**MỤC LỤC**

[LỜI MỞ ĐẦU 4](#_Toc188441341)

[BẢNG PHÂN CÔNG CÔNG VIỆC 5](#_Toc188441342)

[NHẬN XÉT CỦA MENTOR 6](#_Toc188441343)

[Chương 1 – Giới thiệu 8](#_Toc188441344)

[1.1 Tổng quan: 8](#_Toc188441345)

[1.2.1 Loại người dùng 9](#_Toc188441346)

[1.2.2. Chức năng chính 10](#_Toc188441347)

[Chương 2 – Phân tích và xác định yêu cầu 11](#_Toc188441348)

[2.1 Chức năng chính của hệ thống: 11](#_Toc188441349)

[2.2 Biểu đồ usecase 12](#_Toc188441350)

[2.2.1 Mô tả tác nhân 12](#_Toc188441351)

[2.2.2 Mô tả Use Case 13](#_Toc188441352)

[2.3 Màn hình 15](#_Toc188441353)

[2.3.1 Luồng sự kiện màn hình 15](#_Toc188441354)

[2.3.2 Mô tả màn hình 15](#_Toc188441355)

[2.4 Yêu cầu cụ thể 19](#_Toc188441356)

[2.4.1 Yêu cầu chức năng 19](#_Toc188441357)

[2.4.2 Điều kiện hợp lệ cho các trường dữ liệu 48](#_Toc188441358)

[2.5 Yêu cầu phi chức năng 50](#_Toc188441359)

[2.5.1 Tính sử dụng 50](#_Toc188441360)

[2.5.2 Độ tin cậy 51](#_Toc188441361)

[2.5.3 Xử lý lỗi 51](#_Toc188441362)

[2.5.4 Tính khả dụng 51](#_Toc188441363)

[2.5.5 Các phụ thuộc của hệ thống 52](#_Toc188441364)

[2.5.6 Bugs and Defect Rates 52](#_Toc188441365)

[2.5.7 Hiệu năng 54](#_Toc188441366)

[2.5.8 Khả năng hỗ trợ 54](#_Toc188441367)

[Chương 3 – Thiết kế kiến trúc và tổ chức CODE 56](#_Toc188441368)

[3.1 Mô tả tổng quan các thành phần trong dự án: 56](#_Toc188441369)

[3.2 Xác định các quy định cần tuân thủ khi viết code: 58](#_Toc188441370)

[3.3 Trình bày cách tổ chức CODE trong dự án: 59](#_Toc188441371)

[3.4. Liệt kê các thư viện thirdparty sử dụng trong dự án: 60](#_Toc188441372)

[Chương 4 – Thiết kế cơ sở dữ liệu 61](#_Toc188441373)

[4.1. Mô hình thực thể liên kết (ERD): 61](#_Toc188441374)

[4.2 Lược đồ cơ sở dữ liệu quan hệ: 61](#_Toc188441375)

[4.2.1 Mô hình cơ sở dữ liệu quan hệ 62](#_Toc188441376)

[4.2.2 Mô tả cấu trúc bảng 62](#_Toc188441377)

[Chương 5 – Thiết kế giao diện 68](#_Toc188441378)

[5.1 Giao diện chức năng 1: Giao diện trang chủ 68](#_Toc188441379)

[5.2 Giao diện chức năng 2: Giao diện trang đăng nhập 68](#_Toc188441380)

[5.3 Giao diện chức năng 3: Giao diện trang đăng ký 69](#_Toc188441381)

[5.4 Giao diện chức năng 4: Giao diện trang danh sách sản phẩm 69](#_Toc188441382)

[5.5 Giao diện chức năng 5: Giao diện trang chi tiết sản phẩm 70](#_Toc188441383)

[5.6 Giao diện chức năng 6: Giao diện trang giỏ hàng 70](#_Toc188441384)

[5.7 Giao diện chức năng 7: Giao diện trang đặt hàng 71](#_Toc188441385)

[5.8 Giao diện chức năng 8: Giao diện trang danh sách blog 71](#_Toc188441386)

[5.9 Giao diện chức năng 9: Giao diện trang chi tiết blog 72](#_Toc188441387)

[5.10 Giao diện chức năng 10: Giao diện trang quản lý thông tin cá nhân 72](#_Toc188441388)

[5.11 Giao diện chức năng 11: Giao diện trang đổi mật khẩu 73](#_Toc188441389)

[5.12 Giao diện chức năng 12: Giao diện trang theo dõi đơn hàng 73](#_Toc188441390)

[5.13 Giao diện chức năng 13: Giao diện trang quản lý sản phẩm 74](#_Toc188441391)

[5.14 Giao diện chức năng 14: Giao diện trang quản lý brand 74](#_Toc188441392)

[5.15 Giao diện chức năng 15: Giao diện trang khách hàng 75](#_Toc188441393)

[Chương 6 - Kiểm thử 76](#_Toc188441394)

[6.1 Kiểm thử đơn vị 76](#_Toc188441395)

[6.1.1 Unit test cho authentication service 76](#_Toc188441396)

[6.2 Kiểm thử hệ thống 77](#_Toc188441397)

[Chương 7 – Hướng dẫn cài đặt và sử dụng 80](#_Toc188441398)

[7.1 Hướng dẫn cài đặt 80](#_Toc188441399)

[7.1.1 Yêu cầu hệ thống: 80](#_Toc188441400)

[7.1.2 Các bước cài đặt: 80](#_Toc188441401)

[7.1.3 Kiểm tra ứng dụng: 81](#_Toc188441402)

[7.2 Hướng dẫn sử dụng 81](#_Toc188441403)

[KẾT LUẬN 83](#_Toc188441404)

[Kết quả đạt được 83](#_Toc188441405)

[Hướng phát triển 83](#_Toc188441406)

[PHỤ LỤC 84](#_Toc188441407)

[TÀI LIỆU THAM KHẢO 85](#_Toc188441408)

# LỜI MỞ ĐẦU

Để có thể hoàn thành dự án này, lời đầu tiên em thay mặt nhóm 6 xin phép gửi lời cảm ơn tới anh Hà Khánh Toàn và các anh chị khác trong công ty đã tạo điều kiện thuận lợi cho nhóm em thực hiện dự án này.

Đặc biệt em xin chân thành cảm ơn anh Hà Khánh Toàn đã rất tận tình hướng dẫn, chỉ bảo em trong suốt thời gian thực hiện dự án vừa qua.

Em xin cam đoan dự án “Xây dựng website cho cửa hàng bán giày” là kết quả thực hiện của nhóm em dưới sự hướng dẫn của anh Hà Khánh Toàn.

Những phần sử dụng tài liệu tham khảo trong đồ án đã được nêu rõ trong phần tài liệu tham khảo. Các kết quả trình bày trong đồ án và chương trình xây dựng được hoàn toàn là kết quả do bản thân em thực hiện.

Mặc dù nhóm em đã có cố gắng, nhưng với trình độ còn hạn chế, trong quá trình thực hiện đề tài không tránh khỏi những thiếu sót. Nhóm em hi vọng sẽ nhận được những ý kiến nhận xét, góp ý của các anh, chị về những kết quả triển khai trong dự án.

Dự án đã được đẩy lên github:

https://github.com/nglong145/MockProject\_ShoesStore

Em xin trân trọng cảm ơn!

# BẢNG PHÂN CÔNG CÔNG VIỆC

|  |  |  |
| --- | --- | --- |
| Họ và tên | Công việc đã làm | Tiến độ |
| Nguyễn Văn Long | - Thiết kế API Authentication. - Thiết kế API Product. - Chỉnh sửa API Size. - Thiết kế API Blog. - Thiết kế API Review. - Thiết kế các API upload image. - Viết unit test cho authentication service.  - Thiết kế giao diện đăng nhập, đăng ký, trang chủ. - Thiết kế giao diện danh sách sản phẩm, chi tiết sản phẩm. - Thiết kế giao diện danh sách blog, xem chi tiết blog.  - Tích hợp API vào giao diện đăng nhập, đăng ký, quản lý thông  tin người dùng, thay đổi mật khẩu. - Tích hợp API vào giao diện lấy danh sách blog, xem chi tiết blog (User). - Tích hợp API vào giao diện lấy danh sách sản phẩm, lọc sản phẩm, xem chi tiết sản phẩm, đánh giá sản phẩm(User). | 80% |
| Trương Quang Linh | - Thiết kế API đặt hàng, CRUD giỏ hàng, CRUD phương thức thanh toán - Thiết kế giao diện giỏ hàng, đặt hàng, thông tin chi tiết người dùng, theo dõi đơn hàng của người dùng - Xây dựng unit test cho chức năng đặt hàng | 100% |
| Triệu Đức Hiếu | - thiết kê giao diện thêm, sửa product - thiết kê giao diện thêm, sửa brand - thiết kê giao diện thêm, sửa size - tích hợp API và làm chức năng add product, update product - tích hợp API và làm chức năng add product, update brand - tích hợp API và làm chức năng add product, update size |  |
| Hoàng Văn Hải | - Thiết kế giao diện tổng quan (Admin) - Thiết kế giao diện hiển thị danh sách Product, Brand, Blog (Admin) - Thiết kế API Brand, Blog, Size - Chỉnh sửa API User và thiết kế giao diện hiển thị, thay đổi trạng thái Customer - Tích hợp API và thiết kế giao diện thêm, sửa, xóa Blog (Admin) | 80% |

# NHẬN XÉT CỦA MENTOR

……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………........................................................................................................

# Chương 1 – Giới thiệu

## **1.1 Tổng quan:**

- Hệ thống quản lý bán giày được phát triển nhằm:

* Tạo một nền tảng trực tuyến thân thiện, dễ sử dụng cho khách hàng để xem, tìm kiếm, và mua giày mọi lúc, mọi nơi.
* Hỗ trợ doanh nghiệp quản lý sản phẩm, đơn hàng, khách hàng và doanh thu một cách hiệu quả.
* Cải thiện trải nghiệm khách hàng thông qua các tính năng như giỏ hàng, đánh giá sản phẩm, và khuyến mãi.
* Nâng cao khả năng tiếp cận của doanh nghiệp đến thị trường trực tuyến.

- Hệ thống sẽ tập trung vào các chức năng cơ bản để hỗ trợ hoạt động bán giày trực tuyến, bao gồm:

* Quản lý danh mục sản phẩm, thương hiệu, kích cỡ và màu sắc giày.
* Hỗ trợ khách hàng tìm kiếm, lọc, và đặt mua sản phẩm trực tuyến.
* Quản lý giỏ hàng, thanh toán và tình trạng đơn hàng.
* Cung cấp công cụ để quản trị viên theo dõi doanh số, quản lý kho, và điều hành các chương trình khuyến mãi.

#### **- Phạm vi sử dụng:**

* **Khách hàng cá nhân**: Người dùng muốn mua sắm giày trực tuyến.
* **Doanh nghiệp bán lẻ giày**: Chủ sở hữu cửa hàng hoặc nhà phân phối muốn quản lý sản phẩm và bán hàng trực tuyến.
* **Quản trị viên hệ thống**: Nhân viên chịu trách nhiệm quản lý sản phẩm, khách hàng, và đơn hàng.

Hệ thống quản lý bán giày là một nền tảng trực tuyến hỗ trợ khách hàng mua sắm giày dễ dàng và nhanh chóng, đồng thời cung cấp giao diện quản trị để quản lý sản phẩm, khách hàng, và đơn hàng một cách hiệu quả.

Hệ thống sẽ cung cấp các tính năng chính như: quản lý danh mục sản phẩm, giỏ hàng, thanh toán, đánh giá sản phẩm, quản lý khuyến mãi, và báo cáo doanh thu.

**1.2 Yêu cầu chức năng**

### ***1.2.1 Loại người dùng***

Hệ thống hỗ trợ 3 nhóm người dùng:

1. **Người dùng chưa đăng nhập**:
   * Xem sản phẩm, tìm kiếm, và lọc giày theo danh mục.
   * Thêm sản phẩm vào giỏ hàng nhưng cần đăng ký hoặc đăng nhập để hoàn tất giao dịch.
2. **Khách hàng (đã đăng nhập)**:
   * Quản lý tài khoản cá nhân (thay đổi thông tin, mật khẩu, quên mật khẩu).
   * Xem và chỉnh sửa giỏ hàng.
   * Đặt hàng và theo dõi tình trạng đơn hàng.
   * Đánh giá và nhận xét sản phẩm đã mua.
3. **Quản trị viên (Admin)**:
   * Quản lý sản phẩm (thêm, sửa, xóa sản phẩm).
   * Quản lý danh mục, thương hiệu, kích thước, và màu sắc giày.
   * Xử lý đơn hàng (cập nhật trạng thái, kiểm tra lịch sử mua hàng).
   * Quản lý khách hàng (xem, phân quyền, chặn/mở tài khoản khách hàng).
   * Theo dõi và xuất báo cáo doanh thu.

### ***1.2.2. Chức năng chính***

1. **Quản lý Sản phẩm**:
   * Hiển thị danh sách giày với thông tin chi tiết (hình ảnh, giá, kích cỡ, màu sắc).
   * Tìm kiếm và lọc sản phẩm theo thương hiệu, giá, kích cỡ.
2. **Giỏ hàng và Thanh toán**:
   * Hỗ trợ thêm, sửa, và xóa sản phẩm trong giỏ hàng.
   * Cung cấp giao diện thanh toán an toàn với các phương thức thanh toán phổ biến (thẻ tín dụng, ví điện tử).
3. **Đánh giá và nhận xét sản phẩm**:
   * Cho phép khách hàng đánh giá sản phẩm bằng sao và viết nhận xét.
   * Hiển thị các đánh giá cho khách hàng khác tham khảo.
4. **Quản lý Báo cáo**:
   * Xuất báo cáo doanh thu, sản phẩm bán chạy, và lịch sử giao dịch.

# Chương 2 – Phân tích và xác định yêu cầu

## **2.1 Chức năng chính của hệ thống:**

+ Update Personal Details

+ Write Review

+ Order Tracking

+ Add to Cart

+ Manage Cart

+ Checkout

+ Authenticated User

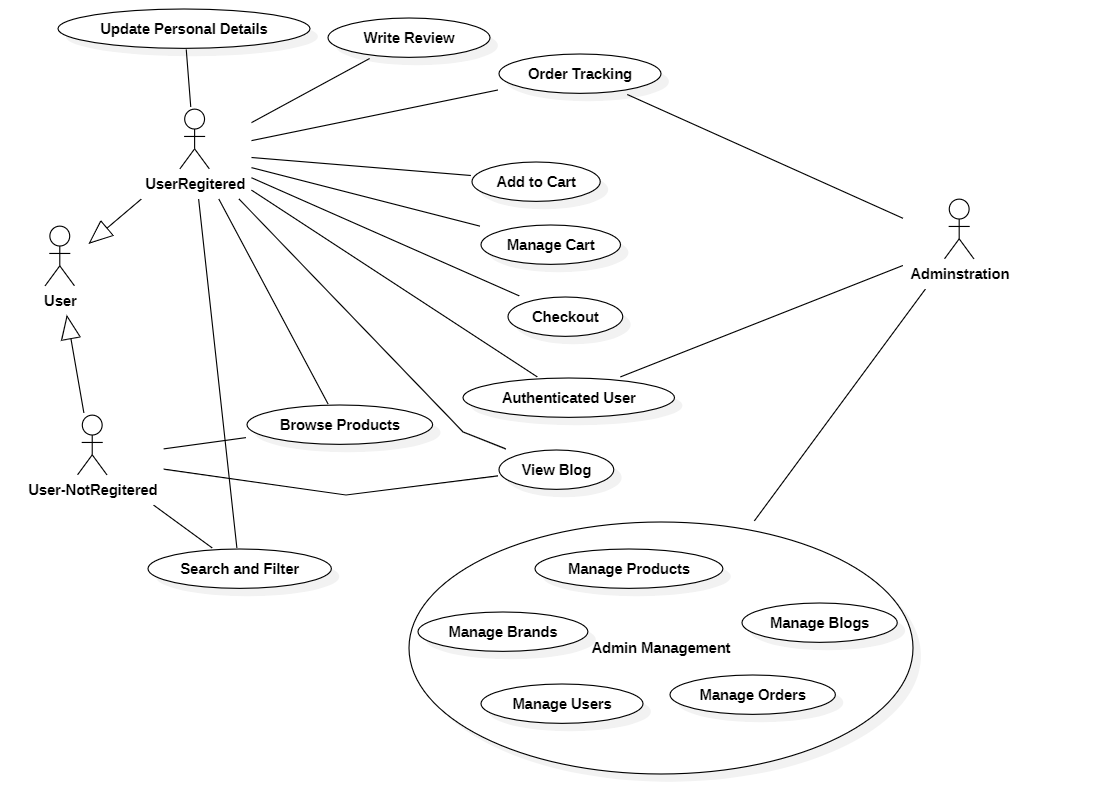
+ Browse Products

+ View Blog

+ Search and Filter

+ Admin Management

## **2.2 Biểu đồ usecase**



### ***2.2.1 Mô tả tác nhân***

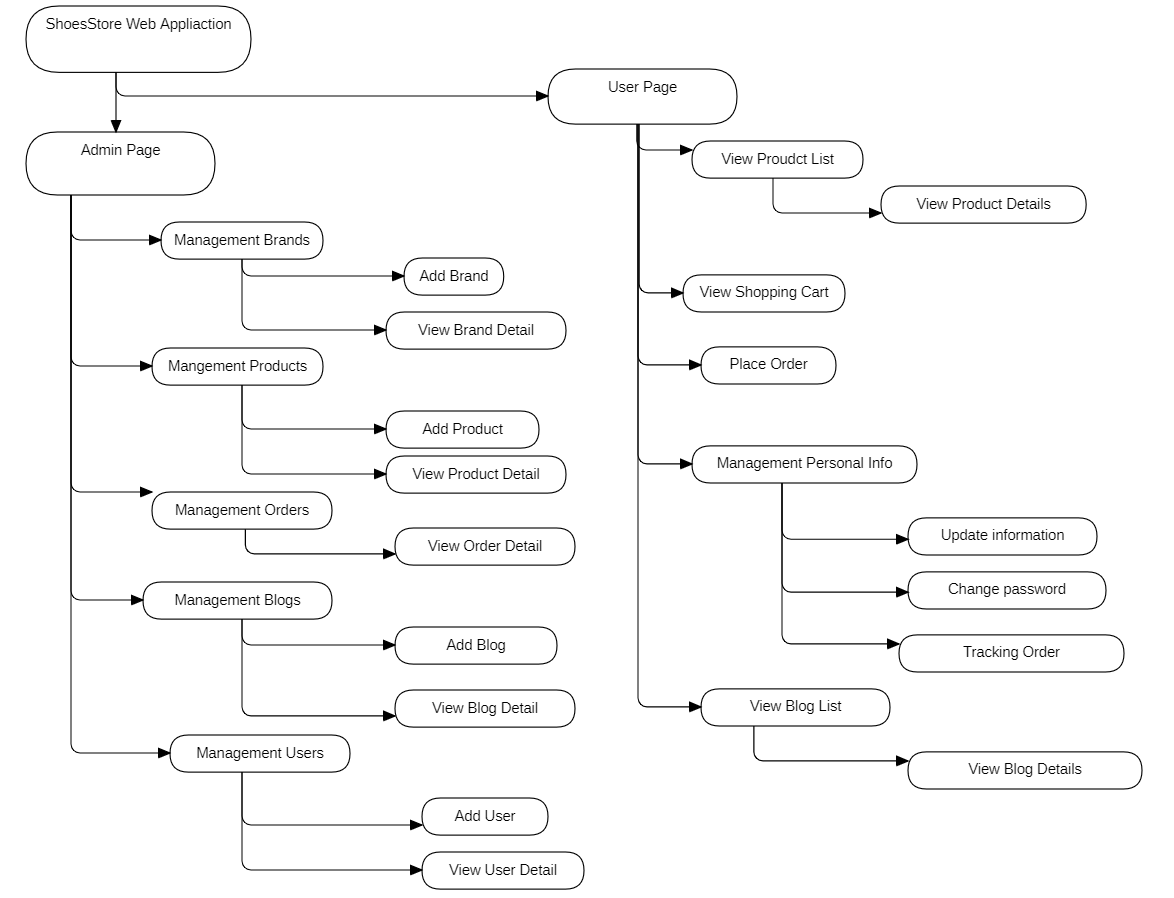
|  |  |  |
| --- | --- | --- |
| # | Actor | Description |
| 1 | User-NotRegisterd | Can view prodcut list, product detai and view blogs. |
| 2 | User-Registerd | Can Add product to cart, manage cart, checkout, tracking order and feedback product. In additon, they can change their personal information. |
| 3 | Administrator | Can manage brands, products, imports, orders, blogs of system. |

