|  |  |
| --- | --- |
|  |  |
| ĐÀM VĂN HIẾU | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
|  |
| ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC  NGÀNH CÔNG NGHỆ THÔNG TIN |
|  |
| **Xây dựng Website bán giày**  **sử dụng VueJs và Asp.net** |
|  |
|  |
| **CBHD: TS. Vũ Việt Thắng** |
| CÔNG NGHỆ THÔNG TIN | **Sinh viên: Đàm Văn Hiếu** |
| **Mã số sinh viên: 2019603858** |
|  |
|  |
|  |
| Hà Nội – Năm 2023 |
|  |

**MỤC LỤC**

[LỜI NÓI ĐẦU iii](#_Toc133824903)

[DANH MỤC CÁC CHỮ VIẾT TẮT v](#_Toc133824904)

[DANH MỤC BẢNG BIỂU v](#_Toc133824905)

[DANH MỤC HÌNH ẢNH vii](#_Toc133824906)

[CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI 1](#_Toc133824907)

[1.1. Tên đề tài 1](#_Toc133824908)

[1.2. Lý do chọn đề tài 1](#_Toc133824909)

[1.3. Mục tiêu của đề tài 2](#_Toc133824910)

[1.4. Đối tượng và phạm vi 3](#_Toc133824911)

[1.5. Kết quả dự kiến đạt được 3](#_Toc133824912)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT VÀ CÔNG CỤ SỬ DỤNG 4](#_Toc133824913)

[2.1. Cơ sở lý thuyết 4](#_Toc133824914)

[2.1.1. Ngôn ngữ UML 4](#_Toc133824915)

[2.1.2. Ngôn ngữ ASP.NET Core API 6](#_Toc133824916)

[2.1.3. Ngôn ngữ JavaScript 7](#_Toc133824917)

[2.1.4. VueJS 9](#_Toc133824918)

[2.1.5. Cơ sở dữ liệu MySQL 11](#_Toc133824919)

[2.2. Công cụ sử dụng 12](#_Toc133824920)

[CHƯƠNG 3. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 13](#_Toc133824921)

[3.1. Phân tích hệ thống 13](#_Toc133824922)

[3.1.1. Vai trò của người dùng 13](#_Toc133824923)

[3.1.2. Yêu cầu chức năng 13](#_Toc133824924)

[3.1.3. Yêu cầu phi chức năng 14](#_Toc133824925)

[3.2. Biểu đồ use case tổng quát 15](#_Toc133824926)

[3.3. Mô hình thực thể liên kết 16](#_Toc133824927)

[3.4. Đặc tả use case 17](#_Toc133824928)

[3.4.1. Use case đăng ký 17](#_Toc133824929)

[3.4.2. Use case đăng nhập 18](#_Toc133824930)

[3.4.3. Use case quên mật khẩu 20](#_Toc133824931)

[3.4.4. Use case quản lý tài khoản 21](#_Toc133824932)

[3.4.5. Use case xem thông tin sản phẩm 23](#_Toc133824933)

[3.4.6. Use case quản lý giỏ hàng 25](#_Toc133824934)

[3.4.7. Use case mua hàng 27](#_Toc133824935)

[3.4.8. Use case quản lý đơn hàng 29](#_Toc133824936)

[3.4.9. Use case sắp xếp sản phẩm 31](#_Toc133824937)

[3.4.10. Use case lọc sản phẩm theo giá bán 32](#_Toc133824938)

[3.4.11. Use case xem danh mục sản phẩm 33](#_Toc133824939)

[3.4.12. Use case đánh giá sản phẩm 34](#_Toc133824940)

[3.4.13. Use case đóng góp ý kiến cho cửa hàng 36](#_Toc133824941)

[3.4.14. Use case quản lý sản phẩm 37](#_Toc133824942)

[3.4.15. Use case quản lý người dùng 39](#_Toc133824943)

[3.4.16. Use case quản lý đơn hàng 41](#_Toc133824944)

[3.4.17. Use case quản lý danh mục 42](#_Toc133824945)

[3.4.18. Use case quản lý góp ý cửa hàng 43](#_Toc133824946)

[3.4.19. Use case quản lý thống kê 45](#_Toc133824947)

[3.5. Thiết kế hệ thống 46](#_Toc133824948)

[3.5.1. Thiết kế cơ sở dữ liệu 46](#_Toc133824949)

[3.5.2. Thiết kế giao diện 53](#_Toc133824950)

[KẾT LUẬN 54](#_Toc133824951)

[TÀI LIỆU THAM KHẢO 56](#_Toc133824952)

# LỜI NÓI ĐẦU

Thời đại công nghệ thông tin bùng nổ, cùng với sự phát triển của các nền tảng online đã tạo ra một xu hướng mới là kinh doanh online. Nếu vẫn giữ nguyên phương pháp bán hàng truyền thống, bạn sẽ khó mà cạnh tranh được với những đối thủ đã sớm tiến quân và chiếm thị phần trên các kênh bán hàng online.

Năm 2020 vừa qua, Nghị định 100 của Chính phủ và cơn đại dịch Covid-19 là minh chứng rõ ràng nhất cho sự phát triển của kinh doanh online. Qua quãng thời gian nghỉ dịch, mọi người đều có thể thấy được kinh doanh online là xu hướng kinh doanh tất yếu. Kinh doanh Offline thôi là chưa đủ: Chúng ta có thể nhìn lại qua năm 2020, với sự bắt đầu bằng những tín hiệu không được lạc quan với ngành kinh doanh offline do sự tác động của dịch Covid-19. Các lệnh giãn cách xã hội kéo dài, lượng khách hàng giảm đi đáng kể, nhu cầu mua sắm cũng giảm mạnh với một số mặt hàng.

Với tình hình dịch bệnh kéo dài, nếu chỉ khăng khăng duy trì các kênh bán hàng offline, doanh nghiệp sẽ mất đi một lượng lớn khách hàng. Do hành vi của người tiêu dùng đã bị tác động và thay đổi. Sự hoàn thiện của hệ sinh thái cho người kinh doanh online: Khoảng một vài năm gần đây, hệ sinh thái cho người kinh doanh online đã được xây dựng gần như hoàn thiện. Các phân đoạn từ chọn nguồn hàng, kênh bán hàng hay đơn vị vận chuyển đều đã có sẵn. Tạo điều kiện cho nhiều hộ kinh doanh nhỏ lẻ. Bạn có thể xây dựng các Fanpage ở Facebook để chạy quảng cáo sản phẩm. Có thể xây dựng website riêng cho nhãn hàng hoặc liên kết với các sàn thương mại điện tử như tiki, shoppe, ….

Sau khi tìm hiểu và dưới sự hướng dẫn của thầy **Vũ Việt Thắng**, em đã phân tích và thiết kế được **Website bán giày sử dụng VueJs và Asp.net**. Để giúp khách hàng đặt mua được những đôi giày theo ý thích của mình, giúp cho chủ cửa hàng có thể quản lý cửa hàng một cách nhanh chóng, dễ dàng và tiện lợi.

Để hoàn thành được đồ án tốt nghiệp này, em xin được gửi lời cảm ơn chân thành đến các thầy cô trong khoa Công nghệ thông tin Trường Đại học Công Nghiệp Hà Nội đã tận tình giảng dạy và trang bị kiến thức cho em trong suốt thời gian em học tập tại trường. Thầy giáo hướng dẫn đề tài – **Tiến Sĩ Vũ Việt Thắng**, giảng viên Khoa Công nghệ Thông tin Trường Đại học Công Nghiệp Hà Nội – đã tận tụy hết lòng giúp đỡ, hướng dẫn, chỉ dẫn tận tình để giúp em hoàn thành được đồ án. Bạn bè đã góp ý và giúp đỡ tôi trong suốt thời gian tôi làm đồ án.

***Hà Nội, Ngày tháng năm 2023***

# DANH MỤC CÁC CHỮ VIẾT TẮT

|  |  |
| --- | --- |
| Từ viết tắt | Giải thích |
| API | Application Programming Interface |
| HTML | HyperText Markup Language |
| CSS | Cascading Style Sheets |
| MVC | Model-View-Controller |
| UI | User Interface |
| PK | Primary key |
| FK | Foreign Key |

# DANH MỤC BẢNG BIỂU

[Bảng 1. Bảng Category để lưu thông tin danh mục sản phẩm 34](#_Toc133772745)

[Bảng 2. Bảng Size để lưu thông size sản phẩm 34](#_Toc133772746)

[Bảng 3. Bảng Evaluate để lưu thông tin đánh giá của sản phẩm 35](#_Toc133772747)

[Bảng 4. Bảng Shoe để lưu thông tin sản phẩm 35](#_Toc133772748)

[Bảng 5. Bảng Customer để lưu thông tin người dùng 36](#_Toc133772749)

[Bảng 6. Bảng FeedBack để lưu thông tin đóng góp ý kiến của người dùng 37](#_Toc133772750)

[Bảng 7. Bảng Cart để lưu thông tin giỏ hàng 37](#_Toc133772751)

[Bảng 8. Bảng CartDetail để lưu thông tin chi tiết giỏ hàng 38](#_Toc133772752)

[Bảng 9. Bảng Bill để lưu thông tin đơn hàng 38](#_Toc133772753)

[Bảng 10. Bảng BillDetail để lưu thông tin chi tiết đơn hàng 39](#_Toc133772754)

# DANH MỤC HÌNH ẢNH

[Hình 1. ASP.NET Core 6](#_Toc133846818)

[Hình 2. Javascript 7](https://d.docs.live.net/c2563e8ec37aad03/Máy%20tính/DATN.docx#_Toc133846819)

[Hình 3. VueJS 9](#_Toc133846820)

[Hình 4. MySQL 11](#_Toc133846821)

[Hình 5. Biểu đồ use case tổng quát 15](#_Toc133846822)

[Hình 6. Biều đồ thực thể liên kết 16](#_Toc133846823)

[Hình 7. Biểu đồ phân rã đăng ký 17](#_Toc133846824)

[Hình 8. Biều đồ trình tự đăng ký 18](#_Toc133846825)

[Hình 9. Biều đồ phân rã đăng nhập 18](#_Toc133846826)

[Hình 10. Biểu đồ trình tự đăng nhập 19](#_Toc133846827)

[Hình 11. Biều đồ phân rã quên mật khẩu 20](#_Toc133846828)

[Hình 12. Biểu đồ trình tự quên mật khẩu 21](#_Toc133846829)

[Hình 13. Biểu đồ phân rã quản lý tài khoản 21](#_Toc133846830)

[Hình 14. Biểu đồ trình tự quản lý tài khoản 23](#_Toc133846831)

[Hình 15. Biểu đồ phân rã thông tin sản phẩm 23](#_Toc133846832)

[Hình 16. Biểu đồ trình tự xem thông tin sản phẩm 24](#_Toc133846833)

[Hình 17. Biểu đồ phân rã quản lý giỏ hàng 25](#_Toc133846834)

[Hình 18. Biểu đồ trình tự quản lý giỏ hàng 26](#_Toc133846835)

[Hình 19. Biểu đồ phân ra mua hàng 27](#_Toc133846836)

[Hình 20. Biểu đồ trình tự mua hàng 29](#_Toc133846837)

[Hình 21. Biểu đồ phân rã quản lý đơn hàng 29](#_Toc133846838)

[Hình 22. Biểu đồ trình tự quản lý đơn hàng 31](#_Toc133846839)

[Hình 23. Biểu đồ phân rã sắp xếp sản phẩm 31](#_Toc133846840)

[Hình 24. Biểu đồ trình tự sắp xếp sản phẩm 32](#_Toc133846841)

[Hình 25. Biểu đồ phân rã lọc sản phẩm theo giá bán 32](#_Toc133846842)

[Hình 26. Biểu đồ trình tự lọc sản phẩm theo giá bán 33](#_Toc133846843)

[Hình 27. Biểu đồ phân rã xem danh mục sản phẩm 33](#_Toc133846844)

[Hình 28. Biểu đồ phân rã xem danh mục sản phẩm 34](#_Toc133846845)

[Hình 29. Biểu đồ phân rã đánh giá sản phẩm 35](#_Toc133846846)

[Hình 30. Biểu đồ trình tự đánh giá sản phẩm 36](#_Toc133846847)

[Hình 31. Biểu đồ phân rã đóng góp ý kiến cho cửa hàng 36](#_Toc133846848)

[Hình 32. Biểu đồ trình tự đóng góp ý kiến cho cửa hàng 37](#_Toc133846849)

[Hình 33. Biểu đồ phân rã quản lý sản phẩm 38](#_Toc133846850)

[Hình 34. Biểu đồ trình tự quản lý sản phẩm 40](#_Toc133846851)

[Hình 35. Biểu đồ phân rã quản lý người dùng 41](#_Toc133846852)

[Hình 36. Biểu đồ trình tự quản lý người dùng 43](#_Toc133846853)

[Hình 37. Biểu đồ phân rã quản lý đơn hàng 43](#_Toc133846854)

[Hình 38. Biểu đồ trình tự quản lý đơn hàng 45](#_Toc133846855)

[Hình 39. Biểu đồ phân rã quản lý danh mục 46](#_Toc133846856)

[Hình 40. Biểu đồ trình tự quản lý danh mục 48](#_Toc133846857)

[Hình 41. Biểu đồ phân rã quản lý góp ý cửa hàng 48](#_Toc133846858)

[Hình 42. Biểu đồ trình tự quản lý góp ý cửa hàng 50](#_Toc133846859)

[Hình 43. Giao diện trang chủ 58](#_Toc133846860)

[Hình 44. Giao diện trang giới thiệu 59](#_Toc133846861)

[Hình 45. Giao diện trang sản phẩm 60](#_Toc133846862)

[Hình 46. Giao diện trang liên hệ 61](#_Toc133846863)

[Hình 47. Màn hình đăng nhập, đăng ký 61](#_Toc133846864)

[Hình 48. Giao diện màn hình quên mật khẩu 62](#_Toc133846865)

[Hình 49. Giao diện trang chi tiết sản phẩm 63](#_Toc133846866)

[Hình 50. Giao diện giỏ hàng 64](#_Toc133846867)

[Hình 51. Giao diện thanh toán 64](#_Toc133846868)

[Hình 52. Giao diện quản lý đơn hàng 65](#_Toc133846869)

[Hình 53. Giao diện trang chủ 65](#_Toc133846870)

[Hình 54. Giao diện trang quản lý tài khoản người dùng 66](#_Toc133846871)

[Hình 55. Giao diện trang quản lý sản phẩm 66](#_Toc133846872)

[Hình 56. Giao diện trang quản lý đơn hàng 67](#_Toc133846873)

[Hình 57. Giao diện trang quản lý danh mục 67](#_Toc133846874)

# TỔNG QUAN VỀ ĐỀ TÀI

## Tên đề tài

***Xây dựng Website bán giày sử dụng VueJs và Asp.net***

## Lý do chọn đề tài

Ngày nay, các hoạt động mua bán đồ dùng thông qua website bán hàng là một chuyện hết sức bình thường. Website bán hàng trở thành nơi quảng cáo, thực hiện giao dịch mua bán, thậm chí nguồn thu từ việc bán hàng qua website đã trở thành doanh số chính của nhiều cửa hàng. Website đã trở thành một phần không thể thiếu trong việc kinh doanh theo hình thức thương mại điện tử.

