

**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**  
**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**  
**KHOA CÔNG NGHỆ THÔNG TIN 1**

---



**ĐỒ ÁN**  
**TỐT NGHIỆP ĐẠI HỌC**

**ĐỀ TÀI:**  
**XÂY DỰNG HỆ THỐNG CHIA SẺ TÀI LIỆU THEO KIẾN TRÚC**  
**HƯỚNG DỊCH VỤ**

**Giảng viên hướng dẫn: TS. Đặng Ngọc Hùng**

**Sinh viên thực hiện: Phan Quốc Anh**

**Lớp: D19CNPM1**

**Mã sinh viên: B19DCCN040**

**Hệ đào tạo: Chính quy**

**Hà Nội - 2023**

## LỜI CẢM ƠN

Đầu tiên, em xin được gửi lời cảm ơn đến Ban Giám đốc Học Viện Công Nghệ Bưu chính Viễn thông đã tạo môi trường tốt cho em có thể học tập, tiếp thu được những kiến thức bổ ích trong những năm qua.

Em xin bày tỏ lòng biết ơn của em tới TS. Đặng Ngọc Hùng vì sự hướng dẫn tận tình, chu đáo, tạo điều kiện thuận lợi cho em trong quá trình thực hiện và hoàn thành đồ án tốt nghiệp.

Em xin cảm ơn tất cả các thầy các cô, đặc biệt là các thầy cô trong khoa Công nghệ thông tin 1 đã tận tình, chỉ dạy những kiến thức để em có thể hoàn thành được đồ án tốt nghiệp cũng như là hành trang để em bước lên con đường sự nghiệp sau này.

Cuối cùng, em xin chúc các thầy cô, gia đình và bạn bè sức khỏe dồi dào, hạnh phúc, thành công trong sự nghiệp.

Em xin chân thành cảm ơn!

Hà Nội, Ngày 31 tháng 12 năm 2023

Sinh viên

Phan Quốc Anh

**NHẬN XÉT, ĐÁNH GIÁ, CHO ĐIỂM**  
**(Của giảng viên phản biện)**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Điểm.....(Bằng chữ.....)

Hà Nội, Ngày.....tháng.....năm.....  
CÁN BỘ - GIẢNG VIÊN PHẢN BIỆN  
(Ký, họ tên)

**MỤC LỤC**

---

|   |  |
|---|--|
| <b>MỞ ĐẦU</b> .....   |  |
| <b>CHƯƠNG 1: GIỚI THIỆU VỀ ĐỀ TÀI, KIẾN TRÚC VÀ CÔNG NGHỆ SỬ DỤNG</b> ..... |  |
| <b>1.1. Đặt vấn đề</b> .....  |  |
| <b>1.2 Tìm hiểu các giải pháp</b> .....                                     |  |
| <b>1.2.1. Tổng quan về yêu cầu</b> .....                                    |  |
| 1.2.1.1. Giới thiệu về bài toán chia sẻ tài liệu.....                       |  |
| 1.2.1.2. Giới thiệu sơ qua về đề tài.....                                   |  |
| <b>1.2.2. Giải pháp</b> .....   |  |
| 1.2.2.1. Kiến trúc Monolithic.....  |  |
| 1.2.2.2. Kiến trúc Microservices.....                                       |  |
| 1.2.2.3. So sánh giữa Monolithic và Microservices.....                      |  |
| <b>1.3. Giải pháp</b> .....   |  |
| <b>1.3.1. Kiến trúc</b> .....   |  |
| 1.3.1.1. Khái niệm kiến trúc Microservices.....                             |  |
| 1.3.1.2. Một số đặc điểm chính của kiến trúc Microservices.....             |  |
| 1.3.1.3. Lợi ích của kiến trúc Microservices.....                           |  |
| 1.3.1.4. Ưu điểm của Microservices.....                                     |  |
| 1.3.1.5. Nhược điểm của Microservices.....                                  |  |
| <b>1.3.2. Công nghệ</b> .....   |  |
| 1.3.2.1. Giới thiệu về Spring Boot.....                                     |  |
| 1.3.2.2. Giới thiệu về Spring Data JPA.....                                 |  |
| 1.3.2.3 Giới thiệu về Git.....  |  |
| 1.3.2.4 Giới thiệu về MySQL.....  |  |
| <b>1.4 Kết luận chương 1</b> .....  |  |
| <b>CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG</b> .....                          |  |
| <b>2.1. Phân tích yêu cầu hệ thống</b> .....                                |  |
| 2.1.1. Xác định yêu cầu chức năng.....                                      |  |
| 2.1.2. Xác định yêu cầu phi chức năng.....                                  |  |
| 2.1.3. Xác định các tác nhân của hệ thống.....                              |  |
| 2.1.4. Xác định và mô tả các ca sử dụng.....                                |  |
| <b>2.2. Biểu đồ Use case</b> .....  |  |
| 2.2.1 Biểu đồ Use case tổng quan.....                                       |  |
| 2.2.2 Biểu đồ Use case phân rã.....   |  |
| 2.2.2.1. Phân rã Use case Quản lý người dùng.....                           |  |
| 2.2.2.2. Phân rã Use case quản lý danh mục.....                             |  |

- 2.2.2.3. Phân rã Use case quản lý tài liệu.....
- 2.2.2.4. Phân rã Use case Upload tài liệu lên kho.....
- 2.2.2.5. Phân rã Use case Quản lý bình luận.....
- 2.2.2.6. Phân rã Use case Quản lý thông tin người dùng.....
- 2.2.2.7. Phân rã Use case bình luận.....
- 2.3. Kịch bản.....
- 2.3.1 Kịch bản cho chức năng Đăng nhập.....
- 2.3.2 Kịch bản cho chức năng Đăng ký tài khoản.....
- 2.3.3 Kịch bản cho chức năng Đăng xuất.....
- 2.3.4 Kịch bản cho chức năng quản lý người dùng.....
  - 2.3.4.1. Kịch bản “ Thêm mới người dùng”.....
  - 2.3.4.2. Kịch bản “ Sửa thông tin người dùng”.....
  - 2.3.4.3. Kịch bản “ Xóa người dùng”.....
- 2.3.5 Kịch bản cho chức năng quản lý danh mục.....
  - 2.3.5.1. Kịch bản “ Thêm mới danh mục”.....
  - 2.3.5.2. Kịch bản “ Sửa thông tin danh mục”.....
  - 2.3.5.3. Kịch bản “ Xóa danh mục”.....
- 2.3.6 Kịch bản cho chức năng quản lý tài liệu.....
  - 2.3.6.1. Kịch bản “ Thêm mới tài liệu”.....
  - 2.3.6.2. Kịch bản “ Sửa thông tin tài liệu”.....
  - 2.3.6.3. Kịch bản “ Xóa tài liệu”.....
- 2.3.7 Kịch bản cho chức năng quản lý bình luận.....
  - 2.3.7.1. Kịch bản “ Thêm mới bình luận”.....
  - 2.3.7.2. Kịch bản “ Sửa thông tin bình luận”.....
  - 2.3.7.3. Kịch bản “ Xóa bình luận”.....
- 2.3.8. Kịch bản cho chức năng Upload tài liệu lên kho.....
- 2.3.9. Kịch bản cho chức năng cập nhật thông tin cá nhân.....
- 2.3.10. Kịch bản cho chức năng tìm kiếm tài liệu theo tên.....
- 2.3.11. Kịch bản cho chức năng xem chi tiết tài liệu.....
- 2.3.12. Kịch bản cho chức năng Viết bình luận.....
- 2.3.13. Kịch bản cho chức năng Thay đổi mật khẩu.....
- 2.4 Biểu đồ lớp.....
  - 2.4.1. Xác định các lớp.....
  - 2.4.2. Biểu đồ lớp phân tích.....
  - 2.4.3. Biểu đồ lớp thiết kế.....
- 2.5. Biểu đồ giao tiếp.....
  - 2.5.1. Biểu đồ giao tiếp cho chức năng đăng nhập.....
  - 2.5.2. Biểu đồ giao tiếp cho chức năng đăng ký.....

|  |  |
|--|--|
| 2.5.3. Biểu đồ giao tiếp cho chức năng quản lý người dùng.....                 |  |
| 2.5.3.1. Biểu đồ giao tiếp “Thêm người dùng”.....                              |  |
| 2.5.3.2. Biểu đồ giao tiếp “Sửa thông tin người dùng”.....                     |  |
| 2.5.3.3. Biểu đồ giao tiếp “xóa người dùng”.....                               |  |
| 2.5.4. Biểu đồ giao tiếp cho chức năng quản lý danh mục.....                   |  |
| 2.5.4.1. Biểu đồ giao tiếp “Thêm danh mục”.....                                |  |
| 2.5.4.2. Biểu đồ giao tiếp “Sửa thông tin danh mục”.....                       |  |
| 2.5.4.3. Biểu đồ giao tiếp “xóa danh mục”.....                                 |  |
| 2.5.5. Biểu đồ giao tiếp cho chức năng quản lý tài liệu.....                   |  |
| 2.5.5.1. Biểu đồ giao tiếp “Thêm tài liệu”.....                                |  |
| 2.5.5.2. Biểu đồ giao tiếp “Sửa thông tin tài liệu”.....                       |  |
| 2.5.5.3. Biểu đồ giao tiếp “xóa tài liệu”.....                                 |  |
| 2.5.6. Biểu đồ giao tiếp cho chức năng quản lý bình luận.....                  |  |
| 2.5.6.1. Biểu đồ giao tiếp “Thêm bình luận”.....                               |  |
| 2.5.6.2. Biểu đồ giao tiếp “Sửa thông tin bình luận”.....                      |  |
| 2.5.6.3. Biểu đồ giao tiếp “xóa bình luận”.....                                |  |
| 2.5.7. Biểu đồ giao tiếp cho chức năng Upload tài liệu lên kho.....            |  |
| 2.5.8. Biểu đồ giao tiếp cho chức năng cập nhật thông tin người dùng.....      |  |
| 2.5.9. Biểu đồ giao tiếp cho chức năng đổi mật khẩu.....                       |  |
| 2.5.10. Biểu đồ giao tiếp cho chức năng viết bình luận.....                    |  |
| 2.6. Phân rã thành các Service.....  |  |
| 2.7. Mô tả cấu trúc dữ liệu.....   |  |
| 2.8. Cơ sở dữ liệu.....  |  |
| 2.9. Kết luận chương 2.....  |  |
| <b>CHƯƠNG 3: CÀI ĐẶT HỆ THỐNG.....</b>   |  |
| 3.1. Các yêu cầu về cài đặt về hệ thống.....                                   |  |
| 3.1.1. Yêu cầu về phần mềm.....  |  |
| 3.1.2. Yêu cầu về phần cứng.....   |  |
| 3.2. Giao diện website trao đổi tài liệu dựa trên kiến trúc hướng dịch vụ..... |  |
| 3.3 Kết luận chương 3.....   |  |
| <b>KẾT LUẬN.....</b>   |  |
| <b>TÀI LIỆU THAM KHẢO.....</b>   |  |

## DANH MỤC CÁC HÌNH VẼ

|   |    |
|---|----|
| Hình 1: Kiến trúc Microservices.....  | 5  |
| Hình 2: Một kiến trúc của Spring Boot.....                                  | 8  |
| Hình 3: Tính năng của Spring Boot.....                                      | 9  |
| Hình 4: Các method của Spring Data JPA.....                                 | 11 |
| Hình 5: Công cụ quản lý mã nguồn Git.....                                   | 12 |
| Hình 6: Hệ quản trị cơ sở dữ liệu MySQL.....                                | 14 |
| Hình 7: Sơ đồ use case tổng quát.....                                       | 18 |
| Hình 8: Sơ đồ use case quản lý người dùng.....                              | 19 |
| Hình 9: Sơ đồ use case quản lý danh mục.....                                | 20 |
| Hình 10: Sơ đồ use case quản lý tài liệu.....                               | 21 |
| Hình 11: Sơ đồ use case Upload tài liệu lên kho.....                        | 22 |
| Hình 12: Sơ đồ use case quản lý bình luận.....                              | 22 |
| Hình 13: Sơ đồ use case Quản lý thông tin người dùng.....                   | 23 |
| Hình 14: Sơ đồ use case bình luận.....                                      | 24 |
| Hình 15: Biểu đồ lớp phân tích.....   | 45 |
| Hình 16: Biểu đồ lớp thiết kế.....  | 46 |
| Hình 17: Biểu đồ giao tiếp cho chức năng Login.....                         | 46 |
| Hình 18: Biểu đồ giao tiếp cho chức năng Register.....                      | 47 |
| Hình 19: Biểu đồ giao tiếp cho chức năng add user.....                      | 47 |
| Hình 20: Biểu đồ giao tiếp cho chức năng edit user.....                     | 48 |
| Hình 21: Biểu đồ giao tiếp cho chức năng delete user.....                   | 48 |
| Hình 22: Biểu đồ giao tiếp cho chức năng add category.....                  | 49 |
| Hình 23: Biểu đồ giao tiếp cho chức năng edit category.....                 | 49 |
| Hình 24: Biểu đồ giao tiếp cho chức năng delete category.....               | 50 |
| Hình 25: Biểu đồ giao tiếp cho chức năng add document.....                  | 50 |
| Hình 26: Biểu đồ giao tiếp cho chức năng edit document.....                 | 51 |
| Hình 27: Biểu đồ giao tiếp cho chức năng delete document.....               | 51 |
| Hình 28: Biểu đồ giao tiếp cho chức năng add comment.....                   | 52 |
| Hình 29: Biểu đồ giao tiếp cho chức năng edit comment.....                  | 52 |
| Hình 30: Biểu đồ giao tiếp cho chức năng delete comment.....                | 53 |
| Hình 31: Biểu đồ giao tiếp cho chức năng upload tài liệu lên kho.....       | 53 |
| Hình 32: Biểu đồ giao tiếp cho chức năng cập nhật thông tin người dùng..... | 54 |
| Hình 33: Biểu đồ giao tiếp cho chức năng đổi mật khẩu.....                  | 54 |
| Hình 34: Biểu đồ giao tiếp cho chức năng viết bình luận.....                | 55 |
| Hình 35: Biểu đồ phân rã các Service.....                                   | 55 |
| Hình 36: Cơ sở dữ liệu của UserService.....                                 | 59 |
| Hình 37: Cơ sở dữ liệu của CategoryService.....                             | 59 |
| Hình 38: Cơ sở dữ liệu của DocumentService.....                             | 60 |
| Hình 39: Cơ sở dữ liệu của CommentService.....                              | 60 |

|  |    |
|--|----|
| Hình 40: Cơ sở dữ liệu của AdminService.....                 | 61 |
| Hình 41: Giao diện danh sách user.....                       | 62 |
| Hình 42: Giao diện sửa user.....                             | 63 |
| Hình 43: Giao diện xóa user.....                             | 63 |
| Hình 44: Giao diện danh sách category.....                   | 64 |
| Hình 45: Giao diện sửa category.....                         | 64 |
| Hình 46: Giao diện xóa category.....                         | 65 |
| Hình 47: Giao diện danh sách document.....                   | 65 |
| Hình 48: Giao diện sửa document.....                         | 66 |
| Hình 49: Giao diện xóa document.....                         | 66 |
| Hình 50: Giao diện Login.....                                | 67 |
| Hình 51: Giao diện Register.....                             | 67 |
| Hình 52: Giao diện trang chủ.....                            | 68 |
| Hình 53: Giao diện thông tin cá nhân.....                    | 68 |
| Hình 54: Giao diện sửa thông tin cá nhân.....                | 69 |
| Hình 55: Giao diện đổi password.....                         | 69 |
| Hình 56: Giao diện danh sách tài liệu theo danh mục.....     | 70 |
| Hình 57: Giao diện chi tiết tài liệu.....                    | 70 |
| Hình 58: Giao diện upload file từ google drive.....          | 71 |
| Hình 59: Giao diện chọn tài khoản.....                       | 71 |
| Hình 60: Giao diện danh sách các tệp trong google drive..... | 72 |
| Hình 61: Giao diện trang xem danh sách tài liệu của tôi..... | 72 |

## DANH MỤC CÁC BẢNG



|   |    |
|---|----|
| Bảng 1: So sánh giữa kiến trúc Monolithic và Microservices..... | 4  |
| Bảng 2: Yêu cầu phi chức năng của hệ thống.....                 | 16 |
| Bảng 3: Các tác nhân của hệ thống.....                          | 17 |
| Bảng 4: Mô tả các ca sử dụng.....                               | 17 |
| Bảng 5: Kịch bản cho use case đăng nhập.....                    | 25 |
| Bảng 6: Kịch bản cho use case đăng ký tài khoản.....            | 26 |
| Bảng 7: Kịch bản cho use case đăng xuất.....                    | 27 |
| Bảng 8: Kịch bản cho use case thêm mới người dùng.....          | 28 |
| Bảng 9: Kịch bản cho use case sửa thông tin người dùng.....     | 29 |
| Bảng 10: Kịch bản cho use case xóa người dùng.....              | 30 |
| Bảng 11: Kịch bản cho use case thêm mới danh mục.....           | 31 |
| Bảng 12: Kịch bản cho use case sửa thông tin danh mục.....      | 32 |
| Bảng 13: Kịch bản cho use case xóa danh mục.....                | 33 |
| Bảng 14: Kịch bản cho use case thêm mới tài liệu.....           | 34 |
| Bảng 15: Kịch bản cho use case sửa thông tin tài liệu.....      | 35 |
| Bảng 16: Kịch bản cho use case xóa tài liệu.....                | 36 |
| Bảng 17: Kịch bản cho use case thêm mới bình luận.....          | 37 |
| Bảng 18: Kịch bản cho use case sửa thông tin bình luận.....     | 38 |
| Bảng 19: Kịch bản cho use case xóa bình luận.....               | 39 |
| Bảng 20: Kịch bản cho use case upload tài liệu lên kho.....     | 40 |
| Bảng 21: Kịch bản cho use case cập nhật thông tin cá nhân.....  | 41 |
| Bảng 22: Kịch bản cho use case tìm kiếm tài liệu theo tên.....  | 42 |
| Bảng 23: Kịch bản cho use case xem chi tiết tài liệu.....       | 42 |
| Bảng 24: Kịch bản cho use case viết bình luận.....              | 43 |
| Bảng 25: Kịch bản cho use case thay đổi mật khẩu.....           | 44 |
| Bảng 26: Bảng Users.....  | 56 |
| Bảng 27: Bảng Address.....                                      | 56 |
| Bảng 28: Bảng PhoneNumber.....                                  | 57 |
| Bảng 29: Bảng Document.....                                     | 57 |
| Bảng 30: Bảng Category_Document.....                            | 57 |
| Bảng 31: bảng Cateogry.....                                     | 58 |
| Bảng 32: Bảng AdminSv.....                                      | 58 |
| Bảng 33: Bảng Comment.....                                      | 58 |

## DANH MỤC CÁC TỪ VIẾT TẮT

| <b>TỪ VIẾT TẮT</b> | <b>Ý NGHĨA</b>                    |
|--------------------|-----------------------------------|
| SOA                | Service-Oriented Architecture     |
| API                | Application Programming Interface |
| AMQP               | Advanced Message Queuing Protocol |
| AWS                | Amazon Web Services               |
| RAD                | Rapid Application Development     |

## MỞ ĐẦU

Trong thời kỳ công nghệ thông tin đang phát triển mạnh trong nhiều năm trở lại đây, việc chia sẻ tài liệu là một phần quan trọng. Một hệ thống chia sẻ tài liệu phù hợp và hiệu quả có thể giúp tổ chức lưu trữ, quản lý và truy cập tài liệu một cách dễ dàng và nhanh chóng, tăng năng suất trong công việc

Một hệ thống chia sẻ tài liệu tốt có thể giảm thiểu thời gian và tài nguyên cần thiết. Thay vì phải gửi tài liệu qua email, facebook, hoặc trên các nền tảng khác, cũng như phải sao chép thành nhiều bản để chia sẻ thì bây giờ người dùng có thể đơn giản truy cập và tải xuống tài liệu từ hệ thống chia sẻ tài liệu.

Đặc biệt hệ thống thường cung cấp các cơ chế bảo mật để bảo vệ thông tin quan trọng. Người dùng có thể áp dụng quyền truy cập và phân quyền để chỉ cho phép những người được ủy quyền truy cập vào các tài liệu nhất định. Điều này giúp bảo mật thông tin hơn là các nền tảng khác.

Xuất phát từ những yêu cầu trên, em đã chọn đề tài làm đề án tốt nghiệp đại học là “Xây dựng hệ thống chia sẻ tài liệu theo kiến trúc hướng dịch vụ”

Mục tiêu của đề án là xây dựng một hệ thống chia sẻ tài liệu có thể dễ dàng quản lý tài liệu, phân chia tài liệu cho hợp lý, người dùng có thể đăng tài liệu, tìm kiếm tài liệu cần tìm một cách dễ dàng

Nội dung của đề án gồm có 3 chương chính sau:

### **Chương 1: Cơ sở lý thuyết**

Chương 1 sẽ giới thiệu về đặt vấn đề về bài toán chia sẻ tài liệu, lý do lựa chọn đề tài và giải pháp cho việc giải quyết đề tài đó bao gồm kiến trúc, các công nghệ sử dụng để xây dựng hệ thống.

### **Chương 2: Phân tích thiết kế hệ thống**

Chương 2 sẽ phân tích yêu cầu, nghiệp vụ của hệ thống, từ đó thiết kế các tác nhân, sơ đồ use case, kịch bản, sơ đồ thiết kế của hệ thống, biểu đồ giao tiếp, biểu đồ triển khai của hệ thống

### **Chương 3: Cài đặt hệ thống**

Chương 3 sẽ mô tả môi trường cài đặt hệ thống, quá trình cài đặt hệ thống và trình bày kết quả thực hiện hệ thống

## CHƯƠNG 1: GIỚI THIỆU VỀ ĐỀ TÀI, KIẾN TRÚC VÀ CÔNG NGHỆ SỬ DỤNG

Trong chương này sẽ trình bày về một số nội dung như: đặt vấn đề, lý do xây dựng bài toán chia sẻ tài liệu, đưa ra các giải pháp để giải quyết vấn đề, từ đó lựa chọn giải pháp phù hợp. Tiếp đó giới thiệu về kiến trúc, các công nghệ được sử dụng cho hệ thống chia sẻ tài liệu theo kiến trúc hướng dịch vụ.

### 1.1. Đặt vấn đề

Như chúng ta đã thấy trong môi trường kinh doanh hiện đại, việc chia sẻ tài liệu là một phần quan trọng của quy trình làm việc và hợp tác. Việc này mang lại nhiều lợi ích về cộng tác, hiệu suất làm việc, quản lý thông tin và linh hoạt làm việc.