### ***2.2.2 Mô tả Use Case***

|  |  |  |  |
| --- | --- | --- | --- |
| # | Code | Name | Brief Description |
| 1 | UC01 | Update Personal Details | Allows user to update their personal information and change their password. |
| 2 | UC02 | Write Review | Allows user to share their feedback about the product |
| 3 | UC03 | Order Tracking | Allows user to track their orders |
| 4 | UC04 | Add to Cart | Allows user to add items to their shopping cart |
| 5 | UC05 | Manage Cart | Allows user to update their shopping cart |
| 6 | UC06 | Checkout | Allows user to place and pay for orders |
| 7 | UC07 | Browse Products | Allows user to see products curently avaiable in system |
| 8 | UC08 | View Blog | Allows user to see blogs curently avaiable in system |
| 9 | UC09 | Search and Filter | Allows user to search and filter products by options |
| 10 | UC10 | Manage Brands | Allows adminstrator to manage brands in the system by adding, updating, deleting, and viewing them. |
| 11 | UC11 | Manage Products | Allows adminstrator to manage products in the system by adding, updating, deleting, and viewing them. |
| 12 | UC12 | Manage Blogs | Allows adminstrator to manage blogs in the system by adding, updating, deleting, and viewing them. |
| 13 | UC13 | Manage Import | Allows adminstrator to view and add product import orders. |
| 14 | UC14 | Manage Orders | Allows user to check and modify the status of orders |
| 15 | UC15 | Authenticate User | System verifies permissions of user |

## **Màn hình**

### ***Luồng sự kiện màn hình***



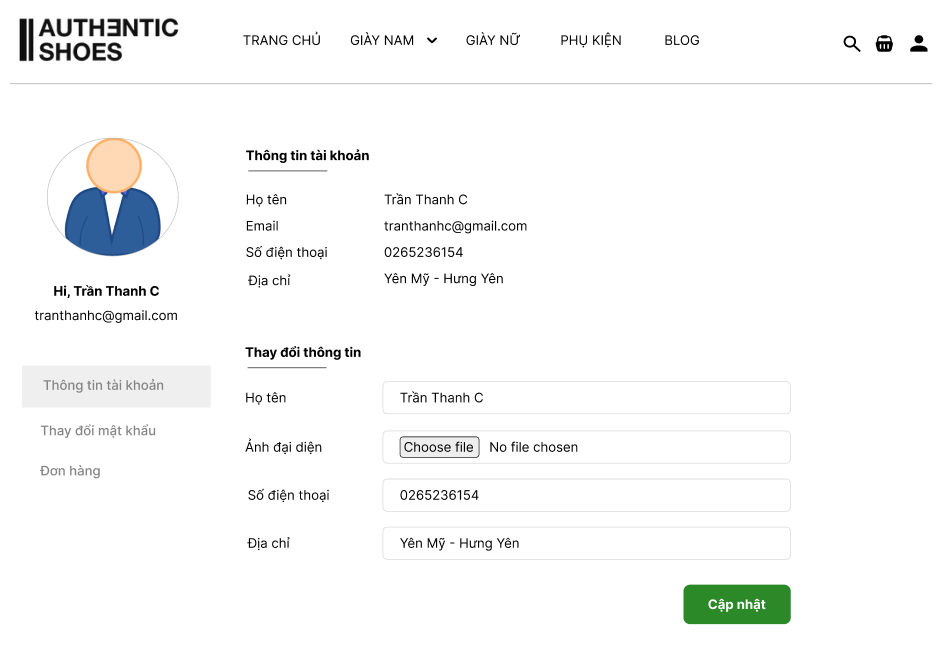
### ***Mô tả màn hình***

|  |  |  |
| --- | --- | --- |
| # | Screen name | Brief Description |
| 1 | Management Brands | Allows admin to view the list of bradns on the system website |
| 2 | Add Brand | Allows admin to add new brand on the system website |
| 3 | View Brand Detail | Allows admin to view the detail of brand and update brand |
| 4 | Management Products | Allows admin to view the list of products on the system website |
| 5 | Add Product | Allows admin to add new product on the system website |
| 6 | View Product Detail | Allows admin to view the detail of product and update product |
| 7 | Management Orders | Allows admin to view the list of ordes on the system website |
| 8 | View Order Detail | Allows admin to view the detail of order and update status |
| 9 | Management Blogs | Allows admin to view the list of blogs on the system website |
| 10 | Add Blog | Allows admin to add new blog on the system website |
| 11 | View Blog Detail | Allows admin to view the detail of blog and update blog |
| 12 | Management Users | Allows admin to view the list of users on the system website |
| 13 | Add User | Allows admin to add new user on the system website |
| 14 | View User Detail | Allows admin to view the detail of user and update user |
| 15 | View Product List | Allows user to view the list of products on the system website |
| 16 | View Product Details | Allows user to view the detail of product on the system website |
| 17 | View Review | Allows user to view feedback from other customers about the product |
| 18 | Write review | Allows user to write their feedback about the product after testing it |
| 19 | Add to Cart | Allows user to add products to the shopping cart |
| 20 | View Shopping Cart | Allows user to view products in the shopping cart |
| 21 | Update quantity/size | Allows user to modify the quantity or size of item in the shopping cart |
| 22 | Delete Product | Allows user to remove item from the shopping cart |
| 23 | Place Order | Allows user to enter and confirm information to place an order |
| 24 | Payment | Allows user to choose a payment method to place an order |
| 25 | Management Personal Info | Allows user to view and manage personal information |
| 26 | Update information | Allows user to update personal information |
| 27 | Change password | Allows user to change password |
| 28 | Tracking Order | Allows user to track orders |
| 29 | View Blog List | Allows user to view the list of blogs on the system website |
| 30 | View Blog Details | Allows user to view the detail of blog on the system website |

# Yêu cầu cụ thể

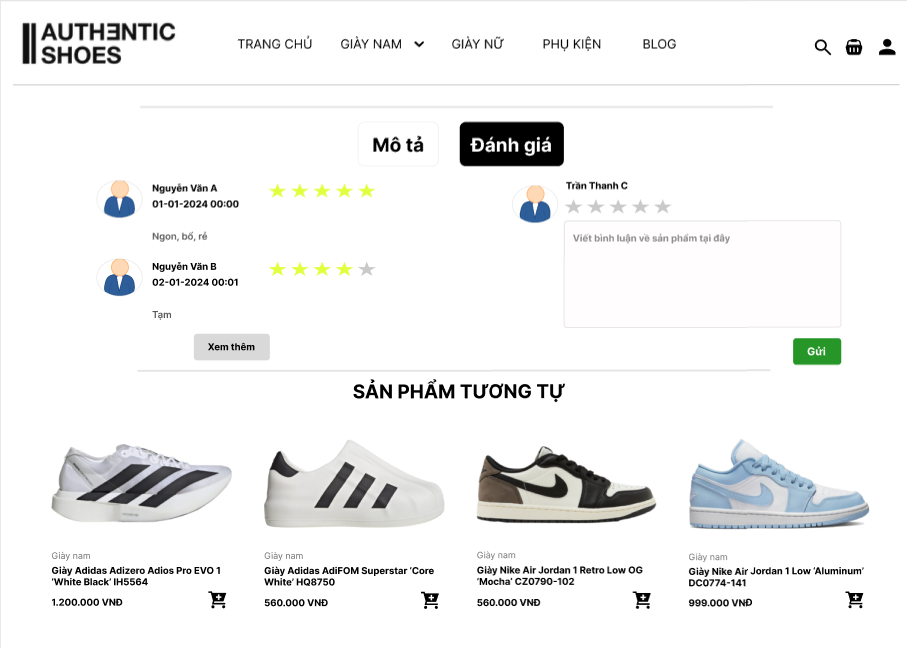
## ***2.4.1 Yêu cầu chức năng***

1. *UC01: Update Personal Information*



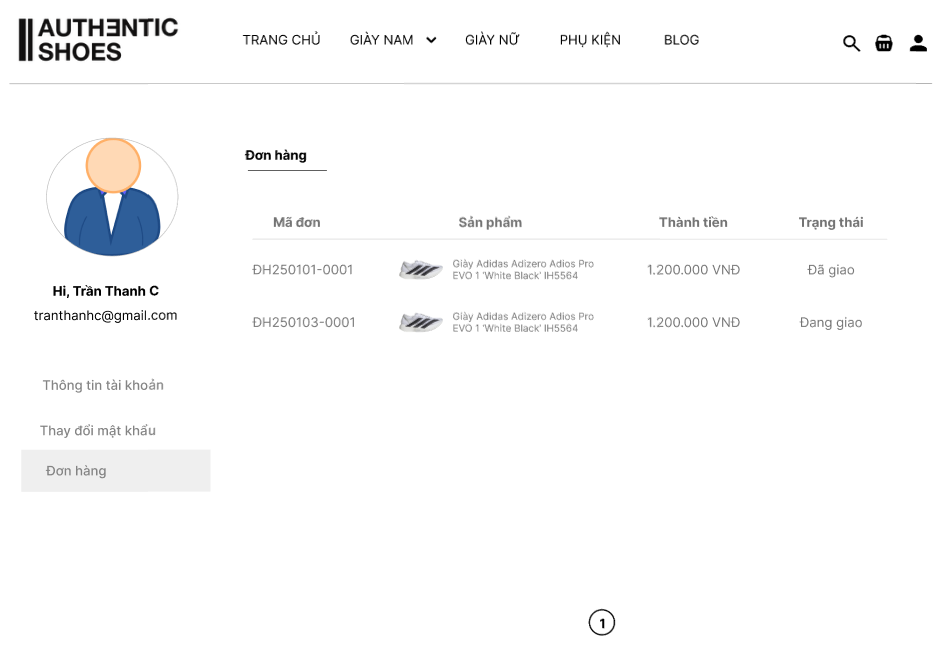
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC01: Update Personal Details** | | | | | |
| Goal in Context | | | This use case allows the actor to view the personal information of their account.  The actor can update their personal information. | | |
| Pre-Condition | | | The actor has successfully logged into the website application. | | |
| Post-Condition | | | The tool shall bring back data that is appropriate to the actor’s request.  The system update data into database. | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | The actor has successfully logged into the website application and purchased this product. | | | |
| 1. | User | Click | | “User” Icon in the Header bar | Website displays detailed personal information for the account. |
| 2. | User | Input | | Cells | Input data into “Full name”, “Phone number”, “Address” cells, selected file in “Avatar image” cells.  (see [3.2. Data field valid conditions](#_Data_field_validation)). |
| 3. | User | Click | | Button “Update” | System checks empty condition on field “Full name”, “Phone number” and “Address”. If fields is empty then displays message “Field is empty. Please fill!”  Else  If validation process is completed successfully without any warning message then system saves the personal information into database. |

1. *UC02: Write Review*

****

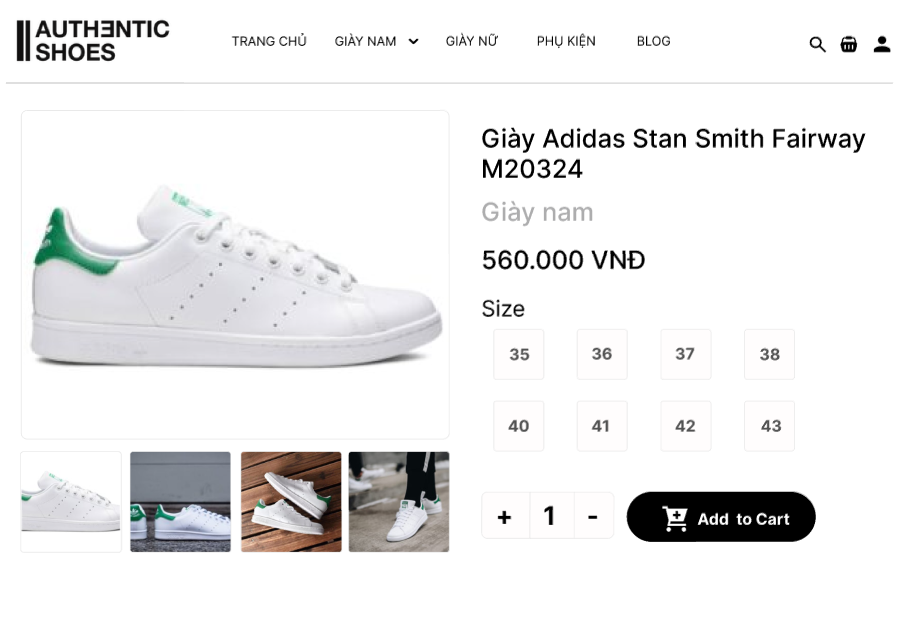
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC02: Write Review** | | | | | |
| Goal in Context | | | This use case allows the actor to display feedback from other customer.  The actor can write their own review about the product. | | |
| Pre-Condition | | | The actor has successfully logged into the website application.  The actor has purchased this product. | | |
| Post-Condition | | | The tool shall bring back data that is appropriate to the actor’s request | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | The actor has successfully logged into the website application and purchased this product. | | | |
| 1. | User | Click | | “Product” in “Product List” Table | Website displays detailed information about the product. |
| 2. | User | Click | | The “Review” section under the product image | System displays the feedback from other customers and a form to user can write their review about the product |
| 3. | User | Click | | “Star” Icon | System highlights the selected star icons is yellow. |
| 4. | User | Input | | Cells | Input data into “Your feedback” cell (see [3.2. Data field valid conditions](#_Data_field_validation)).  Actor cannot submit a review until all valid conditions of fields on that row are satisfied. |
| 5. | User | Click | | Button “Send” | System checks empty condition on field “Your Feedback" and number of star. If fields is empty then displays message “You can send a review empty.”  Else  System checks duplicate data, if the user's review already exists then displays message “You already review this product.”  If validation process is completed successfully without any warning message then system saves the review into database. |

1. UC03: Order Tracking

****

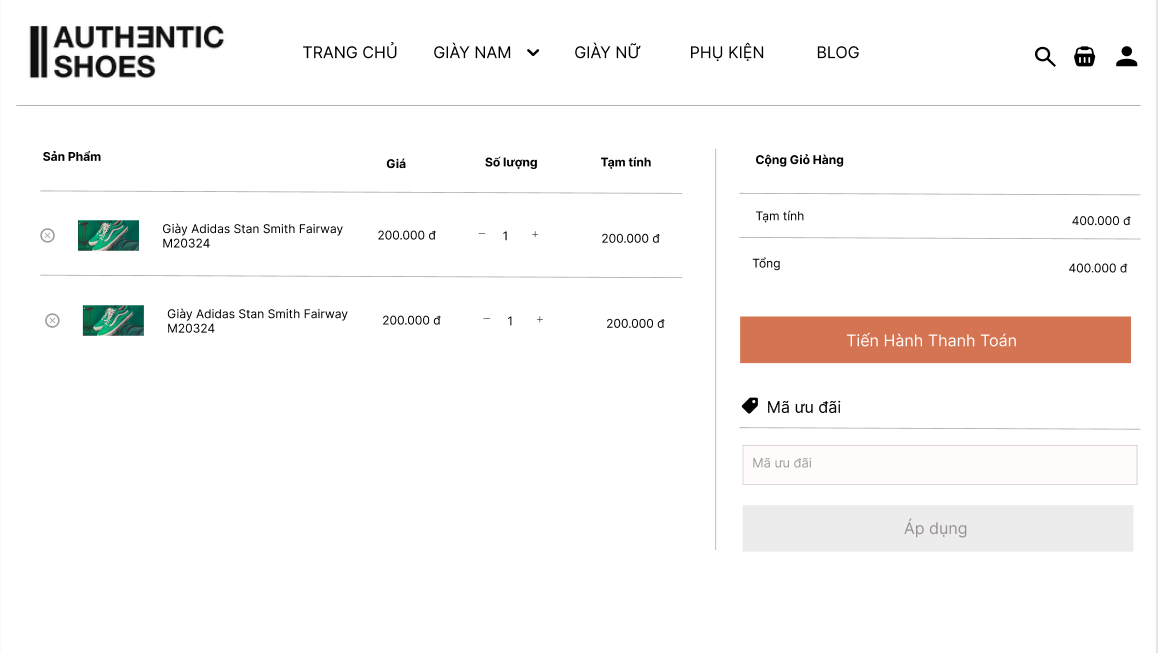
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC03: Order Tracking** | | | | | |
| Goal in Context | | | This use case allows the actor to view the list of all orders. | | |
| Pre-Condition | | | The actor has successfully logged into the website application. | | |
| Post-Condition | | | The tool shall bring back data that is appropriate to the actor’s request | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | The actor has successfully logged into the website application. | | | |
| 1. | User | Click | | The “Order” section on the left sidebar | Website displays detailed information about the product. |
| 2. | User | Click | | Any Order | System returns and displays the detailed information of the selected order on the screen. |

1. UC04: Add to Cart

****

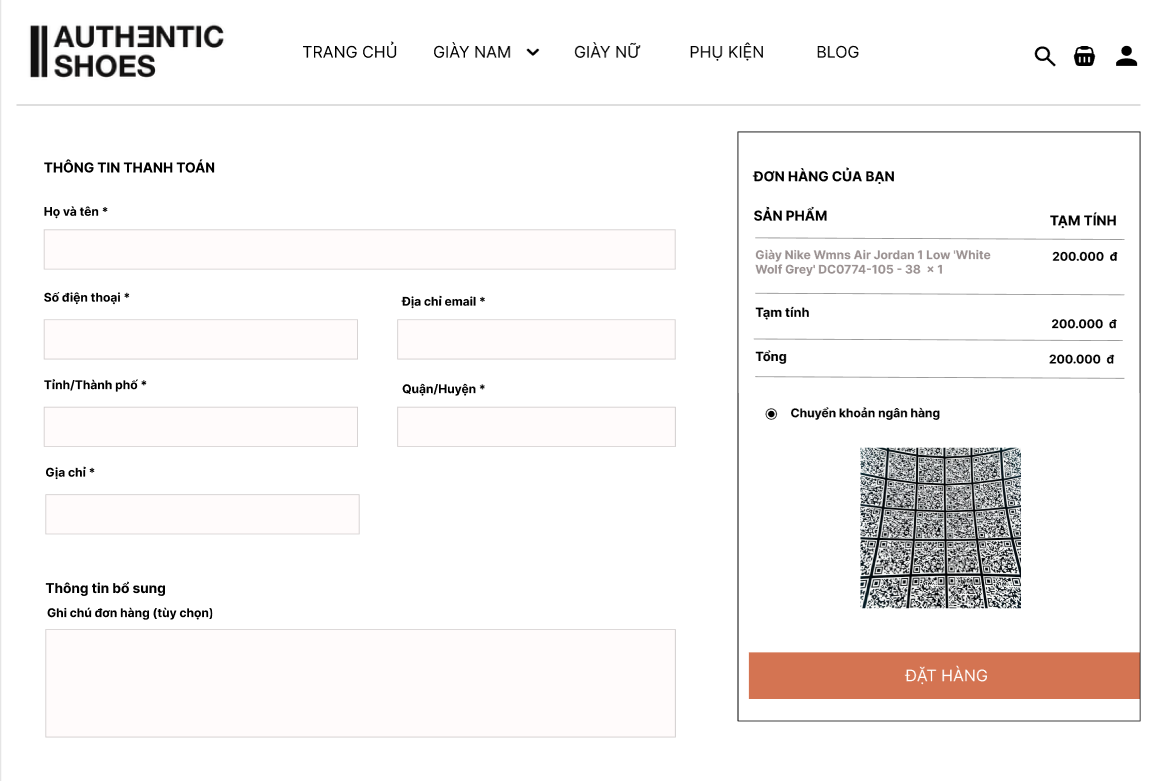
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC04: Add to Cart** | | | | | |
| Goal in Context | | | This use case allows the actor to add products to the shopping cart. | | |
| Pre-Condition | | | The actor has successfully logged into the website application. | | |
| Post-Condition | | | The tool shall bring back data that is appropriate to the actor’s request. | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | The actor has successfully logged into the website application. | | | |
| 1. | User | Click | | “Product” in “Product List” Table | Website displays detailed information about the product. |
| 2. | User | Click | | The “Size” in the table Size. | Website checks quantity of size product. If quantiy is sufficent, the size box is highlighted.  Else  Website displays message “This size is out of stock. Please select a different size”. |
| 3. | User | Input/Click | | “Quantity” under the table Size | Website checks quantity of size product. If quantiy is sufficent, display the product quantity based on the number entered by the user.  Else  Website displays message “This size is out of stock. Please select a different size”. |
| 4. | User | Click | | “Add to cart” Button | Website add product into the shopping cart and update the quantity of products currently in the shopping cart. |

