**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP LỚN**

**HỌC PHẦN: CÔNG NGHỆ JAVA**

**CHỦ ĐỀ 7: BÀI TOÁN KHÁC**

**MÃ ĐỀ 154: XÂY DỰNG ỨNG DỤNG WEB ĐỂ TRA CỨU TIN TỨC SỬ DỤNG JAVA SERVLET**

|  |  |  |
| --- | --- | --- |
| **Sinh viên thực hiện** | **Lớp** | **Khóa** |
| **Nguyễn Văn Hải** | **DC.CNTT12.10.3** | **12** |
| **Đỗ Minh Hiếu** | **DC.CNTT12.10.3** | **12** |
| **Phan Đức Huy** | **DC.CNTT12.10.3** | **12** |
| **Nguyễn Văn Tuân** | **DC.CNTT12.10.3** | **12** |
| **Nguyễn Huy Hoàng** | **DC.CNTT12.10.3** | **12** |

**Bắc Ninh, năm 2024**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á**

(bìa 1, bìa đóng giấy màu)



**BÀI TẬP LỚN**

**HỌC PHẦN: CÔNG NGHỆ JAVA**

**TÊN BÀI TẬP LỚN: XÂY DỰNG ỨNG DỤNG WEB ĐỂ TRA CỨU TIN TỨC SỬ DỤNG JAVA SERVLET**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sinh viên thực hiện** | **Khóa** | **Lớp** | **Mã sinh viên** |
| **Nguyễn Văn Hải** | **K12** | **DC.CNTT12.10.3** |  |
| **Đỗ Minh Hiếu** | **K12** | **DC.CNTT12.10.3** |  |
| **Phan Đức Huy** | **K12** | **DC.CNTT12.10.3** |  |
| **Nguyễn Văn Tuân** | **K12** | **DC.CNTT12.10.3** |  |
| **Nguyễn Huy Hoàng** | **K12** | **DC.CNTT12.10.3** |  |

**Bắc Ninh, tháng… năm 20…**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÀI TẬP LỚN**

**HỌC PHẦN: CÔNG NGHỆ JAVA**

**Nhóm: 8**

**CHỦ ĐỀ 7: BÀI TOÁN KHÁC**

**MÃ ĐỀ 154: XÂY DỰNG ỨNG DỤNG WEB ĐỂ TRA CỨU TIN TỨC SỬ DỤNG JAVA SERVLET**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Sinh viên thực hiện** | **Mã sinh viên** | **Điểm bằng số** | **Điểm bằng chữ** |
| **1** | **Nguyễn Văn Hải** | **20210864** |  |  |
| **2** | **Đỗ Minh Hiếu** | **20210805** |  |  |
| **3** | **Phan Đức Huy** | **20210664** |  |  |
| **4** | **Nguyễn Văn Tuân** | **20210724** |  |  |
| **5** | **Nguyễn Huy Hoàng** | **20210763** |  |  |

|  |  |
| --- | --- |
| **CÁN BỘ CHẤM 1**  *(Ký và ghi rõ họ tên)* | **CÁN BỘ CHẤM 2**  *(Ký và ghi rõ họ tên)* |

**Bắc Ninh, tháng ….. Năm 20 …**

**MỤC LỤC**

[DANH MỤC CÁC TỪ VIẾT TẮT 4](#_Toc171064156)

[DANH MỤC HÌNH ẢNH 5](#_Toc171064157)

[CHƯƠNG 1: GIỚI THIỆU VỀ ĐỀ TÀI 10](#_Toc171064158)

[1.1. Giới thiệu về đề tài 10](#_Toc171064159)

[1.2. Kế hoạch làm đề tài 11](#_Toc171064160)

[1.2.1. Phân tích yêu cầu 11](#_Toc171064161)

[1.2.2. Thiết kế hệ thống 11](#_Toc171064162)

[1.2.3. Phát triển ứng dụng 12](#_Toc171064163)

[1.2.4. Kiểm thử và triển khai 12](#_Toc171064164)

[1.3. Các công nghệ sử dụng 12](#_Toc171064165)

[CHƯƠNG 2: MÔ HÌNH HỆ THỐNG 16](#_Toc171064166)

[2.1. Các chức năng của hệ thống 16](#_Toc171064167)

[2.2. Các giao diện có trong hệ thống. 21](#_Toc171064168)

[2.3. Biểu đồ use case. 27](#_Toc171064169)

[2.4. Mô hình lớp. 32](#_Toc171064170)

[2.5. Mô hình cơ sở dữ liệu. 33](#_Toc171064171)

[2.6. Các tiện ích. 42](#_Toc171064172)

[CHƯƠNG 3: PHÁT TRIỂN HỆ THỐNG 43](#_Toc171064173)

[3.1. Lớp GUI 43](#_Toc171064174)

[3.2. Lớp BUS 54](#_Toc171064175)

[3.3. Lớp DAL. 64](#_Toc171064176)

[3.4 Kết chương. 67](#_Toc171064177)

[KẾT LUẬN 68](#_Toc171064178)

[DANH MỤC SÁCH THAM KHẢO 70](#_Toc171064179)

DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| STT | Chữ viết tắt | Giải thích |
| 1 | HTTP | HyperText Transfer Protocol |
| 2 | JSP | JavaServer Pages |

DANH MỤC HÌNH ẢNH

|  |  |  |
| --- | --- | --- |
| Số hiệu | Tên | Trang |
| 2.2.1 | Giao diện đăng nhập | **21** |
| 2.2.2 | Giao diện đăng ký | **21** |
| 2.2.3: | Giao diện trang chủ | **22** |
| 2.2.4 | Giao diện trang chi tiết tin tức | **22** |
| 2.2.5 | Giao diện bình luận | **23** |
| 2.2.6 | Giao diện tin tức đã lưu | **23** |
| 2.2.7 | Giao diện quản lý tin tức | **24** |
| 2.2.8 | Giao diện thêm tin tức | **24** |
| 2.2.9 | Giao diện chỉnh sửa tin tức | **25** |
| 2.2.10 | Giao diện chỉnh sửa thông tin cá nhân | **25** |
| 2.2.11 | Giao diện tìm kiếm bài viết | **26** |
| 2.3.1 | Biểu đồ use case tổng quát của hệ thống | **27** |
| 2.3.2 | Use case chức năng đăng nhập và đăng ký | **28** |
| 2.3.3 | Use case chức năng xem và thảo luận tin tức | **29** |
| 2.3.4 | Use case chức năng quản lý tài khoản cá nhân | **30** |
| 2.3.5 | Use case chức năng quản lý tin tức | **31** |
| 2.4.1 | Mô hình lớp | **32** |
| 2.5.1 | Mô hình cơ sở dữ liệu | **33** |
| 2.5.2 | Phương thức Login | **34** |
| 2.5.3 | Phương thức register | **34** |
| 2.5.4 | Phương thức checkemail | **34** |
| 2.5.5 | Phương thức getUserById | **34** |
| 2.5.6 | Phương thức getUserByEmail | **35** |
| 2.5.7 | Phương thức getTheloais | **35** |
| 2.5.8 | Phương thức getTheLoaiTinsByIdTheLoai | **35** |
| 2.5.9 | Phương thức getTheLoaiByTextUrl | **35** |
| 2.5.10 | Phương thức getTheLoaiTinById | **35** |
| 2.5.11 | Phương thức getTheLoaiById | **36** |
| 2.5.12 | Phương thức getAllEmotion | **36** |
| 2.5.13 | Phương thức getEmotionById | **36** |
| 2.5.14 | Phương thức GetEmotionsByIdBinhLuan | **36** |
| 2.5.15 | Phương thức getEmotionBy\_IdBinhLuan\_idEmotion | **36** |
| 2.5.16 | Phương thức EmotionAction | **37** |
| 2.5.17 | Phương thức GetBinhLuanRepliesByIdBinhLuan | **37** |
| 2.5.18 | Phương thức GetBinhLuansByIdBaiViet | **37** |
| 2.5.19 | Phương thức GetBaiViets | **37** |
| 2.5.20 | Phương thức GetBaiVietsByTheLoaiTin | **38** |
| 2.5.21 | Phương thức GetBaiVietsByTheLoai | **38** |
| 2.5.22 | Phương thức GetBaiVietsWithLimit | **38** |
| 2.5.23 | Phương thức GetBaiVietSaved | **38** |
| 2.5.24 | Phương thức SaveBaiViet | **39** |
| 2.5.25 | Phương thức RemoveSavedBaiViet | **39** |
| 2.5.26 | Phương thức deleteBaiViet | **39** |
| 2.5.27 | Phương thức AddBinhLuan | **39** |
| 2.5.28 | Phương thức addBaiViet | **40** |
| 2.5.29 | Phương thức updateUserName | **40** |
| 2.5.30 | Phương thức updateGender | **40** |
| 2.5.31 | Phương thức updateBirthday | **40** |
| 2.5.32 | Phương thức updateNumberphone | **40** |
| 2.5.33 | Phương thức updateAddress | **41** |
| 2.5.34 | Phương thức updateEmail | **41** |
| 2.5.35 | Phương thức updateAvatar | **41** |
| 2.5.36 | Phương thức updatePassword | **41** |
| 2.5.37 | Phương thức updateBaiViet | **42** |
| 3.1.1 | Trang đăng nhập | **43** |
| 3.1.2 | Trang đăng kí | **44** |
| 3.1.3 | Trang chủ | **45** |
| 3.1.4 | Trang chi tiết tin tức | **46** |
| 3.1.5 | Mục bình luận | **47** |
| 3.1.6 | Trang tin tức đã lưu | **48** |
| 3.1.7 | Trang thông tin cá nhân | **49** |
| 3.1.8 | Trang quản lý bài viết | **50** |
| 3.1.9 | Trang thêm bài viết | **51** |
| 3.1.10 | Trang cập nhật bài viết | **52** |
| 3.1.11 | Trang tìm kiếm bài viết | **53** |
| 3.2.1 | Lớp thực thể User | **54** |
| 3.2.2 | Lớp thực thể BaiViet | **55** |
| 3.2.3 | Lớp thực thể BaiVietSave | **56** |
| 3.2.4 | Lớp thực thể BinhLuan | **57** |
| 3.2.5 | Lớp thực thể BinhLuanEmotion | **58** |
| 3.2.6 | Lớp thực thể Emotion | **59** |
| 3.2.7 | Lớp thực thể TheLoai | **60** |
| 3.2.8 | Lớp thực thể TheLoaiTin | **61** |
| 3.2.9 | Interface IBaiViet | **61** |
| 3.2.10 | Interface IBinhLuan | **62** |
| 3.2.11 | Interface IEmotion | **62** |
| 3.2.12 | Interface ITheLoai | **62** |
| 3.2.13 | Interface IUser | **63** |
| 3.3.1 | Lớp DAL BaiVietService | **64** |
| 3.3.2 | Lớp DAL BinhLuanService | **64** |
| 3.3.3 | Lớp DAL EmotionService | **65** |
| 3.3.4 | Lớp DAL TheLoaiService | **65** |
| 3.3.5 | Lớp DAL UserService | **66** |

# CHƯƠNG 1: GIỚI THIỆU VỀ ĐỀ TÀI

# 1.1. Giới thiệu về đề tài

Trong bối cảnh thông tin ngày càng phát triển và nhu cầu cập nhật tin tức của người dùng ngày càng tăng, việc phát triển một ứng dụng web tra cứu tin tức trở nên cần thiết hơn bao giờ hết. Ứng dụng web tra cứu tin tức là một hệ thống giúp người dùng tìm kiếm và cập nhật các tin tức mới nhất từ nhiều nguồn khác nhau. Ứng dụng này không chỉ giúp người dùng tiết kiệm thời gian trong việc tìm kiếm thông tin mà còn cung cấp một nền tảng thuận tiện để theo dõi các tin tức quan trọng theo từng chủ đề cụ thể.

Ứng dụng web tra cứu tin tức sẽ thu thập, xử lý và hiển thị các bài viết tin tức theo các danh mục khác nhau như công nghệ, thể thao, giải trí, kinh doanh, và thời sự. Điều này giúp người dùng dễ dàng tìm kiếm thông tin trong lĩnh vực mà họ quan tâm mà không cần phải truy cập nhiều trang web khác nhau. Bằng cách cung cấp một giao diện thân thiện và dễ sử dụng, ứng dụng sẽ mang lại trải nghiệm tốt nhất cho người dùng, giúp họ tiếp cận nhanh chóng với các tin tức mới nhất và quan trọng nhất.

Ứng dụng được phát triển sử dụng Java Servlet, một công nghệ mạnh mẽ của Java EE (Enterprise Edition) cho phép xây dựng các ứng dụng web động. Java Servlet sẽ xử lý các yêu cầu từ phía người dùng, tương tác với các nguồn dữ liệu, và trả về các trang web được tạo động dựa trên thông tin mà người dùng yêu cầu. Việc sử dụng Java Servlet giúp đảm bảo rằng ứng dụng sẽ hoạt động ổn định, bảo mật và có khả năng mở rộng trong tương lai.

Để thu thập tin tức, ứng dụng sẽ tích hợp với các API của nguồn tin tức như News API và RSS feeds từ các trang báo nổi tiếng. News API là một dịch vụ cung cấp thông tin từ nhiều nguồn tin tức khác nhau trên toàn thế giới. Nó hỗ trợ nhiều tính năng như tìm kiếm theo từ khóa, lọc tin tức theo nguồn, ngôn ngữ, và ngày tháng. Việc sử dụng các API này giúp ứng dụng luôn cập nhật được các bài viết mới nhất, đảm bảo rằng người dùng sẽ không bỏ lỡ bất kỳ tin tức quan trọng nào.

Một phần quan trọng của ứng dụng là giao diện người dùng (UI). Giao diện sẽ được thiết kế để dễ dàng sử dụng và trực quan, với các chức năng tìm kiếm mạnh mẽ cho phép người dùng nhập từ khóa và nhận lại danh sách các bài viết liên quan trong thời gian thực. Các bài viết sẽ được hiển thị với tiêu đề, tóm tắt và liên kết đến bài viết gốc, giúp người dùng dễ dàng duyệt qua và lựa chọn những bài viết mà họ quan tâm.

