

TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG



NIÊN LUẬN CƠ SỞ NGÀNH CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI
XÂY DỰNG TRANG WEB BLOG TIN TỨC BẰNG
FRAMEWORK LARAVEL

Sinh viên thực hiện

Bùi Văn Khoa

MSSV: B1606808

Khóa: 42

Cần Thơ, 11/2019

TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG



NIÊN LUẬN CƠ SỞ NGÀNH CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI
XÂY DỰNG TRANG WEB BLOG TIN TỨC BẰNG
FRAMEWORK LARAVEL

Giáo viên hướng dẫn
Th.S Phạm Thị Xuân Diễm

Sinh viên thực hiện
Bùi Văn Khoa
MSSV: B1606808
Khóa: 42

Cần Thơ, 11/2019

ĐÁNH GIÁ KẾT QUẢ THỰC HIỆN NIÊN LUẬN

(Học kỳ: 01, Năm học 2019-2020)

TT	Điểm thành phần	Quy định	Trọng số	GVHD
1	Điểm chuyên cần	Thực hiện chăm chỉ, đúng tiến độ theo yêu cầu của GVHD.	10%	
2	Điểm quyền báo cáo		40%	
	Hình thức	Bìa (0.25 điểm) Bố cục (0.25 điểm) <ul style="list-style-type: none">- Trang đánh giá.- Mục lục.- Phụ lục.- Tài liệu tham khảo.		
	Nội dung	Giới thiệu (1 điểm) Ứng dụng (1,5 điểm) Kết luận (1 điểm)		
3	Điểm chương trình demo	- Giao diện thân thiện, dễ sử dụng (1 điểm) - Kết quả chính xác, đúng theo bản thiết kế hệ thống (3 điểm)	40%	
4	Điểm sáng tạo	- Có ý tưởng mới trong thiết kế hay trong phát triển ứng dụng - Giao diện có tính sáng tạo	10%	

Cần Thơ, ngày tháng năm 2019
GIÁO VIÊN CHẤM

Phạm Thị Xuân Diễm

MỤC LỤC

CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI	1
1.1 ĐẶT VẤN ĐỀ	1
1.2 MỤC TIÊU ĐỀ TÀI	2
1.3 NỘI DUNG NGHIÊN CỨU	2
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	3
2.1 TỔNG QUAN VỀ LARAVEL	3
2.1.1 Mô hình MVC	3
2.1.2 Mô hình MVC trong Laravel hoạt động như thế nào?	3
2.1.3 Cài đặt Laravel	4
2.1.4 Cấu trúc thư mục	4
2.2 CÔNG NGHỆ VÀ CÁC NGÔN NGỮ ĐƯỢC SỬ DỤNG	5
CHƯƠNG 3: MÔ TẢ VÀ THIẾT KẾ HỆ THỐNG	7
3.1 MÔ TẢ HỆ THỐNG	7
3.1.1 Mục đích của hệ thống	7
3.1.2 Yêu cầu hệ thống	7
3.1.3 Chức năng của hệ thống	7
3.2 THIẾT KẾ HỆ THỐNG	8
3.2.1 Mô hình usecase	8
3.2.2 Mô hình dữ liệu mức quan niệm - CDM (Conceptual Data Model)	10
3.3 THIẾT KẾ CÁC LỚP	14
3.3.1 Model	14
3.3.2 Controller	14
3.3.3 Middleware	14
3.3.4 View	15
CHƯƠNG 4: KẾT QUẢ CHƯƠNG TRÌNH	16
4.1 GIAO DIỆN TRANG ADMIN	16
4.1.1 Trang đăng nhập admin	16
4.1.2 Trang quản lý danh mục bài viết	16
4.1.3 Trang quản lý bài viết	17
4.1.4 Quản lý tài khoản người dùng	18
4.2 TRANG HIỂN THỊ	20
4.2.1 Trang chủ	20
4.2.2 Trang đăng ký và đăng nhập	21

4.2.3	Trang danh mục.....	21
4.2.4	Trang nội dung bài viết	22
4.2.5	Trang tìm kiếm.....	23
4.2.6	Trang báo lỗi	23
4.2.7	Trang giới thiệu	23
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN		25
5.1	KẾT LUẬN.....	25
5.2	HƯỚNG PHÁT TRIỂN.....	25
TÀI LIỆU THAM KHẢO		26

DANH MỤC HÌNH

Hình 1: Biểu đồ thể hiện sự phổ biến của Laravel so với các Framework PHP	1
Hình 2 Cách thức hoạt động của mô hình MVC	3
Hình 3: Mô hình MVC trong Laravel.....	4
Hình 4: Cây thư mục trong Laravel.....	5
Hình 5: Sơ đồ usecase trang quản trị.....	8
Hình 6: Sơ đồ trang chủ.....	9
Hình 7: Sơ đồ CDM.....	10
Hình 8: Giao diện trang đăng nhập admin.....	16
Hình 9: Giao diện trang quản lý danh mục bài viết.....	16
Hình 10: Giao diện trang sửa danh mục	17
Hình 11 :Giao diện trang quản lý bài viết	17
Hình 12: Giao diện trang thêm mới bài viết.....	18
Hình 13: Giao diện trang sửa bài viết.....	18
Hình 14: Giao diện trang quản lý users	19
Hình 15: Trang tạo tài khoản người dùng mới	19
Hình 16: Trang sửa tài khoản người dùng.....	20
Hình 17: Trang chủ.....	20
Hình 18: Trang đăng ký.....	21
Hình 19: Trang đăng nhập.....	21
Hình 20: Trang danh mục loại tin.....	22
Hình 21: Giao diện trang nội dung bài viết	22
Hình 22: Giao diện trang tìm kiếm.....	23
Hình 23: Giao diện trang báo lỗi	23
Hình 24: Giao diện trang giới thiệu.....	24

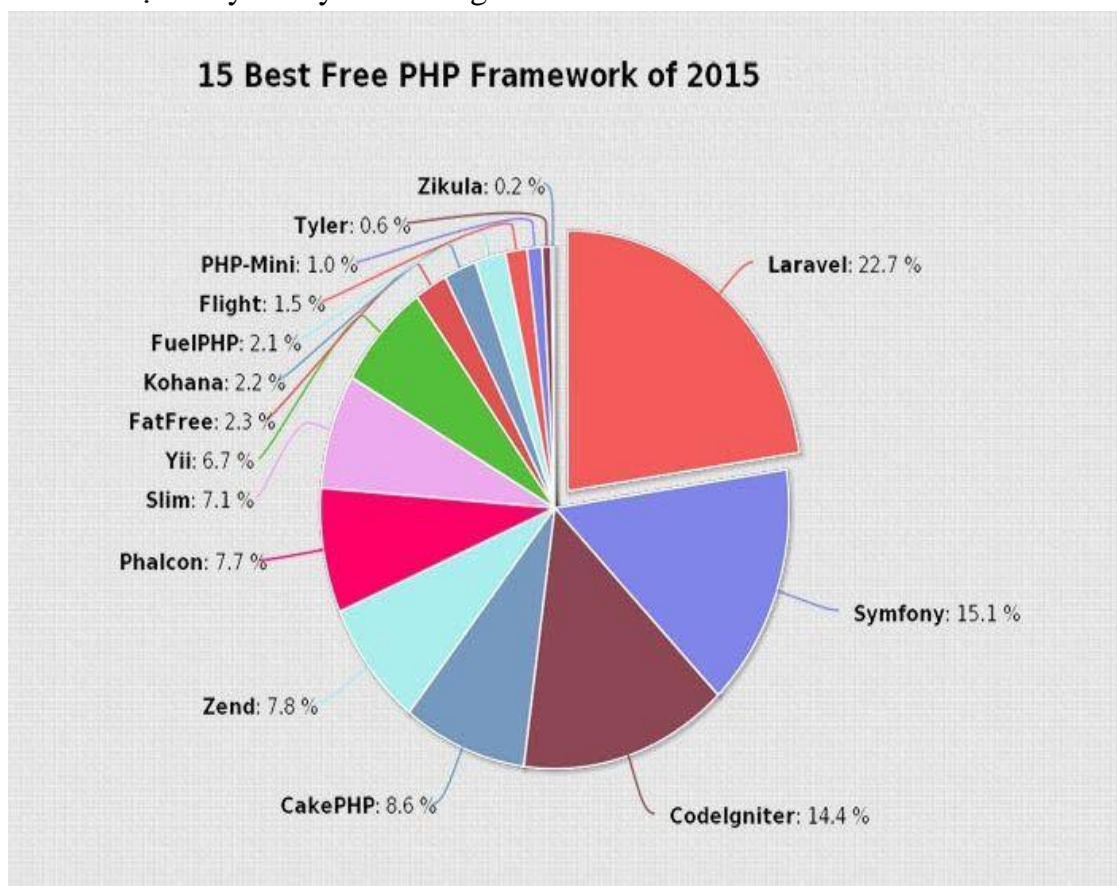
CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

1.1 ĐẶT VẤN ĐỀ

Công nghệ thông tin ngày càng phát triển kéo theo đó các ứng dụng web liên tục tạo ra, đánh dấu mức nhảy vọt trong thế giới lập trình. Các nền tảng lập trình cũng được xây dựng hàng loạt nhằm mục đích cải thiện tốc độ thực thi và nâng cao tính bảo mật cho trang web. Với nỗ lực làm giảm thời gian phát triển một trang web thì framework ra đời là một điều tất yếu. Trên thị trường có rất nhiều Framework mã nguồn mở giúp người lập trình viên có thể tiếp cận dễ dàng để hoàn thành công việc một cách nhanh nhất.

Laravel là một PHP Framework mã nguồn mở và miễn phí, được phát triển bởi Taylor Otwell và nhằm mục tiêu hỗ trợ phát triển các ứng dụng web theo cấu trúc MVC (Model- View- Controller). Những tính năng nổi bật của Laravel bao gồm cú pháp dễ hiểu- rõ ràng, một hệ thống đóng gói Modular và quản lý gói phụ thuộc, nhiều cách khác nhau để truy cập vào các cơ sở dữ liệu quan hệ, nhiều tiện ích khác nhau hỗ trợ việc triển khai vào bảo trì ứng dụng.[1]

Vào tháng 3 năm 2015, các lập trình viên đã có cuộc bình chọn PHP Framework phổ biến nhất, kết quả Laravel đã giành giải quán quân cho Framework PHP năm 2015. Theo sau lần lượt là Symfony và CodeIgniter và các Framework khác.



