

**THỰC TẬP CHUYÊN ĐỀ VÀ ĐỒ ÁN CHUYÊN MÔN**

***ĐỀ TÀI***

**Hệ thống website xe ô tô**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | **Th.S Trịnh Văn Chung** |
| **Sinh viên thực hiện:** | **Nguyễn Huy Thông** |
| **Mã sinh viên:** | **231090102** |
| **Lớp:** | **K23CNT2** |
| **Khóa:** | **23** |

**@FITNTU-PJ** 12/2024

**MỤC LỤC**

[**DANH MỤC CÁC TỪ VIẾT TẮT VÀ THUẬT NGỮ** 5](#_heading=h.kgcv8k)

[**DANH MỤC BẢNG BIỂU** 8](#_heading=h.34g0dwd)

[**DANH MỤC HÌNH ẢNH** 10](#_heading=h.1jlao46)

[**CHƯƠNG 1: NGHIÊN CỨU CƠ SỞ LÝ THUYẾT** 11](#_heading=h.43ky6rz)

[1.1.](#_heading=h.2iq8gzs) Tổng quan về lập trình thiết kế giao diện website 11

[1.1.1.](#_heading=h.xvir7l) Giới thiệu về HTML 11

[1.1.2.](#_heading=h.3hv69ve) Giới thiệu về CSS3 11

[1.1.3.](#_heading=h.1x0gk37) Tổng quan về ngôn ngữ lập trình Javascript 11

[1.1.4.](#_heading=h.4h042r0) Thư viện Jquery 11

[1.1.5.](#_heading=h.2w5ecyt) Framework Bootstrap 11

[1.2.](#_heading=h.1baon6m) Tổng quan về hệ quản trị cơ sở dữ liệu SQL Server 11

[1.2.1.](#_heading=h.3vac5uf) Giới thiệu về SQL Server 11

[1.2.2.](#_heading=h.2afmg28) Ưu điểm của SQL Server 11

[1.2.3.](#_heading=h.pkwqa1) Nhược điểm của SQL Server 11

[1.3.](#_heading=h.39kk8xu) Tổng quan về ngôn ngữ lập trình phía máy chủ 11

[1.3.1.](#_heading=h.1opuj5n) Giới thiệu về nền tảng công nghệ .NET 11

[1.3.2.](#_heading=h.48pi1tg) Giới thiệu về ngôn ngữ lập trình C# 11

[1.3.3.](#_heading=h.2nusc19) Tổng quan về Framework ASP.NET MVC 5 12

[**CHƯƠNG 2: TỔNG QUAN VỀ ĐỀ TÀI** 13](#_heading=h.1302m92)

[2.1.](#_heading=h.3mzq4wv) Đặt vấn đề 13

[2.2.](#_heading=h.2250f4o) Hệ thống hiện tại 13

[2.3.](#_heading=h.haapch) Hệ thống đề xuất 13

[2.4.](#_heading=h.319y80a) Giới hạn của hệ thống 13

[2.5.](#_heading=h.1gf8i83) Yêu cầu về phần cứng và phần mềm 13

[2.5.1.](#_heading=h.40ew0vw) Yêu cầu tối thiểu 13

[2.5.2.](#_heading=h.2fk6b3p) Yêu cầu đề nghị 13

[**CHƯƠNG 3:** 15](#_heading=h.upglbi)

[**PHÂN TÍCH YÊU CẦU KHÁCH HÀNG VÀ THIẾT KẾ HỆ THỐNG** 15](#_heading=h.3ep43zb)

[3.1.](#_heading=h.1tuee74) Người dùng của hệ thống 15

[3.2.](#_heading=h.4du1wux) Chức năng của hệ thống 15

[3.3.](#_heading=h.2szc72q) Systeme Designs (Thiết kế hệ thống) 18

[3.3.1.](#_heading=h.184mhaj) Entity Relationship Diagram (Mô hình quan hệ thực thể) 18

[3.4.](#_heading=h.3s49zyc) Database Design (Thiết kế cơ sở dữ liệu) 18

[3.5.](#_heading=h.279ka65) Table RelationShip Diagram (Mô hình quan hệ bảng) 19

[3.6.](#_heading=h.meukdy) SiteMap (Sơ đồ cấu trúc site) 19

[3.6.1.](#_heading=h.36ei31r) Site map (Cấu trúc trang): Dành cho khách vãng lai 20

[3.6.2.](#_heading=h.1ljsd9k) Sitemap (Cấu trúc trang): Dành cho thành viên 20

[3.6.3.](#_heading=h.45jfvxd) Sitemap (Cấu trúc trang): Dành cho quản trị viên 20

[3.7.](#_heading=h.2koq656) Algorithms (Giải thuật) 21

[3.7.1.](#_heading=h.zu0gcz) Đăng ký 21

[3.7.2.](#_heading=h.3jtnz0s) Đăng nhập 22

[3.7.3.](#_heading=h.1yyy98l) Đăng xuất 22

[3.7.4.](#_heading=h.4iylrwe) Đổi mật khẩu 23

[3.7.5.](#_heading=h.2y3w247) Thay đổi thông tin cá nhân 24

[3.7.6.](#_heading=h.1d96cc0) Thêm sản phẩm 25

[3.7.7.](#_heading=h.3x8tuzt) Xóa sản phẩm 26

[3.7.8.](#_heading=h.2ce457m) Đặt mua 27

[3.7.9.](#_heading=h.rjefff) Tìm kiếm 28

[**CHƯƠNG 4: XÂY DỰNG HỆ THỐNG** 29](#_heading=h.3bj1y38)

[4.1.](#_heading=h.1qoc8b1) Xây dựng phần mềm 29

[4.1.1.](#_heading=h.4anzqyu) Một số mã nguồn chương trình 29

[4.1.2.](#_heading=h.2pta16n) Một số giao diện người dùng và chức năng cụ thể 29

[1.](#_heading=h.14ykbeg) Giao diện trang chủ 29

[2.](#_heading=h.3oy7u29) Giao diện trang quản trị 29

[3.](#_heading=h.243i4a2) ….. 29

[**KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN** 30](#_heading=h.j8sehv)

[5.1.](#_heading=h.338fx5o) Kết luận 30

[5.2.](#_heading=h.1idq7dh) Task Sheet (bảng kế hoạch công việc) 31

[5.3.](#_heading=h.42ddq1a) Checklist (bảng kiểm tra chức năng) 32

[5.3.1.](#_heading=h.2hio093) Kiểm tra dữ liệu nhập 32

[5.3.2.](#_heading=h.wnyagw) Kiểm tra các liên kết 33

[**TÀI LIỆU THAM KHẢO** 34](#_heading=h.3gnlt4p)

# **DANH MỤC CÁC TỪ VIẾT TẮT VÀ THUẬT NGỮ**

1. **Thuật Ngữ và Từ Viết Tắt Trong Cơ Sở Dữ Liệu**

* **DBMS**: **Database Management System** – Hệ quản trị cơ sở dữ liệu.
* **RDBMS**: **Relational Database Management System** – Hệ quản trị cơ sở dữ liệu quan hệ.
* **SQL**: **Structured Query Language** – Ngôn ngữ truy vấn có cấu trúc.
* **NoSQL**: **Not Only SQL** – Cơ sở dữ liệu không phải quan hệ, ví dụ MongoDB.
* **ACID**: **Atomicity, Consistency, Isolation, Durability** – Các đặc tính của giao dịch cơ sở dữ liệu.
* **JOIN**: Lệnh trong SQL dùng để kết nối các bảng trong cơ sở dữ liệu.
* **Index**: Một cấu trúc dữ liệu trong cơ sở dữ liệu giúp tăng tốc độ truy vấn.
* **Normalization**: Quá trình tổ chức dữ liệu trong cơ sở dữ liệu để giảm sự dư thừa.

1. **Các Từ Viết Tắt và Thuật Ngữ Cơ Bản**

* MVC: Model-View-Controller – Kiến trúc phần mềm chia thành ba thành phần: Mô hình (Model), Giao diện người dùng (View), và Bộ điều khiển (Controller).
* Eloquent ORM: Object-Relational Mapping – Là một hệ thống ORM mạnh mẽ trong Laravel giúp bạn làm việc với cơ sở dữ liệu như là các đối tượng.
* CLI: Command Line Interface – Giao diện dòng lệnh. Laravel cung cấp một công cụ mạnh mẽ gọi là Artisan CLI để thực hiện các tác vụ như tạo controller, migration, và nhiều tác vụ khác.
* Artisan: Công cụ dòng lệnh của Laravel giúp tự động hóa các tác vụ lập trình thông qua các lệnh như php artisan make:model, php artisan migrate, v.v.
* Blade: Blade là templating engine của Laravel, giúp bạn xây dựng giao diện động một cách dễ dàng và nhanh chóng. Nó cung cấp các cú pháp đơn giản để làm việc với các dữ liệu động trong view.
* Env: Các tệp cấu hình trong Laravel thường được quản lý bằng cách sử dụng tệp .env để lưu trữ các biến môi trường (như thông tin cấu hình cơ sở dữ liệu, API keys, v.v.).
* Seeder: Là các tệp dùng để nhập dữ liệu mẫu vào cơ sở dữ liệu khi cần thiết (ví dụ: dữ liệu ban đầu cho các bảng trong cơ sở dữ liệu).
* Migration: Là công cụ để quản lý cấu trúc cơ sở dữ liệu trong Laravel. Migrations giúp bạn tạo và chỉnh sửa bảng cơ sở dữ liệu bằng cách sử dụng code thay vì làm thủ công.
* Middleware: Là các lớp trung gian giúp bạn kiểm tra và xử lý yêu cầu HTTP trước khi gửi đến Controller. Ví dụ: kiểm tra xác thực người dùng, kiểm tra quyền truy cập.
* Route: Là các định tuyến HTTP trong Laravel, xác định các URL và các hành động mà ứng dụng sẽ thực hiện khi người dùng truy cập các URL đó.
* Request: Là đối tượng chứa thông tin về yêu cầu HTTP của người dùng, như phương thức (GET, POST), dữ liệu gửi lên, v.v.
* Response: Là đối tượng chứa thông tin về phản hồi của ứng dụng khi người dùng yêu cầu một tài nguyên nào đó.
* Controller: Là lớp giúp xử lý các logic của ứng dụng khi nhận được yêu cầu từ các route. Controllers phân tách logic xử lý khỏi views.
* Service Provider: Là các lớp trong Laravel giúp đăng ký các dịch vụ hoặc các phụ thuộc của ứng dụng trong container (IoC container).
* Container (IoC): Inversion of Control – Mô hình quản lý các phụ thuộc (dependencies). Laravel sử dụng container để quản lý các lớp và đối tượng của ứng dụng.

1. **Các Tính Năng và Công Cụ Laravel**

* Queues: Hàng đợi giúp xử lý các tác vụ không đồng bộ như gửi email, xử lý ảnh, hoặc các tác vụ tốn thời gian khác mà không làm gián đoạn quá trình xử lý yêu cầu của người dùng.
* Events & Listeners: Laravel hỗ trợ mô hình Event-Listener, cho phép bạn xử lý các sự kiện và phản ứng lại với chúng, ví dụ như gửi thông báo khi có một người dùng mới đăng ký.
* Job: Tác vụ trong Laravel có thể được xử lý trong các hàng đợi (queue), giúp bạn tách biệt các tác vụ phức tạp hoặc lâu dài khỏi các thao tác yêu cầu chính.
* Testing: Laravel có hỗ trợ rất tốt cho việc Test ứng dụng. Laravel hỗ trợ các công cụ như PHPUnit để kiểm tra tính đúng đắn của ứng dụng và mã nguồn.
* Sanctum: Laravel Sanctum là một giải pháp nhẹ để bảo mật API trong các ứng dụng Laravel, cho phép cấp phát và xác thực các token API cho người dùng.
* Passport: Laravel Passport là một hệ thống hoàn chỉnh để xác thực OAuth2 cho API trong Laravel. Nó cung cấp khả năng tạo các token API cho người dùng và hệ thống của bạn.
* Validation: Là quy trình kiểm tra các dữ liệu đầu vào (request) của người dùng. Laravel cung cấp một hệ thống validation mạnh mẽ, dễ sử dụng để đảm bảo tính toàn vẹn của dữ liệu.
* Collection: Laravel cung cấp Collections, một bộ công cụ mạnh mẽ giúp bạn xử lý các mảng hoặc dữ liệu như một đối tượng, với các phương thức như map(), filter(), reduce(), v.v.
* Resource: Là một lớp giúp biến dữ liệu (thường là dữ liệu từ cơ sở dữ liệu) thành JSON hoặc các định dạng khác để trả về cho người dùng.

1. **Thuật Ngữ trong Laravel API**

* API Resource: Các đối tượng tài nguyên (Resource) trong Laravel giúp bạn định dạng dữ liệu JSON trả về từ API.
* Throttle: Throttle trong Laravel dùng để giới hạn số lần người dùng có thể gửi yêu cầu đến API trong một khoảng thời gian nhất định, nhằm bảo vệ API khỏi các cuộc tấn công DDOS hoặc sử dụng quá mức.
* CORS: Cross-Origin Resource Sharing – Chính sách bảo mật cho phép các tài nguyên trên web có thể được chia sẻ giữa các domain khác nhau. Laravel hỗ trợ cấu hình CORS dễ dàng.
* Rate Limiting: Giới hạn tần suất yêu cầu mà một người dùng có thể gửi đến API trong một khoảng thời gian nhất định.

1. **Các Từ Viết Tắt trong Laravel**

* DB: Database – Cơ sở dữ liệu. Laravel cung cấp một bộ công cụ giúp dễ dàng làm việc với cơ sở dữ liệu thông qua Eloquent ORM hoặc Query Builder.
* PDO: PHP Data Objects – Một cách để giao tiếp với cơ sở dữ liệu trong PHP, Laravel hỗ trợ PDO qua DB facade.
* JWT: JSON Web Token – Một chuẩn để truyền tải thông tin giữa các hệ thống qua các token, đặc biệt trong xác thực API.
* CSRF: Cross-Site Request Forgery – Tấn công mạo danh yêu cầu của người dùng từ một trang web khác. Laravel sử dụng token CSRF để bảo vệ ứng dụng.
* XSS: Cross-Site Scripting – Một loại tấn công bảo mật khi kẻ tấn công đưa mã JavaScript vào trang web của bạn. Laravel bảo vệ các form và đầu vào khỏi XSS.

1. **Thuật Ngữ Liên Quan đến Quản Lý Cơ Sở Dữ Liệu**

* Schema: Là một lớp trong Laravel để tạo hoặc thay đổi cấu trúc cơ sở dữ liệu. Nó hỗ trợ các phương thức như create(), table(), drop().
* Pivot Table: Là bảng liên kết giữa các bảng trong cơ sở dữ liệu khi bạn sử dụng mối quan hệ nhiều-nhiều (Many-to-Many).
* Seeder: Là các lớp được sử dụng để tạo dữ liệu mẫu vào cơ sở dữ liệu. Dữ liệu này có thể được sử dụng trong quá trình phát triển hoặc thử nghiệm.
* Factory: Các lớp giúp tạo dữ liệu mẫu (fake data) cho quá trình thử nghiệm hoặc phát triển.

1. **Thuật Ngữ và Tính Năng Mới**

* Laravel Mix: Một công cụ build dành cho các asset front-end (CSS, JavaScript, v.v.) tích hợp với Webpack. Laravel Mix giúp bạn dễ dàng biên dịch và tối ưu hóa các tài nguyên này.
* Livewire: Một thư viện của Laravel cho phép bạn xây dựng các ứng dụng web động, tương tác mà không cần phải viết nhiều JavaScript.
* Telescope: Là một công cụ giám sát của Laravel, cho phép bạn theo dõi các truy vấn cơ sở dữ liệu, các lỗi, các yêu cầu HTTP, và các sự kiện trong ứng dụng.
* Horizon: Là một công cụ giám sát các hàng đợi trong Laravel, giúp bạn dễ dàng theo dõi và quản lý các job trong queue.

