

**BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP. HCM
KHOA CÔNG NGHỆ THÔNG TIN**



WEBSITE TIN TỨC

Đồ án môn học:

Lập trình mã nguồn mở

SINH VIÊN THỰC HIỆN:

2001225916 – Tạ Nguyên Vũ (NT)

2001222641 – Trần Công Minh

2001225676 – Lê Đức Trung

2001224227 – Nguyễn Chí Tài

TP. HỒ CHÍ MINH, THÁNG 11 NĂM 2025

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP. HCM
KHOA CÔNG NGHỆ THÔNG TIN



WEBSITE TIN TỨC

Đồ án môn học:

Lập trình mã nguồn mở

SINH VIÊN THỰC HIỆN:

2001225916 – Tạ Nguyên Vũ (NT)

2001222641 – Trần Công Minh

2001225676 – Lê Đức Trung

2001224227 – Nguyễn Chí Tài

TP. HỒ CHÍ MINH, THÁNG 11 NĂM 2025

LỜI CẢM ƠN

Để hoàn thành đồ án học phần **Lập trình mã nguồn mở** này, bên cạnh sự nỗ lực của bản thân các thành viên, nhóm chúng em đã nhận được rất nhiều sự quan tâm, giúp đỡ và hỗ trợ nhiệt tình.

Trước hết, chúng em xin gửi lời cảm ơn chân thành và sâu sắc nhất đến Thầy **Huỳnh Khắc Duy**. Thầy đã tận tình chỉ bảo, định hướng, giải đáp các thắc mắc và cung cấp cho chúng em những kiến thức chuyên môn quý báu trong suốt quá trình thực hiện đề tài.

Mặc dù đã rất cố gắng, nhưng do hạn chế về kiến thức và thời gian, đồ án chắc chắn không thể tránh khỏi những thiếu sót. Chúng em rất mong nhận được những ý kiến đóng góp quý báu từ Thầy và các bạn để sản phẩm được hoàn thiện hơn.

Chúng em xin chân thành cảm ơn!

LỜI CAM ĐOAN

Chúng em xin cam đoan đồ án với đề tài "Website Tin Tức" là công trình nghiên cứu và thực hiện của chính nhóm chúng em.

Chúng em xin cam kết:

1. Nội dung trình bày trong báo cáo này là trung thực và nguyên bản.
2. Tất cả các ý tưởng, kết quả và tài liệu tham khảo không phải của chúng em đều đã được trích dẫn và ghi rõ nguồn gốc một cách minh bạch, tuân thủ các quy định về sở hữu trí tuệ.
3. Mã nguồn (codebase) của dự án là kết quả của quá trình làm việc chung của tất cả các thành viên trong nhóm, không sao chép toàn bộ từ bất kỳ dự án có sẵn nào khác mà không có trích dẫn.
4. Chúng em hoàn toàn chịu trách nhiệm về tính xác thực của nội dung báo cáo và sản phẩm đồ án.

TP.HCM, tháng 11 năm 2025

BẢNG PHÂN CÔNG CÔNG VIỆC

MSSV	Tên	Công việc	Đánh giá
2001225916	Tạ Nguyên Vũ	<ul style="list-style-type: none"> Phân tích & Thiết kế hệ thống (Cấu trúc CSDL, ERD). Lập trình Backend (Core: Models, Migrations, Hệ thống Xác thực - Auth, Google OAuth). Quản lý source code (Git), merge code. Tổng hợp và hoàn thiện báo cáo tổng thể. 	100%
2001222641	Trần Công Minh	<ul style="list-style-type: none"> Lập trình Frontend (Thiết kế layout chính, Blade, TailwindCSS). Xây dựng giao diện (UI/UX) phía Người dùng (Client). Chức năng: Trang chủ, Trang chi tiết bài viết, Trang chuyên mục. Hỗ trợ viết báo cáo (Chương 4 - Giao diện người dùng). 	100%
2001225676	Lê Đức Trung	<ul style="list-style-type: none"> Lập trình Backend (Controllers: Post, Category). Tích hợp API tải ảnh (ImgBB Service). - Xây dựng toàn bộ Giao diện và Logic trang Quản trị (Admin). Chức năng: Dashboard, CRUD Chuyên mục, CRUD Bài viết. 	100%
2001224227	Nguyễn Chí Tài	<ul style="list-style-type: none"> Lập trình Backend (Controllers: Comment, User, Profile). Xây dựng chức năng Tìm kiếm (SearchService) và Bình luận. Xây dựng trang Quản lý Người dùng, Quản lý Bình luận (Admin). Chức năng: Trang Profile (Hồ sơ cá nhân) người dùng. Kiểm thử (Testing) toàn bộ hệ thống, rà soát lỗi. 	100%

MỤC LỤC

CHƯƠNG 1. TỔNG QUAN VỀ DỰ ÁN	1
1.1. Mục tiêu & Phạm vi	1
1.1.1. Mục tiêu dự án	1
1.1.2. Phạm vi dự án	2
1.2. Cấu trúc & Kiến trúc	2
1.2.1. Mô hình kiến trúc	2
1.2.2. Cấu trúc thư mục dự án	3
1.3. Chức năng chính	5
1.3.1. Chức năng cho Người dùng vắng lai (Guest)	5
1.3.2. Chức năng cho Người dùng đã đăng nhập (Authenticated User)	6
1.3.3. Chức năng cho Quản trị viên (Admin)	6
CHƯƠNG 2. PHÂN TÍCH & THIẾT KẾ HỆ THỐNG	8
2.1. Thiết kế Cơ sở dữ liệu	8
2.1.1. Sơ đồ quan hệ thực thể (ERD)	8
2.1.2. Bảng users	8
2.1.3. Bảng categories	9
2.1.4. Bảng posts	9
2.1.5. Bảng comments	10
2.1.6. Bảng post_images	11
2.2. Usecase và luồng hoạt động	12
2.2.1. Usecase hệ thống	12
2.2.2. Luồng hoạt động (Tạo bài viết)	13

CHƯƠNG 3. TRIỂN KHAI KỸ THUẬT.....	15
3.1. Các công nghệ & thư viện chính.....	15
3.2. Triển khai các yêu cầu phi chức năng.....	16
3.2.1. Xác thực & Phân quyền	16
3.2.2. Tối ưu hiệu suất.....	17
3.2.3. Tương tác bên ngoài.....	18
3.3. Bảo mật	18
CHƯƠNG 4. GIAO DIỆN NGƯỜI DÙNG	20
4.1. Luồng chức năng phía người dùng	20
4.1.1. Trang chủ (Home).....	20
4.1.2. Trang Đăng nhập & Đăng ký.....	22
4.1.3. Trang chi tiết bài viết	23
4.1.4. Trang chuyên mục và Trang tìm kiếm.....	26
4.1.5. Trang quản lý thông tin cá nhân (Profile).....	28
4.2. Luồng chức năng phía Quản trị viên (Admin).....	30
4.2.1. Trang Dashboard (Tổng quan).....	30
4.2.2. Trang quản lý Chuyên mục (CRUD)	31
4.2.3. Trang quản lý Bài viết (CRUD & Duyệt bài).....	32
4.2.4. Trang quản lý Người dùng và Bình luận	35
KẾT LUẬN & HƯỚNG PHÁT TRIỂN	37

DANH MỤC HÌNH

Hình 1. Sơ đồ minh họa luồng xử lý của mô hình MVC trong dự án	3
Hình 2. Sơ đồ cây thư mục (tree diagram) của dự án	4
Hình 3. Sơ đồ quan hệ thực thể (ERD) của CSDL	8
Hình 4. Usecase hệ thống.....	12
Hình 5. Sơ đồ hoạt động (Tạo bài viết).....	13
Hình 6. Màn hình Trang chủ (Home)	21
Hình 7. Màn hình Trang đăng nhập	22
Hình 8. Màn hình Trang đăng kí.....	23
Hình 9. Màn hình Trang chi tiết bài viết.....	25
Hình 10. Màn hình Trang chuyên mục	26
Hình 11. Màn hình Trang tìm kiếm	27
Hình 12. Màn hình Trang hồ sơ cá nhân.....	28
Hình 13. Màn hình Trang chỉnh sửa hồ sơ.....	29
Hình 14. Màn hình Trang cài đặt	30
Hình 15. Màn hình Trang Dashboard (Admin)	31
Hình 16. Màn hình Trang Quản lý chuyên mục	32
Hình 17. Màn hình Trang Quản lý bài viết.....	33
Hình 18. Màn hình Trang tạo/chỉnh sửa bài viết	34
Hình 19. Màn hình Trang Quản lý người dùng	35
Hình 20. Màn hình Trang Quản lý bình luận.....	36

DANH MỤC BẢNG

Bảng 1. Mô tả các thư mục chính trong dự án.....	5
Bảng 2. Danh sách công nghệ và thư viện chính.....	16

CHƯƠNG 1. TỔNG QUAN VỀ DỰ ÁN

1.1. Mục tiêu & Phạm vi

1.1.1. Mục tiêu dự án

Dự án tập trung vào việc phát triển một ứng dụng web tin tức hoàn chỉnh, đáp ứng các mục tiêu chính sau:



- **Xây dựng hệ thống tin tức phía người dùng:** Cung cấp giao diện trực quan, thân thiện, cho phép người dùng đọc tin, xem tin theo chuyên mục, tìm kiếm bài viết và tương tác qua lại (bình luận).
- **Phát triển hệ thống quản trị nội dung (CMS):** Cung cấp cho Quản trị viên (Admin) một trang quản trị (Dashboard) mạnh mẽ để thực hiện các nghiệp vụ quản lý bài viết, chuyên mục, người dùng và các nội dung liên quan.
- **Đảm bảo tính linh hoạt và an toàn:** Xây dựng hệ thống xác thực người dùng an toàn, hỗ trợ đăng nhập qua mạng xã hội (Google), và áp dụng các biện pháp bảo mật cơ bản để bảo vệ dữ liệu.

- **Tối ưu trải nghiệm:** Sử dụng các công nghệ hiện đại (Laravel, TailwindCSS) để đảm bảo website có hiệu suất tốt, giao diện đáp ứng (responsive) tốt trên nhiều thiết bị.

1.1.2. Phạm vi dự án

Dự án tập trung triển khai các nhóm chức năng cốt lõi sau:

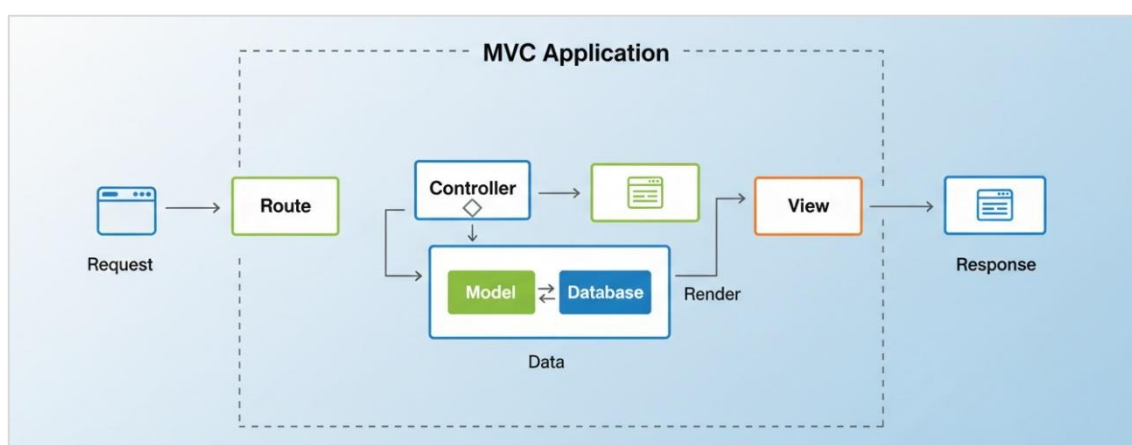
- **Các chức năng đã thực hiện:**
 - **Xác thực:** Đăng ký tài khoản, Đăng nhập (bằng email/mật khẩu và Google), Đăng xuất, Quên mật khẩu.
 - **Người dùng (Client):** Xem trang chủ, xem bài viết theo chuyên mục, xem chi tiết bài viết, tìm kiếm bài viết, bình luận (thêm/sửa/xóa bình luận của mình), quản lý thông tin hồ sơ cá nhân (cập nhật thông tin, đổi mật khẩu).
 - **Quản trị viên (Admin):** Truy cập trang Dashboard thống kê, Quản lý CRUD (Thêm, Sửa, Xóa, Xem) cho Chuyên mục, Quản lý CRUD cho Bài viết (bao gồm cả việc duyệt bài), Quản lý Người dùng, và Quản lý Bình luận.
- **Các chức năng nằm ngoài phạm vi (Hạn chế):**
 - Hệ thống phân quyền chi tiết (ví dụ: các vai trò như Biên tập viên, Cộng tác viên).
 - Các tính năng cá nhân hóa tin tức nâng cao (gợi ý tin tức dựa trên hành vi).
 - Hệ thống thông báo (notification) thời gian thực.
 - Tối ưu SEO (Search Engine Optimization) chuyên sâu.