Ngoài ra, ứng dụng cũng sẽ có các tính năng bổ sung như đăng ký và đăng nhập người dùng, cho phép người dùng lưu trữ và quản lý các bài viết yêu thích của họ. Tính năng thảo luận bài viết cũng sẽ được tích hợp, cho phép người dùng bình luận và chia sẻ ý kiến về các bài viết, tạo ra một cộng đồng thảo luận sôi động và gắn kết.

# 1.2. Kế hoạch làm đề tài

## 1.2.1. Phân tích yêu cầu

* Xác định các chức năng chính của hệ thống: Trước hết, cần xác định các chức năng mà hệ thống sẽ cung cấp. Các chức năng cơ bản bao gồm tìm kiếm tin tức, xem chi tiết bài viết, và thảo luận bài viết. Bên cạnh đó, hệ thống có thể cần chức năng đăng ký và đăng nhập người dùng, quản lý bài viết, và lưu trữ bài viết.
* Thiết kế giao diện người dùng sơ bộ: Tạo các bản vẽ giao diện sơ bộ (wireframe) cho các trang chính của ứng dụng, bao gồm trang chủ, trang tìm kiếm, trang chi tiết bài viết, và trang quản lý bài viết. Giao diện cần thân thiện, dễ sử dụng và trực quan.

## 1.2.2. Thiết kế hệ thống

* Thiết kế kiến trúc hệ thống: Hệ thống sẽ được thiết kế theo mô hình Client-Server, trong đó phía client là trình duyệt web của người dùng, và phía server là hệ thống xử lý yêu cầu sử dụng Java Servlet. Client sẽ gửi các yêu cầu HTTP tới server, và server sẽ xử lý các yêu cầu này và trả về phản hồi thích hợp.
* Thiết kế cơ sở dữ liệu: Thiết kế cơ sở dữ liệu để lưu trữ các thông tin cần thiết, bao gồm thông tin người dùng, bài viết, bình luận, và các danh mục tin tức. Cơ sở dữ liệu cần được thiết kế sao cho tối ưu cho việc truy xuất và lưu trữ dữ liệu.
* Lên kế hoạch chi tiết về các servlet và JSP cần thiết: Xác định các servlet cần thiết để xử lý các yêu cầu từ phía người dùng, chẳng hạn như Servlet xử lý tìm kiếm, Servlet xử lý xem chi tiết bài viết, và Servlet xử lý bình luận. Các trang JSP sẽ được sử dụng để hiển thị dữ liệu cho người dùng.

## 1.2.3. Phát triển ứng dụng

* Thiết lập môi trường phát triển: Cài đặt các công cụ cần thiết cho việc phát triển, bao gồm IDE (Eclipse, IntelliJ), và server (Apache Tomcat). Thiết lập cấu hình ban đầu cho dự án.
* Phát triển các thành phần front-end: Sử dụng HTML, CSS, và JavaScript để phát triển các giao diện người dùng. Đảm bảo giao diện thân thiện và đáp ứng tốt trên các thiết bị khác nhau.
* Phát triển các servlet để xử lý yêu cầu từ người dùng: Viết các servlet để xử lý các yêu cầu HTTP từ phía người dùng, chẳng hạn như tìm kiếm tin tức, xem chi tiết bài viết, và thảo luận bài viết. Đảm bảo các servlet xử lý đúng logic và trả về dữ liệu phù hợp.

## 1.2.4. Kiểm thử và triển khai

* Kiểm thử chức năng từng phần của ứng dụng: Thực hiện kiểm thử từng phần của ứng dụng để đảm bảo các chức năng hoạt động đúng. Sử dụng các công cụ kiểm thử và viết các test case để kiểm thử.
* Kiểm thử tích hợp và kiểm thử hệ thống: Sau khi kiểm thử từng phần, thực hiện kiểm thử tích hợp để đảm bảo các phần của hệ thống hoạt động tốt khi kết hợp với nhau. Kiểm thử toàn bộ hệ thống để đảm bảo không có lỗi.
* Triển khai ứng dụng lên server thực tế: Sau khi hoàn tất kiểm thử, triển khai ứng dụng lên server thực tế. Đảm bảo cấu hình server đúng và ứng dụng hoạt động ổn định.

# 1.3. Các công nghệ sử dụng

Ngôn ngữ lập trình: Java:

* Lý do chọn Java: Java là một ngôn ngữ lập trình hướng đối tượng mạnh mẽ và phổ biến, được sử dụng rộng rãi trong phát triển các ứng dụng web và doanh nghiệp. Java cung cấp tính bảo mật cao, hiệu năng ổn định và cộng đồng hỗ trợ lớn.
* Ứng dụng trong dự án: Java sẽ được sử dụng để viết các servlet, xử lý logic ứng dụng, kết nối với cơ sở dữ liệu và thực hiện các tác vụ liên quan đến backend.

Server: Apache Tomcat:

* Lý do chọn Apache Tomcat: Tomcat là một server ứng dụng mã nguồn mở phổ biến, hỗ trợ triển khai các ứng dụng web dựa trên Java Servlet và JSP. Tomcat dễ cài đặt, cấu hình và sử dụng, đồng thời có khả năng mở rộng và tích hợp tốt với các công cụ phát triển khác.
* Ứng dụng trong dự án: Tomcat sẽ được sử dụng để triển khai và chạy các servlet và JSP, phục vụ các yêu cầu từ trình duyệt của người dùng và phản hồi với nội dung động.

Giao diện người dùng:

* HTML:
* Lý do chọn HTML: HTML (HyperText Markup Language) là ngôn ngữ đánh dấu tiêu chuẩn để tạo ra các trang web. HTML định nghĩa cấu trúc cơ bản của trang web, cho phép nhúng các thành phần như văn bản, hình ảnh, liên kết, và các biểu mẫu.
* Ứng dụng trong dự án: HTML sẽ được sử dụng để xây dựng cấu trúc của các trang web, bao gồm trang chủ, trang tìm kiếm, trang chi tiết bài viết và trang thảo luận.
* CSS:
* Lý do chọn CSS: CSS (Cascading Style Sheets) là ngôn ngữ để mô tả cách trình bày của các phần tử HTML trên trang web. CSS giúp tách biệt phần nội dung và phần trình bày, dễ dàng quản lý và tùy chỉnh giao diện.
* Ứng dụng trong dự án: CSS sẽ được sử dụng để tạo kiểu dáng và bố cục cho các trang web, đảm bảo giao diện đẹp mắt, thân thiện và tương thích trên nhiều thiết bị.
* JavaScript:
* Lý do chọn JavaScript: JavaScript là ngôn ngữ lập trình kịch bản phía client, giúp tạo ra các tương tác động trên trang web. JavaScript có thể thay đổi nội dung trang mà không cần tải lại trang, nâng cao trải nghiệm người dùng.
* Ứng dụng trong dự án: JavaScript sẽ được sử dụng để xử lý các sự kiện từ người dùng, cập nhật nội dung động, thực hiện các truy vấn AJAX để tương tác với server mà không cần tải lại trang.

Java EE:

* Servlet:
* Lý do chọn Servlet: Servlet là thành phần quan trọng của Java EE, cho phép xử lý các yêu cầu HTTP và tạo ra các phản hồi động. Servlet dễ phát triển, mở rộng và tích hợp với các thành phần khác của Java EE.
* Ứng dụng trong dự án: Servlet sẽ được sử dụng để xử lý các yêu cầu từ phía client như tìm kiếm tin tức, xem chi tiết bài viết, thảo luận bài viết, và quản lý người dùng.
* JSP (JavaServer Pages):
* Lý do chọn JSP: JSP là công nghệ cho phép tạo ra các trang web động bằng cách nhúng mã Java vào các trang HTML. JSP dễ sử dụng và tích hợp tốt với các servlet và các thành phần khác của Java EE.
* Ứng dụng trong dự án: JSP sẽ được sử dụng để tạo ra các trang web động, hiển thị dữ liệu từ server và tương tác với người dùng.

Cơ sở dữ liệu: MySQL:

* Lý do chọn MySQL: MySQL là hệ quản trị cơ sở dữ liệu mã nguồn mở, phổ biến và đáng tin cậy, hỗ trợ SQL (Structured Query Language) để quản lý và truy vấn dữ liệu. MySQL có hiệu năng cao, dễ sử dụng và có cộng đồng hỗ trợ lớn.
* Ứng dụng trong dự án: MySQL sẽ được sử dụng để lưu trữ các thông tin cần thiết cho ứng dụng, bao gồm thông tin người dùng, bài viết, bình luận và danh mục tin tức. Các servlet sẽ kết nối với MySQL để thực hiện các thao tác CRUD (Create, Read, Update, Delete) trên dữ liệu.

Công cụ phát triển: IDE:

* Eclipse: Eclipse là môi trường phát triển tích hợp mã nguồn mở, hỗ trợ nhiều ngôn ngữ lập trình và có rất nhiều plugin hữu ích. Eclipse cung cấp nhiều tính năng như tự động hoàn thành mã, gỡ lỗi, và tích hợp với các công cụ quản lý dự án.
* IntelliJ IDEA: IntelliJ IDEA là một IDE mạnh mẽ và phổ biến, được biết đến với các tính năng thông minh như mã hóa tự động, gỡ lỗi, và tích hợp tốt với nhiều công cụ và framework khác.
* Ứng dụng trong dự án: Eclipse hoặc IntelliJ IDEA sẽ được sử dụng để viết mã, quản lý dự án và gỡ lỗi trong quá trình phát triển ứng dụng.

Quản lý phiên bản: GitHub:

* Lý do chọn GitHub: GitHub là dịch vụ lưu trữ mã nguồn và quản lý phiên bản dựa trên Git, cho phép các nhóm phát triển phần mềm hợp tác làm việc, theo dõi các thay đổi và quản lý các phiên bản của mã nguồn. GitHub cung cấp giao diện web trực quan và nhiều tính năng hỗ trợ làm việc nhóm như pull request, issues, và wiki.
* Ứng dụng trong dự án: GitHub sẽ được sử dụng để lưu trữ mã nguồn của dự án, theo dõi các thay đổi trong mã nguồn và hợp tác làm việc nhóm. Các thành viên trong nhóm có thể tạo nhánh (branch) để phát triển các tính năng mới, gửi pull request để đề xuất các thay đổi, và hợp nhất (merge) các nhánh vào nhánh chính (main branch) sau khi kiểm tra.

# CHƯƠNG 2: MÔ HÌNH HỆ THỐNG

# 2.1. Các chức năng của hệ thống

Đăng nhập, đăng ký tài khoản

1. Đăng nhập

Mô tả chức năng: Chức năng "Đăng nhập" cho phép người dùng truy cập vào hệ thống bằng tài khoản đã đăng ký. Người dùng cần cung cấp thông tin đăng nhập bao gồm email và mật khẩu.

Yêu cầu chi tiết:

* Giao diện người dùng: Form đăng nhập bao gồm các trường nhập liệu sau:
* Email: Trường nhập email (dạng văn bản).
* Mật khẩu: Trường nhập mật khẩu (dạng mật khẩu).
* Nút "Đăng nhập": Nút để gửi thông tin đăng nhập.
* Xử lý đăng nhập:
* Kiểm tra tính hợp lệ của email và mật khẩu.
* Kiểm tra xem các trường nhập email và mật khẩu có đang để trống hay không.
* Kiểm tra tài khoản có tồn tại trong hệ thống hay không.
* Nếu thông tin đăng nhập đúng, chuyển hướng người dùng đến trang chủ.
* Nếu thông tin đăng nhập sai, hiển thị thông báo lỗi và yêu cầu người dùng nhập lại.

Luồng hoạt động:

* Người dùng truy cập trang đăng nhập.
* Nhập email và mật khẩu.
* Nhấn nút "Đăng nhập".
* Hệ thống kiểm tra thông tin đăng nhập:
* Nếu hợp lệ: Đăng nhập thành công và chuyển hướng người dùng đến trang chủ.
* Nếu không hợp lệ: Hiển thị thông báo lỗi.

1. Đăng ký tài khoản

Mô tả chức năng: Chức năng "Đăng ký tài khoản" cho phép người dùng tạo tài khoản mới trên hệ thống. Người dùng cần cung cấp các thông tin cá nhân để đăng ký.

Yêu cầu chi tiết:

* Giao diện người dùng: Form đăng ký bao gồm các trường nhập liệu sau:
* Email: Trường nhập email (dạng văn bản).
* Mật khẩu: Trường nhập mật khẩu (dạng mật khẩu).
* Xác nhận mật khẩu: Trường nhập lại mật khẩu để xác nhận.
* Nút "Đăng ký": Nút để gửi thông tin đăng ký.
* Xử lý đăng ký:
* Kiểm tra tính hợp lệ của các trường nhập liệu (email hợp lệ, mật khẩu và xác nhận mật khẩu khớp nhau).
* Kiểm tra email có đã được sử dụng để đăng ký tài khoản trước đó hay chưa.
* Kiểm tra xem các trường nhập vào có đang để trống hay không.
* Nếu thông tin hợp lệ, tạo tài khoản mới và lưu trữ thông tin người dùng trong cơ sở dữ liệu.
* Nếu thông tin không hợp lệ, hiển thị thông báo lỗi và yêu cầu người dùng nhập lại.

Luồng hoạt động:

* Người dùng truy cập trang đăng ký.
* Nhập các thông tin cá nhân: tên người dùng, email, mật khẩu và xác nhận mật khẩu.
* Nhấn nút "Đăng ký".
* Hệ thống kiểm tra thông tin đăng ký:
* Nếu hợp lệ: Tạo tài khoản mới và chuyển hướng người dùng đến trang đăng nhập.
* Nếu không hợp lệ: Hiển thị thông báo lỗi và yêu cầu người dùng nhập lại.

Xem tin tức theo danh mục.

Mô tả chức năng: Chức năng "Hiển thị tin tức theo danh mục" cho phép người dùng xem các tin tức được phân loại theo từng danh mục cụ thể. Danh mục có thể bao gồm các chủ đề khác nhau như công nghệ, giải trí, thể thao, sức khỏe, v.v.

Yêu cầu chi tiết:

* Giao diện người dùng:
* Danh sách danh mục: Hiển thị danh sách các danh mục tin tức. Người dùng có thể chọn một danh mục để xem các tin tức thuộc danh mục đó.
* Danh sách tin tức: Hiển thị danh sách các tin tức thuộc danh mục đã chọn. Mỗi tin tức bao gồm các thông tin sau: Tiêu đề tin tức, tóm tắt ngắn gọn về nội dung tin tức, tác giả, ngày đăng, hình ảnh.