1. **Các Loại Quan Hệ (Relationships)**

* One-to-One: Mối quan hệ một-một giữa hai bảng trong cơ sở dữ liệu (ví dụ: mỗi người dùng chỉ có một profile).
* One-to-Many: Mối quan hệ một-nhiều giữa hai bảng (ví dụ: một bài viết có nhiều bình luận).
* Many-to-Many: Mối quan hệ nhiều-nhiều giữa các bảng (ví dụ: một bài viết có thể có nhiều tag, và một tag có thể thuộc nhiều bài viết).
* Polymorphic Relationships: Mối quan hệ cho phép một mô hình có thể thuộc về nhiều mô hình khác nhau, như bình luận có thể thuộc về bài viết hoặc video.

1. **Các Công Cụ Kiểm Tra (Testing)**

* PHPUnit: Một thư viện PHP để viết các bài kiểm tra (unit test) cho mã nguồn. Laravel tích hợp sẵn PHPUnit để bạn kiểm tra tính đúng đắn của ứng dụng.
* Feature Test: Là bài kiểm tra kiểm tra các tính năng hoặc luồng ứng dụng (ví dụ: đăng nhập, thêm sản phẩm vào giỏ hàng, v.v.).
* Unit Test: Kiểm tra một đơn vị nhỏ của mã nguồn, chẳng hạn như một phương thức trong controller hoặc một hàm trong model.

1. **Các Tính Năng Khác**

* Job: Một lớp trong Laravel dùng để xử lý các tác vụ nặng hoặc các tác vụ không đồng bộ. Các job thường được đẩy vào queue và xử lý sau.
* Queue: Là hệ thống xử lý các tác vụ không đồng bộ, giúp ứng dụng không bị gián đoạn khi thực hiện các tác vụ lâu dài (ví dụ: gửi email, xử lý ảnh).
* Queue Worker: Là tiến trình giúp lấy các job từ hàng đợi và xử lý chúng.

1. **Các Thuật Ngữ Liên Quan đến Tối Ưu Hiệu Suất**

* Caching: Là việc lưu trữ tạm thời dữ liệu để tránh phải tính toán lại hoặc truy vấn cơ sở dữ liệu nhiều lần.
* Redis: Một hệ thống lưu trữ dữ liệu trong bộ nhớ (in-memory data store) thường được dùng trong Laravel để cache hoặc làm queue driver.
* Lazy Loading: Kỹ thuật trong Eloquent giúp giảm số lượng truy vấn đến cơ sở dữ liệu bằng cách chỉ tải các đối tượng liên quan khi cần thiết.
* Eager Loading: Kỹ thuật trong Eloquent giúp tải tất cả các đối tượng liên quan trong một truy vấn duy nhất, giảm thiểu số lượng truy vấn.

# **DANH MỤC BẢNG BIỂU**

# **DANH MỤC HÌNH ẢNH**

# **CHƯƠNG 1: NGHIÊN CỨU CƠ SỞ LÝ THUYẾT**

## Tổng quan về lập trình thiết kế giao diện website

## Giới thiệu về HTML5

HTML5 là một ngôn ngữ cấu trúc và trình bày nội dung cho [**World Wide Web**](https://vi.wikipedia.org/wiki/World_Wide_Web) và sẽ là công nghệ cốt lõi của [Internet](https://vi.wikipedia.org/wiki/Internet) trong tương lai không xa, được đề xuất đầu tiên bởi [Opera Software](https://vi.wikipedia.org/wiki/Opera_Software). Đây là phiên bản thứ năm của ngôn ngữ [HTML](https://vi.wikipedia.org/wiki/HTML) - được tạo ra năm 1990 và chuẩn hóa như HTML4 năm 1997 - và xuất hiện vào tháng 12 năm 2012, là 1 ứng viên được giới thiệu bởi [World Wide Web Consortium](https://vi.wikipedia.org/wiki/W3C) (W3C). Mục tiêu cốt lõi khi thiết kế ngôn ngữ là cải thiện khả năng hỗ trợ cho đa phương tiện mới nhất trong khi vẫn giữ nó dễ dàng đọc được bởi con người và luôn hiểu được bởi các thiết bị và các chương trình máy tính như [trình duyệt web](https://vi.wikipedia.org/wiki/Tr%C3%ACnh_duy%E1%BB%87t_web" \o "Trình duyệt web), phân tích cú pháp, v.v... HTML5 vẫn sẽ giữ lại những đặc điểm cơ bản của [HTML4](https://vi.wikipedia.org/w/index.php?title=HTML4&action=edit&redlink=1) và bổ sung thêm các đặc tả nổi trội của [XHTML](https://vi.wikipedia.org/wiki/XHTML), [DOM](https://vi.wikipedia.org/wiki/DOM) cấp 2,[[3]](https://vi.wikipedia.org/wiki/HTML5#cite_note-HTML5-20110405-3) đặc biệt là [JavaScript](https://vi.wikipedia.org/wiki/JavaScript).

Là phiên bản tiếp sau của HTML 4.01 và XHTML 1.1, HTML5 phản ánh thực tế rằng HTML và XHTML được sử dụng phổ biến trên World Wide Web là một hỗn hợp các tính năng với các thông số kĩ thuật khác nhau, được giới thiệu bởi nhiều nhà sản xuất phần mềm, cùng với các sản phẩm phần mềm được giới thiệu như trình duyệt web, những người thành lập phổ biến thực tế và có quá nhiều lỗi cú pháp trong các văn bản web. Đây là một nỗ lực để xác định một [ngôn ngữ đánh dấu](https://vi.wikipedia.org/wiki/Ng%C3%B4n_ng%E1%BB%AF_%C4%91%C3%A1nh_d%E1%BA%A5u" \o "Ngôn ngữ đánh dấu) duy nhất có thể được viết bằng cú pháp HTML hoặc XHTML. Nó bao gồm các mô hình xử lý chi tiết để tăng tính tương thích, mở rộng, cải thiện và chuẩn hóa các đánh dấu có sẵn cho tài liệu, đưa ra các đánh đấu mới và giới thiệu [giao diện lập trình ứng dụng](https://vi.wikipedia.org/wiki/Giao_di%E1%BB%87n_l%E1%BA%ADp_tr%C3%ACnh_%E1%BB%A9ng_d%E1%BB%A5ng" \o "Giao diện lập trình ứng dụng) (application programming interfaces API) để tạo ra các [ứng dụng Web](https://vi.wikipedia.org/wiki/%E1%BB%A8ng_d%E1%BB%A5ng_web" \o "Ứng dụng web) phức tạp. Cùng một lý do như vây, HTML5 là một ứng cử viên tiềm năng cho nền tảng ứng dụng di động. Nhiều tính năng của HTML5 được xây dựng với việc xem xét chúng có thể sử dụng được trên các thiết bị di động như [điện thoại thông minh](https://vi.wikipedia.org/wiki/%C4%90i%E1%BB%87n_tho%E1%BA%A1i_th%C3%B4ng_minh" \o "Điện thoại thông minh) và [máy tính bảng](https://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh_b%E1%BA%A3ng" \o "Máy tính bảng) hay không. Trong tháng 12 năm 2011, công ty nghiên cứu Strategy Analytics dự báo doanh số bán hàng của điện thoại tương thích HTML5 sẽ đứng đầu 1 tỷ vào năm 2013.[[4]](https://vi.wikipedia.org/wiki/HTML5#cite_note-4)

Đặc biệt, HTML5 có thêm nhiều tính năng cú pháp mới. Chúng bao gồm các thẻ mới như <video>, <audio> và các thành phần <canvas>, cũng như sự tích hợp của đồ họa vector có khả năng mở rộng (Scalable Vector Graphics) nội dung (thay thế việc sử dụng thẻ chung <object>) và MathML cho các công thức toán học. Những tính năng này được thiết kế để làm cho nó dễ dàng bao quát, xử lý đa phương tiện và nội dung đồ họa trên web mà không cần phải dùng đến quyền sở hữu bổ sung và APIs. Các yếu tố mới khác, chẳng hạn như <section>, <article>, <header> và <nav>, được thiết kế để làm phong phú thêm nội dung ngữ nghĩa của tài liệu. Thuộc tính mới đã được giới thiệu với mục đích tương tự, trong khi một số yếu tố và các thuộc tính đã được loại bỏ. Một số yếu tố, chẳng hạn như <a>, <cite> và <menu> đã được thay đổi, xác định lại hoặc chuẩn hóa. APIs và Document Object Model (DOM) không phải suy nghĩ muộn hơn quá nhiều, nhưng là bộ phận cơ bản của đặc điểm kỹ thuật HTML5. HTML5 cũng xác định cụ thể một số các xử lý cần thiết cho các tài liệu không hợp lệ để các lỗi cú pháp sẽ được xử lý thống nhất của tất cả các trình duyệt phù hợp và các tác nhân người dùng khác.[[5]](https://vi.wikipedia.org/wiki/HTML5#cite_note-:0-5)

## Giới thiệu về CSS3

CSS3 (Cascading Style Sheets) là phiên bản mới nhất của ngôn ngữ định kiểu được sử dụng rộng rãi trong phát triển web. CSS3 cung cấp cho các nhà phát triển web một bộ công cụ mạnh mẽ để tạo kiểu cho các trang web, cho phép kiểm soát chính xác giao diện và trải nghiệm người dùng.

Đi sâu hơn, CSS3 còn giới thiệu các tính năng mới như flexbox, grid layout, chuyển đổi mượt mà, và biến đổi, giúp xây dựng các bố cục phức tạp và hiệu ứng động một cách dễ dàng. Các tính năng này mở ra nhiều khả năng sáng tạo, nâng cao tính thẩm mỹ và tương tác của website.

Ngoài ra, CSS3 còn được tích hợp tốt với các công cụ phát triển web khác, cho phép tạo ra các website đáp ứng và tương thích với nhiều thiết bị khác nhau.

Ưu điểm của CSS3 là gì?

CSS3 cung cấp vị trí nhất quán và chính xác của các phần tử có thể điều hướng. Có thể dễ dàng tùy chỉnh một trang web vì nó có thể được thực hiện chỉ bằng cách thay đổi một tệp module. Đồ họa dễ dàng hơn trong CSS3, do đó dễ dàng làm cho trang web trở nên hấp dẫn.

CSS3 cho phép xem các video trực tuyến mà không cần sử dụng các plugin của bên thứ ba và giúp tiết kiệm thời gian hầu hết các trình duyệt đều hỗ trợ nó.

## Tổng quan về ngôn ngữ lập trình Javascript

JavaScript là ngôn ngữ lập trình có kịch bản dựa trên những đối tượng có sẵn hoặc chính người lập trình tự nghĩ ra.

Ngôn ngữ này được sử dụng rộng rãi trong các ứng dụng website. Hiện tại, nó cũng được sử dụng trên các trình duyệt phổ biến mà bạn vẫn hay dùng như Chrome, Firefox, Cốc Cốc,…

JavaScript thường được nhúng trực tiếp vào một trang web hoặc được tham chiếu qua file .js. Nó là ngôn ngữ phía Client. Script được tải về máy khách truy cập và được xử lý tai đó thay vì pía server là xử lsy trên server rồi mới đưa kết quả tới khách truy cập.

Ngôn ngữ JS giúp hỗ trợ đầy đủ việc khách hàng muốn tắt hay mở trên các trình duyệt web được ứng dụng cơ bản hiện nay. Nhờ vậy, việc có thể xác định được website đang hoạt động như thế nào khi không có ngôn ngữ JS đang hoạt động.

Hoạt động đơn giản và hiệu quả là những gì mà ngôn ngữ JavaScript sở hữu. Nó ứng dụng trong nhiều môi trường, nhiều nền tảng khác nhau.

a. Ưu điểm

JavaScript có rất nhiều ưu điểm giúp nó trở nên vượt trội hơn hẳn so với các ngôn ngữ khác. Một số ưu điểm có thể kể đến như:

Dễ dàng, đơn giản nên nó dễ học hơn so với các ngôn ngữ khác

Lỗi dễ phát hiện và dễ sửa

Có thể hoạt động trên nhiều trình duyệt và nền tảng

JavaScript có thể được gắn trên một số element của trang web hay event của trang web như là thông qua click chuột hoặc di chuột

Bạn không cần một compiler vì web browser có thể biên dịch nó bằng HTML

Nó dễ học hơn so với các ngôn ngữ khác

Mang tới các tính năng bổ sung cho các website

Giao diện thân thiện, nhiều tính năng

b. Nhược điểm

Cái gì cũng có hai mặt của nó. Đồng nghĩa với việc nổi tiếng như JavaScript thì sẽ có khá nhiều Hacker luôn tìm kiếm những lỗ hổng và các lỗi bảo mật để lợi dụng.

Một số khuyết điểm có thể kể đến như:

Dễ bị khai thác

Có thể sử dụng để thực thi mã độc trên máy tính của người dùng

JS code snippets khá lớn

Triển khai khác nhau tuỳ từng thiết bị khiến nguy cơ không đồng nhất có thể xảy ra, gây ảnh hưởng tới quá trình thực hiện mục tiêu cuối cùng

## Thư viện Jquery

jQuery là một thư viện JavaScript rất phổ biến được sử dụng để tạo ra các hiệu ứng động, thay đổi nội dung của trang web, tương tác với người dùng và thao tác với tài liệu HTML và CSS một cách dễ dàng và tiện lợi. jQuery cung cấp các phương thức để thao tác với HTML DOM, xử lý sự kiện, tạo ra hiệu ứng động, gọi Ajax để gửi và nhận dữ liệu từ máy chủ, thao tác với CSS và nhiều tính năng khác.

jQuery được phát triển bởi John Resig vào năm 2006 và được phát hành dưới giấy phép MIT. jQuery rất phổ biến và được sử dụng rộng rãi trong phát triển web, đặc biệt là trong việc tạo ra các trang web tương tác và ứng dụng web đơn trang (Single Page Application - SPA).

Để sử dụng jQuery, bạn cần đưa thư viện vào trang web của mình bằng cách thêm đoạn mã sau vào phần head của tài liệu HTML:

<head>

    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

</head>

Sau đó, bạn có thể sử dụng các phương thức của jQuery bằng cách viết mã JavaScript trong phần script của tài liệu HTML.

Các tính năng nổi bật của jQuery

Một số tính năng chính của jQuery bao gồm:

Thao tác với HTML DOM: jQuery cho phép bạn thao tác với HTML DOM một cách dễ dàng và tiện lợi hơn bằng cách sử dụng các phương thức như $(selector).method() để chọn các phần tử trên trang và thay đổi nội dung của chúng, thêm hoặc xóa các phần tử, thay đổi thuộc tính và lớp CSS.

Xử lý sự kiện: jQuery cung cấp các phương thức để xử lý sự kiện như click, hover, submit, change, và keypress. Bạn có thể sử dụng các phương thức này để xử lý các sự kiện của người dùng một cách dễ dàng.

Hiệu ứng động: jQuery cung cấp các phương thức để tạo ra các hiệu ứng động trên trang web như fadeIn, fadeOut, slideUp, slideDown, animate và toggle. Bạn có thể sử dụng các phương thức này để thêm tính năng động cho trang web của mình.

Ajax: jQuery cho phép bạn gửi và nhận dữ liệu từ máy chủ bằng cách sử dụng phương thức Ajax. Bạn có thể sử dụng các phương thức như $.ajax() và $.getJSON() để gửi và nhận dữ liệu từ máy chủ một cách nhanh chóng và tiện lợi.

