|  |
| --- |
| TRƯỜNG CAO ĐẲNG CÔNG THƯƠNG TPHCM  **KHOA CÔNG NGHỆ THÔNG TIN**                **BÁO CÁO THỰC TẬP**  **TỐT NGHIỆP**    GV hướng dẫn: Chu Thị Mai  Sinh viên thực hiện: Nguyễn Anh Quân  Mã số sinh viên: 2122110006  Lớp: CCQ2211A Khoá: 46  **TPHCM – Tháng 04 Năm 2025** |

# LỜI CẢM ƠN

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

**NHẬN XÉT KẾT QUẢ THỰC TẬP**

*(Dành cho đơn vị nhận sinh viên thực tập)*

Họ tên sinh viên thực tập:

Ngày tháng năm sinh:

Cán bộ hướng dẫn thực tập:

Bộ phận:

Sau thời gian sinh viên …………………………………. thực tập tại đơn vị, chúng tôi có nhận xét sau:

1. **Về ý thức chấp hành nội quy, quy định của công ty:**

1. **Về đạo đức, tác phong:**

1. **Về năng lực chuyên môn:**

1. **Kết luận:**

Nhận xét:

Điểm:

*…………….., Ngày…… tháng…… năm…….*

|  |  |
| --- | --- |
| **Cán bộ hướng dẫn** | **Xác nhận của đơn vị** |

**NHẬN XÉT KẾT QUẢ THỰC TẬP**

*(Dành cho Giảng viên hướng dẫn sinh viên thực tập)*

Họ tên sinh viên thực tập:

Ngày tháng năm sinh:

Giảng viên hướng dẫn thực tập:

Sau thời gian sinh viên …………………………………. thực tập tốt nghiệp, tôi có nhận xét sau:

1. **Về ý thức, thái độ của sinh viên:**

1. **Về đạo đức, tác phong:**

1. **Về năng lực chuyên môn:**

1. **Kết luận:**

Nhận xét:

Điểm:

*…………….., Ngày…… tháng…… năm…….*

**Giảng viên hướng dẫn**

[LỜI CẢM ƠN ii](#_Toc201729844)

[1. Lý do chọn đề tài: 3](#_Toc201729845)

[2. Mục tiêu: 3](#_Toc201729846)

[3. Đối tượng nghiên cứu: 4](#_Toc201729847)

[4. Phạm vi nghiên cứu: 4](#_Toc201729848)

[5. Phương pháp nghiên cứu: Tiến trình nghiên cứu gồm: 5](#_Toc201729849)

[CHƯƠNG 1. TỔNG QUAN VỀ CƠ SỞ THỰC TẬP 6](#_Toc201729850)

[1.1. Giới thiệu về doanh nghiệp 6](#_Toc201729851)

[1.2. Cơ cấu tổ chức và bộ phận thực tập 6](#_Toc201729852)

[1.2.1. Cơ cấu tổ chức: 6](#_Toc201729853)

[1.2.2. Bộ phận thực tập: 6](#_Toc201729854)

[1.3. Nội quy và quy trình làm việc tại công ty 7](#_Toc201729855)

[CHƯƠNG 2. LÝ THUYẾT ÁP DỤNG THEO LĨNH VỰC THỰC TẬP 8](#_Toc201729856)

[2.1. Tổng quan về công nghệ sử dụng: 8](#_Toc201729857)

[2.2. Kiến trúc hệ thống phần mềm: 9](#_Toc201729858)

[2.3. Laravel 9](#_Toc201729859)

[2.4. MySQL 10](#_Toc201729860)

[2.5. Composer 11](#_Toc201729861)

[2.6. Node.js và npm 11](#_Toc201729862)

[2.7. Blade Template 12](#_Toc201729863)

[2.8. Git: Quản lý source code 13](#_Toc201729864)

[CHƯƠNG 3. NỘI DUNG THỰC TẬP 14](#_Toc201729865)

[3.1. Mục tiêu thực tập 14](#_Toc201729866)

[3.2. Công việc được giao 14](#_Toc201729867)

[3.3. Kết quả làm việc 15](#_Toc201729868)

[3.4. Nhật ký thực tập 15](#_Toc201729869)

[3.5. Phân tích thiết kế hệ thống & triển khai 16](#_Toc201729870)

[3.5.1. Phân tích thiết kế hệ thống: 16](#_Toc201729871)

[3.5.2. Triển khai: 27](#_Toc201729872)

[3.6. Hình ảnh minh họa 41](#_Toc201729873)

[3.7. Biểu đồ thể hiện tiến độ công việc theo tuần 42](#_Toc201729874)

[CHƯƠNG 4. KẾT LUẬN VÀ KIẾN NGHỊ 43](#_Toc201729875)

[4.1. Kết quả đạt được 43](#_Toc201729876)

[4.2. Những khó khăn gặp phải và cách khắc phục 43](#_Toc201729877)

[4.3. Bài học kinh nghiệm 43](#_Toc201729878)

[4.4. Đề xuất và hướng phát triển 43](#_Toc201729879)

ĐỀ CƯƠNG THỰC TẬP

ĐỀ TÀI: *XÂY DỰNG WEBSITE BÁN HÀNG THIẾT BỊ MÁY ẢNH BẰNG LARAVELL*

## Lý do chọn đề tài:

* Trong bối cảnh hiện nay, việc số hóa các hoạt động kinh doanh và quản lý dữ liệu đã trở thành một xu hướng tất yếu, đặc biệt là trong lĩnh vực thương mại điện tử. Các doanh nghiệp, từ quy mô nhỏ đến lớn, đều có nhu cầu xây dựng một hệ thống quản lý bán hàng và dữ liệu sản phẩm một cách bài bản, hiệu quả và dễ vận hành.
* Thiết bị máy ảnh là mặt hàng có giá trị cao, đòi hỏi thông tin sản phẩm minh bạch, hình ảnh trực quan và khả năng tương tác tốt với người mua. Do đó, một hệ thống website bán hàng chuyên nghiệp là rất cần thiết.
* Đề tài “Xây dựng website bán hàng thiết bị máy ảnh bằng Laravel” được lựa chọn dựa trên thực tiễn đó. Ngoài mục đích phục vụ cho hoạt động kinh doanh, hệ thống còn giúp quản lý người dùng, danh mục sản phẩm, thông tin đặt hàng và các thao tác quản trị cơ bản một cách tập trung.
* Laravel được lựa chọn làm framework phát triển vì:

Cấu trúc rõ ràng, hỗ trợ MVC (Model – View – Controller)

Hệ thống routing, auth, migration tích hợp sẵn, dễ tùy chỉnh

Cộng đồng lớn và tài liệu phong phú

Dễ mở rộng cho các tính năng trong tương lai

* Việc chọn đề tài này cho phép sinh viên áp dụng tổng hợp các kỹ năng:

Vận dụng toàn diện kiến thức đã học từ frontend đến backend.

Trải nghiệm thực tế quy trình xây dựng phần mềm từ phân tích yêu cầu đến triển khai.

Nâng cao khả năng giải quyết vấn đề, làm việc độc lập và teamwork trong môi trường dự án thật.

## Mục tiêu:

* Mục tiêu tổng quát:

Xây dựng website bán hàng chuyên nghiệp, đầy đủ chức năng như xem và tìm kiếm sản phẩm, thêm vào giỏ hàng, đặt hàng và quản lý đơn hàng.

Cung cấp giao diện quản trị (admin) để dễ dàng kiểm soát dữ liệu, người dùng, sản phẩm và đơn hàng.