1. UC05: Manage Cart



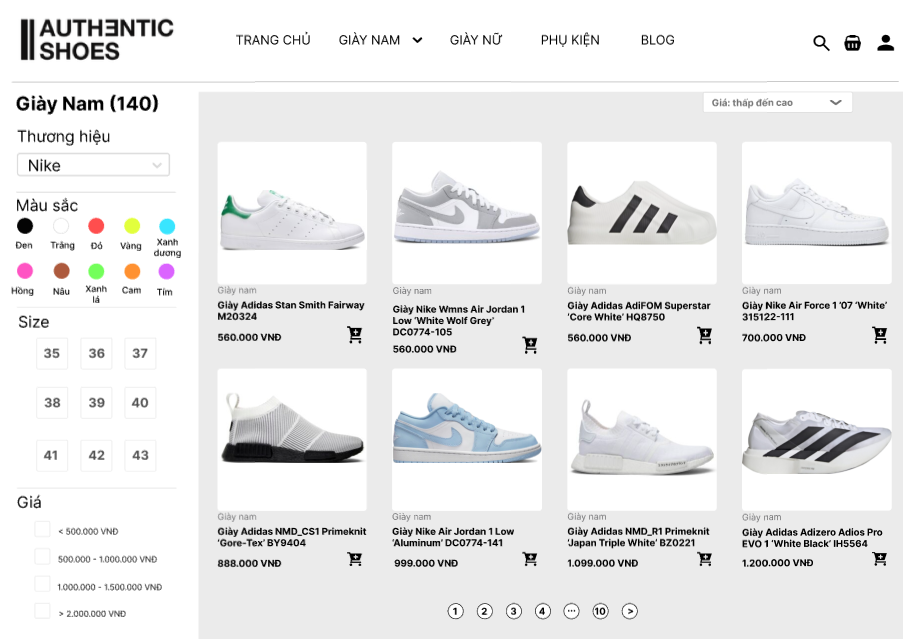
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC05: Manage Cart** | | | | | |
| Goal in Context | | | This use case allows users to view the list of products in their shopping cart. Users can add products to the cart, update product quantities or size, or remove products from the cart. Users can also proceed to checkout and make payments for the products in the shopping cart. | | |
| Pre-Condition | | | The actor has successfully logged into the website application. | | |
| Post-Condition | | | The tool shall bring back data that is appropriate to the actor’s request | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | The actor has successfully logged into the website application. | | | |
| 1. | User | Click | | Icon “Shopping Cart” on the menu bar | Website displays the shopping cart screen. |
| 2. | User | Click | | Icon “ X ” next to the Product | The product is removed from the shopping cart, and the cart list is updated. |
| 3. | User | Input/Click | | Product Quantity in the Input Box Or Button “+”, Buton” - “ | The product quantity is updated, and the total amount is automatically recalculated. |
| 4. | User | Input | | "Proceed to Checkout" Button | The system navigates to the checkout screen. |

1. UC06: Checkout



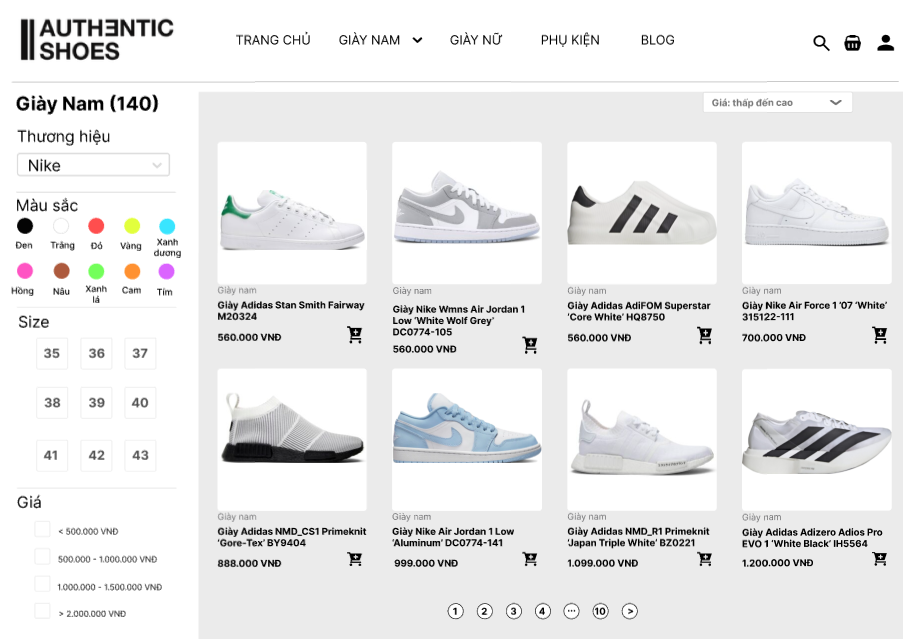
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC06: Checkout** | | | | | |
| Goal in Context | | | This use case allows users to complete their order payment by:   * Entering shipping information (full name, phone number, and address). * Selecting a payment method. * Confirming the order and receiving a success notification. | | |
| Pre-Condition | | | The user has added at least one product to the shopping cart.  The user has logged in or provided contact information. | | |
| Post-Condition | | | The order is stored in the system.  The system notifies the user of successful payment confirmation. | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | The user has logged in or provided the required information (full name, address, phone number, and email). | | | |
| 1. | User | Input | | Full Name, Phone Number, Email, Address | The system verifies and validates the provided information. |
| 2. | User | Select | | City/Province and District | The system displays a list of applicable areas. |
| 3. | User | Choose | | Payment Method (Bank Transfer, COD, etc.) | The system saves the selected payment method. |
| 4. | User | Click | | "Place Order" Button | The system checks the information, creates the order, and displays a success notification. |

1. UC07: Browse Products

****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC07: Browse Products** | | | | | |
| Goal in Context | | | This use case allows user to browse products by category. | | |
| Pre-Condition | | | None. | | |
| Post-Condition | | | The tool shall bring back data that is appropriate to the actor’s request. | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | None. | | | |
| 1. | User | Click | | Category in the header bar | Website displays the list of products along with their summary information. |
| 2. | User | Select | | Any Product | Website returns and displays the detailed information of the selected product on the screen. |

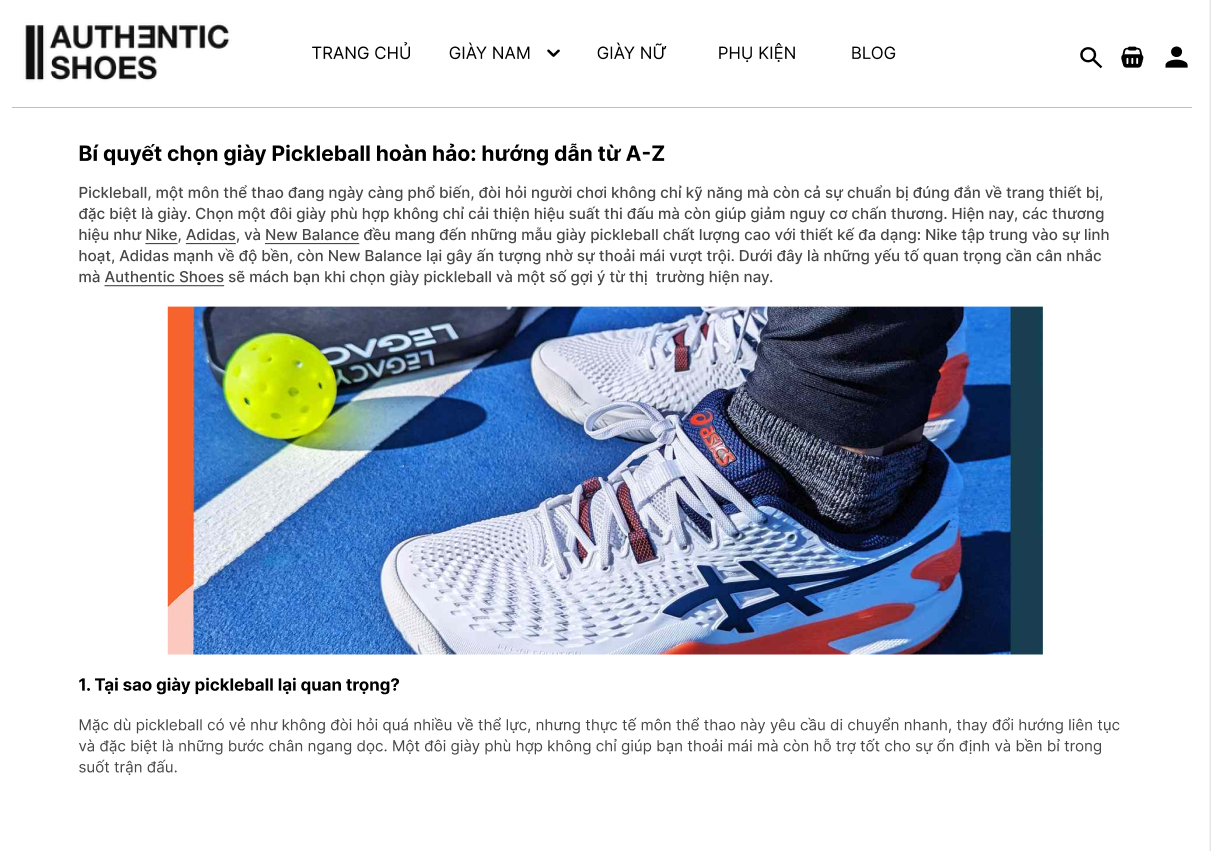
1. UC08: Search and Filter

****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC08: Search and Filter** | | | | | |
| Goal in Context | | | This use case allows user to search and filter products by brand, colour, size, price as well as sort them by other criteria. | | |
| Pre-Condition | | | None. | | |
| Post-Condition | | | The tool shall bring back data that is appropriate to the actor’s request. | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | None. | | | |
| 1. | User | Select | | Brand combobox | Website returns and displays the list of products on the screen by brand. |
| 2. | User | Click | | Colour, Size box and “Price” checkbox | Website returns and displays the list of products on the screen by colour, size and price. |
| 3 | User | Select | | Sort combobox | Website returns and displays the list of products on the screen by criteria. |

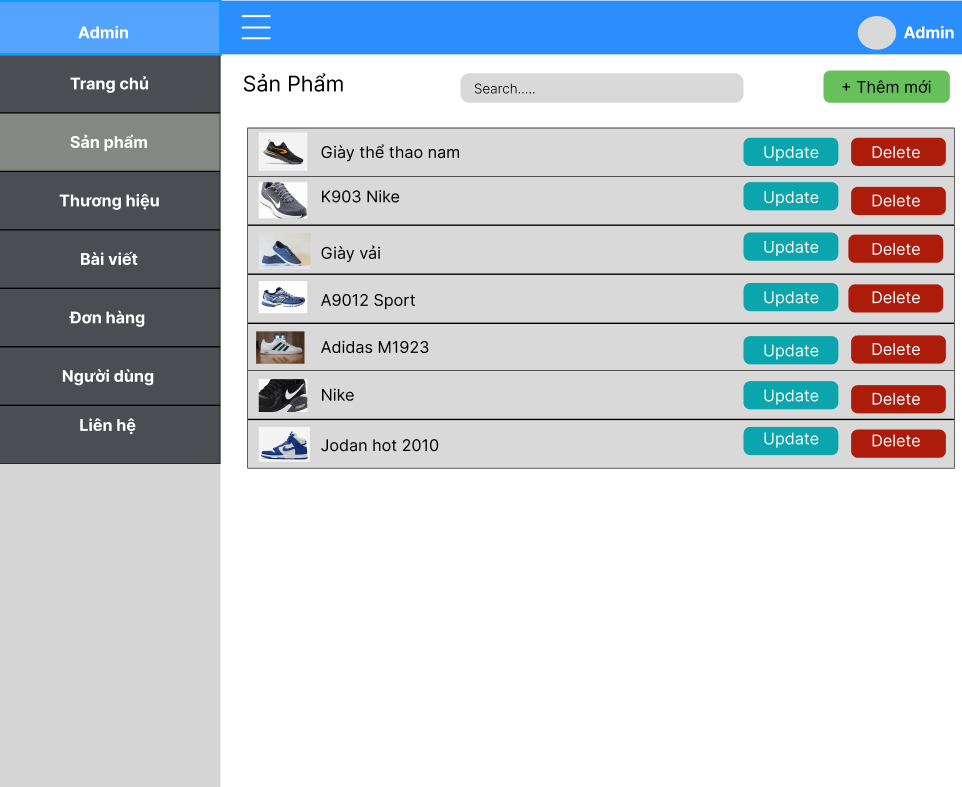
1. UC09: View Blog

****

****

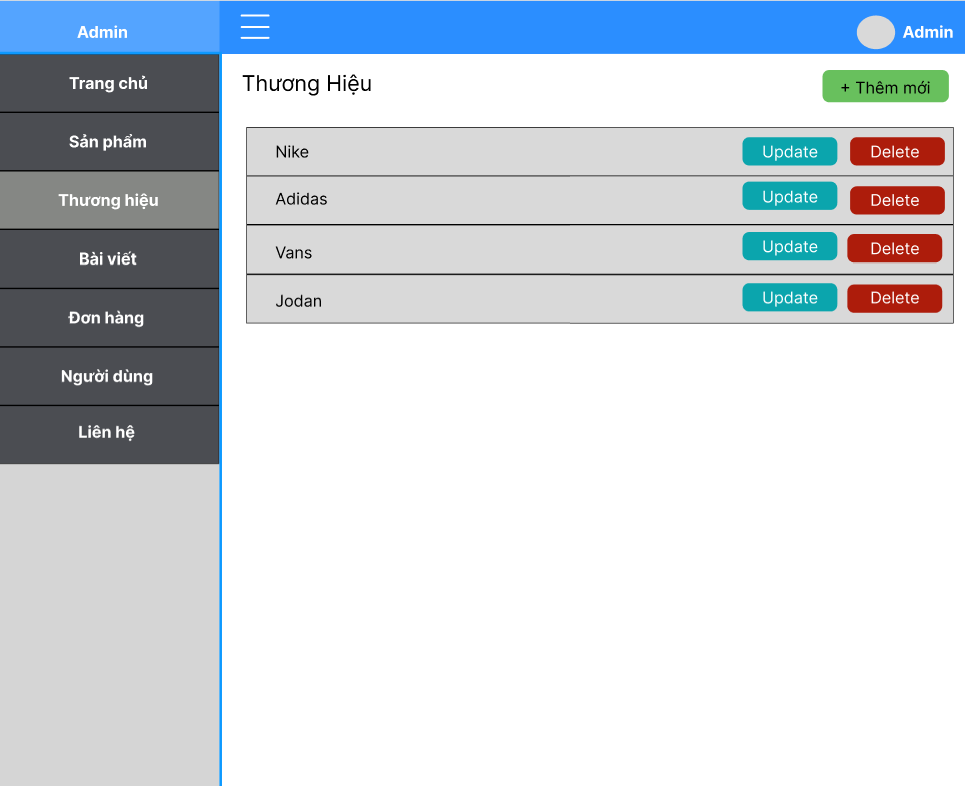
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC09: View Blog** | | | | | |
| Goal in Context | | | This use case allows the actor to view the list of all blogs.  They can select a blog from the list and view its details. | | |
| Pre-Condition | | | None | | |
| Post-Condition | | | The tool shall bring back data that is appropriate to the actor’s request | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | None. | | | |
| 1. | User | Click | | “Blog” in the Header bar | Website displays the blogs along with their summary information.  (see Figure 9a: View Blog List) |
| 2. | User | Select | | Any Blog | System returns and displays the detailed information of the selected article on the screen.  (see Figure 9b: View Blog Details) |

1. UC10: Management Products



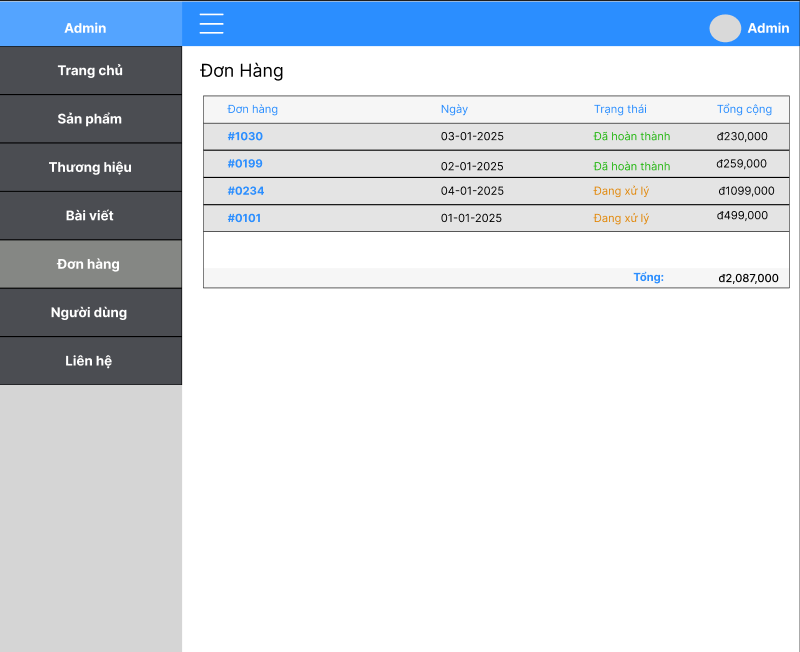
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC10: Management Product** | | | | | |
| Goal in Context | | | This use case allows the admin to view the list of all products. The admin can add a new product, update existing products, or delete products. | | |
| Pre-Condition | | | The admin must be successfully logged into the system. | | |
| Post-Condition | | | The system responds with data that matches the admin's requests. | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | The admin has successfully logged into the system. | | | |
| 1. | User | Select | | "Add New" Button on the Screen | The system displays the "Add New Product" screen. The admin enters the information and clicks the "Add Product" button to add a new product. |
| 2. | User | Click | | "Delete" Button | The product is removed from the website, and the product list is updated. |
| 3 | User | Select | | "Update" Button | The system displays the product update screen. The admin edits the information and clicks the "Update" button to save the changes. |

1. UC11: Management Brands



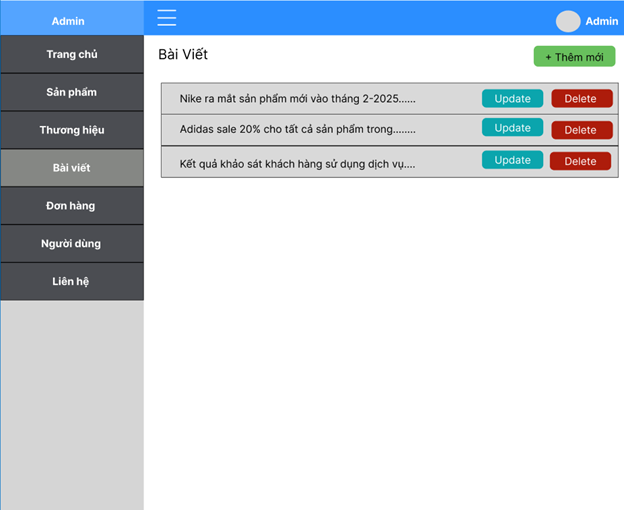
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC11: Management Brands** | | | | | |
| Goal in Context | | | This use case allows the admin to view the list of all brands. The admin can add a new brand, update an existing brand, or delete a brand. | | |
| Pre-Condition | | | The admin must be successfully logged into the system. | | |
| Post-Condition | | | The system responds with data that matches the admin's requests. | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | The admin has successfully logged into the system. | | | |
| 1. | User | Select | | "Add New" Button on the Screen | The system displays the "Add New Brand" screen. The admin enters the information and clicks the "Add Brand" button to add a new brand. |
| 2. | User | Click | | "Delete" Button | The brand is removed from the website, and the brand list is updated. |
| 3 | User | Select | | "Update" Button | The system displays the brand update screen. The admin edits the information and clicks the "Update" button to save the changes. |

