

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP

ĐỀ TÀI

XÂY DỰNG WEBSITE CHIA SẺ TÀI LIỆU

| | |
|----------------------|-----------------------|
| Giảng viên hướng dẫn | : TS. Hoàng Văn Thông |
| Sinh viên thực hiện | : Nguyễn Việt Anh |
| Mã sinh viên | : 201200016 |
| Lớp | : CNTT1 |
| Khóa | : K61 |

Hà Nội – 2024

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP

ĐỀ TÀI

XÂY DỰNG WEBSITE CHIA SẺ TÀI LIỆU

| | |
|----------------------|-----------------------|
| Giảng viên hướng dẫn | : TS. Hoàng Văn Thông |
| Sinh viên thực hiện | : Nguyễn Việt Anh |
| Mã sinh viên | : 201200016 |
| Lớp | : CNTT1 |
| Khóa | : K61 |

Hà Nội – 2024

LỜI CẢM ƠN

Để hoàn thành đồ án, trước hết em xin gửi lời cảm ơn chân thành và sâu sắc đến thầy Hoàng Văn Thông, người đã tận tình hướng dẫn em trong suốt quá trình nghiên cứu vừa qua. Trong thời gian được thầy hướng dẫn, em không những tiếp thu thêm nhiều kiến thức bổ ích mà còn học tập được tinh thần, thái độ làm việc nghiêm túc, hiệu quả. Đây là điều rất cần thiết cho em trong quá trình học tập và làm việc sau này.

Em chân thành cảm ơn các thầy cô trong khoa Công nghệ thông tin, Đại học Giao thông vận tải đã tâm huyết dạy dỗ, truyền đạt những kiến thức quý báu cho em trong suốt 4 năm học đại học. Những kiến thức đó không chỉ giúp em hoàn thành đồ án tốt nghiệp này mà còn là thứ hành trang quý báu để em có thể tự tin trên quá trình theo đuổi thành công của mình.

Em cũng gửi lời cảm ơn đến tất cả các anh chị trong khoá trước cũng như các bạn trong cùng lớp và khoa đã luôn đồng hành và hỗ trợ em trong suốt quãng thời gian học tập. Sự chia sẻ kiến thức, kinh nghiệm, sự đồng cảm và sự hỗ trợ từ mọi người không chỉ giúp em vượt qua những thách thức trong học tập mà còn là nguồn động viên và động lực mạnh mẽ để em tiếp tục phấn đấu. Em rất biết ơn và trân trọng mọi đóng góp và sự hỗ trợ từ tất cả mọi người.

Hơn nữa, em xin cảm ơn gia đình, bạn bè, người thân và đặc biệt là tập thể CNTT1_K61 đã hỗ trợ em hết mình trong những năm tháng sinh viên.

Với điều kiện thời gian cũng như kinh nghiệm còn hạn chế, đồ án tốt nghiệp này không thể tránh được những thiếu sót. Em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của các thầy cô để em có điều kiện bổ sung, nâng cao ý thức của mình, phục vụ tốt hơn cho chuyên ngành của bản thân mình.

Em xin chân thành cảm ơn!

SINH VIÊN THỰC HIỆN ĐỒ ÁN

Nguyễn Việt Anh

MỤC LỤC

| | |
|--|-----------|
| LỜI CẢM ƠN..... | 3 |
| DANH MỤC CÁC TỪ VIẾT TẮT | 6 |
| DANH MỤC BẢNG BIÊU | 7 |
| DANH MỤC HÌNH ẢNH | 8 |
| MỞ ĐẦU | 11 |
| CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI VÀ CÁC KIẾN THỨC LIÊN QUAN | 13 |
| 1.1. Tìm hiểu về giao thức HTTP. | 13 |
| 1.1.1. HTTP là gì ? | 13 |
| 1.1.2. Sơ đồ hoạt động của HTTP | 13 |
| 1.1.3. Uniform Resource Locator (URL) | 13 |
| 1.1.4. Các thành phần chính của HTTP | 13 |
| 1.2. Tổng quan về UML..... | 15 |
| 1.2.1. UML là gì ? | 15 |
| 1.2.2. Một số loại sơ đồ UML | 15 |
| 1.3. Tìm hiểu tổng quan hệ thống cơ sở dữ liệu và hệ quản trị cơ sở dữ liệu..... | 18 |
| 1.3.1. Khái niệm dữ liệu và cơ sở dữ liệu. | 18 |
| 1.3.2. Các mô hình cơ sở dữ liệu..... | 19 |
| 1.3.3. Hệ quản trị cơ sở dữ liệu là gì ? | 20 |
| CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH THIẾT KẾ HỆ THỐNG | 21 |
| 2.1. Khảo sát..... | 21 |
| 2.1.1. Khảo sát website tương tự..... | 21 |
| 2.1.2. Phân tích yêu cầu..... | 24 |
| 2.2. Phân tích và thiết kế hệ thống | 26 |
| 2.2.1. Biểu đồ use case | 26 |
| 2.2.2. Biểu đồ hành động | 39 |
| 2.2.3. Sơ đồ tuần tự chức năng..... | 45 |
| 2.2.4. Phân tích thiết kế cơ sở dữ liệu | 50 |

| | |
|--|----|
| CHƯƠNG 3. CÀI ĐẶT VÀ THỬ NGHIỆM WEBSITE | 58 |
| 3.1. Các công nghệ sử dụng. | 58 |
| 3.1.1. Giới thiệu về NodeJS. | 58 |
| 3.1.2. Giới thiệu về ExpressJS (Express Framework). | 60 |
| 3.1.3. Ngôn ngữ lập trình Javascript | 61 |
| 3.1.4. Giới thiệu về Bootstrap. | 61 |
| 3.1.5. HTML và CSS..... | 62 |
| 3.1.6. Hệ quản trị dữ liệu MySQL | 63 |
| 3.2. Xây dựng chương trình | 64 |
| 3.2.1. Giao diện người dùng..... | 64 |
| 3.2.2. Giao diện người quản trị viên | 72 |
| KẾT LUẬN VÀ KIẾN NGHỊ..... | 78 |
| TÀI LIỆU THAM KHẢO | 79 |

DANH MỤC CÁC TỪ VIẾT TẮT

| Thuật ngữ/Từ viết tắt | Ý nghĩa |
|------------------------------|---------------------------------------|
| API | Application Programming Interface |
| CPU | Central Processing Unit |
| CSDL | Cơ sở dữ liệu |
| CSS | Cascading Style Sheets |
| RDBMS | Relational Database Management System |
| DNS | Domain Name System |
| FK | Foreign Key |
| HTML | HyperText Markup Language |
| HTTP | Hypertext Transfer Protocol |
| I/O | Input/Output |
| JS | Javascript |
| MVC | Model - View – Controller |
| NPM | Node Package Manager |
| PK | Primary Key |
| SD | Sequence diagram |
| SQL | Structured Query Language |
| TCP | Transmission Control Protocol |
| UML | Unified Modeling Language |
| URL | Uniform Resource Locator |
| WWW | World Wide Web |

DANH MỤC BẢNG BIỂU

| | |
|--|----|
| Bảng 1.1. Bảng giải thích ký hiệu dùng để vẽ trong use-case..... | 15 |
| Bảng 1.2. Bảng giải thích ký hiệu dùng để vẽ trong Activity Diagram. | 18 |
| Bảng 2.1. Bảng các tác nhân. | 26 |
| Bảng 2.2. Bảng các ca sử dụng. | 26 |
| Bảng 2.3. Bảng mô tả use case đăng nhập. | 28 |
| Bảng 2.4. Bảng mô tả use case đăng ký. | 29 |
| Bảng 2.5. Bảng mô tả use case cập nhật tài khoản. | 30 |
| Bảng 2.6. Bảng mô tả use case đăng tài liệu/bài viết. | 31 |
| Bảng 2.7. Bảng mô tả use case yêu thích. | 32 |
| Bảng 2.8. Bảng mô tả use case bình luận. | 33 |
| Bảng 2.9. Bảng mô tả use case follow. | 34 |
| Bảng 2.10. Bảng mô tả use case quản lý người dùng. | 35 |
| Bảng 2.11. Bảng mô tả use case quản lý tài liệu. | 36 |
| Bảng 2.12. Bảng mô tả use case quản lý bài viết. | 37 |
| Bảng 2.13. Bảng mô tả use case thông kê. | 38 |
| Bảng 2.14. Bảng các thực thể. | 50 |
| Bảng 2.15. Bảng mô tả chi tiết bảng Tag. | 51 |
| Bảng 2.16. Bảng mô tả chi tiết bảng Types. | 52 |
| Bảng 2.17. Bảng mô tả chi tiết bảng Users. | 52 |
| Bảng 2.18. Bảng mô tả chi tiết bảng Posts. | 53 |
| Bảng 2.19. Bảng mô tả chi tiết bảng tags_post | 53 |
| Bảng 2.20. Bảng mô tả bảng Document..... | 54 |
| Bảng 2.21. Bảng mô tả bảng Comments. | 55 |
| Bảng 2.22. Bảng mô tả chi tiết bảng Likes. | 56 |
| Bảng 2.23. Bảng mô tả chi tiết bảng Notification. | 56 |
| Bảng 2.24. Bảng mô tả bảng follow. | 57 |
| Bảng 2.25. Bảng mô tả bảng report..... | 57 |

DANH MỤC HÌNH ẢNH

| | |
|---|----|
| Hình 1.1. Hình ví dụ về sơ đồ trình tự | 17 |
| Hình 2.1. Hình trang chủ website Slideshare..... | 21 |
| Hình 2.2. Giao diện trang tìm kiếm website SlideShare..... | 22 |
| Hình 2.3. Giao diện trang thông tin cá nhân website SlideShare..... | 22 |
| Hình 2.4. Giao diện trang upload tài liệu website SlideShare. | 23 |
| Hình 2.5. Giao diện trang chi tiết tài liệu website SlideShare. | 23 |
| Hình 2.6. Sơ đồ use case tổng quát. | 27 |
| Hình 2.7. Sơ đồ use case đăng nhập..... | 28 |
| Hình 2.8. Sơ đồ use case đăng ký. | 29 |
| Hình 2.9. Sơ đồ use case cập nhật tài khoản. | 30 |
| Hình 2.10. Sơ đồ use case yêu thích. | 31 |
| Hình 2.11. Sơ đồ use case bình luận. | 32 |
| Hình 2.12. Sơ đồ use case theo dõi. | 33 |
| Hình 2.13. Sơ đồ use case quản lý người dùng. | 34 |
| Hình 2.14. Sơ đồ use case quản lý tài liệu. | 36 |
| Hình 2.15. Sơ đồ use case quản lý bài viết. | 37 |
| Hình 2.16. Sơ đồ use case thống kê. | 38 |
| Hình 2.17. Biểu đồ hành động chức năng đăng nhập. | 39 |
| Hình 2.18. Biểu đồ hành động chức năng đăng ký tài khoản. | 40 |
| Hình 2.19. Biểu đồ hành động chức năng cập nhật tài khoản..... | 40 |
| Hình 2.20. Biểu đồ hành động chức năng tìm kiếm. | 41 |
| Hình 2.21. Biểu đồ hành động chức năng thêm mới bình luận. | 41 |
| Hình 2.22. Biểu đồ hành động chức năng yêu thích. | 42 |
| Hình 2.23. Biểu đồ hành động chức năng thêm mới tài liệu. | 42 |
| Hình 2.24. Biểu đồ hành động chức năng cập nhật tài liệu. | 43 |
| Hình 2.25. Biểu đồ hành động chức năng thêm mới bài viết..... | 43 |
| Hình 2.26. Biểu đồ hành động chức năng cập nhật bài viết. | 44 |

| | |
|---|----|
| Hình 2.27. Biểu đồ hành động chức năng theo dõi..... | 44 |
| Hình 2.28. Biểu đồ hành động chức năng thông kê..... | 45 |
| Hình 2.29. Sơ đồ tuần tự chức năng đăng nhập..... | 45 |
| Hình 2.30. Sơ đồ tuần tự chức năng đăng ký..... | 46 |
| Hình 2.31. Sơ đồ tuần tự chức năng quên mật khẩu..... | 46 |
| Hình 2.32. Sơ đồ tuần tự chức năng cập nhật thông tin tài khoản..... | 47 |
| Hình 2.33. Sơ đồ tuần tự chức năng tìm kiếm..... | 47 |
| Hình 2.34. Sơ đồ tuần tự chức năng đăng tài liệu..... | 48 |
| Hình 2.35. Sơ đồ tuần tự chức năng đăng bài viết..... | 48 |
| Hình 2.36. Sơ đồ tuần tự chức năng bình luận..... | 49 |
| Hình 2.37. Sơ đồ tuần tự chức năng yêu thích..... | 49 |
| Hình 2.38. Sơ đồ ERD..... | 51 |
| Hình 3.1. Giao diện đăng nhập..... | 64 |
| Hình 3.2. Giao diện trang đăng ký tài khoản..... | 65 |
| Hình 3.3. Giao diện trang quên mật khẩu..... | 65 |
| Hình 3.4. Mật khẩu mới gửi về email..... | 66 |
| Hình 3.5. Giao diện trang thông tin cá nhân..... | 66 |
| Hình 3.6. Giao diện chức năng đổi mật khẩu..... | 67 |
| Hình 3.7. Giao diện trang chủ..... | 68 |
| Hình 3.8. Giao diện trang upload tài liệu..... | 69 |
| Hình 3.9. Giao diện trang đăng bài viết..... | 69 |
| Hình 3.10. Giao diện trang chi tiết tài liệu..... | 70 |
| Hình 3.11. Hình gửi mail sau khi download tài liệu..... | 70 |
| Hình 3.12. Giao diện trang chi tiết bài viết..... | 71 |
| Hình 3.13. Giao diện trang thông tin người dùng khác..... | 71 |
| Hình 3.14. Danh sách thông báo..... | 72 |
| Hình 3.15. Giao diện trang thống kê (1)..... | 73 |
| Hình 3.16. Giao diện trang thống kê (2)..... | 73 |
| Hình 3.17. Giao diện trang quản lý người dùng..... | 74 |

| | |
|---|----|
| Hình 3.18. Giao diện trang quản lý tài liệu..... | 74 |
| Hình 3.19. Giao diện trang quản lý bài viết..... | 75 |
| Hình 3.20. Giao diện trang quản lý bình luận người dùng..... | 75 |
| Hình 3.21. Giao diện trang quản lý danh mục tài liệu..... | 76 |
| Hình 3.22. Giao diện trang quản lý tag bài viết..... | 76 |
| Hình 3.23. Giao diện trang quản lý báo cáo vi phạm..... | 77 |
| Hình 3.24. Yêu cầu cung cấp bằng chứng vi phạm..... | 77 |

MỞ ĐẦU

Trong thời đại ngày nay, công nghệ thông tin đang được phát triển một cách mạnh mẽ với một tốc độ chóng mặt. Quá trình chuyển đổi số đã và đang diễn ra khắp nơi trên toàn thế giới. Công nghệ thông tin được áp dụng vào rất nhiều lĩnh vực để tối ưu hóa hiệu suất và đáp ứng các nhu cầu ngày càng cao của con người. Trong thời đại công nghệ số bùng nổ, việc tiếp cận và chia sẻ thông tin trở nên dễ dàng hơn bao giờ hết. Nhu cầu tìm kiếm tài liệu học tập, nghiên cứu và tham khảo ngày càng tăng cao, đặc biệt là trong môi trường giáo dục và học thuật. Tuy nhiên, việc tìm kiếm tài liệu chất lượng và phù hợp vẫn còn nhiều khó khăn do nguồn tài liệu phân tán, thiếu hệ thống và cập nhật chưa thường xuyên.

Các website chia sẻ tài liệu hoạt động theo một cơ chế rõ ràng. Người dùng có thể đăng tải tài liệu của mình lên website, các tài liệu sau đó được phân loại theo chủ đề, lĩnh vực và định dạng. Việc tìm kiếm tài liệu được thực hiện dựa trên từ khóa, tên tác giả hoặc nội dung cụ thể. Người dùng có thể tải về tài liệu miễn phí hoặc trả phí, và có thể đánh giá, bình luận về chất lượng tài liệu. Ngoài ra, các website này cũng quản lý tài liệu bằng cách gỡ bỏ những tài liệu vi phạm bản quyền hoặc có nội dung độc hại.

Các website chia sẻ tài liệu mang đến nhiều lợi ích thiết thực. Đầu tiên, người dùng có thể tiếp cận kho tàng tri thức miễn phí hoặc với chi phí thấp, tiết kiệm đáng kể chi phí mua sắm sách vở và tài liệu. Nhờ vào tài liệu đa dạng và phong phú, việc học tập trở nên hiệu quả hơn, giúp cung cấp kiến thức và kỹ năng một cách toàn diện. Ngoài ra, việc tìm kiếm tài liệu trên các trang web này rất dễ dàng và nhanh chóng, tiết kiệm thời gian cho người dùng. Hơn nữa, các trang web này còn là nơi giao lưu, chia sẻ kinh nghiệm và hỗ trợ lẫn nhau trong cộng đồng học tập, tạo môi trường cho người học giao lưu, chia sẻ kinh nghiệm học tập và hỗ trợ lẫn nhau. Nhờ đó, người học có thể học tập hiệu quả hơn và xây dựng cộng đồng học tập đoàn kết.

Bên cạnh những lợi ích, các website chia sẻ tài liệu cũng tồn tại nhiều hạn chế. Chất lượng tài liệu không đồng đều, có nguy cơ tiếp cận thông tin sai lệch hoặc thiếu chính xác. Vi phạm bản quyền là một vấn đề nhức nhối, khi việc sử dụng tài liệu trái phép ảnh hưởng đến quyền lợi của tác giả. Quản lý nội dung trên các trang web này chưa hiệu quả, dẫn đến nguy cơ xuất hiện nội dung độc hại, vi phạm đạo đức. Ngoài ra, người dùng còn phải đối mặt với nguy cơ mất an toàn thông tin, bị virus hoặc mã độc tấn công.

Các website chia sẻ tài liệu là công cụ hữu ích cho học tập và nghiên cứu. Tuy nhiên, việc sử dụng thông minh và an toàn là điều vô cùng quan trọng để tránh những tác hại tiềm ẩn. Nhận thức được vấn đề này, em đã chọn đề tài "Xây dựng website chia sẻ tài liệu" với mục đích tạo dựng một nền tảng trực tuyến giúp người dùng dễ dàng tìm

kiếm, chia sẻ và quản lý tài liệu một cách hiệu quả. Em mong muốn tạo ra một nền tảng đáng tin cậy, nơi người dùng có thể yên tâm truy cập và sử dụng tài liệu một cách hiệu quả. Đồng thời, em cũng hy vọng website này sẽ góp phần xây dựng một cộng đồng học tập tích cực, nơi mọi người có thể giao lưu, chia sẻ kiến thức và hỗ trợ lẫn nhau trong quá trình học tập và nghiên cứu, thúc đẩy nghiên cứu khoa học, phát triển cộng đồng và khuyến khích tinh thần tự học.

Nội dung đồ án gồm 3 chương với nội dung tóm tắt như sau:

Chương 1: Các kiến thức liên quan.

Chương 2: Khảo sát và phân tích thiết kế hệ thống.

Chương 3: Cài đặt và thử nghiệm hệ thống.

CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI VÀ CÁC KIẾN THỨC LIÊN QUAN

1.1. Tìm hiểu về giao thức HTTP.

1.1.1. HTTP là gì ?

HTTP (HyperText Transfer Protocol - Giao thức truyền tải siêu văn bản) là một trong các giao thức chuẩn về mạng Internet, được dùng để liên hệ thông tin giữa Máy cung cấp dịch vụ (Web server) và Máy sử dụng dịch vụ (Web client), là giao thức Client/Server dùng cho World Wide Web – WWW.

HTTP là một giao thức ứng dụng của bộ giao thức TCP/IP (các giao thức nền tảng cho Internet).

1.1.2. Sơ đồ hoạt động của HTTP

HTTP hoạt động dựa trên mô hình Client – Server. Trong mô hình này, các máy tính của người dùng sẽ đóng vai trò làm máy khách (Client). Sau một thao tác nào đó của người dùng, các máy khách sẽ gửi yêu cầu đến máy chủ (Server) và chờ đợi câu trả lời từ những máy chủ này.

HTTP là một stateless protocol. Hay nói cách khác, request hiện tại không biết những gì đã hoàn thành trong request trước đó.

HTTP cho phép tạo các yêu cầu gửi và nhận các kiểu dữ liệu, do đó cho phép xây dựng hệ thống độc lập với dữ liệu được chuyển giao.

1.1.3. Uniform Resource Locator (URL)

Một URL (Uniform Resource Locator) được sử dụng để xác định duy nhất một tài nguyên trên Web. Một URL có cấu trúc như sau:

“protocol://hostname:port/path-and-file-name”

Trong một URL có 4 thành phần:

- Protocol: giao thức tầng ứng dụng được sử dụng bởi client và server
- Hostname: tên DNS domain
- Port: Cổng TCP để server lắng nghe request từ client
- Path-and-file-name: Tên và vị trí của tài nguyên yêu cầu.

1.1.4. Các thành phần chính của HTTP

- **HTTP – Requests:** HTTP Request Method: Là phương thức để chỉ ra hành động mong muốn được thực hiện trên tài nguyên đã xác định. Cấu trúc của một HTTP Request:

- Một Request-line = Phương thức + URI–Request + Phiên bản HTTP . Giao thức HTTP định nghĩa một tập các giao thức GET, POST, HEAD, PUT ... Client có thể sử dụng một trong các phương thức đó để gửi request lên server.
- Có thể có hoặc không các trường header.
- Một dòng trống để đánh dấu sự kết thúc của các trường Header.
- Request Header Fields: Các trường header cho phép client truyền thông tin bổ sung về yêu cầu, và về chính client, đến server. Một số trường: Accept-Charset, Accept-Encoding, Accept-Language, Authorization, Expect, From, Host, ...

Khi request đến server, server thực hiện một trong 3 hành động sau:

- Server phân tích request nhận được, maps yêu cầu với tập tin trong tập tài liệu của server, và trả lại tập tin yêu cầu cho client.
- Server phân tích request nhận được, maps yêu cầu vào một chương trình trên server, thực thi chương trình và trả lại kết quả của chương trình đó.
- Request từ client không thể đáp ứng, server trả lại thông báo lỗi.

HTTP định nghĩa một số phương thức cho việc trao đổi dữ liệu, bao gồm:

- GET: Truy vấn dữ liệu từ máy chủ.
- POST: Gửi dữ liệu đến máy chủ để xử lý.
- PUT: Cập nhật dữ liệu trên máy chủ.
- DELETE: Xóa dữ liệu trên máy chủ.
- Và nhiều phương thức khác như HEAD, OPTIONS, PATCH, v.v.

HTTP Response: HTTP Response được gọi là “thông báo phản hồi HTTP“. Đây là kết quả server trả về cho client.

Cấu trúc HTTP Response gần giống với HTTP request. Điểm khác biệt là thay vì Request-Line, HTTP có response có Status-Line.

Tương tự như Request-Line, Status-Line cũng có ba phần như sau:

- **HTTP-version:** phiên bản HTTP cao nhất mà server hỗ trợ.
- **Status-Code:** mã kết quả trả về.

- **Reason-Phrase:** mô tả về Status-Code.

Ý nghĩa một số Status-Code: Yếu tố Status-Code là một số nguyên 3 ký tự. Ký tự đầu tiên của mã hóa trạng thái định nghĩa hạng (loại) phản hồi và hai ký tự cuối không có bất cứ vai trò phân loại nào.

Có 5 giá trị của ký tự đầu tiên:

- **1xx:** Thông tin. Mã này nghĩa là yêu cầu đã được nhận và tiến trình đang tiếp tục.
- **2xx:** Thành công. Mã này nghĩa là hoạt động đã được nhận, được hiểu, và được chấp nhận một cách thành công.
- **3xx:** Sự điều hướng lại. Mã này nghĩa là hoạt động phải được thực hiện để hoàn thành yêu cầu.
- **4xx:** Lỗi Client. Mã này nghĩa là yêu cầu chứa cú pháp không chính xác hoặc không được thực hiện.
- **5xx:** Lỗi Server. Mã này nghĩa là Server thất bại với việc thực hiện một yêu cầu nhìn như có vẻ khả thi.

1.2. Tổng quan về UML.

1.2.1. UML là gì ?

Ngôn ngữ mô hình hóa thống nhất (Unified Modeling Language - UML) là một ngôn ngữ mô hình gồm các ký hiệu đồ họa mà các phương pháp hướng đối tượng sử dụng để thiết kế các hệ thống thông tin một cách nhanh chóng.

1.2.2. Một số loại sơ đồ UML

a. **Biểu đồ Use case (Use Case Diagram):** là một biểu đồ mô tả về các tương tác có thể xảy ra giữa người dùng (user) và hệ thống (system). Biểu đồ này bao gồm các use case cũng như các loại người dùng khác nhau của hệ thống và thường đi kèm với một số loại biểu đồ khác. Các use case được thể hiện bằng hình tròn hoặc hình elip, còn tác nhân (actor) thường được biểu diễn dưới dạng hình người.