1.2. Cấu trúc & Kiến trúc

1.2.1. Mô hình kiến trúc

Dự án được phát triển dựa trên mô hình kiến trúc MVC (Model - View - Controller), đây là kiến trúc tiêu chuẩn được áp dụng bởi framework Laravel:

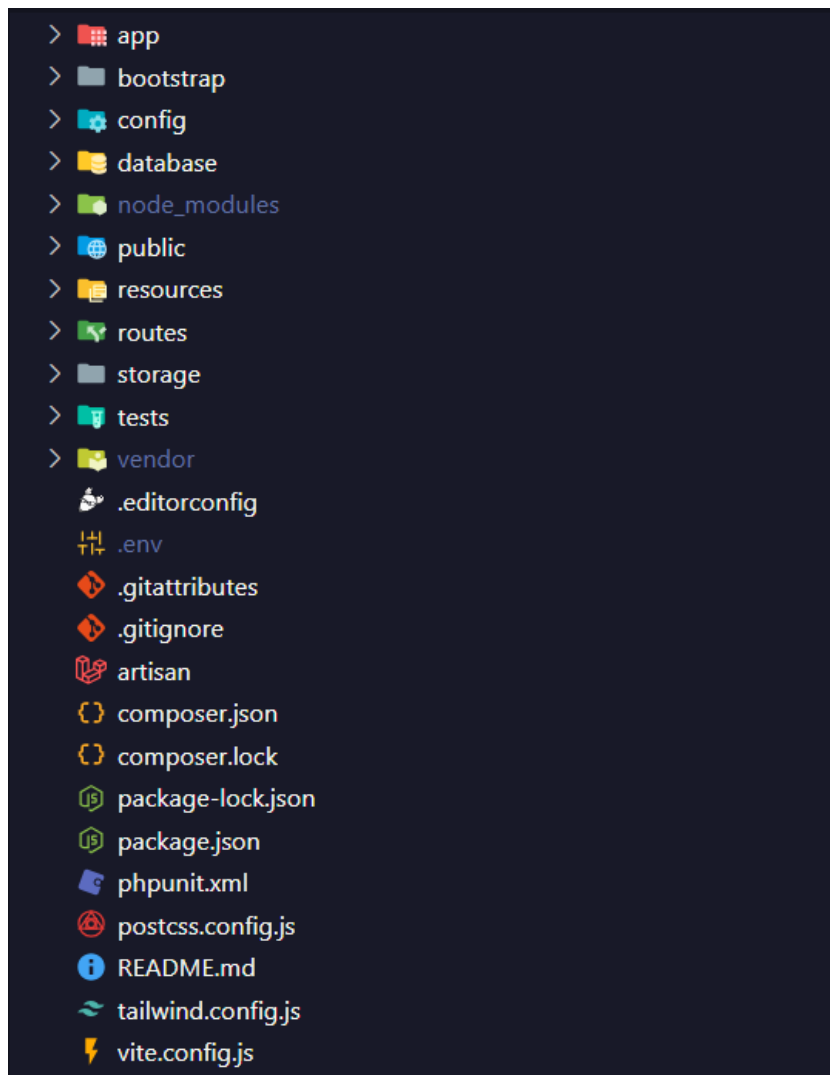
- **Model:** Đại diện cho cấu trúc dữ liệu và logic nghiệp vụ. Chúng chịu trách nhiệm tương tác trực tiếp với cơ sở dữ liệu (MySQL). Trong dự án, đây là các tệp Eloquent Model nằm trong thư mục **app/Models** (ví dụ: **User.php**, **Post.php**, **Category.php**, **Comment.php**).
- **View:** Đại diện cho tầng giao diện người dùng (UI). Đây là các tệp mẫu (template) Blade nằm trong thư mục **resources/views**. Chúng có nhiệm vụ tiếp nhận dữ liệu từ Controller và hiển thị nội dung cho người dùng.
- **Controller:** Đóng vai trò trung gian, tiếp nhận yêu cầu (Request) từ người dùng (được điều hướng bởi Routes), xử lý logic nghiệp vụ (gọi đến Model để truy vấn/thao tác dữ liệu), sau đó truyền dữ liệu và trả về kết quả cho View (Response). Các Controller nằm trong **app/Http/Controllers** (ví dụ: **PostController.php**, **AdminController.php**).



Hình 1. Sơ đồ minh họa luồng xử lý của mô hình MVC trong dự án

1.2.2. Cấu trúc thư mục dự án

Dự án tuân thủ cấu trúc thư mục tiêu chuẩn của framework Laravel. Cấu trúc này giúp tổ chức mã nguồn một cách rõ ràng, tách bạch các thành phần và logic của ứng dụng.



Hình 2. Sơ đồ cây thư mục (tree diagram) của dự án

Thư mục	Vai trò và Chức năng chính
app	Thư mục cốt lõi, chứa toàn bộ logic nghiệp vụ của ứng dụng.
app/Http/Controllers	Chứa các Controller xử lý yêu cầu HTTP.
app/Http/Middleware	Chứa các lớp trung gian (ví dụ: AdminMiddleware.php để kiểm tra quyền admin).
app/Models	Chứa các Eloquent Model định nghĩa cấu trúc bảng và quan hệ CSDL.
app/Services	Chứa các lớp dịch vụ (ví dụ: ImgBBSservice.php, SearchService.php).

config	Chứa toàn bộ các tệp cấu hình của ứng dụng (CSDL, xác thực, dịch vụ...).
database	Chứa các tệp di chuyển (migrations), dữ liệu mẫu (seeders) và factories.
public	Thư mục gốc (document root) của máy chủ web. Chứa tệp index.php (entry point) và các tài sản (assets) đã được biên dịch (CSS, JS).
resources	Chứa các tài sản gốc (chưa biên dịch) và các tệp View.
resources/views	Chứa toàn bộ các tệp Blade template (giao diện) của ứng dụng.
routes	Chứa các tệp định nghĩa tuyến đường (URL) của ứng dụng (web.php, auth.php).
storage	Chứa các tệp do framework tạo ra (logs, cache, sessions).
.env	Tệp cấu hình biến môi trường, chứa các thông tin nhạy cảm (API keys, CSDL...).

Bảng 1. Mô tả các thư mục chính trong dự án

1.3. Chức năng chính

1.3.1. Chức năng cho Người dùng vắng lai (Guest)

- **Xem tin tức:**
 - Xem danh sách các bài viết mới nhất, bài viết nổi bật tại Trang chủ.
 - Xem chi tiết nội dung, hình ảnh, tác giả của một bài viết.
- **Xem theo chuyên mục:**
 - Điều hướng đến trang danh sách các bài viết thuộc một chuyên mục cụ thể.
- **Tìm kiếm:**
 - Sử dụng thanh tìm kiếm để tìm các bài viết dựa trên từ khóa.
- **Xác thực:**

- Truy cập các trang Đăng nhập, Đăng ký và Quên mật khẩu.

1.3.2. Chức năng cho Người dùng đã đăng nhập (Authenticated User)

- Bao gồm tất cả các chức năng của Người dùng vắng lai, và thêm:
- **Bình luận:**
 - Viết bình luận mới cho một bài viết.
 - Chỉnh sửa hoặc xóa các bình luận do chính mình tạo ra.
- **Quản lý hồ sơ cá nhân:**
 - Xem và cập nhật thông tin cá nhân (tên, email, ảnh đại diện).
 - Thay đổi mật khẩu tài khoản.
- **Tương tác bài viết (Dự kiến/Tùy chọn):**
 - Người dùng có thể được cấp quyền đăng bài, các bài đăng này có thể cần chờ Quản trị viên duyệt.
- **Đăng xuất: Thoát khỏi phiên đăng nhập hiện tại.**

1.3.3. Chức năng cho Quản trị viên (Admin)

- Được bảo vệ bởi AdminMiddleware, bao gồm tất cả chức năng của người dùng đã đăng nhập, và thêm:
- **Quản lý Chuyên mục:**
 - Xem danh sách, thêm, sửa, xóa các chuyên mục tin tức.
- **Quản lý Bài viết:**
 - Xem danh sách toàn bộ bài viết trên hệ thống.
 - Tạo bài viết mới, chỉnh sửa và xóa bất kỳ bài viết nào.
 - Duyệt hoặc gỡ bài viết.
- **Quản lý Người dùng:**
 - Xem danh sách tất cả người dùng.
 - Chỉnh sửa thông tin và phân quyền (gán/gỡ quyền admin) cho người dùng.
 - Xóa tài khoản người dùng khỏi hệ thống.

- **Quản lý Bình luận:**

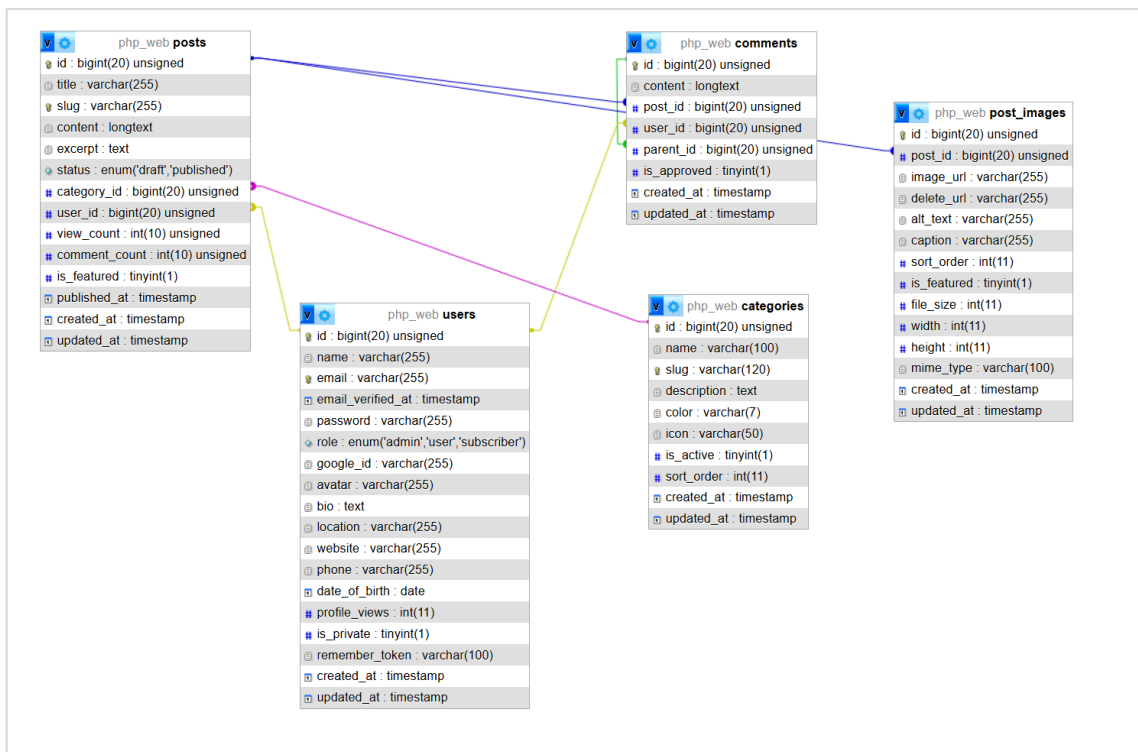
- Xem danh sách tất cả bình luận trên hệ thống.
- Xóa các bình luận vi phạm.