* Mục tiêu cụ thể:

Thiết kế giao diện người dùng hiện đại, thân thiện trên cả desktop và thiết bị di động.

Phân quyền tài khoản: khách hàng và quản trị viên.

Bảo mật thông tin và kiểm tra tính hợp lệ khi nhập liệu.

Kết nối frontend và backend xử lý dữ liệu động.

Đảm bảo hệ thống dễ bảo trì và mở rộng về sau.

## Đối tượng nghiên cứu:

* Ngôn ngữ/Lập trình chính: PHP với Laravel Framework.
* Cơ sở dữ liệu: MySQL – lưu trữ toàn bộ dữ liệu hệ thống.
* Giao diện người dùng: HTML, CSS, JavaScript kết hợp với Blade Template (Laravel).
* Quản lý package: Composer (PHP) và npm (frontend).
* Các chức năng chính mô phỏng qua hệ thống:

Quản lý tài khoản người dùng: đăng ký, đăng nhập, đổi mật khẩu, phân quyền.

Quản lý sản phẩm: thêm/sửa/xoá sản phẩm, upload ảnh, phân loại theo danh mục và thương hiệu.

Quản lý đơn hàng: xử lý đặt hàng, theo dõi trạng thái đơn.

Tìm kiếm và lọc sản phẩm theo tên, danh mục.

Giao diện quản trị riêng biệt với giao diện người dùng.

## Phạm vi nghiên cứu:

* Đề tài tập trung vào việc xây dựng chức năng cốt lõi của một hệ thống bán hàng đơn giản, cụ thể là:

Giao diện người dùng và chức năng đặt hàng

Giao diện quản trị viên để quản lý nội dung

Phân quyền tài khoản đăng nhập (admin và user)

Lưu trữ và xử lý dữ liệu nội bộ qua cơ sở dữ liệu

* Các yếu tố không nằm trong phạm vi nghiên cứu gồm:

Tích hợp thanh toán trực tuyến (như VNPay, PayPal)

Tích hợp API vận chuyển hoặc các dịch vụ bên thứ ba

Phát triển ứng dụng di động (mobile app)

Tính năng báo cáo thống kê nâng cao hoặc AI gợi ý sản phẩm

* Phạm vi này giúp đảm bảo tính khả thi trong thời gian thực tập, đồng thời tập trung vào chất lượng và hoàn thiện sản phẩm nền tảng.

## Phương pháp nghiên cứu: Tiến trình nghiên cứu gồm:

* Thu thập tài liệu và học công nghệ:

Nghiên cứu tài liệu chính thức từ Laravel, MySQL, Bootstrap...

Xem video hướng dẫn thực hành trên YouTube và các khóa học online.

* Khảo sát thực tế:

Xem và phân tích các website bán hàng chuyên nghiệp (TGDĐ, FPT Shop, ...).

Rút ra yêu cầu người dùng và các tính năng cần thiết.

* Phát triển hệ thống:

Thiết kế sơ đồ Use Case, ERD.

Phát triển frontend bằng Blade Template và Bootstrap.

Viết logic backend với Laravel Controller/Model.

Thử nghiệm từng chức năng, ghi lại lỗi và fix.

Hoàn thiện hệ thống, đảm bảo các chức năng chạy ổn định.

# CHƯƠNG 1. TỔNG QUAN VỀ CƠ SỞ THỰC TẬP

## Giới thiệu về doanh nghiệp

* Tên công ty: CÔNG TY CỔ PHẦN PROPTECHSOFT
* Địa chỉ: Tầng 3, The Vista, 628C Xa Lộ Hà Nội, Phường An Phú, Quận 2, Thành phố Hồ Chí Minh, Việt Nam
* Lĩnh vực hoạt động: Bán lẻ máy vi tính, thiết bị ngoại vi, phần mềm và thiết bị viễn thông trong các cửa hàng chuyên doanh

## Cơ cấu tổ chức và bộ phận thực tập

### Cơ cấu tổ chức:

* Công ty Cổ phần Proptechsoft là doanh nghiệp hoạt động trong lĩnh vực công nghệ thông tin, chuyên cung cấp giải pháp phần mềm và ứng dụng phục vụ ngành bất động sản (PropTech). Cơ cấu tổ chức của công ty được xây dựng theo mô hình phân cấp chức năng, bao gồm các bộ phận chuyên trách nhằm đảm bảo hiệu quả trong quản lý và vận hành hệ thống phần mềm.
* Sơ đồ cơ cấu tổ chức (mô phỏng):

A diagram of a company

AI-generated content may be incorrect.

### Bộ phận thực tập:

* Trong thời gian thực tập tại công ty, tôi được phân công về Phòng Kỹ Thuật – Nhóm lập trình web, chịu trách nhiệm triển khai một dự án website bán hàng cụ thể từ đầu đén cuối, dưới hình thức thực hiện cá nhân

## Nội quy và quy trình làm việc tại công ty

* Thời gian làm việc: Công ty áp dụng giờ làm việc hành chính từ 8h – 17h30, nghỉ trưa từ 12h – 13-30, từ thứ Hai đến thứ Sáu hàng tuần. Trong suốt thời gian làm việc, nhân viên và thực tập sinh cần đảm bảo có mặt đầy đủ, đúng giờ và tuân thủ các quy định về nội quy lao động của công ty.
* Nội quy cơ bản:

Tuân thủ giờ giấc làm việc, không tự ý nghỉ làm nếu không có sự cho phép của quản lý.

Giữ gìn vệ sinh nơi làm việc, ăn mặc gọn gàn lịch sự.

Không truy cập các trang web, phần mềm không liên quan đến công việc trong giờ làm việc.

Tôn trọng đồng nghiệp, cấp trên, có tinh thần trách nhiệm và hợp tác trong công việc nhóm.

Thực hiện đầy đủ và nghiêm túc các nhiệm vụ được giao.

* Quy trình quản lý làm việc:

Người giao công việc: Các nhiệm vụ được giao trực tiếp bởi anh **Hồ Anh Tuấn – CEO,** đồng thời cũng là người hướng dẫn chính thức trong suốt ký thực tập.

Quy trình nhận và triển khai công việc: Tôi thường xuyên được trao đổi công việc thông qua các lần gặp mặt trực tiếp trong tuần. Các nhiệm vụ sẽ được mô tả rõ ràng yêu cầu, thời gian hoàn thành và tài liệu tham khảo (nếu có). Sau khi nhận nhiệm vụ, tôi tiến hành:

* + - Tìm hiểu yêu cầu công việc và đọc tài liệu liên quan.
    - Tự nghiên cứu và viết mã nguồn theo tài liệu đã có hoặc thành phần đã có.
    - Kiểm thử kết quả, so sánh với yêu cầu ban đầu.
    - Kiểm tra lại code và tiến thành push code lên git.
* Các công cụ hỗ trợ làm việc và lập trình:

Visual Studio Sode.

Xampp

StarUML

…

# LÝ THUYẾT ÁP DỤNG THEO LĨNH VỰC THỰC TẬP

## Tổng quan về công nghệ sử dụng:

* Trong đề tài này, công nghệ chính được sử dụng là Laravel – một framework (khung làm việc) viết bằng ngôn ngữ PHP. Laravel rất nổi tiếng và được dùng rộng rãi để xây dựng các website và ứng dụng web. Điểm mạnh của Laravel là hỗ trợ lập trình theo mô hình MVC (Model - View - Controller), giúp chia nhỏ phần mềm thành 3 phần riêng biệt, dễ quản lý và bảo trì.
* Ngoài Laravel, dự án còn sử dụng các công nghệ khác để hỗ trợ quá trình phát triển:

Laravel: Framework PHP hiện đại, cung cấp đầy đủ công cụ để phát triển ứng dụng web theo kiến trúc chuẩn, tích hợp sẵn hệ thống định tuyến, quản lý middleware, cơ chế bảo mật, xử lý cơ sở dữ liệu…

MySQL: Hệ quản trị cơ sở dữ liệu quan hệ, dùng để lưu trữ toàn bộ dữ liệu như người dùng, sản phẩm, đơn hàng…

Composer: Trình quản lý thư viện PHP. Laravel và các package mở rộng đều được cài đặt, cập nhật và quản lý thông qua Composer.