Luồng hoạt động:

* Người dùng truy cập trang danh mục.
* Trang danh mục hiển thị danh sách các danh mục tin tức.
* Người dùng chọn một danh mục cụ thể.
* Trang hiển thị danh sách các tin tức thuộc danh mục đã chọn.

Tìm kiếm tin tức.

Giao diện người dùng:

* Trường tìm kiếm: Người dùng nhập từ khóa tìm kiếm vào trường tìm kiếm.
* Nút tìm kiếm: Nút để gửi yêu cầu tìm kiếm.
* Kết quả tìm kiếm: Hiển thị danh sách các tin tức phù hợp với từ khóa tìm kiếm. Mỗi tin tức bao gồm các thông tin sau:
* Tiêu đề tin tức.
* Tóm tắt ngắn gọn về nội dung tin tức.
* Hình ảnh.

Luồng hoạt động:

* Người dùng truy cập trang tìm kiếm.
* Nhập từ khóa vào ô tìm kiếm.
* Nhấn nút "Tìm kiếm".
* Hệ thống xử lý yêu cầu tìm kiếm và hiển thị danh sách các tin tức phù hợp.

Quản lý tài khoản cá nhân.

Mô tả chức năng: Chức năng "Quản lý tài khoản cá nhân" cho phép người dùng xem và cập nhật thông tin cá nhân của họ trên hệ thống. Người dùng có thể chỉnh sửa các thông tin như tên, email, mật khẩu, và ảnh đại diện.

Giao diện người dùng:

* Thông tin tài khoản: Hiển thị các thông tin cá nhân của người dùng bao gồm: Tên người dùng, email, ảnh đại diện. Cung cấp form để người dùng chỉnh sửa các thông tin cá nhân.
* Nút "Lưu thay đổi": Nút để gửi thông tin cập nhật.
* Nút "Hủy bỏ": Nút để hủy bỏ các thay đổi và trở về trang thông tin tài khoản.

Luồng hoạt động:

* Người dùng truy cập vào trang thông tin tài khoản.
* Hệ thống hiển thị các thông tin cá nhân hiện tại của người dùng.
* Người dùng có thể nhấn nút "Chỉnh sửa" để vào chế độ chỉnh sửa thông tin.
* Người dùng chỉnh sửa các thông tin cần thay đổi và nhấn nút "Lưu thay đổi".
* Hệ thống kiểm tra và lưu các thay đổi, cập nhật thông tin trong cơ sở dữ liệu.
* Nếu người dùng không muốn thay đổi, họ có thể nhấn nút "Hủy bỏ" để quay lại trang thông tin tài khoản mà không lưu các thay đổi.

Lưu tin tức.

Giao diện người dùng:

* Nút lưu tin tức: Hiển thị nút "Lưu" hoặc biểu tượng cái cờ tại mỗi tin tức để người dùng có thể lưu tin tức.
* Danh sách tin tức đã lưu: Trang riêng biệt hiển thị danh sách các tin tức mà người dùng đã lưu. Mỗi tin tức trong danh sách bao gồm:
* Tiêu đề tin tức.
* Tóm tắt ngắn gọn về nội dung tin tức.
* Tác giả.
* Ngày đăng.
* Hình ảnh.
* Nút "Xóa" hoặc biểu tượng thùng rác để xóa tin tức khỏi danh sách lưu.

Luồng hoạt động:

* Người dùng xem danh sách hoặc chi tiết tin tức.
* Người dùng nhấn nút "Lưu" hoặc biểu tượng cái cờ để lưu tin tức vào danh sách tin đã lưu.
* Tin tức được thêm vào danh sách tin tức đã lưu của người dùng.
* Người dùng có thể truy cập trang danh sách tin tức đã lưu để xem lại các tin tức đã lưu.
* Tại trang danh sách tin tức đã lưu, người dùng có thể xóa tin tức khỏi danh sách bằng cách nhấn nút "Xóa" hoặc biểu tượng thùng rác.

Xem và thảo luận tin tức.

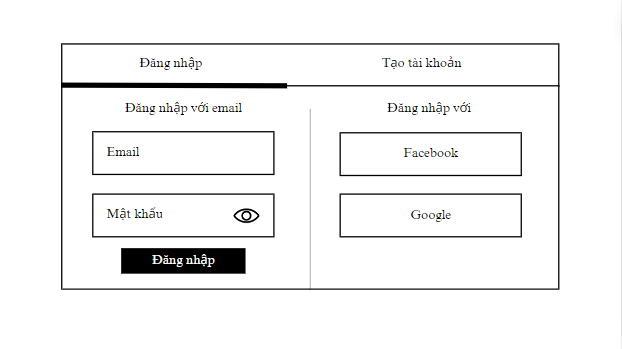
Mô tả chức năng: Chức năng "Xem và thảo luận tin tức" cho phép người dùng đọc chi tiết tin tức và tham gia thảo luận thông qua việc bình luận, trả lời bình luận, và bày tỏ cảm xúc về bài viết.

Luồng hoạt động:

* Người dùng truy cập trang chi tiết tin tức.
* Người dùng đọc nội dung tin tức và có thể bày tỏ cảm xúc bằng cách nhấn nút "Thích".
* Người dùng có thể xem danh sách các bình luận dưới bài viết.
* Người dùng nhập nội dung vào form bình luận và nhấn nút "Gửi" để đăng bình luận.
* Bình luận của người dùng được hiển thị trong danh sách bình luận.
* Người dùng có thể nhấn nút "Trả lời" để trả lời các bình luận khác.
* Người dùng có thể nhấn nút "Thích" để bày tỏ cảm xúc về các bình luận.

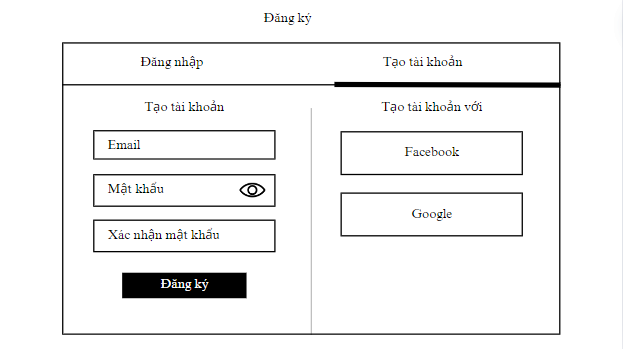
# 2.2. Các giao diện có trong hệ thống.

Giao diện đăng nhập



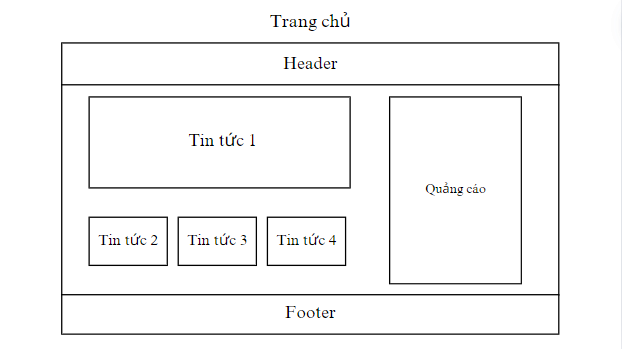
Hình 2.2.1: Giao diện đăng nhập

Giao diện đăng ký



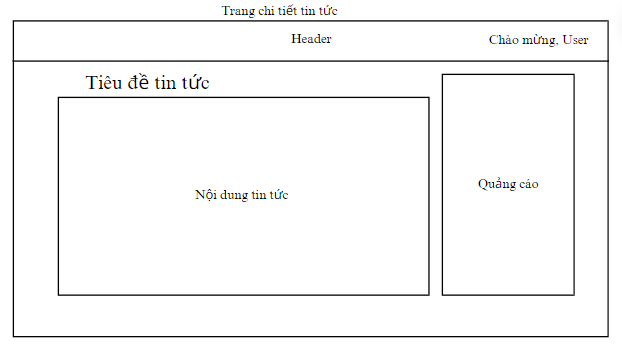
Hình 2.2.2: Giao diện đăng ký

Giao diện trang chủ



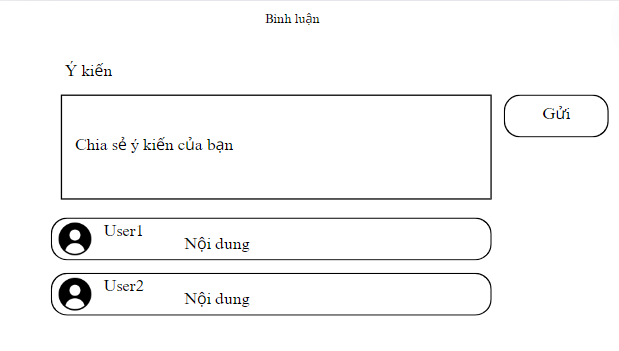
Hình 2.2.3: Giao diện trang chủ

Giao diện trang chi tiết tin tức



Hình 2.2.4: Giao diện trang chi tiết tin tức

Giao diện bình luận



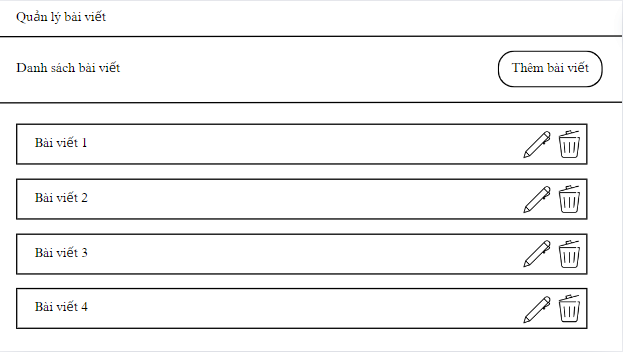
Hình 2.2.5: Giao diện bình luận

Giao diện tin tức đã lưu



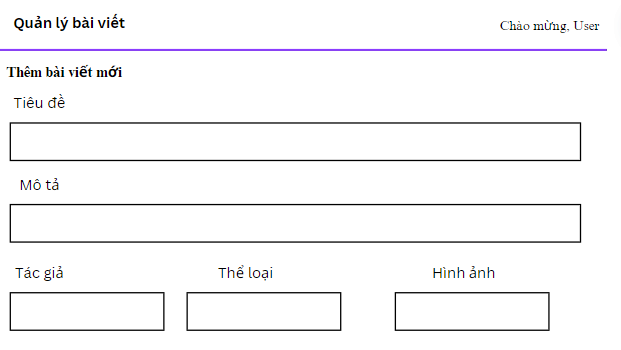
Hình 2.2.6: Giao diện tin tức đã lưu

Giao diện quản lý bài viết



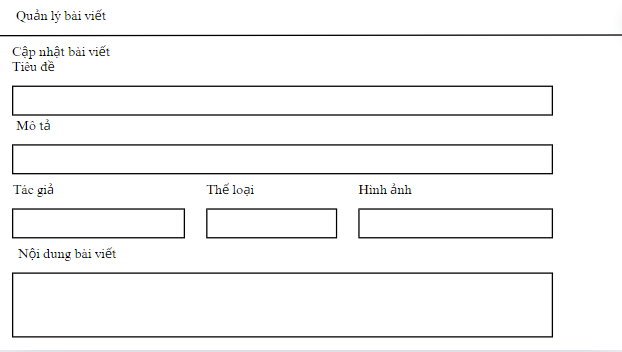
Hình 2.2.7: Giao diện quản lý tin tức

Giao diện thêm bài viết



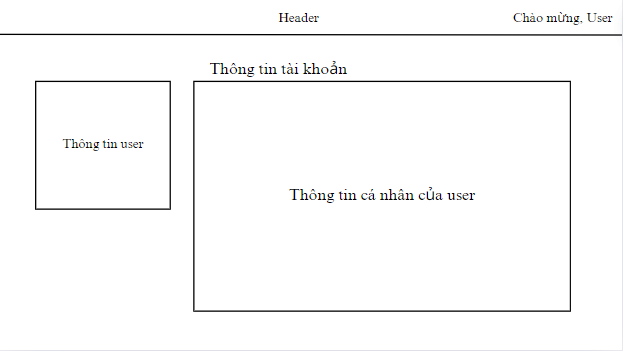
Hình 2.2.8: Giao diện thêm tin tức

Giao diện sửa bài viết



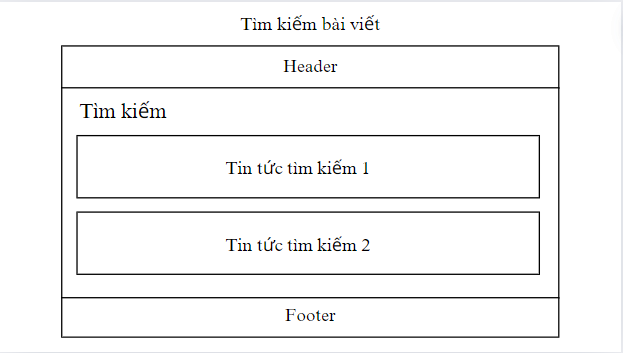
Hình 2.2.9: Giao diện chỉnh sửa tin tức

Giao diện chỉnh sửa thông tin cá nhân



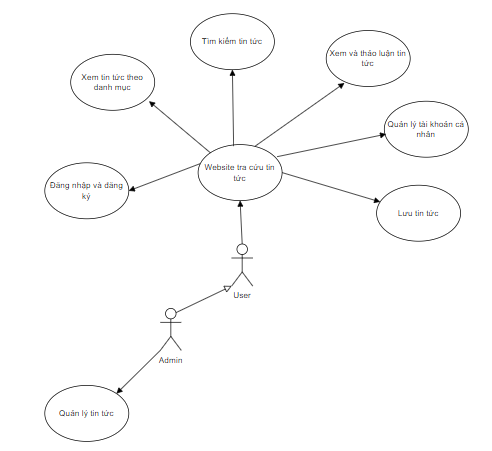
Hình 2.2.10: Giao diện chỉnh sửa thông tin cá nhân

Giao diện tìm kiếm bài viết



Hình 2.2.11: Giao diện tìm kiếm bài viết

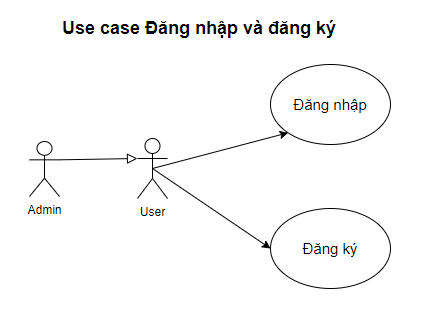
# 2.3. Biểu đồ use case.



