HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY

INSTITUTE OF INTERNATIONAL EDUCATION

FINAL ASSIGNMENT REPORT

Major: **INFORMATION TECHNOLOGY**

Course code: CMP186

Course name: **TOOL & ENVIRONMENT FOR SOFTWARE DEVELOPMENT**

Lecturer's name:

Student's name:

1. Nguyễn Cao Long	ID: 2011063433	Class: 20DTHQB1
2. Nguyễn Quốc Huy	ID: 2080601122	Class: 20DTHQB1

Ho Chi Minh City

2024

Table of contents

Table of Figure, Image, Table

PROLOGUE

First and foremost, I would like to express my utmost gratitude to Ms.Hanh. Throughout the process of studying and exploring the subject of New Programing Language, I have received tremendous care, support, passionate guidance, and dedication from you. You have helped me accumulate a wealth of knowledge in this subject, enabling me to complete the essay on the topic of “Notes”.

Undoubtedly, there may be some shortcomings in my work. Therefore, I sincerely hope to receive your feedback and suggestions to further enhance the quality of my project.

Thank you very much!

Chapter 1. INTRODUCE

1.1 Reporting purpose

This report is created to evaluate and analyze the effectiveness of using a new programming language in the course .The main purpose of this report is:

Assessment of Learning Effectiveness:

- Determine the effectiveness of using the new programming language in the learning process of students in the course.
- Analyze the approach and understanding of students regarding the new programming language compared to other languages they have learned before.

Evaluation of Teaching Effectiveness:

- Analyze the capability of the new programming language in conveying knowledge and programming skills to students.
- Assess feedback from students regarding teaching methods and support from instructors in learning the new programming language.

Analysis of Applicability:

- Evaluate the applicability of students after completing the course in applying knowledge and programming skills using the new language in real-world projects.
- Analyze the integration capability of the new programming language into other fields and technological applications.

Proposed Improvements:

- Propose new teaching methods and strategies to optimize the use of the new programming language in the course.
- Identify areas for improvement and development in integrating the new language into the curriculum.

1.2 Project Introduction: Fruit Selling Website

This is a project dedicated to providing fresh and high-quality fruits directly to consumers' doorsteps. Our goal is to create a convenient and reliable online platform where everyone can easily purchase delicious fruits from home, while also supporting access to clean and health-safe fruit products.

1.3 Target Users of the Fruit Selling Website

1.3.1 Individual Consumers

Health-Conscious Individuals: Those who prioritize healthy eating and seek fresh and nutritious fruits for their daily diet.

Busy Professionals: Individuals with hectic schedules who prefer the convenience of online shopping for groceries, including fruits.

Fruit Enthusiasts: People who enjoy exploring and trying out different types of fruits, looking for unique or exotic options.

1.3.2 Families

Households:Families looking to provide their loved ones with healthy snacks and ingredients for meals.

Parents: Parents who want to encourage their children to consume more fruits as part of a balanced diet

1.3.3 Office and Corporate Clients

Corporate Offices: Companies and organizations interested in providing fresh fruit options for their employees in the workplace.

Event Planners: Professionals organizing corporate events or gatherings where fresh fruit arrangements are desired.

1.3.4 Health and Wellness Organizations

Fitness Centers and Gyms: Businesses focused on promoting health and fitness may offer fresh fruits as part of their amenities or nutrition programs.

Nutritionists and Dietitians: Professionals who recommend incorporating fresh fruits into their clients' diets for overall health and well-being.

1.3.5 Hospitality Industry

Hotels and Resorts: Hospitality establishments may offer fresh fruit baskets or amenities to enhance the guest experience.

Restaurants and Cafes: Food establishments seeking reliable suppliers of fresh fruits for their menu items or beverages.

1.3.6 Educational Institutions

Schools and Day cares: Institutions interested in providing healthy snack options for students or incorporating fruit-based learning activities.

Colleges and Universities: Higher education institutions may offer fresh fruits as part of their campus dining options or wellness programs.

1.3.7 Special Occasion Planners:

Party Planners: Individuals organizing events such as weddings, birthdays, or celebrations may include fresh fruit arrangements as part of their decorations or catering options.

1.3.8 Gift Givers:

Individual Gift Shoppers: Those looking for thoughtful and healthy gift options for special occasions or to express appreciation to loved ones.

Corporate Gift Buyers: Companies and organizations interested in sending fruit baskets or gift sets as corporate gifts to clients or employees

Chapter 2. REQUIREMENT ANALYSIS

2.1 Functional Requirements of the Fruit Selling Website

1. User Registration and Authentication:
   - Allow users to register for an account and log in securely.
   - Provide options for social media login integration for convenience.

2. Browsing and Product Selection:
   - Display a user-friendly interface for browsing different categories of fruits.
   - Enable users to search for specific fruits by name, category, or other criteria.
   - Include high-quality images and detailed descriptions for each fruit product.

3. Shopping Cart and Checkout:
   - Allow users to add fruits to their shopping cart for later purchase.
   - Provide a seamless checkout process with multiple payment options (credit/debit card, PayPal, etc.).
   - Include features for order review, modification, and order confirmation.

4. User Profile Management:
   - Allow users to manage their account information, including personal details, addresses, and payment methods.
   - Provide options for order history tracking and status updates.

5. Product Reviews and Ratings:
   - Allow users to leave reviews and ratings for purchased fruits.
   - Display average ratings and reviews to help users make informed purchasing decisions.

6. Special Offers and Promotions:
   - Display promotional banners or sections highlighting special offers, discounts, or seasonal deals.

- Implement a newsletter subscription feature to notify users about new arrivals and promotions.

7. Responsive Design:

- Ensure the website is responsive and compatible with various devices (desktops, laptops, tablets, smartphones).
- Optimize the user experience for both desktop and mobile users.

8. Search Engine Optimization (SEO):

- Implement SEO best practices to improve visibility and ranking on search engines.
- Include meta tags, keywords, and structured data to enhance search engine visibility.

9. Contact and Support:

- Provide contact information (email, phone number) for customer support inquiries.
- Include a live chat feature or FAQ section to address common questions and concerns.

10. Order Tracking and Notifications:

- Enable users to track the status of their orders in real-time.
- Send email or SMS notifications to users regarding order updates, delivery status, and promotions.

11. Integration with Delivery Services:

- Integrate with third-party delivery services for efficient and reliable order fulfillment.
- Provide options for scheduled delivery dates and times.

12. Accessibility and Security:

- Ensure the website complies with accessibility standards to accommodate users with disabilities.
- Implement security measures such as SSL encryption to protect user data and transactions.

2.2 Non-Functional Requirements of the Fruit Selling Website

1. Ease of Access:

- Ensure the website is easily accessible from all devices and popular web browsers such as Chrome, Firefox, and Safari.

2. Consistent Interface:

- Design an intuitive and understandable interface to create a consistent user experience across all pages and functions of the website.

3. Fast Page Loading Times:

- Optimize website performance to reduce page loading times and enhance user experience.

4. Detailed and Accurate Data:

- Provide detailed and accurate information about each type of fruit, including origin, characteristics, and nutritional information.

5. User Data Security:

- Protect users' personal information and transactions with security measures such as SSL encryption and two-factor authentication.

6. Multi-Language Support:

- Offer website translation feature into multiple languages to serve diverse users.

7. Compliance with Web Standards:

- Adhere to international web standards and regulations to ensure compatibility and support across different platforms and browsers.

8. Diverse Payment Support:

- Provide multiple secure and convenient payment methods such as credit cards, bank transfers, and e-wallets.

9. Advanced Search Functionality:

- Support advanced search functionality to allow users to filter and search for fruits based on criteria such as price, fruit type, or origin.

10. Flexible Return Policies:

- Offer flexible and fair return policies to enhance customer trust and satisfaction with the service.

11. Social Media Integration and Sharing:

- Integrate social media sharing buttons and links to allow users to easily share their favorite products with friends and family.

12. 24/7 Customer Support:

- Provide continuous customer support channels such as live chat, email, and phone to address any inquiries and requests from customers.

Chapter 3. SYSTEM ARCHITECTURE

3.1 System architecture for a fruit-selling website

1. Front-end Layer:

- Client Interface: The user interface displayed on web browsers or mobile apps.
- UI/UX Components: Ensure user-friendly interface and ease of use.

2. Back-end Layer:

- Web Server: Handles requests from users and sends back corresponding web pages.
- Application Server: Processes business logic and interacts with the database.
- Business Logic Layer: Manages core functions such as product management, orders, payments, user authentication, etc.
- RESTful API: Provides endpoints for communication between frontend and backend.

3. Database Layer:

- Database Server: Stores data about products, users, orders, etc.
- Relational Database Management System (RDBMS) or NoSQL Database: Depending on specific application requirements.

4. Authentication and Authorization:

- Authentication Service: Manages user authentication process.
- Authorization Service: Manages user access rights to resources.

5. Third-party Integrations:

- Payment Gateways: Integrates payment gateways like PayPal, Stripe, etc.
- Shipping Services: Integrates shipping services for calculating shipping fees and tracking orders.

6. Infrastructure Layer:

- Hosting: Can use dedicated servers, virtual servers, or cloud services like AWS, Google Cloud, Azure.
- Load Balancer: Distributes load to increase performance and ensure system availability.
- Security Services: Protects the system from threats like DDoS attacks, SQL Injection, etc.
- Backup and Recovery: Regularly backs up data and ensures quick recovery.

7. Monitoring and Logging:

- Monitoring Tools: Monitors system performance and detects issues.
- Logging Services: Records system activities for analysis and tracking.

3.2 Data flow diagram

1. User browsing products:

- User accesses the fruit-selling website.
- The website displays a list of available fruit products.
- User browses through the list of products and views detailed information about each product.

