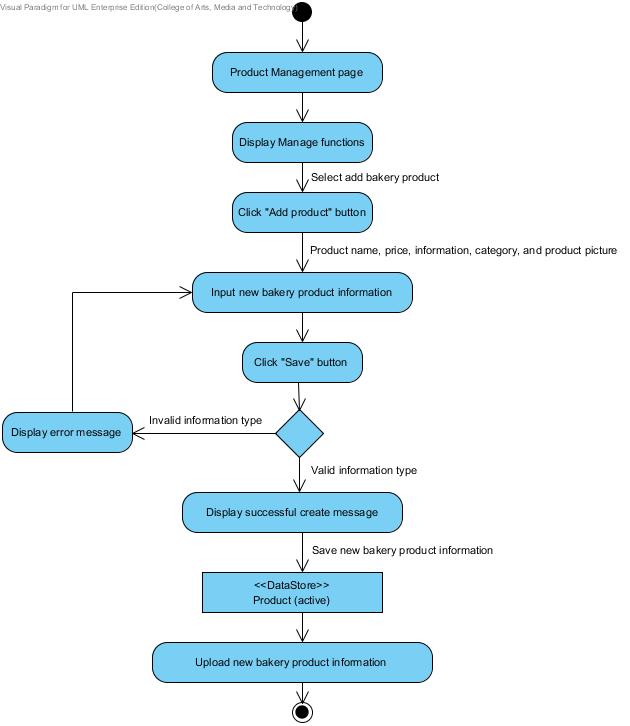
**URS-07:** Admin can manage all product information; product name, price, category, information, and product picture in the database.

**Precondition:** The admin must have login to the system.

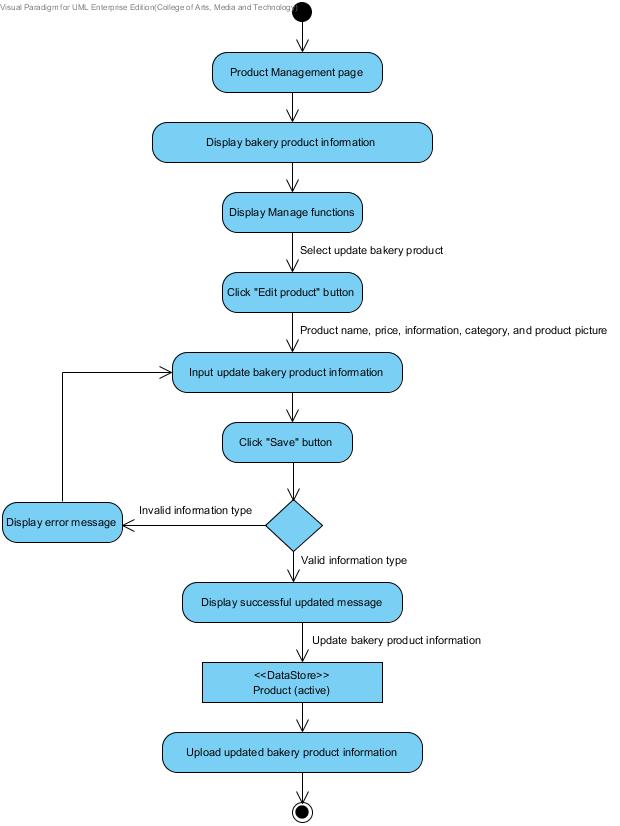
**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Product Name | The product name length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Brownie” |
| Product Information | The product description can explain briefly about the product. The character can be the small letter, capital letter, and number. The length unlimited characters. | “A chocolate brownie is a flat, [baked](http://en.wikipedia.org/wiki/Baking) square or [bar](http://en.wikipedia.org/wiki/Dessert_bar) developed in the [United States](http://en.wikipedia.org/wiki/United_States).” |
| Price | The value of the bakery product. | 29.00 Bath |
| Category | The category length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Bread” |
| Product Picture | The product picture can be any image file. The size of a picture is unlimited. | Aglayan pasta.jpg  Brownie.jpg |

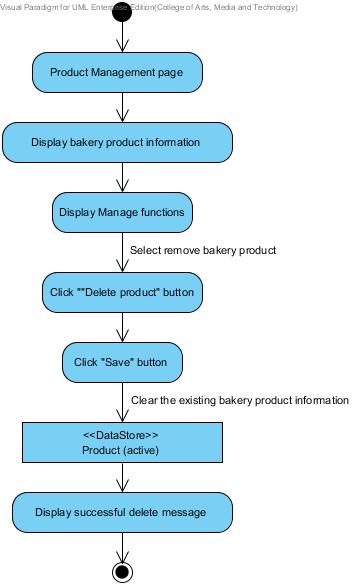
**Action:**



**Figure 12: AD-09 Admin add bakery product information**

****

**Figure 13: AD-10 Admin update bakery product information**



**Figure 14: AD-11 Admin delete bakery product information**

**Output:** Admin can manage all bakery product information and record to the database.

**System Requirement Specification:**

**SRS-34:** The system can connect to the phungnoi database and access webpages\_Roles and Product tables.

**SRS-35:** The system shall provide a link for the admin creating new bakery product information in the product management interface.

**SRS-36:** The system shall provide a bakery product form with a creating new product page for the admin input product name, price, category, information, and product picture.

**SRS-37:** The system shall generate new product ID by sort all product ID indefinitely.

**SRS-38:** The system shall provide a link for the admin updating bakery product information in the product management interface.

**SRS-39:** The system shall provide the existing bakery product information in product management page for the admin update new bakery product information; product name, price, category, information, and product picture into the database.

**SRS-40:** The system shall provide a button for the admin saving new bakery product in the database.

**SRS-41:** The system shall provide a service to validation input information; product name, price, category, information, and product picture.

**SRS-42:** The system shall provide a message to display a notification after the bakery product form has saved.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-43:** The system shall provide a service to upload new bakery product information to display in the product interface.

**SRS-44:** The system shall provide a link for the admin deleting bakery product information from the product interface.

**SRS-25:** The system shall provide a confirmation dialog box after delete link has chosen.

**SRS-45:** The system shall provide a service to remove bakery product information which chosen from the database.

**SRS-46:** The system shall provide a message to display a notification after the bakery product information has removed.

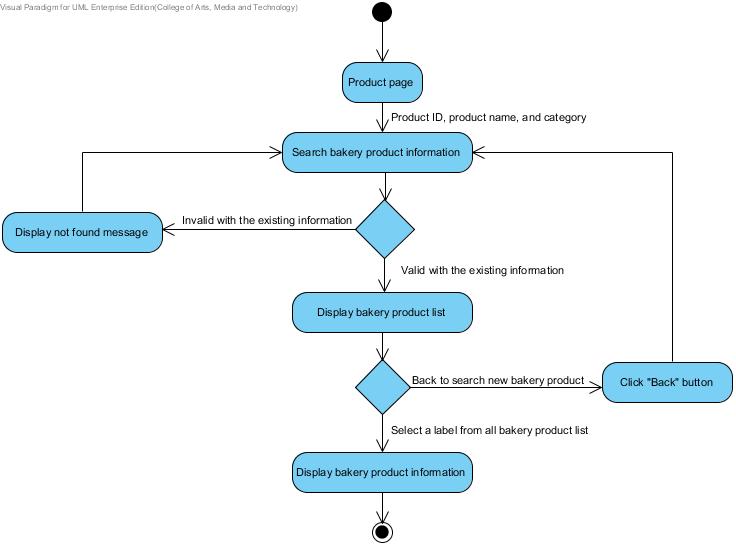
**URS-08:** Member and Visitor can view all bakery product information by search filter keywords of the product ID, product name, price, and category in the database.

