**TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT**

**Khoa Kỹ thuật - Công nghệ**

****

**ĐỒ ÁN MÔN HỌC**

**ĐỒ ÁN CƠ SỞ NGÀNH**

**Đề tài**

**XÂY DỰNG WEBSITE BÁN ĐIỆN THOẠI DI ĐỘNG CHO THEGIOIALO**

**GVHD:** TH.S HỒ ĐẮC HƯNG

**SVTH:**  **1.**ĐÀO XUÂN HOÀNG

MSSV: 1724801030188

**2.**NGUYỄN QUỐC ANH

MSSV: 1724801030188

**3.**NGUYỄN ĐẠI NGHĨA

MSSV: 1724801030084

**LỚP:** D17PM02

|  |
| --- |
| Bình Dương – 12/2019 |

**TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT**

**Khoa Kỹ thuật - Công nghệ**

****

**ĐỒ ÁN MÔN HỌC**

**ĐỒ ÁN CƠ SỞ NGÀNH**

**Đề tài**

**XÂY DỰNG WEBSITE BÁN ĐIỆN THOẠI DI ĐỘNG CHO THEGIOIALO**

**GVHD:** TH.S HỒ ĐẮC HƯNG

**SVTH:**  **1.**ĐÀO XUÂN HOÀNG

MSSV: 1724801030188

**2.**NGUYỄN QUỐC ANH

MSSV: 1724801030191

**3.**NGUYỄN ĐẠI NGHĨA

MSSV: 1724801030084

**LỚP:** D17PM02

|  |
| --- |
| Bình Dương – 12/2019 |

**TRƯỜNG ĐẠI HỌC THỦ DẦU MỘT**

**KHOA KỸ THUẬT – CÔNG NGHỆ**

**NHẬN XÉT VÀ CHẤM ĐIỂM CỦA GIẢNG VIÊN**

Họ và tên giảng viên: TH.S HỒ ĐẮC HƯNG

Tên đề tài: **XÂY DỰNG WEBSITE BÁN ĐIỆN THOẠI DI ĐỘNG CHO THEGIOIALO**

Nội dung nhận xét:

**Điểm:**

Bằng số:

Bằng chữ:

|  |  |
| --- | --- |
|  | **GIẢNG VIÊN CHẤM**  *(Ký, ghi rõ họ tên)* |

**TH.S HỒ ĐẮC HƯNG**

# CATEGORY

CATEGORY iii

1.INTRODUCTION 4

1.1 Purpose 4

1.2 Scope 4

1.3 OverView 4

1.3.1. Topic Target 4

1.3.2. Request Of The Subject 4

*1.3.2.1. Equipment And Software 4*

*1.3.2.2. Site Requirements 4*

1.4 Reference Material 2

2.SYSTEM OVERVIEW 2

3.SYSTEM ARCHITECTURE 2

3.1 Architectural Design 2

3.2 Decomposition Description 3

3.3 Design Rationale 4

4.DATA DESIGN 6

4.1 Data Description 6

4.2 Data Dictionary 7

5.COMPONENT DESIGN 11

5.1 Product module 11

5.2 Online payment module 11

5.3 Customer account management module 12

5.4 Search base module 12

5.5 Cart module 12

5.6 New product module 12

5.7 Introduction of products, service modules 12

5.8 Product management module 12

6.HUMAN INTERFACE DESIGN 12

6.1 Overview of User Interface 12

6.2 Screen Images 13

6.3 Screen Objects and Actions 19

7.REQUIREMENTS MATRIX 23

7.1 Class Diagram 23

7.2 Use Case Diagram 24

7.3 Sequence Diagram 24

APPENDICES 29

1. **INTRODUCTION**
   1. Purpose

- This software design document describes the architecture and system design of the mobile sales website for ThegioiAlo store.

* 1. Scope

- In the era of Industry 4.0, everyone wants to own a phone for themselves to facilitate communication. So the demand for phones is huge.

- Therefore, if there is a website that can help them buy the phone smoothly without going out of the store, just order and will be delivered in place, it will be very beneficial.

- To meet this need, we decided to build a mobile selling website. This site will support and interact with users so that they can search for their favorite products and order it through the website.

* 1. OverView

### 1.3.1. Topic Target

- Help us to reach potential customers.

- Helping enterprises save costs.

- Help build and promote brands.

- Helping increasing business efficiency.

- Upgraded products and adding new customers.

- Better profit.

- Help to receive feedback from customers more conveniently.

- Help build credibility.

### 1.3.2. Request Of The Subject

#### 1.3.2.1. Equipment And Software

- Computers can design the web.

- Sql server management studio

- Software design: dreamwear, visual,…

#### 1.3.2.2. Site Requirements

The system has 2 parts:

- First: The customer part

+ Customers are those who want to buy goods, they will search

+ Necessary items from the system and order these items. So must have the following functions.

+ Display the list of items of the store for customers to see, choose and buy.

+ Customers view news, new descriptions on the website.

+ After the customer selects and orders successfully, the system must send the order email to the customer email.

- Second: For administrators

+ The owner of the site has control over all activities of the system. This user is provided with a username and password to log into the system to perform his function.

+ Functional update, modify and delete products.

+ Customer account management.

* 1. Reference Material

- Bryan Syverson, Joel Murach, *SQL Server hướng dẫn học qua ví dụ*, NXB Khoa học và Kỹ thuật, 2013.

- Floyd Sally, Jacobson Van (1993), *Random Early Detection gateways for Congestion Avoidance*, IEEE/ACM Transactions on Networking.

- http://itexpressvn.com/vi/tin-tuc/video-huong-dan-thiet-ke-website-ban-hang--bang-asp.net-tu-a-z-142.html.

1. **SYSTEM OVERVIEW**

- The system of websites selling mobile phones includes interface pages so that customers and administrators can operate in a decentralized manner. Customers can register for an account and log into the system. After logging in, customers can order products and pay their orders in the form of online payment. Customers can perform operations such as product vehicles, view website news, search products as they wish, contact and manage their shopping carts.

- Each customer order has been ordered, the system will send an email to the customer email to inform the customer of the order.

- The administrator will have the same properties as the user, but will have the highest license level. They have most of their own user functions and functions: Customer account management, product management, product management by category, feedback management.

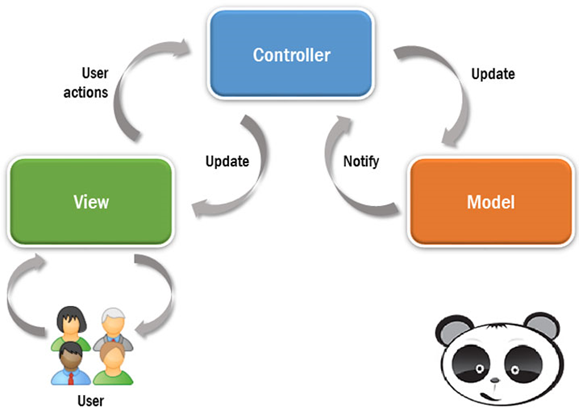
1. **SYSTEM ARCHITECTURE**
   1. Architectural Design

- Model MVC (Model - View - Controller) is a software architecture or design model used in software engineering. It helps developers to separate their applications into 3 different components Model, View and Controller. Each component has a separate task and is independent of the other components. Components of MVC:

+ Model: This component contains all the logic business, processing methods, database access, data description objects such as classes, processing functions, etc.

+ View: Responsible for displaying information, interacting with users, containing all GUI objects such as textboxes, images ... Simply put, it is a collection of forms or HTML files.

+ Controller: Take responsibility to receive requests from the user and call the correct methods to handle them ... For example, this component will receive requests from url and forms to manipulate directly with the Model.



- How MVC model works: Here is a simple way to describe the event flow processed in MVC:

+ User interacts with View, by clicking the button, the user sends the request.

+ Controller receives and redirects them to the right method in Model.

+ Model receives information and executes requests.

+ When the Model finishes processing, the View receives the results from the Model and displays it to the user again.

* 1. Decomposition Description

|  |  |  |
| --- | --- | --- |
| **STT** | **Use case** | Description |
| 1 | Sign up | Sign up for an account |
| 2 | Log in | Log into the system |
| 3 | See the product | See details of the products |
| 4 | Find products | Search product |
| 5 | Order | Order |
| 6 | Pay | Pay by cash on delivery or through online payment services |
| 7 | Watch news | See the news on the website |
| 8 | Contact | Send requests to the website and see the address of the store on the map |
| 9 | Cart management. | Customers manage their own shopping carts (can add, edit, delete selected items) |
| 10 | Manage contact information | Admin manages customer contact information |
| 11 | Customer account management. | Admin manages accounts of customers  + Add new  + Edit  + Show  + Delete |
| 12 | Product Management. | Admin manages the products  + Add new  + Edit  + Show  + Delete |
| 13 | Order management | Admin manages customer orders placed  + Add new  + Edit  + Show  + Delete |

* 1. Design Rationale

- Advantages of MVC model.

