

VIETNAM NATIONAL UNIVERSITY OF HO CHI MINH CITY
UNIVERSITY OF INFORMATION TECHNOLOGY



PROJECT REPORT

HOTEO – ONLINE GREEN TEA STORE
WEBSITE

COURSE: SE104.N22.PMCL – Software Engineering

Instructor: Ms. Thai Thuy Han Uyen

Members:

Tran Ngoc Nhu Hao – 21520224

Phan Nhat Hoang – 21520874

Ly Gia Hieu – 21522074

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to the Faculty of Software Engineering at University of Information Technology for providing us a favorable condition to carry out this research project. We would like to extend our heartfelt appreciation to all the lecturers who have taught us with dedication and equipped us with valuable knowledge during the past semester, which has enabled us to have a strong foundation to complete this project. We especially want to thank Ms. Thai Thuy Han Uyen for her enthusiastic guidance and valuable feedback, which helped us to complete this report.

Although we have made our utmost effort to complete the report to the best of our ability and within the given scope, we are aware that there may still be some limitations. We hope to receive your understanding and constructive contributions. Once again, we would like to express our sincere thanks.

Table of Contents

CHAPTER 1: OVERVIEW OF THE PROJECT	5
1. Introduction:.....	5
2. Scope of the project:	5
3. Objectives:	5
4. Methodology:	6
5. Expected Outcomes:	6
CHAPTER 2: PROJECT MANAGEMENT	6
CHAPTER 3: HOTEQ SYSTEM	13
1. Business Model:	13
2. System Architecture:.....	13
3. Requirements Identification:	14
3.1. User requirements:	14
3.2. System requirements:	14
3.3. Functional requirements:.....	15
4. Functional Requirements Table:.....	15
5. Functional Requirements Description:	16
5.1. Account:.....	16
5.2. Product:	16
5.3. Order:.....	17
5.4. Dashboard Statistics:.....	17
6. Use case Diagram:	17
6.1. List of actors:	17
6.2. HOTEQ System Use Case Diagram:	18
6.3. Shopping Cart Use Case Diagram:.....	20
6.4. Product Management Use Case Diagram:.....	22
6.5. Product View Use Case Diagram:.....	24
7. Activity Diagram:	26
7.1. Login Activity Diagram:	26
7.2. Add To Cart Activity Diagram:.....	27
7.3. Add Product Activity Diagram:	28
7.4. Search Activity Diagram:	29

8. Class Diagram:	30
9. Database:.....	31
10. Interfaces:.....	33
10.1. Login interface	33
10.2. Homepage interface	34
10.3. Product Page Interface.....	34
10.4. Introduction page interface	35
10.5. Cart page interface	35
10.6. Dashboard Interface.....	36
10.7. Searching Interface.....	36
11. Testing:	37
11.1. Login testing:	37
11.2. Remove Product testing:.....	37
CHAPTER 4: CONCLUSION	37
1. Results and advantages:	37
2. Limitations of the project:	38
3. Future development:	38

CHAPTER 1: OVERVIEW OF THE PROJECT

1. Introduction:

HOTEO is a green tea store located in Ho Chi Minh City, Vietnam. The store offers a wide range of high-quality green tea products, including loose leaf tea, teabags, and accessories. HOTEO is committed to providing its customers with the freshest and most authentic green tea products, sourced directly from the best tea gardens in Asia.

To expand its customer base and reach a wider audience, HOTEO intends to develop an e-commerce website that can showcase its products and provide customers with a convenient and secure online shopping experience. The website will be designed to meet the needs and expectations of HOTEO's target customers, who are health-conscious and value high-quality products.

2. Scope of the project:

The project will focus on the development of an e-commerce website for HOTEO. The website will have the following features:

- Product catalog: An online catalog of HOTEO's green tea products, with high-quality images, detailed descriptions, and reviews from other customers.
- Shopping cart: A user-friendly shopping cart that allows customers to add or remove items, view the total cost, and make changes before checkout.
- Checkout and payment: A secure and convenient checkout process that supports multiple payment options, including credit cards, debit cards, and bank transfers.
- Shipping and delivery: A reliable shipping and delivery system that provides customers with real-time tracking, delivery status updates, and estimated delivery times.
- Customer service: A responsive and helpful customer service team that can assist customers with any questions or issues they may encounter during the shopping process.

3. Objectives:

The main objective of this project is to develop an e-commerce website for HOTEO, a green tea store based in Ho Chi Minh City, Vietnam. The website will provide customers with an easy-to-use platform to browse and purchase a wide range of high-quality green tea products online. Specifically, the objectives of the project are:

- To develop a user-friendly e-commerce website that provides a seamless online shopping experience for customers

- To showcase the quality of HOTE0's green tea products and build a strong brand image
- To provide customers with secure and convenient payment methods and delivery options
- To gather and analyze feedback from customers to continuously improve the website and enhance customer satisfaction

4. Methodology:

The project will follow a systematic approach to website development, with the following phases:

- Requirements gathering: Identify the needs, goals, and expectations of HOTE0 and its customers, and define the requirements for the e-commerce website.
- Design: Develop a user-friendly interface design that meets the requirements and provides a seamless shopping experience for customers.
- Development: Code and test the e-commerce website using modern web development technologies and programming languages.
- Deployment: Deploy the website to a production environment, configure the servers and databases, and test for performance and scalability.
- Maintenance and support: Provide ongoing maintenance and support for the website to ensure that it remains up-to-date, secure, and reliable.

5. Expected Outcomes:

The expected outcomes of this project include:

- A user-friendly e-commerce website for HOTE0 that provides a seamless online shopping experience for customers.
- A reliable and secure payment and delivery system that ensures customer satisfaction and builds trust in HOTE0's brand.
- Increased sales and revenue for HOTE0, with the potential to expand to other areas and regions in Vietnam.
- Valuable insights into customer preferences, behavior, and feedback, which can be used to improve the website and enhance customer satisfaction.

CHAPTER 2: PROJECT MANAGEMENT

Project Estimation.

Stage	Task	Description	Participants	Duration	Budget
Project manager	Project requirements survey	Collect general requirements about the project	1 person		

	Project Initialization	Notice of project implementation, project creation	1 person	1 day	
	Plan the project scope	Make a project scope plan	1 person	5 days	
	Write a summary report	Summarize all the work into a report	1 person	The final phase of the project	
	Learn from experience	Give the best experience for the next project	1 person	The final phase of the project	
System analysis and design	Detailed specification of requirements from customers	From the requirements obtained from the survey, describe more specifically the necessary requirements of the website	1 person	2 days	
	Describe the system architecture with use case diagrams,	Build website architecture using use case diagrams, sequence of activities	1 person	5 days	

	sequences, and activities				
	Conduct system database design	Build a database structure for the website (including specific tables)	1 person	3 days	
	Conduct website interface design	Build basic forms of website based on UML diagrams	1 person	4 days	
	Write a detailed system analysis	Synthesize reports for a complete design analysis	1 person	2 days	
	Analyze specific requirements for modules	From the information gathered above, make a detailed construction plan the tasks of the building material dealer manager.	1 person	2 days	
Modules	Conduct module design	- Build functions. - Build login form	1 person	3 days	

	Write code for the module	Build forms for website functions	1 person	5 days	
	Proceed to set up the module	Run the built module.	1 person	1 days	
	Module testing	<ul style="list-style-type: none"> - Test the interface, check whether the interface meets the requirements, on the form of the interface layout of the website, the layout is reasonable or not. - Test the ability of the functions to perform. 	1 person	3 days	
	Write a report on the module	Build a detailed description of the module	1 person	1 day	
	Integration of modules	Complete arrangement of modules into a	1 person	1 day	

		unified program for testing			
Integrating and perfecting products	System-wide integration testing	Test overview of the whole system	1 person	1 day	
	Fix existing errors	Fix errors that arise while testing	1 person	2 days	
Complete the website and deliver	Write documentation on how to use the tools in the website	Create a website user manual	1 person	1 day	
	Set up a site maintenance plan	Delivery of project products to customers	1 person	1 day	
	Handing over products as well as related documents to customers	Give customers a maintenance plan	1 person	1 day	
	Project finished	Project summary		1 day	

Process Analysis Of Implementation.

Assignment structure	Activity	Activity name	Inheritance operation	Execution time
J1: Management model survey.	1.1	Project survey, needs analysis	1.1	1 day
	1.2	Write project charter	1.1	1 day
	1.3	Project implementation report	1.2	1 day
	1.4	Create a project scope description	1.3	1 day
	1.5	Create a work breakdown table	1.4	1 day
	1.6	Make a project plan table	1.5	1 day
J2: System analysis and design	2.1	Process analysis	1.6	1 day
	2.2	Make a system use case diagram	2.1	1 day
	2.3	Create a system sequence diagram	2.2	1 day
	2.4	Unify the design diagrams	2.3	1 day
J3:	3.1	Audience analysis	2.2, 2.3, 2.4	1 day

Create a database				
	3.2	Build audience attribute	3.1	1 day
	3.3	Set up the base and enter data for the system	3.1, 3.2	1 day
J4: Build system login function	4.1	Design the login form to the system	3.3	1 day
	4.2	Code of login function to the system	4.1	1 day
	4.3	Test login function	4.1, 4.2	1 day
J5: Build system functionality	5.1	Form design	4.3	1 day
	5.2	Write triggers, stored procedure	5.1	1 day
	5.3	Code functions	5.2	1 day
	5.4	Test the built modules finished	5.3	1 day
J6: Test	6.1	Check if the interface is reasonable and	4.3	1 day

		convenient for users		
	6.2	Check if the input is correct	6.1	1 day
	6.3	System testing	6.2	1 day
	6.4	Fix existing errors	6.3	1 day
J7: Website Setup	7.1	Write a report of the whole module	6.3, 6.4	1 day
	7.2	Write user manuals	7.1	1 day
J8: Integration and maintenance	8.1	Make a website maintenance plan	6.4, 7.1	1 day
	8.2	Project finished		

CHAPTER 3: HOTE0 SYSTEM

1. Business Model:

HOTE0's business model is based on selling high-quality green tea products directly to customers through its physical store and online channels. The store offers a wide range of green tea products and accessories, sourced directly from the best tea gardens in Asia. HOTE0's products are priced competitively, and the store offers discounts and promotions from time to time to attract and retain customers.

