

**HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY**  
**FACULTY OF COMPUTER SCIENCE AND ENGINEERING**



**PROJECT REPORT**  
**PRACTICE ON SOFTWARE ENGINEERING**

# **SMART FOOD COURT SYSTEM**

Lecturer:     Pham Thi Vuong

Group SE5:

Nguyen Pham Ngoc Quy	1810473
Ly Kien Minh	1813080
Chau Thanh Tan	1810501
To Duy Hung	1810198
Le Trung Son	1810482

Ho Chi Minh, 30 July, 2020



# Table of Contents

<b>I. Introduction .....</b>	<b>3</b>
<b>II. Functional requirements.....</b>	<b>4</b>
<i>1. Functions.....</i>	<i>4</i>
1.1 Registration.....	4
1.2 Login.....	4
1.3 Display the menu .....	4
1.4 Modify menu.....	4
1.5 Search engine .....	4
1.6 Order .....	4
1.7 Place order .....	4
<i>2. Usecase diagram.....</i>	<i>5</i>
<i>3. Methods of implementation .....</i>	<i>6</i>
<i>4. Usecase description .....</i>	<i>6</i>
4.1 Registration.....	6
4.2 Login.....	9
4.3 Display the menu .....	11
4.4 Modify the menu.....	13
4.5 Search engine .....	15
4.6 Order .....	18
4.7 Place order .....	22
<b>III. Implement view .....</b>	<b>24</b>
<b>IV. Architecture design .....</b>	<b>25</b>



## Table of Tables

Table 1: Use-case scenario for registration .....	6
Table 2: Description of function registration .....	8
Table 3: Use-case scenario for login .....	9
Table 4: Description of function login .....	11
Table 5: Use-case scenario for displaying the menu .....	11
Table 6: Description of function display the menu .....	12
Table 7: Use-case scenario for modifying the menu .....	13
Table 8: Description of function modify the menu .....	14
Table 9: Use-case scenario for searching engine .....	15
Table 10: Use-case diagram for searching engine.....	16
Table 11: Description of function search engine.....	17
Table 12: Use-case scenario for adding food .....	18
Table 13: Use-case scenario for deleting food .....	18
Table 14: Description of function ordering food.....	21
Table 15: Use-case scenario for placing order .....	22
Table 16: Description of function place order.....	23

---

## Table of Figures

Figure 1: Use-case diagram for the whole system.....	5
Figure 2: Use-case diagram for registration .....	7
Figure 3: Registration page .....	7
Figure 4: Use-case diagram for login .....	10
Figure 5: Login page.....	10
Figure 6: Display the menu with many of choices .....	12
Figure 7: Admin can edit menu in admin site .....	14
Figure 8: Search food with key word "sua" .....	16
Figure 9: Use-case diagram for ordering food .....	19
Figure 10: After finding food, select "add to cart" btn if you want to add this food to cart .....	19
Figure 11: Change quantity, size, topping. After that, click "add to cart" button.....	20
Figure 12: Order cart with total money and payment method .....	20
Figure 13: Use-case diagram for placing order .....	22
Figure 14: Place order in order cart.....	23
Figure 15: Implementation view .....	24
Figure 16: Deployment view for the whole system.....	25

## I. Introduction

The university is currently having one food court located in its Ly Thuong Kiet campus and is going to build another one in Di An campus. All food courts consist of a number of vendors at food stalls or service counters. Meals are ordered at one of the vendors and then carried to a common area for consumption. The food may also be ordered to take-away. Note that, all food courts are self-service and there is no food delivery service, but may be applied in the future.

When a meal needs time in preparing, or in busy moments, a vendor may provide a pager calling device for notifying the customers when their food is ready.

In 2020, the university wish to build a smart food court system (SFCS) to make the university smarter. The system is for customers to order foods at the food courts or before coming to that places.

There will be some machine with touch screen around the food court for ordering food. When the payment is made, the order is confirmed and will be put in the order list at the food stall kitchen. An order slip will be printed by the machine. The customer can grab a near-by pager that has the number matched with the order number and wait for its notification then goes get the food. He/she can also simply wait at the food stall and then shows the order slip to take the food. Note that, separated orders should be placed for different vendors.

The payment can be made by some online payment service such as Momo wallet, Samsung Pay, Apple Pay, etc. It should be secured to make sure that no important information is leaked to non-authorized persons.

There should be also a mobile app for members of the food court system that allow them to order food before coming to the food courts. In this case, the notification from the kitchen can be pushed to the mobile app instead of the pager. The estimated preparing time of the order can also be informed to the members.

The mobile app has to have an ability to make an online payment or allow members to re-charge and use their payment account for the payment.

Of course, the software system has to have the managing features. For the cooks, they want to see the orders, inform the ready of the food, or inform the system that some food is out-of-order, etc. For the vendor owners, they want to see the (online) report about their stalls, etc. For the managers of the food courts, they want to see the (online)



report of the whole food court, etc. For the IT staffs, sometime, they want to put the system in the maintenance mode, that shutdowns all online services, etc.

## **II. Functional requirements**

### **1. Functions**

#### **1.1 Registration**

If customer wants to order the food then he/she must be registered, unregistered user can not go for ordering.

#### **1.2 Login**

The customer login to the system by entering valid user id and password for ordering.

#### **1.3 Display the menu**

In the system all the items are displayed with their rates

#### **1.4 Modify menu**

System can make changings in menu like adding or removing food items which are not available.

#### **1.5 Search engine**

Search engine for users to find food , based on the key word of the food.

#### **1.6 Order**

Customer can view food detail, delete or add food item in order cart.