**Precondition:** The member and visitor must have enters to the system.

**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Product ID | The digit number that identify the bakery product. | 1234 |
| Product Name | The product name length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Brownie” |
| Category | The category length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Bread” |

**Action:**



**Figure 15: AD-12 Member and Visitor view bakery product information**

**Output:** Member and Visitor can view all bakery product information in the database.

**System Requirement Specification:**

**SRS-47:** The system can connect to the phungnoi database and access Product tables.

**SRS-48:** The system shall provide a textbox in product interface to search filter keywords of the bakery product information; product ID, product name, price, and category in the database.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-49:** The system shall provide a service to match input information; product ID, product name, price, and category with the existing database.

**SRS-50:** The system shall provide a user interface to display all list of bakery product. In each label, shall display the product ID, product name text, and price that related to filter keywords.

**SRS-51:** The system shall provide a link for viewing bakery product information in the product interface.

**SRS-52:** The system shall provide a user interface to display the existing bakery product information which are the product ID, product name, price, category, information, and product picture on the product page.

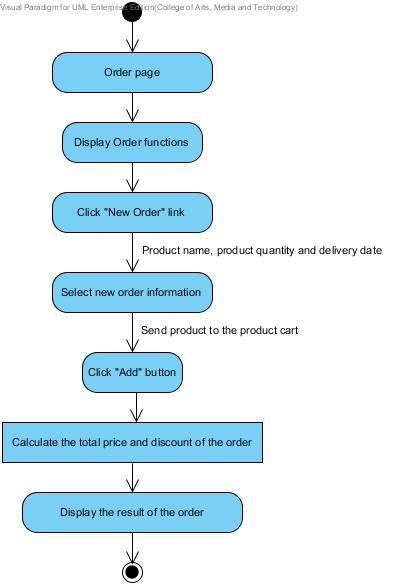
**URS-09:** Wholesaler can order bakery products from the existing bakery product in the system. An order can select by input product name, product quantity, and delivery date.

**Precondition:** The wholesaler must has login to the system.

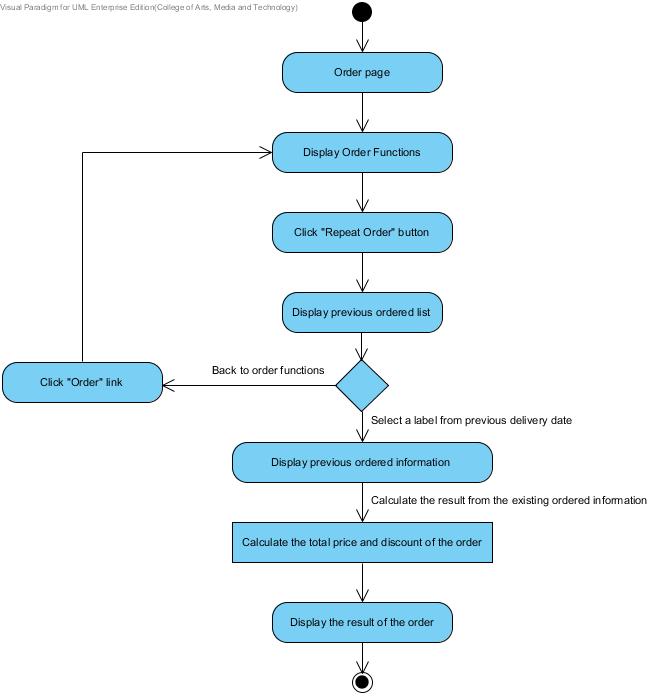
**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Product Name | The product name length must not be longer than 40 characters. Only alphabet is allowed. No special character can be used. | “Brownie” |
| Product Quantity | The amount of number to indicate bakery product in the order. | 1200 |
| Delivery Date | The date to check out and schedule for receiving the order. | 17 October 2014 |

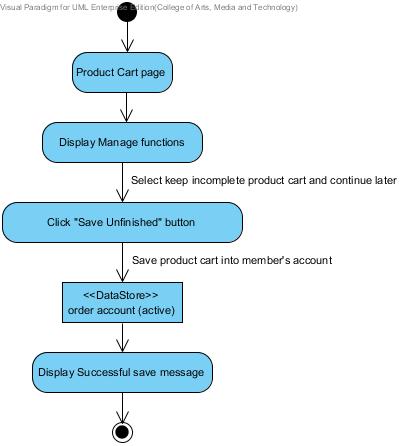
**Action:**

****

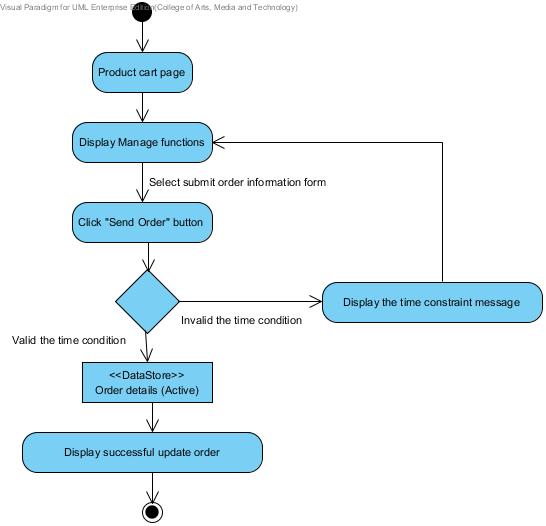
**Figure 16: AD-13 Wholesaler order new bakery product**

****

**Figure 17: AD-14 Wholesaler repeat previous ordered information**

****

**Figure 18: AD-15 Wholesaler save unfinished product list**

****

**Figure 19: AD-16 Wholesaler send order**

**Output:** Wholesaler can view ordering the bakery product and send the order information to the database.

**System Requirement Specification:**

**SRS-53:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Product, Order, OrderDetail, and Status tables.

**SRS-54:** The system shall provide a link for the wholesaler ordering in the order interface within 8:00 am – 12:00 pm of daily.

**SRS-55:** The system shall provide a product cart in order page for the wholesaler select product name, product quantity, and delivery date.

**SRS-56:** The system shall generate the other information from product name field consist of the product ID and price in each label of bakery product list.

**SRS-57:** The system shall provide a button for the wholesaler repeating previous ordered information from the ordered history interface.

**SRS-58:** The system shall provide a user interface to display all ordered history list. In each label, shall display the delivery date, order ID, order status text and total price which related to filter keywords.

**SRS-59:** The system shall provide a link for the wholesaler requesting to use previously ordered list in the order interface.

**SRS-60:** The system shall provide a confirmation dialog box after previously ordered link has chosen.

**SRS-61:** The system shall provide a service to reuse all bakery product name and quantity information from the chosen previously ordered.

**SRS-62:** The system shall provide a calculating function to compute the total price and discount of each order.

**SRS-63:** The system shall provide a user interface to display the result of the total price and discount from all bakery product list.