Through its e-commerce website, HOTE0 aims to expand its customer base and increase revenue by providing customers with a convenient and secure online shopping experience. The website will offer a user-friendly interface, a wide range of products, reliable payment and delivery options, and helpful customer service.

2. System Architecture:

The e-commerce website for HOTE0 will be developed using a modern web development stack, including the following components:

- Front-end: The front-end of the website will be developed using HTML, CSS, and JavaScript. The website will provide a user-friendly interface that is responsive and compatible with different devices and screen sizes.

- **Back-end:** The back-end of the website will be developed using Node.js, a popular platform for building scalable and highly performant web applications. The back-end will handle the business logic of the website, including user authentication, shopping cart management, order processing, and payment and delivery integration.
- **Database:** MySQL is essential for storing data and optimizing queries on ecommerce websites such as HOTE0. It's a reliable and widely used relational database management system for web development. With MySQL, you can store information about products, orders, customers, and integrate payment and delivery. This allows HOTE0 to easily manage and control customer data through the MySQL database management interface. Additionally, MySQL supports query optimization, helping the HOTE0 website load faster and better respond to customer demands.

3. Requirements Identification:

3.1. User requirements:

The website management team requires website for managing posts and related features, including:

- Information about products
- Product quantity status
- Quick and accurate price updates

3.2. System requirements:

With perfect features of the tea website, it should support users in search, statistics, and fast access. Customers will undoubtedly be impressed with the user-friendly interface, perfect functions, and the development of new features and applications for the tea website:

- Optimize the database, the website is designed as an information management system for products, specifically tea.
- Upgrade system features comprehensively, web applications that help users maximize the information on the website.
- The website should perform well on common browsers such as Google Chrome, Firefox... (note after)
- Support website development during website operation.
- In addition, we build an expanded website system to facilitate upgrades without the need for new designs when requested by customers.
- The system must be easy to use, with fast and accurate data access, and simple operations.
- The user interface and machine should be scientifically designed and user-friendly.

3.3. Functional requirements:

Data storage:

- User information: Name, phone number, address, ...
- Product information: prices, quantity, tea descriptions...

Processing:

- Process orders quickly and accurately.
- Process product status and update in a timely manner.

Non-functional requirements:

- User-friendly interface.
- Stability, efficient processing.
- Backup and restore in case of emergency.
- Inheritance.

Performance requirements of the system:

- Efficiently perform system management functions, support system management activities.
- The reporting system is accurate and complete, accessible, and able to monitor all system activities at any time, helping managers control system activities effectively.

4. Functional Requirements Table:

ID	Name of requirement	Priority	Diagram	Test case
1 Account				
1.1	Display	Critical		
1.3	Log in the system	High		
1.4	Register	High		
1.5	Admin Register	High		
1.5	Log out the system	Critical		
1.6	User management	High		
2 Product				
2.1	Display	Critical		
2.2	Search	Critical		
2.3	View product information	High		
2.4	Shopping Cart	High		
2.5	Product Management	High		
3 Order				
3.1	Display	Critical		
3.2	Checkout order	High		
3.3	Process order	Low		
4 Dashboard Statitics				
4.1	Display	Critical		

4.2	Store orders information	High		
4.3	Analyze orders and produce the top best seller products.	High		
4.4	Analyze orders and produce revenue statistics by time.	High		

5. Functional Requirements Description:

5.1. Account:

- Log in the system: Admin and user log in in the same session.
- Log out the system
- Register:
 - + Username: Customer enter the username they want.
 - + Password: Customer enter the password they want.
- Admin Register:
 - Admin will register in the register website for admin without the navigation from the main website to secure store information.
 - + Username: Users enter the username they want.
 - + Password: User enter the password they want.
 - Type of account:
 - + Admin: can edit all of data
 - + Customer: can do operations on products and place orders.
- User Management: The admin can do some action in this page of HOTE0 website such as: storage the log in information, listing the account and remove the invalid account.

5.2. Product:

- Search product information: users are allowed to search for product by typing product's keyword on the search bar and press Enter.
- Product information display
 - Users have the option to explore all available products by visiting the Delight page or conducting a search for the feature of product.
 - On the event of clicking on the product, users will navigate to that specific product page.
- Shopping Cart
 - Add product to the shopping cart: user clicking on the "Cart" button on the Delights page or the "Add to cart" button on the product specific product page.

- Remove product form the cart: On the event of clicking “-“ button, the product will be delete form the user’s shopping cart.
- Product Management
 - Add new product: User clicking on the “Add” button on the Product Management page and navigate to the Add new product page.
 - Remove product: User clicking on “Remove” button on the product cards to remove all product information on Product Management information.
 - Update product information: User clicking on the “Update” button on the Product Management page and navigate to the Update page.

5.3. Order:

- Check out order:
- After user clicking the button “Order/ Make the payment” on Shopping Cart page, website will navigate to the Order page.
- User can see what product being purchased, the information include:
 - ✚ Product image
 - ✚ Product title
 - ✚ Quantity
 - ✚ Price
- Order process: After completing order decision, user will click “Order / Make the payment” and start entering order information.
The order process start with entering order information such as: adress, full name,...

5.4. Dashboard Statistics:

- Order information storage: The order information will be storage and listing in this page. Admin is the actor can see and update it.
- Analysist: With data coming from the order information store, the system will automatically calculate and give dashboards to admins about parameters such as top 5 best-selling lists, revenue over time.

6. Use case Diagram:

6.1. List of actors:

SN	Name of actor	Meaning, Note
1	Admin	Updating the website, looking up order information, creating revenue dashboard by time.
2	Customer	Looking up product information, placing orders, looking up order information.

6.2. HOTEQ System Use Case Diagram:

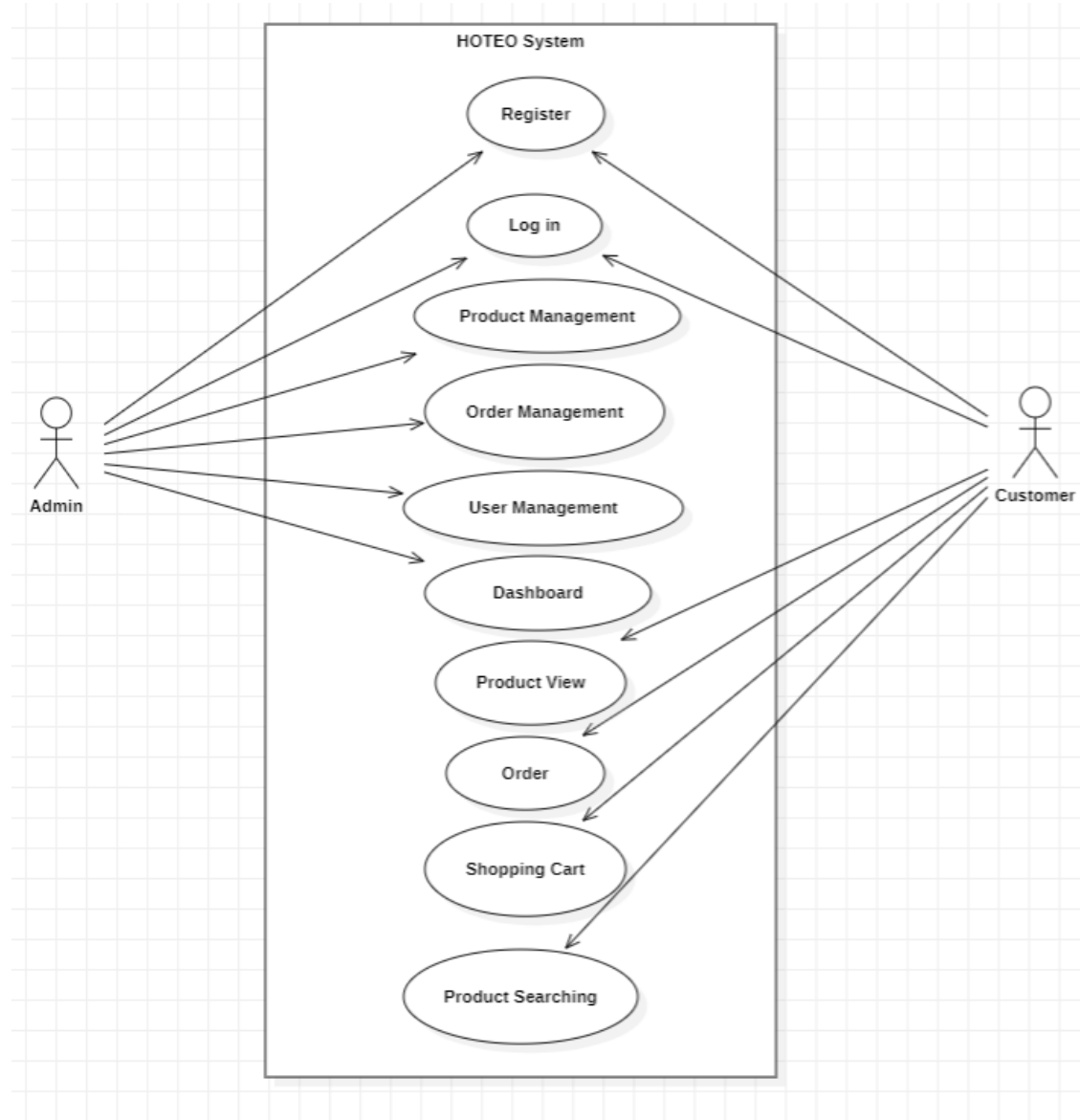


Image 1: HOTEQ use cases

Table 1: List of HOTEQ use cases

STT	USE CASE	Meaning, Note
1	Register	Users can register by providing their personal information such as email and password.