Hình 2.3.1: Biểu đồ use case tổng quát của hệ thống

Mô tả chi tiết:

* User:
* Đăng nhập và đăng ký: Người dùng có thể đăng nhập hoặc đăng ký để truy cập vào trang web.
* Quản lý tài khoản cá nhân: Người dùng có thể thay đổi thông tin cá nhân của mình.
* Xem tin tức theo danh mục: Người dùng có thể xem tin tức theo danh mục của tin tức đã được phân loại theo chủ đề.
* Tìm kiếm tin tức: Người dùng có thể tìm kiếm tin tức theo từ khoá có trong tiêu đề tin tức.
* Xem và thảo luận tin tức: Người dùng có thể để lại bình luận, cảm xúc của bản thân ở dưới tin tức, trả lời bình luận của user khác.
* Lưu tin tức: Người dùng có thể lưu lại tin tức đã xem vào 1 trang riêng.
* Admin:
* Quản lý tin tức: Thêm sửa xoá tin tức có ở trên trang web.



Hình 2.3.2: Use case chức năng đăng nhập và đăng ký

**Đặc tả use-case:**

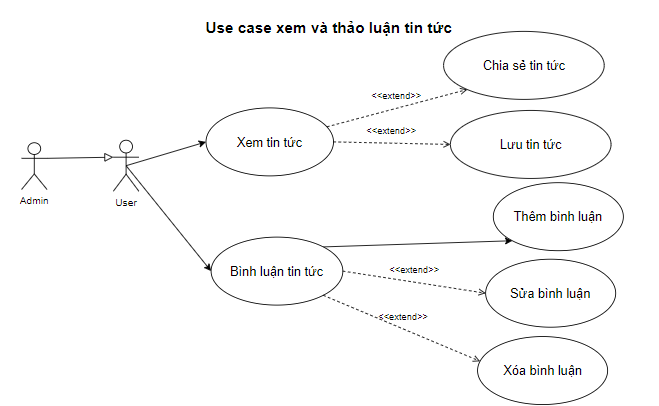
Actor: Admin, User

**Mô tả:**

Admin kế thừa chức năng đăng nhập và đăng ký từ User.

**Luồng sự kiện:**

* User truy cập vào trang web, nếu chưa có tài khoản có thể đăng ký còn có tài khoản rồi thì tiến hành đăng nhập.
* Hệ thống kiểm tra tài khoản và xác định quyền của người dùng,
* Đăng nhập thành công hiển thị chức năng sử dụng của tài khoản. User tiến hành sử dụng chức năng con của web.



Hình 2.3.3: Use case chức năng xem và thảo luận tin tức

**Đặc tả use-case:**

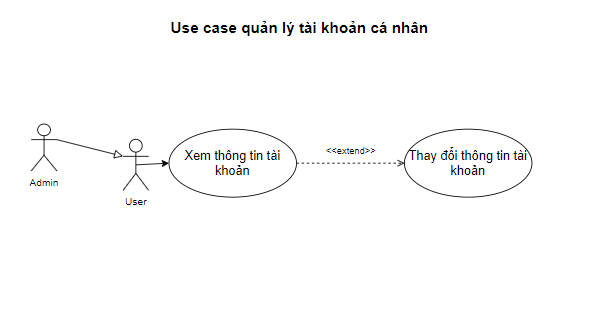
Actor: User, Admin

**Mô tả:**

Admin kế thừa chức năng xem và thảo luận từ User. User và Admin đều được sử dụng những chức năng con của nó.

**Luồng sự kiện:**

* Người dùng truy cập trang chi tiết tin tức.
* Người dùng đọc nội dung tin tức và có thể bày tỏ cảm xúc bằng cách nhấn nút "Thích".
* Người dùng có thể xem danh sách các bình luận dưới bài viết.
* Người dùng nhập nội dung vào form bình luận và nhấn nút "Gửi" để đăng bình luận.
* Bình luận của người dùng được hiển thị trong danh sách bình luận.
* Người dùng có thể nhấn nút "Trả lời" để trả lời các bình luận khác.
* Người dùng có thể nhấn nút "Thích" để bày tỏ cảm xúc về các bình luận.



Hình 2.3.4: Use case chức năng quản lý tài khoản cá nhân

**Đặc tả use-case:**

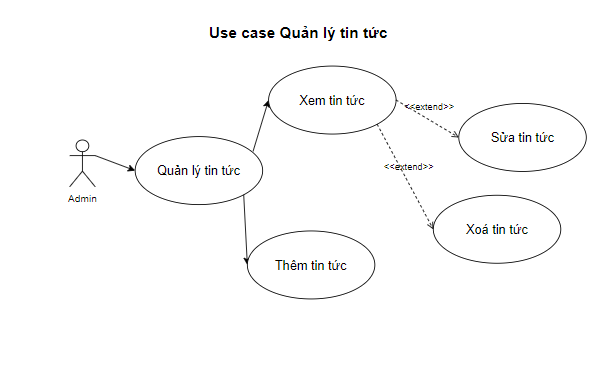
Actor: User, Admin

**Mô tả:**

Admin kế thừa chức năng quản lý tài khoản cá nhân từ User. User và Admin đều được sử dụng những chức năng con của nó.

**Luồng sự kiện:**

* Người dùng truy cập vào trang thông tin tài khoản.
* Hệ thống hiển thị các thông tin cá nhân hiện tại của người dùng.
* Người dùng có thể nhấn nút "Chỉnh sửa" để vào chế độ chỉnh sửa thông tin.
* Người dùng chỉnh sửa các thông tin cần thay đổi và nhấn nút "Lưu thay đổi".
* Hệ thống kiểm tra và lưu các thay đổi, cập nhật thông tin trong cơ sở dữ liệu.
* Nếu người dùng không muốn thay đổi, họ có thể nhấn nút "Hủy bỏ" để quay lại trang thông tin tài khoản mà không lưu các thay đổi.



Hình 2.3.5: Use case chức năng quản lý tin tức

**Đặc tả use-case:**

Actor: Admin

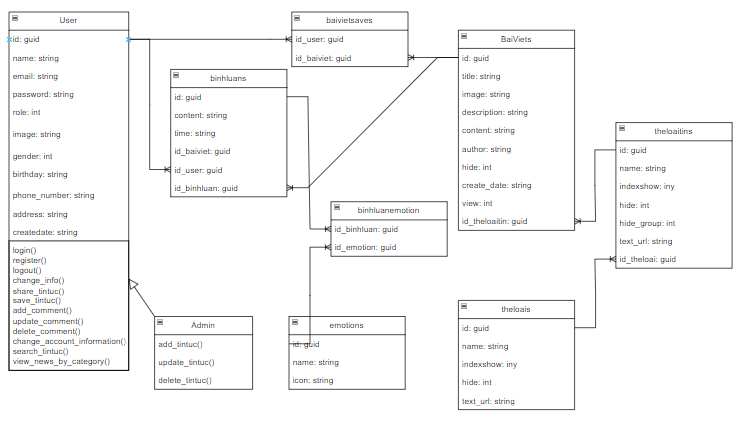
**Mô tả:**

Thể hiện sự tương tác của admin với chức năng quản lý tin tức trong trang web.

**Luồng sự kiện:**

* Admin truy cập vào trang web.
* Admin chuyển tiếp sang trang quản lý bài viết.
* Admin thực hiện thêm, sửa, xoá bài viết.

# 2.4. Mô hình lớp.

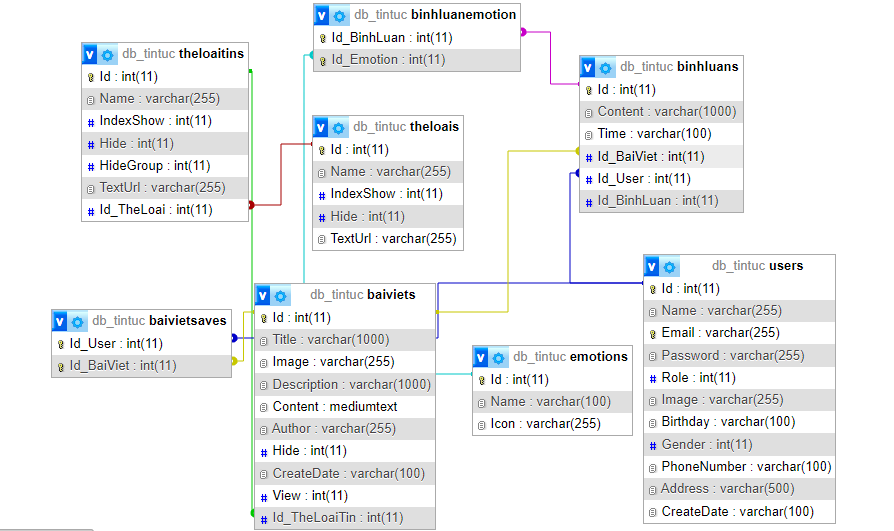


Hình 2.4.1: Mô hình lớp

Trong mô hình lóp trên có các lớp sau đây:

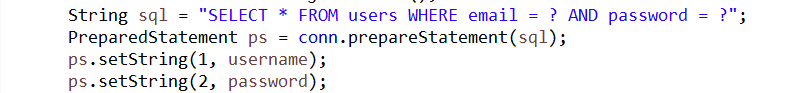
* Lớp User: lớp này bao gồm các thuộc tính: ID, name, email, password, role, image, gender, birthday, phone\_number, address, create\_date. Lớp này được lớp Admin kế thừa. Các thuộc tính trên giúp cho các lớp kế thừa không cần định nghĩa lại, mà chỉ cần sử dụng luôn. Lớp này có các phương thức:
* Login(): Giúp user có thể đăng nhập bằng tài khoản đã đăng kí.
* Register(): Giúp user có thể đăng ký tài khoản. Phương thức này yêu cầu user nhập vào thông tin hợp lệ mới có thể đăng ký tài khoản.
* Logout(): Giúp user có thể đăng xuất khi đang đăng nhập.
* Change\_infor(): Giúp user có thể thay đổi thông tin cá nhân.
* Share\_tintuc(): Giúp user có thể chia sẻ tin tức lên các mạng xã hội khác.
* Save\_tintuc(): Giúp user có thể lưu lại tin tức vào trang tin tức đã lưu.
* Add\_comment(): Giúp user có thể để lại bình luận, bàn luận, cảm xúc của bản thân vào dưới bài báo.
* Update\_comment(): Giúp user có thể sửa lại bình luận, cảm xúc.
* Delete\_comment(): Giúp user có thể xoá bình luận, cảm xúc.
* Change\_account\_information(): Giúp user thay đổi thông tin cá nhân.
* Search\_tintuc():Giúp user tìm kiếm tin tức theo từ khoá.
* View\_news\_by\_category(): Giúp user đọc tin tức theo danh mục.
* Lớp Admin: Lớp này gồm những thuộc tính được kế thừa từ User. Lớp này có các phương thức:
* Add\_tintuc(): Giúp admin thêm tin tức vào trang web.
* Update\_tintuc(): Giúp admin sửa tin tức trang web.
* Delete\_tintuc(): Giúp admin xoá tin tức có trong trang web.
* Lớp User được kế thừa bởi lớp Admin.

# 2.5. Mô hình cơ sở dữ liệu.

******

Hình 2.5.1: Mô hình cơ sở dữ liệu

***Các truy vấn:***

******

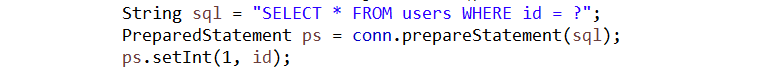
Hình 2.5.2: Phương thức Login



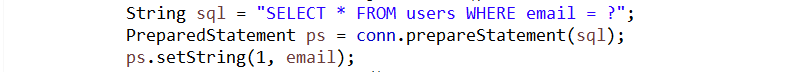
Hình 2.5.3: Phương thức register



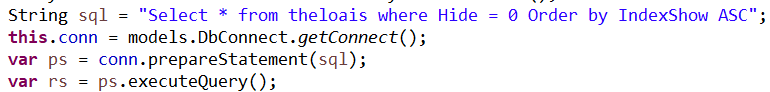
Hình 2.5.4: Phương thức checkemail



Hình 2.5.5: Phương thức getUserById



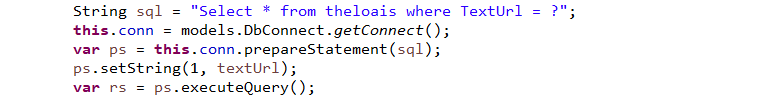
Hình 2.5.6: Phương thức getUserByEmail



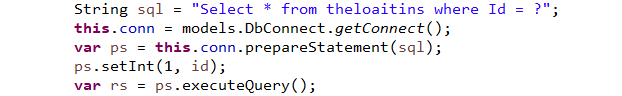
Hình 2.5.7: Phương thức getTheloais



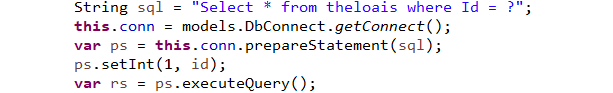
Hình 2.5.8: Phương thức getTheLoaiTinsByIdTheLoai



Hình 2.5.9: Phương thức getTheLoaiByTextUrl



Hình 2.5.10: Phương thức getTheLoaiTinById



Hình 2.5.11: Phương thức getTheLoaiById



Hình 2.5.12: Phương thức getAllEmotion



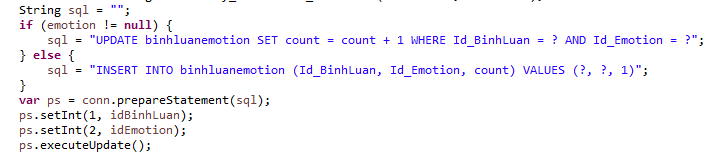
Hình 2.5.13: Phương thức getEmotionById



Hình 2.5.14: Phương thức GetEmotionsByIdBinhLuan



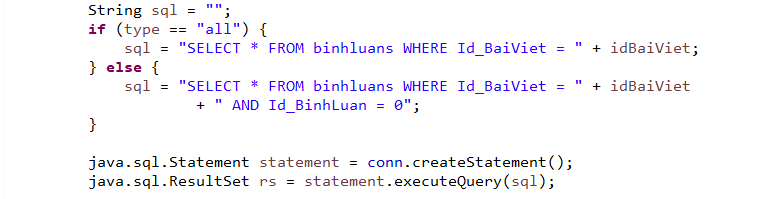
Hình 2.5.15: Phương thức getEmotionBy\_IdBinhLuan\_idEmotion