Với sự bùng nổ của cuộc cách mạng công nghiệp 4.0, internet đã có ảnh hưởng sâu sắc đến đời sống, kinh tế và xã hội. Có thể nói internet là một công cụ không thể thiếu trong cuộc sống. Để tồn tại và phát triển, con người cần phải học tập và tiếp cận nhanh chóng với những công nghệ mới, theo kịp sự phát triển của thời đại.

Trong nền kinh tế thị trường hiện nay, việc cạnh tranh sản phẩm ngày càng quyết liệt. Mục tiêu lớn nhất của cạnh tranh đó là thu hút nhiều khách hàng quan tâm đến sản phẩm của mình. Để thu hút được nhiều khách hàng, bên cạnh chất lượng sản phẩm và giá cả, các chiến lược kinh doanh, chiến lược marketing cũng rất quan trọng. Website trở thành công cụ, một phương tiện không thể thiếu để thực hiện marketing, quảng bá sản phẩm, thương hiệu của công ty, doanh nghiệp. Website trở thành một phương tiện để cửa hàng giới thiệu sản phẩm ra thị trường trong nước. Đây là nơi khách hàng tìm thấy thông tin giới thiệu về cửa hàng, các thông tin chi tiết về các sản phẩm. Nó cũng là một kênh giao tiếp, tư vấn trực tuyến với khách hàng, giải quyết kịp thời thắc mắc và vấn đề của khách hàng, giúp khách hàng hài lòng.

Thiết kế website là nhu cầu cần thiết đối với các cửa hàng. Website càng tốt càng mang lại trải nghiệm tốt cho khách hàng, càng hướng khách hàng tới sản phẩm của mình. Website có ảnh hưởng rất lớn tới việc kinh doanh của cửa hàng như doanh thu, chi phí quản lý, sự hài lòng và đánh giá của khách.

Đối với website bán hàng nói chung và website bán giày nói riêng, bên cạnh giao diện đẹp thì website hoạt động như thế nào là rất quan trọng. Khách hàng khi ghé thăm website bán giày thì khách hàng sẽ giao tiếp và thao tác với giao diện của website đó. Giao diện website phải đẹp, phải gây ấn tượng với khách hàng và cũng sẽ tạo nên điểm đặc biệt của cửa hàng. Khách hàng nhìn vào website có thể dễ dàng sử dụng các công cụ có trên đó. Về hoạt động của website thì website phải chạy mượt mà để khách hàng thoải mái, không bị khó chịu khi sử dụng. Website giúp khách hàng có thể xem được thông tin mình cần, có thể tìm tới sản phẩm mình cần.

Nhận thức được sự quan trọng và cũng như để có thể áp dụng được những kiến thức đã được học và tìm hiểu, em xin được áp dụng những kiến thức đã được học và tìm hiểu đó để xây dựng một website cụ thể là: ***“Xây dựng Website bán giày sử dụng VueJs và Asp.net”.***

## Mục tiêu của đề tài

Đề tài: ***Xây dựng Website bán giày sử dụng VueJs và Asp.net*** đáp ứng được những mục tiêu:

* Hiểu biết những kiến thức cần thiết về xây dựng giao diện trang web, phân tích thiết kế hệ thống, cơ sở dữ liệu.
* Nắm được các kiến thức cơ bản về Asp.net, JavaScript, VueJS.
* Lợi ích đầu tiên khi sở hữu một website bán hàng đó chính là việc tiếp cận được những khách hàng tiềm năng. Dù khách hàng có ở bất cứ đâu trên thế giới, chỉ cần có mạng internet là có thể tìm thấy website bán hàng, biết đến sản phẩm và thương hiệu của cửa hàng. Từ đó, chắc chắn rằng lượng khách hàng sẽ tăng lên đáng kể.
* Tiết kiệm được thời gian cho khách hàng, khách hàng không cần phải đến tận cửa hàng mới mua được hàng mà họ có thể thông qua website xem và mua hàng. Hơn nữa mặt hàng khách mua còn có thể được chuyển về địa chỉ yêu cầu đúng thời gian khách hàng mong muốn, tiết kiệm được cả chi phí đi lại cho khách hàng.
* Thông tin sản phẩm được cập nhật nhanh chính xác qua từng ngày đến khách hàng**.**

## Đối tượng và phạm vi

Bất cứ ai có nhu cầu mua giày trực tuyến trên nền tảng internet.

## Kết quả dự kiến đạt được

Website bao gồm những mục chính và chức năng chính:

* Đăng ký, đăng nhập tài khoản cá nhân.
* Thay đổi mật khẩu, cập nhật thông tin tài khoản.
* Quên mật khẩu của tài khoản.
* Phân quyền giữa người quản trị, nhân viên và khách hàng.
* Chức năng quản trị: quản lý tài khoản, sản phẩm , đơn hàng, danh mục sản phẩm, góp ý của khách hàng.
* Tìm kiếm sản phẩm theo danh mục, giá bán và xem chi tiết sản phẩm.
* Thêm sản phẩm vào giỏ hàng và lưu trữ giỏ hàng của mỗi người dùng.
* Đặt hàng, thanh toán trực tuyến.
* Gửi góp ý cho cửa hàng

# CƠ SỞ LÝ THUYẾT VÀ CÔNG CỤ SỬ DỤNG

## Cơ sở lý thuyết

### **Ngôn ngữ UML**

UML là viết tắt của “Unified Modeling Language” - là ngôn ngữ mô hình hóa thống nhất, dùng để đặc tả, hình dung, xây dựng và tư liệu hóa phần mềm hướng đối tượng.

UML dùng để:

* Trực quan hoá: Đưa ra các mô hình trực quan về hệ thống để mọi người có thể hiểu được.
* Đặc tả: UML cho phép xây dựng các mô hình đúng đắn và đầy đủ để mô tả hệ thống.
* Xây dựng: Các mô hình UML có thể liên kết trực tiếp với nhiều ngôn ngữ lập trình Java, C++, VB, ...
* Lập tài liệu: UML cho phép tạo các tài liệu như tài liệu về các yêu cầu của người dùng, kiến trúc hệ thống, thiết kế hệ thống, kiểm thử, kế hoạch dự án, ...

UML có thể được sử dụng làm công cụ giao tiếp giữa người dùng, nhà phân tích, nhà thiết kế và nhà phát triển phần mềm.

Các phần tử của UML:

View (Quan sát): Theo các phương diện khác nhau của hệ thống cần phân tích, thiết kế. Dựa vào các quan sát để thiết lập kiến trúc cho hệ thống cần phát triển.

***Có 9 loại biểu đồ:***

* Use Case Diagram (Biểu đồ ca sử dụng): Mô tả sự tương tác giữa các tác nhân và hệ thống thông qua các ca sử dụng.
* Class Diagram (Biểu đồ lớp): Biểu đồ lớp chỉ ra sự tồn tại giữa các lớp và mối quan hệ giữa chúng trong thiết kế logic trong hệ thống. Khung nhìn tĩnh của hệ thống chủ yếu hỗ trợ các chức năng của hệ thống.
* Object Diagram (Biểu đồ đối tượng): Biểu đồ đối tượng đại diện cho một thể hiện của biểu đồ lớp, cho nên các khái niệm cơ bản là giống biểu đồ lớp. Biểu đồ đối tượng được sử dụng để hiện thị một tập hợp các đối tượng và các mối quan hệ của chúng như một thể hiện.
* Sequence Diagram (Biểu đồ trình tự): Là biểu đồ để xác định và chỉ rõ vai trò của đối tượng tham gia vào luồng sự kiện của use case.
* Collaboration Diagram (Biểu đồ cộng tác): Tương tự như biểu đồ trình tự nhưng nhấn mạnh vào sự tương tác của các đối tượng trên cơ sở cộng tác với nhau bằng cách trao đổi các thông điệp để thực hiện các yêu cầu theo ngữ cảnh công việc.
* State Diagram (Biểu đồ trạng thái): Thể hiện chu kỳ hoạt động của các đối tượng, của các hệ thống con và của cả hệ thống.
* Activity Diagram (Biểu đồ hành động): Được sử dụng để mô tả các hoạt động và các hành động được thực hiện trong một use case.
* Component Diagram (Biểu đồ thành phần): Chỉ ra cấu trúc vật lý của các thành phần trong hệ thống.
* Deployment Diagram (Biểu đồ triển khai): Chỉ ra cách bố trí vật lý các thành phần theo kiến trúc được thiết kế của hệ thống.

Relationship (Quan hệ):

* Realization (Hiện thực hoá):Là quan hệ ngữ nghĩa giữa giao diện và lớp.
* Dependency (Phụ thuộc):Là quan hệ ngữ nghĩa giữa hai phần tử, trong đó sự thay đổi của một tử sẽ tác động đến ngữ nghĩa của phần tử phụ thuộc.
* Generalization (Kế thừa): Là quan hệ mô tả sự khái quát hoá mà trong đó một số đối tượng cụ thể (của lớp con) sẽ được kế thừa các thuộc tính, các phương thức của các đối tượng tổng quát (lớp cơ sở).
* Association (Kết hợp): Là quan hệ cấu trúc xác định mối liên kết giữa các lớp đối tượng.

### **Ngôn ngữ ASP.NET Core API**

ASP.NET là framework được phát triển bởi gã khổng lồ công nghệ Microsoft từ lâu. Trong ASP.NET sử dụng nhiều ngôn ngữ lập trình như C#, F, Visual Basic để xây dựng ứng dụng. Trong đó phổ biến hơn cả là C#.

ASP.NET có nhiều thể hiện gồm: ASP.NET MVC, ASP.NET Web API, ASP.NET Core MVC, ASP.NET Core API.

Trong đó ASP.NET Core là phiên bản được xây dựng để phát triển ứng dụng chạy trên đa nền tảng của Microsoft. Hiện nay các phiên bản cũ của ASP.NET đã ngừng phát triển trong khi nền tảng mới .NET 6.0 ra đời gần đây đã xây dựng mới để hoạt động đa nền tảng.

Icon

Description automatically generated

Hình 1. ASP.NET Core

### File:Unofficial JavaScript logo 2.svg - Wikimedia CommonsN**gôn ngữ JavaScript**

Hình 2. Javascript

JavaScript là một ngôn ngữ lập trình đa nền tảng (cross-platform), ngôn ngữ lập trình kịch bản, hướng đối tượng. JavaScript là một ngôn ngữ nhỏ và nhẹ (small and lightweight). JavaScript chứa các thư viện tiêu chuẩn cho các đối tượng (object), ví dụ như: Array, Date, Math, và các yếu tố cốt lõi của ngôn ngữ lập trình như: toán tử (operators), cấu trúc điều khiển (control structures), và câu lệnh. JavaScript có thể được mở rộng cho nhiều mục đích bằng việc bổ sung thêm các object; ví dụ:

* Client-side JavaScript - JavaScript phía máy khách, JavaScript được mở rộng bằng cách cung cấp các object để quản lý trình duyệt và Document Object Model (DOM) của nó. Ví dụ, phần mở rộng phía máy khách cho phép một ứng dụng tác động tới các yếu tố trên một trang HTML và phản hồi giống các tác động của người dùng như click chuột, nhập form, và chuyển trang.
* Server-side JavaScript - JavaScript phía máy chủ, JavaScript được mở rộng bằng cách cung cấp thêm các đối tượng cần thiết để để chạy JavaScript trên máy chủ. Ví dụ, phần mở rộng phía server này cho phép ứng dụng kết nối với cơ sở dữ liệu (database), cung cấp thông tin một cách liên tục từ một yêu cầu tới phần khác của ứng dụng, hoặc thực hiện thao tác với các tập tin trên máy chủ.

JavaScript được phát triển bởi Brendan Eich, một nhân viên của Netscape, vào tháng 9 năm 1995. Được biết đến lần đầu tiên là Mocha, rồi sau đó được đổi tên thành LiveScript và cuối cùng là JavaScript nổi tiếng như hiện giờ. Phiên bản đầu tiên của ngôn ngữ này bị giới hạn độc quyền bởi Netscape và chỉ có các tính năng hạn chế, nhưng nó tiếp tục phát triển theo thời gian.

Năm 1996, JavaScript được chính thức đặt tên là ECMAScript. ECMAScript 2 phát hành năm 1998 và ECMAScript 3 tiếp tục ra mắt vào năm 1999. Nó liên tục phát triển thành JavaScript ngày nay, giờ đã hoạt động trên khắp mọi trình duyệt và trên khắp các thiết bị từ di động đến máy tính bàn.

JavaScript liên tục phát triển kể từ đó, có lúc đạt đến 92% website đang sử dụng JavaScript vào năm 2016. Chỉ trong 20 năm, nó từ một ngôn ngữ lập trình riêng trở thành công cụ quan trọng nhất trên bộ công cụ của các chuyên viên lập trình web.

JavaScript cùng với HTML, CSS đã trở thành ngôn ngữ không thể thiếu trong lập trình web hiện nay:

* Với việc phát triển mạnh mẽ của công nghệ, JavaScript đã dần trở thành một ngôn ngữ đa nền tảng với nhiều ứng dụng khác nhau.
* Các Framework, Library cho lập trình front-end mạnh mẽ hỗ trợ xây dựng các trang Single-page application (SPA) như: ReactJS, Angular, VueJS.
* Hỗ trợ lập trình mobile bằng thư viện React Native.
* Hỗ trợ lập trình Window bằng Electron.
* Không chỉ phát triển ở khía cạnh lập trình phía client-side, JavaScript cũng hỗ trợ viết server-side rất mạnh mẽ với NodeJS.

Với hệ sinh thái mã nguồn mở khổng lồ cùng sự phát triển như hiện nay, JavaScript được dự đoán sẽ phát triển mạnh mẽ hơn nữa trong những năm tới.

### **VueJS**

#### **VueJS là gì?**

VueJS là một framework linh động dùng để xây dựng giao diện người dùng.

VueJS được thiết kế từ đầu theo hướng cho phép và khuyến khích việc phát triển ứng dụng theo từng bước, VueJs cũng đáp ứng được dễ dàng nhu cầu xây dựng những ứng dụng SPA (Single Page Applications).

A blue and green logo

Description automatically generated with low confidence

Hình 3. VueJS

#### **Lịch sử phát triển của VueJS**

Vue được thai nghén và phát triển bởi **Evan You** khi tác giả đang làm kỹ sư cho ông lớn Google. Khi đang sử dụng Angular cho một số dự án khác thì Evan có ý tưởng trích một phần mà mình thích của Angular rồi từ đó xây dựng một cái gì đó nhẹ nhàng hơn Angular mà không cần phải chú ý đến những khái niệm bổ sung rườm rà. Đây được xem là tiền đề giúp VueJS đã, đang có mặt và phát triển mạnh mẽ tại thời điểm hiện tại.

Vue đã được phát hành đầu năm 2014 và dự án này đã được đăng tải lên HackerNew, Echo Js ngay trong ngày đầu tiên ra mắt.

Vue 2.0 được release vào đầu năm 2016 để ganh đua với ReactJs. Thực tế VueJs được chứng minh là nhanh hơn và gọn hơn so với ReactJs và Angular 2.0. Ngoài ra, thời gian tiếp cần và làm chủ của Vue tương đối ngắn.

#### **Tính khả dụng**

* Tạo ra những Website chạy thật mượt mà, nhanh, có khả năng mở rộng cao và đơn giản thực hiện.
* Cho phép thiết kế linh hoạt giao diện người dùng, biến nó từ phức tạp thành đơn giản hơn.