2. Adding to Cart:

- User selects the fruit products they want to purchase.
- The system adds the selected products to the user's shopping cart and updates the quantity and total price.

3. Checkout:

- User proceeds to checkout after completing product selection.
- The system requests payment information from the user, such as shipping address, payment method, etc.

4. Order Confirmation:

- User confirms the order after reviewing and ensuring the accuracy of payment information.
- The system sends an order confirmation to the user and begins processing the order.

5. Order Management (Administrator):

- Administrator accesses the order management system.
- The system displays a list of new and pending orders.
- Administrator processes orders by confirming and updating their status.

6. Shipping:

- The system sends order confirmation details to the shipping team.
- The shipping team delivers the products to the specified shipping address.

7. Payment Processing:

- The system processes payment based on the payment method chosen by the user (e.g., credit card, cash on delivery, etc.).
- The system sends a payment request to the payment gateway or linked payment service for confirmation and processing.

8. Product Management (Administrator):

- Administrator accesses the product management system.
- The system allows the administrator to add, edit, delete products, update pricing, and manage inventory.

3.3 Database

3.3.1 Use case Diagram

General Use case

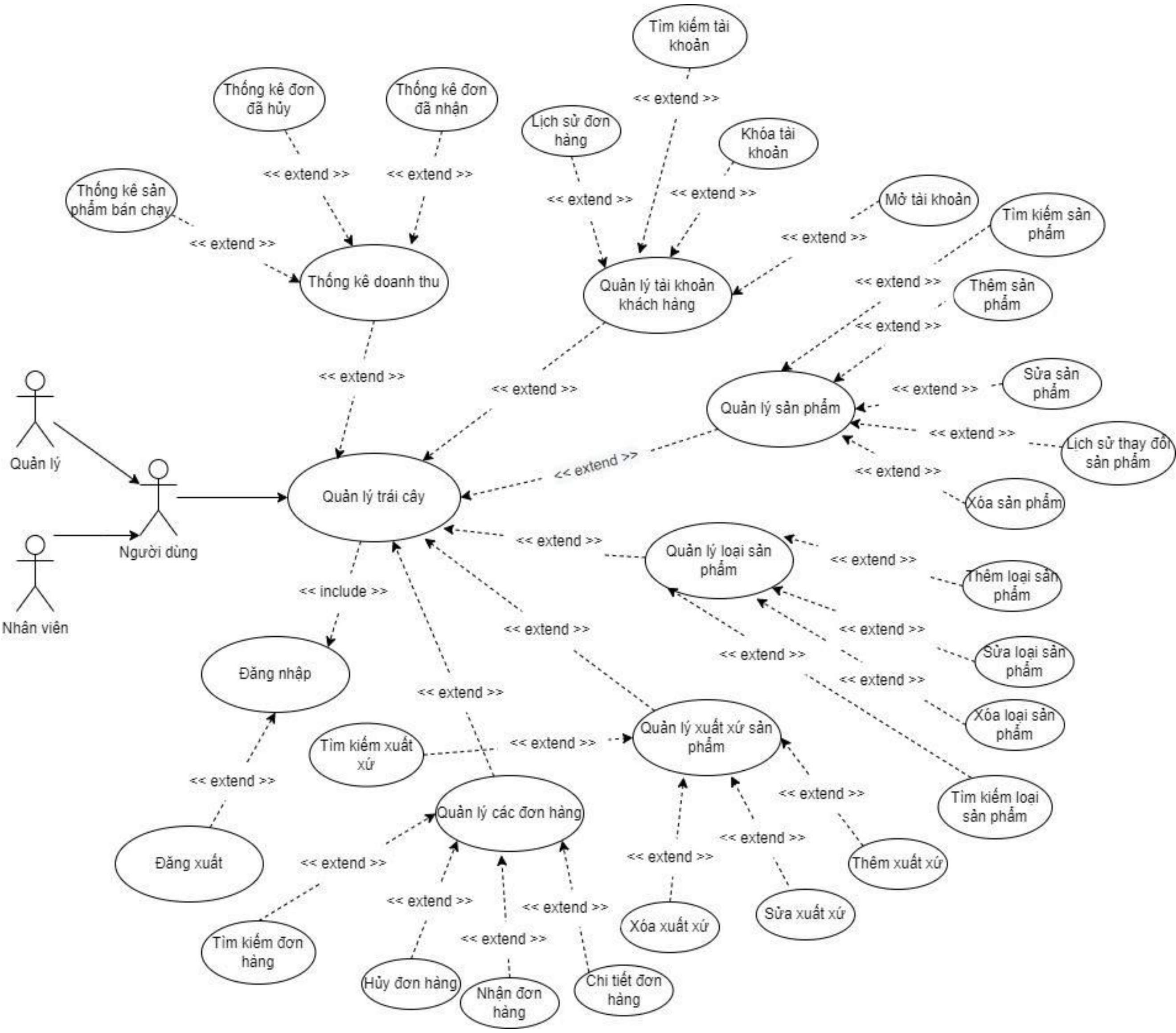


Figure 1: General Use case

Login Use case

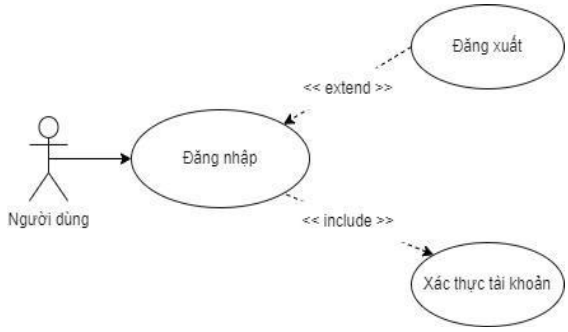


Figure 2: Login use case

Use case	Content
Name of Use case	Log in
Description	Allows users to log into the system to execute show its functions
Actors	User
Trigger event	When the user chooses the login function
Preconditions	Users must have an account on the system
Postconditions	User logged in successfully
Main event stream	The system displays the login screen. User enters login name and password. The system displays and checks the login information. If successful, the system displays the successful login screen End the Use case

Secondary event stream	A. Wrong password: When the user enters the wrong login name and password 1. The system displays the login screen again for the user to re-enter information with a notification that the login name and password are incorrect. 2. Return to step 2 in the main event flow B. Forgot password: When the user selects the forgot password function on the login screen 1. The system displays a screen for the user to enter email 2. User enters email and selects password retrieval function 3. The system checks the email is valid and sends a link to reset the password to the user via email 4. The system displays a success notification screen 5. Use case ends

Table 1: Login Event Stream

Account management function Use case

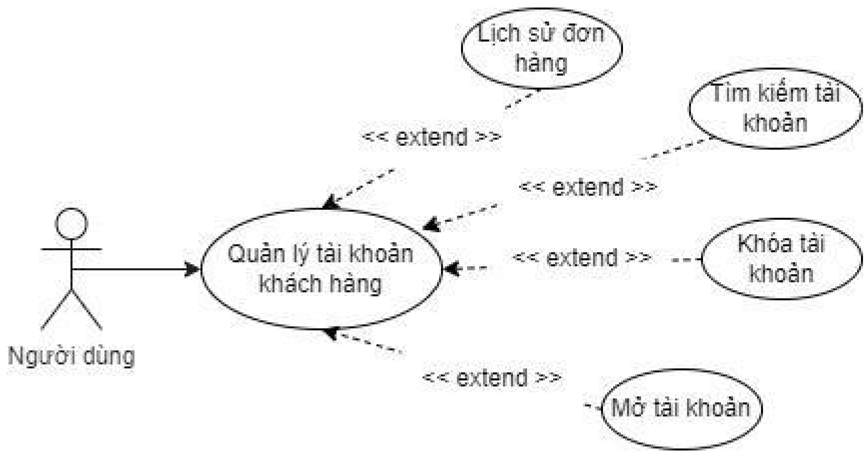


Figure 3: Account management function Use case

Use case	Content
Name of Use case	Account management
Description	Allows admin to manage customer accounts
Actors	Admin
Trigger event	After logging in to your admin account and click manage account clause
Preconditions	Have an admin account
Postconditions	Account management
Main event stream	Case 1: 1. Click the account management button. 2. View orders 3. Return to the account management screen Case 2: 1. Lock account or open account

Table 2: Account management event flow

Product management function Use case

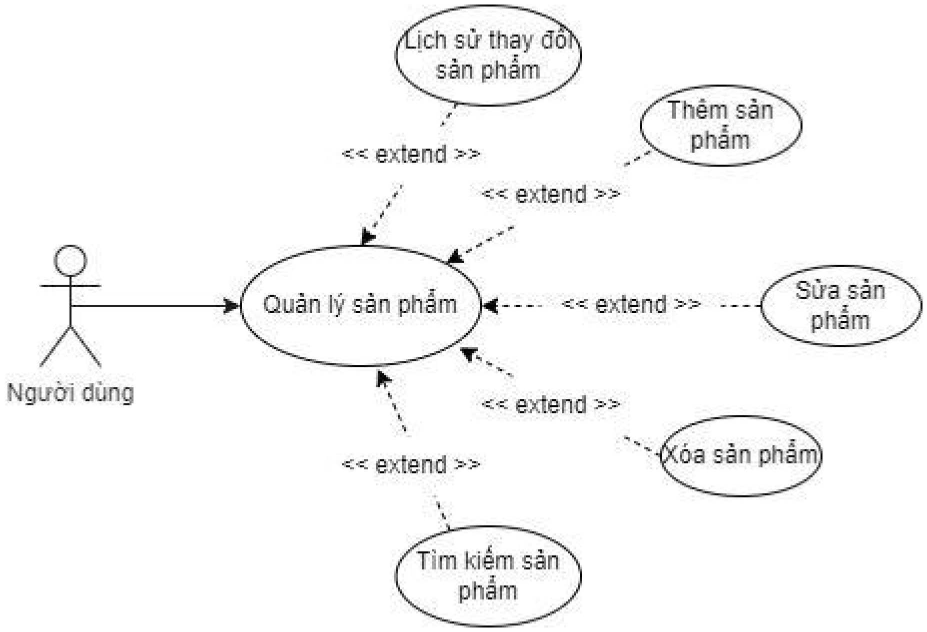


Figure 4: Product management use cases

Use case	Content
Name of Use case	Product Management
Description	Allow admin to manage products
Actors	Admin
Trigger event	After logging in to your admin account and click on property management Products
Preconditions	Have an admin account
Postconditions	Product Management
Main event stream	Case 1: 1. Product management display system 2. Select more products 3. Enter product information 4. Save more products Case 2: 1. Product management display system 2. Select edit product 3. Edit product information