Hiện nay, càng nhiều doanh nghiệp chú trọng phát triển hệ thống chia sẻ tài liệu, điển hình như Google cho ra google drive, Microsoft OneDrive của hãng Microsoft, github,.... Theo khảo sát có rất nhiều công ty ở Việt Nam cũng chú trọng về hệ thống chia sẻ tài liệu.

Việc có hệ thống chia sẻ tài liệu đem lại hiệu quả to lớn trong doanh nghiệp. So với việc phải photo tài liệu và phân phát cho mọi người thì giờ đây ta có thể thực hiện thao tác một cách đơn giản, nhanh chóng, đỡ tốn thời gian và chi phí, đem lại hiệu quả công việc cao.

Ngoài ra nó còn đem lại các lợi ích như tiết kiệm thời gian và tăng hiệu suất làm việc, quản lý tài liệu dễ dàng hơn, cải thiện sự cộng tác và làm việc nhóm, tiết kiệm chi phí, dễ dàng sao lưu và khôi phục dữ liệu.

Tuy nhiên, để phát triển một hệ thống chia sẻ tài liệu ta cần đối mặt những thách thức sau:

1. **Quản lý và tìm kiếm tài liệu:** Trong các tổ chức lớn, việc quản lý và tìm kiếm tài liệu trở nên phức tạp. Có hàng nghìn, thậm chí hàng triệu tài liệu được tạo ra và lưu trữ, và việc tìm kiếm một tài liệu cụ thể hoặc các tài liệu liên quan có thể là một thách thức. Đặc biệt là tài liệu được lưu trữ trên nhiều nền tảng và hệ thống khác nhau, việc tìm kiếm và quản lý lại càng trở lên khó khăn.
2. **Đồng bộ hóa và quản lý phiên bản:** Trong quá trình làm việc nhóm và hợp tác, nhiều người cùng truy cập và chỉnh sửa cùng một tài liệu. Điều này có thể dẫn đến sự mất đồng bộ hóa và trùng lặp phiên bản. Quản lý phiên bản tài liệu trở nên quan trọng để đảm bảo rằng mọi người đang làm việc trên phiên bản mới nhất và có thể phục hồi lại phiên bản trước đó nếu cần.

3. **Quyền truy cập và bảo mật:** Trong việc chia sẻ tài liệu, việc xác định và quản lý quyền truy cập là rất quan trọng. Cần xác định ai có quyền truy cập vào tài liệu nào và có thể thực hiện những thao tác gì, như chỉnh sửa, xem, hoặc chia sẻ. Đồng thời, bảo mật thông tin trong tài liệu trở thành một yếu tố quan trọng để đảm bảo rằng tài liệu không bị truy cập trái phép hoặc rò rỉ.
4. **Tích hợp và tương tác với các hệ thống khác:** Trong môi trường kinh doanh hiện đại, việc chia sẻ tài liệu thường xuyên liên quan đến việc tích hợp và tương tác với các hệ thống khác hoặc các ứng dụng và nền tảng khác. Điều này đòi hỏi khả năng tương thích và tích hợp của hệ thống chia sẻ tài liệu.

Từ những vấn đề trên, em chọn đề án tốt nghiệp “Xây dựng hệ thống chia sẻ tài liệu theo kiến trúc hướng dịch vụ”.

## 1.2 Tìm hiểu các giải pháp

### 1.2.1. Tổng quan về yêu cầu

#### 1.2.1.1. Giới thiệu về bài toán chia sẻ tài liệu

Bài toán chia sẻ tài liệu là một vấn đề trong truyền thông tin giữa các cá nhân với nhau. Trước khi công nghệ phát triển, việc này phải thực hiện qua thư, từ, in ấn, phân phối giấy tờ,... Ngày nay với công nghệ phát triển ta có thể chia sẻ qua internet, các trang web,...

Bài toán chia sẻ tài liệu đặt ra một số thách thức là bảo mật và quyền riêng tư, cũng như về bản quyền,..

Tóm lại, bài toán chia sẻ tài liệu là một vấn đề quan trọng trong việc truyền tải thông tin và kiến thức. Nó đòi hỏi sự quản lý, tổ chức và bảo mật tài liệu một cách hiệu quả để đảm bảo rằng thông tin có thể được chia sẻ một cách dễ dàng, an toàn và có ý nghĩa.

#### 1.2.1.2. Giới thiệu sơ qua về đề tài

Đây là website nhằm đáp ứng trao đổi tài liệu.

website sau khi xây dựng xong sẽ đáp ứng được các yêu cầu sau:

- Quản lý người dùng
- Quản lý danh mục
- Quản lý tài liệu
- Xem danh sách tài liệu theo danh mục
- Upload tài liệu lên kho
- Bình luận, đánh giá tài liệu

## 1.2.2. Giải pháp

### 1.2.2.1. Kiến trúc Monolithic

Monolithic(kiến trúc nguyên khối) là ứng dụng mà mọi dịch vụ đều nằm trên một hệ thống và dùng chung một database. Người dùng thao tác trực tiếp với giao diện, giao diện gọi đến chức năng, chức năng lại tìm đến dữ liệu trong database.

### 1.2.2.2. Kiến trúc Microservices

Microservices là kiểu kiến trúc thay vì các chức năng được đặt cùng nhau trong hệ thống thì nó sẽ được tách riêng lẻ và có thể với mỗi chức năng đó sẽ tương tác đến các database khác nhau.

### 1.2.2.3. So sánh giữa Monolithic và Microservices

| Kiến trúc Monolithic         | Kiến trúc Microservices                     |
|------------------------------|---|
| Hiệu suất truyền tải tốt hơn | Dễ mở rộng                                  |
| Đơn giản hơn                 | Đảm bảo an toàn khi gặp sự cố, dễ chỉnh sửa |
|                              | Tài nguyên dùng ít hơn                      |

*Bảng 1: So sánh giữa kiến trúc Monolithic và Microservices*

Ta thấy đối với bài toán chia sẻ tài liệu, tài nguyên sẽ lớn, phức tạp, và có thể mở rộng ra nhiều lĩnh vực, nhiều chức năng trong tương lai, và đặc biệt các chức năng sẽ được tách riêng lẻ, dễ chỉnh sửa. Vì thế nên trong trường hợp này, kiến trúc Microservices phù hợp hơn kiến trúc Monolithic nên em quyết định chọn kiến trúc Microservices để xây dựng đề tài.

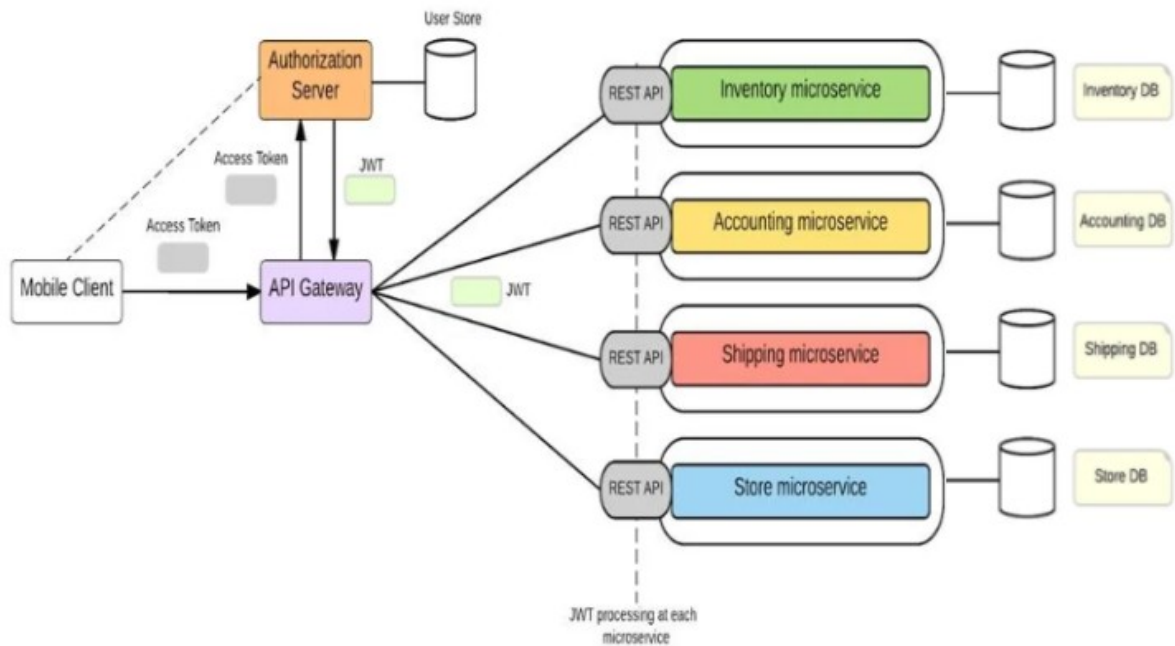
## 1.3. Giải pháp

### 1.3.1. Kiến trúc

Ta sử dụng kiến trúc Microservices áp dụng cho hệ thống chia sẻ tài liệu.

#### 1.3.1.1. Khái niệm kiến trúc Microservices

Microservices là một mô hình kiến trúc mà trong đó ứng dụng được phát triển và triển khai dưới dạng một tập hợp các dịch vụ độc lập và nhỏ gọn. Mỗi dịch vụ đảm nhận một phần nhỏ của toàn bộ ứng dụng và chúng tương tác với nhau thông qua các giao diện được định rõ. Mỗi dịch vụ có thể được triển khai, mở rộng và quản lý độc lập, giúp tăng tính linh hoạt và mở rộng của hệ thống.



Hình 1: Kiến trúc Microservices

API Gateway: hoạt động như một điểm truy cập duy nhất cho tất cả khách hàng với vai trò là dịch vụ biên giúp hiển thị các vi dịch vụ ra thế giới bên ngoài dưới dạng API được quản lý. Ngoài ra nó còn có nhiệm vụ bổ sung như cân bằng tải đơn giản, xác thực và ủy quyền, xử lý lỗi, kiểm tra, dịch giao thức và định tuyến.

Ưu điểm của việc sử dụng API Gateway:

- Tách khách hàng khỏi các service. Các service có thể được cập nhật hoặc tái cấu trúc mà không cần cập nhật tất cả các máy khách.
- Các dịch vụ có thể sử dụng các giao thức nhắn tin không thân thiện với web, chẳng hạn như AMQP.
- API Gateway có thể thực hiện các chức năng xuyên suốt như xác thực, ghi nhật ký và cân bằng tải.

Ví dụ về Microservices [6]

Một vài công ty sử dụng kiến trúc Microservices như:

- Netflix đã quyết định chia kiến trúc Monolithic của mình thành các vi dịch vụ vào năm 2009, sau khi gặp phải nhiều thất bại và sự gián đoạn máy chủ thường xuyên do thiết kế nguyên khối của nó. Netflix phải mất 2 năm để chuyển hoàn toàn các giải pháp kiến trúc của họ sang microservice và họ đã công bố ứng dụng được thiết kế lại vào năm 2011.

- Amazon - Amazon là một trong những công ty đầu tiên sử dụng microservice để cách mạng hóa toàn bộ công ty của họ. Là một trong những nhà quảng bá vi dịch vụ nổi tiếng nhất hiện nay, Amazon Web Services (AWS) cung cấp cho các doanh nghiệp cơ sở hạ tầng mà họ cần để xây dựng và quản lý các bộ chứa cũng như dịch vụ vi mô một cách nhanh chóng.

### 1.3.1.2. Một số đặc điểm chính của kiến trúc Microservices

Nhiều thành phần: Bản thân đặc điểm nổi bật của microservice là sự phân chia thành từng ngăn. Điều này giúp hệ thống triển khai lại bất kỳ thành phần đơn lẻ nào sau khi cấu hình lại mà không cần phải tắt hoạt động của toàn bộ ngăn xếp ứng dụng. [6]

Dành cho doanh nghiệp: Toàn bộ kiến trúc vi dịch vụ tập trung vào khả năng kinh doanh. Về cơ bản nó được thực hiện để phản ánh sự thuận tiện trong kinh doanh. Mỗi nhóm được giao một nhiệm vụ phụ cụ thể và họ cần tập trung phát triển giao diện đó một mình. Truyền thông nội bộ được thực hiện thông qua dịch vụ bus tin nhắn. [6]

Phi tập trung - Một hệ thống nguyên khối sử dụng một cơ sở dữ liệu logic duy nhất cho nhiều ứng dụng. Trong khi các ứng dụng microservice quản lý cơ sở dữ liệu riêng thì mỗi dịch vụ thường quản lý cơ sở dữ liệu riêng. Phân cấp giúp nâng cao năng suất và quản lý nhóm tốt hơn.

Định tuyến đơn giản: Giống như các hệ thống UNIX tiêu chuẩn, vi dịch vụ nhận yêu cầu, xử lý chúng và phản hồi chính xác. Trong nhiều sản phẩm khác, chẳng hạn như Xe buýt dịch vụ doanh nghiệp (ESB) , hệ thống công nghệ cao xử lý việc định tuyến tin nhắn, bố trí và các quy tắc kinh doanh. Vi dịch vụ giúp việc định tuyến dễ dàng hơn với các điểm cuối thông minh có khả năng phân tích dữ liệu và áp dụng các quy trình logic cũng như quy trình cam. [6]

Khắc phục lỗi: Điều gì xảy ra nếu một trong các ứng dụng đột nhiên bị hỏng? Thông thường, điều này có nghĩa là toàn bộ chương trình sẽ bị tắt để bảo trì. Nhưng microservice xử lý những lỗi này một cách dễ dàng và duy trì hoạt động của framework.

Thiết kế tiến hóa: Nhiều ứng dụng bắt đầu với kiến trúc nguyên khối, nhưng khi xuất hiện các yêu cầu bổ sung, nó có thể được cải tiến dần dần thành các dịch vụ vi mô thông qua các API tương tác trên kiến trúc nguyên khối cũ hơn.



#### 1.3.1.3. Lợi ích của kiến trúc Microservices

Khả năng mở rộng: Mỗi microservice có thể được điều chỉnh quy mô độc lập để đáp ứng nhu cầu về các tính năng mà nó hỗ trợ trong ứng dụng. Do đó, các nhóm có thể điều chỉnh quy mô phù hợp với nhu cầu cơ sở hạ tầng của mình, ước tính chính xác chi phí của các tính năng mới và duy trì tính khả dụng của dịch vụ trong thời gian cao điểm.

Quản trị dữ liệu tốt hơn: Vì microservice tăng khả năng hiển thị mã nên việc quản lý dữ liệu trên đám mây/máy chủ trở nên dễ dàng hơn. [6]

Bất khả tri về ngôn ngữ và công nghệ: Microservices cho phép các nhà phát triển xây dựng mã bằng các ngôn ngữ khác nhau.

Cải thiện bảo mật dữ liệu: cung cấp giải pháp tập trung để bảo vệ ứng dụng của bạn khỏi các chương trình độc hại và kẻ tấn công mạng.[6]

#### 1.3.1.4. Ưu điểm của Microservices

Dễ mở rộng quy mô: Mỗi Service được thiết kế, phát triển và triển khai độc lập khi chỉnh sửa một service thì sẽ không ảnh hưởng đến cả hệ thống.

Chống chịu lỗi tốt hơn: Các ứng dụng trong microservices vẫn có thể hoạt động dù cho bất kỳ service nào khác gặp lỗi, bởi các service trong kiến trúc này gần như là độc lập với nhau.

Cho phép thử nghiệm nhiều công nghệ khác nhau: Các Developer có thể linh hoạt thử nghiệm nhiều loại công nghệ trong quá trình tạo ra service.

#### 1.3.1.5. Nhược điểm của Microservices

Giao tiếp giữa các service trở nên phức tạp , khó khăn hơn: Việc chia một ứng dụng thành nhiều module nhỏ sẽ khiến việc giao tiếp trở nên phức tạp và tốn nhiều chi phí hơn, đặc biệt là trong các hệ thống khác nhau thì việc giao tiếp giữa các service càng phức tạp hơn, tốn nhiều thời gian hơn.

Cần nhiều tài nguyên hơn: càng nhiều microservices thì sẽ cần tương ứng lượng tài nguyên cần thiết để triển khai hệ thống, trong đó database cũng sẽ nhiều hơn

Testing và debugging toàn cục: Trong kiến trúc microservices thì các service cần phải được khởi chạy và thử nghiệm riêng lẻ khiến việc test trở lên khó khăn hơn, khó tìm lỗi.

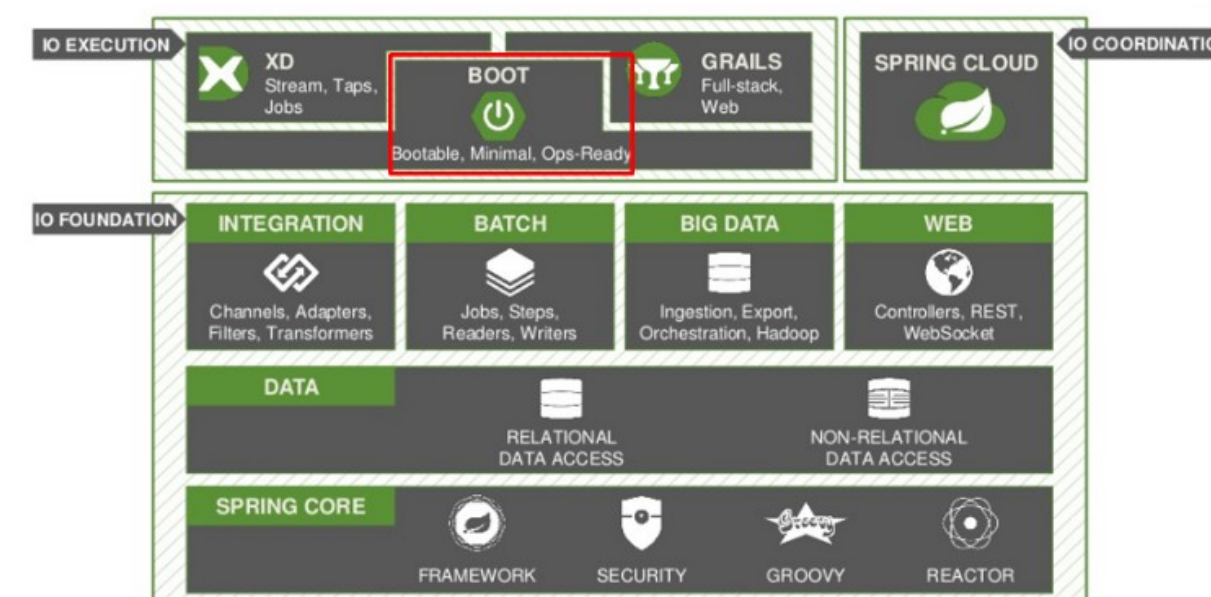
Không thiết thực cho các ứng dụng nhỏ: Microservices là một lựa chọn tốt cho các ứng dụng có quy mô lớn, tuy nhiên những ứng dụng nhỏ sử dụng kiến trúc này sẽ cần phối hợp nhiều service khác nhau trong quá trình triển khai.

## 1.3.2. Công nghệ

### 1.3.2.1. Giới thiệu về Spring Boot

#### a) Khái niệm

Spring Boot là một trong những module của Spring Framework cung cấp tính năng RAD (Phát triển ứng dụng nhanh). Nó được sử dụng để tạo một ứng dụng dựa trên Spring độc lập mà không yêu cầu cấu hình XML.



Hình 2: Một kiến trúc của Spring Boot

#### b) Ưu điểm

- Tạo ứng dụng một cách độc lập, có thể chạy cho Java Web.
- Có cấu hình sẵn, giúp giảm thời gian viết code, tiết kiệm chi phí.
- Có các tính năng của Spring Framework.
- Cung cấp nhiều plugin chuẩn cho Microservices .

#### c) Đặc điểm và tính năng

##### - Đặc điểm

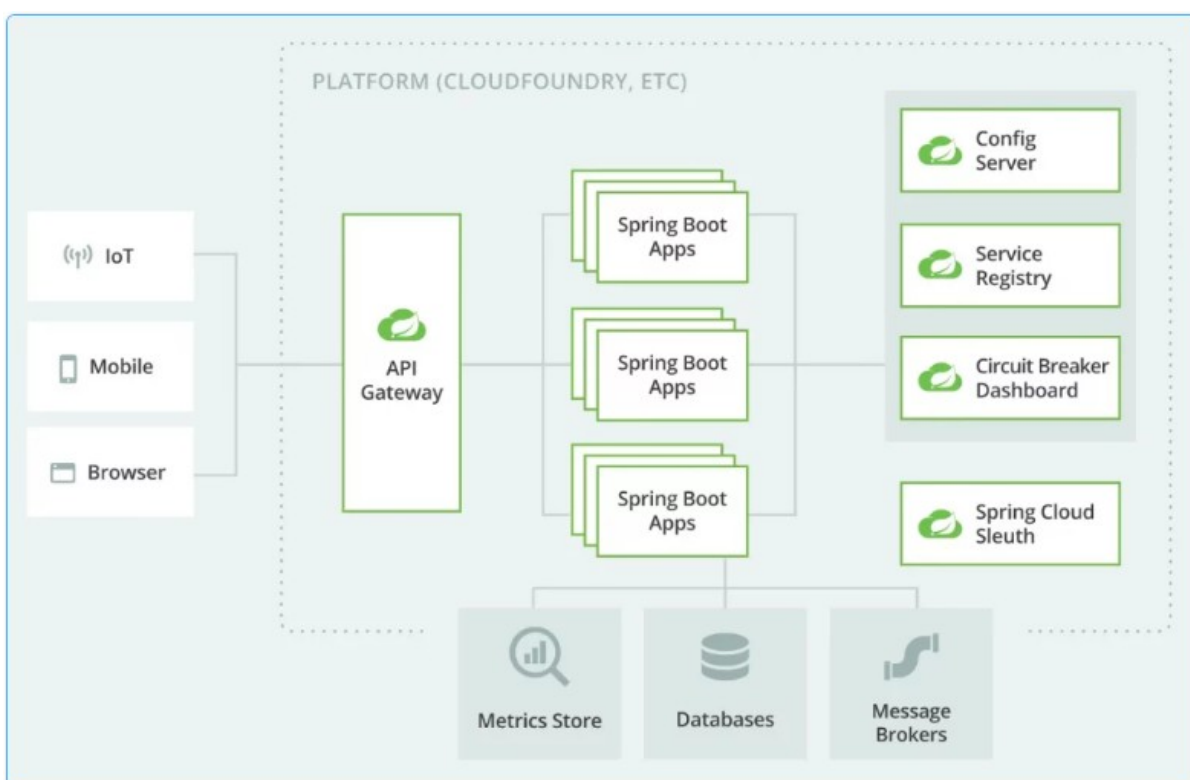
- Cấu hình XML trở nên đơn giản nhanh chóng và dễ dàng.

