|  |  |
| --- | --- |
|  |  |
| NGUYỄN PHÚ HUY | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
|  |
| ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC  NGÀNH KỸ THUẬT PHẦN MỀM |
|  |
| **XÂY DỰNG WEBSITE BÁN TRANG SỨC** |
| **BẰNG MERN** |
|  |
| **CBHD: ThS. Nguyễn Tuấn Tú** |
| KỸ THUẬT PHẦN MỀM | **Sinh viên: Nguyễn Phú Huy** |
| **Mã số sinh viên: 2020600052** |
|  |
| Hà Nội – Năm 2024 |

**MỤC LỤC**

[DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT 4](#_Toc166621547)

[DANH MỤC CÁC HÌNH ẢNH, HÌNH VẼ, ĐỒ THỊ 5](#_Toc166621548)

[DANH MỤC CÁC BẢNG 8](#_Toc166621549)

[LỜI CẢM ƠN 9](#_Toc166621550)

[MỞ ĐẦU 10](#_Toc166621551)

[1. Lý do chọn đề tài 10](#_Toc166621552)

[2. Đối tượng, mục tiêu và phạm vi nghiên cứu 11](#_Toc166621553)

[3. Ý nghĩa khoa học và thực tiễn của đề tài 12](#_Toc166621554)

[CHƯƠNG 1. TỔNG QUAN VỀ NỘI DUNG NGHIÊN CỨU 14](#_Toc166621555)

[1.1. Giới thiệu chung 14](#_Toc166621556)

[1.2. Định hướng phát triển 15](#_Toc166621557)

[1.3. Cơ sở lý thuyết 17](#_Toc166621558)

[1.3.1. Giới thiệu về ngôn ngữ lập trình Javascript 17](#_Toc166621559)

[1.3.2. Giới thiệu về ReactJS 18](#_Toc166621560)

[1.3.3. Giới thiệu về NodeJS 21](#_Toc166621561)

[1.3.4. Giới thiệu về Express Framework 23](#_Toc166621562)

[1.3.5. Giới thiệu về MongoDB 24](#_Toc166621563)

[CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG 26](#_Toc166621564)

[2.1. Đặc tả bài toán 26](#_Toc166621565)

[2.2. Phân tích và thiết kế 27](#_Toc166621566)

[2.2.1. Mô hình hoá chức năng: 27](#_Toc166621567)

[2.2.2. Biểu đồ Use case 28](#_Toc166621568)

[2.2.3. Mô tả các use case 28](#_Toc166621569)

[2.2.4. Mô hình cơ sở dữ liệu hệ thống 62](#_Toc166621570)

[2.2.5. Thiết kế các bảng cơ sở dữ liệu 63](#_Toc166621571)

[2.2.6. Thiết kế giao diện 67](#_Toc166621572)

[CHƯƠNG 3. CÀI ĐẶT CHƯƠNG TRÌNH VÀ KẾT QUẢ ĐẠT ĐƯỢC 77](#_Toc166621573)

[3.1. Giao diện của hệ thống 77](#_Toc166621574)

[3.1.1. Giao diện phía khách hàng 77](#_Toc166621575)

[3.1.2. Giao diện phía quản trị 84](#_Toc166621576)

[3.2. Kiểm thử chức năng hệ thống 87](#_Toc166621577)

[3.2.1. Kiểm thử chức năng phía khách hàng 87](#_Toc166621578)

[3.2.2. Kiểm thử chức năng phía quản trị 90](#_Toc166621579)

[KẾT LUẬN 93](#_Toc166621580)

[TÀI LIỆU THAM KHẢO 94](#_Toc166621581)

# DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **STT** | **Từ viết tắt, ký hiệu** | **Giải thích** |
| 1 | VECOM | Hiệp hội Thương mại Điện tử Việt Nam |
| 2 | O2O | Online – To – Offline (Mô hình kinh doanh kết hợp giữa trực tuyến và trực tiếp) |
| 3 | API | Application Programming Interface |
| 4 | HTML | HyperText Markup Language |
| 5 | CSS | Cascading Style Sheets |
| 6 | MVC | Model-View-Controller |
| 7 | CPU | Central Processing Unit |
| 8 | UI | User Interface |

# DANH MỤC CÁC HÌNH ẢNH, HÌNH VẼ, ĐỒ THỊ

[Hình ảnh 1‑1: Redux 20](#_Toc166537048)

[Hình ảnh 2‑1: Biểu đồ Use case tổng quan 28](#_Toc166537049)

[Hình ảnh 2‑2: Biểu đồ trình tự use case đăng ký 30](#_Toc166537050)

[Hình ảnh 2‑3: Biểu đồ trình tự use case đăng nhập 32](#_Toc166537051)

[Hình ảnh 2‑4: Biểu đồ trình tự use case xem danh mục 33](#_Toc166537052)

[Hình ảnh 2‑5: Biểu đồ trình tự use case tìm kiếm sản phẩm 35](#_Toc166537053)

[Hình ảnh 2‑6: Biểu đồ trình tự xem thông tin chi tiết sản phẩm 36](#_Toc166537054)

[Hình ảnh 2‑7: Biểu đồ trình tự use case xem các bình luận sản phẩm 37](#_Toc166537055)

[Hình ảnh 2‑8: Biểu đồ trình tự use case bình luận sản phẩm 38](#_Toc166537056)

[Hình ảnh 2‑9: Mô tả chi tiết use case quản lý giỏ hàng 40](#_Toc166537057)

[Hình ảnh 2‑10: Biểu đồ trình tự use case đặt hàng 42](#_Toc166537058)

[Hình ảnh 2‑11: Biểu đồ trình tự use case quản lý thông tin cá nhân 43](#_Toc166537059)

[Hình ảnh 2‑12: Biểu đồ trình tự use case đổi mật khẩu 45](#_Toc166537060)

[Hình ảnh 2‑13: Biểu đồ trình tự use case quên mật khẩu 46](#_Toc166537061)

[Hình ảnh 2‑14: Biểu đồ trình tự use case theo dõi đơn hàng 47](#_Toc166537062)

[Hình ảnh 2‑15: Biểu đồ trình tự use case quản lý danh mục sản phẩm 50](#_Toc166537063)

[Hình ảnh 2‑16: Biểu đồ trình tự use case quản lý sản phẩm 53](#_Toc166537064)

[Hình ảnh 2‑17: Biểu đồ trình tự use case quản lý đơn hàng 55](#_Toc166537065)

[Hình ảnh 2‑18: Biểu đồ trình tự use case quản lý khách hàng 58](#_Toc166537066)

[Hình ảnh 2‑19: Biểu đồ trình tự use case quản lý kho 60](#_Toc166537067)

[Hình ảnh 2‑20: Biểu đồ trình tự use case thống kê 62](#_Toc166537068)

[Hình ảnh 2‑21: Thiết kế cơ sở dữ liệu 63](#_Toc166537069)

[Hình ảnh 2‑22: Bảng Users 64](#_Toc166537070)

[Hình ảnh 2‑23: Bảng Categories 64](#_Toc166537071)

[Hình ảnh 2‑24: Bảng Products 65](#_Toc166537072)

[Hình ảnh 2‑25: Bảng Comments 65](#_Toc166537073)

[Hình ảnh 2‑26: Bảng Orders 66](#_Toc166537074)

[Hình ảnh 2‑27: Bảng Inventories 67](#_Toc166537075)

[Hình ảnh 2‑28: Bảng Carts 67](#_Toc166537076)

[Hình ảnh 2‑29: Thiết kế màn hình trang chủ 69](#_Toc166537077)

[Hình ảnh 2‑30: Thiết kế màn hình đăng nhập 69](#_Toc166537078)

[Hình ảnh 2‑31: Thiết kế màn hình đăng ký 69](#_Toc166537079)

[Hình ảnh 2‑32: : Thiết kế màn hình sản phẩm 70](#_Toc166537080)

[Hình ảnh 2‑33: : Thiết kế màn hình thông tin chi tiết sản phẩm 71](#_Toc166537081)

[Hình ảnh 2‑34: Thiết kế màn hình quản lý giỏ hàng 72](#_Toc166537082)

[Hình ảnh 2‑35: Thiết kế màn hình quản lý thông tin cá nhân 72](#_Toc166537083)

[Hình ảnh 2‑36: Thiết kế màn hình đổi mật khẩu 73](#_Toc166537084)

[Hình ảnh 2‑37: Thiết kế màn hình quên mật khẩu 73](#_Toc166537085)

[Hình ảnh 2‑38: Thiết kế màn hình theo dõi đơn hàng 74](#_Toc166537086)

[Hình ảnh 2‑39: Thiết kế màn hình quản lý danh mục sản phẩm 74](#_Toc166537087)

[Hình ảnh 2‑40: Thiết kế màn hình quản lý sản phẩm 75](#_Toc166537088)

[Hình ảnh 2‑41: Thiết kế màn hình quản lý đơn hàng 75](#_Toc166537089)

[Hình ảnh 2‑42: Thiết kế màn hình quản lý khách hàng 76](#_Toc166537090)

[Hình ảnh 2‑43: Thiết kế màn hình quản lý kho 76](#_Toc166537091)

[Hình ảnh 2‑44: Thiết kế màn hình thống kê 77](#_Toc166537092)

[Hình ảnh 3‑1: Màn hình trang chủ 78](#_Toc166537093)

[Hình ảnh 3‑2: Màn hình đăng nhập 79](#_Toc166537094)

[Hình ảnh 3‑3: Màn hình đăng ký 79](#_Toc166537095)

[Hình ảnh 3‑4: Màn hình sản phẩm 80](#_Toc166537096)

[Hình ảnh 3‑5: Màn hình thông tin chi tiết 81](#_Toc166537097)

[Hình ảnh 3‑6: Màn hình quản lý giỏ hàng 82](#_Toc166537098)

[Hình ảnh 3‑7: Màn hình quản lý thông tin cá nhân 82](#_Toc166537099)

[Hình ảnh 3‑8: Màn hình đổi mật khẩu 83](#_Toc166537100)

[Hình ảnh 3‑9: Màn hình quên mật khẩu 83](#_Toc166537101)

[Hình ảnh 3‑10: Màn hình theo dõi đơn hàng 84](#_Toc166537102)

[Hình ảnh 3‑11: Màn hình quản lý danh mục sản phẩm 85](#_Toc166537103)

[Hình ảnh 3‑12: Màn hình quản lý sản phẩm 85](#_Toc166537104)

[Hình ảnh 3‑13: Màn hình quản lý đơn hàng 86](#_Toc166537105)

[Hình ảnh 3‑14: Màn hình quản lý khách hàng 86](#_Toc166537106)

[Hình ảnh 3‑15: Màn hình quản lý kho 87](#_Toc166537107)

[Hình ảnh 3‑16: Màn hình thống kê 88](#_Toc166537108)

# DANH MỤC CÁC BẢNG

[Bảng 2‑1:Bảng mô hình hoá chức năng 28](#_Toc166621585)

[Bảng 2‑2: Mô tả chi tiết use case Đăng ký 29](#_Toc166621586)

[Bảng 2‑3: Mô tả chi tiết use case Đăng ký 31](#_Toc166621587)

[Bảng 2‑4: Mô tả chi tiết use case xem danh mục 32](#_Toc166621588)

[Bảng 2‑5: Mô tả chi tiết use case tìm kiếm sản phẩm 34](#_Toc166621589)

[Bảng 2‑6: Mô tả chi tiết use case xem thông tin chi tiết sản phẩm 35](#_Toc166621590)

[Bảng 2‑7: Use case xem các bình luận về sản phẩm 37](#_Toc166621591)

[Bảng 2‑8: Mô tả chi tiết use case bình luận về sản phẩm 38](#_Toc166621592)

[Bảng 2‑9: Mô tả chi tiết use case quản lý giỏ hàng 39](#_Toc166621593)

[Bảng 2‑10: Mô tả chi tiết use case đặt hàng 41](#_Toc166621594)

[Bảng 2‑11: Mô tả chi tiết use case quản lý thông tin cá nhân 43](#_Toc166621595)

[Bảng 2‑12: Mô tả chi tiết use case đổi mật khẩu 44](#_Toc166621596)

[Bảng 2‑13: Mô tả chi tiết use case quên mật khẩu 46](#_Toc166621597)

[Bảng 2‑14: Mô tả chi tiết use case theo dõi đơn hàng 47](#_Toc166621598)

[Bảng 2‑15: Mô tả chi tiết use case quản lý danh mục sản phẩm 49](#_Toc166621599)

[Bảng 2‑16: Mô tả chi tiết use case quản lý sản phẩm 52](#_Toc166621600)

[Bảng 2‑17: Mô tả chi tiết use case quản lý đơn hàng 54](#_Toc166621601)

[Bảng 2‑18: Mô tả chi tiết use case quản lý khách hàng 56](#_Toc166621602)

[Bảng 2‑19: Mô tả chi tiết use case quản lý kho 58](#_Toc166621603)

[Bảng 2‑20: Mô tả chi tiết use case thống kê 60](#_Toc166621604)

[Bảng 3‑1: Bảng kiểm thử chức năng phía khách hàng 90](#_Toc166621605)

[Bảng 3‑2: Bảng kiểm thử chức năng phía quản trị 92](#_Toc166621606)

# LỜI CẢM ƠN

Đối với một sinh viên trường Đại học Công nghiệp Hà Nội, đồ án tốt nghiệp là một minh chứng cho những kiến thức đã có được sau bốn năm học tập. Em xin gửi lời cảm ơn đến trường Đại học Công nghiệp Hà Nội, khoa Công nghệ thông tin đã tạo điều kiện cho em được học tập và làm đồ án tốt nghiệp. Em xin chân thành cảm ơn thầy cô trong khoa Công nghệ thông tin đã đã giảng dạy với tất cả tâm huyết, truyền dạy cho chúng em tất cả những tri thức quý báu của mình để chúng em có thể vững bước tiếp trên con đường lập nghiệp của mình. Đặc biệt em xin chân thành cảm ơn đến thầy giáo **ThS. Nguyễn Tuấn Tú**. Trong suốt quá trình nghiên cứu, thực hiện đồ án tốt nghiệp thầy đã luôn giúp đỡ, hướng dẫn tận tình để em có thể hoàn thành đề tài tốt nghiệp này.

Em đã cố gắng hoàn thiện báo cáo đồ án tốt nghiệp một cách tốt nhất trong khả năng của mình, tuy nhiên không thể tránh được những thiếu sót. Em rất mong nhận được sự góp ý của các thầy cô để đồ án tốt nghiệp này của em được hoàn thiện hơn.

Lời cuối cùng em xin chúc thầy cô luôn dồi dào sức khỏe, luôn vui vẻ và thành công trong cuộc sống!

Em xin chân thành cảm ơn!

# MỞ ĐẦU

## Lý do chọn đề tài

Ngày nay, nhu cầu làm đẹp và khẳng định bản thân của con người ngày càng tăng cao, dẫn đến việc thị trường trang sức ngày càng sôi động. Các cửa hàng trang sức liên tục được mở ra để đáp ứng nhu cầu này. Tuy nhiên, để cạnh tranh trong thị trường đầy tiềm năng nhưng cũng đầy rẫy thách thức này, việc sở hữu một website bán hàng là vô cùng quan trọng.

Thiết kế website chuyên bán trang sức mang lại nhiều lợi ích cho cả doanh nghiệp và khách hàng:

**Đối với doanh nghiệp:**

* **Mở rộng thị trường:** Website giúp doanh nghiệp tiếp cận được nhiều khách hàng tiềm năng hơn, không chỉ giới hạn trong khu vực địa lý mà cửa hàng tọa lạc. Khách hàng từ mọi nơi trên thế giới đều có thể truy cập website và mua sắm sản phẩm.
* **Tiết kiệm chi phí:** So với việc mở cửa hàng truyền thống, việc thiết kế website có chi phí thấp hơn nhiều. Doanh nghiệp không cần phải tốn tiền thuê mặt bằng, trang trí cửa hàng, thuê nhân viên bán hàng,...
* **Quản lý sản phẩm dễ dàng:** Website giúp doanh nghiệp dễ dàng quản lý sản phẩm, cập nhật giá cả, hình ảnh và thông tin chi tiết về sản phẩm một cách nhanh chóng và hiệu quả.
* **Tăng doanh số bán hàng:** Website giúp doanh nghiệp tăng doanh số bán hàng nhờ khả năng tiếp cận được nhiều khách hàng hơn và việc mua sắm trở nên dễ dàng hơn cho khách hàng.
* **Nâng cao uy tín thương hiệu:** Một website được thiết kế đẹp mắt, chuyên nghiệp sẽ giúp nâng cao uy tín thương hiệu của doanh nghiệp trong mắt khách hàng.

**Đối với khách hàng:**

* **Tiết kiệm thời gian:** Khách hàng có thể tham khảo sản phẩm và mua sắm trực tuyến mà không cần phải đến trực tiếp cửa hàng.
* **So sánh giá cả:** Khách hàng có thể dễ dàng so sánh giá cả của các sản phẩm khác nhau từ các cửa hàng khác nhau trước khi mua hàng.
* **Mua sắm mọi lúc mọi nơi:** Khách hàng có thể mua sắm bất cứ lúc nào, bất cứ ở đâu chỉ cần có kết nối internet.
* **Nhận được nhiều ưu đãi:** Các doanh nghiệp thường xuyên có các chương trình khuyến mãi, ưu đãi dành riêng cho khách hàng mua sắm online.

Với những lợi ích trên, việc thiết kế website bán trang sức là một khoản đầu tư vô cùng hiệu quả cho các doanh nghiệp kinh doanh trong lĩnh vực này.

## Đối tượng, mục tiêu và phạm vi nghiên cứu

* + **Đối tượng nghiên cứu bao gồm:**
* Cửa hàng bán trang sức Helios.
* Công nghệ: ngôn ngữ JS, NodeJS(Express), ReactJS.
* Nghiên cứu các website bán hàng trang sức và quy trình đặt hàng.
* Nghiên cứu về công cụ Visual Studio, Rational Rose, Code Planner, Balsamiq Mockups
* **Mục đích nghiên cứu:**
* Xác định và làm rõ các vấn đề liên quan đến đề tài.
* Khảo sát, phân tích thiết kế hệ thống cho website.
* Khách hàng có thể tham khảo thông tin sản phẩm qua website chỉ bằng vài thao tác với Internet.
* Giúp công việc mua sắm có thể tiến hành một cách nhanh chóng, tiện lợi, tiết kiệm thời gian, đáp ứng được nhu cầu thực tế cho khách hàng.
* Hệ thống tìm kiếm dễ dàng, giao diện thân thiện.
* Chủ cửa hàng có thể dễ dàng quản lý sản phẩm, quản lý đơn hàng, thống kê được doanh số bán hàng, v.v. Có thể xem được thống kê theo từng tháng, theo từng mặt hàng để điều chỉnh hàng hóa một cách hợp lý.
  + **Phạm vi nghiên cứu:**
* Khảo sát về thực trạng tại các cơ sở, cửa hàng kinh doanh. Tìm hiểu những yếu tố và điều kiện cần thiết phải có để có thể hỗ trợ kinh doanh cho cửa hàng một cách tối ưu nhất cũng như tiện lợi cho khách hàng.
* Nghiên cứu qua Internet: đánh giá và đưa ra các ý kiến khách quan về vấn đề.
* Thực hiện phân tích các chức năng chính cần thiết cho hệ thống.
* Lựa chọn công nghệ phù hợp để phát triển cũng như các công cụ cần thiết để xây dựng website

## Ý nghĩa khoa học và thực tiễn của đề tài

Đề tài em muốn thực hiện là xây dựng website bán trang sức .Với đề tài này website sẽ cho phép người sử dụng thực hiện một cách nhanh chóng, chính xác việc nhập dữ liệu, lưu trữ, cập nhập thông tin các mặt hàng, thực hiện yêu cầu báo cáo…với giao diện làm việc đơn giản, dễ sử dụng.

Website mang đến nguồn thông tin về sản phẩm chính xác cho khách hàng dễ dàng tra cứu và chọn lựa sản phẩm, cho phép khách hàng ở xa hay gần đều có thể mua hàng qua website một cách nhanh chóng nhất.

* **Một số kết quả thực nghiệm cần đạt được:**
* Website đạt yêu cầu quản lý của chủ cửa hàng.
* Website giải quyết nhu cầu về mua bán trực tuyến sản phẩm cho khách hàng.
* Quảng bá thông tin sản phẩm, tăng tương tác giữa người mua hàng và người bán hàng qua online.
* Giao diện thân thiện, đẹp mắt, dễ sử dụng.