1. UC12: Management Orders



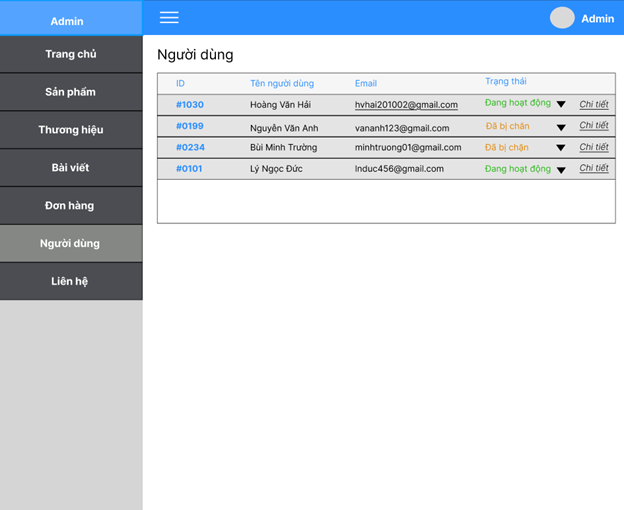
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC12: Management Orders** | | | | | |
| Goal in Context | | | This use case allows the admin to view the list of all orders. The admin can view the details of each order. | | |
| Pre-Condition | | | The admin must be successfully logged into the system. | | |
| Post-Condition | | | The system responds with data that matches the admin's requests. | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | The admin has successfully logged into the system. | | | |
| 1. | User | Click | | Any Order | The system displays the detailed information of that order. |
| 2. | User | Select | | “Status” combobox | The system updates status into database and displays status after update. |

1. UC13: Management Blogs



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC13: Management Blogs** | | | | | |
| Goal in Context | | | This use case allows the admin to view the list of all articles. The admin can add a new article, update an existing article, or delete an article. | | |
| Pre-Condition | | | The admin must be successfully logged into the system. | | |
| Post-Condition | | | The system responds with data that matches the admin's requests. | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | The admin has successfully logged into the system. | | | |
| 1. | User | Select | | "Add New" Button on the Screen | The system displays the "Add New Article" screen. The admin enters the information and clicks the "Add Article" button to add a new article. |
| 2. | User | Click | | "Delete" Button | The system displays the "Add New Article" screen. The admin enters the information and clicks the "Add Article" button to add a new article. |
| 3 | User | Select | | "Update" Button | The system displays the article update screen. The admin edits the information and clicks the "Update" button to save the changes. |

1. UC14: Management Users



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **UC14: Management Users** | | | | | |
| Goal in Context | | | This use case allows the admin to view the list of all users. The admin can update the status of users and view detailed user information. | | |
| Pre-Condition | | | The admin must be successfully logged into the system. | | |
| Post-Condition | | | The system responds with data that matches the admin's requests. | | |
|  | | | | | |
| **Step** | **Subject** (Actor) | **Verb** (Event) | | **Object** (Entity) | **Response** |
| **Main Success Scenario** | | | | | |
| Pre-Condition | | The admin has successfully logged into the system. | | | |
| 1. | User | Select | | Triangle Button on the Desired User | The system displays the status for the admin to select. |
| 2. | User | Click | | "Details" Button | The system displays the detailed information of the selected user. |

1. UC15: Authenticate User

As the first task of running ShoesStore Website Application, system checks for the existing of user account, which is used to log in to operation system, in Active Directory.

System displays a message box “You do not have permission to use this application.” in any case of following:

* Account does not exist in Active Directory
* Account does exist but it has not assigned to any role belonging to ShoesStore Website Application. Otherwise, system displays main form with menu items based on the role assigned to user account.

### ***Điều kiện hợp lệ cho các trường dữ liệu***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Field Name | Mandatory | Type | Condition | Error message |
| 1 | Full name | Y | Text | Max length is 128 |  |
| Value should not be empty | “Full name should not be empty.” |
| 2 | Avatar image | Y | File | Max size is 128KB |  |
| 3 | Phone number | Y | Text | Max length is 10 | “The phone number must not exceed 10 characters.” |
| Value should not be empty | “Phone number should not be empty.” |
| 4 | Address | Y | Text | Max length is 500 |  |
| Value should not be empty | “Address should not be empty.” |
| 5 | Quantity | Y | Number | Value > 0 | “Quantity can not be negative” |
| Value should not be empty | “Quantity should not be empty.” |
| 6 | Size | Y | Checkbox | Checkbox not empty | “Checkbox not empty” |
| 7 | Star Icon | Y | Checkbox | Checkbox not empty | “Checkbox not empty” |
| 8 | Your feedback | Y | Text | Max length is 500 |  |
| Value should not be empty | “Your feedback should not be empty.” |
| 9 | Fullname | Y | Text | Max length is 128 |  |
| Value should not be empty | “Fullname should not be empty.” |
| 10 | Phone number | Y | Text | Max length is 10 | “Phone not exceed 10 characters” |
| Value should not be empty | “Phone number should not be empty.” |
| 11 | Email | Y | Text | Include “@” and “.com” | “Email must include “@” and “.com”. Please input again!“ |
| Value should not be empty | “Email should not be empty.” |
| 12 | Address ship | Y | Text | Max length is 500 |  |
| Value should not be empty | “Address ship should not be empty.” |
| 13 | Place order note | N | Text | Max length is 500 |  |
| 14 | Payment method | Y | Checkbox | Checkbox not empty | “Checkbox not empty” |

## **Yêu cầu phi chức năng**

### ***2.5.1 Tính sử dụng***

1. Look and feel

Thiết kế giao diện người dùng (UI) sẽ tuân theo một bố cục mẫu với các nguyên tắc sau:

* Hệ thống phải tuân theo giao diện và trải nghiệm của các trang thương mại điện tử hiện đại, đảm bảo giao diện **sạch sẽ**, **hấp dẫn về mặt thẩm mỹ**, và **thân thiện với người dùng**.
* Các yếu tố thiết kế như **bảng màu**, **phông chữ**, và **kiểu dáng nút bấm** phải được sử dụng nhất quán trên toàn bộ trang web. Điều này giúp nâng cao trải nghiệm người dùng và củng cố nhận diện thương hiệu.

1. Screen Styles

Kiểu dáng giao diện màn hình mô tả cách bố cục và cấu trúc của giao diện người dùng trên các trang khác nhau của website. Các yêu cầu cụ thể bao gồm:

* Kiểu dáng nên nhất quán giữa các trang (product pages, checkout, shopping cart, etc.).
* Thiết kế phải thích ứng, linh hoạt với nhiều kích thước màn hình khác nhau, từ điện thoại di động đến máy tính để bàn.

### ***2.5.2 Độ tin cậy***

Màn hình của người dùng cuối phải hiển thị thông báo lỗi thân thiện với người dùng (không được hiển thị stack dump).

### ***2.5.3 Xử lý lỗi***

Website nên hiển thị rõ ràng, thông báo lỗi dễ hiểu trong trường hợp xảy ra sự cố (ví dụ: nhập dữ liệu không hợp lệ, hết sản phẩm). Các lỗi kỹ thuật (ví dụ: stack dumps) không được phép hiển thị cho người dùng cuối. Một số ví dụ bao gồm:

* “Vui lòng kiểm tra thông tin thanh toán của bạn và thử lại.”
* “Sản phẩm này hiện tại đã hết hàng.”

### ***2.5.4 Tính khả dụng***

* **Thời gian hoạt động:** Website nên duy trì thời gian gian hoạt động ít nhất là 99.9% mỗi tháng, đảm bảo rằng hệ thống này gần như luôn sẵn sàng cho người dùng truy cập.
* **Khôi phục sau thảm họa:**Hệ thống phải có các quy trình sao lưu và khôi phục, với mục tiêu khôi phục chức năng trong vòng 30 phút trong trường hợp xảy ra sự cố nghiêm trọng.

### ***2.5.5 Các phụ thuộc của hệ thống***

**Database Back-end:** Hệ thống sẽ sử dụng MS SQL Server 2022 cho lưu trữ thông tin chi tiết sẩn phẩm, dữ liệu khách hàng, và thông tin đơn hàng.

### ***2.5.6 Bugs and Defect Rates***

Không có yêu cầu cụ thể nào được nêu ra. Các mục tiêu về tỷ lệ lỗi có thể được lấy từ các tiêu chuẩn và quy định của hệ thống quản lý chất lượng HVN SDC.

Mức độ nghiêm trọng của lỗi sẽ được định nghĩa như sau:

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Mức độ | Giá trị trọng số | Mô tả |
| 1 | Nghiêm trọng | 10 | Một chức năng quan trọng hoặc thành phần cốt lõi của hệ thống bị hỏng hoàn toàn, không có cách khắc phục nào khả thi. Lỗi này khiến quá trình kiểm thử hoặc vận hành không thể tiếp tục. |
| 2 | Cao | 5 | Một chức năng hoặc thành phần quan trọng của hệ thống không hoạt động đúng cách, bị lỗi nghiêm trọng hoặc được tài liệu hóa sai lệch. Lỗi này ảnh hưởng trực tiếp đến hoạt động chính của hệ thống. |
| 3 | Trung bình | 3 | Các vấn đề chức năng nhỏ, chẳng hạn như tính năng hoạt động sai hoặc được tài liệu hóa không chính xác nhưng không ảnh hưởng nghiêm trọng đến hệ thống tổng thể. |
| 4 | Thẩm mỹ | 1 | Các vấn đề thẩm mỹ hoặc hiển thị, như lỗi về hình ảnh hoặc giao diện, không ảnh hưởng đến chức năng hoặc hiệu suất cốt lõi của hệ thống. |

**Tỷ lệ lỗi** được đo lường dựa trên công thức: **Tổng lỗi có trọng số / Điểm use case** (viết tắt: **Wdef / UCP**). Một điểm use case tương đương với 20 ngày công.

Theo tiêu chuẩn của SDC, **tỷ lệ lỗi chuẩn** là **5.5 Wdef / UCP**.

### ***2.5.7 Hiệu năng***

1. Thời gian phản hồi

* **Thời gian tải trang:** Website phải tải các trang chính (ví dụ: trang chủ, trang sản phẩm, trang thanh toán) trong vòng 3 giây.
* **Tương tác người dùng:** Các hành động như: thêm sản phẩm vào giỏ hàng hoặc cập nhật số lượng phải phản hồi trong thời gian dưới 1 giây.

1. Thông lượng

**Người dùng đồng thời: website phải có khả năng xử lý ít nhất 500 người dùng đông thời duyệt web và thực hiện mua hàng mà không xảy ra chậm chễ hoặc gián đoạn.**

1. Khối lượng dữ liệu

* Xử lý dữ liệu: Website phải có khả năng quản lý dữ liệu sản phẩm ít nhất 5 năm, bao gồm: ảnh, giá và mức độ tồn kho. Ngoài ra, hệ thống phải hỗ trợ ít nhất **10,000 sản phẩm** và tối đa **100,000 hồ sơ khách hàng.**
* Người dùng: Hệ thống phải được thiết kế để hỗ trợ **5 đến 10 quản trị viên hoạt động đồng thời** và hàng ngàn khách hàng tiềm năng truy cập cùng lúc.

### ***2.5.8 Khả năng hỗ trợ***

1. Khả năng bảo trì

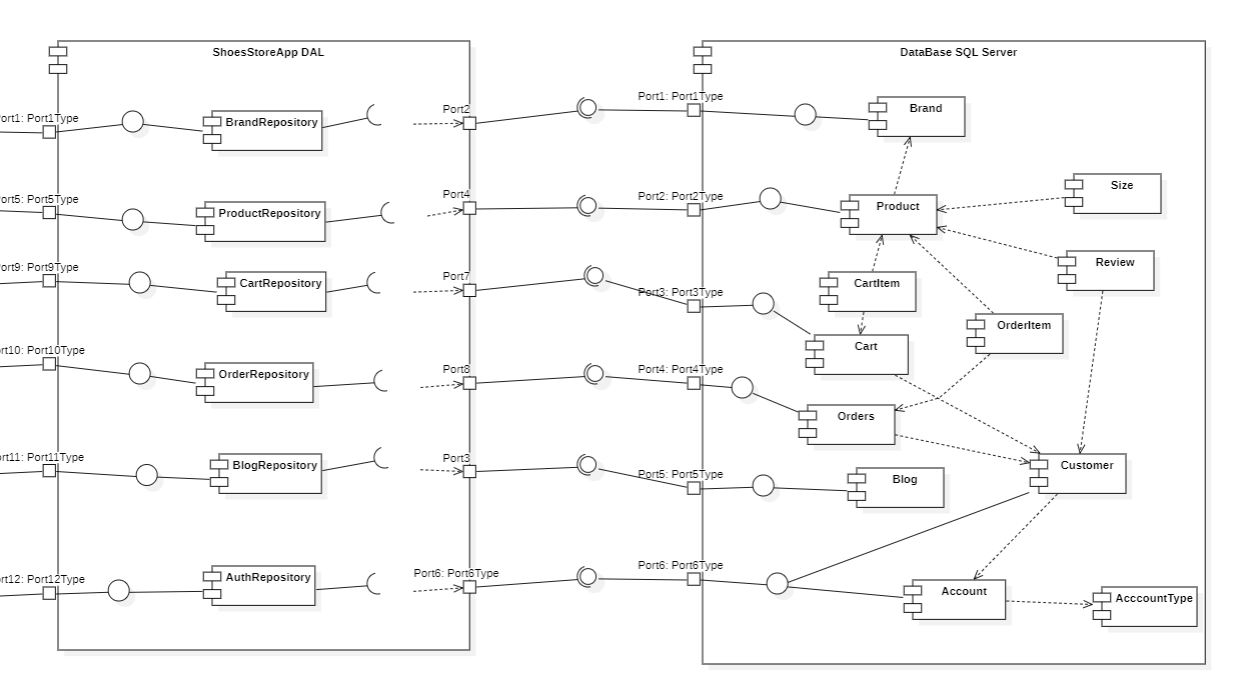
* **Dễ dàng cập nhật:** Tất cả các giá trị được định nghĩa trước, chẳng hạn như danh mục sản phẩm, kích cỡ, và các giá trị trong danh sách thả xuống, phải dễ dàng bảo trì. Quản trị viên phải có thể thêm, cập nhật, hoặc xóa sản phẩm và các giá trị này nhanh chóng thông qua hệ thống quản lý nội dung (CMS).
* **Hệ thống quản lý nội dung:** Hệ thống CMS (Content Management System) phía backend phải thân thiện với người dùng, cho phép nhân viên cập nhật danh sách sản phẩm, chương trình khuyến mãi, và tồn kho mà không cần kiến thức kỹ thuật.

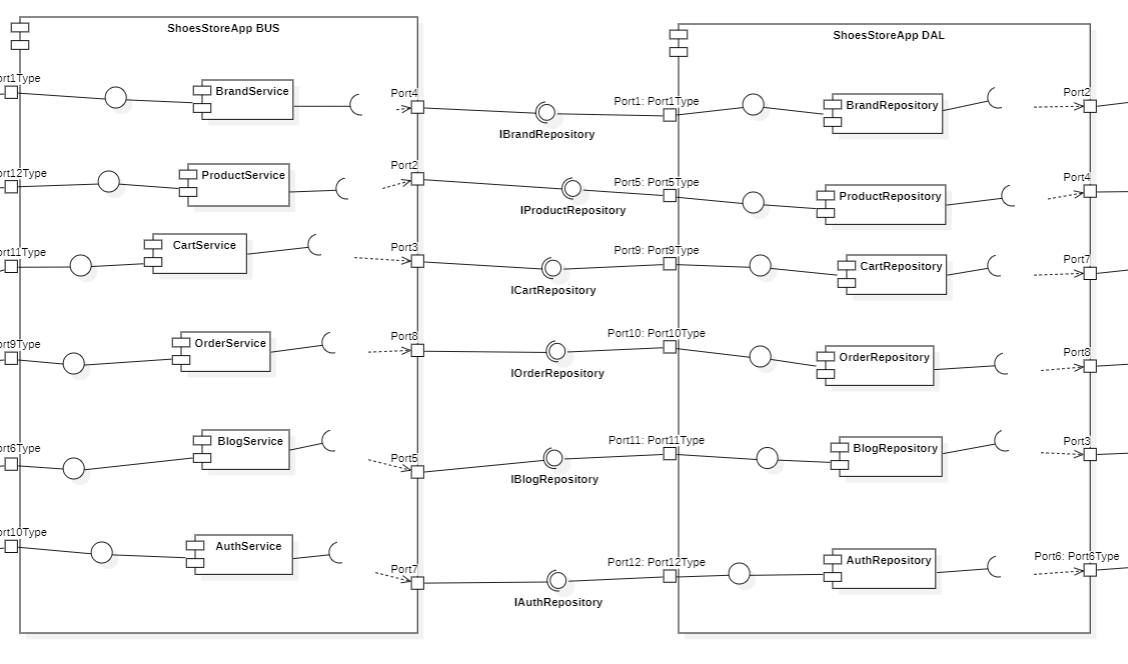
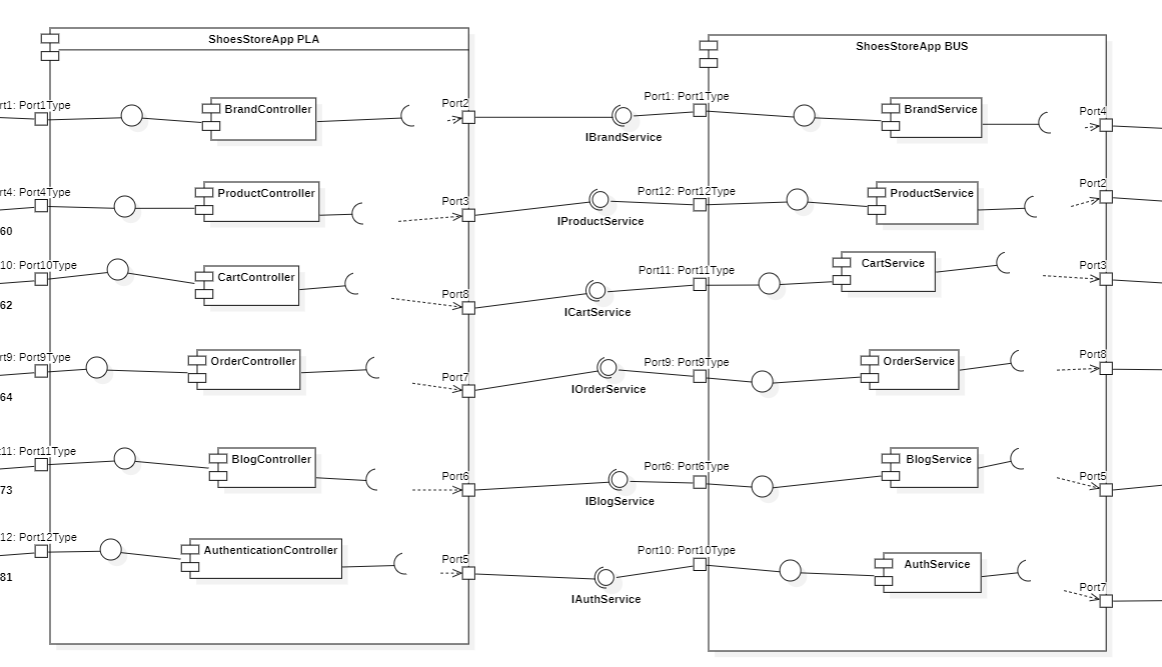
1. Nhật ký lỗi