#### **1.7 Place order**

Allow user to place order after selected food

## 2. Usecase diagram

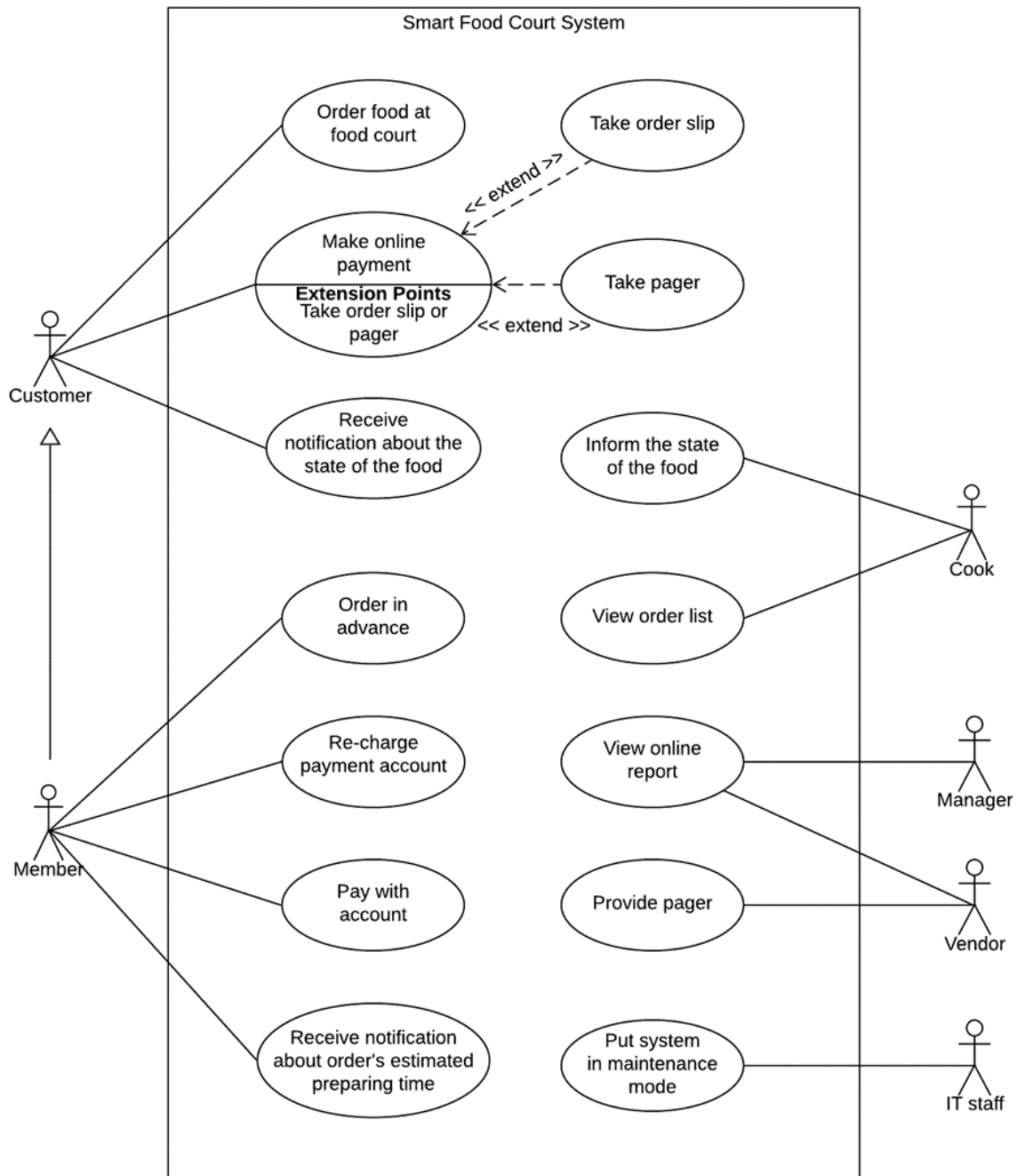


Figure 1: Use-case diagram for the whole system

### 3. Methods of implementation

Platform: Web

Front-end: JavaServer Pages, HTML, CSS, Javascript

Back-end: Java

Database: mySQL

### 4. Usecase description

#### 4.1 Registration

*Table 1: Use-case scenario for registration*

Use-case name	<b>Registration</b>
Actors	Customer.
Description	The customer register a new account.
Triggers	The customer select to register button on website.
Preconditions	The app have to connect to the Internet. The customer open the mobile app. No account is logged in.
Postconditions	The customer registered sucessfully.
Normal Flow	<ol style="list-style-type: none"><li>1. The customer select register button.</li><li>2. The system asks for the personal details of the customer.</li><li>3. The customer enter the personal details including credentials, i.e name, username, password, number phone/email and submit.</li><li>4. The system check the entered information is valid.</li><li>5. The system store user information in database and notify that registration was successful.</li></ol>
Exception	Exception 1 at step 3: If the entered infomation is not valid, the customer is required to retype.

Alternative Flow	None.
------------------	-------

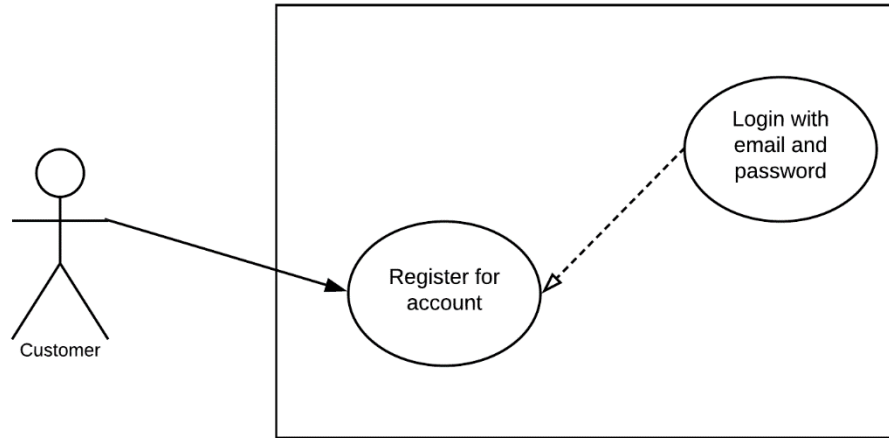


Figure 2: Use-case diagram for registration

### User story:

As a customer, he/she want to register for a account, so that he/she can use that to log in to the web, view and order/buy products sold at food court.

### Mock-up:

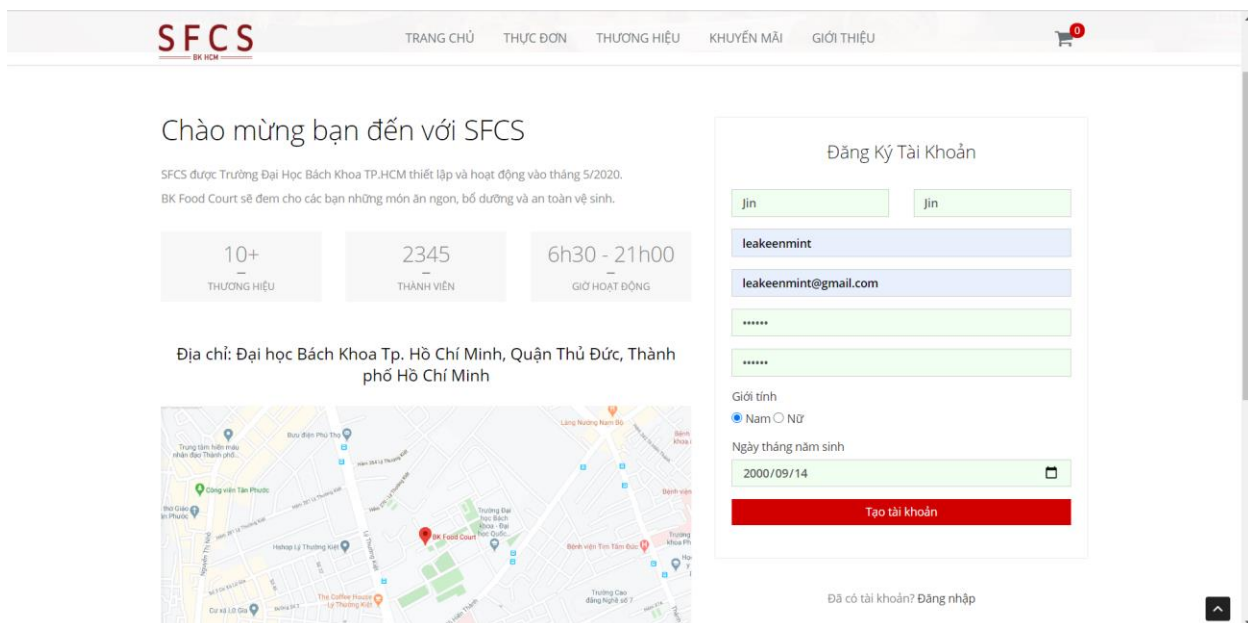


Figure 3: Registration page



**Description:**

No	Field Name	Description	Control Type	Data Type	Mandatory	Default Value
1	First name input	Add username	Text input	Text	Yes	N/A
2	Last name input	Add password	Text input	Text	Yes	N/A
3	User name input	Login to SFCS	Text input	Text	Yes	N/A
4	Email input	Add email info	Text input	Text	Yes	N/A
5	Password input	Add password info	Text input	Text	Yes	N/A
6	Confirm password input	Add confirm password info	Text input	Text	Yes	N/A
7	Gender input	Add gender info	Choice input	Enum	Yes	N/A
8	Date of birth input	Add date of birthday	Date input	Date	Yes	N/A
9	Register button	Register for account	Button	N/A	Yes	N/A