	4. Update products Case 3: 1. Product management display system 2. Select delete product 3. Update product management
Secondary event stream	Case1: 1. Product management display system 2. Select more products 3. Return to the product management form Case 2: 1. Product management display system 2. Select edit product 3. Return to the product management form

Table 3: Product management event flow

Product management Use case

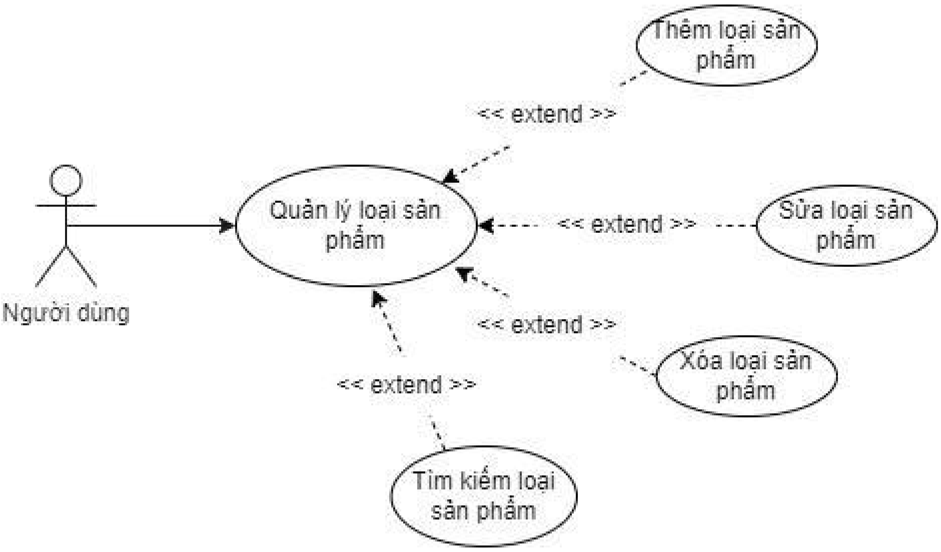


Figure 5: Product management Use case

Use case	Content
Name of Use case	Manage product types
Description	Allows admin to manage product types
Actors	Admin
Trigger event	After logging in to the admin account and click manage categories product
Preconditions	Have an admin account
Postconditions	Manage product types
Main event stream	Case 1: 1. Product type management display system 2. Select more product types 3. Enter product type information - Save more product types Case 2: 1. Product type management display system 2. Select and edit product type 3. Edit product type information 4. Update product type Case 3: 1. The system displays product type management 2. Select delete product type 3. Update product category management
Secondary event stream	Case 1: 1. Product type management display system 2. Select more product types 3. Return to the product type management form Case 2: 1. Product type management display system 2. Select and edit product type 3. Return to the product type management form

Table 4: Product type management event flow

Use case of product origin management function

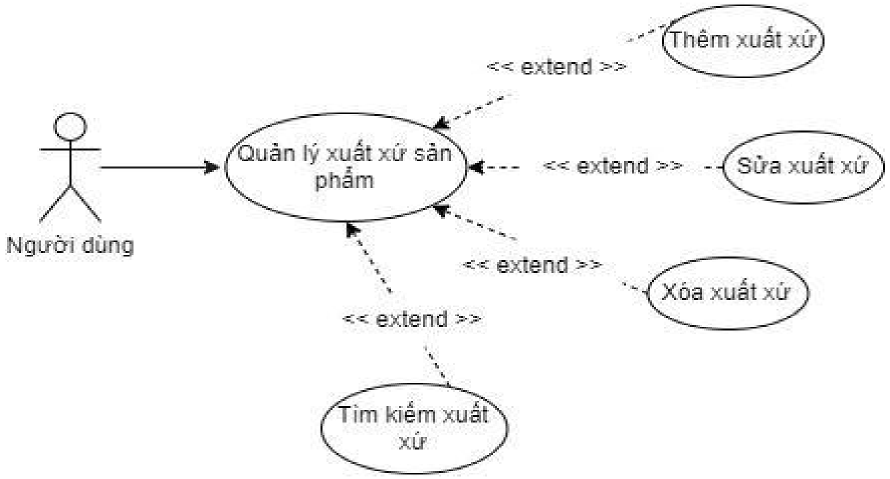


Figure 6: Use case for product origin management

Use case	Content
Name of Use case	Origin management
Description	Allow admin to manage origin
Actors	Admin
Trigger event	After logging in to the admin account and clicking manage origin
Preconditions	Have an admin account
Postconditions	Origin management
Main event stream	Case1: Origin management display system Select additional origin Enter origin information Save additional origins Case 2: Origin management display system Select edit origin Correct origin information Update origin Case 3: Origin management display system Select delete origin Update origin management

Secondary event stream	Case 1: 1. Origin management display system 2. Select additional origin 3. Return to the origin management form Case 2: 1. Origin management display system 2. Select edit origin 3. Return to the origin management form

Table 5: Provenance management event flow

Use case of order management function

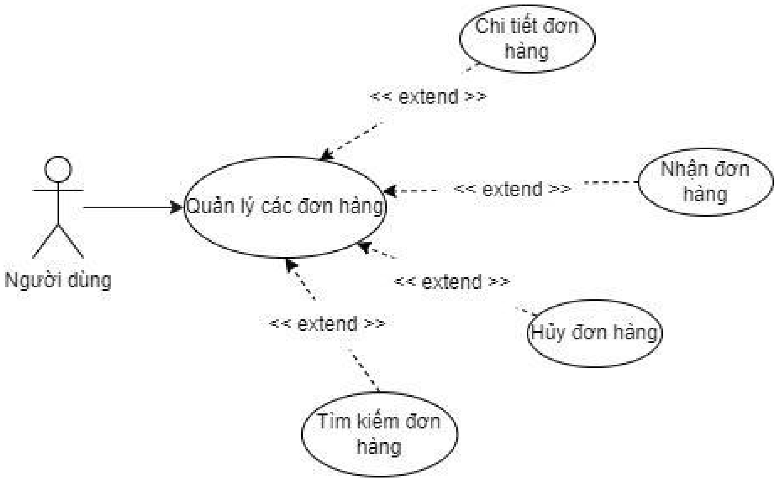


Figure 7: Use case for order management

Use case	Content
Name of Use case	Manage orders
Description	Allow admin to manage orders
Actors	Admin
Trigger event	After logging in to the admin account and clicking manage the order
Preconditions	Have an admin account
Postconditions	Manage orders

Main event stream	Case 1: Order management display system Select to view order details Select return Case 2: Order management display system Order manipulation Goods have been received or canceled Update operations

Table 6: Order management event flow

Use case of revenue statistics function

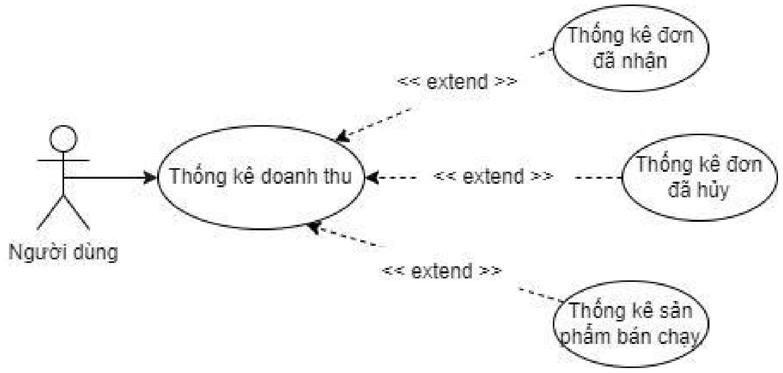


Figure 8: Use case for revenue statistics

Use case	Nội dung
Name of Use case	Revenue statistics
Description	Allows admin to compile revenue statistics
Actors	Admin
Trigger event	After logging in to your admin account and clicking statistics
Preconditions	Have an admin account
Postconditions	Revenue statistics
Main event stream	- Select start date and end date - Select statistics on received orders and canceled orders or best-selling products - Display necessary information.

Table 7: Revenue statistics event flow

3.4 User Interface

Main Page:

Header:

- Website logo.
- Important links such as Home, Products, About, Contact.
- Search box for finding specific types of fruits.

Carousel:

- Displays prominent images of popular or discounted fruits.
- Provides quick links to specific product categories.
- Product Categories:
- List of fruit categories such as fruits, seasonal fruits, imported fruits, organic fruits, etc.
- Images or icons representing each category.

Featured Products:

- Showcases best-selling or special products.
- Each product includes an image, name, price, and "Add to Cart" button.

Offers or Advertisements:

- Displays special offers or advertising announcements if available.

Footer:

- Contact information.
- Links to other information pages such as Privacy Policy, Terms and Conditions.