CHƯƠNG 2. PHÂN TÍCH & THIẾT KẾ HỆ THỐNG

2.1. Thiết kế Cơ sở dữ liệu

Hệ thống sử dụng Hệ quản trị Cơ sở dữ liệu quan hệ (RDBMS) là MySQL. Thiết kế CSDL được xây dựng dựa trên nhu cầu lưu trữ thông tin về người dùng, chuyên mục, bài viết, bình luận và các hình ảnh liên quan.

2.1.1. Sơ đồ quan hệ thực thể (ERD)



Hình 3. Sơ đồ quan hệ thực thể (ERD) của CSDL

2.1.2. Bảng users

Mục đích: Lưu trữ thông tin tài khoản của tất cả người dùng, bao gồm cả quản trị viên.

Tên trường	Kiểu dữ liệu (Ví dụ)	Mô tả
id	BIGINT (Khóa chính)	Mã định danh duy nhất cho mỗi người dùng.
name	VARCHAR	Tên hiển thị của người dùng.

email	VARCHAR (Unique)	Địa chỉ email (dùng để đăng nhập).
password	VARCHAR	Mật khẩu đã được mã hóa (băm).
google_id	VARCHAR (Nullable)	ID người dùng từ Google (dùng cho đăng nhập OAuth).
avatar	VARCHAR (Nullable)	URL ảnh đại diện của người dùng.
is_admin	BOOLEAN (Default: 0)	Cờ xác định vai trò: 1 là Admin, 0 là người dùng thường.
created_at	TIMESTAMP	Thời điểm tài khoản được tạo.

2.1.3. Bảng *categories*

Mục đích: Lưu trữ các chuyên mục (danh mục) tin tức của website.

Tên trường	Kiểu dữ liệu (Ví dụ)	Mô tả
id	BIGINT (Khóa chính)	Mã định danh duy nhất cho chuyên mục.
name	VARCHAR	Tên của chuyên mục (ví dụ: "Thể thao", "Công nghệ").
slug	VARCHAR (Unique)	Chuỗi định danh ngắn, dùng trên URL (ví dụ: "the-thao").
created_at	TIMESTAMP	Thời điểm chuyên mục được tạo.

2.1.4. Bảng *posts*

Mục đích: Lưu trữ nội dung chính của các bài viết (tin tức). Đây là bảng trung tâm của ứng dụng.

Tên trường	Kiểu dữ liệu (Ví dụ)	Mô tả
id	BIGINT (Khóa chính)	Mã định danh duy nhất cho bài viết.

user_id	BIGINT (Khóa ngoại)	Liên kết đến users(id). Xác định tác giả của bài viết.
category_id	BIGINT (Khóa ngoại)	Liên kết đến categories(id). Xác định chuyên mục của bài viết.
title	VARCHAR	Tiêu đề của bài viết.
slug	VARCHAR (Unique)	Chuỗi định danh ngắn, dùng trên URL cho bài viết.
excerpt	TEXT	Đoạn tóm tắt ngắn (mô tả) cho bài viết.
body	LONGTEXT	Nội dung chi tiết, đầy đủ của bài viết.
is_published	BOOLEAN (Default: 0)	Cờ xác định trạng thái: 1 đã được duyệt và hiển thị, 0 là bản nháp.
views_count	INT (Default: 0)	Bộ đếm số lượt xem của bài viết.
created_at	TIMESTAMP	Thời điểm bài viết được tạo/đăng.

2.1.5. Bảng comments

Mục đích: Lưu trữ các bình luận của người dùng về bài viết.

Tên trường	Kiểu dữ liệu (Ví dụ)	Mô tả
id	BIGINT (Khóa chính)	Mã định danh duy nhất cho bình luận.
user_id	BIGINT (Khóa ngoại)	Liên kết đến users(id). Xác định người viết bình luận.
post_id	BIGINT (Khóa ngoại)	Liên kết đến posts(id). Xác định bài viết được bình luận.
parent_id	BIGINT (Nullable)	Liên kết đến comments(id). Dùng để trả lời bình luận (hệ thống cha-con).
body	TEXT	Nội dung của bình luận.

created_at	TIMESTAMP	Thời điểm bình luận được gửi.
-------------------	-----------	-------------------------------

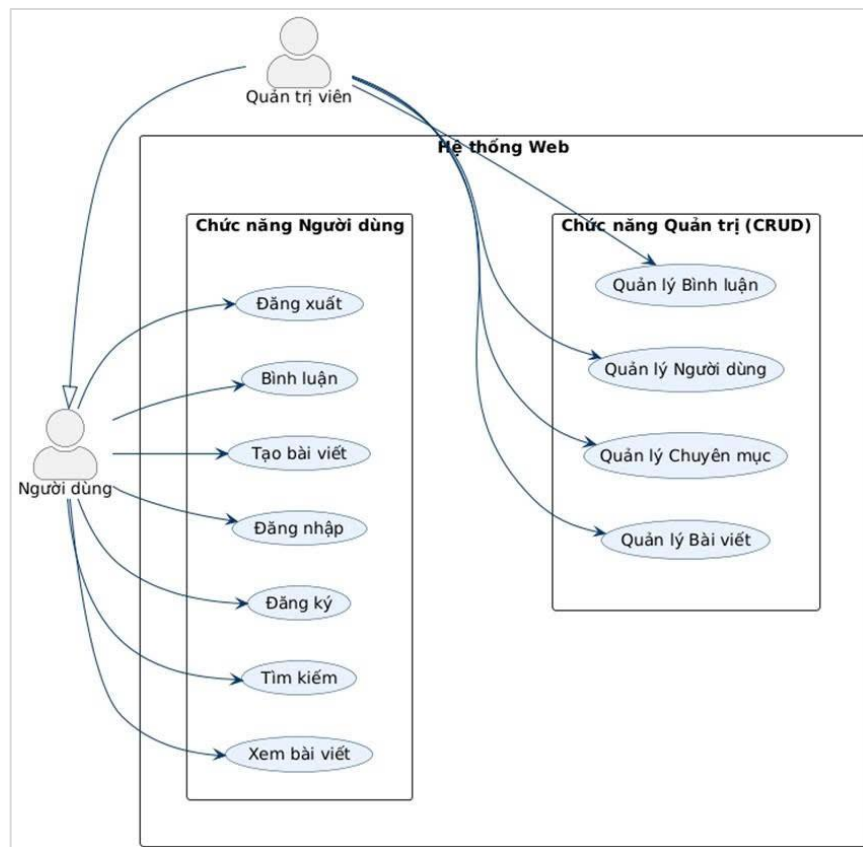
2.1.6. Bảng *post_images*

Mục đích: Lưu trữ các URL hình ảnh được tải lên và liên kết với một bài viết.

Tên trường	Kiểu dữ liệu (Ví dụ)	Mô tả
id	BIGINT (Khóa chính)	Mã định danh duy nhất cho hình ảnh.
post_id	BIGINT (Khóa ngoại)	Liên kết đến posts(id). Xác định bài viết sở hữu hình ảnh này.
image_url	VARCHAR	URL của hình ảnh (trả về từ dịch vụ ImgBB).
created_at	TIMESTAMP	Thời điểm ảnh được tải lên.

2.2. Usecase và luồng hoạt động

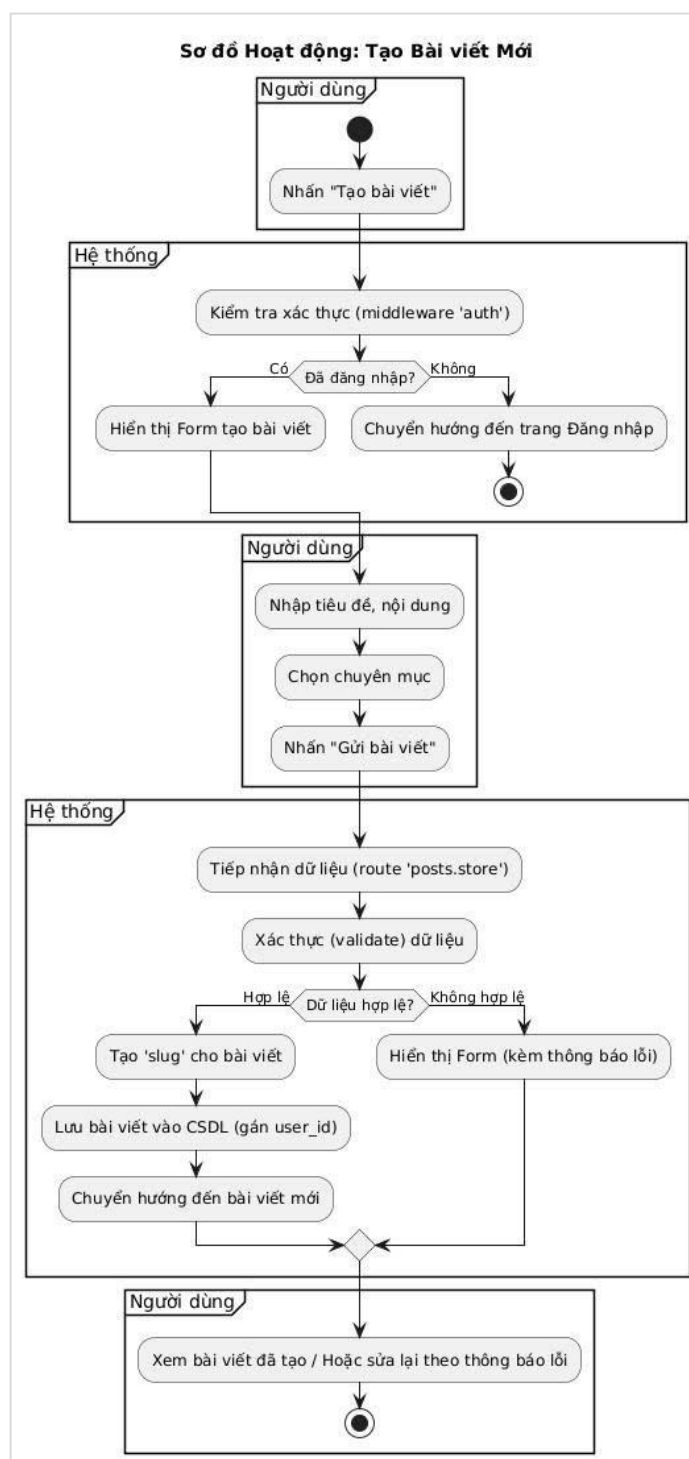
2.2.1. Usecase hệ thống



Hình 4. Usecase hệ thống

- **Người dùng (User):**
 - Các chức năng "Xem bài viết" và "Tìm kiếm"
 - Các chức năng "Đăng ký" và "Đăng nhập"
 - Các chức năng "Tạo bài viết", "Bình luận" và "Đăng xuất" với người đã đăng nhập.
- **Quản trị viên (Admin):**
 - Kế thừa tất cả các chức năng của "Người dùng".
 - Có thêm các quyền quản lý toàn bộ hệ thống như quản lý bài viết, chuyên mục, người dùng và bình luận.

2.2.2. Luồng hoạt động (Tạo bài viết)



Hình 5. Sơ đồ hoạt động (Tạo bài viết)

1. Người dùng bắt đầu bằng cách nhấn "Tạo bài viết".
2. Hệ thống (thông qua middleware auth) kiểm tra xem người dùng đã đăng nhập chưa.

