.

KHOA CÔNG NGHỆ THÔNG TIN

BỘ MÔN CÔNG NGHỆ PHẦN MỀM

\*

Icon

Description automatically generated

BÁO CÁO ĐỒ ÁN MÔN HỌC

**Thiết kế kiến trúc phần mềm**

**Đề tài**

**Blog Website**

Nhóm sinh viên thực hiện: Nhóm 9

DSSV: **Đỗ Minh Đức - 0187266**

**Nguyễn Đức Nguyên - 0198966**

Lớp học phần: **66PM**

Giảng viên hướng dẫn: **Phạm Hữu Tùng**

Hà Nội 5/2025

Mục lục

[Chương 1: Giới thiệu tổng quan về đề tài 4](#_Toc199200493)

[1.1. Lý do chọn đề tài 4](#_Toc199200494)

[1.2. Mục tiêu của đồ án 4](#_Toc199200495)

[1.3. Phạm vi của hệ thống 4](#_Toc199200496)

[1.3.1. Đối với người dùng ( User ) 4](#_Toc199200497)

[1.3.2. Đối với quản trị viên ( Admin ) 5](#_Toc199200498)

[1.3.3. Về mặt kỹ thuật 5](#_Toc199200499)

[1.4. Giới hạn của hệ thống 5](#_Toc199200500)

[1.4.1. Tính năng chưa hỗ trợ 5](#_Toc199200501)

[1.4.2. Bảo mật ở mức căn bản 5](#_Toc199200502)

[1.4.3. Giao diện và trải nghiệm người dùng 5](#_Toc199200503)

[1.4.4. Quy mô dữ liệu nhỏ 5](#_Toc199200504)

[1.5. Phương pháp thực hiện 6](#_Toc199200505)

[1.5.1. Thu thập và phân tích yêu cầu 6](#_Toc199200506)

[1.5.2. Thiết kế hệ thống 6](#_Toc199200507)

[1.5.3. Cài đặt ( lập trình hệ thống ) 6](#_Toc199200508)

[1.5.4. Kiểm thử hệ thống 6](#_Toc199200509)

[1.5.5. Đánh giá và hoàn thiện 6](#_Toc199200510)

[Chương 2: Phân tích hệ thống 7](#_Toc199200511)

[2.1. Mô tả bài toán 7](#_Toc199200512)

[2.2. Yêu cầu chức năng 7](#_Toc199200513)

[2.3. Yêu cầu phi chức năng 7](#_Toc199200514)

[2.4. Use Case Diagram 7](#_Toc199200515)

[2.4.1. Use Case Diagram tổng quát 8](#_Toc199200516)

[2.4.2. Use Case Diagram “Quản lý bài viết” 9](#_Toc199200517)

[2.4.3. Use Case Diagram “Quản lý danh mục bài viết” 10](#_Toc199200518)

[2.4.4. Use Case Diagram “Quản lý tài khoản người dùng” 11](#_Toc199200519)

[2.5. Mô tả các Use Case chính 12](#_Toc199200520)

[2.5.1. Use case “Xem danh sách bài viết (trang chủ)” 12](#_Toc199200521)

[2.5.2. Use case “Xem chi tiết bài viết” 12](#_Toc199200522)

[2.5.3. Use case “Đăng nhập” 13](#_Toc199200523)

[2.5.4. Use case “Đăng xuất” 14](#_Toc199200524)

[2.5.5. Use case “Gửi liên hệ” 15](#_Toc199200525)

[2.5.6. Use case “Chỉnh sửa thông tin cá nhân” 16](#_Toc199200526)

[2.5.7. Use case “Bình luận bài viết” 17](#_Toc199200527)

[2.5.8. Use case “Thêm bài viết/danh mục bài viết/tài khoản người dùng” 18](#_Toc199200528)

[2.5.9. Use case “Chỉnh sửa bài viết/danh mục bài viết/tài khoản người dùng” 19](#_Toc199200529)

[2.5.10. Use case “Xoá bài viết/danh mục bài viết/tài khoản người dùng” 20](#_Toc199200530)

[2.5.11. Use case “Xem danh sách liên hệ” 20](#_Toc199200531)

[2.6. Biểu đồ hoạt động 21](#_Toc199200532)

[2.6.1. Activity diagram “Đăng nhập” 22](#_Toc199200533)

[2.6.2. Activity diagram “Gửi liên hệ” 23](#_Toc199200534)

[2.6.3. Activity diagram “Bình luận bài viết” 24](#_Toc199200535)

[2.6.4. Activity diagram “Quản trị” 25](#_Toc199200536)

[Chương 3: Thiết kế hệ thống 26](#_Toc199200538)

[3.1. Lựa chọn mô hình kiến trúc 26](#_Toc199200539)

[3.2. Biểu đồ tuần tự 26](#_Toc199200540)

[3.2.1. Sequence diagram “Thêm danh mục” 27](#_Toc199200541)

[3.2.2. Sequence diagram “Chỉnh sửa danh mục” 28](#_Toc199200542)

[3.2.3. Sequence diagram “Xoá danh mục” 29](#_Toc199200543)

[3.2.4. Sequence diagram “Đăng bài viết mới” 30](#_Toc199200544)

[3.2.5. Sequence diagram “Chỉnh sửa bài viết” 31](#_Toc199200545)

[3.2.6. Sequence diagram “Xoá bài viết” 32](#_Toc199200546)

[3.2.7. Sequence diagram “Mở tài khoản” 33](#_Toc199200547)

[3.2.8. Sequence diagram “Chỉnh sửa thông tin tài khoản” 34](#_Toc199200549)

[3.2.9. Sequence diagram “Xoá tài khoản người dùng” 35](#_Toc199200550)

[3.2.10. Sequence diagram “Đăng nhập” 36](#_Toc199200551)

[3.2.11. Sequence diagram “Gửi liên hệ” 37](#_Toc199200552)

[3.2.12. Sequence diagram “Bình luận bài viết” 38](#_Toc199200553)

[3.3. Thiết kế lớp 39](#_Toc199200554)

[3.4. Thiết kế cơ sở dữ liệu 40](#_Toc199200555)

[3.5. Thiết kế giao diện người dùng 40](#_Toc199200556)

[3.5.1. Giao diện phần đầu trang 40](#_Toc199200557)

[3.5.2. Giao diện phần chân trang 41](#_Toc199200558)

[3.5.3. Giao diện trang chủ 41](#_Toc199200559)

[3.5.4. Giao diện danh sách bài viết chia theo danh mục 42](#_Toc199200560)

[3.5.5. Giao diện trang gửi liên hệ 42](#_Toc199200561)

[3.5.6. Giao diện trang đăng nhập(User) 43](#_Toc199200562)

[3.5.7. Giao diện trang đăng nhập(Admin) 43](#_Toc199200563)

[3.5.8. Giao diện trang quản trị 44](#_Toc199200564)

[3.5.9. Giao diện trang chỉnh sửa thông tin cá nhân 44](#_Toc199200565)

[3.5.10. Giao diện trang danh sách liên hệ 45](#_Toc199200566)

[Chương 4: Cài đặt và triển khai hệ thống 46](#_Toc199200567)

[4.1. Môi trường triển khai 46](#_Toc199200568)

[4.2. Cài đặt hệ thống 46](#_Toc199200569)

[Chương 5: Kết quả và đánh giá 47](#_Toc199200570)

[5.1. Kết quả thử nghiệm hệ thống 47](#_Toc199200571)

[5.2. Đánh giá hiệu quả hệ thống 47](#_Toc199200572)

[5.2.1. Ưu điểm 47](#_Toc199200573)

[5.2.2. Hạn chế 47](#_Toc199200574)

**Lời cảm ơn**

Để hoàn thành đồ án này, em xin gửi lời cảm ơn đến tất cả các thầy cô giáo tại Trường Đại Xây dụng Hà Nội đã truyền dạy cho chúng em những nền tảng kiến thức cơ bản giúp chúng em hoàn thành đồ án này. Chúng em xin gửi lời cảm ơn đến các thầy cô đã cùng với tri thức và tâm huyết của mình truyền đạt vốn kiến thức quý báu cho chúng em trong suốt quá trình học tập.

Đặc biệt, xin gửi lời cảm ơn sâu sắc nhất đến thầy Phạm Hữu Tùng, thầy đã trực tiếp hướng dẫn chúng em hoàn thành đồ án này với sự nhiệt tình và ân cần chỉ bảo, đồng thời cung cấp cho em những kiến thức chuyên môn để em có thể hoàn thiện đồ án này. Cuối cùng, chúng em xin gửi lời cảm ơn chân thành đến gia đình, bạn bè và người thân, những người đã bên cạnh và động viên chúng em trong suốt quá trình học tập và hoàn thành đồ án. Mặc dù chúng em đã rất cố gắng nhưng trong quá trình làm đồ án sẽ không tránh khỏi những thiếu sót. Rất mong nhận được sự góp ý quý báu của quý thầy cô và bạn bè để đồ án được hoàn chỉnh hơn.

Hà Nội, ngày 25 tháng 5 năm 2025

# Chương 1: Giới thiệu tổng quan về đề tài

# Lý do chọn đề tài

* Blog là nền tảng lý tưởng để chia sẻ kiến thức, kinh nghiệm cá nhân hoặc chuyên môn trong lĩnh vực cụ thể như công nghệ, giáo dục, sức khỏe,...Giúp lan tỏa thông tin hữu ích đến cộng đồng một cách có hệ thống.
* Một blog chuyên nghiệp giúp khẳng định uy tín, phong cách viết và khả năng chuyên môn.
* Blog giúp lưu trữ thông tin theo thời gian, người đọc có thể tra cứu lại khi cần.
* Việc thiết kế và vận hành blog giúp cải thiện các kỹ năng như: viết nội dung, thiết kế UI/UX, lập trình web, quản trị hệ thống,...
* Blog có thể tích hợp quảng cáo (Google AdSense), liên kết tiếp thị (affiliate), hoặc bán sản phẩm/dịch vụ trực tiếp.
* Blog cho phép người đọc bình luận, góp ý, tạo ra sự tương tác hai chiều giữa người viết và người đọc.

# Mục tiêu của đồ án

* **Xây dựng một nền tảng trực tuyến** cho phép người dùng tạo, đăng, quản lý và chia sẻ bài viết (bài blog) một cách dễ dàng, nhanh chóng và hiệu quả.
* **Tạo hệ thống có tính tổ chức** rõ ràng về mặt nội dung, dễ sử dụng và dễ mở rộng trong tương lai.

# Phạm vi của hệ thống

Hệ thống blog website được thiết kế với các chức năng chính sau:

# Đối với người dùng ( User )

* Đăng nhập/đăng xuất hệ thống.
* Xem danh sách các bài viết blog.
* Xem chi tiết bài viết.
* Lọc bài viết theo thể loại.
* Phân trang danh sách bài viết.

# Đối với quản trị viên ( Admin )

* Đăng nhập vào hệ thống quản trị.
* Quản lý **danh mục bài viết** (thêm, sửa, xoá).
* Quản lý **bài viết/blog** (thêm, sửa, xoá).
* Giao diện quản trị đơn giản, dễ sử dụng.

# Về mặt kỹ thuật

* Thiết kế hệ thống theo mô hình **MVC (Model - View - Controller).**
* Sử dụng **PHP Laravel**, kết hợp **MySQL** làm hệ quản trị cơ sở dữ liệu.
* Giao diện thiết kế bằng **HTML/CSS đơn giản** và thân thiện với người dùng.

# Giới hạn của hệ thống

Mặc dù hệ thống đáp ứng các chức năng cơ bản, nhưng có một số giới hạn như sau:

# Tính năng chưa hỗ trợ

* **Chưa có tính năng trả lời bình luận giữa các người dùng**.
* **Chưa hỗ trợ đa ngôn ngữ** hoặc thay đổi giao diện theo người dùng.
* **Chưa tích hợp SEO** hoặc chia sẻ bài viết qua mạng xã hội.

# Bảo mật ở mức căn bản

* Hệ thống mới chỉ áp dụng một số kỹ thuật bảo mật đơn giản như kiểm tra đăng nhập bằng session.
* Chưa xử lý sâu các lỗ hổng như CSRF, XSS, hay mã hóa mật khẩu nâng cao.

# Giao diện và trải nghiệm người dùng

* Giao diện đơn giản, chưa tối ưu.
* Không có tính năng cá nhân hóa trải nghiệm người dùng.

# Quy mô dữ liệu nhỏ

* Thiết kế phục vụ mục đích học tập hoặc cá nhân, không tối ưu cho lượng dữ liệu lớn hay nhiều người truy cập đồng thời.

# Phương pháp thực hiện

Quá trình xây dựng hệ thống blog website được thực hiện theo các bước chính như sau:

# Thu thập và phân tích yêu cầu

* Tìm hiểu nhu cầu thực tế về chức năng của một hệ thống blog.
* Xác định các đối tượng sử dụng (admin, người dùng).
* Phân tích các chức năng chính: đăng nhập, quản lý bài viết, thể loại, hiển thị bài viết,...
* Xác định các ràng buộc kỹ thuật (ngôn ngữ lập trình, CSDL, giao diện,...).

# Thiết kế hệ thống

* Thiết kế kiến trúc tổng thể: Áp dụng mô hình **MVC (Model - View - Controller)** để phân chia logic rõ ràng.
* Thiết kế cơ sở dữ liệu.
* Thiết kế giao diện người dùng.

# Cài đặt ( lập trình hệ thống )

* Sử dụng **PHP Laravel** để lập trình backend.
* Sử dụng **MySQL** để xây dựng cơ sở dữ liệu.
* Tạo các chức năng.
* Kết nối CSDL.