Thao tác với CSS: jQuery cho phép bạn thao tác với CSS trên trang web của mình bằng cách sử dụng các phương thức như addClass, removeClass, và css. Bạn có thể sử dụng các phương thức này để thêm hoặc xóa lớp CSS hoặc thay đổi thuộc tính CSS của các phần tử trên trang.

Hỗ trợ đa trình duyệt: jQuery được thiết kế để hoạt động trên tất cả các trình duyệt phổ biến, bao gồm cả Internet Explorer, Firefox, Chrome, Safari và Opera.

## Framework Bootstrap

Bootstrap là một framework [HTML](https://vietnix.vn/html-la-gi/), [CSS](https://vietnix.vn/css-la-gi/), và [JavaScript](https://vietnix.vn/javascript-la-gi/) phổ biến nhất để phát triển website chuẩn responsive.

Bootstrap giúp cho quá trình thiết kế website của bạn diễn ra nhanh chóng dựa trên những thành tố cơ bản sẵn có như typography, forms, buttons, tables, grids, navigation, image carousels,…

Lịch sử về Bootstrap

Tiếp theo, ta cùng tìm hiểu về lịch sử Bootstrap một chút để hiểu rõ hơn Bootstrap là gì. Bootstrap là sản phẩm của Mark Otto và Jacob Thornton tại Twitter. Tên gọi ban đầu của Bootstrap là Twitter Blueprint. Mục đích lúc đầu của nó như một mã nguồn mở vào ngày 19/09/2011 trên GitHub dùng để cải thiện tính nhất quán giữa các công cụ nội bộ.

Vào ngày 31 tháng 1 năm 2012, Bootstrap phiên bản 2 chính thức được phát hành. Phiên bản mới này đã được hỗ trợ từ Glyphicons cũng như thay đổi một số thành phần có sẵn. Đồng thời, phiên bản này cũng hỗ trợ responsive web design (RWD) – tức là bố cục của các trang web sẽ tự động điều chỉnh theo đặc điểm của thiết bị được sử dụng (bất kể đó là desktop, tablet hay mobile).

Ngày 19/08/2013 đánh dấu sự ra đời của phiên bản tiếp theo – Bootstrap phiên bản 3. Phiên bản này đã tái thiết kế lại các thành phần để sử dụng flat design, đồng thời tương thích với các thiết bị di động hơn. Bên cạnh đó, Bootstrap 3 cũng có một hệ thống plugin hoàn toàn mới với các namespaced event. Dù Bootstrap 3 không còn hỗ trợ Internet Explorer 7 và Firefox 3.6 nữa, nhưng vẫn có các polyfill cho những trình duyệt này.

Phải đến tận tháng 1 năm 2018, người dùng mới được sử dụng phiên bản mới của Bootstrap – Bootstrap 4. Và gần đây nhất chính là Bootstrap 5 vào 5/5/2021. Các bạn có thể [tải Bootstrap](https://getbootstrap.com/" \t "_blank) phiên bản mới nhất tại trang web chính thức.

3 file chính Bootstrap

Để quản lý giao diện người dùng và các chức năng của website với 3 file chính dưới đây:

Bootstrap.CSS

Bootstrap.JS

Glyphicons

## Tổng quan về hệ quản trị cơ sở đữ liệu MySQL

## Giới thiệu về MySQL

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System – RDBMS) mã nguồn mở, được sử dụng rộng rãi để lưu trữ, quản lý và truy xuất dữ liệu. MySQL hoàn toàn miễn phí nằm trong nhóm LAMP (Linux – Apache – MySQL – PHP).

MySQL đọc như thế nào? Một số người sẽ đọc là My Sequel một số khác lại phát âm chúng là MY-ES-KYOO-EL. Cả hai cách đọc trên đều đúng và những ai đã biết đến MySQL đều sẽ hiểu.

Xem tiếp...

Vì hệ quản trị cơ sở dữ liệu MySQL ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh và Mysql cũng có cùng một cách truy xuất và mã lệnh tương tự với ngôn ngữ SQL chính vì thế nên MySQL được sử dụng và hỗ trợ của những lập trình viên yêu thích mã nguồn mở.

Nhưng MySQL không bao quát toàn bộ những câu truy vấn cao cấp như SQL Server. Vì vậy MySQL chỉ đáp ứng việc truy xuất đơn giản trong quá trình vận hành của website, thích hợp cho các ứng dụng có truy cập [CSDL](https://topdev.vn/blog/phan-1-gioi-thieu-co-so-du-lieu-csdl-he-quan-tri-csdl/) trên internet và có thể giải quyết hầu hết các bài toán trong PHP, Perl.

MySQL có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows,Linux, Mac OSX, Unix, FreeBSD, NetBSD, Novell NetWare, SGI Irix, Solaris, SunOS,…

MySQLlà một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL).

Xem tiếp...

Lịch sử hình thành và phát triển của MySQL

MySQL được phát triển vào năm 1994 bởi công ty Thụy Điển MySQL AB. Sau khi trải qua nhiều giai đoạn phát triển và chuyển giao, với những cột mốc quan trọng như việc trở thành mã nguồn mở vào năm 2000, sau đó Sun Microsystems đã mua lại MySQL vào năm 2008. Nhưng sau đó chỉ hai năm gã khổng lồ Oracle đã giữ quyền sở hữu MySQL vào năm 2010

Sự ra đời của MariaDB như một “fork” của MySQL đã tạo ra một cuộc cạnh tranh lành mạnh, thúc đẩy cả hai hệ thống không ngừng cải tiến. Nhờ tính năng mã nguồn mở, hiệu suất cao, dễ sử dụng và cộng đồng lớn, MySQL đã trở thành một trong những hệ quản trị cơ sở dữ liệu phổ biến nhất thế giới, được sử dụng rộng rãi trong các ứng dụng web và các hệ thống lưu trữ dữ liệu.

Đặc điểm của MySQL

Hiệu suất cao và khả năng mở rộng: MySQL được tối ưu hóa để xử lý một lượng lớn dữ liệu một cách nhanh chóng và hiệu quả. Hệ thống có thể dễ dàng mở rộng để đáp ứng nhu cầu lưu trữ và xử lý dữ liệu ngày càng tăng.

Bảo mật và tính toàn vẹn dữ liệu: MySQL cung cấp các tính năng bảo mật mạnh mẽ như xác thực người dùng, phân quyền, mã hóa dữ liệu và phát hiện xâm nhập, giúp bảo vệ dữ liệu khỏi các mối đe dọa an ninh.

Tính linh hoạt và tùy biến cao: MySQL hỗ trợ nhiều nền tảng, nhiều loại engine lưu trữ và cho phép người dùng tùy chỉnh cấu hình để phù hợp với nhu cầu cụ thể của ứng dụng.

Cộng đồng lớn và hỗ trợ mạnh mẽ: MySQL có một cộng đồng người dùng rất lớn, sẵn sàng chia sẻ kinh nghiệm và giải đáp thắc mắc.

Mã nguồn mở: MySQL là một phần mềm mã nguồn mở, người dùng có thể tự do sử dụng, sửa đổi và phân phối.

Cơ chế quản lý dữ liệu: MySQL quản lý dữ liệu thông qua các cơ sở dữ liệu (database). Mỗi cơ sở dữ liệu có thể chứa nhiều bảng (table), mỗi bảng lưu trữ một loại dữ liệu cụ thể. Các bảng được liên kết với nhau thông qua các khóa (key) để tạo thành các mối quan hệ.

Cơ chế phân quyền: MySQL có cơ chế phân quyền người dùng rất chi tiết. Mỗi người dùng có thể được cấp quyền truy cập đến một hoặc nhiều cơ sở dữ liệu, với các quyền hạn khác nhau như đọc, ghi, sửa đổi cấu trúc bảng, v.v.

Ngôn ngữ SQL: MySQL sử dụng ngôn ngữ SQL (Structured Query Language) để tương tác với dữ liệu. SQL là ngôn ngữ chuẩn để truy vấn, cập nhật, xóa và quản lý dữ liệu trong các hệ quản trị cơ sở dữ liệu quan hệ.

## Ưu điểm của MySQL

Nhanh chóng và tiết kiệm chi phí: MySQL thực hiện công việc một cách nhanh chóng nhờ vào việc thiết lập các tiêu chuẩn và tận dụng khả năng làm việc hiệu quả. Điều này giúp tăng tốc độ thực thi và tối ưu chi phí cho doanh nghiệp.

Mạnh mẽ và mở rộng: MySQL không chỉ xử lý một lượng lớn dữ liệu mà còn có khả năng mở rộng linh hoạt theo nhu cầu, đặc biệt phù hợp với các ứng dụng đòi hỏi khả năng mở rộng khi cần thiế

Đa tính năng: MySQL hỗ trợ nhiều tính năng SQL, cung cấp những chức năng được mong đợi từ một hệ quản trị CSDL quan hệ, cả trực tiếp và gián tiếp.

Độ bảo mật cao: MySQL là lựa chọn lý tưởng cho các ứng dụng truy cập CSDL qua internet, nhờ vào tính năng bảo mật và đặc biệt ở mức độ an toàn cao.

Dễ sử dụng: MySQL không chỉ là cơ sở dữ liệu dễ sử dụng mà còn đảm bảo ổn định và tốc độ cao. Nó hoạt động trên nhiều hệ điều hành và cung cấp một hệ thống chức năng mạnh mẽ.

## Nhược điểm của MySQL

Hạn chế dung lượng: Trong trường hợp số lượng bản ghi ngày càng gia tăng, quá trình truy xuất dữ liệu sẽ trở nên khó khăn. Để giải quyết vấn đề này, bạn có thể áp dụng nhiều biện pháp như tạo cache MySQL hoặc phân tải cơ sở dữ liệu ra nhiều máy chủ.

Độ tin cậy thấp: Một nhược điểm khác của MySQL là trong quá trình thực hiện các chức năng cụ thể như kiểm toán, giao dịch và quản lý tài liệu tham khảo, MySQL có thể trở nên kém tin cậy hơn một số hệ quản trị cơ sở dữ liệu quan hệ khác.

Giới hạn chức năng: MySQL không thiết kế để thực hiện toàn bộ các chức năng và nó đi kèm với những hạn chế về chức năng mà một số ứng dụng có thể cần.

## Tổng quan về ngôn ngữ lập trình phía máy chủ

## Giới thiệu về nền tảng công nghệ nguồn mở

Nền tảng công nghệ nguồn mở (open-source technology platform) là các hệ thống, phần mềm, công cụ hoặc môi trường phát triển mà mã nguồn của chúng được công khai và cho phép người dùng tự do sử dụng, thay đổi và phân phối lại mà không bị hạn chế bởi các bản quyền phần mềm.

Dưới đây là một số đặc điểm và lợi ích chính của nền tảng công nghệ nguồn mở:

1. Mã nguồn mở

Mã nguồn của phần mềm hoặc công nghệ là công khai và có sẵn cho bất kỳ ai. Điều này giúp các nhà phát triển có thể tiếp cận và chỉnh sửa phần mềm theo nhu cầu của mình.

Các phần mềm nguồn mở thường được phát triển và duy trì bởi cộng đồng người dùng và lập trình viên toàn cầu.

2. Tự do sử dụng và phát triển

Người dùng có thể tự do tải về, sử dụng, thay đổi và phân phối lại phần mềm mà không gặp phải các rào cản pháp lý.

Công nghệ nguồn mở khuyến khích sự đổi mới và sáng tạo trong cộng đồng lập trình viên, khi họ có thể xây dựng và cải tiến phần mềm một cách tự do.

3. Khả năng tùy chỉnh và linh hoạt

Với mã nguồn mở, người dùng có thể tùy chỉnh phần mềm để phù hợp với yêu cầu và môi trường sử dụng riêng của mình.

Các tổ chức có thể phát triển các tính năng đặc biệt hoặc sửa lỗi mà phần mềm chính thức không hỗ trợ.

4. An toàn và bảo mật

Do mã nguồn được công khai, nhiều người có thể kiểm tra và phát hiện các lỗ hổng bảo mật, giúp cải thiện độ an toàn của phần mềm.

Nhiều tổ chức, doanh nghiệp hoặc cộng đồng kỹ thuật cũng đóng góp vào việc vá lỗi bảo mật, làm tăng độ tin cậy của phần mềm.

5. Tiết kiệm chi phí

Các nền tảng công nghệ nguồn mở thường miễn phí hoặc có chi phí rất thấp, điều này giúp giảm chi phí bản quyền phần mềm.

Các tổ chức và cá nhân có thể tiết kiệm chi phí đáng kể trong việc phát triển và duy trì hệ thống công nghệ của mình.

6. Cộng đồng hỗ trợ

Các nền tảng công nghệ nguồn mở thường có cộng đồng người dùng, lập trình viên và chuyên gia đóng góp, trao đổi kiến thức và hỗ trợ nhau.

Các diễn đàn, blog, tài liệu và mã nguồn sẵn có giúp người dùng nhanh chóng giải quyết vấn đề.

7. Ví dụ về nền tảng công nghệ nguồn mở

Linux: Một hệ điều hành nguồn mở phổ biến, được sử dụng rộng rãi trên các máy chủ, thiết bị di động và hệ thống nhúng.

Apache: Máy chủ web mã nguồn mở, là một trong những phần mềm quan trọng nhất cho việc cung cấp dịch vụ web.

WordPress: Hệ quản trị nội dung (CMS) phổ biến mã nguồn mở, dùng để xây dựng các trang web và blog.

MySQL: Một hệ quản trị cơ sở dữ liệu mã nguồn mở, được sử dụng trong nhiều ứng dụng web và phần mềm.

Python: Ngôn ngữ lập trình phổ biến với cộng đồng mã nguồn mở mạnh mẽ và nhiều thư viện hỗ trợ phát triển phần mềm.

Kết luận

Nền tảng công nghệ nguồn mở mang lại nhiều lợi ích cho các tổ chức và cá nhân, từ việc tiết kiệm chi phí đến việc cải tiến và phát triển phần mềm linh hoạt. Sự cộng tác trong cộng đồng cũng giúp thúc đẩy sự đổi mới và bảo mật, góp phần làm cho các công nghệ này ngày càng trở nên mạnh mẽ và phổ biến.

## Giới thiệu về ngôn ngữ lập trình PHP

PHP (Hypertext Preprocessor) là một ngôn ngữ lập trình kịch bản (scripting language) mã nguồn mở, chủ yếu được sử dụng để phát triển các ứng dụng web động. PHP được thiết kế để dễ dàng tích hợp vào HTML, giúp các lập trình viên xây dựng các trang web tương tác và động một cách nhanh chóng và hiệu quả.

1. Lịch sử và phát triển

Ra đời: PHP được tạo ra lần đầu vào năm 1994 bởi Rasmus Lerdorf. Lúc đầu, PHP chỉ là một tập hợp các công cụ giúp theo dõi các hồ sơ của người truy cập trang web.

Cải tiến: Qua thời gian, PHP đã phát triển mạnh mẽ và trở thành một ngôn ngữ lập trình mạnh mẽ cho các ứng dụng web. Phiên bản hiện tại của PHP (tính đến 2025) là PHP 8.x với nhiều tính năng cải tiến về hiệu suất và bảo mật.

2. Đặc điểm của PHP

Mã nguồn mở: PHP là một ngôn ngữ mã nguồn mở, có thể sử dụng miễn phí và được phát triển bởi cộng đồng lập trình viên toàn cầu.

Chạy trên máy chủ (server-side): PHP là một ngôn ngữ phía máy chủ, có nghĩa là mã PHP được xử lý trên máy chủ trước khi được gửi đến trình duyệt của người dùng. Điều này khác với JavaScript, vốn được thực thi trên trình duyệt của người dùng.