+ Easy testing: With MVC, you can easily check and review software errors before reaching consumers, ensuring higher quality and reputation.

+ Control function: On website platforms, programming languages ​​like CSS, HTML, Javascript have an extremely important role. Using MVC pattern will help you have a superior control set on the basis of modern languages ​​in many different forms.

+ View and size: View will be the place to store the data. The more requests made, the bigger the file size. Meanwhile, the network connection also reduces the load speed. Using MVC model will help you save the optimal bandwidth area.

+ Soc function (Separation of Concern): This function allows you to clearly separate parts such as Model, interface, data, business.

+ Combine: The integration in MVC model allows you to freely write code on the website platform. At that time, your server will be greatly reduced load.

+ Simple: This is a model with relatively simple structure. Even if you don't have too much expertise, you can use it.

+ Light, save bandwidth: MVC does not use viewstate, so it saves quite a lot of bandwidth. When used, users can use web applications that need to interact and send and receive data continuously. Therefore, reducing bandwidth helps the website work better and more stable.

- Why choose MVC without choosing 3 Layer:

+ The similarity between two models:

* Both are architectural models applied in programming divided into 3 components.
* Separation of dependencies between programming languages, development environment, and logic processing
* The components in each model have many similarities such as Presentation Layer and View

+ Different:

* The most obvious difference is the handling of requests (workflow) of the two models
* In MVC, the Model component holds the functionality of both Business and Data Access layers in the 3 Layer model.
* In the 3-layer model: Once encountering an error (in the case of incorrect data), whichever layer is used, then toss it one step above the upper layer to the GUI and then throw it to the user. The data is transferred between the layers through an object called Data Transfer Object (DTO), this is simply the Class representing the objects stored in the Database. Organize 3 layer model: There are many ways to name the components of 3 layers such as: 1: GUI, BUS, DAL, way 2: GUI, BLL, DAO, DTO, way 3: Presentation, BLL, DAL

- Therefore, 3 Layer is often used in application programming. Meanwhile, MVC is preferred in website programming because of its flexibility and does not require DTOs like 3-Layer.

1. **DATA DESIGN**
   1. Data Description

- A website that sells phones will post the latest and featured phones to customers, the site also manages the regular upload of the latest phone models and promotional items on the site. The following is a description of the site's data:

- A website consists of many pages, each with its own theme and function. Each topic must have a link to each page, with the name and topic, and content of each page. The homepage will summarize the content of the entire website including the latest phone models, a summary of each topic, promotional items and manufacturer's information. The topic page will post news about that topic. And login page to buy products. Working site of customers and administrators.

- Each topic must be different and different from other topics. Administrators will manage the number of topics available on the site and the name of each topic. Each topic will have a keyword to identify the topic. Each topic will have its own interface for each topic and be suitable for that topic.

- Each customer must provide their personal information as well as their address and phone and email. Each customer will have an account to distinguish them from each other.

- Administrator will store the number of times and the date that customers log in, log out, purchase and order time.

4.1.1. Registration

- Each customer must register personal information including: full name, date of birth, address, email, phone number, username, login password.

4.1.2. Log in

- Customers must login to purchase products including: login name, login password.

- Administrators must log in to manage accounts added, deleted, updated products

4.1.3. Statistical reports are generated

- Statistics on the customer situation for the site:

+ How many customers come to the site in a day?

+ Which row is best for?

+ Which product is bought a lot?

* 1. Data Dictionary

**-** Overview scheme

|  |  |  |
| --- | --- | --- |
| **STT** | **Name** | **Description** |
| 1 | Products | List of products information |
| 2 | Content | News information of the store |
| 3 | FeedBack | Save feedback |
| 4 | Category | List of product lines |
| 5 | Contact | Contact Info |
| 6 | User | List of account information |
| 7 | Order | General information when customers order |
| 8 | OrderDetail | Details of the order when the customer orders |

- Detailed scheme

+ The table contains information Products

|  |  |  |  |
| --- | --- | --- | --- |
| **Products** | | | |
| Field name | Type | Max Length | Descrition |
| Id | bigint |  | Save ID code |
| Name | nvarchar | 250 | Save product's name |
| MetaTitle | varchar | 250 | Save product title name |
| Description | nvarchar | 500 | Save product description |
| Image | nvarchar | 250 | Save product image |
| Price | decimal | 18, 0 | Save product price |
| PromotionPrice | decimal |  | Save price when promotion |
| IncludedVAT | bit |  | Save tax |
| Quantity | int |  | Save the number of products |
| CategoryID | bigint |  | Save type Id |
| Detail | ntext |  | Save product details |
| Warranty | int |  | Save product warranty period |
| CreatedDate | datetime |  | Save product creation date |
| CreatedBy | varchar | 50 | Save creator information |
| ModifiedDate | datetime |  | Save date modified product information |
| ModefiedBy | varchar | 50 | Save modifier information |
| ViewCount | int |  | Save views of the product |

+ The table contains information Content

|  |  |  |  |
| --- | --- | --- | --- |
| **Content** | | | |
| Field name | Type | Max Length | Descrition |
| Id | bigint |  | Save ID code |
| Name | nvarchar | 250 | Save content name |
| MetaTitle | varchar | 250 | Save content headline name |
| Description | nvarchar | 500 | Save content description |
| Image | nvarchar | 250 | Save Image |
| CategoryID | bigint |  | Save Type Id |
| Detail | ntext |  | Save details of the content |
| Warranty | int |  | Save uptime of the content |
| CreatedDate | datetime |  | Save content creation date |
| CreatedBy | varchar | 50 | Save originator information |
| ModifiedDate | datetime |  | Save Date modified content information |
| ModefiedBy | varchar | 50 | Save modified person information |
| ViewCount | int |  | Save views of the content |

+ The table contains information FeedBack

|  |  |  |  |
| --- | --- | --- | --- |
| **FeedBack** | | | |
| Field name | Type | Max Length | Descrition |
| ID | int |  | Save ID code |
| Name | nvarchar | 50 | Save name of the respondent |
| Phone | nvarchar | 50 | Save telephone number of respondent |
| Email | nvarchar | 50 | Save email the respondent |
| Address | nvarchar | 50 | Save responder's address |
| Content | nvarchar | 250 | Save feedback content |
| CreateDate | datetime |  | Save timely send feedback |
| Status | bit |  | Save status |

+ The table contains information Category

|  |  |  |  |
| --- | --- | --- | --- |
| **Category** | | | |
| Field name | Type | Max Length | Descrition |
| Id | bigint |  | Save ID code |
| Name | nvarchar | 250 | Save product's name |
| MetaTitle | varchar | 250 | Save product title name |
| DisplayOrder | int |  | Save displayOrder |
| SeoTitle | nvarchar | 250 | Save title |
| CreatedDate | datetime |  | Save time from creating product information |
| CreatedBy | varchar | 50 | Save creator information |
| ModifiedDate | datetime |  | Save date modified product information |
| ModefiedBy | varchar | 50 | Save modifier information |
| Status | int |  | Save status |

+ The table contains information User

|  |  |  |  |
| --- | --- | --- | --- |
| **User** | | | |
| Field name | Type | Max Length | Descrition |
| Id | bigint |  | Save ID code |
| UserName | varchar | 50 | Save login username |
| PassWord | varchar | 32 | Save account password |
| Name | nvarchar | 50 | Save user name |
| Address | nvarchar | 50 | Save address |
| Email | nvarchar | 50 | Save email |
| Phone | nvarchar | 50 | Save phone number |
| CreatedDate | datetime |  | Save time to create an account |
| CreatedBy | varchar | 50 | Save information creator |
| ModifiedDate | datetime | 50 | Save date of modification of account information |
| ModefiedBy | varchar | 50 | Save modifier information |
| Status | bit |  | Save status |

+ The table contains information Order

|  |  |  |  |
| --- | --- | --- | --- |
| **Order** | | | |
| Field name | Type | Max Length | Descrition |
| Id | bigint |  | Save ID code |
| CreatedDate | datetime |  | Save order creation time |
| CustomerID | bigint |  | Save Customer ID |
| ShipName | nvarchar | 50 | Save customer name |
| ShipMobile | varchar | 50 | Save Customer phone number |
| ShipAddress | nvarchar | 50 | Save customer address |
| ShipEmail | nvarchar | 50 | Save customer email |
| Status | int |  | Save status |

+ The table contains information OrderDetail

|  |  |  |  |
| --- | --- | --- | --- |
| **OrderDetail** | | | |
| Field name | Type | Max Length | Descrition |
| ProductID | bigint |  | Save product ID number |
| OrderID | bigint |  | Save order ID number |
| Quantity | int |  | Save amount |
| Price | decimal | 18, 0 | Save price |

1. **COMPONENT DESIGN**

## 5.1 Product module

- In this module, you can share your products with a group of products according to a certain group criteria, making it convenient for viewers to search for the product.