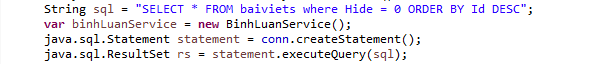
Hình 2.5.16: Phương thức EmotionAction



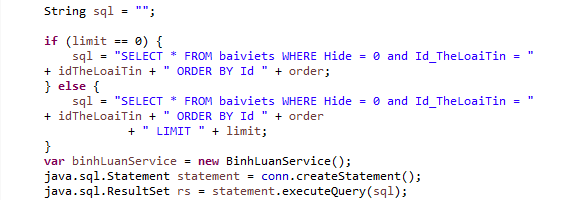
Hình 2.5.17: Phương thức GetBinhLuanRepliesByIdBinhLuan



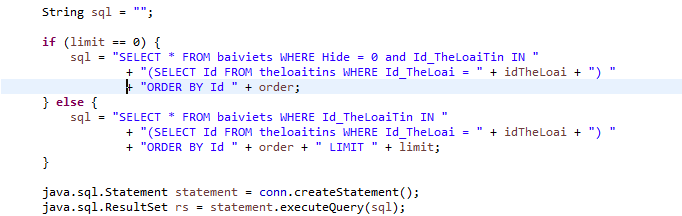
Hình 2.5.18: Phương thức GetBinhLuansByIdBaiViet



Hình 2.5.19: Phương thức GetBaiViets



Hình 2.5.20: Phương thức GetBaiVietsByTheLoaiTin



Hình 2.5.21: Phương thức GetBaiVietsByTheLoai

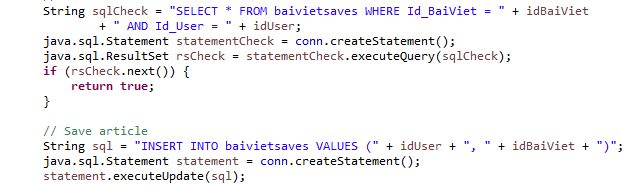


Hình 2.5.22: Phương thức GetBaiVietsWithLimit



Hình 2.5.23: Phương thức GetBaiVietSaved

**Các cập nhật dữ liệu:**



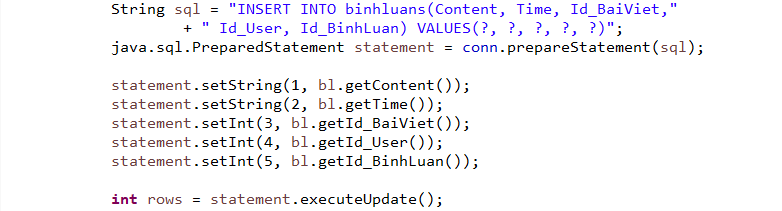
Hình 2.5.24: Phương thức SaveBaiViet



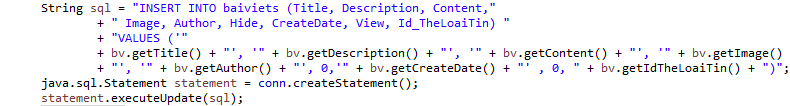
Hình 2.5.25: Phương thức RemoveSavedBaiViet



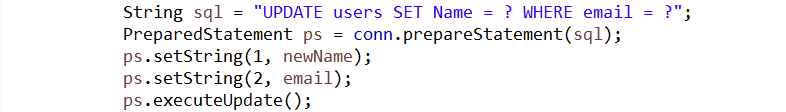
Hình 2.5.26: Phương thức deleteBaiViet



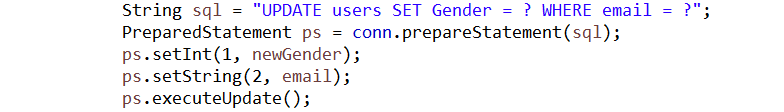
Hình 2.5.27: Phương thức AddBinhLuan



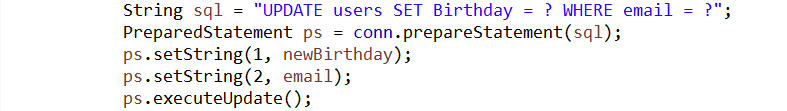
Hình 2.5.28: Phương thức addBaiViet



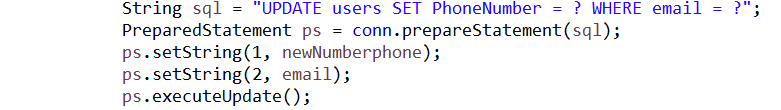
Hình 2.5.29: Phương thức updateUserName



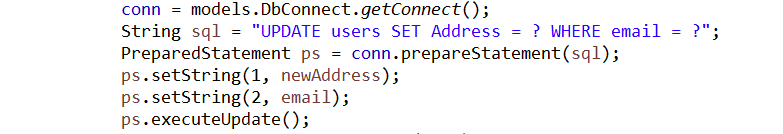
Hình 2.5.30: Phương thức updateGender



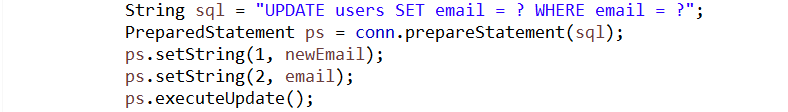
Hình 2.5.31: Phương thức updateBirthday



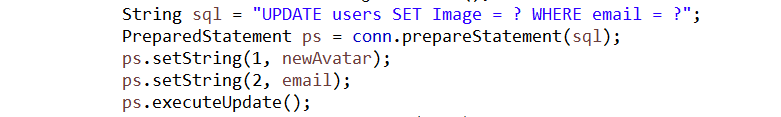
Hình 2.5.32: Phương thức updateNumberphone



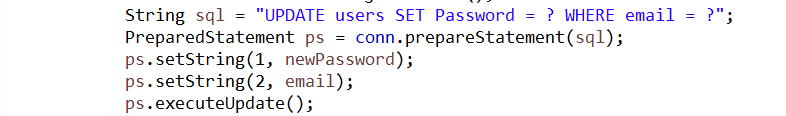
Hình 2.5.33: Phương thức updateAddress



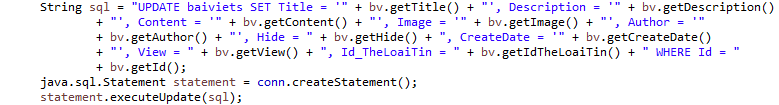
Hình 2.5.34: Phương thức updateEmail



Hình 2.5.35: Phương thức updateAvatar



Hình 2.5.36: Phương thức updatePassword



Hình 2.5.37: Phương thức updateBaiViet

# 2.6. Các tiện ích.

**Các tiện ích trên trang web bao gồm:**

**Tiện ích tìm kiếm:** Cho phép người dùng dễ dàng tìm kiếm tin tức mà họ đang cần.

**Tiện ích lưu tin tức:** Hiển thị các tin tức đã được lưu và cho phép người dùng thêm/xóa tin tức khỏi trang tin tức đã lưu.

**Tiện ích đăng nhập/tạo tài khoản:** Cho phép người dùng đăng nhập vào tài khoản của họ hoặc tạo tài khoản mới để xem tin tức.

**Tiện ích bình luận:** Người dùng có thể để lại bình luận và cảm xúc tại bên dưới tin tức.

**Tiện ích xem tin tức theo danh mục:** Đây là tiện ích quan trọng cho phép người dùng xem tin tức theo danh mục đã được phân loại một cách nhanh chóng, đúng chủ đề.

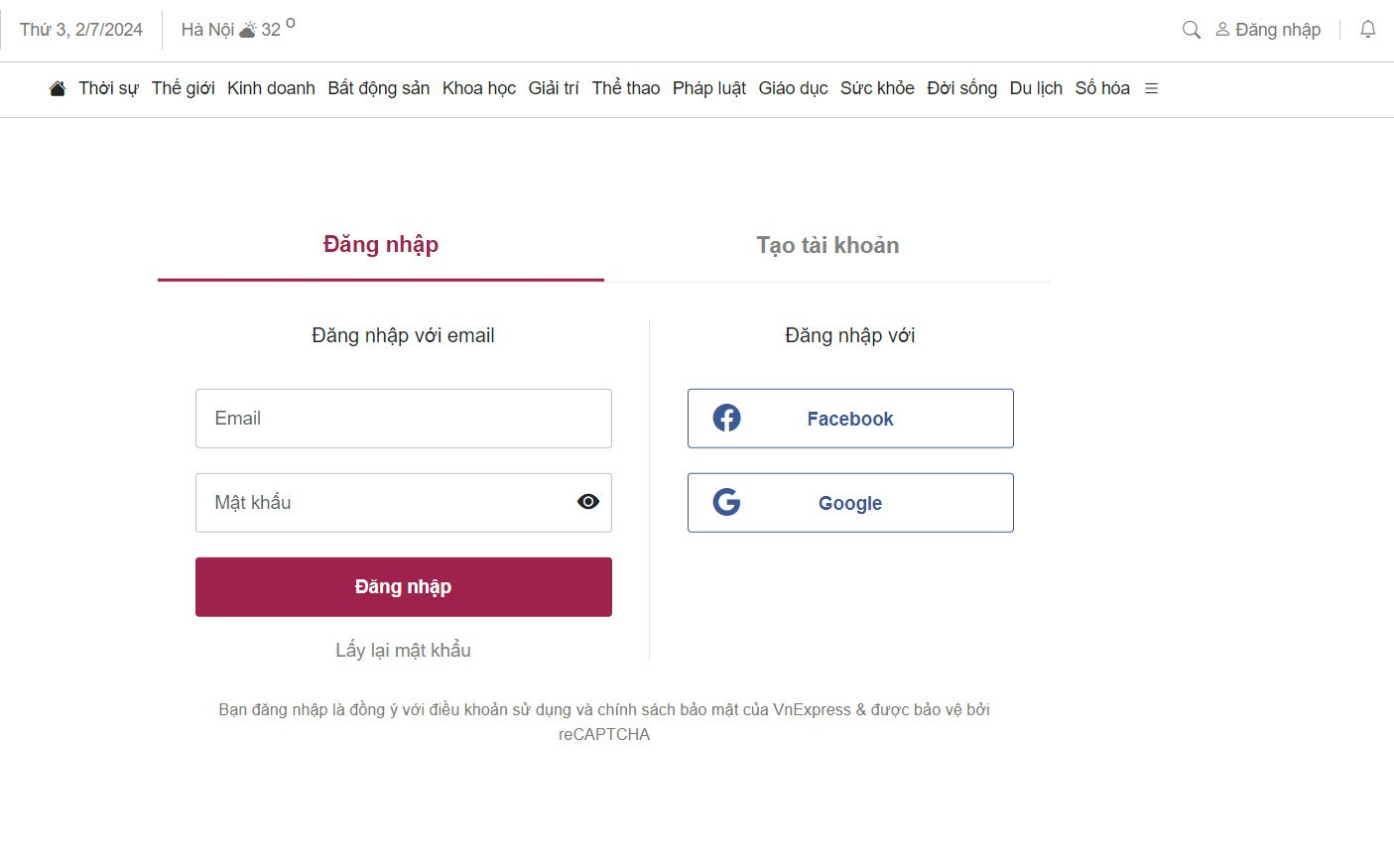
# CHƯƠNG 3: PHÁT TRIỂN HỆ THỐNG

# 3.1. Lớp GUI

**Giao diện trang đăng nhập**

Mối quan hệ:

1. Khi bấm vào button đăng nhập hệ thống sẽ kiểm tra thông tin tên đăng nhập, mật khẩu của khách hàng nếu thông tin chính xác thì trang chủ sẽ được mở lên.
2. Khi bấm vào tab đăng ký thì sẽ chuyển sang tab đăng ký sẽ được mở lên.
3. Khi nhập tài khoản admin thì sẽ thêm chức năng quản lý bài viết.

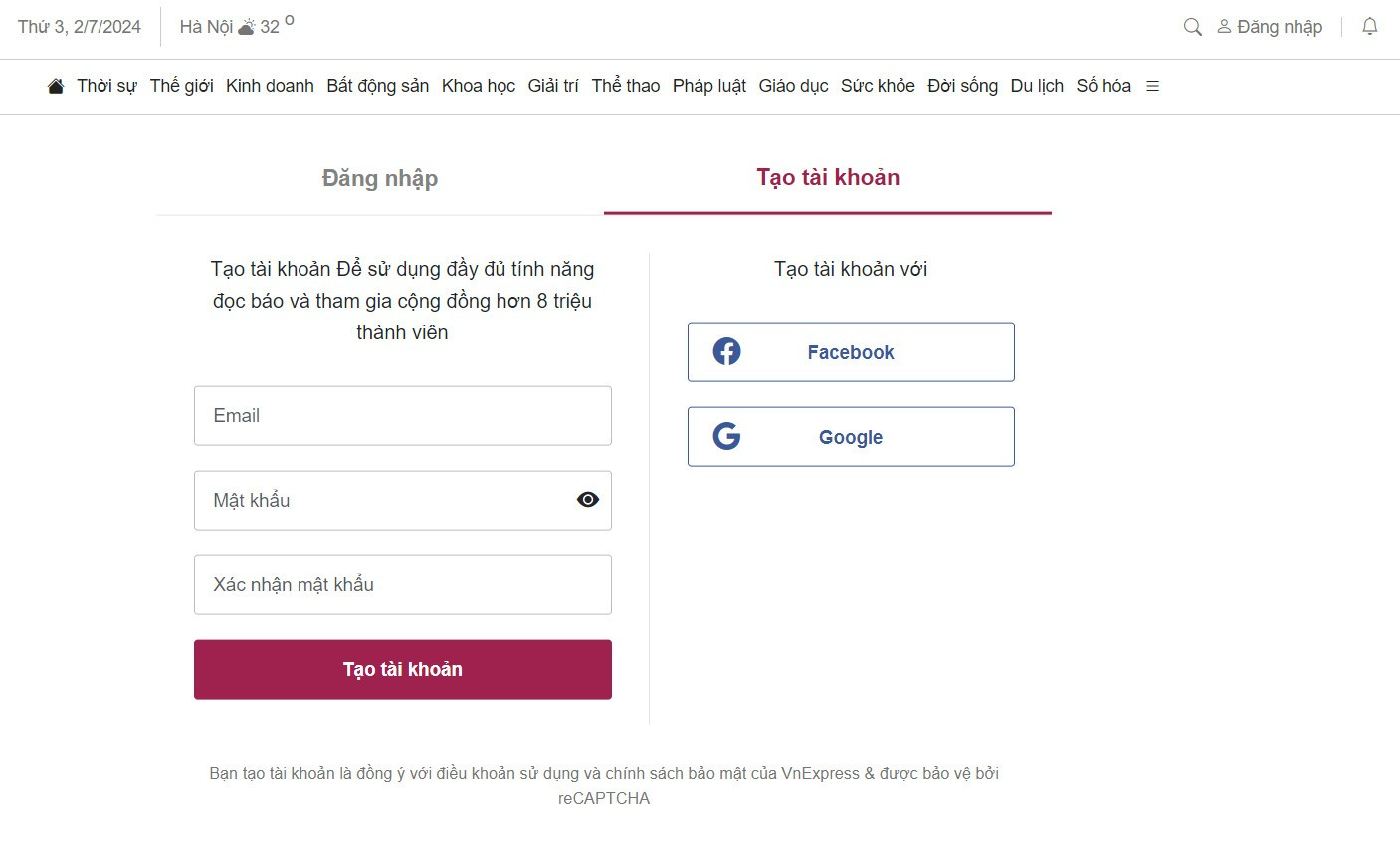


Hình 3.1.1: Trang đăng nhập

**Giao diện trang đăng ký**

Mối quan hệ:

1. Khi bấm button tạo tài khoản nếu đăng ký thành công thì vào trang chủ.



Hình 3.1.2: Trang đăng kí

**Giao diện trang chủ**

Mối quan hệ:

1. Khi bấm vào 1 bài báo bất kì thì sẽ hiển thị thông tin chi tiết của bài báo đó.
2. Khi bấm vào thanh thể loại thì sẽ hiển thị những bài báo với thể loại đã được phân.
3. Khi bấm vào hình ngôi nhà sẽ quay lai trang chủ



Hình 3.1.3: Trang chủ

**Giao diện trang chi tiết tin tức**

Mối quan hệ:

1. Khi bấm vào 1 bài báo bất kì ở mục xem nhiều thì sẽ hiển thị thông tin chi tiết của bài báo đó.
2. Khi bấm vào thanh thể loại thì sẽ hiển thị những bài báo với thể loại đã được phân.
3. Khi bấm vào hình ngôi nhà sẽ quay lai trang chủ.
4. Khi bấm vào mục bình luận sẽ được bình luận.
5. Khi bấm nút lưu, bài viết sẽ được lưu vào phần bài viết đã lưu.



Hình 3.1.4: Trang chi tiết tin tức

**Giao diện bình luận**

Mối quan hệ:

1. Khi bấm vào chia sẻ ý kiến, nhập ý kiến rồi ấn gửi thì bình luận sẽ được đẩy lên.
2. Bấm vào nút thích để thả biểu cảm
3. Bấm trả lời để trả lời bình luận

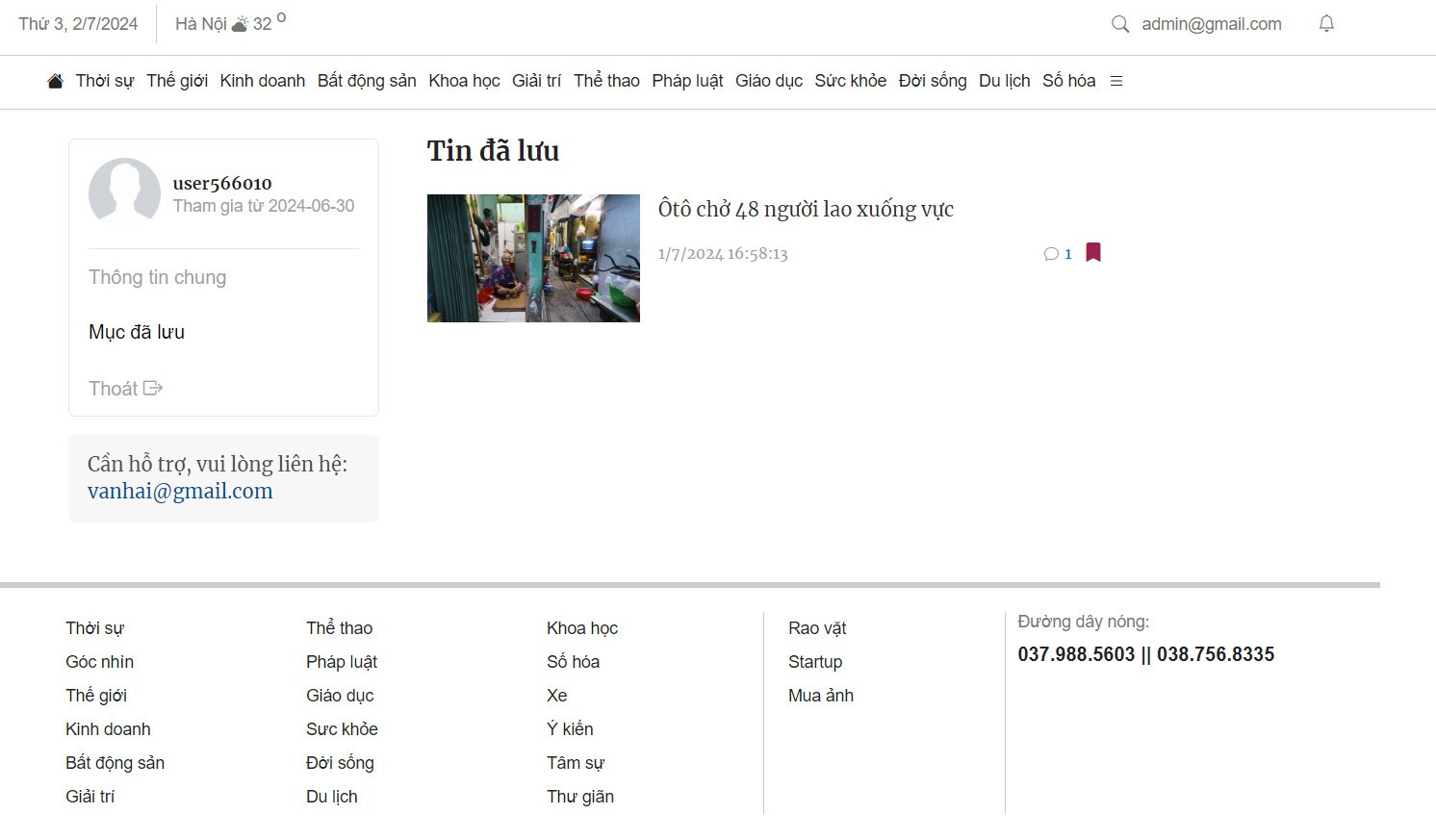


Hình 3.1.5: Mục bình luận

**Giao diện trang tin tức đã lưu**

Mối quan hệ:

1. Khi bấm vào bài viết đã lưu thì sẽ hiển thị chi tiết bài viết đó.
2. Khi bấm vào thanh thể loại thì sẽ hiển thị những bài báo với thể loại đã được phân.
3. Khi bấm vào hình ngôi nhà sẽ quay lai trang chủ.
4. Bấm thoát để quay lai trang chủ.
5. Khi bấm vào email, sẽ được chọn chuyển tới đăng xuất, tin đã lưu, chỉnh sửa thông tin cá nhân.

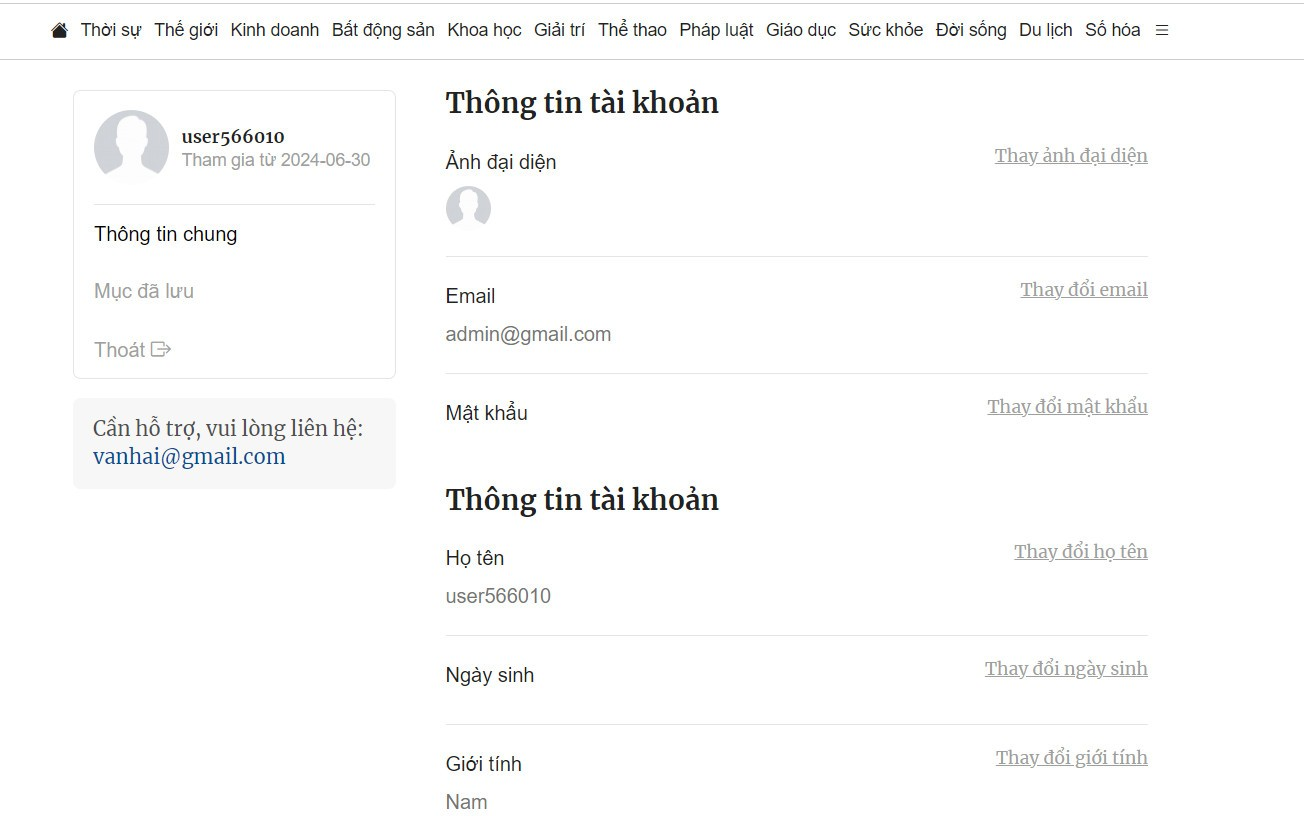


Hình 3.1.6: Trang tin tức đã lưu

**Giao diện trang chỉnh sửa thông tin cá nhân**

Mối quan hệ:

1. Khi bấm vào bài viết đã lưu thì sẽ tới trang bài viết đã lưu.
2. Thoát sẽ thoát ra trang chủ.
3. Tuỳ chọn thay đổi để thay đổi thông tin cần thay đổi.
4. Chọn thể loại để đi tới những bài viết thể loại đó.

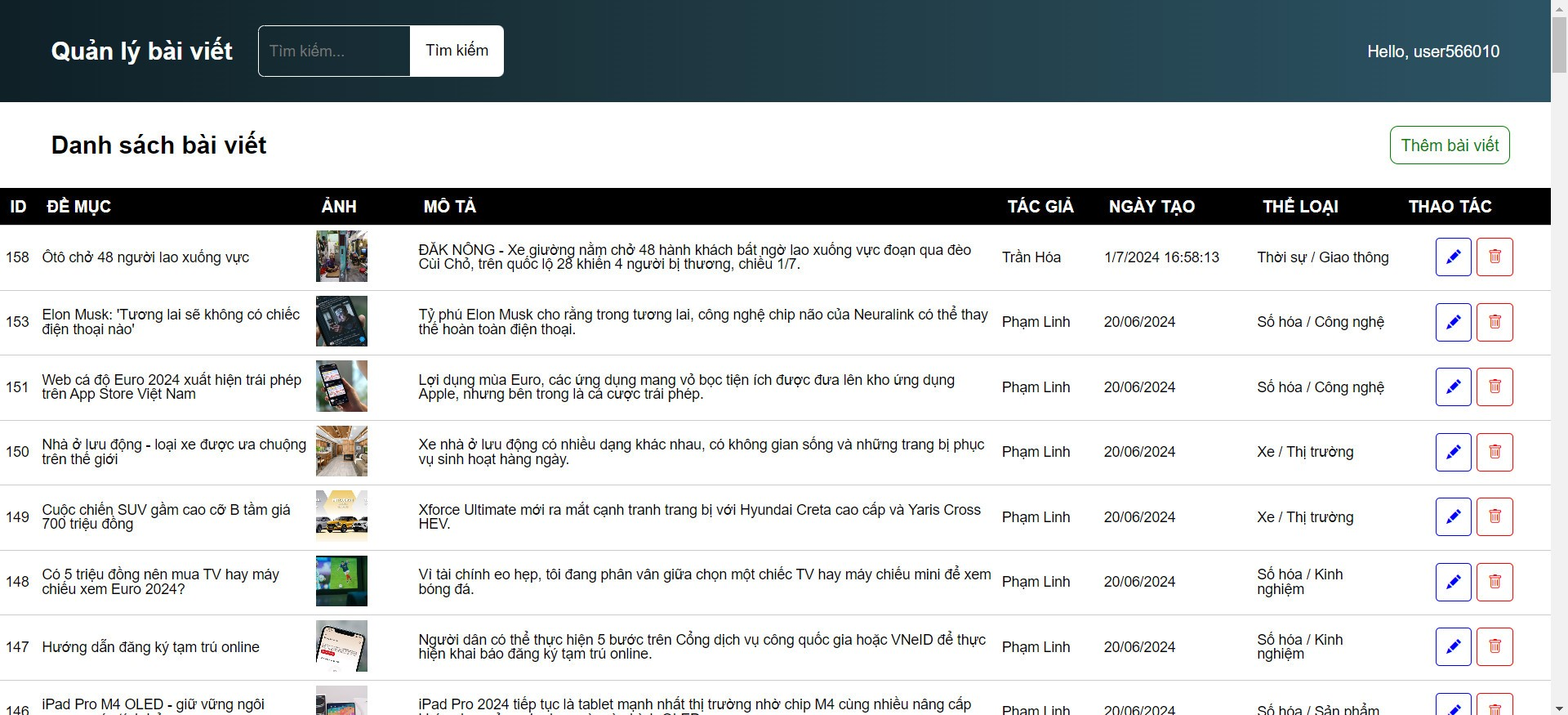


Hình 3.1.7: Trang thông tin cá nhân

**Giao diện trang quản lý bài viết**

Mối quan hệ:

1. Khi bấm vào tìm kiếm bài viết thì sẽ tìm kiếm theo từ khoá.
2. Khi bấm vào nút thêm bài viết sẽ chuyển tới trang thêm bài viết.
3. Khi bấm vào nút bút chì màu xanh sẽ chuyển tới trang sửa bài viết.
4. Khi bấm vào nút thùng rác màu đỏ thì thông báo hiện lên hỏi có chắc chắn muốn xoá bài viết hay không.