Product Page:

Product Image:

- Displays a large image of the product with the ability to zoom in.
- Detailed description of the product including name, description, price, and nutritional information if available.

Quantity Selection and Add to Cart:

- Quantity selection box for the product.
- "Add to Cart" button to purchase the product.

Reviews and Ratings:

- Section allows users to rate and write reviews about the product.

Related Products:

- Displays similar or related products to encourage users to explore further.

Shopping Cart:

Product List:

- Displays a list of products added to the shopping cart.
- Allows users to edit the quantity or remove products from the cart.

Total Amount:

- Shows the total amount of the products in the shopping cart.

Checkout Button:

- "Checkout" button redirects users to the checkout page.

Checkout Page:

Delivery Information:

- Form for users to input delivery information.

Payment Method:

- Selection of payment methods such as credit card, bank transfer, PayPal, etc.

Order Summary:

- Recap of the product information and total payment.

Confirm Order Button:

- "Confirm Order" button to complete the purchasing process.

Chapter 4. SYSTEM DEPLOYMENT

4.1 Choose the NodeJS framework

Image 1: NodeJS logo

NodeJS is a powerful programming platform, built on the Chrome V8 JavaScript engine, used to develop efficient and flexible server applications. In our system, Node.js plays an extremely important role and it is the core of the infrastructure.

With asynchronous processing capabilities, Node.js allows us to build multi-threaded applications without having to worry about performance. This is especially important in handling concurrent requests from multiple users. Node.js provides a flexible and convenient environment for handling I/O tasks without blocking, while minimizing waiting time for users.

One of the strengths of Node.js is the large and rich community of npm modules. This helps us quickly integrate extended features and functionality into our applications with ease. By using off-the-shelf modules, we can leverage existing solutions without having to write code from scratch, saving development time and effort.

Additionally, Node.js also allows us to build real-time applications. This is important for some applications with low latency and data synchronization requirements. Thanks to this capability, we can provide the best user experience and ensure the application always operates smoothly and stably.

4.2 Choose the Express framework

Image 2: ExpressJS logo

Express is a popular web framework built on Node.js, widely used in developing web applications and APIs. In our system, Express plays an important role in building APIs and handling HTTP requests easily and robustly.

One of the main advantages of Express is its simplicity and flexibility. Express provides a minimalist approach to handling requests and responses, helping us focus on developing the main features of our application without having to worry about complex technical details.

Express also allows us to easily organize the application's source code in a modular way and reuse the source code effectively. This helps us maintain the source code easily and flexibly scale the application as needed.

One of the powerful features of Express is middleware. Middleware allows us to extend the functionality of our application by adding intermediate layers to handle requests and responses before they reach the final routes. This helps us perform tasks such as authentication, access checking, and logging with ease and flexibility.

4.3 Choose the MongoDB framework

Image 3: MongoDB framework logo

MongoDB is a popular non-relational database management system (NoSQL) used in storing and managing data for web and mobile applications. In our system, MongoDB plays an important role in storing data and providing a flexible and scalable database to cater to the needs of our application.

One of the biggest advantages of MongoDB is its flexible data structure. MongoDB uses JSON-style documents to store data, allowing us to store structured and unstructured data in a natural and flexible way. This makes changing data structures easy and does not cause the burden of traditional relational database management systems.

MongoDB also offers linear scalability. This means we can scale our database by adding nodes to the MongoDB cluster easily, helping us handle large data traffic and boost application performance without any hassle. No scalability issues.

Another powerful feature of MongoDB is its flexible querying capabilities. MongoDB provides a powerful and flexible query language that allows us to query data according to a variety of criteria easily and efficiently.

4.4 Choose the React framework

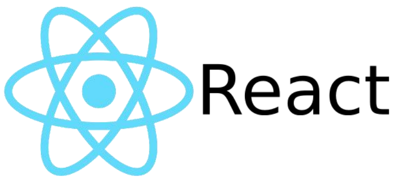


Image 4: React logo

React is a popular JavaScript library used for building user interfaces in web applications. In our system, React plays a key role in building user interfaces that are flexible, efficient, and easy to maintain.

One of React's strengths is its ability to reuse components. React allows us to divide the user interface into small, independent components, each performing a specific task. This helps us easily reuse components, increase efficiency, and reduce code repetition.

React also allows us to write code in a declarative way, rather than in a concrete way to update the user interface (imperative). This makes code development and maintenance easier, as we only need to focus on describing "what" to display instead of "how" to display it.

React also integrates well with other technologies and provides a powerful approach to managing application state. With libraries like Redux or Context API, we can manage application state easily and efficiently.

In addition, React also has a large and rich community, with many useful documents and online learning resources. This makes it easy for us to learn and use React in developing and maintaining our applications.

4.5 Install and configure the environment

Introduce:

Our Online Books project uses JavaScript technology and tools like Node.js, Express, MongoDB for the backend and React for the frontend. The project's goal is to provide a platform to manage book products, provide login features for admins and students, and display statistical information on the admin dashboard page.

Environmental Settings:

Backend:

- ❖ Node.js and npm: First, we installed Node.js and npm on our computer so we can run JavaScript code on the server. This is necessary for application deployment using Node.js.
- ❖ MongoDB: Next, we installed MongoDB to store data for our application. MongoDB provides a flexible unstructured database that fits your project's needs.
- ❖ Clone Project: We cloned the project from the repository to start working on its source code. This helps us save time and start development right from existing facilities.

Frontend:

- ❖ Node.js and npm: Similar to the backend, we installed Node.js and npm to develop the frontend of the application using React.
- ❖ Clone Project: We cloned the React project from the repository to start working on the app's user interface.

Environmental Configuration:

Backend:

- We have created an .env file to store environment variables like PORT and MONGODB_URI. This helps us easily configure environment variables for development and deployment environments.
- We then installed the dependencies using npm so we could use the necessary libraries and extension packs for our project.

Frontend:

- There is no special environment configuration for the frontend, but we have also created an .env file so we can define environment variables if necessary.
- Similar to the backend, we installed dependencies using npm so we can use them during development.

Project Launch:

- For the backend, we started the Node.js server with the npm start command and made sure the backend could connect to our MongoDB database.
- For the frontend, we launched the React app using the npm start command and opened a web browser to access the app's user interface.

4.6 Develop each module

4.6.1 Login and Authentication

Description: This module allows users to log in to the system as admin or student. It also includes user authentication to ensure proper access to other application functions.

Development Work:

- Build a user interface for entering login information.
- Handle login requests and authenticate users on the backend.
- Uses sessions and cookies to maintain user login status.

4.6.2 Book Product Management

Description: This module allows admins to perform CRUD (Create, Read, Update, Delete) operations on the list of book products, including adding new books, editing book information, and deleting books from the database. Whether.

Development Work:

- Build a user interface to display book lists and book management functions.
- Handle requests from users and perform corresponding operations on the MongoDB database.

4.6.3 User Management:

Description: This module allows admins to perform user management, including adding new students and displaying information about all users on the system.

Development Work:

- Build a user interface to display user lists and user management functions.
- Handle requests from users and perform corresponding operations on the MongoDB database.

4.6.4 Admin Dashboard page:

Description: This module provides a dashboard page for admins to display general information about the system, including total number of books, total number of students, and total number of admins.

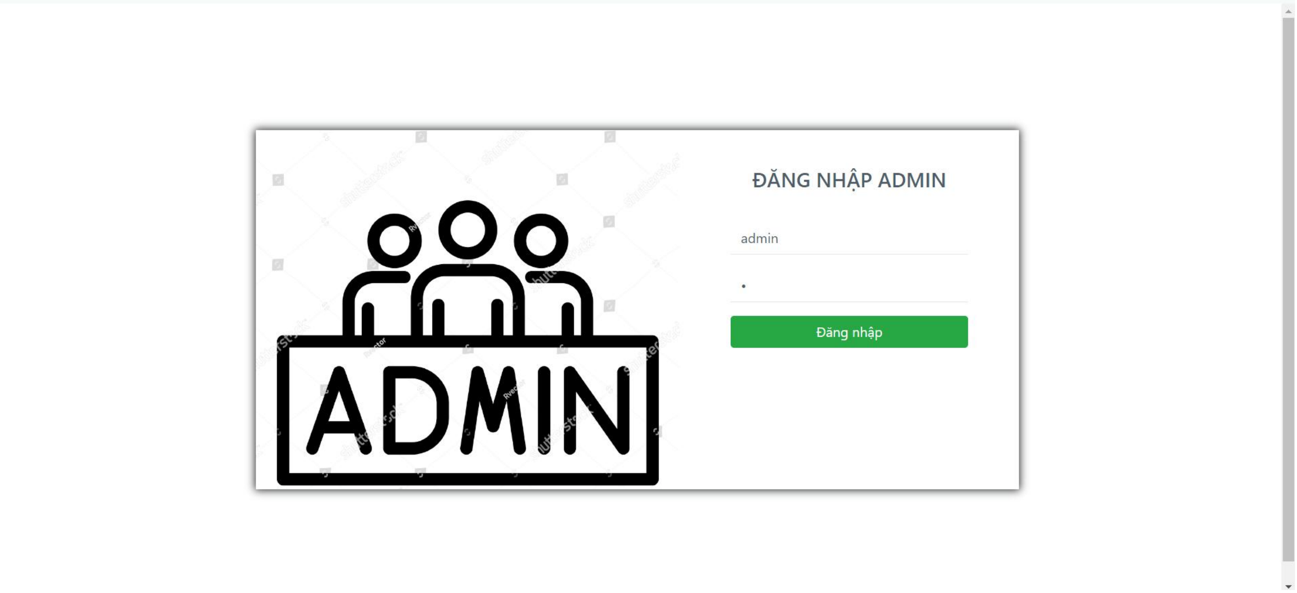
Development Work:

- Query the database to get statistical information about books, students, and admin.
- Display statistical information on the dashboard user interface.