#### **Các đặc trưng của VueJS**

* No build step required: Vue không cần sử dụng build tool quá phức tạp quá nhiều bước mới có thể xây dựng ứng dụng, bạn chỉ cần khai báo một script là có thể phát triển một ứng dụng bằng Vue.
* Command line interface: Vue có thể tạo cấu trúc project nhanh chóng hơn nhờ vào command line interface.
* Đa dạng tài liệu: Hiện nay, nguồn tài liệu về Vue ngày càng đa dạng, rõ ràng về ngôn ngữ nên có thể dễ dàng trở thành chuyên gia về nó.
* Hệ sinh thái vững chắc: Vue sở hữu một hệ sinh thái cực kỳ vững chắc và đầy đủ nên có thể cung cấp một số add – ons rất hữu ích cho việc xây dựng một ứng dụng fontend điển hình nhất.
* Core Vue linh hoạt: VueJs tập trung vào việc render giao diện cho người dùng và các tương tác của nó vì vậy, nó sẽ cung cấp tối thiểu những tính năng cần thiết cho việc thiết kế và xây dựng kiến trúc, đồng thời sẽ giúp bạn loại bỏ được các tính năng không cần thiết ra khỏi thư viện trong lõi VueJs và đảm bảo cho framework nhỏ gọn và mềm dẻo hơn.

### **Cơ sở dữ liệu MySQL**

#### **MySQL là gì?**

****

Hình 4. MySQL

MySQL là 1 hệ thống quản trị về cơ sở dữ liệu với mã nguồn mở (được gọi tắt là RDBMS) và đang hoạt động theo mô hình dạng client-server. Đối với RDBMS – Relational Database Management System thì MySQL đã được tích hợp apache và PHP.

Được phát hành chính thức từ thập niên 90, MySQL hiện đang quản lý dữ liệu qua những cơ sở dữ liệu, với mỗi một cơ sở dữ liệu hoàn toàn có thể có rất nhiều những bản quan hệ có chứa dữ liệu. Ngoài ra, MySQL cũng có cùng 1 cách thức truy xuất cũng như mã lệnh tương tự cùng với ngôn ngữ SQL.

#### **Ưu điểm của MySQL?**

* **Nhanh chóng**: Nhờ vào việc đưa ra một số những tiêu chuẩn và cho phép MySQL làm việc hiệu quả cũng như tiết kiệm chi phí, giúp gia tăng tốc độ thực thi.
* **Mạnh mẽ và khả năng mở rộng:** MySQL hoàn toàn có thể xử lý số lượng lớn dữ liệu và đặc biệt hơn thế nữa thì nó còn có thể mở rộng nếu như cần thiết.
* **Đa tính năng:** Ưu điểm MySQL là gì? MySQL hiện đang hỗ trợ nhiều những chức năng SQL rất được mong chờ từ 1 hệ quản trị CSDL quan hệ cả gián tiếp cũng như trực tiếp.
* **Độ bảo mật cao:** MySQL là gì? Hiện tại nó đang rất thích hợp cho những ứng dụng truy cập CSDL thông qua internet khi sở hữu rất nhiều những tính năng về bảo mật và thậm chí là đang ở cấp cao.
* **Dễ dàng sử dụng:**MySQL đang là cơ sở dữ liệu dễ sử dụng, ổn định, tốc độ cao và hoạt động trên rất nhiều những hệ điều hành đang cung cấp 1 hệ thống lớn những hàm tiện ích rất mạnh.

#### **Các yếu tố giúp MySQL nổi bật**

* MySQL  hiện đang là CSDL có tốc độ ổn định, khá cao và đặc biệt là rất dễ sử dụng, hoạt động được trên nhiều những hệ điều hành.
* Không những có thể sử dụng trên nhiều ứng dụng được, tính bảo mật mạnh mà nó còn được sử dụng miễn phí hoàn toàn.
* MySQL không những bổ trợ cho Perl và PHP, mà còn bổ trợ cho rất nhiều những ngôn ngữ khác. Đây chính là nơi để thực hiện lưu trữ tất cả những thông tin trên những trang Web đang được viết bằng Perl hoặc PHP.

## Công cụ sử dụng

* Visual Studio Code: phục vụ mục đích code asp.net.
* VS Code: phục vụ mục đích code VueJS.
* MySQL: cơ sở dữ liệu dùng để lưu trữ dữ liệu.
* dbForge Studio: hệ quản trị cơ sở dữ liệu.
* Postman: sử dụng để thử nghiệm API.
* Chrome DevTools: gỡ lỗi khi phát triển web.
* Case Studio: thiết kế mô hình thực thể liên kết.
* Rational Rose: thiết kế hệ thống.

# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Phân tích hệ thống

### **Vai trò của người dùng**

* Hệ thống chia làm 3 nhóm người dùng chính:
  + Khách hàng
  + Nhân vien
  + Người quản trị
* Mô tả nhóm người dùng.
* Khách hàng (thành viên của trang web): những khách hàng đã có tài khoản đăng nhập trên trang web. Khách hàng khi mua hàng trên trang web thì thông tin của khách hàng đã được lưu lại trong cơ sở dữ liệu và khi người dùng muốn mua hàng chỉ cần đăng nhập bằng tài khoản thông tin của khách hàng sẽ hiển thị và khách hàng chỉ cần tiến hành mua.
* Nhân viên: là người quản lý trang web thay cho người quản trị nhưng sẽ bị giới hạn các chức năng.
* Người quản trị: là người quản lý trang web và được truy cập vào các chức năng quản lý của hệ thống.

### **Yêu cầu chức năng**

* Thành viên của trang web:
  + Đăng nhập
  + Đăng ký
  + Đăng xuất
  + Xem chi tiết sản phẩm
  + Lọc sản phẩm
  + Đặt hàng
  + Quản lý đơn hàng
  + Sửa thông tin cá nhân
  + Quên mật khẩu
  + Đánh giá sản phẩm
  + Xem danh mục sản phẩm
  + Sắp xếp sản phẩm
* Quản trị viên:
  + Sửa thông tin cá nhân
  + Quản lý tài khoản người dùng
  + Quản lý danh mục sản phẩm
  + Quản lý sản phẩm
  + Quản lý góp ý của người dùng
  + Quản lý danh sách đơn hàng

### **Yêu cầu phi chức năng**

* Giao diện trang web được thiết kế dễ sử dụng và có tính thẩm mỹ cao.
* Hiệu năng: tốc độ, khả năng và độ tin cậy của hệ thống.
* Website bảo mật phân quyền để thực hiện các chức năng của hệ thống.

## Biểu đồ use case tổng quát

Diagram

Description automatically generated

Hình 5. Biểu đồ use case tổng quát

## Mô hình thực thể liên kết

Diagram, schematic, timeline

Description automatically generated

Hình 6. Biều đồ thực thể liên kết

## Đặc tả use case

### **Use case đăng ký**

a) Biểu đồ use case

Shape

Description automatically generated with medium confidence

Hình 7. Biểu đồ phân rã đăng ký

b) Đặc tả use case

Mô tả: use case này cho phép khách hàng đăng kí tài khoản trên hệ thống.

Luồng sự kiện chính:

1. Người dùng chọn mục đăng ký.
2. Form đăng ký hiển thị.
3. Người dùng nhập thông tin cần thiết vào form đăng ký.
4. Nhất nút đăng ký.
5. Hệ thống thông báo kết quả quá trình nhập thông tin cá nhân. Nếu thông tin nhập không chính xác thì thực hiện luồng nhánh A1. Nếu nhập chính xác thì thực hiện bước 6.
6. Hệ thống cập nhật thông tin của người dùng vào danh sách người dùng trong cơ sở dữ liệu.
7. Usecase kết thúc.

Luồng nhánh A1: Quá trình nhập thông tin không chính xác.

1. Hệ thống thông báo thông tin nhập không chính xác.
2. Hệ thống yêu cầu Người dùng nhập lại thông tin.
3. Nếu người dùng muốn đăng ký thì quay về bước 2 của luồng sự kiện chính. Nếu không đồng ý thì usecase kết thúc.

c) Biểu đồ trình tự

Graphical user interface, diagram

Description automatically generated

Hình 8. Biều đồ trình tự đăng ký

### **Use case đăng nhập**

a) Biểu đồ use case

A picture containing diagram

Description automatically generated

Hình 9. Biều đồ phân rã đăng nhập

b) Đặc tả use case

Mô tả: Cho phép người dùng đăng nhập vào hệ thống.

Tiền điều kiện: Người dùng chưa đăng nhập vào hệ thống.

Luồng sự kiện chính:

1. Người dùng chọn chức năng đăng nhập.
2. Form đăng nhập hiển thị.
3. Nhập user, mật khẩu vào form đăng nhập.
4. Hệ thống kiểm tra thông tin.
5. Nếu việc đăng nhập thành công thì hệ thống chuyển về trang chủ. Nếu người dùng nhập sai thông tin thì chuyển sang luồng nhánh A1.

Luồng nhánh A1: Người dùng đăng nhập không thành công

1. Hệ thống thông báo quá trình đăng nhập không thành công.
2. Chọn: Đăng ký hay nhập lại. Nếu chọn đăng ký thì hệ thống sẽ thực hiện usecase đăng ký từ bước 2 của luồng sự kiện chính. Nếu chọn nhập lại thì thực hiện bước 3.
3. Hệ thống yêu cầu người dùng nhập lại thông tin.
4. Nếu người dùng đồng ý thì quay về bước 2 của luồng sự kiện chính, nếu không đồng ý thì usecase kết thúc.

Hậu điều kiện: Người dùng đăng nhập thành công và có thể sử dụng các chức năng tương.

c) Biểu đồ trình tự

Diagram

Description automatically generated

Hình 10. Biểu đồ trình tự đăng nhập

### **Use case quên mật khẩu**

a) Biểu đồ use case

A picture containing shape

Description automatically generated

Hình 11. Biều đồ phân rã quên mật khẩu

b) Đặc tả use case

Mô tả: Cho phép người dùng đặt lại mật khẩu khi quên mật khẩu đăng nhập.

Tiền điều kiện: Người dùng chưa đăng nhập vào hệ thống.

Luồng sự kiện chính:

1. Người dùng chọn mục quên mật khẩu trên form đăng nhập
2. Hệ thống sẽ hiển thị form và yêu cầu người dùng nhập email đăng ký.
3. Nếu email có tồn tại trong cơ sở dữ liệu thì hệ thống sẽ gửi mã đến email qua gmail. Nếu email không tồn tại thì sẽ thông báo lỗi.
4. Người dùng nhập mã nhận được để tiến hành đặt lại mật khẩu.
5. Nếu mã khớp với mã trong email và trong thời gian quy định thì hệ thống sẽ hiển thị form đặt lại mật khẩu, người dùng nhập mật khẩu mới . Ngược lại, hệ thống thông báo lỗi.
6. Usecase kết thúc.

c) Biểu đồ trình tự

Graphical user interface

Description automatically generated

Hình 12. Biểu đồ trình tự quên mật khẩu

### **Use case quản lý tài khoản**

a) Biểu đồ use case

Diagram, schematic

Description automatically generated

Hình 13. Biểu đồ phân rã quản lý tài khoản

b) Đặc tả use case

Mô tả: Cho phép người dùng thay đổi mật khẩu của tài khoản.

Tiền điều kiện: Người dùng bắt buộc đăng nhập vào hệ thống.

Luồng sự kiện chính:

1. Người dùng chọn mục quản lý tài khoản.
2. Hệ thống sẽ hiển thị ra thông tin tài khoản người dùng.
3. Người dùng có thể xem và thay đổi dữ liệu.
4. Nếu người dùng muốn thay đổi dữ liệu thì sẽ điền lại các thông tin vào form và nhấn xác nhận để tiến hành yêu cầu thay đổi.
5. Hệ thống kiểm tra và nếu thông tin hợp lệ thì hiển thị thông báo thành công. Ngược lại thông báo lỗi. Usecase kết thúc.

Hậu điều kiện: Người dùng thay thông tin thành công.

c) Biểu đồ trình tự

Diagram

Description automatically generated

Hình 14. Biểu đồ trình tự quản lý tài khoản

### **Use case xem thông tin sản phẩm**

a) Biểu đồ use case

Diagram

Description automatically generated

Hình 15. Biểu đồ phân rã thông tin sản phẩm

b) Đặc tả use case

Mô tả: Cho phép người dùng xem thông tin về các sản phẩm có trong cửa hàng.

Luồng sự kiện chính:

1. Người dùng chọn sản phẩm cần xem.
2. Form xem thông tin sản phẩm xuất hiện, hệ thống hiển thị thông tin về sản phẩm đã chọn.
3. Người quản lý, nhân viên, khách hàng xem thông tin chi tiết về sản phẩm được hiển thị.
4. Usecase kết thúc.

c) Biểu đồ trình tự



Hình 16. Biểu đồ trình tự xem thông tin sản phẩm

### **Use case quản lý giỏ hàng**

a) Biểu đồ use case

Diagram

Description automatically generated

Hình 17. Biểu đồ phân rã quản lý giỏ hàng

b) Đặc tả use case

Mô tả: Cho phép người dùng quản lý giỏ hàng của mình.

Tiền điều kiện: Người dùng bắt buộc đăng nhập vào hệ thống.

Luồng sự kiện chính:

1. Người dùng chọn giỏ hàng.
2. Form xem thông tin giỏ hàng xuất hiện, hệ thống hiển thị thông tin về giỏ hàng của người dùng hiện tại.
3. Người dùng xem thông tin chi tiết về giỏ hàng được hiển thị và có thể thêm, xóa sản phẩm trong giỏ hàng.
4. Usecase kết thúc.

c) Biểu đồ trình tự

Diagram

Description automatically generated

Hình 18. Biểu đồ trình tự quản lý giỏ hàng

### **Use case mua hàng**

a) Biểu đồ use case

Shape

Description automatically generated with low confidence

Hình 19. Biểu đồ phân ra mua hàng

b) Đặc tả use case

Mô tả: Cho phép người dùng tạo đơn hàng để tiến hành thanh toán giỏ hàng sản phẩm.

Tiền điều kiện: Người dùng bắt buộc đăng nhập vào hệ thống.

Luồng sự kiện chính:

1. Người dùng nhấn mua hàng để tạo đơn hàng.
2. Form thanh toán xuất hiện, hệ thống hiển thị đơn hàng và một số thông tin mặc định của đơn hàng.
3. Khách hàng nhập tiếp các thông tin: họ tên, phương thức thanh toán, địa chỉ giao hàng, địa chỉ email, ….
4. Chọn đặt hàng. Nếu đơn hàng đã nhập đúng thì thực hiện bước 5. Nếu sai thì thực hiện luồng sự kiện rẽ nhánh A1.
5. Hệ thống kiểm tra số tài khoản của khách hàng và phương thức thanh toán. Nếu hợp lệ thì thực hiện bước 6. Nếu sai thì thực hiện luồng sự kiện rẽ nhánh A2.
6. Hệ thống thông báo giao dịch thành công.
7. Usecase kết thúc.

Luồng sự kiện A1: đơn hàng nhập sai.