Node.js và npm: Hệ thống quản lý các gói frontend như CSS/JS (ví dụ: Bootstrap, jQuery...) phục vụ việc thiết kế giao diện người dùng đẹp mắt, hiện đại.

Blade Template (Laravel Template Engine):

* + - Cho phép xây dựng giao diện bằng cách kết hợp giữa HTML và cú pháp mở rộng thân thiện của Laravel.
    - Hỗ trợ các lệnh điều kiện như @if, @foreach, ..., giúp viết giao diện có logic mà vẫn dễ đọc.
    - Hệ thống @extends, @section, @yield giúp tái sử dụng layout và tránh trùng lặp mã nguồn.
    - Hiệu suất cao do Blade được biên dịch thành PHP thuần và lưu vào bộ nhớ đệm.
* Nhờ việc kết hợp các công nghệ trên, quá trình phát triển và triển khai dự án trở nên linh hoạt, nhanh chóng, và dễ mở rộng trong tương lai.

## Kiến trúc hệ thống phần mềm:

* Hệ thống phần mềm được xây dựng theo mô hình MVC – đây là cách tổ chức dự án phổ biến trong các ứng dụng web hiện đại. MVC chia phần mềm thành 3 thành phần chính:

Model: Là nơi xử lý và lưu trữ dữ liệu. Ví dụ, khi người dùng tạo tài khoản, phần Model sẽ chịu trách nhiệm lưu thông tin này vào cơ sở dữ liệu. Model cũng có thể lấy dữ liệu từ database để hiển thị ra ngoài.

View: Là phần giao diện mà người dùng nhìn thấy và tương tác trực tiếp. Ví dụ như các trang đăng ký, đăng nhập, xem danh sách sản phẩm… View chỉ lo việc hiển thị mà không xử lý logic.

Controller: Là “người điều phối” – nhận yêu cầu từ người dùng (như bấm nút, gửi form), sau đó xử lý hoặc chuyển tiếp thông tin cho Model hoặc View. Ví dụ: khi người dùng bấm “Lưu”, Controller sẽ kiểm tra dữ liệu và yêu cầu Model lưu vào database.

* Ngoài ra, Laravel còn có 2 cơ chế quan trọng:

Route: Dùng để xác định các đường dẫn URL trong hệ thống. Ví dụ: khi người dùng truy cập đường dẫn /san-pham, hệ thống biết phải gọi đến chức năng nào.

Middleware: Giống như “người gác cổng” – giúp kiểm tra điều kiện trước khi cho phép truy cập một chức năng nào đó. Ví dụ: chỉ người đăng nhập mới được vào trang quản lý.

* Nhờ vào mô hình MVC kết hợp với Route và Middleware, hệ thống trở nên rõ ràng, dễ quản lý, bảo trì và phát triển thêm tính năng mới.

## Laravel

* Laravel là một framework PHP hiện đại, được phát triển nhằm hỗ trợ lập trình viên xây dựng các ứng dụng web một cách nhanh chóng, có tổ chức và dễ bảo trì. Đây là công nghệ trọng tâm của đề tài vì khả năng tích hợp sẵn nhiều tính năng thiết yếu như:

Kiến trúc MVC rõ ràng: Tách biệt phần giao diện (View), xử lý nghiệp vụ (Controller) và dữ liệu (Model), giúp tăng tính tổ chức, dễ quản lý khi dự án mở rộng.

Routing mạnh mẽ: Laravel cho phép định nghĩa rõ ràng các đường dẫn (URL) và liên kết chúng đến các controller, giúp người dùng tương tác với hệ thống dễ dàng.

Tích hợp hệ thống xác thực người dùng (Authentication): Có sẵn chức năng đăng ký, đăng nhập, phân quyền giúp đảm bảo bảo mật hệ thống.

Migration & Seeder: Quản lý thay đổi cấu trúc cơ sở dữ liệu qua các phiên bản; dễ dàng khởi tạo dữ liệu mẫu khi cần thiết.

Validation và xử lý lỗi: Laravel hỗ trợ kiểm tra dữ liệu đầu vào từ phía server một cách dễ dàng và bảo mật.

Hệ sinh thái phong phú: Laravel hỗ trợ rất nhiều gói mở rộng (package) phục vụ cho các tính năng nâng cao như gửi mail, thanh toán, API, chatbot...

* Laravel được lựa chọn trong đề tài không chỉ vì sự phổ biến, mà còn bởi khả năng mở rộng tốt, giúp sinh viên dễ dàng triển khai thêm các tính năng nâng cao trong tương lai nếu cần thiết (ví dụ: thanh toán trực tuyến, phân tích dữ liệu, hoặc tích hợp ứng dụng di động).

## MySQL

* MySQL là hệ quản trị cơ sở dữ liệu quan mã nguồn mở rất phổ biến, và được sử dụng để lưu trữ toàn bộ dữ liệu trong dự án.
* Trong hệ thống bán hàng này, MySQL đảm nhiệm việc lưu trữ và truy xuất các thông tin như:

Tài khoản người dùng (admin và khách hàng).

Danh mục, thương hiệu và sản phẩm.

Giỏ hàng và đơn hàng.

Lịch sử giao dịch và trạng thái đơn hàng.

* Một số ưu điểm khi dùng MySQL trong đề tài:

Khả năng xử lý dữ liệu lớn nhanh chóng: giúp truy vấn hàng trăm sản phẩm, đơn hàng một cách hiệu quả.

Dễ dàng tích hợp với Laravel thông qua Eloquent ORM: Giúp sinh viên không cần viết câu lệnh SQL phức tạp, mà vẫn có thể thao tác dữ liệu linh hoạt.

Hỗ trợ các ràng buộc (foreign key) giúp đảm bảo tính toàn vẹn dữ liệu trong quan hệ giữa các bảng như: người dùng – đơn hàng, đơn hàng – chi tiết đơn hàng,...

* Cấu trúc cơ sở dữ liệu trong đề tài được thiết kế theo hướng chuẩn hóa để tránh dư thừa, dễ bảo trì và mở rộng.

## Composer

* Composer là một công cụ quản lý thư viện và phụ thuộc chuyên dụng cho ngôn ngữ lập trình PHP. Trong các dự án phần mềm hiện đại, việc sử dụng Composer là không thể thiếu để tự động hóa việc cài đặt, cập nhật và quản lý các gói thư viện từ bên thứ ba hoặc các module mở rộng của chính dự án.
* Vai trò của Composer trong dự án:

Quản lý thư viện tập trung: Composer giúp dự án dễ dàng tích hợp các gói mở rộng cần thiết như Laravel, thư viện xử lý ảnh, gửi email, hoặc các module bảo mật, mà không cần tải về thủ công từng gói.

Tự động hoá quá trình cài đặt: Thay vì mất thời gian cấu hình thủ công, Composer tự động tải về các phiên bản thư viện tương thích dựa trên thông số đã được khai báo.