- Gia tăng được năng suất và giảm thiểu thời gian viết code.
- Giúp nhà phát triển dễ dàng hơn trong quá trình phát triển ứng dụng mà không cần biết quá nhiều kiến thức.

- Tính năng

- SpringApplication: giúp cho việc chạy dự án trở lên dễ dàng hơn.
- Externalized Configuration: Có thể chạy trên nhiều môi trường khác nhau.
- Profiles: Sử dụng Profile để phân chia từng loại cho từng môi trường để dễ dàng quản lý hơn.
- Logging: Tính năng này được sử dụng cho toàn bộ chức năng log trong phạm vi nội bộ và nó được quản lý mặc định.

Còn có một vài tính năng khác như Security, Developing web Applications, Working with SQL Technologies, Caching, Sending Email,...



Hình 3: Tính năng của Spring Boot

### 1.3.2.2. Giới thiệu về Spring Data JPA

#### a) Khái niệm về JPA

JPA (Java Persistence API) là một tiêu chuẩn Java để quản lý đối tượng và truy cập cơ sở dữ liệu quan hệ. Nó cung cấp một cách thuận tiện và linh hoạt để lưu trữ và truy xuất dữ liệu từ các cơ sở dữ liệu quan hệ trong ứng dụng Java.

JPA định nghĩa một tập hợp các giao diện và annotation để mô tả cấu trúc cơ sở dữ liệu và quan hệ giữa các đối tượng Java. Nó cho phép bạn ánh xạ các đối tượng Java vào các bảng trong cơ sở dữ liệu và thực hiện các thao tác CRUD (Create, Read, Update, Delete) trên dữ liệu.

#### b) Khái niệm Spring Data JPA

Spring Data JPA là một phần của Spring Data, là một dự án của Spring Framework nhằm cung cấp một cách tiếp cận thuận tiện để làm việc với cơ sở dữ liệu trong ứng dụng Java. Nó làm việc trên nền tảng JPA (Java Persistence API) và cung cấp các tính năng mạnh mẽ để giảm thiểu công việc lặp lại và tăng hiệu suất phát triển.

Spring Data JPA tập trung vào việc triển khai các repository dựa trên JPA. Nó cung cấp một cách đơn giản để tạo ra các repository dựa trên JPA mà không cần triển khai các phương thức cơ bản như tìm kiếm, thêm, sửa, xóa dữ liệu. Bằng cách sử dụng các annotation và quy ước mặc định, Spring Data JPA tự động tạo các truy vấn JPA và cung cấp các phương thức để truy cập dữ liệu một cách dễ dàng.

Các method của Spring Data JPA

| Keyword            | Sample  | JPQL snippet  |
|--------------------|---|---|
| Distinct           | findDistinctByLastnameAndFirstname                        | <code>select distinct ... where x.lastname = ?1 and x.firstname = ?2</code>   |
| And                | findByLastnameAndFirstname                                | <code>... where x.lastname = ?1 and x.firstname = ?2</code>                   |
| Or                 | findByLastnameOrFirstname                                 | <code>... where x.lastname = ?1 or x.firstname = ?2</code>                    |
| Is, Equals         | findByFirstname, findByFirstnameIs, findByFirstnameEquals | <code>... where x.firstname = ?1</code>                                       |
| Between            | findByStartDateBetween                                    | <code>... where x.startDate between ?1 and ?2</code>                          |
| LessThan           | findByAgeLessThan   | <code>... where x.age &lt; ?1</code>  |
| LessThanEqual      | findByAgeLessThanEqual                                    | <code>... where x.age &lt;= ?1</code>   |
| GreaterThan        | findByAgeGreaterThan                                      | <code>... where x.age &gt; ?1</code>  |
| GreaterThanEqual   | findByAgeGreaterThanEqual                                 | <code>... where x.age &gt;= ?1</code>   |
| After              | findByStartDateAfter                                      | <code>... where x.startDate &gt; ?1</code>                                    |
| Before             | findByStartDateBefore                                     | <code>... where x.startDate &lt; ?1</code>                                    |
| IsNull, Null       | findByAge(Is)Null   | <code>... where x.age is null</code>  |
| IsNotNull, NotNull | findByAge(Is)NotNull                                      | <code>... where x.age not null</code>   |
| Like               | findByFirstnameLike                                       | <code>... where x.firstname like ?1</code>                                    |
| NotLike            | findByFirstnameNotLike                                    | <code>... where x.firstname not like ?1</code>                                |
| StartingWith       | findByFirstnameStartingWith                               | <code>... where x.firstname like ?1 (parameter bound with appended %)</code>  |
| EndingWith         | findByFirstnameEndingWith                                 | <code>... where x.firstname like ?1 (parameter bound with prepended %)</code> |
| Containing         | findByFirstnameContaining                                 | <code>... where x.firstname like ?1 (parameter bound wrapped in %)</code>     |
| OrderBy            | findByAgeOrderByLastnameDesc                              | <code>... where x.age = ?1 order by x.lastname desc</code>                    |
| Not                | findByLastnameNot   | <code>... where x.lastname &lt;&gt; ?1</code>                                 |
| In                 | findByAgeIn(Collection<Age> ages)                         | <code>... where x.age in ?1</code>  |
| NotIn              | findByAgeNotIn(Collection<Age> ages)                      | <code>... where x.age not in ?1</code>  |
| True               | findByActiveTrue()  | <code>... where x.active = true</code>  |
| False              | findByActiveFalse()                                       | <code>... where x.active = false</code>                                       |
| IgnoreCase         | findByFirstnameIgnoreCase                                 | <code>... where UPPER(x.firstname) = UPPER(?1)</code>                         |

*Hình 4: các method của Spring Data JPA*

### 1.3.2.3 Giới thiệu về Git

#### a) Khái niệm về Git

Git là hệ thống quản lý các phiên bản dưới dạng phân tán. Đây là hệ thống quản lý phổ biến nhất hiện nay. GIT là một phiên bản của Distributed Version Control System – DVCS hay còn gọi là VCS. Sử dụng GIT sẽ đem tới cho các lập trình viên một kho lưu trữ, trong đó chứa đầy đủ các lịch sử thay đổi của hệ thống.



*Hình 5: Công cụ quản lý mã nguồn Git*

#### b) Một số khái niệm trong git

**Repository:** Repository là nơi lưu trữ tất cả các phiên bản và lịch sử của dự án. Nó có thể là một thư mục cục bộ trên máy tính của bạn (local repository) hoặc một kho lưu trữ trên máy chủ từ xa (remote repository).

**Commit:** Một commit trong Git đại diện cho một bản ghi của tệp tin hoặc một tập hợp các tệp tin trong repository. Mỗi commit kèm theo thông điệp mô tả những thay đổi đã được thực hiện.

**Branch:** Một branch trong Git là một nhánh công việc độc lập trong repository. Nó cho phép bạn làm việc song song trên các tính năng, sửa lỗi hoặc thay đổi mà

không ảnh hưởng đến nhánh chính (master branch) hoặc các nhánh khác. Branch giúp quản lý và đồng bộ các phiên bản khác nhau của dự án.

**Merge:** Merge là quá trình kết hợp các thay đổi từ một nhánh vào nhánh khác. Khi bạn đã hoàn thành công việc trên một branch riêng, bạn có thể merge nhánh đó vào nhánh chính hoặc nhánh khác để đưa các thay đổi vào phiên bản chính thức của dự án.

**Pull và Push:** Pull là hành động lấy các thay đổi từ remote repository về local repository, cập nhật phiên bản trên máy tính của bạn với những thay đổi mới nhất. Push là hành động đẩy các thay đổi từ local repository lên remote repository, cập nhật phiên bản chung của dự án.

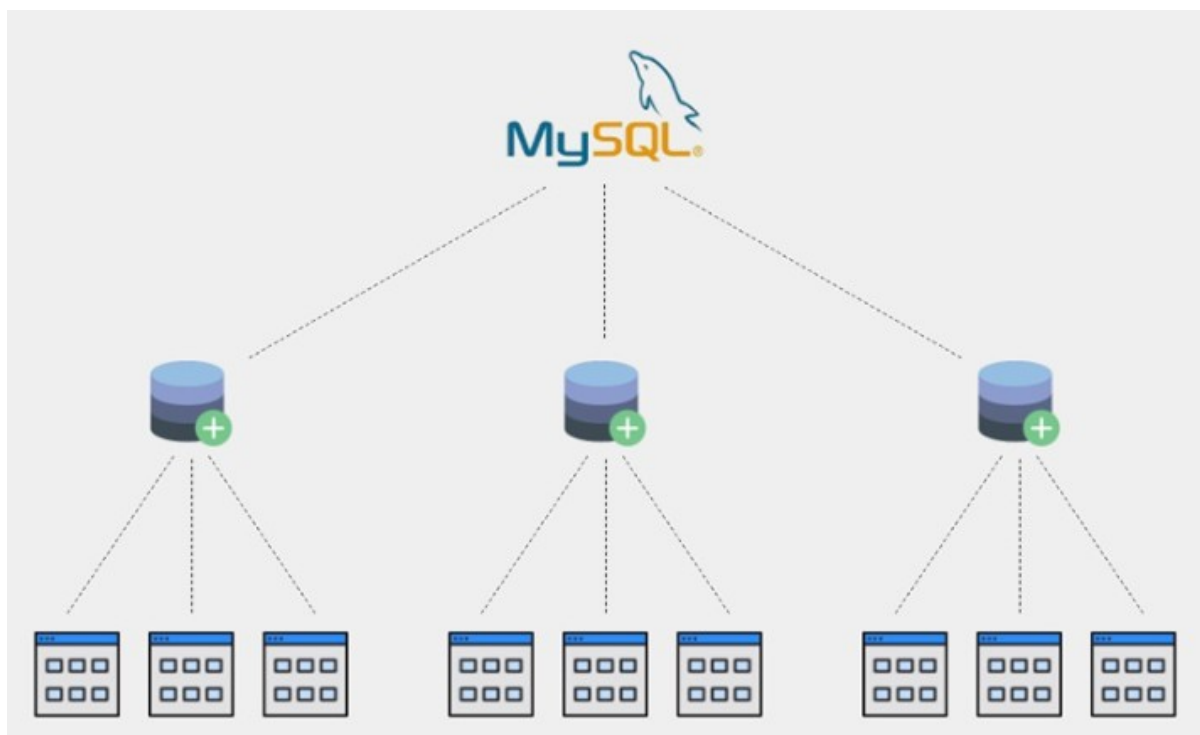
### c) Lợi ích của git

- Thao tác thực hiện nhanh, gọn, lẹ hơn, việc sử dụng an toàn hơn.
- Người dùng có thể dễ dàng kết hợp các nhánh với nhau. Qua đó giúp đơn giản hóa quy trình code theo nhóm.
- Khi sử dụng GIT, người dùng chỉ cần sử dụng clone mã nguồn từ kho chứa/phiên bản nào đó có sẵn trong kho lưu trữ/ một nhánh trong kho lưu trữ. Hệ thống lưu trữ chính xác sẽ cho phép bạn có thể làm việc bất cứ lúc nào, ở bất cứ đâu.

### 1.3.2.4 Giới thiệu về MySQL

#### a) Khái niệm

MySQL là hệ quản trị cơ sở dữ liệu mã nguồn mở và được hoạt động theo mô hình Client-Server. Nó được nhiều nhà phát triển cũng như doanh nghiệp lựa chọn trong quá trình phát triển một dự án mới.



Hình 6: Hệ quản trị cơ sở dữ liệu MySQL

**b) Ưu điểm**

- Có tính bảo mật cao, dễ sử dụng, và đặc biệt là tốc độ truy cập cao.
- MySQL sở hữu nhiều tính năng.
- Có khả năng mở rộng cơ sở dữ liệu khi cần thiết và có thể xử lý được nhiều dữ liệu cùng một lúc.
- Tiết kiệm chi phí.

**c) Nhược điểm**

- Độ tin cậy của một vài chức năng thấp.
- Dung lượng bộ nhớ dùng để lưu trữ dữ liệu thấp, khó phù hợp đối với một vài dự án cần lưu trữ cơ sở dữ liệu rất lớn.



#### **1.4 Kết luận chương 1**

Chương này đã giới thiệu về phân đặt vấn đề, từ đó giúp cho bạn đọc hiểu rõ hơn về lý do lựa chọn đề tài dựa trên bài toán chia sẻ tài liệu. Đồng thời cũng giới thiệu được kiến trúc và các công nghệ sử dụng cho việc xây dựng hệ thống chia sẻ tài liệu. Chương tiếp theo sẽ đi sâu hơn về quy trình phát triển phần mềm để bạn đọc hiểu rõ hơn về hệ thống chia sẻ tài liệu.

## CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

Chương 2 sẽ đi sâu hơn về quy trình phát triển phần mềm của hệ thống cụ thể là xác định yêu cầu, phân tích yêu cầu của hệ thống như biểu đồ ca sử dụng, biểu đồ lớp, kịch bản và biểu đồ giao tiếp cùng như cơ sở dữ liệu và thiết kế hệ thống chia sẻ tài liệu theo kiến trúc hướng dịch vụ.

### 2.1. Phân tích yêu cầu hệ thống

#### 2.1.1. Xác định yêu cầu chức năng

- Hệ thống cần xử lý được các chức năng sau:
  - Chức năng đăng ký tài khoản, đăng nhập, đổi mật khẩu
  - Chức năng xem thông tin người dùng, cập nhật thông tin người dùng
  - Chức năng quản lý người dùng: Tạo, xem sửa, xóa người dùng
  - Chức năng quản lý danh mục: Tạo, xem sửa, xóa danh mục
  - Chức năng quản lý tài liệu: Tạo, xem sửa, xóa tài liệu
  - Chức năng quản lý bình luận: Tạo, xem sửa, xóa bình luận
  - Chức năng upload tài liệu lên kho dùng google drive
  - Chức năng bình luận, đánh giá tài liệu
  - Chức năng xem tài liệu đã upload, xem chi tiết tài liệu
  - Chức năng tìm kiếm tài liệu theo tên tài liệu
- Hệ thống cần lưu trữ
  - Hệ thống cần lưu trữ thông tin về người dùng, tài liệu, danh mục, bình luận
  - Hệ thống cần lưu trữ thông tin thông báo của hệ thống

#### 2.1.2. Xác định yêu cầu phi chức năng

| Yêu cầu phi chức năng | Mô tả                           | Ví dụ   |
|-----------------------|---------------------------------|---|
| Thao tác              | Môi trường hoạt động            | Hệ thống hoạt động tốt trên chrome, coccoc, edge  |
| Hiệu năng             | Tốc độ, độ tin cậy của hệ thống | Cơ sở dữ liệu của hệ thống được cập nhật theo thời gian thực<br>Thời gian phản hồi của API là 3s, của trang web tối đa là 10s |
| Bảo mật               | Cơ chế xác thực người dùng      | Sử dụng JWT(Json Web Token) làm access token đính kèm vào header khi gọi API để thực thi cơ chế xác thực người dùng           |

Bảng 2: Yêu cầu phi chức năng của hệ thống

**2.1.3. Xác định các tác nhân của hệ thống**

| STT | Tên Actor | Mô tả   |
|-----|-----------|---|
| 1   | User      | Là người dùng truy cập vào hệ thống để thao tác các chức năng xem danh sách tài liệu theo danh mục, Upload tài liệu lên kho, bình luận, đánh giá tài liệu,... |
| 2   | Admin     | Là người quản trị để quản lý tài liệu, quản lý người dùng, quản lý danh mục, quản lý bình luận  |

*Bảng 3: Các tác nhân của hệ thống*

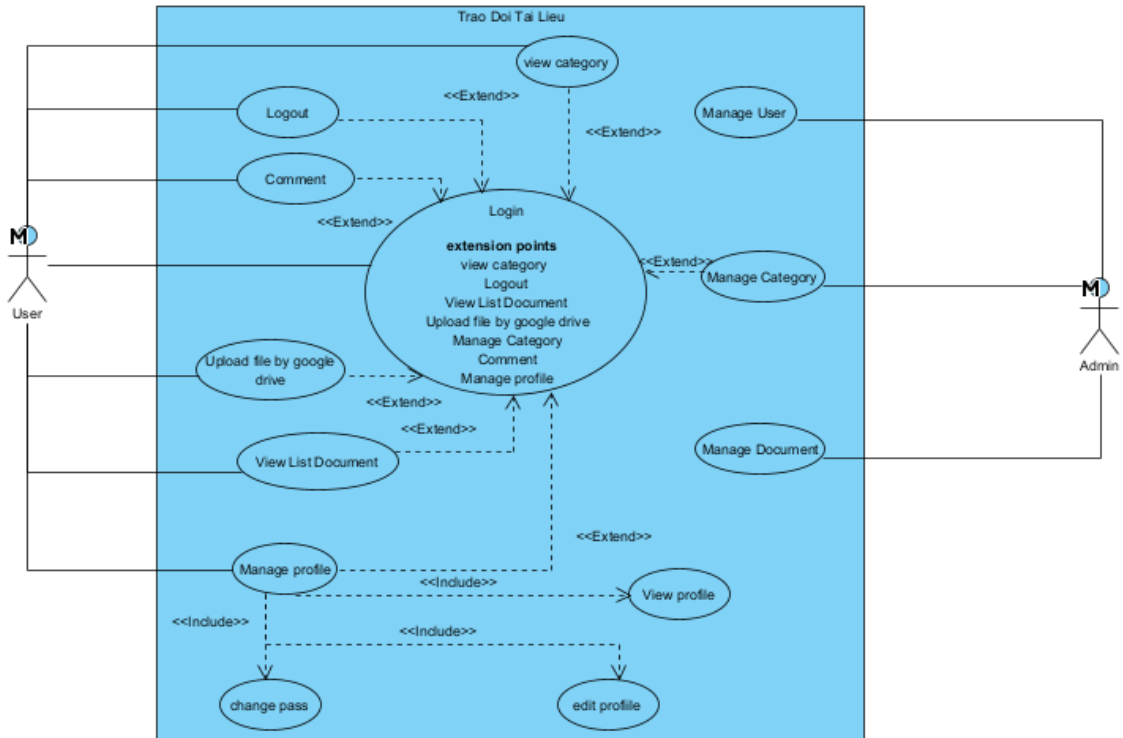
**2.1.4. Xác định và mô tả các ca sử dụng**

| STT | Tên use case                 | Mô tả   |
|-----|------------------------------|---|
| 1   | Quản lý người dùng           | Người quản trị xem, thêm, sửa, xóa người dùng   |
| 2   | Quản lý danh mục             | Người quản trị xem, thêm, sửa, xóa danh mục   |
| 3   | Quản lý tài liệu             | Người quản trị xem, thêm, sửa, xóa tài liệu   |
| 4   | Quản lý bình luận            | Người quản trị xem, thêm, sửa, xóa bình luận  |
| 5   | Upload tài liệu lên kho      | Người dùng Chọn tài liệu từ các nền tảng sau đó upload tài liệu lên kho tài liệu hệ thống |
| 6   | Quản lý thông tin người dùng | Người dùng xem thông tin chi tiết, cập nhật thông tin người dùng                          |
| 7   | Bình luận                    | Người dùng viết bình luận về một tài liệu   |