		This feature is of critical priority as it is the first step for users to interact with the online tea-selling website.
2	Log in	This use case describe the compulsory function of the action before user entered to the HOTEQ system when they want to contact with the website.
3	Product Management	This use case describes the function of updating product information, adding product information or removing product in the Admin's system. When information about a product changes, the Admin is responsible for doing action to the product information in the system.
4	Order Management	Admin manages orders when customers place orders for the store's products.
5	User Management	Admin manages the user information when the customer make an account in the website. Admin can see the account storage, do some action such as: remove account, take the information
6	Dashboard	This use case describes the function of statistics on inventory items, best-selling items, and revenue statistics by week and month. It also involves order management, including tracking unresolved orders, identifying orders that have been resolved, and how they were resolved.
7	Product View	This use case describes the function of viewing product information by users such as viewing the product origin, technical specifications of the product, etc.
8	Order	This use case describes the function of customer placing orders, in which customers can change the quantity of products as desired. They can cancel the order if they change their mind.

9	Shopping Cart	The function of a shopping cart is to hold the items that a customer has selected to purchase.
10	Product Searching	This use case describes the function of searching for products by the customers. This function makes it easier to search for products when there are many products stored in the system, as finding a specific product according to specific requirements can be a difficult task.

6.3. Shopping Cart Use Case Diagram:

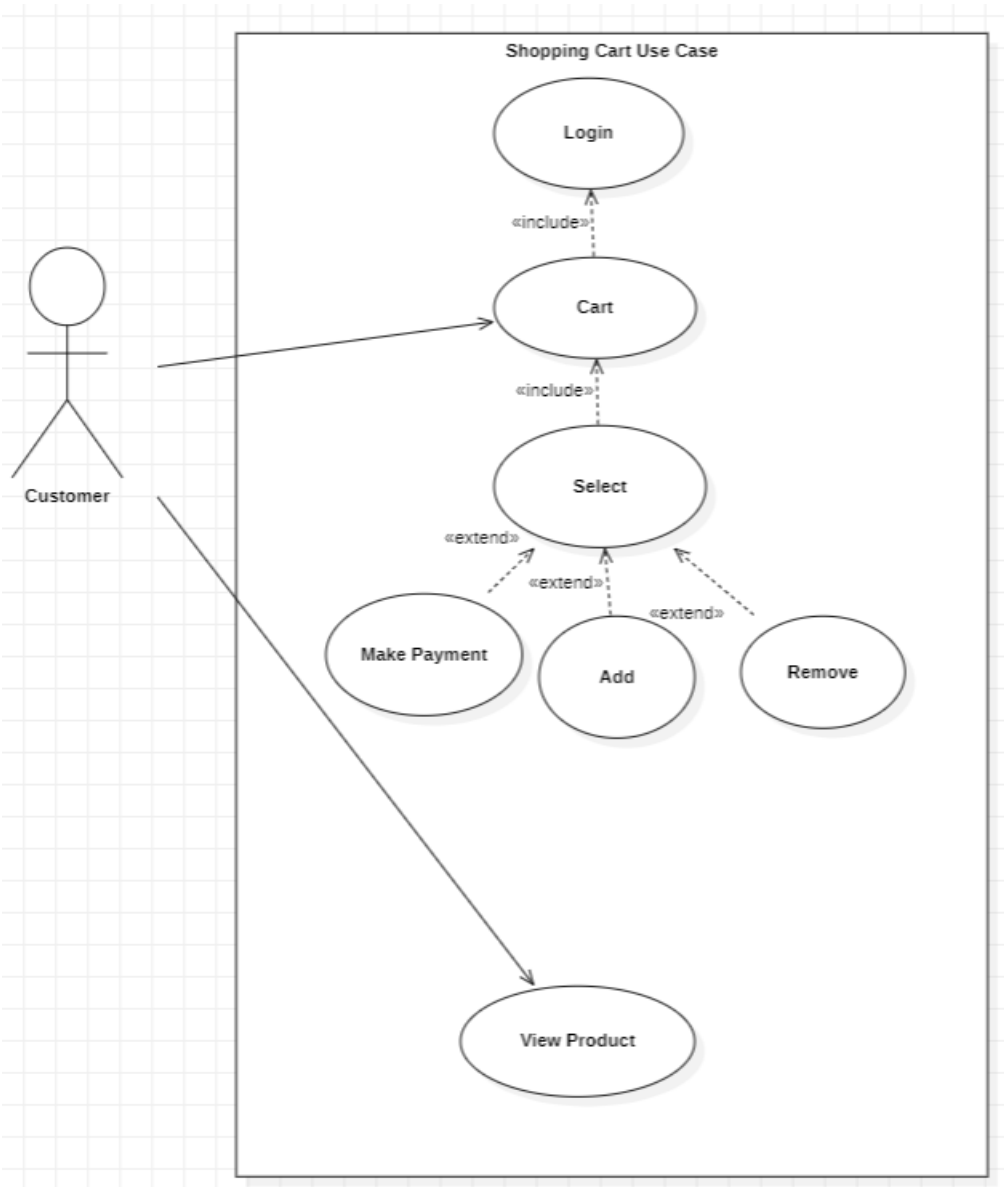


Image 2: Shopping Cart Use Case

Table 2: Shopping Cart Use Case Description

SN	Section	Context/ Explanation
1	Name	Shopping Cart
2	Priority	High
3	Description	Users use this use case when you want to choose which products, add or remove the number of products you have put in the cart.
4	Actors	Customer
7	Pre-conditions	<p>The customer must have logged in or registered an account of the website.</p> <p>The customer must have added at least one item to their cart.</p> <p>The inventory of the added items must be available in stock and not out of stock.</p> <p>The internet connection must be stable and reliable for the customer to complete the transaction.</p>
8	Post conditions	<p>The items added to the shopping cart are now marked as reserved until the transaction is completed.</p> <p>The inventory for the items in the shopping cart has been updated to reflect the purchases made by the customer.</p> <p>The customer has received an order confirmation with details of the transaction, including the order number and delivery date.</p> <p>The shopping cart has been emptied, and all items have been removed from the cart.</p>
9	Main scenario	<ol style="list-style-type: none">1. The customer browses the website and adds items they want to purchase to the shopping cart.2. The customer reviews the items in the shopping cart and makes adjustments if necessary, such as changing quantities or removing items.3. The customer proceeds to checkout and enters their shipping and billing information.

		4. The customer reviews the order details, including the total cost and estimated delivery date, and submits the order.
10	Exception scenario	Insufficient inventory: the system determines that one or more items in the customer's shopping cart are no longer available due to insufficient inventory.