Hình 1: Biểu đồ thể hiện sự phổ biến của Laravel so với các Framework PHP

* Nguồn ảnh: <https://hocvps.com/cai-dat-laravel/> [4]

1.2 MỤC TIÊU ĐỀ TÀI

- Xây dựng thành công website blog bằng framework Laravel dựa trên nền tảng ngôn ngữ PHP.
- Website hiển thị nội dung đẹp, thu hút người dùng
- Ứng dụng thành công mô hình MVC vào việc phát triển website

1.3 NỘI DUNG NGHIÊN CỨU

- Tìm hiểu về các công nghệ được sử dụng trong việc phát triển đề tài này
- Biết cách thiết kế website bằng công nghệ bootstrap
- Thực hiện nghiên cứu framework Laravel và mô hình MVC để ứng dụng phát triển website

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1 TỔNG QUAN VỀ LARAVEL

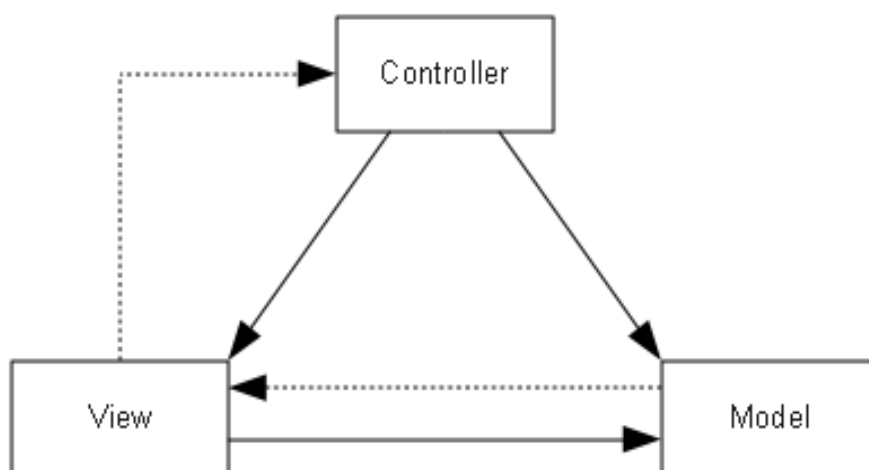
2.1.1 Mô hình MVC

Mô hình MVC là viết tắt của 3 chữ Model, View, Controller. Mô hình này tách một ứng dụng web ra làm 3 thành phần đảm nhiệm chức năng tách biệt, thuận tiện cho việc xử lý và bảo trì.

Model: Chịu trách nhiệm quản lý dữ liệu, nó lưu trữ và truy xuất các thực thể từ cơ sở dữ liệu như mysql, sql server, postgresQL, ... đồng thời các logic được thực thi bởi ứng dụng

View: Chịu trách nhiệm hiển thị dữ liệu đã được truy xuất từ model theo một format nào đó theo ý đồ của lập trình viên. Cách sử dụng của View tương tự như các module templates thường thấy trong các ứng dụng web phổ biến như WordPress, Joomla, ...

Controller: trung gian, làm nhiệm vụ xử lý cho model và view tương tác với nhau. Controller nhận request từ client, sau đó gọi các model để thực hiện các hoạt động được yêu cầu và gửi ra ngoài View. View sẽ chịu trách nhiệm format lại data từ controller gửi ra và trình bày dữ liệu theo 1 định dạng đầu ra (html).[2]

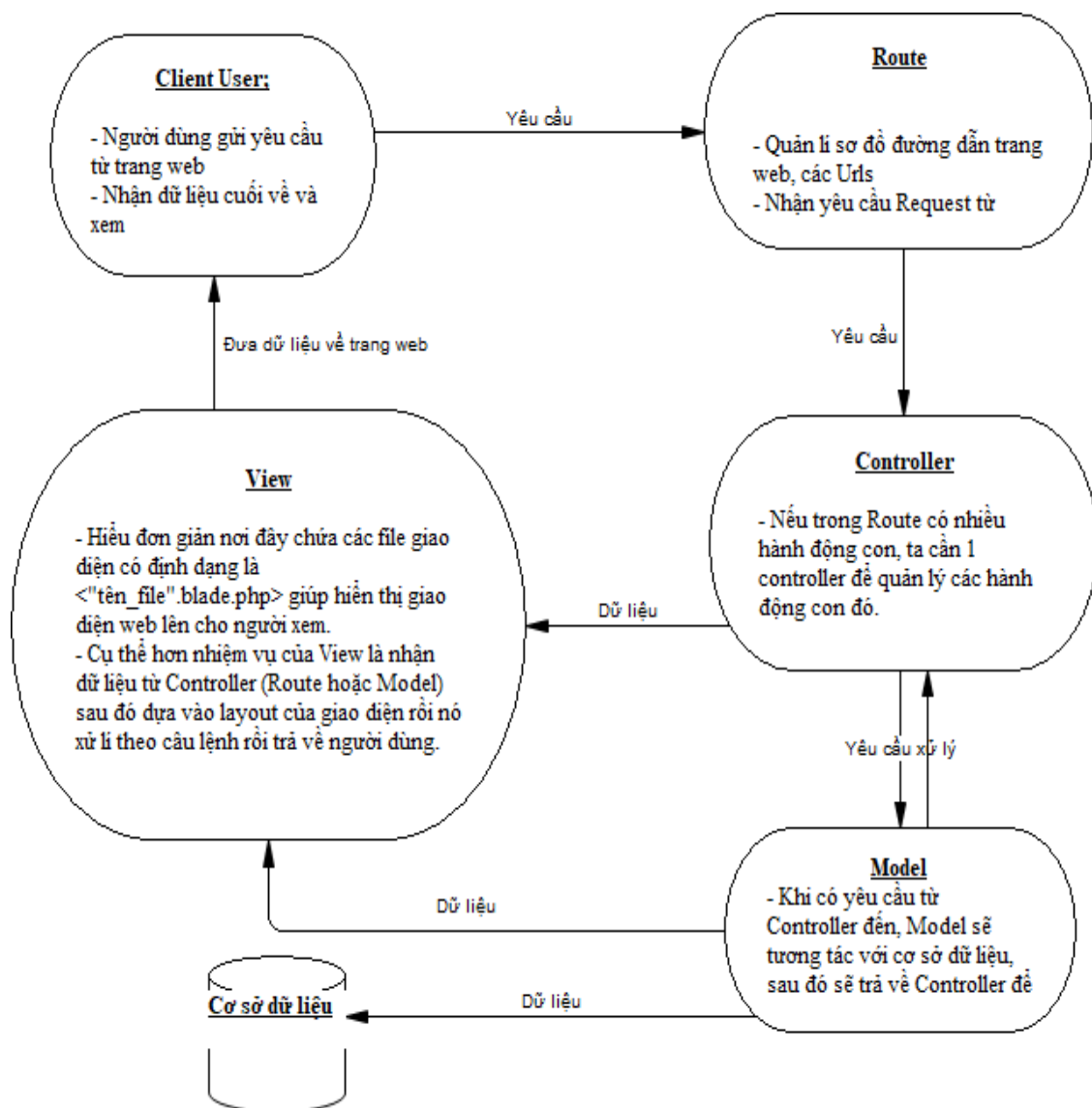


Hình 2 Cách thức hoạt động của mô hình MVC

2.1.2 Mô hình MVC trong Laravel hoạt động như thế nào?

Laravel Framework hỗ trợ lập trình theo mô hình MVC khá mạnh mẽ. Ngoài các thành phần chính Model, View, Controller thì Routes được sử dụng định tuyến người dùng theo đúng Urls. Ở đây, mọi Request từ phía người dùng đều phải qua Route, dữ liệu được gửi xuống Controller để xử lý, cần dữ liệu sẽ lấy từ Model lên hoặc cập nhật dữ liệu xuống Model, kết quả gửi ra View cho người sử dụng.

Cách thức hoạt động MVC trong Laravel được thể hiện như hình sau



Hình 3: Mô hình MVC trong Laravel

2.1.3 Cài đặt Laravel

Để cài đặt Laravel, trước tiên chúng ta cần phải cài đặt Composer. Sau khi cài đặt Composer thành công, chúng ta cần khởi động Terminal lên và truy cập vào thư mục cần cài đặt.

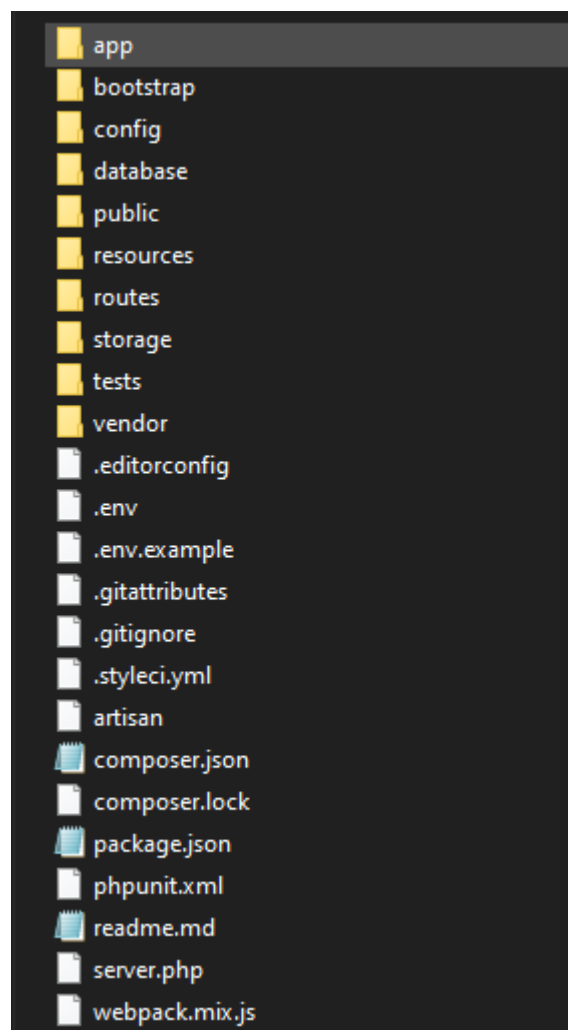
Thực hiện câu lệnh như sau để cài đặt Laravel:

```
composer create-project -prefer-dist laravel/laravel {Tên project Laravel}
```

2.1.4 Cấu trúc thư mục

- app: chứa mã nguồn của ứng dụng mà chúng ta sẽ viết trong laravel tại
- đây chứa phần lớn source code xử lý logic, helper, Model, Middleware và điều hướng controller ...
- bootstrap: chứa các file bootstrap quan trọng và các file cấu hình nạp tự động
- config: chứa các file cấu hình cho ứng dụng như xác thực, cache, mail,
- cơ sở dữ liệu ...
- database: chứa các file migration và các mẫu seeder

- public: chứa các hình ảnh và các file thư viện như css, javascript, bootstrap
- resource: chứa view, các raw asset và các file ngôn ngữ
 - Thư mục view chứa các file giao diện có định dạng “<<ten_file>>.blade.php” được sử dụng bởi Controller hoặc route.
 - Thư mục lang chứa những file PHP chứa mảng của những chuỗi cho phép dễ dàng xác định vị trí ứng dụng
- routes: chứa các file route
- test: chứa các test tự động của bạn
- storage: chứa các file cần lưu trữ như cache, session, logs ...
- vendor: chứa core laravel và thư viện
- .env: đây là file môi trường để kết nối project Laravel tới cơ sở dữ liệu



Hình 4: Cây thư mục trong Laravel

2.2 CÔNG NGHỆ VÀ CÁC NGÔN NGỮ ĐƯỢC SỬ DỤNG

Ngoài Laravel, website còn sử dụng nhiều công nghệ khác như sau:

- **HTML** (Hyper Text Markup Language) được hiểu là “ngôn ngữ đánh dấu siêu văn bản” là một ngôn ngữ được thiết kế ra để tạo nên các trang web tĩnh
- **CSS** (Cascading Style Sheets) giúp ta định dạng lại bố cục, font chữ, màu nền,...