Bảng 1.1. Bảng giải thích ký hiệu dùng để vẽ trong use-case.

| Ký hiệu | Mô tả |
|--|--|
|  Actor | Biểu diễn các tác nhân (actor) có thể là người dùng hay hệ thống khác tương tác với hệ thống hiện tại. |

| | |
|--|---|
| | Biểu diễn các ca sử dụng (use-case) thể hiện các hành động mà hệ thống thực hiện. |
| | Biểu diễn mối quan hệ giữa actor và use-case. |
| | Quan hệ <<include>> thể hiện quan hệ bao hàm giữa hai use-case. |
| | Quan hệ <<extend>> thể hiện hệ khái quát mở rộng của hai use-case. |

b. Biểu đồ trình tự (Sequence diagram – SD):

Được sử dụng để xác định và chỉ rõ vai trò của các đối tượng tham gia vào luồng sự kiện của use case.

Là một loại biểu đồ tương tác, mô tả mô hình tương tác giữa các đối tượng, trong đó nhấn mạnh vào trình tự thời gian của các thông điệp trao đổi giữa các đối tượng đó.

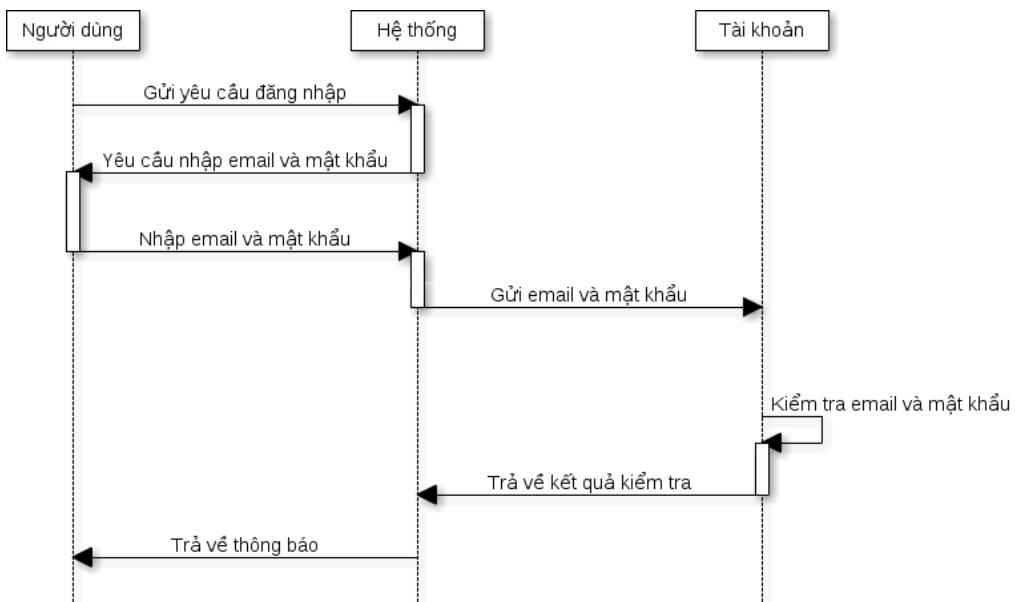
Biểu đồ trình tự chỉ ra:

- Các đối tượng tham gia vào tương tác.
- Thời gian sống của các đối tượng .
- Trình tự các thông điệp được trao đổi.

Các thành phần của biểu đồ tuần tự:

- Đối tượng (object hoặc class): biểu diễn bằng các hình chữ nhật
- Đường đời đối tượng (Lifelines): biểu diễn bằng các đường gạch rời thẳng đứng bên dưới các đối tượng
- Thông điệp (Message): biểu diễn bằng các đường mũi tên.
- Xử lý bên trong đối tượng (biểu diễn bằng các đoạn hình chữ nhật rỗng nối với các đường đời đối tượng).

Ví dụ:



Hình 1.1. Hình ví dụ về sơ đồ trình tự

Trong sơ đồ trên có 3 đối tượng là : người dùng, hệ thống và tài khoản. Luồng xử lý của chức năng đăng nhập có thể diễn giải như sau:

- Người dùng gửi yêu cầu đăng nhập đến hệ thống.
- Hệ thống yêu cầu người dùng nhập email và mật khẩu.
- Người dùng nhập email và mật khẩu.
- Hệ thống gửi email và mật khẩu của người dùng để kiểm tra.
- Tài khoản kiểm tra thông tin email và password có đúng hay không.
- Tài khoản trả về kết quả kiểm tra cho hệ thống.
- Hệ thống trả về thông báo cho người dùng.

c. Biểu đồ hoạt động (Activity Diagram)

Biểu đồ hành động là công cụ để mô hình các quá trình phức tạp

Thường dùng để mô tả:

- Luồng công việc (workflow) của hệ thống
- Những hành động trong từng kịch bản của ca sử dụng
- Các chi tiết hoạt động của một chức năng

Bảng 1.2. Bảng giải thích ký hiệu dùng để vẽ trong Activity Diagram.

| Ký hiệu | Đặc điểm |
|---------|--|
| | <ul style="list-style-type: none"> Khởi tạo một hoạt động. Một activity diagram có thể có nhiều trạng thái Start. |
| | Mô tả sự chuyển đổi trạng thái của các hoạt động |
| | <ul style="list-style-type: none"> Mô tả hành vi của đối tượng trong quy trình Tên hoạt động phải ngắn gọn, đủ nghĩa. Nên đặt là động từ và mô tả đầy đủ ý nghĩa tổng thể của hoạt động nhất có thể. |
| | Đây là kí hiệu biểu thị nút điều kiện chuyển hướng. Tùy theo trường hợp đúng hay sai của kết quả mà có hướng di chuyển tiếp theo tương ứng. |
| | <p>Synchronization bar bao gồm hai loại sau: Join và Fork</p> <ul style="list-style-type: none"> Khi có các trường hợp cần hội tụ đủ nhiều luồng điều khiển một lúc để gộp thành một luồng xử lý thì cần dùng JOIN. Khi cần phải tách một luồng điều khiển ra hai hoặc nhiều luồng khác biệt nhau thì cần dùng FORK. Mỗi luồng của FORK hoàn toàn không lệ thuộc nhau. |
| | <ul style="list-style-type: none"> Mô tả trạng thái kết thúc quy trình Một activity diagram có thể có một hoặc nhiều trạng thái kết thúc |

1.3. Tìm hiểu tổng quan hệ thống cơ sở dữ liệu và hệ quản trị cơ sở dữ liệu.

1.3.1. Khái niệm dữ liệu và cơ sở dữ liệu.

Dữ liệu là các thông tin liên quan đến đối tượng (ví dụ như người, vật, sự việc,...) được lưu trữ trên máy tính. Dữ liệu được mô tả dưới nhiều dạng khác nhau (như ký tự, ký hiệu, hình ảnh, , âm thanh...)

Dữ liệu về đối tượng có thể khác nhau tùy thuộc vào mục đích quản lý. Ví dụ: cùng là dữ liệu về đối tượng sinh viên nhưng với mục đích quản lý khác nhau thì sẽ khác

nhau, cụ thể như sau: Với mục đích quản lý điểm, dữ liệu về đối tượng sinh viên sẽ bao gồm: Tên, Mã sinh viên, Điểm môn 1, Điểm môn 2, Điểm môn 3. Với mục đích quản lý thông tin cá nhân, dữ liệu về đối tượng sinh viên sẽ bao gồm: Tên, Địa chỉ, Ngày Sinh, Quê Quán, Lớp.

Cơ sở dữ liệu là tập hợp dữ liệu được tổ chức một cách có cấu trúc liên quan với nhau và được lưu trữ trong máy tính. Cơ sở dữ liệu được thiết kế, xây dựng cho phép người dùng lưu trữ dữ liệu, truy xuất thông tin hoặc cập nhật dữ liệu.

Lợi ích của việc quản lý dữ liệu bằng cơ sở dữ liệu:

- Giúp dữ liệu được lưu trữ một cách hiệu quả và có tổ chức.
- Tránh dữ thửa, trùng lặp dữ liệu.
- Đảm bảo sự nhất quán trong CSDL.
- Các dữ liệu được lưu trữ có thể được chia sẻ.
- Duy trì tính toàn vẹn dữ liệu.
- Đảm bảo dữ liệu được bảo mật.

1.3.2. Các mô hình cơ sở dữ liệu.

a. Mô hình dữ liệu file phẳng (Flat file): dùng cho các CSDL đơn giản. CSDL dạng file phẳng thường là file kiểu văn bản chứa dữ liệu dạng bảng.

b. Mô hình dữ liệu mạng (Network model): Các file trong mô hình dữ liệu mạng được gọi là các bản ghi. Tập hợp các bản ghi cùng kiểu tạo thành một kiểu thực thể dữ liệu. Các kiểu thực thể kết nối với nhau thông qua mối quan hệ cha-con. Mô hình này được biểu diễn bởi một đồ thị có hướng và các mũi tên chỉ từ kiểu thực thể cha sang kiểu thực thể con. Ưu điểm của mô hình này là có thể biểu diễn được các mối quan hệ phức tạp. Nhược điểm là truy xuất chậm và không thích hợp với các CSDL có quy mô lớn.

c. Mô hình dữ liệu phân cấp (Hierarchical model): Tổ chức theo hình cây, mỗi nút biểu diễn một thực thể dữ liệu. Mỗi nút cha có thể có một hoặc nhiều nút con nhưng mỗi nút con chỉ có một nút cha. Do đó mô hình dữ liệu phân cấp có thể có các kiểu quan hệ: 1-1, 1-N. Nhược điểm của mô hình này là một nút con không thể có quá một nút cha nên không biểu diễn được các quan hệ phức tạp.

d. Mô hình dữ liệu quan hệ (Relational model): Trong mô hình dữ liệu quan hệ không có các liên kết vật lý. Dữ liệu được biểu diễn dưới dạng bảng với các hàng và các cột. Dữ liệu trong hai bảng liên kết với nhau thông qua các cột chung và có các toán tử để thao tác trên các hàng của bảng.

e. Mô hình dữ liệu hướng đối tượng (Object-Oriented model): Mô hình này ra đời vào khoảng đầu những năm 90, dựa trên cách tiếp cận của phương pháp lập trình hướng đối tượng. Cơ sở dữ liệu bao gồm các đối tượng:

- Mỗi đối tượng bao gồm các thuộc tính, phương thức của đối tượng
- Các đối tượng trao đổi với nhau thông qua các phương thức
- Một đối tượng có thể được sinh ra từ việc kế thừa từ đối tượng khác

1.3.3. Hệ quản trị cơ sở dữ liệu là gì ?

Hệ quản trị cơ sở dữ liệu (Database Management System - DBMS): Là một hệ thống phần mềm cho phép tạo lập cơ sở dữ liệu và cung cấp cơ chế lưu trữ, truy cập dựa trên các mô hình CSDL.

Trên thị trường phần mềm hiện nay xuất hiện khá nhiều phần mềm hệ quản trị cơ sở dữ liệu như: Microsoft Access, Foxpro, SQL Server, Oracle...Trong đó:

- SQL Server, Microsoft Access, Oracle là các hệ quản trị CSDL điển hình cho mô hình quan hệ
- IMS của IBM là hệ quản trị CSDL điển hình cho mô hình dữ liệu phân cấp
- IDMS là hệ quản trị CSDL điển hình cho mô hình dữ liệu mạng

Những lợi ích mà DBMS mang lại:

- Quản trị các CSDL
- Cơ chế an toàn, bảo mật cao
- Hỗ trợ các ngôn ngữ giao tiếp. Ví dụ như ngôn ngữ truy vấn dữ liệu có cấu trúc - SQL, ngôn ngữ thao tác dữ liệu - DML, ngôn ngữ mô tả, định nghĩa dữ liệu – DDL

CHƯƠNG 2. KHẢO SÁT VÀ PHÂN TÍCH THIẾT KẾ HỆ THỐNG

2.1. Khảo sát.

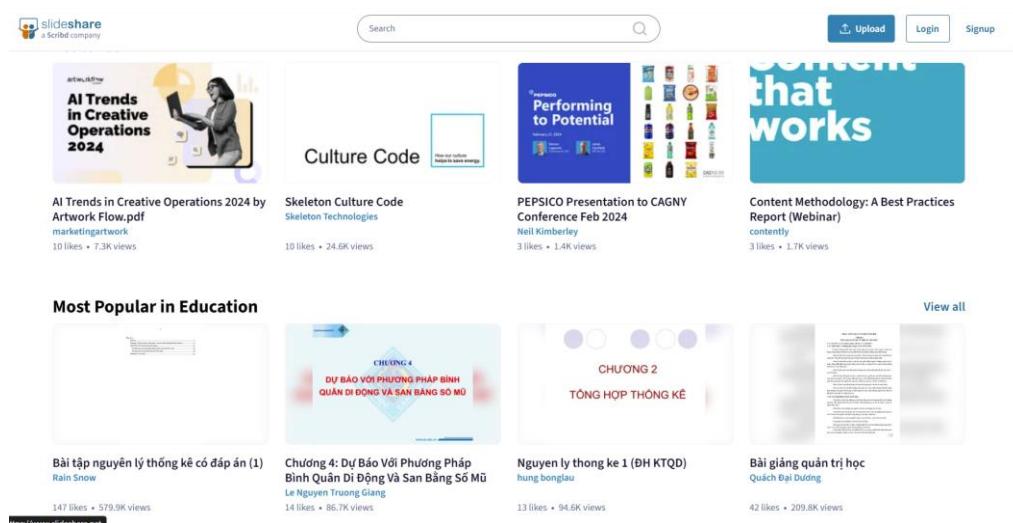
2.1.1. Khảo sát website tương tự

Hiện nay trên thị trường có rất nhiều website chia sẻ tài liệu đa dạng như slideshare, scribd,...

Trang web Slideshare

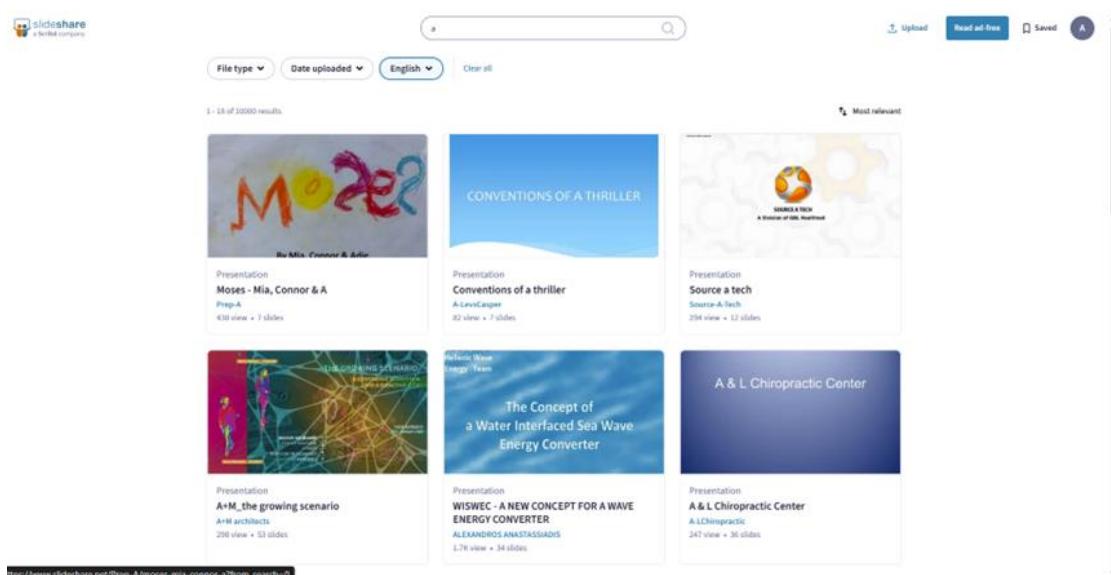
Slideshare.net là một nền tảng chia sẻ bài thuyết trình, tài liệu PDF và infographics. Giao diện ứng dụng: giao diện của ứng dụng khá thân thiện, đơn giản về màu sắc, lối thiết kế theo hướng tối giản, dễ dàng sử dụng.

- Giao diện trang chủ: Hiển thị các tài liệu mới nhất, nổi bật nhất của nhiều thể loại tài liệu khác nhau.



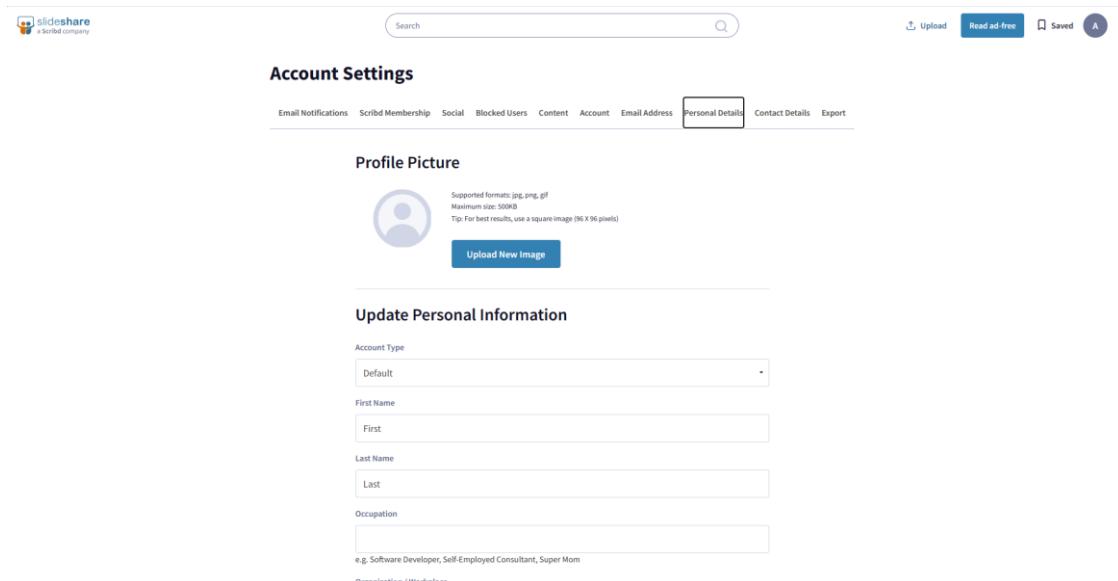
Hình 2.1. Hình trang chủ website Slideshare.

- Giao diện trang tìm kiếm: Tìm kiếm các tài liệu theo các chủ đề, các từ khóa, tác giả,...



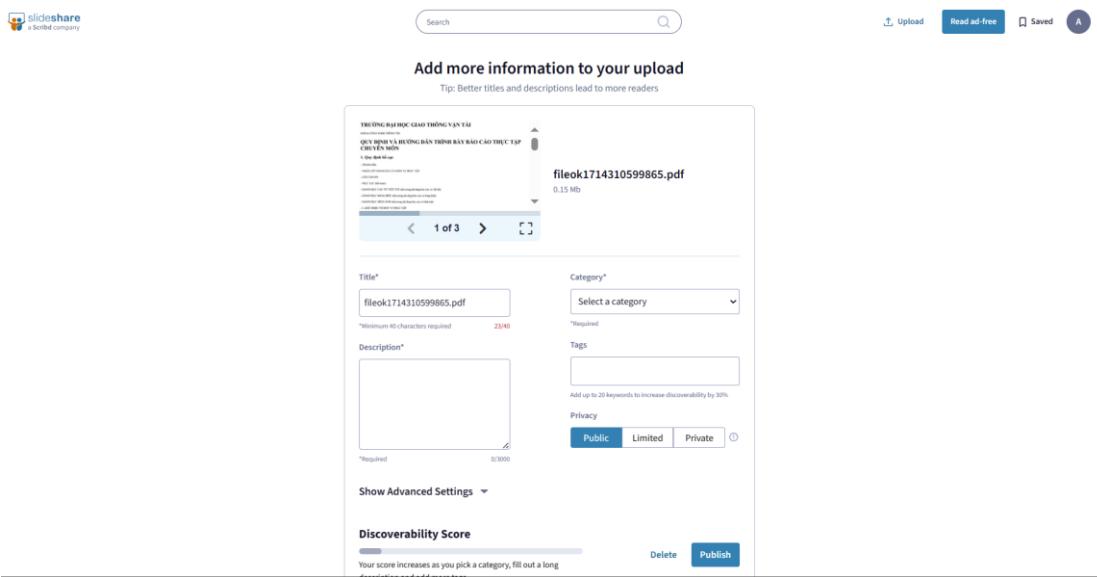
Hình 2.2. Giao diện trang tìm kiếm website SlideShare.

- Giao diện trang thông tin cá nhân: Hiển thị thông tin chi tiết của tài khoản như: họ và tên, ảnh đại diện,...



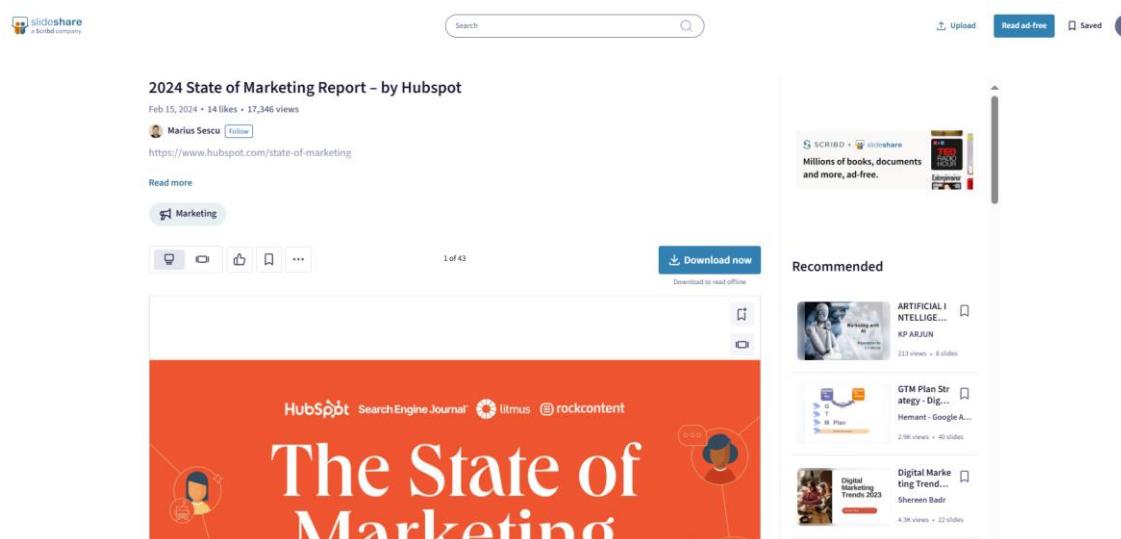
Hình 2.3. Giao diện trang thông tin cá nhân website SlideShare.

- Giao diện trang upload tài liệu: Người dùng có thể upload tài liệu dưới dạng PDF, DOCX,... Thông tin chi tiết một tài liệu bao gồm như: tiêu đề tài liệu, mô tả, chủ đề, quyền riêng tư,...



Hình 2.4. Giao diện trang upload tài liệu website SlideShare.

- Giao diện chi tiết tài liệu: Hiển thị một số thông tin của tài liệu như: tên tài liệu, loại tài liệu, lượt xem tài liệu, lượt yêu thích, nội dung tài liệu, người đăng,...



Hình 2.5. Giao diện trang chi tiết tài liệu website SlideShare.

Tính năng nổi bật:

- Tải lên bài thuyết trình PowerPoint, Keynote, PDF, và các tài liệu khác.
- Xem bài thuyết trình: Người dùng có thể xem bài thuyết trình trực tuyến hoặc tải về máy tính
- Tìm kiếm bài thuyết trình: Người dùng có thể tìm kiếm bài thuyết trình theo từ khóa, chủ đề, tác giả, v.v. Tạo các bộ sưu tập bài thuyết trình.

- Theo dõi các người dùng khác và nhận thông báo khi họ tải lên bài thuyết trình mới.
- Thích, bình luận và chia sẻ bài thuyết trình: Người dùng có thể thích, bình luận và chia sẻ bài thuyết trình với những người khác...

Hạn chế:

- Chất lượng nội dung không đồng đều.
- Nguy cơ mất an toàn thông tin.

Quy trình đăng và quản lý tài liệu:

- Đăng Ký và Đăng Nhập: Người dùng cần có tài khoản để tải lên và chia sẻ tài liệu.
- Tải Lên Tài Liệu: Tài liệu được tải lên dưới dạng PDF, PPT, DOCX hoặc video, sau đó được phân loại theo chủ đề và thẻ từ khóa.
- Quản Lý Tài Liệu: Người dùng có thể chỉnh sửa, xóa hoặc thay đổi cài đặt quyền riêng tư của tài liệu bất cứ lúc nào.