# Kiểm thử hệ thống

* **Kiểm thử chức năng**: đảm bảo các chức năng như thêm, sửa, xoá hoạt động đúng.
* **Kiểm thử giao diện**: xem trên nhiều trình duyệt, thiết bị khác nhau.
* **Kiểm thử dữ liệu:** thử với nhiều loại dữ liệu để đảm bảo không lỗi.
* **Kiểm thử đăng nhập**: đúng sai tài khoản, bảo mật session.

# Đánh giá và hoàn thiện

* Rà soát, sửa lỗi, tối ưu mã nguồn.
* Ghi nhận những phần còn hạn chế.
* Đề xuất hướng phát triển trong tương lai.

# Chương 2: Phân tích hệ thống

# Mô tả bài toán

Trong thời đại công nghệ thông tin phát triển mạnh mẽ, việc chia sẻ thông tin, kiến thức và ý tưởng thông qua internet ngày càng phổ biến. Một trong những phương tiện hiệu quả để thực hiện điều này là thông qua **blog** – một dạng nhật ký trực tuyến cho phép người dùng đăng tải bài viết, hình ảnh, và tương tác với người đọc.

Tuy nhiên, nhiều người dùng cá nhân hoặc tổ chức nhỏ chưa có hệ thống blog phù hợp, dễ sử dụng, có thể tùy chỉnh theo nhu cầu. Do đó, cần thiết kế một hệ thống blog đơn giản, dễ quản lý và có thể phát triển thêm trong tương lai.

# Yêu cầu chức năng

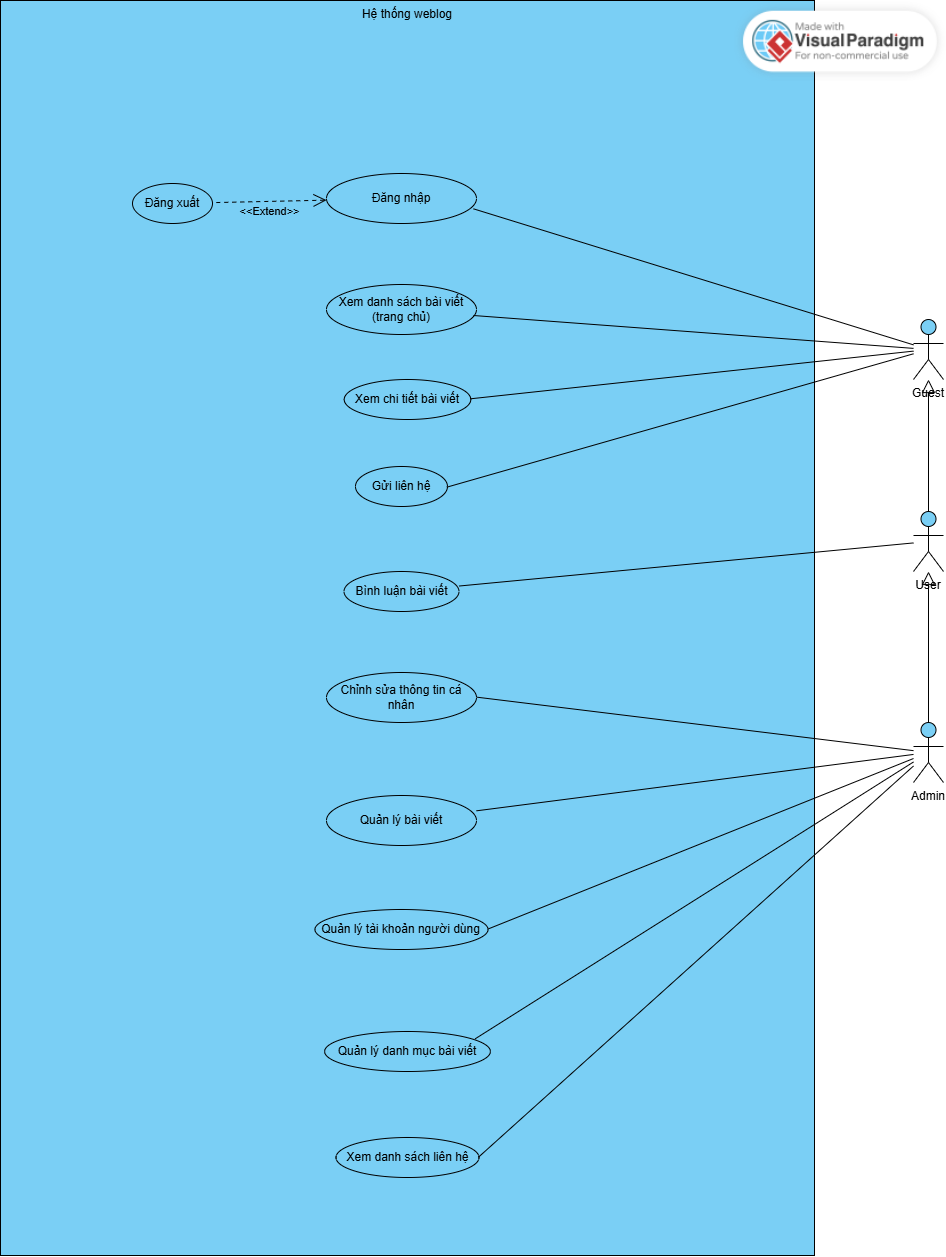
* Cho phép người dùng, quản trị viên đăng nhập vào hệ thống.
* Quản lý thông tin **danh mục** (thêm, sửa, xóa).
* Quản lý thông tin **bài viết/blog** (thêm, sửa, xóa, phân loại theo thể loại).
* Quản lý thông tin **tài khoản người dùng** (thêm, sửa, xóa).
* Hiển thị danh sách các bài viết ra giao diện người dùng, có hỗ trợ **phân trang** và lọc theo danh mục.
* Hiển thị chi tiết nội dung bài viết.
* Chức năng bình luận danh cho dùng đã đăng nhập.
* Chức năng gửi liên hệ danh cho tất cả người dùng truy cập hệ thống.

# Yêu cầu phi chức năng

* **Giao diện thân thiện, dễ sử dụng**, tương thích với nhiều thiết bị.
* **Hiệu năng tốt**: tải nhanh, xử lý yêu cầu hiệu quả.
* **Bảo mật**: bảo vệ thông tin người dùng, tránh tấn công như SQL Injection, XSS,...
* **Dễ bảo trì và mở rộng**: viết code theo mô hình chuẩn (MVC,...), có thể thêm tính năng mới sau này.

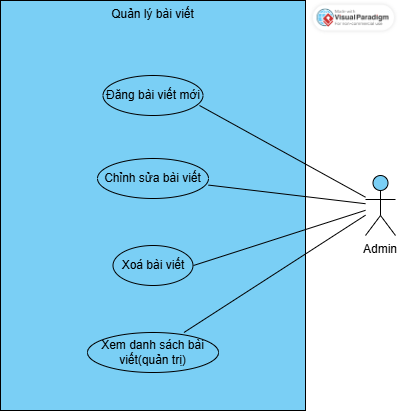
# Use Case Diagram

# Use Case Diagram tổng quát



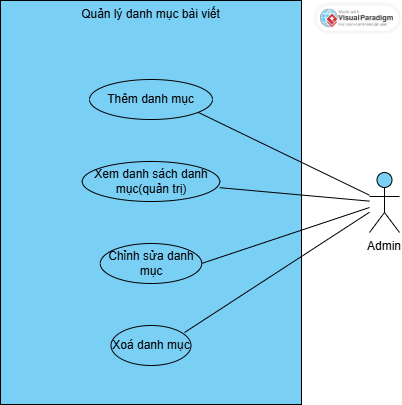
Hình 1-1: Use Case Diagram tổng quát

# Use Case Diagram “Quản lý bài viết”



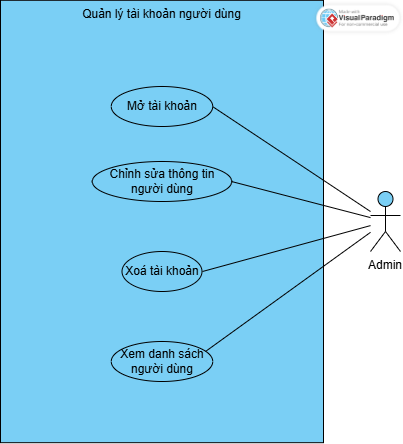
Hình 1-2: Use Case Diagram Quản lý bài viết

# Use Case Diagram “Quản lý danh mục bài viết”



Hình 1-3: Use Case Diagram Quản lý danh mục bài viết

# Use Case Diagram “Quản lý tài khoản người dùng”



Hình 1-4: Use Case Diagram Quản lý tài khoản người dùng

# Mô tả các Use Case chính

Để hiểu rõ hơn về chức năng của hệ thống cũng như cách các tác nhân tương tác với hệ thống, chúng em tiến hành xây dựng và đặc tả các Use Case . Mỗi Use Case sẽ mô tả một chức năng cụ thể mà người dùng hoặc quản trị viên có thể thực hiện, từ đó làm cơ sở cho việc thiết kế và cài đặt hệ thống.

# Use case “Xem danh sách bài viết (trang chủ)”

|  |  |
| --- | --- |
| **ID** | UC1 |
| **Name** | Xem danh sách bài viết(trang chủ) |
| **Brief description** | |  | | --- | |  |  |  | | --- | | Chức năng cho phép người dùng xem danh sách tất cả bài viết có trong hệ thống, không phân loại theo danh mục. | |
| **Actors** | Tất cả |
| **Pre-conditions** | Người dùng đã truy cập vào hệ thống. |
| **Basic flow** | |  |  | | --- | --- | | • Người dùng truy cập vào giao diện chính của trang web. |  |  |  |  | | --- | --- | | • Hệ thống hiển thị danh sách tất cả bài viết hiện có, bao gồm: hình ảnh bài viết, tiêu đề bài viết, mô tả ngắn, ngày đăng. |  |  |  |  | | --- | --- | |  |  |  |  | | --- | | • Khi người dùng chọn một bài viết, hệ thống sẽ chuyển đế trang xem chi tiết nội dung bài viết. | |
| **Alternative flow** |  |
| **Result** | Người dùng xem được danh sách bài viết đang có trên hệ thống và có thể chọn để xem nội dung chi tiết. |
| **Exceptions** | • Kết nối mạng không ổn định, danh sách bài viết không tải được. • Dữ liệu bài viết bị lỗi hoặc không hiển thị chính xác. |

# Use case “Xem chi tiết bài viết”

|  |  |
| --- | --- |
| **ID** | UC2 |
| **Name** | Xem chi tiết bài viết |
| **Brief description** | |  | | --- | |  |  |  |  |  | | --- | --- | --- | | |  | | --- | |  |  |  | | --- | | Chức năng cho phép người dùng xem nội dung chi tiết của một bài viết bao gồm tiêu đề, nội dung, hình ảnh, ngày đăng, và các thông tin liên quan. | | |
| **Actors** | Tất cả |
| **Pre-conditions** | Người dùng đang ở giao diện danh sách bài viết. |
| **Basic flow** | |  |  | | --- | --- | | • Người dùng duyệt danh sách bài viết trên giao diện chính. |  |  |  |  | | --- | --- | | • Người dùng chọn một bài viết bất kỳ bằng cách nhấn vào tiêu đề hoặchình ảnh đại diện. |  |  |  |  | | --- | --- | | • Hệ thống tải và hiển thị trang chi tiết bài viết, bao gồm: tiêu đề, nội dung đầy đủ, hình ảnh (nếu có), ngày đăng, tên tác giả, thể loại. |  |  |  | | --- | | • Người dùng có thể đọc nội dung, bình luận(khi đã đăng nhập). | |
| **Alternative flow** | • Nếu người dùng truy cập qua đường dẫn trực tiếp (ví dụ từ bên ngoài), hệ thống kiểm tra ID bài viết và hiển thị nội dung.  • Nếu bài viết đã bị xóa hoặc không tồn tại, hệ thống hiển thị thông báo lỗi. |
| **Result** | Người dùng xem được nội dung đầy đủ của bài viết đã chọn. |
| **Exceptions** | • Bài viết không tồn tại hoặc đã bị xóa.  • Lỗi kết nối mạng khiến bài viết không thể hiển thị.  • Dữ liệu bài viết bị lỗi (ví dụ: thiếu nội dung, lỗi mã hóa ký tự…). |

# Use case “Đăng nhập”

|  |  |
| --- | --- |
| **ID** | UC3 |
| **Name** | |  | | --- | |  |  |  | | --- | | Đăng nhập | |
| **Brief description** | |  | | --- | |  |  |  |  |  | | --- | --- | --- | | |  | | --- | |  |  |  | | --- | | Chức năng cho phép người dùng đăng nhập vào tài khoản đã được quản trị viên tạo sẵn để sử dụng các tính năng của hệ thống như quản trị hệ thống(với admin), bình luận(cả web user với admin) | | |
| **Actors** | Người dùng, Quản trị viên |
| **Pre-conditions** | Người dùng đã có tài khoản hợp lệ và đang ở trạng thái chưa đăng nhập. |
| **Basic flow** | |  |  | | --- | --- | | • Người dùng truy cập vào trang đăng nhập từ giao diện chính. |  |  |  |  | | --- | --- | | • Người dùng nhập email và mật khẩu. |  |  |  |  | | --- | --- | | • Người dùng nhấn nút "Đăng nhập". |  |  |  |  | | --- | --- | | • Hệ thống kiểm tra tính hợp lệ của thông tin đăng nhập. |  |  |  | | --- | | • Nếu hợp lệ, hệ thống cấp quyền truy cập, tạo phiên làm việc (session), và chuyển đến trang chính hoặc trang người dùng yêu cầu trước đó. | |
| **Alternative flow** | • Người dùng nhập sai email hoặc mật khẩu → hệ thống hiển thị thông báo lỗi và cho phép nhập lại. |
| **Result** | Người dùng đăng nhập thành công và có thể sử dụng các chức năng dành cho tài khoản đã xác thực. |
| **Exceptions** | • Lỗi kết nối mạng khiến không thể xác thực tài khoản.  • Hệ thống đang bảo trì hoặc có lỗi server. |