**SRS-64:** The system shall provide a button for the wholesaler sending finished product cart.

**SRS-65:** The system shall provide a service to validation incoming order follow the time condition. (According to time constraint of 8:00 am – 12:00 pm in daily)

**SRS-66:** The system shall provide a message to display a notification after the product cart has sent.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-67:** The system shall provide a service to upload new ordered information to display in the order information interface.

**SRS-68:** The system shall provide a button for the wholesaler saving unfinished product cart in their account.

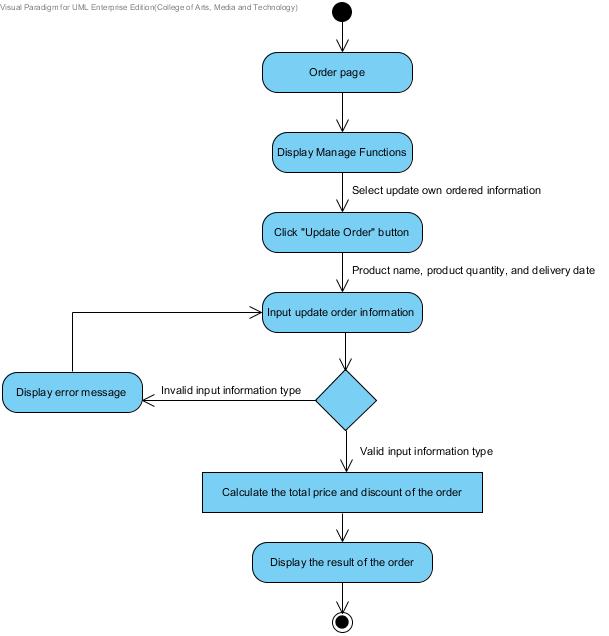
**URS-10:** Wholesaler can manage own ordered information; product name, product quantity, and delivery date. (According to time constraint of 8:00 am – 12:00 pm, same day with current order created)

**Precondition:** The wholesaler must has login to the system and current ordered information has created.

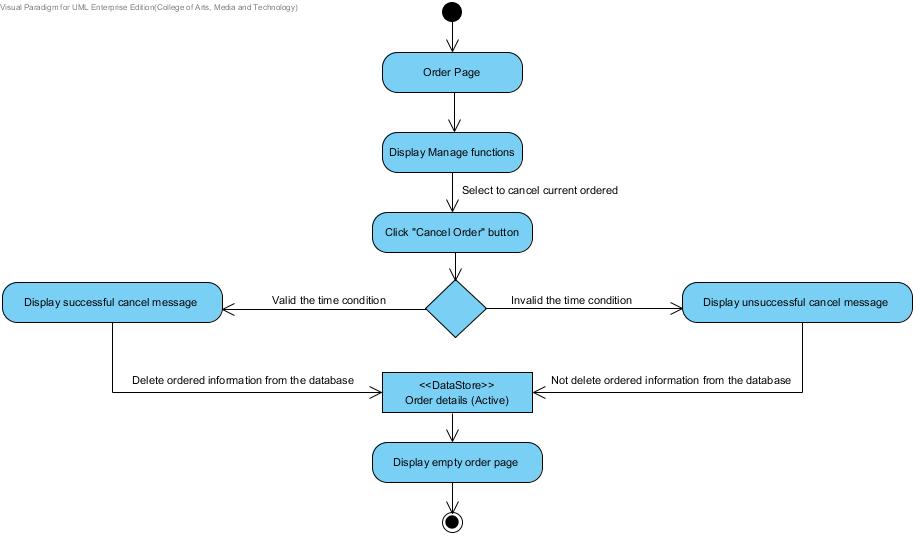
**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Product Name | The product name length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Brownie” |
| Product Quantity | The amount of number to indicate bakery product in the order. | 1200 |
| Delivery Date | The date to check out and schedule for receiving the order. | 17 October 2014 |

**Action:**

****

**Figure 20: AD-17 Wholesaler update own ordered information**

****

**Figure 21: AD-18 Wholesaler cancel own ordered**

**Output:** Wholesaler can manage own order information, and all updated information will record to the database.

**System Requirement Specification:**

**SRS-53:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Product, Order, OrderDetail, and Status tables.

**SRS-69:** The system shall provide a link for the wholesaler updating current order information in the order interface.

**SRS-70:** The system shall provide a user interface to display current order information in the order page for the wholesaler update order information; product name, product quantity, and delivery date into the database.

**SRS-56:** The system shall generate the other information from product name field consist of the product ID and price in each label of bakery product list.

**SRS-62:** The system shall provide a calculating function to compute the total price and discount of each order.

**SRS-63:** The system shall provide a user interface to display the result of the total price and discount from all bakery product list.

**SRS-64:** The system shall provide a button for the wholesaler sending finished product cart.

**SRS-65:** The system shall provide a service to validation incoming order follow the time condition. (According to time constraint of 8:00 am – 12:00 pm in daily)

**SRS-66:** The system shall provide a message to display a notification after the product cart has sent.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-67:** The system shall provide a service to upload new ordered information to display in the order information interface.

**SRS-71:** The system shall provide a cancel link for the wholesaler canceling current order information in the order interface.

**SRS-72:** The system shall provide a confirmation dialog box after cancel link has chosen.

**SRS-73:** The system shall provide a service to remove current order information from the database.

**SRS-74:** The system shall provide a message to display a notification after the current order information has canceled.

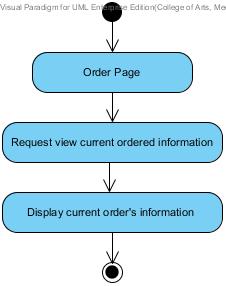
**URS-11:** Wholesaler can view their current order information; order ID, first name,

last name, product ID, product name, product quantity, price, total price, discount, order date, delivery date, and order status.

**Precondition:** The wholesaler must has login to the system.

**Input:** -

**Action:**



**Figure 22: AD-19 Wholesaler view current order information**

**Output:** Wholesaler can view current order information in the database.

**System Requirement Specification:**

**SRS-75:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Status, and OrderDetail tables.

**SRS-76:** The system shall provide a link for viewing current order information in the order information interface.

**SRS-77:** The system shall provide a user interface to display current order information which are the order ID, member ID, first name, last name, product ID, product name, product quantity, order status, delivery date, order date, total price and discount on the current order information page.

**URS-12:** Worker can view current order information of all wholesalers by search

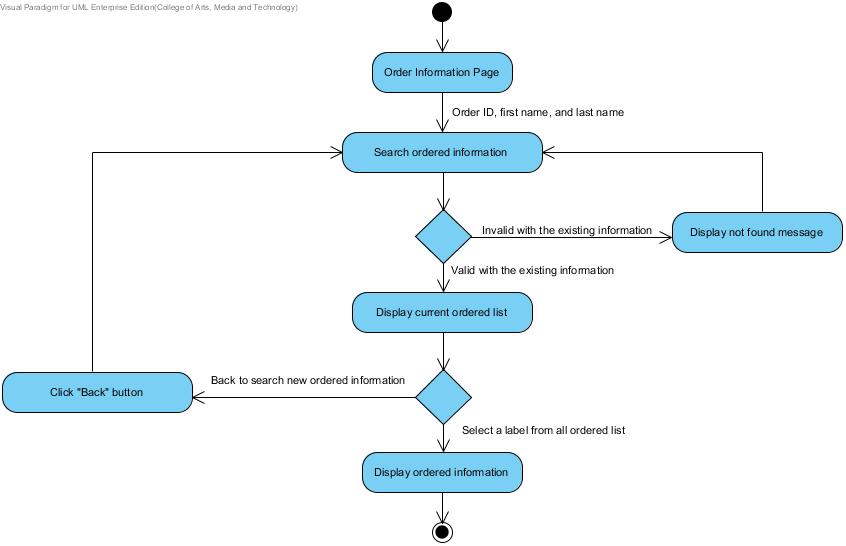
filter keywords of the order ID, member ID, first name, last name, order status, and delivery date in the database.