Đảm bảo đồng nhất môi trường phát triển: Composer tạo ra tệp cấu hình chi tiết (composer.lock) để đảm bảo rằng tất cả thành viên trong nhóm, hoặc các môi trường máy chủ khác nhau, đều cài đúng phiên bản thư viện giống nhau.

Tăng khả năng mở rộng và bảo trì: Khi muốn nâng cấp hệ thống, lập trình viên chỉ cần cập nhật Composer để thay đổi phiên bản các gói hoặc thêm mới mà không ảnh hưởng đến phần còn lại của hệ thống.

* Tầm quan trọng trong thực tiễn:

Composer giúp sinh viên tiếp cận với cách tổ chức dự án chuyên nghiệp, hiện đại.

Rút ngắn thời gian triển khai và cấu hình khi khởi tạo một dự án web.

Giúp đảm bảo an toàn, ổn định và dễ dàng phục hồi môi trường phát triển nếu có sự cố xảy ra.

Là công cụ tiêu chuẩn trong hầu hết dự án PHP từ quy mô nhỏ đến lớn

## Node.js và npm

* Node.js là một môi trường chạy JavaScript phía máy chủ. npm (Node Package Manager) đi kèm theo Node.js và được dùng để quản lý các gói thư viện frontend.
* Trong đề tài, npm chủ yếu phục vụ cho:

Quản lý các thư viện giao diện: như Bootstrap, jQuery, ...

Biên dịch và đóng gói các file tĩnh: như SCSS/CSS, JS thông qua công cụ Vite hoặc Webpack Mix.

Cải thiện hiệu suất hiển thị giao diện: Thông qua việc nén file, chia nhỏ tài nguyên, hoặc tải không đồng bộ (lazy loading).

* Một số lệnh thường dùng:

npm install – cài đặt tất cả các thư viện đã khai báo.

npm run dev – chạy dự án ở chế độ phát triển (không nén file).

* Npm giúp frontend luôn gọn nhẹ, dễ chỉnh sửa và đảm bảo hiệu năng khi chạy thực tế.

## Blade Template

* Blade Template là hệ thống xây dựng giao diện (template engine) được tích hợp sẵn trong Laravel. Đây là một công cụ hỗ trợ lập trình viên tạo giao diện web một cách hiệu quả, có tổ chức và dễ bảo trì – đặc biệt là khi giao diện có nhiều phần lặp lại hoặc có điều kiện hiển thị khác nhau theo ngữ cảnh.
* Vai trò và đặc điểm chính:

Phân tách giao diện và xử lý nghiệp vụ: Blade giúp tách biệt phần hiển thị (View) ra khỏi các xử lý logic trong Controller, đúng theo mô hình MVC. Điều này đảm bảo mã nguồn dễ hiểu, dễ sửa đổi và tăng tính tổ chức cho toàn hệ thống.

Hỗ trợ xây dựng giao diện động: Blade cho phép kết hợp dữ liệu từ hệ thống vào giao diện một cách linh hoạt.

Khả năng tái sử dụng giao diện: Với Blade, lập trình viên có thể xây dựng các bố cục (layout) dùng chung cho toàn hệ thống như khung header, footer, menu và chỉ thay đổi nội dung chính. Điều này giúp tiết kiệm thời gian và tránh trùng lặp giao diện giữa các trang.

Tối ưu hiệu suất xử lý: Blade được thiết kế để biên dịch sang mã PHP thuần khi chạy, điều này giúp quá trình hiển thị giao diện nhanh và hiệu quả, không gây chậm hệ thống dù có nhiều thao tác logic hiển thị.

* Vai trò trong dự án thực tế:

Blade Template được sử dụng để xây dựng tất cả các trang giao diện trong hệ thống từ phần khách hàng đến phần quản trị viên.

Giao diện được chia thành các phần riêng biệt: bố cục chung, các khối nội dung và các thành phần hiển thị điều kiện (theo người dùng, sản phẩm, trạng thái…).

Nhờ Blade, quá trình phát triển giao diện trở nên khoa học, nhất quán, dễ nâng cấp và duy trì lâu dài.

* Blade không chỉ là một công cụ hiển thị mà còn là cầu nối hiệu quả giữa backend và frontend trong toàn bộ hệ sinh thái Laravel.

## Git: Quản lý source code

* Git là hệ thống quản lý phiên bản phân tán được sử dụng phổ biến nhất hiện nay trong phát triển phần mềm. Git cho phép nhiều lập trình viên cùng làm việc trên cùng một dự án mà không gây xung đột về mã nguồn, đồng thời lưu trữ đầy đủ lịch sử thay đổi của từng tệp tin theo thời gian.
* Đặc điểm nổi bật:

Quản lý phiên bản hiệu quả: Git lưu trữ mọi thay đổi trong mã nguồn thông qua các commit giúp dễ dàng quay lại trạng thái trước đó khi có lỗi xảy ra.

Phân nhánh (branching) linh hoạt: Cho phép tạo các nhánh để phát triển tính năng mới, sửa lỗi, hoặc thử nghiệm, rồi gộp lại bằng thao tác merge hoặc rebase.

Làm việc offline: Vì là hệ thống phân tán, mọi thao tác như commit, diff, log... đều có thể thực hiện mà không cần kết nối Internet.

Tích hợp với các nền tảng: Git thường được kết hợp với các dịch vụ như GitHub, GitLab, Bitbucket để hỗ trợ cộng tác nhóm và quản lý mã nguồn từ xa.

* Lợi ích trong dự án thực tế:

Giúp kiểm soát thay đổi mã nguồn chặt chẽ và minh bạch.

Tăng khả năng cộng tác giữa các thành viên trong nhóm dự án.

Dễ dàng quản lý phiên bản, phân quyền và tự động triển khai (CI/CD) qua các công cụ hỗ trợ như GitHub Actions hoặc GitLab CI.

# NỘI DUNG THỰC TẬP

## Mục tiêu thực tập

* Tiếp cận môi trường làm việc thực tế, quy trình phát triển dự án.
* Nâng cao kỹ năng lập trình và làm việc nhóm.
* Áp dụng kiến thức học được vào các dự án thực tế.
* Phát triển thêm kỹ năng mềm.
* Rèn luyện khả năng tự học và tự trau dồi.
* Luyện tập giải quyết vấn đề độc lập.

## Công việc được giao

* Lập đề cương thực tập: Xác định đề tài về "Xây dựng website bán hàng", lên kế hoạch và mục tiêu thực hiện.
* Cài đặt môi trường phát triển: Thiết lập PHP, Composer, Laravel, Node.js, MySQL; tạo project và cấu hình kết nối database.
* Phân tích và thiết kế hệ thống:

Sử dụng sơ đồ Use Case để xác định chức năng và luồng nghiệp vụ.

Vẽ sơ đồ quy trình để xác định các bước xử lý của hệ thống.

Thiết kế cơ sở dữ liệu.

* Xây dựng giao diện người dùng:

Giao diện khách hàng: Trang chủ, danh mục sản phẩm, chi tiết sản phẩm, giỏ hàng, tìm kiếm sản phẩm.

Giao diện admin: Dashboard quản trị, quản lý danh mục, thương hiệu, sản phẩm, đơn hàng, người dùng.

Giao diện đăng nhập, đăng ký, quên mật khẩu.

* Phát triển chức năng hệ thống:

Đăng ký, đăng nhập, đổi mật khẩu, quên mật khẩu.

Quản lý danh mục, sản phẩm, thương hiệu (CRUD).

Quản lý khách hàng, đơn hàng (CRUD).