# Use case “Đăng xuất”

|  |  |
| --- | --- |
| **ID** | UC4 |
| **Name** | |  | | --- | |  |  |  | | --- | | Đăng xuất | |
| **Brief description** | Chức năng cho phép người dùng kết thúc phiên làm việc và rời khỏi hệ thống một cách an toàn. |
| **Actors** | Người dùng, Quản trị viên |
| **Pre-conditions** | |  | | --- | |  |  |  | | --- | | Người dùng đã đăng nhập vào hệ thống. | |
| **Basic flow** | |  |  | | --- | --- | | • Người dùng chọn chức năng "Đăng xuất" từ giao diện của hệ thống ( ở menu, góc phải trên cùng, v.v.). |  |  |  |  | | --- | --- | | • Hệ thống xóa thông tin phiên làm việc (session). |  |  |  |  | | --- | --- | | • Hệ thống chuyển người dùng về trang chính hoặc trang đăng nhập. |  |  |  | | --- | | • Trạng thái người dùng chuyển về "khách" (guest). | |
| **Alternative flow** | • Người dùng đóng trình duyệt mà không chọn “Đăng xuất” → hệ thống tự động kết thúc session sau thời gian không hoạt động (timeout). |
| **Result** | Người dùng đăng xuất khỏi hệ thống, session bị hủy, không thể truy cập các chức năng yêu cầu xác thực trừ khi đăng nhập lại. |
| **Exceptions** | • Lỗi hệ thống khiến session chưa được xóa hoàn toàn (hiếm). • Trình duyệt lưu cache làm người dùng vẫn thấy giao diện như đã đăng nhập, nhưng thực tế đã bị đăng xuất (cần tải lại trang). |

# Use case “Gửi liên hệ”

|  |  |
| --- | --- |
| **ID** | UC5 |
| **Name** | |  | | --- | |  |  |  | | --- | | Gửi liên hệ | |
| **Brief description** | Chức năng cho phép người dùng (kể cả chưa đăng nhập) gửi phản hồi, góp ý hoặc câu hỏi đến ban quản trị hệ thống thông qua form liên hệ. |
| **Actors** | Tất cả |
| **Pre-conditions** | |  | | --- | |  |  |  |  |  | | --- | --- | --- | | |  | | --- | |  |  |  | | --- | | Người dùng hoặc khách đang truy cập vào hệ thống (không yêu cầu đăng nhập). | | |
| **Basic flow** | |  |  | | --- | --- | | • Người dùng truy cập trang “Liên hệ” từ menu. |  |  |  |  | | --- | --- | | • Hệ thống hiển thị form liên hệ gồm các trường: Họ tên, Email, Tiêu đề, Nội dung. |  |  |  |  | | --- | --- | | • Người dùng nhập thông tin cần thiết vào các trường. |  |  |  |  | | --- | --- | | • Nhấn nút “Gửi liên hệ”. |  |  |  |  | | --- | --- | | • Hệ thống kiểm tra dữ liệu (ví dụ: email đúng định dạng, nội dung không để trống). |  |  |  |  | | --- | --- | | • Nếu hợp lệ, hệ thống lưu thông tin liên hệ vào cơ sở dữ liệu hoặc gửi đến email quản trị. |  |  |  | | --- | | • Hệ thống hiển thị thông báo gửi thành công. | |
| **Alternative flow** | • Nếu người dùng đang đăng nhập, hệ thống có thể tự động điền sẵn Họ tên và Email từ tài khoản.  • Người dùng có thể hủy thao tác và quay lại trang trước. |
| **Result** | Thông tin liên hệ được gửi đến hệ thống thành công để ban quản trị phản hồi. |
| **Exceptions** | • Thông tin không hợp lệ (email sai định dạng, nội dung trống…) → hệ thống hiển thị lỗi và yêu cầu nhập lại.  • Lỗi kết nối hoặc server khiến không thể gửi liên hệ → hiển thị thông báo lỗi. |

# Use case “Chỉnh sửa thông tin cá nhân”

|  |  |
| --- | --- |
| **ID** | UC6 |
| **Name** | Chỉnh sửa thông tin cá nhân |
| **Brief description** | Chức năng cho phép quản trị viên chỉnh sửa và cập nhật thông tin cá nhân trên tài khoản của mình, như tên, địa chỉ email, số điện thoại, ảnh đại diện, hoặc mật khẩu. |
| **Actors** | Quản trị viên |
| **Pre-conditions** | Quản trị viên đã đăng nhập vào tài khoản. |
| **Basic flow** |  Quản trị viên truy cập vào mục "Thông tin cá nhân" trên web.   Quản trị viên chọn phần thông tin cần cập nhật (ví dụ: tên, email hoặc mật khẩu).   Quản trị viên nhập thông tin mới hoặc thay đổi thông tin hiện tại.   Hệ thống kiểm tra tính hợp lệ của thông tin cập nhật, như:   * Email phải hợp lệ và chưa được sử dụng. * Mật khẩu mới phải tuân thủ các yêu cầu bảo mật.    Hệ thống lưu thông tin đã cập nhật.   Quản trị viên nhận thông báo "Cập nhật thông tin thành công." |
| **Alternative flow** | Nếu thông tin nhập không hợp lệ (ví dụ: email không đúng định dạng, số điện thoại không hợp lệ):   * Hệ thống hiển thị thông báo lỗi, ví dụ: "Vui lòng nhập thông tin hợp lệ."   Nếu quản trị viên cố gắng cập nhật thông tin đã được sử dụng bởi tài khoản khác:   * Hệ thống hiển thị thông báo lỗi, ví dụ: "Email này đã được sử dụng." |
| **Result** | Thông tin cá nhân của quản trị viên được cập nhật thành công trong hệ thống. |
| **Exceptions** |  Mất kết nối mạng khi gửi yêu cầu cập nhật (hệ thống hiển thị thông báo: "Cập nhật không thành công. Vui lòng thử lại.").   Lỗi máy chủ hoặc hệ thống không thể xử lý yêu cầu cập nhật. |

# Use case “Bình luận bài viết”

|  |  |
| --- | --- |
| **ID** | UC7 |
| **Name** | Bình luận bài viết |
| **Brief description** | |  | | --- | |  |  |  | | --- | | Chức năng cho phép người dùng đã đăng nhập viết bình luận về nội dung của một bài viết cụ thể trên hệ thống. | |
| **Actors** | Người dùng(đã đăng nhập) |
| **Pre-conditions** | Người dùng đã đăng nhập và đang xem chi tiết một bài viết. |
| **Basic flow** | |  |  | | --- | --- | | • Người dùng truy cập vào trang xem chi tiết bài viết. |  |  |  |  | | --- | --- | | • Ở cuối bài viết, hệ thống hiển thị form bình luận gồm trường nhập nội dung. |  |  |  |  | | --- | --- | | • Người dùng nhập nội dung bình luận. |  |  |  |  | | --- | --- | | • Nhấn nút “Gửi bình luận”. |  |  |  |  | | --- | --- | | • Hệ thống kiểm tra nội dung không được để trống. |  |  |  |  | | --- | --- | | • Nếu hợp lệ, hệ thống lưu bình luận và hiển thị trong danh sách bình luận của bài viết, kèm thông tin người đăng và thời gian. |  | |
| **Alternative flow** | • Người dùng chưa đăng nhập → khi nhấn “Gửi bình luận”, hệ thống hiển thị thông báo yêu cầu đăng nhập.  • Người dùng có thể chỉnh sửa hoặc xóa bình luận của chính mình (nếu chức năng này được hỗ trợ). |
| **Result** | Bình luận được hiển thị thành công dưới bài viết, gắn với người dùng đã đăng nhập. |
| **Exceptions** | • Nội dung bình luận để trống hoặc chứa từ ngữ không hợp lệ → hệ thống hiển thị lỗi.  • Lỗi hệ thống hoặc kết nối mạng khiến không gửi được bình luận. |

# Use case “Thêm bài viết/danh mục bài viết/tài khoản người dùng”

|  |  |
| --- | --- |
| **ID** | UC8 |
| **Name** | Thêm bài viết/danh mục bài viết/tài khoản người dùng |
| **Brief description** | |  | | --- | |  |  |  | | --- | | Chức năng cho phép quản trị viên thêm bài viết/danh mục bài viết/tài khoản người dùng mới lên hệ thống | |
| **Actors** | Quản trị viên |
| **Pre-conditions** | Quản trị viên đã đăng nhập. |
| **Basic flow** | |  |  | | --- | --- | | • Quản trị viên truy cập trang “Thêm bài viết/danh mục bài viết/tài khoản người dùng”. |  |  |  |  | | --- | --- | | • Hệ thống hiển thị form thêm bài viết/danh mục bài viết/tài khoản người dùng gồm nhiều trường |  |  |  |  | | --- | --- | | • Quản trị viên nhập đầy đủ thông tin và nhấn nút “Thêm bài viết/danh mục bài viết/tài khoản người dùng”. |  |  |  |  | | --- | --- | | • Hệ thống kiểm tra dữ liệu đầu vào không được trống. |  |  |  |  | | --- | --- | | • Nếu hợp lệ, hệ thống lưu bài viết/danh mục bài viết/tài khoản người dùng vào cơ sở dữ liệu và hiển thị bài viết/danh mục bài viết/tài khoản người dùng trong danh sách bài viết/danh mục bài viết/tài khoản người dùng. |  |  |  | | --- | | • Hệ thống hiển thị thông báo thêm thành công. | |
| **Alternative flow** | • Quản trị viên nhấn “Hủy” → hệ thống quay về trang trước mà không lưu dữ liệu. |
| **Result** | Bài viết/danh mục bài viết/tài khoản người dùng được thêm thành công và hiển thị trên hệ thống. |
| **Exceptions** | • Dữ liệu không hợp lệ (ví dụ: thiếu tiêu đề,...) → hệ thống hiển thị lỗi và yêu cầu nhập lại.  • Lỗi kết nối hoặc hệ thống khiến không thể lưu bài viết. |

# Use case “Chỉnh sửa bài viết/danh mục bài viết/tài khoản người dùng”

|  |  |
| --- | --- |
| **ID** | UC9 |
| **Name** | |  | | --- | |  |  |  | | --- | | Chỉnh sửa bài viết/danh mục bài viết/tài khoản người dùng | |
| **Brief description** | |  | | --- | |  |  |  |  |  | | --- | --- | --- | | |  | | --- | |  |  |  | | --- | | Chức năng cho phép quản trị chỉnh sửa thông tin của bài viết/danh mục bài viết/tài khoản người dùng đã đăng. | | |
| **Actors** | Quản trị viên |
| **Pre-conditions** | Quản trị viên đã đăng nhập |
| **Basic flow** | |  |  | | --- | --- | | • Quản trị viên truy cập danh sách bài viết/danh mục bài viết/tài khoản người dùng của mình hoặc từ giao diện quản trị. |  |  |  |  | | --- | --- | | • Quản trị viên chọn một bài viết/danh mục bài viết/tài khoản người dùng và nhấn nút “Chỉnh sửa”. |  |  |  |  | | --- | --- | | • Hệ thống hiển thị form chỉnh sửa với dữ liệu hiện tại của bài viết/danh mục bài viết/tài khoản người dùng. |  |  |  | | --- | | • Quản trị viên cập nhật lại thông tin |  |  |  | | --- | --- | | • Nhấn nút “Cập nhật”. |  |  |  |  | | --- | --- | | • Hệ thống kiểm tra dữ liệu mới (không để trống thông tin) |  |  |  | | --- | | • Nếu hợp lệ, hệ thống cập nhật dữ liệu trong cơ sở dữ liệu và hiển thị thông báo thành công. | |
| **Alternative flow** | • Quản trị viên nhấn “Hủy” → hệ thống quay lại trang trước, không lưu thay đổi. • Nếu hệ thống áp dụng kiểm duyệt → bài viết sau khi sửa được chuyển về trạng thái “chờ duyệt lại”. |
| **Result** | Bài viết/danh mục bài viết/tài khoản người dùng được cập nhật thành công với nội dung mới. |
| **Exceptions** | • Dữ liệu chỉnh sửa không hợp lệ → hệ thống hiển thị thông báo lỗi và yêu cầu sửa lại.  • Lỗi hệ thống khiến bài viết không được cập nhật. |