2.1.2. Phân tích yêu cầu

Từ việc khảo sát các website ở trên, em xác định xây dựng website với các chức năng sau:

a. Xây dựng giao diện

- Thiết kế giao diện thân thiện, mới mẻ, dễ sử dụng với người dùng.
- Sắp xếp bố cục hợp lý.
- Có responsive, tiện lợi và phù hợp với máy tính và điện thoại.

b. Chức năng của người dùng

- Quản lý tài khoản: Tạo mới tài khoản, chỉnh sửa thông tin cá nhân, thay đổi mật khẩu, quên mật khẩu,...
- Tìm kiếm thông tin: Tìm kiếm tài liệu, bài viết, người dùng khác, xem chi tiết bài viết, đọc tài liệu, tải xuống tài liệu,...
- Quản lý tài liệu: Thêm mới tài liệu, xem các tài liệu của mình, cập nhật tài liệu, xóa tài liệu.
- Quản lý bài viết: Thêm mới bài viết, xem các bài viết của mình, cập nhật bài viết xóa bài viết.
- Quản lý bình luận: Xem bình luận của bản thân, thêm bình luận mới vào bài viết hoặc tài liệu, xóa bình luận, cập nhật bình luận.

- Quản lý like: Thêm mới hoặc hủy bỏ yêu thích vào tài liệu hoặc bài viết.
- Kết nối với bạn bè: Người dùng có thể kết nối với bạn bè bằng cách nhấn follow người khác.

c. Chức năng của người quản trị hệ thống

- Quản lý người dùng: người quản trị hệ thống có thể xem danh sách người dùng, xem thông tin chi tiết người dùng, cập nhật thông tin người dùng, khóa tài khoản người dùng, xóa người dùng.
- Quản lý tài liệu: người quản trị hệ thống có thể xem danh sách tài liệu, xem thông tin chi tiết tài liệu, tìm kiếm tài liệu, xóa tài liệu và xử lý các bài viết vi phạm.
- Quản lý người dùng: người quản trị hệ thống có thể xem danh sách bài viết, thêm mới bài viết, xem thông tin chi tiết bài viết, cập nhật bài viết, xóa bài viết.
- Quản lý bình luận: người quản trị hệ thống có thể xem danh sách bình luận, xóa bình luận.
- Thông kê: người quản trị viên có thể xem thống kê dữ liệu về người dùng, bài viết, tài liệu.

d. Yêu cầu phi chức năng

Các yêu cầu phi chức năng của một mạng xã hội bao gồm các yếu tố không phải là tính năng cụ thể của trang web, mà thường liên quan đến các khía cạnh khác nhau như hiệu suất, bảo mật, quyền riêng tư, và trải nghiệm người dùng. Dưới đây là một số yêu cầu phi chức năng quan trọng cho một mạng xã hội:

- Hiệu suất và tốc độ: Trang web phải có hiệu suất tốt và thời gian tải nhanh để đảm bảo rằng người dùng có trải nghiệm mượt mà.
- Bảo mật và quyền riêng tư: Cung cấp các biện pháp bảo mật mạnh mẽ để bảo vệ thông tin cá nhân của người dùng, bao gồm cả quyền riêng tư và cách xử lý dữ liệu cá nhân.
- Độ tin cậy và sẵn sàng hoạt động cao: Mạng xã hội cần phải luôn hoạt động và sẵn sàng phục vụ người dùng mọi lúc, ngay cả khi có lưu lượng truy cập lớn.
- Tích hợp công cụ phân tích và theo dõi: Cung cấp các công cụ phân tích và theo dõi để đo lường hiệu quả và hoạt động của trang web.
- Trải nghiệm người dùng tốt: Đảm bảo rằng giao diện người dùng thân thiện, dễ sử dụng và hỗ trợ người dùng tìm kiếm và tương tác dễ dàng.
- Độ tin cậy và ổn định: Trang web cần phải ổn định và không gặp lỗi thường xuyên để duy trì niềm tin của người dùng.

2.2. Phân tích và thiết kế hệ thống

2.2.1. Biểu đồ use case

a. Xác định các tác nhân

Bảng 2.1. Bảng các tác nhân.

| STT | Tác nhân | Mô tả |
|-----|---------------|--|
| 1 | Người dùng | Là người sử dụng hệ thống, có quyền truy cập vào hệ thống. |
| 2 | Quản trị viên | Là người có quyền xem các báo cáo, quản lý tài khoản người dùng, quản lý tài liệu, quản lý bài viết. |

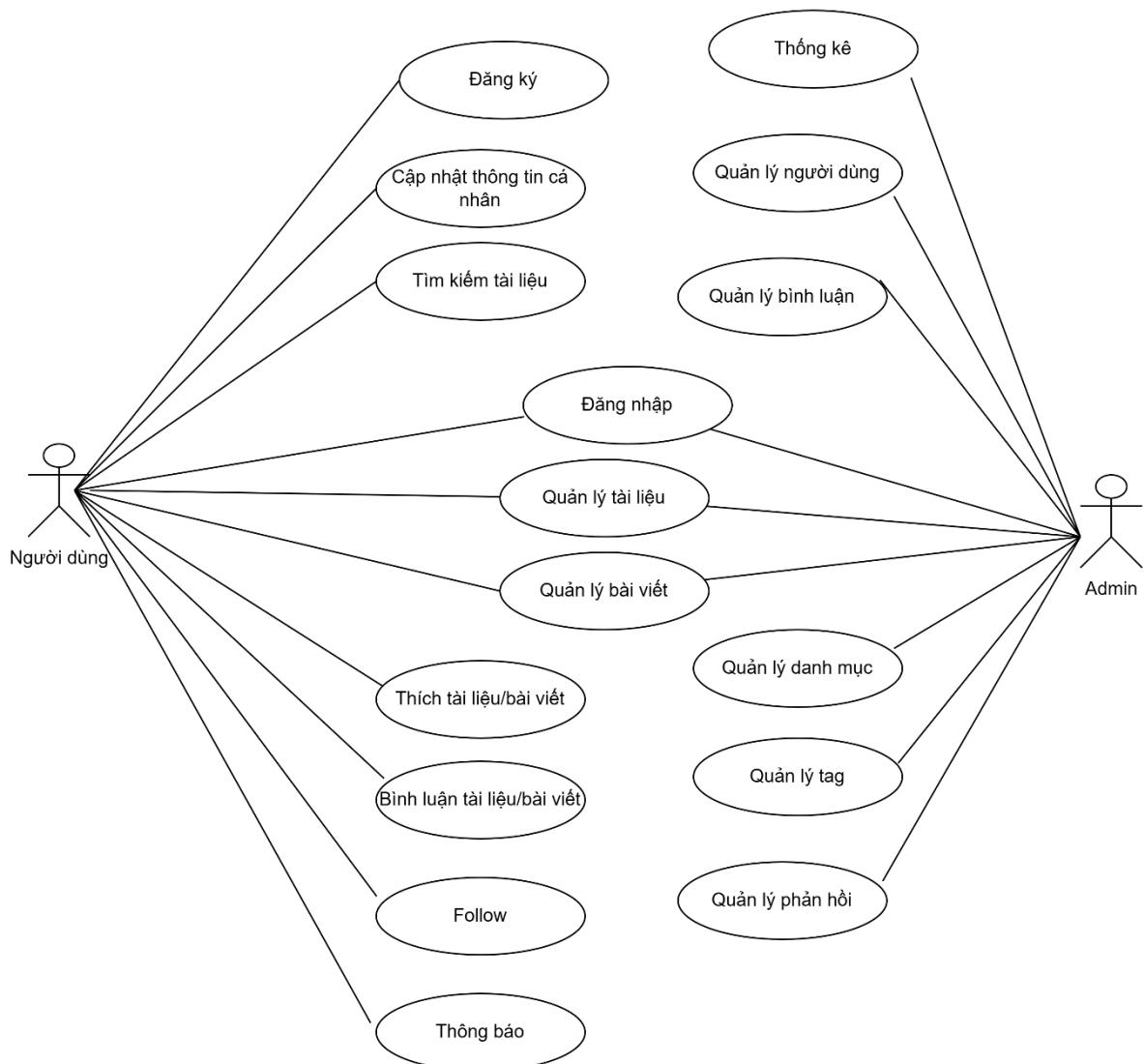
b. Xác định các ca sử dụng

Bảng 2.2. Bảng các ca sử dụng.

| STT | Tác nhân | Use case | Ý nghĩa |
|-----|---------------|-------------------------|---|
| 1 | Người dùng | Đăng nhập | Đăng nhập vào website với tài khoản đã được đăng ký |
| 2 | | Đăng ký | Đăng ký một tài khoản mới |
| 3 | | Đăng tài liệu/bài viết | Đăng tài liệu/bài viết mới lên website. |
| 4 | | Yêu thích | Yêu thích tài liệu/ bài viết bất kỳ |
| 5 | | Bình luận | Bình luận vào tài liệu/ bài viết bất kỳ |
| 6 | | Theo dõi | Theo dõi người khác |
| 7 | | Tìm kiếm | Tìm kiếm tài liệu, bài viết,... |
| 8 | | Chỉnh sửa trang cá nhân | Chỉnh sửa các thông tin cá nhân |
| 9 | | Quên mật khẩu | Lấy lại mật khẩu của người dùng |
| 10 | | Quản lý bài viết | Xem, xóa, sửa bài viết của bản thân |
| 11 | | Quản lý tài liệu | Xem, xóa, sửa tài liệu của bản thân |
| 12 | Quản trị viên | Quản lý người dùng | Xem thông tin, cập nhật, xóa, khóa tài khoản người dùng |
| 13 | | Quản lý tài liệu | Xem thông tin, xóa tài liệu |

| | | |
|----|------------------|---|
| 14 | Quản lý bài viết | Xem thông tin, thêm mới, sửa, xóa bài viết |
| 15 | Thống kê | Thống kê số lượng tài khoản, tài liệu, bài viết,... |

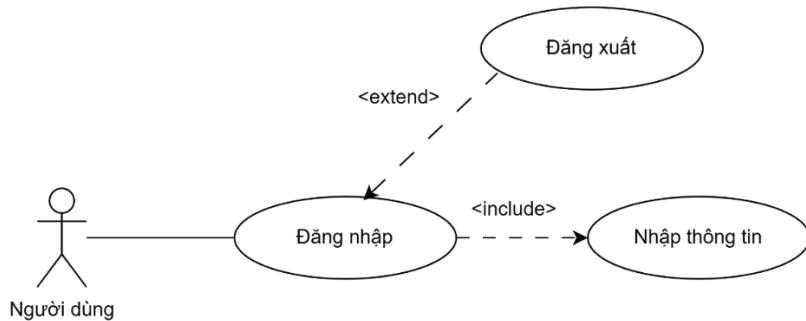
c. Sơ đồ use case tổng quát



Hình 2.6. Sơ đồ use case tổng quát.

d. Mô tả các ca sử dụng.

- **Use case đăng nhập:**



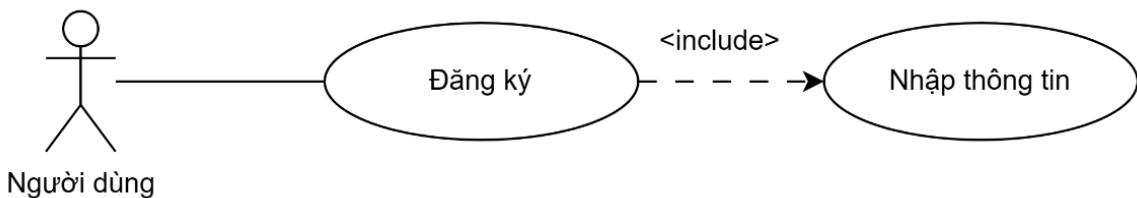
Hình 2.7. Sơ đồ use case đăng nhập.

Đặc tả use case đăng nhập:

Bảng 2.3. Bảng mô tả use case đăng nhập.

| STT | Thành phần | Nội dung |
|-----|--------------|--|
| 1 | Tóm tắt | Use case mô tả cách người dùng đăng nhập vào hệ thống |
| 2 | Actor | Người dùng |
| 3 | Dòng sự kiện | <p>Dòng sự kiện chính: Use case này bắt đầu khi người dùng muốn đăng nhập vào ứng dụng.</p> <ul style="list-style-type: none"> • Hệ thống hiển thị mẫu đăng ký/đăng nhập. • Người dùng nhập tên đăng nhập và mật khẩu. • Hệ thống kiểm tra thông tin đăng nhập(Dữ liệu nhập đã đủ và đúng định dạng chưa, tài khoản được nanding ký khi định dạng đúng hoặc tài khoản được đăng nhập khi tài khoản đã tồn tại trong hệ thống) <p>Nếu thông tin nhập chính xác thì chuyển qua màn hình chính.</p> <p>Dòng sự kiện khác:</p> <ul style="list-style-type: none"> • Nếu trong dòng sự kiện chính, người dùng nhập sai tên đăng nhập, mật khẩu thì hệ thống sẽ hiển thị thông báo lỗi. Người dùng có thể chọn trở về đầu dòng sự kiện chính hoặc hủy bỏ việc đăng nhập, lúc này use case kết thúc. • Nếu use case thành công, người dùng có thể sử dụng các chức năng của website. |
| 4 | Kết quả | Người dùng có thể đăng nhập vào hệ thống. |

- **Use case đăng ký:**



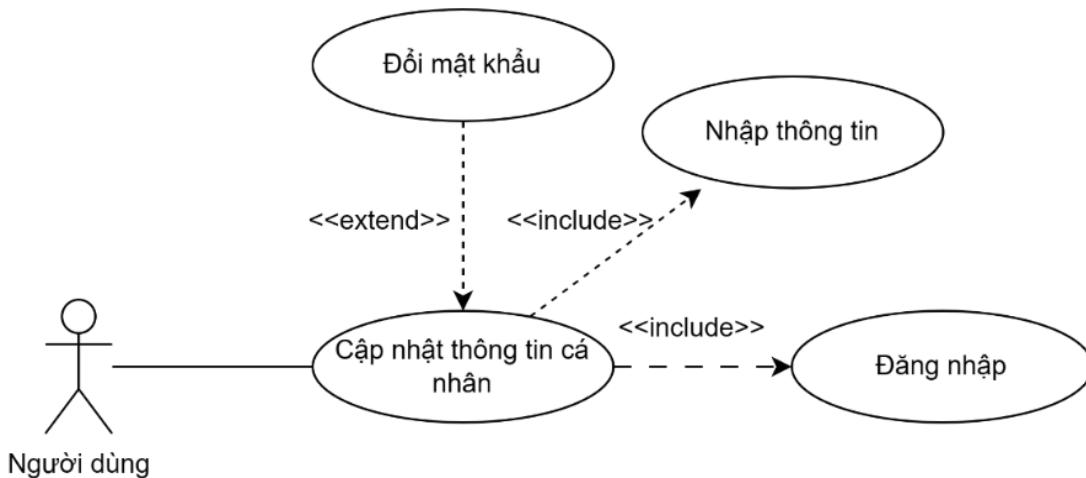
Hình 2.8. Sơ đồ use case đăng ký.

Đặc tả use case đăng ký:

Bảng 2.4. Bảng mô tả use case đăng ký.

| STT | Thành phần | Nội dung |
|-----|--------------|--|
| 1 | Tóm tắt | Use case mô tả cách người dùng đăng ký tài khoản mới |
| 2 | Actor | Người dùng |
| 3 | Dòng sự kiện | <p>Dòng sự kiện chính: Use case này bắt đầu khi người dùng cuối muốn đăng ký một tài khoản trên hệ thống.</p> <ul style="list-style-type: none"> Người dùng nhấn nút Đăng ký, sau đó nhập những thông tin cần thiết. Hệ thống kiểm tra thông tin đăng ký (dữ liệu nhập đã đủ và đúng định dạng chưa, tài khoản được đăng ký khi định dạng đúng hoặc tài khoản đã tồn tại trong hệ thống). <p>Thông tin hợp lệ thì lưu vào CSDL và chuyển người dùng về trang đăng nhập.</p> <p>Dòng sự kiện khác:</p> <ul style="list-style-type: none"> Nếu trong dòng sự kiện chính, người dùng nhập dữ liệu không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi. Người dùng có thể chọn trở về đầu dòng sự kiện chính hoặc hủy bỏ việc đăng nhập, lúc này use case kết thúc. Nếu use case thành công, người dùng có thể đăng nhập và sử dụng các chức năng của website. |
| 4 | Kết quả | Thêm mới dữ liệu vào cơ sở dữ liệu và người dùng có tài khoản mới dùng để đăng nhập vào hệ thống |

- Use case cập nhật tài khoản:



Hình 2.9. Sơ đồ use case cập nhật tài khoản.

Đặc tả use case:

Bảng 2.5. Bảng mô tả use case cập nhật tài khoản.

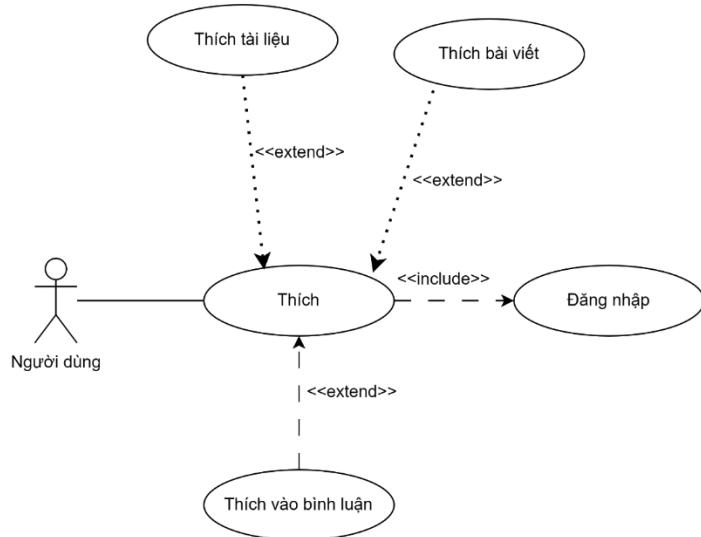
| STT | Thành phần | Nội dung |
|-----|--------------|--|
| 1 | Tóm tắt | Use case mô tả cách người dùng chỉnh sửa thông tin cá nhân. |
| 2 | Actor | Người dùng |
| 3 | Dòng sự kiện | <p>Dòng sự kiện chính: Use case này bắt đầu khi người dùng muốn chỉnh sửa thông tin cá nhân của mình.</p> <ul style="list-style-type: none"> Tại trang cá nhân, người dùng nhập thông tin cần chỉnh sửa và nhấn cập nhật. Hệ thống sẽ xử lý và lưu vào CSDL. Thành công người dùng sẽ được chuyển đến trang cá nhân với thông tin đã được chỉnh sửa. <p>Dòng sự kiện khác: Nếu dữ liệu người dùng nhập vào không hợp lệ, hệ thống sẽ thông báo lỗi và yêu cầu nhập lại.</p> |
| 4 | Kết quả | Người dùng sẽ được trả về trang cá nhân với thông tin đã được cập nhật. |

- Use case đăng tài liệu/bài viết:

Bảng 2.6. Bảng mô tả use case đăng tài liệu/bài viết.

| STT | Thành phần | Nội dung |
|-----|--------------|--|
| 1 | Tóm tắt | Use case mô tả cách người dùng đăng tài liệu/bài viết của mình |
| 2 | Actor | Người dùng |
| 3 | Dòng sự kiện | <p>Dòng sự kiện chính: Use case này bắt đầu khi người dùng muốn đăng một tài liệu/bài viết mới.</p> <ul style="list-style-type: none"> • Tại màn hình chính, người dùng nhấn vào ô đăng tài liệu/bài viết và nhập những thông tin cần thiết. • Hệ thống xử lý dữ liệu và lưu vào CSDL, bài viết mới sẽ được hiển thị trên trang chủ. <p>Dòng sự kiện khác: Nếu dữ liệu người dùng nhập vào không hợp lệ, hệ thống sẽ thông báo lỗi và yêu cầu người dùng nhập lại.</p> |
| 4 | Kết quả | Dữ liệu tài liệu/bài viết của người dùng được thêm mới hiển thị lên trang chủ |

- Use case yêu thích:



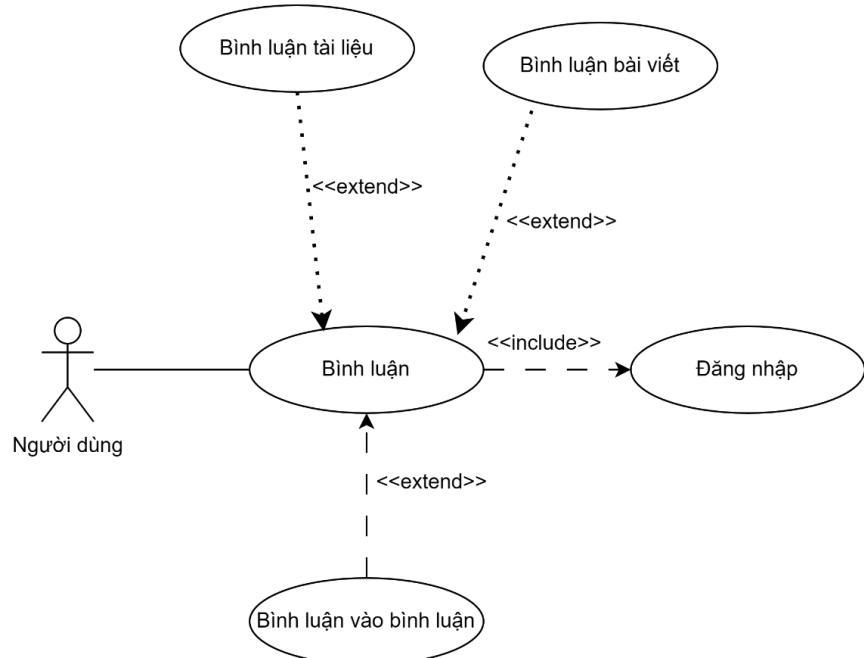
Hình 2.10. Sơ đồ use case yêu thích.

Đặc tả use case:

Bảng 2.7. Bảng mô tả use case yêu thích.

| STT | Thành phần | Nội dung |
|-----|--------------|---|
| 1 | Tóm tắt | Use case mô tả cách người dùng thích một tài liệu/bài viết/bình luận |
| 2 | Actor | Người dùng |
| 3 | Dòng sự kiện | <p>Dòng sự kiện chính:</p> <ul style="list-style-type: none"> Use case này bắt đầu khi người dùng muốn thích một tài liệu/bài viết Tại màn hình chính, người dùng tìm đến tài liệu/bài viết/bình luận muốn thích và nhấn vào biểu tượng trái tim trên mỗi tài liệu/bài viết. Biểu tượng trái tim sẽ chuyển sang màu đỏ và số người thích tại bài viết sẽ được cập nhật. <p>Dòng sự kiện khác: Không.</p> |
| 4 | Kết quả | Dữ liệu được cập nhật |

- **Use case bình luận:**



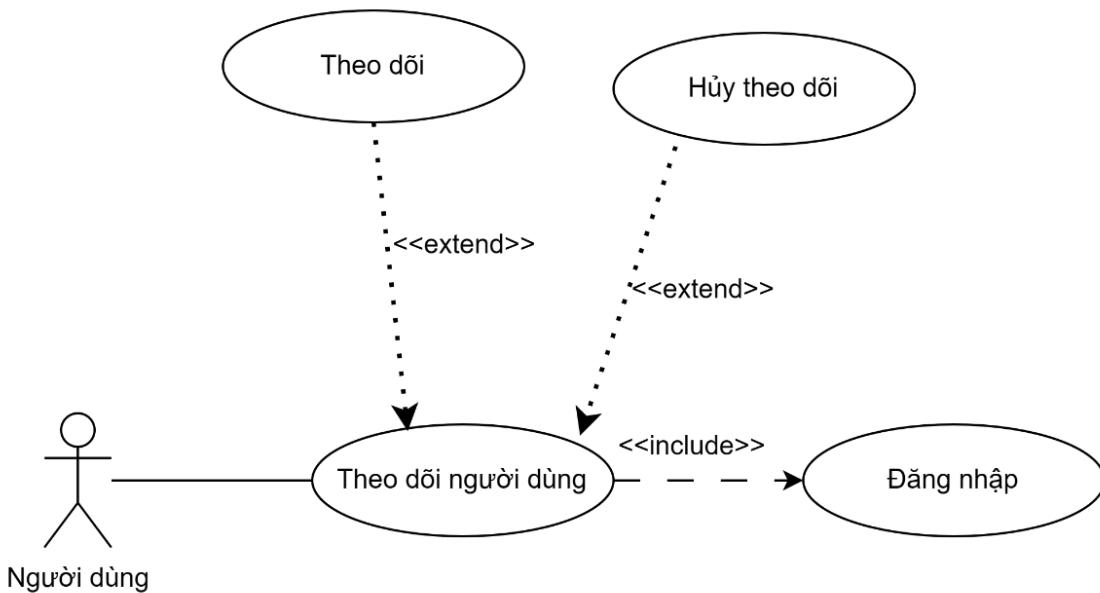
Hình 2.11. Sơ đồ use case bình luận.