Chapter 5. DEMO WEBSITE

5.1 Admin

5.1.1 Login

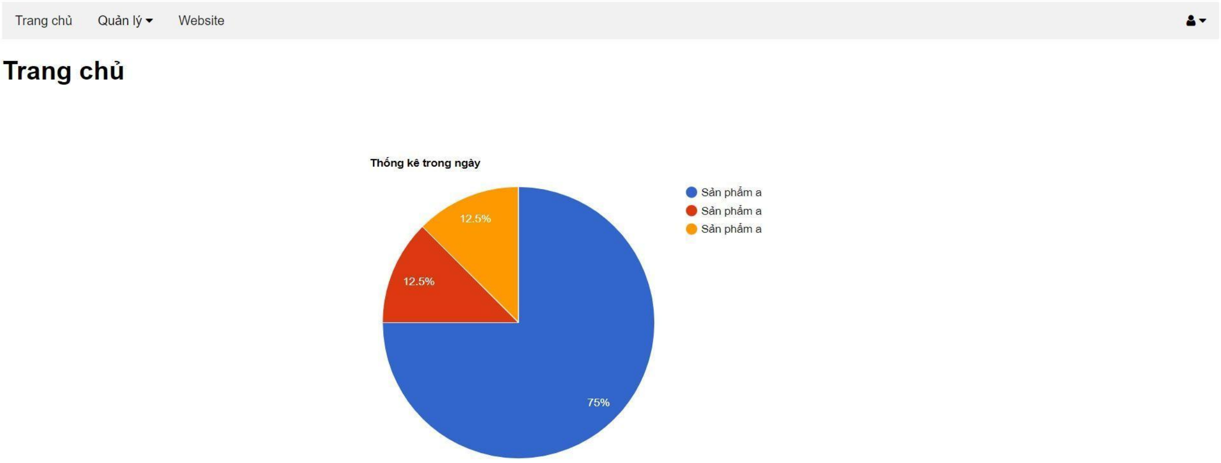


ĐĂNG NHẬP ADMIN

admin

Image 5: Admin login

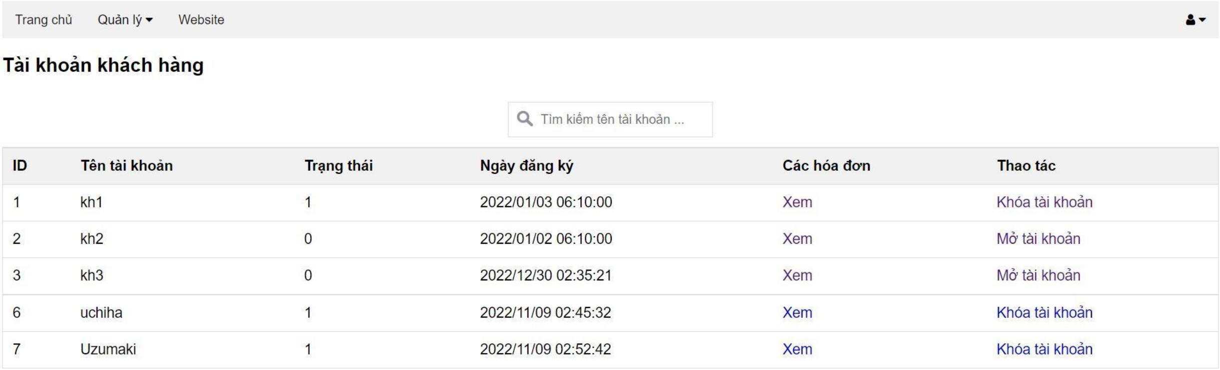
5.1.2 Home Page



Trang chủ

Image 6: Admin home page

5.1.3 Manage customer accounts

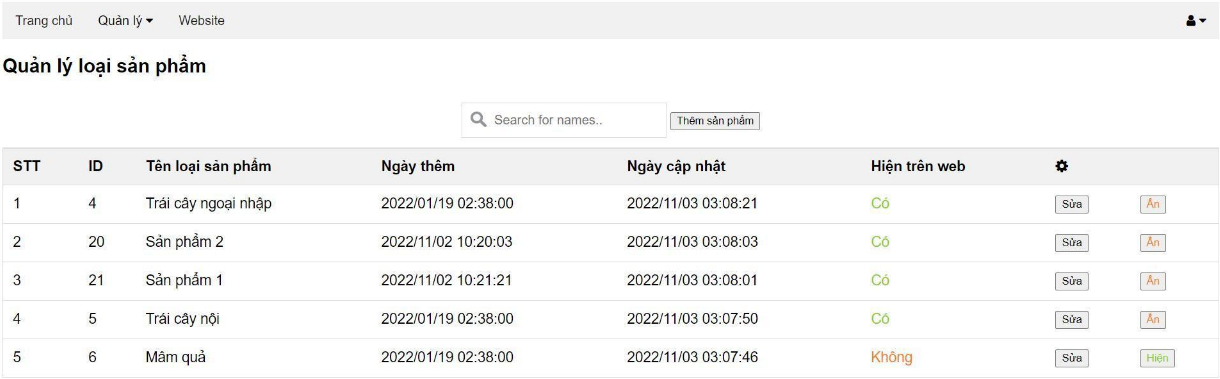


Tài khoản khách hàng

ID	Tên tài khoản	Trạng thái	Ngày đăng ký	Các hóa đơn	Thao tác
1	kh1	1	2022/01/03 06:10:00	Xem	Khóa tài khoản
2	kh2	0	2022/01/02 06:10:00	Xem	Mở tài khoản
3	kh3	0	2022/12/30 02:35:21	Xem	Mở tài khoản
6	uchiha	1	2022/11/09 02:45:32	Xem	Khóa tài khoản
7	Uzumaki	1	2022/11/09 02:52:42	Xem	Khóa tài khoản

Image 7: Manage customer accounts

5.1.4 Manage product types

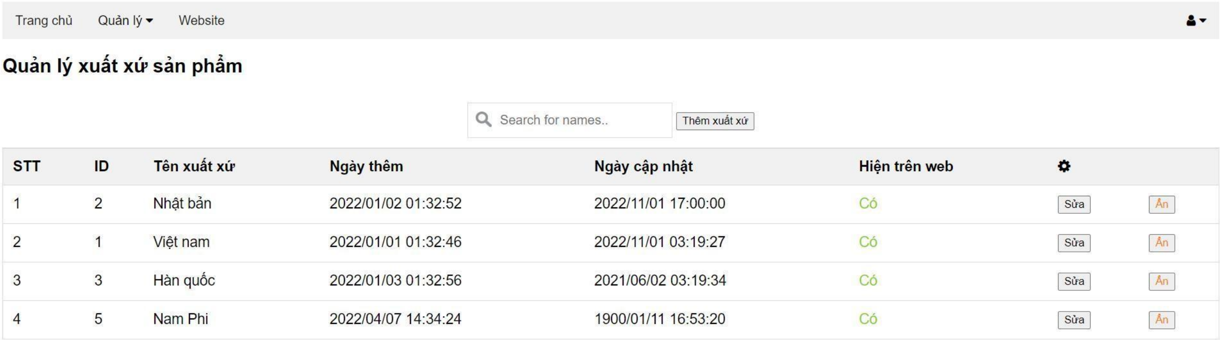


Quản lý loại sản phẩm

STT	ID	Tên loại sản phẩm	Ngày thêm	Ngày cập nhật	Hiện trên web	⚙	
1	4	Trái cây ngoại nhập	2022/01/19 02:38:00	2022/11/03 03:08:21	Có	Sửa	Ẩn
2	20	Sản phẩm 2	2022/11/02 10:20:03	2022/11/03 03:08:03	Có	Sửa	Ẩn
3	21	Sản phẩm 1	2022/11/02 10:21:21	2022/11/03 03:08:01	Có	Sửa	Ẩn
4	5	Trái cây nội	2022/01/19 02:38:00	2022/11/03 03:07:50	Có	Sửa	Ẩn
5	6	Mâm quả	2022/01/19 02:38:00	2022/11/03 03:07:46	Không	Sửa	Hiện

Image 8: Manage product types

5.1.5 Origin management

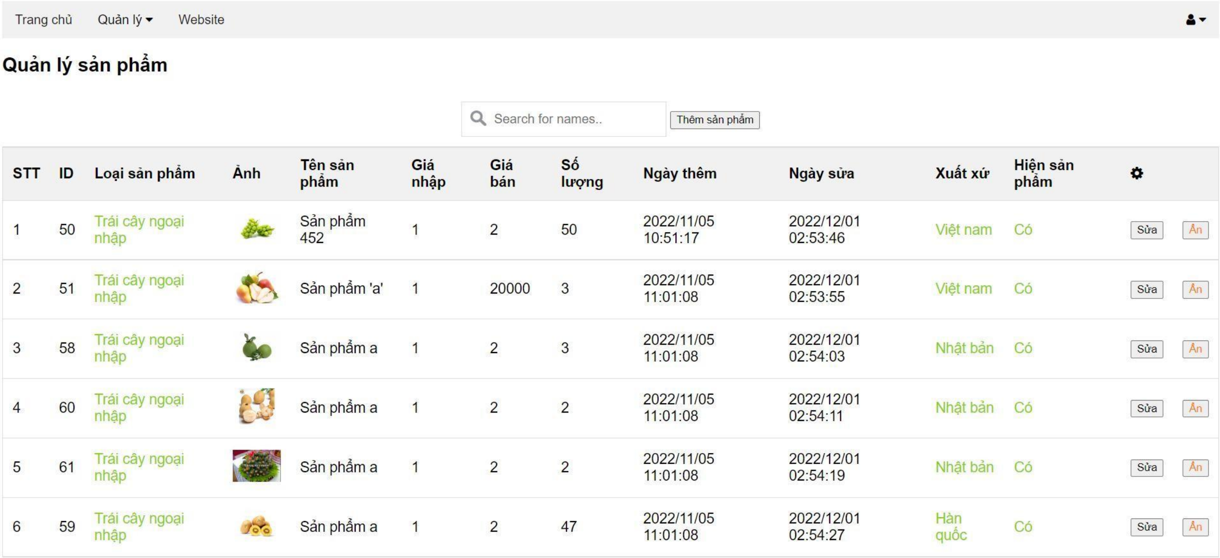


Quản lý xuất xứ sản phẩm

STT	ID	Tên xuất xứ	Ngày thêm	Ngày cập nhật	Hiện trên web	⚙	
1	2	Nhật bản	2022/01/02 01:32:52	2022/11/01 17:00:00	Có	Sửa	Ẩn
2	1	Việt nam	2022/01/01 01:32:46	2022/11/01 03:19:27	Có	Sửa	Ẩn
3	3	Hàn quốc	2022/01/03 01:32:56	2021/06/02 03:19:34	Có	Sửa	Ẩn
4	5	Nam Phi	2022/04/07 14:34:24	1900/01/11 16:53:20	Có	Sửa	Ẩn