# Use case “Xoá bài viết/danh mục bài viết/tài khoản người dùng”

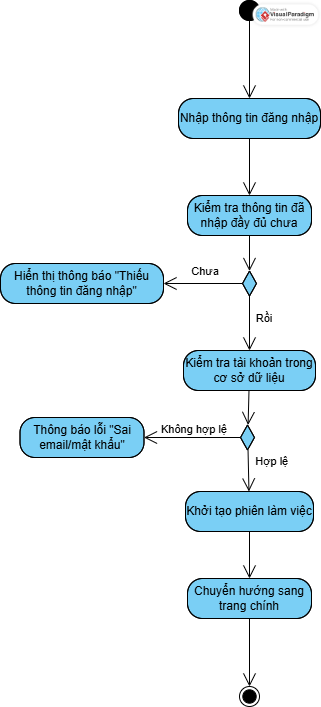
|  |  |
| --- | --- |
| **ID** | UC10 |
| **Name** | |  | | --- | |  |  |  |  |  | | --- | --- | --- | | |  | | --- | |  |  |  | | --- | | Xóa bài viết/danh mục bài viết/tài khoản người dùng | | |
| **Brief description** | |  | | --- | |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | | |  | | --- | |  |  |  |  |  | | --- | --- | --- | | |  | | --- | |  |  |  | | --- | | Chức năng cho phép quản trị viên xóa bài viết/danh mục bài viết/tài khoản người dùng khỏi hệ thống. | | | |
| **Actors** | Quản trị viên |
| **Pre-conditions** | Quản trị viên đã đăng nhập |
| **Basic flow** | |  |  | | --- | --- | | • Quản trị viên truy cập danh sách bài viết/danh mục bài viết/tài khoản người dùng |  |  |  |  | | --- | --- | | • Chọn bài viết/danh mục bài viết/tài khoản người dùng muốn xóa và nhấn nút “Xóa”. |  |  |  |  | | --- | --- | |  |  |  |  |  | | --- | --- | | • Quản trị viên xác nhận xóa. |  |  |  |  | | --- | --- | | • Hệ thống thực hiện xóa bài viết/danh mục bài viết/tài khoản người dùng khỏi cơ sở dữ liệu hoặc đánh dấu là “đã xóa”. |  |  |  | | --- | | • Hệ thống hiển thị thông báo xóa thành công và cập nhật lại danh sách bài viết/danh mục bài viết/tài khoản người dùng. | |
| **Alternative flow** |  |
| **Result** | Bài viết/danh mục bài viết/tài khoản người dùng bị xóa khỏi hệ thống. |
| **Exceptions** |  |

# Use case “Xem danh sách liên hệ”

|  |  |
| --- | --- |
| **ID** | UC11 |
| **Name** | |  | | --- | |  |  |  | | --- | | Xem danh sách liên hệ | |
| **Brief description** | |  | | --- | |  |  |  |  |  | | --- | --- | --- | | |  | | --- | |  |  |  | | --- | | Cho phép quản trị viên xem toàn bộ danh sách liên hệ được gửi từ người dùng thông qua biểu mẫu liên hệ (Contact Form). Danh sách được hiển thị theo bảng, có thể bao gồm tên, email, nội dung liên hệ, và thời gian gửi. | | |
| **Actors** | Quản trị viên |
| **Pre-conditions** | Quản trị viên đã đăng nhập |
| **Basic flow** | |  |  | | --- | --- | | • Quản trị viên chọn chức năng “Xem liên hệ” từ menu quản trị. |  |  |  |  | | --- | --- | | • Hệ thống truy vấn cơ sở dữ liệu để lấy danh sách liên hệ. |  |  |  |  | | --- | --- | |  |  |  |  |  | | --- | --- | | • Hệ thống hiển thị danh sách liên hệ dưới dạng bảng. |  | |
| **Alternative flow** |  |
| **Result** | Danh sách liên hệ được hiển thị ra màn hình. |
| **Exceptions** |  |

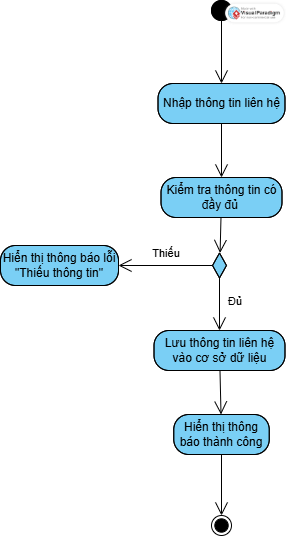
# Biểu đồ hoạt động

# Activity diagram “Đăng nhập”



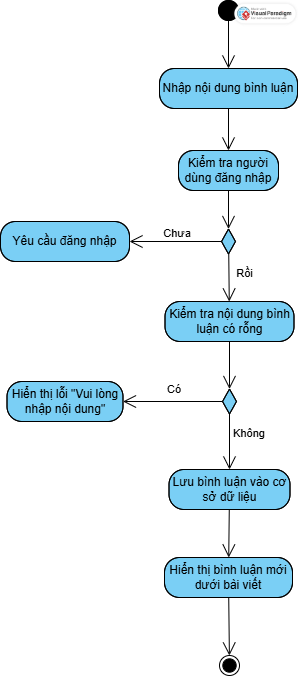
Hình 2-1: Activity Diagram Đăng nhập

# Activity diagram “Gửi liên hệ”



Hình 2-2: Activity Diagram Gửi liên hệ

# Activity diagram “Bình luận bài viết”



Hình 2-3: Activity Diagram Bình luận bài viết

# Activity diagram “Quản trị”

# C:\Users\Admin\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Weblog Activity -Quản lý (3).png

Hình 2-4: Activity Diagram Quản trị

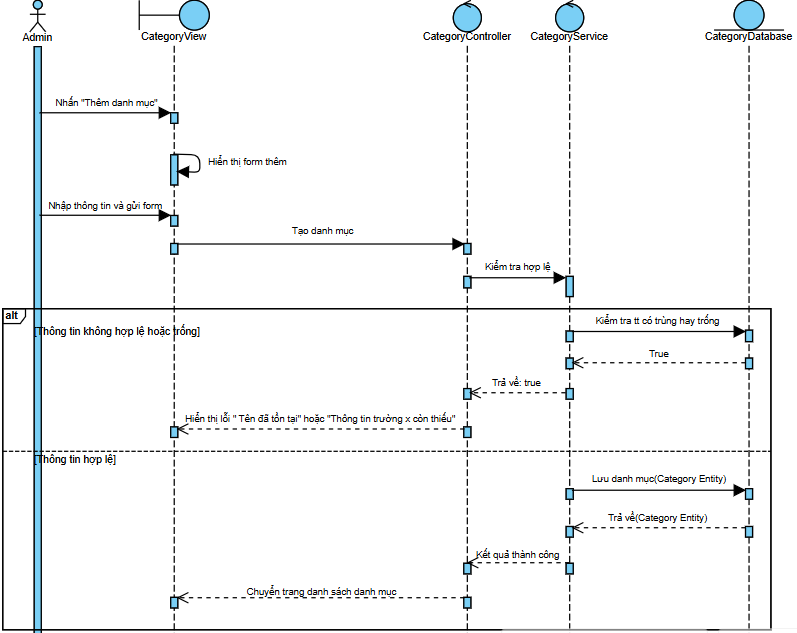
# Chương 3: Thiết kế hệ thống

# Lựa chọn mô hình kiến trúc

Trong dự án này, mô hình kiến trúc **MVC (Model - View - Controller)** được chúng em lựa chọn vì tính hiệu quả và khả năng tổ chức mã nguồn rõ ràng. MVC phân chia ứng dụng thành ba phần riêng biệt: **Model** xử lý dữ liệu và logic nghiệp vụ, **View** chịu trách nhiệm hiển thị giao diện cho người dùng, và **Controller** đóng vai trò trung gian điều phối, tiếp nhận yêu cầu từ người dùng, xử lý và trả về kết quả. Việc áp dụng MVC giúp mã nguồn dễ bảo trì, dễ mở rộng và hỗ trợ phát triển song song giữa giao diện và xử lý nghiệp vụ. Đồng thời, mô hình này còn tăng cường khả năng kiểm thử và bảo mật, phù hợp với yêu cầu của các ứng dụng web hiện đại. Vì vậy, MVC là lựa chọn hợp lý và tối ưu cho việc phát triển dự án này.

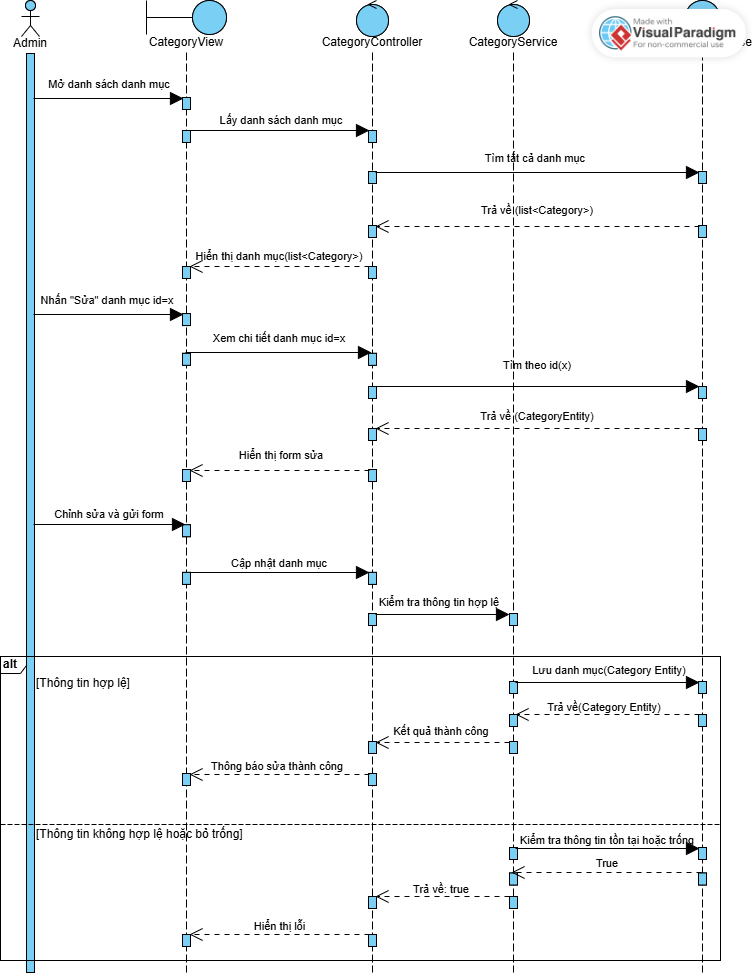
# Biểu đồ tuần tự

# Sequence diagram “Thêm danh mục”



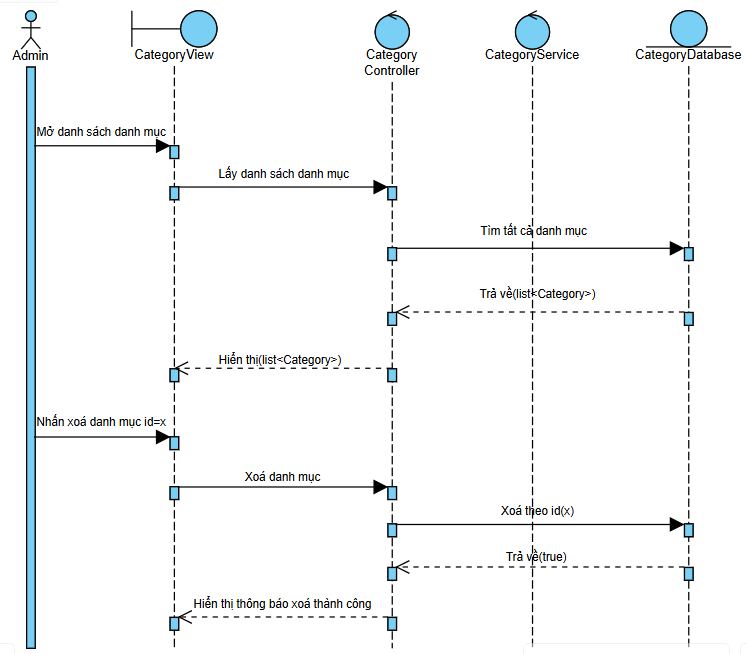
Hình 3-1: Sequence Diagram Thêm danh mục

# Sequence diagram “Chỉnh sửa danh mục”



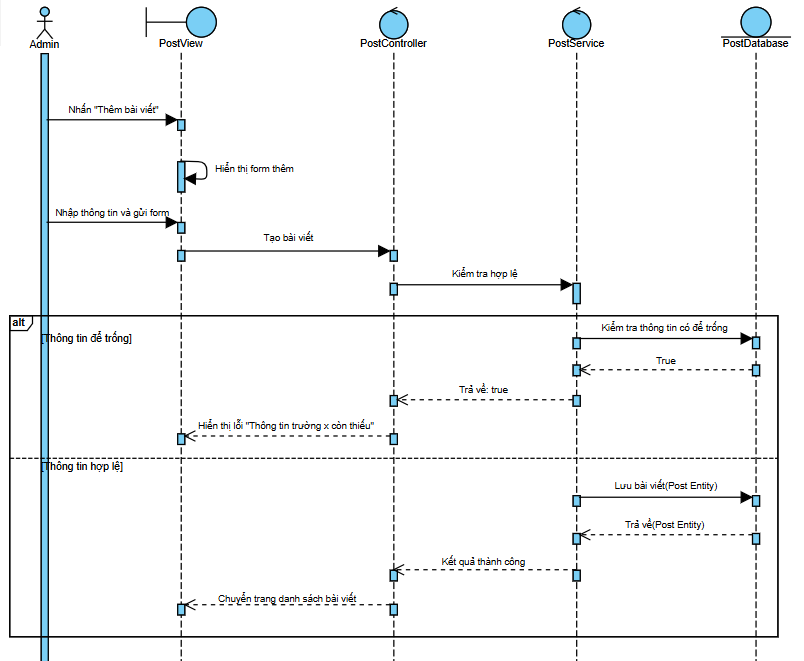
Hình 3-2: Sequence Diagram Chỉnh sửa danh mục