- Presenting detailed information, photos and prices of products to help customers easily compare and choose between products.

## 5.2 Online payment module

- A website cannot afford to operate. Online payment is a form of payment that is becoming popular today due to the strong development of e-commerce.

- Online payment helps customers be more convenient in shopping, thereby stimulating shopping of customers.

## 5.3 Customer account management module

- Keep information about customers, add, delete, edit, search and lock accounts and account authorization

## 5.4 Search base module

- This module will directly affect the user experience and attract customers. Their customers are always looking for product information as quickly as possible without making too much progress and time. Help them find all the information on the site with relevant keywords.

## 5.5 Cart module

- The module includes the shopping cart function, order, customer management to help customers easily select goods, register orders or add products. This module should integrate online payment on the site.

## 5.6 New product module

- Your website cannot be without this module for product or business update. It will help your customers pay attention to you.

## 5.7 Introduction of products, service modules

- Any sales website needs this module to show customers what specific information about the product you are offering? Information about features, origins, product prices and non-intuitive.

- The display of products should be divided by categories, each category will have many products inside and presented in the most convenient list to track customers.

## 5.8 Product management module

- The module will assist in updating, editing, adding or deleting products and services. This module needs to be optimized for administrators.

1. **HUMAN INTERFACE DESIGN**
   1. Overview of User Interface

- Order function

+ Description: When users want to order products, users must log in to the system if they do not have an account, they must register.

+ Time can order: 0: 00h - 24h

+ If the purchase is successful, the user will receive an email from the system sent to the email entered by the customer in the account information

- Contact function

+ Description: When users want to contact the administrator, they can use the contact function on the website and enter the content they want to feedback to the store. Administrator will contact customer via email or phone number provided by customer.

- Find products

+ Description: When users want to find the product they want, they can enter the product name or related keywords. The system will display products related to the keyword that you have entered. If the product is not in the system or the user enters the word without thinking, the system will display a message that there is no product that the customer wants to find.

- View product details

+ Description: When users want to view a product details, users can click on the image of that product. The system will automatically go to the product details page for customers to view and add to their shopping carts

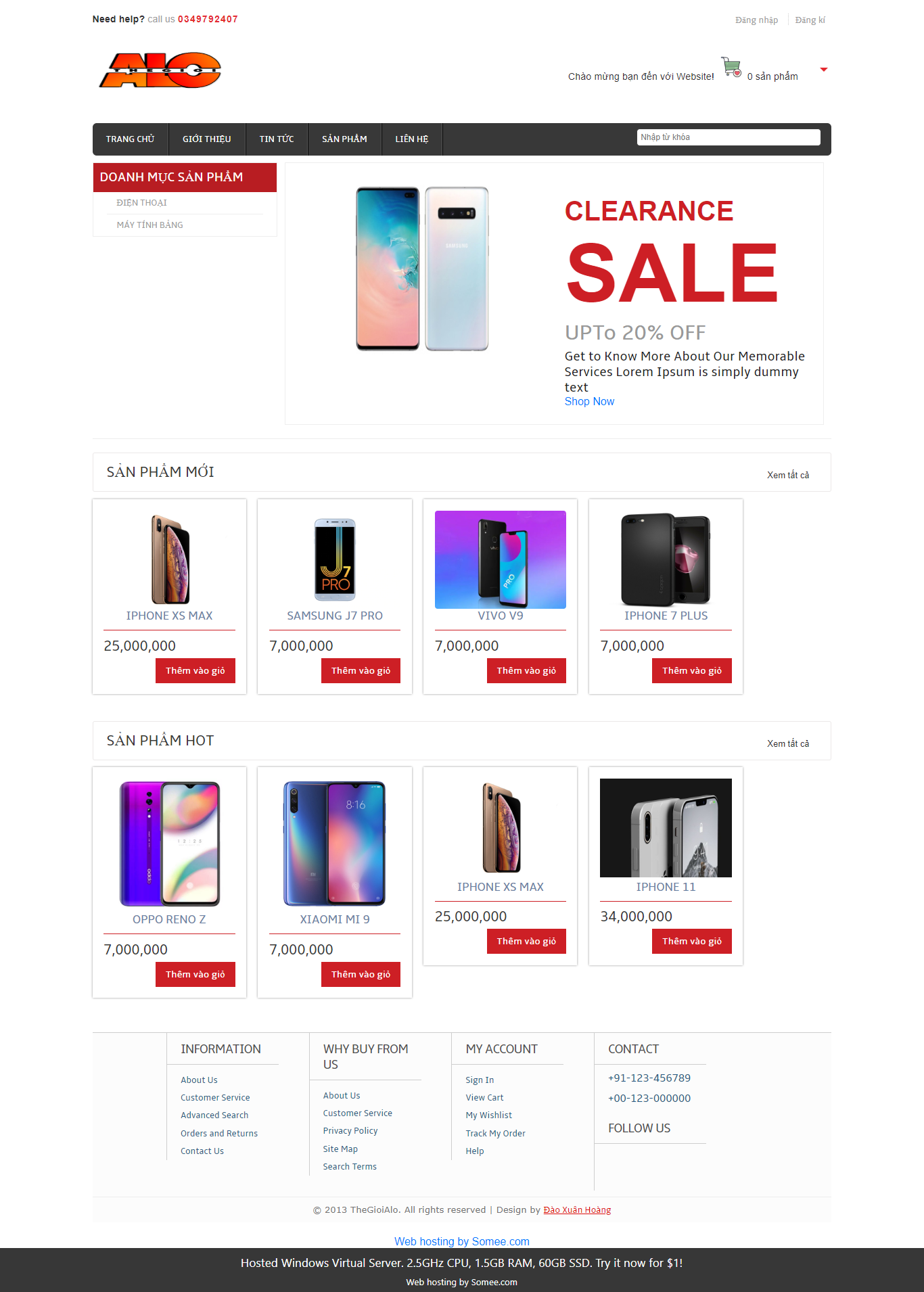
- Shopping cart management

+ Description: Customers can add and edit products in their shopping cart and customers can increase or decrease products already in the cart and click on the system update function will update the shopping cart again. for customers. In addition, customers can delete all products already in the cart by deleting the cart in the page.

* 1. Screen Images

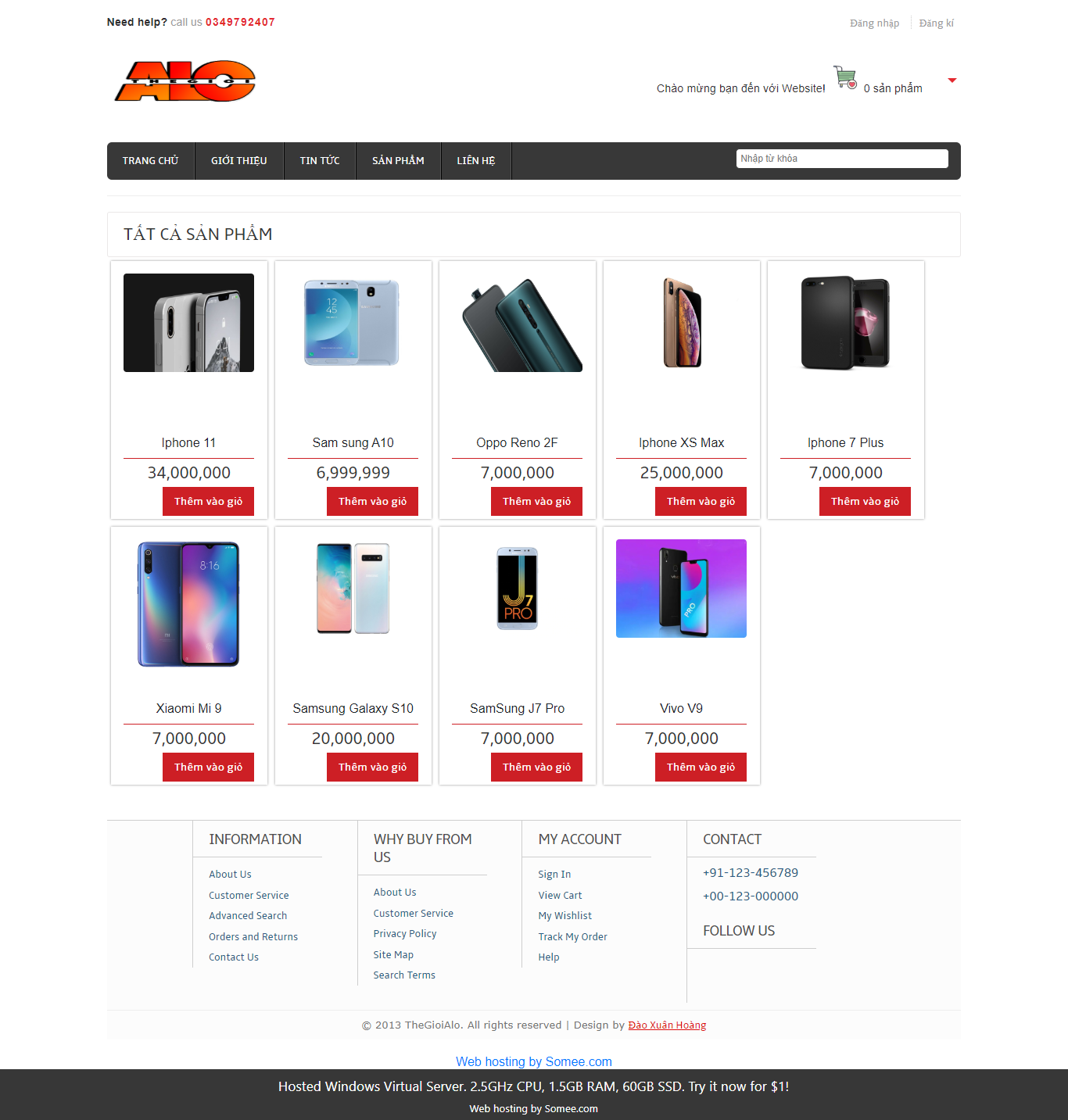
- User interface.