**Precondition:** The worker must has login to the system.

**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Order ID | The number to identify the ordered. | 1234 |
| Member ID | The number to identify the member. | 5678 |
| First name | The wholesaler’s first name. | “Nontra” |
| Last name | The wholesaler’s last name. | “Mahachanont” |
| Order status | The order status can identify progressive of the order. | “Preparing” |
| Delivery Date | The date to check out and schedule for receive the order. | 17 October 2014 |

**Action:**



**Figure 23:** AD-20 Worker view **current order information**

**Output:** Worker can view current order information in the database.

**System Requirement Specification:**

**SRS-75:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Status, and OrderDetail tables.

**SRS-78:** The system shall provide a textbox in order interface to search the order information; order ID, member ID, first name, last name, and order status.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-79:** The system shall provide a service to match input information; order ID, member ID, first name, last name, and order status with the existing current order information in the database.

**SRS-80:** The system shall provide a user interface to display all current order list. In each label, shall display the order ID, first name text, last name text, total price, and order status text which related to filter keywords.

**SRS-76:** The system shall provide a link for viewing current order information in the order information interface.

**SRS-77:** The system shall provide a user interface to display current order information which are the order ID, member ID, first name, last name, product ID, product name, product quantity, order status, delivery date, order date, total price and discount on the current order information page.

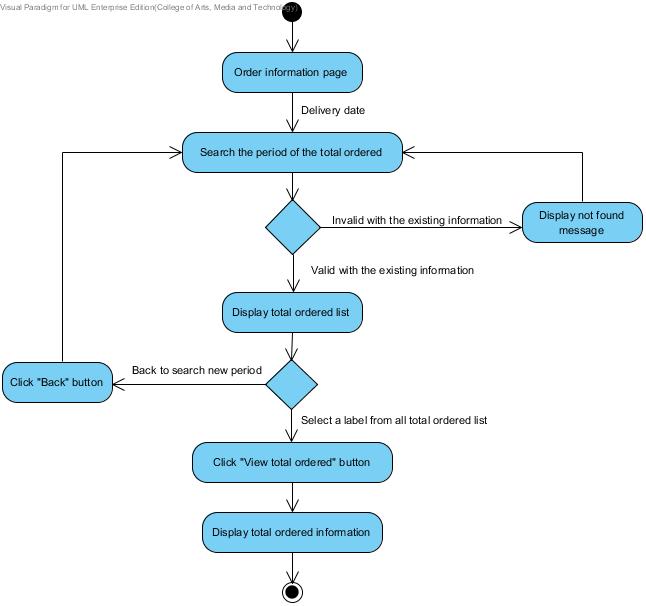
**URS-13:** Worker can view the summary of total bakery product are ordered in daily.

**Precondition:** The worker must has login to the system.

**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Delivery Date | The date to check out and schedule for receive the order. | 17 October 2014 |

**Action:**



**Figure 24: AD-21 Worker view total ordered**

**Output:** Worker can know the total ordered result from the delivery date in daily.

**System Requirement Specification:**

**SRS-81:** The system can connect to the phungnoi database and access webpages\_Roles and OrderDetail tables.

**SRS-82:** The system shall provide a user interface to display the current total ordered information which are the product ID, product name, total number, and summary price on the total ordered page.

**SRS-83:** The system shall provide a textbox in the total ordered interface to search the filter keywords of the day, month, and year of the previously total ordered result.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-84:** The system shall provide a service to match the period of the day, month, and year in the field textbox with the previously total ordered information in the database.

**SRS-85:** The system shall provide a user interface to display all total ordered list. In each label, shall display the delivery date and summary price which related to filter keywords.

**SRS-86:** The system shall provide a link for the worker viewing the total number of all bakery product and summary price in the total ordered interface.

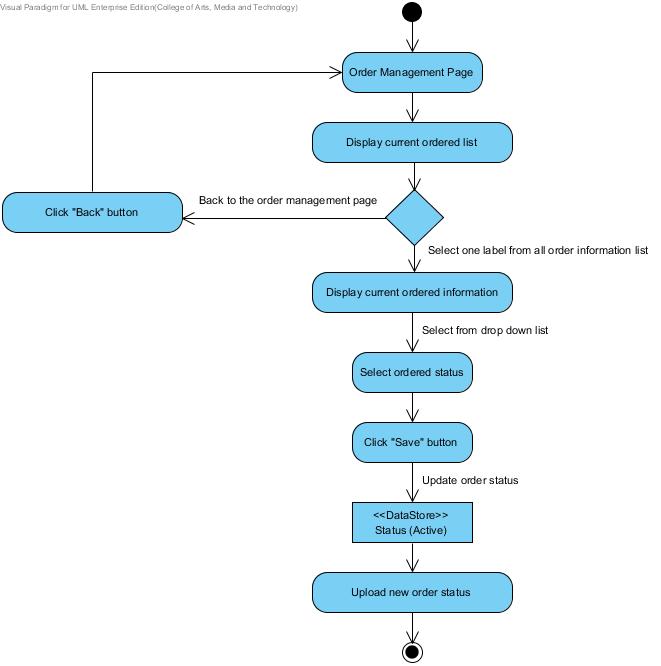
**SRS-87:** The system shall provide a user interface to display the total ordered information which chosen by the worker.

**URS-14:** Worker can mark progressive of all order status while the order in the stage.

**Precondition:** The worker must has login to the system.

**Input: -**

**Action:**



**Figure 25: AD-22 Worker mark order status**

**Output:** Worker can mark progressive of the order status and record to the database.

**System Requirement Specification:**

**SRS-73:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Status, and OrderDetail tables.

**SRS-75:** The system shall provide a user interface to display current order information which are the order ID, member ID, first name, last name, product ID, product name, product quantity, order status, delivery date, order date, total price and discount on the current order information page.

**SRS-86:** The system shall provide a drop down list for the worker selecting progressive of order status in the order management interface.

**SRS-87:** The system shall provide a button for the worker saving changed order status in the order management interface.

**SRS-88:** The system shall provide a message to display a notification after the order status in the preparation stage has changed.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-89:** The system shall provide a service to upload new progressive of order status to display in the order information interface.