Tích hợp HTML và SQL: PHP có thể dễ dàng tích hợp với HTML để xây dựng các trang web động, và kết nối với các cơ sở dữ liệu như MySQL, PostgreSQL, hoặc SQLite để lưu trữ và truy xuất dữ liệu.

Cộng đồng phát triển mạnh mẽ: Với một cộng đồng phát triển rộng lớn, PHP luôn được cập nhật và cải tiến liên tục. Các thư viện, framework và tài liệu hỗ trợ PHP rất phong phú.

3. Lợi ích của PHP

Dễ học và sử dụng: PHP có cú pháp dễ hiểu và gần gũi với các ngôn ngữ lập trình khác, giúp các lập trình viên mới làm quen nhanh chóng.

Hiệu suất cao: Các phiên bản PHP mới liên tục cải tiến hiệu suất, giúp chạy các ứng dụng web nhanh và hiệu quả.

Hỗ trợ mạnh mẽ cho cơ sở dữ liệu: PHP hỗ trợ kết nối với nhiều loại cơ sở dữ liệu khác nhau, đặc biệt là MySQL và MariaDB, giúp phát triển các ứng dụng web động với dữ liệu được lưu trữ trong cơ sở dữ liệu.

Phát triển ứng dụng web nhanh chóng: Với khả năng tích hợp với các framework như Laravel, Symfony, và CodeIgniter, PHP giúp việc phát triển ứng dụng web trở nên nhanh chóng và dễ dàng hơn.

4. Ứng dụng của PHP

PHP thường được sử dụng để phát triển các ứng dụng web động, bao gồm:

Website thương mại điện tử: Nhiều nền tảng thương mại điện tử như Magento, WooCommerce (WordPress), và PrestaShop được xây dựng bằng PHP.

Hệ quản trị nội dung (CMS): WordPress, Joomla, và Drupal đều được xây dựng trên nền tảng PHP, giúp người dùng tạo ra và quản lý nội dung trang web một cách dễ dàng.

Ứng dụng web: PHP được sử dụng để xây dựng các ứng dụng web từ nhỏ đến lớn, ví dụ như các trang web xã hội, diễn đàn trực tuyến, và các công cụ quản lý công việc.

API và dịch vụ web: PHP có thể được sử dụng để xây dựng các dịch vụ web RESTful, cung cấp các API cho các ứng dụng di động hoặc web khác.

5. Framework PHP phổ biến

Để tối ưu hóa quá trình phát triển, nhiều framework PHP đã được phát triển, giúp lập trình viên tiết kiệm thời gian và giảm thiểu lỗi. Một số framework PHP phổ biến bao gồm:

Laravel: Đây là một framework PHP rất mạnh mẽ và phổ biến, cung cấp nhiều tính năng như routing, authentication, và ORM (Object-Relational Mapping).

Symfony: Một framework linh hoạt và mở rộng, phù hợp với các dự án web quy mô lớn.

CodeIgniter: Đơn giản và nhẹ, CodeIgniter rất thích hợp cho các dự án nhỏ và vừa.

Yii: Framework PHP được tối ưu hóa cho các ứng dụng web có hiệu suất cao và bảo mật tốt.

6. PHP trong thời đại hiện nay

Tính phổ biến: PHP vẫn là một trong những ngôn ngữ lập trình phổ biến nhất cho phát triển web, mặc dù có sự cạnh tranh từ các ngôn ngữ khác như JavaScript (với Node.js) và Python.

Hỗ trợ hiện đại: PHP liên tục cải tiến với các phiên bản mới như PHP 7 và PHP 8, cung cấp các tính năng hiện đại, tối ưu hóa hiệu suất và cải thiện bảo mật.

Tính linh hoạt và phát triển liên tục: PHP không chỉ giới hạn trong các ứng dụng web cơ bản mà còn có thể sử dụng để phát triển các ứng dụng web phức tạp và quy mô lớn, nhờ vào các công cụ và thư viện mạnh mẽ.

Kết luận

PHP là một ngôn ngữ lập trình mạnh mẽ và linh hoạt, đặc biệt phù hợp với phát triển ứng dụng web. Với cộng đồng lớn mạnh và khả năng tích hợp tốt với cơ sở dữ liệu và HTML, PHP vẫn là một lựa chọn phổ biến cho các lập trình viên khi xây dựng các website và ứng dụng web.

## Tổng quan về Framework PHP Laravel

Laravel là một framework PHP mã nguồn mở miễn phí được phát hành lần đầu tiên vào năm 2011. Taylor Otwell đã tạo ra Laravel nhằm giúp các Developer xây dựng các ứng dụng web bằng cách sử dụng mẫu kiến ​​trúc Model-View-Controller (MVC).

Laravel được biết đến với cú pháp tinh tế, cách tiếp cận thân thiện và bộ tính năng phong phú giúp đơn giản hóa các tác vụ phát triển web phổ biến, chẳng hạn như CodeIgniter, Yii và các ngôn ngữ lập trình khác như Ruby on Rails.

Laravel tận dụng tối đa các tính năng tiên tiến của PHP, chẳng hạn như lập trình hướng đối tượng (OOP), cung cấp cú pháp rõ ràng và có khả năng truyền đạt tốt, giúp dễ dàng viết mã code duy trì và mở rộng. Nếu đã quen thuộc với Core PHP và PHP nâng cao, Laravel sẽ giúp công việc của bạn trở nên dễ dàng hơn, tiết kiệm nhiều thời gian, đặc biệt nếu bạn dự định phát triển một trang web từ đầu.

Ngoài ra, Laravel còn mang đến sự an toàn cho ứng dụng web, giúp ngăn chặn các cuộc tấn công mạng phổ biến.

Laravel là một framework back-end, được viết bằng ngôn ngữ lập trình PHP. Laravel cung cấp các công cụ và cấu trúc để xây dựng phần logic của ứng dụng web ví dụ như: Xử lý yêu cầu và phản hồi HTTP, kết nối và thao tác với cơ sở dữ liệu, xác thực và phân quyền người dùng, quản lý cache, gửi email và nhiều tính năng khác.

Laravel không phải là framework front-end. Laravel không cung cấp các công cụ để xây dựng giao diện người dùng (UI) của ứng dụng web, chẳng hạn như HTML, CSS và JavaScript. Tuy nhiên, Laravel có thể được tích hợp với các framework front-end khác để tạo ra các ứng dụng web hoàn chỉnh. Một số framework front-end phổ biến được sử dụng với Laravel có thể kể đến như: Vue.js, React, Angular…

Nhược điểm của Laravel Framework

Lộ trình học tập khó khăn

Laravel có lộ trình học tập khó khăn đối với các Developer mới làm quen với framework này. Điều này là do Laravel có rất nhiều tính năng và khái niệm cần tìm hiểu và có thể mất một thời gian để thành thạo.

Hiệu suất chậm

Hiệu suất của ứng dụng web Laravel có thể chậm hơn so với một số khung PHP khác, đặc biệt là khi xử lý lượng lớn dữ liệu hoặc lưu lượng truy cập. Tuy nhiên, có một số kỹ thuật tối ưu hóa mà Developer có thể sử dụng để cải thiện hiệu suất của framework.

Khó khăn khi nâng cấp

Việc nâng cấp lên phiên bản mới của Laravel có thể gặp khó khăn, đặc biệt nếu ứng dụng sử dụng nhiều gói và thư viện Laravel của bên thứ ba. Sở dĩ do các phiên bản mới của Laravel có thể đưa ra những thay đổi yêu cầu cập nhật code.

**CHƯƠNG 2:**

**TỔNG QUAN VỀ ĐỀ TÀI**

**2.1. Đặt vấn đề**

Trong bối cảnh hiện nay, thị trường siêu xe đang phát triển mạnh mẽ với sự xuất hiện của nhiều mẫu xe sang trọng, công nghệ tiên tiến và các thương hiệu nổi tiếng toàn cầu. Khách hàng trong phân khúc này không chỉ quan tâm đến chất lượng sản phẩm mà còn đòi hỏi sự chuyên nghiệp trong quá trình tư vấn, bán hàng và dịch vụ hậu mãi.

Các doanh nghiệp kinh doanh siêu xe cần một hệ thống quản lý bán hàng hiệu quả để tối ưu hóa quy trình kinh doanh, quản lý thông tin khách hàng VIP, theo dõi các tùy chọn cá nhân hóa của xe và đảm bảo dịch vụ hậu mãi hoàn hảo. Tuy nhiên, nhiều doanh nghiệp vẫn chưa áp dụng công nghệ quản lý hiện đại, dẫn đến thiếu sự chuyên nghiệp, chậm trễ trong quy trình vận hành, và hạn chế trong khả năng đáp ứng nhu cầu đặc thù của khách hàng cao cấp.

Việc xây dựng hệ thống quản lý bán siêu xe sử dụng framework Laravel là một giải pháp toàn diện giúp doanh nghiệp khắc phục những hạn chế này. Laravel, với các tính năng mạnh mẽ như Eloquent ORM, Blade Template Engine, hệ thống bảo mật vượt trội, và khả năng mở rộng linh hoạt, là một nền tảng lý tưởng để phát triển các hệ thống quản lý bán hàng cho ngành siêu xe.

Mục tiêu của việc nghiên cứu và phát triển hệ thống này là cung cấp một nền tảng quản lý bán hàng chuyên nghiệp, dễ sử dụng, hỗ trợ tối ưu các quy trình từ quản lý sản phẩm, khách hàng, kho hàng đến dịch vụ hậu mãi. Đồng thời, hệ thống cần có khả năng mở rộng, tích hợp thêm các tính năng mới khi doanh nghiệp phát triển trong tương lai.

**2.2. Hệ thống hiện tại**

Hiện nay, nhiều doanh nghiệp kinh doanh siêu xe vẫn đang đối mặt với các vấn đề về quản lý bán hàng do sử dụng các phương pháp thủ công hoặc phần mềm không phù hợp. Một số vấn đề chính bao gồm:

1. Quản lý thủ công hoặc bằng công cụ cơ bản:
   * Nhiều doanh nghiệp vẫn sử dụng sổ sách hoặc bảng tính Excel để theo dõi thông tin sản phẩm, khách hàng và giao dịch.
   * Điều này không chỉ làm tăng nguy cơ sai sót mà còn khiến việc quản lý dữ liệu trở nên khó khăn, đặc biệt với khách hàng VIP đòi hỏi sự chính xác và nhanh chóng.
2. Khả năng mở rộng và linh hoạt hạn chế:
   * Khi doanh nghiệp mở rộng quy mô, hệ thống cũ không thể đáp ứng các yêu cầu mới, như tích hợp các dịch vụ cao cấp hoặc nâng cấp giao diện thân thiện hơn cho nhân viên và khách hàng.
   * Việc tùy chỉnh hoặc nâng cấp phần mềm hiện tại gặp nhiều khó khăn và tốn kém chi phí.
3. Thiếu tính tự động hóa và trải nghiệm khách hàng:
   * Hệ thống hiện tại không hỗ trợ tự động hóa các quy trình như thông báo lịch bảo trì, nhắc nhở khách hàng về các dịch vụ hậu mãi, hoặc cập nhật trạng thái đơn hàng theo thời gian thực.
   * Điều này làm giảm sự hài lòng của khách hàng, đặc biệt trong phân khúc cao cấp nơi chất lượng dịch vụ đóng vai trò quyết định.

**2.3. Hệ thống đề xuất**

Hệ thống quản lý bán siêu xe sử dụng framework Laravel được đề xuất với mục tiêu khắc phục những hạn chế của hệ thống hiện tại, đồng thời mang lại các lợi ích vượt trội cho doanh nghiệp:

1. Tính năng mạnh mẽ và linh hoạt:
   * Laravel cung cấp công cụ quản lý cơ sở dữ liệu hiệu quả, giao diện thiết kế thân thiện và hệ thống bảo mật cao, đảm bảo vận hành mượt mà và an toàn.
   * Hệ thống hỗ trợ tích hợp với các dịch vụ cao cấp như API của các hãng xe, cổng thanh toán, và các dịch vụ vận chuyển siêu xe.
2. Tự động hóa và cá nhân hóa:
   * Hệ thống tự động gửi thông báo lịch bảo trì, nhắc nhở khách hàng về các dịch vụ đã đặt, và cung cấp báo cáo chi tiết về tình trạng kinh doanh.
   * Cung cấp gợi ý cá nhân hóa dựa trên sở thích và nhu cầu của khách hàng, nâng cao trải nghiệm mua sắm.
3. Khả năng mở rộng và tùy chỉnh linh hoạt:
   * Hệ thống dễ dàng mở rộng để tích hợp các tính năng như chương trình khách hàng thân thiết, quản lý sự kiện ra mắt xe, và báo cáo phân tích thị trường.
   * Tùy chỉnh theo nhu cầu của từng doanh nghiệp, đảm bảo đáp ứng các yêu cầu đặc thù của ngành siêu xe.

Mục tiêu của hệ thống:

* Tối ưu hóa quy trình kinh doanh và dịch vụ hậu mãi.
* Đảm bảo độ chính xác và chuyên nghiệp trong mọi hoạt động quản lý.
* Nâng cao trải nghiệm khách hàng, giúp doanh nghiệp xây dựng thương hiệu và phát triển bền vững.

Hệ thống đề xuất sẽ bao gồm các thành phần chính sau:

NHT\_QUAN\_TRI : Danh mục Quản Trị, các chức năng CRUD

NHT\_LOAI\_SAN\_PHAM : Danh mục Loại Sản Phẩm, các chức năng CRUD

NHT\_SAN\_PHAM : Danh mục Sản Phẩm, các chức năng CRUD

NHT\_KHACH\_HANG : Danh mục Khách Hàng, các chức năng CRUD

NHT\_HOA\_DON : Danh mục Hóa Đơn, các chức năng CRUD

NHT\_CT\_HOA\_DON : Danh mục Hóa Đơn, các chức năng CRUD

NHT\_TIN\_TUC : Danh mục Tin Tức, các chức năng CRUD

Hệ thống quản lý bán siêu xe sử dụng Laravel sẽ mang lại một giải pháp toàn diện, mạnh mẽ và linh hoạt để doanh nghiệp có thể quản lý các khía cạnh quan trọng trong quá trình bán hàng. Hệ thống sẽ giúp cải thiện hiệu suất công việc, nâng cao trải nghiệm khách hàng, đồng thời tăng trưởng doanh thu và giảm thiểu các rủi ro trong quản lý kho và đơn hàng.

## Giới hạn của hệ thống

Mặc dù hệ thống quản lý bán hàng siêu xe sử dụng framework Laravel có nhiều tính năng hữu ích và sẽ giúp doanh nghiệp tối ưu hóa quy trình bán hàng, quản lý kho và khách hàng, tuy nhiên, hệ thống vẫn có những giới hạn cần phải lưu ý. Dưới đây là một số giới hạn của hệ thống:

1. Giới Hạn về Tính Năng và Tính Mở Rộng

2. Yêu Cầu về Cơ Sở Hạ Tầng và Tài Nguyên

3. Vấn Đề về Bảo Mật và Dữ Liệu

4. Hạn Chế về Tích Hợp với Các Hệ Thống Khác

5. Giới Hạn về Khả Năng Xử Lý Tình Huống Ngoại Lệ

## Yêu cầu về phần cứng và phần mềm

## Yêu cầu tối thiểu

|  |  |  |
| --- | --- | --- |
|  | **Server** | **Client** |
| Hardware | \* Ram:….  \* HDD >=500MB free  \* CPU: …  \* Connect the Internet. | \* Ram: ….  \* CPU :….  \* Connect the Internet. |
| Software | \* Hệ điều hành: ….  \* SQL Server …  \* NetFramework:… | \* Hệ điều hành: …  \* IE 8.0, Firefox 4.0, Chrome 15….. |

## Yêu cầu đề nghị

|  |  |  |
| --- | --- | --- |
|  | **Server** | **Client** |
| Hardware | \* Ram: >= 2Gb  \* HDD: >= 1GB free  \* CPU: Intel Core 2 Duo  \* Connect the Internet. | \* Ram:….  \* CPU: ….  \*Connect the Internet. |
| Software | \* Hệ điều hành: Windows Server …..  \* SQL Server ….  \* Net Framework:…. | \* Từ Windows XP / Vista / 7 / 8 / 8.1… trở lên.  \* IE 11.0 hoặc Firefox 16.0 trở lên, Chrome 19.0 trở lên. |

# **CHƯƠNG 3:**

# **PHÂN TÍCH YÊU CẦU KHÁCH HÀNG VÀ THIẾT KẾ HỆ THỐNG**

## Người dùng của hệ thống

***Đối tượng người dùng của hệ thống bao gồm***

* Guest (Khách vãng lai)
* Member (Thành viên)
* Admin (Quản trị hệ thống).