+ Homepage interface



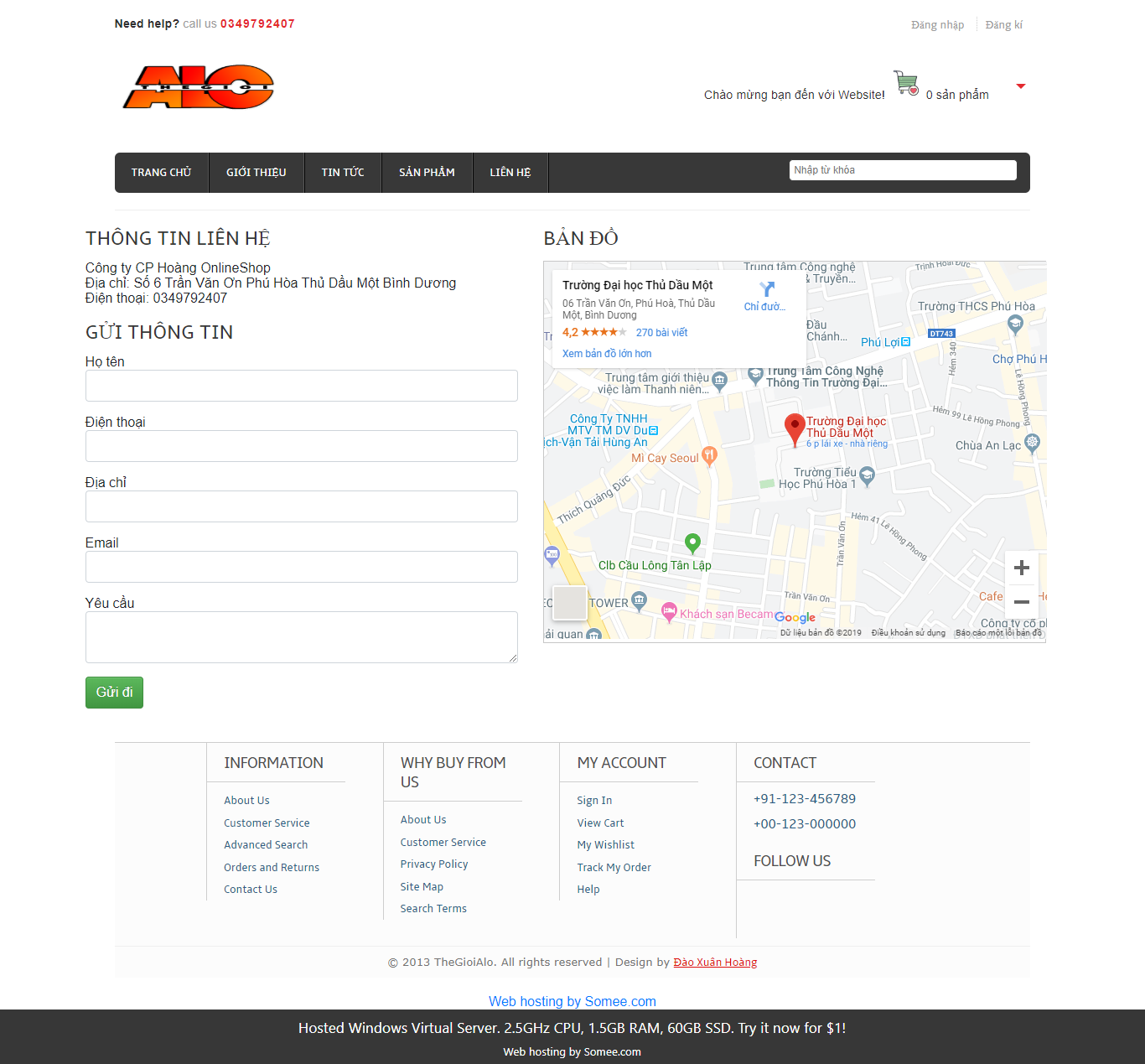
**Figure 6.1** The homepage interface

+ Product page interface



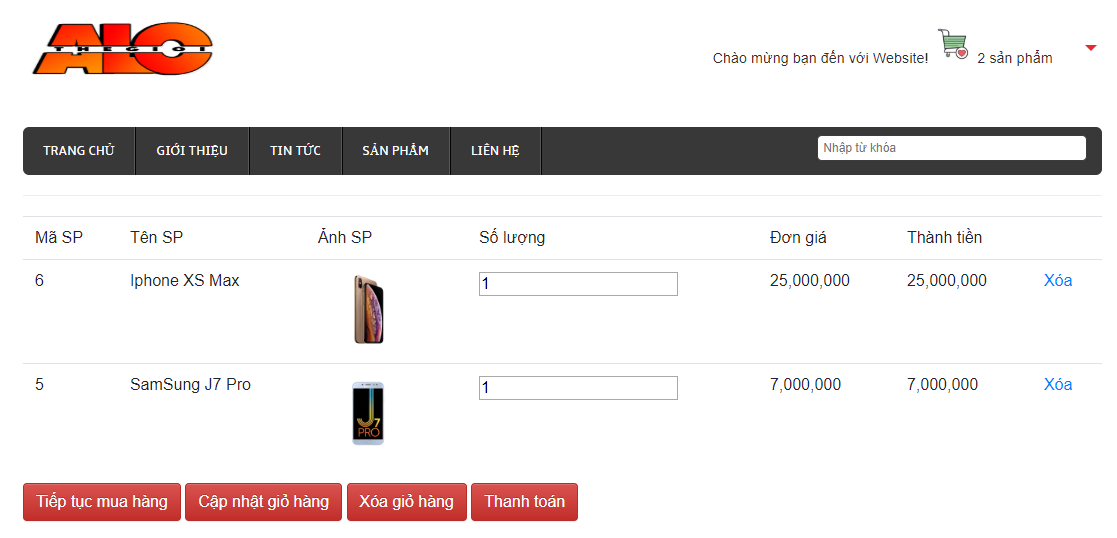
**Figure 6.2** The product page interface