# TỔNG QUAN VỀ NỘI DUNG NGHIÊN CỨU

## Giới thiệu chung

Trong những năm gần đây, trước sự bùng nổ của công nghệ số, hoạt động thương mại điện tử đã và đang làm thay đổi đáng kể hành vi tiêu dùng. Người tiêu dùng đang có xu hướng chuyển qua mua bán sản phẩm qua ứng dụng trực tuyến. Tuy nhiên, cần phải nhìn thực tế rằng, tại Việt Nam, xu thế mua bán trực tuyến chưa thể lấn át và thay thế hoàn toàn việc bán hàng trực tiếp. Trong thời gian tới, nếu sự kết hợp tốt bán hàng trực tuyến và bán hàng trực tiếp sẽ là nền tảng để các doanh nghiệp nhỏ mở rộng và phát triển bền vững.

Đặc biệt là thị trường về trang sức như là phụ kiện trang sức, nhẫn, vòng tay, đông hồ, ….thì mua bán trực tuyến chính là giải pháp toàn diện cho những doanh nghiệp mong muốn mở rộng phạm vi kinh doanh, tận dụng nguồn khách hàng trên internet để đem lại doanh thu đáng kể cho mình.

Sở hữu một website bán hàng chính là công cụ hữu ích để giải quyết vấn đề này cho các doanh nghiệp và các cửa hàng nhỏ mở rộng quy mô kinh doanh, bên cạnh đó tăng tương tác với nhiều người tiêu dùng.

* **Những vấn đề còn tồn tại**
* Cụ thể, đối với các cửa hàng nhỏ và tầm trung sẽ thường bán hàng trực tiếp đơn thuần, việc truyền tải thông tin đến khách hàng sẽ vô cùng hạn chế, bởi phạm vi khách hàng lui tới cửa hàng chỉ trong khu vực địa lý gần cửa hàng.
* Các công ty thương mại điện tử, bán hàng trực tuyến có giá cả và khả năng đáp ứng nhu cầu lựa chọn đa dạng của khách hàng chính là lý do khiến các nhà bán lẻ từng lo ngại rằng họ sẽ không đủ khả năng để cạnh tranh.
* Các cửa hàng bán hàng trực tiếp đòi hỏi chi phí thuê mặt bằng khá cao và chi phí thuê nhân viên để điều hành các cửa hàng đó.
* Đặc biệt, cửa hàng nhỏ không thể cung cấp nhiều và đa dạng các chủng loại hàng hóa nếu không gian nhỏ hẹp. Trong khi đó, các trang bán hàng trực tuyến có thể cung cấp tất cả thông tin về hàng hóa rất đa dạng mà không tốn nhiều chi phí cho mặt bằng và nhân công, họ chỉ cần một nhà kho nhỏ chứa hàng và tự mình hoặc thuê dịch vụ vận chuyển hàng hóa đến tay khách hàng.

## Định hướng phát triển

Trong bối cảnh Internet ngày càng phát triển, không ít doanh nghiệp nhỏ đang triển khai Mô hình kinh doanh kết hợp giữa trực tuyến và trực tiếp (O2O). Với mô hình kinh doanh này, doanh nghiệp sẽ kết hợp cả hai hình thức nêu trên một cách tối ưu nhất. Mô hình O2O sẽ cung cấp các thông tin sản phẩm hữu ích nhất để khách hàng có thể mua sắm tiện lợi và tiết kiệm nhiều thời gian.

Mô hình O2O sẽ được thực thi với mục tiêu rõ ràng:

* Kênh online làm nhiệm vụ giới thiệu tới khách hàng về sản phẩm và dịch vụ, từ đó cho phép khách hàng tiềm năng tìm kiếm đầy đủ thông tin cần thiết nhằm dẫn dắt họ đến tận cửa hàng để thực hiện mua bán hoặc có thể mua online ngay cả khi họ ở rất xa cửa hàng.
* Trong khi đó, kênh offline sẽ là nơi cung cấp mọi trải nghiệm đầy đủ nhất, thỏa mãn nhất cho khách hàng về sản phẩm mà khách hàng muốn mua đã có thông tin trên website.
* Đây là cách thức bán hàng rất thu hút khách hàng từ kênh trực tiếp đến với cửa hàng thực tế của mình. Thông qua các website bán hàng để có thể truyền đạt thông tin sản phẩm đến người tiêu dùng một cách nhanh chóng và chính xác.
* Người dùng sẽ tìm kiếm sản phẩm mình muốn mua trước, so sánh hiệu suất cũng như các tiêu chí sản phẩm khác nhau để đưa ra sản phẩm cuối cùng mình muốn mua, sau đó tới cửa hàng để có thể trải nghiệm trực tiếp sản phẩm.

Với những thông tin trên có thể thấy rằng mô hình O2O là hình thức kinh doanh rất phù hợp với phong cách của người Việt. Một khi việc mua sắm được thực hiện bằng cách tìm kiếm thông tin trên internet, khách hàng có thể tiến hành đặt hàng trước, sau đó khách hàng đến các cửa hàng thực tế để trải nghiệm sản phẩm và mua hàng, như vậy khách hàng sẽ không còn lo ngại về chất lượng hàng hóa không ổn định hay không hài lòng về mẫu mã hàng hóa thực tế.

Không những vậy, kỹ thuật mà các công ty sử dụng mô hình O2O có thể dùng là kỹ thuật “Buy online pick – Up in store”, tức là đặt hàng trực tuyến nhưng nhận hàng tại cửa hàng, hoặc cho phép khách hàng đổi trả sản phẩm đã mua trực tuyến tại các cửa hàng thực tế hoặc với số điểm tích lũy có được khi mua hàng trực tuyến, khách hàng được tặng phiếu đến các cửa hàng thực tế để mua sắm sản phẩm.

Với cách thức tổ chức kinh doanh như vậy, các công ty sẽ vẫn có khách hàng đến các cửa hàng thực tế để xem các sản phẩm họ nhìn thấy và đặt mua trên mạng, như vậy công ty cũng sẽ thu hút được một lượng khách hàng đáng kể đến cửa hàng thực tế lấy sản phẩm nếu tiện đường đi thay vì ngồi chờ đợi nhân viên giao hàng tới.

Còn đối với các khách hàng có vị trí địa lý ở xa cửa hàng hơn, ngoài việc tìm kiếm thông tin qua Internet để có thể nắm rõ thông số, cấu hình sản phẩm thì cách để tạo lòng tin đối với khách hàng là cửa hàng cần liên hệ trực tiếp và cho khách hàng kiểm tra hàng trước khi nhận. Thứ nhất là tạo lòng tin nơi khách hàng, thứ hai là xây dựng cầu nối cho nhiều khách hàng khác biết tới dịch vụ của cửa hàng tốt. Tùy thuộc vào vị trí địa lý xa hay gần mà cửa hàng có thể đưa ra mức cọc đối với sản phẩm ship đi, và có thể cho phép khách hàng thanh toán nốt số tiền sản phẩm còn lại khi nhận được hàng.

Việc xây dựng một website bán hàng sẽ giải quyết được nhu cầu mua bán cũng như quản lý cho chủ doanh nghiệp nhỏ và cửa hàng bán lẻ:

* Phù hợp với cách thức kinh doanh trực tuyến và trực tiếp.
* Tiện lợi cho cả khách hàng ở xa và khách hàng ở gần.
* Giải quyết được bài toán hiệu suất kinh doanh mà chủ cửa hàng mong muốn.
* Vừa có thể đưa thông tin tới khách hàng nhanh gọn và đầy đủ nhất, vừa có thể nâng cao hiệu suất kinh doanh cho cửa hàng.
* Không những vậy, website giúp chủ cửa hàng quản lý sản phẩm, quản lý đơn hàng chặt chẽ hơn, v.v tiết kiệm được những chi phí đắt đỏ trước đó.

Trong báo cáo này, đề tài em thực hiện là “**Xây dựng website bán trang sức bằng MERN**”, để có thể đáp ứng nhu cầu cần thiết tối thiểu phù hợp cho cả người tiêu dùng lẫn người kinh doanh, ngoài ra website có giao diện dễ sử dụng, tối ưu nhất cho mọi đối tượng khách hàng.

## Cơ sở lý thuyết

### Giới thiệu về ngôn ngữ lập trình 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.

### Giới thiệu về ReactJS

#### ReactJS là gì?

ReactJS được hiểu nôm na là một thư viện mã nguồn mở chứa nhiều JavaScript và người tạo ra ReactJS chính là ông trùm với cái tên quen thuộc Facebook, được ra đời vào năm 2013. Mục đích của việc tạo ReactJS là tạo ra các ứng dụng web nhanh, hiệu quả và hấp dẫn với nỗ lực viết mã tối thiểu. Mục tiêu chính của ReactJS là bất kỳ trang web nào sử dụng ReactJS phải mượt mà, nhanh chóng, khả năng mở rộng cao và dễ thực hiện.

Nhìn chung, các tính năng và điểm mạnh của ReactJS thường đến từ việc tập trung vào các phần riêng lẻ. Do đó, khi làm việc với web thay vì toàn bộ ứng dụng của một trang web bằng ReactJS, các nhà phát triển tính năng có thể tách rời và chuyển đổi giao diện người dùng từ những cách phức tạp và biến nó thành những phần đơn giản hơn. Điều này có nghĩa là kết xuất dữ liệu không chỉ ở phía máy chủ. Thực hiện với các vị trí, nhưng bạn cũng có thể thực hiện tại vị trí khách hàng khi sử dụng ReactJS.

#### Lịch sử phát triển của ReactJS

Trước thời điểm năm 2013 việc xây dựng UI cho Facebook hoàn toàn dựa trên framework AngularJS của Google, tuy nhiên sau thời điểm năm 2013 khi lượng người dùng trở nên quá lớn và dữ liệu Facebook lưu trữ ngày càng nhiều, kéo theo đó việc xử lý dữ liệu ở ngoài UI trở nên quá khó khăn vì AngularJS chậm và khá nặng nề. Facebook không thể tìm kiếm được một framework nào khác đủ khả năng thay thế cho AngularJS, nên đã quyết định tự tạo ra ra một công nghệ mới là “React” gồm 2 phần: ReactJS (dùng cho web app) và React Native (dùng cho mobile app).

React được tạo bởi Jordan Walke, một kỹ sư phần mềm làm việc tại Facebook, người đã tại ra bản mẫu đầu tiền của ReactJS gọi là “FaxJS”. Jordan Walke bị ảnh hưởng bới XHP (một nền tảng thành phần HTML cho PHP). React lần đầu tiên được triển khai cho ứng dụng Newsfeed của Facebook năm 2011, rồi sau đó được triển khai cho Instagram vào năm 2012. Tháng 5 năm 2013, nó được mở mã nguồn (open-sourced) tại JSConfUS nhằm mục đích có nhiều người góp phần phát triển thư viện này và sớm trở thành một framework.

Ngày 26/04/2022, React đã cho ra phiên bản mới nhất là phiên bản 18.1.0.

#### Ưu điểm của ReactJS

Reactjs cực kì hiệu quả: Reactjs tạo ra cho chính nó DOM ảo – nơi mà các component thực sự tồn tại trên đó. Điều này sẽ giúp cải thiện hiệu suất rất nhiều. Reactjs cũng tính toán những thay đổi nào cần cập nhật len DOM và chỉ thực hiện chúng. Điều này giúp Reactjs tránh những thao tác cần trên DOM mà nhiều chi phí.

Khi đó ta có thể viết các methods, các funtions tác động lên component Hoge (thay đổi Model data), View sẽ lập tức được cập nhật cho dù ta không phải động chạm trực tiếp gì vào phần tử DOM trên View.

Reactjs giúp việc viết các đoạn code JS dễ dàng hơn: Nó dung cú pháp đặc biệt là JSX (Javascript mở rộng) cho phép ta trộn giữa code HTML và Javascript. Ta có thể them vào các đoạn HTML vào trong hàm render mà không cần phải nối chuỗi. Đây là đặc tính thú vị của Reactjs. Nó sẽ chuyển đổi các đoạn HTML thành các hàm khởi tạo đối tượng HTML bằng bộ biến đổi JSX.

Nó có nhiều công cụ phát triển: Khi bạn bắt đầu Reactjs, đừng quên cài đặt ứng dụng mở rộng của Chrome dành cho Reactjs. Nó giúp bạn debug code dễ dàng hơn. Sau khi bạn cài đặt ứng dụng này, bạn sẽ có cái nhìn trực tiếp vào virtual DOM như thể bạn đang xem cây DOM thông thường.

Render tầng server: Một trong những vấn đề với các ứng dụng đơn trang là tối ưu SEO và thời gian tải trang. Nếu tất cả việc xây dựng và hiển thị trang đều thực hiện ở client, thì người dung sẽ phải chờ cho trang được khởi tạo và hiển thị lên. Điều này thực tế là chậm. Hoặc nếu giả sử người dung vô hiệu hóa Javascript thì sao? Reactjs là một thư viện component, nó có thể vừa render ở ngoài trình duyệt sử dụng DOM và cũng có thể render bằng các chuỗi HTML mà server trả về.

Làm việc với vấn đề test giao diện: Nó cực kì dễ để viết các test case giao diện vì virtual DOM được cài đặt hoàn toàn bằng JS.Hiệu năng cao đối với các ứng dụng có dữ liệu thay đổi liên tục, dễ dàng cho bảo trì và sửa lỗi.

#### Nhược điểm ReactJS là gì

Reactjs chỉ phục vụ cho tầng View. React chỉ là View Library nó không phải là một MVC framework như những framework khác. Đây chỉ là thư viện của Facebook giúp render ra phần view. Vì thế React sẽ không có phần Model và Controller, mà phải kết hợp với các thư viện khác. React cũng sẽ không có 2-way binding hay là Ajax

Tích hợp Reactjs vào các framework MVC truyền thống yêu cầu cần phải cấu hình lại.

React khá nặng nếu so với các framework khác React có kích thước tương tương với Angular (Khoảng 35kb so với 39kb của Angular). Trong khi đó Angular là một framework hoàn chỉnh.

#### React Router

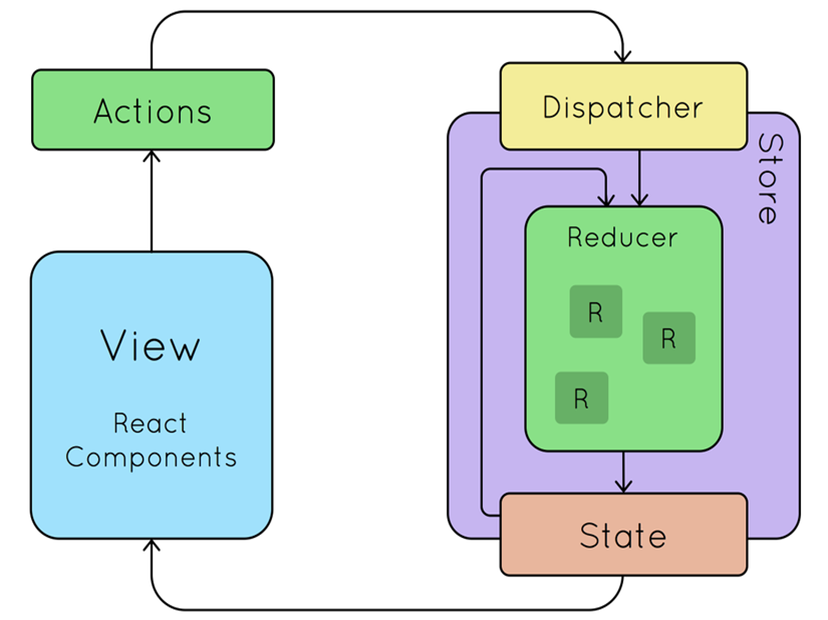
React-Router là một thư viện định tuyến (routing) tiêu chuẩn trong React. Nó giữ cho giao diện của ứng dụng đồng bộ với URL trên trình duyệt. React-Router cho phép bạn định tuyến "luồng dữ liệu" trong ứng dụng của bạn một cách rõ ràng. Nó tương đương với sự khẳng định, nếu bạn có URL này, nó sẽ tương đương với Route này, và giao diện tương ứng

#### Redux

Redux là một thư viện giúp bạn quản lí trạng thái (state) của application. Nó được thiết kế dựa trên Flux, nhưng giảm bớt những đau khổ thường gặp phải khi viết một ứng dụng Flux. Nếu bạn đã từng làm việc với Flux, bạn sẽ dễ dàng nhận ra rằng Redux đã xây dựng sẵn cho bạn rất nhiều thứ mà lẽ ra bạn tự phải làm. Ngoài ra, không giống như Flux, bạn có một state container duy nhất. Đó là một ưu điểm lớn, bởi vì nó sẽ giúp chia sẻ state và tái sử dụng code dễ dàng hơn.

Stores: Một store đơn giản là một state container. Đây là nơi lưu trữ state và nơi những action được phát đi và xử lí. Khi bạn bắt đầu xây dựng ứng dụng bằng Redux, bạn phải nghĩ bạn muốn model ứng dụng của bạn như nào và các state sẽ được lưu trữ như nào. Đây là điều rất quan trọng, bởi vì với Redux bạn chỉ nên có 1 store và state thì được chia sẻ, nên bạn cần phải suy nghĩ kĩ trước khi bắt đầu.

Actions: là những đối tượng mô tả cách chúng ta muốn thay đổi state. Bạn có thể hình dung action như là những API cho state của bạn



Hình ảnh ‑: Redux

### Giới thiệu về NodeJS

#### Giới thiệu về NodeJS

NodeJS là một ngôn ngữ lập trình và là một Platform.

Ngôn ngữ lập trình:NodeJS là một nền tảng được xây dựng trên nên tảng JavaScript V8 Engine. Được xây dựng để phát triển những ứng dụng sever side.

Platform: Bất cứ môi trường phần cứng hoặc phần mềm nào mà trong đó có một chương trình chạy, thì được hiểu như là một Platform.

NodeJS xuất hiện lần đầu vào năm 2009 bởi Ryan Dahl. Phiên bản đầu tiên của NodeJs chỉ hộ trợ hệ điều hành Linux và MacOS X. Ban đầu, việc phát triển và bảo trì được thực hiển bởi Dahl. Sau đó được Joyent, một công ty phần mềm và dịch vụ tài trợ.

Vào tháng 1 năm 2010, một trình quản lí của NodeJS đã được giới thiệu để giúp các lập trình viên dễ dàng xuất bản và chia sẻ mã nguồn của các gói NodeJS, đồng thời đơn giản hóa việc cài đặt, gỡ cài đặt và cập nhập các phiên bản.

Vào năm 2011, Microsoft và Joyent cùng nhau phát triển phiên bản Windows của NodeJS, mở rộng số lượng hệ điều hành mà công cụ này có thể hỗ trợ và cung cấp thêm nhiều lựa chọn cho các nhà phát triển.

Nền tảng NodeJS đã được hợp nhất với JS Foundation vào năm 2019 để tạo thành OpenJS Foundation gồm những nền tảng giúp quản lý dự án phát triển mã nguồn mở, phân tán của NodeJS.

**Các đặc điểm của NodeJS:**

* **Không đồng bộ và Phát sinh sự kiện(Event Driven):**Tất cả các APIs của thư viện NodeJS đều không đồng bộ, nghĩa là không blocking(khóa). Nó rất cần thiết vì NodeJS khong bao giờ đợi một API trả về dữ liệu. Server chuyển sang một API sau khi gọi nó và có cơ chế thông báo sự kiện của NodeJS giúp Server nhận được phản hồi từ các API gọi trước đó.
* **Chạy rất nhanh:** Dựa trên V8 Javascropt Engine của Google Chome, thư viện NodeJS
* **Đơn giản – hiệu quả:** Tiến trình vận hành của NodeJS đơn giản song lại mang đến hiệu quả cao nhờ ứng dụng mô hình luồng đơn(single thread) với các sự kiện lặp. Các cơ chế sự kiện giúp Server trả lại các phản hồi với một cách không block, đồng thời tăng hiệu quả sử dụng. Các luông đơn cung cấp dịch vụ cho nhiều request hơn hẳn Server truyền thống.
* **Không đệm:** Nền tảng NodeJS không có vùng đệm, tức không cung cấp khả năng lưu trữ dữ liệu buffer.