## Chức năng của hệ thống

* + 1. **Guest (Chức năng dành cho khách vãng lai)**
* **Xem sản phẩm**

|  |  |
| --- | --- |
| **Mô tả** | Khách hàng có thể xem thông tin chi tiết từng loại SmartPhone như tên SmartPhone, hình ảnh, giá cả,mô tả … |
| **Nhập** | Lựa chọn loại SmartPhone cần xem. |
| **Xử lý** | Tìm kiếm trong CSDL. |
| **Xuất** | Hiển thị thông tin mà khách hàng muốn xem. |

* **Tìm Kiếm Sản Phẩm**

|  |  |
| --- | --- |
| **Mô tả** | Khách hàng có thể search sản phẩm thông qua thanh search |
| **Nhập** | Nhập tên sản phẩm muốn tìm |
| **Xử lý** | Tìm kiếm trong CSDL. |
| **Xuất** | Hiển thị thông tin mà khách hàng muốn xem. |

* …………….
* ………………..
* ……………..
* **Đăng ký thành viên**

|  |  |
| --- | --- |
| **Mô tả** | Khách vãng lai có thể đăng ký làm thành viên. |
| **Nhập** | Đưa vào các thông tin đã nhập trong form để kiểm tra. |
| **Xử lý** | * Kiểm tra username, email đã được đăng ký hay chưa. * Mật khẩu phải gồm ít nhất 6 ký tự và không có ký tự trắng. * Email nhập vào phải đúng định dạng. * ... * Yêu cầu nhập các thông tin bắt buộc. * Thêm mới các thông tin khách hàng, mã hóa mật khẩu vào CSDL nếu đúng. Nếu không yêu cầu nhập lại. |
| **Xuất** | Thông báo khách hàng đăng ký thành công sau khi đã điền đầy đủ thông tin hợp lệ. Dẫn đến trang Đăng Nhập. |

* + 1. ***Member functions*** *(Chức năng dành cho thành viên)*

Thành viên chính thức của website sẽ có các chức năng giống như khách vãng lai như:

* + Xem sản phẩm.
  + Tìm kiếm sản phẩm.
  + Lựa chọn sản phẩm vào giỏ hàng.

***Ngoài ra thành viên chính thức còn có các chức năng sau***:

* **Đăng nhập**

|  |  |
| --- | --- |
| **Mô tả** | Đăng nhập vào website bằng cách nhập Email và password đã đăng ký. |
| **Nhập** | Nhập Email và mật khẩu đã đăng ký. |
| **Xử lý** | Truy cập kiểm tra xem tài khoản có tồn tại hay không. |
| **Xuất** | Hiển thị thông báo đăng nhập thành công khi khách hàng nhập đúng thông tin đã đăng ký và quay về trang chủ. Thông báo sai thông tin đăng nhập nếu tài khoản chưa đăng ký hoặc bị khóa, yêu cầu nhập lại thông tin đăng nhập! |

* **Đăng xuất**

|  |  |
| --- | --- |
| **Mô tả** | Đăng xuất khỏi website. |
| **Nhập** | Nhấn nút thoát để đăng xuất tài khoản. |
| **Xử lý** | * Hủy toàn bộ session của người dùng hiện tại * Đưa trở về chức năng của khách vãng lai. |
| **Xuất** | Hiển thị giao diện như khách vãng lai sau khi đã đăng xuất khỏi hệ thống. |

* **Đặt hàng**

|  |  |
| --- | --- |
| **Mô tả** | Khi nhấn mua một sản phẩm nào đó ở giao diện khách vãng lai thì hệ thống chuyển đến trang đăng nhập.khi đăng nhập rồi mua một sản phẩm nào đó hệ thống sẽ chuyển bạn đến trang hóa đơn lấy dữ liệu từ csdl (readonly) và cột số lượng. khi người dùng nhập xong số lượng muốn mua thì nhấn vào nút mua sản phẩm để tạo hóa đơn và chuyển đến trang xem hóa đơn.ở trang hóa đơn có nút tạo chi tiết hóa đơn khi ấn vào sẽ tạo được bảng chi tiết hóa đơn vào csdl |
| **Nhập** | Nhấn Mua và nhập số lượng mua để tạo hóa đơn, nhấn tạo chi tiết hóa đơn để tạo chi tiết hóa đơn |
| **Xử lý** | Kiểm tra xem đúng thông tin khách hàng đăng ký chưa nếu sai không mua được |
| **Xuất** | Hóa đơn và chi tiêt hóa đơn được lưu vào csdl |

* **………………..**
* **………………**
  + 1. **Admin functions (Chức năng dành cho quản trị viên)**

***Quản trị viên có các chức năng chính như sau:***

* **Đăng nhập quản trị**

|  |  |
| --- | --- |
| **Mô tả** | Quản trị viên đăng nhập vào hệ thống bằng cách nhập username và password vào form đăng nhập. |
| **Nhập** | Nhập tài khoản và mật khẩu của quản trị viên. |
| **Xử lý** | Kiểm tra thông tin nhập vào có hợp lệ hay không. |
| **Xuất** | * Chuyển đến trang quản trị nếu đăng nhập thành công * Quay lại trang đăng nhập nếu nhập sai. |

* **Đăng xuất quản trị**

|  |  |
| --- | --- |
| **Mô tả** | Thoát khỏi tài khoản quản trị viên. |
| **Nhập** | Click vào nút Đăng Xuất từ giao diện quản trị. |
| **Xử lý** | Hủy session quản trị viên đã đăng nhập. |
| **Xuất** | Quay lại trang chủ. |

* **Quản lý tài khoản khách hàng**

|  |  |
| --- | --- |
| **Mô tả** | Quản trị viên có quyền xem thông tin hoặc khóa/kích hoạt, xóa các tài khoản của thành viên. |
| **Nhập** | Chọn lệnh tương ứng của tài khoản cần thao tác. |
| **Xử lý** | * Với việc xem thông tin tài khoản: * Hiển thị thông tin của khách hàng. * Đối với việc xóa tài khoản: * Thay đổi trạng thái quyền đăng nhập website. * Khóa/kích hoạt tài khoản: * Admin có quyền khóa/kích hoạt tài khoản của khách hàng. |
| **Xuất** | Hiện thông tin đã thay đổi. |

* **Quản lý sản phẩm**

|  |  |
| --- | --- |
| **Mô tả** | Quản trị viên có quyền xem thông tin,Thêm,Sửa, xóa các Sản Phẩm |
| **Nhập** | Chọn lệnh tương ứng của Sản Phẩm cần thao tác. |
| **Xử lý** | * Với việc xem thông tin Sản Phẩm: * Hiển thị thông tin của Sản Phẩm. * Đối với việc xóa Sản Phẩm: * Xóa dữu liệu của sản phẩm * Sửa Thông tin sản phẩm: * Thay đổi thông tin của sản phẩm |
| **Xuất** | Hiện thông tin đã thay đổi. |

* **Quản lý Loại sản phẩm**

|  |  |
| --- | --- |
| **Mô tả** | Quản trị viên có quyền xem thông tin,Thêm,Sửa, xóa các Loại Sản Phẩm |
| **Nhập** | Chọn lệnh tương ứng của Loại Sản Phẩm cần thao tác. |
| **Xử lý** | * Với việc xem thông tin Loại Sản Phẩm: * Hiển thị thông tin của Loại Sản Phẩm. * Đối với việc xóa Loại Sản Phẩm: * Xóa dữu liệu của Loại Sản Phẩm * Sửa Thông tin sản phẩm: * Thay đổi thông tin của Loại Sản Phẩm / trạng thái |
| **Xuất** | Hiện thông tin đã thay đổi. |

* **Quản lý hóa đơn**

|  |  |
| --- | --- |
| **Mô tả** | Quản trị viên có quyền xem thông tin,Thêm,Sửa, xóa các Hóa Đơn |
| **Nhập** | Chọn lệnh tương ứng của Hóa Đơn cần thao tác. |
| **Xử lý** | * Với việc xem thông tin Hóa Đơn Khi Mua một sản phẩm: * Hiển thị thông tin của hóa đơn đó. * Đối với việc xóa hóa đơn : * Xóa dữu liệu của hóa đơn đó * Sửa Thông tin hóa đơn: * Thay đổi trạng thái của hóa đơn |
| **Xuất** | Hiện thông tin đã thay đổi. |

* **Quản Lý Chi Tiết Hóa Đơn**

|  |  |
| --- | --- |
| **Mô tả** | Quản trị viên có quyền xem thông tin,Thêm,Sửa, xóa các Chi Tiết Hóa Đơn |
| **Nhập** | Chọn lệnh tương ứng của Chi Tiết Hóa Đơn cần thao tác. |
| **Xử lý** | * Với việc xem thông tin Chi Tiết Hóa Đơn Khi đã có hóa đơn: * Hiển thị thông tin của chi tiết hóa đơn đó. * Đối với việc xóa chi tiết hóa đơn : * Xóa dữu liệu của chi tiết hóa đơn đó * Sửa Thông tin hóa đơn: * Thay đổi trạng thái của chi tiết hóa đơn |
| **Xuất** | Hiện thông tin đã thay đổi. |

* **Quản Lý Tin Tức**

|  |  |
| --- | --- |
| **Mô tả** | Quản trị viên có quyền xem thông tin,Thêm,Sửa, xóa các Tin Tức |
| **Nhập** | Chọn lệnh tương ứng của Tin Tức cần thao tác. |
| **Xử lý** | * Với việc xem thông tin Tin Tức: * Hiển thị thông tin của chi tiết Tin Tức với MaTT đó. * Đối với việc xóa Tin Tức : * Xóa dữu liệu của Tin Tức * Sửa Thông tin Tin Tức: * Thay đổi dữ diệu trong csdl |
| **Xuất** | Hiện thông tin đã thay đổi. |

## Systeme Designs (Thiết kế hệ thống)

## Entity Relationship Diagram (Mô hình quan hệ thực thể)

Gồm có :

NHT\_QUAN\_TRI;

NHT\_LOAI\_SAN\_PHAM;

NHT\_SAN\_PHAM;

NHT\_KHACH\_HANG;

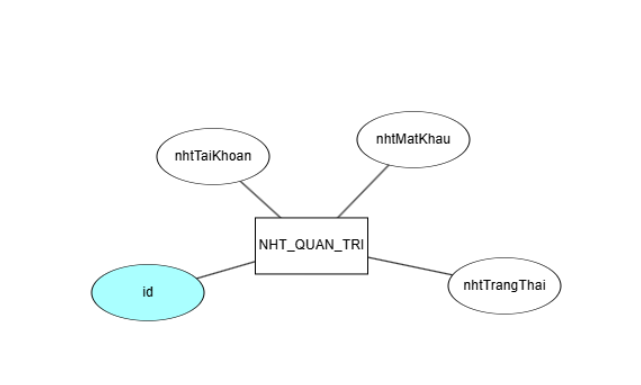
NHT\_HOA\_DON;

NHT\_CT\_HOA\_HOA;

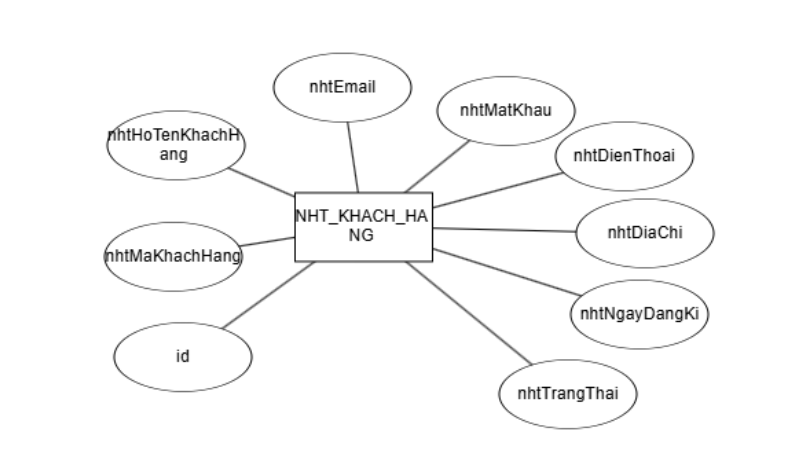
NHT\_TIN\_TUC

***Mô tả chi tiết các thuộc tính của các thực thể***:

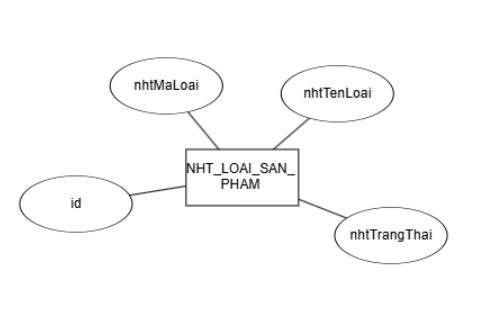
1. Thực thể quản trị (**NHT\_QUAN\_TRI**):   
   Lưu trữ thông tin người quản trị viên hệ thống.



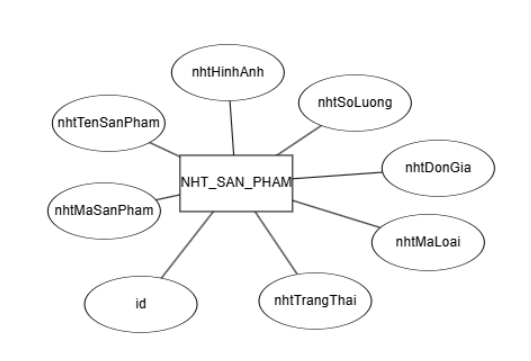
1. Thực thể khách hàng (**NHT\_KHACH\_HANG**): Lưu trữ thông tin khách hàng khi đăng ký, mua hàng tại hệ thống.