+ Contact page interface



**Figure 6.3** The Contact page interface

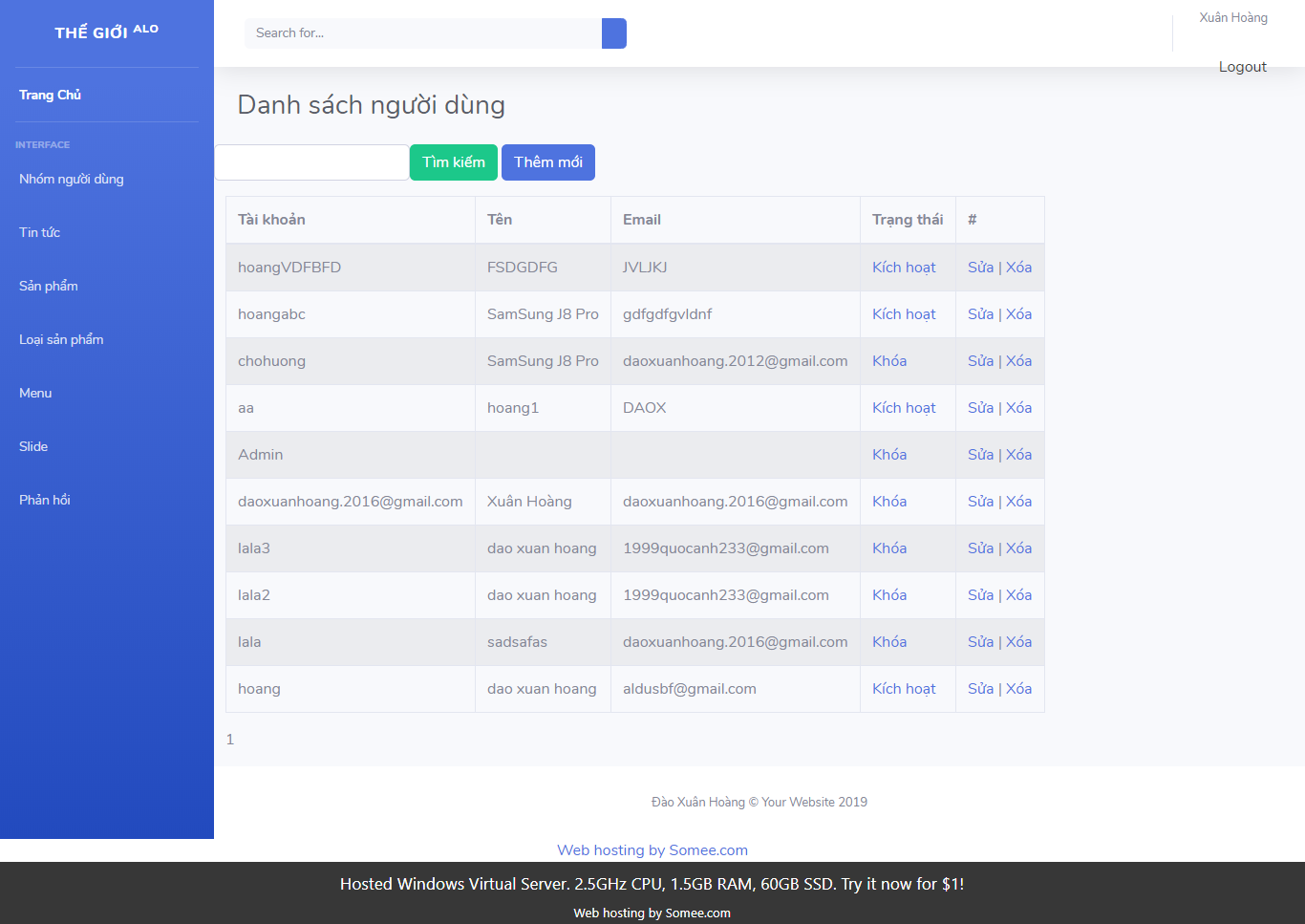
+ Cart management interface (when products are available)



**Figure 6.4** Cart management interface (when products are available)

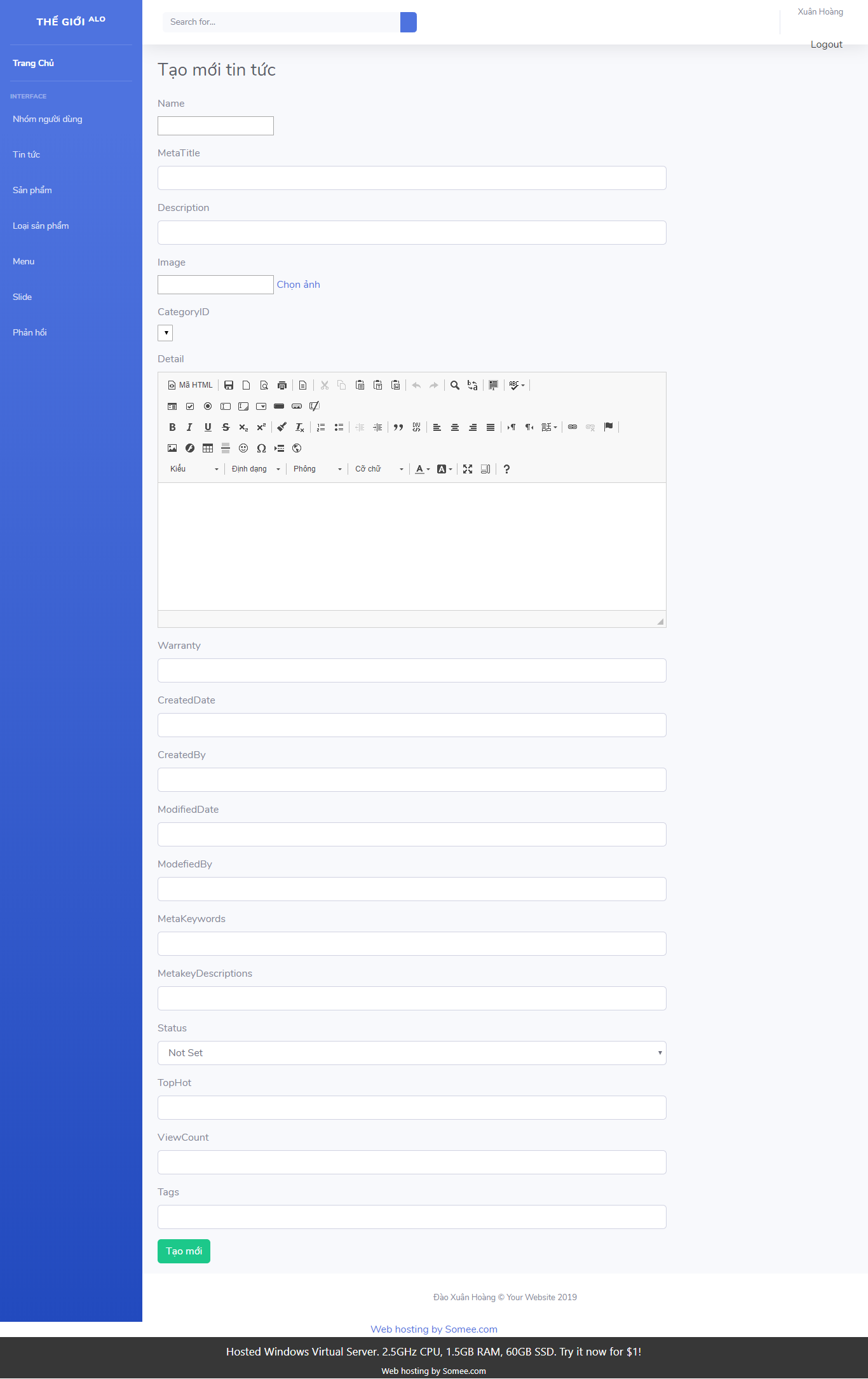
- Admin interface

+ Customer account management interface



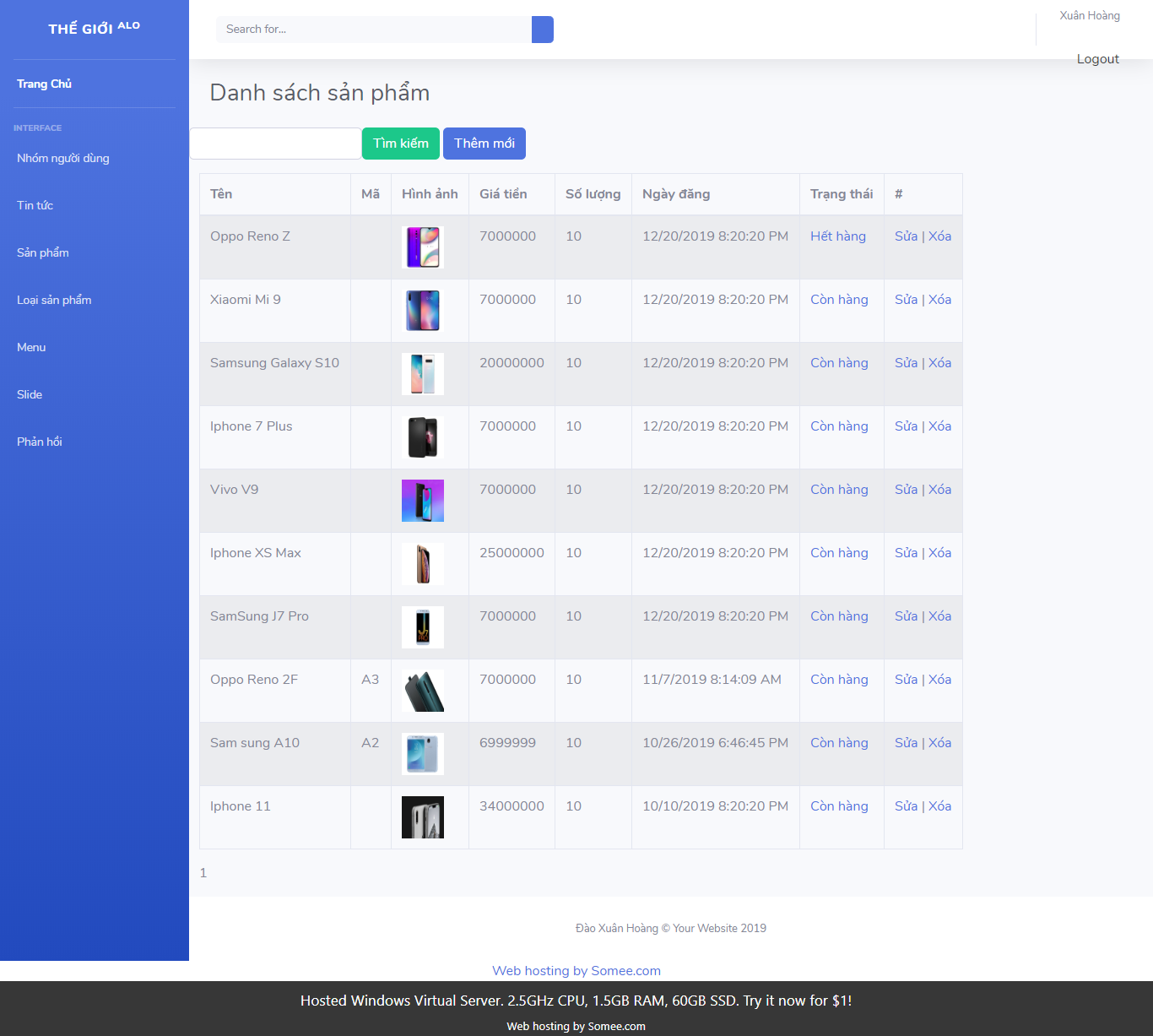
**Figure 6.5** Customer account management interface