**Lý do nên sử dụng NodeJS:**

* Ứng dụng phần đông đều được viết bằng ngôn ngữ lập trình javascript – một ngôn ngữ thông dụng, được sử dụng rộng rãi và chạy được trên nhiều trình duyệt, nền tảng, hệ điều hành,...
* NodeJS khá nhẹ nhưng lại hiệu quả nhờ vào vơ chế non-blocing I/O, chạy đa nền tảng trên Server và dùng Event-driven.
* Tương thích với nhiều thiết bị. Bạn có thể chạy các ứng dụng phát triển bởi NodeJS trên bất cứ thiết bị nào dù là Mac, Linux, Windows,..
* Cộng đồng NodeJS khá lớn và được cung cấp miễn phí cho người dùng
* Ứng dụng NodeJS có khả năng chạy đa nền tảng, thiết bị, đồng thời đáp ứng được yêu cầu về thời gian thực.
* NodeJS có tốc độ cực nhanh, xử lý được nhu cầu sử dụng của lượng khách truy cập ‘khổng lồ’ trong thời gian cực ngắn.
* Ứng dụng phát triển bởi NodeJS có khả năng xử lý nhiều yêu cầu truy cập cùng lúc, ’cứu’ website của bạn khỏi nguy cơ bị ‘sập’ khi lương truy cập quá nhiều.

### Giới thiệu về Express Framework

#### Express Framework là gì?

ExpressJS là một Framework nhỏ, nhưng linh hoạt được xây dựng trên nền tảng của Nodejs. Nó cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile

Về các package hỗ trợ: ExpressJS có vô số các package hỗ trợ nên các bạn không phải lo lắng khi làm việc với Framework này.

Về performance: Express cung cấp thêm về các tính năng (feature) để dev lập trình tốt hơn. Chứ không làm giảm tốc độ của NodeJS.

Và hơn hết, các Framework nổi tiếng của NodeJS hiện nay đều sử dụng ExpressJS như một core function, chẳng hạn: SailsJS, MEAN,…

#### Các tính năng của ExpressJS

* **Phát triển máy chủ nhanh hơn:** ExpressJS cung cấp cho bạn nhiều tính năng phổ biến của Node.js dưới dạng hàm có thể dễ dàng sử dụng ở bất kỳ đâu trong chương trình. Điều này sẽ giúp rút ngắn thời gian để viết code.
* **Phần mềm trung gian:** Phần mềm trung gian là một phần trong chương trình cho phép truy cập vào cơ sở dữ liệu, xem xét yêu cầu của khách hàng và các phần mềm trung gian khác. Tính năng này chịu trách nhiệm chính cho việc tổ chức chức năng khác nhau của ExpressJS.
* **Định tuyến:** ExpressJS cung cấp một cơ chế định tuyến nâng cao giúp duy trì trạng thái của trang web.
* **Khuôn mẫu:** ExpressJS cung cấp các công cụ tạo khuôn mẫu cho phép các nhà phát triển tạo nội dung động trên các trang web bằng việc xây dựng các mẫu HTML ở phía máy chủ.
* **Gỡ lỗi:** Gỡ lỗi là yếu tố quan trọng để phát triển các ứng dụng web. ExpressJS giúp gỡ lỗi dễ dàng hơn bằng cách cung cấp một cơ chế có khả năng xác định chính xác phần ứng dụng web có lỗi.

#### Ưu điểm và nhược điểm khi sử dụng ExpressJS

1. **Ưu điểm của ExpressJS:**

* Rất dễ học, chỉ cần bạn biết JavaScript, bạn sẽ không cần phải học một ngôn ngữ mới để học ExpressJS
* Giúp cho việc phát triển back-end dễ dàng hơn nhiều khi sử dụng ExpressJS
* Mã JavaScript được diễn giải thông qua Google V8 JavaScript Engine của Node.js. Do đó, mã sẽ được thực hiện một cách nhanh chóng và dễ dàng.
* ExpressJS rất đơn giản để tùy chỉnh và sử dụng theo nhu cầu.
* Cung cấp một module phần mềm trung gian linh hoạt và rất hữu ích để thực hiện các tác vụ bổ sung theo phản hồi và yêu cầu.
* Hỗ trợ phát triển ứng dụng theo mô hình MVC, đây là mô hình phổ biến cho việc lập trình web hiện nay.

1. **Nhược điểm của ExpressJS:**

Nếu bạn cần xử lý các ứng dụng tốn tài nguyên CPU như encoding video, convert file, decoding encryption... hoặc các ứng dụng tương tự như vậy thì không nên dùng NodeJS (Lý do: NodeJS được viết bằng C++ & Javascript, nên phải thông qua thêm 1 trình biên dịch của NodeJS sẽ lâu hơn 1 chút ).

### Giới thiệu về MongoDB

MongoDB là một hệ quản trị cơ sở dữ liệu NoSQL, là một lựa chọn phổ biến trong cộng đồng phát triển ứng dụng. Nó chú trọng vào tính linh hoạt và khả năng mở rộng, giúp đáp ứng nhu cầu lưu trữ dữ liệu động và thay đổi liên tục.

**Cấu Trúc Dữ Liệu và Bộ Sưu Tập:**

* Tài Liệu BSON: MongoDB lưu trữ dữ liệu dưới dạng các tài liệu BSON, mỗi tài liệu là một bản ghi JSON-like. Các tài liệu có thể chứa các trường với định dạng và kiểu dữ liệu đa dạng.
* Bộ Sưu Tập: Dữ liệu được tổ chức vào các bộ sưu tập, tương tự như bảng trong hệ quản trị cơ sở dữ liệu quan hệ. Mỗi bộ sưu tập có thể chứa nhiều tài liệu có cấu trúc khác nhau.

**Tính Linh Hoạt và Khả Năng Mở Rộng:**

* Không Yêu Cầu Schema Cứng: MongoDB cho phép tài liệu trong cùng một bộ sưu tập có cấu trúc khác nhau, không yêu cầu một schema cứng.
* Chia Nhỏ Dữ Liệu (Sharding): Hỗ trợ chia nhỏ dữ liệu để tăng khả năng mở rộng và phân phối dữ liệu trên nhiều máy chủ.

**Truy Vấn và Tìm Kiếm:**

* Ngôn Ngữ Truy Vấn JSON-like: Truy vấn MongoDB sử dụng cú pháp dựa trên JSON, giúp truy vấn và tìm kiếm dữ liệu dễ dàng.
* Chỉ Mục (Indexes): Hỗ trợ chỉ mục để tối ưu hóa quá trình truy xuất dữ liệu.

**Khả Năng Mở Rộng và Hỗ Trợ Ngôn Ngữ:**

* Hỗ Trợ Nhiều Ngôn Ngữ Lập Trình: MongoDB có các driver cho nhiều ngôn ngữ lập trình, giúp tích hợp dễ dàng với các ứng dụng được phát triển bằng Node.js, PHP, Python, và nhiều ngôn ngữ khác.
* Hệ Thống Lớn Các Hàm Tiện Ích: Với tốc độ và tính ổn định, MongoDB là sự lựa chọn phù hợp cho ứng dụng có truy cập cơ sở dữ liệu mạnh trên internet.

**Bảo Mật và An Toàn:**

* Tính Bảo Mật Cao: MongoDB cung cấp các tính năng bảo mật như xác thực, kiểm soát truy cập, và mã hóa dữ liệu để bảo vệ thông tin.

MongoDB, với tính linh hoạt, khả năng mở rộng, và hỗ trợ đa ngôn ngữ, là một giải pháp mạnh mẽ cho việc lưu trữ và quản lý dữ liệu trong các ứng dụng hiện đại.

# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Đặc tả bài toán

Ngày nay, việc sở hữu một website bán hàng là điều không thể thiếu đối với bất kỳ doanh nghiệp nào, đặc biệt là trong lĩnh vực trang sức. Khi khách hàng có nhu cầu mua sắm, họ thường có xu hướng tìm kiếm thông tin sản phẩm online trước khi quyết định mua. Website bán trang sức chính là công cụ hiệu quả giúp doanh nghiệp tiếp cận khách hàng tiềm năng, gia tăng nhận diện thương hiệu và thúc đẩy doanh số bán hàng.

So với việc đầu tư vào một cửa hàng truyền thống tốn kém chi phí thuê mặt bằng, nhân viên, trang thiết bị, v.v., việc sở hữu website bán trang sức mang lại nhiều lợi ích vượt trội:

* **Tiết kiệm chi phí:** Chi phí xây dựng và vận hành website rẻ hơn nhiều so với chi phí thuê mặt bằng, trang trí cửa hàng, tuyển dụng và trả lương nhân viên.
* **Mở rộng phạm vi khách hàng:** Website giúp doanh nghiệp tiếp cận khách hàng tiềm năng trên toàn quốc, thậm chí là quốc tế, vượt qua giới hạn địa lý của cửa hàng truyền thống.
* **Tăng cường khả năng nhận diện thương hiệu:** Website là kênh giao tiếp trực tiếp với khách hàng, giúp doanh nghiệp giới thiệu thương hiệu, sản phẩm và dịch vụ đến khách hàng một cách hiệu quả.
* **Cải thiện trải nghiệm mua sắm:** Khách hàng có thể dễ dàng tham khảo thông tin sản phẩm, so sánh giá cả và đặt hàng trực tuyến mọi lúc mọi nơi.
* **Quản lý đơn hàng hiệu quả:** Website giúp doanh nghiệp quản lý đơn hàng, theo dõi tình trạng giao hàng và thanh toán một cách dễ dàng và hiệu quả.
* **Tăng doanh số bán hàng:** Website là kênh bán hàng 24/7, giúp doanh nghiệp tăng doanh số bán hàng bất kể ngày đêm.

**Yêu cầu chức năng của hệ thống website bán trang sức:**

* **Đối với khách hàng:**
* Đăng nhập
* Đăng ký
* Xem danh mục các loại sản phẩm
* Tìm kiếm sản phẩm
* Xem thông tin chi tiết sản phẩm
* Xem các bình luận về sản phẩm
* Bình luận sản phẩm
* Quản lý giỏ hàng (Thêm, sửa, xoá sản phẩm trong giỏ hàng)
* Đặt hàng
* Quản lý thông tin cá nhân
* Đổi mật khẩu
* Quên mật khẩu
* Theo dõi đơn hàng
* **Đối với quản trị viên:**
* Quản lý danh mục sản phẩm
* Quản lý sản phẩm (thêm, sửa, xóa)
* Quản lý đơn hàng
* Quản lý khách hàng
* Quản lý kho
* Thống kê (thống kê doanh thu, đơn hàng theo tháng, ngày, năm)

**Yêu cầu phi chức năng của hệ thống:**

* Giao diện đẹp mắt, dễ sử dụng, thân thiện với người dùng
* Tốc độ truy cập nhanh chóng, ổn định
* Tương thích với nhiều thiết bị (máy tính, điện thoại, máy tính bảng)
* Dễ dàng bảo trì, nâng cấp
* An toàn bảo mật

Website bán trang sức là công cụ thiết yếu giúp doanh nghiệp gia tăng lợi thế cạnh tranh trong thị trường ngày càng sôi động. Việc đầu tư vào một website bán trang sức hiệu quả sẽ mang lại nhiều lợi ích cho doanh nghiệp, góp phần thúc đẩy sự phát triển bền vững trong tương lai.

## Phân tích và thiết kế

### Mô hình hoá chức năng:

|  |  |  |
| --- | --- | --- |
| STT | Tên Actor | Mô tả |
| 1 | Admin  (Quản trị viên) | Admin có toàn quyền tương tác với hệ  thống, quản trị hệ thống |
| 2 | User(Khách hàng) | User có thể đăng ký, đăng nhập, xem thông tin về sản phẩm, bình luận về sản phẩm, tìm kiếm sản phẩm, đặt hàng. |

Bảng ‑:Bảng mô hình hoá chức năng

### Biểu đồ Use case



Hình ảnh ‑: Biểu đồ Use case tổng quan

### Mô tả các use case

#### Use case Đăng ký

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng đăng ký tài khoản để đăng nhập vào hệ thống website thực hiện các thao tác mua hàng trực tuyến qua Internet. |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Không |
| Luồng sự kiện chính | Use case này bắt đầu khi khách hàng click vào “Đăng ký”. Hệ thống hiển thị màn hình nhập các thông tin đăng ký tài khoản.  Khách hàng nhập các thông tin theo yêu cầu của hệ thống và click vào nút “Đăng ký”. Hệ thống lưu thông tin khách hàng vào bảng “Users” và hiển thị lên màn hình thông báo đăng ký thành công. Use case kết thúc. |
| Luồng rẽ nhánh | Nếu nhập không hợp lệ một trong các trường dữ liệu thì hệ thống hiển thị thông báo lỗi, khách hàng có thể nhập lại trường dữ liệu đó hoặc thoát khỏi màn hình đăng ký.  Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| Hậu điều kiện | Nếu Use case thành công, hệ thống sẽ lưu dữ liệu mới của khách hàng vào bảng “Users”. Ngược lại hiển thị màn hình thông báo lỗi. |

Bảng ‑: Mô tả chi tiết use case Đăng ký



Hình ảnh ‑: Biểu đồ trình tự use case đăng ký

#### Use case Đăng nhập

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng đăng nhập vào hệ thống website thực hiện các thao tác mua hàng trực tuyến qua Internet. |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Đã có tài khoản của hệ thống |
| Luồng sự kiện chính | Use case này bắt đầu khi khách hàng kích vào nút “Đăng nhập” trên giao diện của trang web. Hệ thống hiển thị một màn hình đăng nhập.  Khách hàng nhập tài khoản và mật khẩu và nhấn “Đăng nhập”. Hệ thống kiểm tra thông tin chi tiết trong bảng “Users” và hiển thị thông báo. Use case kết thúc. |
| Luồng rẽ nhánh | Khách hàng nhập sai thông tin tài khoản hoặc mật khẩu thì hệ thống hiển thị thông báo lỗi, khách hàng có thể nhập lại thông tin tài khoản hoặc thoát khỏi màn hình đăng nhập.  Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| Hậu điều kiện | Cho phép sử dụng các chức năng của hệ thống theo quyền của tài khoản đã đăng ký. |

Bảng ‑: Mô tả chi tiết use case Đăng ký



Hình ảnh ‑: Biểu đồ trình tự use case đăng nhập

#### Use case Xem danh mục

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng xem danh mục sản phẩm |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Không |
| Luồng sự kiện chính | Use case này bắt đầu khi khách hàng truy cập vào website trên menu chính. Hệ thống lấy thông tin chi tiết của tên các danh mục sản phẩm từ bảng “Categories” và hiển thị ra một danh sách tên các danh mục sản phẩm.  Khách hàng click vào một tên danh mục sản phẩm bất kỳ. Hệ thống sẽ lấy thông tin chi tiết của các sản phẩm thuộc danh mục đó và hiển thị lên màn hình. Use case kết thúc. |
| Luồng rẽ nhánh | Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| Hậu điều kiện | Không |

Bảng ‑: Mô tả chi tiết use case xem danh mục



Hình ảnh ‑: Biểu đồ trình tự use case xem danh mục

#### Use case Tìm kiếm sản phẩm

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng tìm kiếm các sản phẩm trên Website. |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Không |
| Luồng sự kiện chính | Use case này bắt đầu khi khách hàng nhập tên sản phẩm và kích vào nút tìm kiếm sản phẩm trên giao diện của website.  Hệ thống kiểm tra thông tin bảng “Products” và hiển thị màn hình chứa danh sách các sản phẩm có tên giống như khách hàng đã nhập bao gồm hình ảnh, tên và giá tiền cho mỗi sản phẩm. Use case kết thúc. |
| Luồng rẽ nhánh | Nếu không tìm thấy kết quả thì hệ thống hiển thị một thông báo không có sản phẩm tồn tại.  Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| Hậu điều kiện | Không |

Bảng ‑: Mô tả chi tiết use case tìm kiếm sản phẩm



Hình ảnh ‑: Biểu đồ trình tự use case tìm kiếm sản phẩm

#### Use case Xem thông tin chi tiết sản phẩm

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng xem thông tin chi tiết của sản phẩm |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Không |
| Luồng sự kiện chính | Use case này bắt đầu khi khách hàng click vào nút “Sản phẩm” trên menu chính. Hệ thống hiển thị danh sách các sản phẩm theo danh mục ra màn hình.  Khách hàng click vào một sản phẩm bất kỳ. Hệ thống lấy thông tin chi tiết của sản phẩm từ bảng “Products” và hiển thị một màn hình gồm hình ảnh, tên sản phẩm, giá tiền, mã sản phẩm và mô tả của sản phẩm. Use case kết thúc. |
| Luồng rẽ nhánh | Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| Hậu điều kiện | Không |

Bảng ‑: Mô tả chi tiết use case xem thông tin chi tiết sản phẩm



Hình ảnh ‑: Biểu đồ trình tự xem thông tin chi tiết sản phẩm

#### Use case Xem các bình luận về sản phẩm

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng bình luận về sản phẩm trên website. |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Không |
| Luồng sự kiện chính | Use case này bắt đầu khi khách hàng click vào một sản phẩm bất kỳ. Hệ thống lấy thông tin chi tiết các bình luận trong bảng “Comments” hiển thị ra màn hình. Use case kết thúc |
| Luồng rẽ nhánh | Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm  nào trong quá trình thực hiện use case nếu không kết nối |
| Hậu điều kiện | Không |

Bảng ‑: Use case xem các bình luận về sản phẩm



Hình ảnh ‑: Biểu đồ trình tự use case xem các bình luận sản phẩm

#### Use case Bình luận sản phẩm

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng bình luận về sản phẩm trên website. |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Khách hàng đã đăng nhập hệ thống |
| Luồng sự kiện chính | Use case này bắt đầu. Khách hàng nhập bình luận và chọn sao đánh giá về chất lượng sản phẩm cũng như dịch vụ của cửa hàng và click vào nút “Đăng bình luận”.  Hệ thống kiểm tra thông tin bảng “Comments”, lưu thông tin bình luận của khách hàng và hiển thị ra màn hình. Use case kết thúc. |
| Luồng rẽ nhánh | Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm  nào trong quá trình thực hiện use case nếu không kết nối |
| Hậu điều kiện | Không |

Bảng ‑: Mô tả chi tiết use case bình luận về sản phẩm



Hình ảnh ‑: Biểu đồ trình tự use case bình luận sản phẩm

#### Use case Quản lý giỏ hàng

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng quản lý giỏ hàng của mình trên website. |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Khách hàng đã đăng nhập hệ thống. Có ít nhất một sản phẩm trong giỏ hàng |
| Luồng sự kiện chính | Use case này bắt đầu khi khách hàng click vào nút “Giỏ hàng” trên menu chính.  Hệ thống lấy thông tin từ bảng “Carts” và hiển thị danh sách các mặt hàng có trong giỏ hàng ra màn hình. Khách hàng có thể thực hiện thay đổi số lượng hoặc xóa sản phẩm ra khỏi giỏ hàng.  Hệ thống cập nhật lại giỏ hàng nếu có sự thay đổi về sản phẩm của khách hàng. Use case kết thúc. |
| Luồng rẽ nhánh | Nếu trong giỏ hàng chưa có sản phẩm nào được chọn thì hệ thống hiển thị thông báo không có sản phẩm nào trong giỏ hàng và gợi ý chọn sản phẩm thêm vào giỏ.  Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| Hậu điều kiện | Không |

Bảng ‑: Mô tả chi tiết use case quản lý giỏ hàng



Hình ảnh ‑: Mô tả chi tiết use case quản lý giỏ hàng

#### Use case Đặt hàng

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng đặt hàng trực tuyến qua internet trên website. |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Khách hàng đã đăng nhập hệ thống, Sản phẩm đã tồn tại trong giỏ hàng. |
| Luồng sự kiện chính | Use case này bắt đầu khi khách hàng click vào nút “Thanh toán” trong giỏ hàng và hệ thống lấy thông tin danh sách sản phẩm trong bảng “Carts”, thông tin khách hàng trong bảng “Users”.  Khách hàng nhập thông tin của đơn hàng sau đó nhấn nút đặt hàng và hệ thống lưu thông tin đơn hàng vào bảng “Order”. Use case kết thúc. |
| Luồng rẽ nhánh | Nếu khách hàng chưa đăng nhập thì hệ thống yêu cầu khách hàng cần phải đăng nhập hệ thống, sau đó hiển thị màn hình đăng nhập.  Không kết nối được với cơ sở dữ liệu : tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc |
| Hậu điều kiện | Không |