*Bảng 4: Mô tả các ca sử dụng*

## 2.2. Biểu đồ Use case

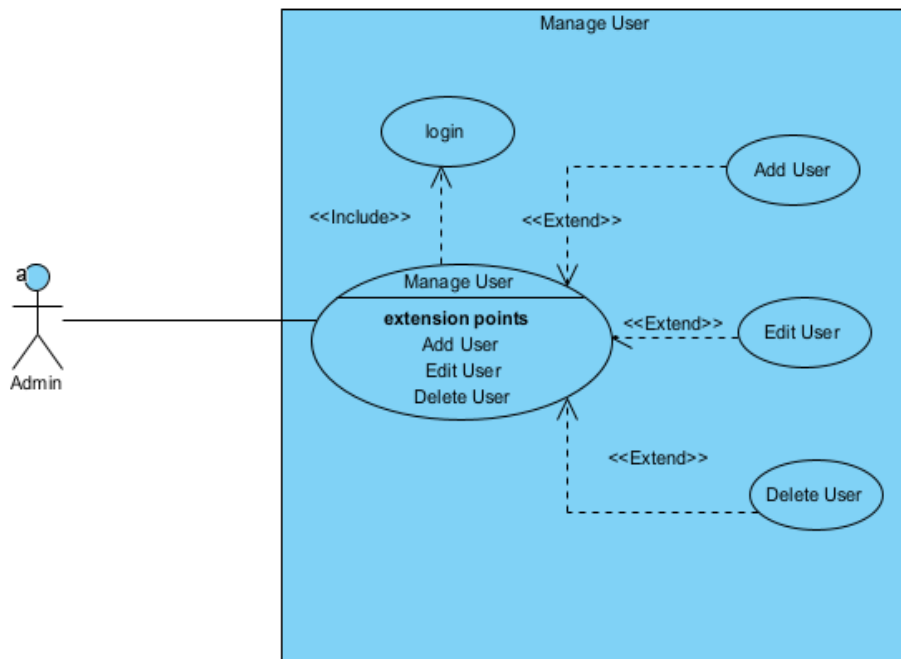
### 2.2.1 Biểu đồ Use case tổng quan



Hình 7: Sơ đồ use case tổng quát

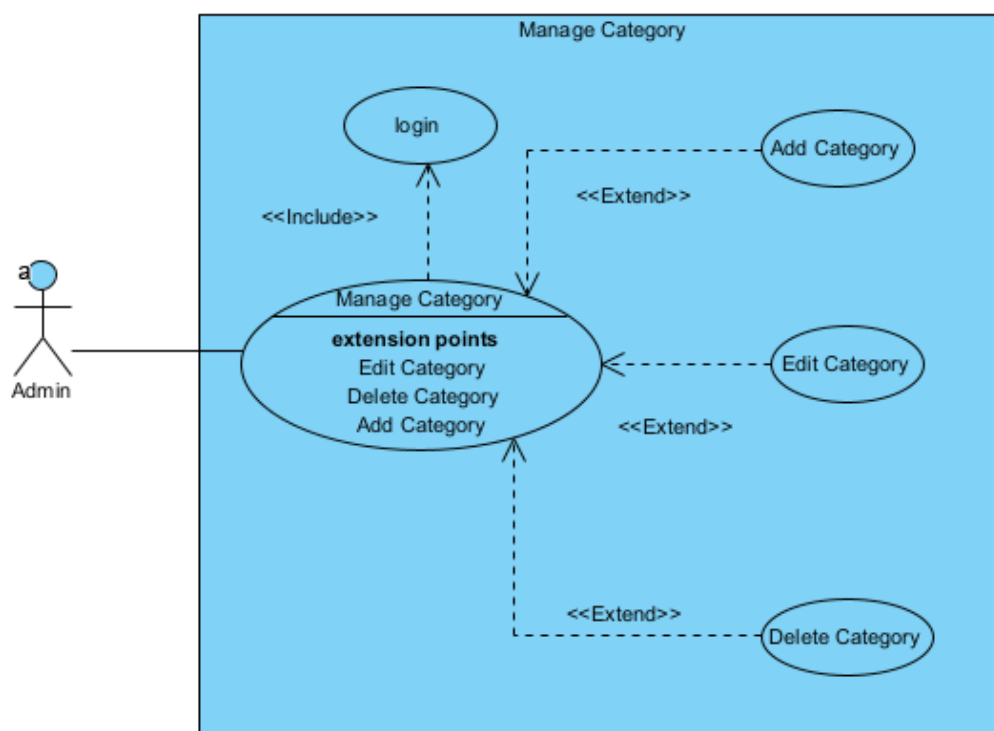
## 2.2.2 Biểu đồ Use case phân rã

### 2.2.2.1. Phân rã Use case Quản lý người dùng



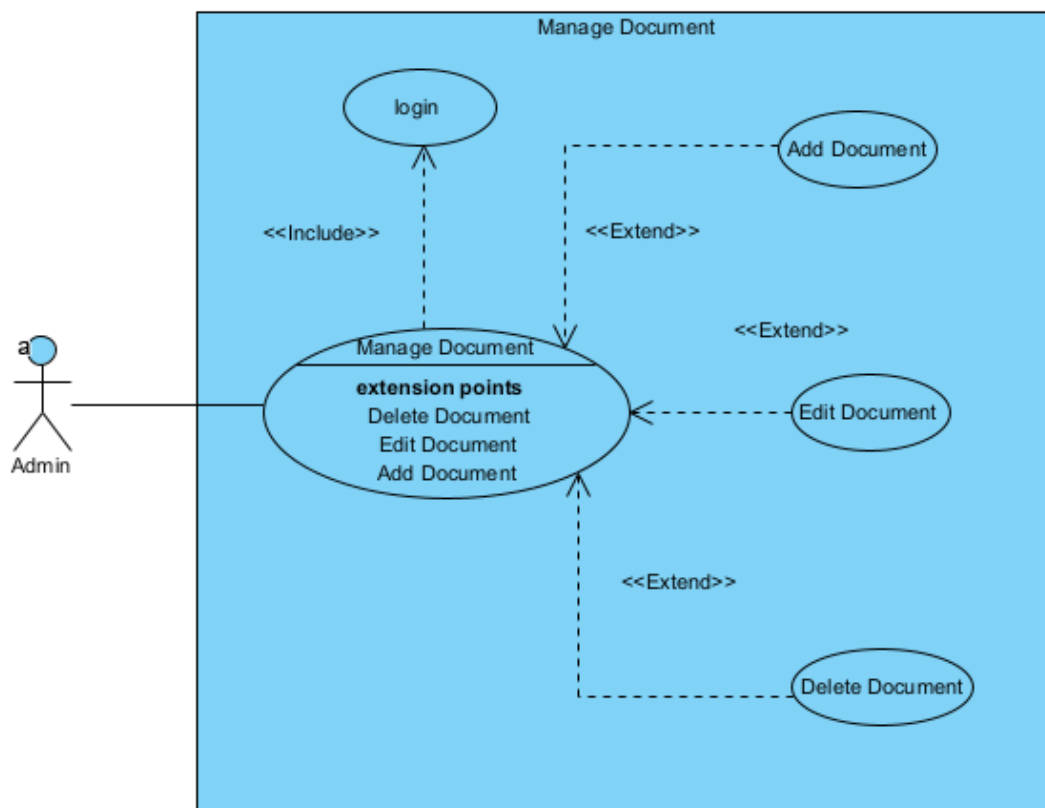
Hình 8: Sơ đồ use case Quản lý người dùng

### 2.2.2.2. Phân rã Use case quản lý danh mục



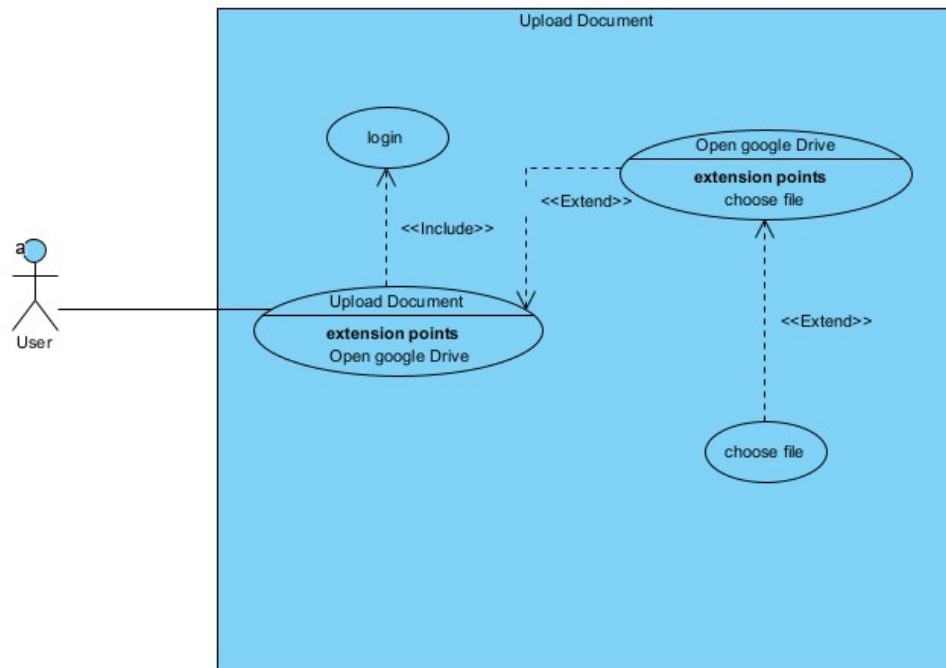
Hình 9: Sơ đồ use case Quản lý danh mục

### 2.2.2.3. Phân rã Use case quản lý tài liệu



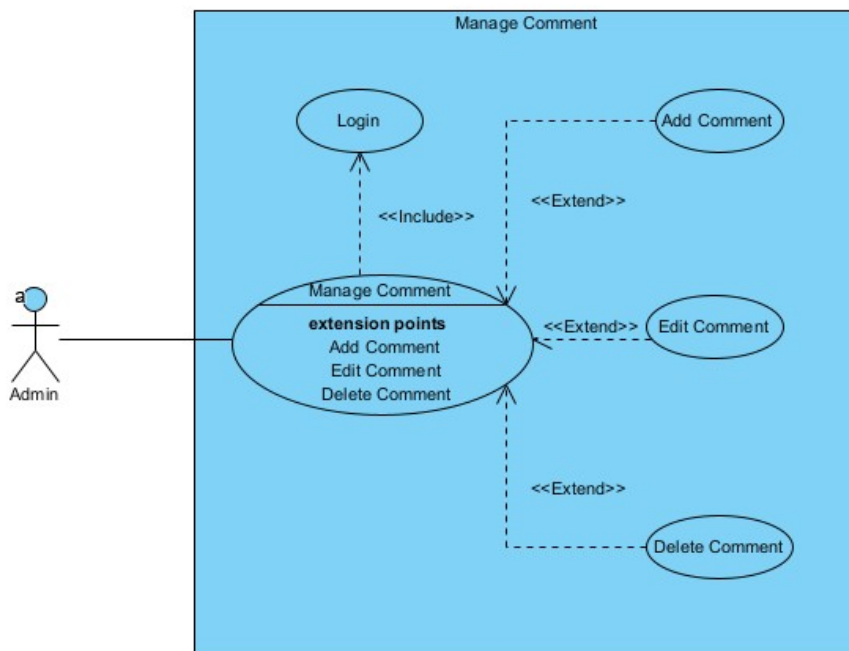
Hình 10: Sơ đồ use case Quản lý tài liệu

### 2.2.2.4. Phân rã Use case Upload tài liệu lên kho



Hình 11: Sơ đồ use case Upload tài liệu lên kho

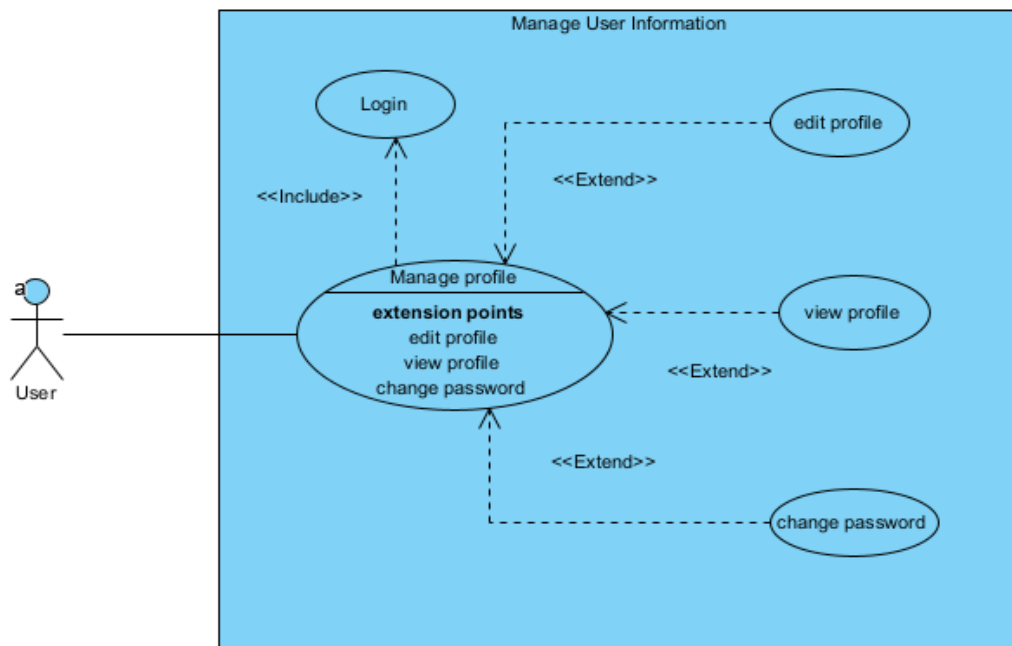
### 2.2.2.5. Phân rã Use case Quản lý bình luận



Hình 12: Sơ đồ use case quản lý bình luận

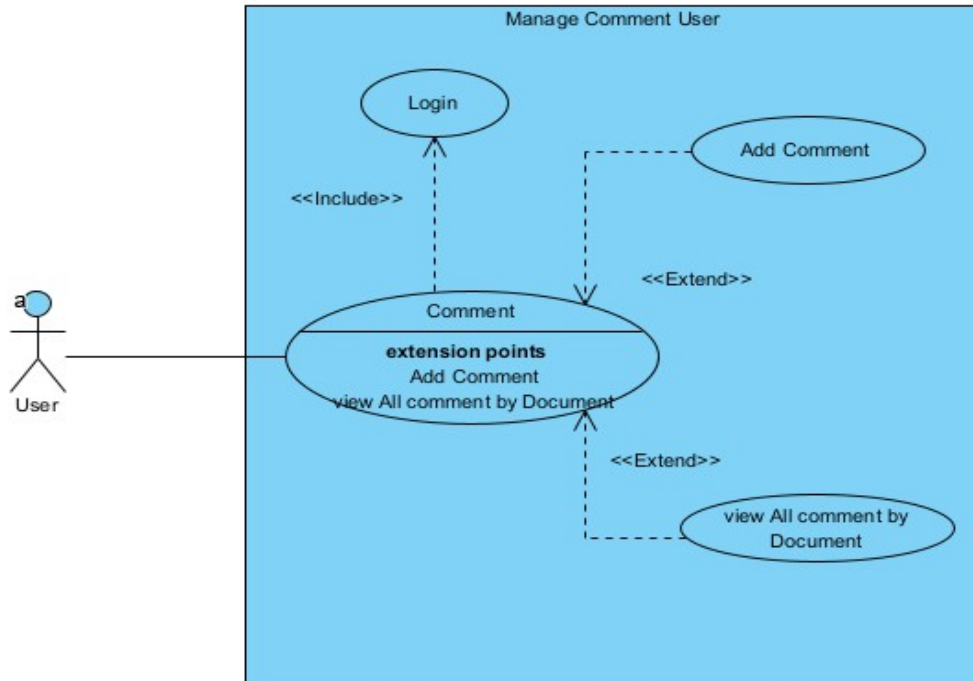


### 2.2.2.6. Phân rã Use case Quản lý thông tin người dùng



Hình 13: Sơ đồ use case quản lý thông tin người dùng

### 2.2.2.7. Phân rã Use case bình luận



Hình 14: Sơ đồ use case bình luận

## 2.3. Kịch bản

### 2.3.1 Kịch bản cho chức năng Đăng nhập

|  |   |
|--|---|
| Tên use case   | Đăng nhập   |
| Tác nhân chính   | Admin, User   |
| Tiền điều kiện   | Người dùng đã truy cập vào hệ thống                           |
| Đảm bảo thành công   | Hệ thống thông báo đăng nhập thành công, chuyển đến trang chủ |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Người dùng truy cập vào website và nhấn nút “login”</li> <li>2. Giao diện đăng nhập hiện ra</li> <li>3. Người dùng nhập thông tin và nhấn nút “Submit”</li> <li>4. Hệ thống kiểm tra thông tin đăng nhập người dùng vừa nhập, thông báo đăng nhập thành công và chuyển tới trang chủ</li> </ol>   |   |
| <p>Ngoại lệ</p> <ol style="list-style-type: none"> <li>3.1. Người dùng nhập thiếu thông tin, ấn nút “Submit” <ol style="list-style-type: none"> <li>3.1.a. Hệ thống thông báo “enter the email” và yêu cầu người dùng điền thông tin vào</li> <li>3.1.b. Hệ thống thông báo “enter the password” và yêu cầu người dùng điền thông tin vào</li> <li>3.1.c. Hệ thống thông báo “enter the email”, “enter the password” và yêu cầu người dùng điền thông tin vào</li> </ol> </li> <li>4.1. Hệ thống thông báo “Tên đăng nhập hoặc mật khẩu không chính xác” và yêu cầu người dùng nhập lại</li> </ol> |   |

*Bảng 5: Kịch bản cho use case đăng nhập*

### 2.3.2 Kịch bản cho chức năng Đăng ký tài khoản

|  |   |
|--|---|
| Tên use case   | Đăng ký tài khoản   |
| Tác nhân chính   | User  |
| Tiền điều kiện   | Người dùng đang ở màn hình đăng ký  |
| Đảm bảo thành công   | Người dùng đăng ký tài khoản thành công nếu email và username chưa tồn tại trong hệ thống |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Người dùng truy cập vào website và nhấn nút “register”</li> <li>2. Giao diện đăng ký tài khoản hiện ra</li> <li>3. Người dùng nhập thông tin cần thiết và nhấn nút “Submit”</li> <li>4. Hệ thống kiểm tra thông tin đăng ký vừa nhập và thông báo đăng ký thành công sau đó chuyển tới trang đăng nhập</li> </ol>   |   |
| <p>Ngoại lệ</p> <ol style="list-style-type: none"> <li>3.1. Người dùng nhập thiếu thông tin, ấn nút “Submit” <ol style="list-style-type: none"> <li>3.1.a. Hệ thống thông báo “enter the email” và yêu cầu người dùng điền thông tin vào</li> <li>3.1.b. Hệ thống thông báo “enter the password” và yêu cầu người dùng điền thông tin vào</li> <li>3.1.c. Hệ thống thông báo “enter the name” và yêu cầu người dùng điền thông tin vào</li> <li>3.1.d. Hệ thống thông báo “enter the address” và yêu cầu người dùng điền thông tin vào</li> <li>3.1.e. Hệ thống thông báo “enter the phone number” và yêu cầu người dùng điền thông tin vào</li> <li>3.1.a. Hệ thống thông báo “password phải ít nhất 8 ký tự gồm chữ hoa, chữ thường, số và ký tự đặc biệt” và yêu cầu người dùng điền thông tin vào</li> </ol> </li> </ol> |   |

*Bảng 6: Kịch bản cho use case đăng ký tài khoản*

### 2.3.3 Kịch bản cho chức năng Đăng xuất

|  |   |
|--|---|
| Tên use case   | Đăng xuất   |
| Tác nhân chính   | User  |
| Tiền điều kiện   | Người dùng đã đăng nhập vào hệ thống                                |
| Đảm bảo thành công   | Hệ thống thông báo đăng xuất thành công, trở lại màn hình đăng nhập |
| Chuỗi sự kiện chính  |   |
| <ol style="list-style-type: none"><li>1. Người dùng truy cập vào website và nhấn vào biểu tượng hình người trên trang chủ</li><li>2. Giao diện thông tin người dùng hiện ra</li><li>5. Người dùng nhấn nút “Logout”</li><li>6. Hệ thống thông báo đăng xuất thành công</li></ol> |   |

*Bảng 7: Kịch bản cho use case đăng xuất*

### **2.3.4 Kịch bản cho chức năng quản lý người dùng**

#### **2.3.4.1. Kịch bản “Thêm mới người dùng”**

|  |  |
|--|--|
| Tên use case   | Thêm mới người dùng  |
| Tác nhân chính   | Admin  |
| Tiền điều kiện   | Người quản trị đã đăng nhập vào hệ thống                       |
| Đảm bảo thành công   | Hệ thống cho phép người quản trị tạo người dùng mới thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Admin truy cập vào website và đăng nhập bằng tài khoản admin</li> <li>2. Trang quản lý người dùng hiện ra</li> <li>3. Admin chọn chức năng “Add User”</li> <li>4. Giao diện thêm mới người dùng hiện ra</li> <li>5. Admin nhập thông tin cần thiết, nhấn nút “Submit”</li> <li>6. Hệ thống hiển thị thông báo thêm người dùng thành công</li> </ol> |  |
| <p>Ngoại lệ</p> <p>6.1. Hệ thống hiển thị thông báo “Không phản hồi”, yêu cầu nhập lại thông tin người dùng</p>  |  |

*Bảng 8: Kịch bản cho use case thêm mới người dùng*

#### **2.3.4.2. Kịch bản “ Sửa thông tin người dùng”**

|  |  |
|--|--|
| Tên use case   | Sửa thông tin người dùng   |
| Tác nhân chính   | Admin  |
| Tiền điều kiện   | Người quản trị đã đăng nhập vào hệ thống                                 |
| Đảm bảo thành công   | Hệ thống cho phép người quản trị sửa thông tin người dùng mới thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Admin truy cập vào website và đăng nhập bằng tài khoản admin</li> <li>2. Trang quản lý người dùng hiện ra</li> <li>3. Admin chọn chức năng “Edit User”</li> <li>4. Giao diện thêm sửa thông tin người dùng hiện ra</li> <li>5. Admin chỉnh sửa lại thông tin người dùng, nhấn nút “Submit”</li> <li>6. Hệ thống hiển thị thông báo sửa thông tin người dùng thành công</li> </ol> |  |
| <p>Ngoại lệ</p> <p>6.1. Hệ thống hiển thị thông báo “Không phản hồi”, yêu cầu nhập lại thông tin người dùng</p>  |  |

*Bảng 9: Kịch bản cho use case sửa thông tin người dùng*

#### **2.3.4.3. Kịch bản “ Xóa người dùng”**

|   |  |
|---|--|
| Tên use case  | Xóa người dùng   |
| Tác nhân chính  | Admin  |
| Tiền điều kiện  | Người quản trị đã đăng nhập vào hệ thống                   |
| Đảm bảo thành công  | Hệ thống cho phép người quản trị xóa người dùng thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Admin truy cập vào website và đăng nhập bằng tài khoản admin</li> <li>2. Trang quản lý người dùng hiện ra</li> <li>3. Admin chọn chức năng “List User”</li> <li>4. Giao diện danh sách thông tin người dùng hiện ra</li> <li>5. Admin chọn “Delete”</li> <li>6. Hệ thống hiển thị thông báo xóa người dùng thành công</li> </ol> |  |
| <p>Ngoại lệ</p> <ol style="list-style-type: none"> <li>6.1. Hệ thống hiển thị thông báo “Không phản hồi”.</li> </ol>  |  |

*Bảng 10: Kịch bản cho use case xóa người dùng*

### **2.3.5 Kịch bản cho chức năng quản lý danh mục**

#### **2.3.5.1. Kịch bản “Thêm mới danh mục”**



|  |  |
|--|--|
| Tên use case   | Thêm mới danh mục  |
| Tác nhân chính   | Admin  |
| Tiền điều kiện   | Người quản trị đã đăng nhập vào hệ thống                     |
| Đảm bảo thành công   | Hệ thống cho phép người quản trị tạo danh mục mới thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Admin truy cập vào website và đăng nhập bằng tài khoản admin</li> <li>2. Trang quản lý danh mục hiện ra</li> <li>3. Admin chọn chức năng “Add Category”</li> <li>4. Giao diện thêm mới danh mục hiện ra</li> <li>5. Admin nhập thông tin cần thiết, nhấn nút “Submit”</li> <li>6. Hệ thống hiển thị thông báo thêm danh mục thành công</li> </ol> |  |
| <p>Ngoại lệ</p> <p>6.1. Hệ thống hiển thị thông báo “Không phản hồi”, yêu cầu nhập lại thông tin danh mục</p>  |  |

*Bảng 11: Kịch bản cho use case thêm mới danh mục*

### **2.3.5.2. Kịch bản “ Sửa thông tin danh mục”**

|  |  |
|--|--|
| Tên use case   | Sửa thông tin danh mục   |
| Tác nhân chính   | Admin  |
| Tiền điều kiện   | Người quản trị đã đăng nhập vào hệ thống                               |
| Đảm bảo thành công   | Hệ thống cho phép người quản trị sửa thông tin danh mục mới thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Admin truy cập vào website và đăng nhập bằng tài khoản admin</li> <li>2. Trang quản lý danh mục hiện ra</li> <li>3. Admin chọn chức năng “Edit Category”</li> <li>4. Giao diện thêm sửa thông tin danh mục hiện ra</li> <li>5. Admin chỉnh sửa lại thông tin danh mục, nhấn nút “Submit”</li> <li>6. Hệ thống hiển thị thông báo sửa thông tin danh mục thành công</li> </ol> |  |
| <p>Ngoại lệ</p> <p>6.1. Hệ thống hiển thị thông báo “Không phản hồi”, yêu cầu nhập lại thông tin danh mục</p>  |  |