+ News management interface



**Figure 6.6** News management interface

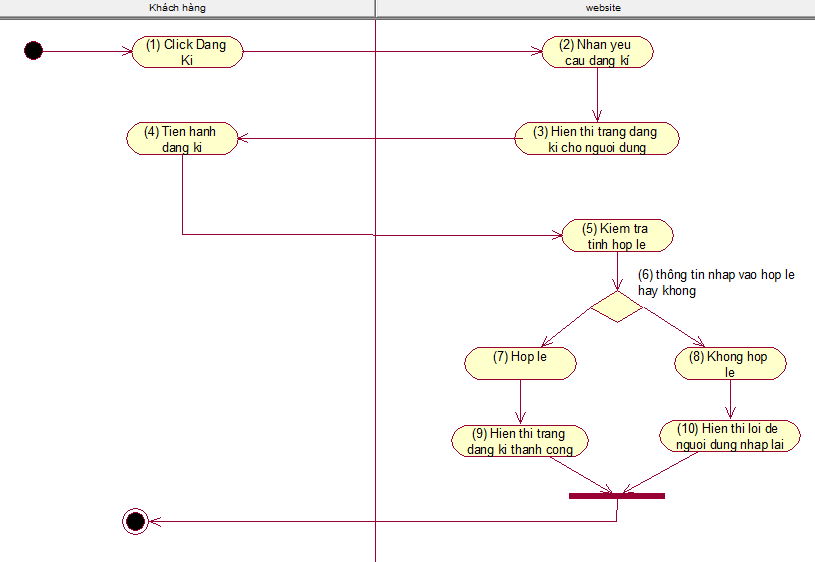
+ Product management interface



**Figure 6.7** Product management interface

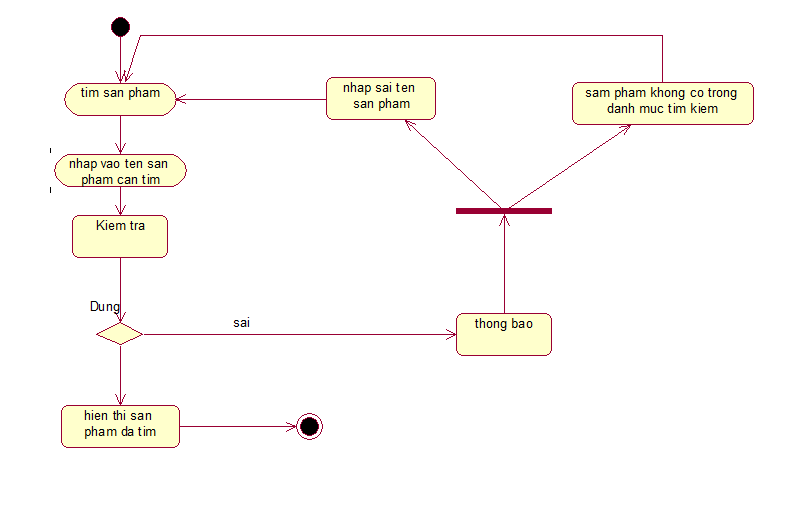
* 1. Screen Objects and Actions

- Register members activity diagram



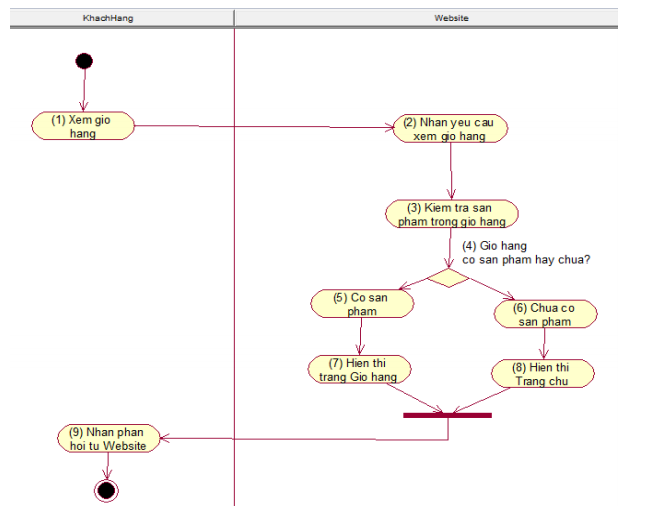
**Figure 6.8** Register members activity diagram

- Product search activity diagram



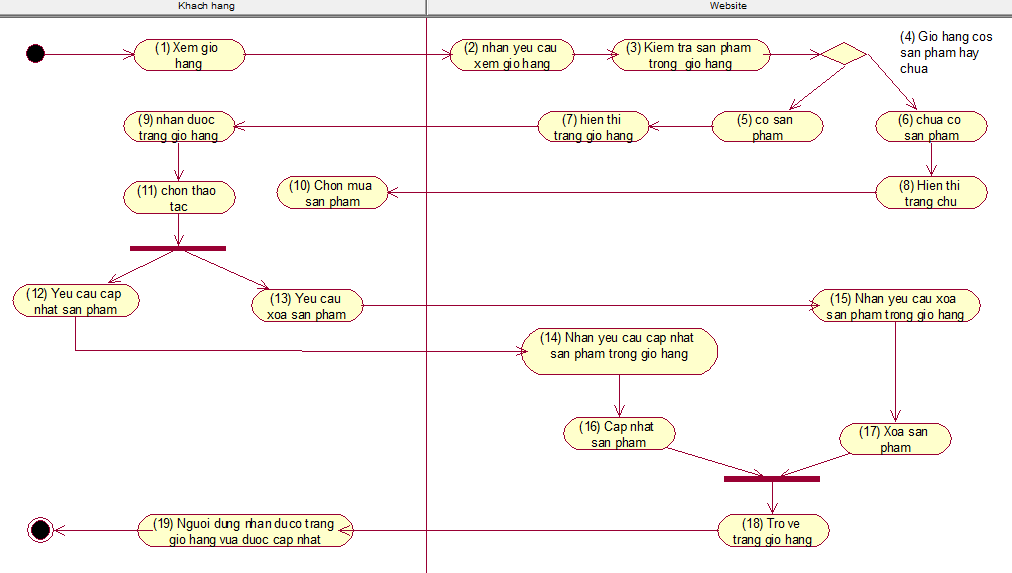
**Figure 6.9** Product search activity diagram