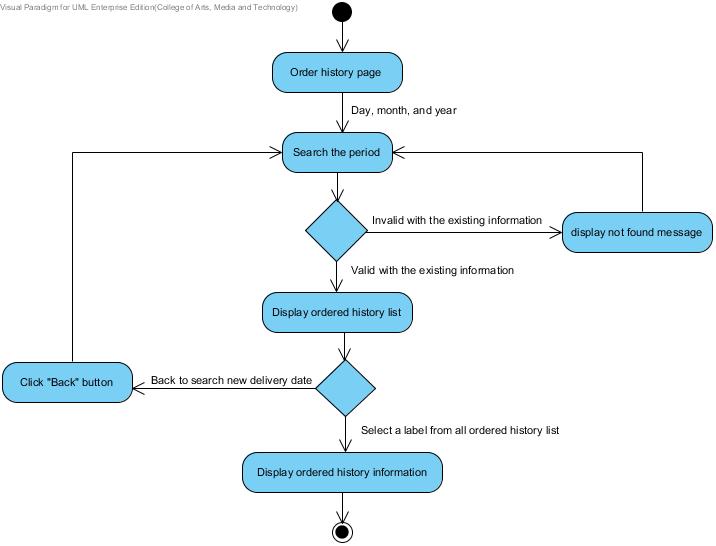
**URS-15:** Wholesaler can view their ordered history by choosing a period of the day, month, and year to searching the previous ordered in the database.

**Precondition:** The wholesaler must has login to the system.

**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Delivery Date | The date to check out and schedule for receive the order. | 17 October 2014 |

**Action:**



**Figure 26: AD-23 Wholesaler view ordered history**

**Output:** Wholesaler can know the ordered history information from the database.

**System Requirement Specification:**

**SRS-73:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Status, and OrderDetail tables.

**SRS-90:** The system shall provide a textbox in order interface to search the filter keywords of the day, month, and year of the delivery date.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-91:** The system shall provide a service to match the period of the day, month, and year in field textbox with the ordered history information in the database.

**SRS-55:** The system shall provide a user interface to display all ordered history list. In each label, shall display the delivery date, order ID, order status text and total price which related to filter keywords.

**SRS-92:** The system shall provide a link for viewing the ordered history information in the ordered history interface.

**SRS-93:** The system shall provide a user interface to display the ordered history information which are the member ID, first name, last name, order ID, product ID, product name, product quantity, total price, discount, order status and delivery date on the ordered history page.

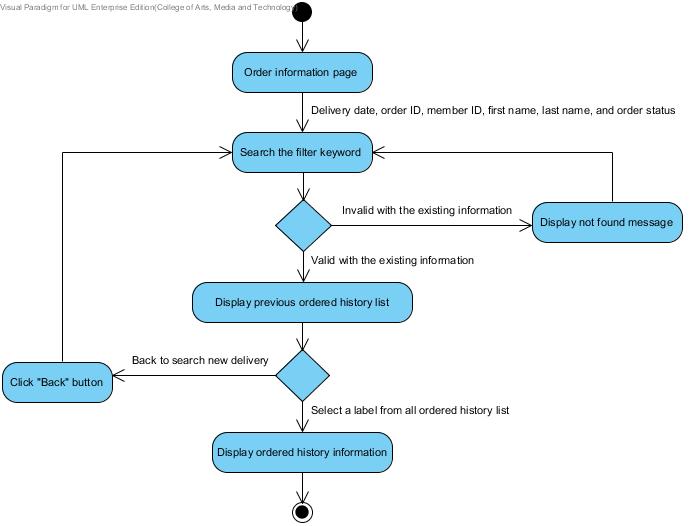
**URS-16:** Worker can view all ordered history by choosing a period of the day, month, and year with another filter keywords. For an example member ID, first name, last name, delivery date, order status, and order ID to searching the result in the database.

**Precondition:** The worker must has login to the system.

**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Delivery Date | The date to check out and schedule for receive the order. | 17 October 2014 |
| Order ID | The number to identify the ordered. | 1234 |
| Member ID | The number to identify the member. | 5678 |
| First name | The wholesaler’s first name. | “Nontra” |
| Last name | The wholesaler’s last name. | “Mahachanont” |
| Order status | The order status can identify progressive of the order. | “Preparing” |

**Action:**

****

**Figure 27: AD-24 Worker view all ordered history**

**Output:** Worker can know all ordered history information from the database.

**System Requirement Specification:**

**SRS-73:** The system can connect to the phungnoi database and access Member, webpages\_Roles, Status, and OrderDetail tables.

**SRS-94:** The system shall provide the first textbox in the order interface to search the filter keywords of the day, month, and year. And the second textbox to search ordered history information; member ID, first name, last name, delivery date, order status, and order ID.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-95:** The system shall provide a service to match the filter keywords in the first textbox and second textbox with the ordered history information in the database.

**SRS-55:** The system shall provide a user interface to display all ordered history list. In each label, shall display the delivery date, order ID, order status text and total price which related to the filter keywords.

**SRS-92:** The system shall provide a link for viewing the ordered history information in the ordered history interface.

**SRS-93:** The system shall provide a user interface to display the ordered history information which are the member ID, first name, last name, order ID, product ID, product name, product quantity, total price, discount, order status and delivery date on the ordered history page.

**URS-17:** Admin can manage all ingredient information; ingredient name, category,

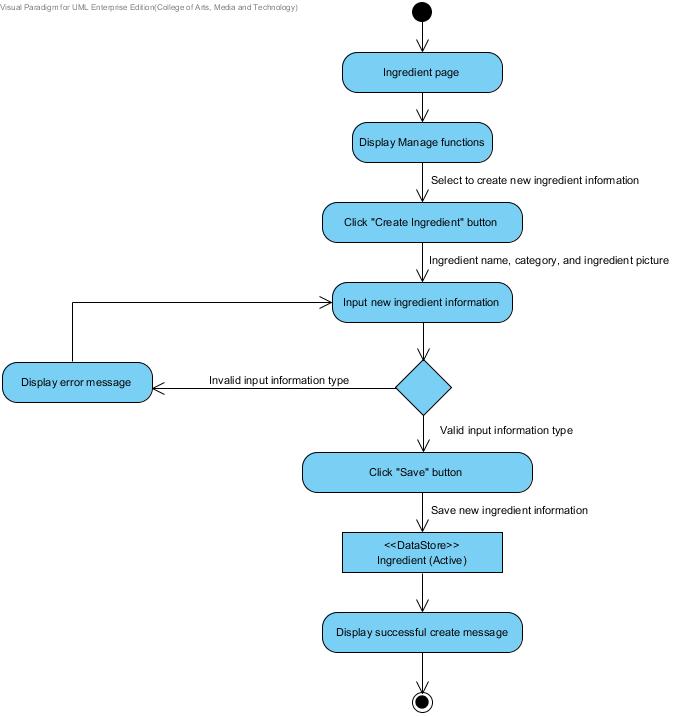
and ingredient picture in the database.