1. Thực thể loại sản phẩm (**NHT\_LOAI\_SAN\_PHAM**)



1. Thực thể sản phẩm (**NHT\_SAN\_PHAM**)



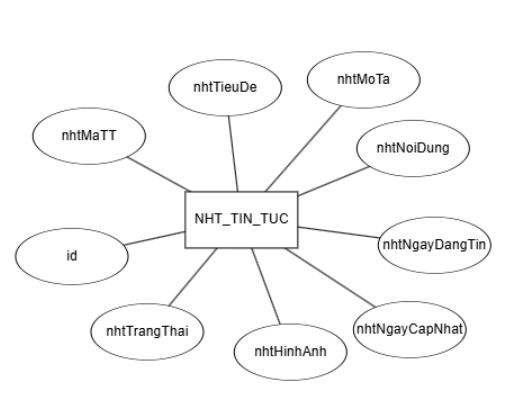
1. Thực thể hóa đơn (**NHT\_HOA\_DON**)



1. Thực thể chi tiết hóa đơn (**NHT\_CT\_HOA\_DON**)



1. Thực thể tin tức (**NHT\_TIN\_TUC**)



## Database Design (Thiết kế cơ sở dữ liệu)

Thiết kế có sở dữ liệu trên hệ quản trị dữ liệu MS SQL (SQL Server)

1. Bảng **NHT\_QUAN\_TRI**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu DataType** | **Null** | **Ràng buộc**  **Constraint** | **Mô tả Content** |
| **id** | bigint | No | Primary Key;AUTO\_INCREMENT | Id tự động tăng |
| nhtTaiKhoan | Varchar(255) | No | Unique | Tài khoản đăng nhập của quản trị |
| nhtMatKhau | Varchar(255) | No |  | Mật khẩu đăng nhập của quản trị. |
| nhtTrangThai | Tinyint | No |  | Trạng thái:  1- Đang hoạt động 0- Đang bị khóa |

1. Bảng **NHT\_KHACH\_HANG**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu DataType** | **Null** | **Ràng buộc**  **Constraint** | **Mô tả Content** |
| **id** | bigint | No | Primary Key; Auto\_increment | Tự động tăng |
| nhtMaKhachHang | Varchar(255) | No | Unique | Mã khách hàng |
| nhtHoTenKhachHang | Varchar(255) | Yes |  | Tên khách hàng |
| nhtEmail | Varchar(255) | No | Unique | Tài khoản đăng nhập |
| nhtMatKhau | Varchar(255) | Yes |  | Mật khẩu |
| nhtDienThoai | Varchar(255) | Yes | Unique | Địa chỉ |
| nhtDiaChi | Varchar(255) | Yes |  | Điện thoại |
| nhtNgayDangKy | date | No |  | Hộp thư điện tử |
| nhtTrangThai | tinyint | Yes |  | Trạng thái:  0- Đang hoạt động  1-Đang Tạm Khóa 2- Đang bị khóa |

1. Bảng **NHT\_LOAI\_SAN\_PHAM**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu DataType** | **Null** | **Ràng buộc**  **Constraint** | **Mô tả Content** |
| **id** | bigint | No | Primary Key;AUTO\_INCREMENT | Id tự động tăng |
| nhtMaLoai | Varchar(255) | No | Unique | Mã Loại Sản Phẩm |
| nhtTenLoai | Varchar(255) | No |  | Tên Loại Sane Phẩm |
| nhtTrangThai | Tinyint | No |  | Trạng thái:  1- Đang hoạt động 0- Đang bị khóa |

1. Bảng **NHT\_SAN\_PHAM**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu DataType** | **Null** | **Ràng buộc**  **Constraint** | **Mô tả Content** |
| **id** | bigint | No | Primary Key;AUTO\_INCREMENT | Id tự động tăng |
| nhtMaSanPham | Varchar(255) | No | Unique | Mã Sản Phẩm |
| nhtTenSanPham | Varchar(255) | No |  | Tên Sản Phẩm |
| nhtHinhAnh | Varchar(255) | yes |  | File ảnh |
| nhtSoLuong | int | No |  | Số lượng |
| nhtDonGia | Double(8,2) | No |  | Đơn giá |
| nhtMaLoai | bigint | No | REFERENCES nht\_LOAI\_SAN\_PHAM(id) | Mã loại lấy từ id bảng nht\_loai\_san\_pham |
| nhtMoTa |  | Yes |  | Mô tả |
| nhtTrangThai | Tinyint | No |  | Trạng thái:  1- khóa 0- hiển thị |

1. Bảng **NHT\_HOA\_DON**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu DataType** | **Null** | **Ràng buộc**  **Constraint** | **Mô tả Content** |
| **id** | bigint | No | Primary Key;AUTO\_INCREMENT | Id tự động tăng |
| nhtMaHoaDon | Varchar(255) | No | Unique | Mã Hóa Đơn |
| nhtMaKhachHang | bigint | No | REFERENCES nht\_KHACH\_HANG(id) | lấy từ id bảng nht\_KHACH\_HANG |
| nhtNgayHoaDon | date | No |  | Ngày Tạo Hóa Đơn |
| nhtNgayNhan | date | Yes |  | Ngày Nhận Hóa Đơn |
| nhtHoTenKhachHang | Varchar(255) | No |  | Họ Tên Khách Hàng |
| nhtEmail | Varchar(255) | No |  | Email |
| nhtDienThoai | Varchar(255) | No |  | Điện thoại |
| nhtDiaChi | Varchar(255) | No |  | Địa chỉ |
| nhtTongGiaTri | Double(15,3) | No |  | Tổng giá trị |
| nhtTrangThai | Tinyint | No |  | Trạng thái:  2-đã hoàn thành 1- đang sử lý 0- chờ sử lý |

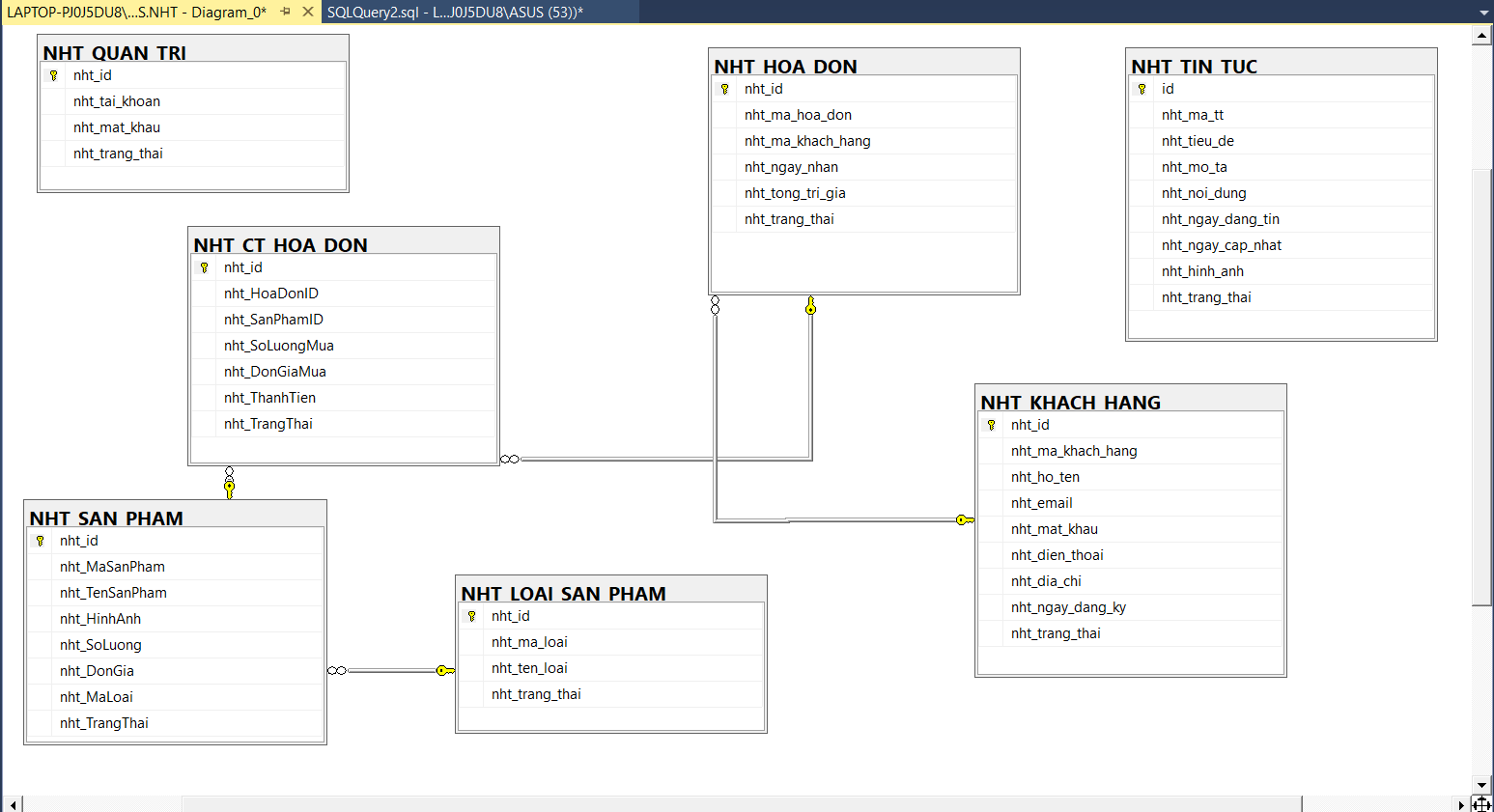
1. Bảng **nht\_CT\_HOA\_DON**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu DataType** | **Null** | **Ràng buộc**  **Constraint** | **Mô tả Content** |
| **id** | bigint | No | Primary Key;AUTO\_INCREMENT | Id tự động tăng |
| nhtHoaDonID | bigint | No | REFERENCES nht\_HOA\_DON(id) | lấy từ id bảng nht\_HOA\_DON |
| nhtSanPhamID | bigint | No | REFERENCES nht\_SAN\_PHAM(id) | lấy từ id bảng nht\_san\_pham |
| nhtSoLuongMua | int | No |  | Số lượng mua |
| nhtDonGiaMua | Double(8,2) | No |  | Đơn giá mua |
| nhtThanhTien | double | No |  | Thành tiền |
| NHTTrangThai | Tinyint | No |  | Trạng thái:  2-khóa 1- trả lại 0- hoàn thành |

1. Bảng **NHT\_TIN\_TUC**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tên cột**  **Field** | **Kiểu dữ liệu DataType** | **Null** | **Ràng buộc**  **Constraint** | **Mô tả Content** |
| **id** | bigint | No | Primary Key;AUTO\_INCREMENT | Id tự động tăng |
| nhtMaTT | varchar(255) | No | Unique | Mã tin tức |
| nhtTieuDe | varchar(255) | No |  | Tiêu đề |
| nhtMoTa | varchar(255) | Yes |  | Mô tả |
| nhtNoiDung | varchar(500) | Yes |  | Nội dung |
| nhtNgayDangTin | date | No |  | Ngày đăng tin |
| nhtNgayCapNhap | date | No |  | Ngày cập nhập |
| nhtHinhAnh | varchar(255) | Yes |  | Hình ảnh |
| nhtTrangThai | Tinyint | No |  | Trạng thái:  1- ẩn 0- hiển thị |

## Table RelationShip Diagram (Mô hình quan hệ bảng)



## SiteMap (Sơ đồ cấu trúc site)

## Site map (Cấu trúc trang): Dành cho khách vãng lai

Trang chủ

Sản phẩm

Nhãn hiệu

Tìm kiếm

Giới thiệu

Liên hệ

Hướng dẫn

Đăng nhập

Đăng ký

Giỏ hàng

## Sitemap (Cấu trúc trang): Dành cho thành viên

Trang chủ

Sản phẩm

Nhãn hiệu

Tìm kiếm

Giới thiệu

Liên hệ

Hướng dẫn

Đăng xuất

Thông tin cá nhân

Giỏ hàng

Đặt mua

Đổi mật khẩu

Sửa thông tin

Bình luận

## Sitemap (Cấu trúc trang): Dành cho quản trị viên

Trang chủ Admin

Quản lý Sản phẩm

Quản lý Tài Khoản

Quản lý Loại

Quản lý user

Quản lý Hóa đơn

Quản lý Liên hệ

Quản lý Phản hồi

Quản lý Tin tức

Quản lý Thành viên

Quản lý Quản trị

## Algorithms (Giải thuật)

<< Bổ dung chỉnh sửa cho phù hợp với đề tài>>

## Đăng ký



|  |  |
| --- | --- |
| * **Kiểm tra thông tin nhập**:   Kiểm tra các thông tin bắt buộc nhập theo ràng buộc khi thiết kế cơ sở dữ liệu:   * + Tài khoản, Mật khẩu, Email, Điện thoại,.. | * **Kiểm tra sự tồn tại của tài khoản:**   Tài khoản là duy nhất đối với mỗi khách hàng. Nếu đã có người dùng rồi thì không cho phép đăng ký. |

## Đăng nhập



## Đăng xuất



## Đổi mật khẩu



## Thay đổi thông tin cá nhân



## Thêm sản phẩm



## Xóa sản phẩm



## Đặt mua



## Tìm kiếm



# **CHƯƠNG 4: XÂY DỰNG HỆ THỐNG**

## Xây dựng phần mềm

## Một số mã nguồn chương trình

## <!-- JavaScript Libraries -->

## <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

## <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.8/dist/umd/popper.min.js"

## integrity="sha384-I7E8VVD/ismYTF4hNIPjVp/Zjvgyol6VFvRkX/vR+Vc4jQkC+hVqc2pM8ODewa9r" crossorigin="anonymous"></script>

## <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.min.js"

## integrity="sha384-0pUGZvbkm6XF6gxjEnlmuGrJXVbNuzT9qBBavbLwCsOGabYfZo0T0to5eqruptLy" crossorigin="anonymous"></script>

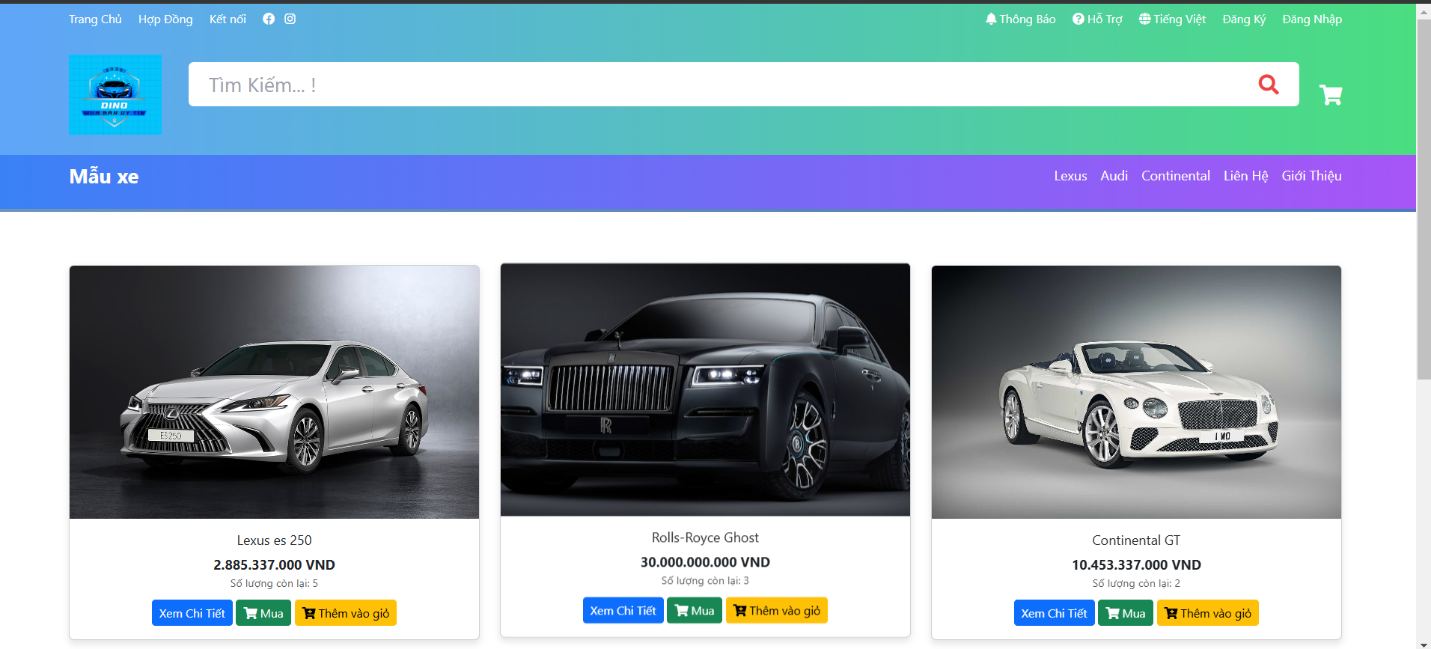
    <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css" rel="stylesheet" />