*Bảng 12: Kịch bản cho use case sửa thông tin danh mục*

### **2.3.5.3. Kịch bản “ Xóa danh mục”**

|   |  |
|---|--|
| Tên use case  | Xóa người dùng   |
| Tác nhân chính  | Admin  |
| Tiền điều kiện  | Người quản trị đã đăng nhập vào hệ thống                 |
| Đảm bảo thành công  | Hệ thống cho phép người quản trị xóa danh mục thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Admin truy cập vào website và đăng nhập bằng tài khoản admin</li> <li>2. Trang quản lý danh mục hiện ra</li> <li>3. Admin chọn chức năng “List Category”</li> <li>4. Giao diện danh sách thông tin danh mục hiện ra</li> <li>5. Admin chọn “Delete”</li> <li>6. Hệ thống hiển thị thông báo xóa danh mục thành công</li> </ol> |  |
| <p>Ngoại lệ</p> <ol style="list-style-type: none"> <li>6.1. Hệ thống hiển thị thông báo “Không phản hồi”.</li> </ol>  |  |

*Bảng 13: Kịch bản cho use case xóa danh mục*

### **2.3.6 Kịch bản cho chức năng quản lý tài liệu**

#### **2.3.6.1. Kịch bản “Thêm mới tài liệu”**

|  |  |
|--|--|
| Tên use case   | Thêm mới tài liệu  |
| Tác nhân chính   | Admin  |
| Tiền điều kiện   | Người quản trị đã đăng nhập vào hệ thống                     |
| Đảm bảo thành công   | Hệ thống cho phép người quản trị tạo tài liệu mới thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Admin truy cập vào website và đăng nhập bằng tài khoản admin</li> <li>2. Trang quản lý tài liệu hiện ra</li> <li>3. Admin chọn chức năng “Add Document”</li> <li>4. Giao diện thêm mới tài liệu hiện ra</li> <li>5. Admin nhập thông tin cần thiết, nhấn nút “Submit”</li> <li>6. Hệ thống hiển thị thông báo thêm tài liệu thành công</li> </ol> |  |
| <p>Ngoại lệ</p> <p>6.1. Hệ thống hiển thị thông báo “Không phản hồi”, yêu cầu nhập lại thông tin tài liệu</p>  |  |

*Bảng 14: Kịch bản cho use case thêm mới tài liệu*

### **2.3.6.2. Kịch bản “ Sửa thông tin tài liệu”**

|  |  |
|--|--|
| Tên use case   | Sửa thông tin tài liệu   |
| Tác nhân chính   | Admin  |
| Tiền điều kiện   | Người quản trị đã đăng nhập vào hệ thống                               |
| Đảm bảo thành công   | Hệ thống cho phép người quản trị sửa thông tin tài liệu mới thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Admin truy cập vào website và đăng nhập bằng tài khoản admin</li> <li>2. Trang quản lý tài liệu hiện ra</li> <li>3. Admin chọn chức năng “Edit Document”</li> <li>4. Giao diện thêm sửa thông tin tài liệu hiện ra</li> <li>5. Admin chỉnh sửa lại thông tin tài liệu, nhấn nút “Submit”</li> <li>6. Hệ thống hiển thị thông báo sửa thông tin tài liệu thành công</li> </ol> |  |
| <p>Ngoại lệ</p> <p>6.1. Hệ thống hiển thị thông báo “Không phản hồi”, yêu cầu nhập lại thông tin tài liệu</p>  |  |

*Bảng 15: Kịch bản cho use case sửa thông tin tài liệu*

### 2.3.6.3. Kịch bản “ Xóa tài liệu”

|   |  |
|---|--|
| Tên use case  | Xóa tài liệu   |
| Tác nhân chính  | Admin  |
| Tiền điều kiện  | Người quản trị đã đăng nhập vào hệ thống                 |
| Đảm bảo thành công  | Hệ thống cho phép người quản trị xóa tài liệu thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Admin truy cập vào website và đăng nhập bằng tài khoản admin</li> <li>2. Trang quản lý tài liệu hiện ra</li> <li>3. Admin chọn chức năng “List Document”</li> <li>4. Giao diện danh sách thông tin tài liệu hiện ra</li> <li>5. Admin chọn “Delete”</li> <li>6. Hệ thống hiển thị thông báo xóa tài liệu thành công</li> </ol> |  |
| <p>Ngoại lệ</p> <ol style="list-style-type: none"> <li>6.1. Hệ thống hiển thị thông báo “Không phản hồi”.</li> </ol>  |  |

*Bảng 16: Kịch bản cho use case xóa tài liệu*

### **2.3.7 Kịch bản cho chức năng quản lý bình luận**

#### **2.3.7.1. Kịch bản “Thêm mới bình luận”**

|  |   |
|--|---|
| Tên use case   | Thêm mới bình luận  |
| Tác nhân chính   | Admin   |
| Tiền điều kiện   | Người quản trị đã đăng nhập vào hệ thống                      |
| Đảm bảo thành công   | Hệ thống cho phép người quản trị tạo bình luận mới thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Admin truy cập vào website và đăng nhập bằng tài khoản admin</li> <li>2. Trang quản lý bình luận hiện ra</li> <li>3. Admin chọn chức năng “Add Comment”</li> <li>4. Giao diện thêm mới bình luận hiện ra</li> <li>5. Admin nhập thông tin cần thiết, nhấn nút “Submit”</li> <li>6. Hệ thống hiển thị thông báo thêm bình luận thành công</li> </ol> |   |
| <p>Ngoại lệ</p> <p>6.1. Hệ thống hiển thị thông báo “Không phản hồi”, yêu cầu nhập lại thông tin bình luận</p>   |   |

*Bảng 17: Kịch bản cho use case thêm mới bình luận*

### **2.3.7.2. Kịch bản “ Sửa thông tin bình luận”**

|   |   |
|---|---|
| Tên use case  | Sửa thông tin bình luận   |
| Tác nhân chính  | Admin   |
| Tiền điều kiện  | Người quản trị đã đăng nhập vào hệ thống                                |
| Đảm bảo thành công  | Hệ thống cho phép người quản trị sửa thông tin bình luận mới thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Admin truy cập vào website và đăng nhập bằng tài khoản admin</li> <li>2. Trang quản lý bình luận hiện ra</li> <li>3. Admin chọn chức năng “Edit Comment”</li> <li>4. Giao diện thêm sửa thông tin bình luận hiện ra</li> <li>5. Admin chỉnh sửa lại thông tin bình luận, nhấn nút “Submit”</li> <li>6. Hệ thống hiển thị thông báo sửa thông tin bình luận thành công</li> </ol> |   |
| <p>Ngoại lệ</p> <p>6.1. Hệ thống hiển thị thông báo “Không phản hồi”, yêu cầu nhập lại thông tin bình luận</p>  |   |

*Bảng 18: Kịch bản cho use case sửa thông tin bình luận*

### 2.3.7.3. Kịch bản “ Xóa bình luận”



|   |   |
|---|---|
| Tên use case  | Xóa bình luận   |
| Tác nhân chính  | Admin   |
| Tiền điều kiện  | Người quản trị đã đăng nhập vào hệ thống                  |
| Đảm bảo thành công  | Hệ thống cho phép người quản trị xóa bình luận thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Admin truy cập vào website và đăng nhập bằng tài khoản admin</li> <li>2. Trang quản lý bình luận hiện ra</li> <li>3. Admin chọn chức năng “List Comment”</li> <li>4. Giao diện danh sách thông tin bình luận hiện ra</li> <li>5. Admin chọn “Delete”</li> <li>6. Hệ thống hiển thị thông báo xóa bình luận thành công</li> </ol> |   |
| <p>Ngoại lệ</p> <ol style="list-style-type: none"> <li>6.1. Hệ thống hiển thị thông báo “Không phản hồi”.</li> </ol>  |   |

*Bảng 19: Kịch bản cho use case xóa bình luận*

### 2.3.8. Kịch bản cho chức năng Upload tài liệu lên kho

|   |   |
|---|---|
| Tên use case  | Upload tài liệu lên kho                                 |
| Tác nhân chính  | User  |
| Tiền điều kiện  | Người dùng đang ở màn hình chọn tài liệu                |
| Đảm bảo thành công  | Hệ thống cho phép người dùng upload tài liệu thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Người dùng nhấn vào nút “Upload” rồi chọn “choose from google drive”</li> <li>2. Trang chọn tài khoản google drive hiện ra</li> <li>3. Người dùng chọn tài khoản muốn upload</li> <li>4. Trang google drive cho tài khoản đó hiện ra</li> <li>5. người dùng chọn file cần upload</li> <li>6. Trở về giao diện Upload tài liệu</li> <li>7. Người dùng điền thông tin cần thiết</li> <li>8. Trang hiển thị upload thành công và trở về giao diện danh sách tài liệu</li> </ol> |   |
| <p>Ngoại lệ</p> <ol style="list-style-type: none"> <li>1.1. Hệ thống hiển thị thông báo “Không phản hồi”.</li> <li>2.1. Hệ thống không hiển thị danh sách các file và folder trên google drive</li> <li>8.1. Hệ thống hiển thị thông báo “Không phản hồi” và yêu cầu người dùng điền lại thông tin</li> </ol>   |   |

*Bảng 20: Kịch bản cho use case upload tài liệu lên kho*

### 2.3.9. Kịch bản cho chức năng cập nhật thông tin cá nhân

|   |   |
|---|---|
| Tên use case  | Cập nhật thông tin cá nhân                            |
| Tác nhân chính  | User  |
| Tiền điều kiện  | Người dùng đã đăng nhập vào hệ thống                  |
| Đảm bảo thành công  | Hệ thống cho phép người dùng sửa thông tin thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Người dùng nhấn vào biểu tượng người dùng trên trang chủ</li> <li>1. Giao diện thông tin cá nhân hiện ra</li> <li>2. Người dùng chọn “Edit Information User”</li> <li>3. Giao diện sửa thông tin cá nhân hiện ra</li> <li>4. Người dùng cập nhật thông tin cho phù hợp rồi nhấn “Submit”</li> <li>5. Hệ thống hiển thị sửa thông tin thành công và trở lại giao diện thông tin người dùng</li> </ol> |   |
| <p>Ngoại lệ</p> <p>5.1. Hệ thống hiển thị thông báo “Không phản hồi”, yêu cầu nhập lại thông tin người dùng</p>   |   |

*Bảng 21: Kịch bản cho use case cập nhật thông tin cá nhân*

### 2.3.10. Kịch bản cho chức năng tìm kiếm tài liệu theo tên

|              |                            |
|--------------|----------------------------|
| Tên use case | Tìm kiếm tài liệu theo tên |
|--------------|----------------------------|

|  |   |
|--|---|
| Tác nhân chính   | User  |
| Tiền điều kiện   | Người dùng đã đăng nhập vào hệ thống                      |
| Đảm bảo thành công   | Hệ thống cho phép danh sách tài liệu có liên quan hiện ra |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Người dùng nhập tên tài liệu cần tìm vào ô tìm kiếm và nhấn nút “search”</li> <li>2. Danh sách tài liệu theo tên hiện ra</li> </ol> |   |
| <p>Ngoại lệ</p> <p>2.1. Hệ thống hiển thị thông báo “Không phản hồi” do người dùng không nhập tên hoặc không có tài liệu cần tìm</p>   |   |

*Bảng 22: Kịch bản cho use case tìm kiếm tài liệu theo tên*

### 2.3.11. Kịch bản cho chức năng xem chi tiết tài liệu

|   |  |
|---|--|
| Tên use case  | Xem chi tiết tài liệu  |
| Tác nhân chính  | User   |
| Tiền điều kiện  | Người dùng đã đăng nhập vào hệ thống và vào trang danh sách tài liệu |
| Đảm bảo thành công  | Hệ thống cho phép thông tin chi tiết về tài liệu hiện ra             |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Người dùng nhấn vào một trong các danh mục trên thanh công cụ</li> <li>2. Giao diện danh sách tài liệu theo danh mục hiện ra</li> <li>3. Người dùng chọn một tài liệu bất kỳ và nhấn “Detail”</li> <li>4. Giao diện chi tiết tài liệu hiện ra</li> </ol> |  |
| <p>Ngoại lệ</p> <p>4.1. Hệ thống hiển thị thông báo “Không phản hồi”</p>  |  |

*Bảng 23: Kịch bản cho use case xem chi tiết tài liệu*

### 2.3.12. Kịch bản cho chức năng Viết bình luận

|   |   |
|---|---|
| Tên use case  | viết bình luận  |
| Tác nhân chính  | User  |
| Tiền điều kiện  | Người dùng đã đăng nhập vào hệ thống                      |
| Đảm bảo thành công  | Hệ thống cho phép người dùng tạo bình luận mới thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Người dùng truy cập vào trang danh sách tài liệu</li> <li>2. Trang danh sách tài liệu hiện ra</li> <li>3. Người dùng chọn tài liệu bất kỳ và nhấn “Detail”</li> <li>4. Trang danh sách chi tiết tài liệu hiện ra</li> <li>5. Người dùng nhập thông tin cần thiết, nhấn nút “Submit”</li> <li>6. Hệ thống hiển thị thông báo thêm bình luận thành công</li> </ol> |   |
| <p>Ngoại lệ</p> <p>6.1. Hệ thống hiển thị thông báo “Không phản hồi”, yêu cầu nhập lại thông tin bình luận</p>  |   |

*Bảng 24: Kịch bản cho use case viết bình luận*

### 2.3.13. Kịch bản cho chức năng Thay đổi mật khẩu

|  |  |
|--|--|
| Tên use case   | Thay đổi mật khẩu                                    |
| Tác nhân chính   | User   |
| Tiền điều kiện   | Người quản trị đã đăng nhập vào hệ thống             |
| Đảm bảo thành công   | Hệ thống cho phép người dùng đổi mật khẩu thành công |
| <p>Chuỗi sự kiện chính</p> <ol style="list-style-type: none"> <li>1. Người dùng nhấn vào biểu tượng người dùng trên trang chủ</li> <li>2. Giao diện thông tin cá nhân hiện ra</li> <li>3. Người dùng chọn “Change Password”</li> <li>4. Giao diện thay đổi mật khẩu</li> <li>5. Người dùng cập nhật thông tin cho phù hợp rồi nhấn “Submit”</li> <li>6. Hệ thống hiển thị đổi mật khẩu thành công và trở lại giao diện thông tin người dùng</li> </ol> |  |
| <p>Ngoại lệ</p> <ol style="list-style-type: none"> <li>5.1 Người dùng nhập không đúng mật khẩu cũ</li> <li>6.1. Hệ thống hiển thị thông báo “Không phản hồi”, yêu cầu nhập lại mật khẩu mới, do mật khẩu mới không đạt yêu cầu</li> </ol>  |  |

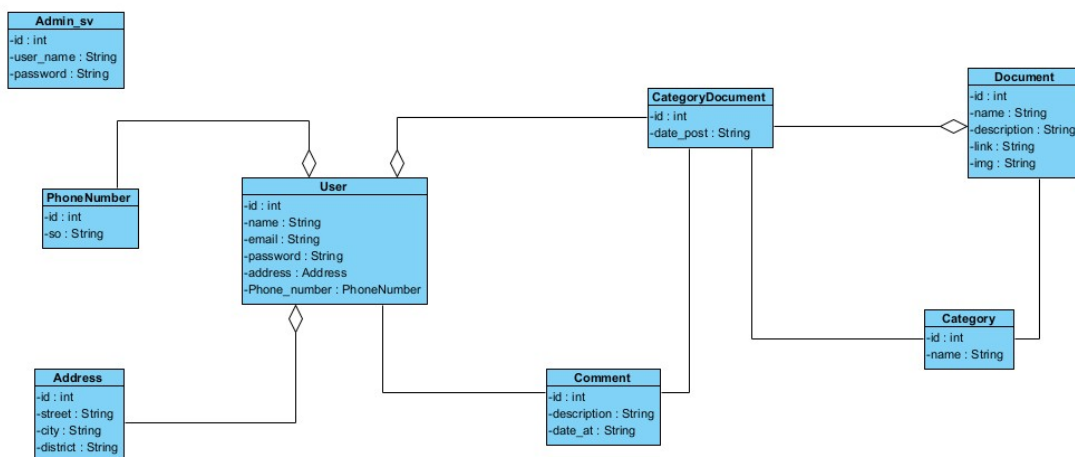
*Bảng 25: Kịch bản cho use case thay đổi mật khẩu*

## 2.4 Biểu đồ lớp

### 2.4.1. Xác định các lớp

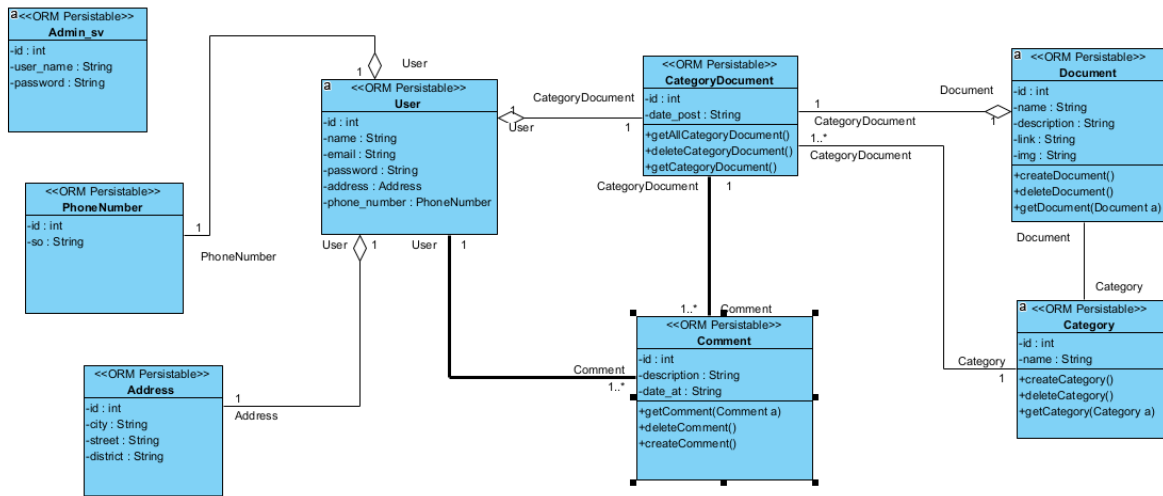
- Ta dùng kỹ thuật trích xuất danh từ trong các ca sử dụng và kịch bản, các danh từ thu được là:
  - Người dùng
  - Người quản trị
  - Tài liệu
  - Danh mục
  - Bình luận
- Loại bỏ các danh từ nằm ngoài phạm vi hệ thống, danh từ trùng lặp,... Ta thu được các danh từ sau để làm ứng viên cho các lớp thực thể:
  - Người dùng(User)
  - Người quản trị(Admin)
  - Danh mục(Category)
  - Tài liệu(Document)
  - Bình luận(Comment).
- Mối quan hệ giữa các thực thể
  - + User quan hệ n-n với Document
  - + User quan hệ 1-n với Comment
  - + Document quan hệ 1-n với Comment
  - + Category quan hệ 1-n với Document

### 2.4.2. Biểu đồ lớp phân tích



Hình 15: Biểu đồ lớp phân tích

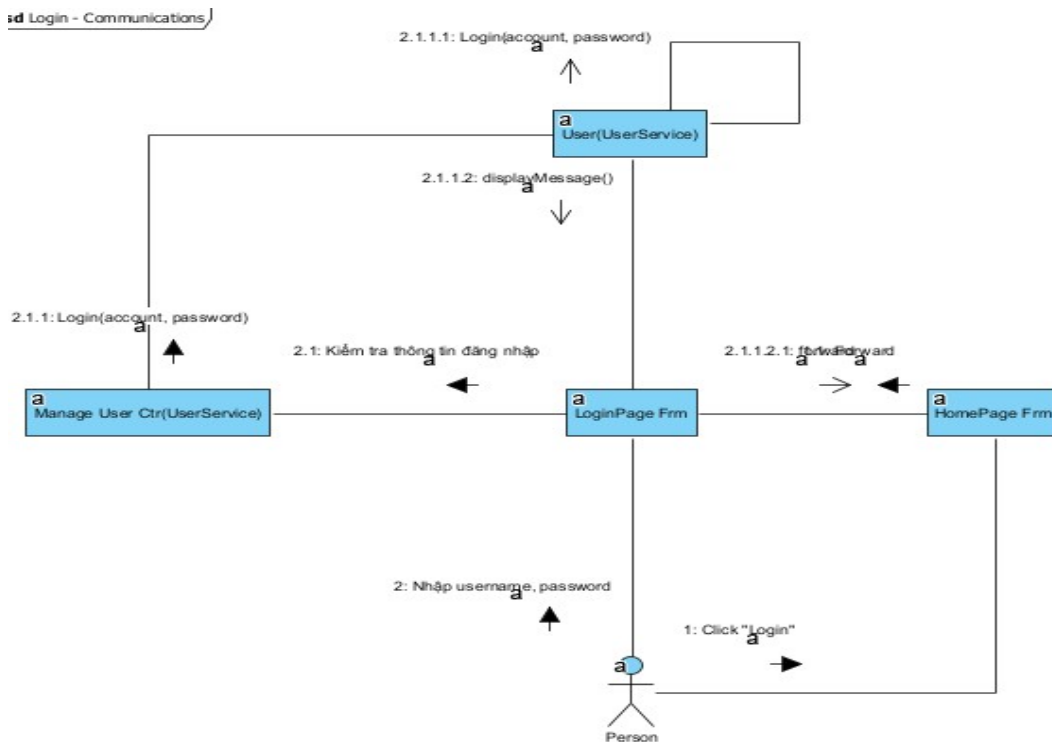
### 2.4.3. Biểu đồ lớp thiết kế



Hình 16: Biểu đồ lớp thiết kế

### 2.5. Biểu đồ giao tiếp

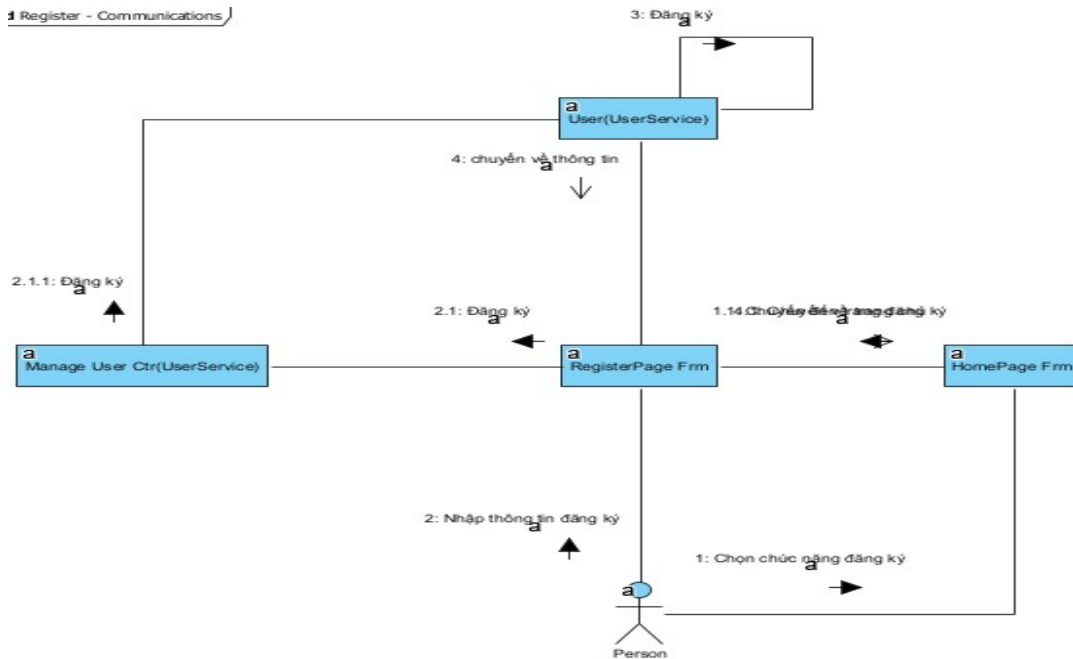
#### 2.5.1. Biểu đồ giao tiếp cho chức năng đăng nhập