1. Hệ thống thông báo việc tạo đơn hàng không thành công.
2. Hệ thống hiển thị form nhập thông tin và các trường nhập sai.
3. Người dùng nhập lại thông tin, nếu ngưng giao dịch thì thực hiện luồng sự kiện rẽ nhánh A3.
4. Quay lại bước 4 của luồng sự kiện chính.

Luồng sự kiện A2: Tài khoản không hợp lệ.

1. Hệ thống thông báo phương thức thanh toán và tài khoản của khách hàng không hợp lệ.
2. Người dùng nhập lại thông tin nếu muốn tiếp tục mua hàng, nếu muốn ngưng giao dịch thì thực hiện luồng sự kiện rẽ nhánh A3.
3. Quay lại bước 4 của luồng sự kiện chính.

Luồng sự kiện A3: Người dùng hủy đơn hàng.

1. Người dùng nhấn hủy bỏ đơn hàng.
2. Hệ thống thực hiện hủy đơn hàng và đưa người dùng về trang chủ.
3. Usecase kết thúc.

Tiền điều kiện: Người dùng đã chọn sản phẩm cần mua.

Hậu điều kiện: Đơn hàng được đặt thành công.

c) Biểu đồ trình tự

Diagram

Description automatically generated

Hình 20. Biểu đồ trình tự mua hàng

### **Use case quản lý đơn hàng**

a) Biểu đồ use case

Diagram, schematic

Description automatically generated

Hình 21. Biểu đồ phân rã quản lý đơn hàng

b) Đặc tả use case

Mô tả: use case này cho phép người dùng xem hoặc hủy đơn hàng online trên hệ thống.

Tiền điều kiện: Người dùng bắt buộc đăng nhập vào hệ thống.

Luồng sự kiện chính:

1. Use case này bắt đầu khi người dùng nhấn vào đơn hàng trong menu thông tin cá nhân.
2. Hệ thống lấy thông tin người dùng và lọc danh sách đơn hàng mà khách hàng đã đặt.
3. Người dùng xem thông tin đơn hàng đã đặt và nhấn hủy đơn hàng tại đơn hàng muốn hủy.
4. Use case kết thúc.

Tiền điều kiện: Người dùng cần đăng nhập.

c) Biểu đồ trình tự

Diagram

Description automatically generated

Hình 22. Biểu đồ trình tự quản lý đơn hàng

### **Use case sắp xếp sản phẩm**

a) Biểu đồ use case

Diagram

Description automatically generated

Hình 23. Biểu đồ phân rã sắp xếp sản phẩm

b) Đặc tả use case

Mô tả: use case này cho phép người dùng lựa chọn thứ tự ưu tiên theo các tiêu chí của sản phẩm.

Luồng sự kiện chính:

1. Người dùng chọn vào combobox để tiến hành sắp xếp sản phẩm.
2. Người dùng chọn các loại ưu tiên.
3. Danh sách sản phẩm sẽ hiển thị theo loại ưu tiên người dùng chọn.
4. Usecase kết thúc.

Hậu điều kiện: Sản phẩm hiển thị theo thứ tự ưu tiên người dùng chọn.

c) Biểu đồ trình tự

Diagram

Description automatically generated

Hình 24. Biểu đồ trình tự sắp xếp sản phẩm

### **Use case lọc sản phẩm theo giá bán**

a) Biểu đồ use case

Diagram

Description automatically generated

Hình 25. Biểu đồ phân rã lọc sản phẩm theo giá bán

b) Đặc tả use case

Mô tả: Cho phép người dùng lọc sản phẩm theo khoảng giá mà người dùng nhập.

Luồng sự kiện chính:

1. Người dùng nhập khoảng giá.
2. Hệ thống hiển thị danh sách sản phẩm trong khoảng giá người dùng nhập.
3. Usecase kết thúc.

Hậu điều kiện: Sản phẩm hiển thị theo thứ tự ưu tiên người dùng chọn.

c) Biểu đồ trình tự

Graphical user interface, diagram

Description automatically generated

Hình 26. Biểu đồ trình tự lọc sản phẩm theo giá bán

### **Use case xem danh mục sản phẩm**

a) Biểu đồ use case

A picture containing application

Description automatically generated

Hình 27. Biểu đồ phân rã xem danh mục sản phẩm

b) Đặc tả use case

Mô tả: Cho phép người dùng xem các danh mục cha, con.

Luồng sự kiện chính:

1. Người dùng di chuột vào các danh mục tên được hiển thị trên thanh navbar.
2. Hệ thống sẽ hiển thị danh mục theo cấp cha con.
3. Người dùng bấm chọn danh mục để hiển thị sản phẩm theo danh mục chọn.

Hậu điều kiện: Sản phẩm hiển thị dựa theo danh mục người dùng chọn.

c) Biểu đồ trình tự

Graphical user interface

Description automatically generated with medium confidence

Hình 28. Biểu đồ phân rã xem danh mục sản phẩm

### **Use case đánh giá sản phẩm**

a) Biểu đồ use case

Diagram, schematic

Description automatically generated

Hình 29. Biểu đồ phân rã đánh giá sản phẩm

b) Đặc tả use case

Mô tả: Cho phép người dùng đánh giá về sản phẩm.

Tiền điều kiện: Người dùng bắt buộc đăng nhập vào hệ thống.

Luồng sự kiện chính:

1. Người dùng chọn sản phẩm muốn đánh giá.
2. Hệ thống sẽ hiển thị chi tiết sản phẩm.
3. Người dùng chọn mục đánh giá.
4. Hệ thống hiển thị form đánh giá.
5. Người dùng nhập form và bấm đánh giá để gửi đánh giá.
6. Usecase kết thúc.

Hậu điều kiện: Lời đánh giá của người dùng sẽ hiển thị trong mục đánh giá của sản phẩm để mọi người đều có thể xem.

c) Biểu đồ trình tự

Graphical user interface

Description automatically generated

Hình 30. Biểu đồ trình tự đánh giá sản phẩm

### **Use case đóng góp ý kiến cho cửa hàng**

a) Biểu đồ use case

Shape

Description automatically generated with medium confidence

Hình 31. Biểu đồ phân rã đóng góp ý kiến cho cửa hàng

b) Đặc tả use case

Mô tả: Cho phép người dùng đánh giá về sản phẩm.

Luồng sự kiện chính:

1. Người dùng chọn phần liên hệ trên thanh navbar.
2. Hệ thống hiển thị thông tin liên hệ và form đóng góp ý kiến.
3. Người dùng có thể đóng góp ý kiến bằng cách nhập vào form và bấm gửi.
4. Usecase kết thúc.

Hậu điều kiện: Ý kiến đóng góp của khách hàng được ghi nhận.

c) Biểu đồ trình tự

Diagram

Description automatically generated

Hình 32. Biểu đồ trình tự đóng góp ý kiến cho cửa hàng

### **Use case quản lý sản phẩm**

a) Biểu đồ use case

Diagram, schematic

Description automatically generated

Hình 33. Biểu đồ phân rã quản lý sản phẩm

b) Đặc tả use case

Mô tả: Cho phép người quản lý thêm, xóa, thay đổi thông tin của các sản phẩm

Tiền điều kiện: Người dùng bắt buộc đăng nhập vào hệ thống với vai trò quản trị viên.

Luồng sự kiện chính:

1. Người quản lý chọn kiểu tác động lên sản phẩm: thêm sản phẩm, thay đổi thông tin sản phẩm, xóa sản phẩm.

A. Thêm sản phẩm:

1. Hệ thống hiển thị form nhập thông tin sản phẩm.
2. Người quản lý nhập thông tin sản phẩm.
3. Nhấn nút lưu thông tin.
4. Nếu nhập thành công thì thực hiện bước 5. Nếu sai thực hiện luồng sự kiện rẽ nhánh A1.
5. Lưu thông tin sản phẩm.

B. Thay đổi thông tin sản phẩm

1. Hệ thống hiển thị form sửa thông tin của sản phẩm.
2. Người quản lý nhập các thông tin cần thay đổi.
3. Nhấn nút lưu thông tin.
4. Nếu việc thay đổi thành công thì thực hiện bước 5. Nếu sau thực hiện luồng sự kiện rẽ nhánh A1.
5. Lưu thông tin sản phẩm.

C. Xóa sản phẩm

1. Người quản trị chọn sản phẩm cần xóa.
2. Nhấn nút xóa để thực hiện loại bỏ sản phẩm.
3. Hệ thống hiển thị thông báo xác nhận xóa. Nếu người quản trị đồng ý thì thực hiện bước 4. Nếu không đồng ý thì thực hiện bước 5.
4. Thông báo sản phẩm đã xóa.
5. Hệ thống hiển thị lại danh sách sản phẩm.
6. Usecase kết thúc.

Luồng sự kiện A1:

1. Hệ thống thông báo việc nhập dữ liệu không hợp lệ.
2. Người quản lý nhập lại thông tin.
3. Quay lai bước 3 của luồng sự kiện chính.

Tiền điều kiện: Người quản lý phải đăng nhập vào hệ thống

Hậu điều kiện: Các thông tin về sản phẩm được cập nhật vào cơ sở dữ liệu

c) Biểu đồ trình tự

Diagram

Description automatically generated

Diagram

Description automatically generated with medium confidence

Hình 34. Biểu đồ trình tự quản lý sản phẩm

### **Use case quản lý người dùng**

a) Biểu đồ use case

Diagram

Description automatically generated

Hình 35. Biểu đồ phân rã quản lý người dùng

b) Đặc tả use case

Mô tả: use case này cho phép quản trị viên quản lý thông tin người dùng, thêm, sửa, xóa thông tin người dùng.

Tiền điều kiện: Người dùng bắt buộc đăng nhập vào hệ thống với vai trò quản trị viên.

Luồng sự kiện chính:

1. Use case này bắt đầu khi quản trị viên nhấn chọn quản lý người dùng.
2. Hệ thống sẽ lấy dữ liệu từ cơ sở dữ liệu và hiển thị lên màn hình, có các nút nút sửa, xóa.
3. Quản trị viên thực hiện chức năng thêm, khi nhấm nút thêm hệ thống sẽ tải trang thêm người dùng để quản trị viên nhập các thông tin đầy đủ của người dùng. Sau khi nhập đầy đủ thông tin xong nhấn nút lưu lại và việc thêm người dùng đã thực hiện xong.
4. Quản trị viên nhấn nút sửa của người dùng muốn sửa thông tin, hệ thống sẽ lấy thông tin của người dùng đã chọn và hiển thị lên màn hình. Quản trị viên sẽ nhập các thông tin cần thay đổi sau đó nhấn nút cập nhật và quá trình sửa thông tin người dùng đã hoàn tất.
5. Quản trị viên nhấn nút xóa trên người dùng muốn xóa, sau khi nhấn nút xóa hệ thống sẽ lấy thông tin của người dùng đã chọn và xóa khỏi cơ sở dữ liệu.
6. Usecase kết thúc.

c) Biểu đồ trình tự

Diagram

Description automatically generated with low confidence

Diagram

Description automatically generated

Hình 36. Biểu đồ trình tự quản lý người dùng

### **Use case quản lý đơn hàng**

a) Biểu đồ use case

Diagram, schematic

Description automatically generated

Hình 37. Biểu đồ phân rã quản lý đơn hàng

b) Đặc tả use case

Mô tả: Cho phép quản trị viên quản lý thông tin đơn hàng.

Tiền điều kiện: Người dùng bắt buộc đăng nhập vào hệ thống với vai trò quản trị viên.

Luồng sự kiện chính:

1. Use case này bắt đầu khi quản trị viên nhấn chọn quản lý đơn hàng.
2. Hệ thống sẽ lấy dữ liệu từ cơ sở dữ liệu và hiển thị lên màn hình, có các nút nút sửa, xóa.
3. Quản trị viên nhấn nút sửa của đơn hàng muốn sửa, hệ thống sẽ lấy thông tin đơn hàng đã chọn và hiển thị lên màn hình. Quản trị viên sẽ thực hiện cập nhật các thông tin cần thay đổi sau đó nhấn nút lưu và quá trình sửa thông tin đã hoàn tất.
4. Quản trị viên nhấn nút xóa trên đơn hàng muốn xóa, hệ thống sẽ hiển thị thông báo xác nhận xóa. Nếu người dùng chọn xóa thì sẽ xóa thông tin đơn hàng khỏi cơ sở dữ liệu.

Usecase kết thúc.

c) Biểu đồ trình tự

Diagram

Description automatically generated

Hình 38. Biểu đồ trình tự quản lý đơn hàng

### **Use case quản lý danh mục**

a) Biểu đồ use case

Diagram, schematic

Description automatically generated

Hình 39. Biểu đồ phân rã quản lý danh mục

b) Đặc tả use case

Mô tả: use case này cho phép quản trị viên quản lý thông tin danh mục, thêm, sửa, xóa thông tin danh mục.

Tiền điều kiện: Người dùng bắt buộc đăng nhập vào hệ thống với vai trò quản trị viên.

Luồng sự kiện chính:

1. Use case này bắt đầu khi quản trị viên nhấn chọn quản lý danh mục.
2. Hệ thống sẽ lấy dữ liệu từ cơ sở dữ liệu và hiển thị lên màn hình, có các nút danh sách ghế danh mục, nút sửa, xóa danh mục.
3. Quản trị viên thực hiện chức năng thêm, khi nhấm nút thêm hệ thống sẽ tải trang thêm danh mục để quản trị viên nhập các thông tin đầy đủ của danh mục. Sau khi nhập đầy đủ thông tin xong nhấn nút lưu lại và việc thêm danh mục đã thực hiện xong.
4. Quản trị viên nhấn nút sửa của danh mục muốn sửa thông tin, hệ thống sẽ lấy thông tin của danh mục đã chọn và hiển thị lên màn hình. Quản trị viên sẽ nhập các thông tin cần thay đổi sau đó nhấn nút cập nhật và quá trình sửa thông tin danh mục đã hoàn tất.
5. Quản trị viên nhấn nút xóa trên danh mục muốn xóa, sau khi nhấn nút xóa hệ thống sẽ lấy thông tin của danh mục đã chọn và xóa khỏi cơ sở dữ liệu.
6. Usecase kết thúc.

c) Biểu đồ trình tự

Diagram

Description automatically generated

A picture containing diagram

Description automatically generated

Hình 40. Biểu đồ trình tự quản lý danh mục

### **Use case quản lý góp ý cửa hàng**

a) Biểu đồ use case

Diagram

Description automatically generated

Hình 41. Biểu đồ phân rã quản lý góp ý cửa hàng

b) Đặc tả use case

Mô tả: Cho phép quản trị viên quản lý thông tin góp ý cửa hàng của người dùng.

Tiền điều kiện: Người dùng bắt buộc đăng nhập vào hệ thống với vai trò quản trị viên.

Luồng sự kiện chính:

1. Use case này bắt đầu khi quản trị viên nhấn chọn quản lý góp ý.
2. Hệ thống sẽ lấy dữ liệu từ cơ sở dữ liệu và hiển thị lên màn hình, có các nút nút gửi email, xóa.
3. Quản trị viên nhấn nút gửi email trên góp ý, hệ thống hiển thị form để quản trị viên nhập các thông tin cần thiết. Quản trị viên nhấn gửi để gửi email.
4. Quản trị viên nhấn nút xóa trên góp ý, hệ thống sẽ hiển thị thông báo xác nhận xóa. Nếu người dùng chọn xóa thì sẽ xóa thông tin đơn hàng khỏi cơ sở dữ liệu.
5. Usecase kết thúc.

c) Biểu đồ trình tự

Diagram

Description automatically generated

Diagram

Description automatically generated with medium confidence

Hình 42. Biểu đồ trình tự quản lý góp ý cửa hàng

### **Use case quản lý thống kê**

Mô tả: Cho phép quản trị viên xuất excel.