**Precondition:** The admin must has login to the system.

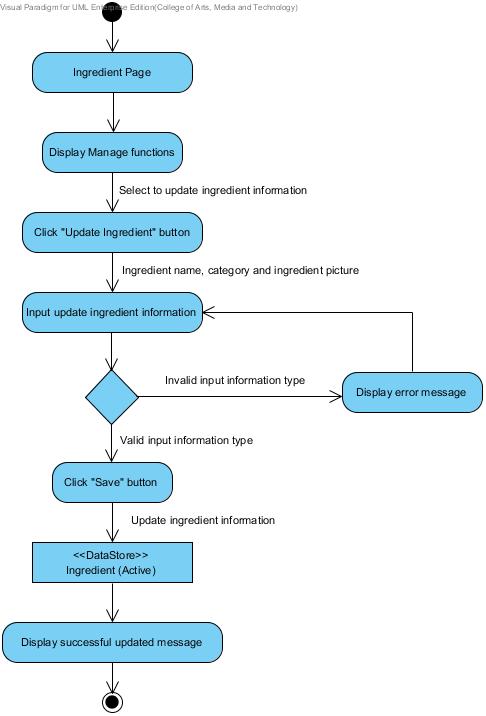
**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Ingredient ID | The number indicate the ingredient. | 123456 |
| Ingredient Name | The ingredient name length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Garlic” |
| Category | The category length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Spices” |
| Ingredient Picture | The ingredient picture can be any image file. The size of a picture is unlimited | http://upload.wikimedia.org/wikipedia/commons/thumb/f/fb/Allium_sativum._Restra_de_allos_de_Oroso-_Galiza.jpg/1024px-Allium_sativum._Restra_de_allos_de_Oroso-_Galiza.jpg  Garlic.jpg |

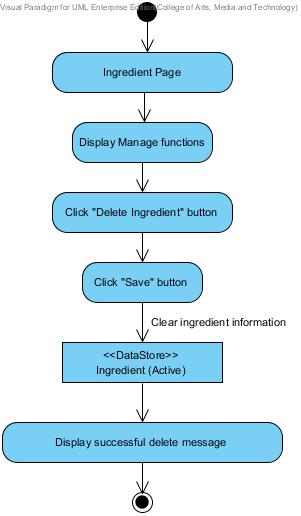
**Action:**

****

**Figure 27: AD-24 Admin create ingredient information**

****

**Figure 28: AD-25 Admin update ingredient information**



**Figure 29: AD-26 Admin delete ingredient information**

**Output:** Admin can manage all bakery product information and record to the database.

**System Requirement Specification:**

**SRS-96:** The system can connect to the phungnoi database and access webpages\_Roles and Ingredient tables.

**SRS-97:** The system shall provide a link for the admin creating new ingredient information in the ingredient management interface.

**SRS-98:** The system shall provide an ingredient form in the ingredient management page for the admin input ingredient information; ingredient name, category, and ingredient picture.

**SRS-99:** The system shall provide a link for the admin updating ingredient information in the ingredient interface.

**SRS-100:** The system shall provide a user interface to display the existing ingredient information in the ingredient management page for the admin update new ingredient information; ingredient name, category, and ingredient picture.

**SRS-101:** The system shall generate the other information from the ingredient name field consist of the ingredient ID in each label of the ingredient list.

**SRS-102:** The system shall provide a button for the admin saving new ingredient in the database.

**SRS-103:** The system shall provide a service to validation input information; ingredient name, category, and ingredient picture.

**SRS-104:** The system shall provide a message to display a notification after the ingredient form has saved.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-105:** The system shall provide a link for the admin deleting ingredient information in the ingredient information interface.

**SRS-106:** The system shall provide a confirmation dialog box after delete link has chosen.

**SRS-107:** The system shall provide a service to remove ingredient information which chosen from the database.

**SRS-108:** The system shall provide a message to display a notification after the ingredient information has removed.

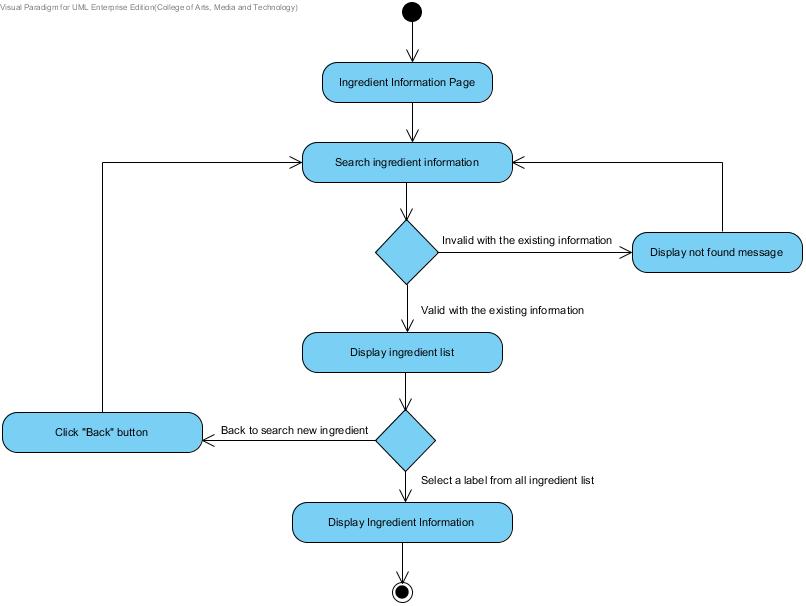
**URS-18:** Admin can view all ingredient information by search filter keywords of the ingredient ID, ingredient name, and category in the database.

**Precondition:** The admin must has login to the system and the ingredient information has created in the system.

**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Ingredient ID | The number indicate the ingredient. | 123456 |
| Ingredient Name | The ingredient name length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Garlic” |
| Category | The category length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Spices” |

**Action:**

****

**Figure 30: AD-27 Admin view ingredient information**

**Output:** Admin can view all ingredient information in the database.

**System Requirement Specification:**

**SRS-96:** The system can connect to the phungnoi database and access webpages\_Roles and Ingredient tables.

**SRS-109:** The system shall provide a textbox in the ingredient interface to search the filter keywords of the ingredient information; ingredient ID, ingredient name, and category.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-120:** The system shall provide a service to match the input information; ingredient ID, ingredient name, category, and ingredient picture with the existing ingredient information in the database.

**SRS-111:** The system shall provide a user interface to display all ingredient list. In each label, shall display the ingredient name text and ingredient ID which related to the filter keywords.

**SRS-112:** The system shall provide a user interface to display the existing ingredient information which are the ingredient ID, ingredient name, category, and ingredient picture on the ingredient information page.

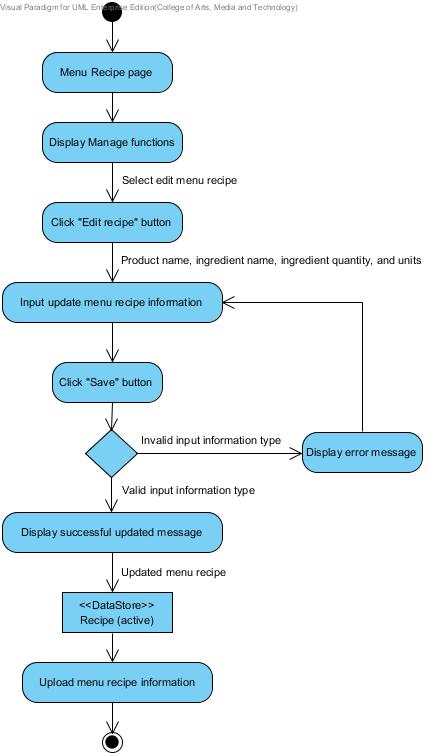
**URS-19:** Admin can manage all menu recipe information; product name, ingredient name, ingredient quantity, and units in the database.