3. Nếu chưa: Sẽ bị chuyển hướng đến trang Đăng nhập và luồng kết thúc.
4. Nếu rồi: Hệ thống hiển thị Form tạo bài viết .
5. Người dùng điền thông tin và nhấn "Gửi bài viết".
6. Hệ thống nhận dữ liệu và tiến hành xác thực các trường như tiêu đề, nội dung...
7. Nếu không hợp lệ: Hệ thống hiển thị lại Form kèm thông báo lỗi. Người dùng phải sửa lại.
8. Nếu hợp lệ: Hệ thống xử lý (tạo slug, gán user_id của người đang đăng nhập) và lưu vào cơ sở dữ liệu.
9. Cuối cùng, hệ thống chuyển hướng người dùng đến trang xem bài viết họ vừa tạo. Luồng kết thúc thành công.

CHƯƠNG 3. TRIỂN KHAI KỸ THUẬT

3.1. Các công nghệ & thư viện chính

Việc lựa chọn nền tảng công nghệ phù hợp là yếu tố then chốt, ảnh hưởng trực tiếp đến tốc độ phát triển, khả năng bảo trì và hiệu suất của dự án.

Hạng mục	Công nghệ / Thư viện	Lý do lựa chọn (Mục đích)
Backend (Ngôn ngữ)	PHP (Phiên bản 8.2+)	Ngôn ngữ kịch bản phía máy chủ phổ biến, ổn định, cộng đồng lớn và hệ sinh thái thư viện phong phú.
Backend (Framework)	Laravel (Phiên bản 11)	Framework PHP theo kiến trúc MVC hàng đầu, cung cấp cấu trúc rõ ràng, nhiều tính năng tích hợp sẵn (ORM, Routing, Auth), giúp tăng tốc độ phát triển.
Cơ sở dữ liệu	MySQL	Hệ quản trị CSDL quan hệ (RDBMS) phổ biến nhất, đáng tin cậy, và tương thích hoàn toàn với Laravel.
Frontend (Styling)	TailwindCSS	Framework CSS dạng "utility-first", cho phép xây dựng giao diện tùy chỉnh nhanh chóng mà không cần rời khỏi HTML, giúp tạo ra thiết kế đồng nhất.
Frontend (Templating)	Blade	Hệ thống template engine tích hợp sẵn của Laravel. Cung cấp cú pháp gọn gàng và khả năng tái sử dụng component (ví dụ: layout, partials).
Frontend (JS)	Vite, Alpine.js	Vite được dùng để biên dịch và đóng gói (bundle) tài sản (JS, CSS) cực nhanh. Alpine.js là thư viện JS gọn nhẹ, cung cấp tính tương tác cho giao diện.
Xác thực	Laravel Socialite	Thư viện chính thức của Laravel, giúp đơn giản hóa quá trình tích hợp đăng

		nhập với các nhà cung cấp OAuth (như Google).
Tương tác API	GuzzleHttp	Thư viện HTTP client, được sử dụng để thực hiện các cuộc gọi API đến dịch vụ bên thứ ba một cách dễ dàng.
Hiệu ứng (UI)	Lottie-web	Thư viện render các file animation (từ Adobe After Effects) dưới dạng JSON, giúp tạo các hiệu ứng động mượt mà (ví dụ: splash screen).

Bảng 2. Danh sách công nghệ và thư viện chính

3.2. Triển khai các yêu cầu phi chức năng

3.2.1. Xác thực & Phân quyền

Một hệ thống tin tức cần cơ chế quản lý danh tính và quyền hạn người dùng một cách rõ ràng.

- **Quản lý phiên (Session-based Authentication):** Dự án sử dụng hệ thống xác thực tích hợp sẵn của Laravel (Laravel Auth). Khi người dùng đăng nhập thành công (`LoginController.php`), một phiên (session) bảo mật được tạo ra và lưu trữ phía máy chủ. Cookie của trình duyệt sẽ được dùng để định danh phiên này cho các yêu cầu tiếp theo, đảm bảo người dùng giữ trạng thái đăng nhập.
- **Đăng nhập qua Mạng xã hội (Google OAuth):** Để tăng tính tiện dụng, hệ thống tích hợp đăng nhập bằng Google thông qua thư viện Laravel Socialite. Luồng hoạt động (`GoogleController.php`):
 1. Người dùng chọn "Đăng nhập bằng Google".
 2. Hệ thống chuyển hướng người dùng đến trang xác thực của Google.
 3. Sau khi xác thực thành công, Google gọi lại (callback) về ứng dụng kèm theo thông tin người dùng.
 4. Hệ thống kiểm tra email trong CSDL:
 - Nếu email đã tồn tại, người dùng được đăng nhập.

- Nếu chưa, một tài khoản mới sẽ được tự động tạo và người dùng được đăng nhập.
- **Phân quyền (Authorization):** Hệ thống phân quyền dựa trên vai trò (Role-based) một cách đơn giản thông qua trường `is_admin` trong bảng `users`:
 - **Middleware:** Lớp `AdminMiddleware.php` được áp dụng cho tất cả các tuyến đường (route) của trang quản trị. Middleware này kiểm tra xem người dùng đã đăng nhập và có cờ `is_admin = 1` hay không. Nếu không, người dùng sẽ bị từ chối truy cập (HTTP 403 Forbidden).
 - **Policies:** Hệ thống sử dụng Policy (`CommentPolicy.php`) để kiểm soát các hành động chi tiết hơn. Ví dụ, một người dùng chỉ có thể "cập nhật" hoặc "xóa" bình luận nếu họ là chủ sở hữu của bình luận đó, hoặc nếu họ là Admin.

3.2.2. Tối ưu hiệu suất

- **Biên dịch tài sản (Asset Compilation):** Dự án sử dụng Vite (`vite.config.js`) thay vì Webpack truyền thống. Vite cung cấp khả năng khởi động máy chủ phát triển (dev server) gần như ngay lập tức và biên dịch tài sản (build) cho production cực kỳ nhanh, giúp giảm thời gian tải trang.
- **Tải ảnh không đồng bộ (Offloading):** Thay vì lưu trữ hình ảnh trực tiếp trên máy chủ của ứng dụng, hệ thống tích hợp dịch vụ lưu trữ ảnh bên thứ ba (ImgBB). Điều này mang lại hai lợi ích lớn:
 1. Giảm tải băng thông và dung lượng lưu trữ cho máy chủ chính.
 2. Hình ảnh được phân phối qua mạng lưới của dịch vụ (CDN), giúp tăng tốc độ tải ảnh cho người dùng.

3.2.3. Tương tác bên ngoài

Hệ thống kết nối với hai dịch vụ API bên ngoài chính:

1. **Google OAuth API (Qua Laravel Socialite):** dùng để xác thực người dùng.
2. **ImgBB API:**
 - Một lớp dịch vụ chuyên biệt (`app/Services/ImgBBService.php`) được tạo ra để xử lý việc tải ảnh lên ImgBB.
 - Khi Admin tạo hoặc cập nhật bài viết, hình ảnh được tải lên sẽ được gửi đến API của ImgBB thông qua thư viện GuzzleHttp.
 - ImgBB API trả về một URL của hình ảnh đã được lưu trữ.
 - Hệ thống chỉ lưu trữ URL này (trong bảng `post_images`) vào cơ sở dữ liệu, thay vì lưu trữ tệp tin ảnh.

3.3. Bảo mật

Dự án đã áp dụng các biện pháp bảo mật tiêu chuẩn được cung cấp bởi framework Laravel để phòng chống các lỗ hổng phổ biến:

- **Chống SQL Injection:**
 - Giải pháp: Sử dụng Eloquent ORM của Laravel.
 - Mô tả: Mọi tương tác với CSDL (truy vấn, thêm, sửa, xóa) đều được thực hiện thông qua các phương thức của Eloquent (ví dụ: `Post::find($id)`, `User::create([...])`). Eloquent sử dụng cơ chế "parameter binding" (ràng buộc tham số), đảm bảo rằng mọi dữ liệu đầu vào từ người dùng đều được coi là văn bản (string) chứ không phải là một phần của câu lệnh SQL. Điều này vô hiệu hóa hoàn toàn nguy cơ tấn công SQL Injection.
- **Chống Cross-Site Scripting (XSS):**

- Giải pháp: Cơ chế thoát (escaping) tự động của Blade Template Engine.
- Mô tả: Khi hiển thị dữ liệu ra giao diện, dự án sử dụng cú pháp `{{ $variable }}` của Blade. Cú pháp này tự động chuyển đổi các ký tự HTML đặc biệt (như `<script>`, `>`, `&`) thành các thực thể HTML an toàn (ví dụ: `<script>`). Điều này ngăn chặn trình duyệt thực thi bất kỳ mã JavaScript độc hại nào do người dùng nhập vào (ví dụ: trong tên người dùng, nội dung bình luận).
- **Chống Cross-Site Request Forgery (CSRF):**
 - Giải pháp: CSRF Token tích hợp sẵn của Laravel.
 - Mô tả: Mọi biểu mẫu (form) thực hiện các hành động thay đổi dữ liệu (POST, PUT, DELETE) trong ứng dụng đều bắt buộc phải kèm theo một mã token ẩn (được sinh ra bởi directive `@csrf` trong Blade). Middleware `VerifyCsrfToken` của Laravel sẽ tự động kiểm tra tính hợp lệ của token này. Nếu token không khớp hoặc không tồn tại, yêu cầu sẽ bị từ chối. Điều này đảm bảo rằng các yêu cầu thay đổi dữ liệu phải đến từ chính website của ứng dụng, chứ không phải từ một trang web độc hại bên ngoài.
- **Bảo mật mật khẩu:**
 - Giải pháp: Hashing bằng Bcrypt.
 - Mô tả: Khi người dùng đăng ký (`RegisterController.php`) hoặc đổi mật khẩu, mật khẩu gốc không bao giờ được lưu vào CSDL. Thay vào đó, hệ thống sử dụng `Hash::make()` (Bcrypt) để tạo ra một chuỗi băm (hash) một chiều, không thể giải mã ngược. Khi đăng nhập, mật khẩu người dùng nhập vào sẽ được băm và so sánh với chuỗi băm trong CSDL.