6.4. Product Management Use Case Diagram:

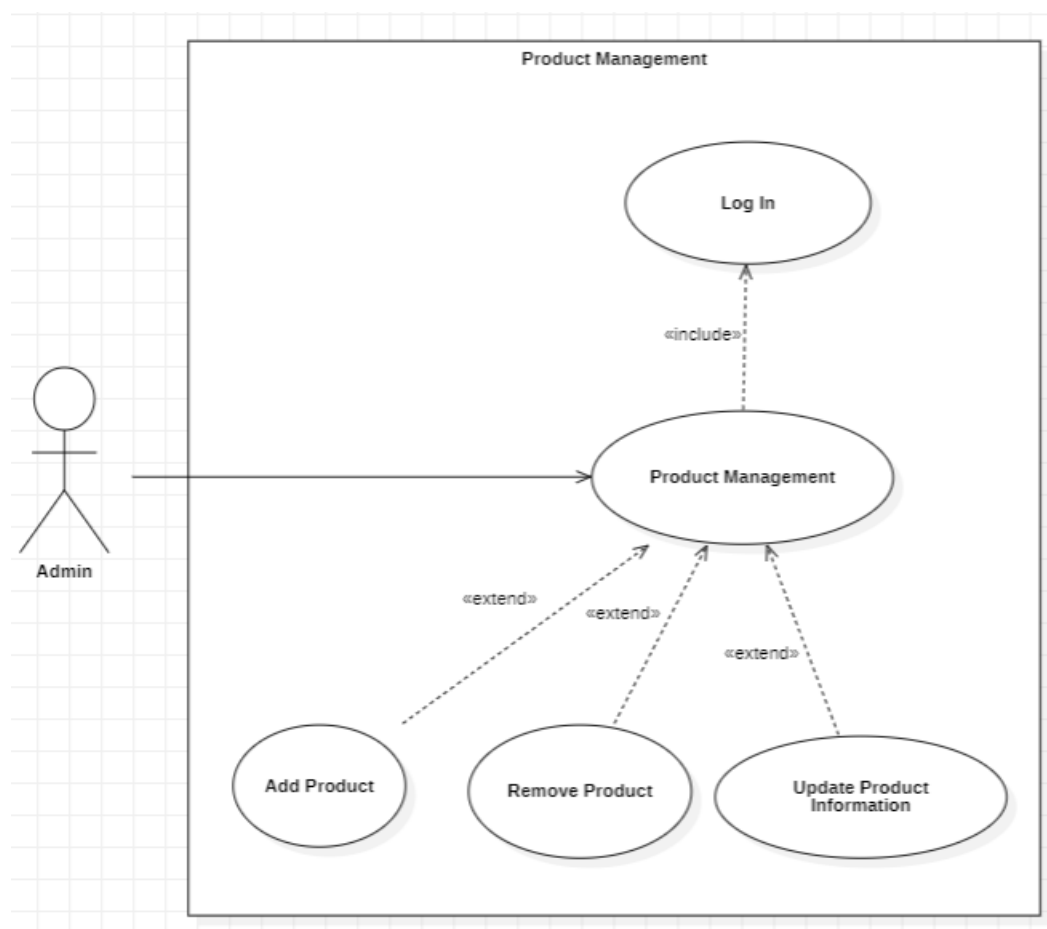


Image 3: Product Management Use Case

Table 3: Product Management Use Case Description

SN	Section	Context/ Explanation
1	Name	Product Management
2	Priority	Critical

3	Description	Users use it to add, remove, or adjust product information that will be saved in the product list for customers to select.
4	Actors	Admin
7	Pre-conditions	<p>There is information of the product that needs to be updated, added or removed will be provided to the admin responsible for website management.</p> <p>The user is logged into the system and goes to the product management interface. ...</p>
8	Post conditions	A product information is updated, deleted or put into the database. Admins will always receive notifications to add, delete or update products successfully. The newly added/updated product appears in the product management interface. The removed product disappears from the product management interface.
9	Main scenario	<ul style="list-style-type: none"> • The system displays the product management interface. • The user selects appropriate actions based on the operation needs for product information. <ul style="list-style-type: none"> ✚ Select "Add product" to enter new product information. ✚ Select "Delete product" to remove product information that exists in the product list. ✚ Select "Update product" to adjust the product information that needs to be updated. • After the operation, the user selects "Confirm." • The system checks the constraints in the database. • The system updates the operation results into the database. • The system updates successfully, returns to the product management page with product information added/updated, and removes product information after deleting the product.
10	Exception scenario	<ul style="list-style-type: none"> • Duplicate products: the system detects that a product being added already exists in the system and cannot be added again. • Invalid inputs: the system encounters invalid inputs while adding or updating a product, such as a missing or invalid product name, description or price.

		<ul style="list-style-type: none"> Product deletion failure: the system is unable to delete a product due to existing links with orders, or other dependencies.
--	--	--

6.5. Product View Use Case Diagram:

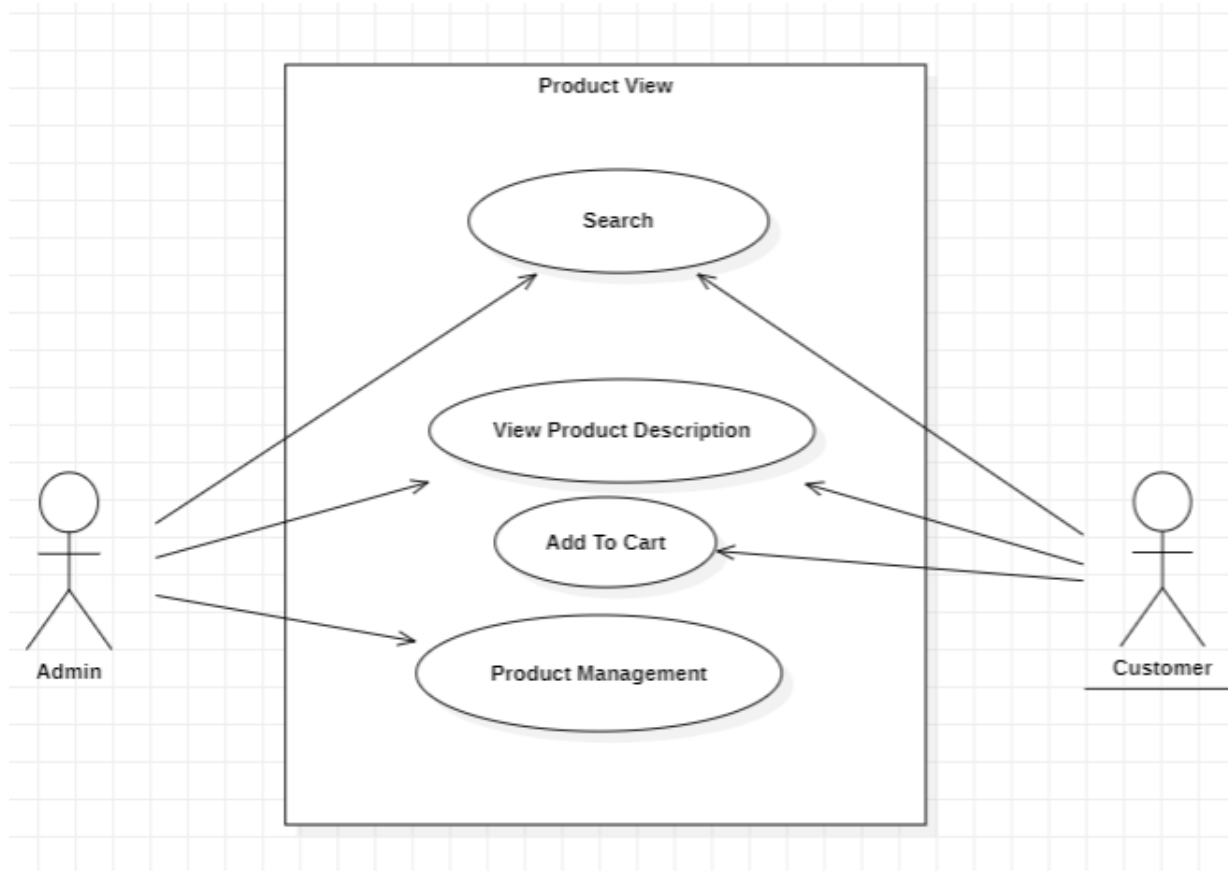


Image 4: Product View Use Case

Table 4: Product View Use Case Description

SN	Section	Context/ Explanation
1	Name	Product View
2	Priority	Critical
3	Description	Users use it to see all about the information of the product that admin had updated before.
4	Actors	Customer
7	Pre-conditions	<ul style="list-style-type: none"> The user has access to the system, either as a admin or as a customer.

		<ul style="list-style-type: none"> • The product exists in the system and is available for viewing. • The user has chosen to view the product, either by searching for it, clicking on a link, or browsing through product cards. • The user has a device with a compatible web browser or mobile application that can access the system and display the information about the product.
8	Post conditions	<ul style="list-style-type: none"> • The user has successfully viewed the product and its associated information, including name, description, price, and any available images or videos. • The user can interact with the product, such as adding it to a shopping cart or wish list, sharing it on social media, or leaving a review. • The user can navigate back to the previous page or continue browsing other products in the system.
9	Main scenario	<ul style="list-style-type: none"> • The user navigates to the product page or conducts a search for the desired product. • The system displays the product information, including name, description, price, and any available images. • The user has the option to click on the product for a closer look or to view additional details. • The system updates the view count for the product. • The user interacts with the product, such as adding it to a cart. • The user can navigate back to the previous page or continue browsing other products in the system.
10	Exception scenario	<ul style="list-style-type: none"> • The product page fails to load due to server or network issues. • The product information is incomplete or inaccurate, causing confusion for the user. • The user encounters technical difficulties such as a broken link or error message.