*Table 2: Description of function registration*

## 4.2 Login

<b>Use-case name:</b>	<b>Login</b>		
<b>Created by:</b>	Chau Thanh Tan	<b>Last updated by:</b>	Chau Thanh Tan
<b>Date created:</b>	April 22 <sup>nd</sup> 2020	<b>Date last updated:</b>	May 6 <sup>th</sup> 2020
<b>Actor:</b>	MS (a customer has already registered to be a member of SFCS)		
<b>Description:</b>	MS logs in to the SFCS on mobile phone or Touch-screen machine		
<b>Trigger:</b>	MS intends to use Smart Food Court online service.		
<b>Includes:</b>	None		
<b>Extends:</b>	None		
<b>Pre-conditions:</b>	Customer has already registered to be a member of SFCS (called MS)		
<b>Post-conditions:</b>	MS accesses to the app successfully		
<b>Normal flow</b>	<ol style="list-style-type: none"> <li>1. MS selects Login button on the top of the page.</li> <li>2. The login box pops up. The following fields are available: <ul style="list-style-type: none"> <li>- Username</li> <li>- Password</li> </ul> </li> <li>3. MS enters Username, Password.</li> <li>4. The popup box disappears and the page is refreshed to the post-login page.</li> </ol>		
<b>Alternative flows</b>	None		
<b>Exceptions</b>	<p>Exception 1: At step 3:</p> <p>If MS types in a wrong username/password, the popup window says: “Wrong username”/”Wrong password”. The field Username/Password becomes blank so that MS enters the correct username/password.</p> <p>Besides, if MS keeps typing wrong password 5 times, system will lock the account of that username for 15 minutes. After that amount of time, MS will be able to try to login again.</p>		

Table 3: Use-case scenario for login

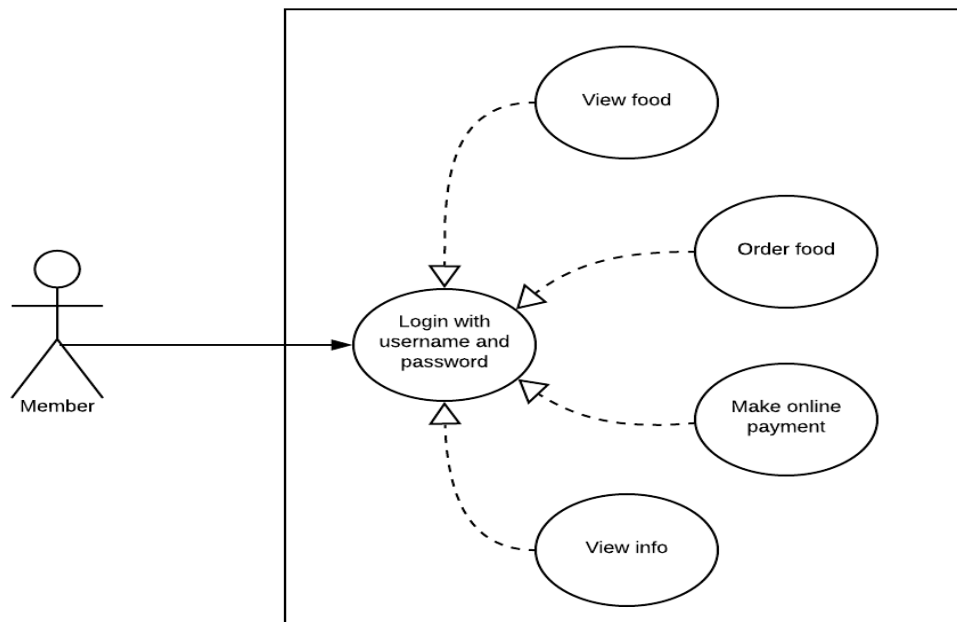
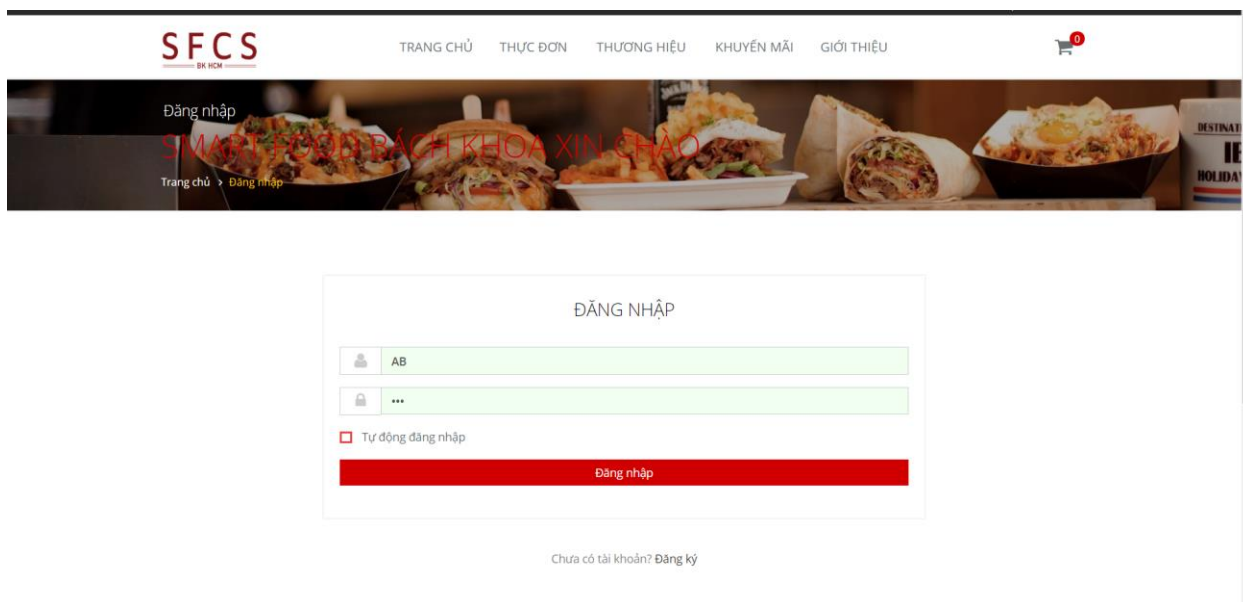


Figure 4: Use-case diagram for login

## User story:

Member logs in to SFCS to take several services in SFCS.

## Mock-up:



The mock-up shows the SFCS website header with navigation links: TRANG CHỦ, THỰC ĐƠN, THƯƠNG HIỆU, KHUYẾN MÃI, and GIỚI THIỆU. Below the header is a banner image of food. The main content area is titled 'ĐĂNG NHẬP' (Login) and contains a form with two input fields for username and password, a checkbox for 'Tự động đăng nhập' (Remember me), and a red 'Đăng nhập' (Login) button. Below the form is a link: 'Chưa có tài khoản? Đăng ký' (Don't have an account? Register).