# Sequence diagram “Xoá danh mục”



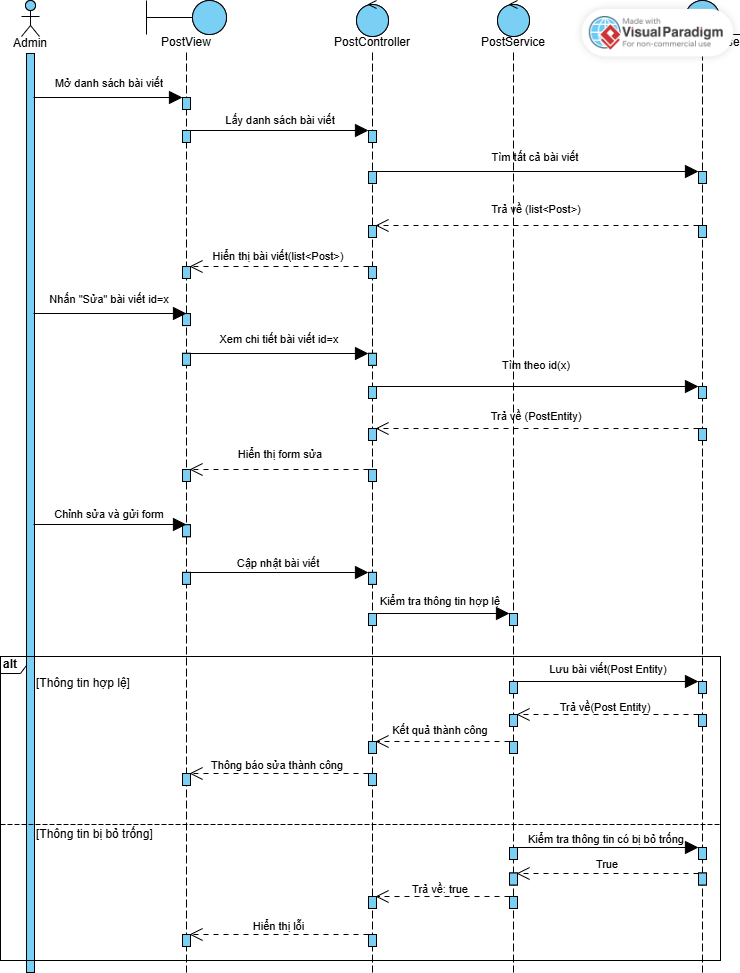
Hình 3-3: Sequence Diagram Xoá danh mục

# Sequence diagram “Đăng bài viết mới”



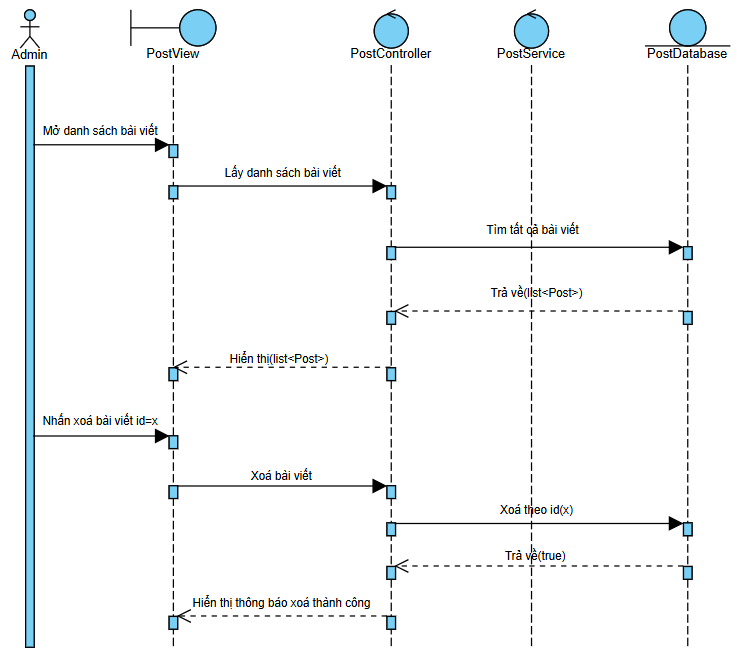
Hình 3-4: Sequence Diagram Đăng bài viết mới

# Sequence diagram “Chỉnh sửa bài viết”



Hình 3-5: Sequence Diagram Chỉnh sửa bài viết

# Sequence diagram “Xoá bài viết”



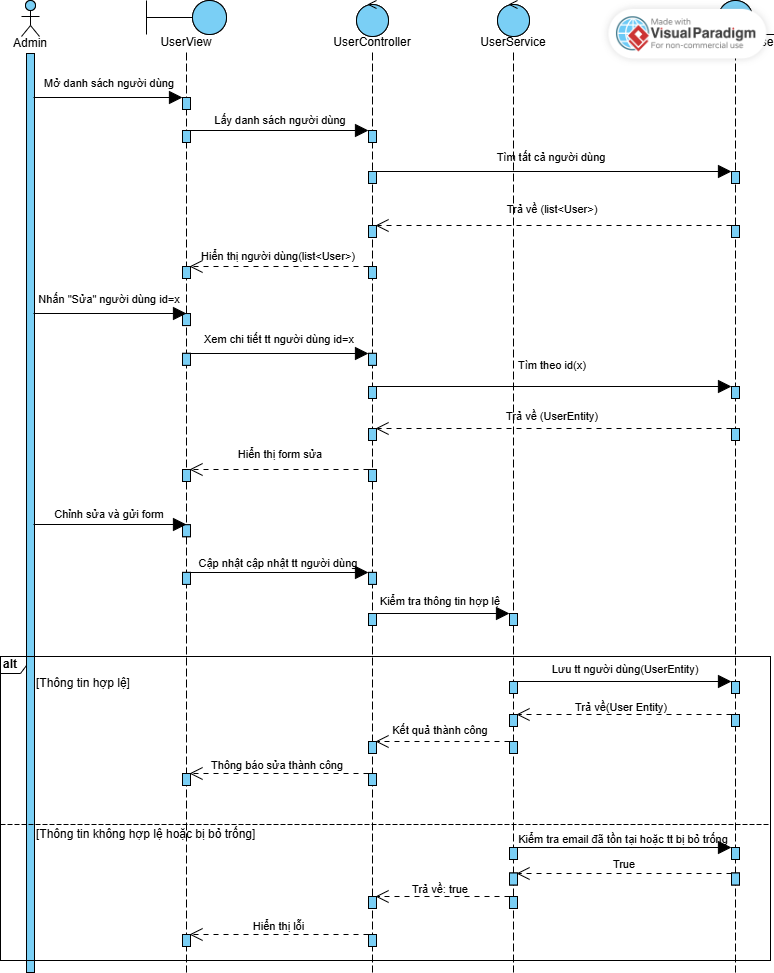
Hình 3-6: Sequence Diagram Xoá bài viết

# Sequence diagram “Mở tài khoản”

# 

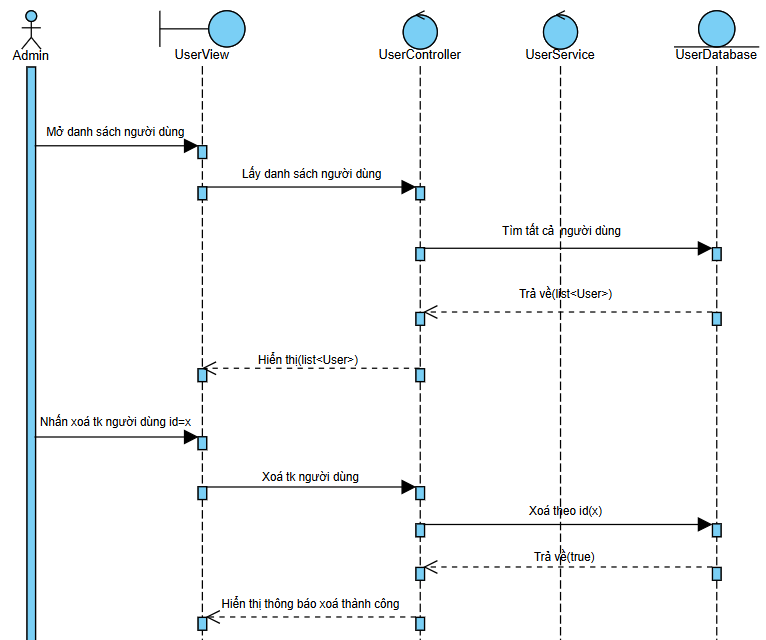
Hình 3-7: Sequence Diagram Mở tài khoản

# Sequence diagram “Chỉnh sửa thông tin tài khoản”



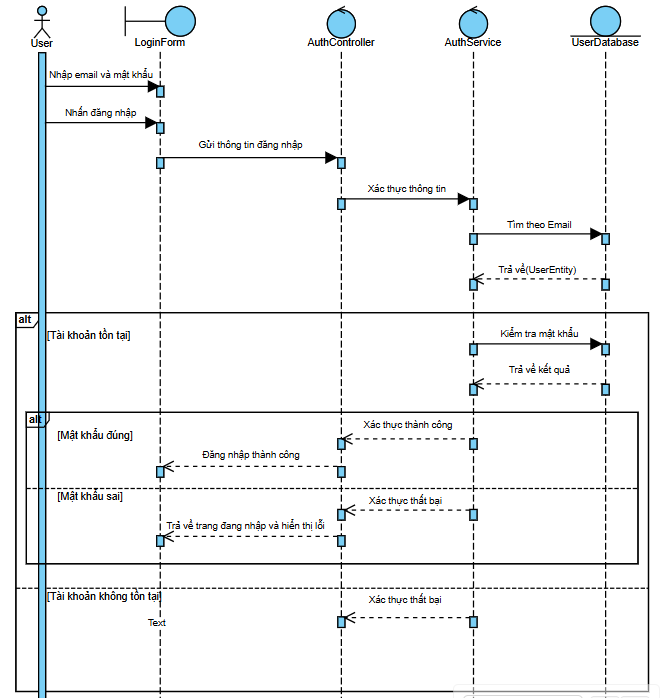
Hình 3-8: Sequence Diagram Chỉnh sửa thông tin tài khoản

# Sequence diagram “Xoá tài khoản người dùng”



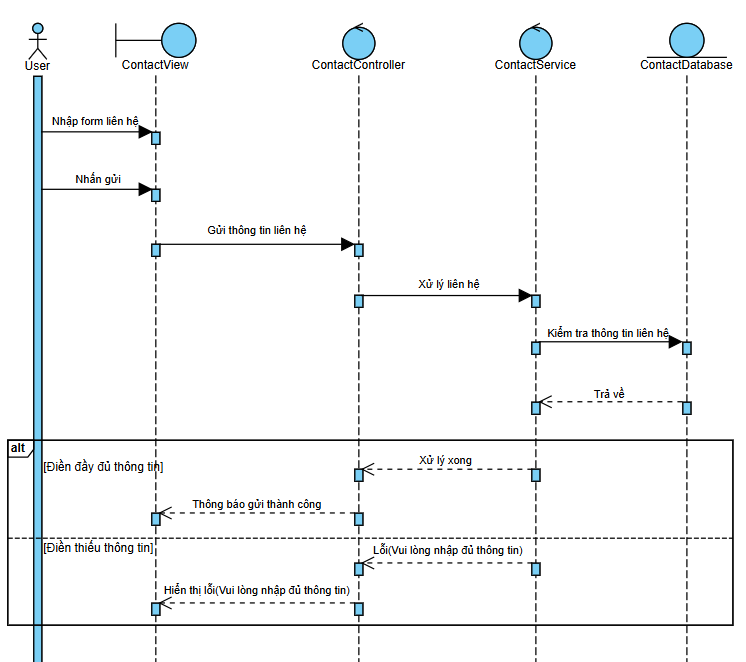
Hình 3-9: Sequence Diagram Xoá tài khoản người dùng