Bảng ‑: Mô tả chi tiết use case đặt hàng



Hình ảnh ‑: Biểu đồ trình tự use case đặt hàng

#### Use case Quản lý thông tin cá nhân

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng xem và thay đổi thông tin cá nhân trên website. |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Khách hàng đã đăng nhập hệ thống. |
| Luồng sự kiện chính | Use case này bắt đầu khi khách hàng click vào nút “Thông tin cá nhân” trong menu tài khoản và hệ thống lấy thông tin cá nhân trong bảng “Users” hiển thị ra màn hình.  Khách hàng có thể nhập thông tin cá nhân mới sau đó nhấn “Thay đổi”, hệ thống sẽ lưu thông tin cá nhân mới trong bảng “Users” và hệ thống hiển thị thông tin cá nhân mới ra màn hình.Use case kết thúc. |
| Luồng rẽ nhánh | Không kết nối được với cơ sở dữ liệu : tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc |
| Hậu điều kiện | Không |

Bảng ‑: Mô tả chi tiết use case quản lý thông tin cá nhân



Hình ảnh ‑: Biểu đồ trình tự use case quản lý thông tin cá nhân

#### Use case Đổi mật khẩu

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng thay đổi mật khẩu trên website. |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Khách hàng đã đăng nhập hệ thống. |
| Luồng sự kiện chính | Use case này bắt đầu khi khách hàng click vào nút “Thay đổi mật khẩu” trong thông tin cá nhân. Khách hàng có thể nhập mật khẩu cũ và mật khẩu mới sau đó nhấn “Lưu” và hệ thống sẽ lưu thông tin mật khẩu mới vào trong bảng “Users”.Use case kết thúc. |
| Luồng rẽ nhánh | Khách hàng nhập sai mật khẩu cũ, hoặc mật khẩu mới không trùng với mật khẩu xác nhận thì hệ thống hiển thị thông báo lỗi, khách hàng có thể nhập lại mật khẩu cũ hoặc thoát khỏi màn hình đăng nhập  Không kết nối được với cơ sở dữ liệu : tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc |
| Hậu điều kiện | Không |

Bảng ‑: Mô tả chi tiết use case đổi mật khẩu



Hình ảnh ‑: Biểu đồ trình tự use case đổi mật khẩu

#### Use case Quên mật khẩu

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng đặt lại mật khẩu trên website. |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Không |
| Luồng sự kiện chính | Use case này bắt đầu khi khách hàng click vào nút “Quên mật khẩu” trong phần đăng nhập. Khách hàng có thể nhập email của tài khoản sau đó nhấn “Gửi link reset mật khẩu” và hệ thống sẽ gửi link reset mật khẩu đến email của khách hàng.  Khách hàng sau khi click vào link reset mật khẩu, sau đó nhập mật khẩu mới vào nhấn “Xác nhận” và hệ thống sẽ lưu lại mật khẩu mới trong bảng “Users”. Use case kết thúc. |
| Luồng rẽ nhánh | Khách hàng nhập emai của tài khoản nếu email không có trong bảng “Users” thì hệ thống hiển thị thông báo lỗi, khách hàng có thể nhập lại email hoặc thoát khỏi màn hình đăng nhập  Không kết nối được với cơ sở dữ liệu : tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc |
| Hậu điều kiện | Không |

Bảng ‑: Mô tả chi tiết use case quên mật khẩu



Hình ảnh ‑: Biểu đồ trình tự use case quên mật khẩu

#### Use case Theo dõi đơn hàng

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép khách hàng theo dõi trạng thái của đơn hàng |
| Người thực hiện | Khách hàng |
| Tiền điều kiện | Khách hàng đã đăng nhập hệ thống |
| Luồng sự kiện chính | Use case này bắt đầu khi khách hàng click vào nút “Đơn hàng của tôi” trong menu tài khoản và hệ thống hiển thị danh sách các đơn hàng của khách hàng đó từ bảng “Orders”. Use case kết thúc. |
| Luồng rẽ nhánh | Không kết nối được với cơ sở dữ liệu : tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc |
| Hậu điều kiện | Không |

Bảng ‑: Mô tả chi tiết use case theo dõi đơn hàng



Hình ảnh ‑: Biểu đồ trình tự use case theo dõi đơn hàng

#### Use case Quản lý danh mục sản phẩm

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép quản trị viên thêm, sửa, xóa thông tin của các danh mục sản phẩm |
| Người thực hiện | Quản trị viên |
| Tiền điều kiện | Đăng nhập vào hệ thống với quyền quản trị |
| Luồng sự kiện chính | Use case này bắt đầu khi quản trị viên kích vào nút “Danh mục” trên thanh menu quản trị. Hệ thống sẽ lấy danh sách thông tin các danh mục sản phẩm và hiển thị lên màn hình.  **Thêm mới danh mục**  Quản trị viên kích vào nút “Thêm Mới” trên cửa sổ danh sách danh mục . Hệ thống hiển thị màn hình yêu cầu nhập thông tin chi tiết cho danh mục mới.  Quản trị viên nhập thông tin cho danh mục mới và kích vào nút “Thêm Mới”. Hệ thống sẽ tạo một bản ghi mới trong bảng “Categories” và hiển thị danh sách các danh mục sản phẩm được cập nhật ra màn hình.  **Sửa danh mục**  Quản trị viên kích vào nút “Sửa” trên một dòng bất kỳ. Hệ thống lấy thông tin cũ của danh mục sản phẩm và hiển thị lên màn hình.  Quản trị viên nhập thông tin mới của danh mục sản phẩm và kích vào nút “Cập Nhật”. Hệ thống sẽ lưu thông tin mới vào bảng DanhMuc và hiển thị danh sách đã cập nhật ra màn hình.  **Xóa danh mục**  Quản trị viên ấn vào nút “Xóa” trên một dòng sản phẩm, hệ thống sẽ hiển thị một yêu cầu xác nhận xóa.  Quản trị viên ấn vào nút “Đồng Ý”. Hệ thống xóa danh mục ra khỏi bảng DanhMuc và hiển thị danh sách danh mục đã cập nhật.  Use case kết thúc. |
| Mô tả | Use case này cho phép quản trị viên thêm, sửa, xóa thông tin của các danh mục sản phẩm |
| Người thực hiện | Quản trị viên |

Bảng ‑: Mô tả chi tiết use case quản lý danh mục sản phẩm



Hình ảnh ‑: Biểu đồ trình tự use case quản lý danh mục sản phẩm

#### Use case Quản lý sản phẩm

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép quản trị viên thêm, sửa, xóa thông tin của các sản phẩm |
| Người thực hiện | Quản trị viên |
| Tiền điều kiện | Đăng nhập vào hệ thống với quyền quản trị |
| Luồng sự kiện chính | Use case này bắt đầu khi quản trị viên kích vào nút “Sản phẩm” trên thanh menu quản trị. Hệ thống sẽ lấy danh sách thông tin sản phẩm và hiển thị lên màn hình.  **Thêm sản phẩm**  Quản trị viên kích vào nút “Thêm Mới” trên cửa sổ danh sách sản phẩm . Hệ thống hiển thị màn hình yêu cầu nhập thông tin chi tiết cho sản phẩm.  Quản trị viên nhập thông tin cho sản phẩm mới và kích vào nút “Thêm Mới”. Hệ thống sẽ tạo một bản ghi mới trong bảng “Products”và hiển thị danh sách các sản phẩm được cập nhật ra màn hình.  **Sửa sản phẩm**  Quản trị viên kích vào nút “Sửa” trên một dòng sản phẩm bất kỳ. Hệ thống lấy thông tin cũ của sản phẩm và hiển thị lên màn hình.  Quản trị viên nhập thông tin mới của sản phẩm và kích vào nút “Cập Nhật”. Hệ thống sẽ lưu thông tin mới vào bảng “Products”và hiển thị danh sách đã cập nhật ra màn hình.  **Xóa sản phẩm**  Quản trị viên ấn vào nút “Xóa” trên một dòng sản phẩm, hệ thống sẽ hiển thị một yêu cầu xác nhận xóa.  Quản trị viên ấn vào nút “Đồng Ý”. Hệ thống xóa sản phẩm ra khỏi bảng “Products” và hiện danh sách sản phẩm đã cập nhật.  Use case kết thúc. |
| Luồng rẽ nhánh | Nếu nhập không hợp lệ một trong các trường dữ liệu thì hệ thống hiển thị thông báo lỗi, quản trị viên có thể nhập lại trường dữ liệu đó.  Khi sản phẩm có tồn tại trong các hóa đơn thì hệ thống hiển thị thông báo lỗi khi xóa sản phẩm.  Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| Hậu điều kiện | Nếu use case kết thúc thành công thì thông tin về sản phẩm sẽ được cập nhật trong cơ sở dữ liệu. |

Bảng ‑: Mô tả chi tiết use case quản lý sản phẩm



Hình ảnh ‑: Biểu đồ trình tự use case quản lý sản phẩm

#### Use case Quản lý đơn hàng

|  |  |
| --- | --- |
| Mô tả | Use case này cho phép quản trị viên quản lý những đơn hàng của khách đã đặt. |
| Tác nhân | Quản trị viên |
| Tiền điều kiện | Đăng nhập vào hệ thống với quyền quản trị |
| Luồng sự kiện chính | Use case này bắt đầu khi quản trị viên kích vào nút “Đơn hàng” trên menu chính của giao diện admin.  Hệ thống kiểm tra thông tin chi tiết dữ liệu trong bảng “Orders” rồi hiển thị lên màn hình danh sách những đơn hàng của khách hàng.  Quản trị viên có thể sử dụng các thao tác như xem chi tiết, xác nhận, huỷ, hoàn thành đơn hàng  Hệ thống kiểm tra tính hợp lệ và lưu các thay đổi vào trong cơ sở dữ liệu. Use case kết thúc. |
| Luồng rẽ nhánh | Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| Hậu điều kiện | Không |

Bảng ‑: Mô tả chi tiết use case quản lý đơn hàng



Hình ảnh ‑: Biểu đồ trình tự use case quản lý đơn hàng

#### Use case Quản lý khách hàng

|  |  |
| --- | --- |
| Mô tả | Cho phép Quản trị viên quản lý danh sách khách hàng |
| Người thực hiện | Quản trị viên |
| Tiền điều kiện | Đăng nhập vào hệ thống với quyền quản trị |
| Luồng sự kiện chính | Use case này bắt đầu khi quản trị viên kích vào nút “Khách hàng” trên thanh menu quản trị. Hệ thống sẽ lấy danh sách thông tin khách hàng và hiển thị lên màn hình.  **Sửa thông tin**  Quản trị viên click “Sửa” của một dòng thông tin khách hàng và hệ thống hiển thị thông tin của khách hàng ra màn hình. Quản trị viên có thể nhập thông tin mới của khách hàng sau đó nhấn nút “Lưu” và hệ thông sẽ lưu lại thông tin mới của khách hàng trong bảng “Users”  **Xóa thông tin**   1. Quản trị viên ấn vào nút “Xóa” trên một dòng thông tin bất kỳ, hệ thống sẽ hiển thị một yêu cầu xác nhận xóa. 2. Quản trị viên ấn vào nút “Đồng Ý”. Hệ thống xóa thông tin khách hàng ra khỏi bảng “Users” và hiển thị danh sách khách hàng đã cập nhật.   Use case kết thúc. |
| Luồng rẽ nhánh | Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi  và use case kết thúc. |
| Hậu điều kiện | Nếu use case kết thúc thành công thì thông tin khách hàng sẽ được cập nhật trong cơ sở dữ liệu. |

Bảng ‑: Mô tả chi tiết use case quản lý khách hàng



Hình ảnh ‑: Biểu đồ trình tự use case quản lý khách hàng

#### Use case Quản lý kho

|  |  |
| --- | --- |
| Mô tả | Cho phép quản trị viên quản lý danh sách kho |
| Người thực hiện | Quản trị viên |
| Tiền điều kiện | Đăng nhập vào hệ thống với quyền quản trị |
| Luồng sự kiện chính | Use case này bắt đầu khi Quản trị viên kích vào nút “Kho” trên thanh menu quản trị. Hệ thống sẽ lấy danh sách thông tin kho và hiển thị lên màn hình.  **Tăng, giảm số lượng**  Quản trị viên click “Số lượng” của một kho hàng và hệ thống hiển thị màn hình thông tin của kho ra màn hình. Quản trị viên có thể nhập số lượng tăng, giảm của kho sau đó nhấn nút Tăng hoặc giảm và hệ thông sẽ lưu lại thông tin của kho trong bảng “Inventories”  Use case kết thúc. |
| Luồng rẽ nhánh | Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi  và use case kết thúc. |
| Hậu điều kiện | Nếu use case kết thúc thành công thì thông tin khách hàng sẽ được cập nhật trong cơ sở dữ liệu. |

Bảng ‑: Mô tả chi tiết use case quản lý kho



Hình ảnh ‑: Biểu đồ trình tự use case quản lý kho

#### Use case Thống kê

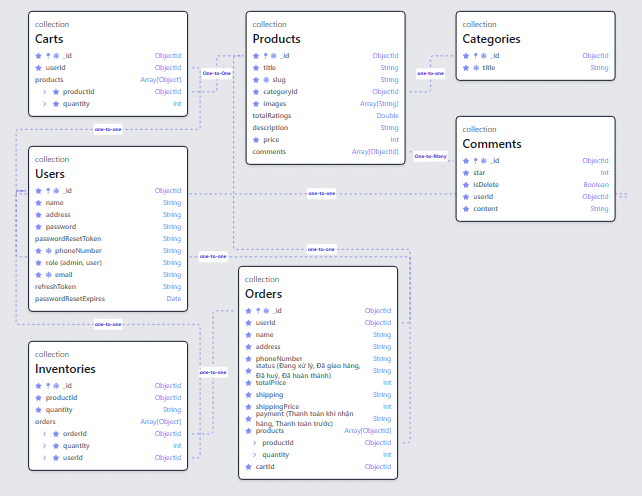
|  |  |
| --- | --- |
| Mô tả | Use case này cho phép Quản trị viên thống kê doanh thu ,đơn hàng trong các khoảng thời gian. |
| Tác nhân | Quản trị viên |
| Tiền điều kiện | Đăng nhập vào hệ thống với quyền quản trị |
| Luồng sự kiện chính | Use case này bắt đầu khi quản trị viên kích vào nút “Thống kê” trên giao diện admin của website.  Hệ thống hiển thị màn hình chứa các khoảng thời gian cho phép quản trị viên nhập thời gian. Quản trị viên nhập thời gian và hệ thống kiểm tra tính hợp lệ của các trường dữ liệu, lấy thông tin chi tiết của các đơn hàng trong bảng “Orders” rồi hiển thị số lượng của đơn hàng được trong khoảng thời gian được chọn. Use case kết thúc. |
| Luồng rẽ nhánh | Không kết nối được cơ sở dữ liệu: tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc. |
| Hậu điều kiện | Không |