- Cart view activity diagram



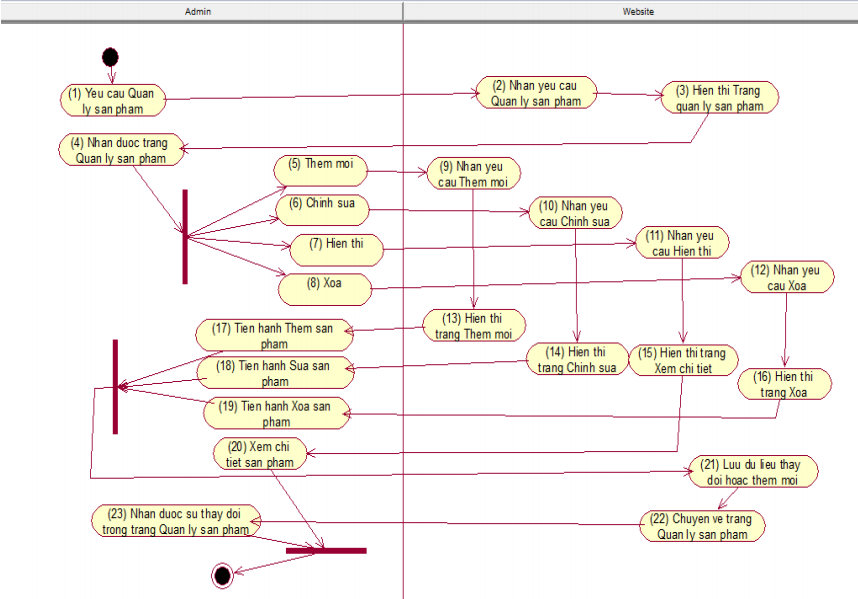
**Figure 6.10** Cart view activity diagram

- Cart update activity diagram



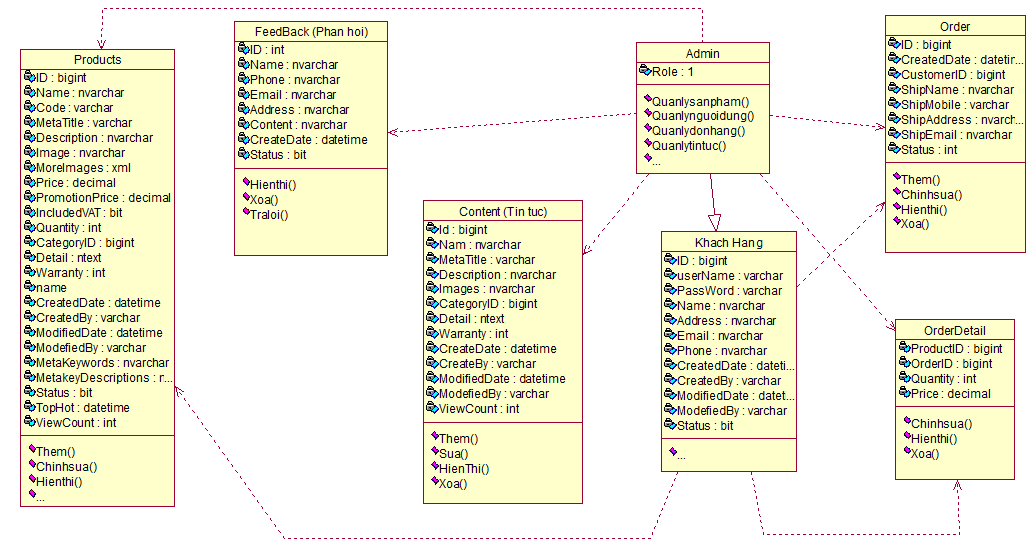
**Figure 6.11** Cart update activity diagram

- Product management activity diagram



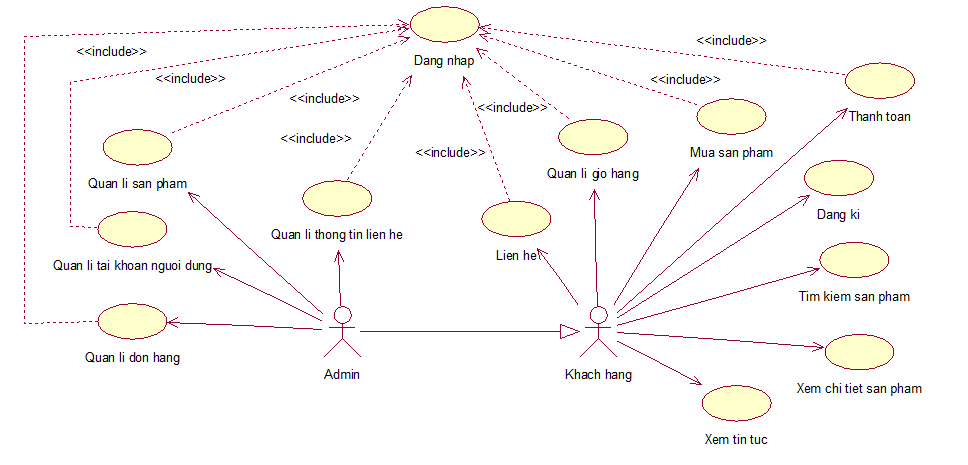
**Figure 6.12** Product management activity diagram

1. **REQUIREMENTS MATRIX**
   1. Class Diagram



**Figure 7.1** Class Diagram

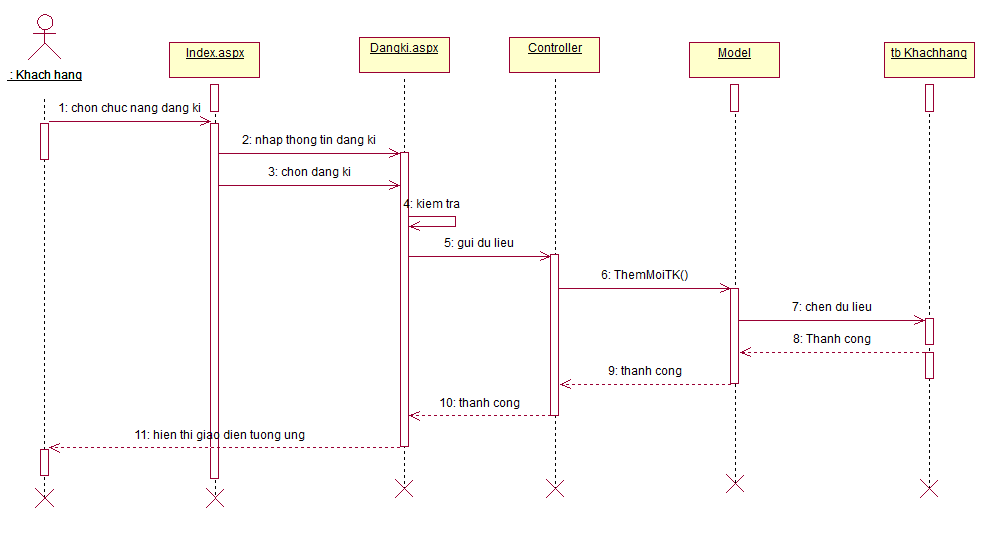
* 1. Use Case Diagram



**Figure 7.2** Use case diagram

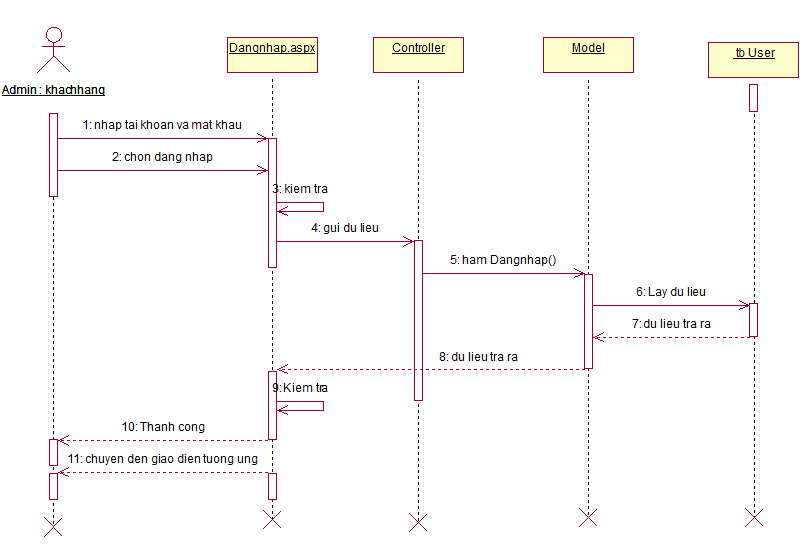
* 1. Sequence Diagram

- Registration



**Figure 7.3** Registration sequence diagram

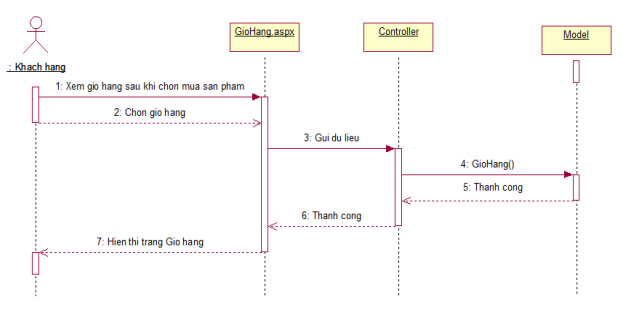
- Login



**Figure 7.4** Login sequence diagram

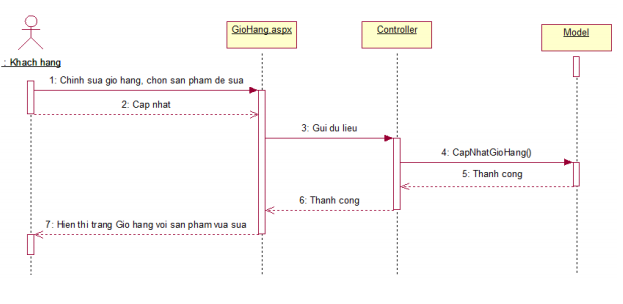
- Sequence management Cart management

+ Sequence chart View cart

****

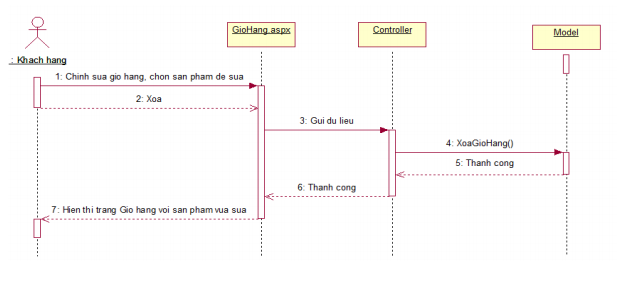
**Figure 7.5** Sequence diagram View cart