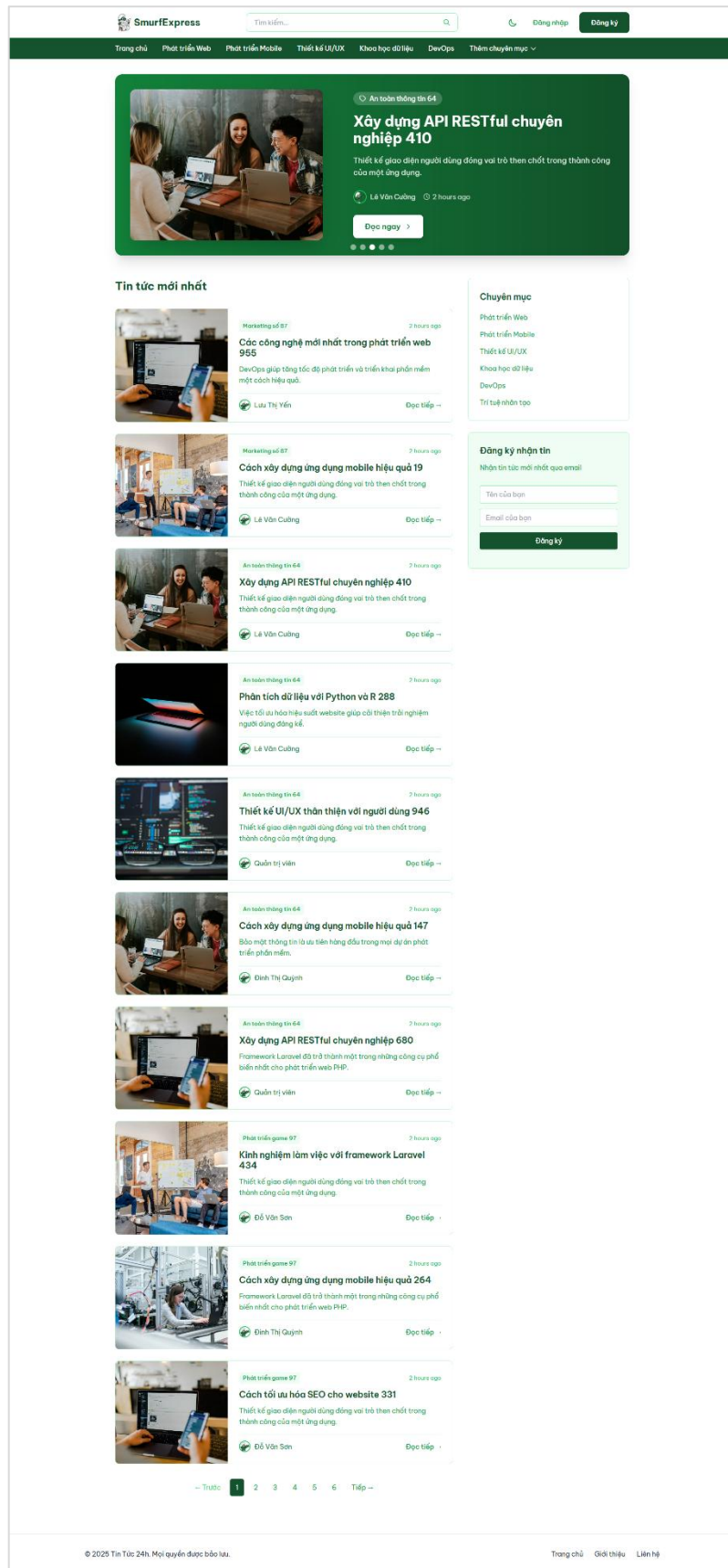
CHƯƠNG 4. GIAO DIỆN NGƯỜI DÙNG

4.1. Luồng chức năng phía người dùng

Đây là các màn hình mà người dùng cuối (khách truy cập, người đọc) sẽ tương tác.

4.1.1. Trang chủ (Home)

- **Mô tả:** Là trang đầu tiên người dùng nhìn thấy. Trang này hiển thị một thanh điều hướng (navbar) rõ ràng, một khu vực "hero" (carousel) cho các bài viết nổi bật, và một lưới (grid) các bài viết mới nhất được phân trang.
- **Tệp View:** `home.blade.php`



Hình 6. Màn hình Trang chủ (Home)

4.1.2. Trang Đăng nhập & Đăng ký

- **Mô tả:** Cung cấp các biểu mẫu (form) cho phép người dùng đăng nhập vào hệ thống bằng email/mật khẩu, hoặc đăng ký một tài khoản mới. Trang đăng nhập cũng tích hợp nổi bật nút "Đăng nhập bằng Google" để xác thực qua OAuth.
- **Tập View:** `auth/login.blade.php`, `auth/register.blade.php`

The screenshot shows the SmurfExpress login and registration page. The header includes the SmurfExpress logo, a search bar, and navigation links: Trang chủ, Phát triển Web, Phát triển Mobile, Thiết kế UI/UX, Khoa học dữ liệu, DevOps, and Thêm chuyên mục. The main content area features a green background with a central white box containing the login and registration forms. The login form includes fields for email and password, a 'Remember me' checkbox, and a 'Forgot password?' link. The registration form includes a 'Sign up with Google' button. The footer contains copyright information: © 2025 Tin Tức 24h. Mọi quyền được bảo lưu. and links to Trang chủ, Giới thiệu, and Liên hệ.

Hình 7. Màn hình Trang đăng nhập

The screenshot displays the 'Tạo tài khoản mới' (Create new account) page on the SmurfExpress website. The page features a dark green header with the site logo, a search bar, and navigation links. The main content area is a light green gradient with a central white form. The form has a title 'Tạo tài khoản mới' and a subtitle 'Tham gia cộng đồng và khám phá nội dung tuyệt vời'. It contains several input fields: 'Họ và tên' (Name), 'Địa chỉ email' (Email), 'Mật khẩu' (Password), and 'Xác nhận mật khẩu' (Confirm password). There is also a checkbox for 'Tôi đồng ý với Điều khoản sử dụng và Chính sách bảo mật' (I agree to the Terms of Use and Privacy Policy). Below the form, there is a link to 'Đăng nhập ngay' (Log in now) and a copyright notice '© 2025 Tin Tức 24h. Mọi quyền được bảo lưu.' (© 2025 Tin Tức 24h. All rights reserved.).

Hình 8. Màn hình Trang đăng kí

4.1.3. Trang chi tiết bài viết

- **Mô tả:** Đây là nơi hiển thị nội dung đầy đủ của một bài viết, bao gồm: Tiêu đề, thông tin tác giả, ngày đăng, nội dung chi tiết (body), và các hình ảnh liên quan đến bài viết.
- **Hệ thống bình luận:** Ngay bên dưới nội dung bài viết là khu vực bình luận tương tác:
 - Hiển thị danh sách các bình luận đã có (hỗ trợ phân cấp trả lời).
 - Nếu đã đăng nhập, người dùng sẽ thấy biểu mẫu để gửi bình luận mới.

- Người dùng có thể chỉnh sửa/xóa bình luận của chính mình.
- **Tệp View:** `posts/show.blade.php`

SmurfExpress

Đăng nhập

Đăng ký

Trang chủ

Phát triển Web

Phát triển Mobile

Thiết kế UI/UX

Khoa học dữ liệu

DevOps

Thêm chuyên mục

Trang chủ > An toàn thông tin 64 > Xây dựng API RESTful chuyên nghiệp 410

An toàn thông tin 64

Xây dựng API RESTful chuyên nghiệp 410

Lê Văn Cường

Tác giả

22/10/2025 13:18

•

3070 lượt xem

•

4 bình luận

1 phút đọc

Thiết kế giao diện người dùng đóng vai trò then chốt trong thành công của một ứng dụng.

Trong thời đại công nghệ số hiện nay, việc nắm vững các kỹ năng lập trình trở nên vô cùng quan trọng. DevOps giúp tăng tốc độ phát triển và triển khai phần mềm một cách hiệu quả. Để trở thành một lập trình viên giỏi, bạn cần không ngừng học hỏi và cập nhật kiến thức mới. Bảo mật thông tin là ưu tiên hàng đầu trong mọi dự án phát triển phần mềm.

Bài viết được đăng lúc 13:18, 22/10/2025

Chia sẻ

Bình luận (4)

Bạn cần đăng nhập để bình luận

Đăng nhập hoặc tạo tài khoản để tham gia thảo luận

Đinh Thị Quỳnh

22/10/2025 13:18

Bài viết rất hay và bổ ích! Cảm ơn tác giả đã chia sẻ.

C

Cao Văn Bảo

22/10/2025 13:18

Có source code mẫu không bạn?

Đỗ Văn Sơn

22/10/2025 13:18

Rất hữu ích cho người mới bắt đầu như mình.

V

Vũ Thị Giang

22/10/2025 13:18

Bạn có thể làm thêm video hướng dẫn không?

Tin tức liên quan

An toàn thông tin 64

2 hours ago

Phân tích dữ liệu với Python và R 288

Việc tối ưu hóa hiệu suất website giúp cải thiện trải nghiệm người dùng đáng kể.

L

Lê Văn Cường

Đọc tiếp

An toàn thông tin 64

2 hours ago

Thiết kế UI/UX thân thiện với người dùng 946

Thiết kế giao diện người dùng đóng vai trò then chốt trong thành công của một ứng...

Q

Quản trị viên

Đọc tiếp

An toàn thông tin 64

2 hours ago

Cách xây dựng ứng dụng mobile hiệu quả 147

Bảo mật thông tin là ưu tiên hàng đầu trong mọi dự án phát triển phần mềm.

Đinh Thị Quỳnh

Đọc tiếp

© 2025 Tin Tức 24h. Mọi quyền được bảo lưu.