Đặc tả use case:

Bảng 2.8. Bảng mô tả use case bình luận.

| STT | Thành phần | Nội dung |
|-----|--------------|---|
| 1 | Tóm tắt | Use case mô tả cách người bình luận vào một tài liệu/bài viết nào đó |
| 2 | Actor | Người dùng |
| 3 | Dòng sự kiện | <p>Dòng sự kiện chính:</p> <ul style="list-style-type: none"> Use case này bắt đầu khi người dùng muốn bình luận vào một tài liệu/bài viết. Tại màn hình chính, người dùng tìm đến bài viết muốn bình luận và viết bình luận của mình vào ô bình luận rồi nhấn Enter. Hệ thống sẽ xử lý và lưu dữ liệu vào CSDL, bình luận sẽ được hiển thị bên dưới tài liệu/bài viết/bình luận. <p>Dòng sự kiện khác: Không.</p> |
| 4 | Kết quả | Dữ liệu bình luận sẽ được cập nhật và hiển thị lên hệ thống. |

- **Use case theo dõi:**



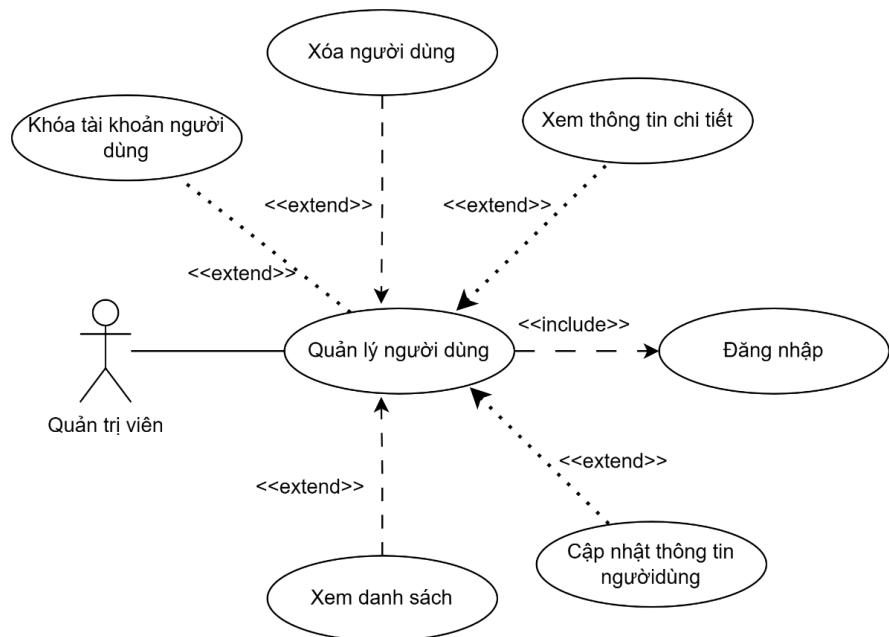
Hình 2.12. Sơ đồ use case theo dõi.

Đặc tả use case:

Bảng 2.9. Bảng mô tả use case theo dõi.

| STT | Thành phần | Nội dung |
|-----|--------------|---|
| 1 | Tóm tắt | Use case mô tả cách người dùng muốn theo dõi người khác |
| 2 | Actor | Người dùng |
| 3 | Dòng sự kiện | <ul style="list-style-type: none"> Dòng sự kiện chính: Use case này bắt đầu khi người dùng muốn theo dõi tài khoản khác. Vào trang cá nhân của người dùng muốn theo dõi.Nhấn nú theo dõi. Hệ thống sẽ xử lý và lưu dữ liệu vào hệ thống. Tài khoản sẽ được bổ sung vào danh sách kết bạn của người dùng, nút “hủy kết bạn” sẽ được chuyển thành nút “kết bạn”. Dòng sự kiện khác: Không. |
| 4 | Kết quả | Danh sách theo dõi của người dùng sẽ được cập nhật. |

- **Use case quản lý người dùng:**



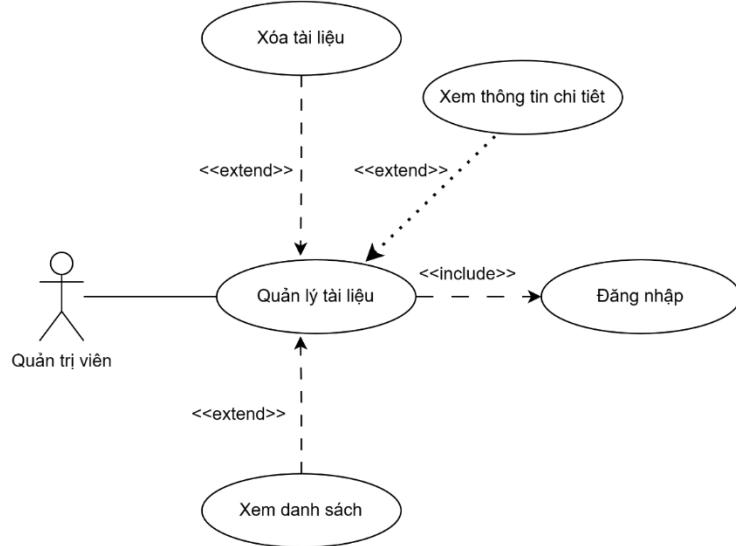
Hình 2.13. Sơ đồ use case quản lý người dùng.

Đặc tả use case:

Bảng 2.10. Bảng mô tả use case quản lý người dùng.

| STT | Thành phần | Nội dung |
|-----|--------------|--|
| 1 | Tóm tắt | Use case mô tả quản trị viên quản lý người dùng trên hệ thống. |
| 2 | Actor | Quản trị viên |
| 3 | Dòng sự kiện | <p>Dòng sự kiện chính:</p> <ul style="list-style-type: none"> • Quản trị viên quản lý thông tin tài khoản của người dùng trên hệ thống. • Quản trị viên chọn chức năng quản lý người dùng để xem danh sách hoặc thực hiện các thao tác quản lý khác. • Hệ thống hiển thị danh sách người hiện có hoặc giao diện cho phép quản trị viên thực hiện các thao tác như sửa, xóa tài khoản, cập nhật thông tin. • Quản trị viên thực hiện các thao tác quản lý tài khoản như sửa thông tin tài khoản, xóa tài khoản nếu tài khoản có vi phạm. • Quản lý cập nhật thông tin và lưu lại thay đổi. • Use case kết thúc <p>Dòng sự kiện khác:</p> <ul style="list-style-type: none"> - Trong trường hợp xảy ra lỗi khi thực hiện các thao tác quản lý người dùng, hệ thống thông báo cho quản trị viên và yêu cầu thử lại hoặc tiếp tục với các thao tác khác. - Trường hợp hủy thao tác, quay lại trang hiển thị danh sách. |
| 4 | Kết quả | Thông tin người dùng được quản lý được cập nhật hoặc xóa thành công |

- Use case quản lý tài liệu:



Hình 2.14. Sơ đồ use case quản lý tài liệu.

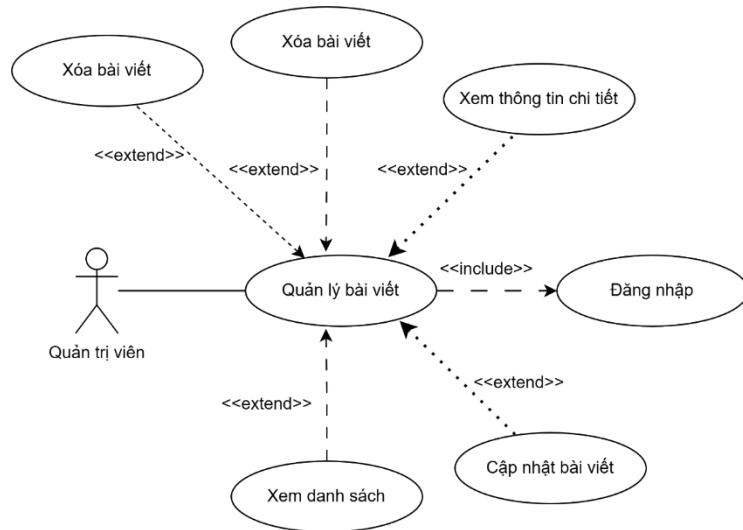
Đặc tả use case:

Bảng 2.11. Bảng mô tả use case quản lý tài liệu.

| STT | Thành phần | Nội dung |
|-----|--------------|--|
| 1 | Tóm tắt | Use case mô tả quản trị viên quản lý tài liệu trên hệ thống. |
| 2 | Actor | Quản trị viên |
| 3 | Dòng sự kiện | <p>Dòng sự kiện chính:</p> <ul style="list-style-type: none"> Quản trị viên quản lý tài liệu trên hệ thống. Quản trị viên chọn chức năng quản lý tài liệu để xem danh sách hoặc thực hiện các thao tác quản lý khác. Hệ thống hiển thị danh sách tài liệu có hoặc giao diện cho phép quản trị viên thực hiện các thao tác như xóa bài viết. Quản trị viên thực hiện các thao tác quản lý tài khoản như xem thông tin, xóa bài viết theo nhu cầu. Quản lý cập nhật thông tin và lưu lại thay đổi. <p>Dòng sự kiện khác:</p> <ul style="list-style-type: none"> Trong trường hợp xảy ra lỗi khi thực hiện các thao tác quản lý người dùng, hệ thống thông báo cho quản |

| | | |
|---|---------|---|
| | | trị viên và yêu cầu thủ lại hoặc tiếp tục với các thao tác khác |
| 4 | Kết quả | Thông tin tài liệu được quản lý được cập nhật hoặc xóa thành công |

- **Use case quản lý bài viết**



Hình 2.15. Sơ đồ use case quản lý bài viết.

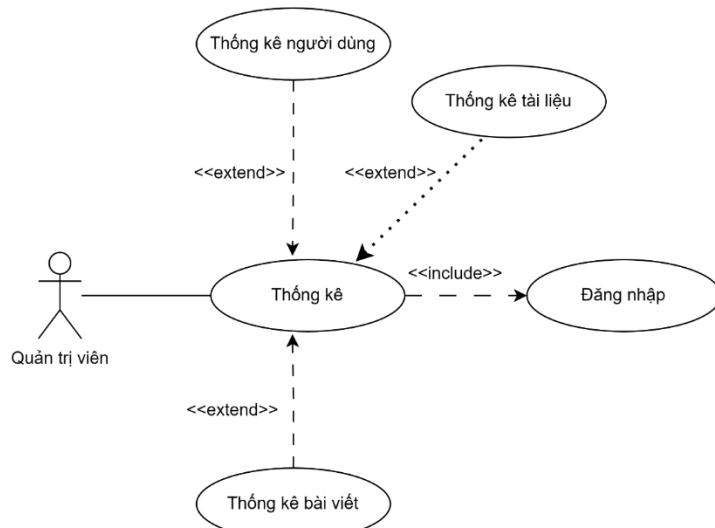
Đặc tả use case:

Bảng 2.12. Bảng mô tả use case quản lý bài viết.

| STT | Thành phần | Nội dung |
|-----|--------------|---|
| 1 | Tóm tắt | Use case mô tả quản trị viên quản lý bài viết trên hệ thống. |
| 2 | Actor | Quản trị viên |
| 3 | Dòng sự kiện | <p>Dòng sự kiện chính:</p> <ul style="list-style-type: none"> • Quản trị viên quản lý bài viết trên hệ thống. • Quản trị viên chọn chức năng quản lý bài viết để xem danh sách hoặc thực hiện các thao tác quản lý khác. • Hệ thống hiển thị danh sách bài viết hiện có hoặc giao diện cho phép quản trị viên thực hiện các thao tác như thêm sửa, xóa bài viết. • Quản trị viên thực hiện các thao tác quản lý bài viết như thêm, sửa, xóa bài viết theo nhu cầu |

| | | |
|---|---------|---|
| | | <ul style="list-style-type: none"> Quản lý cập nhật thông tin và lưu lại thay đổi. Use case kết thúc <p>Dòng sự kiện khác:</p> <ul style="list-style-type: none"> Trong trường hợp xảy ra lỗi khi thực hiện các thao tác quản lý bài viết, hệ thống thông báo cho quản trị viên và yêu cầu thử lại hoặc tiếp tục với các thao tác khác |
| 4 | Kết quả | Thông tin bài viết được quản lý được cập nhật hoặc xóa thành công |

- **Use case thống kê:**



Hình 2.16. Sơ đồ use case thống kê.

Đặc tả use case:

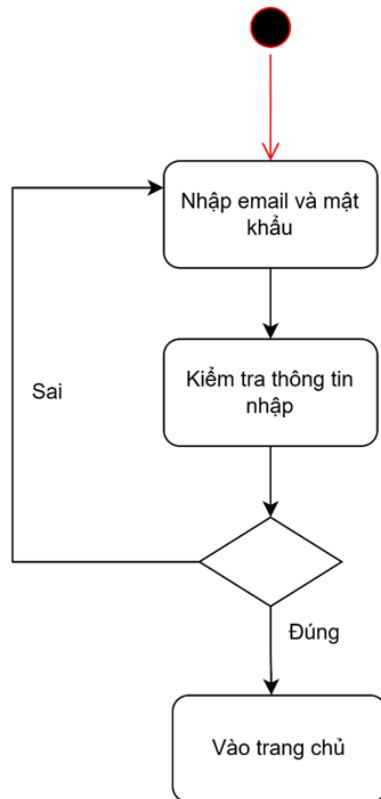
Bảng 2.13. Bảng mô tả use case thống kê.

| STT | Thành phần | Nội dung |
|-----|--------------|--|
| 1 | Tóm tắt | Use case mô tả quản trị viên xem thống kê dữ liệu trên hệ thống |
| 2 | Actor | Quản trị viên |
| 3 | Dòng sự kiện | <p>Dòng sự kiện chính:</p> <ul style="list-style-type: none"> Quản trị viên đăng nhập hệ thống. |

| | | |
|---|---------|---|
| | | <ul style="list-style-type: none"> Quản trị viên chọn chức năng thống kê trên hệ thống. Hệ thống hiển thị các thống kê có sẵn như: thống kê người dùng, thống kê tài liệu, thống kê bài viết,... Quản trị viên xem xét và phân tích dữ liệu trong báo cáo. Use case kết thúc <p>Dòng sự kiện khác:</p> <ul style="list-style-type: none"> Trong trường hợp xảy ra lỗi khi thực hiện các thao tác quản lý người dùng, hệ thống thông báo cho quản trị viên và yêu cầu thử lại hoặc tiếp tục với các thao tác khác |
| 4 | Kết quả | Quản trị viên đã xem báo cáo thống kê và có thể thực hiện các hành động tiếp theo dựa trên thông tin trong báo cáo. |

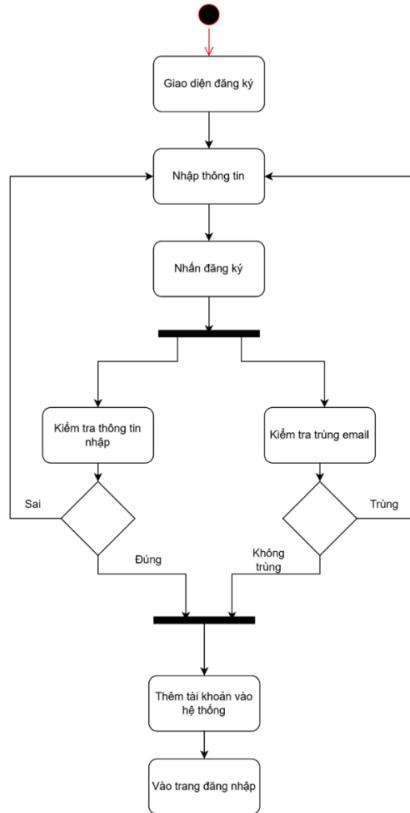
2.2.2. Biểu đồ hành động

- Biểu đồ hành động chức năng đăng nhập:



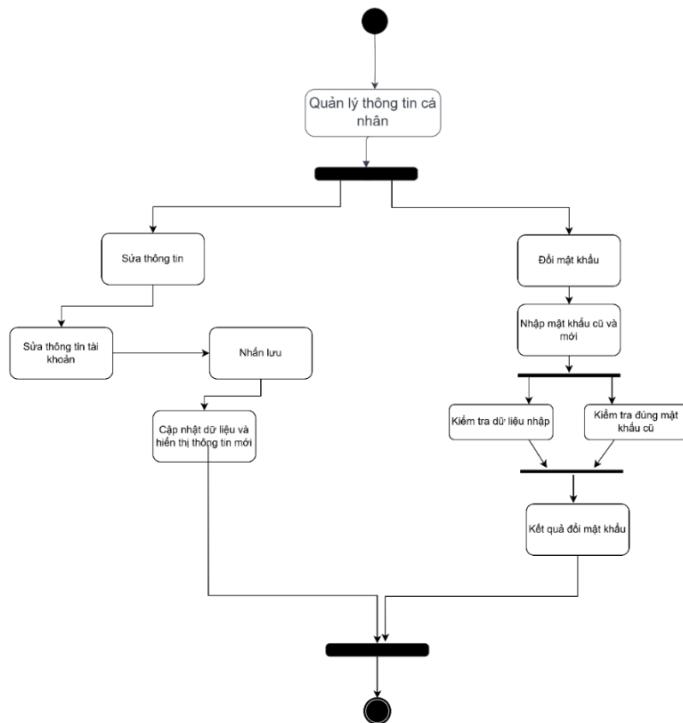
Hình 2.17. Biểu đồ hành động chức năng đăng nhập.

- **Biểu đồ chức năng đăng ký tài khoản:**



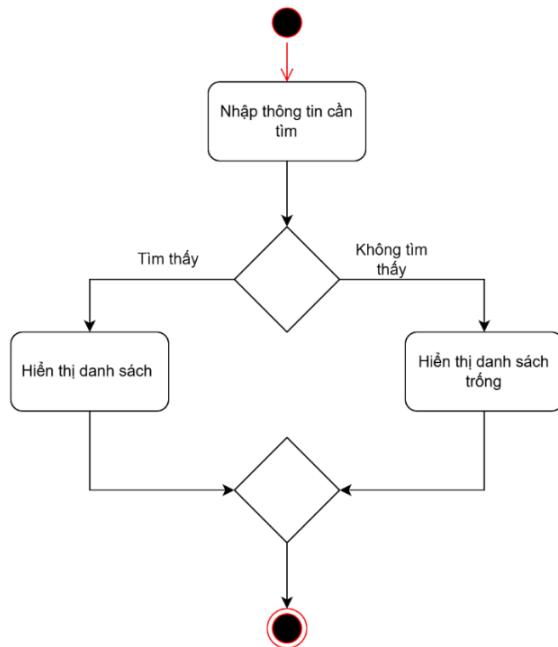
Hình 2.18. Biểu đồ hành động chức năng đăng ký tài khoản.

- **Biểu đồ chức năng cập nhật tài khoản:**



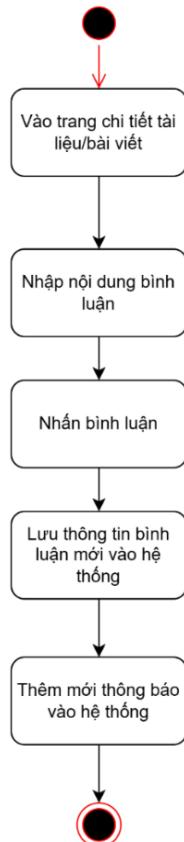
Hình 2.19. Biểu đồ hành động chức năng cập nhật tài khoản.

- **Biểu đồ chức năng tìm kiếm:**



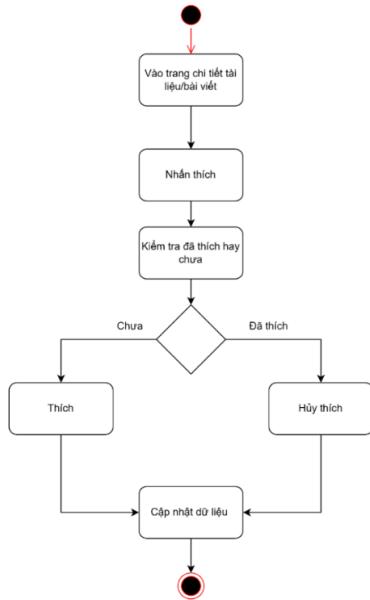
Hình 2.20. Biểu đồ hành động chức năng tìm kiếm.

- **Biểu đồ chức năng thêm mới bình luận**



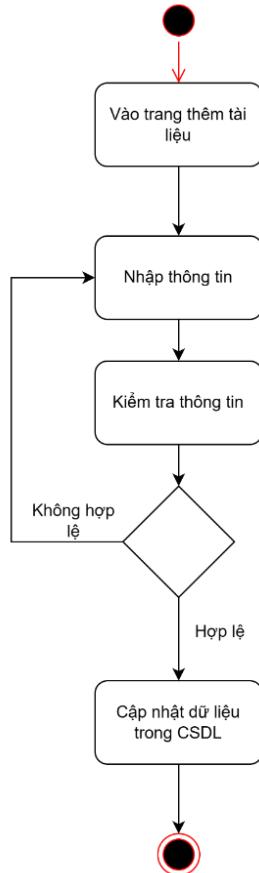
Hình 2.21. Biểu đồ hành động chức năng thêm mới bình luận.

- **Biểu đồ chức năng yêu thích:**



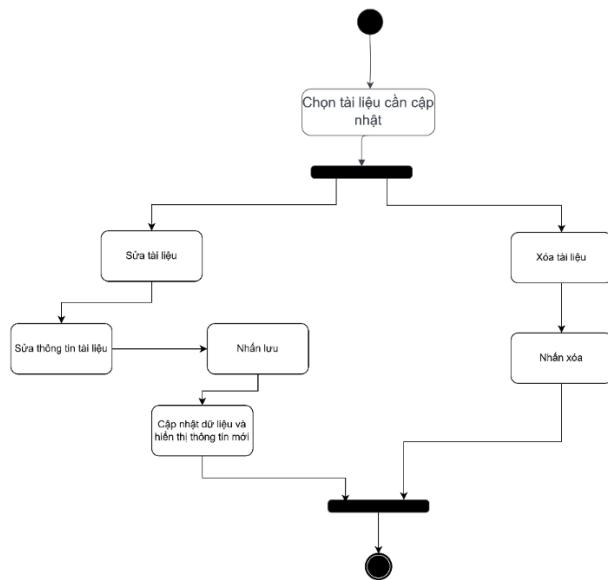
Hình 2.22. Biểu đồ hành động chức năng yêu thích.

- **Biểu đồ hành động chức năng thêm mới tài liệu:**



Hình 2.23. Biểu đồ hành động chức năng thêm mới tài liệu.

- **Biểu đồ hành động cập nhật tài liệu:**



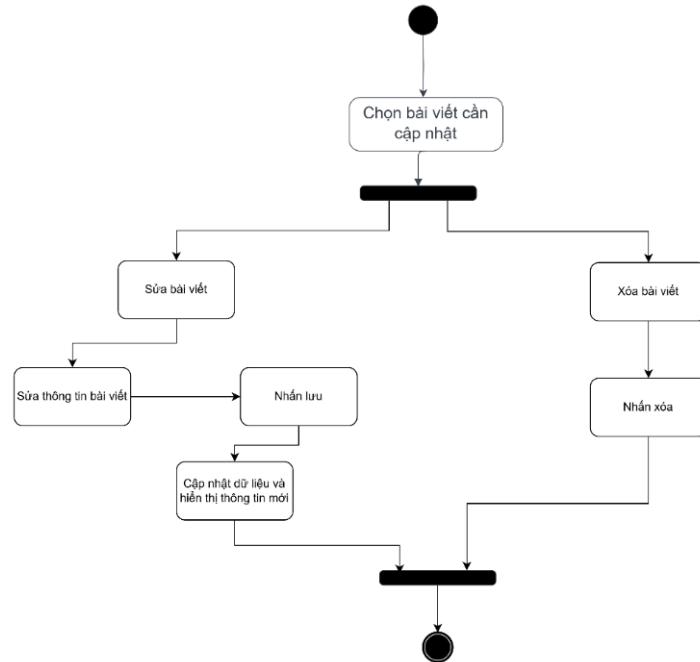
Hình 2.24. Biểu đồ hành động chức năng cập nhật tài liệu.

- **Biểu đồ chức năng thêm mới bài viết:**



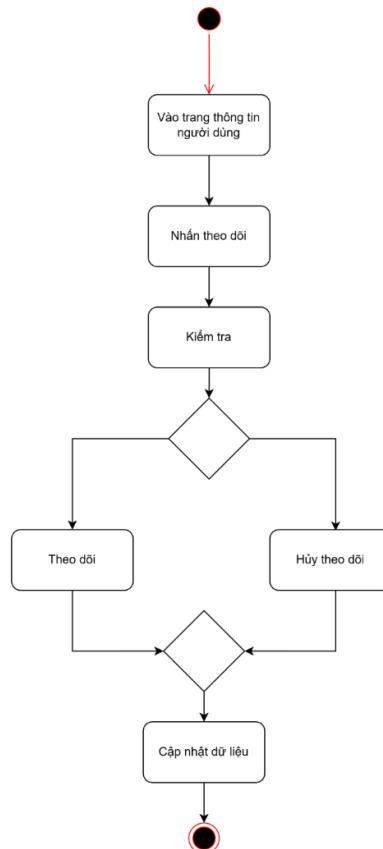
Hình 2.25. Biểu đồ hành động chức năng thêm mới bài viết.