- **Javascript** giúp xây dựng trang web trở nên sinh động và thanh thiện với người dùng hơn
- **Bootstrap** là một bộ framework **CSS** bao gồm các tập tin như **css**, **jquery**, **javascript** được viết sẵn, người thiết kế website chỉ cần gọi chúng để thiết kế website một cách dễ dàng hơn. Tính năng hữu ích trên **Bootstrap** là hỗ trợ **responsive** là khi website viết bằng **Bootstrap** có thể hiển thị đúng bố cục trên mọi thiết bị có kích thước màn hình khác nhau như desktop, tablet, smartphone. Ngoài ra, ta có thể áp dụng các thẻ của **HTML**, **CSS** vào **Bootstrap** để định dạng lại nội dung theo ý muốn.
- **PHP** là ngôn ngữ dùng bên phía server, đây là ngôn ngữ chính của framework Laravel
- **Apache** là một Web server đóng vai trò giao tiếp, xử lý các yêu cầu từ người dùng trên các trang web tĩnh
- **Mysql** là hệ quản trị cơ sở dữ liệu nguồn mở. Lợi thế của **Mysql** là sử dụng tài nguyên ít và khá linh hoạt trong việc lưu trữ và truy vấn cơ sở dữ liệu. **Mysql** hỗ trợ nhiều ngôn ngữ lập trình khác nhau như JSP, PHP, ASP, ...
- **CKEditor** là trình soạn thảo văn bản mã nguồn mở WYSIWYG(tay làm- mắt thấy) được phát triển bởi CKSource
- **CKFinder** là trình quản lý file được phát triển bởi CKSource
- **XAMPP** là chương trình tạo máy chủ (web server) có tích hợp sẵn Apache, MySQL

CHƯƠNG 3: MÔ TẢ VÀ THIẾT KẾ HỆ THỐNG

3.1 MÔ TẢ HỆ THỐNG

3.1.1 Mục đích của hệ thống

Xây dựng trang blog để cho mọi người có thể xem những bài viết với nhiều chủ đề khác nhau do chính tác giả viết. Mặc khác mọi người cũng có thể tương tác bình luận hoặc phản hồi ý kiến cá nhân của mình trong bài viết đó.

3.1.2 Yêu cầu hệ thống

Đây là chương trình được xây dựng trên nền tảng web nên không yêu cầu quá cao về cấu hình thiết bị hay hệ điều hành. Website có xây dựng cơ chế responsive bằng bootstrap nên chúng có thể hoạt động tốt trên nhiều thiết bị có tỉ lệ màn hình khác nhau như TV, tablet, mobile

Giao diện thân thiện, dễ dàng sử dụng đối với những người không có am hiểu nhiều về công nghệ. Các bài viết luôn được cập nhật đều đặn trên hệ thống. Đảm bảo tính bảo mật về quyền truy cập vào trang quản trị.

3.1.3 Chức năng của hệ thống

* Đối với người dùng

- **Người dùng vắng lai và người dùng đã đăng ký thành viên** có thể sử dụng đầy đủ các chức năng của hệ thống nhưng để bình luận bài viết bắt buộc phải đăng nhập
- **Đăng ký và đăng nhập:** người dùng đăng ký và đăng nhập để sử dụng các chức năng của website
- **Đăng xuất:** thoát khỏi tài khoản đã đăng nhập vào website
- **Tìm kiếm:** sử dụng từ khóa để tìm bài viết mà bạn muốn đọc
- **Trang chủ:** hiển thị những bài viết nổi bật của nhiều thể loại được sắp xếp ngẫu nhiên
- **Trang danh mục:** hiển thị bài viết đúng theo danh mục bài viết đó
- **Bài viết:** đây là trang nội dung bài viết bên trong có chèn liên kết những bài viết có liên quan. Có thể xem bình luận của người dùng khác mặc khác cũng có thể bình luận trong đây
- **Bình luận:** chức năng bình luận giành cho người dùng đã đăng nhập

* Đối với người quản trị (admin)

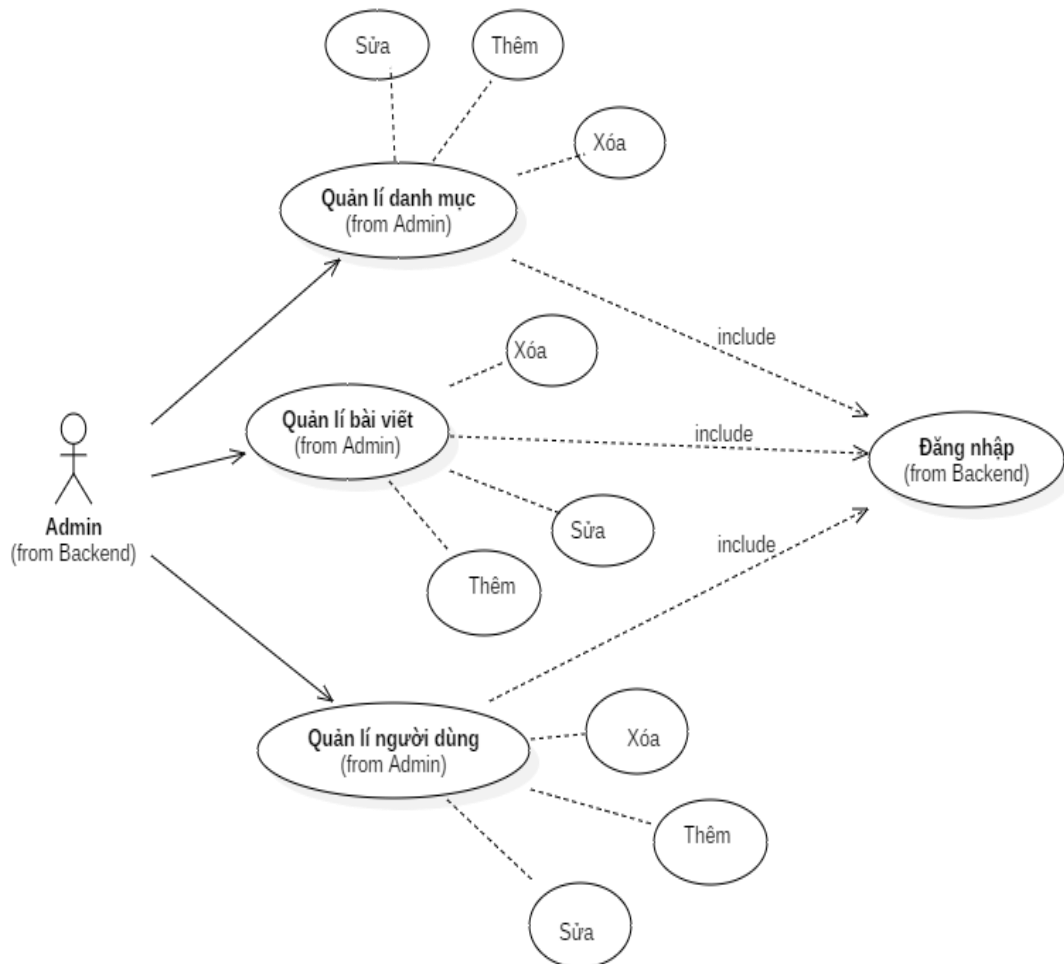
- **Đăng nhập:** chức năng đăng nhập bằng tài khoản admin đến trang quản trị
- **Quản lý danh mục:** chức năng này cho phép người quản trị có thể xem, sửa, xóa và thêm mới danh mục. Khi người quản trị xóa một danh mục nào đó thì tất cả các dữ liệu bài viết và bình luận thuộc nhóm danh mục đó sẽ đồng thời xóa hết
- **Quản lý bài viết:** chức năng này cho phép người quản trị có thể xem, sửa, xóa và thêm mới bài viết. Khi xóa bài viết thì những bình luận có trong bài viết sẽ không còn tồn tại.

- **Quản lý người dùng:** chức năng này cho phép người quản trị có thể xem, sửa, xóa và thêm mới người dùng. Khi xóa người dùng nào đó thì những bình luận của họ sẽ không tồn tại nữa.
- **Đăng xuất:** kết thúc phiên làm việc nếu không còn sử dụng nữa để đảm bảo an toàn cho hệ thống

3.2 THIẾT KẾ HỆ THỐNG

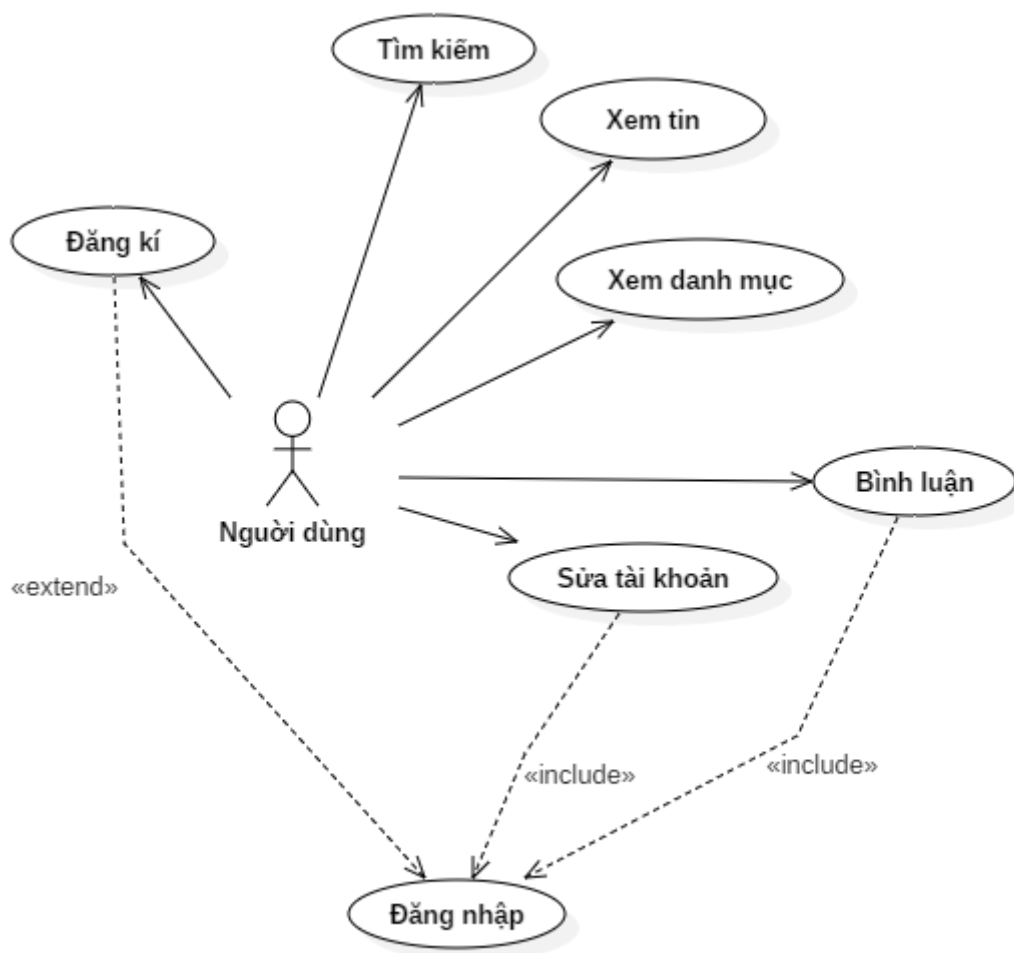
3.2.1 Mô hình usecase

- Sơ đồ trang quản trị (Admin)