Thêm vào giỏ hàng, đặt hàng, tìm kiếm sản phẩm.

Phân quyền người dùng: Admin và khách hàng.

* Kết nối frontend và backend:

Hiển thị dữ liệu từ database ra giao diện.

Xử lý validation, thông báo lỗi nhập liệu.

Bảo mật form và xác thực người dùng với middleware.

## Kết quả làm việc

* Đề cương thực tập: hoàn thành đúng tiến độ, được giáo viên hướng dẫn phê duyệt.
* Xây dựng giao diện: Hoàn thành giao diện frontend cho trang khách hang và admin dashboard.
* Xây dựng backend: hoàn thiện chức năng CRUD cho sản phẩm, danh mục, thương hiệu, người dung và đơn hàng.
* Kết nối frontend và backend: Dữ liệu hiển thị động, xử lý nhập liệu, kiểm tra tính hợp lệ, bảo mật ứng dụng

## Nhật ký thực tập

* Tuần 1 (14/04/2025 - 19/04/2025)
* Nhận đề tài, nghiên cứu tổng quan, viết đề cương thực tập.
* Kết quả: Hoàn thành đề cương.
* Tuần 2 (21/04/2025 - 26/04/2025)
* Phân tích yêu cầu hệ thống, mô tả các chức năng chính và xây dựng các tài liệu mô tả yêu cầu.
* Kết quả: Hoàn thành tài liệu mô tả chức năng hệ thống.
* Tuần 3 (28/04/2025 – 03/05/2025)
* Vẽ sơ đồ Use Case, sơ đồ ERD, thiết kế giao diện UI
* Kết quả: Bộ tài liệu thiết kế hệ thống hoàn chỉnh.
* Tuần 4 (05/05/2025 - 10/05/2025)
* Cài đặt môi trường làm việc Laravel, MySQL, NodeJS.
* Kết quả: Tạo project, khởi tạo database.
* Tuần 5 (12/05/2025 - 17/05/2025)
* Xây dựng giao diện khách hàng và admin.
* Kết quả: Hoàn thành 70% giao diện chính.
* Tuần 6 (19/05/2025 - 24/05/2025)
* Xây dựng các chức năng quản lý người dùng, đơn hàng, sản phẩm, …
* Kết quả: Hoàn thành chức năng CRUD cơ bản.
* Tuần 7 (26/05/2025 - 31/05/2025)
* Phát triển chức năng giỏ hàng, tìm kiếm sản phẩm.
* Kết quả: Kết nối frontend - backend hiển thị dữ liệu động.
* Tuần 8 (02/06/2025 - 07/06/2025)
* Hoàn thiện phân quyền user.
* Kết quả: Hệ thống bảo mật và hoạt động ổn định.
* Tuần 9 (09/06/2025 - 14/06/2025)
* Kiểm thử, fix lỗi, hoàn thiện project.
* Kết quả: Dự án hoàn chỉnh, có thể triển khai.

## Phân tích thiết kế hệ thống & triển khai

### Phân tích thiết kế hệ thống:

* Mô tả tổng quan hệ thống: Hệ thống website bán hàng thiết bị máy ảnh là một ứng dụng web thương mại điện tử, hỗ trợ khách hàng xem thông tin sản phẩm, đặt hàng và theo dõi tình trạng đơn hàng. Đồng thời, hệ thống cũng có khu vực quản trị giúp người quản trị quản lý sản phẩm, người dùng, thương hiệu, đơn hàng và phân quyền tài khoản. Hệ thống được phát triển dựa trên mô hình MVC (Model – View – Controller) sử dụng Laravel Framework.
* Phân tích yêu cầu:

Chức năng người dùng (Frontend):

* + - Đăng ký, đăng nhập, quên mật khẩu.
    - Duyệt danh mục sản phẩm, xem chi tiết, tìm kiếm sản phẩm.
    - Thêm vào giỏ hàng, cập nhật số lượng, đặt hàng.
    - Tra cứu đơn hàng bằng mã đơn hàng và email.

Chức năng quản trị (Admin):

* + - Quản lý danh mục, thương hiệu, sản phẩm.
    - Quản lý người dùng, phân quyền tài khoản.
    - Quản lý đơn hàng và trạng thái xử lý.
    - Quản lý hình ảnh sản phẩm và khôi phục từ thùng rác.

Yêu cầu phi chức năng:

* + - Giao diện thân thiện, phản hồi nhanh.
    - Tốc độ xử lý tốt và bảo mật cao.
    - Có thể mở rộng tính năng trong tương lai.
* Usecase:

A diagram with text and images

AI-generated content may be incorrect.

Hình 1: Use Case Diagram

|  |  |
| --- | --- |
| **Use Case: Xem chi tiết sản phẩm** | |
| Actor | Hành động |
| Người dùng | Khách hàng truy cập trang chủ |
| Người dùng | |  | | --- | | Chọn sản phẩm muốn xem |  |  | | --- | |  | |
| Người dùng | |  | | --- | | Khách hàng có thể xem chi tiết sản phẩm |  |  | | --- | |  | |
| Hệ thống | |  | | --- | | Hệ thống hiển thị nội dung chi tiết sản phẩm |  |  | | --- | |  | |
| **Use Case: Tìm kiếm sản phẩm** | |
| Actor | Hành động |
| Người dùng | Khách hàng truy cập trang chủ |
| Người dùng | |  | | --- | | Khách hàng nhập tên sản phẩm cần tìm |  |  | | --- | |  | |
| Hệ thống | |  | | --- | | Hệ thống lọc và hiển thị kết quả |  |  | | --- | |  | |
| Hệ thống | |  | | --- | | Hệ thống hiển thị danh sách tên sản phẩm liên quan |  |  | | --- | |  | |

|  |  |
| --- | --- |
| **Use Case: Đặt hàng sản phẩm** | |
| Actor | Hành động |
| Người dùng | Khách hàng truy cập trang sản phẩm |
| Người dùng | |  | | --- | | Chọn sản phẩm muốn mua. |  |  | | --- | |  | |
| Người dùng | |  | | --- | | Thêm sản phẩm vào giỏ hàng |  |  | | --- | |  | |
| Người dùng | Kiểm tra giỏ hàng |
| Người dùng | |  | | --- | | Xác nhận thông tin đặt hàng |  |  | | --- | |  | |
| Người dùng | |  | | --- | | Gửi đơn đặt hàng |  |  | | --- | |  | |
| Hệ thống | |  | | --- | | Tạo đơn hàng và lưu thông tin vào bảng orders & order\_items |  |  | | --- | |  | |
| Hệ thống | |  | | --- | | Cập nhật trạng thái đơn đặt hàng |  |  | | --- | |  | |

|  |  |
| --- | --- |
| **Use Case: Quản lý đơn hàng** | |
| Actor | Hành động |
| Admin | |  | | --- | | Xem danh sách đơn hàng |  |  | | --- | |  | |
| Admin | |  | | --- | | Xem chi tiết đơn hàng |  |  | | --- | |  | |
| Admin | |  | | --- | | Cập nhật trạng thái đơn hàng |  |  | | --- | |  | |

|  |  |
| --- | --- |
| **Use Case: Quản lý tài khoản người dùng (Đăng ký, đăng nhập, quên mật khẩu, đổi mật khẩu)** | |
| Actor | Hành động |
| Người dùng | |  | | --- | | Nhập email và mật khẩu để đăng nhập |  |  | | --- | |  | |
| Hệ thống | |  | | --- | | Xác thực tài khoản |  |  | | --- | |  | |
| Hệ thống | |  | | --- | | Cho phép truy cập vào hệ thống |  |  | | --- | |  | |
| Người dùng | |  | | --- | | Đăng ký tạo tài khoản mới (nếu chưa) |  |  | | --- | |  | |
| Người dùng | Nhấn "Quên mật khẩu" nếu không nhớ mật khẩu (nếu đã có tài khoàn) |
| Hệ thống | |  | | --- | | Gửi email chứa mật khẩu mới |  |  | | --- | |  | |
| Người dùng | |  | | --- | | Nhập mật khẩu mới được gửi từ email |  |  | | --- | |  | |
| Người dùng | |  | | --- | | Sau khi đăng nhập có thể đổi mật khẩu trong trang tài khoản cá nhân |  |  | | --- | |  | |
| Hệ thống | |  | | --- | |  |  |  | | --- | | Cập nhật lại mật khẩu mới trong cơ sở dữ liệu | |