Trang chủ

Giới thiệu

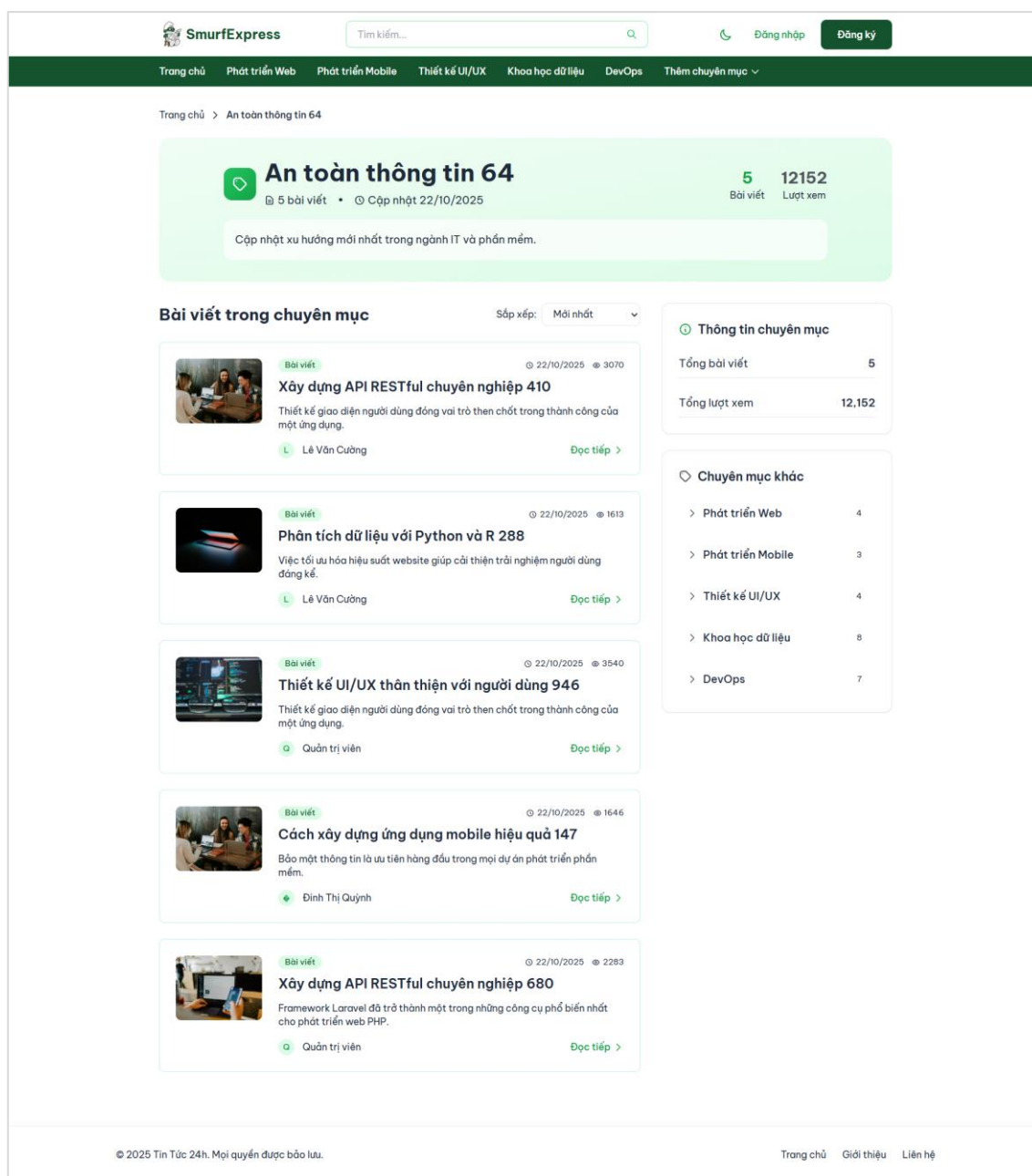
Liên hệ

Hình 9. Màn hình Trang chi tiết bài viết

25

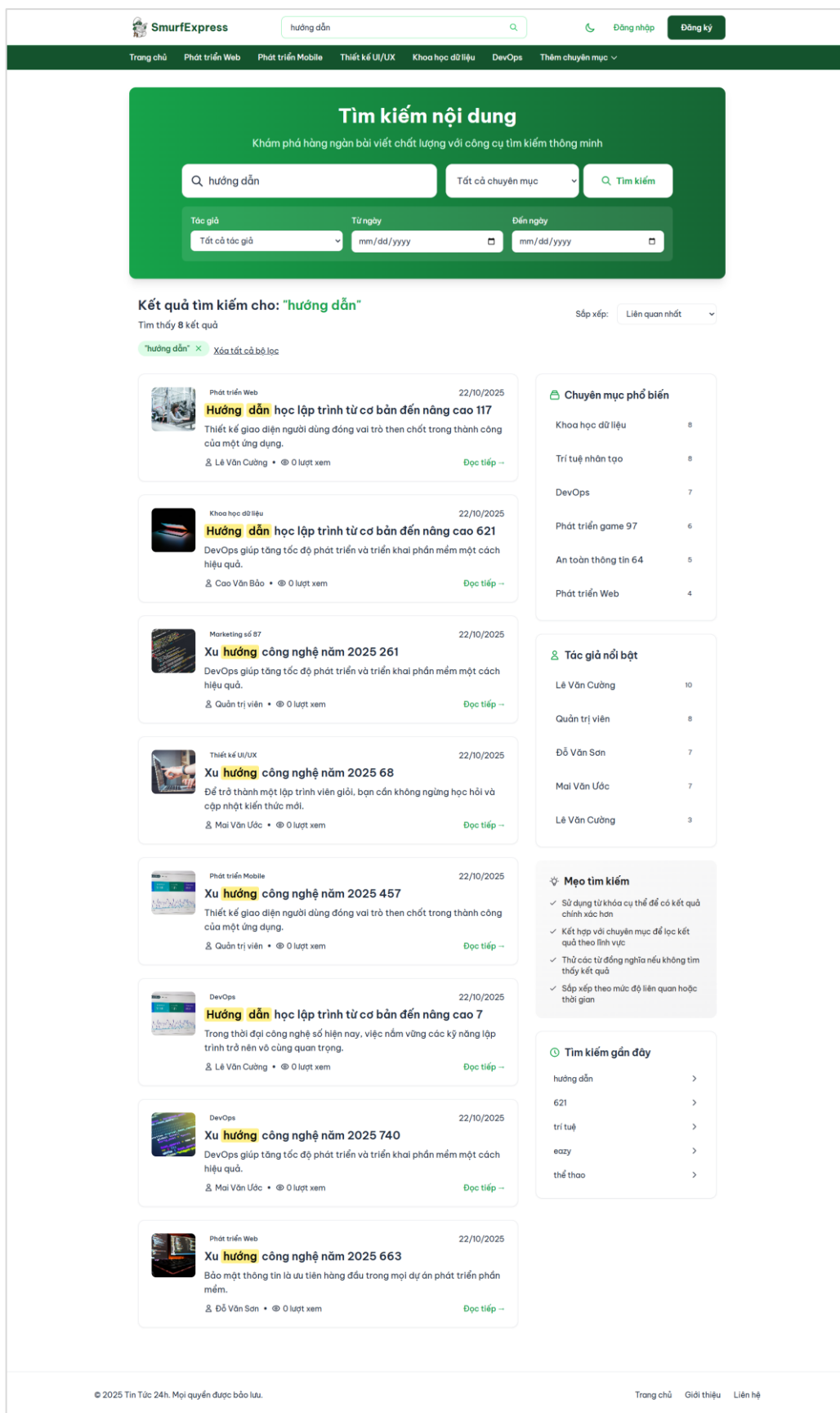
4.1.4. Trang chuyên mục và Trang tìm kiếm

- **Trang chuyên mục:** Hiển thị danh sách tất cả các bài viết thuộc một chuyên mục cụ thể mà người dùng đã chọn. (Tập View: `categories/show.blade.php`).



Hình 10. Màn hình Trang chuyên mục

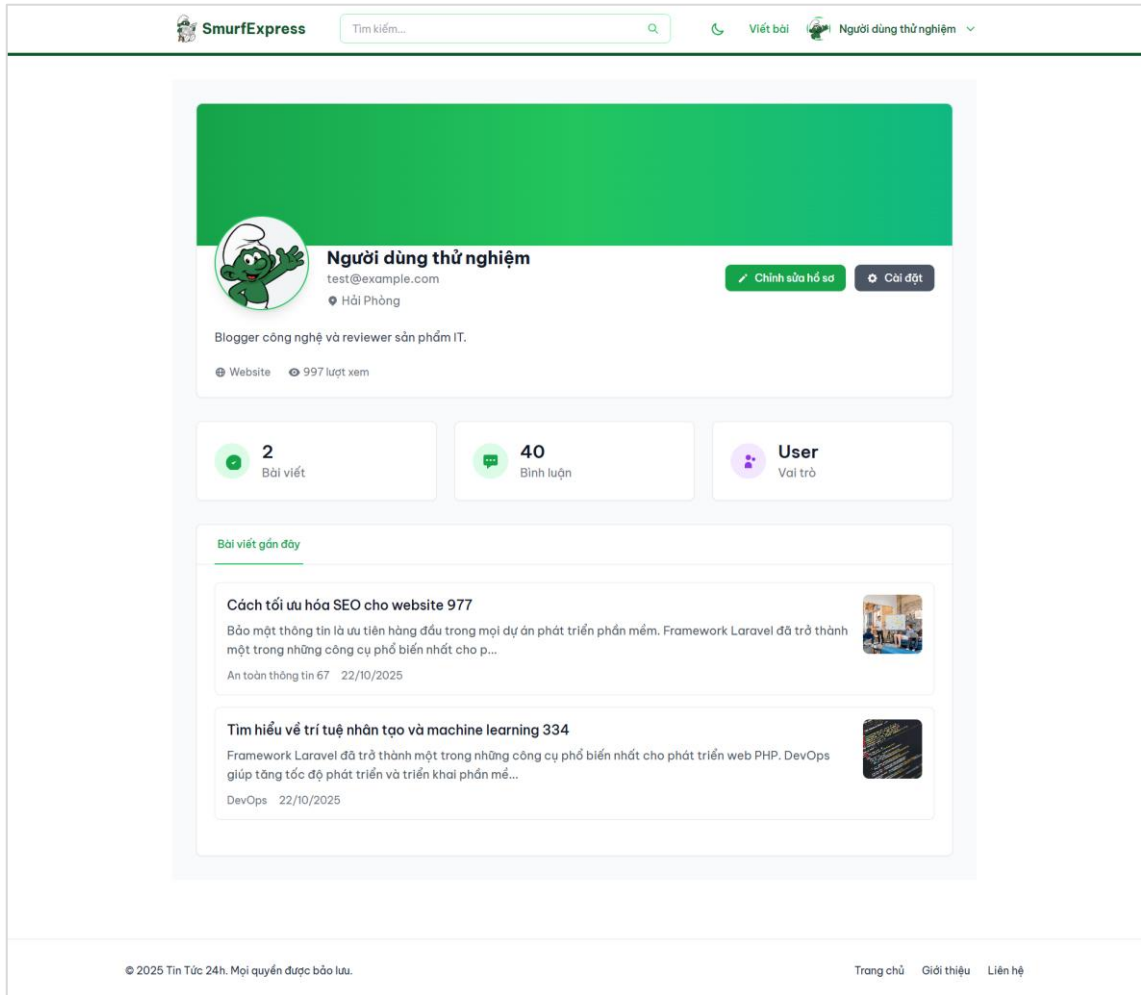
- **Trang tìm kiếm:** Hiển thị kết quả tìm kiếm dựa trên từ khóa người dùng nhập vào. (Tập View: `search/index.blade.php`).



Hình 11. Màn hình Trang tìm kiếm


4.1.5. Trang quản lý thông tin cá nhân (Profile)


- **Mô tả:** Sau khi đăng nhập, người dùng có thể truy cập trang cá nhân của mình để:
- Xem hồ sơ công khai (Tập View: `profile/show.blade.php`).






Hình 12. Màn hình Trang hồ sơ cá nhân

- Cập nhật thông tin cá nhân (tên, email, ảnh đại diện) và đổi mật khẩu (Tập View: `profile/edit.blade.php`, `profile/settings.blade.php`).

 SmurfExpress

 [Viết bài](#)


 [Người dùng thử nghiệm](#) 

[Trang chủ](#) [Phát triển Web](#) [Phát triển Mobile](#) [Thiết kế UI/UX](#) [Khoa học dữ liệu](#) [DevOps](#) [Thêm chuyên mục](#) 

Chỉnh sửa hồ sơ

Cập nhật thông tin cá nhân của bạn

Ảnh đại diện



[Chọn ảnh mới](#)

Thông tin cá nhân

Họ và tên

Email

Email không thể thay đổi

Tiểu sử


Tối đa 500 ký tự

Địa điểm

Website

Số điện thoại

Ngày sinh



Cài đặt quyền riêng tư

☒ Đặt hồ sơ ở chế độ riêng tư

Khi bật tùy chọn này, chỉ bạn mới có thể xem hồ sơ của mình

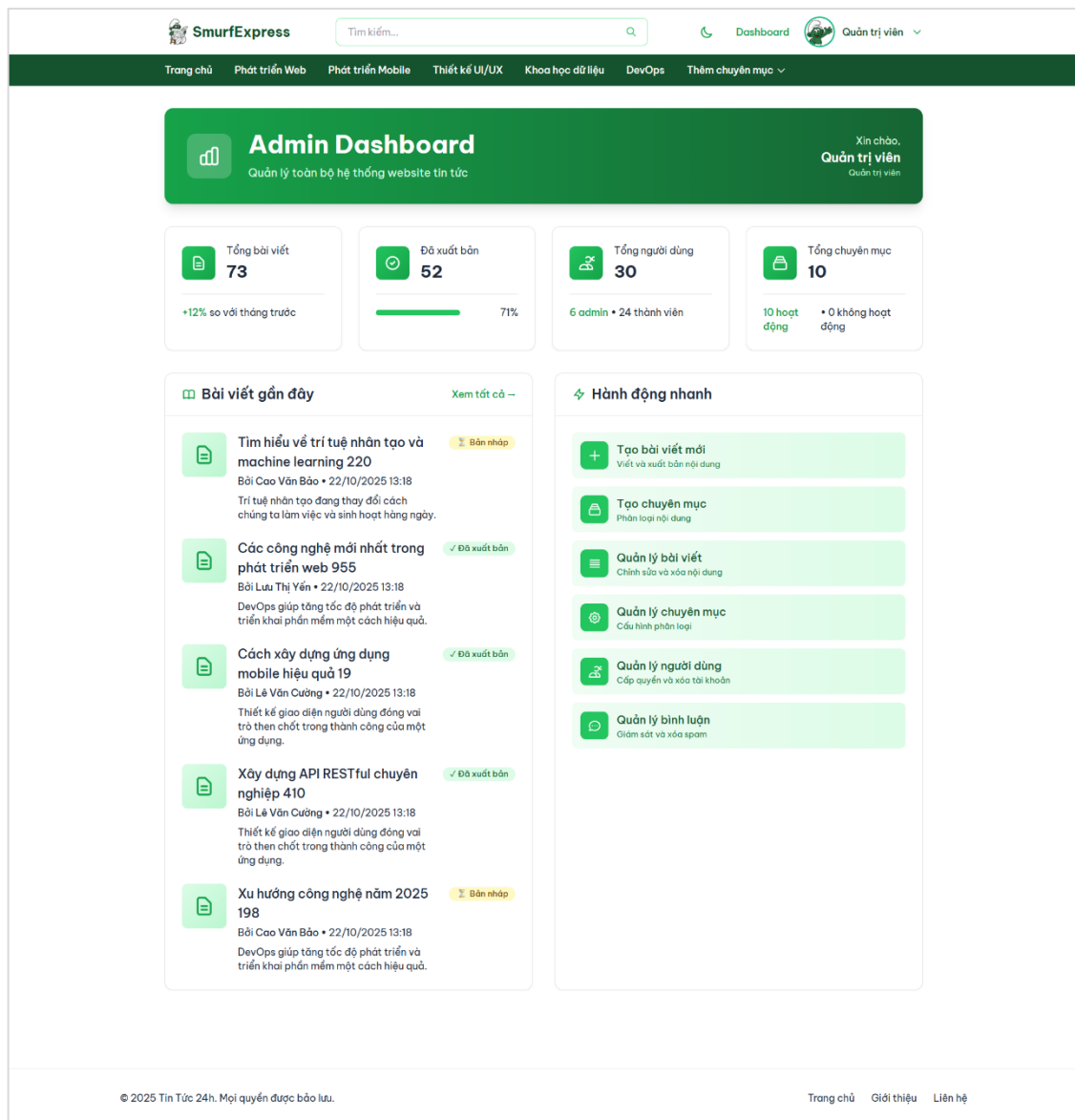
[Hủy bỏ](#)

[Lưu thay đổi](#)

© 2025 Tin Tức 24h. Mọi quyền được bảo lưu.