Figure 5: Login page

### Description:

No	Field Name	Description	Control Type	Data Type	Mandatory	Default Value
1	Username input	Add username	Text input	Text	Yes	N/A
2	Password input	Add password	Text input	Text	Yes	N/A
3	Login button	Login to SFCS	Button	N/A	Yes	N/A

Table 4: Description of function login

### 4.3 Display the menu

<b>Use-case name:</b>	<b>Display the menu</b>
<b>Actor:</b>	Customer
<b>Description:</b>	Customer can view the menu info
<b>Pre-conditions:</b>	Customer need to login to SFCS
<b>Post-conditions:</b>	Screen display the menu
<b>Normal flow</b>	1. Customer selects Product List or “Menu” on the navbar. 2. System show the menu info containing items with rates. If customer want to see more details of each item, select that item.
<b>Alternative flows</b>	None
<b>Exceptions</b>	None

Table 5: Use-case scenario for displaying the menu

### User story:

In the system all the items are displayed with their rates.

## Mock-up:

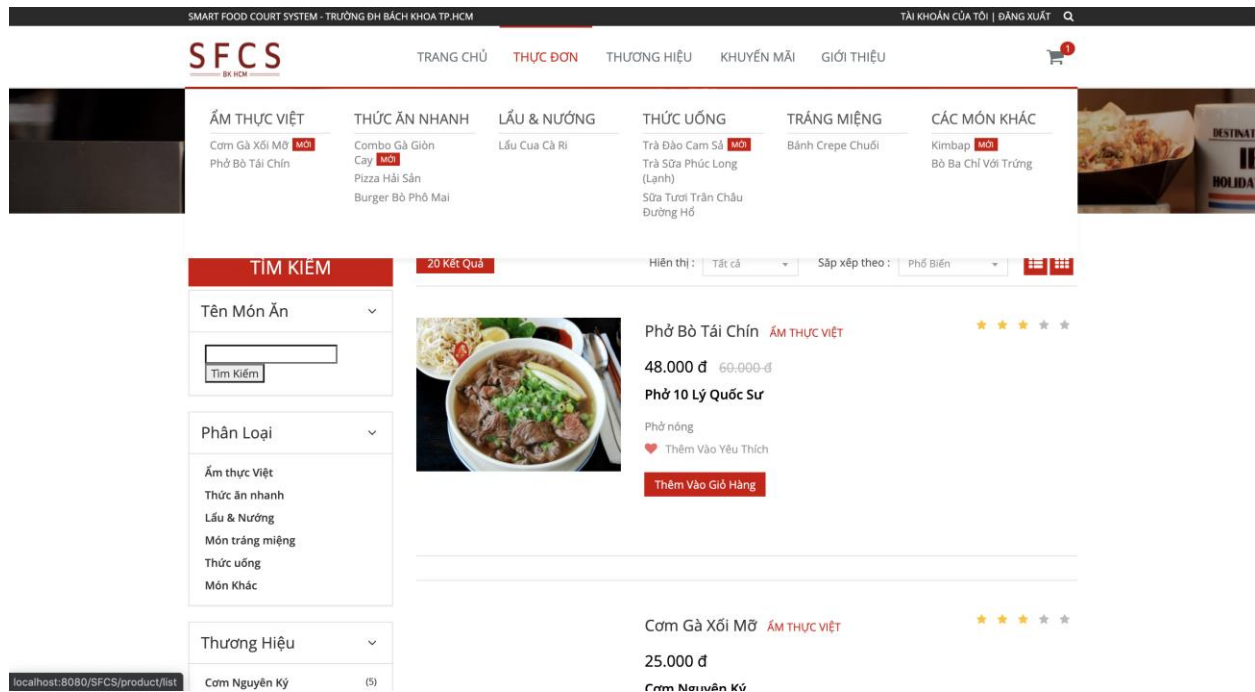


Figure 6: Display the menu with many of choices

## Description:

No	Field Name	Description	Control Type	Data Type	Mandatory	Default Value
1	Menu	Menu	Display	Text	No	N/A
2	Detail	More item details	Button	N/A	No	N/A

Table 6: Description of function display the menu

#### 4.4 Modify the menu

<b>Use-case name:</b>	<b>Modify the menu</b>
<b>Actor:</b>	Admin
<b>Description:</b>	Admin can change the menu like adding or removing food items.
<b>Pre-conditions:</b>	Admin need to login to SFCS
<b>Post-conditions:</b>	Menu has new changings.
<b>Normal flow</b>	<ol style="list-style-type: none"><li>1. Admin select Product Management.</li><li>2.<ul style="list-style-type: none"><li>- Add new food items:<ol style="list-style-type: none"><li>2.1.1 Select “Add new product” button.</li><li>2.1.2 Type info about new item.</li><li>2.1.3 Select “OK”</li></ol></li><li>- Delete food items:<ol style="list-style-type: none"><li>2.2 Admin select “Delete” button beside food items need to be deleted.</li></ol></li><li>- Edit food items:<ol style="list-style-type: none"><li>2.3 Admin select “Edit” button beside food items need to be edited.</li></ol></li></ul></li><li>3. Select “Apply” to make change to the menu.</li></ol>
<b>Alternative flows</b>	None
<b>Exceptions</b>	None

Table 7: Use-case scenario for modifying the menu

#### User story:




System can make changing in menu like editing, adding or removing food items which are not available.

## Mock-up:

### All Products

You can manage products in here

[Add New Product](#)

Advanced Tables									
10 records per page		Search: <input type="text"/>							
ID	Image	Name	Price	Quantity	Discount	Category	Stall	Description	Action
11		Trà Sữa Phúc Long (Lạnh)	45000	60	0	Thức uống	Phúc Long Coffee & Tea		<a href="#">Detail</a>   <a href="#">Edit</a>   <a href="#">Delete</a>
12		Sữa Tươi Trân Châu Đường Hổ	49000	45	28	Thức uống	Trà sữa Toco Toco	Thức uống mới, đậm vị sữa tươi và đường hổ	<a href="#">Detail</a>   <a href="#">Edit</a>   <a href="#">Delete</a>
13		Bánh cuốn	30000	60	5	Ẩm thực Việt	Hoàng Yến Cuisine	Bánh cuốn Hà Nội siêu ngon	<a href="#">Detail</a>   <a href="#">Edit</a>   <a href="#">Delete</a>

Showing 11 to 13 of 13 entries

[Previous](#) [1](#) [2](#) [Next](#)

Figure 7: Admin can edit menu in admin site

## Description:

No	Field Name	Description	Control Type	Data Type	Mandatory	Default Value
1	Add	Add new item to menu	Button	N/A	No	N/A
2	Delete	Delete item from menu	Button	N/A	No	N/A
3	Edit	Edit item in menu	Button	N/A	No	N/A

Table 8: Description of function modify the menu

## 4.5 Search engine

Name	<b>Search engine</b>
Actor	Registered users and guest users
Description	With search engine, users can find many things about the food they are want to eat.
Preconditions	Users need to access to Product/Search page
Normal flow	<ol style="list-style-type: none"><li>1. Users go to home page of website by URL.</li><li>2. Users can find the button link to Product/Search page at the special menu of the web site.</li><li>3. Users input what they want to search.</li><li>4. Website presents a page with many results.</li><li>5. Users can select one of results.</li><li>6. Website presents a page with detail of that result.</li></ol>
Exceptions	Exception at step 4: Users cannot find anything about what they want to find.
Alternative flow	At step 3, users can add more filters to get the correct results.