Tiền điều kiện: Người dùng bắt buộc đăng nhập vào hệ thống với vai trò quản trị viên.

Luồng sự kiện:

1. Quản trị viên nhấn nút xuất excel trong các phân hệ quản lý.
2. Hệ thống thực hiện xuất excel và tải về máy.
3. Usecase kết thúc.

Hậu điều kiện: Quản trị viên nhận được tệp excel.

## Thiết kế hệ thống

### **Thiết kế cơ sở dữ liệu**

#### **Biểu đồ mô tả cấu trúc cơ sở dữ liệu**

Diagram, schematic

Description automatically generated

#### **Mô tả cơ sở dữ liệu**

Bảng 1. Bảng Category để lưu thông tin danh mục sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu | Độ dài | Khóa | Giải thích |
| CategoryId | Char | 36 | PK | Id danh mục |
| CategoryCode | DECIMAL(5, 0) |  |  | Mã danh mục |
| CategoryName | VarChar | 100 |  | Tên danh mục |
| Description | Text |  |  | Mô tả |
| ParentId | DECIMAL(5, 0) |  |  | ID danh mục cha |
| Code | VarChar | 100 |  | Mã code phân nhánh |
| CreatedDate | DateTime |  |  | Ngày tạo |
| CreatedBy | VarChar | 100 |  | Người tạo |
| ModifiedDate | DateTime |  |  | Ngày sửa |
| ModifiedBy | VarChar | 100 |  | Người sửa |

Bảng 2. Bảng Size để lưu thông size sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu | Độ dài | Khóa | Giải thích |
| SizeId | Char | 36 | PK | Id size |
| SizeName | VarChar | 10 |  | Số size |
| CreatedDate | DateTime |  |  | Ngày tạo |
| CreatedBy | VarChar | 100 |  | Người tạo |
| ModifiedDate | DateTime |  |  | Ngày sửa |
| ModifiedBy | VarChar | 100 |  | Người sửa |

Bảng 3. Bảng Evaluate để lưu thông tin đánh giá của sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu | Độ dài | Khóa | Giải thích |
| EvaluateId | Char | 36 | PK | Id đánh giá |
| Star | Int |  |  | Số sao |
| FullName | VarChar | 100 |  | Tên người đánh giá |
| Email | VarChar | 100 |  | Email |
| Comment | Text |  |  | Nhận xét |
| ShoeId | Char | 36 | FK | Id giày |
| CreatedDate | DateTime |  |  | Ngày tạo |
| CreatedBy | VarChar | 100 |  | Người tạo |

Bảng 4. Bảng Shoe để lưu thông tin sản phẩm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu | Độ dài | Khóa | Giải thích |
| ShoeId | Char | 36 | PK | Id giày |
| ShoeCode | VarChar | 255 |  | Mã giày |
| ShoeName | VarChar | 255 |  | Tên giày |
| Descriptions | Text |  |  | Mô tả |
| ImgName | Text |  |  | Source ảnh |
| Price | DECIMAL(10, 0) |  |  | Giá bán |
| Discount | Int |  |  | Giảm giá |
| Material | VarChar | 100 |  | Chất liệu |
| Size | Text |  |  | Size giày |
| CategoyId | Char | 36 | FK | Id danh mục |
| CreatedDate | DateTime |  |  | Ngày tạo |
| CreatedBy | VarChar | 100 |  | Người tạo |
| ModifiedDate | DateTime |  |  | Ngày sửa |
| ModifiedBy | VarChar | 100 |  | Người sửa |

Bảng 5. Bảng Customer để lưu thông tin người dùng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu | Độ dài | Khóa | Giải thích |
| CustomerId | Char | 36 | PK | Id người dùng |
| UserName | VarChar | 100 |  | Tài khoản |
| Password | VarChar | 100 |  | Mật khẩu |
| FullName | VarChar | 50 |  | Họ và tên |
| Email | VarChar | 100 |  | Email |
| ImgName | Text |  |  | Source ảnh |
| PhoneNumber | VarChar | 50 |  | Số điện thoại |
| Address | VarChar | 100 |  | Địa chỉ |
| Role | Int |  |  | Chức vụ |
| Token | VarChar | 50 |  | Mã lấy lại mật khẩu |
| TokenDate | DateTime |  |  | Thời hạn của mã |
| CreatedDate | DateTime |  |  | Ngày tạo |
| CreatedBy | VarChar | 100 |  | Người tạo |
| ModifiedDate | DateTime |  |  | Ngày sửa |
| ModifiedBy | VarChar | 100 |  | Người sửa |

Bảng 6. Bảng FeedBack để lưu thông tin đóng góp ý kiến của người dùng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu | Độ dài | Khóa | Giải thích |
| FeedBackId | Char | 36 | PK | Id đóng góp |
| FullName | VarChar | 100 |  | Tên người giử |
| Email | VarChar | 50 |  | Email |
| PhoneNumber | VarChar | 20 |  | Số điện thoại |
| Address | VarChar | 255 |  | Địa chỉ |
| Description | Text |  |  | Lời nhắn |
| Status | Int |  |  | Trạng thái |
| Reply | VarChar | 255 |  | Trả lời từ cửa hàng |
| CreatedDate | DateTime |  |  | Ngày tạo |
| CreatedBy | VarChar | 100 |  | Người tạo |
| ModifiedDate | DateTime |  |  | Ngày sửa |
| ModifiedBy | VarChar | 100 |  | Người sửa |

Bảng 7. Bảng Cart để lưu thông tin giỏ hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu | Độ dài | Khóa | Giải thích |
| CartId | Char | 36 | PK | Id giỏ hàng |
| CustomerId | Char | 36 | FK | Id người dùng |
| CreatedDate | DateTime |  |  | Ngày tạo |
| CreatedBy | VarChar | 100 |  | Người tạo |

Bảng 8. Bảng CartDetail để lưu thông tin chi tiết giỏ hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu | Độ dài | Khóa | Giải thích |
| CartDetailId | Char | 36 | PK | Id chi tiết giỏ hàng |
| ShoeId | Char | 36 | FK | Id giày |
| Amount | DECIMAL(8, 2) |  |  | Số lượng |
| Size | VarChar | 10 |  | Size |
| CartId | Char | 36 | FK | Id giỏ hàng |
| CreatedDate | DateTime |  |  | Ngày tạo |
| CreatedBy | VarChar | 100 |  | Người tạo |
| ModifiedDate | DateTime |  |  | Ngày sửa |
| ModifiedBy | VarChar | 100 |  | Người sửa |

Bảng 9. Bảng Bill để lưu thông tin đơn hàng

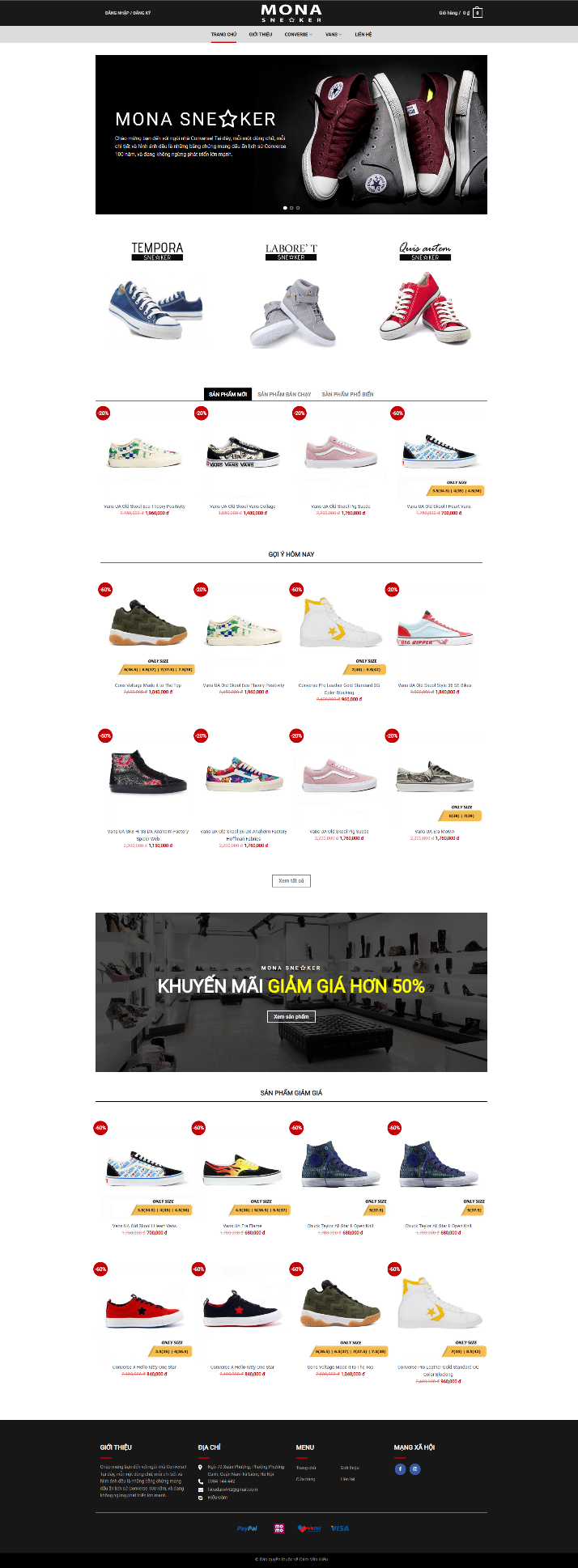
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu | Độ dài | Khóa | Giải thích |
| BillId | Char | 36 | PK | Id hóa đơn |
| FullName | VarChar | 100 |  | Họ và tên |
| Email | VarChar | 100 |  | Email |
| Address | VarChar | 255 |  | Địa chỉ |
| PhoneNumber | VarChar | 20 |  | Số điện thoại |
| TotalPrice | DECIMAL(10, 2) |  |  | Tổng tiền |
| Status | Int |  |  | Trạng thái đơn hàng |
| IsPayment | Int |  |  | Đã thanh toán chưa |
| PaymentMethod | Int |  |  | Phương thức thanh toán |
| CustomerId | Char | 36 | FK | Id người dùng |
| IsCancel | Int |  |  | Người dùng có hủy đơn |
| Country | VarChar | 50 |  | Đất nước |
| City | VarChar | 50 |  | Thành phố / Tỉnh |
| Note | Text |  |  | Ghi chú |
| PaymentId | Text |  |  | Id thanh toán online |
| Reason | VarChar | 100 |  | Lý do |
| CreatedDate | DateTime |  |  | Ngày tạo |
| CreatedBy | VarChar | 100 |  | Người tạo |

Bảng 10. Bảng BillDetail để lưu thông tin chi tiết đơn hàng

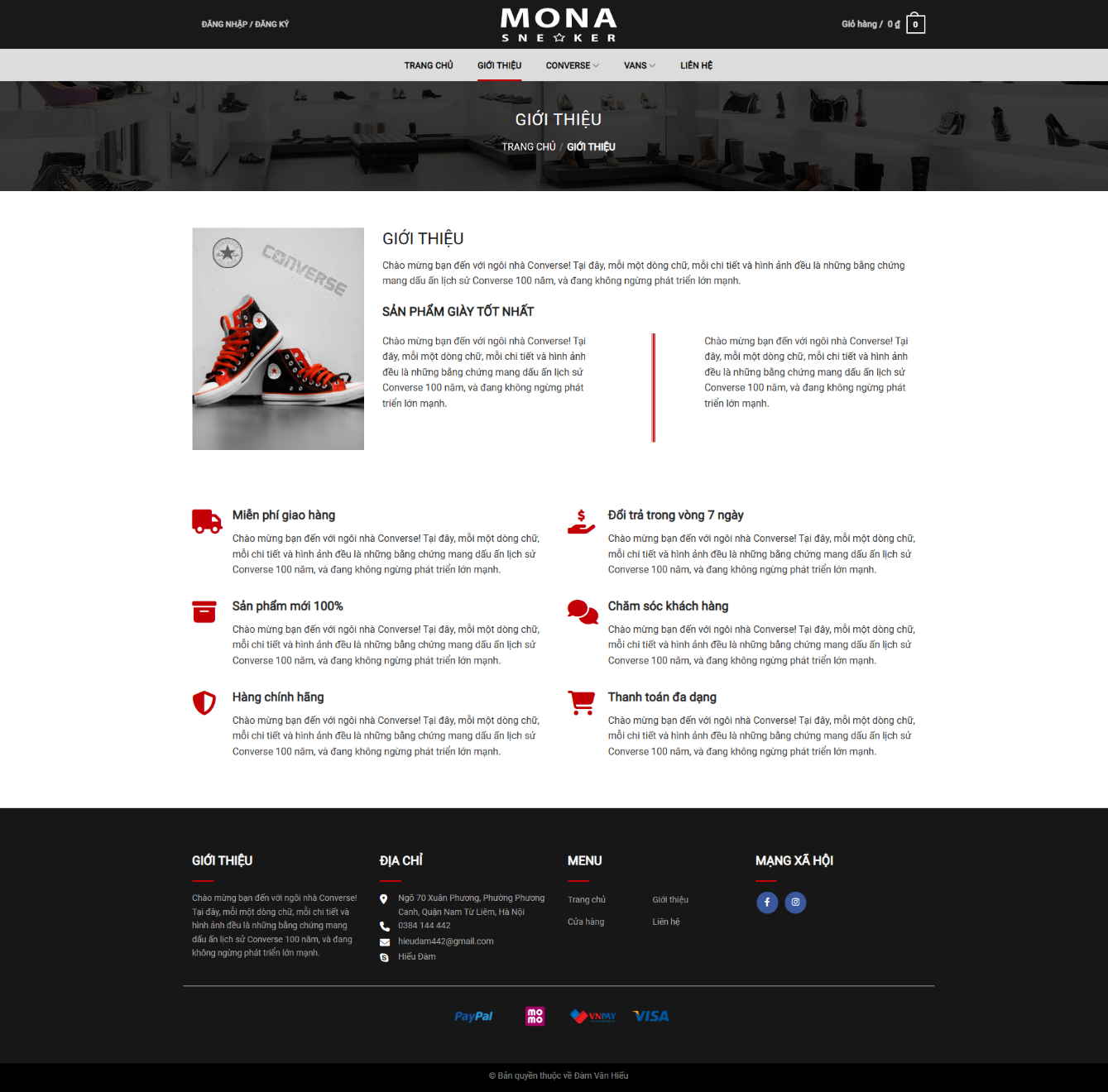
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên trường | Kiểu | Độ dài | Khóa | Giải thích |
| BillDetailId | Char | 36 | PK | Id chi tiết hóa đơn |
| BillId | Char | 36 | FK | Id hóa đơn |
| ShoeId | Char | 36 | FK | Id giày |
| Quantity | DECIMAL(8, 2) |  |  | Số lượng |
| Size | VarChar | 20 |  | Size |
| CreatedDate | DateTime |  |  | Ngày tạo |
| ModifiedDate | VarChar | 100 |  | Ngày sửa |