7. Activity Diagram:

7.1. Login Activity Diagram:

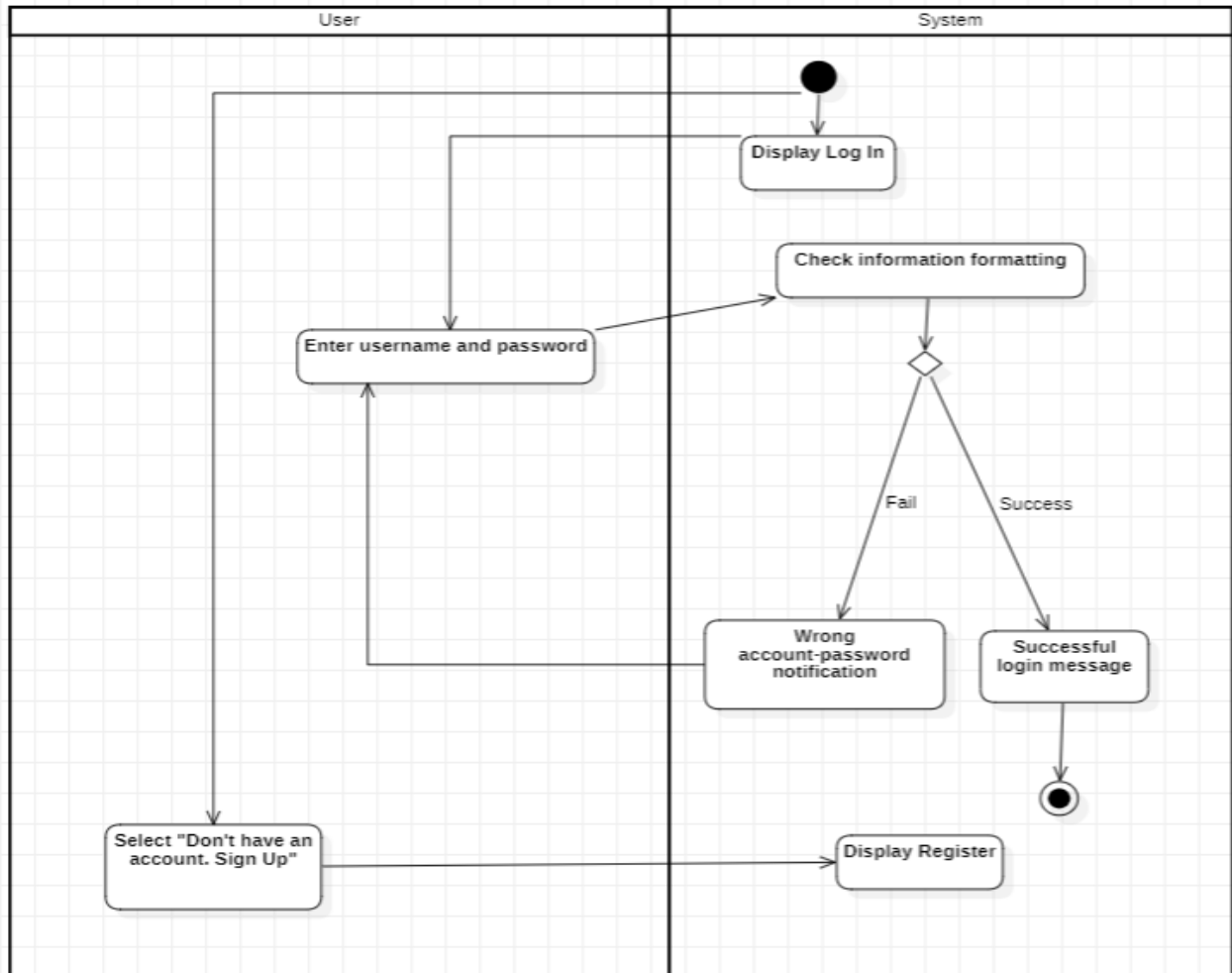


Image 5: Login Activity Diagram

7.2. Add To Cart Activity Diagram:

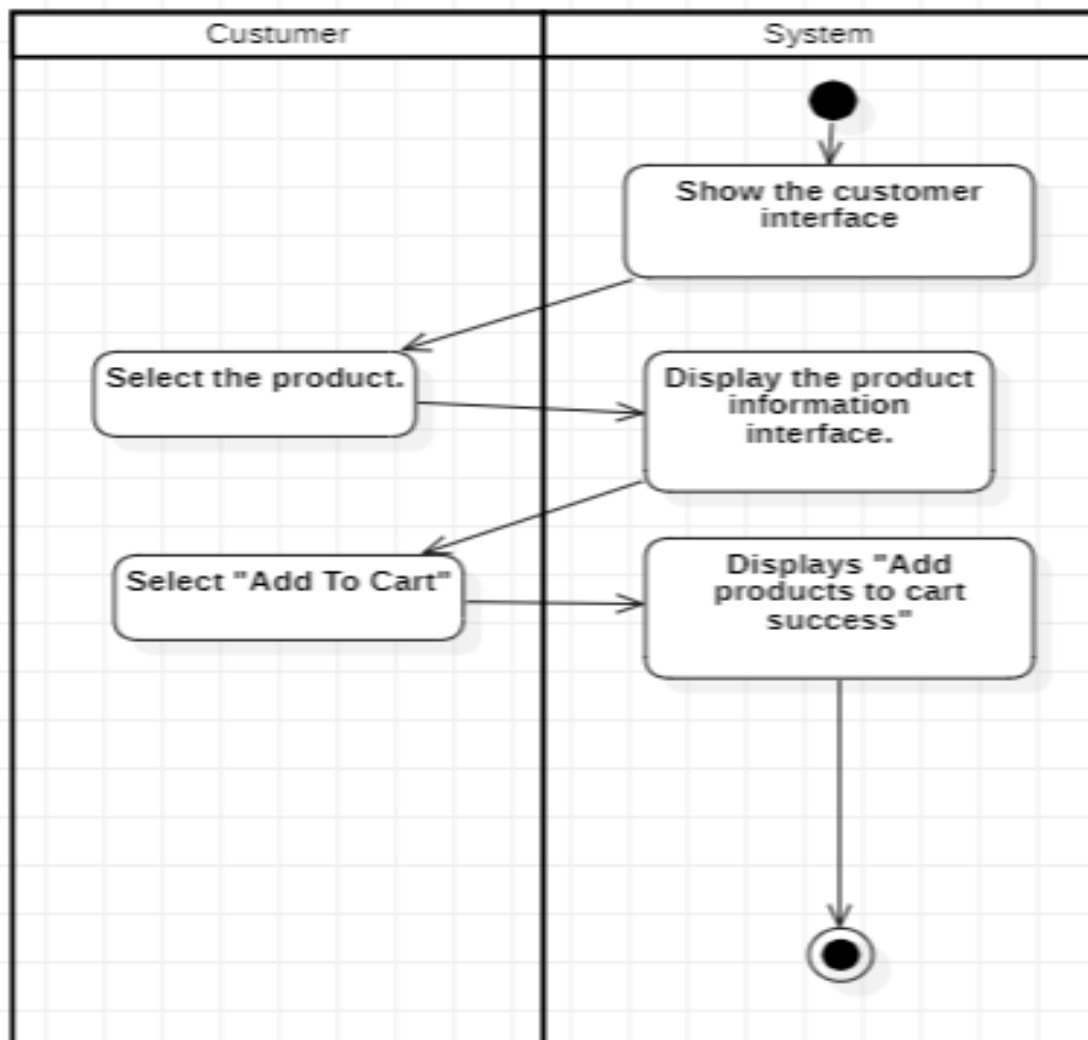


Image 6: Add To Cart Activity Diagram

7.3.Add Product Activity Diagram:

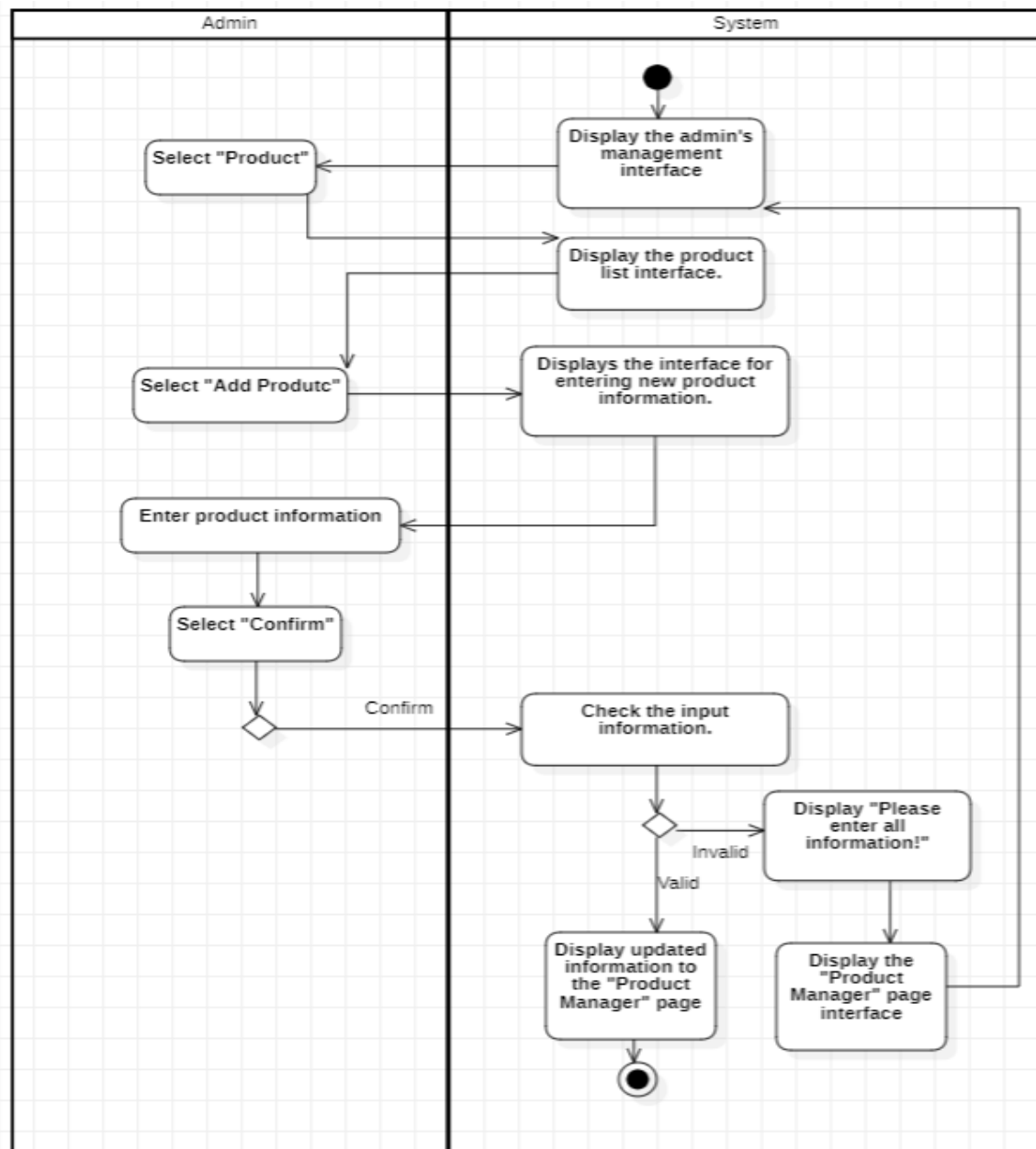


Image 7: Add Product Activity Diagram

7.4. Search Activity Diagram:

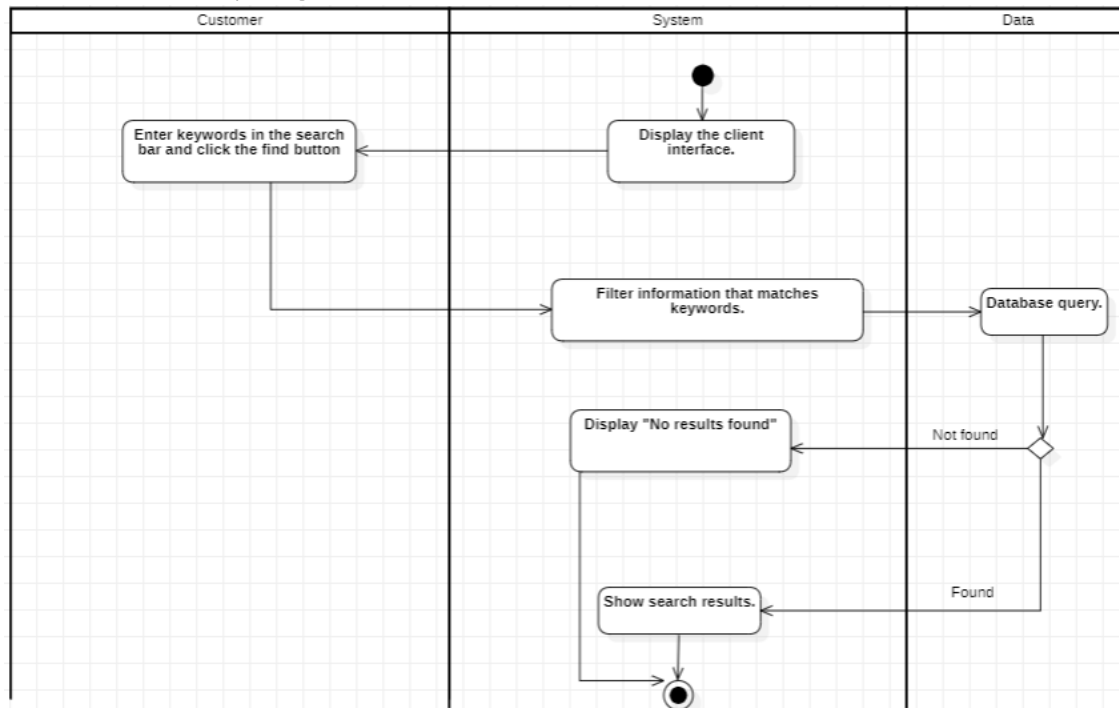


Image 8: Search Activity Diagram

8. Class Diagram:

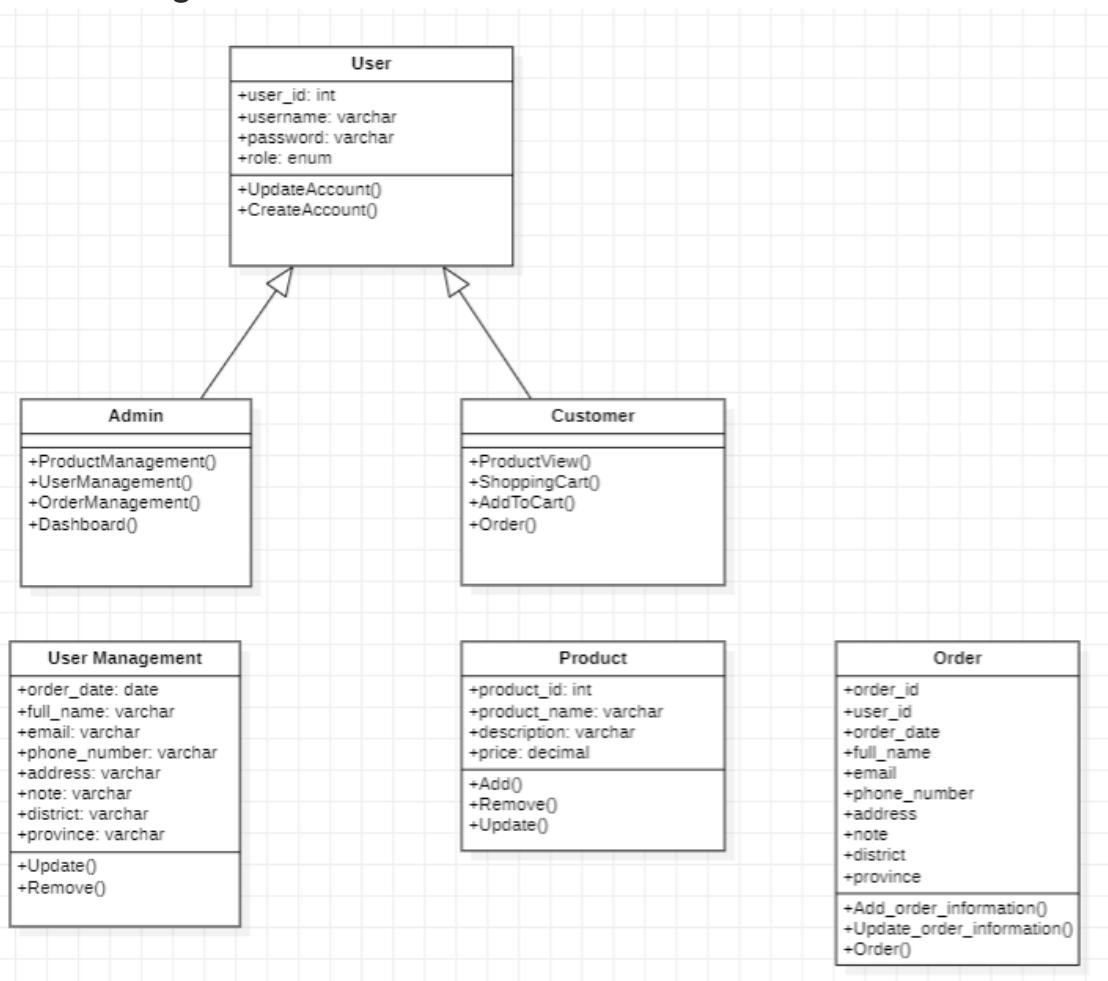


Image 9: HOTE0 Class Diagram

9. Database:

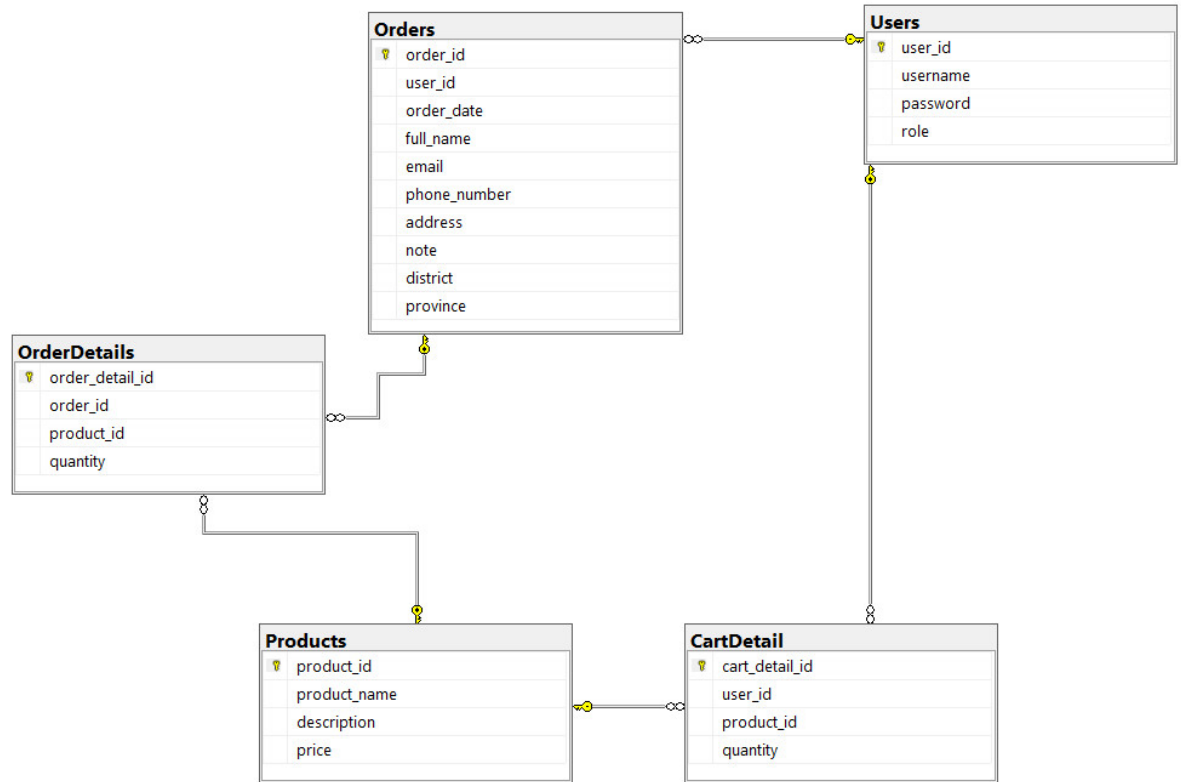


Image 10: HOTE0 Database

Table 5: HOTE0 Database

SN	Relation	Meaning	Attributes	Data type	Key
1	Users	This table stores information about users.	user_id username password role	int varchar(255) varchar(255) enum('admin', 'customer')	user_id
2	Products	This table stores	product_id	int	product_id

		information about products.	product_name description price	varchar(255) varchar(255) decimal(10, 2)	
3	Orders		order_id user_id order_date full_name email phone_number address note district province	int int date varchar(255) varchar(255) varchar(255) varchar(255) varchar(255) varchar(100) varchar(100)	order_id
4	Users		order_date full_name email phone_number address note district province	date varchar(255) varchar(255) varchar(255) varchar(255) varchar(255) varchar(100) varchar(100)	
5	CartDetail		cart_detail_id user_id	int int	cart_detail_id