**Precondition:** The admin must has login to the system.

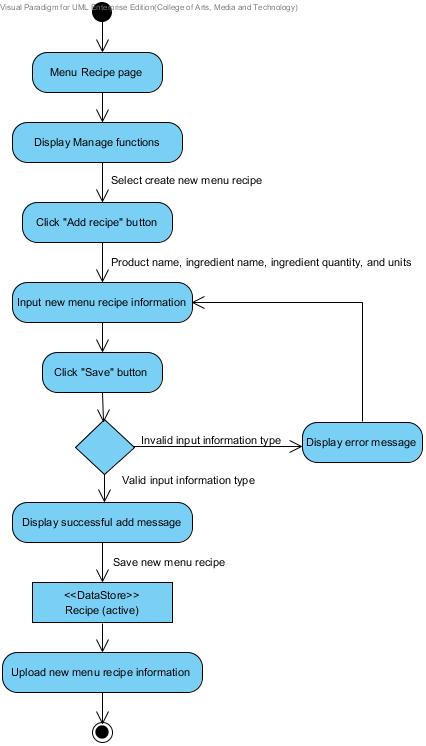
**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Product Name | The product name length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Brownie” |
| Ingredient Name | The ingredient name length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Garlic” |
| Ingredient Quantity | The amount of number to indicate ingredients of the bakery product. | 120 |
| Units | The units to measure an ingredient. | Gram |

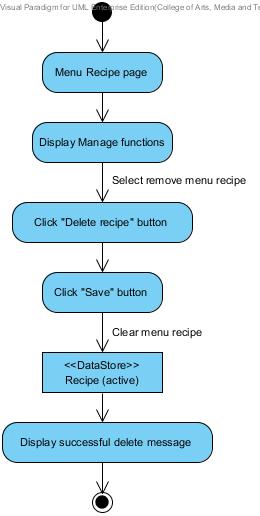
**Action:**



**Figure 31: AD-28 Admin add menu recipe**



**Figure 32: AD-29 Admin edit menu recipe**



**Figure 33: AD-30 Admin delete menu recipe**

**Output:** Admin can manage all menu recipe information and record in the database.

**System Requirement Specification:**

**SRS-113:** The system can connect to the phungnoi database and access webpages\_Roles, Product, Ingredient, Recipe, and Units tables.

**SRS-114:** The system shall provide a link for the admin creating menu recipe information in the menu recipe interface.

**SRS-115:** The system shall provide a menu recipe form in the menu recipe page for the admin input menu recipe information; product name, ingredient name, ingredient quantity, and units.

**SRS-116:** The system shall provide a link for the admin updating menu recipe information in the menu recipe interface.

**SRS-117:** The system shall provide a user interface to display the existing menu recipe information in the menu recipe page for the admin update new menu recipe information; product name, ingredient name, ingredient quantity, and units.

**SRS-118:** The system shall generate the other information from the product name and ingredient name field consist of the product ID, information, category and ingredient ID in each menu recipe.

**SRS-119:** The system shall provide a button for the admin saving new menu recipe in the database.

**SRS-120:** The system shall provide a service to validation input information; product name, ingredient name, ingredient quantity, and units.

**SRS-121:** The system shall provide a message to display a notification after menu recipe form has saved.

**SRS-08:** The system shall provide a service to store new information into the database.

**SRS-122:** The system shall provide a link for deleting menu recipe information in the menu recipe interface.

**SRS-123:** The system shall provide a confirmation dialog box after delete link has chosen.

**SRS-124:** The system shall provide a service to remove menu recipe information which chosen from the database.

**SRS-125:** The system shall provide a message to display a notification after the menu recipe information has removed.

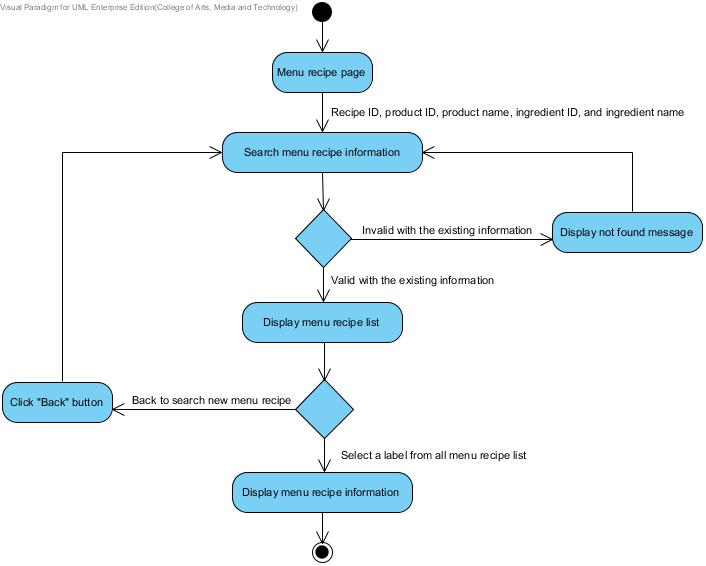
**URS-20:** Admin can view all menu recipe information by search filter keywords of the recipe ID, product ID, product name, ingredient ID, and ingredient name in the database.

**Precondition:** The admin must has login to the system.

**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Recipe ID | The number indicate the menu recipe. | 123456 |
| Product ID | The number indicate the product. | 789 |
| Product Name | The product name length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Brownie” |
| **Name** | **Description** | **Example** |
| Ingredient ID | The number indicate the ingredient. | 6789 |
| Ingredient Name | The ingredient name length must not be longer than 40 characters. Only alphabet is allowed. No special character can used. | “Garlic” |

**Action:**



**Figure 34: AD-31 Admin view menu recipe**

**Output:** Admin can view all menu recipe information in the database.

**System Requirement Specification:**

**SRS-113:** The system can connect to the phungnoi database and access webpages\_Roles, Product, Ingredient, Recipe, and Units tables.

**SRS-126:** The system shall provide a textbox in the menu recipe interface to search filter keywords of the menu recipe information; recipe ID, product ID, product name, ingredient ID, and ingredient name.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-127:** The system shall provide service to match the input information; recipe ID, product ID, product name, ingredient ID, and ingredient name.

**SRS-128:** The system shall provide a user interface to display all menu recipe list. In each label, shall display the product name text and recipe ID on menu recipe list.

**SRS-129:** The system shall provide a user interface to display the existing menu recipe information which are the recipe ID, product ID, product name, product picture, information, category, ingredient ID, ingredient name, ingredient quantity, and units on the menu recipe information page.

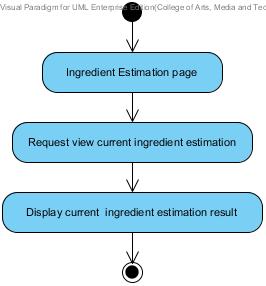
**URS-21:** Worker can view the current ingredient estimation result; product ID,

product name, ingredient ID, ingredient name, ingredient quantity, and units.

**Precondition:** The worker must has login to the system.

**Input: -**

**Action:**



**Figure 35: AD-32 Worker view current ingredient estimation**

**Output:** Worker can view current ingredient estimation information in the database.

**System Requirement Specification:**

**SRS-130:** The system can connect to the phungnoi database and access webpages\_Roles, OrderDetail, Recipe and Units tables.