### **Thiết kế giao diện**

#### **Giao diện trang bán hàng**



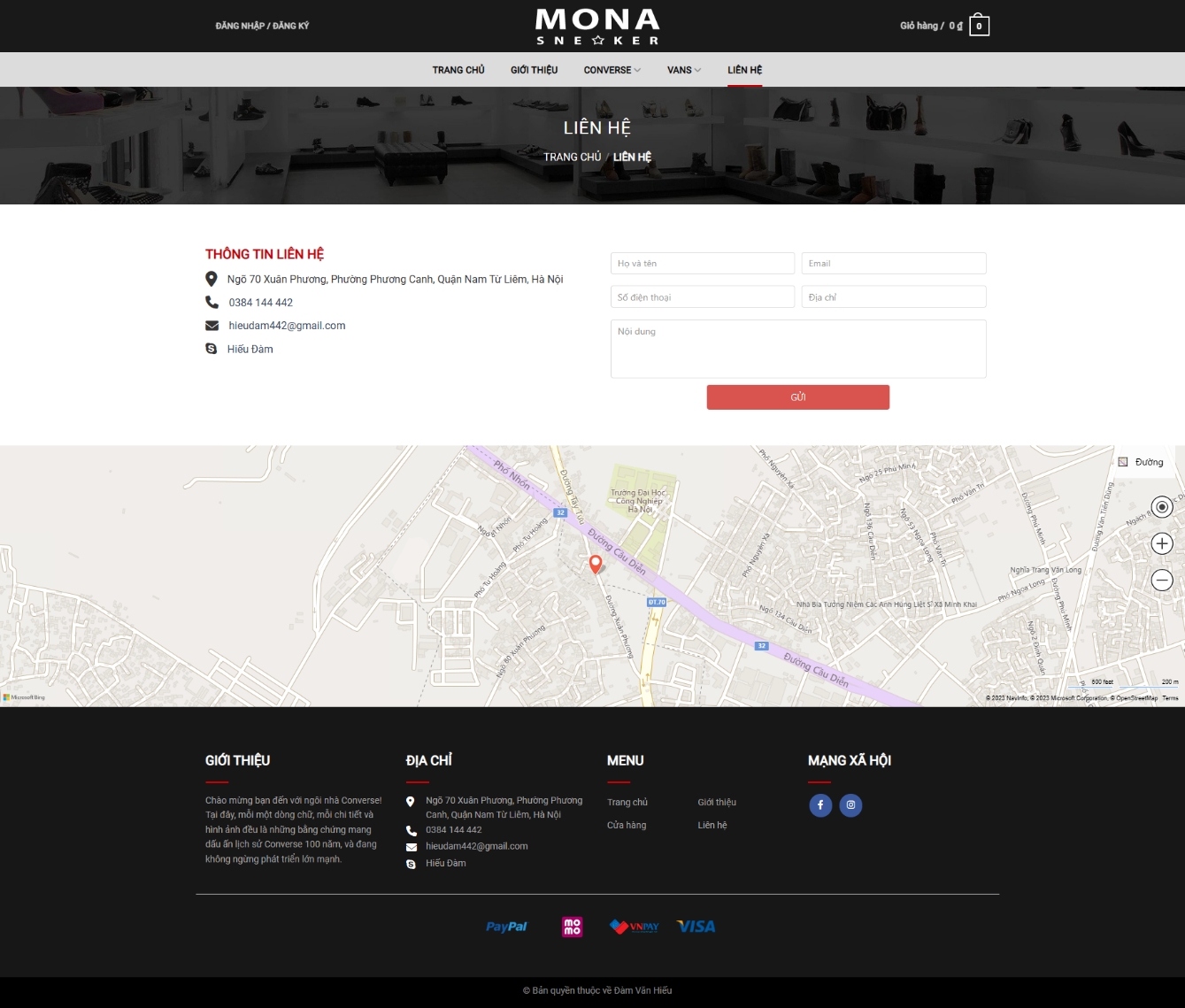
Hình . Giao diện trang chủ



Hình . Giao diện trang giới thiệu



Hình . Giao diện trang sản phẩm

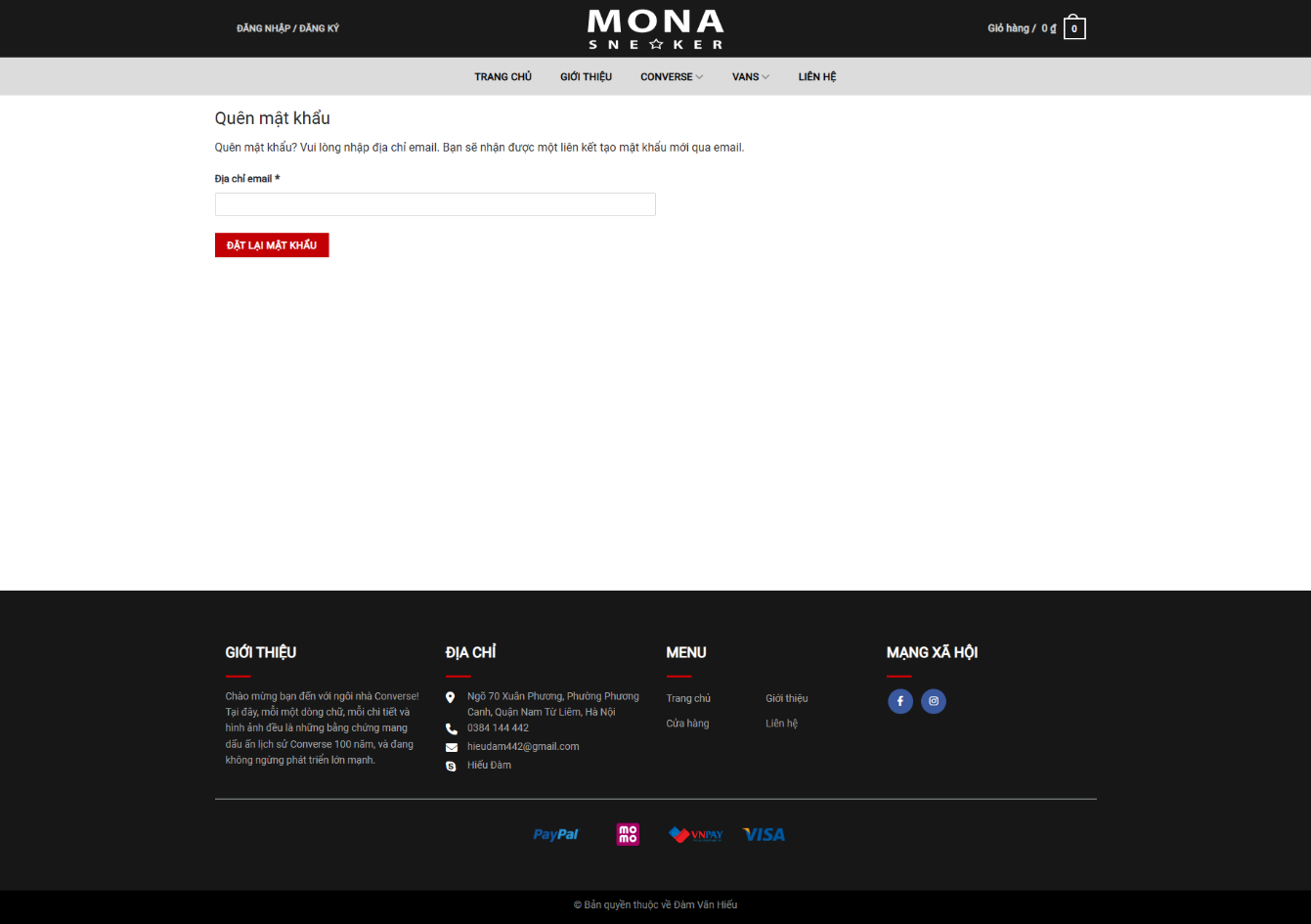


Hình . Giao diện trang liên hệ

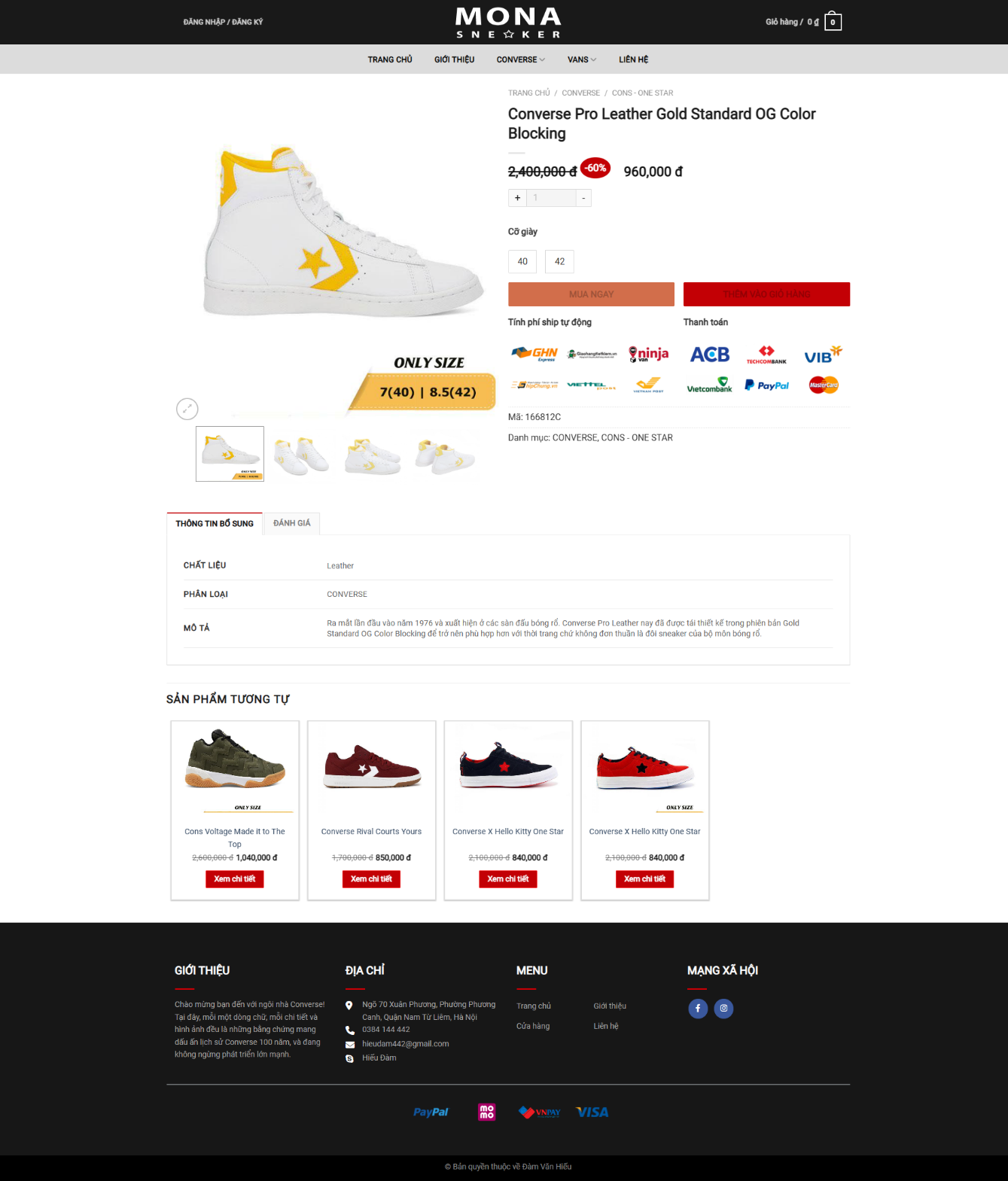
Graphical user interface, text, application, email