* Cơ sở dữ liệu:

Cơ sở dữ liệu được thiết kế chuẩn hóa với các bảng chính sau (dựa trên file final\_assignment.sql):

* + - users: Quản lý thông tin người dùng hệ thống, gồm admin và khách hàng.
    - categories: Lưu trữ danh mục sản phẩm.
    - brands: Lưu thông tin thương hiệu sản phẩm.
    - products: Lưu thông tin sản phẩm như tên, mô tả, giá, ảnh,...
    - product\_images: Lưu nhiều ảnh phụ của từng sản phẩm.
    - customers: Lưu thông tin khách hàng khi đặt hàng.
    - orders: Quản lý đơn hàng, liên kết với customers.
    - order\_items: Lưu chi tiết từng sản phẩm trong mỗi đơn hàng.

Cơ sở dữ liệu hỗ trợ ràng buộc khóa ngoại và timestamp (ngày tạo, cập nhật), đồng thời có tích hợp soft deletes giúp khôi phục dữ liệu khi cần.

Cấu trúc các bảng:

A screenshot of a computer

AI-generated content may be incorrect.

Hình 2: Bảng brands & categories

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3: Bảng customers

A screenshot of a computer

AI-generated content may be incorrect.

Hình 4: Bảng products

A screenshot of a computer

AI-generated content may be incorrect.

Hình 5: Bảng orders

A screenshot of a computer

AI-generated content may be incorrect.

Hình 6: Bảng orders\_items

A screenshot of a computer

AI-generated content may be incorrect.

Hình 7: Bảng users

**A screenshot of a computer

AI-generated content may be incorrect.**

Hình 8: Mối quan hệ giữa các bảng

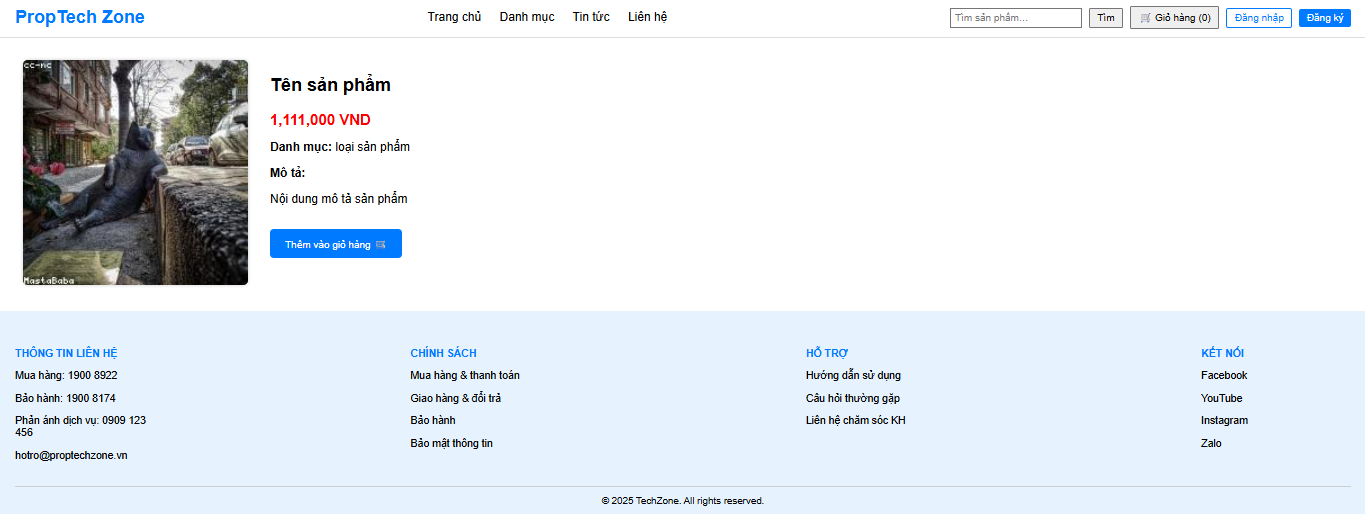
* Giao diện UX/UI:

Giao diện người dùng (khách hàng):

A screenshot of a computer

AI-generated content may be incorrect.

Hình 9: Trang chủ



Hình 10: Chi tiết sản phẩm

A screenshot of a computer

AI-generated content may be incorrect.

Hình 11: Giỏ hàng

A screenshot of a computer

AI-generated content may be incorrect.

Hình 12: Xác nhận đặt hàng

A screenshot of a phone

AI-generated content may be incorrect. A screenshot of a computer

AI-generated content may be incorrect.

Hình 13 & 14: Đăng ký & Đăng nhập

A screenshot of a computer

AI-generated content may be incorrect.

Hình 15: Đổi mật khẩu

Giao diện admin:

A screenshot of a computer

AI-generated content may be incorrect.

Hình 16: Dashboard quản trị tổng quan (thống kê cả doanh thu)

A screenshot of a computer

AI-generated content may be incorrect.

Hình 17: Trang quản lý sản phẩm

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 18 & 19: Trang quản lý thương hiệu và loại sản phẩm

A screenshot of a computer

AI-generated content may be incorrect.

Hình 20: Trang quản lý đơn hàng

A screenshot of a computer

AI-generated content may be incorrect.

Hình 21: Trang quản lý người dùng

A screenshot of a music tab

AI-generated content may be incorrect.

Hình 22: Form thêm / sửa thông tin

A screenshot of a computer

AI-generated content may be incorrect.

*Hình 23: Trang thủng rác*

A screenshot of a video

AI-generated content may be incorrect.

*Hình 24: Phân quyền người dung*

### Triển khai:

* Cài đặt:

Môi trường phát triển:

* + - **Laravel 10 trở lên**: Framework PHP hiện đại, mạnh mẽ.
    - **PHP 8.1 hoặc cao hơn**: Phiên bản PHP hỗ trợ các tính năng mới, bảo mật và tương thích với Laravel.
    - **Composer**: Trình quản lý thư viện cho PHP, dùng để cài đặt Laravel và các package liên quan.
    - **Node.js & npm**: Dùng để biên dịch các tài nguyên frontend (CSS, JS) thông qua công cụ Vite.
    - **MySQL**: Hệ quản trị cơ sở dữ liệu quan hệ, lưu trữ dữ liệu của hệ thống.
    - **XAMPP**: Phần mềm hỗ trợ cài đặt Apache, MySQL và PHP để chạy máy chủ cục bộ.