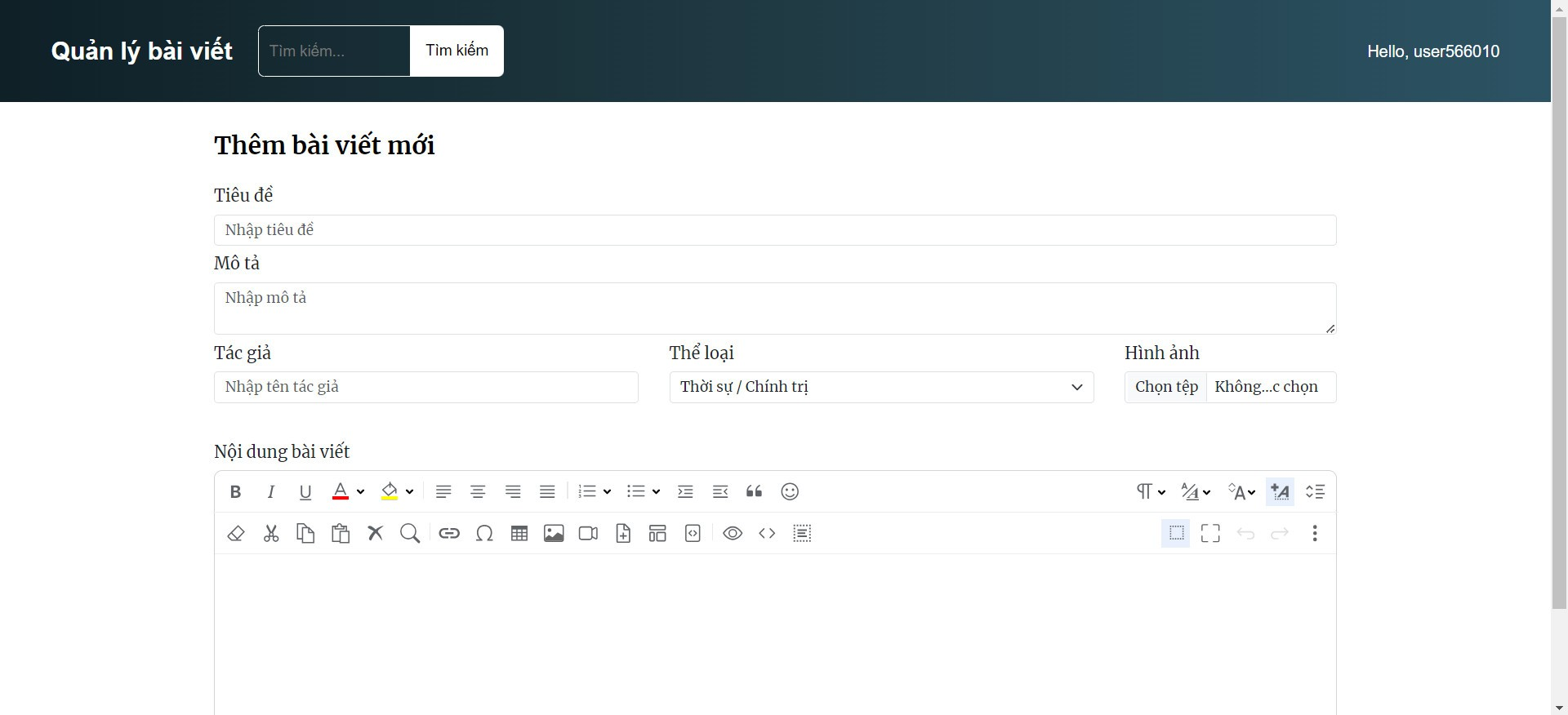


Hình 3.1.8: Trang quản lý bài viết

**Giao diện trang thêm bài viết**

Mối quan hệ:

1. Khi bấm vào tìm kiếm bài viết thì sẽ tìm kiếm theo từ khoá.
2. Khi bấm vào thêm bài viết thì sẽ thông báo thêm bài viết thành công và quay lại trang quản lý bài viết.

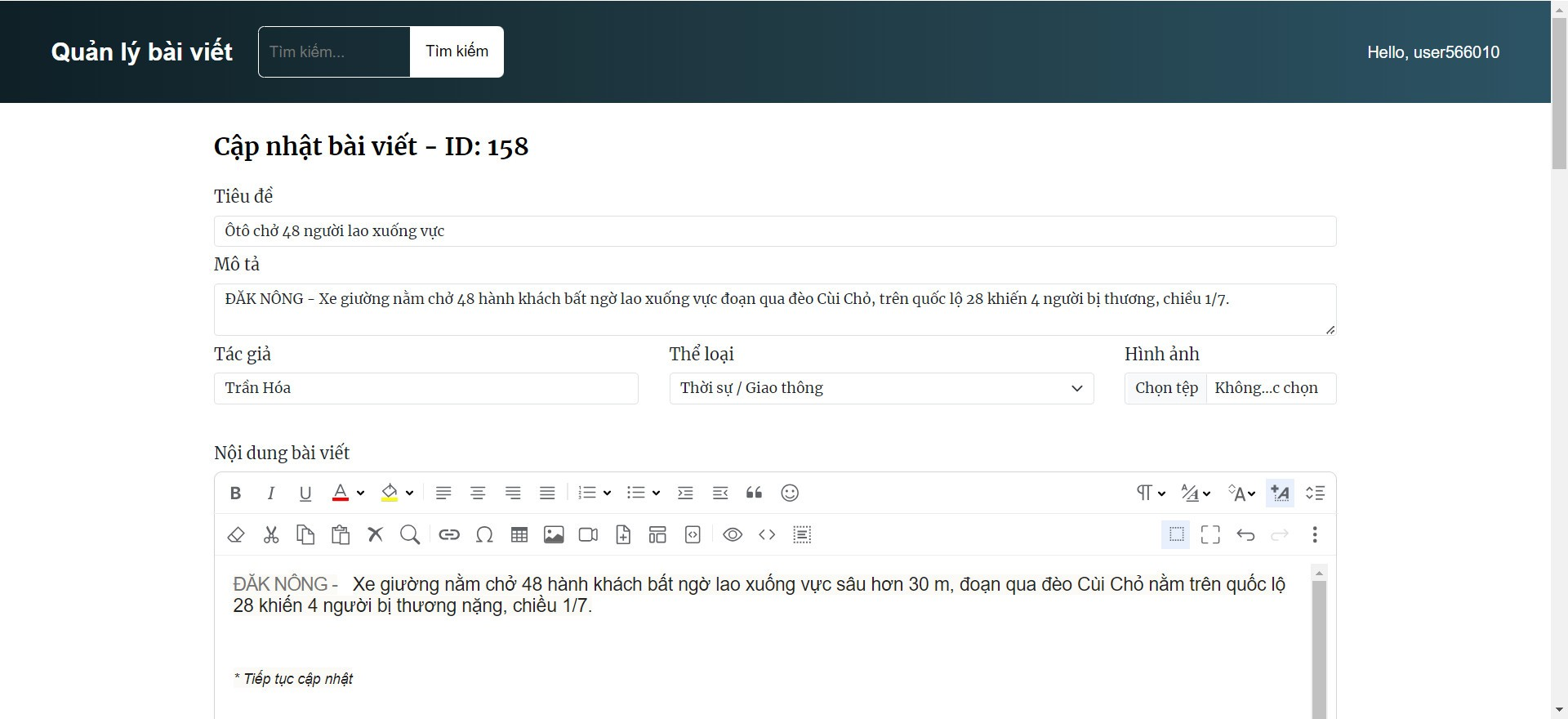


Hình 3.1.9: Trang thêm bài viết

**Giao diện trang cập nhật bài viết**

Mối quan hệ:

1. Khi bấm vào tìm kiếm bài viết thì sẽ tìm kiếm theo từ khoá.
2. Khi bấm vào cập nhật bài viết thì sẽ thông báo thêm bài viết thành công và quay lại trang quản lý bài viết.

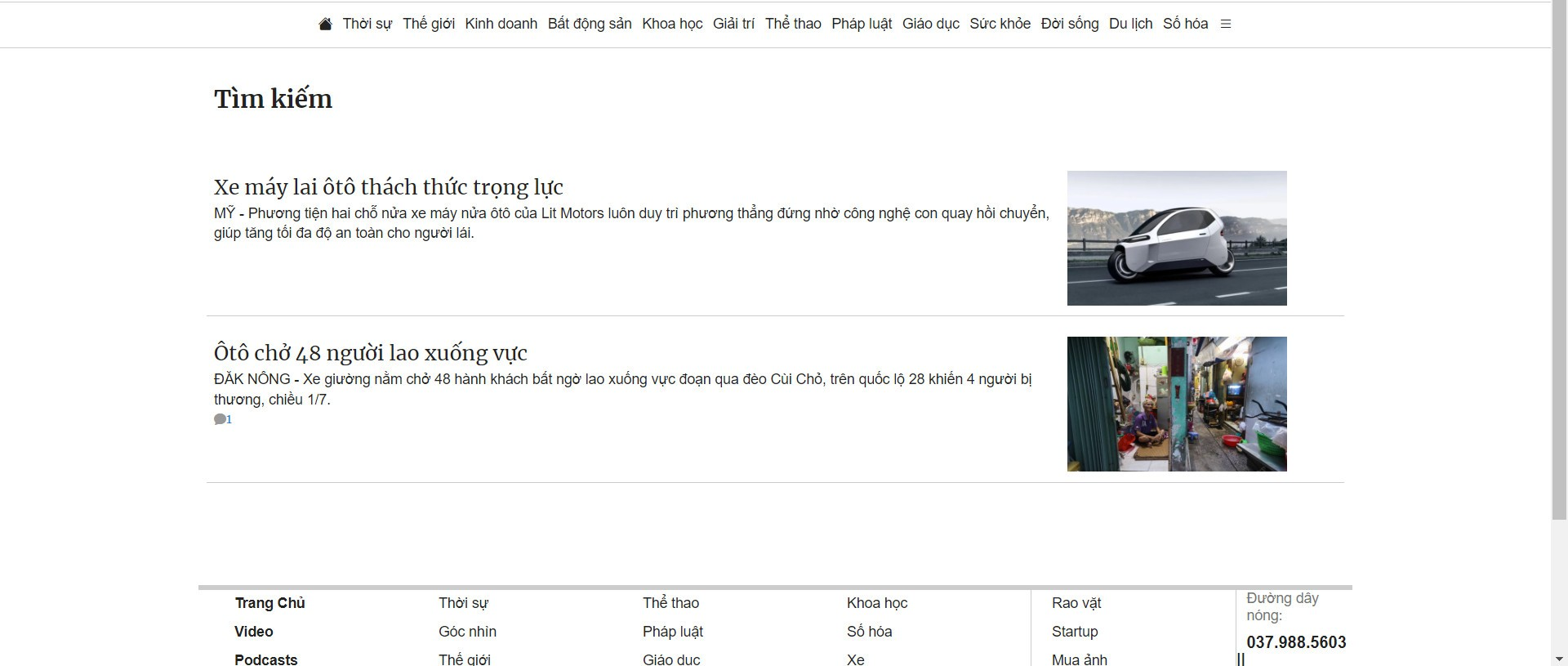


Hình 3.1.10: Trang cập nhật bài viết

**Giao diện trang tìm kiếm bài viết**

Mối quan hệ:

1. Khi bấm vào tìm kiếm bài viết thì sẽ tìm kiếm theo từ khoá.
2. Khi bấm vào bài viết bất kì đã được tìm kiếm thì sẽ hiển thị chi tiết bài viết đó.