Description automatically generated

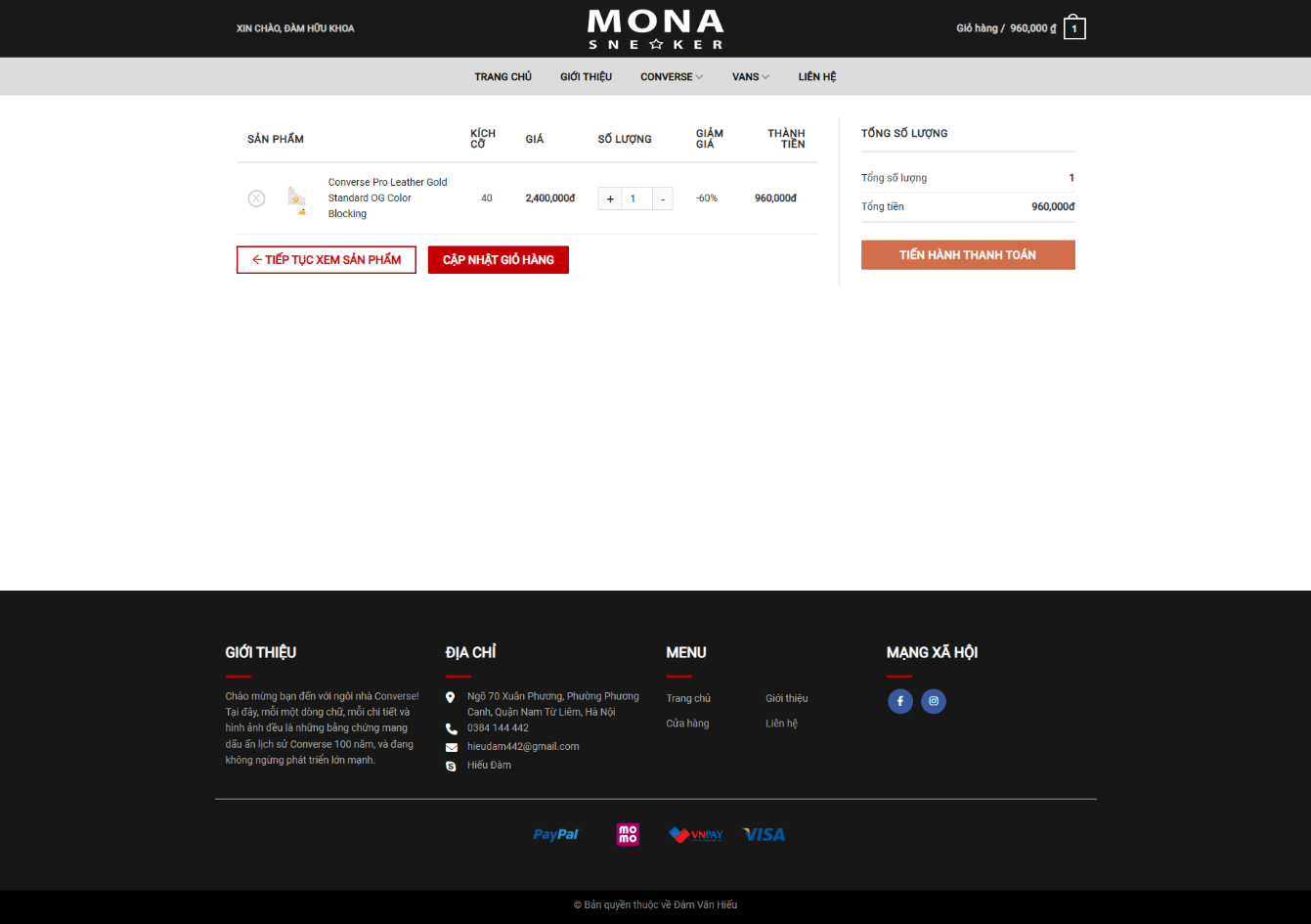
Hình . Màn hình đăng nhập, đăng ký



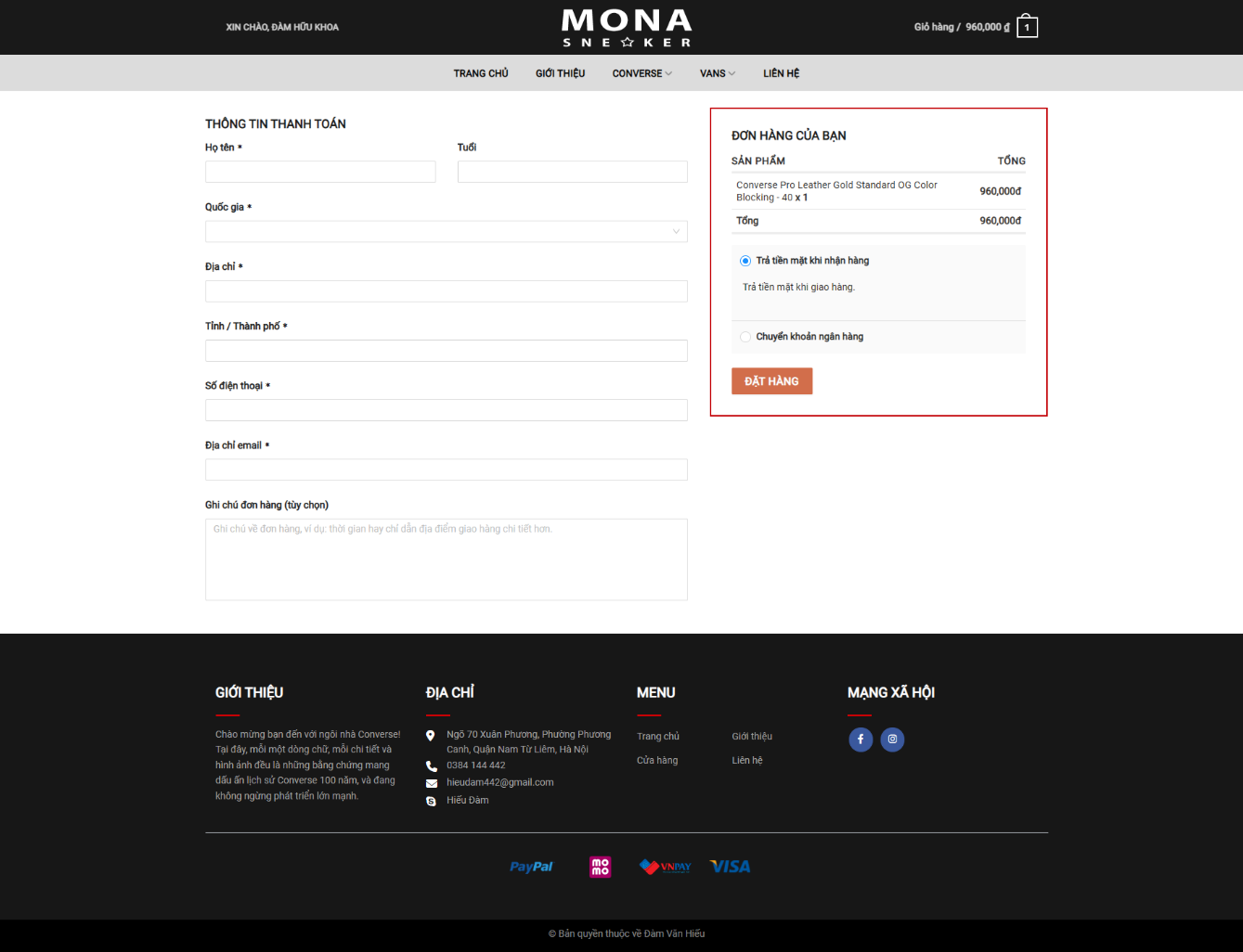
Hình . Giao diện màn hình quên mật khẩu



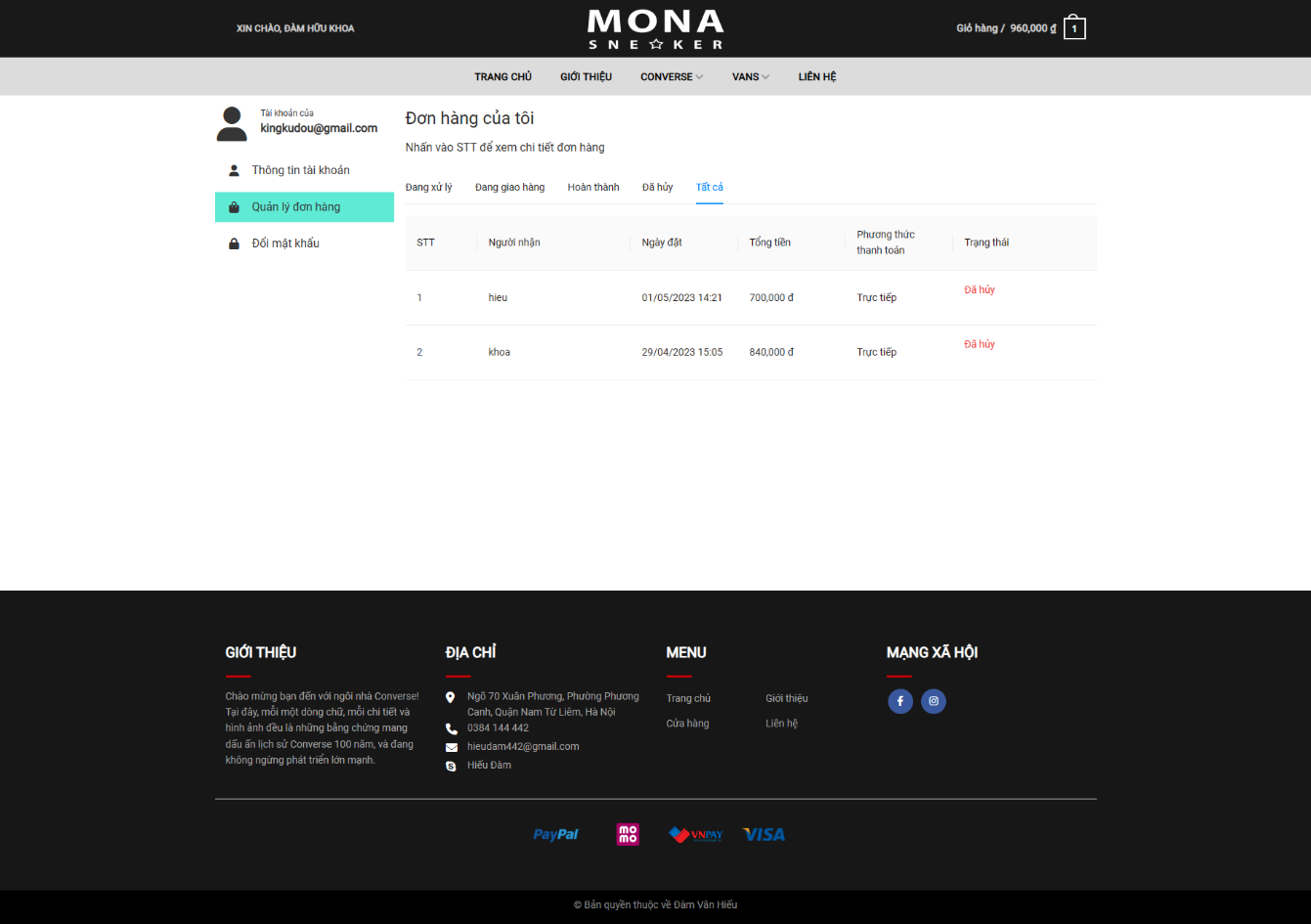
Hình . Giao diện trang chi tiết sản phẩm