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"

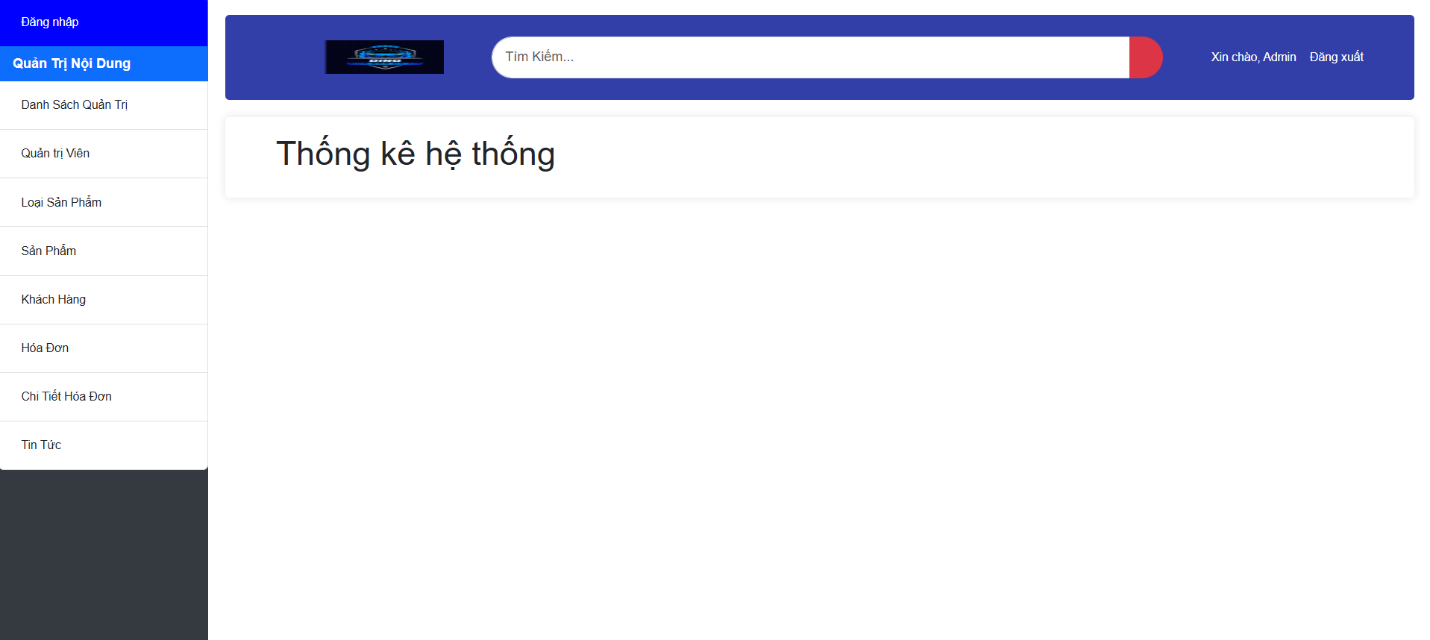
        integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">