Hình 17: Biểu đồ giao tiếp cho chức năng Login



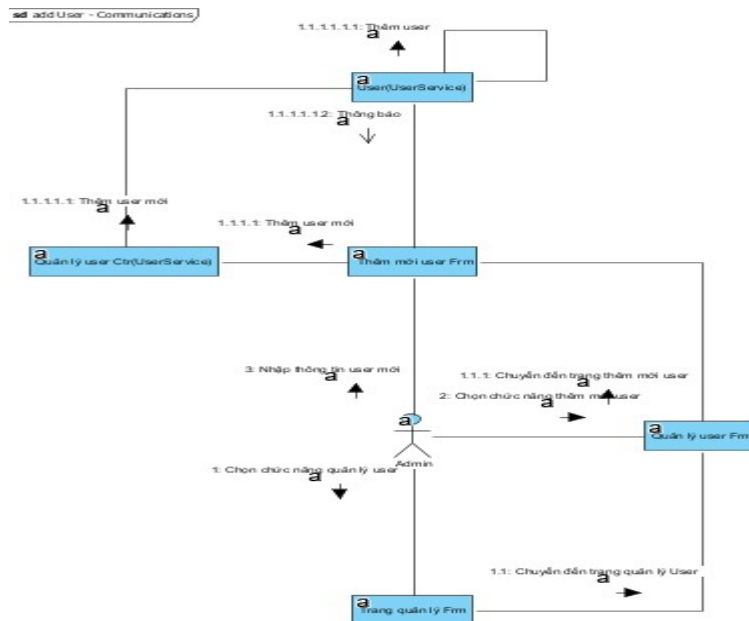
### 2.5.2. Biểu đồ giao tiếp cho chức năng đăng ký



Hình 18: Biểu đồ giao tiếp cho chức năng Register

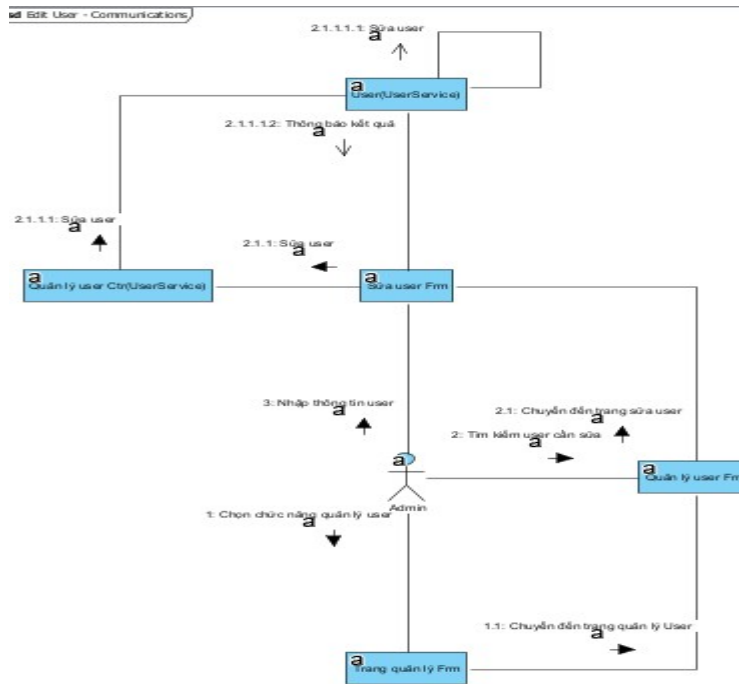
### 2.5.3. Biểu đồ giao tiếp cho chức năng quản lý người dùng

#### 2.5.3.1. Biểu đồ giao tiếp “Thêm người dùng”



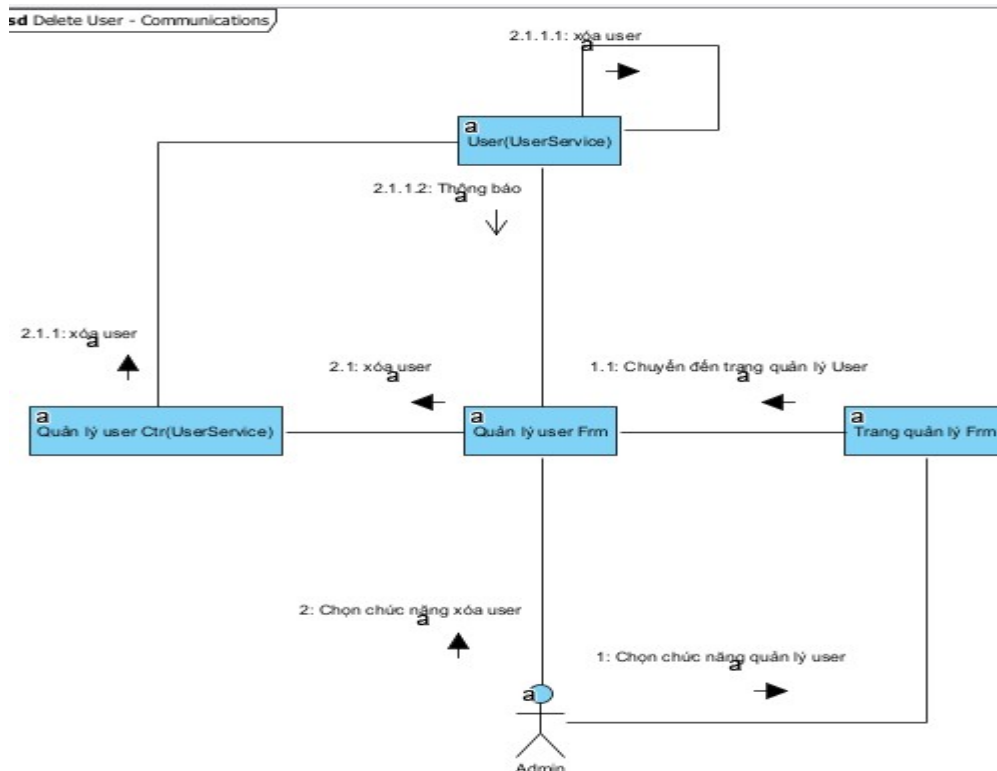
Hình 19: Biểu đồ giao tiếp cho chức năng add user

#### 2.5.3.2. Biểu đồ giao tiếp “Sửa thông tin người dùng”



Hình 20: Biểu đồ giao tiếp cho chức năng edit user

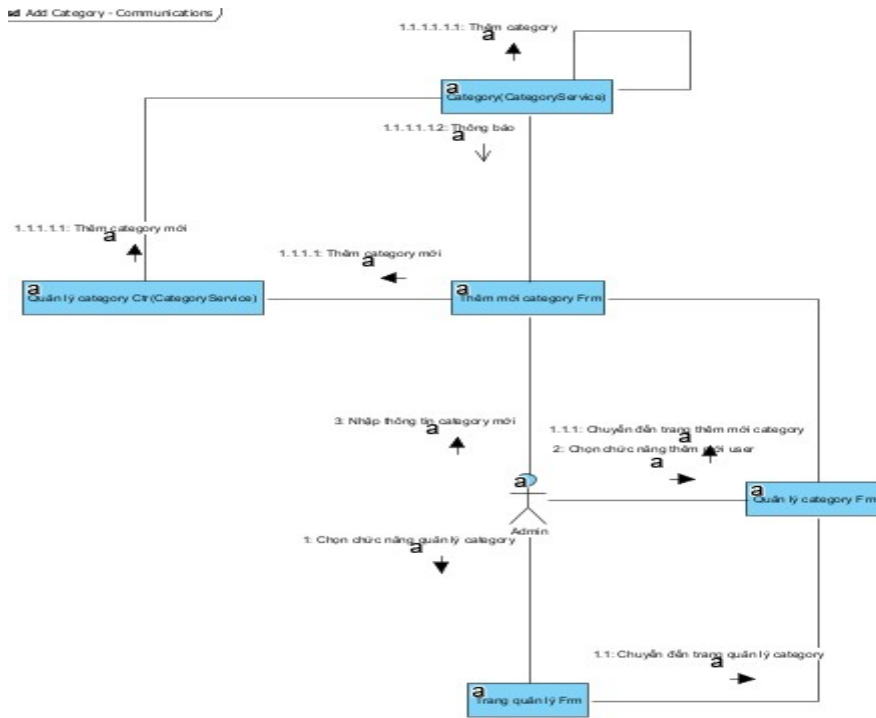
### 2.5.3.3. Biểu đồ giao tiếp “xóa người dùng”



Hình 21: Biểu đồ giao tiếp cho chức năng delete user

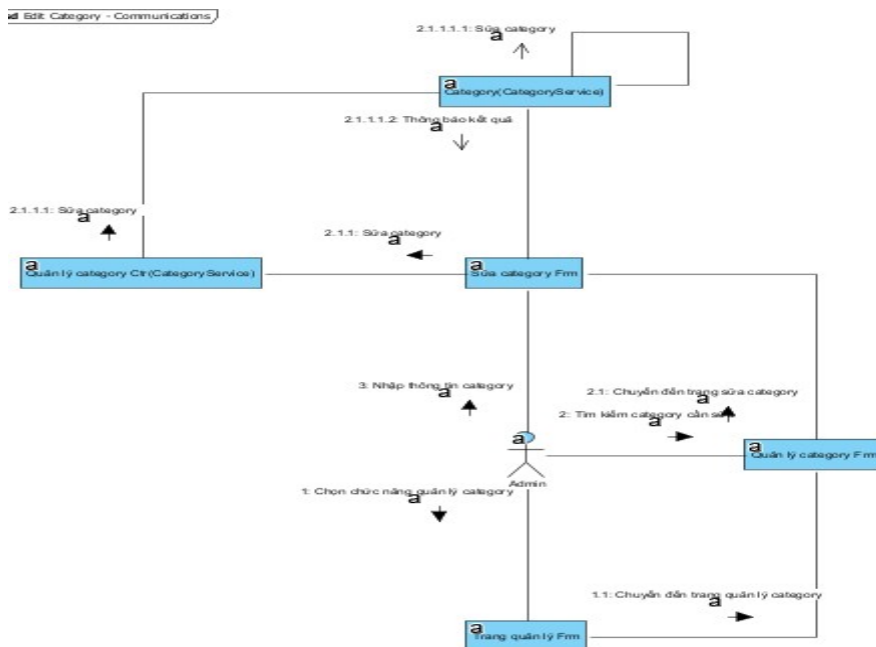
### 2.5.4. Biểu đồ giao tiếp cho chức năng quản lý danh mục

#### 2.5.4.1. Biểu đồ giao tiếp “Thêm danh mục”



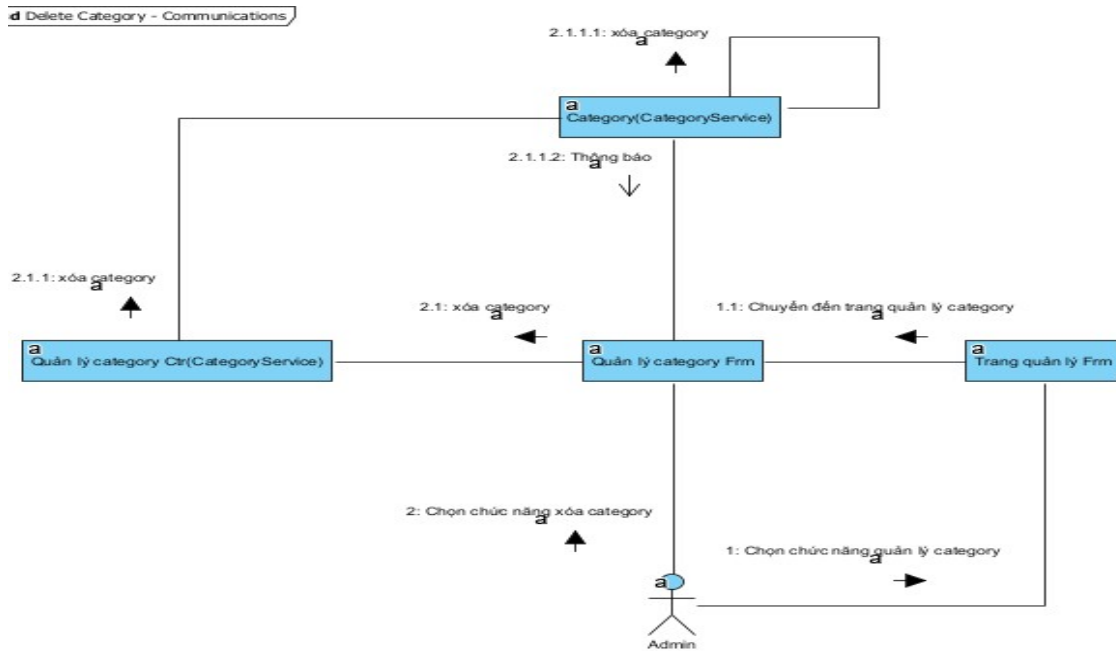
Hình 22: Biểu đồ giao tiếp cho chức năng add category

### 2.5.4.2. Biểu đồ giao tiếp “Sửa thông tin danh mục”



Hình 23: Biểu đồ giao tiếp cho chức năng edit category

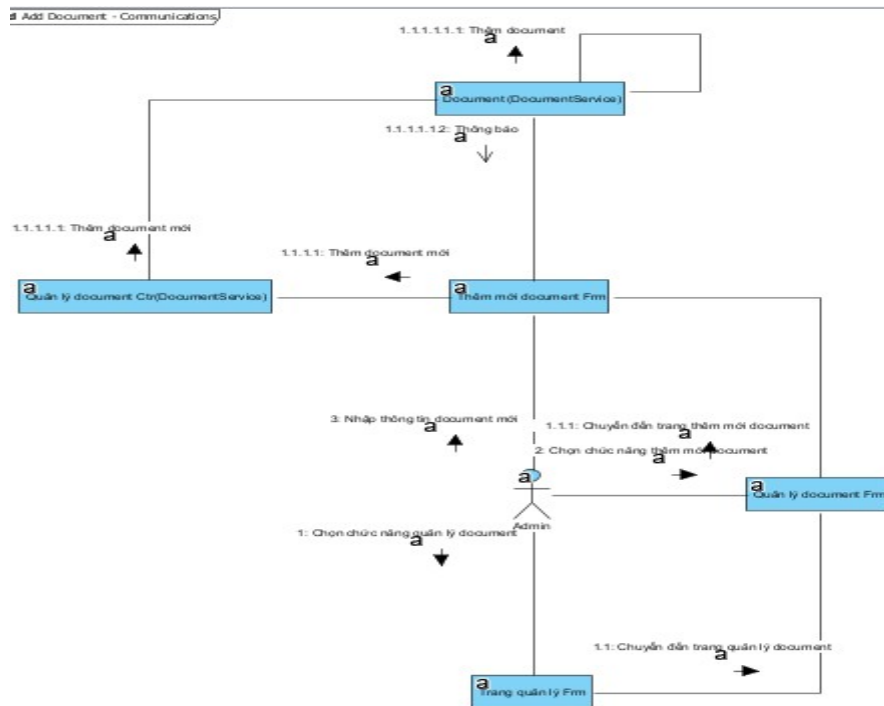
### 2.5.4.3. Biểu đồ giao tiếp “xóa danh mục”



Hình 24: Biểu đồ giao tiếp cho chức năng delete category

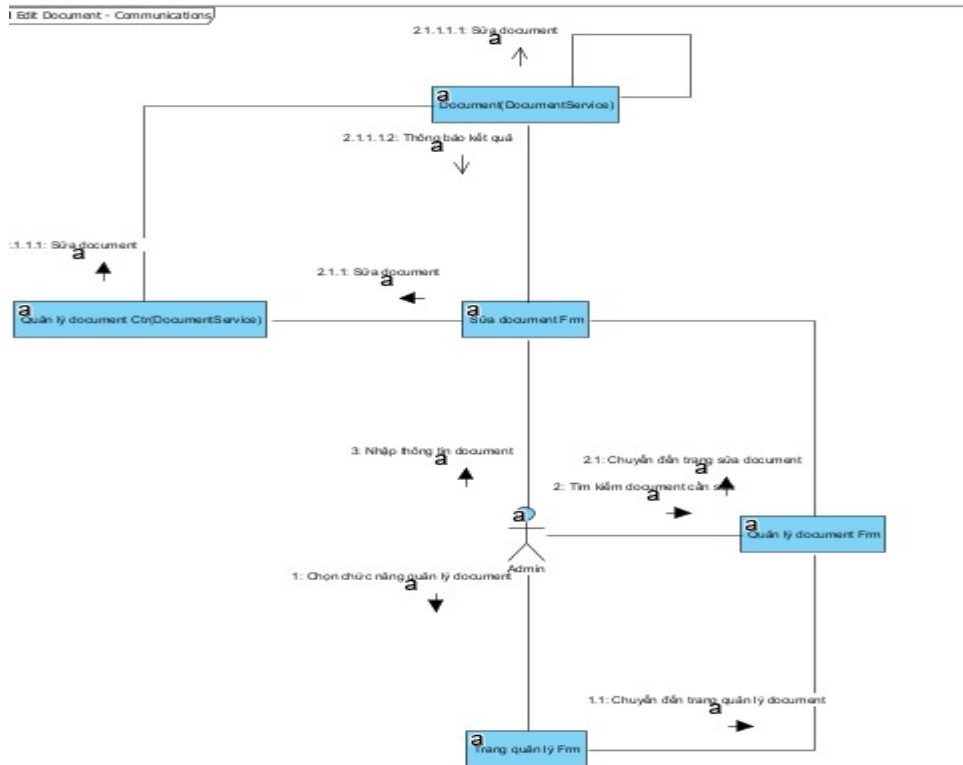
## 2.5.5. Biểu đồ giao tiếp cho chức năng quản lý tài liệu

### 2.5.5.1. Biểu đồ giao tiếp “Thêm tài liệu”



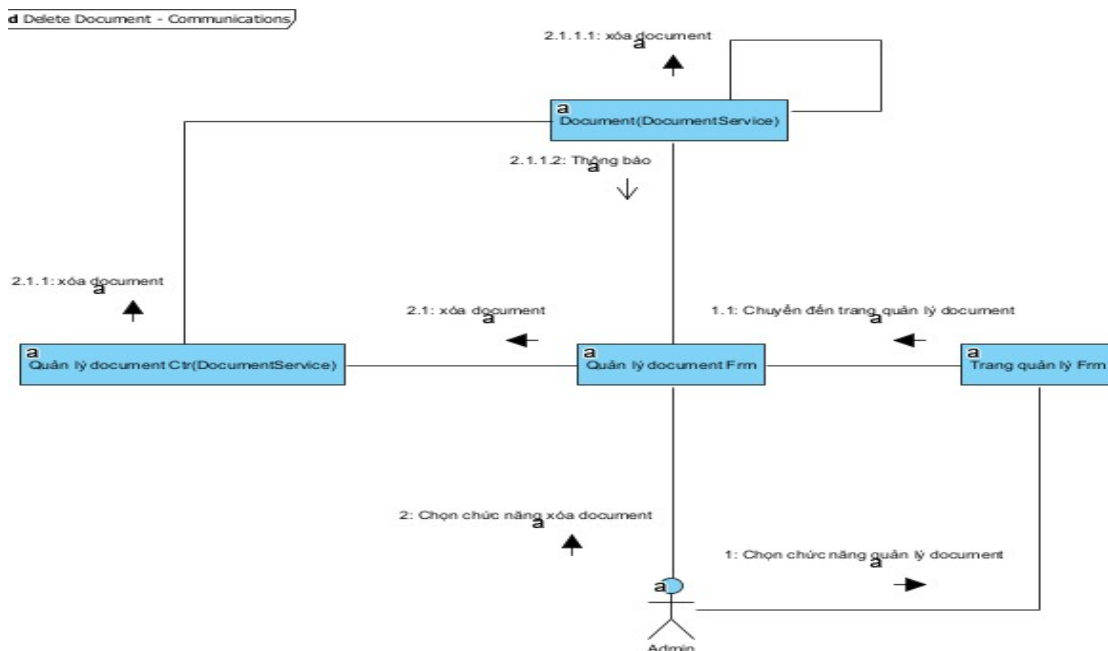
Hình 25: Biểu đồ giao tiếp cho chức năng add document