Bảng ‑: Mô tả chi tiết use case thống kê

****

Hình ảnh ‑: Biểu đồ trình tự use case thống kê

### Mô hình cơ sở dữ liệu hệ thống



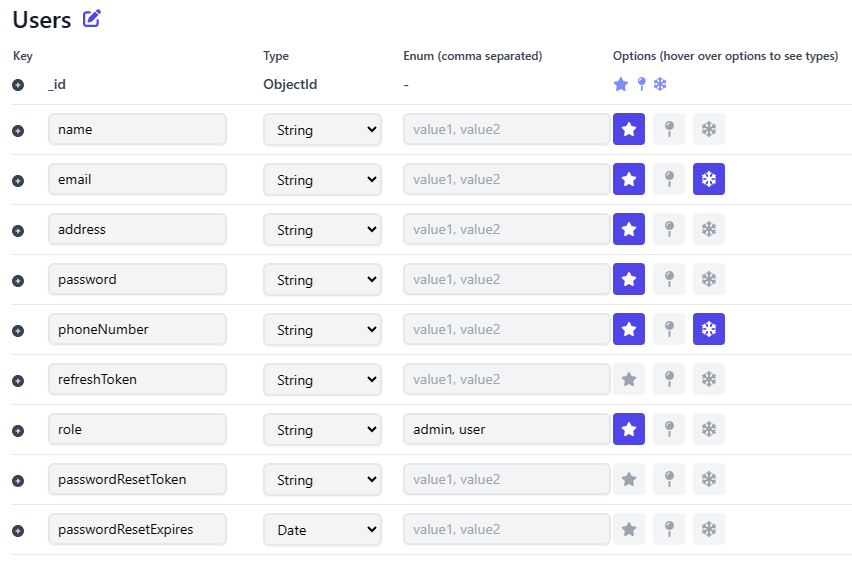
Hình ảnh ‑: Thiết kế cơ sở dữ liệu

**Giải thích ký hiệu:**

****

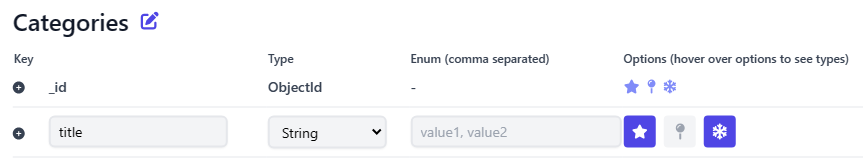
### Thiết kế các bảng cơ sở dữ liệu

#### Bảng Users



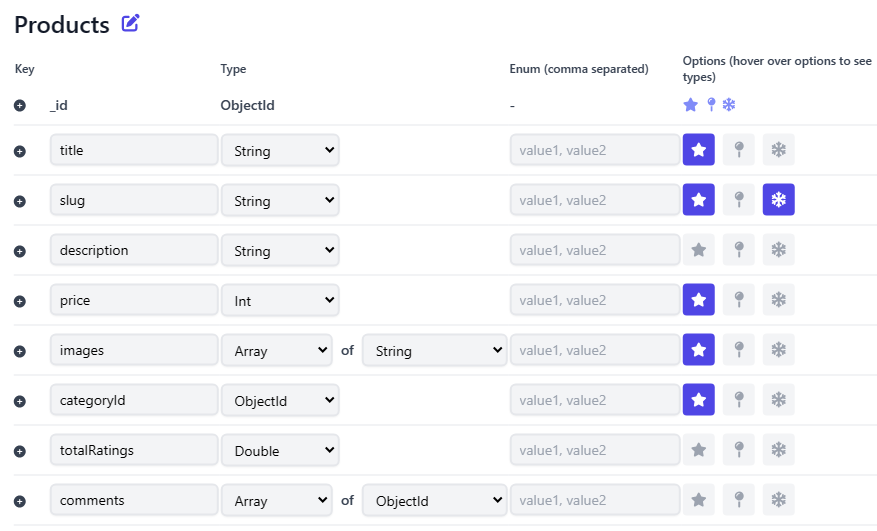
Hình ảnh ‑: Bảng Users

#### Bảng Categories



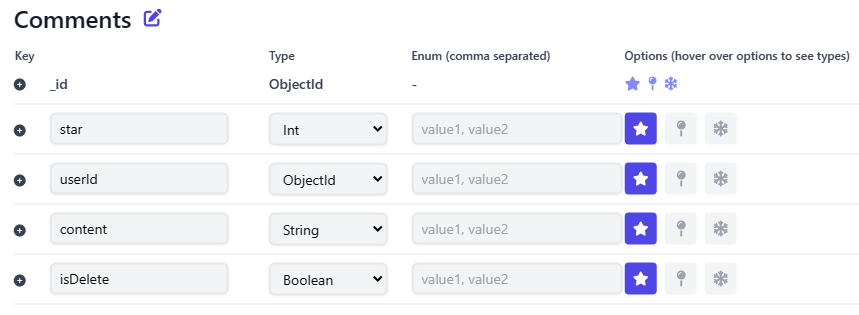
Hình ảnh ‑: Bảng Categories

#### Bảng Products



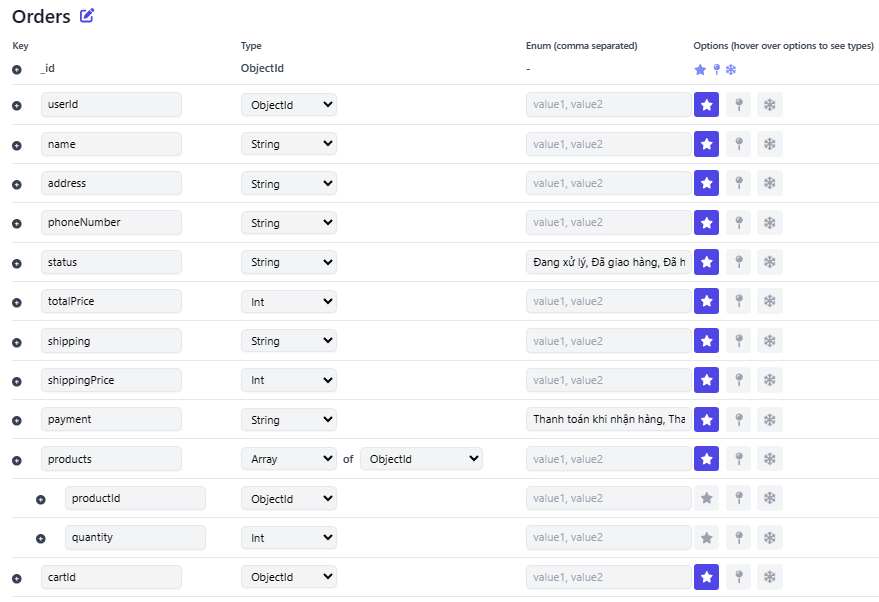
Hình ảnh ‑: Bảng Products

#### Bảng Comments



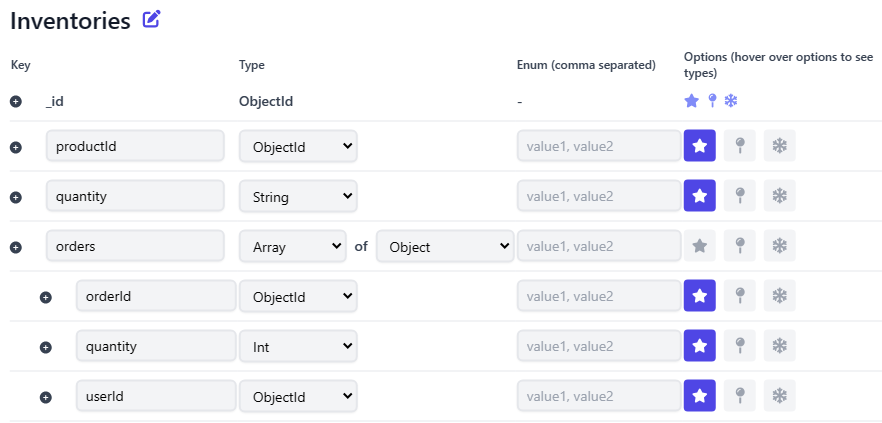
Hình ảnh ‑: Bảng Comments

#### Bảng Orders



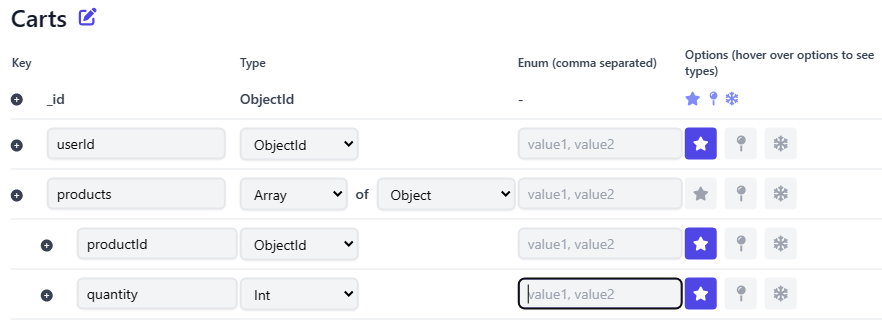
Hình ảnh ‑: Bảng Orders

#### Bảng Inventories



Hình ảnh ‑: Bảng Inventories

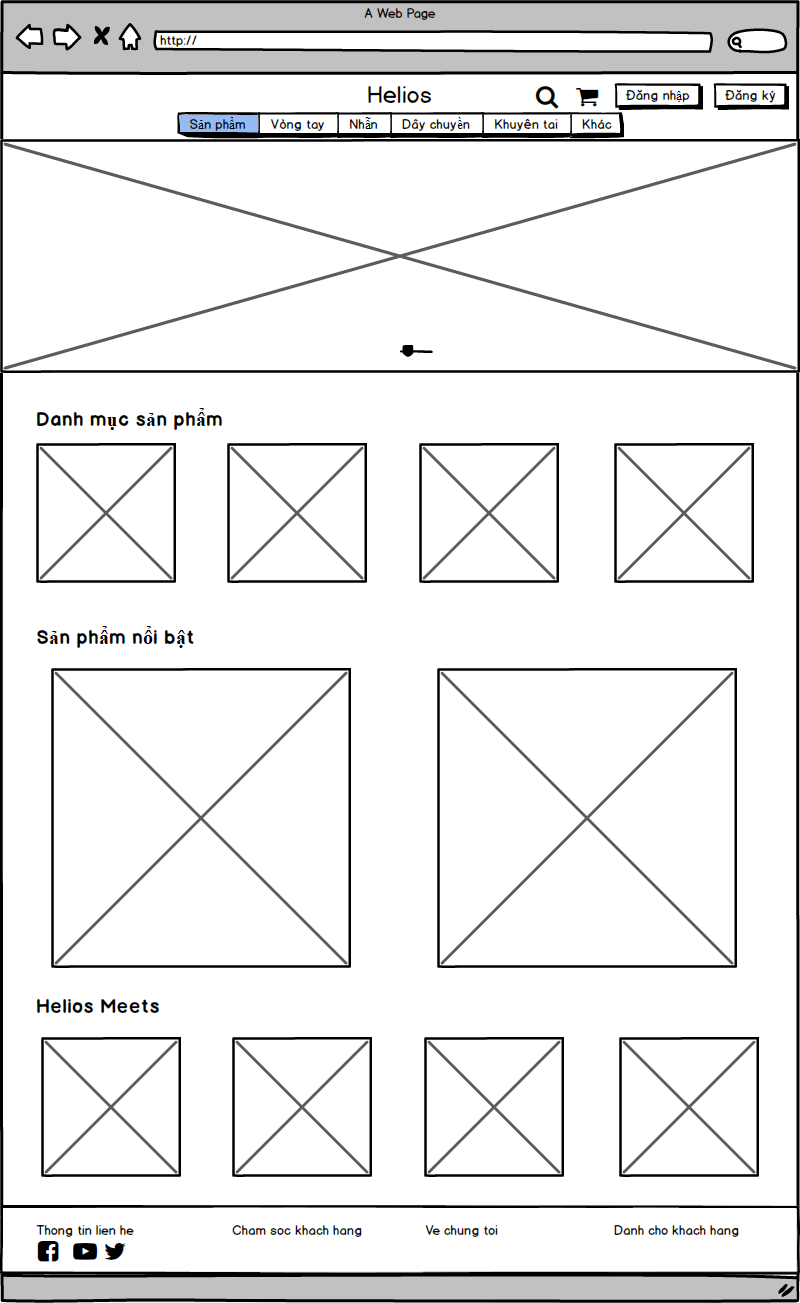
#### Bảng Carts



Hình ảnh ‑: Bảng Carts

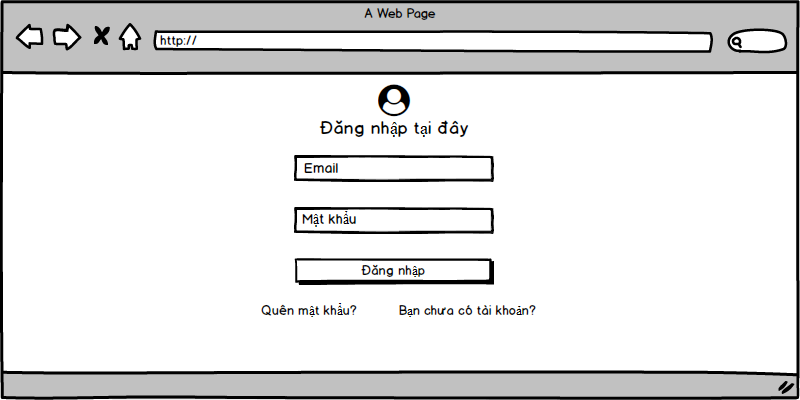
### Thiết kế giao diện

#### Màn hình trang chủ



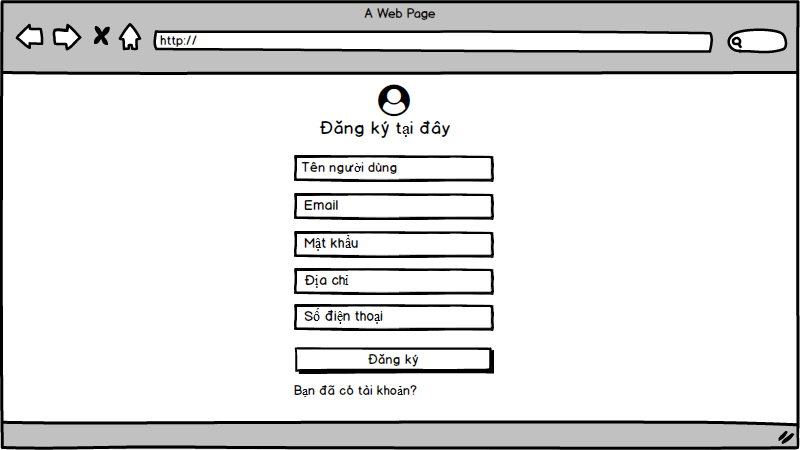
Hình ảnh ‑: Thiết kế màn hình trang chủ

#### Màn hình đăng nhập



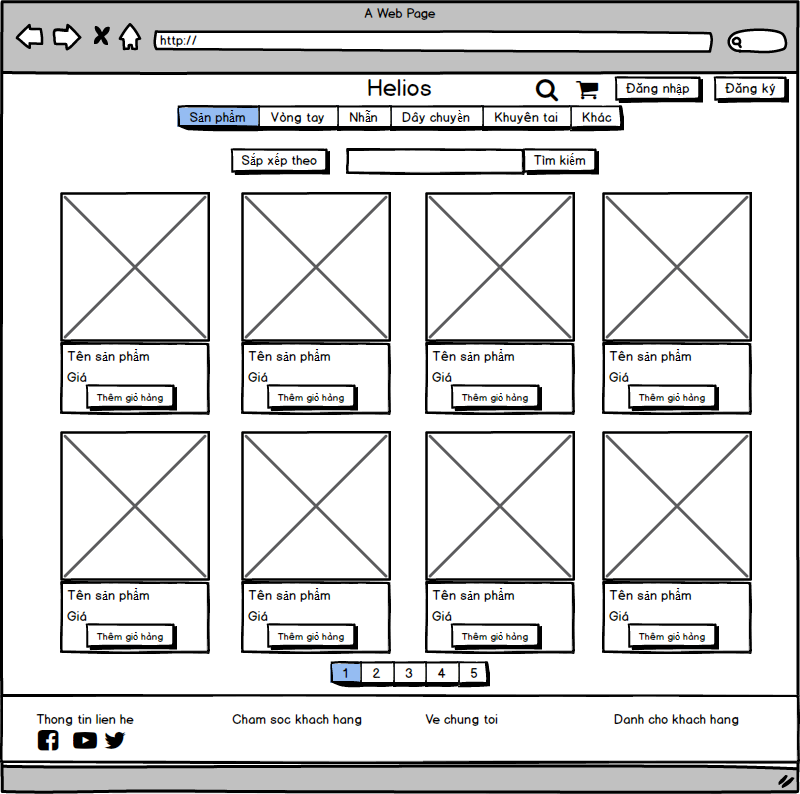
Hình ảnh ‑: Thiết kế màn hình đăng nhập

#### Màn hình đăng ký



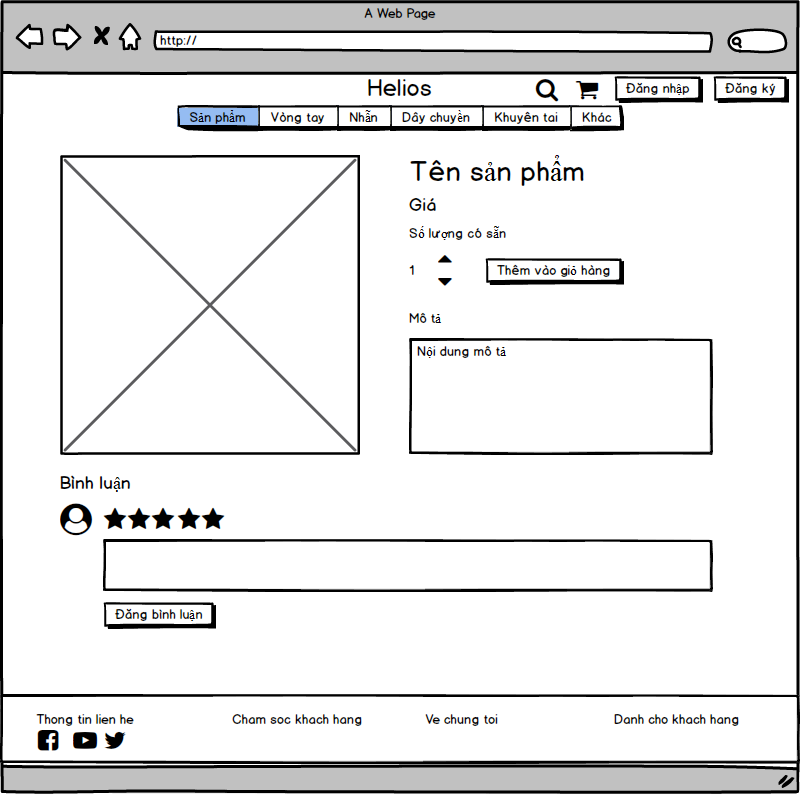
Hình ảnh ‑: Thiết kế màn hình đăng ký

#### Màn hình sản phẩm



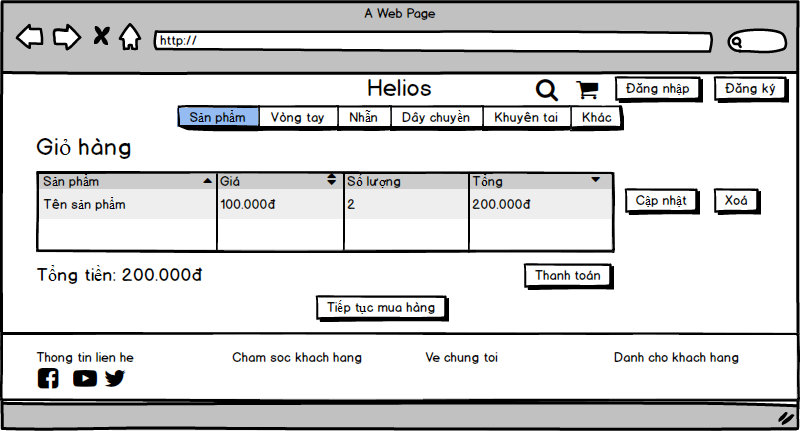
Hình ảnh ‑: : Thiết kế màn hình sản phẩm

#### Màn hình thông tin chi tiết sản phẩm



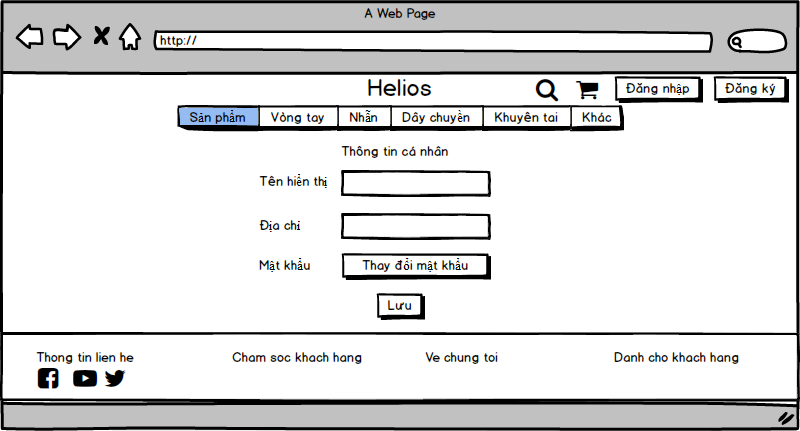
Hình ảnh ‑: : Thiết kế màn hình thông tin chi tiết sản phẩm

#### Màn hình quản lý giỏ hàng



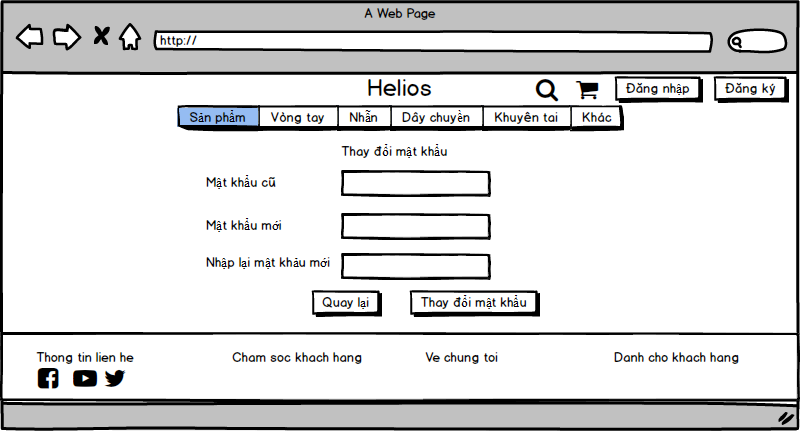
Hình ảnh ‑: Thiết kế màn hình quản lý giỏ hàng

#### Màn hình quản lý thông tin cá nhân



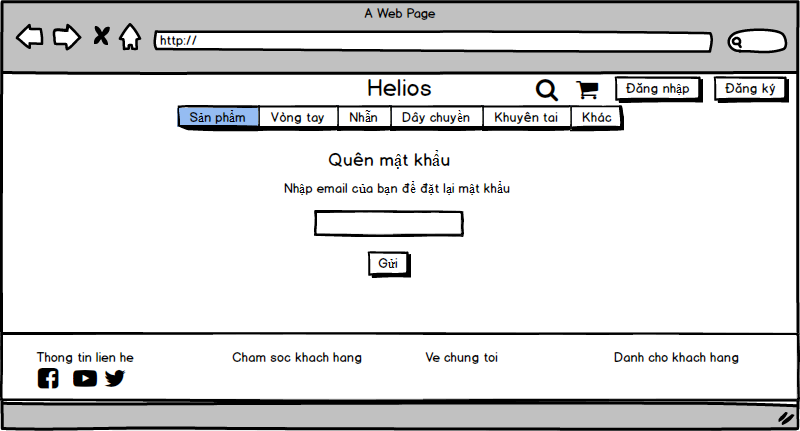
Hình ảnh ‑: Thiết kế màn hình quản lý thông tin cá nhân

#### Màn hình đổi mật khẩu



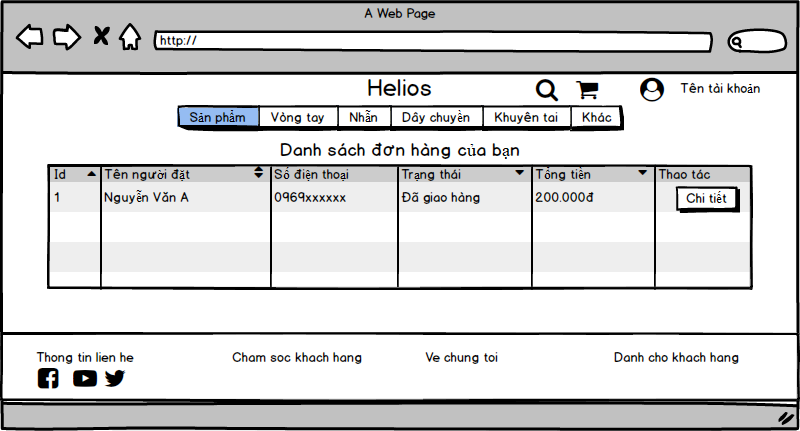
Hình ảnh ‑: Thiết kế màn hình đổi mật khẩu