# Sequence diagram “Đăng nhập”



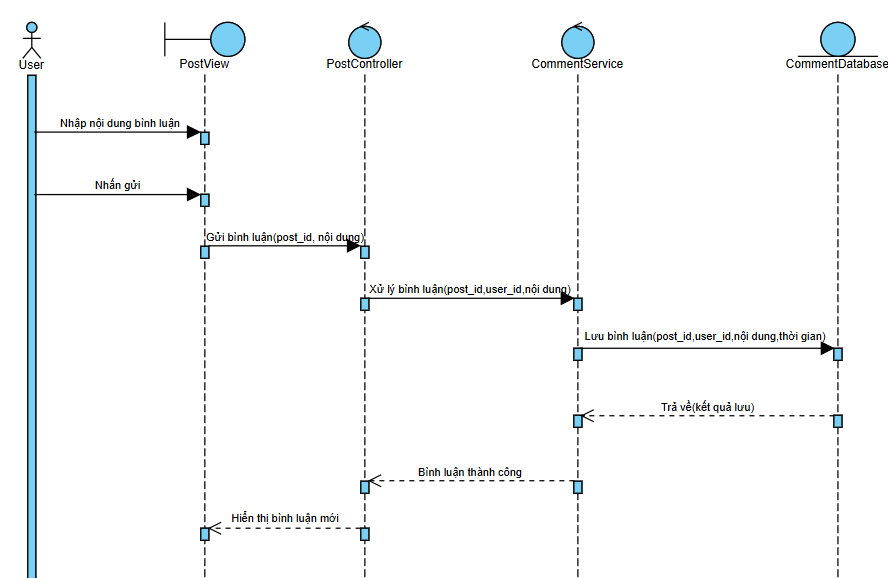
Hình 3-10: Sequence Diagram Đăng nhập

# Sequence diagram “Gửi liên hệ”



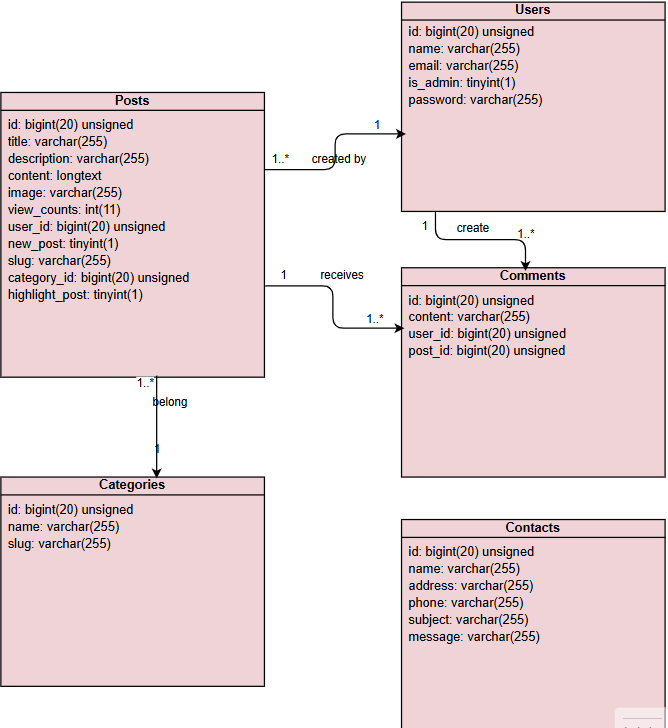
Hình 3-11: Sequence Diagram Gửi liên hệ

# Sequence diagram “Bình luận bài viết”



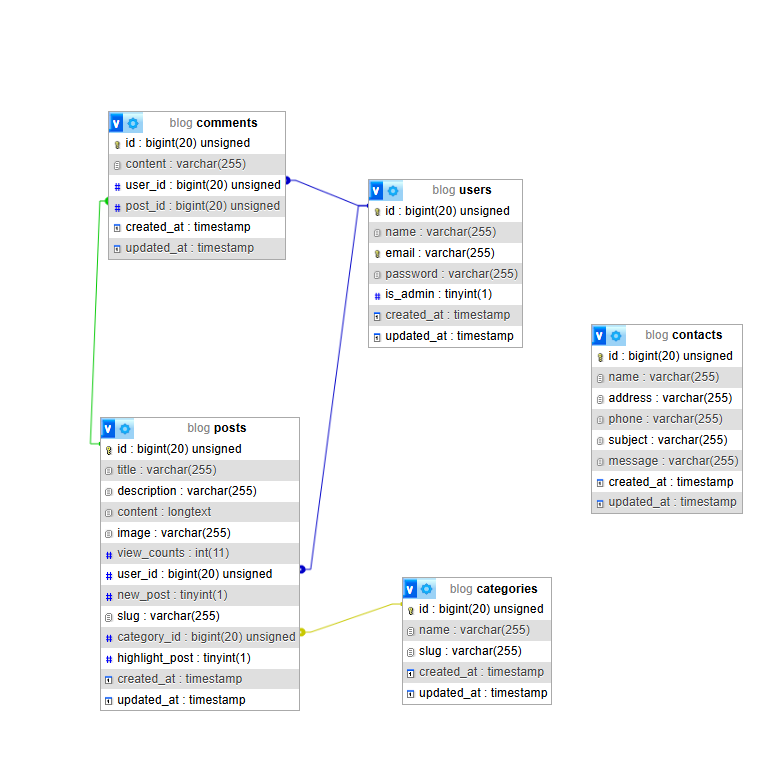
Hình 3-12: Sequence Diagram Bình luận bài viết