Các bước thực hiện:

* + - Bước 1: Cài đặt Laravel bằng Composer.
    - Bước 2: Cấu hình .env để kết nối với cơ sở dữ liệu.
    - Bước 3: Khởi tạo bảng bằng migrate, thêm dữ liệu bằng seeder.
    - Bước 4: Biên dịch giao diện với npm run dev.
* Controller:

Controller đóng vai trò **trung gian giữa Model và View.**

Nó tiếp nhận các hành động từ người dùng (ví dụ: click nút, gửi biểu mẫu), xử lý yêu cầu, gọi Model để thao tác dữ liệu, rồi gửi kết quả đến View để hiển thị.

Controller giúp chia nhỏ logic của từng chức năng, giúp mã nguồn gọn gàng và dễ kiểm soát.

* Model:

Model là nơi làm việc trực tiếp với **cơ sở dữ liệu.**

Trong Laravel, mỗi bảng dữ liệu thường tương ứng với một Model.

Model dùng để thực hiện các thao tác như: **lấy dữ liệu, thêm mới, sửa, xóa,** hoặc **liên kết dữ liệu giữa các bảng.**

Laravel hỗ trợ **Eloquent ORM** – một công cụ giúp thao tác với cơ sở dữ liệu bằng cú pháp PHP đơn giản, thay vì phải viết nhiều câu lệnh SQL thủ công.

* View:

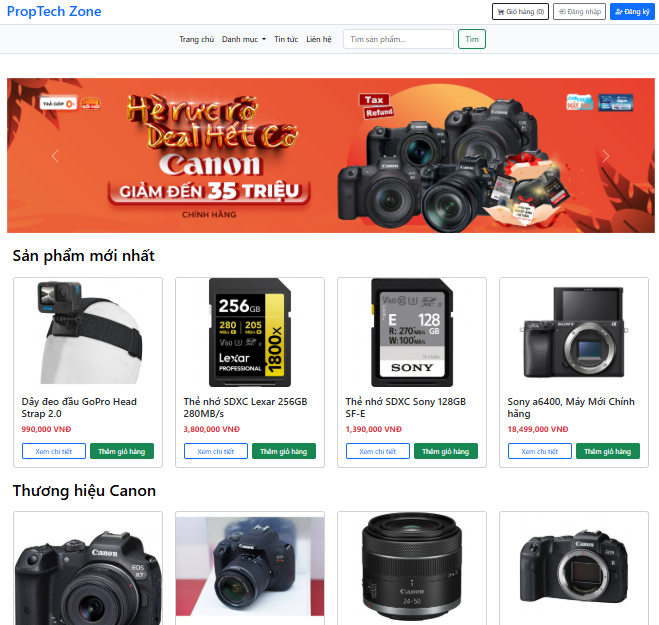
View là nơi hiển thị thông tin cho người dung, là phần **giao diện website.**

Laravel sử dụng hệ thống **Blade Template**, cho phép kết hợp giữa HTML và mã lệnh đơn giản của PHP.

View có thể được chia nhỏ thành các phần như: layout chung (header, footer), các phần tái sử dụng (form, bảng...), giúp dễ bảo trì và quản lý.

Dữ liệu từ Controller sẽ được gửi đến View để hiển thị lên trang web.

* Hình ảnh kết quả đạt được:



Hình 25: Trang chủ.

A screenshot of a web page

AI-generated content may be incorrect.

A screenshot of a web page

AI-generated content may be incorrect.

Hình 26 & 27: Hiển thị sản phẩm theo danh mục.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 28: Chi tiết sản phẩm.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 29 & 30: Giỏ hàng.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 31: Xác nhận đặt hàng

A screenshot of a video game

AI-generated content may be incorrect.

Hình 32: Thông báo sau khi thêm sản phẩm vào giỏ hàng thành công

A screenshot of a web page

AI-generated content may be incorrect.

Hình 33: Thông báo sau khi đặt hàng thành công

A screenshot of a computer

AI-generated content may be incorrect.

Hình 34: Tra cứu đơn hàng sau khi đặt

A screenshot of a computer

AI-generated content may be incorrect.

Hình 35: Trang đăng ký cho khách hàng

A screenshot of a computer

AI-generated content may be incorrect.

Hình 36: Trang đăng nhập cho khách hàng

A screenshot of a computer

AI-generated content may be incorrect.

Hình 37: Trang đổi mật khẩu cho khách hàng

A screenshot of a computer

AI-generated content may be incorrect.

Hình 38: Trang tin tức

A screenshot of a computer

AI-generated content may be incorrect.

Hình 39: Trang liên hệ

A screenshot of a computer

AI-generated content may be incorrect.

*Hình 40: Trang tìm sản phẩm theo tên*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 41: Trang admin: Dashboard*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 42: Trang admin: Đăng nhập*

*A screenshot of a phone

AI-generated content may be incorrect.*

*Hình 43: Trang admin: Đăng ký*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 44: Trang admin: Đổi mật khẩu*

*A screenshot of a computer

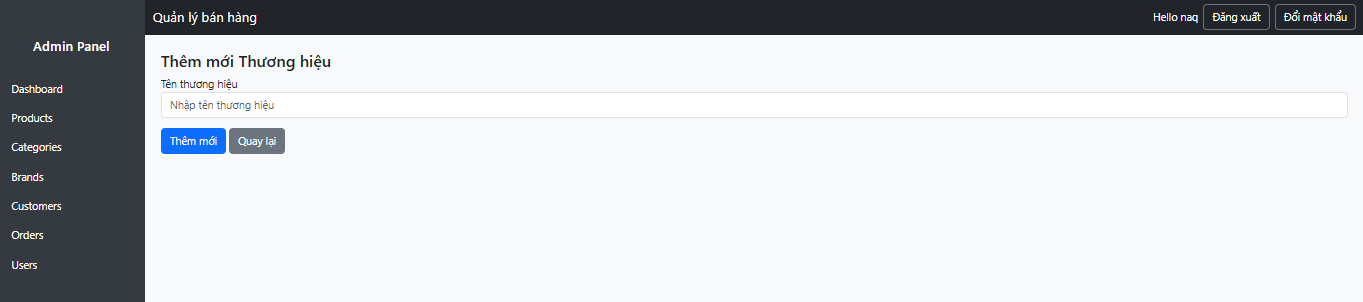
AI-generated content may be incorrect.*

*Hình 45: Trang admin: Quên mật khẩu*

*A screenshot of a phone

AI-generated content may be incorrect.*

*Hình 46: Trang admin: Thương hiệu (loại sản phẩm cũng tương tự)*

**

*Hình 47: Trang admin: Thêm thương hiệu (loại sản phẩm cũng tương tự)*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 48: Trang admin: Chỉnh sửa thương hiệu (loại sản phẩm cũng tương tự)*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 49: Trang admin: Thùng rác thương hiệu (loại sản phẩm cũng tương tự)*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 50: Trang admin: Sản phẩm*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 51: Trang admin: Xem & chỉnh sửa sản phẩm*

*A screenshot of a chat

AI-generated content may be incorrect.*

*Hình 52: Trang admin: Thêm mới sản phẩm*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 53: Trang admin: Thùng rác sản phẩm*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 54: Trang admin: Khách hàng*

*A screenshot of a chat

AI-generated content may be incorrect.*

*Hình 55: Trang admin: Chỉnh sửa khách hàng*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 56: Trang admin: Đơn hàng*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 57: Trang admin: Chi tiết đơn hàng*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 58: Trang admin: Chỉnh sửa đơn hàng*

A screenshot of a computer

AI-generated content may be incorrect.