Hình 3.1.11: Trang tìm kiếm bài viết

# 

# 3.2. Lớp BUS

**Bussiness Entities:**

**Lớp thực thể User**

A computer screen shot of a code

Description automatically generated

Hình 3.2.1: Lớp thực thể User

**Lớp thực thể BaiViet**

A screen shot of a computer code

Description automatically generated

Hình 3.2.2: Lớp thực thể BaiViet

**Lớp thực thể BaiVietSave**

A computer code with text

Description automatically generated

Hình 3.2.3: Lớp thực thể BaiVietSave

**Lớp thực thể BinhLuan**

A screen shot of a computer code

Description automatically generated

Hình 3.2.4: Lớp thực thể BinhLuan

**Lớp thực thể BinhLuanEmotion**

A computer screen shot of a computer code

Description automatically generated

Hình 3.2.5: Lớp thực thể BinhLuanEmotion

**Lớp thực thể Emotion**

A computer code with text

Description automatically generated

Hình 3.2.6: Lớp thực thể Emotion

**Lớp thực thể TheLoai**

A screen shot of a computer code

Description automatically generated

Hình 3.2.7: Lớp thực thể TheLoai

**Lớp thực thể TheLoaiTin**

A screenshot of a computer code

Description automatically generated

Hình 3.2.8: Lớp thực thể TheLoaiTin

**Service Interface:**

**IBaiViet**

**A computer code with text

Description automatically generated**

Hình 3.2.9: Interface IBaiViet

**IbinhLuan**

**A close up of text

Description automatically generated**

Hình 3.2.10: Interface IBinhLuan

**IEmotion**

**A computer screen shot of text

Description automatically generated**

Hình 3.2.11: Interface IEmotion

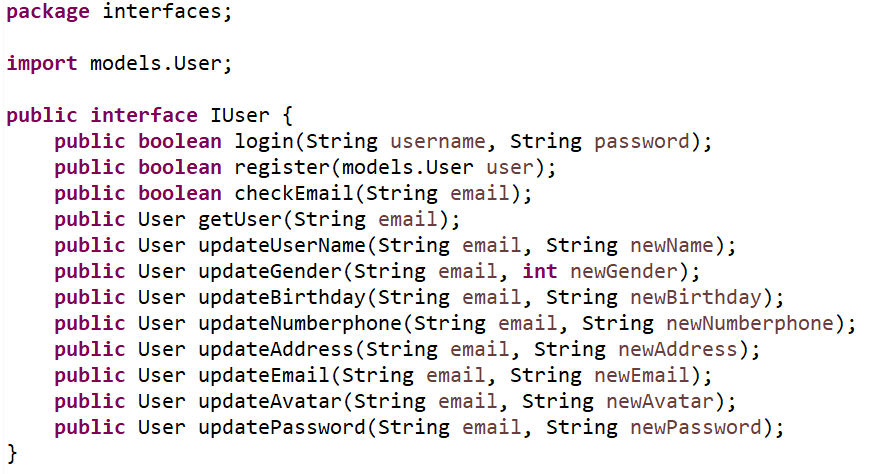
**ITheLoai**

**A computer screen shot of text

Description automatically generated**

Hình 3.2.12: Interface ITheLoai

**IUser**



Hình 3.2.13: Interface IUser

# 3.3. Lớp DAL.

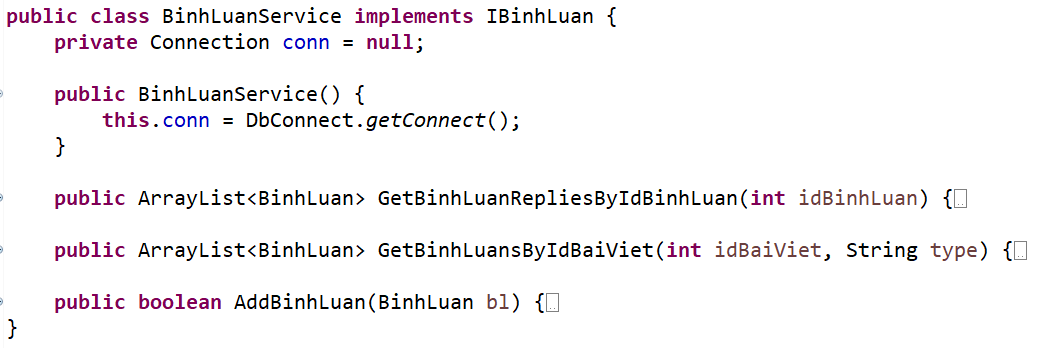
**Lớp BaiVietService:** Lớp này thực hiện các thao tác truy vấn cơ sở dữ liệu liên quan đến bảng bài viết.

A screenshot of a computer program

Description automatically generated

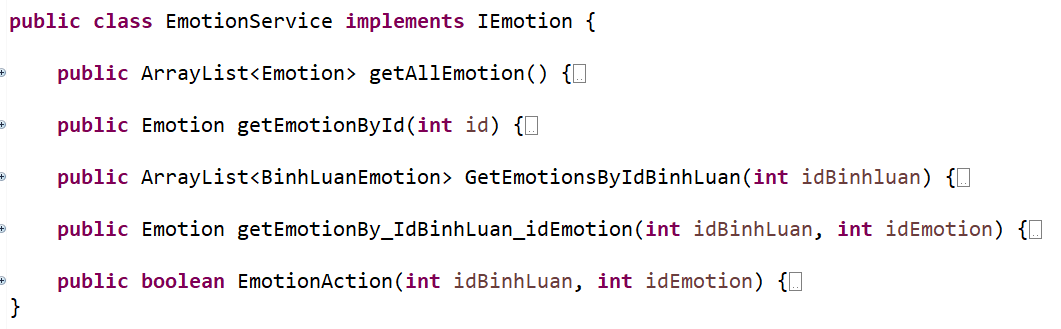
Hình 3.3.1: Lớp DAL BaiVietService

**Lớp BinhLuanService:** Lớp này thực hiện các thao tác truy vấn cơ sở dữ liệu liên quan đến bảng bình luận.



Hình 3.3.2: Lớp DAL BinhLuanService

**Lớp EmotionService:** Lớp này thực hiện các thao tác truy vấn cơ sở dữ liệu liên quan đến bảng emotion.



Hình 3.3.3: Lớp DAL EmotionService

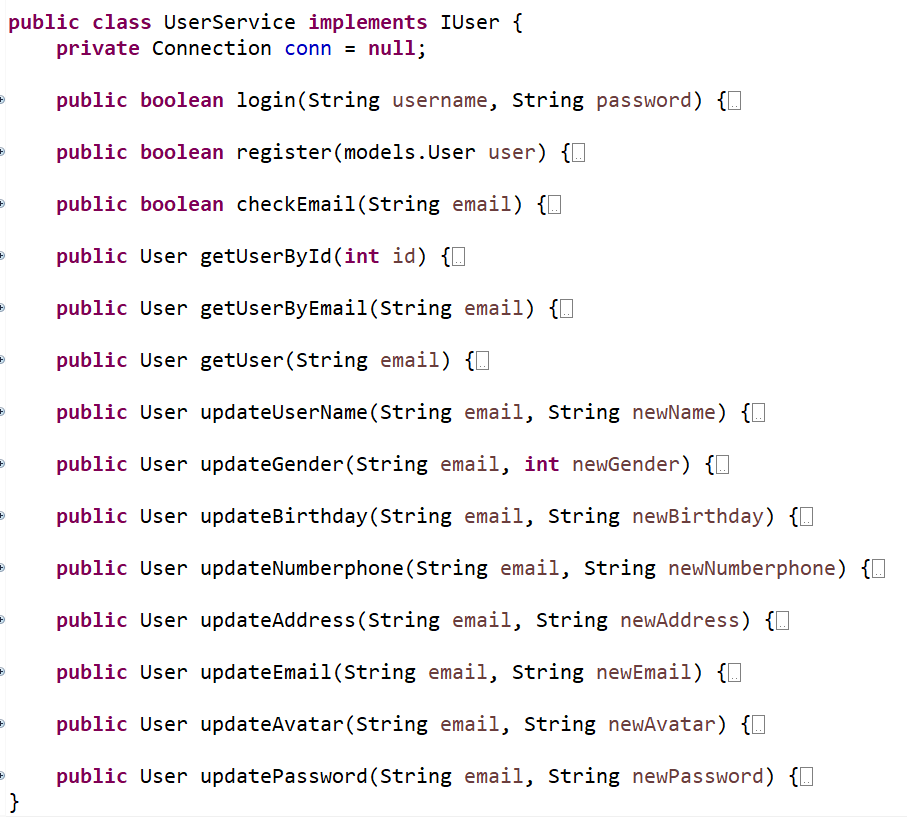
**Lớp TheLoaiService:** Lớp này thực hiện các thao tác truy vấn cơ sở dữ liệu liên quan đến bảng thể loại, thể loại tin.

A screenshot of a computer code

Description automatically generated

Hình 3.3.4: Lớp DAL TheLoaiService

**Lớp UserService:** Lớp này thực hiện các thao tác truy vấn cơ sở dữ liệu liên quan đến bảng người dùng.



Hình 3.3.5: Lớp DAL UserService

# 3.4 Kết chương.

Nhóm chúng em đã xây dựng và phát triển một giao diện người dùng thân thiện, dễ sử dụng cho trang web đọc tin tức. Người dùng có thể dễ dàng thực hiện đăng ký, đăng nhập, và tìm kiếm các bài viết theo từ khóa hoặc chủ đề mong muốn. Hệ thống quản lý bài viết dành cho quản trị viên được thiết kế linh hoạt, cho phép thêm mới, chỉnh sửa và xóa bài viết một cách hiệu quả.

Chức năng bình luận trên bài viết giúp người dùng có thể thảo luận và chia sẻ ý kiến, tạo ra một môi trường tương tác sôi động và phong phú. Những tính năng này đảm bảo mang lại sự thuận tiện và trải nghiệm tốt cho người dùng khi sử dụng trang web.

Những Điều Còn Thiếu:

Tuy nhiên, qua quá trình nghiệm thu, chúng em nhận thấy rằng còn một số khía cạnh cần cải thiện và bổ sung. Ví dụ như tăng tốc độ tải trang để nâng cao trải nghiệm người dùng, đảm bảo trang web hoạt động mượt mà và nhanh chóng hơn.

Chức năng quên mật khẩu, chức năng chia sẻ tin tức lên các mạng xã hội, đăng nhập bằng các tài khoản xã hội như Google và Facebook.

Phương hướng phát triển:

Chúng em cũng đang tiếp tục nỗ lực để cải thiện tính năng SEO, giúp trang web dễ dàng được tìm thấy trên các công cụ tìm kiếm.

Ngoài ra, chúng em nhận thấy cần phát triển các chiến dịch marketing để quảng bá trang web và thu hút người dùng mới. Việc nghiên cứu và áp dụng các chiến lược marketing phù hợp sẽ giúp tăng cường sự hiện diện trực tuyến và thu hút nhiều người đọc hơn.

Kết Luận:

Dù đã hoàn thành giai đoạn nghiệm thu, chúng em nhận ra rằng việc phát triển một trang web đọc tin tức là một quá trình không ngừng nghỉ. Luôn cần phải học hỏi, cải tiến và đáp ứng nhu cầu của người dùng. Với tinh thần đó, chúng em cam kết sẽ tiếp tục phát triển và nâng cao chất lượng trang web để mang lại trải nghiệm tốt nhất cho người dùng và cùng nhau phát triển bền vững trong tương lai.

# KẾT LUẬN

**Kết quả thu được:**

Tóm tắt kết quả đạt được: Trong quá trình này, chúng em đã thành công xây dựng một ứng dụng web đơn giản nhằm mục đích tra cứu tin tức. Trang web của chúng em cho phép người dùng nhập từ khóa để tìm kiếm và hiển thị các tin tức liên quan từ cơ sở dữ liệu.

Phân tích chức năng: Chúng em đã triển khai các chức năng cơ bản bao gồm:

* Giao diện người dùng thân thiện, cho phép người dùng dễ dàng nhập từ khóa tìm kiếm.
* Xử lý yêu cầu tìm kiếm bằng Servlet để truy xuất dữ liệu từ cơ sở dữ liệu.
* Hiển thị kết quả tìm kiếm một cách trực quan và dễ hiểu.

Thử nghiệm và xử lý lỗi: Chúng em đã tiến hành kiểm tra và thử nghiệm trang web để đảm bảo tính ổn định và hiệu quả. Chúng em cũng đã xử lý các lỗi cơ bản và cải thiện hiệu suất của trang web.

Kinh nghiệm học hỏi và cải tiến: Quá trình xây dựng ứng dụng này đã giúp chúng em nắm được kỹ năng cơ bản về Java Servlet và cách tích hợp nó vào một ứng dụng web. Chúng em đã học được cách thiết kế và triển khai các chức năng core của một ứng dụng web đơn giản.

Triển khai và hướng phát triển: Với những kinh nghiệm thu được, chúng em dự định tiếp tục phát triển ứng dụng này bằng cách cải thiện giao diện người dùng, tăng cường tính năng tìm kiếm và tích hợp các công nghệ mới nhằm cải thiện trải nghiệm người dùng.

**Hạn chế và hướng phát triển của đề tài:**

Hạn chế:

* Giao diện người dùng cơ bản: Đối với một ứng dụng tra cứu tin tức, giao diện người dùng có thể được cải thiện để trở nên hấp dẫn hơn và dễ sử dụng hơn cho người dùng cuối.
* Tính năng tìm kiếm cơ bản: Hiện tại, tính năng tìm kiếm có thể chỉ hỗ trợ tìm kiếm theo từ khóa đơn giản. Việc mở rộng để hỗ trợ tìm kiếm nâng cao như tìm kiếm theo thời gian, danh mục, hoặc sắp xếp kết quả sẽ là một điểm mạnh hơn.
* Bảo mật và xử lý lỗi: Vì là một ứng dụng đơn giản, việc xử lý lỗi và bảo mật dữ liệu có thể còn hạn chế. Các biện pháp bảo mật như xác thực người dùng, bảo vệ dữ liệu người dùng và xử lý lỗi tốt hơn có thể cần được cải thiện.

**Hướng phát triển:**

* Cải thiện giao diện người dùng: Tạo ra một giao diện người dùng thân thiện, có thể tương tác được trên nhiều thiết bị và hỗ trợ các tính năng như phân trang kết quả tìm kiếm, giao diện đáp ứng.
* Mở rộng tính năng tìm kiếm: Nâng cao tính năng tìm kiếm để người dùng có thể lọc kết quả theo nhiều tiêu chí khác nhau như thời gian, vị trí, độ phổ biến.
* Tối ưu hóa hiệu suất: Cải thiện hiệu suất của ứng dụng bằng cách tối ưu hóa truy vấn đến cơ sở dữ liệu, sử dụng bộ nhớ cache, và tối ưu hóa mã nguồn.
* Bảo mật và xử lý lỗi: Cải thiện bảo mật bằng cách áp dụng các biện pháp bảo mật như mã hóa dữ liệu, kiểm tra đầu vào và xử lý các lỗi tiềm ẩn một cách chặt chẽ hơn.
* Phát triển tích hợp và mở rộng: Tích hợp các tính năng xã hội, khả năng chia sẻ tin tức, tích hợp các dịch vụ ngoài (APIs), và mở rộng ứng dụng để hỗ trợ nhiều ngôn ngữ, nền tảng khác nhau.

# DANH MỤC SÁCH THAM KHẢO

[1]. Nguyễn Văn Ba (2005), Phân Tích Và Thiết Kế Hệ Thống Thông Tin , NXB Đại Học Quốc Gia Hà Nội 2005.

[2]. Joshua Bloch (2001), Effective Java, ‎ Addison -Wesley Professional .

[3]. https://www.w3schools.com/