*Table 9: Use-case scenario for searching engine*



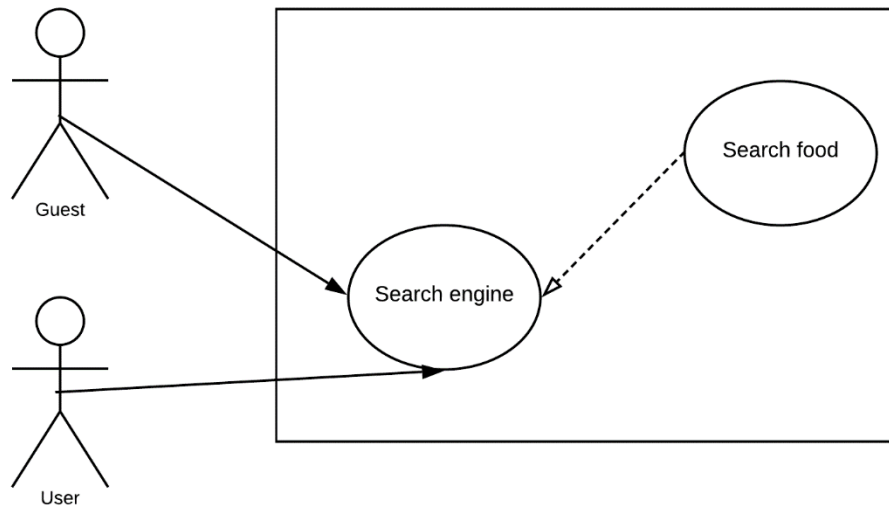


Table 10: Use-case diagram for searching engine

### User story:

As guest users, he or she wants to search information about the food that they are going to eat. Then, get some suggestions (about the same food) from system. After that, they can choose to create an account to use our web application for a long time or order food

### Mock-up:

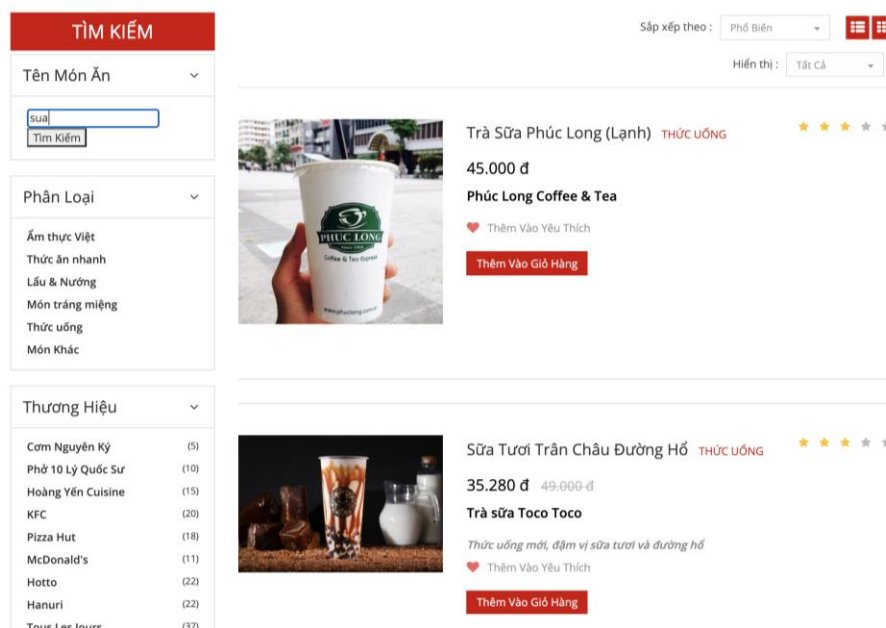


Figure 8: Search food with key word "sua"

## Description

No	Field Name	Description	Control Type	Data Type	Mandatory	Default Value
1	Search panel	Search food name	Text input	Text	Yes	N/A
2	Search result	Display relevant food name when searching	Expand list	Text	Yes	“No results found”
3	Food information	Display food information in food page	Display	Text	Yes	N/A

Table 11: Description of function search engine

## 4.6 Order

Name	<b>Order (Add)</b>
Actor	Customer (Member)
Description	Member add food item in order cart
Preconditions	Member logged into the web app
Normal flow	<ol style="list-style-type: none"> <li>1. Member selects a meal</li> <li>2. System shows food details screen</li> <li>3. Member changes the quantity, size,...</li> <li>4. Member selects “ADD TO CART”</li> <li>5. System display order cart (SFCS/member/cart page)</li> </ol>
Exceptions	No
Alternative flow	No

Table 12: Use-case scenario for adding food

Name	<b>Order (Delete)</b>
Actor	Customer (Member)
Description	Member delete food item in order cart
Preconditions	Item (food selected) is added to cart by member
Normal flow	<ol style="list-style-type: none"> <li>1. Member open order cart (SFCS/member/cart page)</li> <li>2. System displays a list of selected food</li> <li>3. Member click criss-cross (x) to remove the selected food from order cart</li> <li>4. System display order cart (SFCS/member/cart page) after removing</li> </ol>
Exceptions	Exception 1 at step 1 : No item in cart for removing
Alternative flow	No

Table 13: Use-case scenario for deleting food

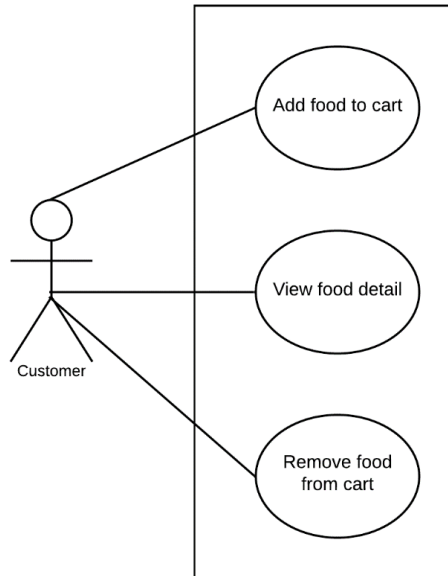


Figure 9: Use-case diagram for ordering food

## User Story:

As a member, he/she wants to check the details of the food, add or delete food in order cart