*Hình 59: Trang admin: Người dùng*

*A screenshot of a phone

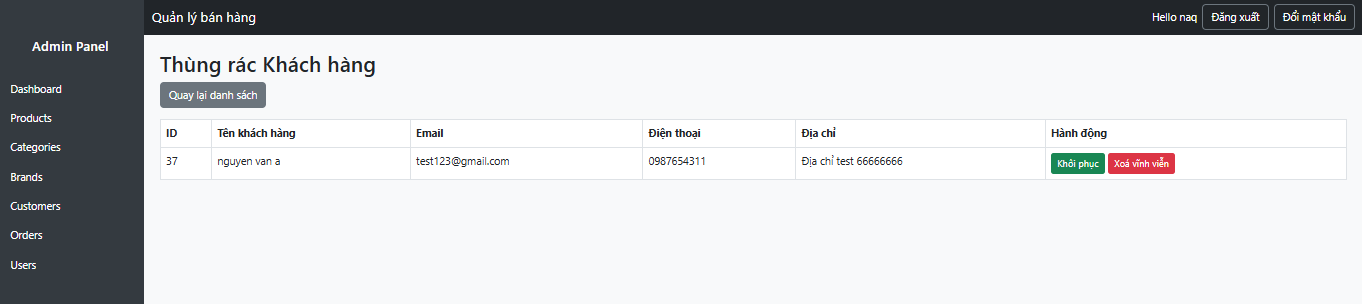
AI-generated content may be incorrect.*

*Hình 60: Trang admin: Chỉnh sửa người dùng*

*A screenshot of a computer

AI-generated content may be incorrect.*

*Hình 61: Trang admin: Thùng rác người dùng*

**

*Hình 62: Trang admin: Thùng rác khách hàng*

## Hình ảnh minh họa

A person sitting at a desk using a computer

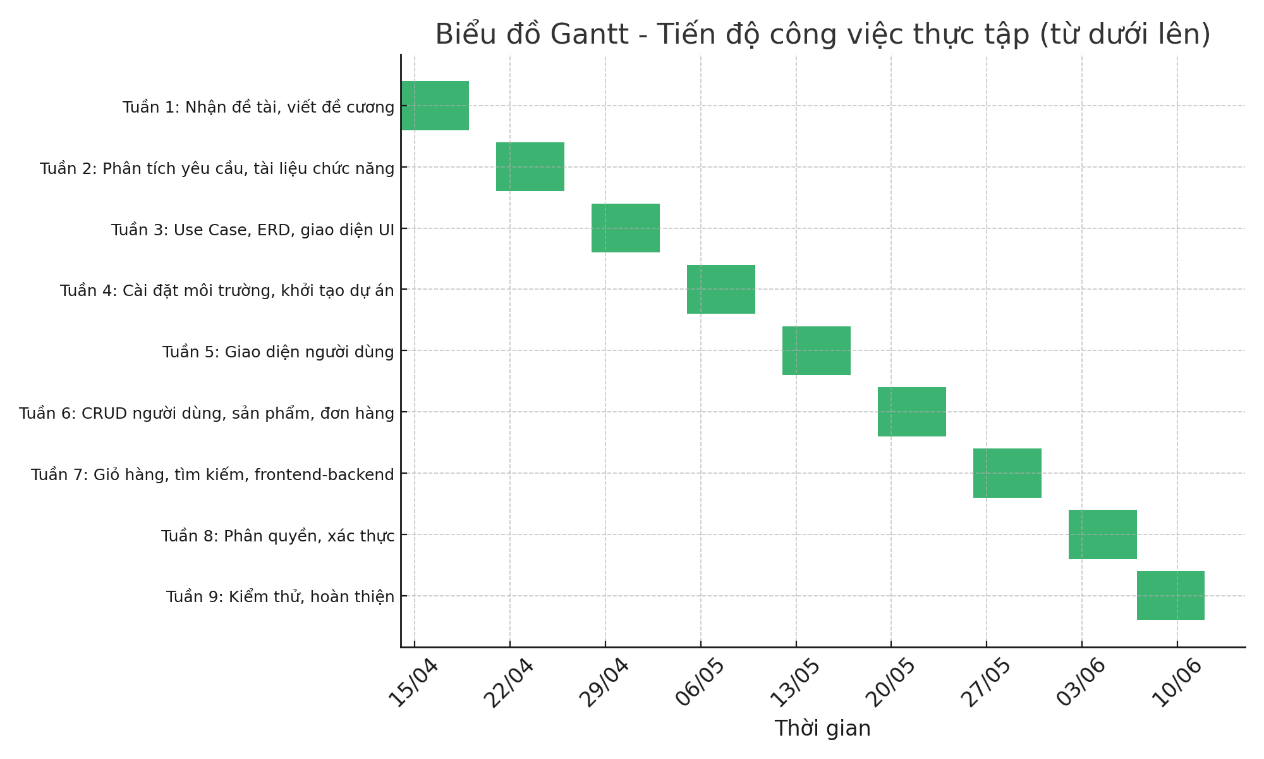
AI-generated content may be incorrect.

A person sitting at a desk with a computer

AI-generated content may be incorrect.

Hình 63 & 64: Hình ảnh thực tập sinh làm việc tại công ty

## Biểu đồ thể hiện tiến độ công việc theo tuần



# KẾT LUẬN VÀ KIẾN NGHỊ

## Kết quả đạt được

* Hoàn thành đề tài đúng tiến độ và yêu cầu đề ra.
* Thiết kế và triển khai được một hệ thống web đầy đủ chức năng quản lý dữ liệu.
* Nắm vững quy trình xây dựng phần mềm từ phân tích, thiết kế đến lập trình và kiểm thử.
* Nâng cao kỹ năng làm việc nhóm, quản lý thời gian và giải quyết vấn đề kỹ thuật thực tế.

## Những khó khăn gặp phải và cách khắc phục

* Khó khăn:

Khó khăn khi cài đặt Laravel vì chưa quen thao tác

Giao diện ban đầu chưa đẹp, chưa hợp lý

* Cách khắc phục:

Tìm đọc tài liệu chính thức, xem video hướng dẫn, hỏi bạn bè và giảng viên.

Tham khảo các mẫu thiết kế và áp dụng Bootstrap để cải thiện giao diện.

## Bài học kinh nghiệm

* Cần tìm hiểu kỹ yêu cầu và xác định phạm vi dự án rõ ràng ngay từ đầu.
* Phân chia thời gian hợp lý cho từng giai đoạn giúp đảm bảo tiến độ công việc.
* Chủ động trong học hỏi công nghệ và tìm kiếm tài liệu là yếu tố quan trọng để thành công.

## Đề xuất và hướng phát triển

* Bổ sung thêm chức năng thống kê báo cáo và gửi thông báo qua email.
* Xây dựng thêm ứng dụng mobile để người dùng tiện truy cập (hoặc có thể thiết kế thêm responsive cho website).
* Áp dụng các công nghệ frontend hiện đại như ReactJS để giao diện đẹp và mượt hơn.

TÀI LIỆU THAM KHẢO

* 1. Laravel Documentation (2024). Laravel - The PHP Framework for Web Artisans. Truy cập tại: <https://laravel.com/docs/10.x>
  2. Vite Documentation (2024). Vite - Next Generation Frontend Tooling.

Truy cập tại: <https://vitejs.dev/guide/>

* 1. MySQL Documentation (2024). MySQL Reference Manual.

Truy cập tại: <https://dev.mysql.com/doc/>

* 1. Bootstrap Documentation (2024). Bootstrap v5.

Truy cập tại: <https://getbootstrap.com/docs/5.3/getting-started/introduction/>

* 1. W3Schools (2024). HTML & CSS, PHP, JavaScript Tutorials.

Truy cập tại: <https://www.w3schools.com/>