- **Biểu đồ hành động cập nhật bài viết**



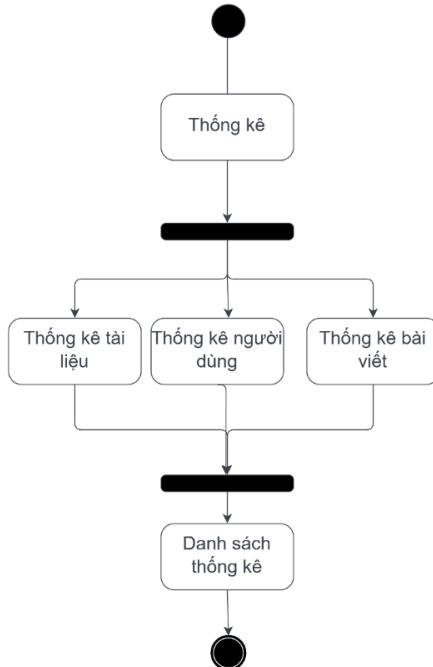
Hình 2.26. Biểu đồ hành động chức năng cập nhật bài viết.

- **Biểu đồ hành động theo dõi người dùng:**



Hình 2.27. Biểu đồ hành động chức năng theo dõi.

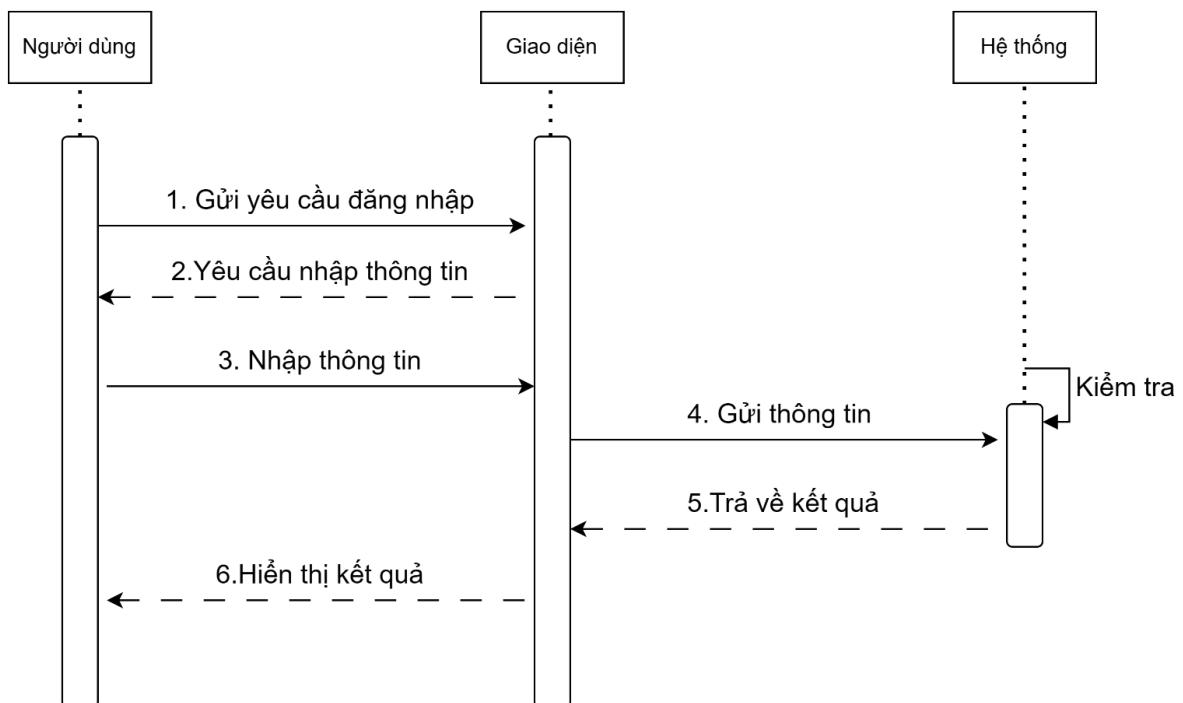
- **Biểu đồ hành động thống kê:**



Hình 2.28. Biểu đồ hành động chức năng thống kê.

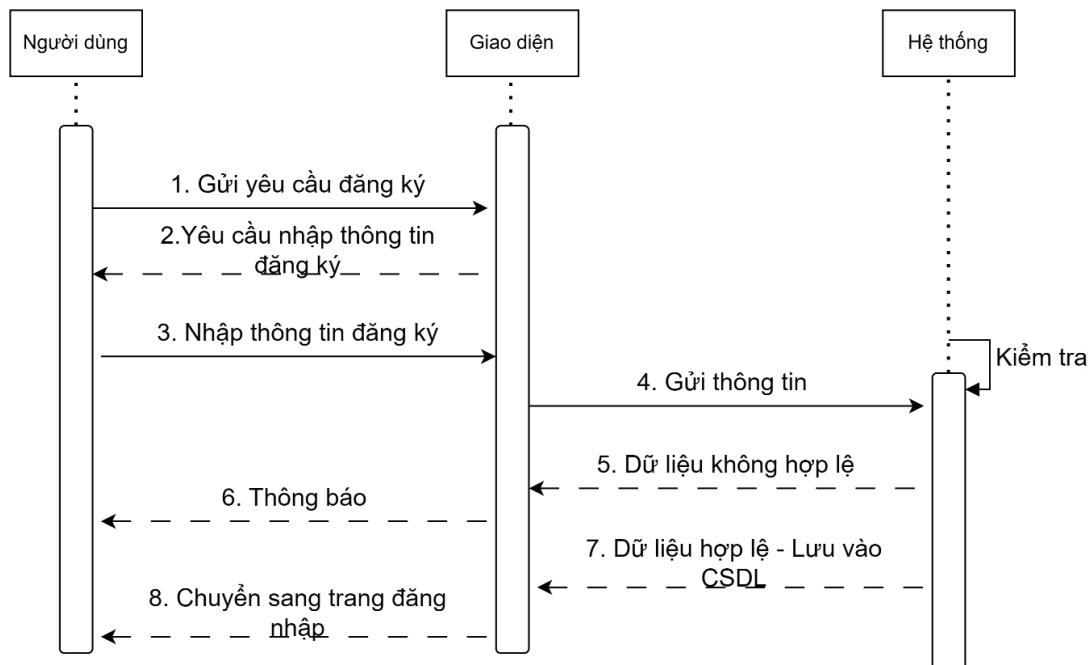
2.2.3. Sơ đồ tuần tự chức năng

- **Sơ đồ tuần tự đăng nhập:**



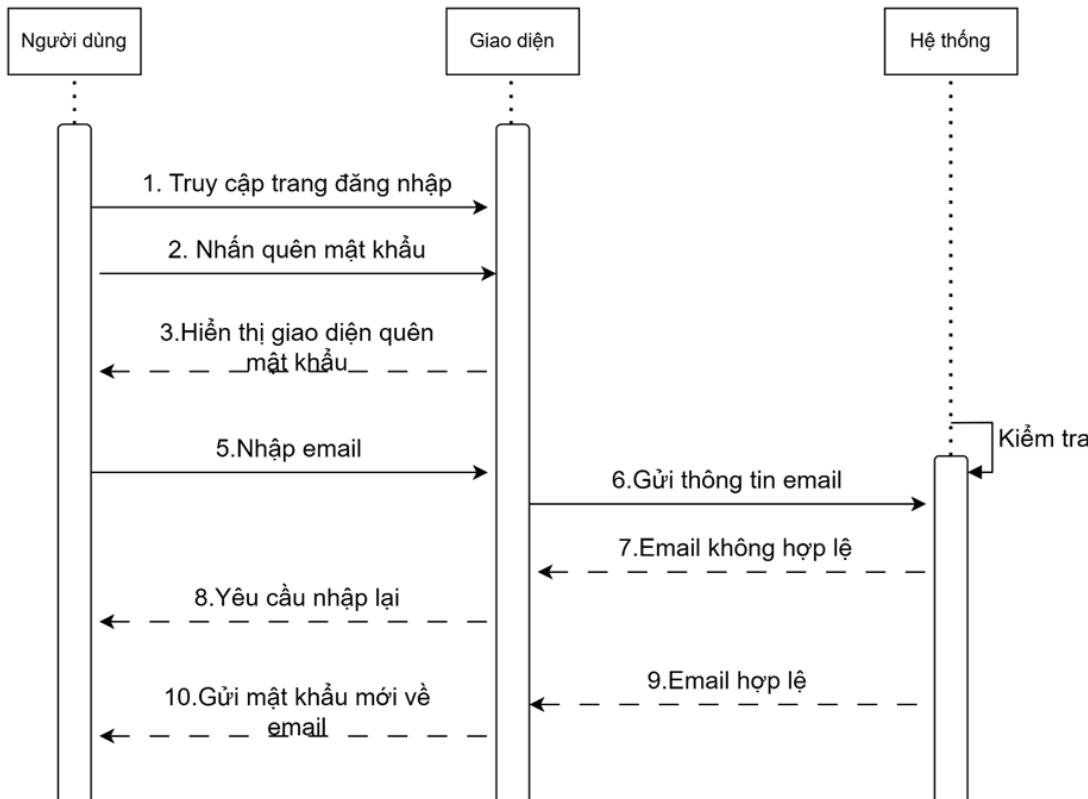
Hình 2.29. Sơ đồ tuần tự chức năng đăng nhập.

- **Sơ đồ tuần tự đăng ký:**



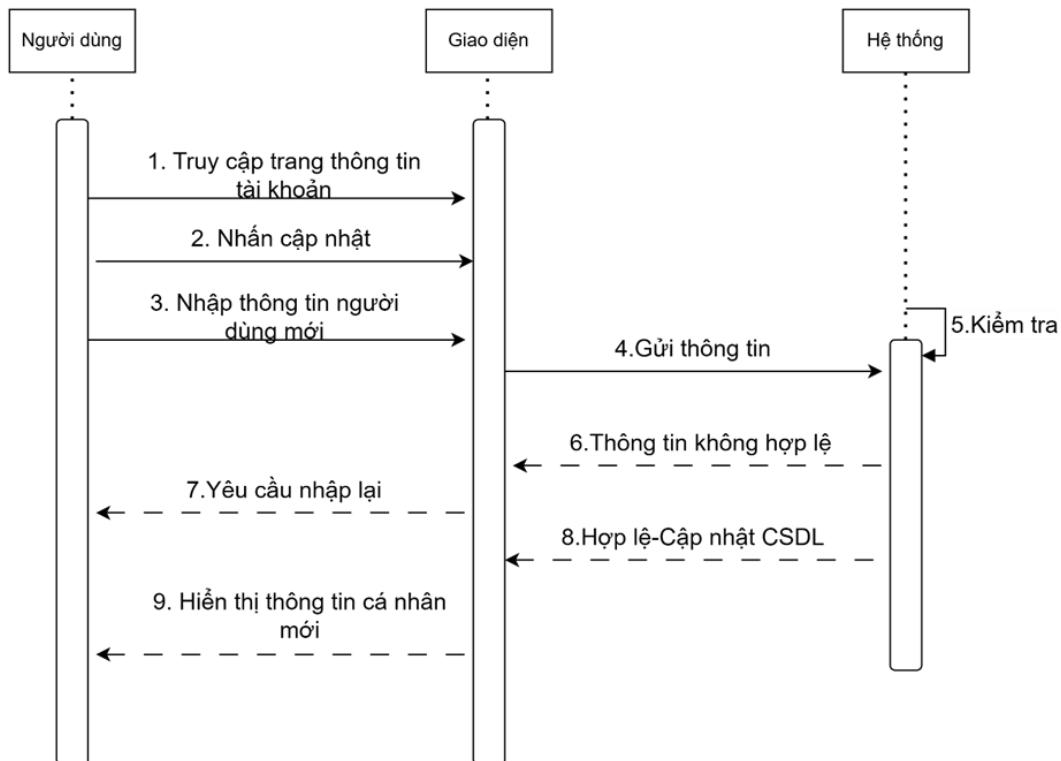
Hình 2.30. Sơ đồ tuần tự chức năng đăng ký.

- **Sơ đồ tuần tự quên mật khẩu:**



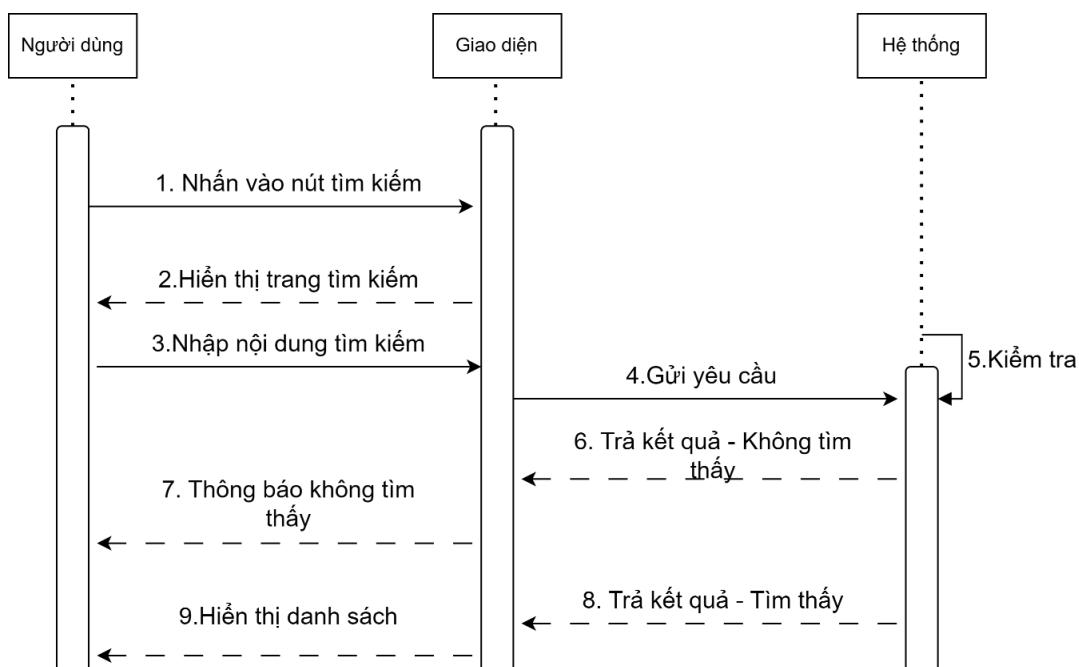
Hình 2.31. Sơ đồ tuần tự chức năng quên mật khẩu.

- **Sơ đồ tuần tự cập nhật thông tin tài khoản:**



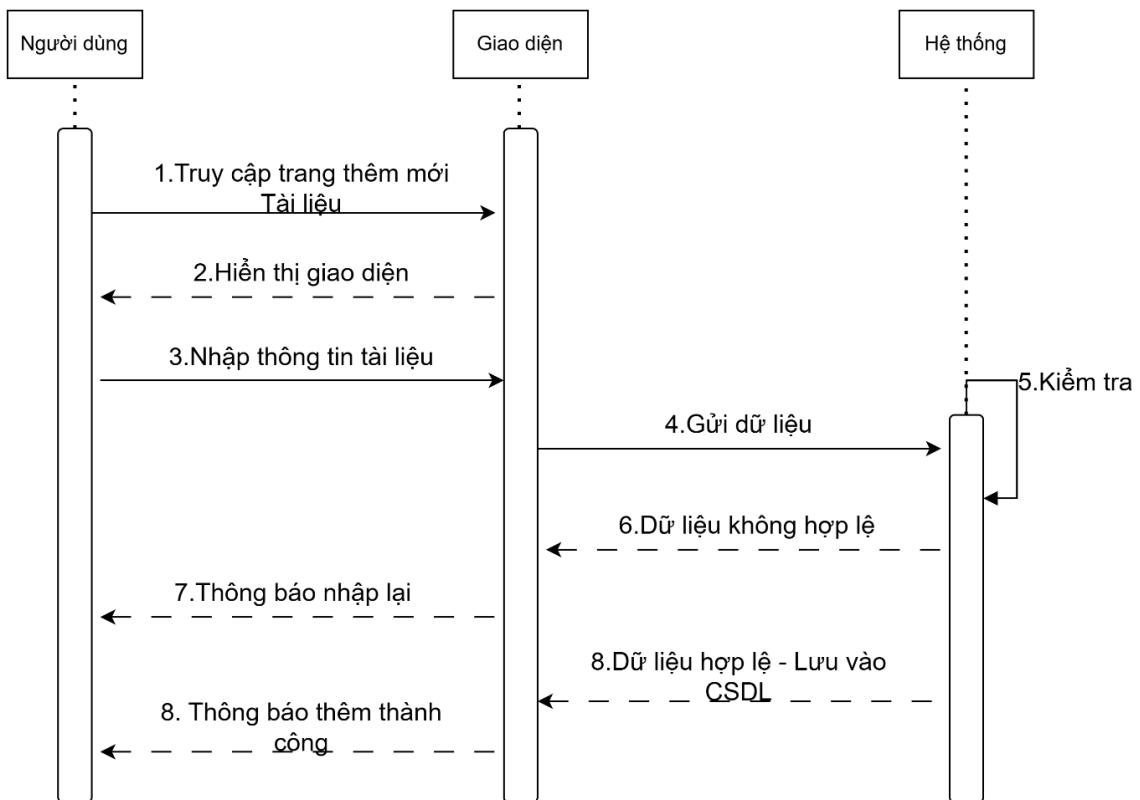
Hình 2.32. Sơ đồ tuần tự chức năng cập nhật thông tin tài khoản.

- **Sơ đồ tuần tự tìm kiếm:**



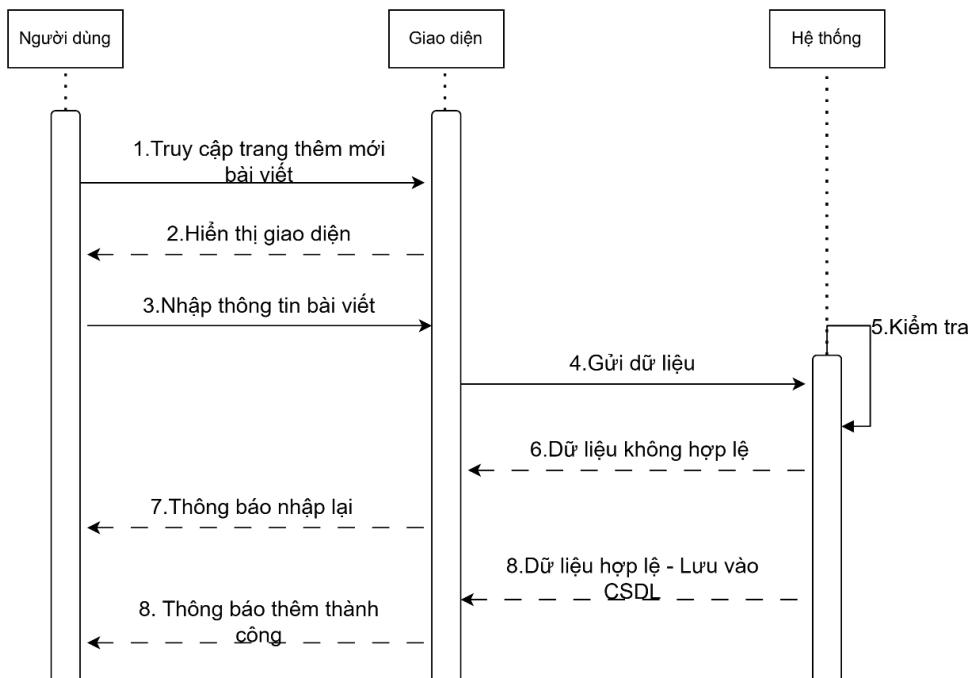
Hình 2.33. Sơ đồ tuần tự chức năng tìm kiếm.

- **Sơ đồ tuần tự đăng tài liệu:**



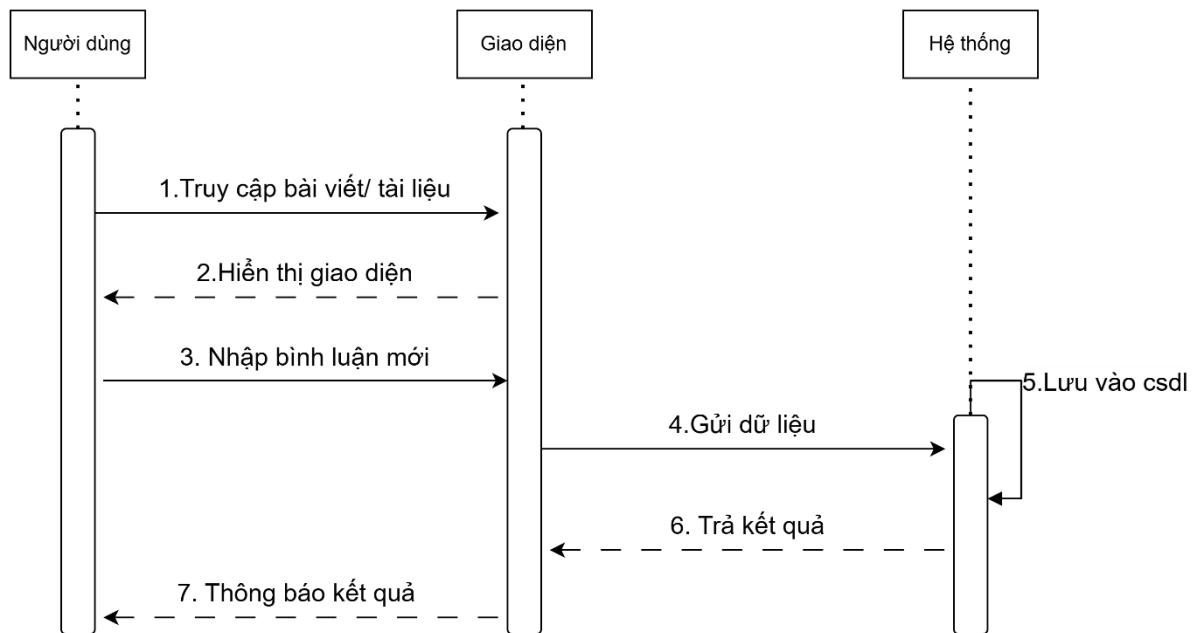
Hình 2.34. Sơ đồ tuần tự chức năng đăng tài liệu.

- **Sơ đồ tuần tự đăng bài viết:**



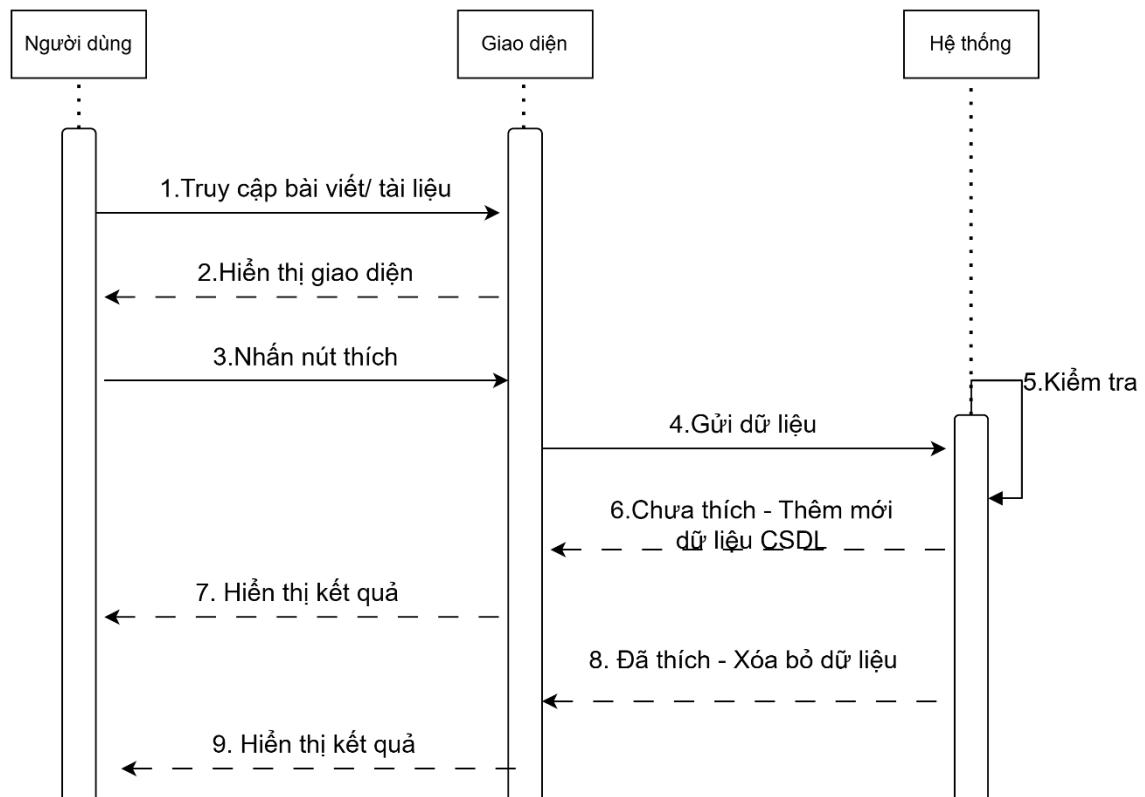
Hình 2.35. Sơ đồ tuần tự chức năng đăng bài viết.