# Thiết kế lớp



Hình 4: Class Diagram

# Thiết kế cơ sở dữ liệu



Hình 5: Thiết kế CSDL

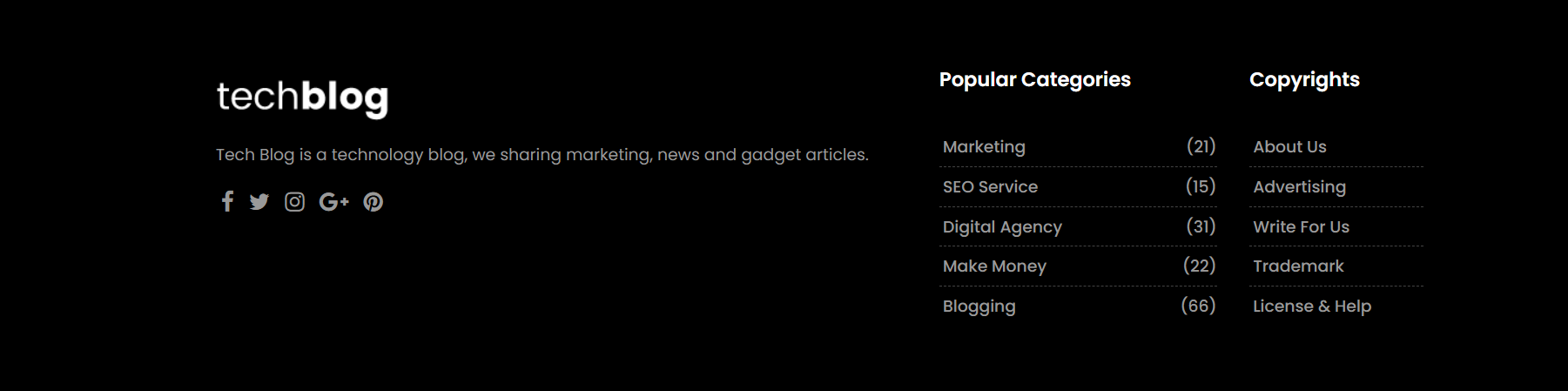
# Thiết kế giao diện người dùng

# Giao diện phần đầu trang



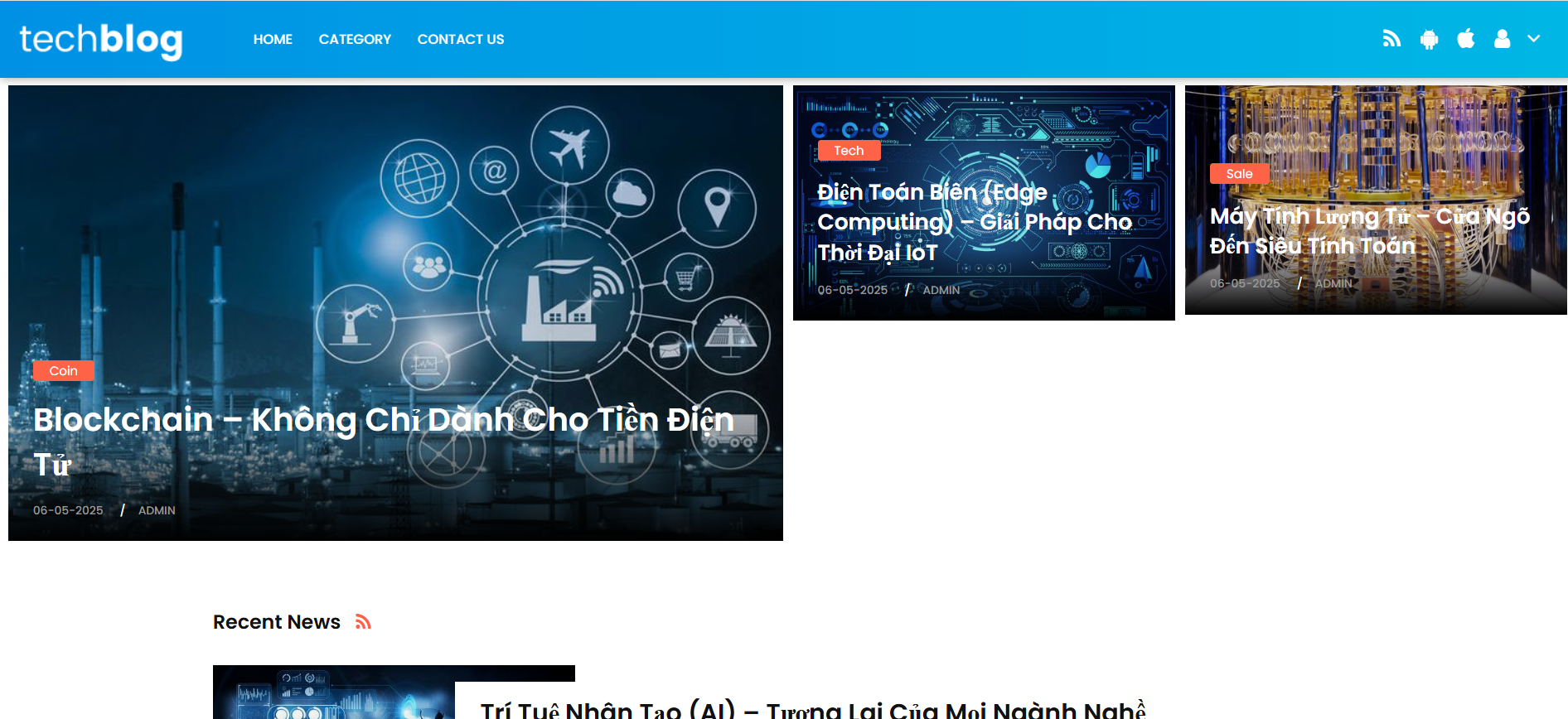
Hình 6.1: Giao diện phần đầu trang

# Giao diện phần chân trang



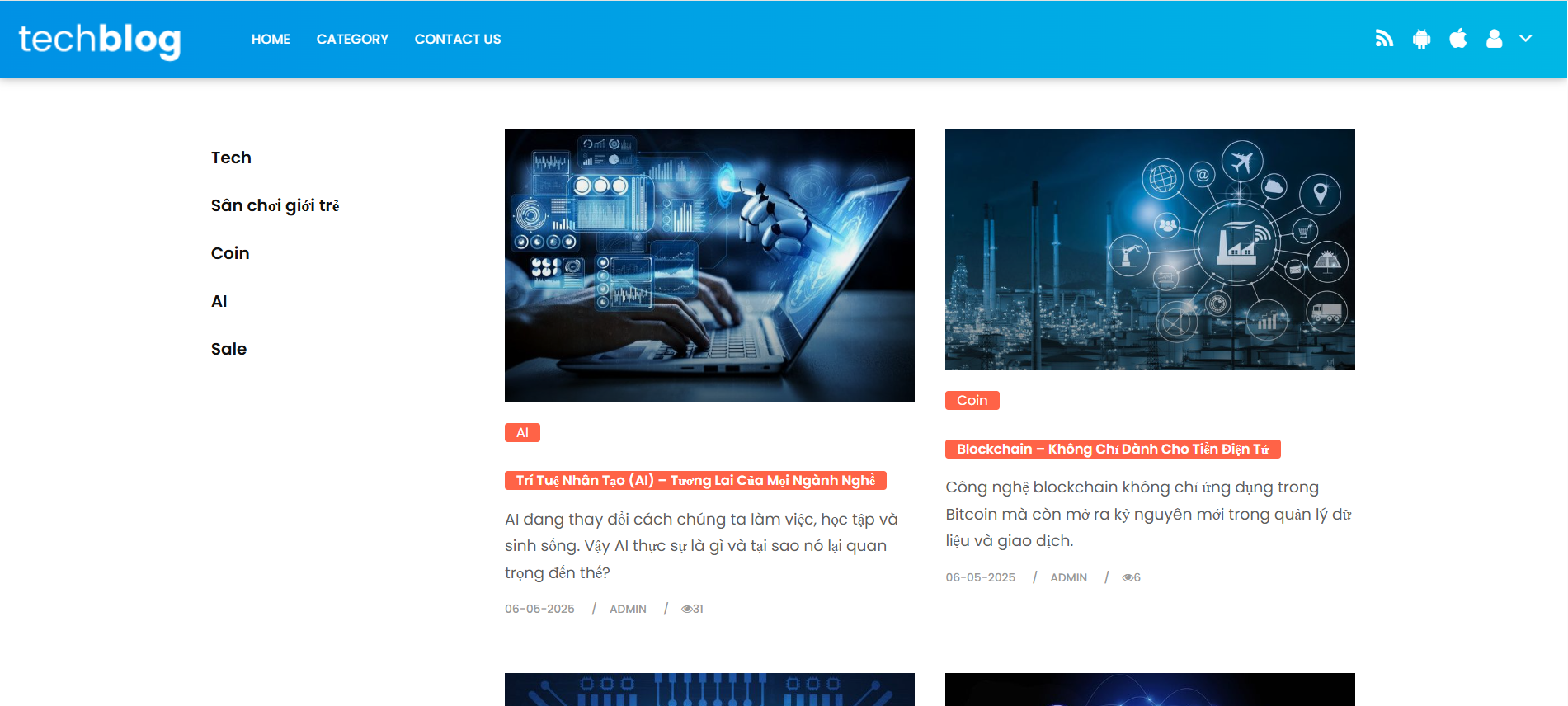
Hình 6.2: Giao diện phần chân trang

# Giao diện trang chủ



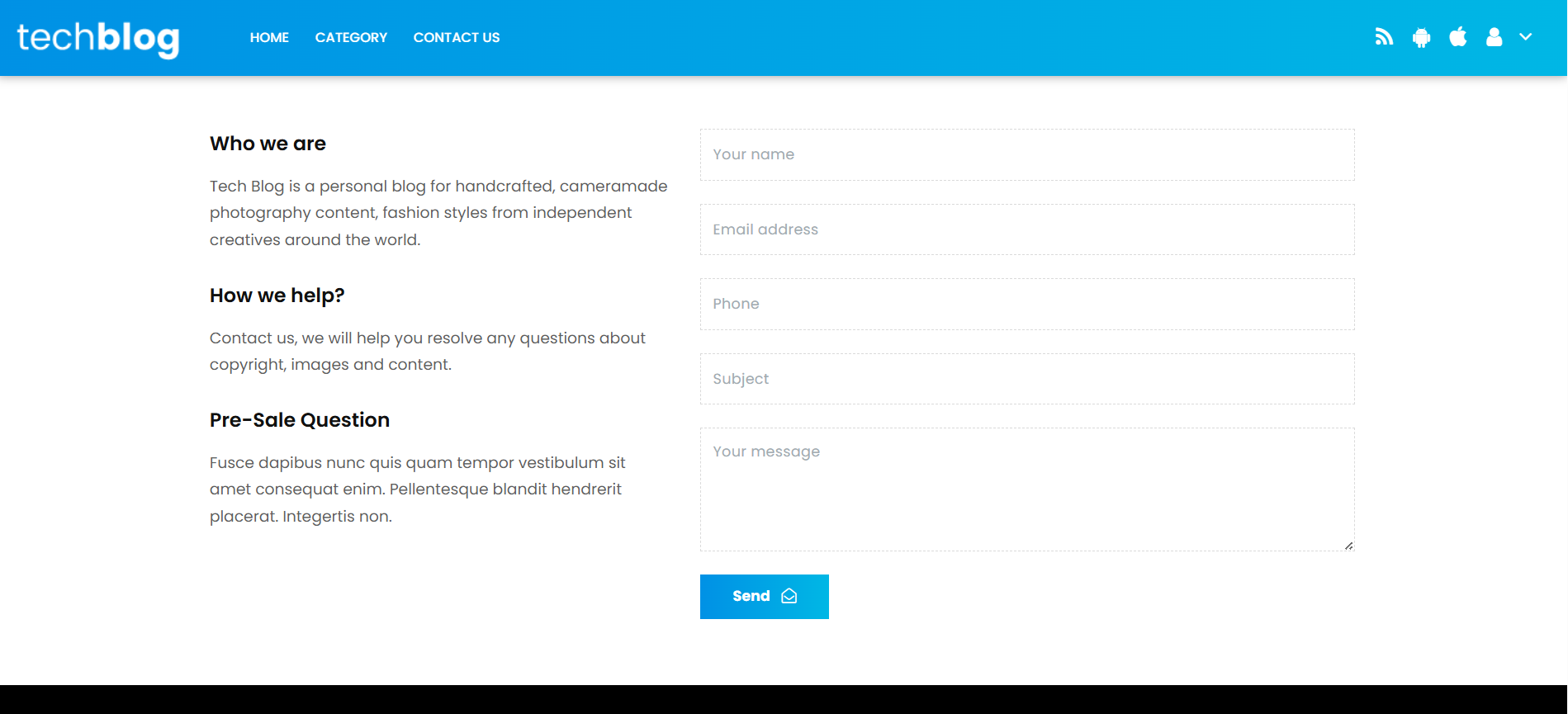
Hình 6.3: Giao diện phần trang chủ

# Giao diện danh sách bài viết chia theo danh mục



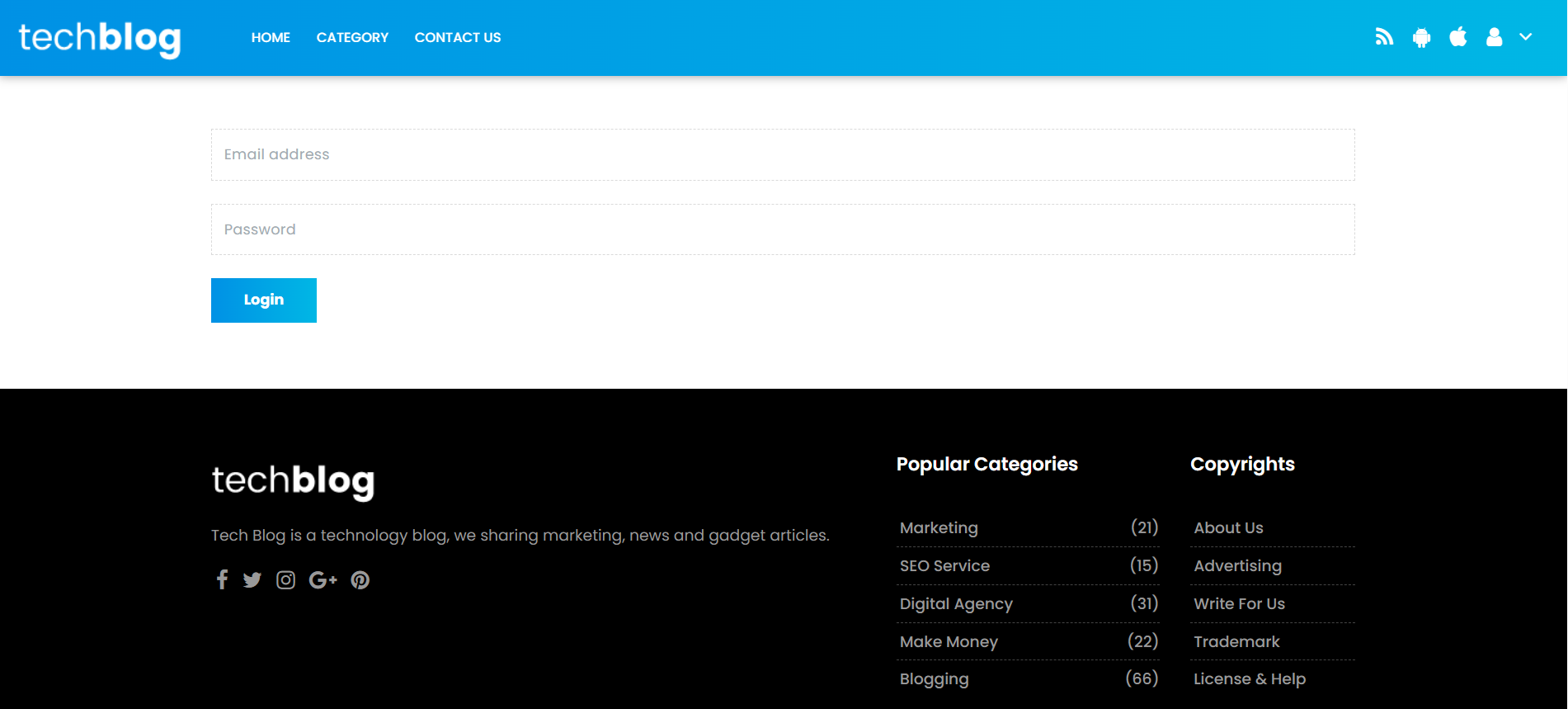
Hình 6.4: Giao diện danh sách bài viết chia theo danh mục

# Giao diện trang gửi liên hệ



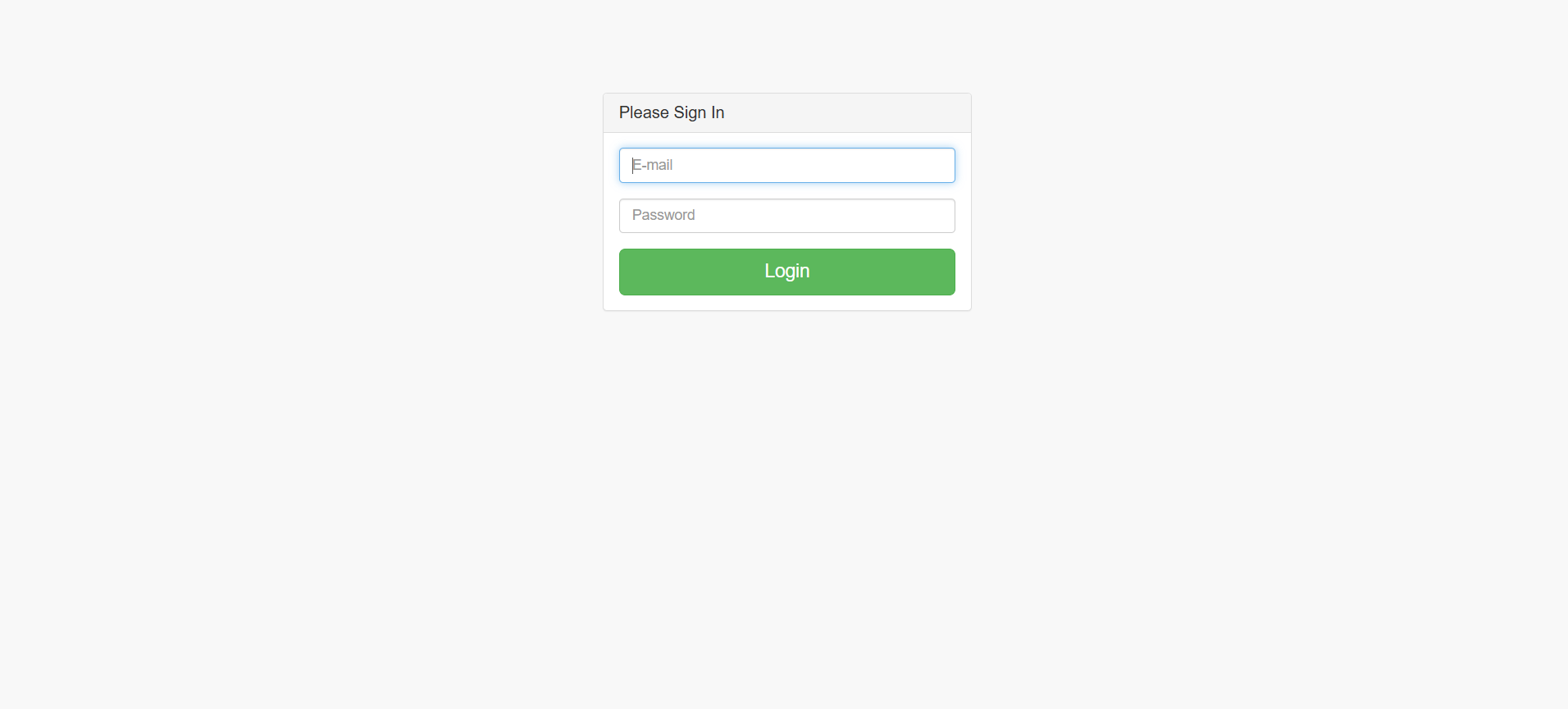
Hình 6.5: Giao diện trang gửi liên hệ

# Giao diện trang đăng nhập(User)



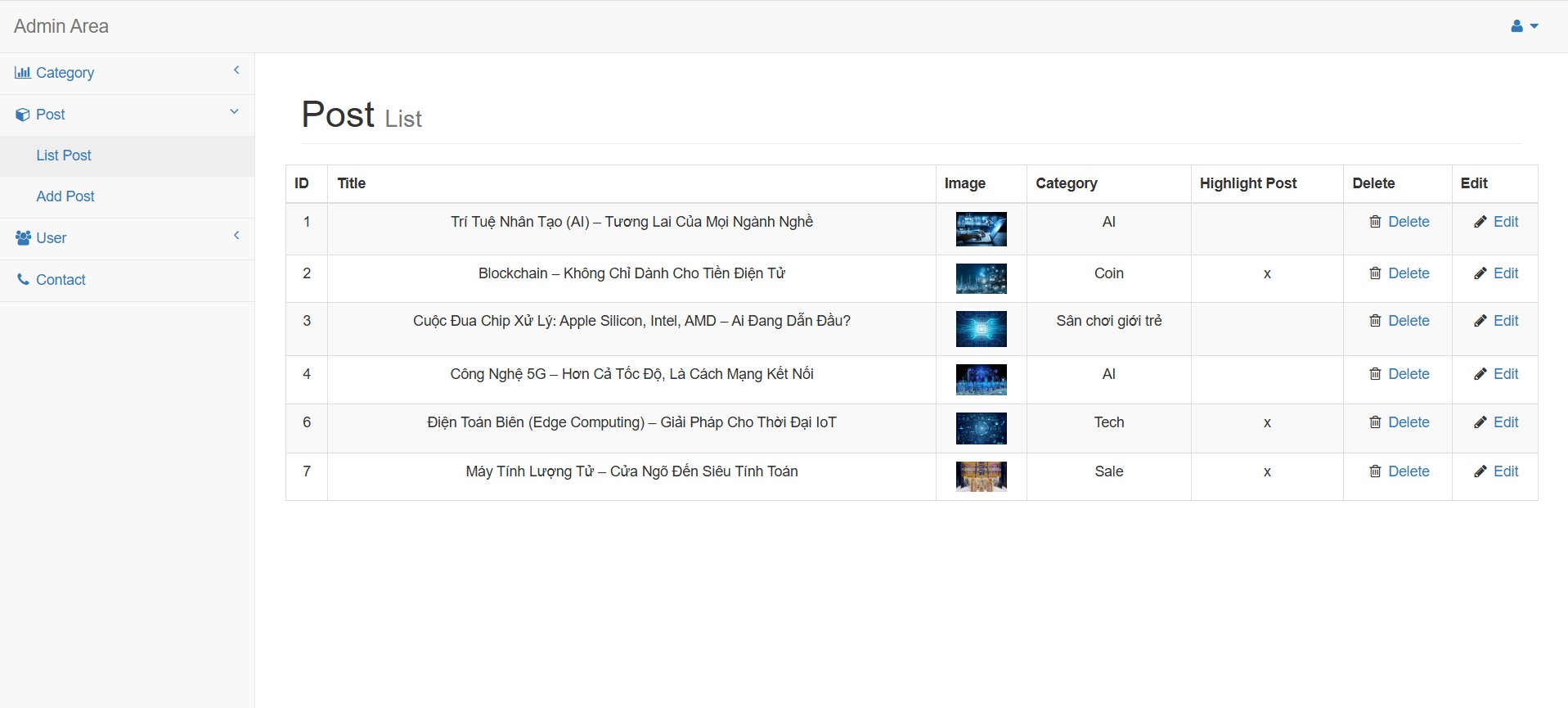
Hình 6.6: Giao diện trang đăng nhập(User)