**SRS-131:** The system shall provide a button for the admin viewing the ingredient usage in the ingredient estimation interface.

**SRS-132:** The system shall provide a calculating function to compute the ingredient usage by using information from the menu recipe and total ordered.

**SRS-133:** The system shall provide a user interface to display the ingredient estimation information which are the product ID, product name, ingredient ID, ingredient name, ingredient quantity, and units on the ingredient estimation page.

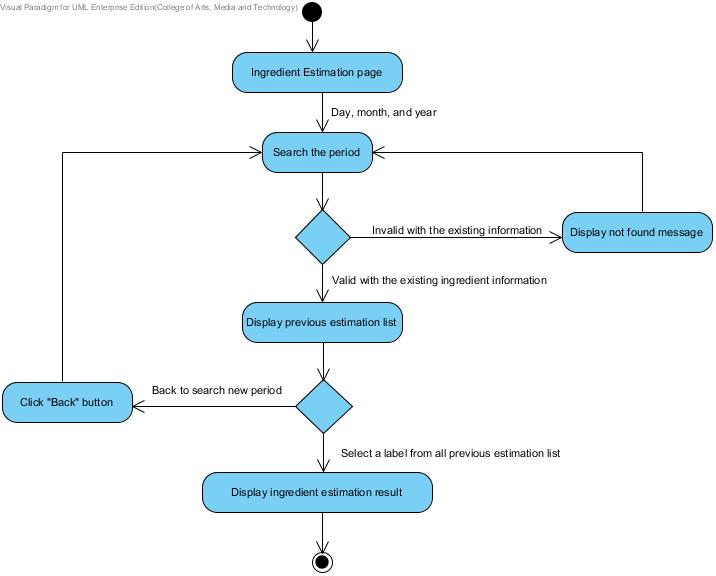
**URS-22:** Worker can view previous ingredient estimation by choosing a period of the day, month, and year.

**Precondition:** The worker must has login to the system.

**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Delivery Date | The date to check out and schedule for receive the order. | 17 October 2014 |

**Action:**



**Figure 36: AD-33 Worker view previous ingredient estimation**

**Output:** Worker can view previous ingredient estimation information in the database.

**System Requirement Specification:**

**SRS-130:** The system can connect to the phungnoi database and access webpages\_Roles, OrderDetail, Recipe and Units tables.

**SRS-134:** The system shall provide a textbox in the ingredient estimation interface to search the filter keywords of the day, month, and year of the previous ingredient estimation result.

**SRS-31:** The system shall provide a button for requesting to search the filter keywords in the textbox.

**SRS-135:** The system shall provide a service to match the period of the day, month, and year in field textbox with the previous ingredient estimation result in the database.

**SRS-136:** The system shall provide a user interface to display the previous ingredient estimation list. In each label, shall display the delivery date which related to the filter keywords.

**SRS-137:** The system shall provide a button for the admin viewing the ingredient usage in the ingredient estimation interface.

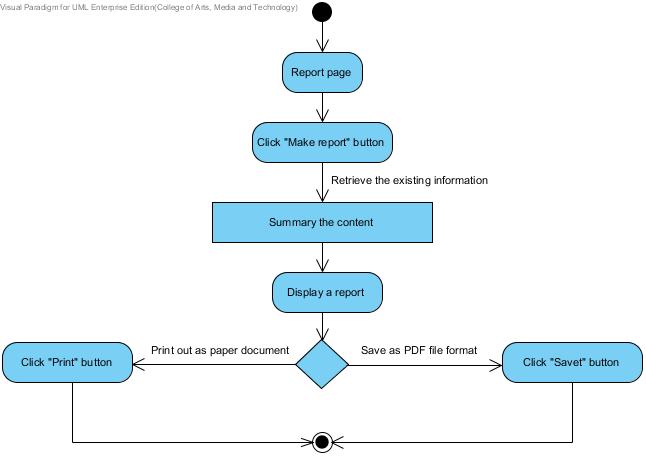
**SRS-138:** The system shall provide a calculating function to compute the ingredient usage by using information from the menu recipe and total ordered.

**SRS-139:** The system shall provide a user interface to display the ingredient estimation information which are the product ID, product name, ingredient ID, ingredient name, ingredient quantity, and units on the ingredient estimation page.

**URS-23:** Wholesaler and Worker can make a different report type according to their member’s role.

**Precondition:** The worker and wholesaler must have login to the system.

**Input: -**

**Action:** 

**Figure 37: AD-34 Worker and wholesaler make report**

**Output:** Worker and Wholesaler make a report from the existing information in the database.

**System Requirement Specification:**

**SRS-140:** The system can connect to the phungnoi database and access Member, webpages\_Roles, OrderDetail, and Recipe tables.

**SRS-141:** The system shall provide a button for making a report.

**SRS-142:** The system shall provide a report function to retrieve the summary information from the database.

**SRS-143:** The system shall provide a user interface to display the summary report contents on the report page.

**SRS-144:** The system shall provide a button for saving a summary report as a PDF file format from the report interface.

**SRS-145:** The system shall provide a service to export a summary report document out.

**SRS-146:** The system shall provide a button for printing a summary report as a documented paper.

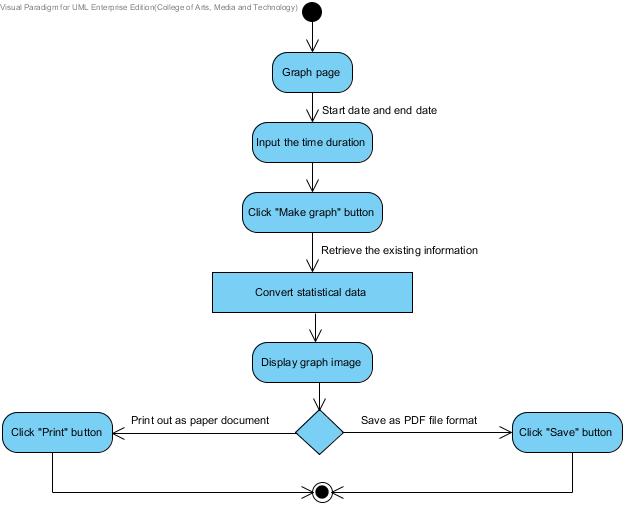
**URS-24:** Wholesaler and Worker can make a graph result from the ordered statistical data.

**Precondition:** The worker and wholesaler must have login to the system.

**Input:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Example** |
| Start Date | The start date for using ordered information. | 17 October 2014 |
| End Date | The end date for using ordered information. | 17 September 2014 |

**Action:**



**Figure 38: AD-35 Worker and wholesaler make graph**

**Output:** Worker and Wholesaler make a graph from the statistical data in the database.

**System Requirement Specification:**

**SRS-73:** The system can connect to the phungnoi database and access Member, webpages\_Roles, and OrderDetail tables.

**SRS-147:** The system shall provide a button for converting ordered information to be a statistical graph.

**SRS-148:** The system shall provide a service to process statistical data became to a graph image.

**SRS-149:** The system shall provide a user interface to display the graph image and statistical portion on the graph page.

**SRS-150:** The system shall provide a button for saving a graph as an image file format from the graph interface.

**SRS-151:** The system shall provide a service to export a graph image out.

**SRS-152:** The system shall provide a button for printing a graph image as a documented paper.