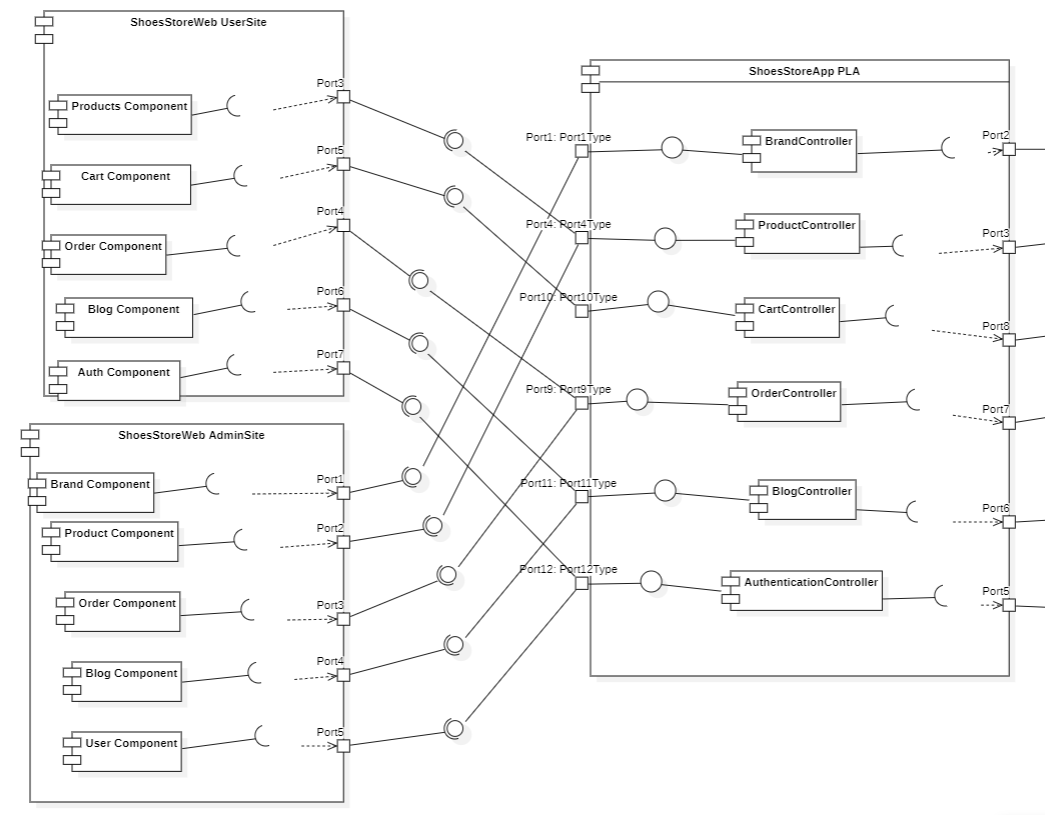
* **Ghi nhật ký:** Tất cả các lỗi và sự kiện hệ thống phải bao gồm định dạng thời gian chuẩn (ví dụ: **YYYY-MM-DD HH:MM:SS**) để dễ dàng theo dõi và xử lý lỗi.
* **Nhật ký bảo mật:** Các sự kiện liên quan đến bảo mật, chẳng hạn như các lần đăng nhập, nỗ lực truy cập thất bại, và lỗi thanh toán, cũng phải được ghi lại kèm theo mức độ nghiêm trọng và thông tin về thời gian xảy ra.

# Chương 3 – Thiết kế kiến trúc và tổ chức CODE

## **3.1 Mô tả tổng quan các thành phần trong dự án:**



**** ****

****

## **3.2** **Xác định các quy định cần tuân thủ khi viết code:**

Trong dự án, để mã nguồn có sự đồng nhất về cú pháp, cách đặt tên biến hàm… chúng ta cần đặt một số quy tắc sau đây để mã nguồn có thể chính xác, dễ đọc dễ bảo trì và phát triển.

* Quy tắc đặt tên
* Naming convention tuân thủ quy tắc Pascal Case. Ví dụ: GetUser.
* Đặt tên biến, lớp và hàm có nghĩa. Ví dụ: GetUser thay vì gu.

Cấu trúc mã nguồn

* **Chia nhỏ mã nguồn:** Sử dụng các hàm và lớp để chia nhỏ công việc, giữ cho mã nguồn gọn gàng và dễ hiểu.
* **Sắp xếp mã:** Theo dõi cấu trúc mã, để các phần liên quan gần nhau.
* Chú thích
* **Ghi chú:** Sử dụng chú thích để giải thích logic phức tạp hoặc các quyết định thiết kế.
* **Chú thích rõ ràng:** Tránh sử dụng chú thích không cần thiết hoặc quá nhiều.
* Quản lý lỗi
* **Xử lý ngoại lệ:** Sử dụng cơ chế xử lý ngoại lệ để xử lý các lỗi có thể xảy ra.
* **Ghi log:** Ghi lại thông tin lỗi để dễ dàng theo dõi và khắc phục.
* Tính hiệu suất
* **Tối ưu hóa:** Tối ưu hóa mã khi cần thiết, nhưng không hy sinh tính rõ ràng.
* **Phân tích hiệu suất:** Sử dụng các công cụ phân tích để theo dõi hiệu suất của mã.
* Quản lý phiên bản
* **Sử dụng hệ thống quản lý mã nguồn:** Sử dụng Git hoặc các hệ thống tương tự để quản lý và theo dõi các thay đổi trong mã nguồn.
* **Viết thông điệp commit rõ ràng:** Giải thích lý do thay đổi trong thông điệp commit.

## **3.3 Trình bày cách tổ chức CODE trong dự án:**

Khi phát triển một dự án, tổ chức code một cách rõ ràng và khoa học là rất quan trọng để đảm bảo khả năng mở rộng, bảo trì và dễ hiểu cho nhóm phát triển:

* Cấu trúc thư mục dự án

Một cấu trúc thư mục chuẩn giúp bạn phân chia trách nhiệm của các thành phần trong ứng dụng.

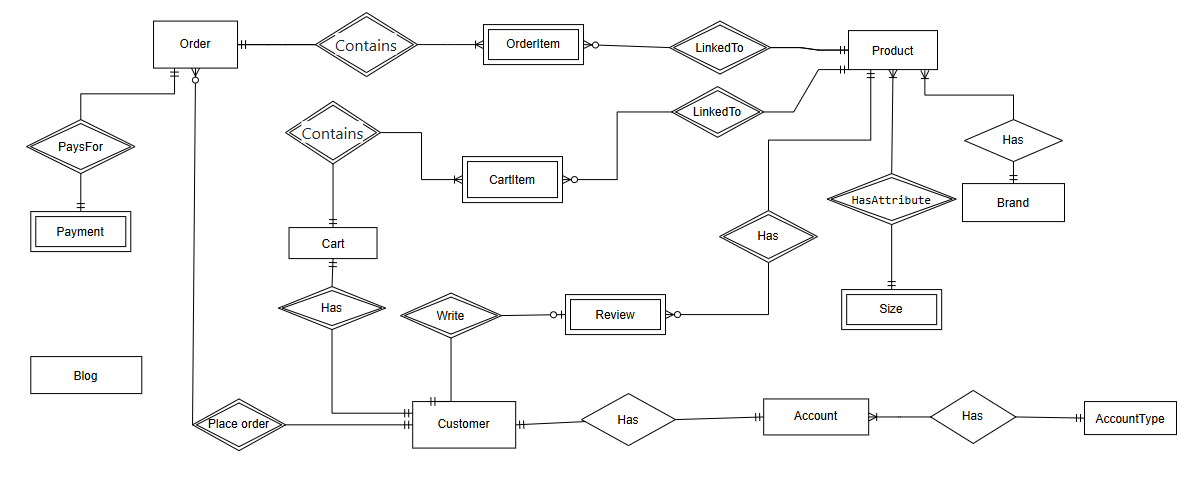
* Phân tầng theo mô hình
* Tất cả các thành phần liên quan đến giao diện (UI), bao gồm Widgets, Components hoặc ViewModels.
* Bao gồm các Entities
* Bao gồm các lớp Models, Repositories, và Data Sources
* Comment và tài liệu hóa
* Dùng doc comments (///) để giải thích lớp, phương thức. - Thêm các TODO comments để đánh dấu các phần cần cải thiện sau.

## **3.4. Liệt kê các thư viện thirdparty sử dụng trong dự án:**

* Microsoft.AspNetCore.Authentication.JwtBearer
* Microsoft.AspNetCore.Identity.EntityFramewor
* Asp.Versioning.Mvc
* Asp.Versioning.Mvc.ApiExplorer
* Microsoft.EntityFrameworkCore.SqlServer
* Microsoft.EntityFrameworkCore.Tools
* Newtonsoft.Json
* Bootstrap
* Font Awesome
* angular2-jwt

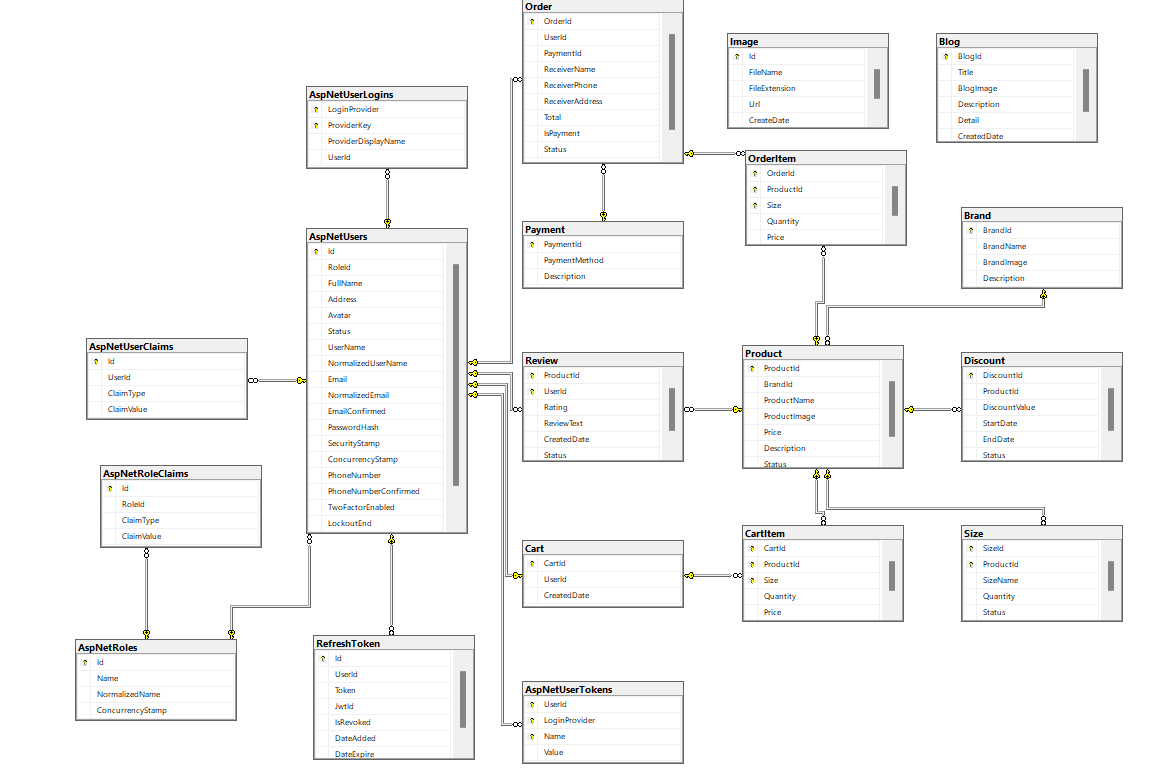
# Chương 4 – Thiết kế cơ sở dữ liệu

## **4.1. Mô hình thực thể liên kết (ERD):**



## **4.2 Lược đồ cơ sở dữ liệu quan hệ:**

### ***4.2.1 Mô hình cơ sở dữ liệu quan hệ***



### ***4.2.2 Mô tả cấu trúc bảng***

- Bảng loại tài khoản

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | AccountTypeId | Int | Khóa chính | Mã loại tài khoản |
| 2 | TypeName | Nvarchar(100) |  | Tên loại tài khoản |

* Bảng tài khoản

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | AccountId | Int | Khóa chính | Mã tài khoản |
| 2 | AccountTypeId | Int | Khóa ngoại | Mã loại tài khoản |
| 3 | Username | char(50) | Không trùng | Tên đăng nhập |
| 4 | Password | char(50) |  | Mật khẩu |
| 5 | Avatar | Varchar(max) |  | Ảnh đại diện |

* Bảng khách hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | CustomerId | Int | Khóa chính | Mã loại tài khoản |
| 2 | AccountId | Int | Khóa ngoại | Tên loại tài khoản |
| 3 | Fullname | nvarchar(255) |  | Họ tên |
| 4 | Email | Varchar(255) |  | Email |
| 5 | PhoneNumber | char(10) |  | Số điện thoại |
| 6 | Address | Nvarchar(max) |  | Địa chỉ |

* Bảng thương hiệu

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | BrandId | Int | Khóa chính | Mã thương hiệu |
| 2 | BrandName | nvarchar(255) | Không trùng | Tên thương hiệu |
| 3 | BrandImage | varchar(max) |  | ảnh thương hiệu |
| 4 | Description | nvarchar(max) |  | Mô tả |

* Bảng sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | ProductId | Int | Khóa chính | Mã sản phẩm |
| 2 | BrandId | Int |  | Mã thương hiệu |
| 3 | ProductName | nvarchar(255) | Không trùng | Tên sản phẩm |
| 4 | ProductImage | Varchar(max) |  | Ảnh sản phẩm |
| 5 | price | Decimal(10,2) |  | Giá |
| 6 | description | nvarchar(max) |  | Mô tả |
| 7 | status | Varchar(50) |  | Trạng thái |

* Bảng size

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | sizeID | Int | Khóa chính | Mã size |
| 2 | productID | Int | Khóa ngoại, Khóa chính | Mã sản phẩm |
| 3 | Size | Varchar(20) |  | Size |
| 4 | Quantity | Int |  | Số lượng |

* Bảng đánh giá

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | ProductId | Int | Khóa ngoại, Khóa chính | Mã sản phẩm |
| 2 | CustomerId | Int | Khóa ngoại, Khóa chính | Mã khách hàng |
| 3 | rating | Int |  | Số sao |
| 4 | ReviewText | nvarchar(max) |  | Văn bản đánh giá |
| 5 | CreatedDate | Datetime | Mặc định ngày thêm | Ngày tạo |
| 6 | Status | Varchar(50) |  | Trạng thái |

* Bảng bài viết

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | BlogId | Int | Khóa chính | Mã bài viết |
| 2 | Title | nvarchar(255) |  | Tiêu đề |
| 3 | BlogImage | varchar(max) |  | ảnh bài viết |
| 4 | Description | nvarchar(300) |  | Mô tả |
| 5 | Detail | nvarchar(max) |  | Chi tiết |
| 6 | CreatedDate | Datetime | Mặc định ngày thêm | Ngày tạo |

* Bảng đơn hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | OrderId | Int | Khóa chính | Mã đơn hàng |
| 2 | CustomerId | Int | Khóa ngoại | Mã khách hàng |
| 3 | ReceiverName | nvarchar(150) |  | Tên người nhận |
| 4 | ReceiverPhone | char(10) |  | Số điện thoại người nhận |
| 5 | ReceiverAddress | nvarchar(max) |  | Địa chỉ người nhận |
| 6 | CreateDate | Datetime |  | Ngày tạo |
| 7 | Total | Decimal(18,2) |  | Tổng tiền |
| 8 | Status | nvarchar(50) |  | Trạng thái |

* Bảng chi tiết đơn hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | OrderId | Int | Khóa ngoại, Khóa chính | Mã đơn hàng |
| 2 | ProductId | Int | Khóa ngoại, Khóa chính | Mã sản phẩm |
| 3 | Quantity | Int |  | Số lượng |
| 4 | Price | Decimal(10,2) |  | Giá |

* Bảng giỏ hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | CartId | Int | Khóa chính | Mã giỏ hàng |
| 2 | CustomerId | Int | Khóa ngoại | Mã khách hàng |
| 3 | CreatedDate | Datetime |  | Ngày tạo |

* Bảng chi tiết giỏ hàng

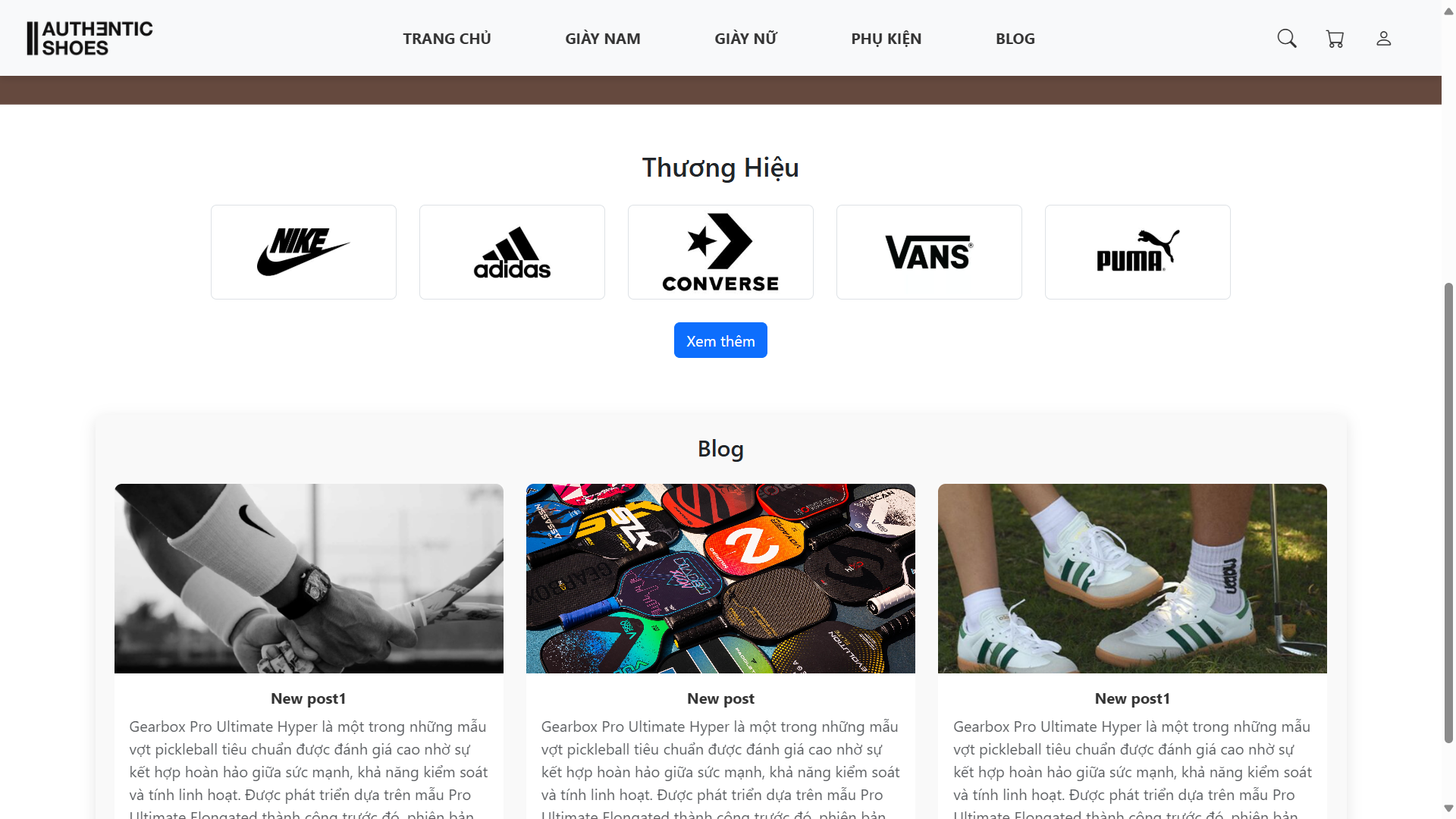
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | CartId | Int | Khóa ngoại, Khóa chính | Mã giỏ hàng |
| 2 | ProductId | Int | Khóa ngoại, Khóa chính | Mã sản phẩm |
| 3 | Quantity | Int |  | Số lượng |
| 4 | Price | Decimal(10,2) |  | Giá |