Hình . Giao diện giỏ hàng

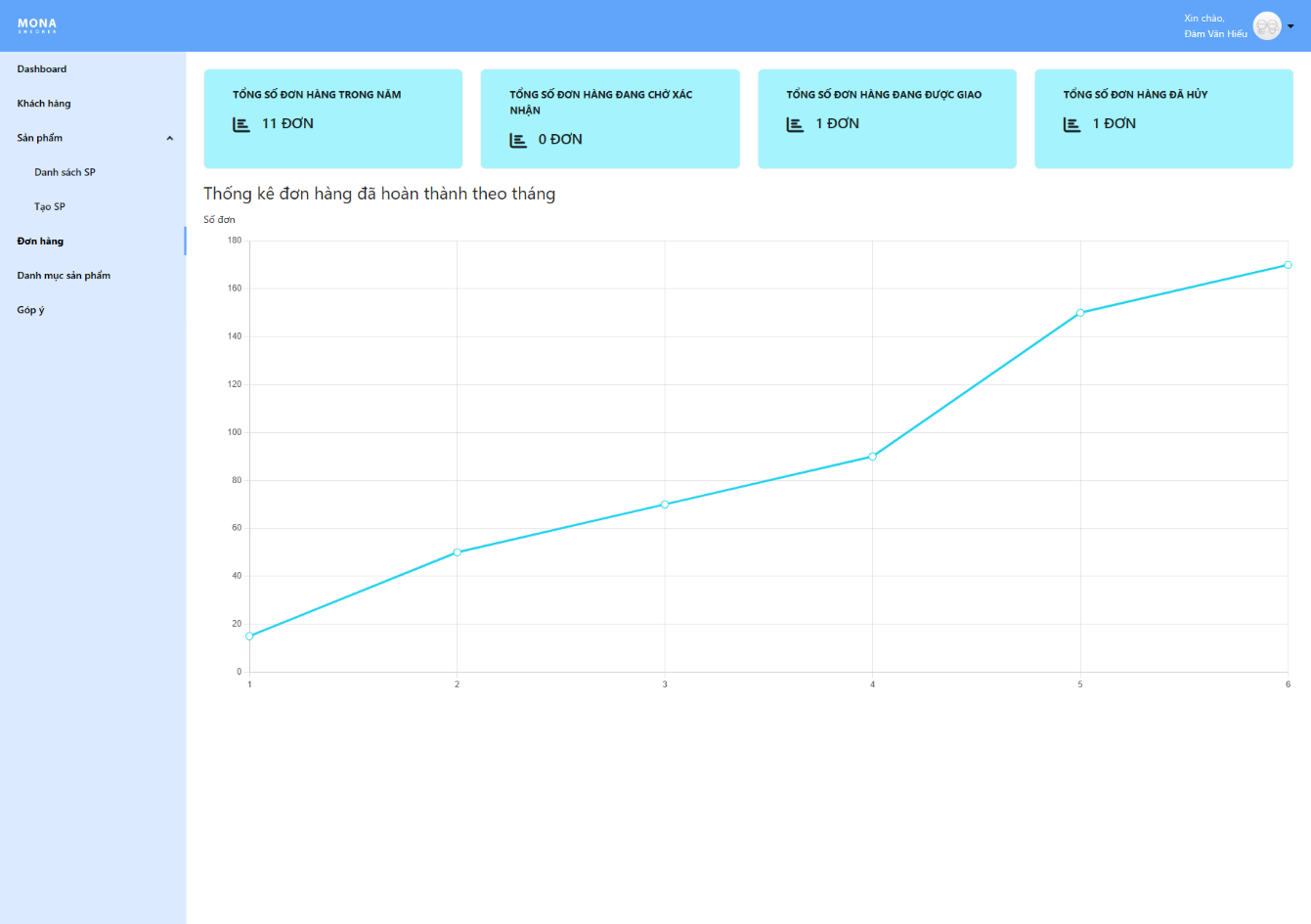


Hình . Giao diện thanh toán

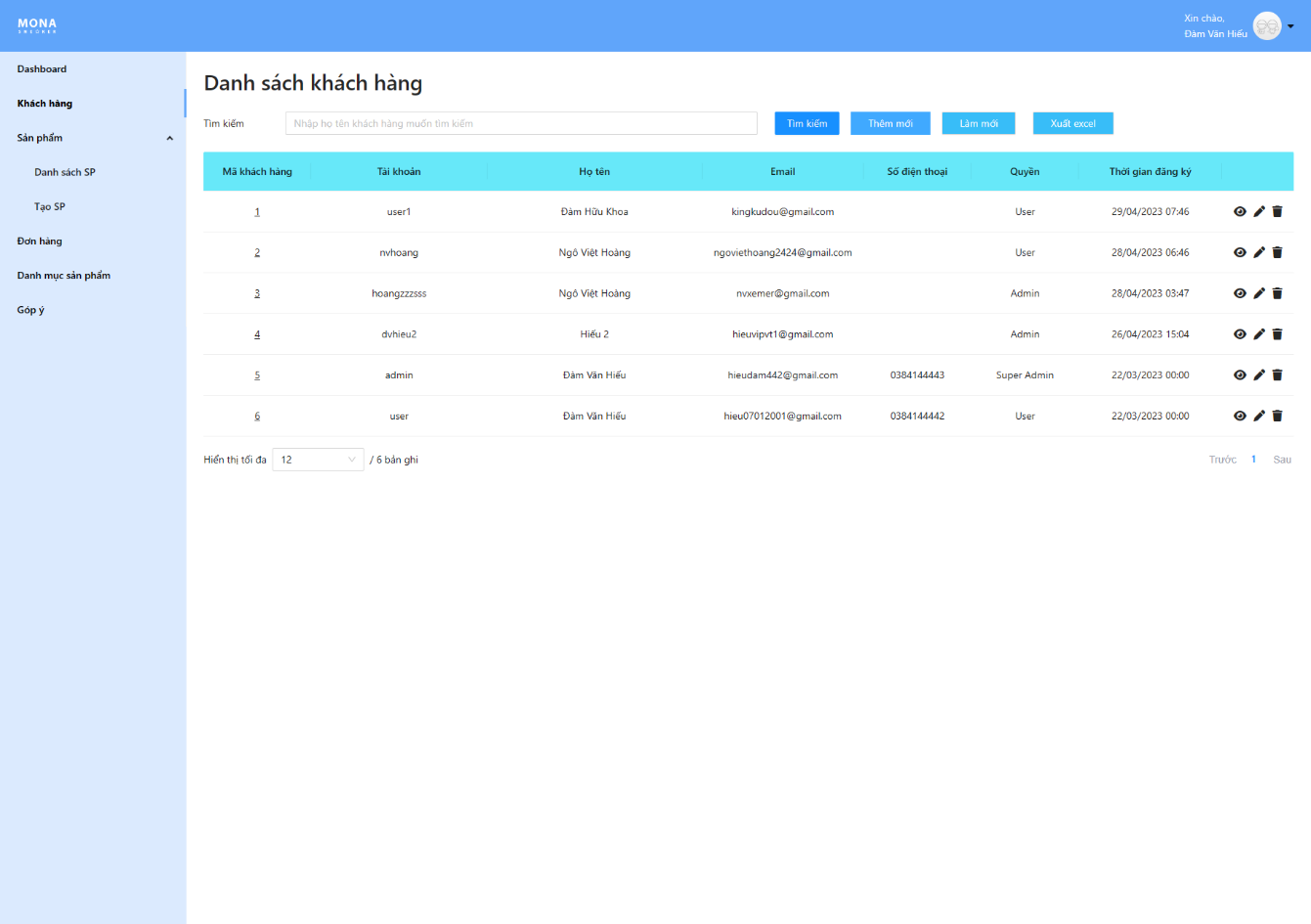


Hình . Giao diện quản lý đơn hàng

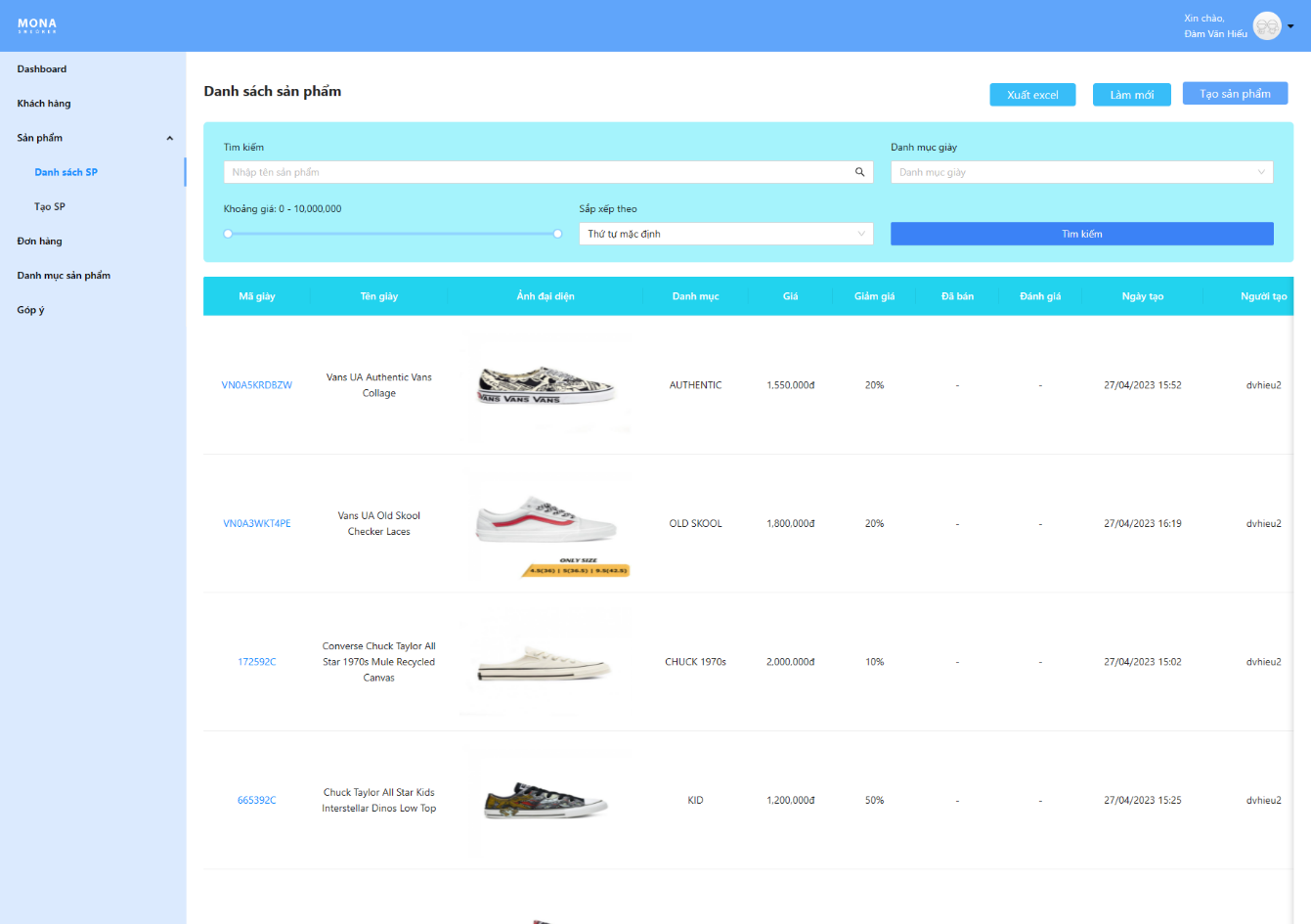
#### **Giao diện trang quản trị**

****

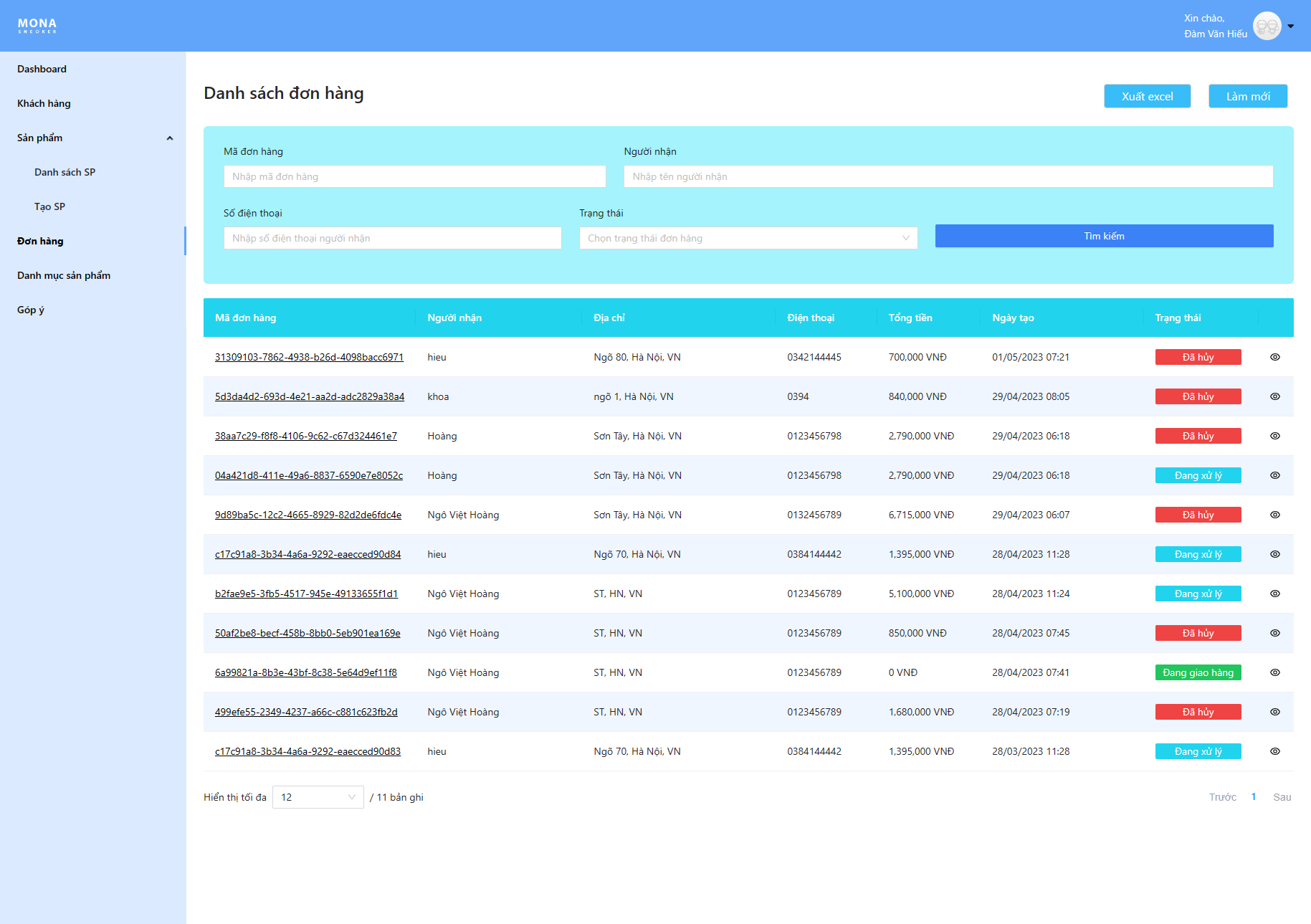
Hình . Giao diện trang chủ



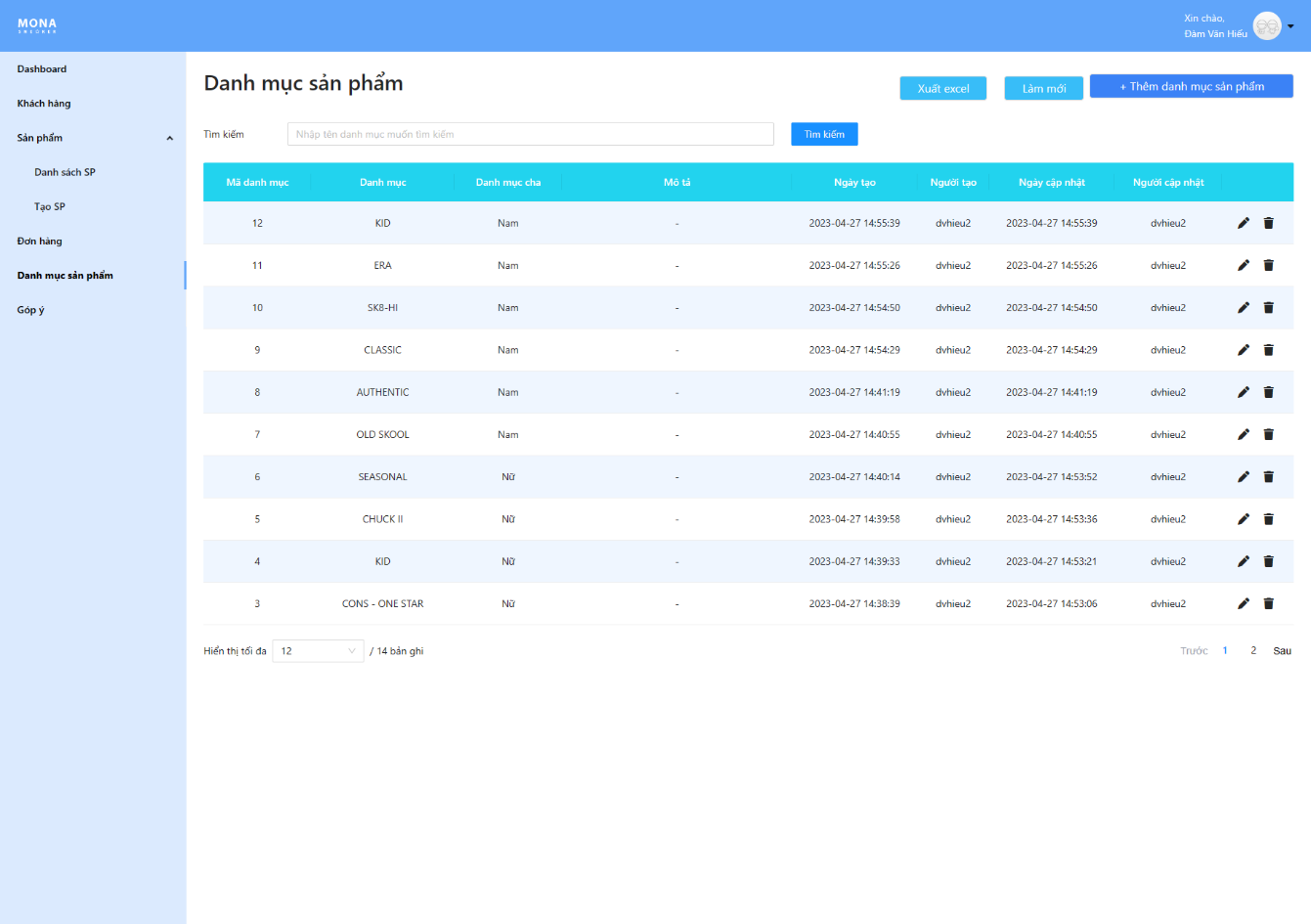
Hình . Giao diện trang quản lý tài khoản người dùng



Hình . Giao diện trang quản lý sản phẩm



Hình . Giao diện trang quản lý đơn hàng



Hình . Giao diện trang quản lý danh mục

# KẾT LUẬN

Đề tài **“Xây dựng Website bán giày sử dụng VueJs và Asp.net”** xuất phát từ những nhu cầu thực tế mà ngày nay trong ngành kinh doanh cần có. Những phương pháp bán hàng truyền thống dần lỗi thời và thay vào đó việc bán hàng qua website nâng cao hiệu quả hơn, khắc phục được những tồn đọng khó khăn mà phương pháp thủ công đang gặp phải.

Qua quá trình thực hiện đề tài, em đã tìm hiểu, tích lũy và học hỏi thêm được nhiều kinh nghiệm cũng như kiến thức công nghệ mới.

**\*Những kết quả đạt được:**

**Về công nghệ:**

* Hiểu được quá trình thiết kế một website thực tế đi từ bước cơ bản: khảo sát dự án, phân tích hệ thống, thiết kế, thực hiện, kiểm thử, triển khai, bảo trì.
* Xây dựng thành công website đáp ứng nhu cầu bán hàng.
* Nắm được các kiến thức xây dựng giao diện website: HTML, CSS, JavaScript.
* Nắm vững về cách xây dựng website thông qua thư viện, framework: Vue, Asp.net trong lập trình.
* Sử dụng thành thạo các công cụ hỗ trợ lập trình: Visual Studio Code, VS Code

**Về cài đặt chương trình:**

* Cho phép khách hàng có thể xem thông tin sản phẩm và có thể mua hàng trực tuyến.
* Tiếp nhận đơn đặt hàng của khách hàng.
* Các mặt hàng được cập nhật liên tục theo nhu cầu của khách hàng.
* Trang quản trị đầy đủ chức năng để quản lý một trang web: quản lý tài khoản, danh mục. sản phẩm, đơn hàng.

**Tính năng khác:**

* Giao diện đẹp gây ấn tượng thân thiện với khách hàng.
* Tính bảo mật.

**Hướng phát triển**

* Tiếp tục hoàn thiện các tính năng mở rộng phần mềm nhằm cải tiến và nâng cấp chương trình.
* Hoàn thiện chương trình, sửa chữa các lỗi, cải thiện tính bảo mật.
* Xây dựng Website quy mô lớn hơn với nhiều ứng dụng như: trò chuyện, tư vấn khách hàng.

# TÀI LIỆU THAM KHẢO

[1] Hoàng Quang Huy, Phùng Đức Hòa, Trịnh Bá Quý, *Nhập môn công nghệ phần mềm*, NXB Đại học Công nghiệp Hà Nội.

[2] Nguyễn Thị Thanh Huyền, Ngô Thị Bích Thúy, Phạm Thị Kim Phượng, *Giáo trình phân tích thiết kế hệ thống*, NXB Giáo dục VN.

[4] *Giáo trình thiết kế web*, Trường đại học Công Nghiệp Hà Nội.

[5] Trang tài liệu UML <https://www.tutorialspoint.com/uml/index.htm>

[6] Trang tài liệu Asp.net [ASP.NET | Open-source web framework for .NET (microsoft.com)](https://dotnet.microsoft.com/en-us/apps/aspnet)

[8] Trang tài liệu JavaScript <https://javascript.info>

[9] Trang tài liệu VueJS [Vue.js - The Progressive JavaScript Framework | Vue.js (vuejs.org)](https://vuejs.org/)

[10] Trang tin tức về lập trình <https://viblo.asia>