Hình 5: Sơ đồ usecase trang quản trị

- Sơ đồ usecase trang trang hiển thị



Hình 6: Sơ đồ trang chủ

*Danh sách các Actor

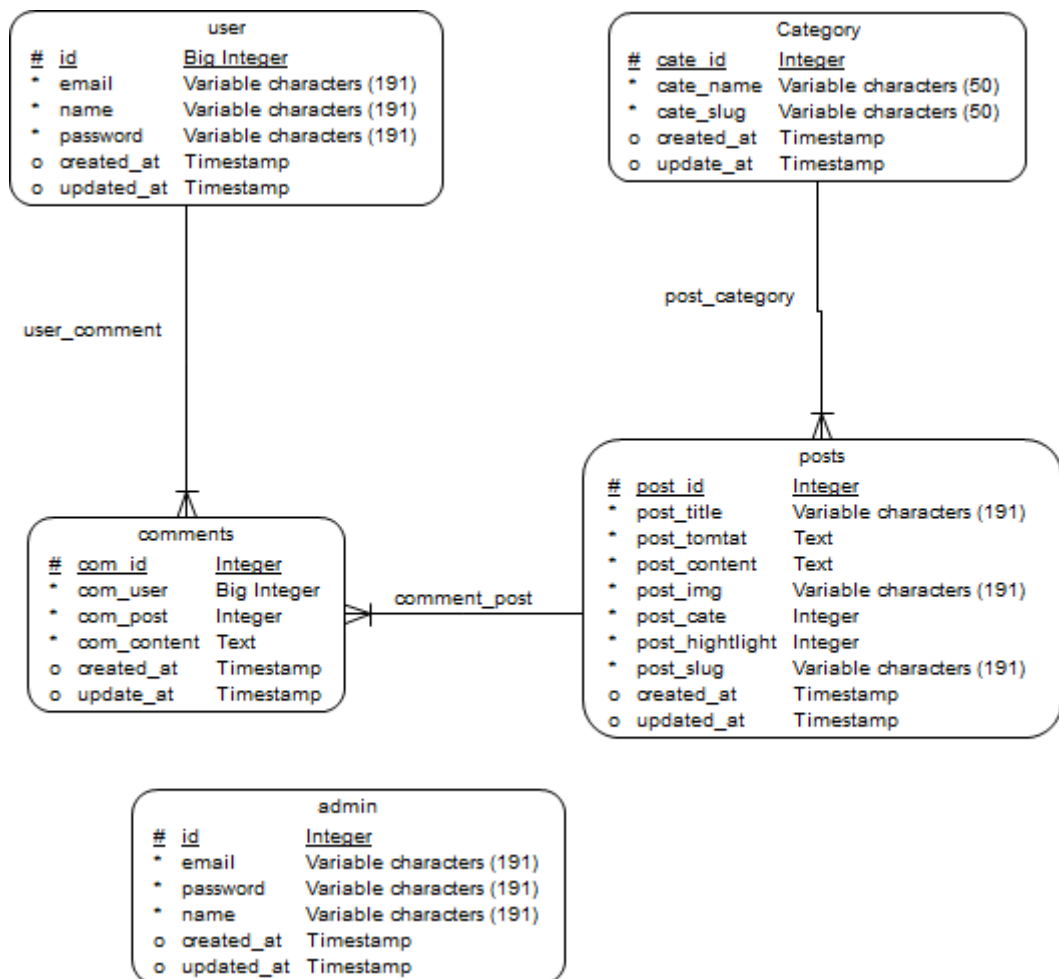
Tên Actor	Mô tả
Admin	Người có quyền hạn cao nhất trong hệ thống
Người dùng	Người dùng có tài khoản hoặc khách vãng lai

*Danh sách các Usecase

Tên Usecase	Mô tả
Đăng nhập	Thành viên có tài khoản đăng nhập vào hệ thống
Sửa tài khoản	Sửa thông tin tài khoản cá nhân như tên, mật khẩu, ...
Quản lí danh mục	Đây là chức năng mà người quản trị có thể thêm, sửa, xóa danh mục bài viết
Quản lí bài viết	Đây là chức năng mà người quản trị có thể thêm, sửa, xóa bài viết có trong hệ thống
Quản lí người dùng	Đây là chức năng mà người quản trị có thể thêm, sửa, xóa người dùng
Xem danh mục	Đây là chức năng xem danh mục mà người dùng khách có thể truy cập được

Xem bài viết	Chức năng xem nội dung bài viết, tất cả mọi người đều cũng có thể xem được
Bình luận	Chức năng này dành cho người đã đăng nhập vào hệ thống
Tìm kiếm	Tìm kiếm bài viết theo từ khóa
Đăng kí	Chức năng này dành cho người muốn tạo tài khoản để đăng nhập vào hệ thống

3.2.2 Mô hình dữ liệu mức quan niệm - CDM (Conceptual Data Model)



Hình 7: Sơ đồ CDM

*Bảng User

- Bảng user ghi lại thông tin người dùng, mặc khác dùng để đăng nhập vào hệ thống. Bảng này nối kết với bảng comments qua nối kết một - nhiều, tức là một user có nhiều bình luận nhưng 1 bình luận chỉ có 1 người duy nhất. Khi xóa 1 một người dùng nào đó thì mọi bình luận của người đó đều không còn tồn tại nữa.

Tên Cột	Kiểu dữ liệu	Null	Kích thước	Mô tả
Id (khóa chính)	Big integer	Không	20	Mã số người dùng tự động tăng khi thêm vào CSDL
email	Varchar	Không	191	Địa chỉ email của người dùng không được trùng với nhau, mặc khác đây cũng là tên đăng nhập
name	Varchar	Không	191	Tên người dùng
password	Varchar	Không	191	Mật khẩu được lưu dưới dạng mã hóa Bcrypt
created_at	timestamp	Có		Ghi lại thời gian khởi tạo
update_at	timestamp	Có		Ghi lại thời gian sửa đổi gần nhất

***Bảng Category**

- Bảng này dùng để ghi lại thể loại bài viết, mỗi thể loại phải khác nhau. Bảng này nối kết với bảng posts qua liên kết một – nhiều. Tức là 1 thể loại sẽ có nhiều bài viết nhưng 1 bài viết sẽ có duy nhất 1 thể loại. Khi xóa thể loại thì toàn bộ những bài viết sẽ xóa hết.

Tên cột	Kiểu dữ liệu	Null	Kích thước	Mô tả
cate_id (khóa chính)	Integer	Không	10	Id của danh mục thể loại bài viết. Đây là trường tự động tăng
cate_name	Varchar	Không	191	Tên của danh mục
cate_slug	Varchar	Không	191	Nơi đây lưu tên danh mục dưới dạng không dấu
created_at	timestamp	Có		Ghi lại thời gian khởi tạo

updated_at	timestamp	Có		Ghi lại thời gian sửa đổi gần nhất
------------	-----------	----	--	------------------------------------

***Bảng posts**

- Bảng posts ghi lại thông tin bài viết, mỗi bài viết phải có tên khác nhau. Bảng này nối kết với bảng category và bảng comments. Nối kết với bảng category bằng liên kết một - nhiều. Cũng tương tự như bảng category, bảng posts khi nối kết qua bảng comments cũng sử dụng liên kết một - nhiều, tức là một bài viết sẽ có nhiều bình luận nhưng một bình luận thì chỉ thuộc một bài viết.

Tên cột	Kiểu dữ liệu	Null	Kích thước	Mô tả
post_id (khóa chính)	Integer	Không	10	Id của bài viết
post_title	Varchar	Không	191	Tiêu đề của bài viết
post_tomtat	Text	Không		Tóm tắt bài viết
post_content	Text	Không		Nội dung bài viết
post_slug	Varchar	Không	191	Tiêu đề không dấu
post_img	Varchar	Không	191	Hình ảnh bài viết
post_cate (khóa ngoại)	Integer	Không		id của danh mục bài viết
post_highlight	Integer	Không		Bài viết nổi bật hay không nổi bật
created_at	Timestamp	Có		Ghi lại thời gian khởi tạo
updated_at	Timestamp	Có		Ghi lại thời gian sửa đổi gần nhất

***Bảng comments**

- Bảng này ghi lại bình luận của một người dùng nào đó. Bảng comments liên kết với bảng posts và bảng user bằng liên kết một – nhiều.

Tên cột	Kiểu dữ liệu	Null	Kích thước	Mô tả
com-id (khóa chính)	Integer	Không		Id của comments

com_user	Integer	Không		Id của user bình luận bài viết đó
com_post	Integer	Không		Id của bài viết
com_content	Text	Không		Nội dung comments
created_at	Timestamp	Có		Ghi lại thời gian khởi tạo
updated_at	Timestamp	Có		Ghi lại thời gian sửa đổi gần nhất

*Bảng admin

- Bảng admin là nơi để truy cập vào hệ thống admin. Đây là bảng riêng lẻ không có liên kết với bất kỳ với một bảng nào khác trong cơ sở dữ liệu.

Tên Cột	Kiểu dữ liệu	Null	Kích thước	Mô tả
Id (khóa chính)	Big integer	Không	20	Mã số người dùng tự động tăng khi thêm vào CSDL
email	Varchar	Không	191	Địa chỉ email của người dùng không được trùng với nhau, mặc khác đây cũng là tên đăng nhập
name	Varchar	Không	191	Tên người dùng
password	Varchar	Không	191	Mật khẩu được lưu dưới dạng mã hóa Bcrypt
created_at	timestamp	Có		Ghi lại thời gian khởi tạo
updated_at	timestamp	Có		Ghi lại thời gian sửa đổi gần nhất

3.3 THIẾT KẾ CÁC LỚP

3.3.1 Model

Model là nơi tương tác với cơ sở dữ liệu sau đó trả về Controller để xử lý. Vì thế mỗi bảng trong cơ sở dữ liệu sẽ tương ứng với một model. Để tạo Model trong laravel ta thực hiện câu lệnh như sau:

```
php artisan make:model {tên_model}
```

* Các Model đã tạo:

- Admin
- User
- Category
- Posts
- Comments

Các model trên sẽ được lưu trên thư mục: App

3.3.2 Controller

Controller là nơi xử lý logic, điều hướng các đối tượng tham gia hệ thống. Để tạo Controller trong laravel ta có thể thực hiện bằng tay hoặc thực hiện theo câu lệnh sau đây

```
php artisan make:controller {tên_controller}
```

* Các Controller đã tạo:

- **CateController:** xử lý danh mục thể loại bài viết
- **LoginController:** xử lý việc đăng nhập vào trang Admin (trang quản trị)
- **PostsController:** xử lý bài viết
- **UsersController:** xử lý những người dùng có trong hệ thống
- **PagesController:** đây là nơi xử lý tất cả nội dung liên quan đến trang chủ

Các Controller sẽ được lưu trong thư mục: App\Http\Controller.