* Bảng thanh toán

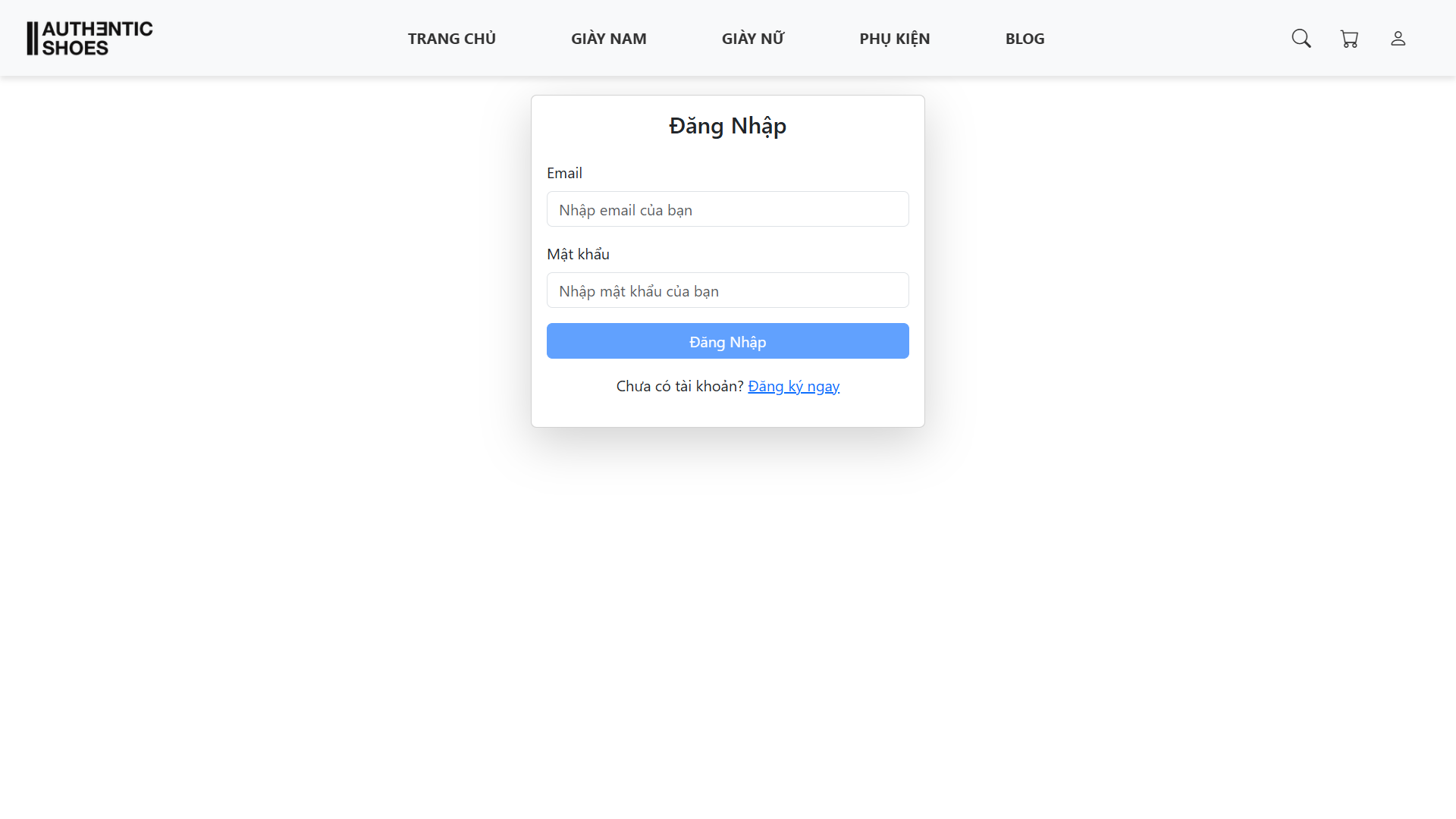
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên Trường** | **Kiểu dữ liệu** | **Ràng buộc** | **Mô tả** |
| 1 | orderID | Int | Khóa ngoại, Khóa chính | Mã đơn hàng |
| 2 | paymentMethod | nvarchar(255) |  | Phương thức thanh toán |
| 3 | Status | nvarchar(150) |  | Trạng thái |

# Chương 5 – Thiết kế giao diện

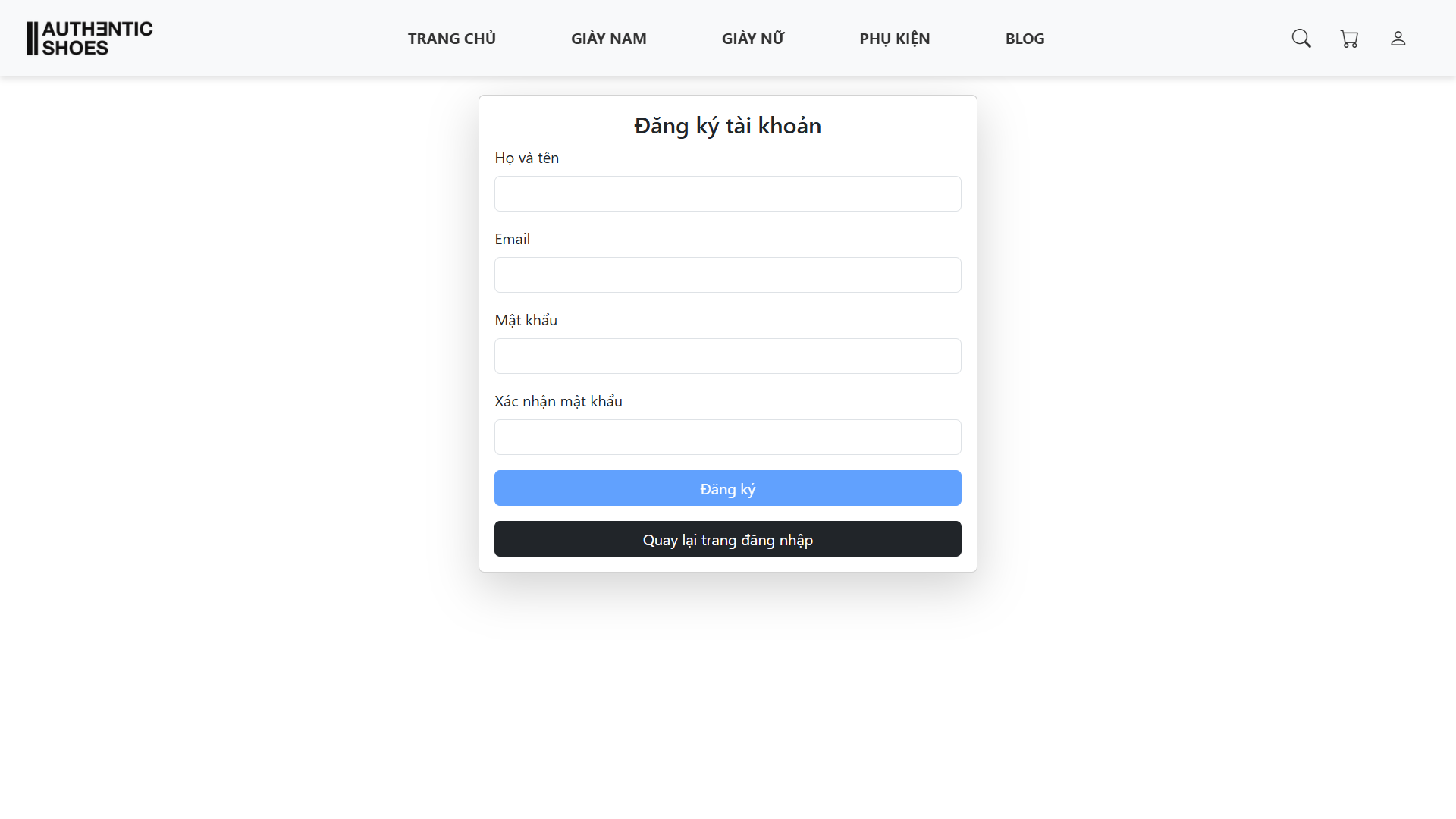
## **5.1 Giao diện chức năng 1: Giao diện trang chủ**

**

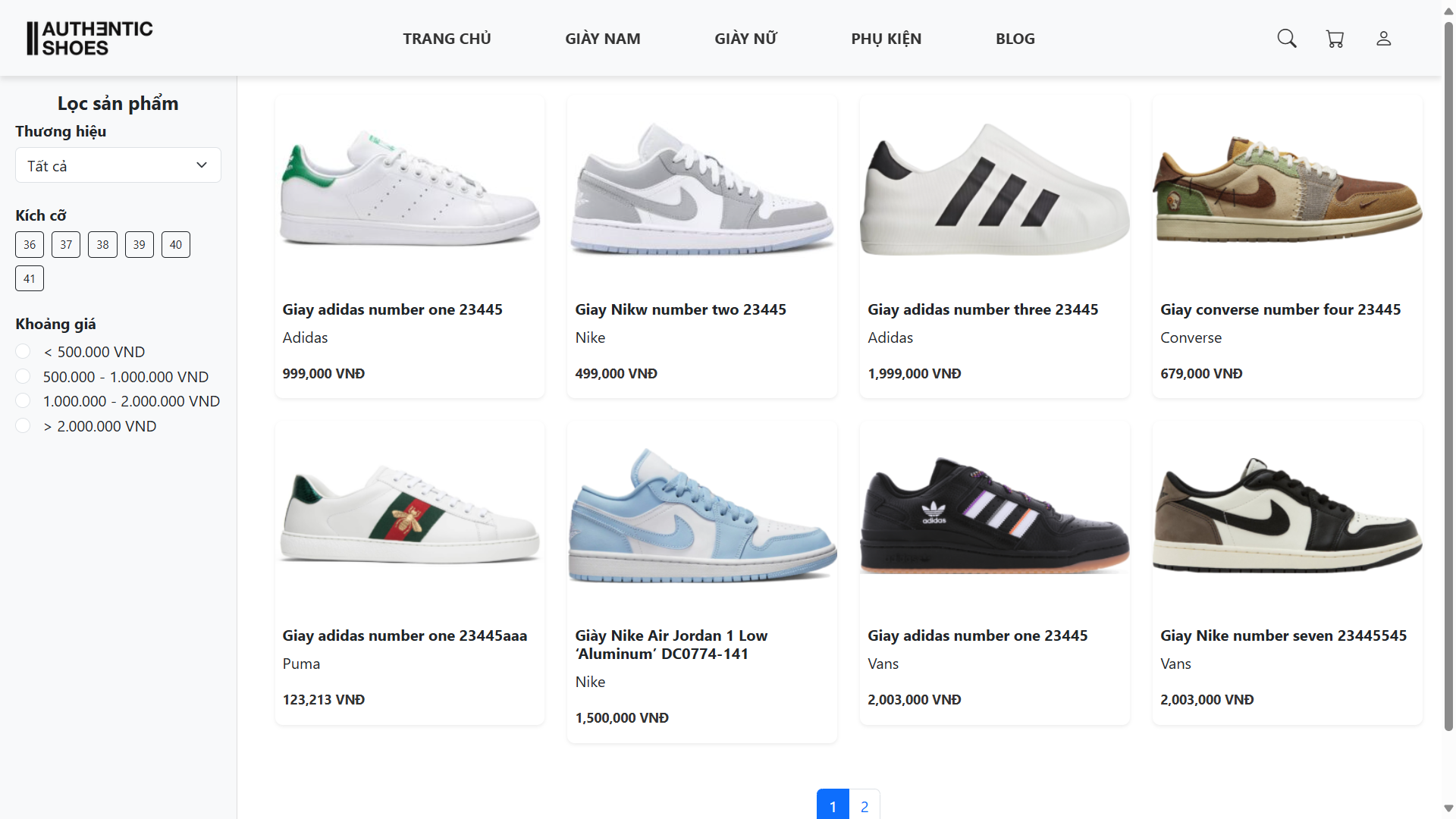
## **5.2 Giao diện chức năng 2: Giao diện trang đăng nhập**

**

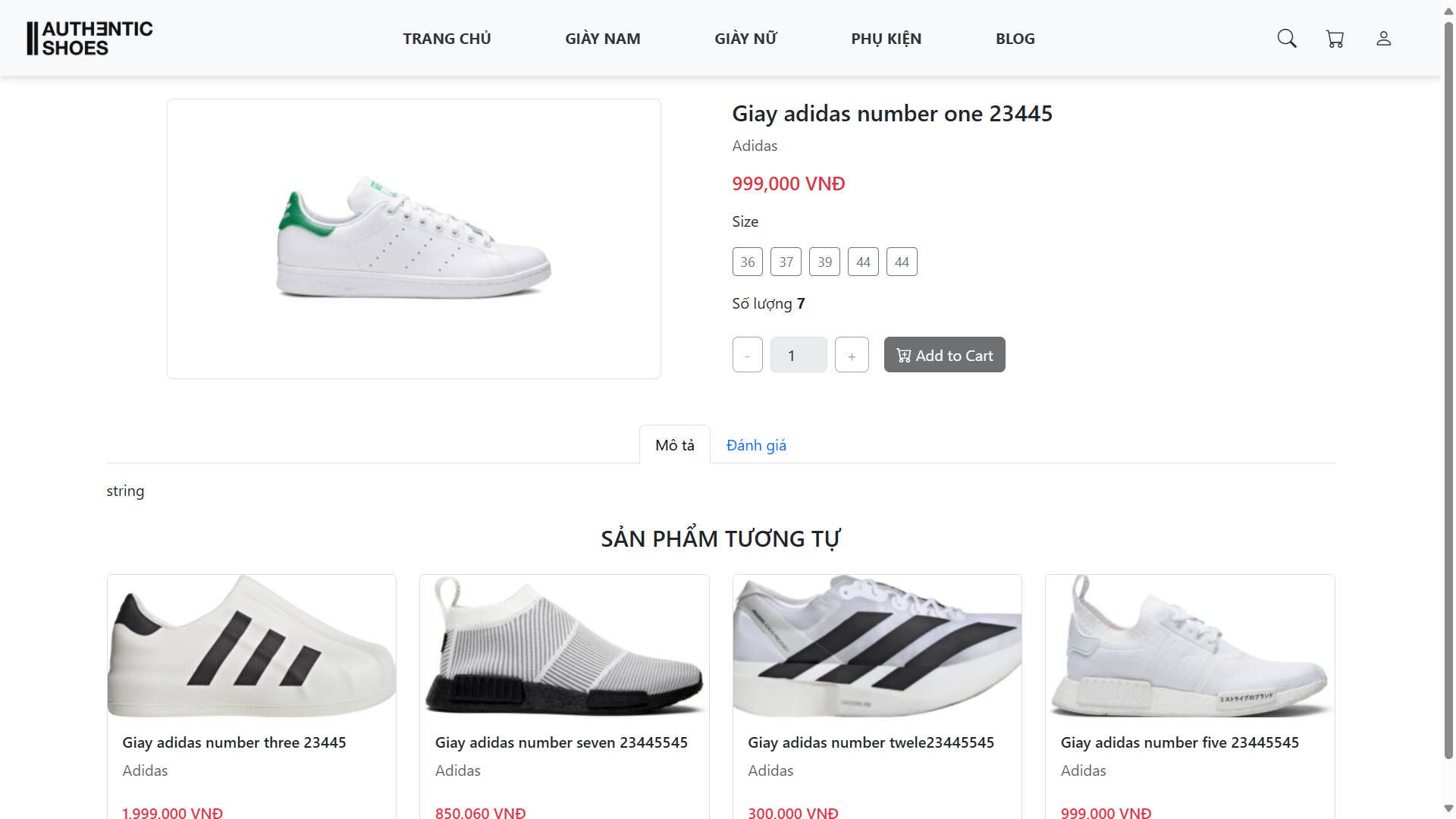
## **5.3 Giao diện chức năng 3: Giao diện trang đăng ký**

**

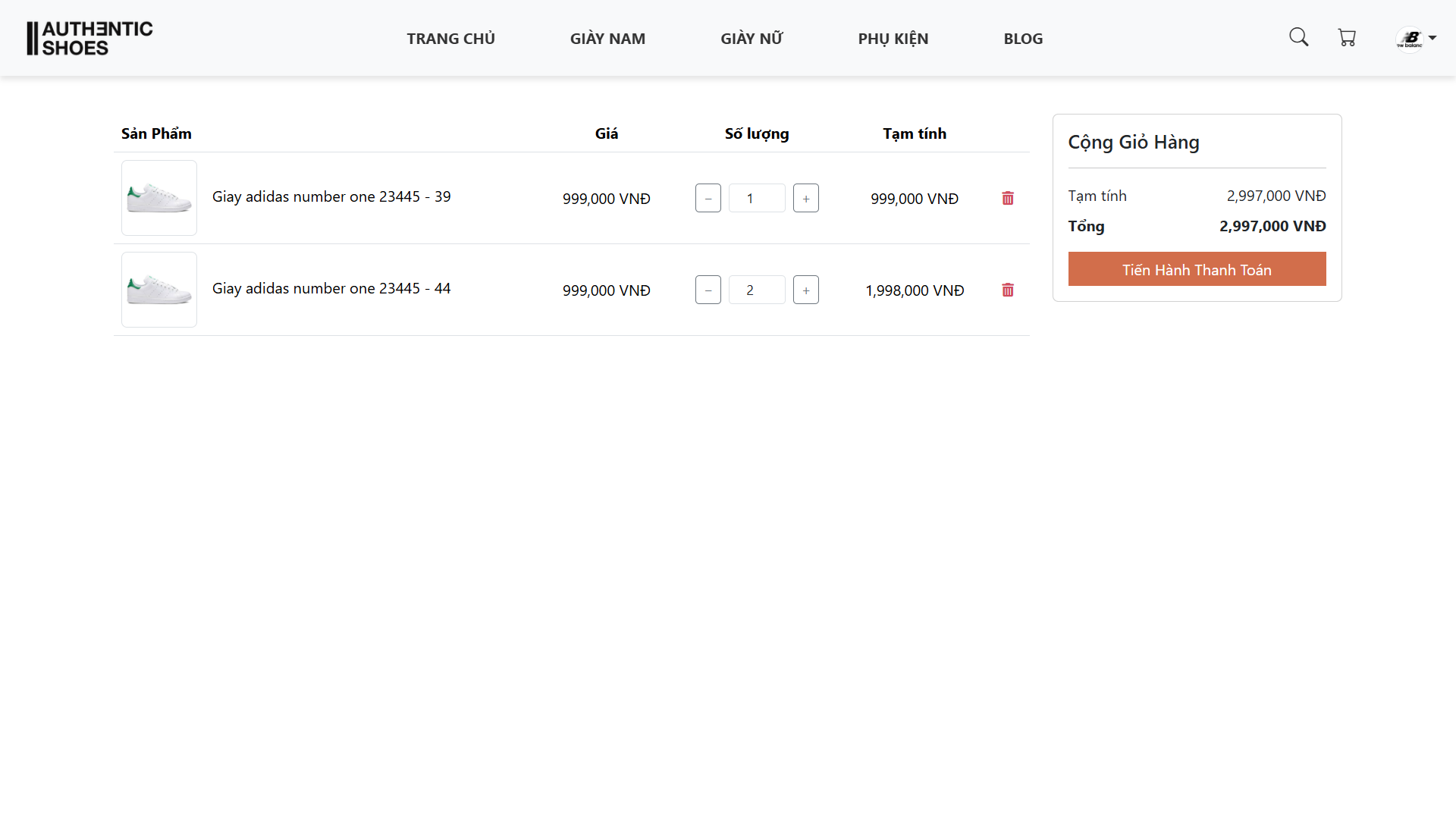
## **5.4 Giao diện chức năng 4: Giao diện trang danh sách sản phẩm**