## Mock-up:

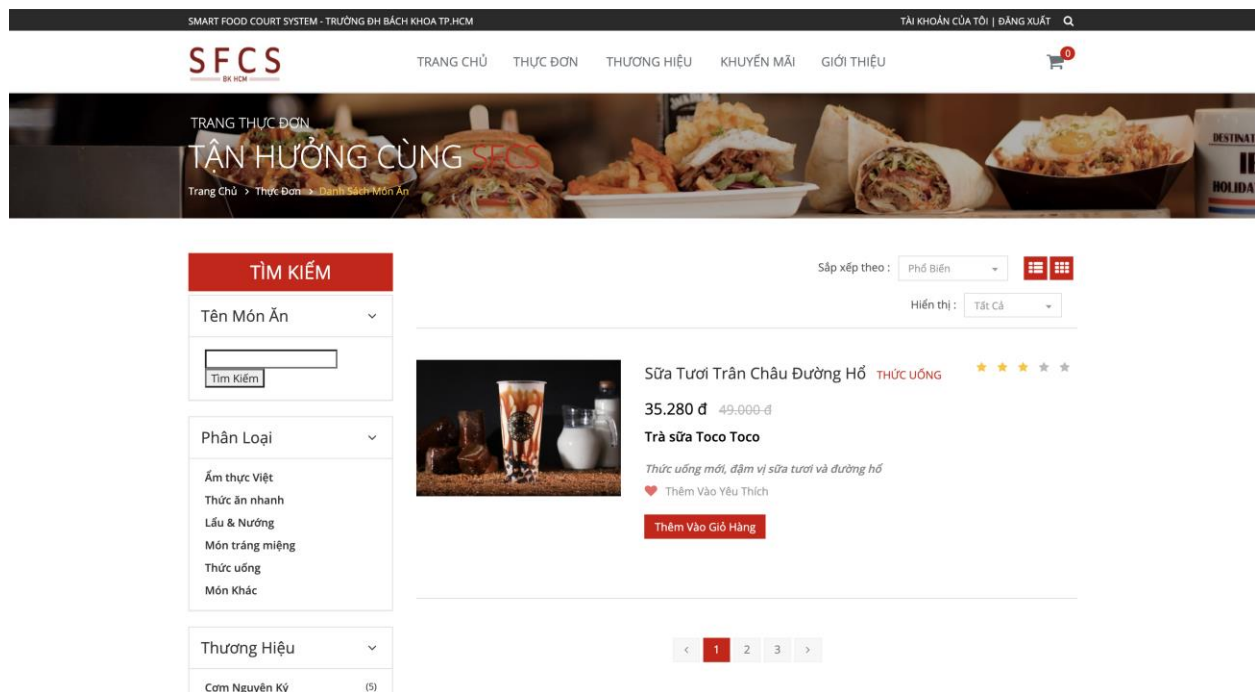


Figure 10: After finding food, select "add to cart" btn if you want to add this food to cart

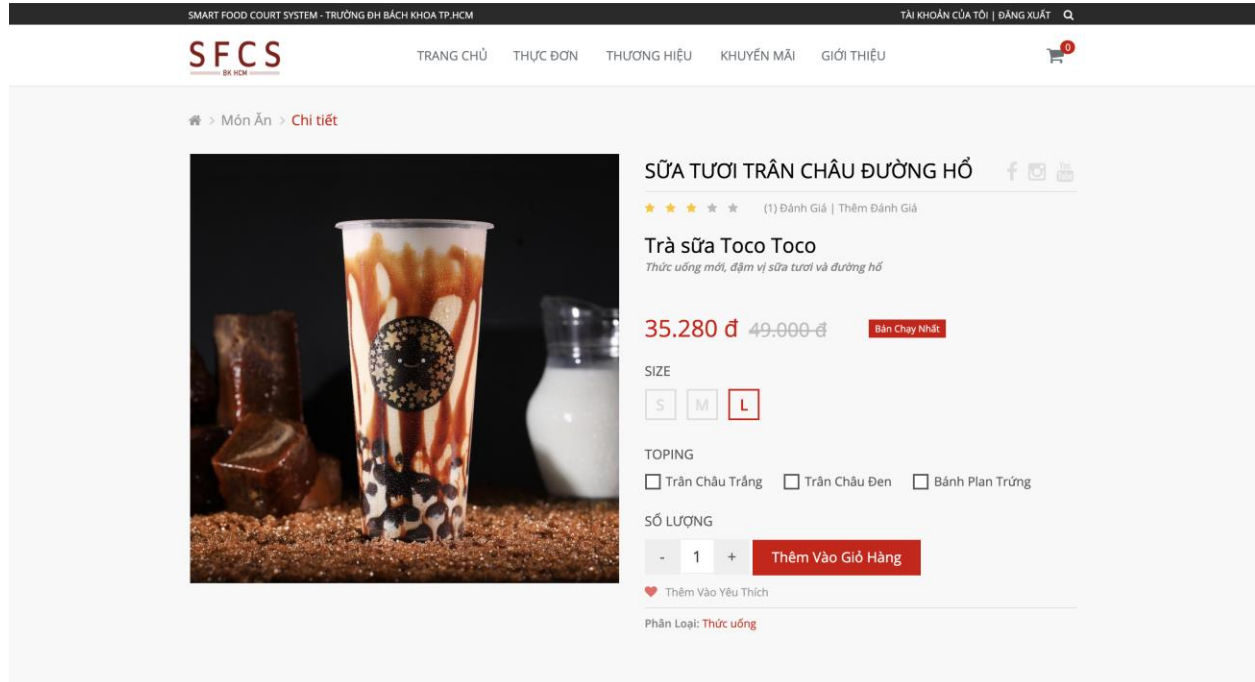


Figure 11: Change quantity, size, topping. After that, click “add to cart” button

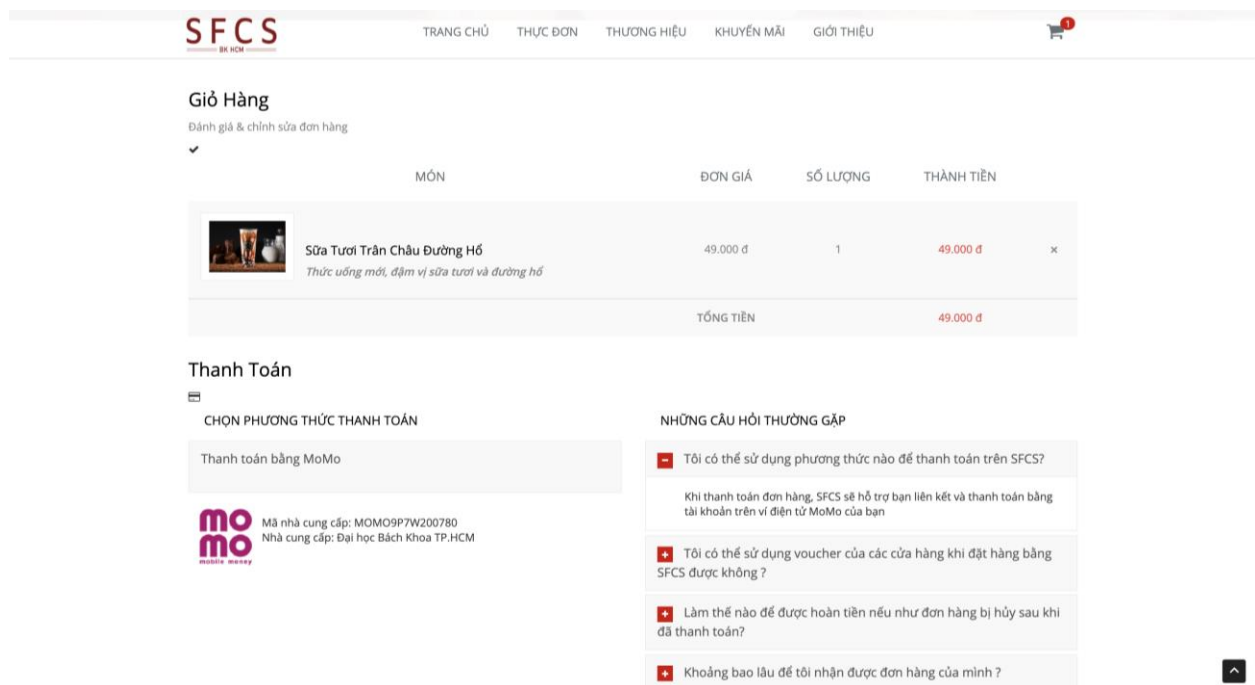


Figure 12: Order cart with total money and payment method

## Description

No	Field Name	Description	Control Type	Data Type	Mandatory	Default Value
1	Food image	Image of the food	Display	Image	Yes	N/A
2	Food name	Name of the food	Display	Text	Yes	N/A
3	Food description	Description about the food	Display	Text	Yes	N/A
4	Food price	Price of the meal	Display	Text	Yes	N/A
5	Quantity button	Change the quantity of the food before adding to cart	Button	N/A	Yes	0
6	Other button (size, topping,...)	Change the size, add topping,.. of the food before adding to cart	Button	N/A	Yes	N/A
7	Add to cart button	Add the meal to cart	Button	N/A	Yes	N/A
8	Delete button	Delete food item in cart	Button	N/A	Yes	N/A