### 2.5.5.2. Biểu đồ giao tiếp “Sửa thông tin tài liệu”



Hình 26: Biểu đồ giao tiếp cho chức năng edit document

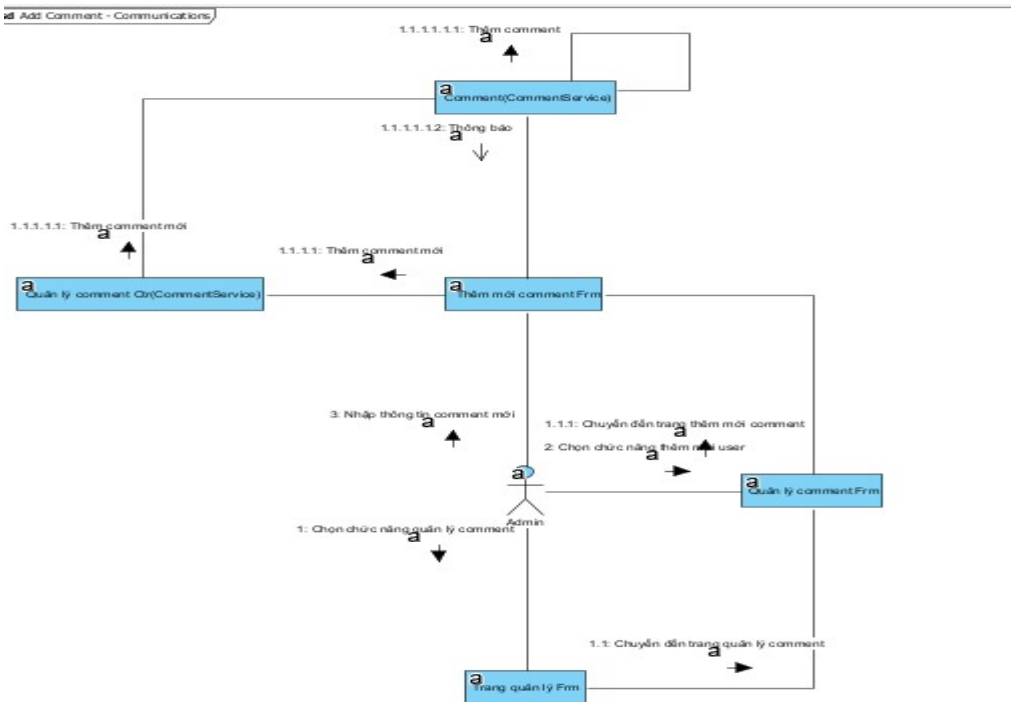
### 2.5.5.3. Biểu đồ giao tiếp “xóa tài liệu”



Hình 27: Biểu đồ giao tiếp cho chức năng delete document

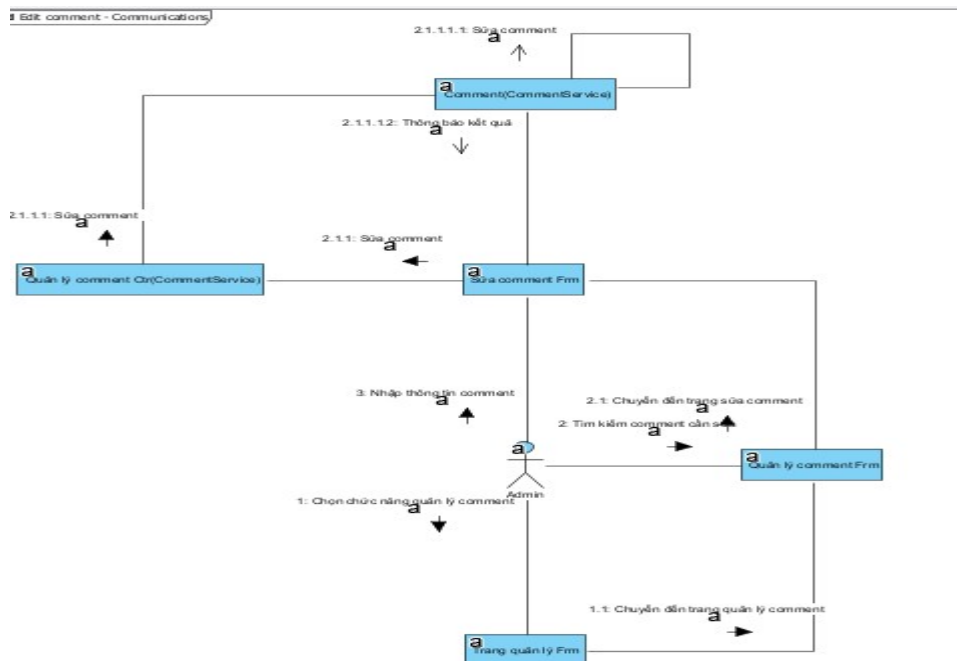
## 2.5.6. Biểu đồ giao tiếp cho chức năng quản lý bình luận

### 2.5.6.1. Biểu đồ giao tiếp “Thêm bình luận”



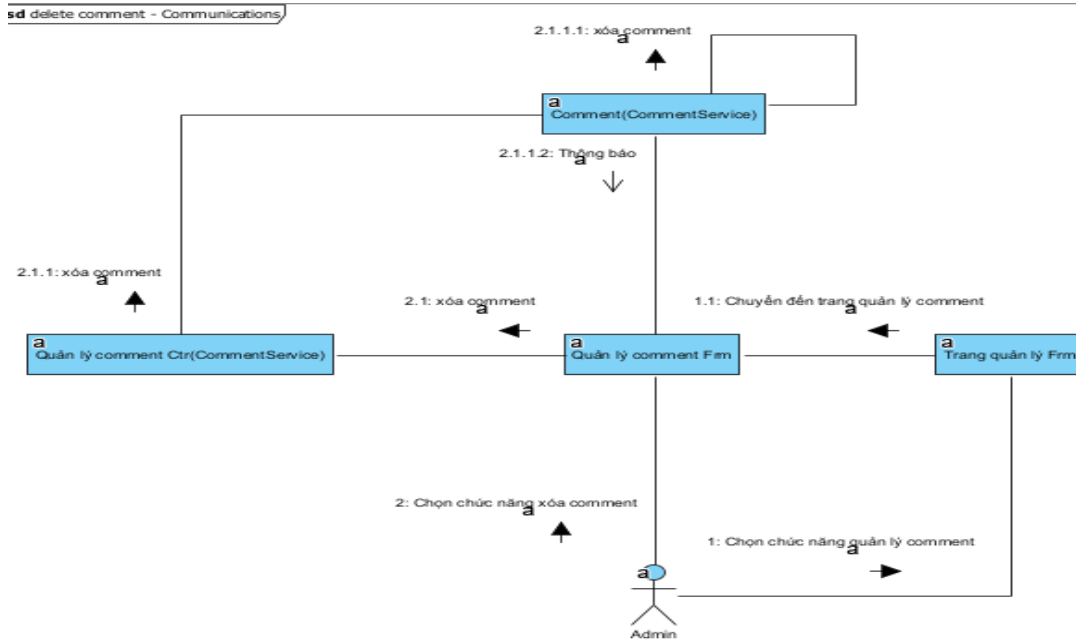
Hình 28: Biểu đồ giao tiếp cho chức năng add comment

### 2.5.6.2. Biểu đồ giao tiếp “Sửa thông tin bình luận”



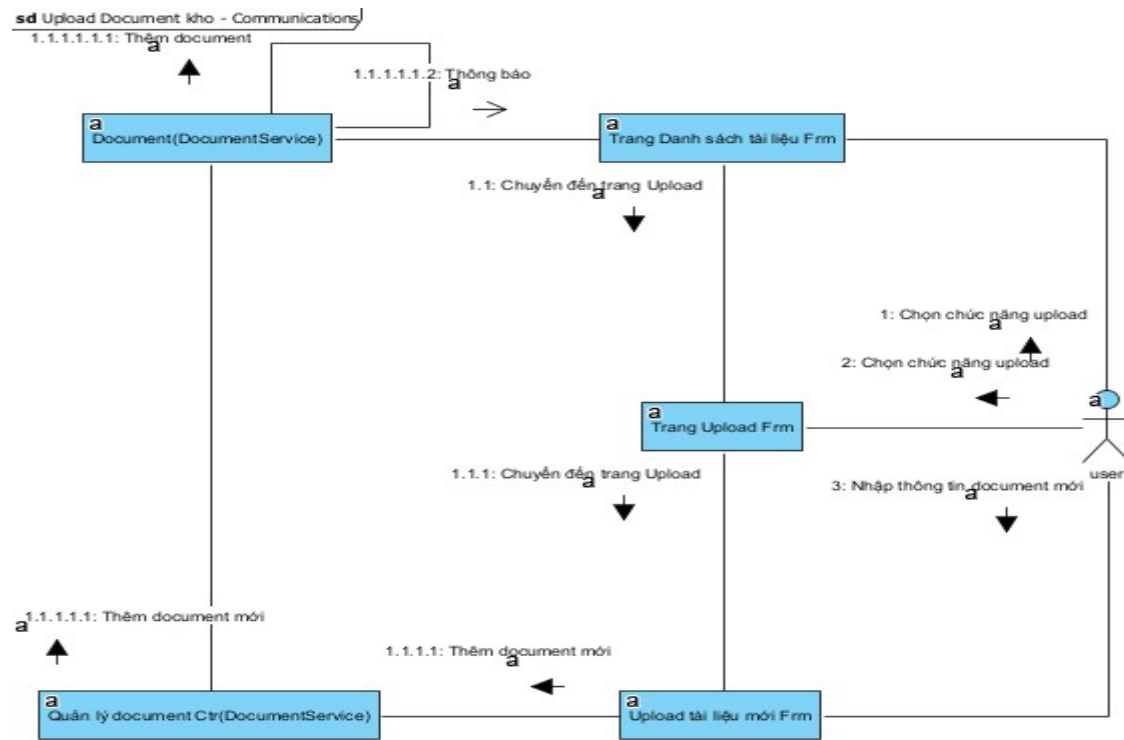
Hình 29: Biểu đồ giao tiếp cho chức năng edit comment

2.5.6.3. Biểu đồ giao tiếp “xóa bình luận”



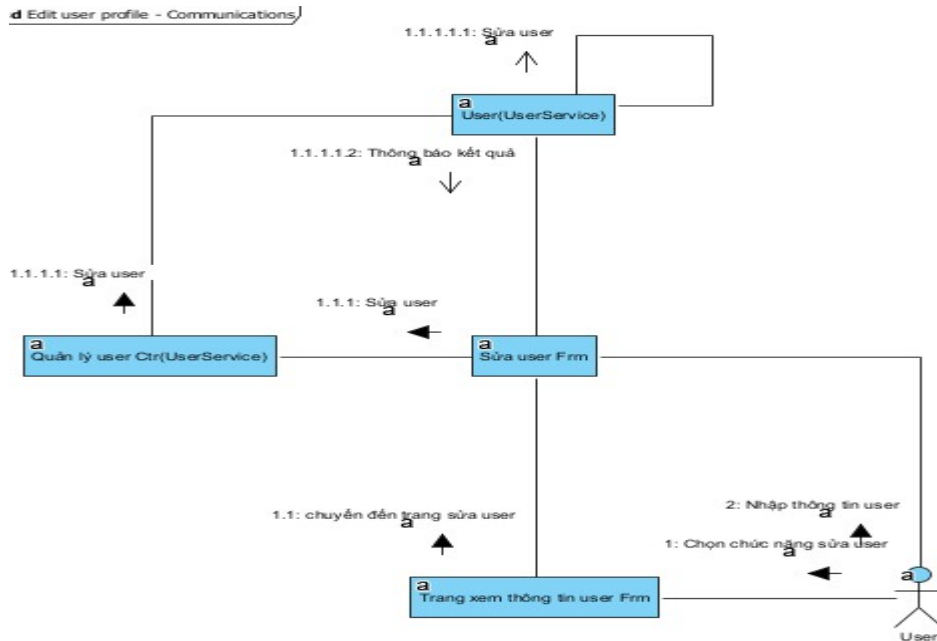
Hình 30: Biểu đồ giao tiếp cho chức năng delete comment

2.5.7. Biểu đồ giao tiếp cho chức năng Upload tài liệu lên kho



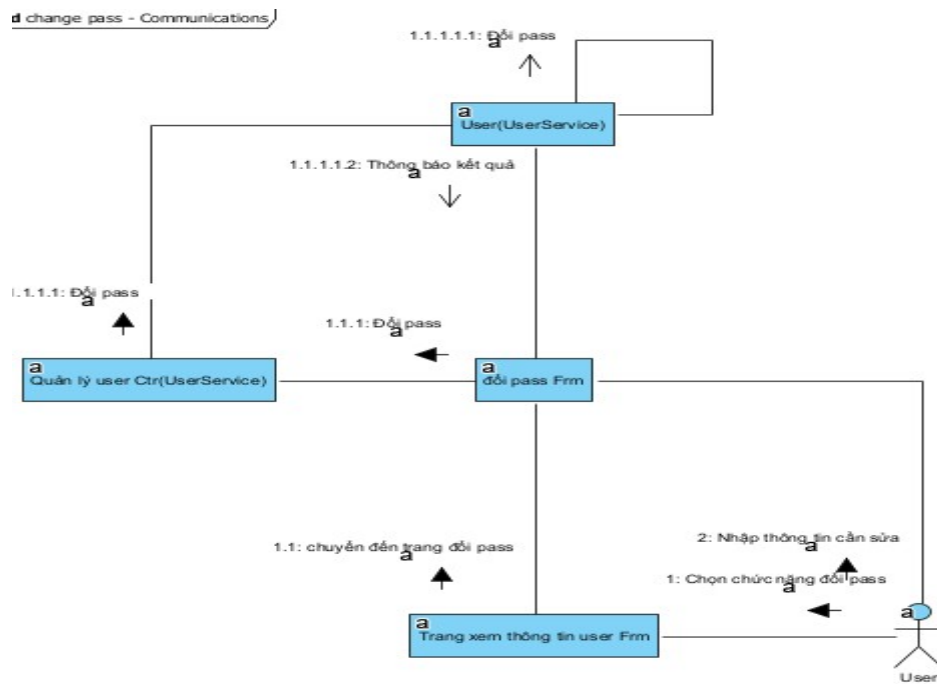
Hình 31: Biểu đồ giao tiếp cho chức năng upload tài liệu lên kho

2.5.8. Biểu đồ giao tiếp cho chức năng cập nhật thông tin người dùng



Hình 32: Biểu đồ giao tiếp cho chức năng cập nhật thông tin người dùng

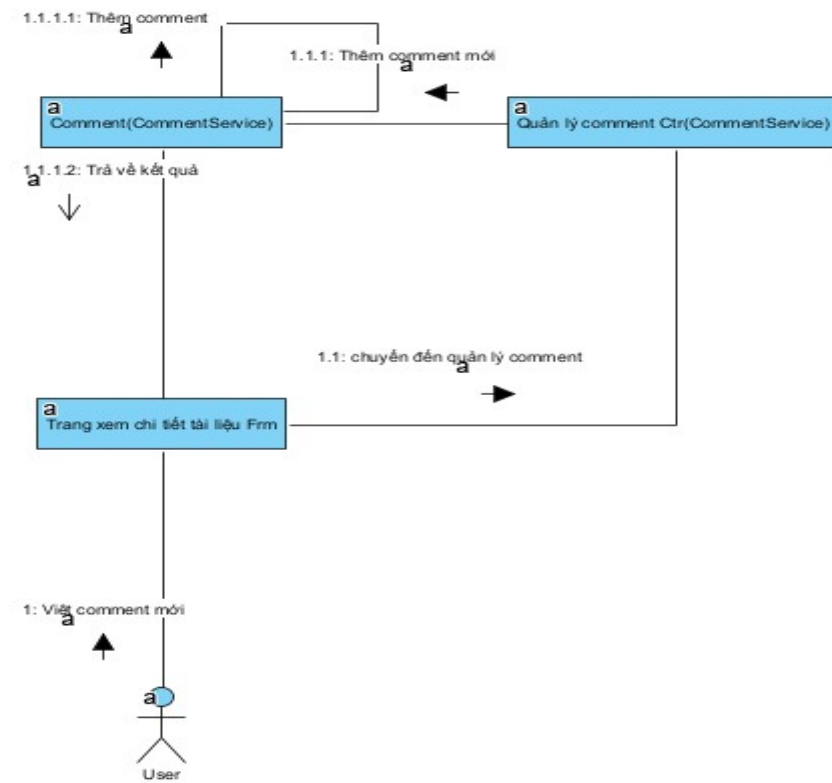
2.5.9. Biểu đồ giao tiếp cho chức năng đổi mật khẩu



Hình 33: Biểu đồ giao tiếp cho chức năng đổi mật khẩu

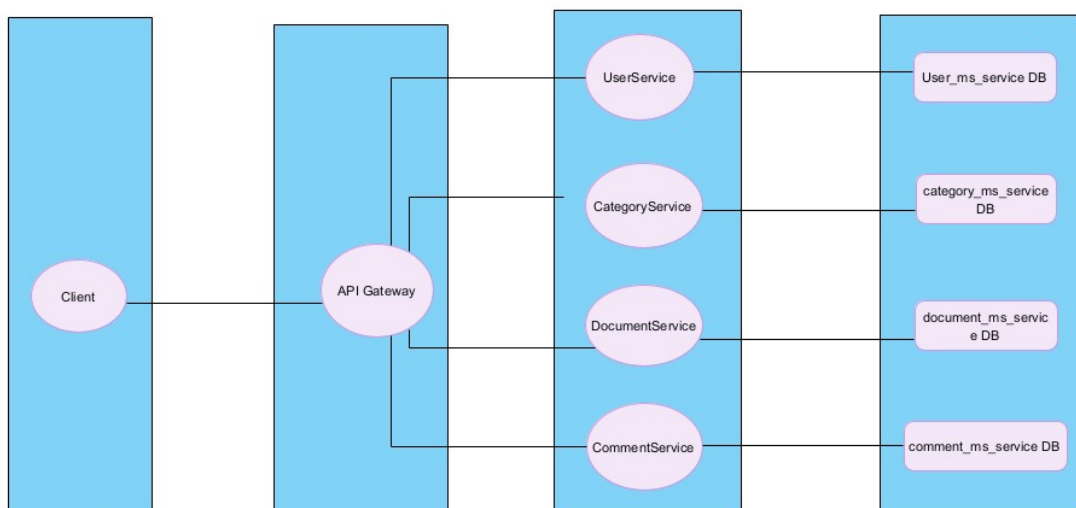


### 2.5.10. Biểu đồ giao tiếp cho chức năng viết bình luận



Hình 34: Biểu đồ giao tiếp cho chức năng viết bình luận

### 2.6. Phân rã thành các Service



Hình 35: Biểu đồ phân rã các service

## 2.7. Mô tả cấu trúc dữ liệu

- a) UserService  
 - Bảng Users

| Tên trường      | Kiểu dữ liệu | Not Null | Ràng buộc  | Mô tả                    |
|-----------------|--------------|----------|------------|--------------------------|
| id              | int          | Yes      | khóa chính | Mã người dùng            |
| name            | varchar(255) | Yes      |            | tên người dùng           |
| email           | varchar(255) | Yes      |            | email người dùng         |
| password        | varchar(255) | Yes      |            | mật khẩu người dùng      |
| address_id      | int          | Yes      | khóa ngoại | địa chỉ người dùng       |
| phone_number_id | int          | Yes      | khóa ngoại | Số điện thoại người dùng |

*Bảng 26: Bảng Users*

- Bảng Address

| Tên trường | Kiểu dữ liệu | Not Null | Ràng buộc  | Mô tả      |
|------------|--------------|----------|------------|------------|
| id         | int          | Yes      | khóa chính | Mã address |
| street     | varchar(255) | Yes      |            | đường      |
| district   | varchar(255) | Yes      |            | quận       |
| city       | varchar(255) | Yes      |            | thành phố  |

*Bảng 27: Bảng Address*

- Bảng PhoneNumber

| Tên trường | Kiểu dữ liệu | Not Null | Ràng buộc  | Mô tả         |
|------------|--------------|----------|------------|---------------|
| id         | int          | Yes      | khóa chính | Mã điện thoại |
| so         | varchar(255) | Yes      |            | số điện thoại |

*Bảng 28: Bảng PhoneNumber*

b) DocumentService

- Bảng Document

| Tên trường  | Kiểu dữ liệu | Not Null | Ràng buộc  | Mô tả          |
|-------------|--------------|----------|------------|----------------|
| id          | int          | Yes      | khóa chính | Mã tài liệu    |
| name        | varchar(255) | Yes      |            | tên tài liệu   |
| description | varchar(255) | Yes      |            | mô tả tài liệu |
| link        | varchar(255) | Yes      |            | link tài liệu  |
| img         | varchar(255) | Yes      |            | ảnh tài liệu   |

*Bảng 29: Bảng Document*

- Bảng Category\_Document

| Tên trường  | Kiểu dữ liệu | Not Null | Ràng buộc  | Mô tả         |
|-------------|--------------|----------|------------|---------------|
| id          | int          | Yes      | khóa chính | Mã bảng       |
| document_id | int          | Yes      |            | mã tài liệu   |
| category_id | int          | Yes      |            | mã danh mục   |
| user_id     | int          | Yes      |            | mã người dùng |
| date_post   | varchar(255) | Yes      |            | ngày upload   |

*Bảng 30: Bảng Category\_Document*

c) CategoryService

- Bảng Category

| Tên trường | Kiểu dữ liệu | Not Null | Ràng buộc  | Mô tả        |
|------------|--------------|----------|------------|--------------|
| id         | int          | Yes      | khóa chính | mã danh mục  |
| name       | varchar(255) | Yes      |            | tên danh mục |

*Bảng 31: bảng Category*

d) AdminService

- Bảng AdminSv

| Tên trường | Kiểu dữ liệu | Not Null | Ràng buộc  | Mô tả               |
|------------|--------------|----------|------------|---------------------|
| id         | int          | Yes      | khóa chính | Mã tài liệu         |
| user_name  | varchar(255) | Yes      |            | tên đăng nhập Admin |
| password   | varchar(255) | Yes      |            | mô tả mật khẩu      |