3.3.3 Middleware

Middleware cung cấp một giải pháp khá tiện ích cho việc filtering HTTP các requests vào ứng dụng. Ví dụ, Laravel có chứa một middleware xác thực người dùng đăng nhập vào hệ thống [3]. Để thực hiện tạo middleware ta có thể thực hiện bằng tay hoặc thực hiện câu lệnh như sau:

```
php artisan make:middleware {tên_middleware}
```

Các Middleware đã được tạo:

- **CheckLoginAdmin:** kiểm tra tài khoản đăng nhập vào trang admin
- **CheckLogoutAdmin:** kiểm tra việc đăng xuất

Các Middleware sẽ được lưu tại thư mục: App\Http\Middleware

3.3.4 View

View là những file HTML có định dạng “.blade.php” nó là một tầng logic trong mô hình MVC. Mục đích sử dụng view là tách phần hiển thị nội dung ra khỏi phần xử lý của PHP. Nói cách khác nhiệm vụ của view chính là nhận dữ liệu từ controller (route, model) sau đó dựa vào câu lệnh đưa về người dùng.

Các view đã được tạo:

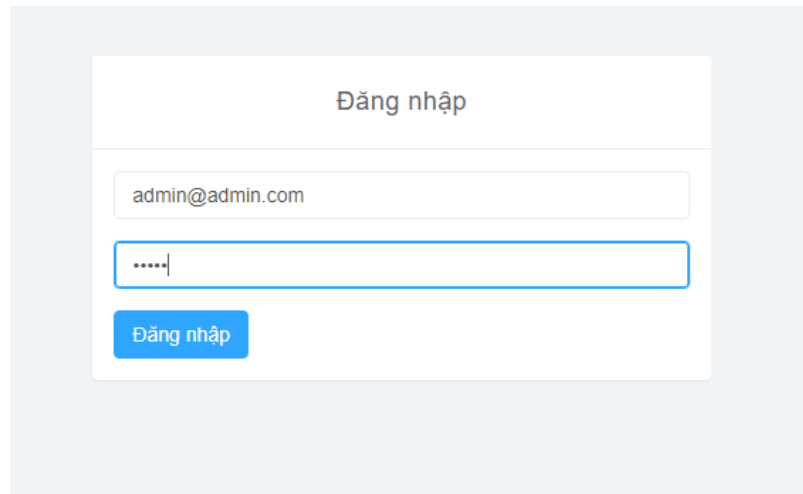
- Thư mục Admin (view trang admin): addposts.blade.php, adduser.blade.php, category.blade.php, editcategory.blade.php, editposts.blade.php, edituser.blade.php, login.blade.php, main.blade.php, posts.blade.php, user.blade.php
- Thư mục error (view báo lỗi): 404.blade.php, emptysearch.blade.php
- Thư mục pages (view trang người dùng): about.blade.php, account.blade.php, category.blade.php, detail.blade.php, index.blade.php, login.blade.php, main.blade.php, menu.blade.php, register.blade.php, search.blade.php
- Thư mục noti (view thông báo): error.blade.php, thôngbao.blade.php

CHƯƠNG 4: KẾT QUẢ CHƯƠNG TRÌNH

4.1 GIAO DIỆN TRANG ADMIN

4.1.1 Trang đăng nhập admin

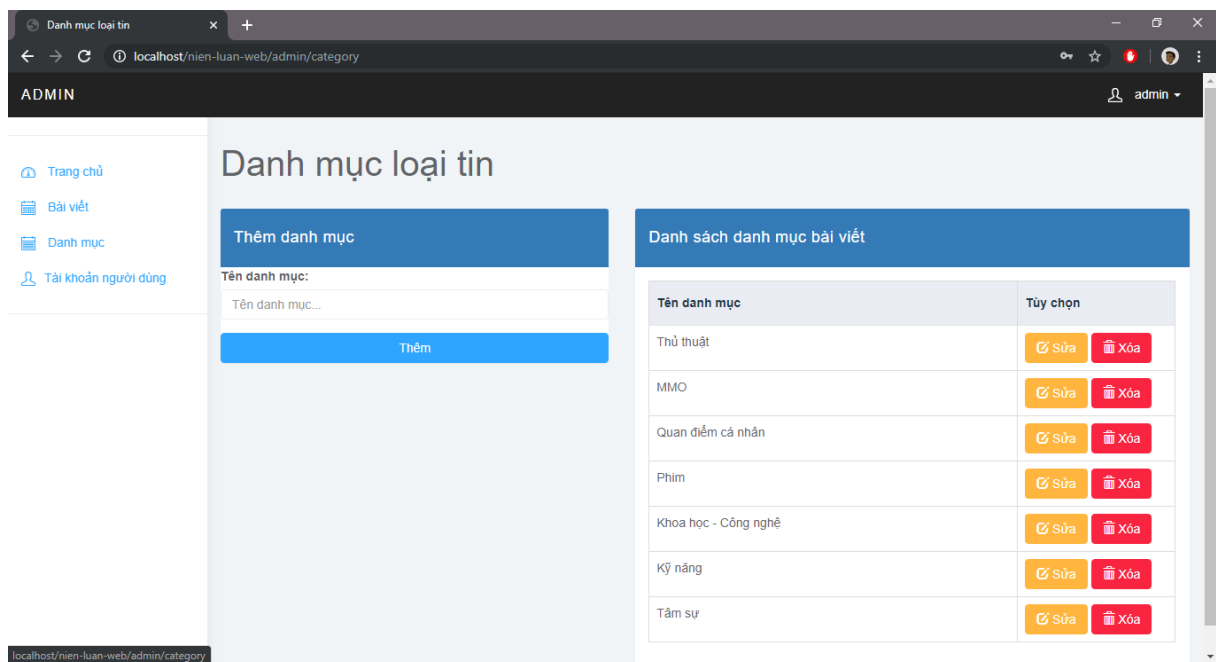
Trang đăng nhập admin truy cập bằng đường dẫn: **/admin/login**



Hình 8: Giao diện trang đăng nhập admin

4.1.2 Trang quản lý danh mục bài viết

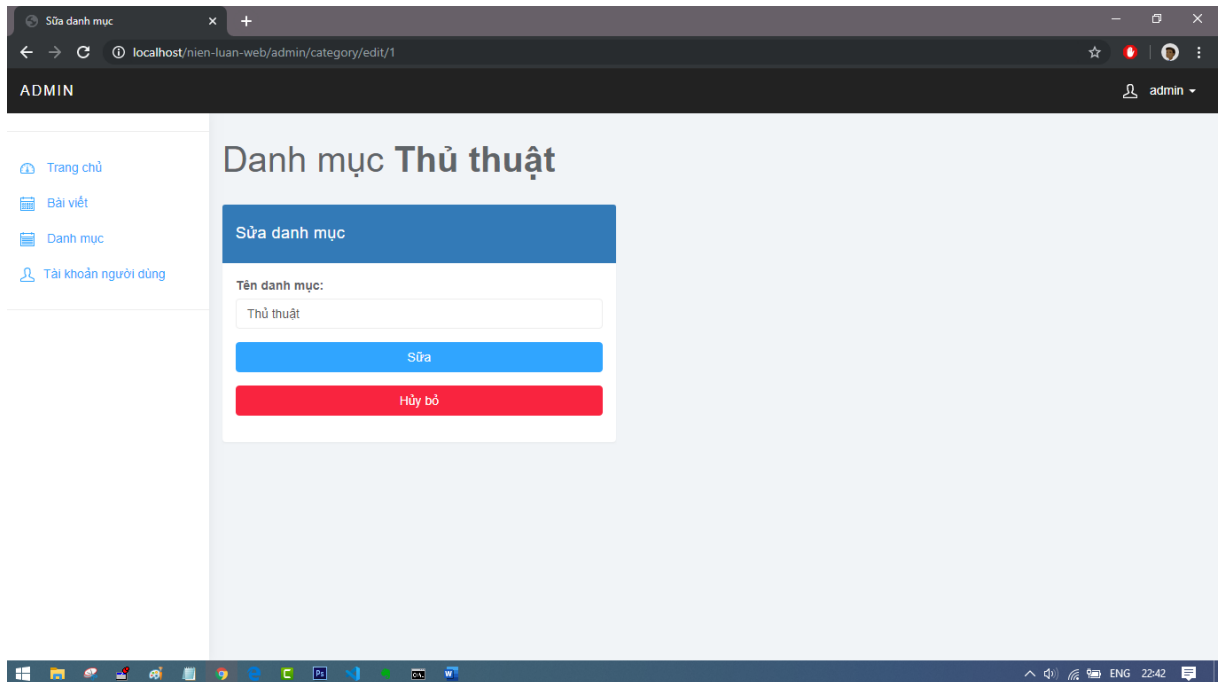
Sau khi đăng nhập thành công nó sẽ chuyển hướng sang trang *danh mục loại tin* có đường dẫn: **/admin/category**. Bên trong trang *danh mục loại tin* có hiển thị danh sách các loại tin và ta có thể sửa và xóa chúng. Góc bên trái là form thêm danh mục loại tin mới



Tên danh mục	Tùy chọn
Thủ thuật	Sửa Xóa
MMO	Sửa Xóa
Quan điểm cá nhân	Sửa Xóa
Phim	Sửa Xóa
Khoa học - Công nghệ	Sửa Xóa
Kỹ năng	Sửa Xóa
Tâm sự	Sửa Xóa

Hình 9: Giao diện trang quản lý danh mục bài viết

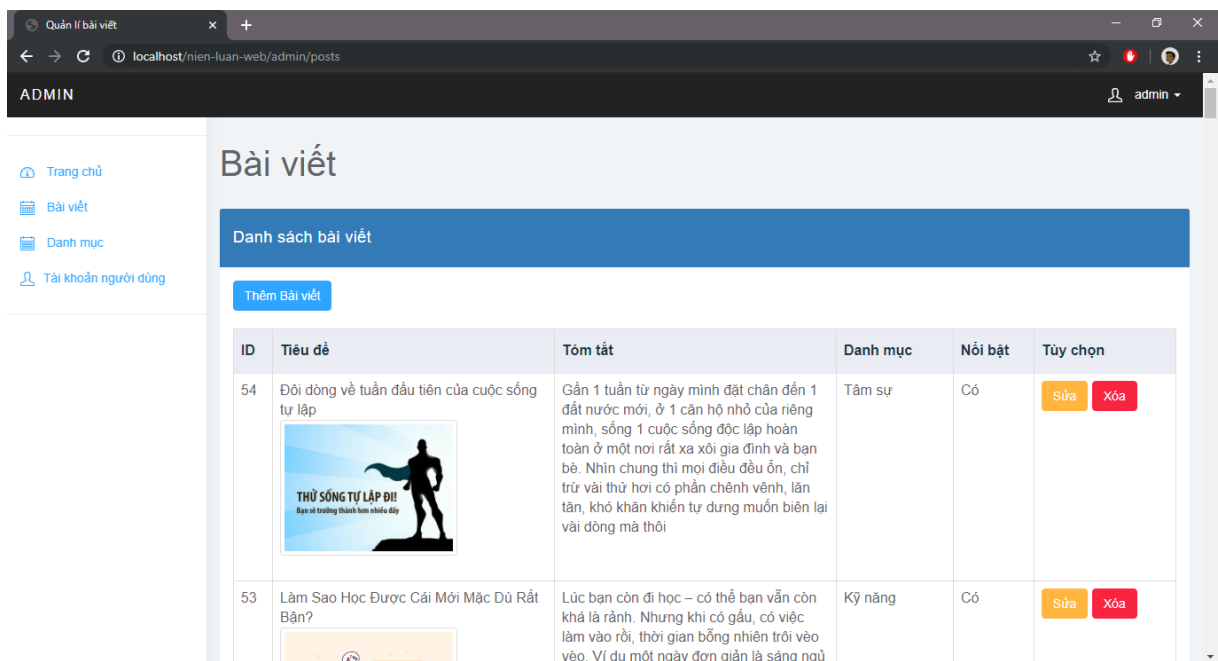
Hình bên dưới là form để sửa *danh mục loại tin* sau khi chúng ta click vào nút sửa. Đường dẫn của chúng là `/admin/category/edit/{id}` với id chính là id của danh mục loại tin trong bảng category trong cơ sở dữ liệu.