Table 14: Description of function ordering food

#### 4.7 Place order

Name	<b>Place order</b>
Actor	Customer (Member)
Description	Member can order all food item in cart
Preconditions	There must be at least one food item in the cart
Normal flow	<ol style="list-style-type: none"> <li>1. Customer pick up any food available that they want to order on web app.</li> <li>2. Customer click symbol “cart” on navbar-header.</li> <li>3. Customer select payment method</li> <li>4. Customer tap “Place Order” button on bottom of the site</li> <li>5. System displays payment instructions for customer</li> <li>6. If payment is successful, notice of order success and display waiting time.</li> </ol>
Exceptions	No
Alternative flow	No

Table 15: Use-case scenario for placing order

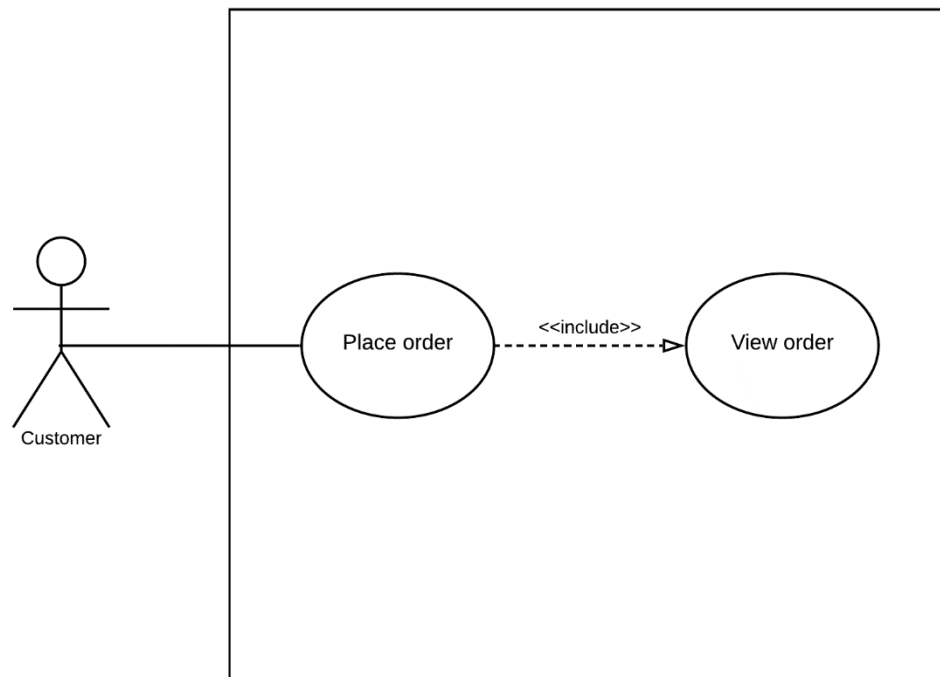


Figure 13: Use-case diagram for placing order

## User Story:

As a member, he/she wants to checkout item (food selected) in cart, place order (confirm to buy food). So, they can click symbol “cart” in “navbar-header” to do that

## Mock-up:

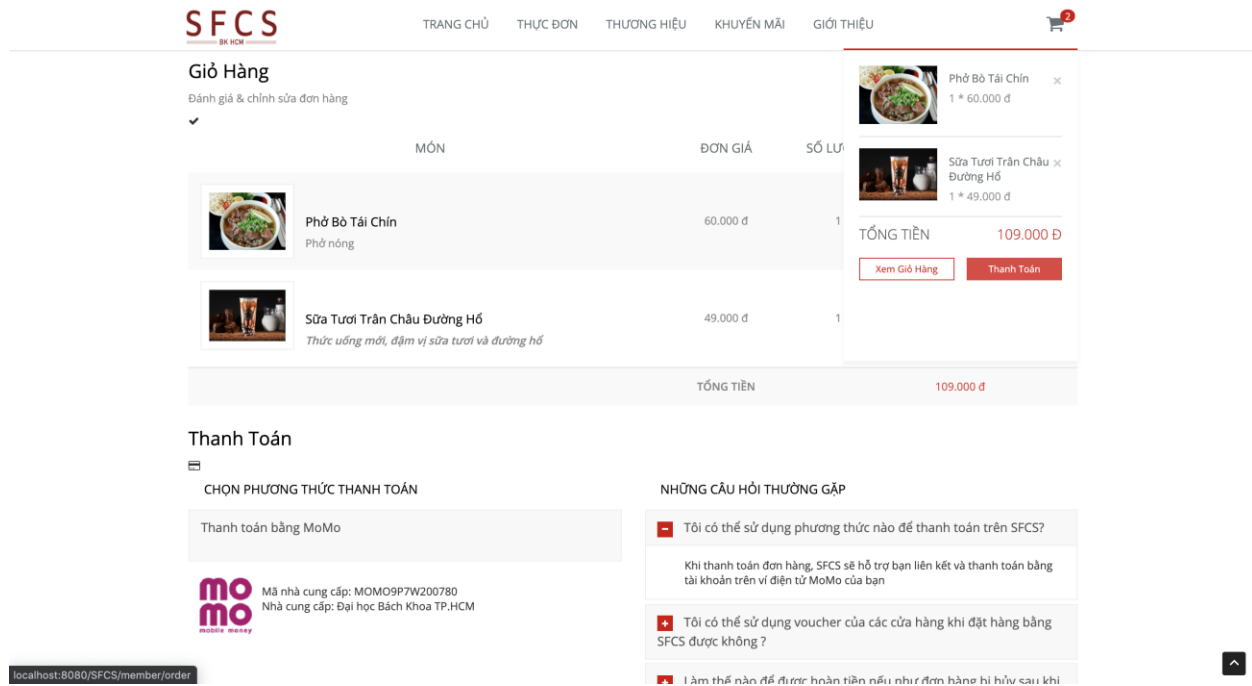


Figure 14: Place order in order cart

## Description:

No	Field Name	Description	Control Type	Data Type	Mandatory	Default Value
1	Cart	Cart All of item which customer selected in category or homepage in Cart.	Button	N/A	Yes	N/A
2	Place order	Select “Place Order” for confirm order food	Button	N/A	Yes	N/A
3	Empty	No food is selected	Display	Text	Yes	N/A