#### Màn hình quên mật khẩu



Hình ảnh ‑: Thiết kế màn hình quên mật khẩu

#### Màn hình theo dõi đơn hàng



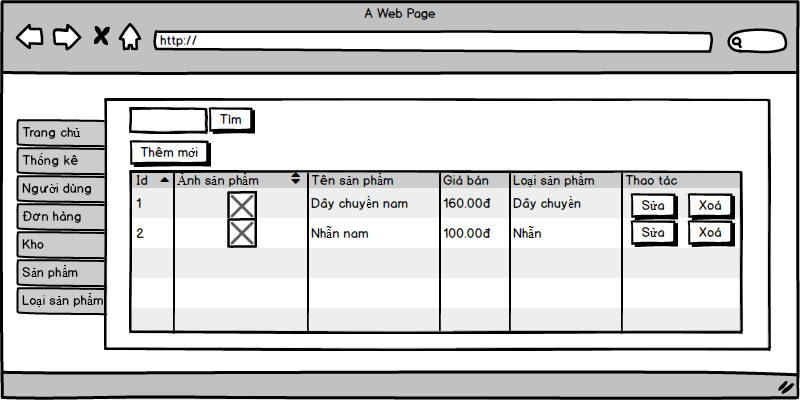
Hình ảnh ‑: Thiết kế màn hình theo dõi đơn hàng

#### Màn hình quản lý danh mục sản phẩm



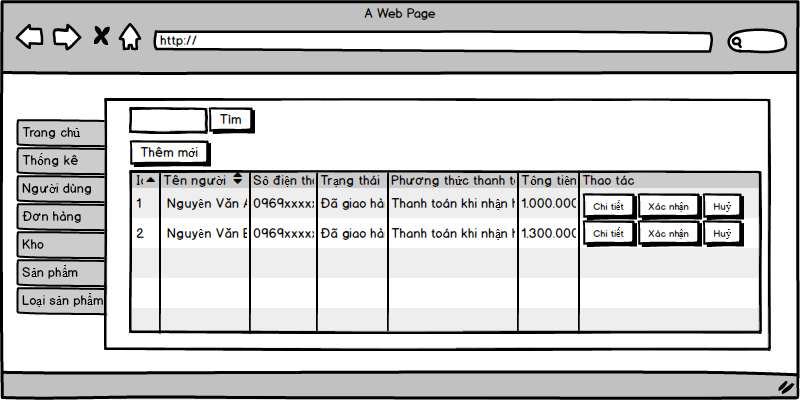
Hình ảnh ‑: Thiết kế màn hình quản lý danh mục sản phẩm

#### Màn hình quản lý sản phẩm



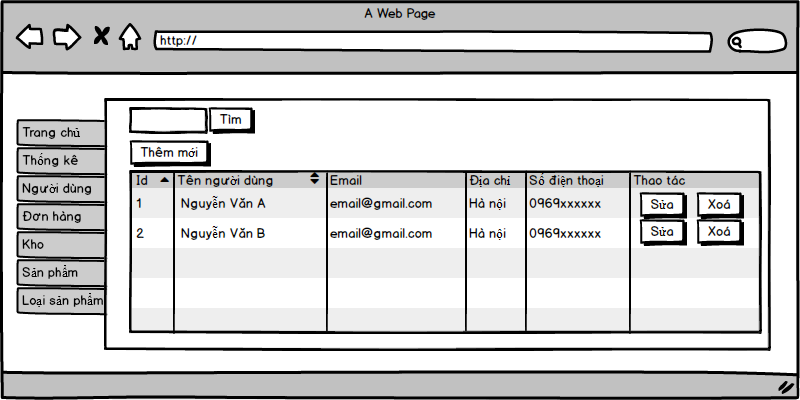
Hình ảnh ‑: Thiết kế màn hình quản lý sản phẩm

#### Màn hình quản lý đơn hàng



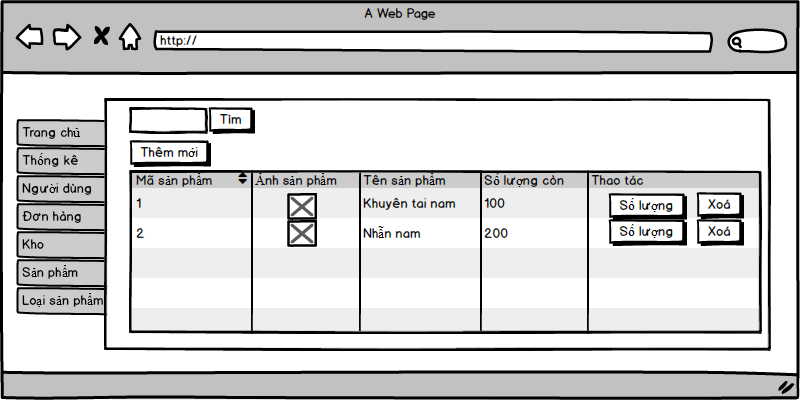
Hình ảnh ‑: Thiết kế màn hình quản lý đơn hàng

#### Màn hình quản lý khách hàng



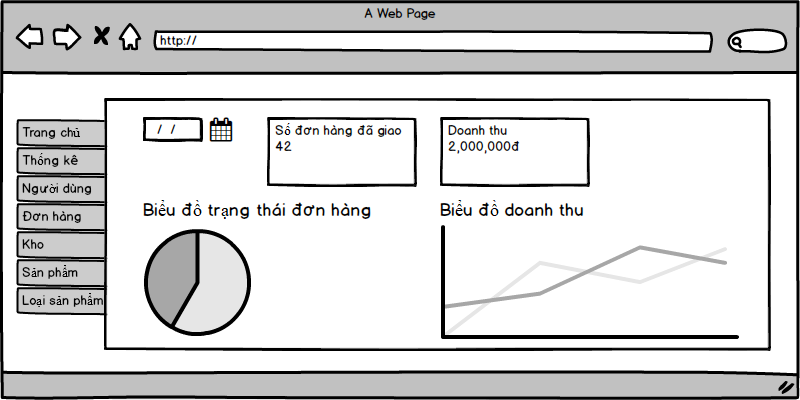
Hình ảnh ‑: Thiết kế màn hình quản lý khách hàng