Hình 10: Giao diện trang sửa danh mục

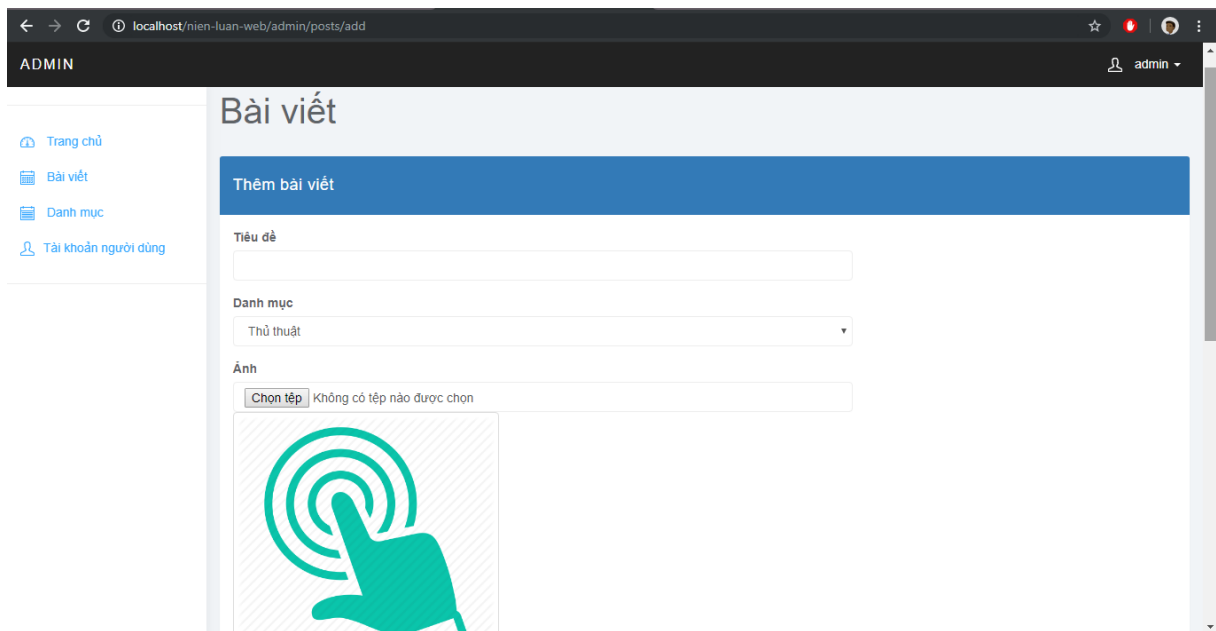
4.1.3 Trang quản lý bài viết

Sau khi click vào menu có liên kết là Bài viết, nó sẽ chuyển hướng cho ta đến trang *Bài viết*. Trang bài viết có đường dẫn là `/admin/posts`. Bên trong có chứa tiêu đề, tóm tắt, hình ảnh,... Mỗi bài viết đều có nút sửa và xóa để người quản trị viên có thể trực tiếp thao tác trên đó



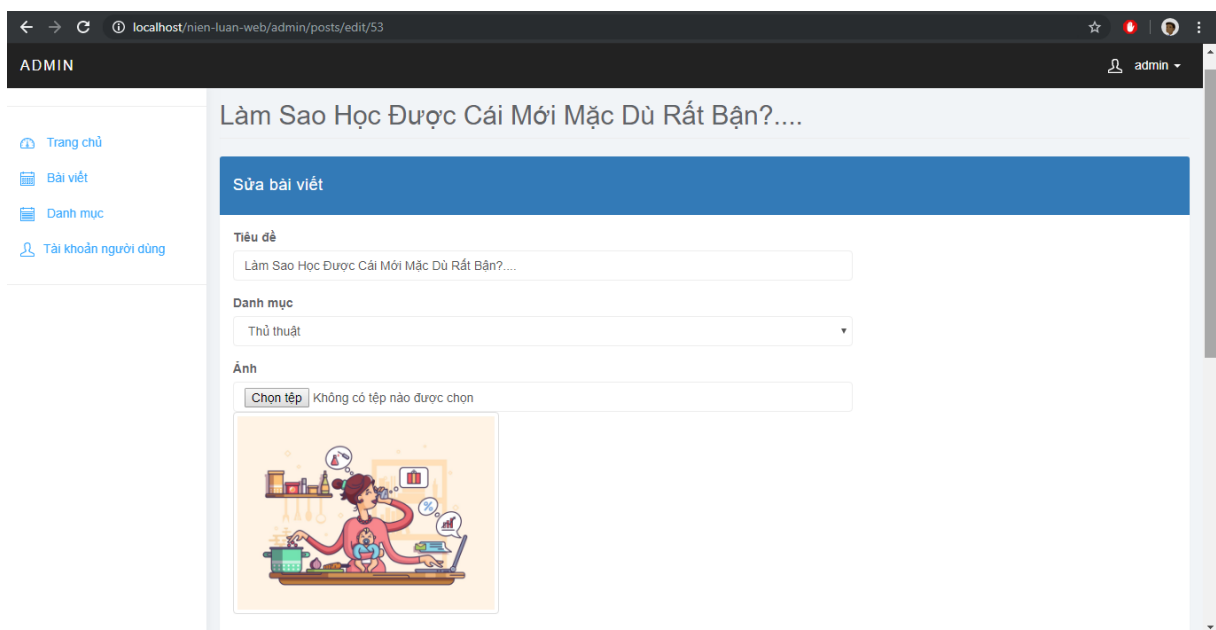
Hình 11 :Giao diện trang quản lý bài viết

Trang thêm bài viết có đường dẫn là: **admin/posts/add**. Tất cả các form đều phải nhập đúng định dạng theo đúng yêu cầu. Nếu nhập sai thì sẽ báo lỗi, yêu cầu phải nhập lại

The screenshot shows a web browser at localhost/nien-luan-web/admin/posts/add. The page is titled 'ADMIN' and 'Bài viết'. On the left is a sidebar with links: 'Trang chủ', 'Bài viết', 'Danh mục', and 'Tài khoản người dùng'. The main content area is titled 'Thêm bài viết'. It contains three form fields: 'Tiêu đề' (Title) with an empty text input, 'Danh mục' (Category) with a dropdown menu showing 'Thủ thuật', and 'Ảnh' (Image) with a 'Chọn tệp' (Choose file) button and the text 'Không có tệp nào được chọn'. Below the image field is a placeholder image of a hand pointing at a target.

Hình 12: Giao diện trang thêm mới bài viết

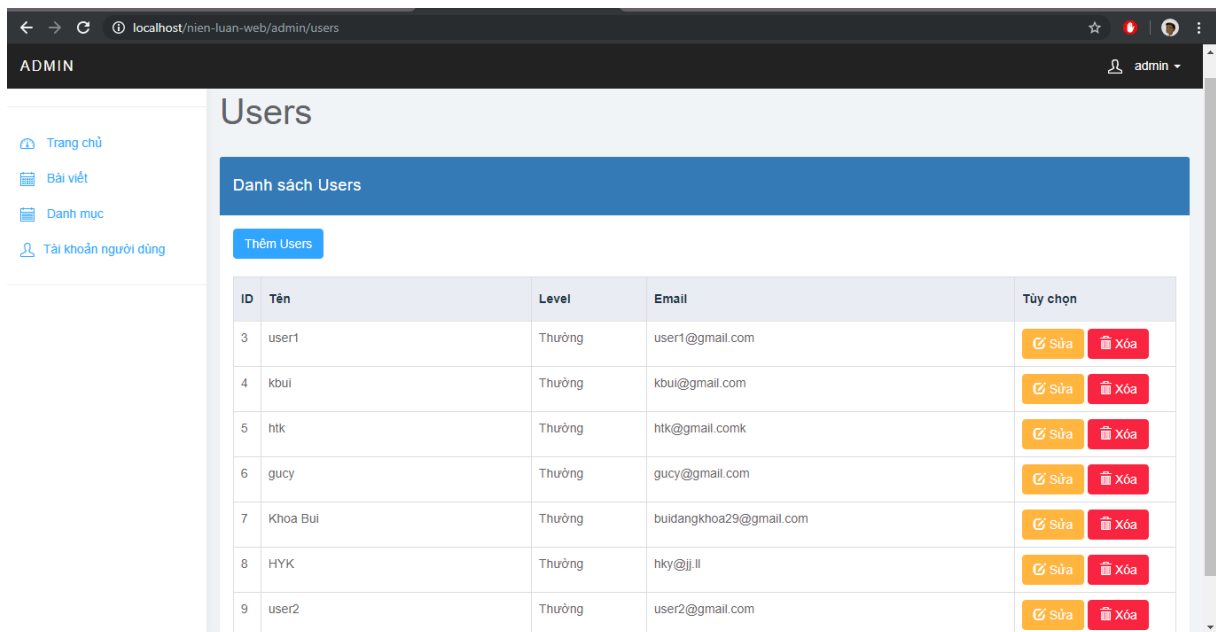
Trang sửa bài viết có giao diện tương tự như trang thêm bài viết. Trang này có đường dẫn là: **admin/posts/edit/{id}** với id chính là id của bài viết

The screenshot shows a web browser at localhost/nien-luan-web/admin/posts/edit/53. The page is titled 'ADMIN' and 'Làm Sao Học Được Cái Mới Mặc Dù Rất Bận?....'. On the left is the same sidebar as in Figure 12. The main content area is titled 'Sửa bài viết'. It contains three form fields: 'Tiêu đề' (Title) with a text input containing 'Làm Sao Học Được Cái Mới Mặc Dù Rất Bận?....', 'Danh mục' (Category) with a dropdown menu showing 'Thủ thuật', and 'Ảnh' (Image) with a 'Chọn tệp' (Choose file) button and the text 'Không có tệp nào được chọn'. Below the image field is a placeholder image of a person cooking at a stove.