- Sơ đồ tuần tự bình luận:



Hình 2.36. Sơ đồ tuần tự chức năng bình luận.

- Sơ đồ tuần tự yêu thích:



Hình 2.37. Sơ đồ tuần tự chức năng yêu thích.

2.2.4. Phân tích thiết kế cơ sở dữ liệu

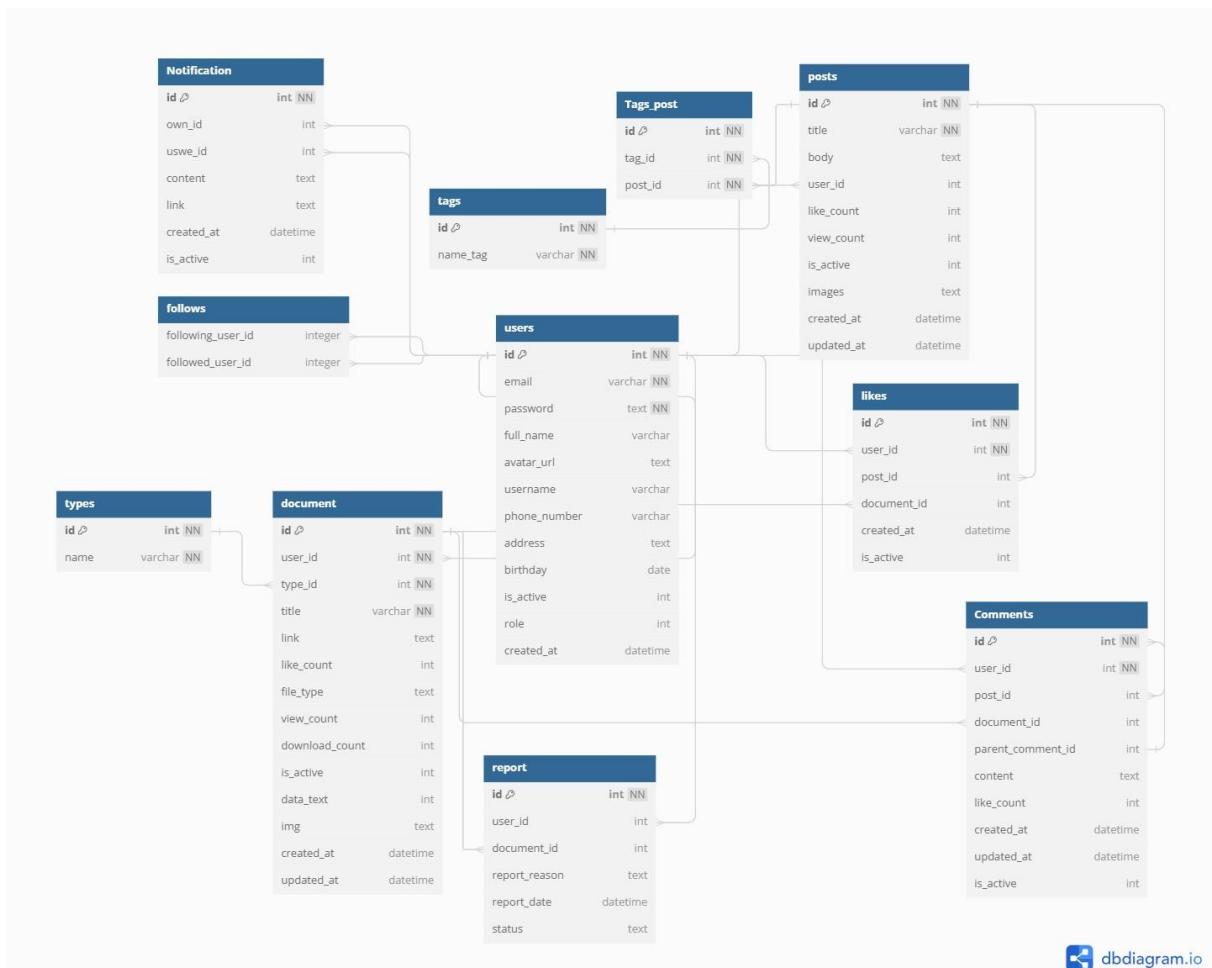
a. Phân tích cơ sở dữ liệu

Bảng 2.14. Bảng các thực thể.

| STT | Tên bảng | Mô tả |
|-----|--------------|--|
| 1 | Users | Lưu thông tin của người dùng |
| 2 | Follow | Lưu thông tin quan hệ của người dùng |
| 3 | Post | Lưu thông tin bài viết |
| 4 | Document | Lưu thông tin của tài liệu |
| 5 | Likes | Lưu các lượt thích bài viết, tài liệu, bình luận |
| 6 | Comments | Lưu thông tin các bình luận |
| 7 | Tags | Lưu thông tin các tags |
| 8 | Types | Lưu thông tin thể loại, lĩnh vực của tài liệu |
| 9 | Notification | Lưu thông tin thông báo của người dùng |
| 10 | Report | Lưu thông tin báo cáo vi phạm từ người dùng |

b. Thiết kế cơ sở dữ liệu

- Sơ đồ thực thể quan hệ:



Hình 2.38. Sơ đồ ERD.

- Mô tả các bảng:

Bảng 2.15. Bảng mô tả chi tiết bảng Tag.

| STT | Tên cột | Kiểu dữ liệu | Nullable | Khóa chính/phụ | Mô tả |
|-----|----------|--------------|----------|----------------|--------------------------|
| 1 | Id | Int | Not null | PK | Id tự động sinh, tự tăng |
| 2 | Name_tag | Varchar(255) | Not null | | Tên của tag |

Bảng 2.16. Bảng mô tả chi tiết bảng Types.

| STT | Tên cột | Kiểu dữ liệu | Nullable | Khóa chính/phụ | Mô tả |
|-----|------------|--------------|----------|----------------|----------------------------|
| 1 | Id | Int | Not null | PK | Id tự động sinh, tự tăng |
| 2 | Name_types | Varchar(255) | Not null | | Tên của thể loại, lĩnh vực |

Bảng 2.17. Bảng mô tả chi tiết bảng Users.

| STT | Tên cột | Kiểu dữ liệu | Nullable | Khóa chính/phụ | Mô tả |
|-----|--------------|--------------|----------|----------------|------------------------------|
| 1 | Id | Int | not null | PK | Id tự động sinh, tự tăng |
| 2 | Email | Varchar(255) | Not null | | Email của tài khoản |
| 3 | Password | Varchar(255) | Not null | | Mật khẩu của tài khoản |
| 4 | Full_name | Varchar(255) | Null | | Tên người dùng |
| 5 | Avatar_url | Text | Null | | Ảnh người dùng |
| 6 | Username | Varchar | Null | | Biệt danh |
| 7 | Phone_number | Varchar | Null | | Số điện thoại của người dùng |
| 8 | Address | Text | Null | | Địa chỉ của người dùng |
| 9 | Birthday | Date | null | | Ngày sinh của người dùng |
| 10 | Is_active | Int | Not null | | Trạng thái của tài khoản |
| 11 | Created_at | Datetime | Not null | | Thời gian tạo tài khoản |

| | | | | | |
|----|------|-----|----------|--|---------------------|
| 12 | Role | Int | Not null | | Quyền của tài khoản |
|----|------|-----|----------|--|---------------------|

Bảng 2.18. Bảng mô tả chi tiết bảng Posts.

| STT | Tên cột | Kiểu dữ liệu | Nullable | Khóa chính/phụ | Mô tả |
|-----|------------|--------------|----------|----------------|-------------------------------------|
| 1 | Id | Int | Not null | PK | Id tự động sinh, tự tăng |
| 2 | User_id | Int | Not null | FK | Người đăng, liên kết với bảng users |
| 3 | Title | Varchar(255) | Not null | | Tiêu đề của bài viết |
| 4 | Body | Text | Null | | Nội dung bài viết |
| 5 | Created_at | Datetime | Not null | | Thời gian tạo bài viết |
| 6 | Updated_at | Datetime | Not null | | Thời gian cập nhật bài viết |
| 7 | Like_count | Int | Null | | Số lượt thích bài viết |
| 8 | View_count | Int | Null | | Số lượt đọc bài viết |
| 9 | Images | Text | Null | | Ảnh của bài viết |
| 10 | Is_active | Int | Not null | | Trạng thái của bài viết |

Bảng 2.19. Bảng mô tả chi tiết bảng tags_post

| STT | Tên cột | Kiểu dữ liệu | Nullable | Khóa chính/phụ | Mô tả |
|-----|---------|--------------|----------|----------------|--------------------------|
| 1 | Id | Int | Not null | PK | Id tự động sinh, tự tăng |
| 2 | Tag_id | Int | Not null | FK/PK | Liên kết với bảng tags |
| 3 | Post_id | Int | Not null | FK/PK | Liên kết với bảng posts |

Bảng 2.20. Bảng mô tả bảng Document.

| STT | Tên cột | Kiểu dữ liệu | Nullable | Khóa chính/phụ | Mô tả |
|-----|----------------|--------------|----------|----------------|--|
| 1 | Id | Int | Not null | PK | Id tự động sinh, tự tăng |
| 2 | User_id | Int | Not null | FK | Người sở hữu tài liệu, liên kết với bảng users |
| 3 | Types_id | Int | Not null | FK | Loại, lĩnh vực của tài liệu, liên kết với bảng types |
| 4 | Title | Varchar(255) | Not null | | Tên của tài liệu |
| 5 | Link | Text | Not null | | Đường dẫn của tài liệu |
| 6 | View_count | Int | | | Lượt xem tài liệu |
| 7 | Download_count | Int | Not null | | Lượt tải tài liệu |
| 8 | View_count | Int | Not null | | Lượt thích tài liệu |
| 9 | File_type | Text | Null | | Kiểu file tài liệu |
| 10 | Data_text | Text | Null | | Nội dung bằng chữ của tài liệu |
| 11 | Img | Text | Null | | Ảnh tài liệu |
| 12 | Created_at | Datetime | Not null | | Thời gian đăng |
| 13 | Updated_at | Datetime | Null | | Thời gian cập nhật |
| 14 | Is_active | Int | Not null | | Trạng thái của tài liệu |

Bảng 2.21. Bảng mô tả bảng Comments.

| STT | Tên cột | Kiểu dữ liệu | Nullable | Khóa chính/phụ | Mô tả |
|-----|-------------------|--------------|----------|----------------|---|
| 1 | Id | Int | Not null | PK | Id tự động sinh, tự tăng |
| 2 | User_id | Int | Not null | FK | Người bình luận, liên kết với bảng users |
| 3 | Post_id | Int | Null | FK | Bài viết được bình luận, liên kết với bảng posts |
| 4 | Document_id | Int | Null | FK | Tài liệu được bình luận, liên kết với bảng document |
| 5 | Parent_comment_id | Int | Null | FK | Bình luận được bình luận, liên kết đến bảng comment |
| 6 | Content | Text | Not null | | Nội dung bình luận |
| 7 | like_count | Int | Not null | | Lượt thích của bình luận |
| 8 | View_count | Int | Not null | | Lượt thích tài liệu |
| 9 | Created_at | Datetime | Not null | | Thời gian viết bình luận |
| 10 | Updated_at | Datetime | Null | | Thời gian cập nhật bình luận |
| 11 | Is_active | Int | Not null | | Trạng thái của bình luận |

Bảng 2.22. Bảng mô tả chi tiết bảng Likes.

| STT | Tên cột | Kiểu dữ liệu | Nullable | Khóa chính/phụ | Mô tả |
|-----|-------------|--------------|----------|----------------|---|
| 1 | Id | Int | Not null | PK | Id tự động sinh, tự tăng |
| 2 | User_id | Int | Not null | FK | Người thích, liên kết với bảng users |
| 3 | post_id | Int | Null | FK | Bài viết được thích, liên kết với bảng posts |
| 4 | Document_id | Int | Null | FK | Tài liệu được thích, liên kết với bảng document |
| 5 | Created_at | Datetime | Not null | | Thời gian thích |
| 6 | Is_active | Int | Not null | | Trạng thái của thích |

Bảng 2.23. Bảng mô tả chi tiết bảng Notification.

| STT | Tên cột | Kiểu dữ liệu | Nullable | Khóa chính/phụ | Mô tả |
|-----|------------|--------------|----------|----------------|--|
| 1 | Id | Int | Not null | PK | Id tự động sinh, tự tăng |
| 2 | Own_id | Int | Not null | FK | Người được nhận thông báo, liên kết với bảng users |
| 3 | User_id | Int | Not null | FK | Người tương tác, liên kết với bảng users |
| 4 | Content | Text | Not null | | Nội dung thông báo |
| 5 | Created_at | Datetime | Not null | | Thời gian tạo thông báo |
| 6 | Link | Text | Not null | | Đường dẫn |

Bảng 2.24. Bảng mô tả bảng follow.

| STT | Tên cột | Kiểu dữ liệu | Nullable | Khóa chính/phụ | Mô tả |
|-----|-------------------|--------------|----------|----------------|--|
| 1 | Id | Int | Not null | PK | Id tự động sinh, tự tăng |
| 2 | Following_user_id | Int | Not null | FK | Người đang theo dõi, liên kết với bảng users |
| 3 | Followed_user_id | Int | Not null | FK | Người theo dõi, liên kết với bảng users |

Bảng 2.25. Bảng mô tả bảng report.

| STT | Tên cột | Kiểu dữ liệu | Nullable | Khóa chính/phụ | Mô tả |
|-----|---------------|--------------|----------|----------------|--|
| 1 | Id | Int | Not null | PK | Id tự động sinh, tự tăng |
| 2 | User_id | Int | Not null | FK | Người gửi báo cáo vi phạm, liên kết với bảng users |
| 3 | Document_id | int | Not null | FK | Tài liệu vi phạm, liên kết với bảng document |
| 4 | report_reason | Text | Not null | | Lý do vi phạm |
| 5 | Report_date | Datetime | Not null | | Thời gian báo cáo vi phạm |
| 6 | Status | Text | Not null | | Trạng thái của report |

CHƯƠNG 3. CÀI ĐẶT VÀ THỬ NGHIỆM WEBSITE

3.1. Các công nghệ sử dụng.

3.1.1. Giới thiệu về NodeJS.

a. Tổng quan:

Nodejs là một nền tảng (Platform) được build dựa trên Chrome's V8 JavaScript engine. Node.js sử dụng mô hình event-driven, non-blocking I/O khiến nó trở nên nhẹ và hiệu quả.

Nodejs được xây dựng và phát triển từ năm 2009, bảo trợ bởi công ty Joyent, trụ sở tại California, Hoa Kỳ.

Phần Core bên dưới của Nodejs được viết hầu hết bằng C++ nên cho tốc độ xử lý và hiệu năng khá cao.

Nodejs tạo ra được các ứng dụng có tốc độ xử lý nhanh, realtime thời gian thực.

Nodejs áp dụng cho các sản phẩm có lượng truy cập lớn, cần mở rộng nhanh, cần đổi mới công nghệ, hoặc tạo ra các dự án Startup nhanh nhất có thể.

b. Điểm nổi bật của Node JS:

- Hiệu suất cao: Node JS sử dụng mô hình lập trình hướng sự kiện (event-driven) và non-blocking I/O, giúp xử lý các yêu cầu đồng thời một cách hiệu quả.
- Khả năng mở rộng: Node JS có thể dễ dàng mở rộng để xử lý nhiều yêu cầu hơn bằng cách thêm CPU cores.
- Cộng đồng lớn: NodeJS có một cộng đồng phát triển lớn và hỗ trợ nhiệt tình.
- Đa dạng thư viện: Có rất nhiều thư viện mã nguồn mở cho NodeJS, giúp bạn thực hiện nhiều chức năng khác nhau.

c. Cài đặt thư viện NodeJS với NPM

NPM (Node Package Manager) là trình quản lý gói được sử dụng để cài đặt các thư viện NodeJS.

Sử dụng lệnh npm install để cài đặt thư viện. Nếu cài đặt một package ta dùng câu lệnh ‘npm install <package-name>’. Nếu cài đặt package với một version cụ thể ta dùng câu lệnh ‘npm install <package-name>@version’.

Sử dụng lệnh npm install --save để cài đặt thư viện và thêm nó vào file package.json của dự án.

Sử dụng lệnh npm uninstall để gỡ cài đặt thư viện.

d. Module

Module là một thành phần cơ bản để xây dựng ứng dụng NodeJS. Nó đóng vai trò như một thư viện chứa các hàm, biến và lớp được định nghĩa sẵn để sử dụng trong ứng dụng.

Lợi ích của việc sử dụng module:

- Tái sử dụng code: Giúp bạn chia nhỏ code thành các module nhỏ gọn, dễ quản lý và tái sử dụng trong các ứng dụng khác nhau.
- Tổ chức code: Giúp bạn tổ chức code một cách khoa học, dễ hiểu và dễ bảo trì.
- Giảm thiểu lỗi: Giúp bạn giảm thiểu lỗi code bằng cách tách biệt các phần code khác nhau.
- Chia sẻ code: Giúp bạn chia sẻ code với các developer khác một cách dễ dàng.

Có hai loại module chính trong NodeJS:

- Module nội bộ: Là các module được cung cấp sẵn bởi NodeJS, ví dụ như fs, http, path, v.v.
- Module bên ngoài: Là các module được cài đặt từ bên ngoài, ví dụ như express, mongoose, lodash, v.v.

Cách sử dụng module:

- Module nội bộ: Sử dụng require() để import module vào ứng dụng.
- Module bên ngoài: Cài đặt module từ npm (Node Package Manager) bằng lệnh npm install <module_name>. Sử dụng require() để import module vào ứng dụng.

e. Nhược điểm của NodeJS:

- Ứng dụng nặng tốn tài nguyên Nếu bạn cần xử lý các ứng dụng tốn tài nguyên CPU như encoding video, convert file, decoding encryption... hoặc các ứng dụng tương tự như vậy thì không nên dùng NodeJS (Lý do: NodeJS được viết bằng C++ & Javascript, nên phải thông qua thêm 1 trình biên dịch của NodeJS sẽ lâu hơn 1 chút).

f. Ứng dụng của NodeJS:

- Web development: Phát triển các ứng dụng web thời gian thực, API, server-side rendering, v.v.

- Network applications: Xây dựng các ứng dụng mạng như chat servers, websockets, v.v.
- Command-line tools: Viết các công cụ dòng lệnh để tự động hóa các tác vụ.
- IoT: Phát triển các ứng dụng cho Internet of Things (IoT).

3.1.2. Giới thiệu về ExpressJS (Express Framework).

Express là một framework được xây dựng trên nền tảng của Nodejs. Nó cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile. Express hỗ trợ các method HTTP và middleware tạo ra API vô cùng mạnh mẽ và dễ sử dụng.

a. Các tính năng nổi bật của Express:

- Thiết lập router cho phép sử dụng với các hành động khác nhau dựa trên phương thức HTTP và URL.
- Hỗ trợ xây dựng theo mô hình MVC.
- Cho phép định nghĩa middleware giúp tổ chức và tái sử dụng code.
- Hỗ trợ RESTful API.

b. Ưu điểm

- Nhẹ và nhanh chóng: Express.js là một framework nhẹ và nhanh chóng, giúp cho việc xây dựng và triển khai các ứng dụng web trở nên dễ dàng hơn.
- Dễ học và sử dụng: Express.js có API đơn giản và dễ hiểu, giúp cho các lập trình viên JavaScript có thể dễ dàng bắt đầu sử dụng nó.
- Hỗ trợ các công cụ động: Express.js hỗ trợ nhiều công cụ động (template engines), giúp dễ dàng hiển thị các trang HTML động. Nhà phát triển có thể lựa chọn từ nhiều công cụ động phổ biến như EJS, Handlebars và Pug. Express.js cũng hỗ trợ tích hợp các tệp tĩnh như hình ảnh, CSS và JavaScript.
- Hỗ trợ các công cụ động: Express.js hỗ trợ nhiều công cụ động (template engines), giúp dễ dàng hiển thị các trang HTML động. Nhà phát triển có thể lựa chọn từ nhiều công cụ động phổ biến như Handlebars và Pug. Express.js cũng hỗ trợ tích hợp các tệp tĩnh như hình ảnh, CSS và JavaScript.

c. Nhược điểm:

- Thiếu cấu trúc: Do ExpressJS không áp đặt một cấu trúc nghiêm ngặt, việc tổ chức dự án và quản lý mã nguồn có thể trở nên khó khăn, đặc biệt khi ứng dụng phát triển lớn và phức tạp.
- Khả năng mở rộng: Khi ứng dụng phát triển lớn và phức tạp, việc quản lý mã nguồn và mở rộng có thể trở nên khó khăn với ExpressJS. Cần có sự kiểm soát

cẩn thận để tránh sự phức tạp và rối rắm trong việc quản lý các module và tương tác giữa chúng.

3.1.3. Ngôn ngữ lập trình Javascript

a. Tổng quan:

JavaScript (được viết tắt JS): là một ngôn ngữ lập trình thông dịch được sử dụng chủ yếu cho phía máy khách(client-side) trong lĩnh vực phát triển web. Nó được sử dụng để tạo ra các tính năng tương tác và động trên các trang web, giúp cải thiện trải nghiệm người dùng trực tuyến.

b. Ưu điểm:

- Phổ biến và năng động
- Đa năng
- Tính tương tác với người dùng
- Cộng đồng lớn và hỗ trợ mạnh mẽ
- Cú pháp dễ đọc và học
- Tích hợp với HTML và CSS
- Khả năng xử lý bất đồng bộ

c. Nhược điểm:

- Phụ thuộc vào trình duyệt
- Bảo mật kém: nếu không kiểm soát và xử lý đầu vào
- Độ phức tạp
- Hiệu suất : do chạy trực tiếp trên trình duyệt vì vậy phụ thuộc vào tốc độ xử lý của trình duyệt
- Khó khăn trong việc gỡ lỗi.

3.1.4. Giới thiệu về Bootstrap.

a. Tổng quan:

Bootstrap là một framework front-end mã nguồn mở được phát triển bởi Twitter, giúp tạo ra các giao diện web hiện đại và responsive một cách nhanh chóng và dễ dàng.

Bootstrap được thiết kế để giúp các nhà phát triển web xây dựng các trang web và ứng dụng web với giao diện thân thiện, linh hoạt và responsive. Nó cung cấp một bộ công cụ và các thành phần giao diện tiêu chuẩn để giúp tăng tốc quá trình phát triển và đảm bảo tính nhất quán trong giao diện của ứng dụng.

b. Các tính năng chính:

- Grid System: Bootstrap cung cấp một hệ thống lưới linh hoạt, giúp căn chỉnh các thành phần của trang web trên các thiết bị với kích thước màn hình khác nhau.
- CSS Components: Bootstrap đi kèm với một loạt các thành phần CSS mà bạn có thể sử dụng để tạo ra nút, biểu mẫu, thẻ, thanh điều hướng, hình ảnh, vv.
- JavaScript Plugins: Ngoài ra, Bootstrap còn đi kèm với một số plugin JavaScript như Carousel, Modal, Scrollspy, Tooltip, vv, giúp thêm các tính năng tương tác động vào trang web của bạn.
- Responsive Design: Bootstrap hỗ trợ thiết kế responsive, tức là các trang web được xây dựng bằng Bootstrap sẽ tự động thích ứng với kích thước màn hình của người dùng.

3.1.5. HTML và CSS.

a. HTML

HTML (Hypertext Markup Language) là ngôn ngữ đánh dấu sử dụng để tạo cấu trúc và hiển thị nội dung trên trang web. HTML là một phần quan trọng của các trang web và là ngôn ngữ cơ bản cho việc phát triển nội dung web.

- **Ưu điểm:**
 - Dễ đọc và sử dụng.
 - Phổ biến và tiêu chuẩn.
 - Dễ duyệt và hiểu.
 - Hỗ trợ đa phương tiện.
 - Tích hợp và nhiều ngôn ngữ khác: CSS, JS,...
- **Nhược điểm:**
 - Giới hạn trong việc mô tả giao diện.
 - Không hỗ trợ lập trình nâng cao.
 - Không thể tạo ứng dụng động phức tạp.
 - Phải cập nhật thường xuyên.

b. CSS

CSS hay Cascading Style Sheets là một ngôn ngữ định dạng được sử dụng để làm đẹp cho các trang web. Nó được sử dụng để điều chỉnh các đặc điểm của các phần tử HTML như font chữ, kích thước, màu sắc,...

Cách mà CSS hoạt động chính là tìm kiếm dựa trên các vùng chọn thẻ như HTML, ID, class,... Sau đó, nó sẽ áp dụng những thuộc tính cần thay đổi lên các vùng đã chọn.

CSS là ngôn ngữ riêng biệt nhưng thường nó được sử dụng cùng với HTML để tạo ra những giao diện đẹp mắt cho các trang web. CSS và HTML thường có mối quan hệ chặt chẽ với nhau. Nếu như HTML tạo nên nền tảng cho website thì CSS lại giúp định hình phong cách cho website đó.