[Trang chủ](#) [Giới thiệu](#) [Liên hệ](#)

Hình 13. Màn hình Trang chỉnh sửa hồ sơ

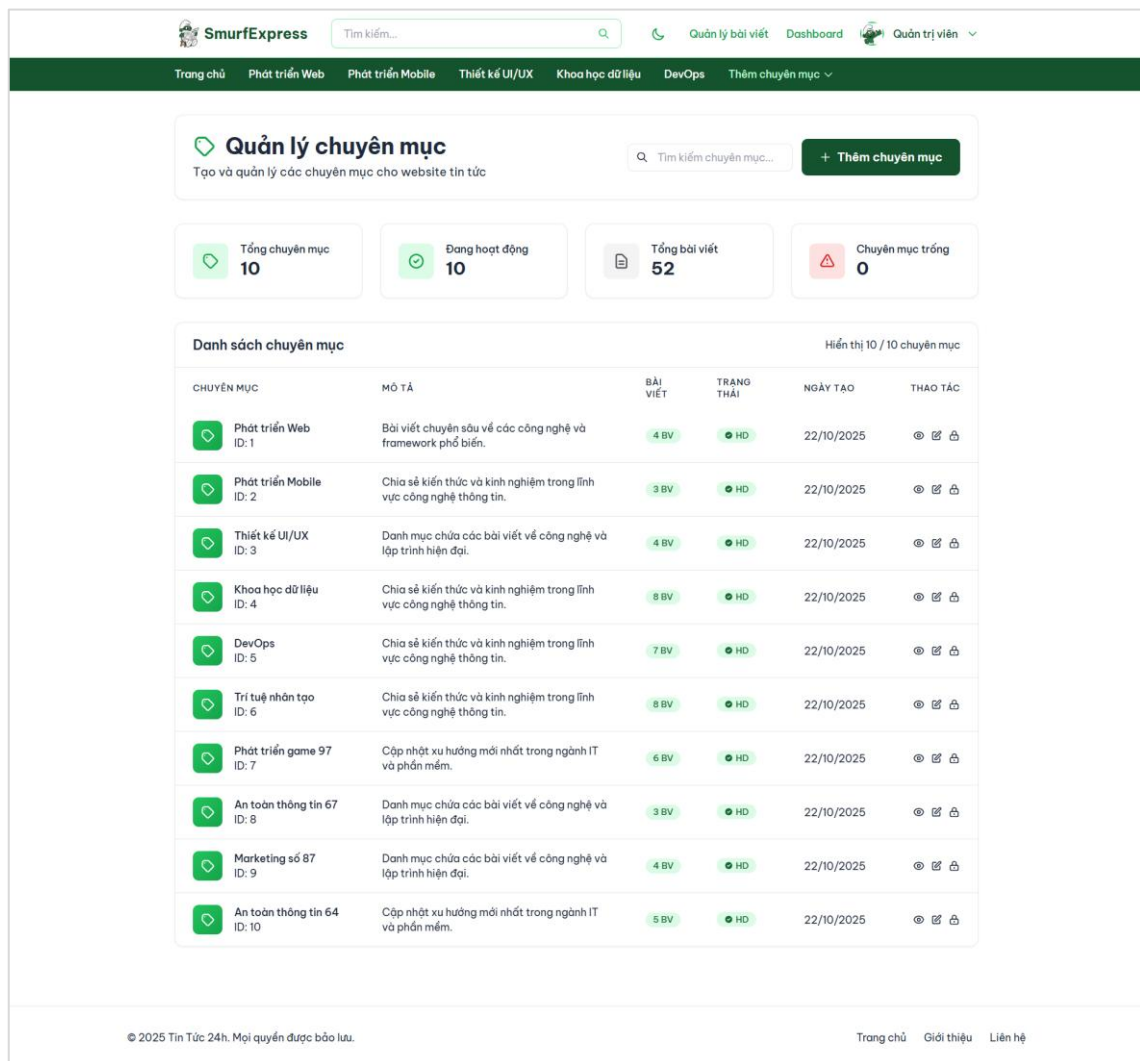


Hình 15. Màn hình Trang Dashboard (Admin)

4.2.2. Trang quản lý Chuyên mục (CRUD)

- **Mô tả:** Cung cấp giao diện bảng (table) liệt kê tất cả các chuyên mục. Admin có thể thực hiện đầy đủ các thao tác CRUD:
 - Create: Mở biểu mẫu để thêm chuyên mục mới (`categories/create.blade.php`).
 - Read: Xem danh sách chuyên mục.
 - Update: Chỉnh sửa tên chuyên mục (`categories/edit.blade.php`).

- Delete: Xóa chuyên mục (có xác nhận).
- **Tập View:** `categories/index.blade.php`

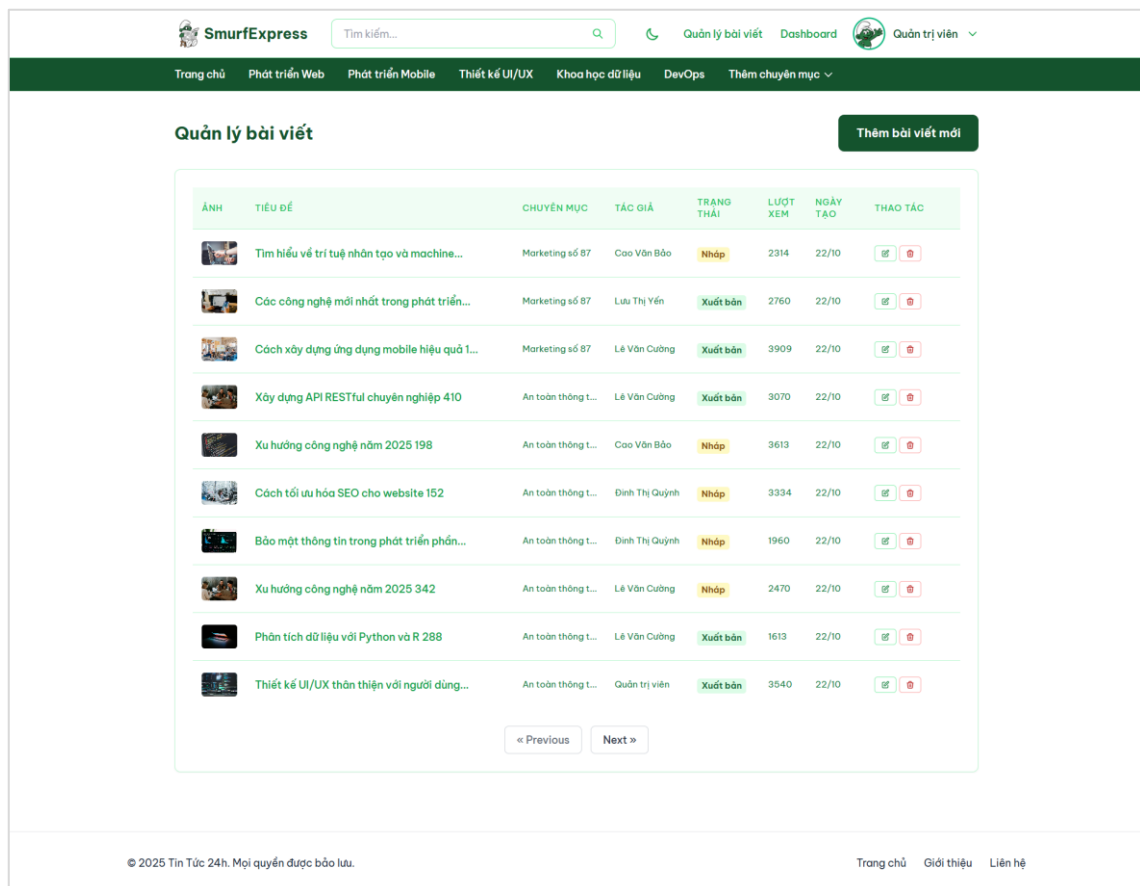


Hình 16. Màn hình Trang Quản lý chuyên mục

4.2.3. Trang quản lý Bài viết (CRUD & Duyệt bài)

- **Mô tả:** Giao diện quản lý toàn diện các bài viết. Bảng liệt kê hiển thị các thông tin chính như Tiêu đề, Tác giả, Chuyên mục, Trạng thái (Đã duyệt/Chưa duyệt) và Lượt xem.
- **Chức năng CRUD:**
 - Create: Mở biểu mẫu tạo bài viết mới. Biểu mẫu này bao gồm trình soạn thảo văn bản (WYSIWYG hoặc Trix editor), ô chọn chuyên mục, và khu vực tải lên hình ảnh (kết nối **ImgBBService**).

- Update: Chỉnh sửa nội dung bài viết.
- Delete: Xóa bài viết.
- Duyệt bài: Admin có thể thay đổi trạng thái `is_published` của bài viết, quyết định bài viết nào được phép hiển thị công khai.
- **Tập View:** `posts/index.blade.php`, `posts/create.blade.php`, `posts/edit.blade.php`



Hình 17. Màn hình Trang Quản lý bài viết

SmurfExpress

Quản lý bài viết

Dashboard

Quản trị viên

Trang chủ

Phát triển Web

Phát triển Mobile

Thiết kế UI/UX

Khoa học dữ liệu

DevOps

Thăm chuyên mục

Trang chủ > Quản lý bài viết >

Chỉnh sửa: Tim hiểu về trí tuệ nhân tạo và machine learning 220

Chỉnh sửa bài viết

Cập nhật nội dung cho "Tim hiểu về trí tuệ nhân tạo và machine learning 220"

Bản nháp

Tiêu đề bài viết

Tiêu đề *

Tim hiểu về trí tuệ nhân tạo và machine learning 220

Tiêu đề sẽ hiển thị trong danh sách bài viết và kết quả tìm kiếm

Chuyên mục

Chọn chuyên mục *

Marketing số 87

Trạng thái

Trạng thái xuất bản *

Bản nháp

Thông tin bài viết

Ngày tạo: 22/10/2025

Cập nhật cuối: 22/10/2025

Tác giả: Cao Văn Bảo

Lượt xem: 0 lượt

Bình luận: 0 bình luận

Thống kê

Số từ: 76

Số ký tự: 367

Thời gian đọc: 1 phút

Tóm tắt bài viết

Tóm tắt ngắn gọn

Trí tuệ nhân tạo đang thay đổi cách chúng ta làm việc và sinh hoạt hàng ngày.