Image 9: Origin management

5.1.6 Product Management

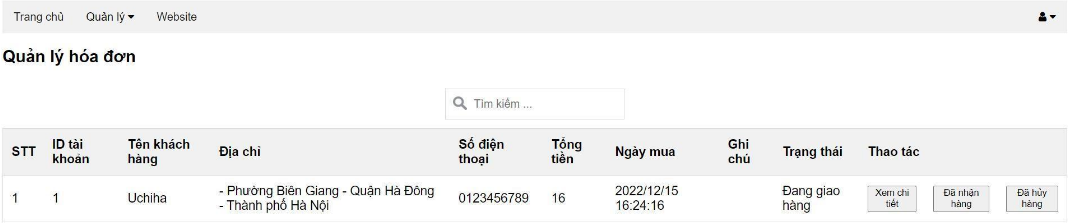


Quản lý sản phẩm

STT	ID	Loại sản phẩm	Ảnh	Tên sản phẩm	Giá nhập	Giá bán	Số lượng	Ngày thêm	Ngày sửa	Xuất xứ	Hiện sản phẩm	⚙	
1	50	Trái cây ngoại nhập		Sản phẩm 452	1	2	50	2022/11/05 10:51:17	2022/12/01 02:53:46	Việt nam	Có	Sửa	Ẩn
2	51	Trái cây ngoại nhập		Sản phẩm 'a'	1	20000	3	2022/11/05 11:01:08	2022/12/01 02:53:55	Việt nam	Có	Sửa	Ẩn
3	58	Trái cây ngoại nhập		Sản phẩm a	1	2	3	2022/11/05 11:01:08	2022/12/01 02:54:03	Nhật bản	Có	Sửa	Ẩn
4	60	Trái cây ngoại nhập		Sản phẩm a	1	2	2	2022/11/05 11:01:08	2022/12/01 02:54:11	Nhật bản	Có	Sửa	Ẩn
5	61	Trái cây ngoại nhập		Sản phẩm a	1	2	2	2022/11/05 11:01:08	2022/12/01 02:54:19	Nhật bản	Có	Sửa	Ẩn
6	59	Trái cây ngoại nhập		Sản phẩm a	1	2	47	2022/11/05 11:01:08	2022/12/01 02:54:27	Hàn quốc	Có	Sửa	Ẩn

Image 10: Product Management

5.1.7 Bill management

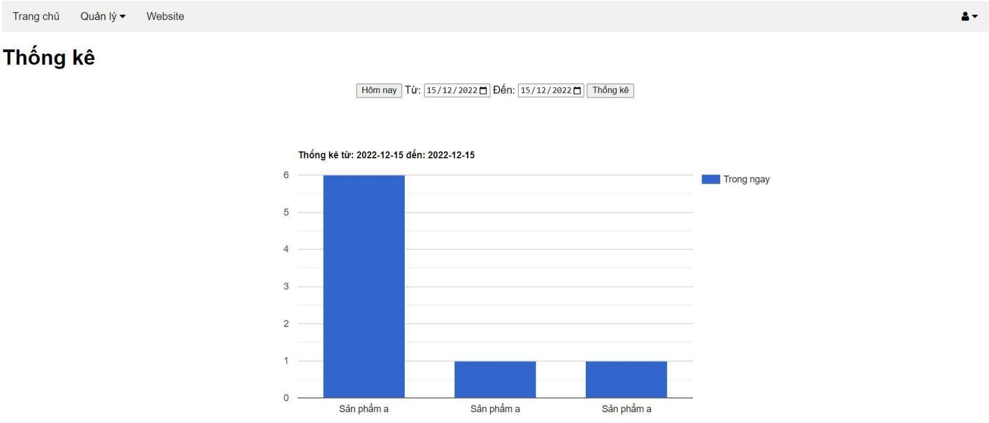


Quản lý hóa đơn

STT	ID tài khoản	Tên khách hàng	Địa chỉ	Số điện thoại	Tổng tiền	Ngày mua	Ghi chú	Trạng thái	Thao tác		
1	1	Uchiha	- Phường Biên Giang - Quận Hà Đông - Thành phố Hà Nội	0123456789	16	2022/12/15 16:24:16		Đang giao hàng	Xem chi tiết	Đã nhận hàng	Đã hủy hàng