- **Ưu điểm:**
 - Tách biệt cấu trúc và giao diện
 - Tích hợp dễ dàng
 - Hiệu suất
 - Khả năng kế thừa
 - Kiểm soát tùy chỉnh chi tiết
 - Khả năng đáp ứng(Responsive)
- **Nhược điểm:**
 - Phức tạp.
 - Khả năng xung đột
 - Quản lý và bảo trì khó khăn với các dự án lớn.
 - Một số chức năng khá phức tạp

3.1.6. Hệ quản trị dữ liệu MySQL

a. SQL là gì ?

SQL là viết tắt của Structured Query Language là ngôn ngữ truy vấn dữ liệu mang tính cấu trúc. SQL là loại ngôn ngữ máy tính phổ biến để tạo, sửa và lấy dữ liệu từ một hệ quản trị cơ sở dữ liệu quan hệ như: MySQL, Oracle, Database, MySQL Server.

b. MySQL là gì?

MySQL là hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) mã nguồn mở phổ biến nhất hiện nay. Nó được sử dụng rộng rãi để lưu trữ dữ liệu cho các ứng dụng web, ứng dụng di động, và nhiều loại ứng dụng khác. Nó là phần mềm mã nguồn mở được hỗ trợ bởi Oracle. Nó có nghĩa là bạn có thể sử dụng MySQL mà không phải trả phí. Ngoài ra, bạn có thể thay đổi mã nguồn của nó để phù hợp với nhu cầu của mình.

- **Ưu điểm của MySQL:**
 - **An toàn:** Vì MySQL sở hữu nhiều tính năng bảo mật cấp cao, mã hóa thông tin đăng nhập và chứng thực từ host đều khả dụng.
 - **Dễ sử dụng:** MySQL ổn định và dễ sử dụng trên nhiều hệ điều hành và cung cấp một hệ thống các hàm tiện ích lớn.

- **Khả năng mở rộng:** Với MySQL có thể xử lý rất nhiều dữ liệu và hơn thế nữa có thể mở rộng khi cần thiết.
- **Hiệu năng cao:** Hỗ trợ nhiều chức năng SQL được mong chờ từ một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS).
- Nhược điểm của MySQL:
 - Giới hạn: Nó vẫn bị hạn chế về một số chức năng cần thiết.
 - Dung lượng hạn chế: Nếu số bản khi càng lớn thì việc truy xuất dữ liệu sẽ phức tạp và khó khăn hơn. Bạn cần phải áp dụng nhiều thủ thuật để nâng cấp tốc độ truy xuất dữ liệu lên.

3.2. Xây dựng chương trình

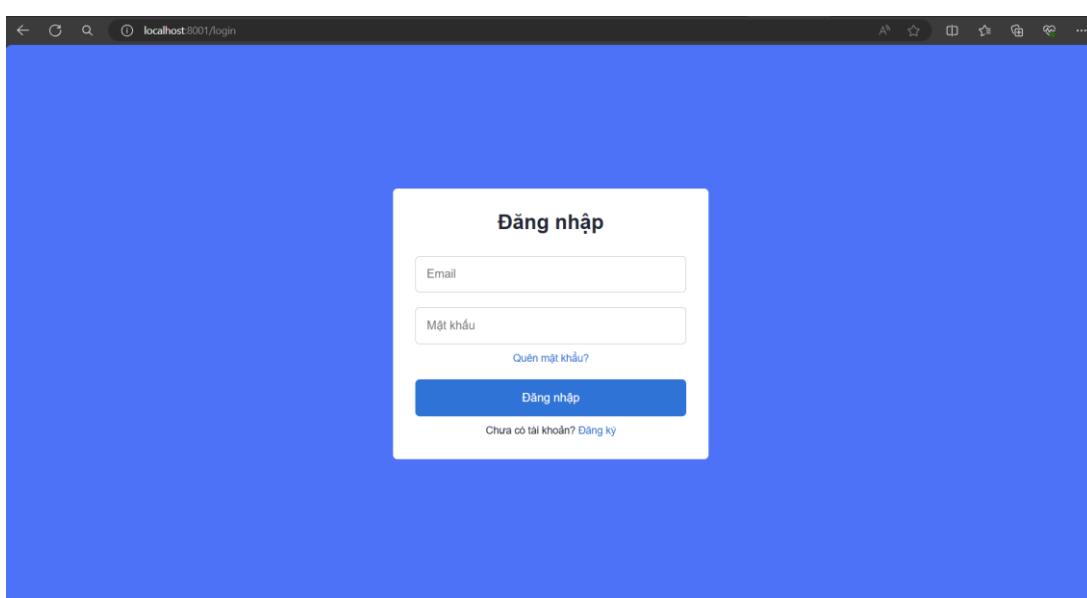
3.2.1. Giao diện người dùng

- **Giao diện trang đăng nhập:**

Ở giao diện đăng nhập người dùng sẽ phải nhập đúng tài khoản và mật khẩu của mình theo đúng định dạng. Nếu tài khoản hợp lệ và đã đăng ký trước đó thì ứng dụng sẽ điều hướng người dùng về màn hình trang chủ. Ngược lại nếu tài khoản này chưa tồn tại (chưa được đăng ký) hoặc thông tin nhập vào không đúng định dạng thì ứng dụng sẽ thông báo lỗi.

Nếu tài khoản có quyền admin tức quản trị viên thì người dùng sẽ được chuyển sang trang quản trị viên.

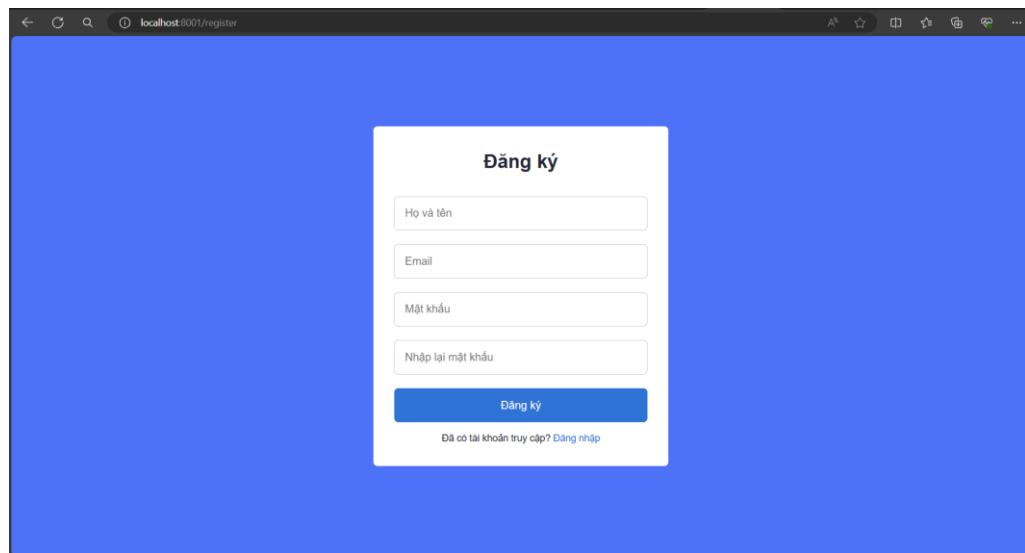
Nếu chưa có tài khoản bạn có thể nhấn vào ‘Đăng ký’ để chuyển sang trang đăng ký tài khoản mới.



Hình 3.1. Giao diện đăng nhập.

- **Giao diện trang đăng ký:**

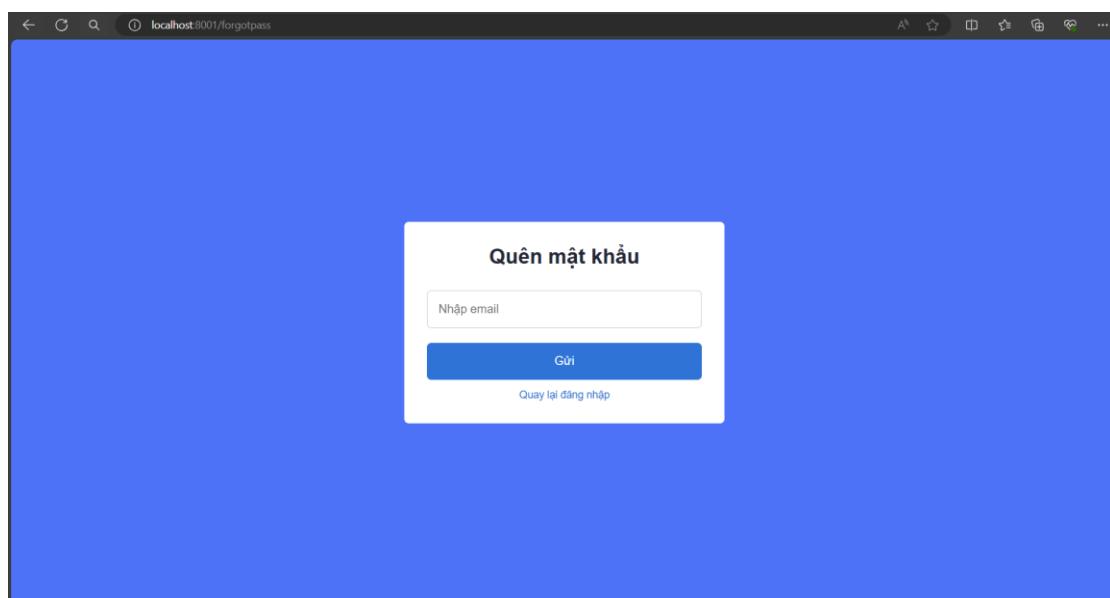
Ở giao diện đăng ký người dùng sẽ phải nhập thông tin tài khoản mới theo đúng định dạng. Nếu email đã đăng ký trước đó thì website đưa ra thông báo. Ngược lại nếu email chưa được đăng ký và các thông tin nhập đúng thì tài khoản mới được tạo và người dùng được chuyển sang trang đăng nhập để đăng nhập vào website.



Hình 3.2. Giao diện trang đăng ký tài khoản..

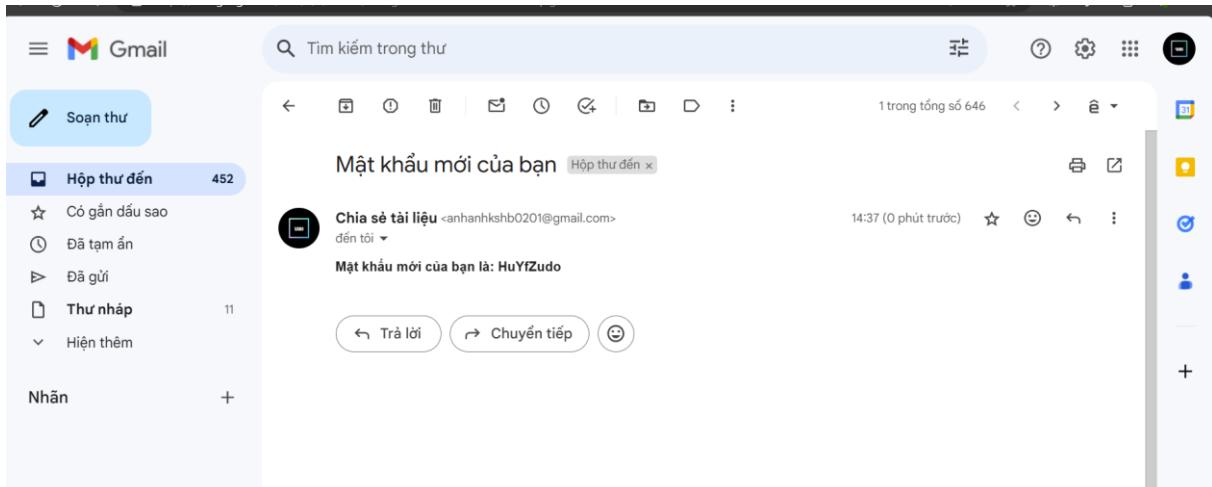
- **Giao diện trang quên mật khẩu:**

Khi người dùng vô tình quên mất mật khẩu của tài khoản thì người dùng đến trang quên mật khẩu và nhập vào email mà mình dùng để đăng ký tài khoản trước đó. Nếu nhập sai định dạng hoặc email không đúng, hệ thống sẽ báo lỗi.



Hình 3.3. Giao diện trang quên mật khẩu.

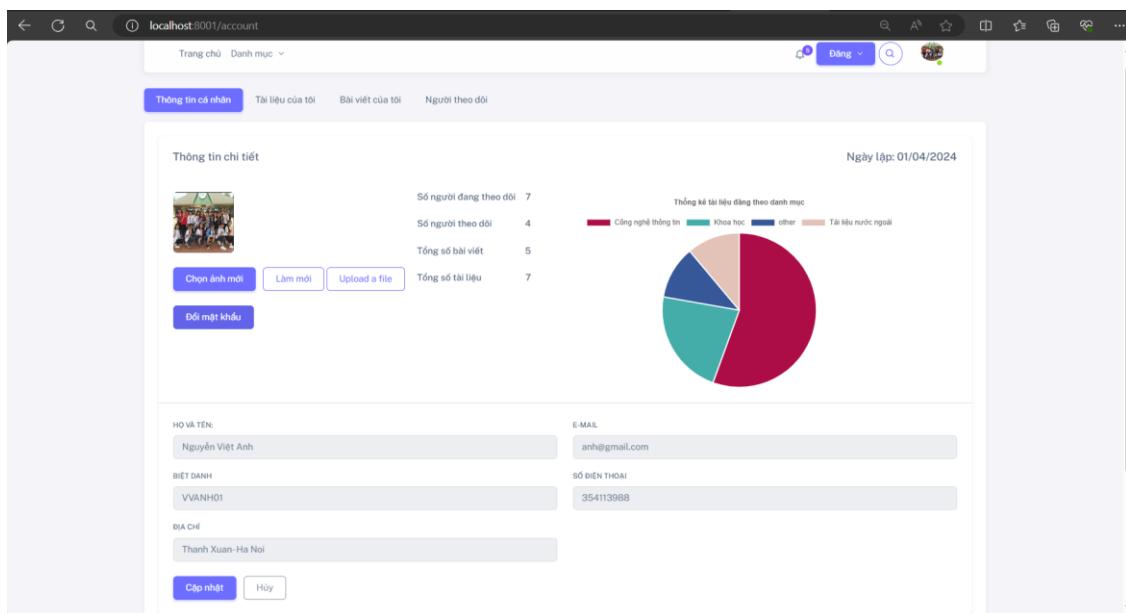
Sau khi nhập đúng email, hệ thống sẽ gửi mật khẩu mới về email đã nhập.



Hình 3.4. Mật khẩu mới gửi về email.

- **Giao diện trang thông tin cá nhân:**

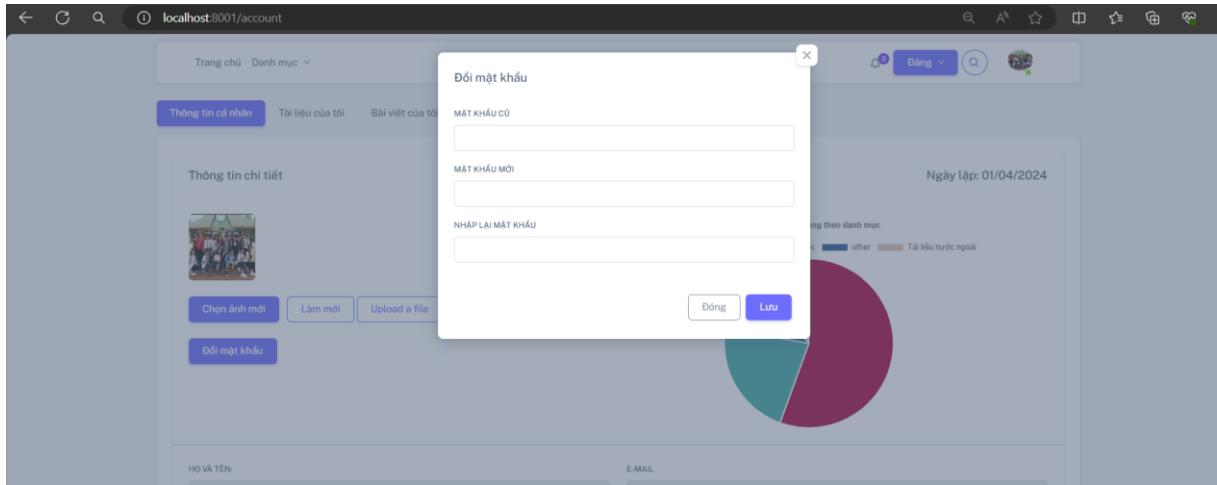
Hiển thị thông tin của người dùng hiện tại đang đăng nhập trên hệ thống. Người dùng có thể cập nhật lại ảnh đại diện cũng như các thông tin khác của tài khoản như họ và tên, địa chỉ,... Ngoài ra, người dùng có thể quản lý tất cả các tài liệu, bài đăng của mình.



Hình 3.5. Giao diện trang thông tin cá nhân.

Để đổi mật khẩu, người dùng click vào nút “Đổi mật khẩu” ứng dụng sẽ hiển thị một hộp thoại cho phép người dùng nhập thông tin cần thiết để đổi mật khẩu.

Người dùng nhập mật khẩu cũ và mật khẩu mới và click vào nút đổi mật khẩu. Ứng dụng sẽ thông báo thao tác của người dùng có thành công hay không. Nếu thành công, ứng dụng ẩn hộp thoại đổi mật khẩu, cập nhật mật khẩu mới và đăng nhập lại hệ thống.



Hình 3.6. Giao diện chức năng đổi mật khẩu.

- **Giao diện trang chủ**

Hiển thị các tài liệu mới nhất, các tài liệu nổi bật có lượt truy cập cao nhất và hiển thị các bài viết mới nhất cũng như nhiều lượt đọc nhất. Khi người dùng nhấp vào một tài liệu hoặc bài viết người dùng có thể xem chi tiết tài liệu cũng như chi tiết bài viết.

The screenshot displays the homepage of a website dedicated to document sharing. The layout is organized into several sections:

- Tài liệu mới nhất:** A grid of eight document thumbnails, each with a title, download links (docx, pdf), and a small preview image.
- Tài liệu nổi bật:** A grid of eight document thumbnails, each with a title, download links (docx, pdf), and a small preview image.
- Bài viết nổi bật:** A grid of eight article thumbnails, each featuring a logo or image, a title, a brief description, and a "Xem chi tiết" button.
- Bài viết mới nhất:** A grid of five article thumbnails, each featuring a photo of a person, a title, a brief description, and a "Xem chi tiết" button.

On the left side, there is a sidebar with a navigation menu titled "DANH MỤC TÀI LIỆU" containing categories such as Công nghệ thông tin, Khoa học, Tài liệu nước ngoài, Bài giảng, Nghiên cứu khoa học, Đô án, Kỹ thuật - Công nghệ, and Kinh tế.

Hình 3.7. Giao diện trang chủ.

- **Giao diện trang upload tài liệu:**

Giao diện cho phép người dùng nhập các thông tin để upload một tài liệu mới. Sau khi người dùng nhập đầy đủ các thông tin và nhấn nút ‘Xác nhận’, hệ thống sẽ thêm mới tài liệu và chuyển về trang cá nhân người dùng.

Upload Tài liệu

Chia sẻ tài liệu của Bạn bằng cách Upload files để mọi người có thể xem, tải và kết nối cùng Bạn.

TÊN TÀI LIỆU
[Input field]

THỂ LOẠI
[Input field] Chọn thể loại

Choose File No file chosen

MÔ TẢ:
[Text area]

Xác nhận

Quy định upload

- LƯU Ý: Không upload tài liệu có bản quyền mà không có sự đồng ý của tác giả.
- Không upload tài liệu vi phạm thuần phong mỹ tục.
- Không được Upload các tài liệu có nội dung phản động.

Hình 3.8. Giao diện trang upload tài liệu.

- **Giao diện trang đăng bài viết:**

Giao diện cho phép người dùng nhập các thông tin để đăng mới bài viết. Sau khi người dùng nhập đầy đủ các thông tin và nhấn nút ‘Xác nhận’, hệ thống sẽ thêm mới bài viết và chuyển về trang cá nhân người dùng.

localhost:8001/uploadpost

Trang chủ Danh mục Đăng

TÊN BÀI VIẾT
[Input field]

THẺ TAG (TỐI ĐA 3 CÁI)
[Input field]

CHỌN ẢNH
Choose File No file chosen

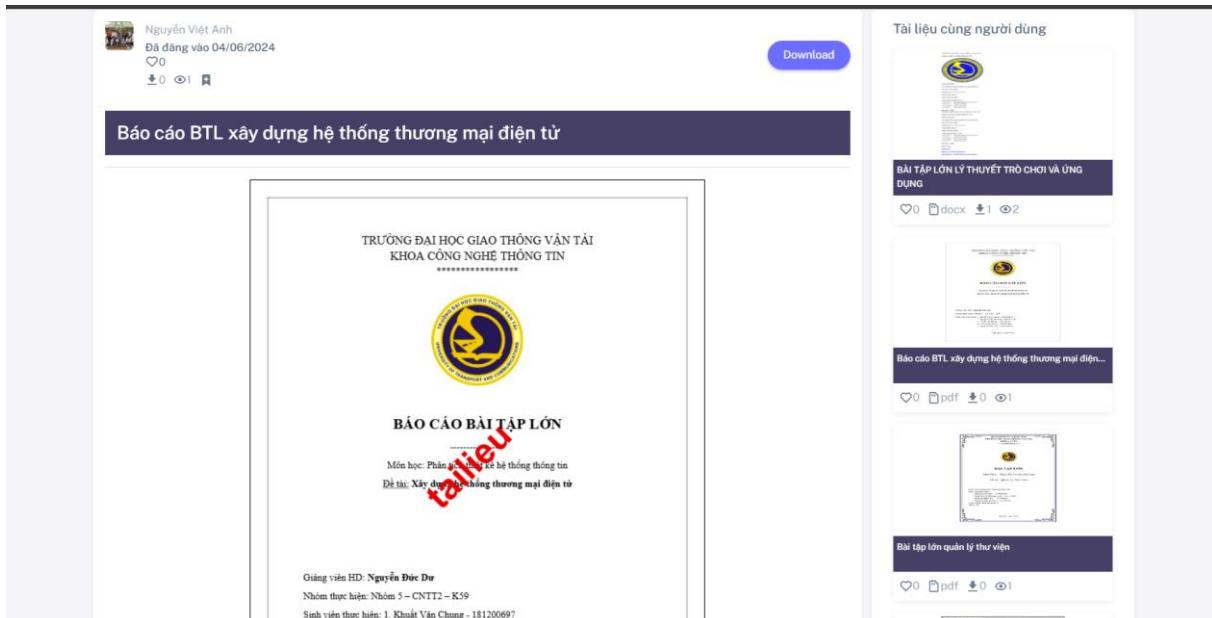
NỘI DUNG
[Rich text editor toolbar] [Content area]

Xác nhận

Hình 3.9. Giao diện trang đăng bài viết.

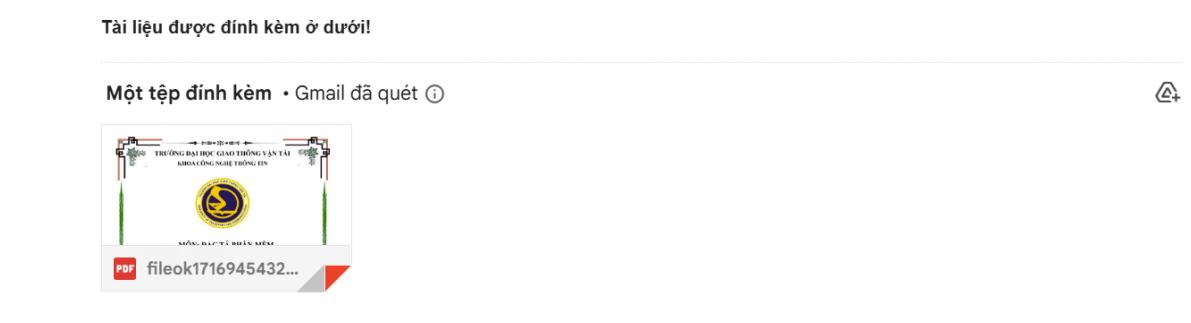
- **Giao diện trang chi tiết tài liệu**

Hiển thị các thông tin như: tên tài liệu, người đăng tài liệu, số lượt tải về, thời gian upload, số lượt đọc, số lượt thích, loại tài liệu, hiển thị một phần nội dung của tài liệu. Người dùng có thể báo cáo bài viết nếu thấy vi phạm. Hiển thị danh sách các bình luận của mọi người. Người dùng có thể thêm mới bình luận, chỉnh sửa bình luận hay xóa bình luận của bản thân. Ngoài ra người dùng còn có thể trả lời bình luận của người khác và thả tim bình luận cũng như thả tim tài liệu.



Hình 3.10. Giao diện trang chi tiết tài liệu.

Nếu muốn đọc hết nội dung của tài liệu thì người dùng phải download tài liệu về máy. Khi người dùng nhấn download thì tài liệu sẽ được gửi về email của người dùng.



Hình 3.11. Hình gửi mail sau khi download tài liệu.