Tóm tắt sẽ hiển thị trong danh sách bài viết và mạng xã hội

Hình ảnh bài viết

Hình ảnh hiện tại

Ảnh đại diện

Thêm hình ảnh mới

Tải lên hình ảnh

hoặc kéo thả vào đây

PNG, JPG, GIF tối đa 5MB mỗi file, tối đa 10 file

Nội dung bài viết

Nội dung chi tiết *

B

Framework Laravel đã trở thành một trong những công cụ phổ biến nhất cho phát triển web PHP.

DevOps giúp tăng tốc độ phát triển và triển khai phần mềm một cách hiệu quả.

Để trở thành một lập trình viên giỏi, bạn cần không ngừng học hỏi và cập nhật kiến thức mới.

Trong thời đại công nghệ số hiện nay, việc nắm vững các kỹ năng lập trình trở nên vô cùng quan trọng.

Hỗ trợ Markdown và HTML. Sử dụng toolbar để chèn ảnh và định dạng nội dung

Các trường có dấu* là bắt buộc

Hủy

Lưu nháp

Cập nhật bài viết

© 2025 Tin Tức 24h. Mọi quyền được bảo lưu.

Trang chủ

Giới thiệu

Liên hệ

Hình 18. Màn hình Trang tạo/chỉnh sửa bài viết

34

4.2.4. Trang quản lý Người dùng và Bình luận

- **Quản lý Người dùng:** Cung cấp danh sách người dùng. Admin có quyền chỉnh sửa thông tin người dùng (ví dụ: cấp/gỡ quyền Admin) hoặc xóa tài khoản người dùng (**AdminController@users**).

Quản lý Người dùng
Cấp quyền và quản lý tài khoản người dùng

Tổng số người dùng: 30

Danh sách Người dùng | Hiển thị 1 - 10 của 30 người dùng

NGƯỜI DÙNG	EMAIL	VAI TRÒ	NGÀY THAM GIA	HÀNH ĐỘNG
Hồ Thị Cúc ID: 12	jaquan.roberts@example.com ✓ Đã xác thực	Admin	22/10/2025	✓ ✖
Hoàng Văn Em ID: 13	vwaters@example.com ✓ Đã xác thực	User	22/10/2025	✓ ✖
Bùi Văn Minh ID: 14	verlie14@example.com ✓ Đã xác thực	Admin	22/10/2025	✓ ✖
Hoàng Văn Em ID: 15	daugherty.mathilde@example.org ✓ Đã xác thực	User	22/10/2025	✓ ✖
Lưu Thị Yến ID: 16	nhane@example.org ✓ Đã xác thực	Admin	22/10/2025	✓ ✖
Bùi Văn Minh ID: 17	hyatt.solon@example.net ✓ Đã xác thực	Admin	22/10/2025	✓ ✖
Lưu Thị Yến ID: 18	schaefer.billie@example.org ✓ Đã xác thực	User	22/10/2025	✓ ✖
Lê Văn Cường ID: 19	nikolaus@example.net ✓ Đã xác thực	User	22/10/2025	✓ ✖
Mai Văn Ước ID: 20	olson.lorenzo@example.com ✓ Đã xác thực	User	22/10/2025	✓ ✖
Hồ Thị Cúc ID: 21	ytillman@example.net ✓ Đã xác thực	User	22/10/2025	✓ ✖

« Previous | Next »

© 2025 Tin Tức 24h. Mọi quyền được bảo lưu. | Trang chủ | Giới thiệu | Liên hệ

Hình 19. Màn hình Trang Quản lý người dùng

- **Quản lý Bình luận:** Cung cấp danh sách tất cả bình luận trên hệ thống. Admin có quyền xóa bất kỳ bình luận nào, đặc biệt là các bình luận spam hoặc vi phạm (**AdminController@comments**).

SmurfExpress

Tìm kiếm...

Dashboard

Quản trị viên

Trang chủ

Phát triển Web

Phát triển Mobile

Thiết kế UI/UX

Khoa học dữ liệu

DevOps

Thêm chuyên mục

Quản lý Bình luận

Tổng số bình luận
474

Giám sát và xóa bình luận spam hoặc vi phạm

Danh sách Bình luận

Hiện thị 1 - 10 của 474 bình luận

NỘI DUNG	NGƯỜI DÙNG	BÀI VIẾT	NGÀY TẠO	HÀNH ĐỘNG
Có thể chia sẻ thêm về phần này được không?	<div><div></div><div>Hồ Thị Cúc trice@example.com</div></div>	<div>Bảo mật thông tin trong phát triển phần mềm 869</div> <div>ID: 50</div>	<div>22/10/2025</div> <div>17:07</div>	<div></div> <div></div>
Tôi đã học được rất nhiều điều từ bài viết này.	<div><div></div><div>Quản trị viên admin@example.com</div></div>	<div>Bảo mật thông tin trong phát triển phần mềm 869</div> <div>ID: 50</div>	<div>22/10/2025</div> <div>17:07</div>	<div></div> <div></div>
Hướng dẫn rất chi tiết và dễ hiểu. Tôi sẽ áp dụng ngay.	<div><div></div><div>Trần Thị Bình crooks.davonte@example.org</div></div>	<div>Bảo mật thông tin trong phát triển phần mềm 869</div> <div>ID: 50</div>	<div>22/10/2025</div> <div>17:07</div>	<div></div> <div></div>
Có thể chia sẻ thêm về phần này được không? <div>Phản hồi</div>	<div><div></div><div>Mai Văn Ước qkupal@example.org</div></div>	<div>Bảo mật thông tin trong phát triển phần mềm 869</div> <div>ID: 50</div>	<div>22/10/2025</div> <div>17:07</div>	<div></div> <div></div>
Hướng dẫn rất chi tiết và dễ hiểu. Tôi sẽ áp dụng ngay.	<div><div></div><div>Quản trị viên admin@example.com</div></div>	<div>Bảo mật thông tin trong phát triển phần mềm 869</div> <div>ID: 50</div>	<div>22/10/2025</div> <div>17:07</div>	<div></div> <div></div>
Bài viết chất lượng, đáng để bookmark.	<div><div></div><div>Hồ Thị Cúc jaquan.roberts@example.com</div></div>	<div>Bảo mật thông tin trong phát triển phần mềm 869</div> <div>ID: 50</div>	<div>22/10/2025</div> <div>17:07</div>	<div></div> <div></div>
Mình đã share bài viết này cho team dev. <div>Phản hồi</div>	<div><div></div><div>Hồ Thị Cúc trice@example.com</div></div>	<div>Bảo mật thông tin trong phát triển phần mềm 869</div> <div>ID: 50</div>	<div>22/10/2025</div> <div>17:07</div>	<div></div> <div></div>
Mình đã share bài viết này cho team dev.	<div><div></div><div>Quản trị viên admin@example.com</div></div>	<div>Cách tối ưu hóa SEO cho website 636</div> <div>ID: 51</div>	<div>22/10/2025</div> <div>17:07</div>	<div></div> <div></div>
Thanks bạn nhiều, đã giải quyết được vấn đề của mình.	<div><div></div><div>Bùi Văn Minh verlie14@example.com</div></div>	<div>Cách tối ưu hóa SEO cho website 636</div> <div>ID: 51</div>	<div>22/10/2025</div> <div>17:07</div>	<div></div> <div></div>
Cảm ơn bạn đã giải thích rất rõ ràng. <div>Phản hồi</div>	<div><div></div><div>Bùi Văn Minh hyatt.solon@example.net</div></div>	<div>Cách tối ưu hóa SEO cho website 636</div> <div>ID: 51</div>	<div>22/10/2025</div> <div>17:07</div>	<div></div> <div></div>

« Previous

Next »

Tổng bình luận
474

Đã duyệt
366

Chờ duyệt
108

© 2025 Tin Tức 24h. Mọi quyền được bảo lưu.

Trang chủ

Giới thiệu

Liên hệ

Hình 20. Màn hình Trang Quản lý bình luận

36

KẾT LUẬN & HƯỚNG PHÁT TRIỂN

1. Kết luận

1.1. Tóm tắt các kết quả đã đạt được

Dự án "Website Tin Tức SmurfExpress" đã hoàn thành các mục tiêu cốt lõi đề ra, xây dựng thành công một ứng dụng web tin tức động (dynamic) bằng framework Laravel 11. Các kết quả chính bao gồm:

- **Hoàn thiện hệ thống xác thực an toàn:** Đã triển khai đầy đủ các chức năng đăng ký, đăng nhập (bằng email và Google OAuth), quên mật khẩu và quản lý phiên làm việc.
- **Xây dựng giao diện người dùng (Client) đầy đủ chức năng:** Người dùng có thể xem tin tức (trang chủ, chi tiết, theo chuyên mục), tìm kiếm bài viết và tham gia tương tác qua lại thông qua hệ thống bình luận (bao gồm cả trả lời bình luận).
- **Xây dựng hệ thống quản trị (Admin) mạnh mẽ:** Cung cấp một trang Dashboard trực quan, cho phép quản trị viên quản lý toàn diện các tài nguyên chính của hệ thống, bao gồm CRUD cho Chuyên mục, Bài viết (có tải ảnh lên dịch vụ ngoài), Người dùng (phân quyền admin) và Bình luận.
- **Áp dụng thành công các kỹ thuật hiện đại:** Tận dụng hiệu quả kiến trúc MVC, Eloquent ORM, Blade template, TailwindCSS, và các biện pháp bảo mật tiêu chuẩn (CSRF, XSS, SQL Injection) để đảm bảo ứng dụng dễ bảo trì, an toàn và có hiệu suất tốt.

1.2. Các hạn chế còn tồn tại

Bên cạnh các kết quả đạt được, dự án vẫn còn một số hạn chế do giới hạn về thời gian và nguồn lực:

- **Hệ thống phân quyền:** Mới chỉ dừng lại ở hai vai trò (Admin và User). Chưa phát triển các vai trò trung gian như "Biên tập viên" (chỉ đăng/sửa bài) hay "Cộng tác viên".
- **Tối ưu trải nghiệm người dùng (UX):** Một số tính năng như thông báo (notification) thời gian thực, gợi ý bài viết liên quan/cá nhân hóa chưa được triển khai.
- **Tối ưu SEO:** Các yếu tố về SEO on-page (meta tags, sitemap...) mới chỉ ở mức cơ bản, chưa được tối ưu chuyên sâu để tăng thứ hạng tìm kiếm.
- **Kiểm thử (Testing):** Dự án chưa xây dựng bộ kiểm thử tự động (automated tests - unit, feature tests), việc kiểm thử chủ yếu vẫn đang thực hiện thủ công.

1.3. Đề xuất hướng phát triển, nâng cấp

Từ những hạn chế trên, dự án có thể được tiếp tục phát triển và nâng cấp trong tương lai theo các hướng sau:

- **Mở rộng hệ thống Role & Permission:** Xây dựng một hệ thống phân quyền chi tiết (ACL) cho phép Admin tùy chỉnh quyền hạn cho từng vai trò (ví dụ: vai trò chỉ được duyệt bài, vai trò chỉ được viết bài).
- **Tích hợp Real-time Notification:** Sử dụng công nghệ (ví dụ: Laravel Echo, Pusher) để gửi thông báo tức thời cho người dùng khi có bình luận mới hoặc bài viết mới được duyệt.
- **Cá nhân hóa nội dung:** Xây dựng thuật toán gợi ý bài viết dựa trên lịch sử xem tin, các chuyên mục người dùng quan tâm.
- **Nâng cao SEO:** Tự động tạo sitemap, tối ưu hóa meta description và keywords cho từng bài viết để thân thiện hơn với các công cụ tìm kiếm.

- **Hoàn thiện bộ kiểm thử (Test Coverage):** Viết unit test và feature test để đảm bảo tính ổn định của mã nguồn khi thêm tính năng mới hoặc tái cấu trúc (refactor).