**

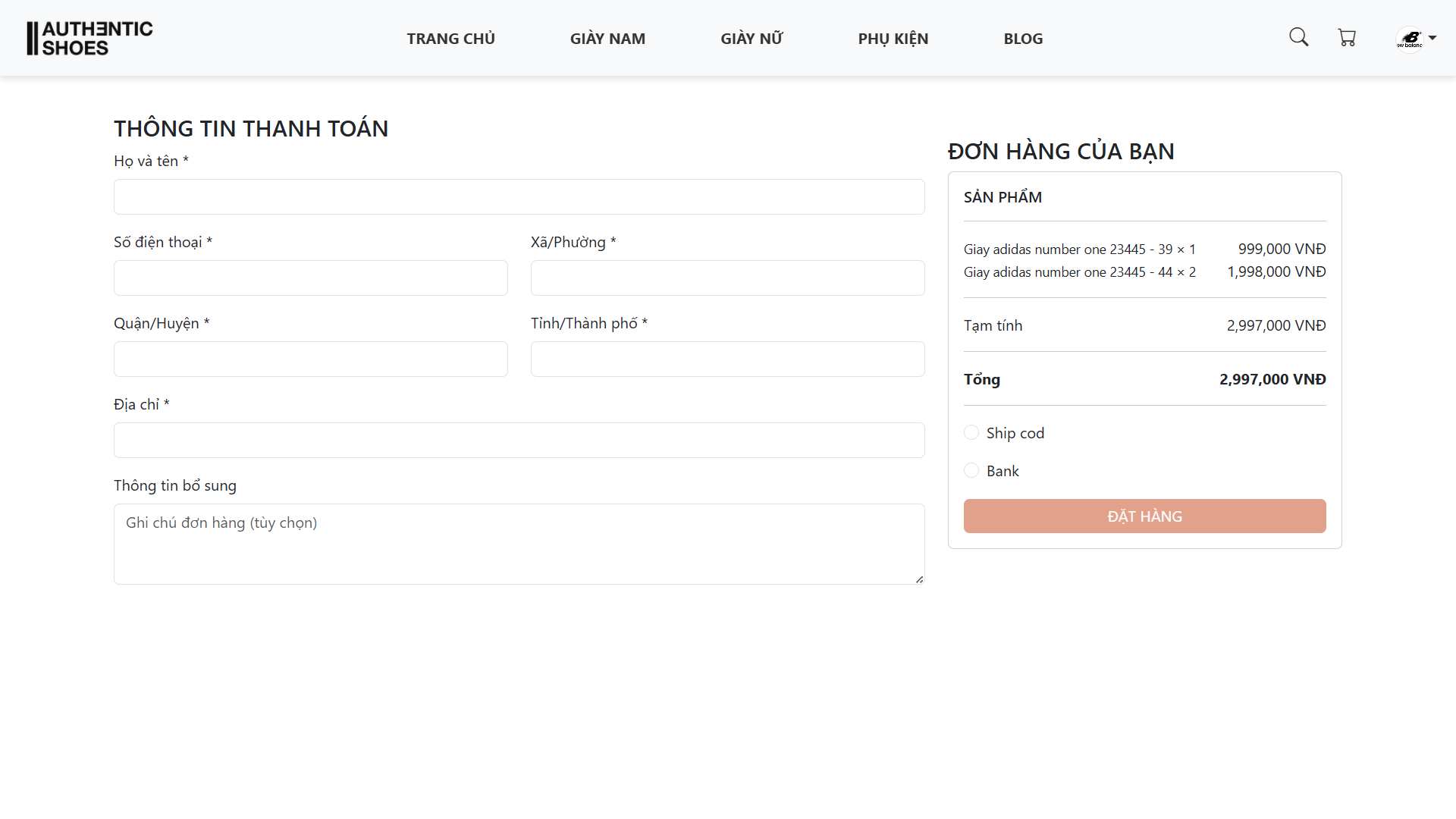
## **5.5 Giao diện chức năng 5: Giao diện trang chi tiết sản phẩm**

**

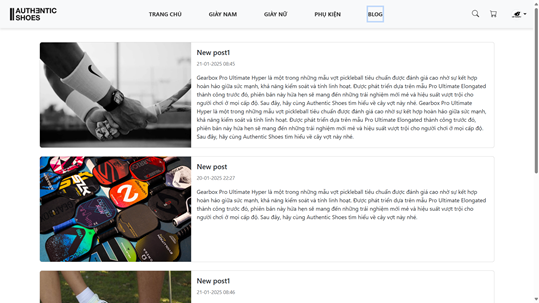
## **5.6 Giao diện chức năng 6: Giao diện trang giỏ hàng**



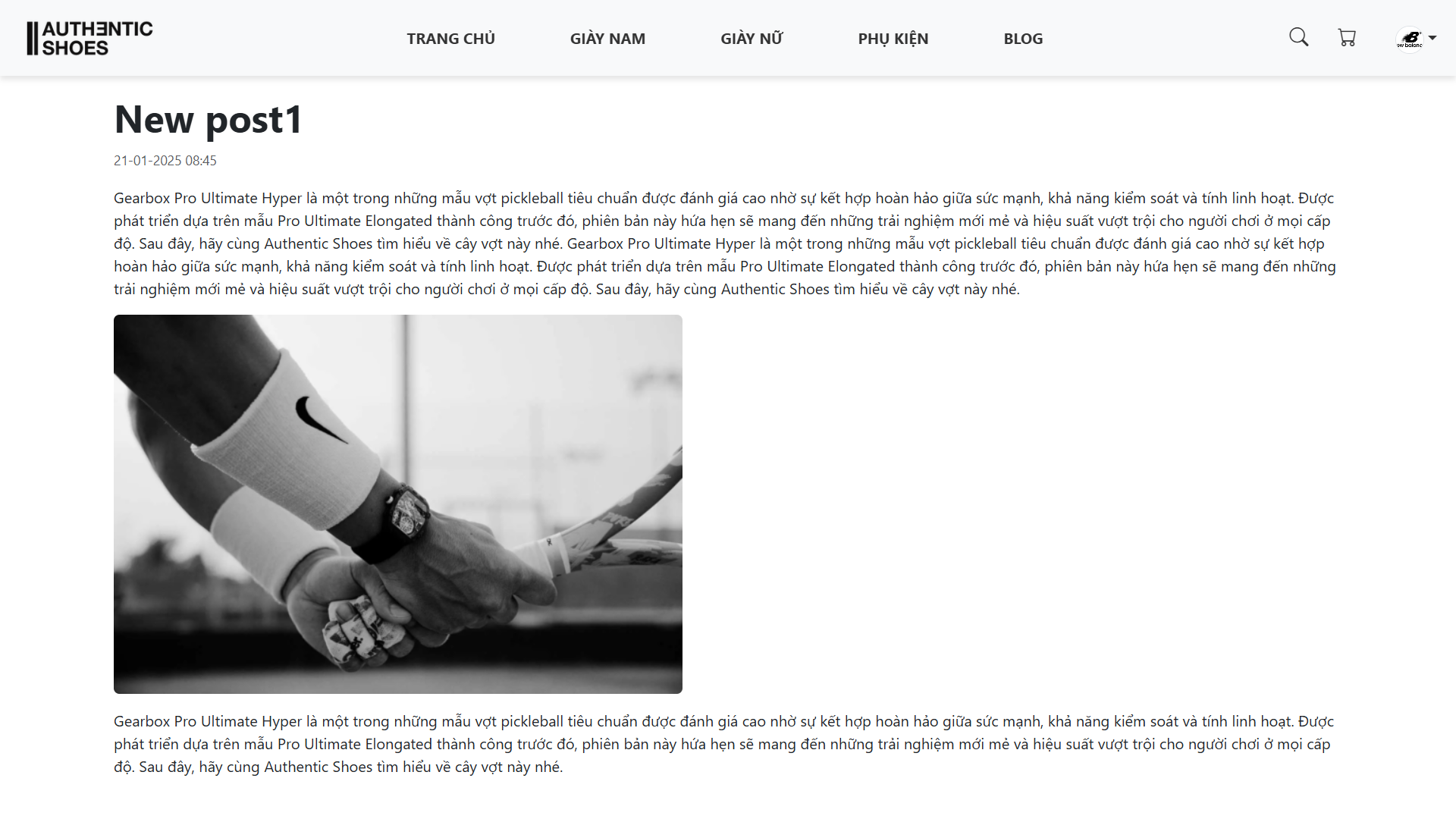
## **5.7 Giao diện chức năng 7: Giao diện trang đặt hàng**

****

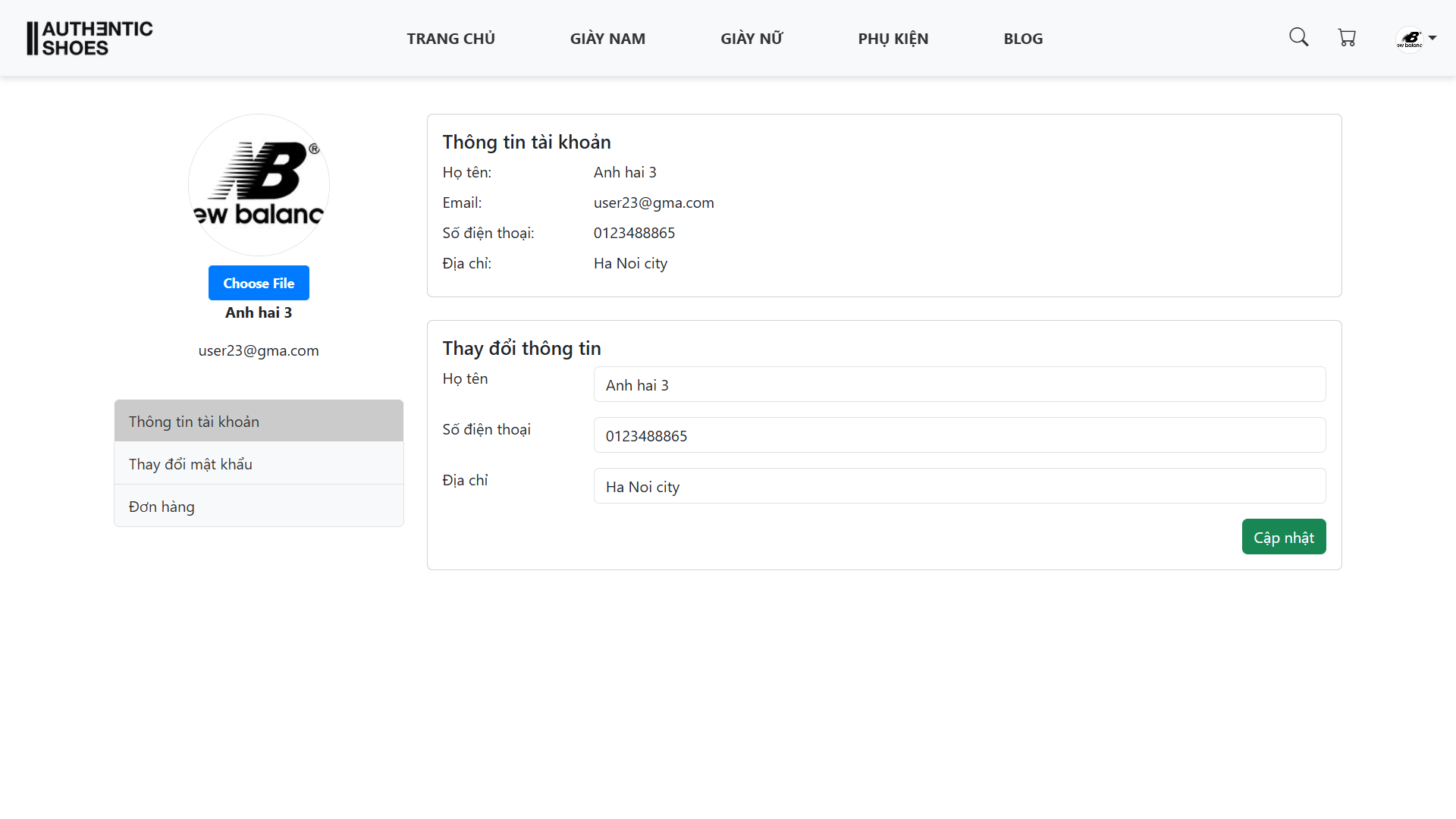
## **5.8 Giao diện chức năng 8: Giao diện trang danh sách blog**



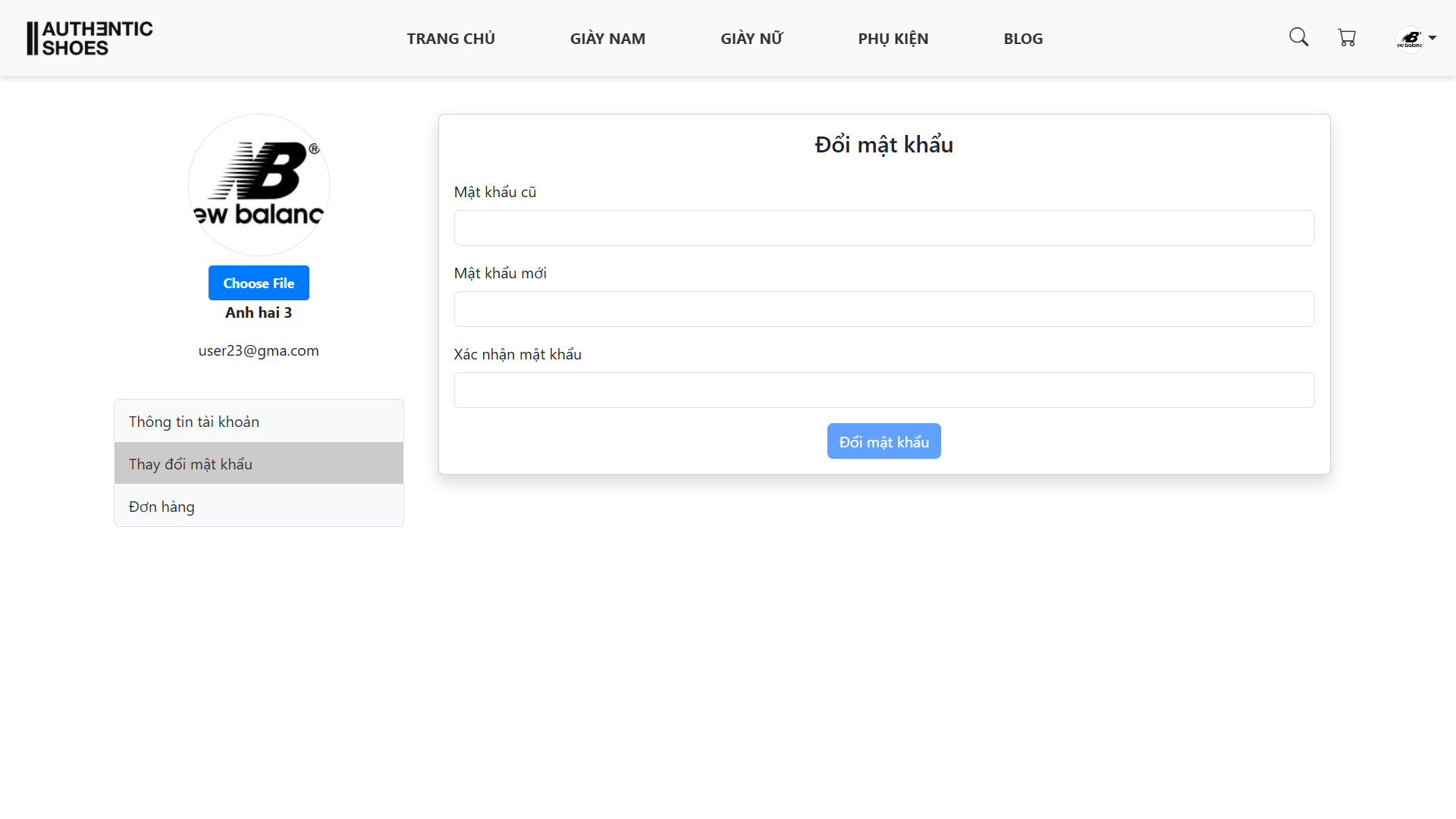
## **5.9 Giao diện chức năng 9: Giao diện trang chi tiết blog**

****

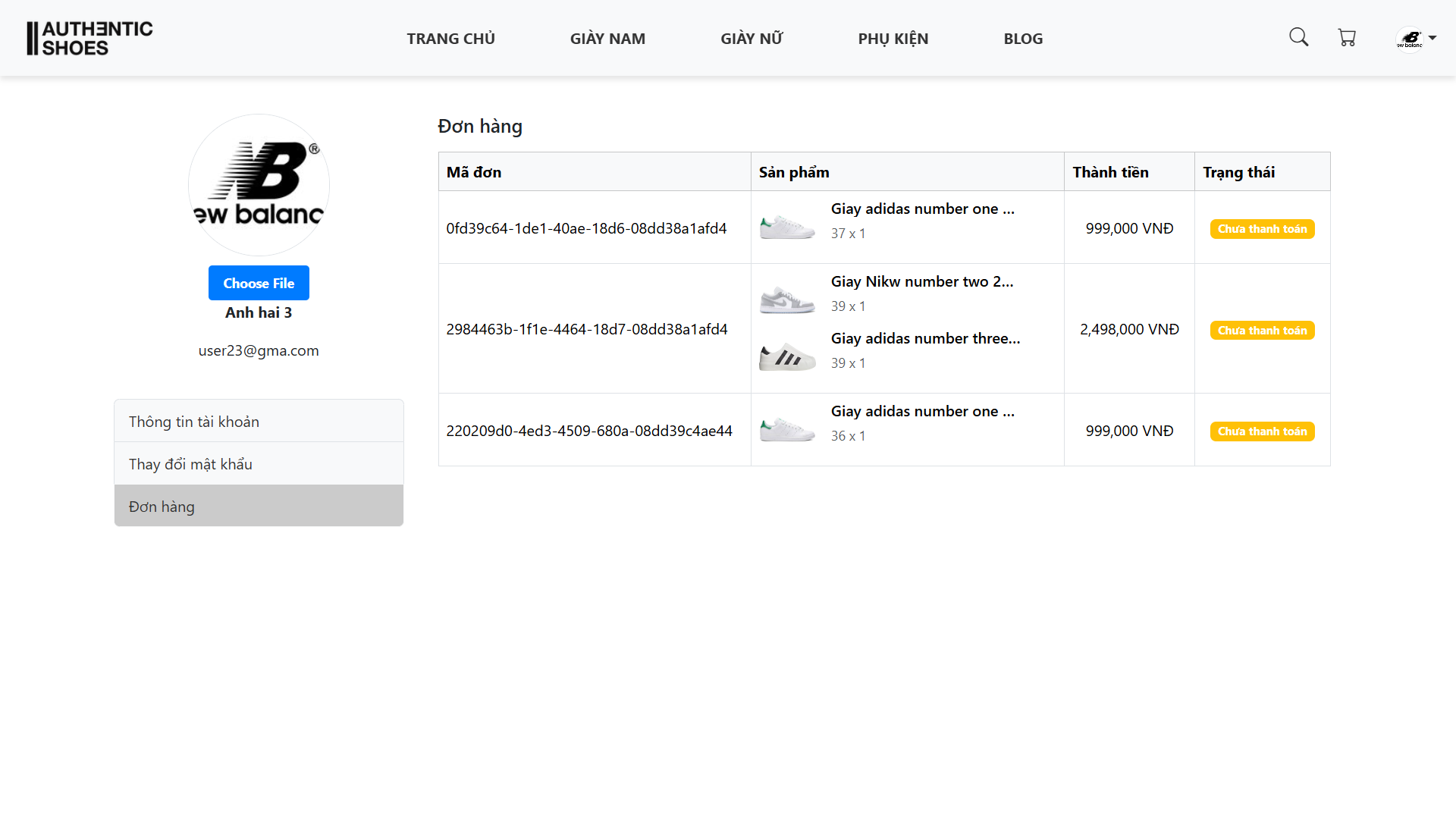
## **5.10 Giao diện chức năng 10: Giao diện trang quản lý thông tin cá nhân**