Hình 13: Giao diện trang sửa bài viết

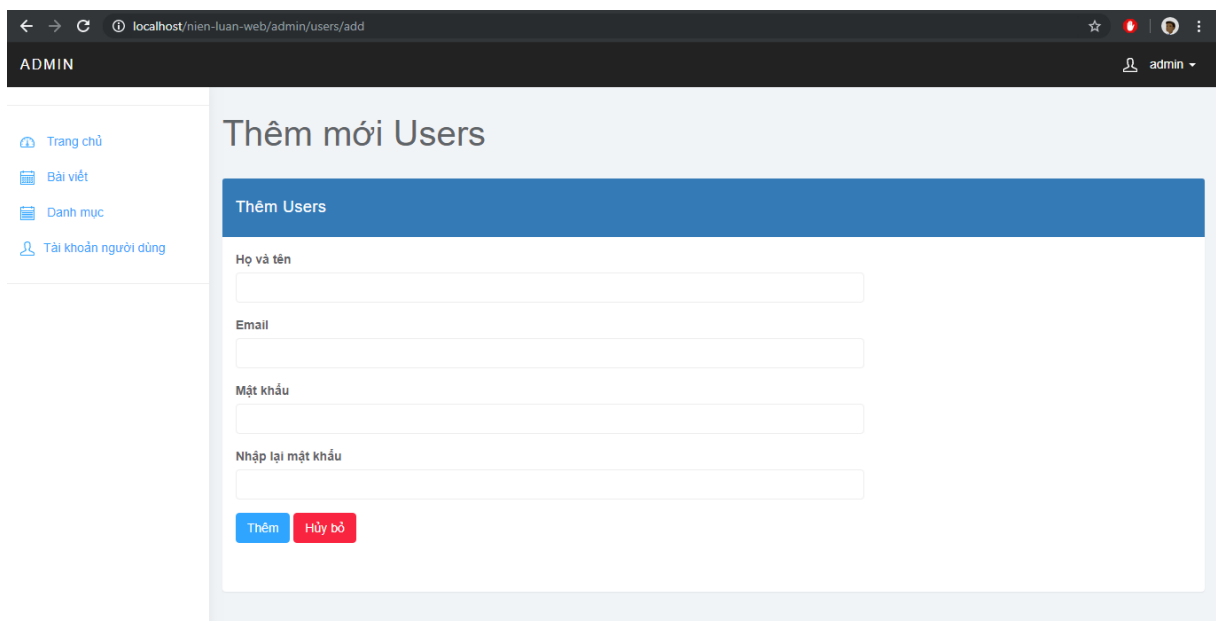
4.1.4 Quản lý tài khoản người dùng

Hình ảnh bên dưới là trang quản lý người dùng, trang này có đường dẫn là: **admin/user**. Trang quản lý người dùng sẽ hiển thị tất cả những tài khoản của bảng **user** trong cơ sở dữ liệu



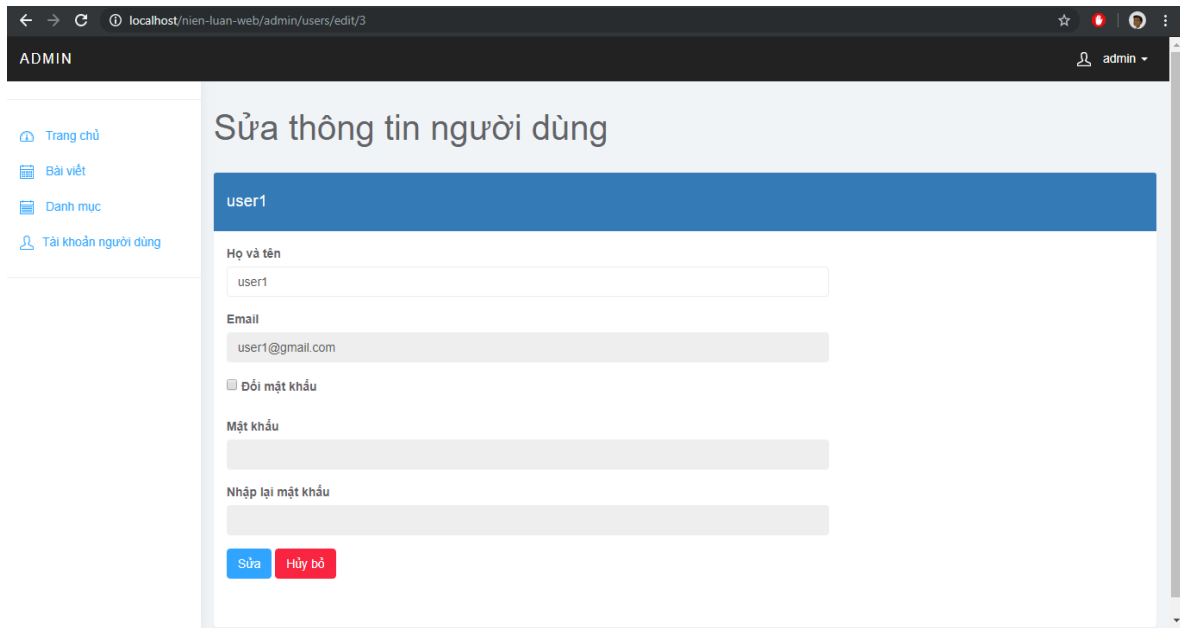
Hình 14: Giao diện trang quản lý users

Trang thêm users có đường là: **admin/users/add**. Trang này yêu cầu cần phải nhập đầy đủ thông tin và đúng theo yêu cầu. Nếu nhập không đúng sẽ báo lỗi và sẽ không được chấp nhận để lưu vào trong cơ sở dữ liệu.



Hình 15: Trang tạo tài khoản người dùng mới

Trang sửa tài khoản người dùng có đường dẫn là **admin/users/edit/{id}** với id chính là id của users. Tuy nhiên chức năng này không được khuyến khích sử dụng vì nó có thể vi phạm quyền riêng tư của người khác. Chỉ được sử dụng khi cần khôi phục lại mật khẩu lỗi khi người chủ tài khoản quên.



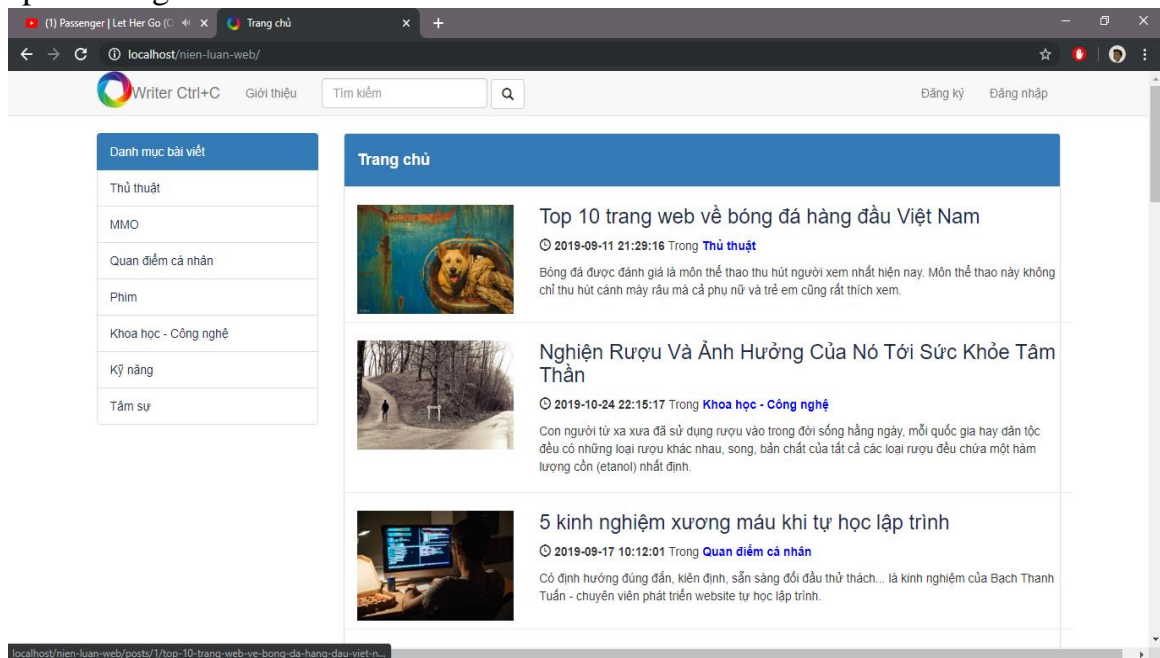
Hình 16: Trang sửa tài khoản người dùng

4.2 TRANG HIỂN THỊ

Bố cục cho trang này được sử dụng cho hầu hết các trang trong đây. Từ phần header, menu và footer đều không thay đổi giữa các trang.

4.2.1 Trang chủ

Trang chủ được truy cập bằng đường dẫn gốc (/), trang này sẽ hiển thị những bài viết nổi bật của từng thể loại. Mỗi trang gồm có 15 bài viết được sắp xếp ngẫu nhiên rồi đó sẽ phân trang.



Hình 17: Trang chủ

4.2.2 Trang đăng ký và đăng nhập

Truy cập trang đăng ký ở trên thanh header hoặc truy cập bằng đường dẫn **/register**. Trang này cho phép mọi người tạo tài khoản để đăng nhập vào hệ thống

Đăng ký tài khoản

Họ tên

Email

Nhập mật khẩu

Nhập lại mật khẩu

Đăng ký

Copyright © Your Website 2019

Hình 18: Trang đăng kí

Sau khi đăng ký thành công nó sẽ chuyển hướng sang trang đăng nhập. Chỉ có đăng nhập bạn mới sử dụng được tính năng bình luận.

Đăng nhập

Email

Mật khẩu

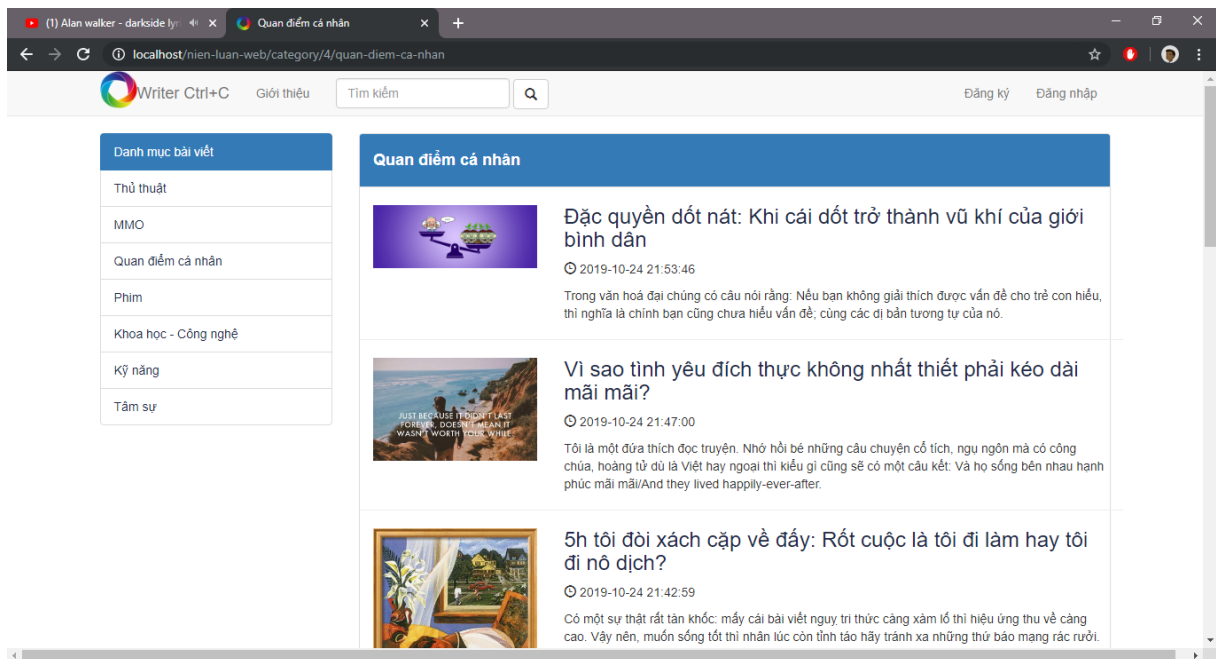
Đăng nhập

Copyright © Your Website 2019

Hình 19: Trang đăng nhập

4.2.3 Trang danh mục

Trang danh mục loại tin sẽ hiển thị những bài viết thuộc thể loại đó. Những bài viết sẽ được sắp xếp theo id từ cao xuống thấp. Đường dẫn của trang này là **/category/{id}/{tên không dấu}** với id chính là id của thể loại bài viết. Cấu trúc trang danh mục tương tự như trang chủ, mỗi trang sẽ hiển thị 10 tin rồi sau đó phân trang.