+ Sequence diagram Updating cart



**Figure 7.6** Sequence diagram Updating cart

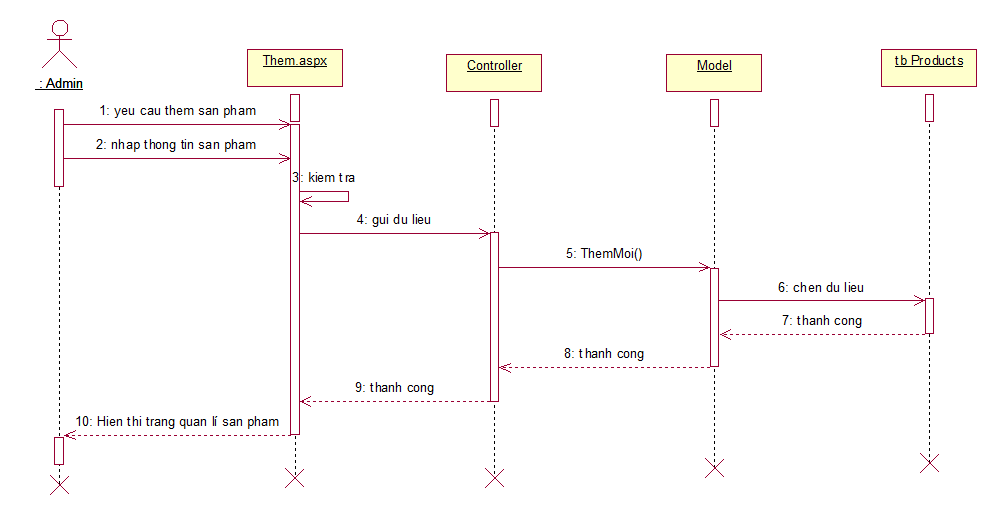
+ Sequence diagram Delete cart



**Figure 7.7** Sequence diagram Delete cart

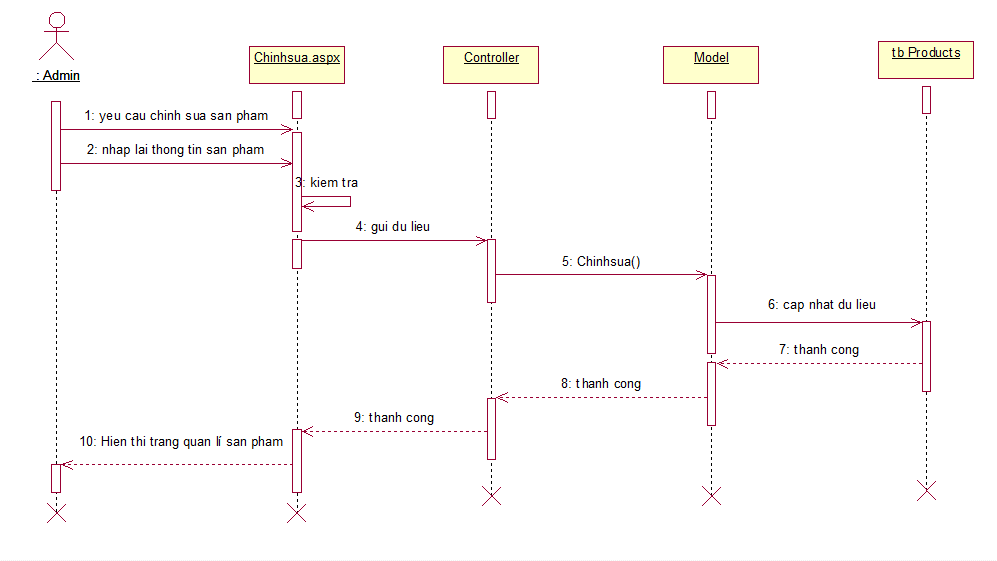
- Sequence diagram Product management

+ Sequence diagram Add products



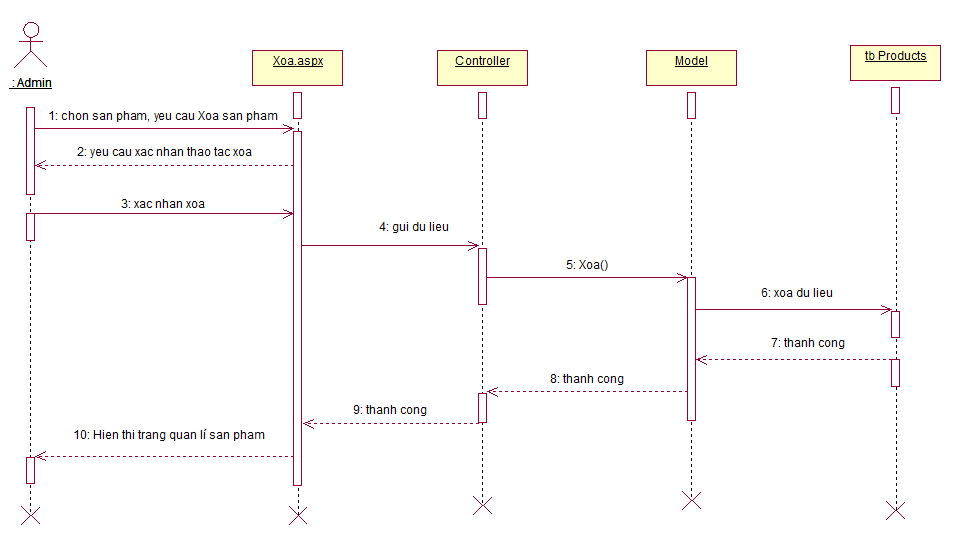
**Figure 7.8** Sequence diagram Add products

+ Sequence diagram Edit products



**Figure 7.9** Sequence diagram Edit products

+ Sequence diagram Delete products



**Figure 7.10** Sequence diagram Delete products

**APPENDICES**

[**Figure 6.1** The homepage interface 13](#_Toc488530548)

[**Figure 6.2** The product page interface 14](#_Toc488530549)

[**Figure 6.3** The Contact page interface 14](#_Toc488530548)

[**Figure 6.4** Cart management interface (when products are available) 15](#_Toc488530548)

[**Figure 6.5** Customer account management interface 16](#_Toc488530549)

[**Figure 6.6** News management interface 17](#_Toc488530548)

[**Figure 6.7** Product management interface 18](#_Toc488530548)

[**Figure 6.8** Register members activity diagram 19](#_Toc488530548)

[**Figure 6.9** Product search activity diagram 20](#_Toc488530548)

[**Figure 6.10** Cart view activity diagram 20](#_Toc488530548)

[**Figure 6.11** Cart update activity diagram 21](#_Toc488530548)

[**Figure 6.12** Product management activity diagram 22](#_Toc488530548)

[**Figure 7.1** Class Diagram 23](#_Toc488530549)

[**Figure 7.2** Use case diagram 24](#_Toc488530548)

[**Figure 7.3** Registration sequence diagram 24](#_Toc488530548)

[**Figure 7.4** Login sequence diagram 25](#_Toc488530549)

[**Figure 7.5** Sequence diagram View cart 25](#_Toc488530548)

[**Figure 7.6** Sequence diagram Updating cart 26](#_Toc488530549)

[**Figure 7.7** Sequence diagram Delete cart 26](#_Toc488530548)

[**Figure 7.8** Sequence diagram Add products 27](#_Toc488530548)

[**Figure 7.9** Sequence diagram Edit products 27](#_Toc488530549)

[**Figure 7.10** Sequence diagram Delete products 28](#_Toc488530548)