## Một số giao diện người dùng và chức năng cụ thể

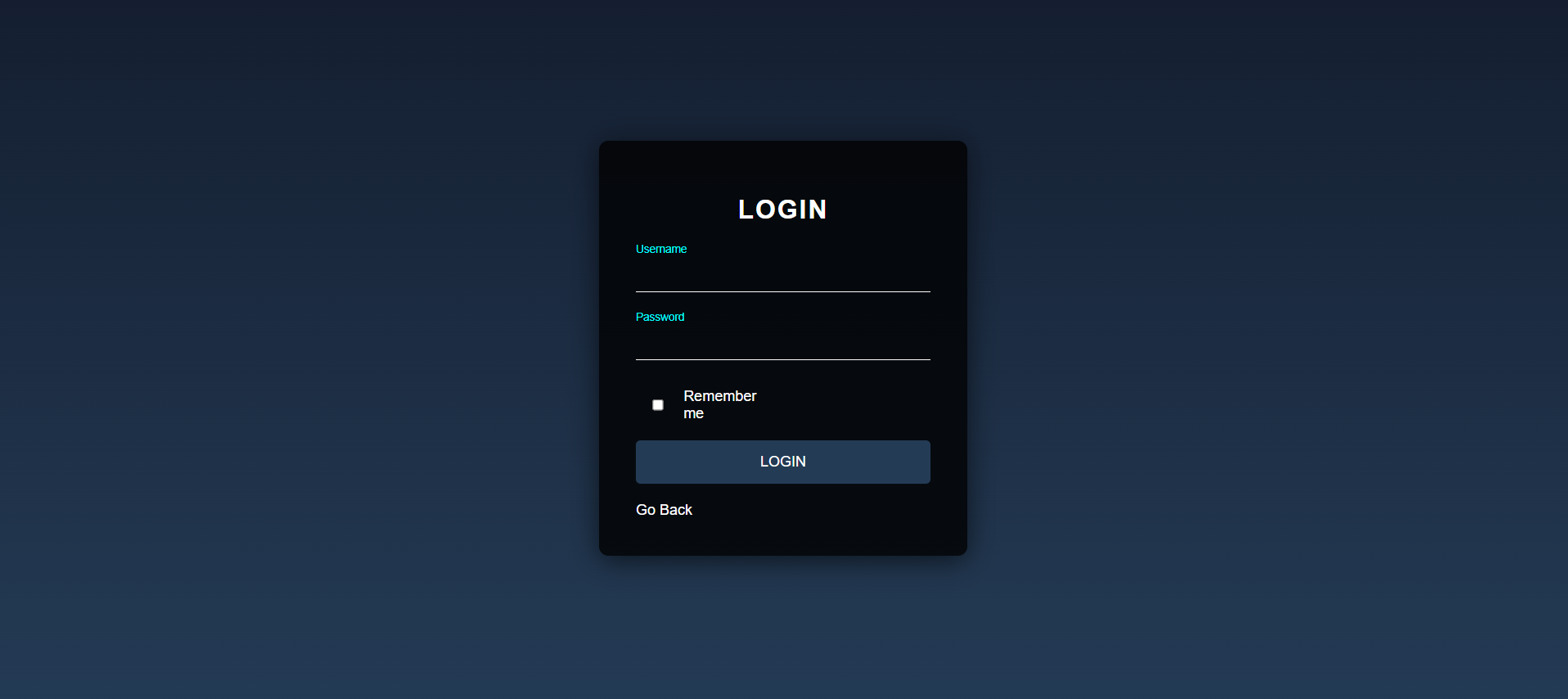
## Giao diện trang chủ



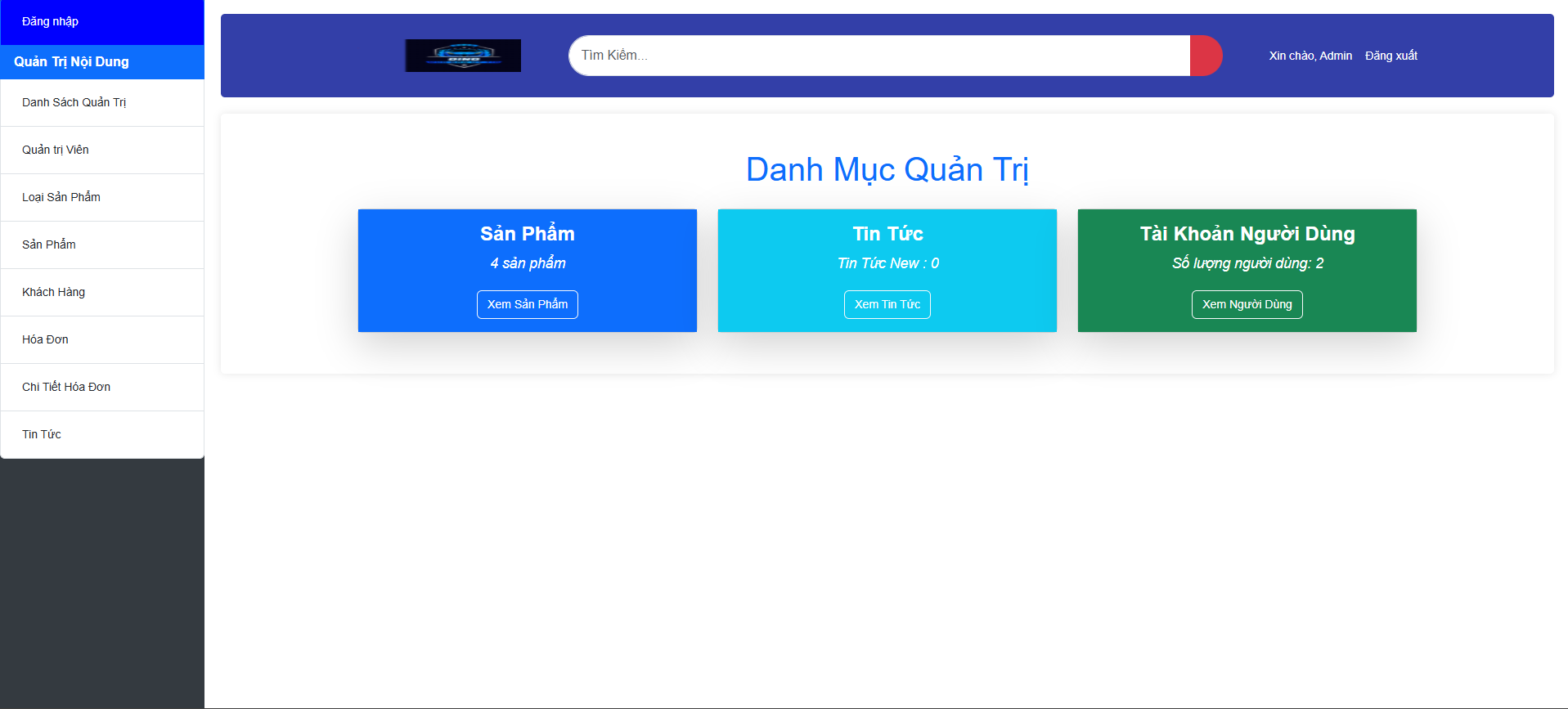
## Giao diện trang quản trị



## Giao diện đăng nhập Admins

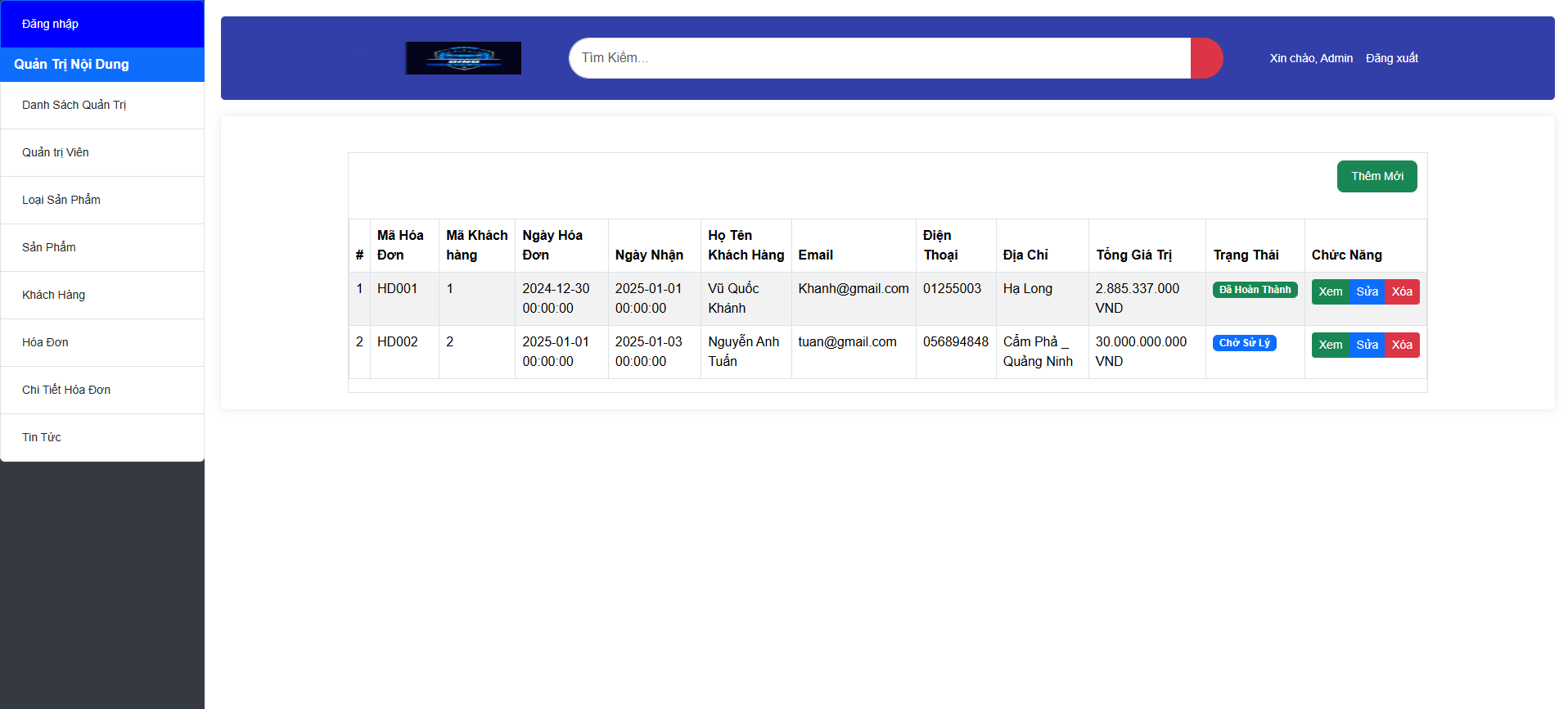


## Giao diện Danh Sách Quản Trị

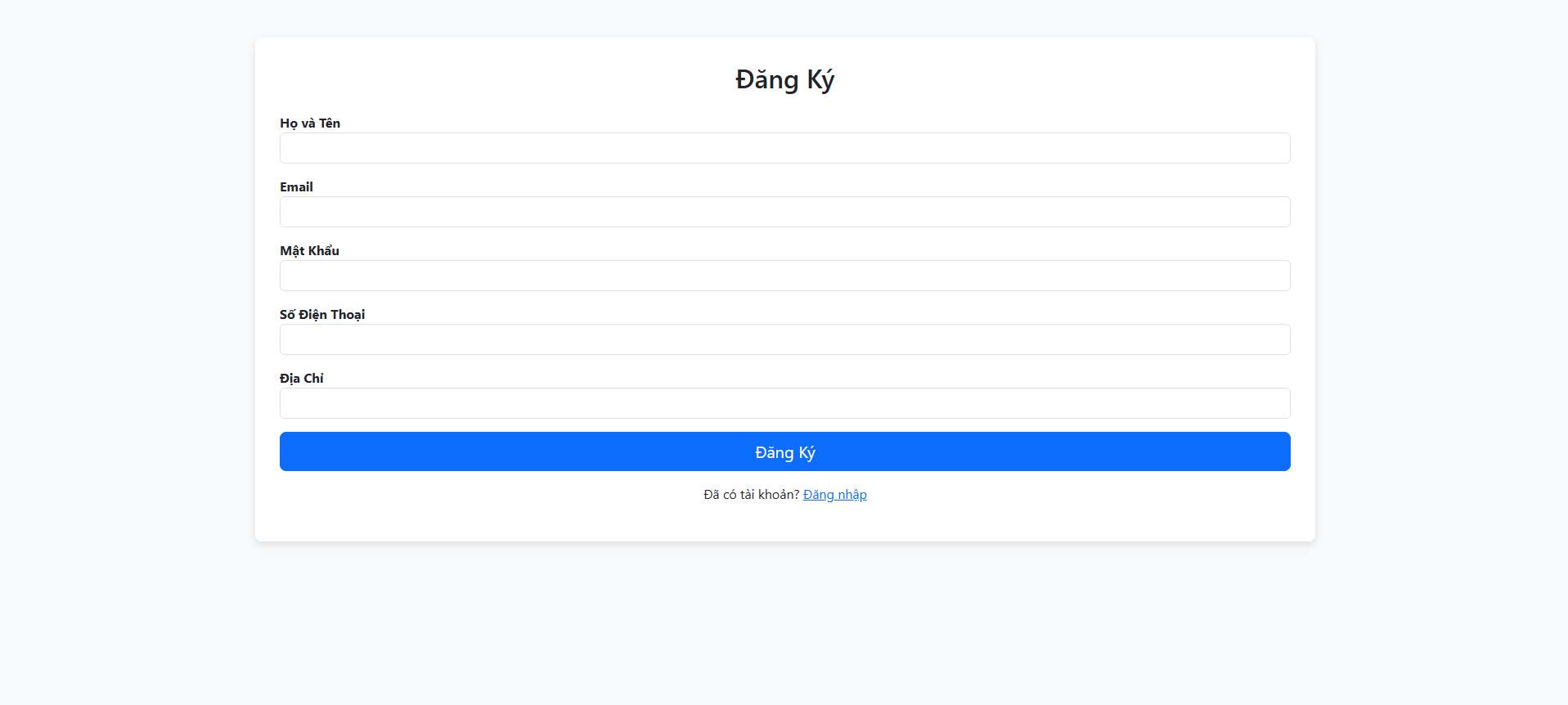


## Giao diện Loại Sản Phẩm , Sản Phẩm

## Giao diện hoá đơn



## Giao diện đăng ký tài khoản

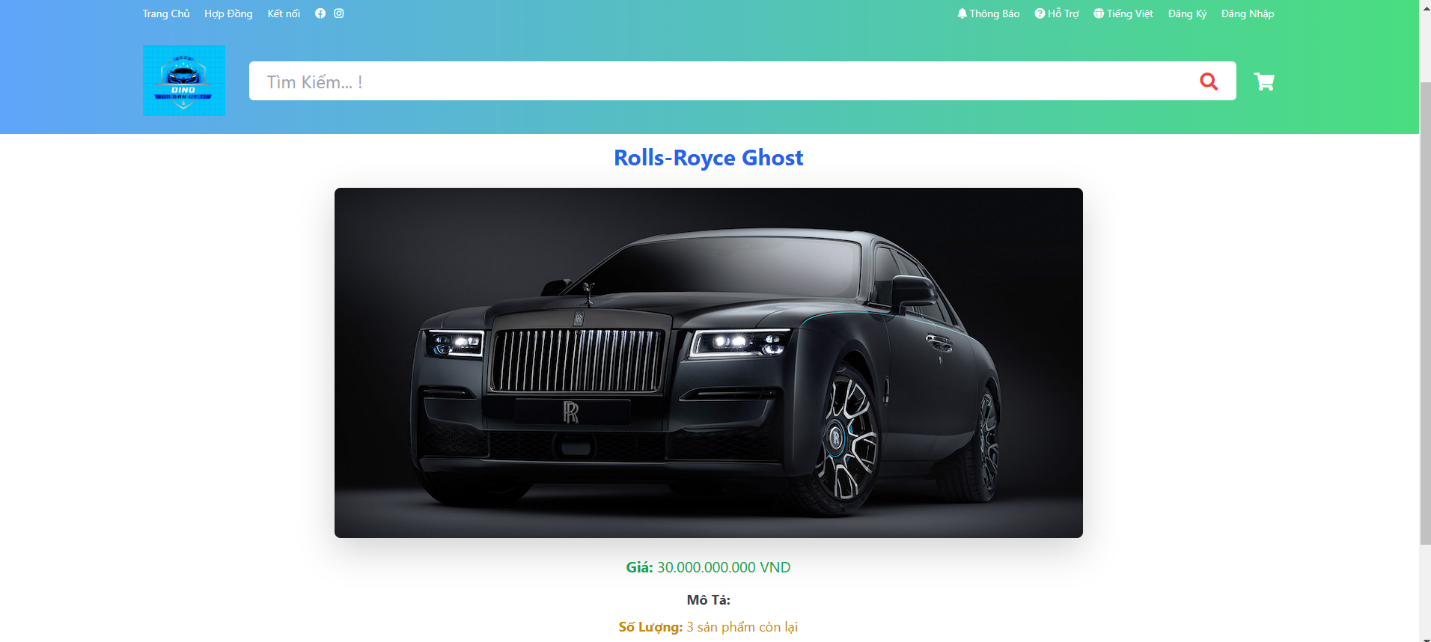


## Giao diện đăng nhập

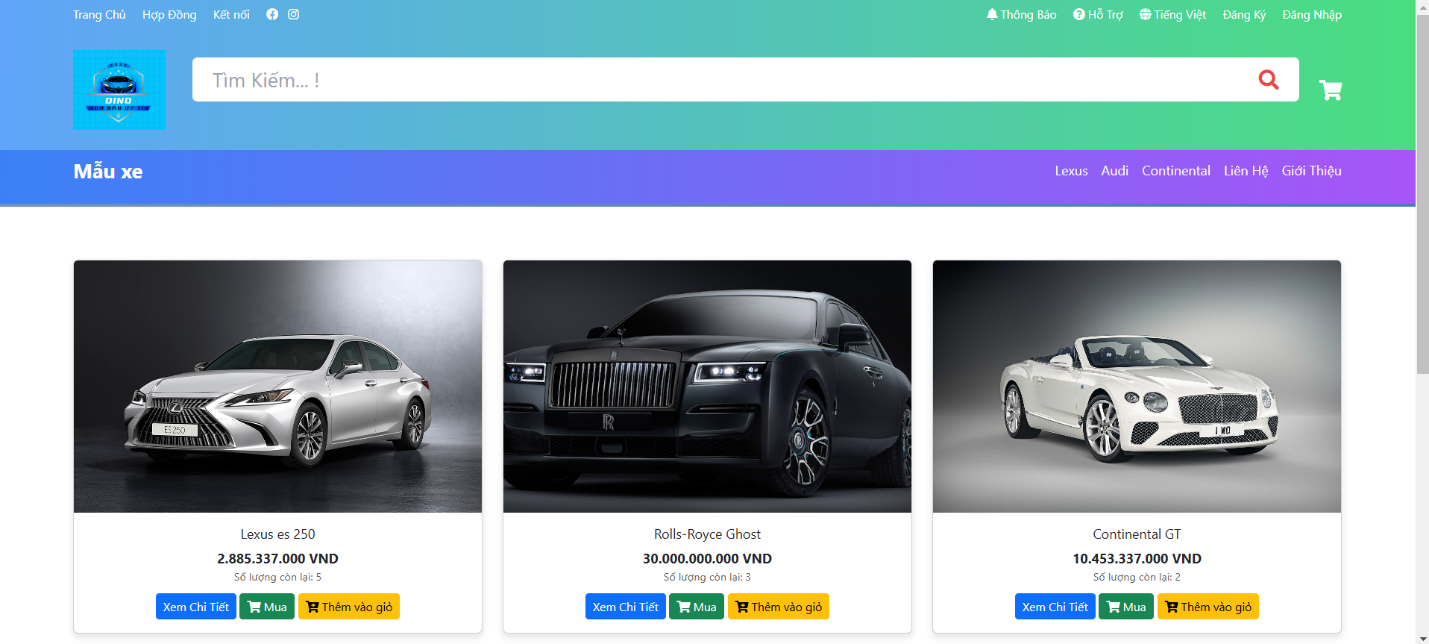
A screenshot of a computer

Description automatically generated

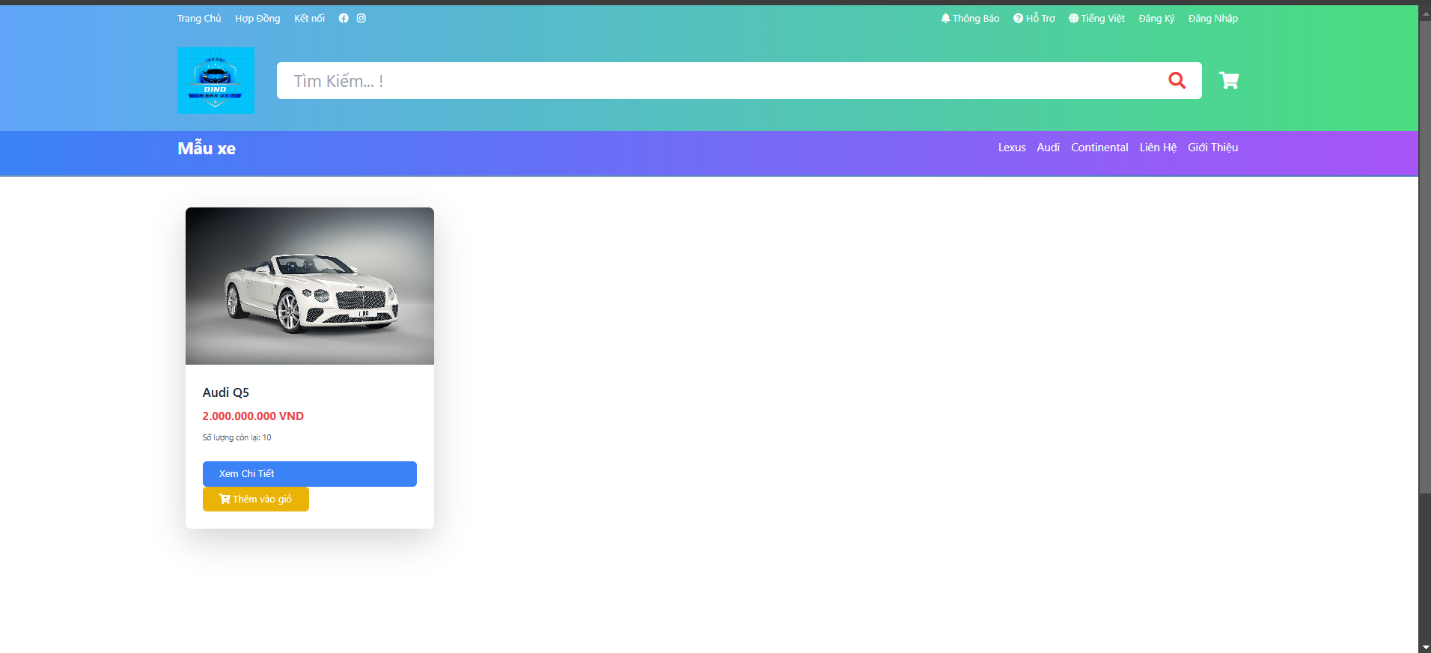
## Giao diện xem chi tiết sản phẩm



## Giao diện khi đăng nhập



## Giao diện khi tìm kiếm sản phẩm



# **KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

## Kết luận

Quá trình làm đồ án chuyên đề với đề tài "Bán siêu xe" đã giúp em có cơ hội để trau dồi các kiến thức được học từ thầy cô trên giảng đường, đồng thời cũng tiếp thu và tìm hiểu được thêm nhiều kiến thức, kinh nghiệm mới, cụ thể là:

* + - Nắm được các quy trình nghiệp vụ trong phân tích và thiết kế một sản phẩm phần mềm.
    - Hiểu biết được một số phương pháp phân tích thiết kế hệ thống, từ đó đã áp dụng các hiểu biết của mình để phân tích thiết kế hệ thống bài toán theo mô hình ba lớp.
    - Tiếp thu được các kinh nghiệm thực tế khi thực hành xây dựng một sản phẩm cụ thể, áp dụng được các kiến thức đã học vào thực tiễn.
    - Củng cố các kiến thức về ngôn ngữ lập trình cũng như các thư viện – framework,…

**Ưu điểm:**

Hệ thống cơ bản đã hoàn thiện đầy đủ các tính năng đã được thiết kế và đưa vào sử dụng phục vụ các yêu cầu của người dùng. Quan trọng hơn, hệ thống đảm bảo duy trì được sự ổn định, tính bảo mật và tính tiện dụng cho người sử dụng. Các tính năng đã hoàn thành:

* + - Hoàn Thành CRUD của csdl
    - Hoàn thành giao diện Admins
    - Hoàn thành giao diện khách vãng lai
    - Hoàn thành giao diện người dùng
    - Có giới thiệu , liên hệ …

**Nhược điểm:**

Trong những phiên bản tiếp theo, ứng dụng có thể được mở rộng để đáp ứng nhiều hơn những yêu cầu từ người sử dụng như:

* + - Giao diện của hệ thống do bản thân tự thiết kế nên vẫn chưa được hoàn thiện và đẹp mắt nhất.
    - Nhiều chức năng hoạt động chưa được mượt mà tạo hiệu ứng sử dụng không tốt với người dùng.
    - Trong thời gian tới em sẽ tiếp tục hoàn thiện các chức năng hiện thời và bổ sung thêm các chức năng mới nhằm mang lại sự tiện dụng nhất cho người dùng.

**Hướng phát triển:**

Trong những phiên bản tiếp theo, ứng dụng có thể được mở rộng để đáp ứng nhiều hơn những yêu cầu từ người sử dụng như:

* + - Cải thiện, bảo trì những lỗi phát sinh ở các tính năng hiện tại.
    - Cải thiện giao diện thân thiện hơn với người dùng.
    - Phát triển thêm tính năng …..

## Task Sheet (bảng kế hoạch công việc)

|  |  |  |  |
| --- | --- | --- | --- |
| **Giai đoạn** | **Từ ngày … đến….** | **Công việc** | **Tỉ lệ hoàn thành** |
| Chương 1:  (100%) |  |  | 100% |
| Chương 2:  (100%) |  |  |  |
| Chương 3:  (100%) |  |  |  |
| Chương 4:  (100%) |  |  |  |
| Chương 5:  (100%) |  |  |  |

## Checklist (bảng kiểm tra chức năng)

## Kiểm tra dữ liệu nhập

|  |  |  |
| --- | --- | --- |
| **TT** | **Các thành phần** | **Tình trạng** |
| 1 | Các trường trên form không được trống theo yêu cầu | OK |
| 2 | Dữ liệu nhập phù hợp (Chuỗi, số, ngày, …) | OK |
| 3 | Tài khoản đăng nhập, email là duy nhất | OK |
| 4 | Các ràng buộc (thêm, sửa, xóa) | OK |
| 5 | Kiểm tra dữ liệu nhập trên form đầy đủ | OK |
| 6 | Xử lý dữ liệu trên giỏ hàng đầy đủ | OK |

## Kiểm tra các liên kết

|  |  |  |
| --- | --- | --- |
| **TT** | **Các liên kết** | **Tình trạng** |
| 1 | Tất cả các trang đều hoạt động | OK |
| 2 | Tất cả các liên kết đều hoạt động đúng | OK |
| 3 | Các trang đều không có lỗi chính tả | OK |
| 4 | Các mẫu sử dụng trên mỗi trang đều phù hợp | OK |
| 5 | Các thành phần trong trang đều hoạt động | OK |
| 6 | Ứng dụng tương thích với nhiều trình duyệt phổ biến | OK |
| 7 | Dữ liệu sau đều có nhắc nhở, thông báo | OK |
| 8 | Các trang đều có liên kết đến trang khác | OK |

# **TÀI LIỆU THAM KHẢO**

#### **Tài liệu:**

#### **Website:**

1. <https://www.w3schools.com/>
2. <https://support.microsoft.com/vi-vn/topic/31846479-c656-f2a4-bc24-c9803a97e62c>
3. <https://topdev.vn/blog/sql-server-la-gi/>