Image 11: Bill management

5.1.8 Statistical



Thống kê

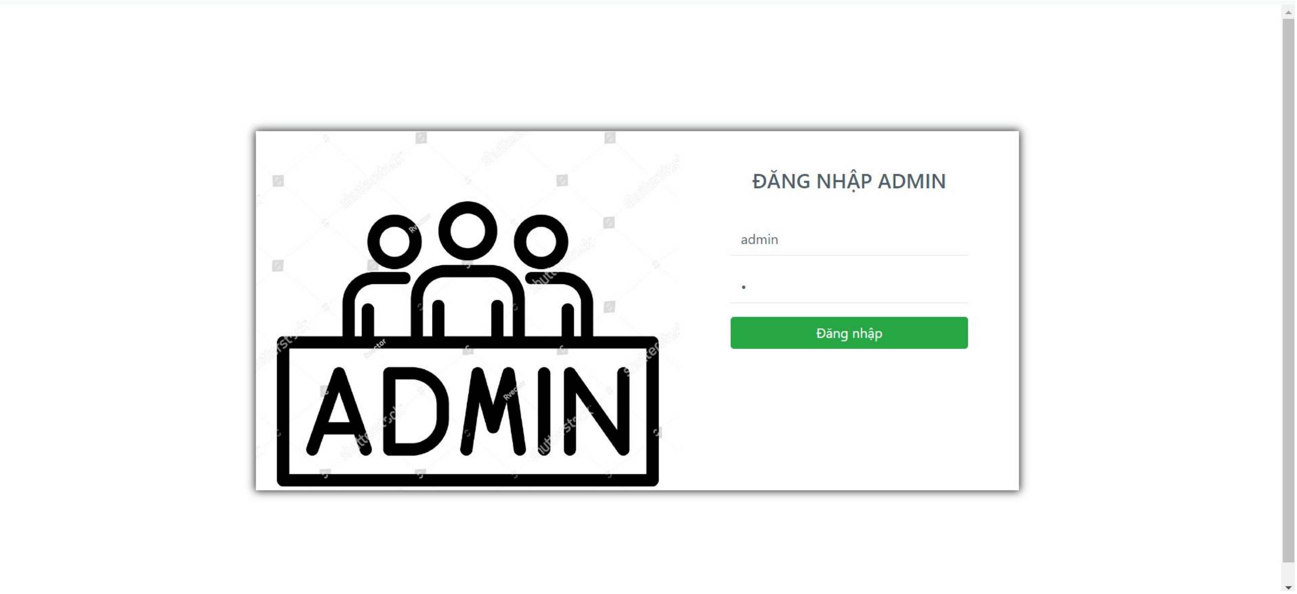
Từ: 15/12/2022 Đến: 15/12/2022

Thống kê từ: 2022-12-15 đến: 2022-12-15

Image 12: Statistical

5.2 Customer

5.2.1 Login



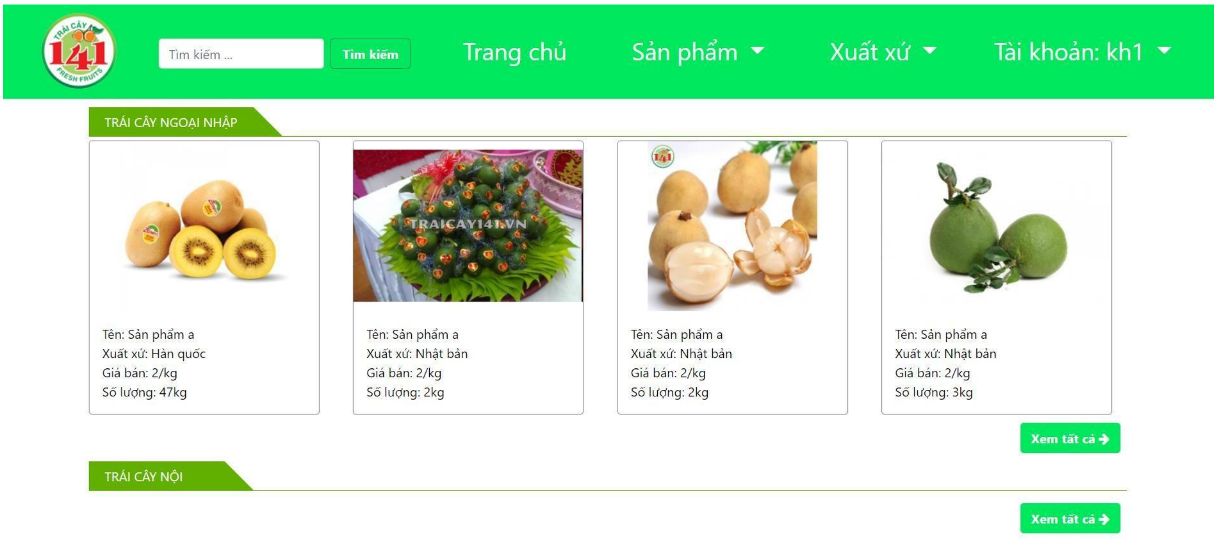
ĐĂNG NHẬP ADMIN

admin

•

Image 13: Login Customer

5.2.2 HomePage



TRÁI CÂY NGOẠI NHẬP

Tên: Sản phẩm a
Xuất xứ: Hàn quốc
Giá bán: 2/kg
Số lượng: 47kg

Tên: Sản phẩm a
Xuất xứ: Nhật bản
Giá bán: 2/kg
Số lượng: 2kg

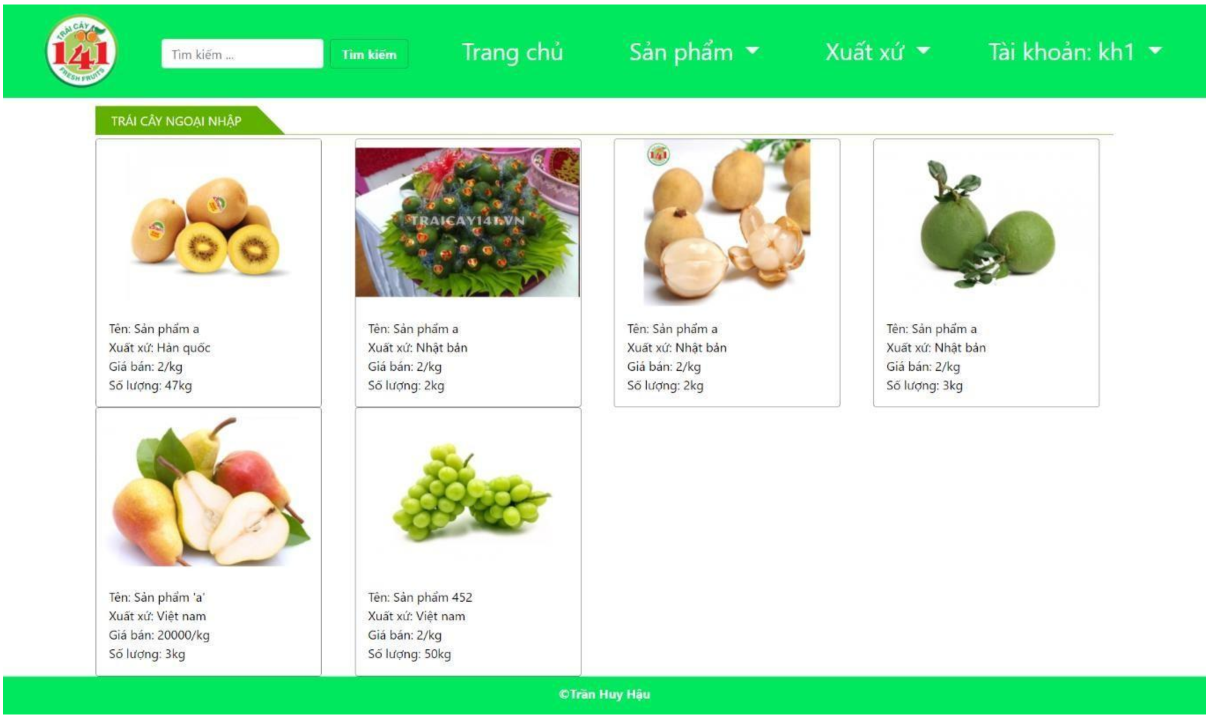
Tên: Sản phẩm a
Xuất xứ: Nhật bản
Giá bán: 2/kg
Số lượng: 2kg

Tên: Sản phẩm a
Xuất xứ: Nhật bản
Giá bán: 2/kg
Số lượng: 3kg

TRÁI CÂY NỘI

Image 14: HomePage Customer

5.2.3 Product



TRÁI CÂY NGOẠI NHẬP

Tên: Sản phẩm a
Xuất xứ: Hàn quốc
Giá bán: 2/kg
Số lượng: 47kg

Tên: Sản phẩm a
Xuất xứ: Nhật bản
Giá bán: 2/kg
Số lượng: 2kg

Tên: Sản phẩm a
Xuất xứ: Nhật bản
Giá bán: 2/kg
Số lượng: 2kg

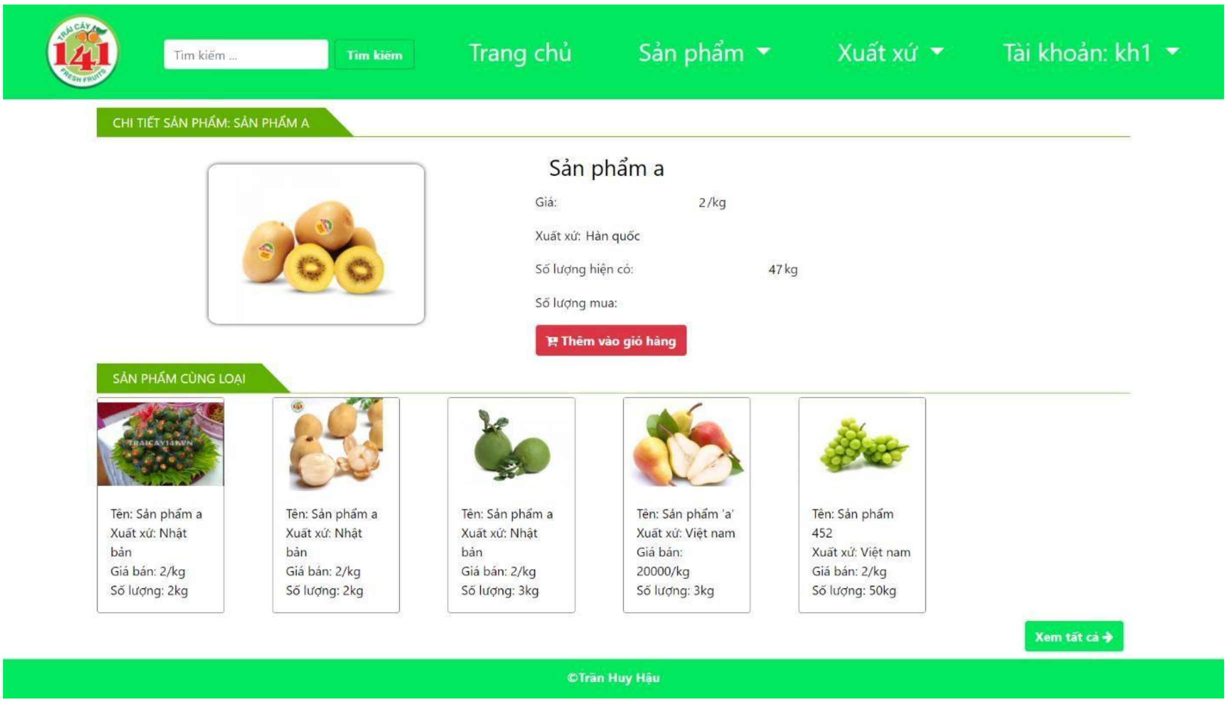
Tên: Sản phẩm a
Xuất xứ: Nhật bản
Giá bán: 2/kg
Số lượng: 3kg

Tên: Sản phẩm 'a'
Xuất xứ: Việt nam
Giá bán: 20000/kg
Số lượng: 3kg

Tên: Sản phẩm 452
Xuất xứ: Việt nam
Giá bán: 2/kg
Số lượng: 50kg

Image 15: Product page

5.2.4 Product details



CHI TIẾT SẢN PHẨM: SẢN PHẨM A

Sản phẩm a
Giá: 2/kg
Xuất xứ: Hàn quốc
Số lượng hiện có: 47kg
Số lượng mua:

SẢN PHẨM CÙNG LOẠI

Tên: Sản phẩm a
Xuất xứ: Nhật bản
Giá bán: 2/kg
Số lượng: 2kg

Tên: Sản phẩm a
Xuất xứ: Nhật bản
Giá bán: 2/kg
Số lượng: 2kg

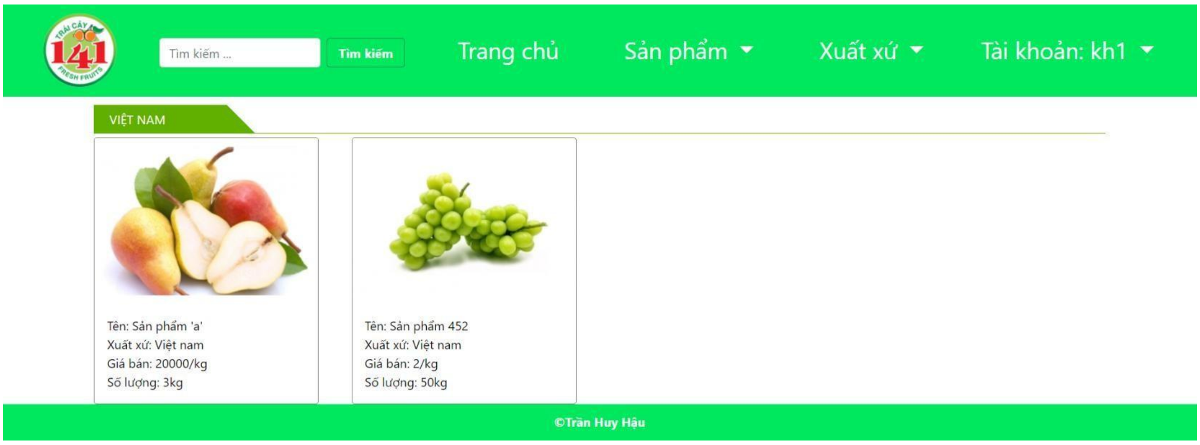
Tên: Sản phẩm a
Xuất xứ: Nhật bản
Giá bán: 2/kg
Số lượng: 3kg