****

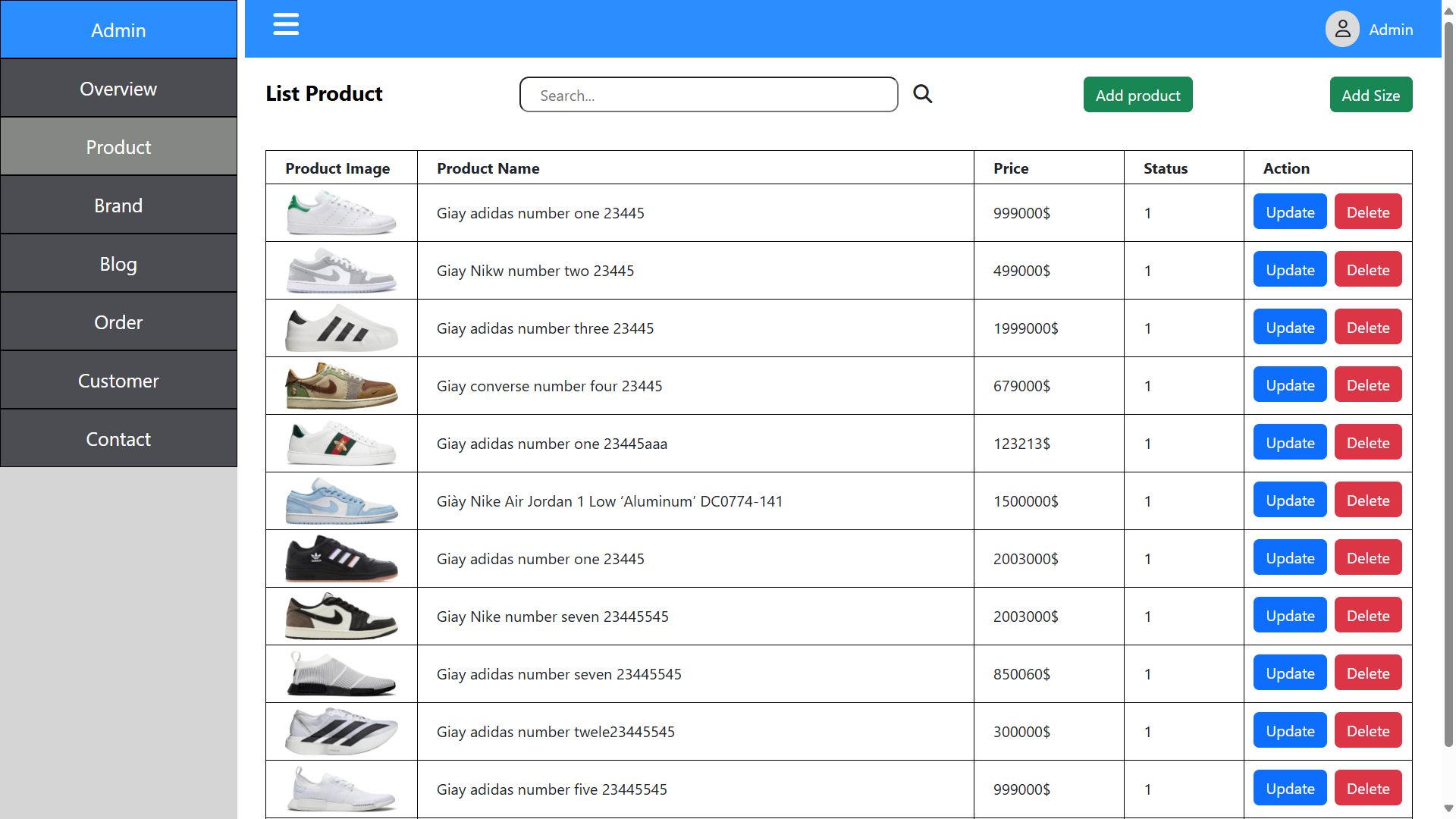
## **5.11 Giao diện chức năng 11: Giao diện trang đổi mật khẩu**

****

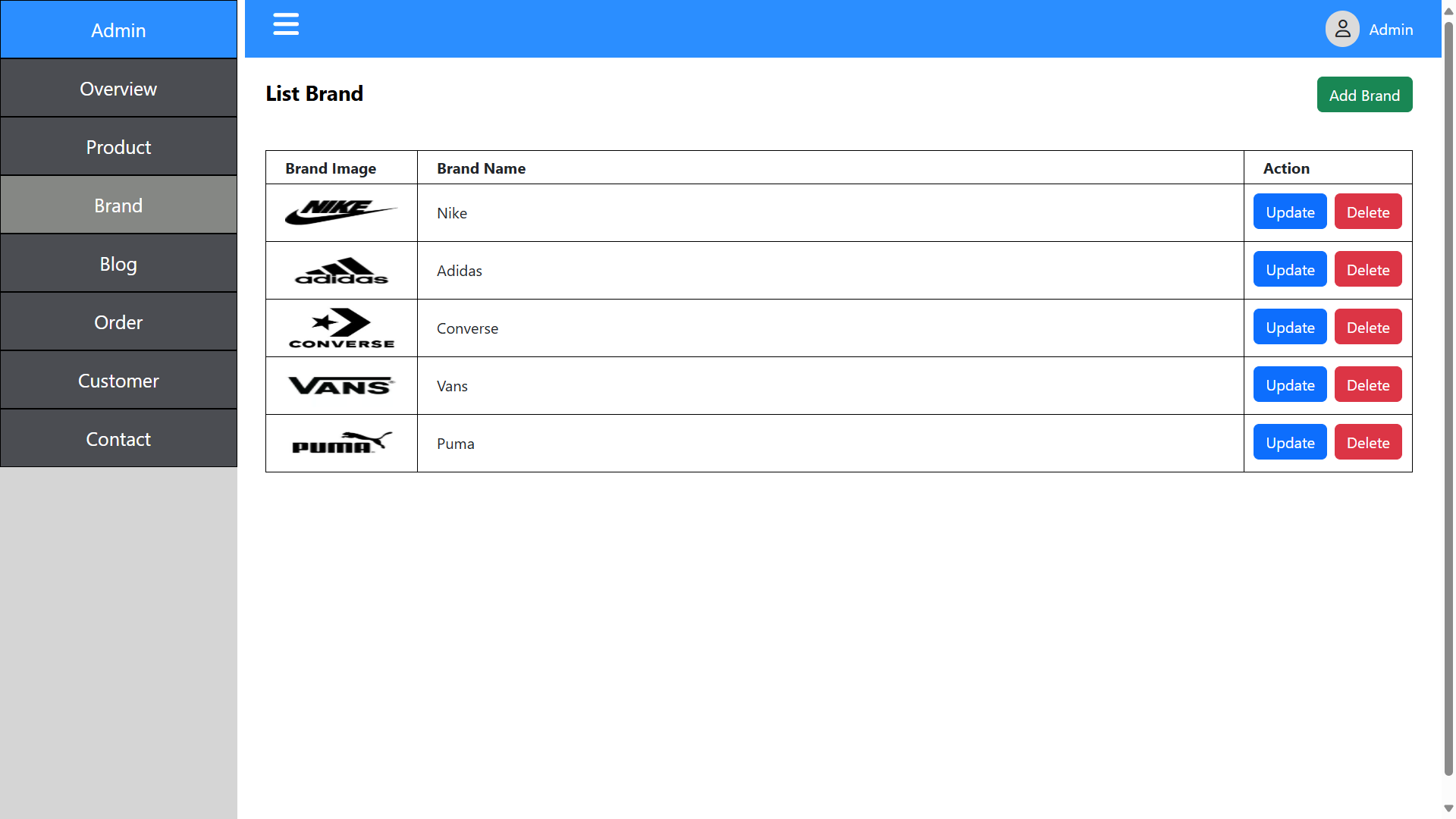
## **5.12 Giao diện chức năng 12: Giao diện trang theo dõi đơn hàng**

****

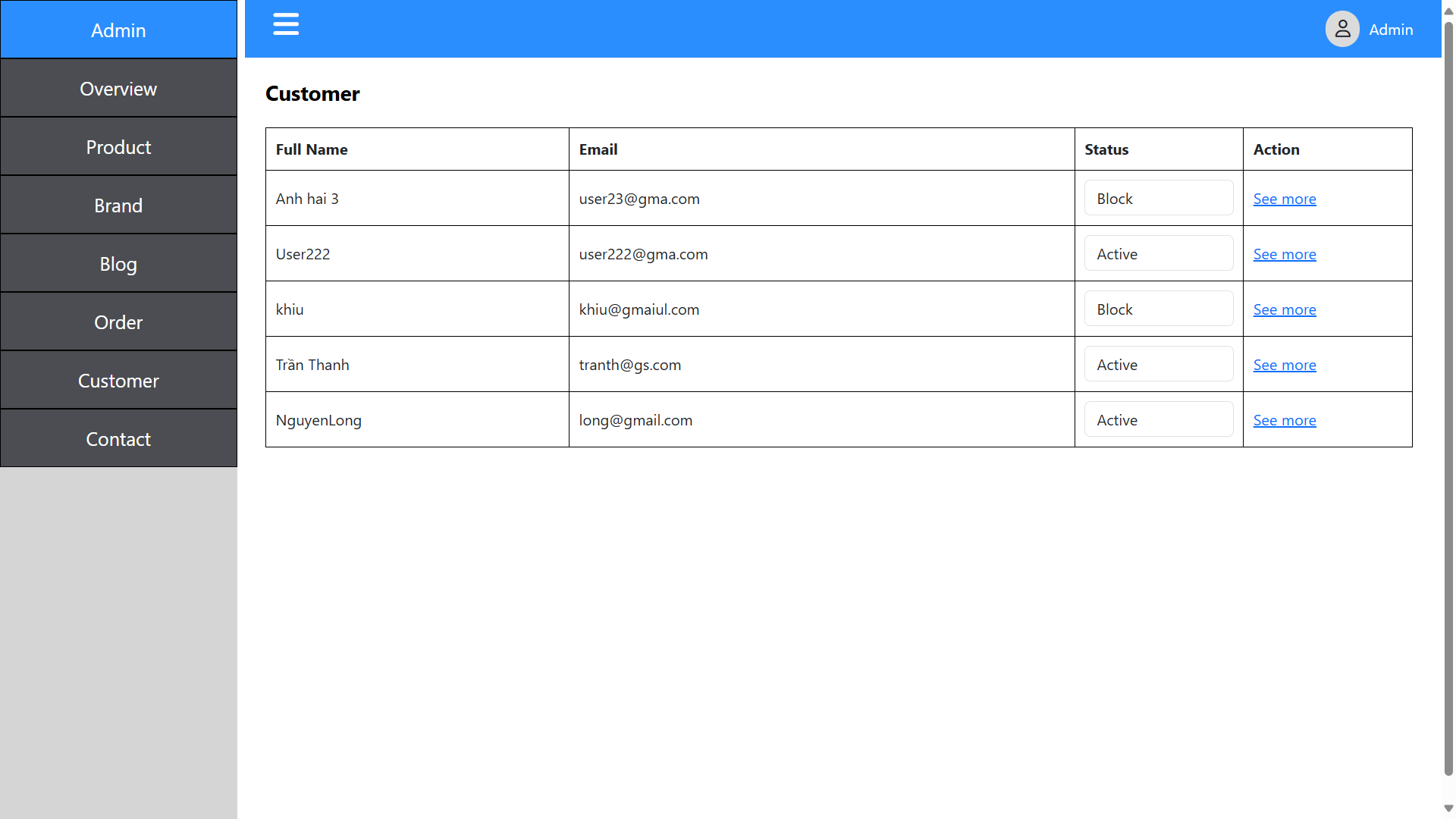
## **5.13 Giao diện chức năng 13: Giao diện trang quản lý sản phẩm**

****

## **5.14 Giao diện chức năng 14: Giao diện trang quản lý brand**

****

## **5.15 Giao diện chức năng 15: Giao diện trang khách hàng**

****

# Chương 6 - Kiểm thử

## **6.1 Kiểm thử đơn vị**

### ***6.1.1 Unit test cho authentication service***

* **RegisterUserAsync\_ShouldReturnSuccessMessage\_WhenUserIsRegistered**  
  Kiểm tra xem khi một người dùng được đăng ký thành công, hệ thống có trả về thông báo xác nhận thành công hay không.
* **RegisterUserAsync\_ShouldThrowException\_WhenEmailAlreadyExists**  
  Kiểm tra hệ thống có ném ra ngoại lệ nếu email đã tồn tại trong hệ thống khi cố gắng đăng ký.
* **RegisterUserAsync\_ShouldThrowException\_WhenUnexpectedErrorOccurs**  
  Xác minh hệ thống xử lý đúng nếu có lỗi không mong muốn xảy ra khi đăng ký người dùng mới.
* **LoginUserAsync\_ShouldThrowException\_WhenCredentialsAreInvalid**  
  Kiểm tra hệ thống ném ngoại lệ khi thông tin đăng nhập không hợp lệ (sai email hoặc mật khẩu).
* **ChangePasswordAsync\_ShouldReturnTrue\_WhenPasswordIsChanged**  
  Xác minh hệ thống trả về giá trị đúng khi người dùng thay đổi mật khẩu thành công.
* **ChangePasswordAsync\_ShouldThrowException\_WhenOldPasswordIsIncorrect**  
  Kiểm tra hệ thống ném ngoại lệ nếu mật khẩu cũ không chính xác khi cố gắng thay đổi mật khẩu.
* **ChangePasswordAsync\_ShouldThrowException\_WhenUserDoesNotExist**  
  Kiểm tra hệ thống ném ngoại lệ nếu người dùng không tồn tại trong cơ sở dữ liệu khi cố gắng thay đổi mật khẩu.
* **RefreshTokenAsync\_ShouldReturnNewAuthResult\_WhenTokenIsValid**  
  Xác minh hệ thống tạo mới token xác thực khi token cũ hợp lệ.
* **RefreshTokenAsync\_ShouldThrowException\_WhenTokenIsExpired**  
  Kiểm tra hệ thống ném ngoại lệ khi token đã hết hạn và không thể sử dụng để làm mới.
* **GetUserInfoAsync\_ShouldReturnUserVm\_WhenUserExists**  
  Kiểm tra hệ thống trả về thông tin chi tiết của người dùng khi người dùng tồn tại trong cơ sở dữ liệu.
* **GetUserInfoAsync\_ShouldThrowException\_WhenUserDoesNotExist**  
  Xác minh hệ thống ném ngoại lệ khi không tìm thấy người dùng trong cơ sở dữ liệu.
* **UpdateUserInfoAsync\_ShouldReturnTrue\_WhenUpdateIsSuccessful**  
  Kiểm tra hệ thống trả về kết quả đúng khi thông tin người dùng được cập nhật thành công.

## **6.2 Kiểm thử hệ thống**

* Kiểm thử chức năng đặt hàng

**Phương thức kiểm thử khi người dùng chưa đăng nhập**



**Phương thức khiểm thử khi có ngoại lệ xẩy ra**



**Phương thức kiểm thử khi tạo đơn hàng thành công**

****

****

# Chương 7 – Hướng dẫn cài đặt và sử dụng

## **7.1 Hướng dẫn cài đặt**

### ***7.1.1 Yêu cầu hệ thống:***

* Hệ điều hành: Windows 10 hoặc cao hơn.
* .NET SDK: Phiên bản .NET 6.0 hoặc cao hơn.
* Trình quản lý cơ sở dữ liệu: SQL Server 2022.
* Các công cụ khác: Visual Studio 2022, Node.js (cho phần frontend nếu cần).

### ***7.1.2 Các bước cài đặt:***

* **Bước 1: Clone dự án từ GitHub**  
  Sử dụng lệnh:

git clone https://github.com/nglong145/MockProject\_ShoesStore

* **Bước 2: Cấu hình cơ sở dữ liệu**
  + Mở file appsettings.json trong thư mục Backend.
  + Cập nhật chuỗi kết nối cơ sở dữ liệu (connection string) để kết nối với SQL Server.
* **Bước 3: Khởi tạo cơ sở dữ liệu**  
  Chạy các lệnh sau để áp dụng migrations:

Add-migration InitialDB

Update-database

* **Bước 4: Cài đặt frontend (nếu có)**
  + Di chuyển vào thư mục frontend:

cd frontend

* + Cài đặt các gói cần thiết:

npm install

* + Khởi chạy frontend:

npm start

* **Bước 5: Chạy backend**  
  Sử dụng Visual Studio hoặc chạy lệnh:

dotnet run

### ***7.1.3 Kiểm tra ứng dụng:***

* Truy cập URL: http://localhost:<port> (Frontend hoặc Backend).
* Đảm bảo mọi thành phần hoạt động bình thường.

## **7.2 Hướng dẫn sử dụng**

Danh sách màn hình và hướng dẫn:

* **Màn hình trang chủ**
  + Truy cập vào danh sách sản phẩm theo các danh mục.
  + Xem các bài blog giới thiệu hoặc cập nhật thông tin.
* **Màn hình đăng ký/đăng nhập**
  + Đăng ký tài khoản mới hoặc đăng nhập với email và mật khẩu.
* **Màn hình danh sách sản phẩm**
  + Sử dụng bộ lọc để tìm kiếm sản phẩm theo thương hiệu, giá, kích cỡ, màu sắc.
* **Màn hình chi tiết sản phẩm**
  + Xem thông tin chi tiết của sản phẩm.
  + Thêm sản phẩm vào giỏ hàng.
* **Màn hình giỏ hàng**
  + Xem và cập nhật số lượng hoặc xóa sản phẩm khỏi giỏ hàng.
  + Tiến hành đặt hàng.
* **Màn hình đặt hàng**
  + Nhập thông tin giao hàng và chọn phương thức thanh toán.
  + Xác nhận đặt hàng và nhận thông báo thành công.
* **Màn hình quản lý cá nhân**
  + Xem hoặc cập nhật thông tin cá nhân như tên, địa chỉ, số điện thoại.
* **Màn hình quản lý admin**
  + **Quản lý sản phẩm**: Thêm, sửa, xóa sản phẩm.
  + **Quản lý thương hiệu**: Quản lý danh mục thương hiệu.
  + **Quản lý bài viết**: Tạo và quản lý các bài blog.

# KẾT LUẬN

## **Kết quả đạt được**

Hoàn thành xây dựng và triển khai hệ thống website bán giầy với các chức năng cốt lõi như:

* **Quản lý người dùng:** Hỗ trợ các tính năng đăng ký, đăng nhập, cập nhật thông tin cá nhân.
* **Quản lý sản phẩm:** Cho phép người dùng xem danh sách, tìm kiếm sản phẩm, thêm sản phẩm vào giỏ hàng, đặt hàng, theo dõi đơn hàng và đánh giá sản phẩm.
* **Quản lý blog:** Hỗ trợ người dùng và quản trị viên trong việc đăng, xem và quản lý các bài viết blog.
* **Quản lý tác vụ quản trị viên:** Bao gồm các chức năng quản lý nhãn hiệu, sản phẩm.
* **Giao diện người dùng:** Thiết kế trực quan, thân thiện.

## **Hướng phát triển**

Nâng cấp các chức năng hiện tại nhằm cải thiện trải nghiệm người dùng:

* Cải thiện thuật toán tìm kiếm và lọc sản phẩm để hiển thị kết quả chính xác hơn.
* Tối ưu hóa hiệu năng để hệ thống hoạt động mượt mà hơn khi số lượng người dùng tăng cao.

Bổ sung các tính năng mới:

* Quản lý các đơn hàng đã đặt.
* Hỗ trợ nhiều phương thức thanh toán hơn

# PHỤ LỤC

|  |  |
| --- | --- |
| [1] | QnA\_Document, tài liệu khảo sát khách hàng |
| [2] | SRS, tài liệu đặc tả yêu cầu hệ thống |
| [3] | Task2\_ComponentDiagram\_ERD\_DB, tài liệu phân tích và thiết kế hệt hống |

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | **ASP.NET Core**: Hướng dẫn chi tiết về phát triển ứng dụng web với .NET Core.  https://learn.microsoft.com/en-us/aspnet/core |
| [2] | **Entity Framework Core**: Hướng dẫn sử dụng EF Core để quản lý cơ sở dữ liệu  https://learn.microsoft.com/en-us/ef/core |
| [3] | **Identity Framework**: Hướng dẫn triển khai xác thực và quản lý người dùng.  https://learn.microsoft.com/en-us/ef/core |
| [4] | Tham khảo cách xây dựng giao diện người dùng responsive  https://getbootstrap.com/docs/5.3/getting-started/introduction/ |
| [5] | Angular Official Documentation  https://angular.dev |
| [6] | SQL Server Documentation  https://learn.microsoft.com/en-us/sql/ |
| [7] | MoQ Library Documentation  https://github.com/moq/moq4/wiki |