#### Màn hình quản lý kho



Hình ảnh ‑: Thiết kế màn hình quản lý kho

#### Màn hình thống kê



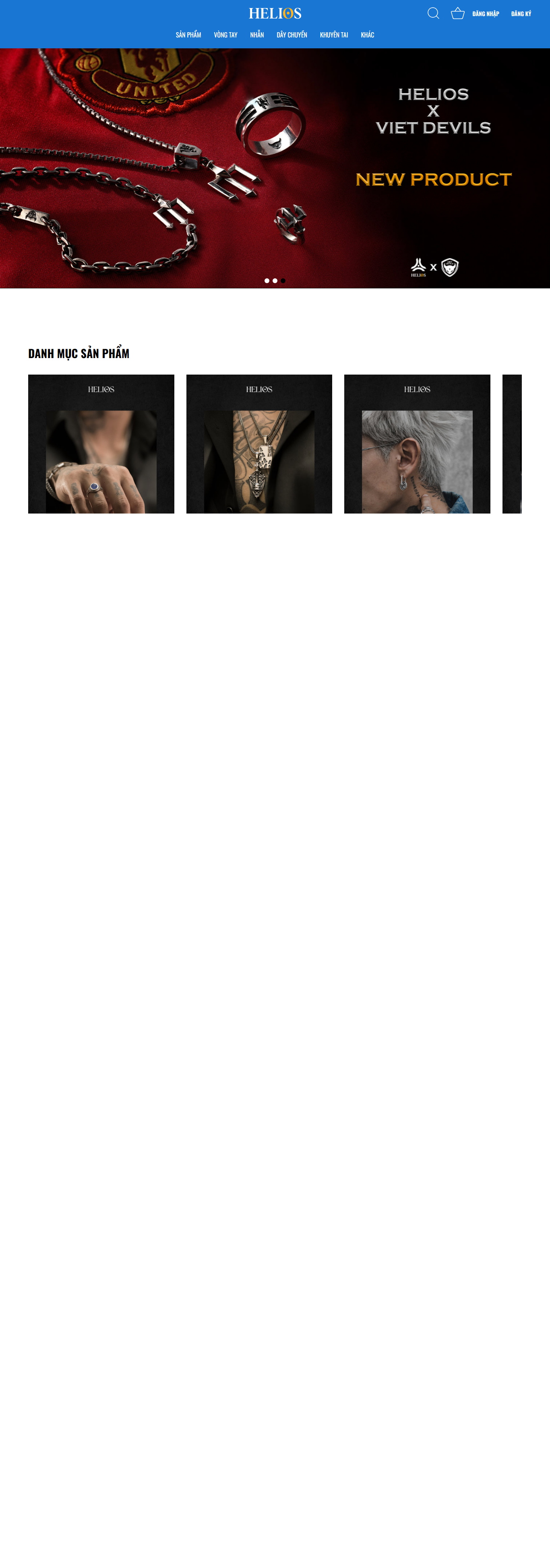
Hình ảnh ‑: Thiết kế màn hình thống kê

# CÀI ĐẶT CHƯƠNG TRÌNH VÀ KẾT QUẢ ĐẠT ĐƯỢC

## Giao diện của hệ thống

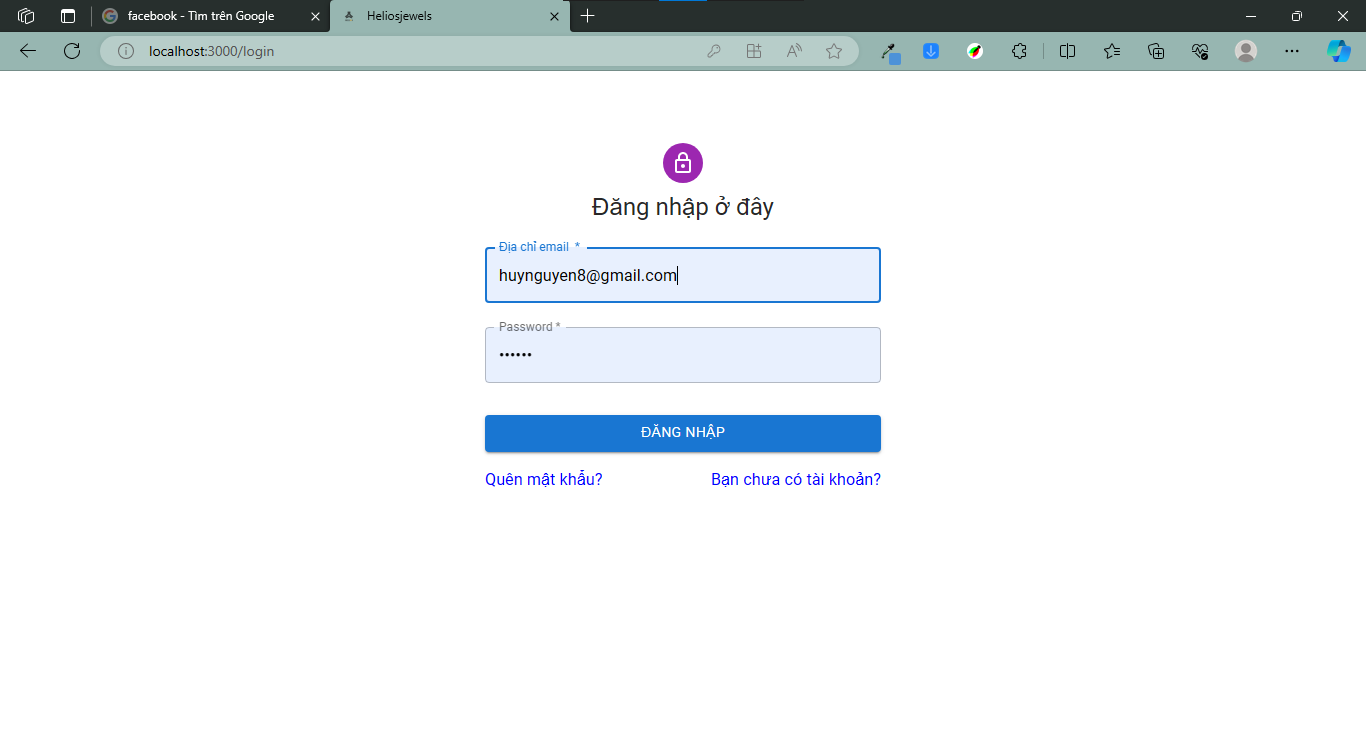
### Giao diện phía khách hàng

#### Màn hình trang chủ



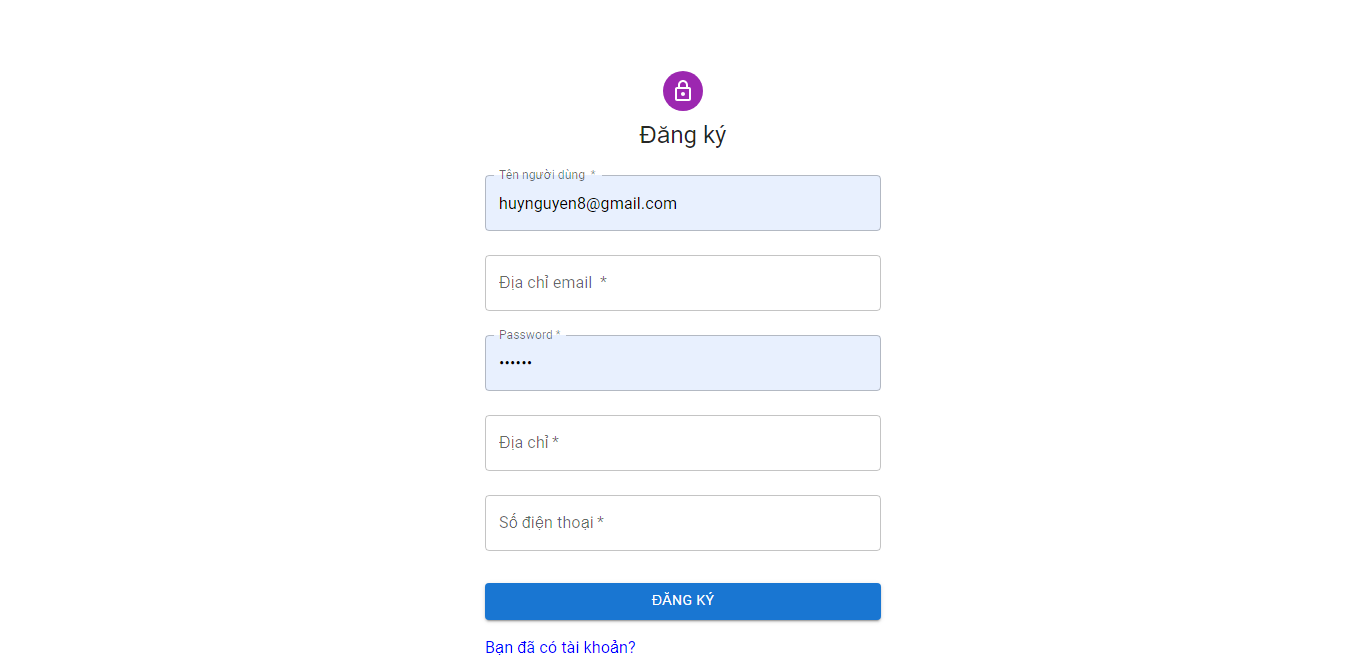
Hình ảnh ‑: Màn hình trang chủ

#### Màn hình đăng nhập



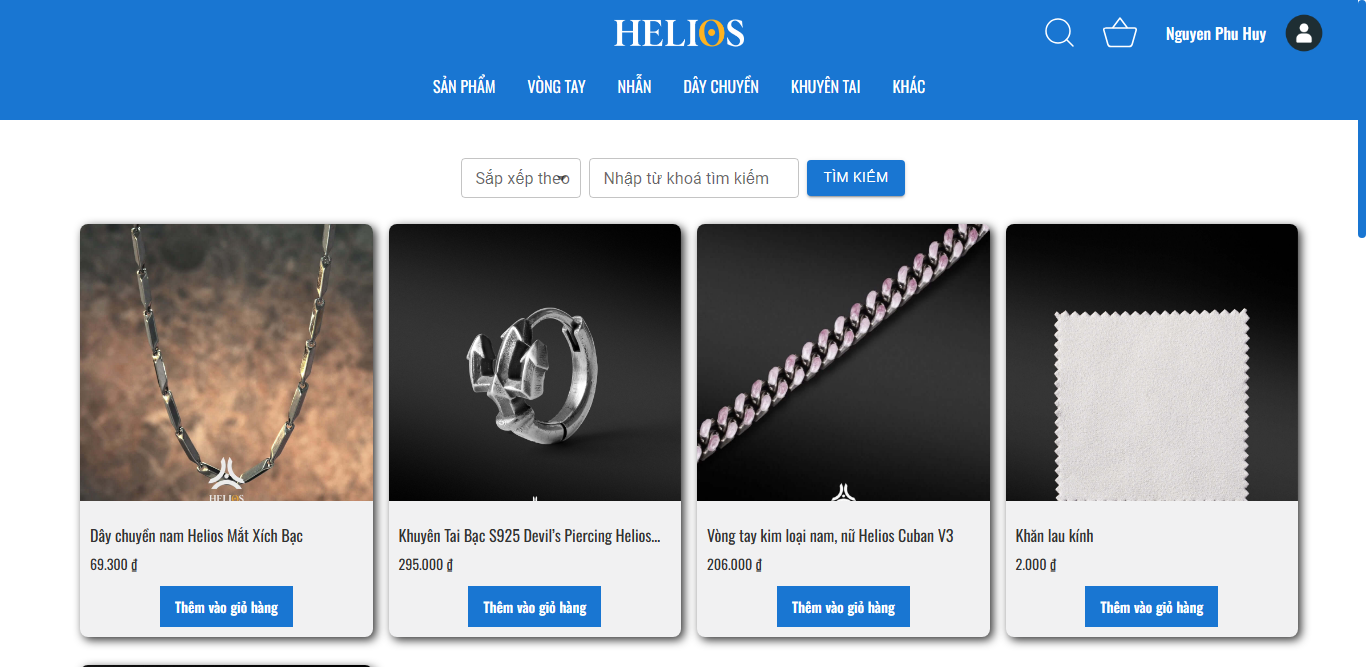
Hình ảnh ‑: Màn hình đăng nhập

#### Màn hình đăng ký



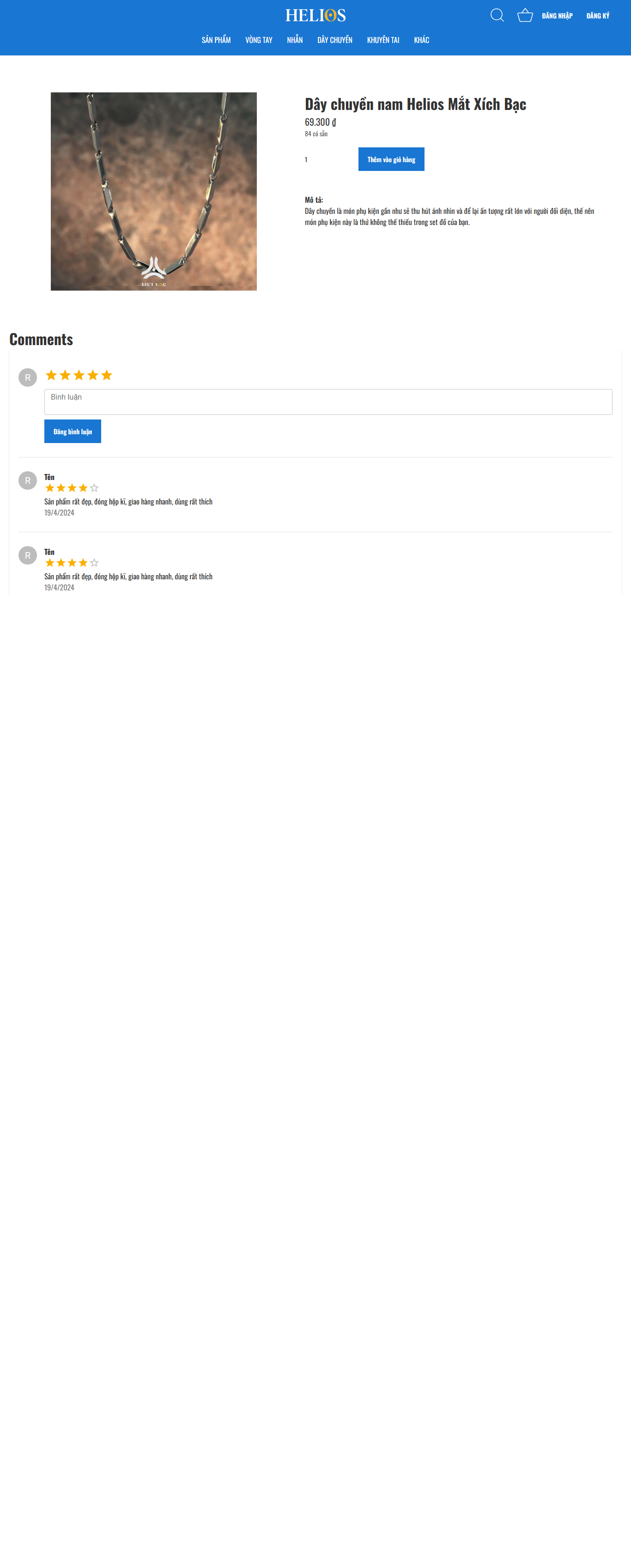
Hình ảnh ‑: Màn hình đăng ký

#### Màn hình sản phẩm



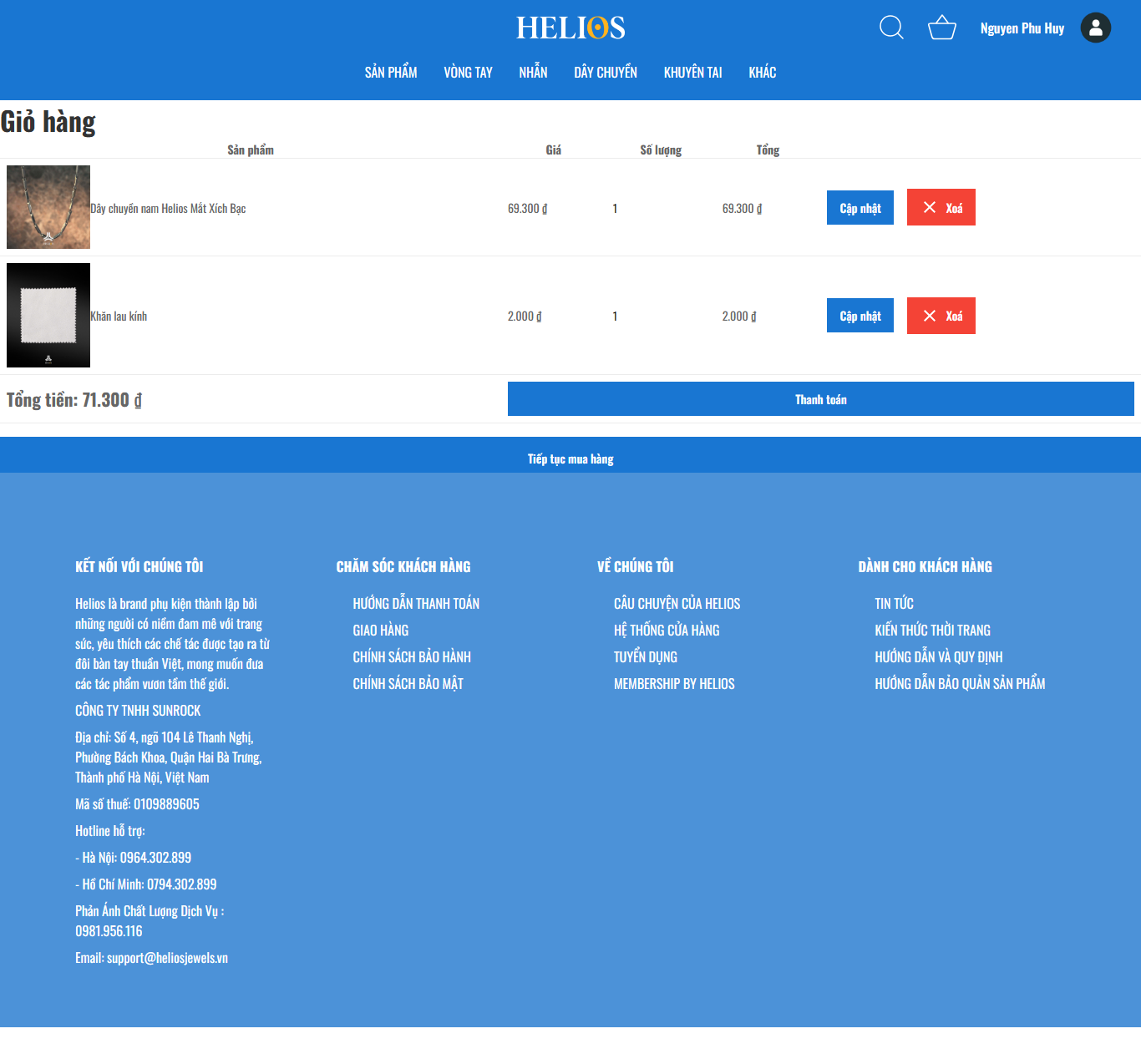
Hình ảnh ‑: Màn hình sản phẩm

#### Màn hình thông tin chi tiết sản phẩm



Hình ảnh ‑: Màn hình thông tin chi tiết

#### Màn hình quản lý giỏ hàng



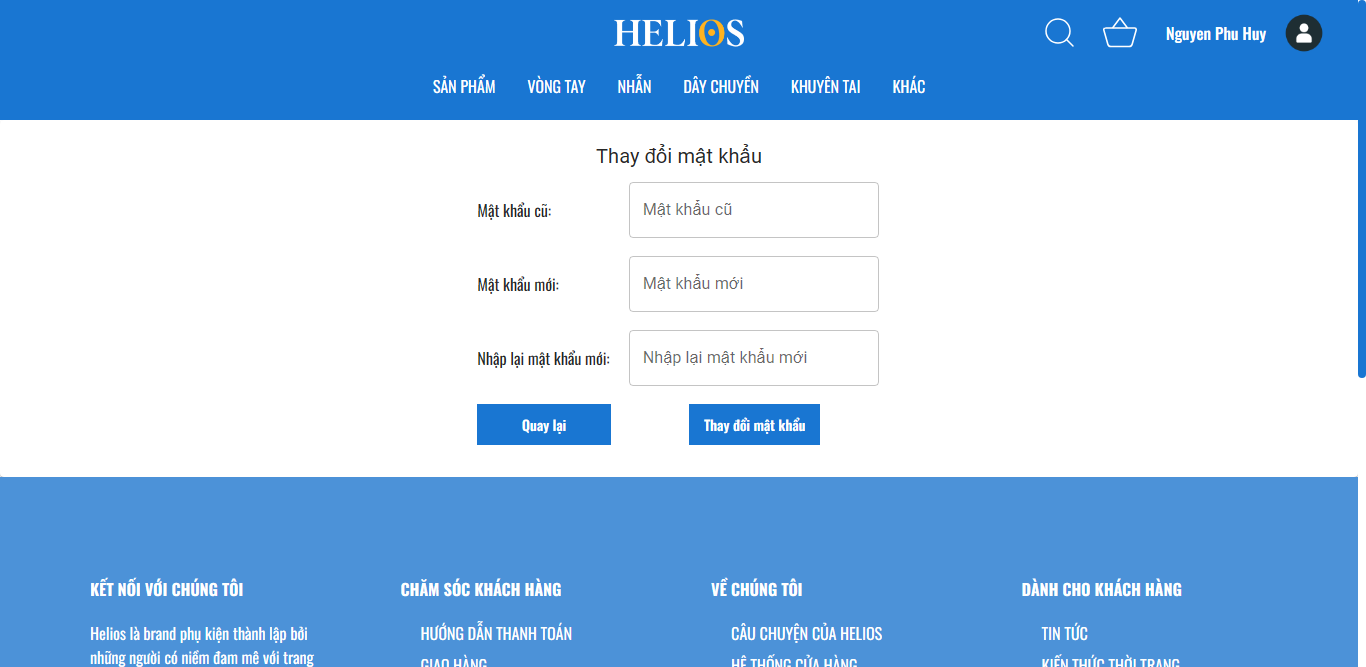
Hình ảnh ‑: Màn hình quản lý giỏ hàng

#### Màn hình quản lý thông tin cá nhân



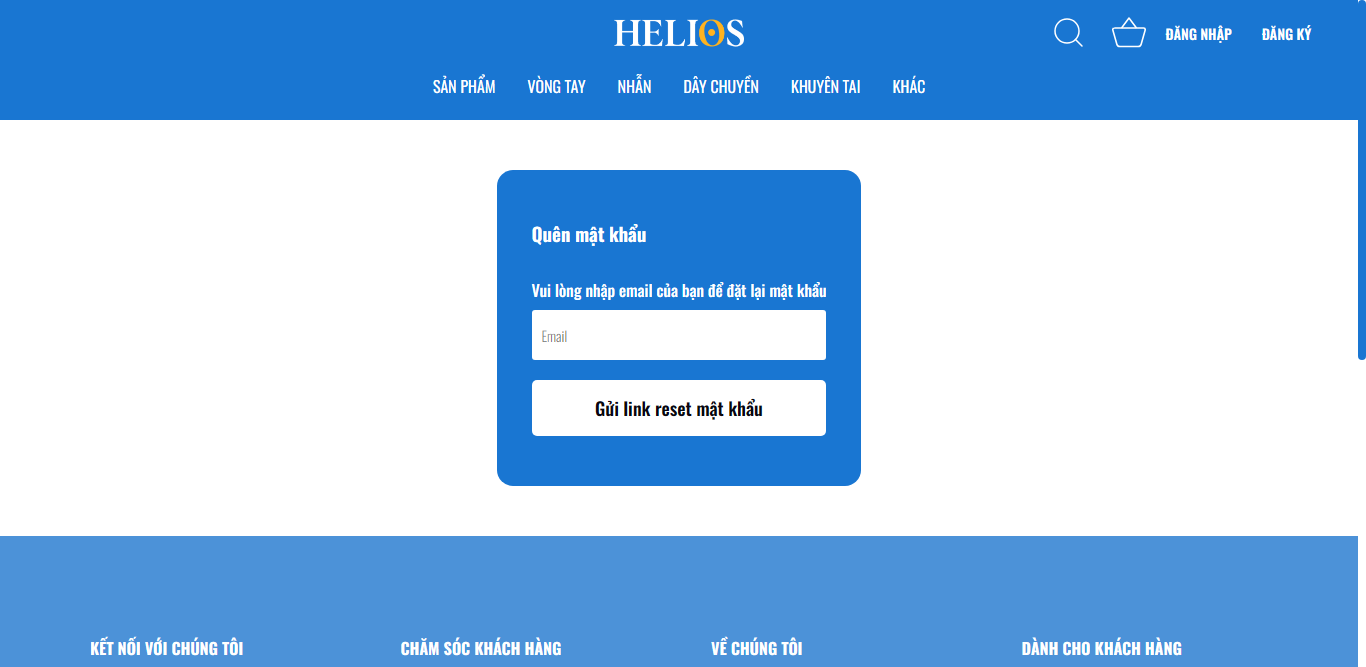
Hình ảnh ‑: Màn hình quản lý thông tin cá nhân

#### Màn hình đổi mật khẩu



Hình ảnh ‑: Màn hình đổi mật khẩu

#### Màn hình quên mật khẩu



Hình ảnh ‑: Màn hình quên mật khẩu

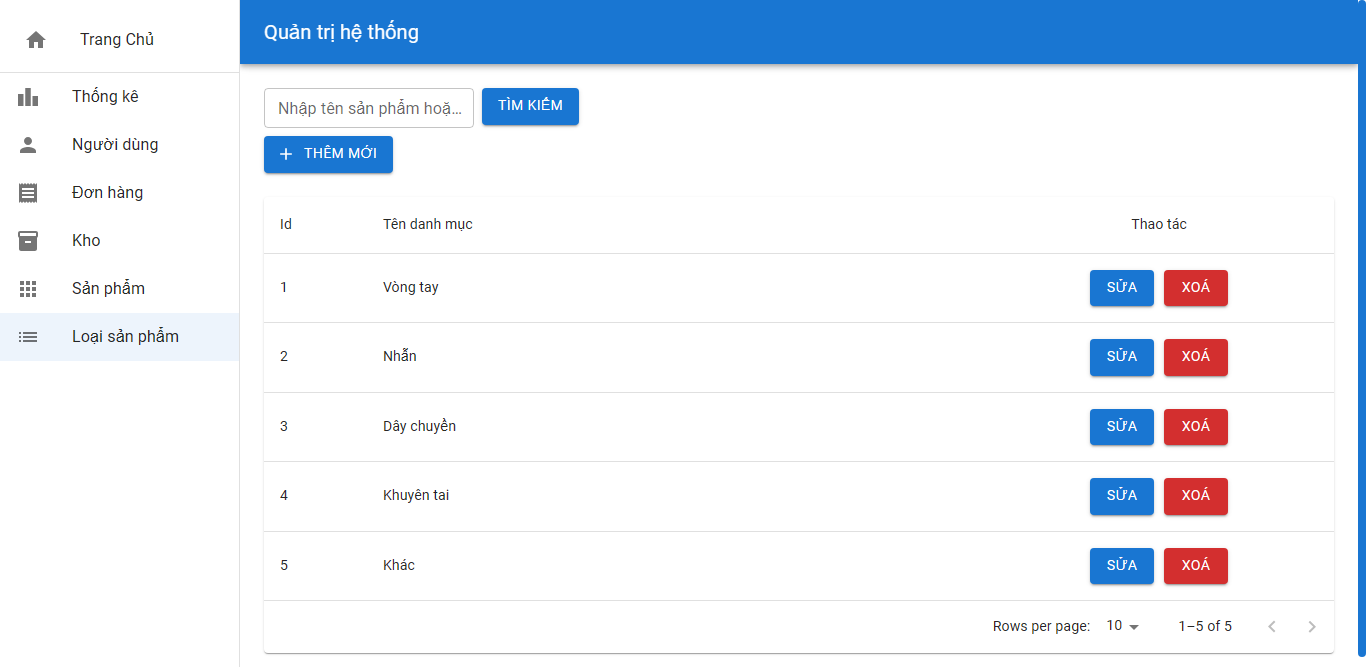
#### Màn hình theo dõi đơn hàng



Hình ảnh ‑: Màn hình theo dõi đơn hàng

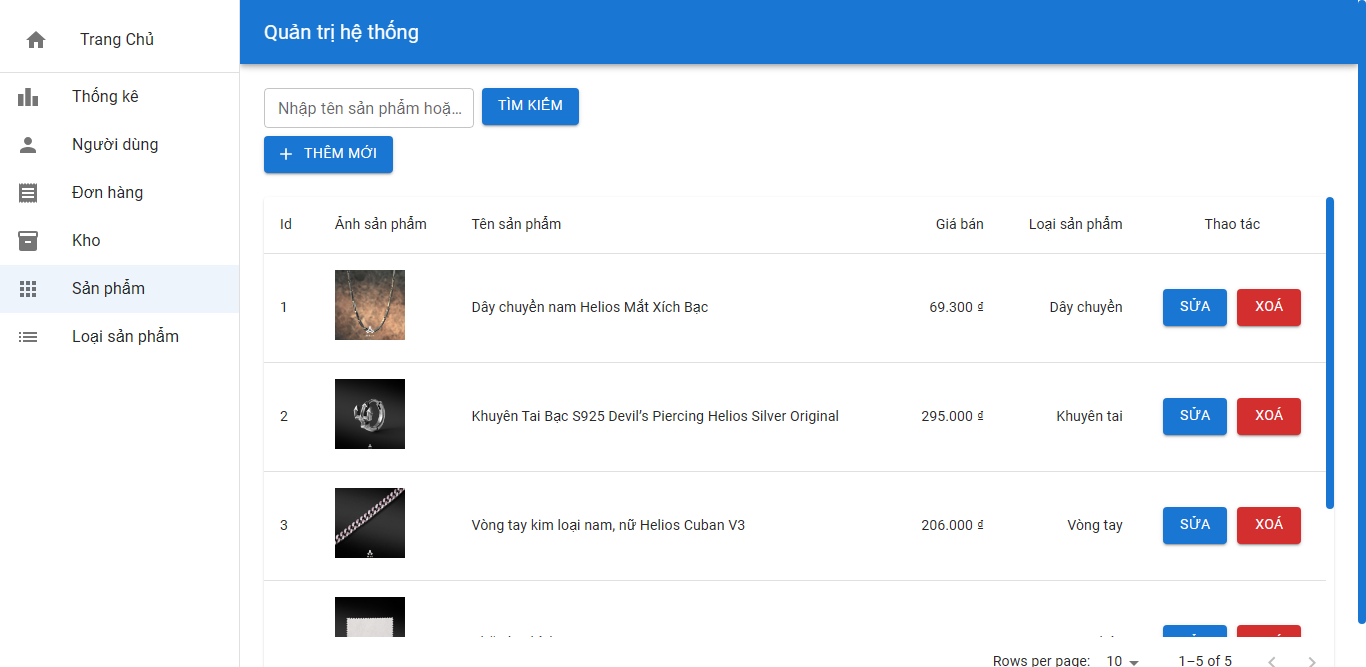
### Giao diện phía quản trị

#### Màn hình quản lý danh mục sản phẩm



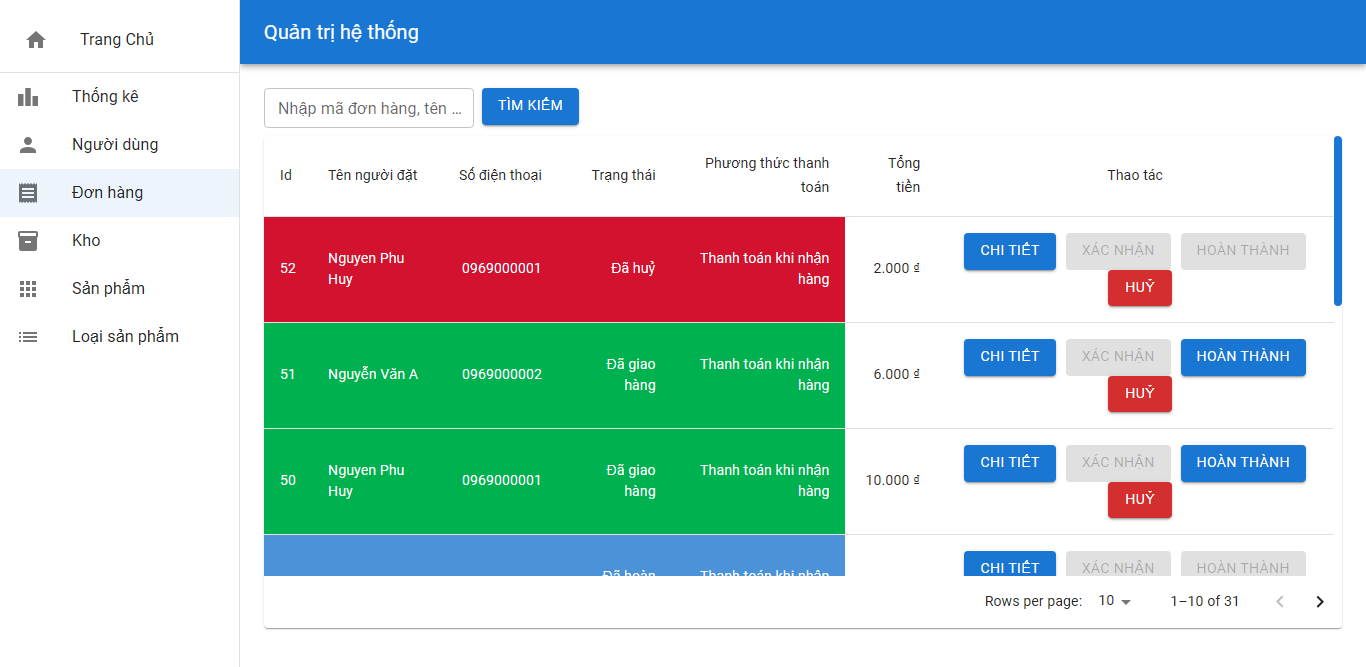
Hình ảnh ‑: Màn hình quản lý danh mục sản phẩm