Tên: Sản phẩm 'a'
Xuất xứ: Việt nam
Giá bán: 20000/kg
Số lượng: 3kg

Tên: Sản phẩm 452
Xuất xứ: Việt nam
Giá bán: 2/kg
Số lượng: 50kg

Image 16: Product details

5.2.5 Sort by Origin

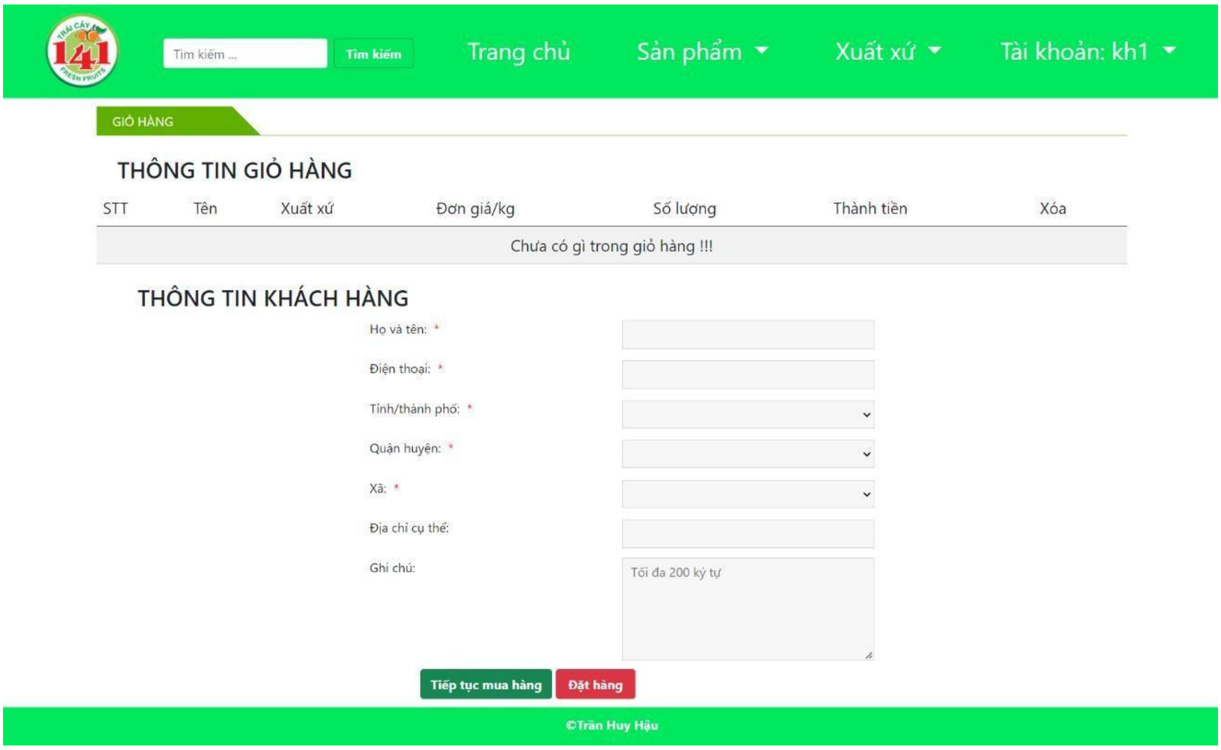


Tên: Sản phẩm 'a'
Xuất xứ: Việt nam
Giá bán: 20000/kg
Số lượng: 3kg

Tên: Sản phẩm 452
Xuất xứ: Việt nam
Giá bán: 2/kg
Số lượng: 50kg

Image 17: Sort by Origin

5.2.6 Cart



THÔNG TIN GIỎ HÀNG

STT	Tên	Xuất xứ	Đơn giá/kg	Số lượng	Thành tiền	Xóa
Chưa có gì trong giỏ hàng !!!						

THÔNG TIN KHÁCH HÀNG

Họ và tên: *

Điện thoại: *

Tỉnh/thành phố: *

Quận huyện: *

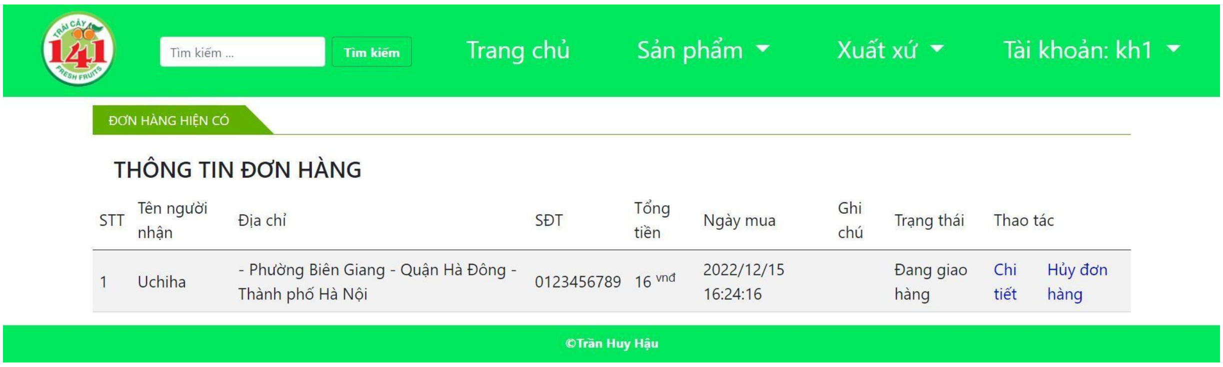
Xã: *

Địa chỉ cụ thể:

Ghi chú: Tối đa 200 ký tự

Image 18: Cart

5.2.7 Existing orders



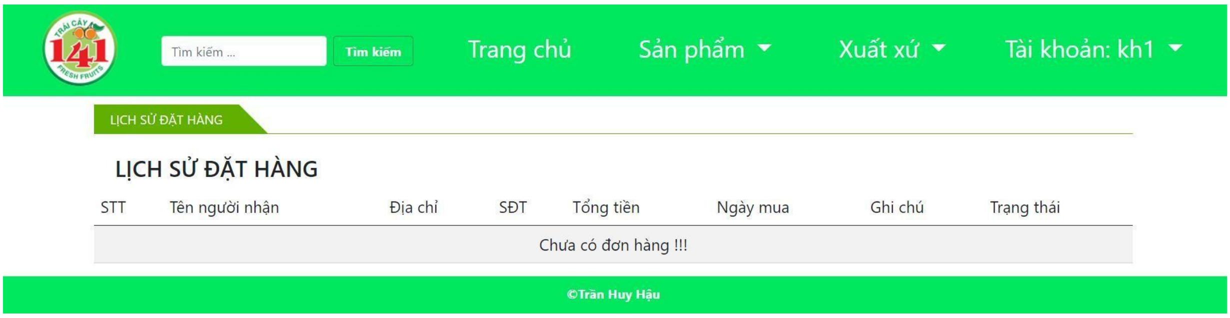
THÔNG TIN ĐƠN HÀNG

STT	Tên người nhận	Địa chỉ	SĐT	Tổng tiền	Ngày mua	Ghi chú	Trạng thái	Thao tác	
1	Uchiha	- Phường Biên Giang - Quận Hà Đông - Thành phố Hà Nội	0123456789	16 ^{vnđ}	2022/12/15 16:24:16		Đang giao hàng	Chi tiết	Hủy đơn hàng

©Trần Huy Hậu

Image 19: Existing orders

5.2.8 Order history



LỊCH SỬ ĐẶT HÀNG

STT	Tên người nhận	Địa chỉ	SĐT	Tổng tiền	Ngày mua	Ghi chú	Trạng thái
Chưa có đơn hàng !!!							

©Trần Huy Hậu

Image 20: Order history

Chapter 6. DEPLOPMENT

1. Diverse Selection Options: Allow users to choose from a variety of fruits and vegetables, including rare or specialty items.
2. Product Rating and Review System: Users can read reviews from others and rate products after purchase.
3. Advanced Search Functionality: Enable users to easily search for products by sorting by price, product type, origin, and other attributes.
4. Custom Order Options: Allow users to select quantities, sizes, or other specific characteristics for each product.
5. Detailed Product Information: Provide detailed information about the origin, cultivation methods, and nutritional information of each type of fruit and vegetable.
6. User-Friendly Mobile Interface: Ensure the website is mobile-friendly for users to shop anytime, anywhere.
7. Flexible Delivery and Payment Options: Offer multiple payment methods and flexible delivery options such as home delivery, same-day delivery, or specific-hour delivery.
8. Quick Reorder Feature: Enable users to quickly reorder previous orders.
9. Promotions and Discounts Program: Provide discount codes or special offers for large orders or regular customers.
10. Online Support: Offer live chat or phone support to assist users with inquiries and issue resolution.

REFERENCE

[1]. https://nodejs.org/en/about

[2]. https://expressjs.com

[3]. https://www.mongodb.com/company/what-is-mongodb

[4]. https://react.dev/learn