# Giao diện trang đăng nhập(Admin)



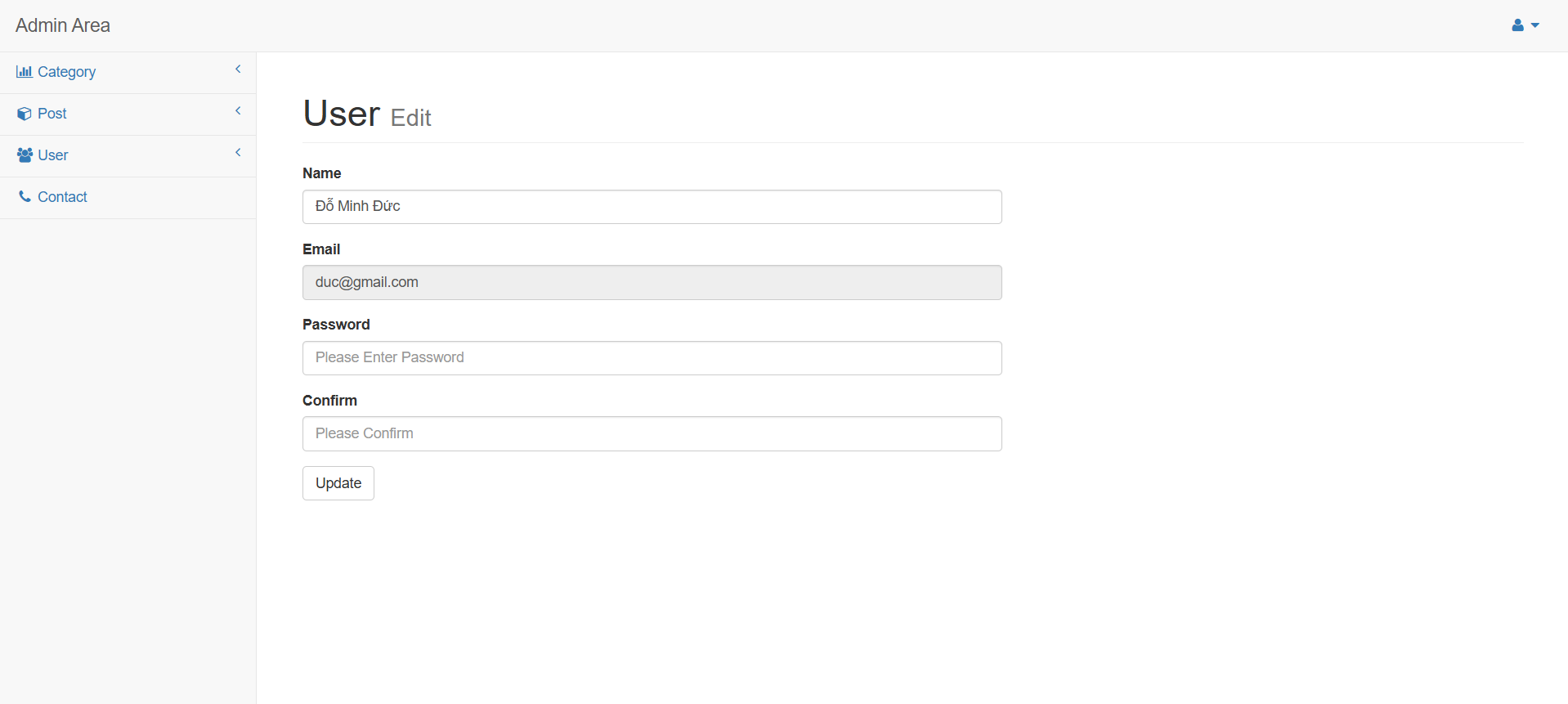
Hình 6.7: Giao diện trang đăng nhập(Admin)

# Giao diện trang quản trị



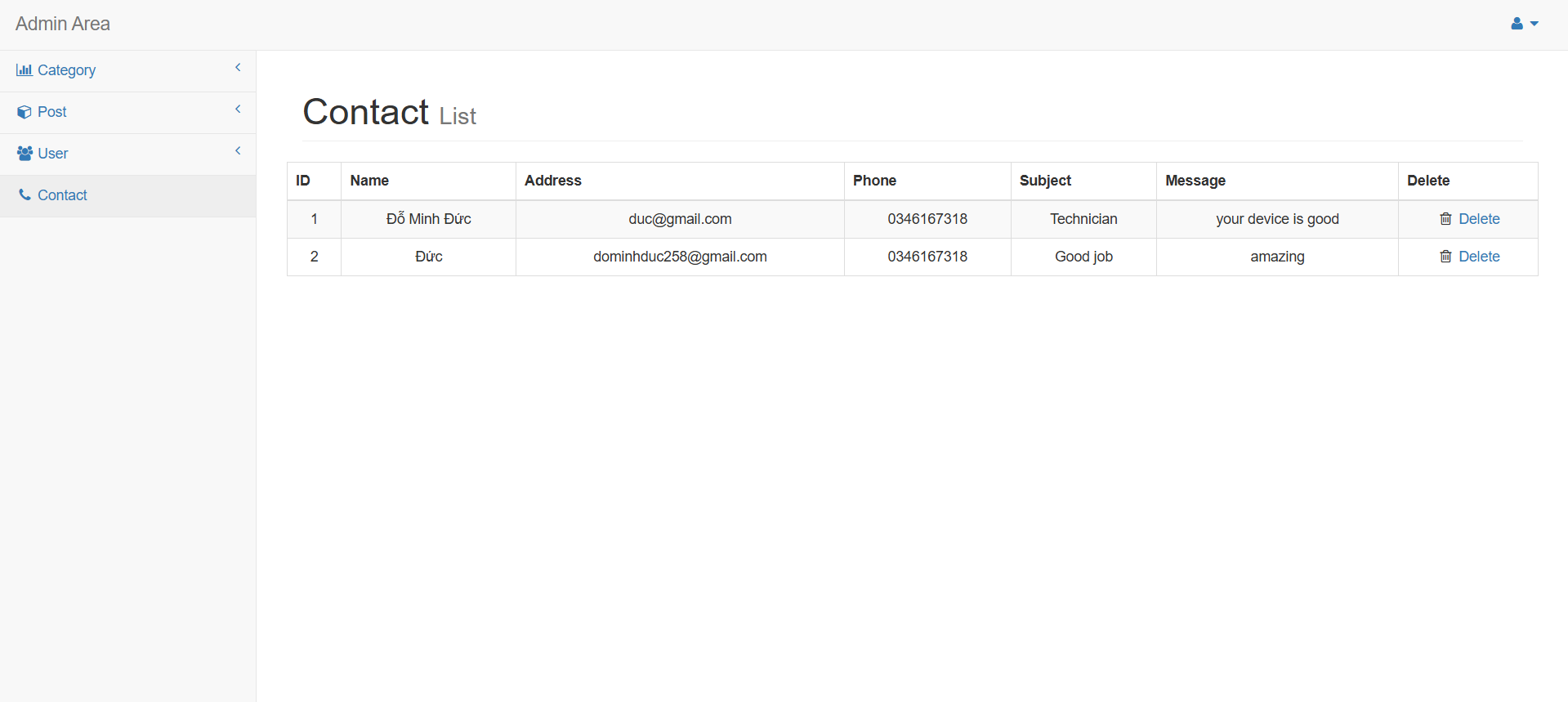
Hình 6.8: Giao diện trang quản trị

# Giao diện trang chỉnh sửa thông tin cá nhân



Hình 6.9: Giao diện trang chỉnh sửa thông tin cá nhân

# Giao diện trang danh sách liên hệ



Hình 6.10: Giao diện trang danh sách liên hệ

# Chương 4: Cài đặt và triển khai hệ thống

# Môi trường triển khai

Để hệ thống Weblog hoạt động ổn định và hiệu quả, cần thiết lập một môi trường bao gồm đầy đủ các thành phần từ máy chủ web, cơ sở dữ liệu cho đến mã nguồn ứng dụng. Hệ thống được xây dựng dựa trên mô hình kiến trúc MVC, giúp tách biệt rõ ràng giữa giao diện, xử lý logic và dữ liệu, từ đó nâng cao khả năng bảo trì và mở rộng. Về phía môi trường triển khai, dự án được cài đặt và chạy thử nghiệm trên nền tảng XAMPP với các thành phần chính gồm Apache (web server), MySQL (hệ quản trị cơ sở dữ liệu) và PHP. Cơ sở dữ liệu được khởi tạo với các bảng như users, posts, categories, comments và contacts, phục vụ cho các chức năng như đăng nhập, quản lý bài viết, phân loại, bình luận và liên hệ. Mã nguồn của hệ thống được đặt trong thư mục htdocs, kết nối đến cơ sở dữ liệu thông qua file cấu hình, và có thể truy cập qua trình duyệt bằng địa chỉ http://localhost/weblog. Sau khi triển khai, hệ thống được kiểm thử toàn diện bao gồm các chức năng người dùng và quản trị, đảm bảo tính ổn định và bảo mật thông qua kiểm soát đầu vào, phân quyền truy cập và mã hóa dữ liệu đăng nhập.

# Cài đặt hệ thống

Để cài đặt hệ thống Weblog, trước hết cần thiết lập môi trường phát triển bằng cách cài đặt phần mềm XAMPP, bao gồm Apache, MySQL và PHP. Sau khi khởi động Apache và MySQL thông qua bảng điều khiển XAMPP, người dùng truy cập vào phpMyAdmin tại địa chỉ http://localhost/phpmyadmin để tạo cơ sở dữ liệu mới, ví dụ tên là weblog\_db, và import dữ liệu mẫu nếu có. Mã nguồn hệ thống được sao chép vào thư mục htdocs của XAMPP, sau đó cấu hình file kết nối cơ sở dữ liệu (như config.php hoặc .env) để đảm bảo kết nối đúng với thông tin máy chủ. Tiếp theo, người dùng mở trình duyệt và truy cập http://localhost/weblog để kiểm tra hoạt động của hệ thống. Hệ thống cần được kiểm thử các chức năng chính như đăng nhập, quản lý bài viết, gửi liên hệ, bình luận,... nhằm đảm bảo hoạt động ổn định, dữ liệu được lưu trữ chính xác và giao diện thân thiện với người dùng.cài đặt phần mềm XAMPP, bao gồm Apache, MySQL và PHP. Sau khi khởi động Apache và MySQL thông qua bảng điều khiển XAMPP, người dùng truy cập vào phpMyAdmin tại địa chỉ http://localhost/phpmyadmin để tạo cơ sở dữ liệu mới, ví dụ tên là weblog\_db, và import dữ liệu mẫu nếu có. Mã nguồn hệ thống được sao chép vào thư mục htdocs của XAMPP, sau đó cấu hình file kết nối cơ sở dữ liệu (như config.php hoặc .env) để đảm bảo kết nối đúng với thông tin máy chủ. Tiếp theo, người dùng mở trình duyệt và truy cập http://localhost/weblog để kiểm tra hoạt động của hệ thống. Hệ thống cần được kiểm thử các chức năng chính như đăng nhập, quản lý bài viết, gửi liên hệ, bình luận,... nhằm đảm bảo hoạt động ổn định, dữ liệu được lưu trữ chính xác và giao diện thân thiện với người dùng.

# Chương 5: Kết quả và đánh giá

# Kết quả thử nghiệm hệ thống

Sau quá trình xây dựng và hoàn thiện hệ thống website blog cá nhân, nhóm đã tiến hành thử nghiệm các chức năng chính để đánh giá mức độ hoàn thiện và khả năng vận hành của hệ thống. Cụ thể:

* Đăng nhập: Người dùng có thể đăng nhập và sử dụng hệ thống một cách dễ dàng. Hệ thống xử lý xác thực đơn giản nhưng an toàn.
* Tạo và quản lý bài viết: Admin có thể viết bài mới, chỉnh sửa hoặc xóa bài viết đã đăng. Giao diện viết bài thân thiện, hỗ trợ định dạng văn bản cơ bản.
* Xem bài viết và bình luận: User có thể xem danh sách bài viết, đọc nội dung chi tiết và để lại bình luận. Các thao tác này diễn ra mượt mà, ít gặp lỗi.
* Tìm kiếm: Hệ thống hỗ trợ tìm kiếm Category theo đề trên giao diện Admin, giúp nâng cao trải nghiệm người dùng.

# Đánh giá hiệu quả hệ thống

Thông qua quá trình thử nghiệm và sử dụng thử, nhóm nhận thấy hệ thống đạt được những điểm tích cực sau:

# Ưu điểm

* Giao diện trực quan, dễ sử dụng: Thiết kế đơn giản, phù hợp với người dùng phổ thông, không cần kiến thức kỹ thuật.
* Khả năng mở rộng tốt: Với kiến trúc đã thiết kế, hệ thống có thể phát triển thêm các chức năng nâng cao trong tương lai.

# Hạn chế

* Chưa cho phép tạo tài khoản, đổi mật khẩu.
* Chưa tích hợp các tính năng mạng xã hội: Chức năng chia sẻ, theo dõi hoặc yêu thích bài viết vẫn chưa được triển khai.