- **Giao diện trang chi tiết bài viết**

Hiển thị thông tin chi tiết bài viết như thông tin người viết, số lượt đọc, lượt thích, số bình luận, tên bài viết, nội dung bài viết, danh sách các bình luận của mọi người. Người dùng có thể thêm mới bình luận, chỉnh sửa bình luận hay xóa bình luận của bản

thân. Ngoài ra người dùng còn có thể trả lời bình luận của người khác và thả tim bình luận cũng như thả tim bài viết.

7 Things You Should Know to Become a Better Node.js Developer

o become a better Node.js developer, you need to have a combination of technical knowledge, following best practices, and having good interpersonal skills. Here are some suggestions to help you improve your Node.js development abilities:

1. Learn the basic concepts of JavaScript.

If you want to become a better Node.js developer, it is important to have a good understanding of JavaScript, since Node.js is based on JavaScript. Here are some resources to help you learn the basics of JavaScript.

- [JavaScript for Beginners](#) — a comprehensive tutorial on the basics of JavaScript, including variables, data types, functions, and control structures.
- [MDN Web Docs: JavaScript](#) — a comprehensive resource on JavaScript, with documentation on all aspects of the language, including its syntax, operators, and objects.
- [Eloquent JavaScript](#) — a free online book that provides a detailed introduction to JavaScript and its features.

2. Master asynchronous programming

Node.js is good at dealing with lots of requests and connections at the same time, which is done using a type of programming called asynchronous programming. Here are some links to help you learn how to do asynchronous programming with Node.js.

- [Asynchronous programming in JavaScript](#) — a tutorial on asynchronous programming in JavaScript, including callbacks, promises, and async/await.
- [Node.js asynchronous programming](#) — a tutorial on asynchronous programming in Node.js, including the event loop, callbacks, and promises.
- [Understanding the Node.js Event Loop](#) — an in-depth article on how the Node.js event loop works and how it enables asynchronous programming.

Bài viết liên quan

- Tại sao hướng dẫn sử dụng Node.js MySQL
- Tại sao hướng dẫn sử dụng Node.js MySQL, Node.js... Người đăng Trần Văn Tùng 23 ngày trước
- Câu chuyện muốn thuở của SQL và NoSQL
- SQL và NoSQL là chủ đề không còn xa lạ trong th... Người đăng Nguyễn Việt Anh 20 ngày trước
- Xử lý đồng thời 10.000 Request trong Node.js
- Khi bạn mới học Node.js, một trong những câu hỏi... Người đăng VVANH 6 ngày trước
- Nvidia ra trợ lý AI cho game thủ
- AI G-Assist của Nvidia hỗ trợ game thủ th... Người đăng Ngọc Anh 8 giờ trước
- Kiểm thử phần mềm
- Kiểm thử là gì? Các mức độ kiểm thử
- Xin chào các bạn lại là mình đây. Hôm nay mình ... Người đăng Ngọc Anh

Hình 3.12. Giao diện trang chi tiết bài viết.

- **Giao diện trang thông tin người dùng:**

Hiển thị thông tin người dùng khác. Bạn có thể xem được danh sách các bài viết của họ và có thể nhấn theo dõi hoặc hủy theo dõi.

Thông tin

Nguyễn Việt Anh

Theo dõi

Số người đang theo dõi: 7

Số người theo dõi: 4

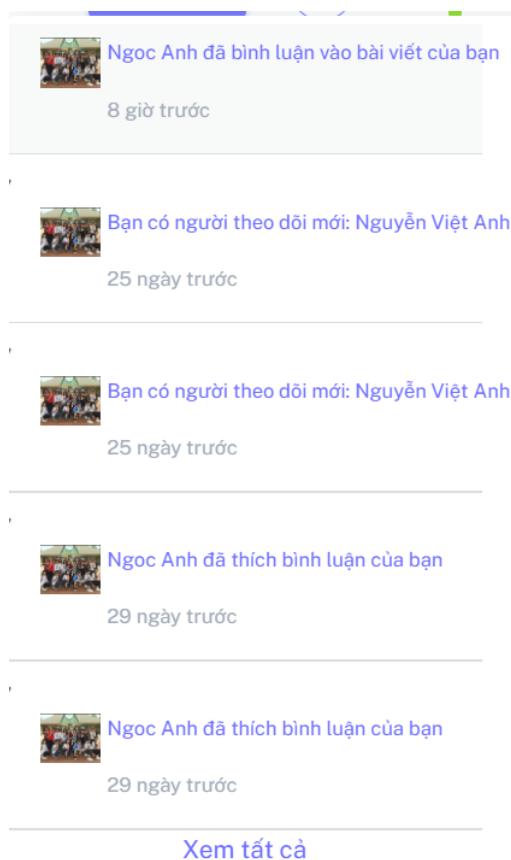
Danh sách bài viết

| Tác giả | Thời gian | Tiêu đề | Lượt thích |
|-----------------|------------------|---------------------------------------|------------|
| Nguyễn Việt Anh | 15th May, 2024 | Câu chuyện muốn thuở của SQL và NoSQL | 0 |
| Nguyễn Việt Anh | 15th May, 2024 | Tim hiểu Express.js Framework | 0 |
| Nguyễn Việt Anh | 30th April, 2024 | Test | 0 |
| Nguyễn Việt Anh | 30th April, 2024 | Test | 0 |
| Nguyễn Việt Anh | 6th April, 2022 | add | 0 |

Hình 3.13. Giao diện trang thông tin người dùng khác.

- **Giao diện danh sách thông báo:**

Mỗi khi có người theo dõi, bình luận vào bài viết hoặc tài liệu thì người dùng sẽ nhận được thông báo mới.

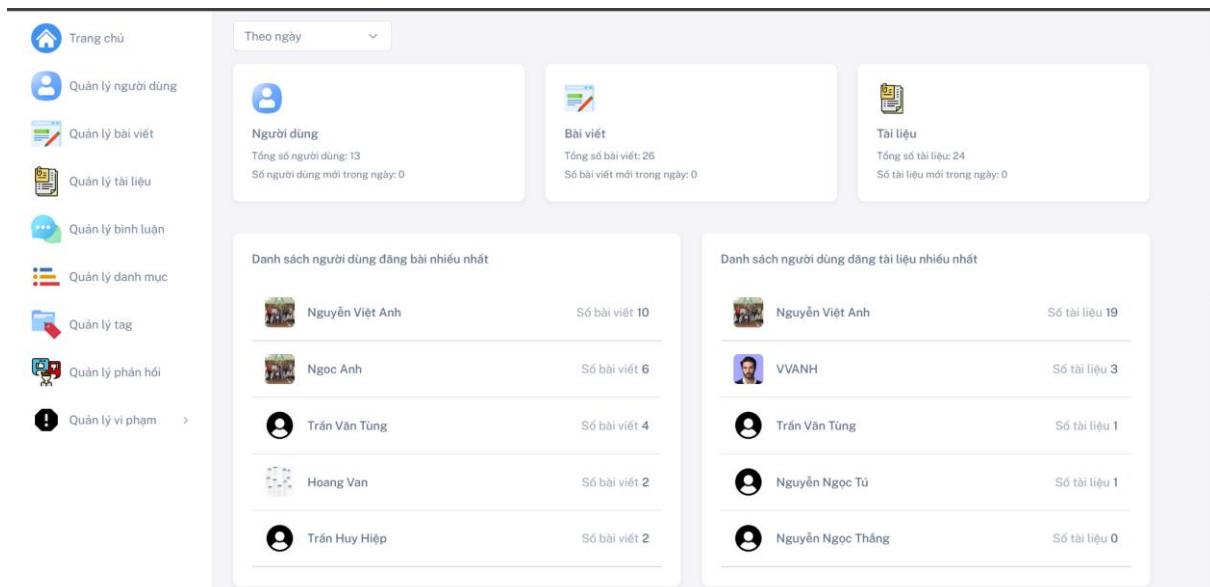


Hình 3.14. Danh sách thông báo.

3.2.2. Giao diện người quản trị viên

- **Giao diện trang thống kê báo cáo:**

Giao diện hiển thị các thống kê về người dùng, tài liệu và về bài viết. Người dùng có thể chọn xem thống kê theo các tiêu chí khoảng thời gian. Click chuột để chọn thống kê theo các mốc thời gian như trong ngày, trong tháng hoặc trong năm.



Hình 3.15. Giao diện trang thống kê (1).



Hình 3.16. Giao diện trang thống kê (2).

• Giao diện trang quản lý người dùng:

Giao diện hiển thị danh sách người dùng trong ứng dụng cùng với một số thông tin cơ bản. Quản trị viên có thể tìm kiếm trong danh sách các người dùng với tiêu chí: Tên người dùng. Khi người quản trị viên click vào ô checkbox thì sẽ khóa tài khoản đó. Người dùng sẽ nhận được email thông báo khóa tài khoản và không thể đăng nhập vào hệ thống. Cuối cùng, người dùng có thể click vào biểu tượng thùng rác để xóa người dùng.

The screenshot shows a sidebar menu on the left with icons and labels: Trang chủ, Quản lý người dùng, Quản lý bài viết, Quản lý tài liệu, Quản lý bình luận, Quản lý danh mục, Quản lý tag, Quản lý phản hồi, and Quản lý vi phạm. The main area is titled "Danh sách / Người dùng" and shows a table for managing users. The table has columns: ID, TÊN (Name), EMAIL, KHÓA TÀI KHOẢN (Account Status), and THAO TÁC (Actions). There are 15 users listed, each with a checkbox in the account status column and a red trash can icon in the actions column. The last user in the list has a checked checkbox in the account status column.

| ID | TÊN | EMAIL | KHÓA TÀI KHOẢN | THAO TÁC |
|----|-----------------|--------------------------|-------------------------------------|----------|
| 1 | Nguyễn Việt Anh | anh@gmail.com | <input type="checkbox"/> | |
| 3 | Ngoc Anh | ngoc@gmail.com | <input type="checkbox"/> | |
| 4 | Trần Huy Hiệp | hiep@gmail.com | <input type="checkbox"/> | |
| 5 | Phạm Công Định | dinh@gmail.com | <input type="checkbox"/> | |
| 6 | Quốc Tuấn | tuan@gmail.com | <input type="checkbox"/> | |
| 7 | Nguyễn Ngọc Tu | tu@gmail.com | <input type="checkbox"/> | |
| 10 | Nguyễn Việt Anh | anh1@gmail.com | <input type="checkbox"/> | |
| 13 | VVANH | anhanhkshb0201@gmail.com | <input type="checkbox"/> | |
| 14 | Trần Văn Tùng | tung@gmail.com | <input type="checkbox"/> | |
| 15 | Nguyễn Ngọc Tú | ngoctu15032008@gmail.com | <input checked="" type="checkbox"/> | |

Hình 3.17. Giao diện trang quản lý người dùng.

- **Giao diện trang quản lý tài liệu:**

Giao diện hiển thị danh sách các tài liệu hiện có trong hệ thống. Người quản trị viên có thể tìm kiếm tài liệu với tiêu chí tên tài liệu. Khi người dùng click vào biểu tượng con xem, hệ thống chuyển đến giao diện chi tiết tài liệu. Cuối cùng, quản trị viên có thể click vào biểu tượng thùng rác để xóa tài liệu.

The screenshot shows a sidebar menu on the left with icons and labels: Trang chủ, Quản lý người dùng, Quản lý bài viết, Quản lý tài liệu, Quản lý bình luận, Quản lý danh mục, Quản lý tag, Quản lý phản hồi, and Quản lý vi phạm. The main area is titled "Danh sách / Tài liệu" and shows a table for managing documents. The table has columns: ID, TÊN TÀI LIỆU (Name), NGƯỜI DÙNG (Author), LOẠI TÀI LIỆU (Type), LUỢT XEM (Views), LUỢT TẢI XUỐNG (Downloads), and LINK. There are 18 documents listed, each with a red trash can icon and a blue file icon in the actions column.

| ID | TÊN TÀI LIỆU | NGƯỜI DÙNG | LOẠI TÀI LIỆU | LUỢT XEM | LUỢT TẢI XUỐNG | LINK |
|----|--|-----------------|---------------|----------|----------------|------|
| 5 | Website giới thiệu bản thân | Nguyễn Việt Anh | pdf | 7 | 13 | |
| 6 | Internet of Things | Nguyễn Việt Anh | pdf | 2 | 0 | |
| 7 | ĐỀ TÀI XÂY DỰNG WEBSITE ĐẶT PHÒNG KHÁCH SẠN | Nguyễn Việt Anh | pdf | 15 | 1 | |
| 11 | ĐỀ TÀI QUẢN LÝ BÁN DIỄN THOẠI | Nguyễn Việt Anh | pdf | 17 | 0 | |
| 14 | Bài tập lớn Đặc tả website xem phim trực tuyến | VVANH | pdf | 20 | 2 | |

Hình 3.18. Giao diện trang quản lý tài liệu.

- **Trang quản lý bài viết:**

Giao diện hiển thị danh sách các bài viết hiện có trong hệ thống. Người quản trị viên có thể tìm kiếm bài viết với tiêu chí tên bài viết. Khi quản trị viên click vào nút “Thêm mới”, hệ thống sẽ chuyển sang trang thêm mới bài viết. Khi quản trị viên click vào biểu tượng xem, hệ thống chuyển đến trang chi tiết bài viết. Cuối cùng, quản trị viên có thể click vào biểu tượng thùng rác để xóa bài viết.

The screenshot shows a web application interface for managing articles. On the left is a sidebar with icons for Trang chủ, Quản lý người dùng, Quản lý bài viết, Quản lý tài liệu, Quản lý bình luận, Quản lý danh mục, Quản lý tag, Quản lý phản hồi, and Quản lý vi phạm. The main area has a search bar at the top labeled "Nhập tên bài viết". Below it is a breadcrumb navigation "Danh sách / Bài viết". A green button labeled "Thêm mới" is visible. The main content area displays a table titled "Bài viết (18)" with columns: ID, TÊN BÀI VIẾT, NGƯỜI DÙNG, and THAO TÁC. The data in the table is as follows:

| ID | TÊN BÀI VIẾT | NGƯỜI DÙNG | THAO TÁC |
|----|---|-----------------|----------|
| 1 | Câu chuyện muôn thuở của SQL và NoSQL | Nguyễn Việt Anh | |
| 2 | Tim hiểu Express js Framework | Nguyễn Việt Anh | |
| 5 | 7 Things You Should Know to Become a Better Node.js Developer | Nguyễn Việt Anh | |
| 6 | Kết nối với cơ sở dữ liệu (mysql) trong nodejs | Nguyễn Việt Anh | |
| 15 | Google giải thích tình trạng AI liên tục trả lời sai | Nguyễn Việt Anh | |

Pagination controls at the bottom show pages 1 through 4.

Hình 3.19. Giao diện trang quản lý bài viết.

- **Giao diện trang quản lý bình luận:**

Giao diện hiển thị danh sách các bình luận. Quản trị viên có thể xem danh sách các bình luận và quản trị viên có thể click vào biểu tượng thùng rác để xóa bình luận.

The screenshot shows a web application interface for managing comments. On the left is a sidebar with icons for Trang chủ, Quản lý người dùng, Quản lý bài viết, Quản lý tài liệu, Quản lý bình luận, Quản lý danh mục, Quản lý tag, Quản lý phản hồi, and Quản lý vi phạm. The main area has a breadcrumb navigation "Danh sách / Bình luận". A green button labeled "Thêm mới" is visible. The main content area displays a table titled "Bình luận(37)" with columns: ID, NGƯỜI DÙNG, NỘI DUNG, TRẠNG THÁI, and THAO TÁC. The data in the table is as follows:

| ID | NGƯỜI DÙNG | NỘI DUNG | TRẠNG THÁI | THAO TÁC |
|----|-----------------|--|------------|----------|
| 64 | Trần Văn Tùng | Cảm ơn bạn nhiều | Hiển thị | |
| 63 | Ngọc Anh | Tài liệu thực sự rất hay ! | Hiển thị | |
| 62 | Ngọc Anh | VN cũng thuộc hàng máu me trong lĩnh vực tiền số | Hiển thị | |
| 58 | Ngọc Anh | Hay quá ạ | Hiển thị | |
| 56 | VVANH | Cảm ơn bạn nhiều | Hiển thị | |
| 55 | Nguyễn Việt Anh | Cảm ơn bạn, tài liệu rất hữu ích | Hiển thị | |
| 54 | VVANH | Cảm ơn bạn | Hiển thị | |
| 53 | Trần Văn Tùng | Bài viết rất hay | Hiển thị | |
| 52 | VVANH | Hay lắm | Hiển thị | |
| 48 | Nguyễn Việt Anh | a | Hiển thị | |

Pagination controls at the bottom show pages 1 through 4.

Hình 3.20. Giao diện trang quản lý bình luận người dùng.

- **Giao diện trang quản lý danh mục:**

Giao diện hiển thị danh sách các danh mục trong hệ thống. Khi quản trị viên click vào nút “Thêm mới”, hệ thống sẽ mở một hộp thoại để thêm mới danh mục. Quản trị viên click vào biểu tượng cập nhật, hệ thống sẽ mở một hộp thoại để có thể cập nhật lại

tên danh mục. Cuối cùng, quản trị viên có thể click vào biểu tượng thùng rác để xóa danh mục đi.

| ID | TÊN DANH MỤC | THAO TÁC |
|----|---------------------|----------|
| 1 | Công nghệ thông tin | |
| 2 | Khoa học | |
| 3 | CTDL-GT | |
| 4 | Tài liệu nước ngoài | |
| 5 | Cơ sở dữ liệu | |

Hình 3.21. Giao diện trang quản lý danh mục tài liệu.

- **Giao diện trang quản lý tag:**

Giao diện hiển thị danh sách các tag của bài viết trong hệ thống. Khi quản trị viên click vào nút “Thêm mới”, hệ thống sẽ mở một hộp thoại để thêm mới tag. Quản trị viên click vào biểu tượng cập nhật, hệ thống sẽ mở một hộp thoại để có thể cập nhật lại tên tag. Cuối cùng, quản trị viên có thể click vào biểu tượng thùng rác để xóa tag đi.

| ID | TÊN TAG | THAO TÁC |
|----|------------------|----------|
| 12 | Blockchain | |
| 13 | AI | |
| 14 | Lập trình | |
| 15 | Machine learning | |
| 16 | Học tiếng anh | |

Hình 3.22. Giao diện trang quản lý tag bài viết

- **Giao diện trang quản lý báo cáo vi phạm:**

Hiển thị danh sách các báo cáo tài liệu vi phạm bản quyền, nội dung sai lệch từ người dùng.

The screenshot shows a user interface for managing violation reports. On the left, a sidebar lists various administrative functions with corresponding icons:

- Trang chủ
- Quản lý người dùng
- Quản lý bài viết
- Quản lý tài liệu
- Quản lý bình luận
- Quản lý danh mục
- Quản lý tag
- Quản lý phản hồi
- Quản lý vi phạm >

The main content area is titled "Danh sách / Báo cáo vi phạm từ người dùng". It displays a table with two rows of data:

| ID | NGƯỜI BÁO CÁO VI PHẠM | TÀI LIỆU VI PHẠM | LÝ DO VI PHẠM | THỜI GIAN | TRẠNG THÁI | THAO TÁC |
|----|-----------------------|--------------------------------------|-------------------|------------|------------|------------------------|
| 4 | Nguyễn Việt Anh | ĐỀ TÀI QUẢN LÝ BẢN ĐIỆN THOẠI | Vi phạm bản quyền | 30/05/2024 | pending | Review |
| 5 | Nguyễn Việt Anh | Tài liệu tiếng anh chuyên ngành CNTT | Nội dung sai lệch | 30/05/2024 | pending | Review |

Hình 3.23. Giao diện trang quản lý báo cáo vi phạm.

Sau khi quản trị viên nhận được các báo cáo vi phạm bản quyền, quy trình xử lý có thể bao gồm các bước sau:

- Xem danh sách các báo cáo: Quản trị viên có thể xem danh sách các báo cáo vi phạm bản quyền.
- Xem chi tiết báo cáo: Quản trị viên có thể xem chi tiết của từng báo cáo, bao gồm thông tin về tài liệu, người báo cáo và lý do báo cáo.
- Xác minh báo cáo: Sau khi xem báo cáo vi phạm, quản trị viên sẽ gửi email đến người báo cáo yêu cầu họ cung cấp thêm các bằng chứng chứng minh tài liệu vi phạm như **hình 3.24**. Nếu báo cáo là chính xác và có vi phạm, quản trị viên sẽ gỡ bỏ tài liệu vi phạm khỏi hệ thống, khóa tài khoản của người tải lên tài liệu vi phạm, gửi email thông báo kết quả cho người báo cáo vi phạm.

The screenshot shows an email inbox with the following details:

Document Sharing Site <anhanhkshb0201@gmail.com>
đến tôi ▾

Kính gửi VVANH,

Chúng tôi đã nhận được báo cáo từ bạn cho tài liệu: Bài tập lớn quản lý thư viện . Lý do: Vi phạm bản quyền.

Để có thể tiến hành xử lý, chúng tôi yêu cầu bạn cung cấp các bằng chứng minh chứng tài liệu đó là của bạn. Ví dụ như đường dẫn tới tài liệu gốc, ảnh chụp màn hình, hoặc mô tả chi tiết vi phạm,...

.Gửi minh chứng về email: documentsharing@gmail.com.

Cảm ơn bạn đã sử dụng dịch vụ của chúng tôi.

Best regards,
Document Sharing Site Team

Hình 3.24. Yêu cầu cung cấp bằng chứng vi phạm.

KẾT LUẬN VÀ KIẾN NGHỊ

Sau một thời gian tập trung nghiên cứu và triển khai đề tài, em đã hoàn thành được đề tài “Xây dựng website chia sẻ tài liệu” với giao diện tương đối đẹp, các thao tác sử dụng dễ dàng với người dùng. Các chức năng hoạt động tốt, thực hiện đúng quy trình nghiệp vụ của hệ thống. Giao diện gần gũi, thao tác thuận lợi, thân thiện với người dùng. Mức độ bảo mật dữ liệu ở mức tương đối.

Khi thực hiện và hoàn thành đồ án, em cũng đã biết thêm được nhiều kinh nghiệm quý giá khi xây dựng website và phát triển phần mềm. Những kỹ năng này sẽ nền tảng để giúp em nâng cao trình độ bản thân, kinh nghiệm trong thực tế để có thể làm việc ngoài các doanh nghiệp sau này.

- Nhữn tồn tại, hạn chế cần khắc phục:
 - Các chức năng còn đơn giản, chưa có chức năng để người dùng chia sẻ thông tin.
 - Web chưa hỗ trợ thanh toán trực tuyến cho người dùng.
 - Tốc độ xử lý chưa cao, vẫn còn một số lỗi phát sinh khi vận hành hệ thống.
 - Website còn ở mức cơ bản, các chức năng nâng cao chưa được triển khai.
- Đề xuất giải pháp khắc phục:
 - Nghiên cứu thiết kế và xây dựng các chức năng: chia sẻ thông tin, thanh toán trực tuyến cho người dùng trong tương lai.
 - Triển khai các chức năng nâng cao hơn, chuyên nghiệp hơn

Sau khi hoàn thành đồ án, em sẽ tiếp tục nghiên cứu và phát triển ứng dụng để khắc phục những hạn chế và mở rộng thêm các tính năng như:

- Phát triển hệ thống theo hướng hỗ trợ đa ngôn ngữ.
- Thêm các chức năng như nhắn tin trực tuyến, xây dựng AI tự động kiểm duyệt tài liệu và bài đăng của người dùng.
- Tối ưu hóa trải nghiệm người dùng: Tiếp tục tối ưu hóa giao diện người dùng và trải nghiệm sử dụng trang web.
- Tăng cường tính năng tìm kiếm và lọc: Nâng cao khả năng tìm kiếm và lọc để người dùng có thể nhanh chóng và dễ dàng tìm kiếm tài liệu hoặc bài viết hợp với nhu cầu của họ.

Một lần nữa em xin chân thành cảm ơn thầy Hoàng Văn Thông đã tận tình hướng dẫn em trong suốt quá trình thực hiện đồ án.

TÀI LIỆU THAM KHẢO

Tiếng Việt:

- [1] Slide dạy học môn Phân tích thiết kế hướng đối tượng – Ths. Nguyễn Hiếu Cường – Bộ môn Công nghệ phần mềm, Khoa Công nghệ thông tin, Trường Đại học Giao Thông vận tải.
- [2] Bài giảng môn “*Đặc tả phần mềm*” – TS. Trần Văn Dũng – Đại học Giao thông vận tải.
- [3] Ý Nhi. (2023, 04 06). *CSS là gì? Ngôn ngữ CSS có ưu điểm và ứng dụng gì ?*, <https://mikotech.vn/css-la-gi/>.

Tiếng Anh:

- [4] MySQL Tutorial. *What is MySQL ?*, <https://www.mysqltutorial.org/getting-started-with-mysql/what-is-mysql/>.
- [5] Express. <https://expressjs.com/en/5x/api.html>.
- [6] Wikipedia. *HTTP*, <https://en.wikipedia.org/wiki/HTTP>.
- [7] Nodejs.org. <https://nodejs.org/en/learn/getting-started/introduction-to-nodejs>.