Hình 20: Trang danh mục loại tin

4.2.4 Trang nội dung bài viết

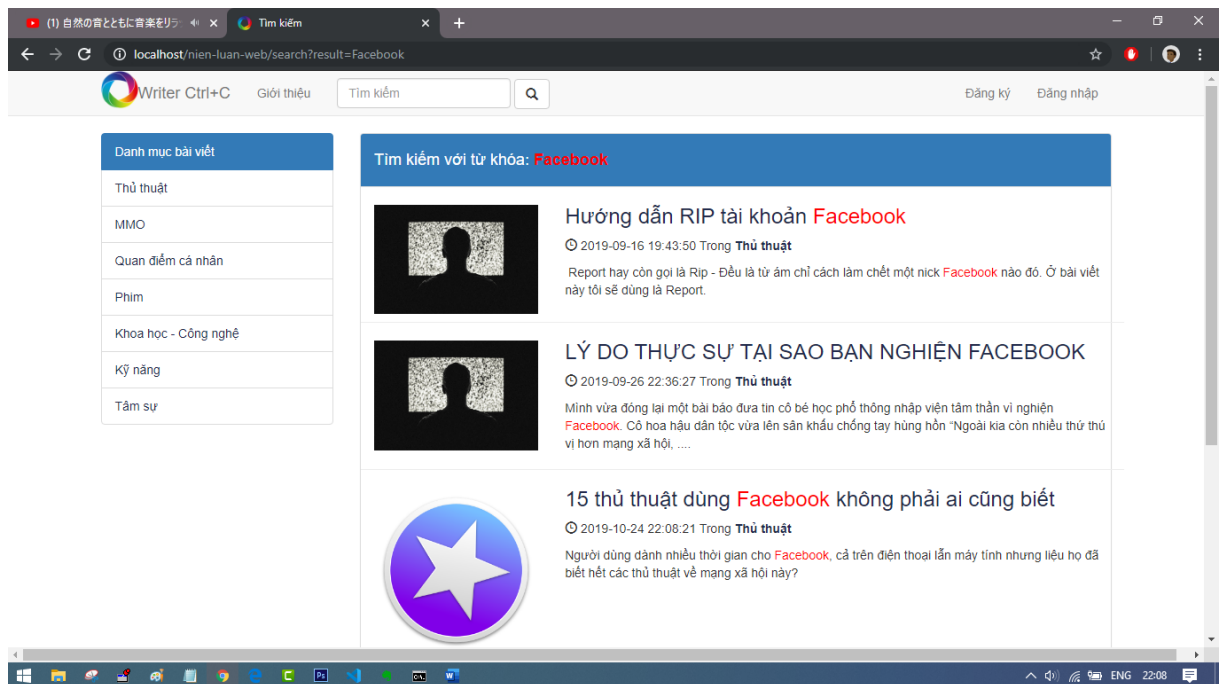
Trang bài viết được truy cập thông qua danh sách những bài viết trên trang chủ hoặc trang mục bài viết. Đường dẫn cho trang bài viết là **/posts/{id}/{tên không dấu của tiêu đề bài viết}**. Trong trang bài viết có hiển thị tiêu đề bài viết trên tab của trình duyệt lẫn trong trang. Mặc khác nó còn hiển thị cho ta thấy danh mục thuộc loại bài viết đó lần thòi gian đăng và cập nhật bài viết. Góc bên phải hiển thị những bài viết có cùng thể loại.



Hình 21: Giao diện trang nội dung bài viết

4.2.5 Trang tìm kiếm

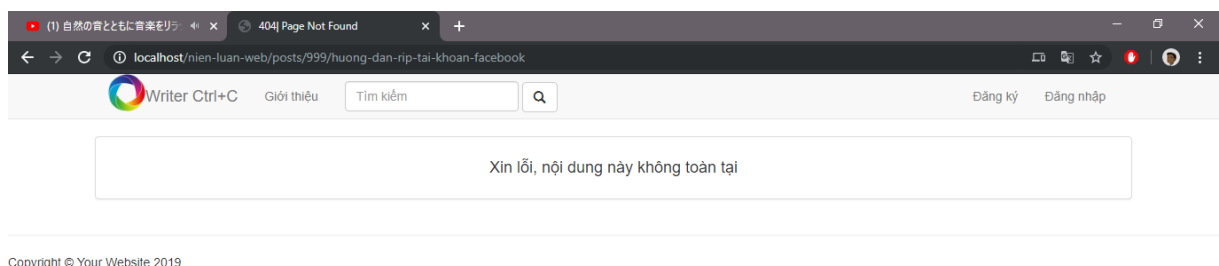
Tìm kiếm với từ khóa ở trên thanh header của bất kì trang nào. Kết quả tìm kiếm dựa vào từ khóa vừa tìm được khớp với những từ có trên tiêu đề và tóm tắt của bảng posts trong cơ sở dữ liệu



Hình 22: Giao diện trang tìm kiếm

4.2.6 Trang báo lỗi

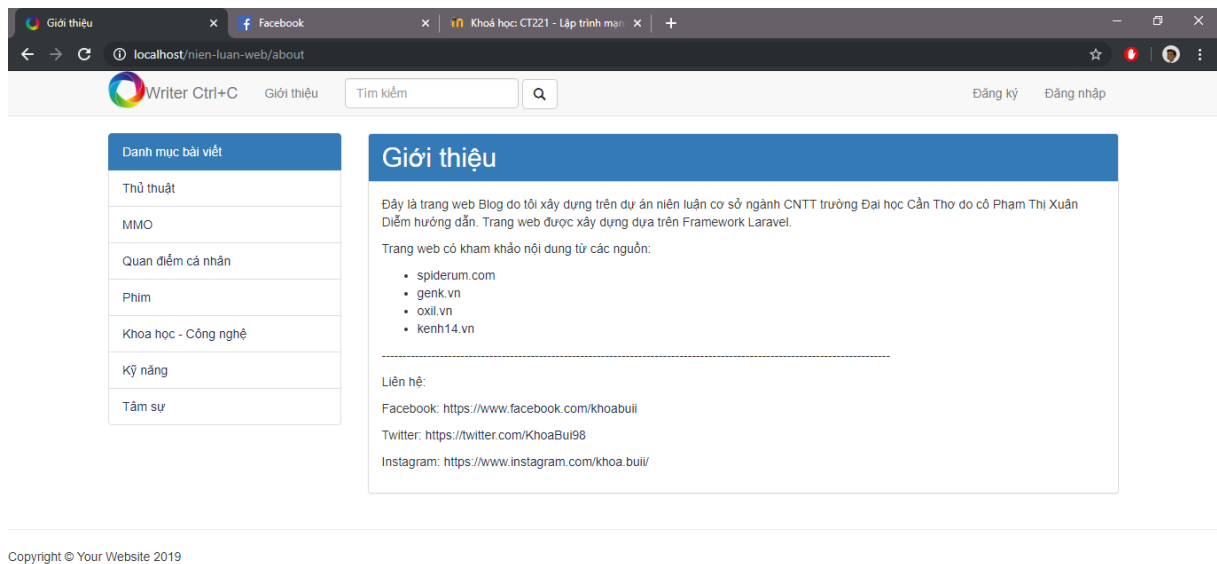
Giả sử khi bạn truy cập vào đường dẫn sai hoặc truy cập đúng đường dẫn nhưng sai về mặt logic. Ví dụ như trang bài viết có đường dẫn là <http://localhost/nien-luan-web/posts/999/huong-dan-rip-tai-khoan-facebook> sẽ truy cập không được do trong cơ sở dữ liệu không có id bằng 999. Nếu như không xây dựng trang báo lỗi, Laravel sẽ hiển thị ra toàn bộ mã nguồn của bên server. Nếu điều đó xảy ra sẽ là một điều tai hại, nó sẽ làm mất đi trải nghiệm tốt cho người dùng.



Hình 23: Giao diện trang báo lỗi

4.2.7 Trang giới thiệu

Trang này có đường dẫn là <http://localhost/nien-luan-web/about>, đây là nơi giới thiệu về trang web.



Hình 24: Giao diện trang giới thiệu

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 KẾT LUẬN

Sau khi hoàn thành xong website này thì đề tài đã đạt những thành công sau đây:

- Giao diện trực quan sinh động phù hợp với những người không am hiểu nhiều về công nghệ
- Đảm bảo an toàn và tính toàn vẹn của hệ thống
- Xây dựng cơ bản đầy đủ các chức năng mà một trang blog bình thường cần phải có
- Mặc khác sau khi thực hiện xong đề tài này đã giúp tôi hiểu thêm được nhiều vấn đề như sau:
- Hiểu được mô hình MVC trong phát triển website
- Từng bước nắm vững những thành phần trong Laravel
- Biết cách sử dụng ngôn ngữ PHP trong việc giải quyết vấn đề xây dựng web

Mặc dù đề tài đã đạt được một số thành công nhưng cũng không tránh những hạn chế nhất định, cụ thể như sau:

- Nội dung bài viết chưa phong phú
- Chưa xây dựng được cơ chế chứng thực email mà người dùng đăng kí tài khoản và reset mật khẩu khi họ quên mật khẩu của mình

5.2 HƯỚNG PHÁT TRIỂN

Đề xuất hướng phát triển trong tương lai:

- Sử dụng tài khoản Google hoặc Facebook để đăng nhập vào hệ thống website, người dùng không cần phải tạo tài khoản bằng cách thủ công thông thường như trước kia
- Người dùng có thể đóng góp nội dung bài viết cho website và đồng thời người quản trị viên kiểm duyệt chúng để bài viết đó được đăng lên hệ thống
- Bổ sung thêm thông tin tài khoản cho người như ảnh đại diện, giới thiệu bản thân, sở thích, giới tính, tài khoản mạng xã hội khác
- Thêm tính năng Like (thích) bài viết, bình luận mà người dùng cảm thấy thích, đồng thời website cần phải có cơ chế đếm view (lượt xem bài viết)
- Bổ sung tính năng bình luận bằng hình ảnh và trả lời bình luận.

TÀI LIỆU THAM KHẢO

- [1]
<https://techmaster.vn/posts/34646/lap-trinh-laravel-cho-nguoi-moi-bat-dau>
- [2]
<https://hoclom.com/hoc-laravel/tim-hieu-ve-mo-hinh-mvc-trong-laravel/>
- [3]
<https://viblo.asia/p/middleware-trong-laravel-MLzGOBKLvpq>
- [4]
<https://hocvps.com/cai-dat-laravel/>

- <https://laravel.com/docs/5.8>
- <https://stackoverflow.com/>