#### Màn hình quản lý sản phẩm



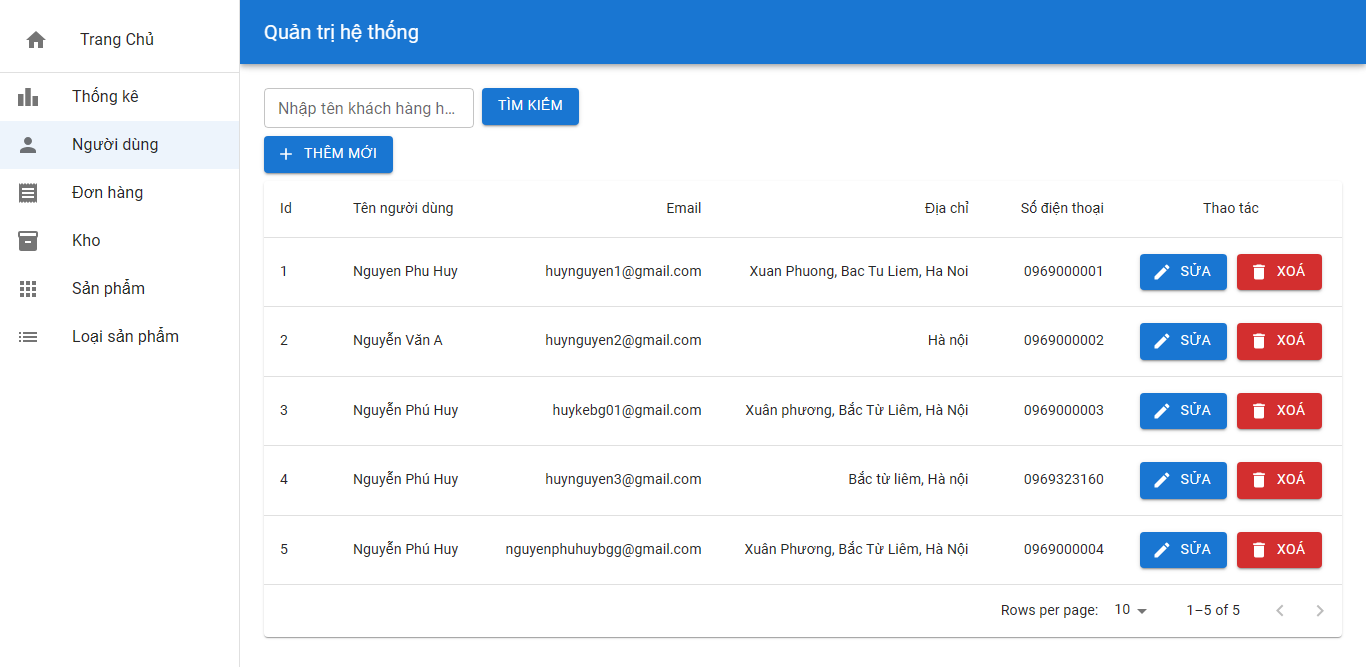
Hình ảnh ‑: Màn hình quản lý sản phẩm

#### Màn hình quản lý đơn hàng



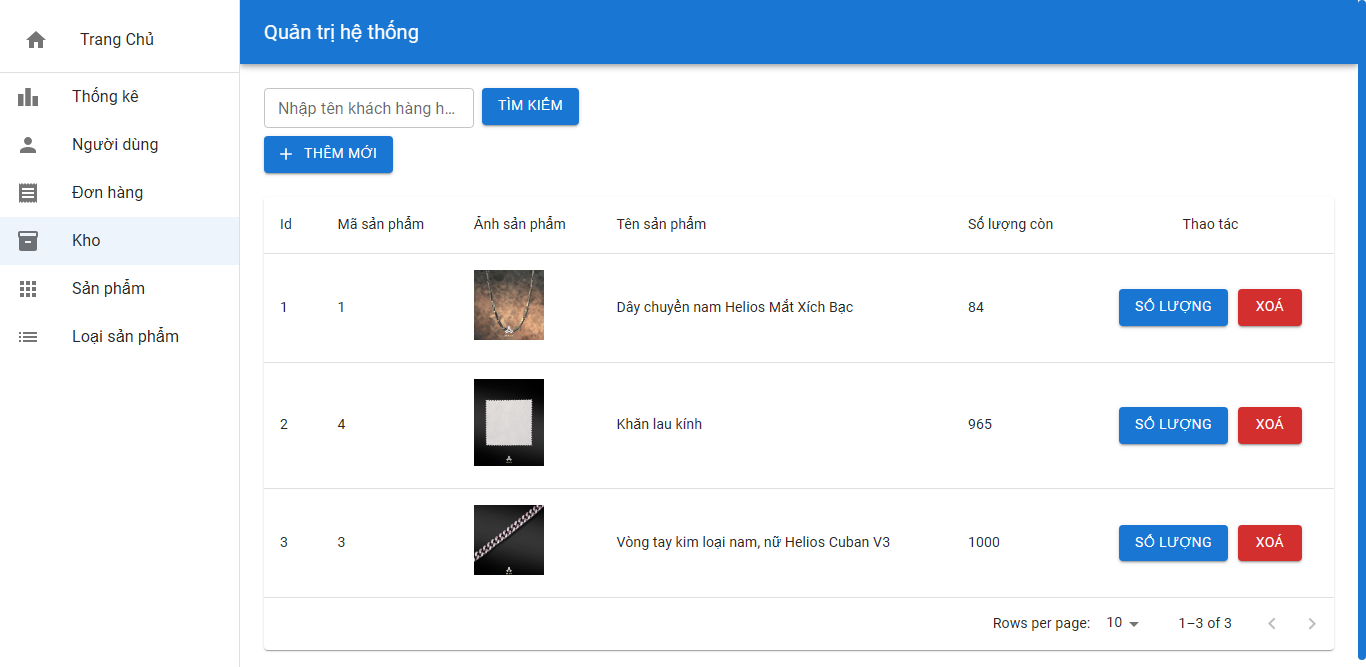
Hình ảnh ‑: Màn hình quản lý đơn hàng

#### Màn hình quản lý khách hàng



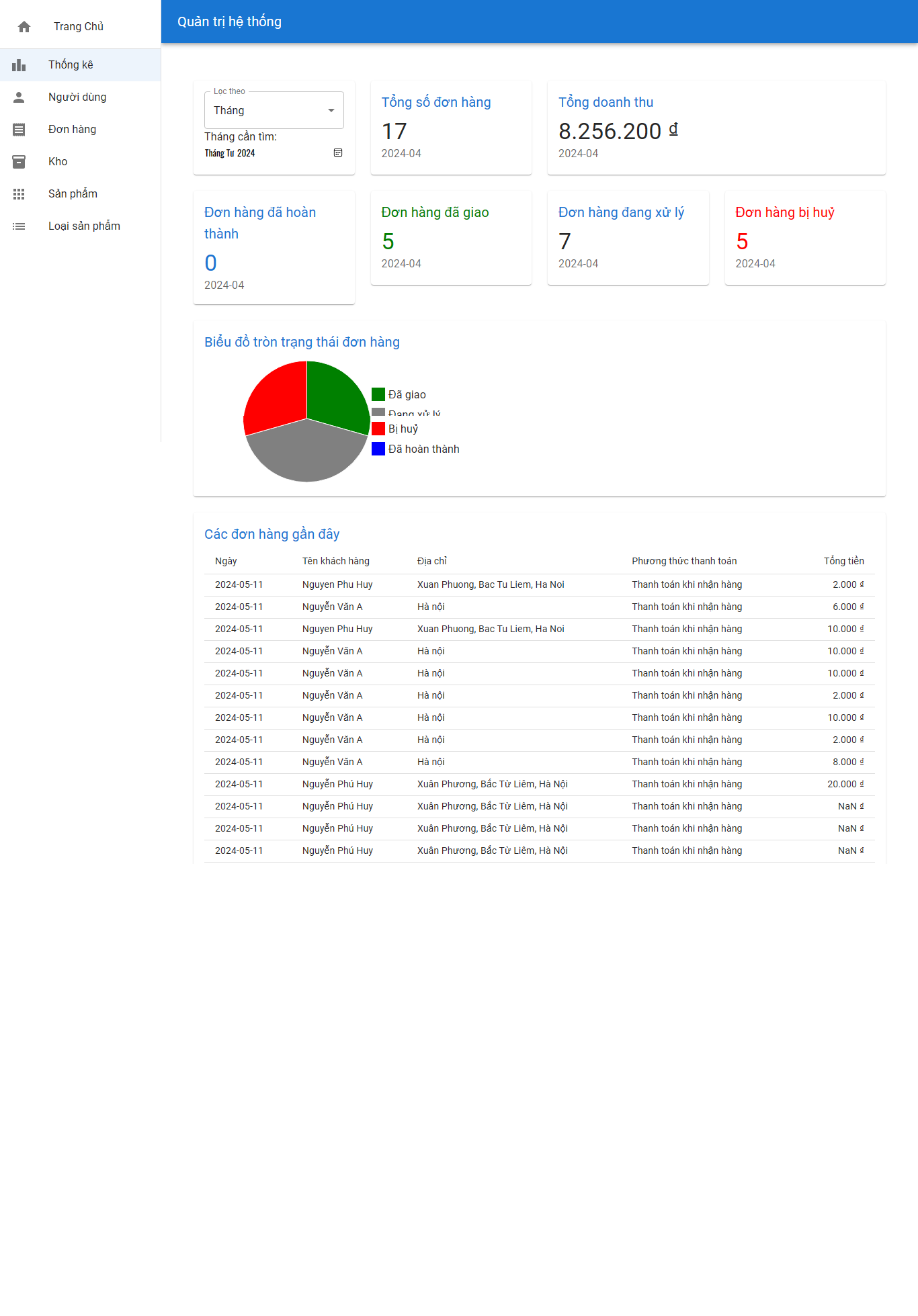
Hình ảnh ‑: Màn hình quản lý khách hàng

#### Màn hình quản lý kho



Hình ảnh ‑: Màn hình quản lý kho

#### Màn hình thống kê



Hình ảnh ‑: Màn hình thống kê

## Kiểm thử chức năng hệ thống

### Kiểm thử chức năng phía khách hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Case** | **Đầu vào** | **Đầu ra mong muốn** | **Kết quả** |
|  | Kiểm tra chức năng đăng nhập | Nhập emai và mật khẩu hợp lệ | Đăng nhập thành công, chuyển sang trang Tài khoản người dùng. | Pass |
| Nhập emai  và mật khẩu không hợp lệ | Đăng nhập không thành công | Pass |
|  | Kiểm tra chức năng đăng ký tài khoản | Nhập thông tin đăng ký hợp lệ | Đăng ký thành công chuyển sang trang đăng nhập hệ thống | Pass |
| Nhập thông tin đăng ký không hợp lệ | Hệ thống thông báo lỗi | Pass |
|  | Kiểm tra chức năng đăng xuất | Đăng xuất tài khoản | Đăng xuất khỏi tài khoản, chuyển sang trang chủ của khách hàng. | Pass |
|  | Kiểm tra chức năng tìm kiếm sản phẩm | Nhập từ khóa sản phẩm cần tìm có tồn tại | Hiển thị tất cả các sản phẩm chứa từ khóa vừa nhập. | Pass |
| Nhập từ khóa  sản phẩm cần tìm không tồn tại | Không tìm thấy sản phẩm nào. | Pass |
|  | Kiểm tra chức năng xem chi tiết  sản phẩm | Đã đăng nhập vào hệ thống | Xem được thông tin chi tiết sản phẩm | Pass |
| Chưa đăng nhập  vào hệ thống | Xem được thông tin chi tiết sản phẩm | Pass |
|  | Kiểm tra chức năng thêm sản phẩm vào giỏ hàng | Đã đăng nhập vào hệ thống | Hệ thống thông báo thêm sản phẩm vào giỏ hàng thành công. | Pass |
| Chưa đăng nhập vào hệ thống | Hệ thống thông báo cần đăng nhập | Pass |
|  | Kiểm tra chức năng cập nhật số lượng sản phẩm trong giỏ hàng | Đã đăng nhập vào hệ thống | Hệ thống thông báo sửa số lượng trong giỏ hàng thành công. | Pass |
| Chưa đăng nhập vào hệ thống | Hệ thống thông báo cần đăng nhập | Pass |
|  | Kiểm tra chức năng xóa sản phẩm trong giỏ hàng | Đã đăng nhập vào hệ thống | Cập nhật các sản phẩm trong giỏ hàng thành công. | Pass |
| Chưa đăng nhập vào hệ thống | Hệ thống thông báo cần đăng nhập | Pass |
|  | Kiểm tra chức năng đặt hàng | Đã đăng nhập vào hệ thống | Cho phép khách hàng tiếp tục đặt hàng và thanh toán. | Pass |
| Chưa đăng nhập vào hệ thống | Hệ thống thông báo cần đăng nhập | Pass |
|  | Kiểm tra chức năng hủy đơn đặt  Hàng | Đã đăng nhập vào hệ thống | Hủy thành công đơn đặt hàng chưa được duyệt. | Pass |
| Chưa đăng nhập vào hệ thống | Không xác định | Pass |
|  | Kiểm tra chức năng xem thông  tin tài khoản | Đã đăng nhập vào hệ thống | Xem được thông tin tài khoản. | Pass |
| Chưa đăng nhập vào hệ thống | Yêu cầu đăng nhập hệ thống. | Pass |
|  | Kiểm tra chức năng sửa thông  tin tài khoản | Đã đăng nhập vào hệ thống | Hệ thống cập nhật thông tin tài khoản | Pass |
| Chưa đăng nhập vào hệ thống | Yêu cầu đăng nhập hệ thống. | Pass |
|  | Kiểm tra chức năng quên mật khẩu | Chưa đăng nhập vào hệ thống | Hệ thống gửi link đặt mật khẩu mới đến email | Pass |
|  | Kiểm tra chức năng  đổi mật khẩu | Đã đăng nhập vào hệ thống và nhập đúng mật khẩu cũ | Hệ thống cập nhập mật khẩu mới | Pass |
| Đã đăng nhập vào hệ thống và nhập đúng mật khẩu chưa nhập đúng mật khẩu cũ | Hệ thống thông báo lỗi | Pass |
|  | Kiểm tra chức năng theo dõi đơn hàng | Đã đăng nhập vào hệ thống | Hệ thống hiển thị các đơn hàng đã đặt | Pass |

Bảng ‑: Bảng kiểm thử chức năng phía khách hàng

### Kiểm thử chức năng phía quản trị

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Case** | **Đầu vào** | **Đầu ra mong muốn** | **Kết quả** |
|  | Kiểm tra chức năng thêm mới danh mục loại sản phẩm | Nhập dữ liệu hợp lệ | Hệ thống thêm mới danh mục loại sản phẩm | Pass |
| Nhập dữ liệu không hợp lệ | Hệ thống thông báo lỗi | Pass |
|  | Kiểm tra chức năng sửa danh mục loại sản phẩm | Nhập dữ liệu hợp lệ | Hệ thông cập nhật danh mục loại sản phẩm | Pass |
| Nhập dữ liệu không hợp lệ | Hệ thống thông báo lỗi | Pass |
|  | Kiểm tra chức năng sửa danh mục loại sản phẩm | Đồng ý xoá | Hệ thống xoá danh mục loại sản phẩm | Pass |
| Huỷ xoá | Hệ thống không xoá danh mục loại sản phẩm | Pass |
|  | Kiểm tra chức năng tìm kiếm danh mục loại sản phẩm | Nhập vào dữ liệu có tồn tại | Hệ thống hiển thị danh sách có từ khoá cần tìm | Pass |
| Nhập vào dữ liệu không tồn tại | Hệ thống không hiển thị bản ghi nào | Pass |
|  | Kiểm tra chức năng thêm mới sản phẩm | Nhập dữ liệu hợp lệ | Hệ thống thêm mới sản phẩm | Pass |
| Nhập dữ liệu không hợp lệ | Hệ thống thông báo lỗi | Pass |
|  | Kiểm tra chức năng sửa sản phẩm | Nhập dữ liệu hợp lệ | Hệ thống cập nhật sản phẩm | Pass |
| Nhập dữ liệu không hợp lệ | Hệ thống thông báo ra lỗi | Pass |
|  | Kiểm tra chức năng xoá sản phẩm | Đồng ý xoá | Hệ thống xoá sản phẩm | Pass |
| Huỷ xoá | Hệ thống không xoá sản phẩm | Pass |
|  | Kiểm tra chức năng tìm kiếm sản phẩm | Nhập vào dữ liệu có tồn tại | Hệ thống hiển thị danh sách có từ khoá cần tìm | Pass |
| Nhập vào dữ liệu không tồn tại | Hệ thống không hiển thị bản ghi nào | Pass |
|  | Kiểm tra chức năng xem danh sách đơn hàng | Chọn xem thông tin đơn hàng | Hệ thống hiển thị thông tin các đơn hàng | Pas |
|  | Kiểm tra chức năng Cập nhật trạng thái các đơn hàng | Chọn: Xác nhận, huỷ đơn, hoàn thành | Hệ thống cập nhật trạng thái các đơn hàng | Pass |
|  | Kiểm tra chức năng xem danh sách khách hàng | Chọn xem thông tin khách hàng | Hệ thống hiển thị thông tin các khách hàng | Pass |
|  | Kiểm tra chức năng sửa thông tin khách hàng | Nhập dữ liệu hợp lệ | Hệ thống cập nhật thông tin khách hàng | Pass |
| Nhập dữ liệu không hợp lệ | Hệ thống thông báo lỗi | Pass |
|  | Kiểm tra chức năng xoá thông tin khách hàng | Đồng ý xoá | Hệ thống xoá thông tin khách hàng | Pass |
| Huỷ xoá | Hệ thống không xoá thông tin khách hàng | Pass |
|  | Kiểm tra chức năng tìm kiếm khách hàng | Nhập vào dữ liệu có tồn tại | Hệ thống hiển thị danh sách có từ khoá cần tìm | Pass |
| Nhập vào dữ liệu không tồn tại | Hệ thống không hiển thị bản ghi nào | Pass |
|  | Kiểm tra chức năng xem kho hàng | Chọn xem thông tin kho hàng | Hệ thống hiển thị thông tin các kho hàng | Pass |
|  | Kiểm tra chức năng tăng, giảm số lượng trong kho hàng | Chọn tăng, giảm số lượng kho hàng | Hệ thống cập nhật số lượng kho hàng | Pass |
|  | Kiểm tra chức năng xoá kho hàng | Đồng ý xoá | Hệ thống xoá thông tin kho hàng | Pass |
| Huỷ xoá | Hệ thông không xoá thông tin kho hàng | Pass |
|  | Kiểm tra chức năng tìm kiếm kho hàng | Nhập vào dữ liệu có tồn tại | Hệ thống hiển thị danh sách có từ khoá cần tìm | Pass |
| Nhập vào dữ liệu không tồn tại | Hệ thống không hiển thị bản ghi nào | Pass |
|  | Kiểm tra chức năng thông kê doanh số, đơn hàng | Chọn thời gian cần thống kê | Hệ thống hiển thị doanh số, số lượng đơn hàng đã thống kê | Pass |

Bảng ‑: Bảng kiểm thử chức năng phía quản trị

# KẾT LUẬN

* **Kết quả thực nghiệm đã đạt được:**
* Hoàn thành khảo sát và phân tích thiết kế hệ thống, thiết kế cơ sở dữ liệu, thiết kế giao diện.
* Hệ thống hiện tại đã xây dựng được cơ bản các chức năng theo như yêu cầu ban đầu: tìm kiếm, xem chi tiết sản phẩm, đăng nhập, đăng ký, đặt hàng và thanh toán, quản lý sản phẩm, quản lý khách hàng, quản lý đơn hàng, v.v.
* Hệ thống được thiết kế tương đối rõ ràng và chi tiết các ca sử dụng và các chức năng được giới thiệu, mô tả chi tiết và dễ hình dung. Mặc dù việc áp dụng các giải pháp công nghệ cũng như các thuật toán vào xây dựng hệ thống vẫn còn những thiếu sót nhưng em sẽ cố gắng hoàn thành và phát triển thêm trong thời gian tới.
* **Hạn chế:**
* Chưa áp dụng được tính phí vận chuyển theo vị trí giao nhận hàng.
* **Hướng phát triển:**
* Tiếp tục tìm hiểu để hoàn thiện các chức năng còn thiếu sót như đã nêu ở mục trên.
* Kết hợp truyền thông online và website để tăng tương tác bán hàng dựa vào các trang mạng xã hội như Facebook, Twitter, Youtube, v.v.

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | N. T. Phú, Giáo trình thiết kế web, NXB thống kê. |
| [2] | Lập trình React thật đơn giản. |
| [3] | Hoàng Quang Huy, Phùng Đức Hoà, Trịnh Bá Quý, Nhập môn công nghệ phần mềm, NXB Đại học công nghiệp Hà Nội. |
| [4] | Nguyễn Thị Thanh Huyền, Ngô Thị Bích Thuý, Phạm Thị Kim Phượng, Giáo trình phân tích thiết kế hệ thống, NXB Giáo dục VN. |
| [5] | "w3schools," [Online]. Available: https://www.w3schools.com/nodejs/. |