Table 16: Description of function place order



### III. Implement view

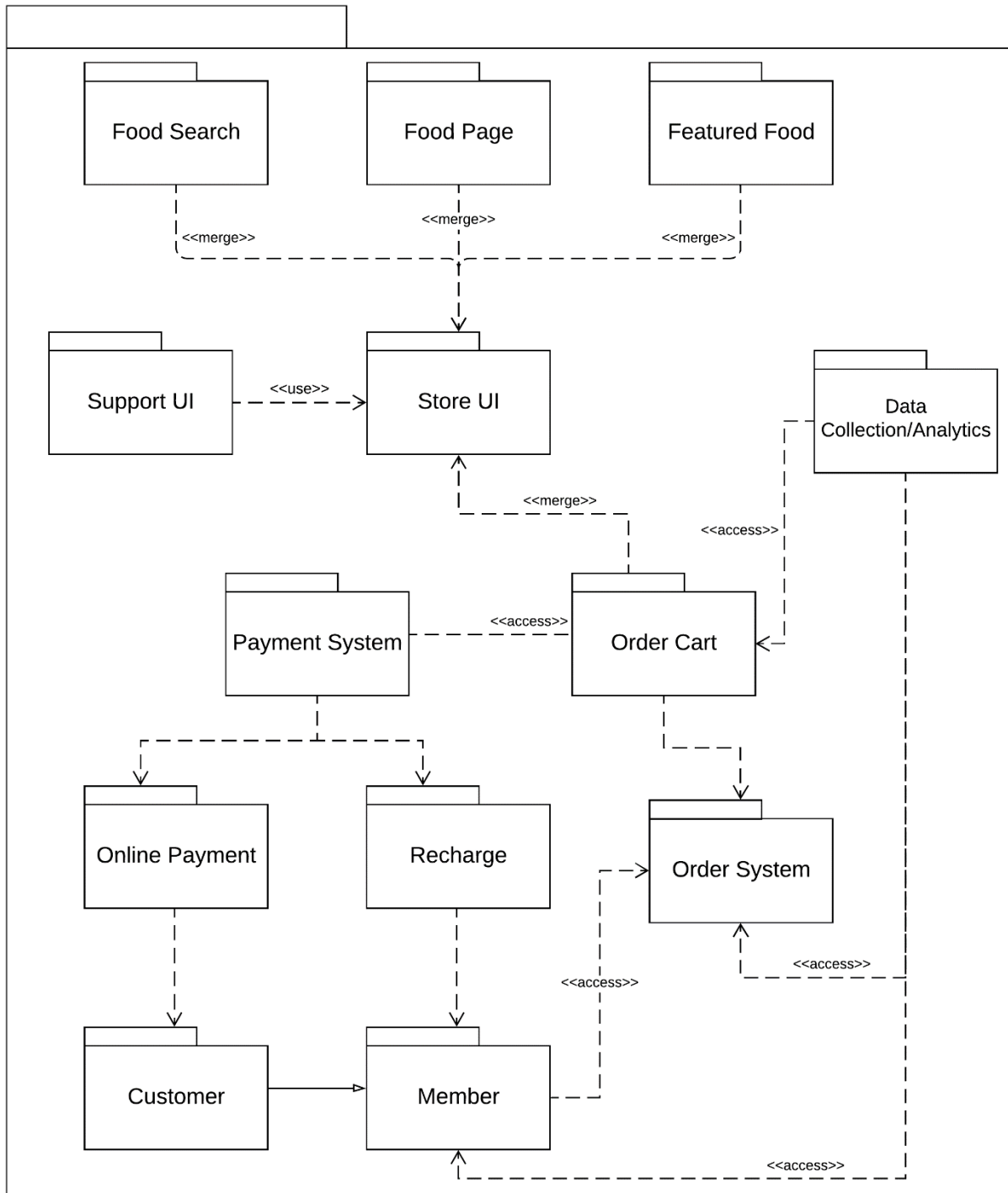


Figure 15: Implementation view

## IV. Architecture design

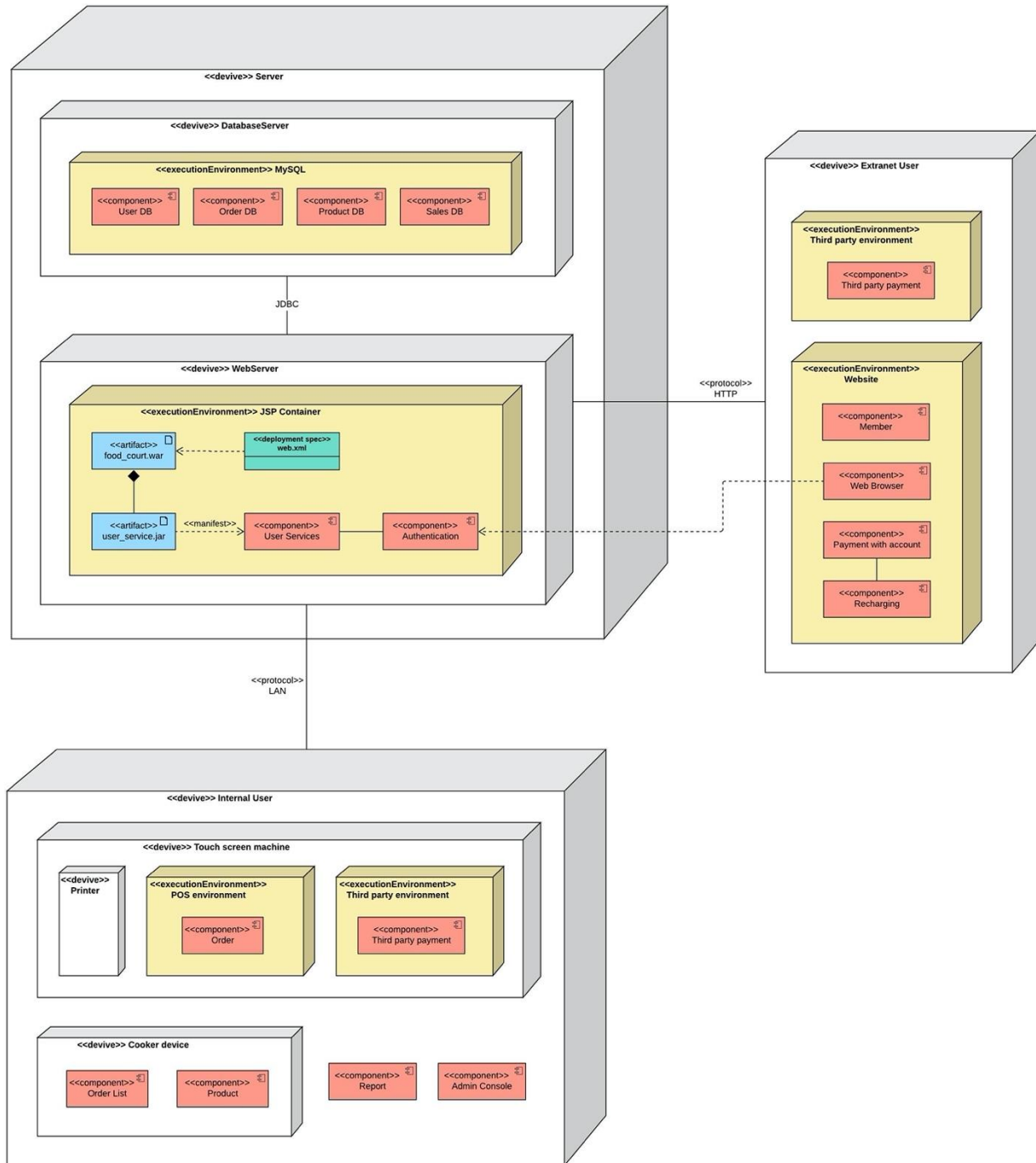


Figure 16: Deployment view for the whole system