10. Interfaces:

10.1. Login interface

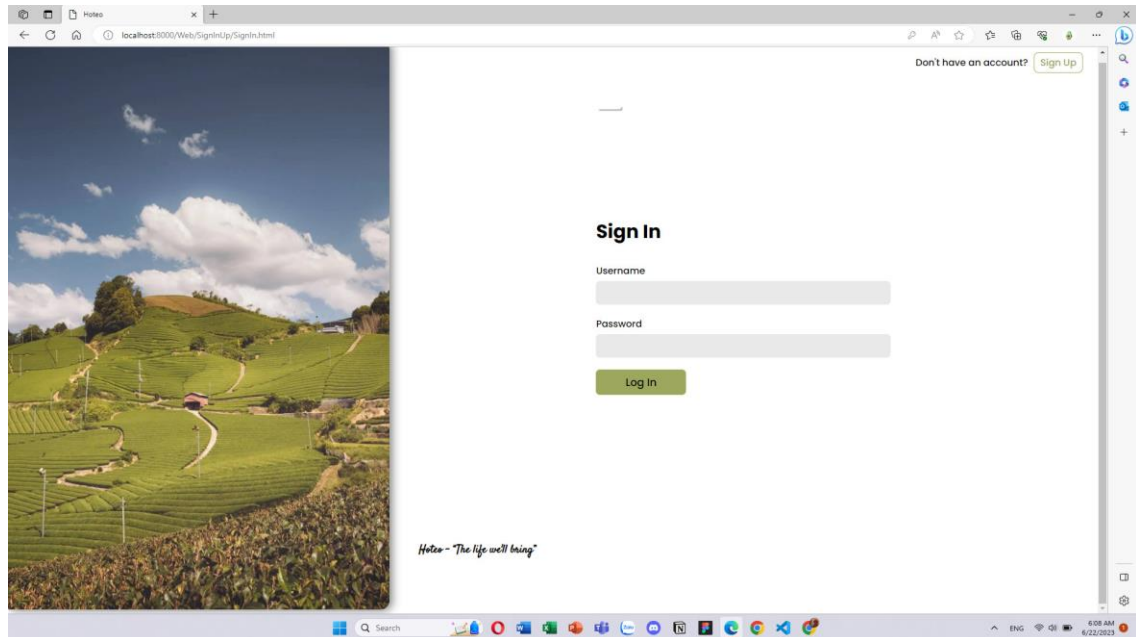


Image 10: Login interface

10.2. Homepage interface

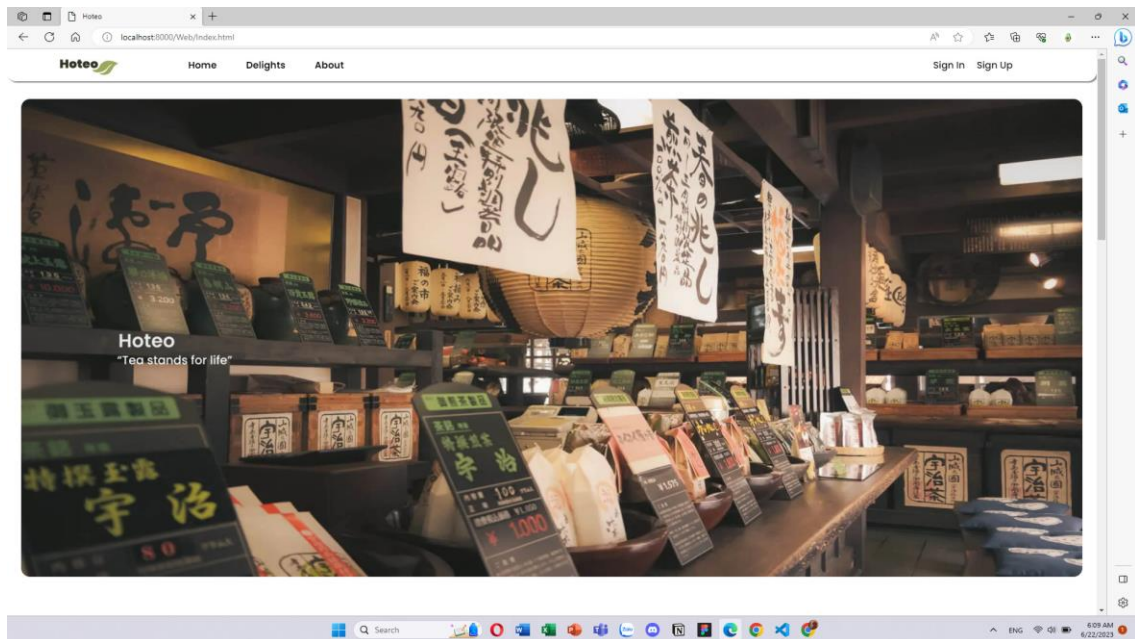


Image 11: Homepage interface

10.3. Product Page Interface

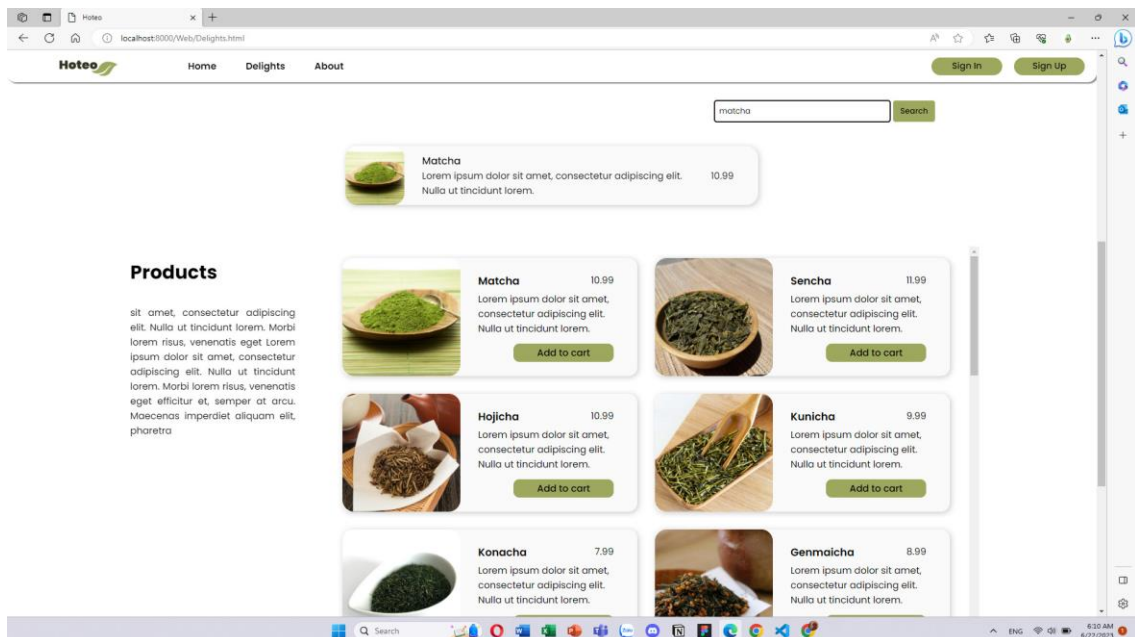


Image 12: Product Page Interface

10.4. Introduction page interface

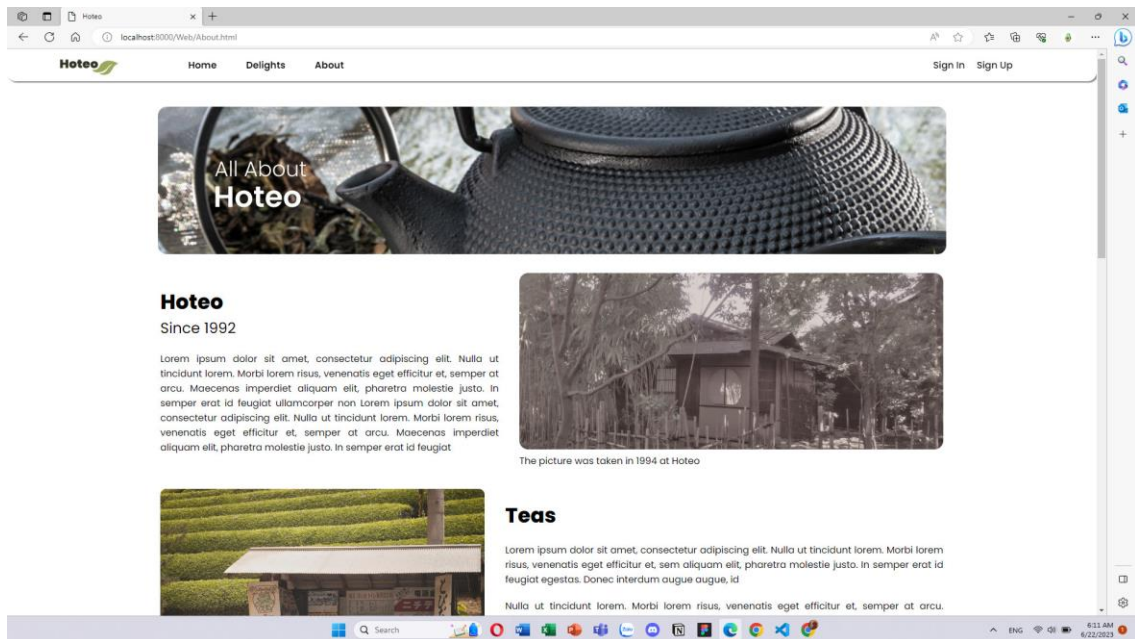


Image 13: Introduction page interface

10.5. Cart page interface

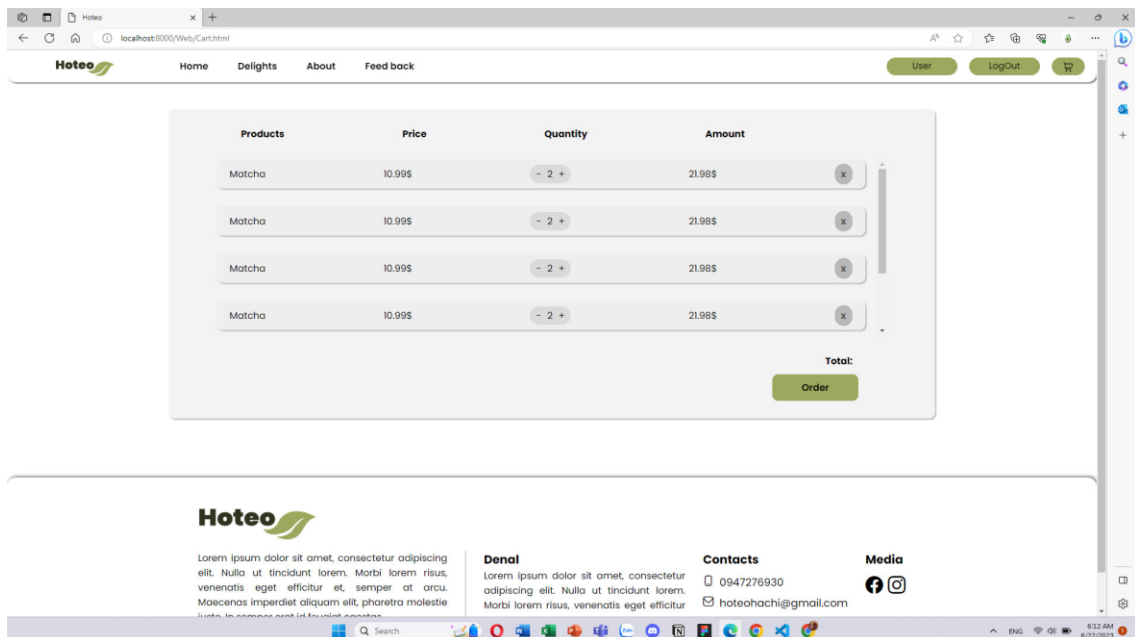


Image 14: Cart page interface

10.6. Dashboard Interface

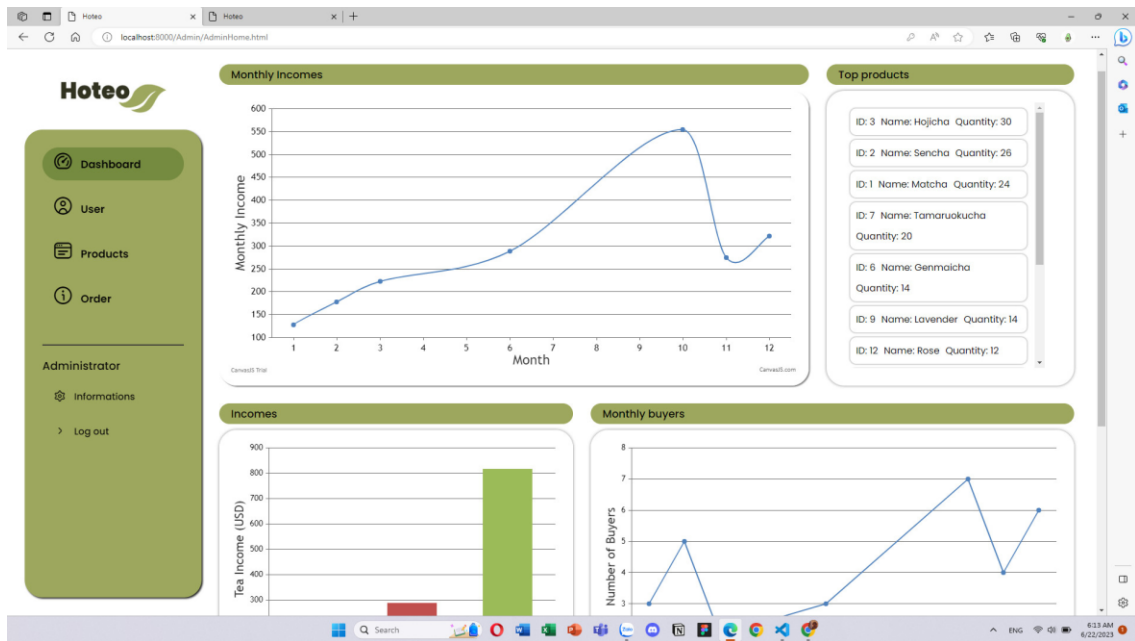


Image 15: Dashboard Interface

10.7. Searching Interface

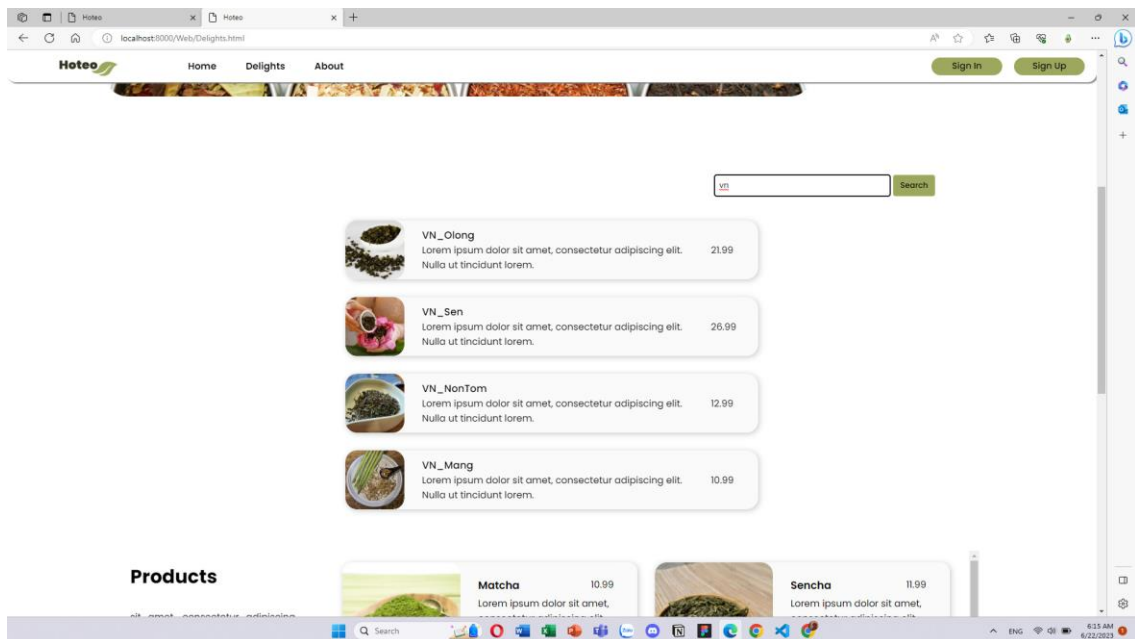


Image 16: Searching Interface

11. Testing:

11.1. Login testing:

Description	Procedure	Expected Result	Test Result
Test with blank username or password	Press "Login" button without any email or password	Display "Please fill in the form."	Pass
Test with account not exist in system	Enter username, password not exist in system Press "Login" button	Display "Invalid Credentials"	Pass
Test with the correct account	Enter username, password. Press "Login" button	Display navigate to the user interface.	Pass

11.2. Remove Product testing:

Description	Procedure	Expected Result	Test Result
Test remove the product but don't confirm	Select the product Don't press the "Remove" button	The product still exists in the list of product	Pass
Test remove the product with the right way.	Select the product Press the "Remove" button	Display navigate to the Product Management interface The product doesn't exist in the list of production in Product Management page, the Delight page and the Product Detail page.	Pass

CHAPTER 4: CONCLUSION

1. Results and advantages:

- The website is practically applicable for tea shops, used to manage information related to their products.

- The website has a user-friendly interface that is easy to use.
- After research, we have developed a tea-selling management website to replace manual management methods that are time-consuming and require effort from managers.

2. Limitations of the project:

After completing the development of the tea-selling website, there are still some shortcomings and limitations that cannot be avoided: the system analysis skills are still limited.

3. Future development:

- Continue improving the program with the hope of meeting all users' needs.
- Continue adding functions that the website does not currently have so that it can be widely applied in all stores.