*Bảng 32: Bảng AdminSv*

e) CommentService

- Bảng Comment

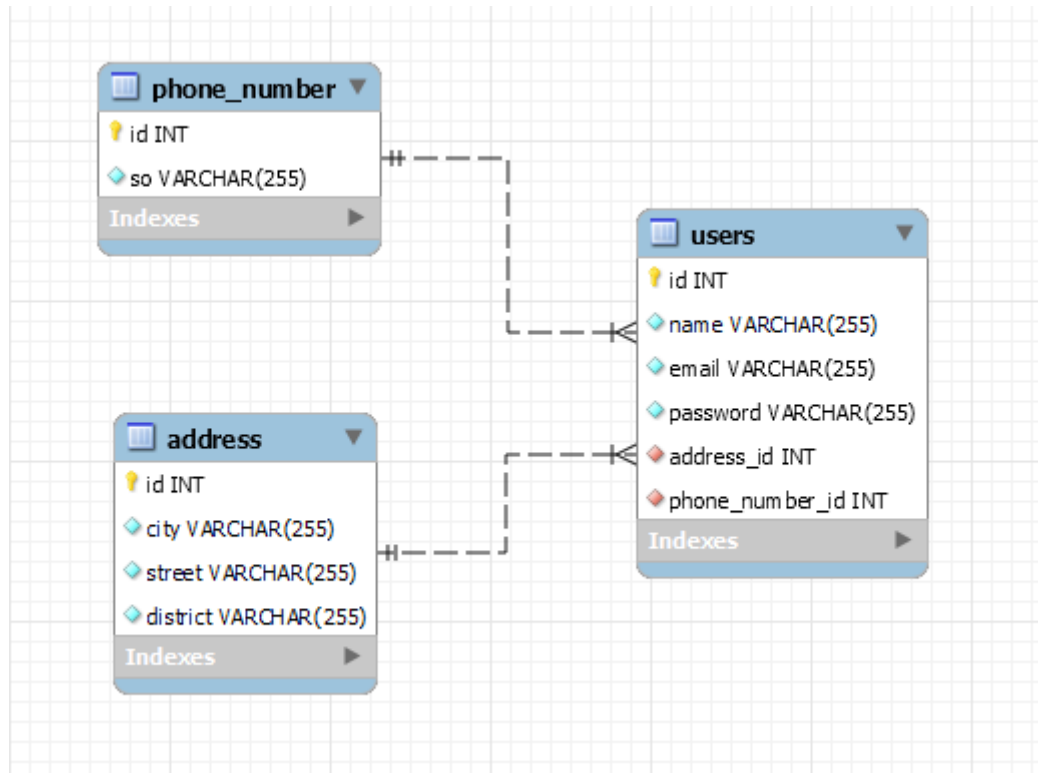
| Tên trường  | Kiểu dữ liệu | Not Null | Ràng buộc  | Mô tả               |
|-------------|--------------|----------|------------|---------------------|
| id          | int          | Yes      | khóa chính | Mã bảng             |
| description | varchar(255) | Yes      |            | mô tả bình luận     |
| date_post   | varchar(255) | Yes      |            | ngày bình luận      |
| catedoc_id  | int          | Yes      |            | mã CategoryDocument |
| user_id     | int          | Yes      |            | mã người dùng       |

*Bảng 33: Bảng Comment*

## 2.8. Cơ sở dữ liệu

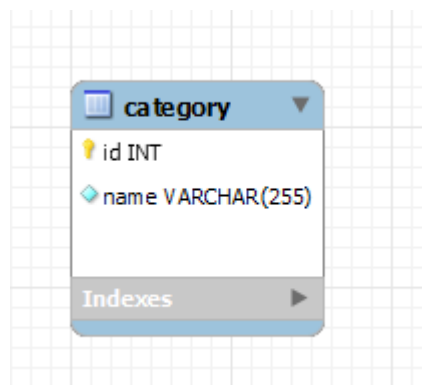
Hệ thống chia thành AdminService, CategoryService, CommentService, UserService, DocumentService. Sau đây là cơ sở dữ liệu của từng Service

### a) UserService



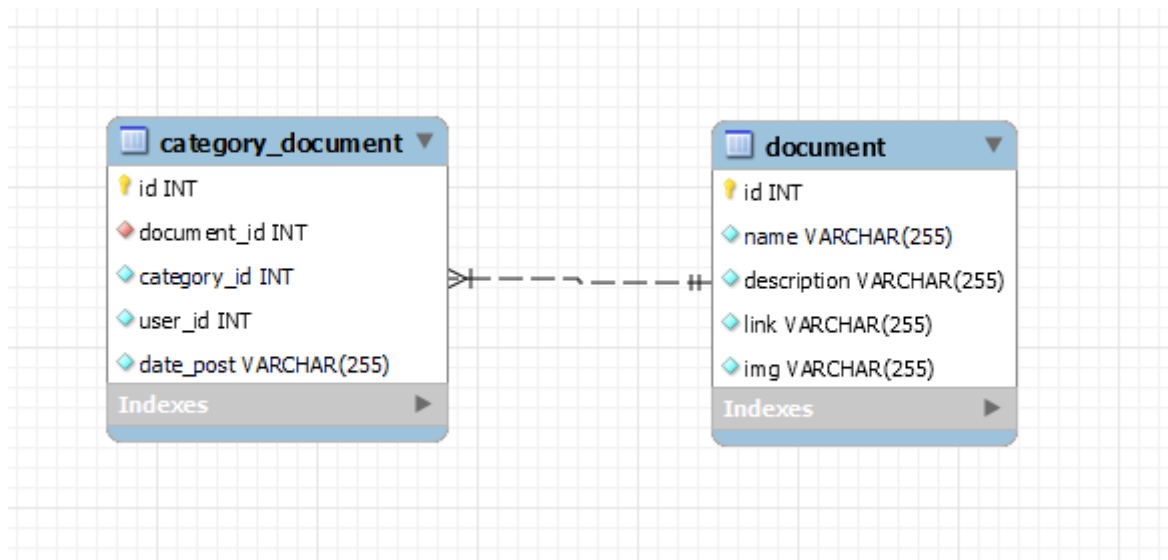
Hình 36: Cơ sở dữ liệu của UserService

### b) CategoryService



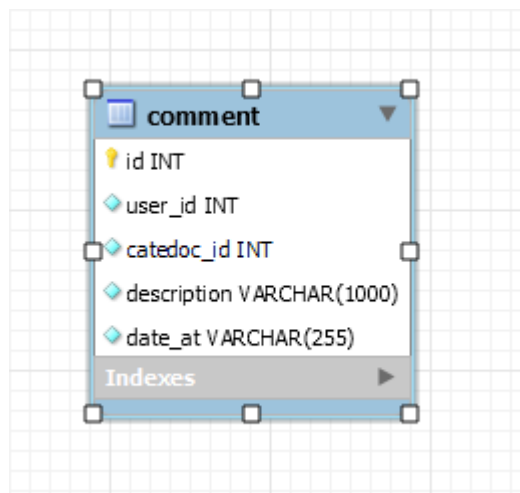
Hình 37: Cơ sở dữ liệu của CategoryService

### c) DocumentService



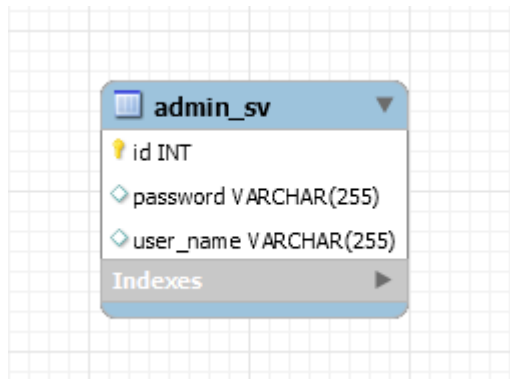
Hình 38: Cơ sở dữ liệu của DocumentService

d) CommentService



Hình 39: Cơ sở dữ liệu của CommentService

e) AdminService



Hình 40: Cơ sở dữ liệu của AdminService

## 2.9. Kết luận chương 2

Trong chương này ta đã trình bày các bước xác định yêu cầu, phân tích yêu cầu và thiết kế hệ thống của hệ thống chia sẻ tài liệu theo kiến trúc hướng dịch vụ. Các biểu đồ UML liên quan như biểu đồ các ca sử dụng, phân rã các ca sử dụng, kịch bản các ca sử dụng, biểu đồ lớp, biểu đồ giao tiếp ứng với chức năng của hệ thống cũng như mô tả được cấu trúc dữ liệu và cơ sở dữ liệu của dự án. Trong chương tiếp theo ta sẽ trình bày về quá trình cài đặt hệ thống.

## CHƯƠNG 3: CÀI ĐẶT HỆ THỐNG

Trong chương 3 ta sẽ tìm hiểu về cách cài đặt hệ thống, yêu cầu về phần mềm, phần cứng để cài đặt hệ thống chia sẻ tài liệu theo kiến trúc hướng dịch vụ cũng như kết quả của hệ thống.

## 3.1. Các yêu cầu về cài đặt về hệ thống

### 3.1.1. Yêu cầu về phần mềm

- Hệ thống chạy tốt trên các trình duyệt, đáp ứng yêu cầu về mặt chức năng
- Yêu cầu về môi trường phần mềm: Java jdk 11
- Phiên bản Spring Boot sử dụng: Version 2.7.3
- Cấu hình cơ sở dữ liệu: Sử dụng hệ quản trị cơ sở dữ liệu MySQL Workbench.
- IDE: Sử dụng IntelliJ Idea.

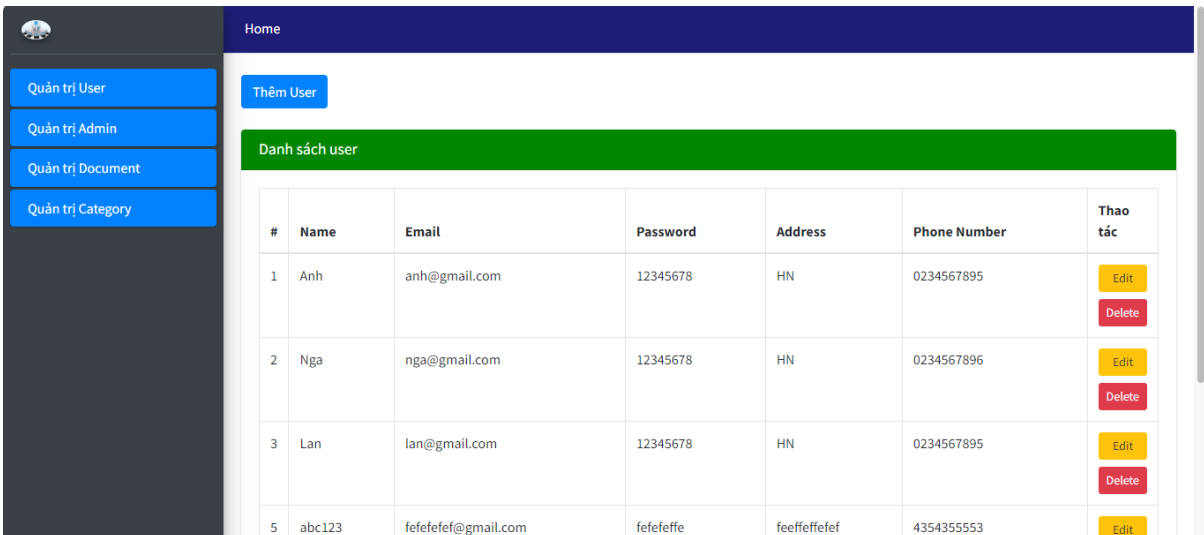
### 3.1.2. Yêu cầu về phần cứng

- Môi trường triển khai: Hệ điều hành window 10, 11.
- RAM: lớn hơn 2GB

## 3.2. Giao diện website trao đổi tài liệu dựa trên kiến trúc hướng dịch vụ

### 1. Giao diện Quản lý người dùng

- Danh sách user



| # | Name   | Email             | Password | Address   | Phone Number | Thao tác                                       |
|---|--------|-------------------|----------|-----------|--------------|--|
| 1 | Anh    | anh@gmail.com     | 12345678 | HN        | 0234567895   | <a href="#">Edit</a><br><a href="#">Delete</a> |
| 2 | Nga    | nga@gmail.com     | 12345678 | HN        | 0234567896   | <a href="#">Edit</a><br><a href="#">Delete</a> |
| 3 | Lan    | lan@gmail.com     | 12345678 | HN        | 0234567895   | <a href="#">Edit</a><br><a href="#">Delete</a> |
| 5 | abc123 | fefefef@gmail.com | fefefefe | feeffefef | 4354355553   | <a href="#">Edit</a>                           |

Hình 41: Giao diện danh sách user

- Sửa user



The screenshot shows a web interface for editing a user. On the left is a dark sidebar with four blue buttons: 'Quản trị User', 'Quản trị Admin', 'Quản trị Document', and 'Quản trị Category'. The main content area has a green header 'Edit User'. Below it are several form fields: 'Name' (containing 'gvđgdg'), 'Email' (containing 'dgdgdg@gmail.com'), 'Password' (containing '\*\*\*\*\*'), 'Address' (containing 'sfsff'), and 'Phone Number' (containing '7899898779'). A blue 'Submit' button is at the bottom of the form.

Hình 42: Giao diện sửa user

- Xóa user

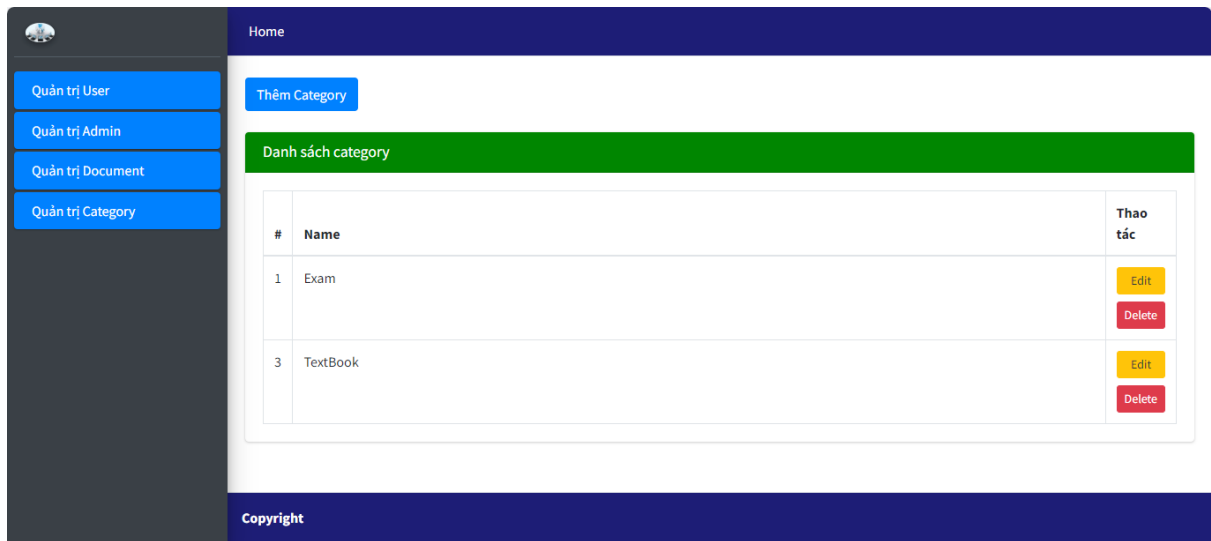
The screenshot shows a table of users with a confirmation dialog box. The dialog box has the text 'User deleted successfully!' and an 'OK' button. The table has the following data:

| # | Name   | Email             | Password  | Address    | Phone Number | tác          |
|---|--------|-------------------|-----------|------------|--------------|--------------|
| 1 | Anh    |                   |           |            | 0234567895   | Edit, Delete |
| 2 | Nga    | nga@gmail.com     | 12345678  | HN         | 0234567896   | Edit, Delete |
| 3 | Lan    | lan@gmail.com     | 12345678  | HN         | 0234567895   | Edit, Delete |
| 5 | abc123 | fefefef@gmail.com | fefefeffe | feeffeffef | 4354355553   | Edit, Delete |
| 8 | gvđgdg | dgdgdg@gmail.com  | MKLm123#  | sfsff      | 7899898779   | Edit, Delete |

Hình 43: Giao diện xóa user

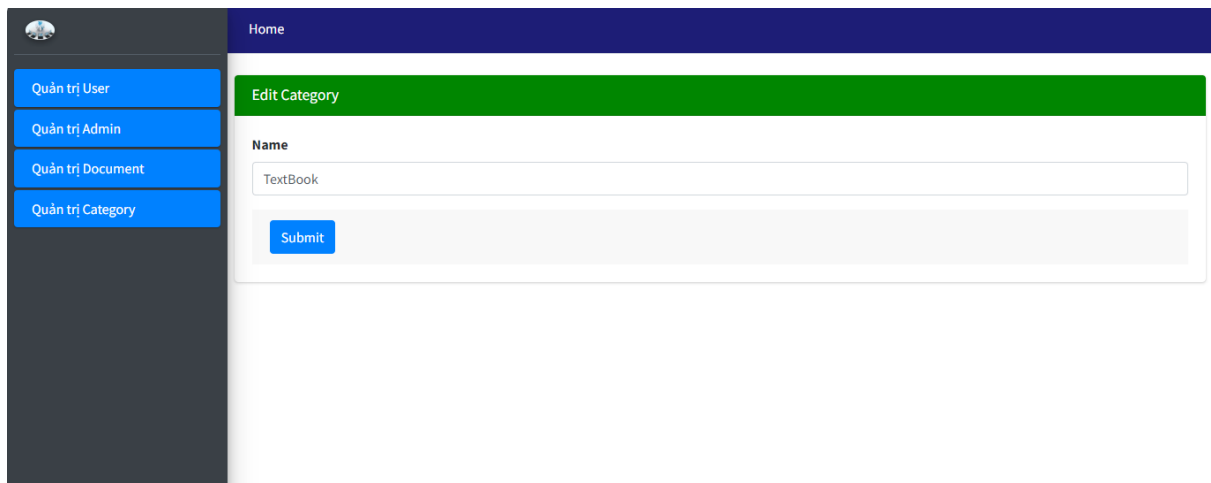
2. Giao diện quản lý Category

- Danh sách Category



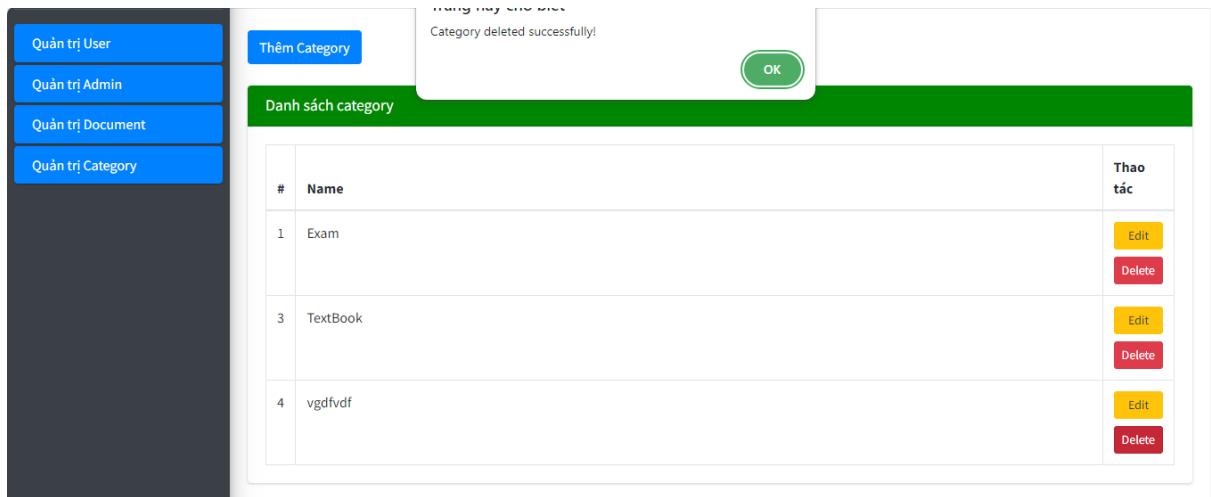
Hình 44: Giao diện danh sách category

- sửa Category



Hình 45: Giao diện sửa category

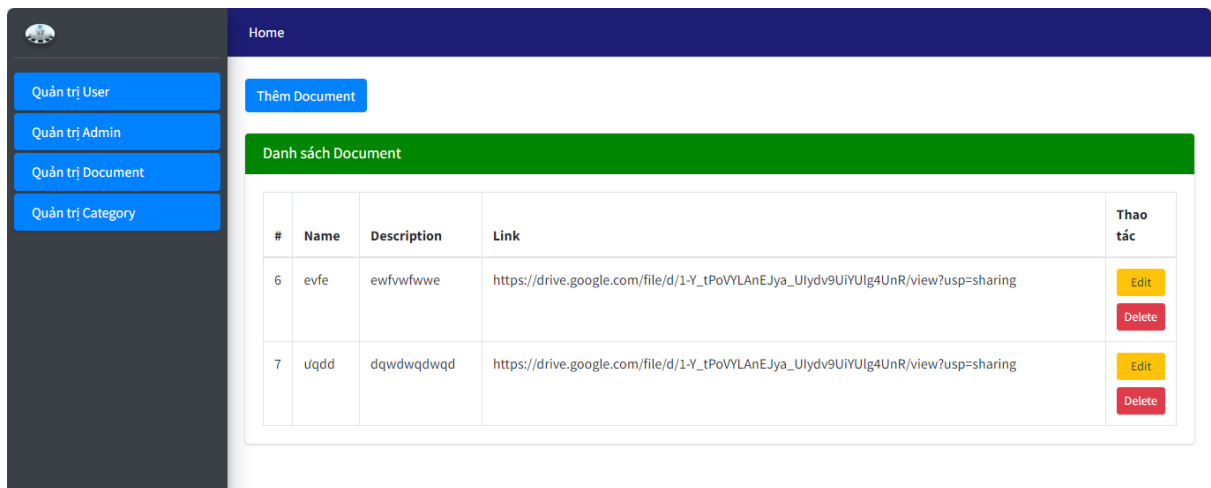
- Xóa Category



Hình 46: Giao diện xóa category

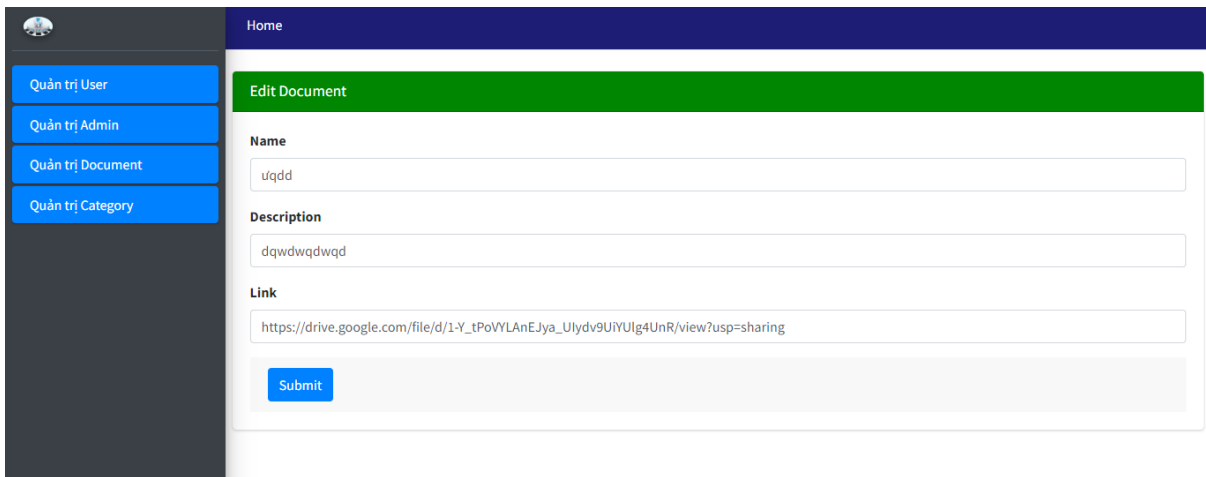
### 3. Giao diện quản lý tài liệu

#### - Danh sách Document



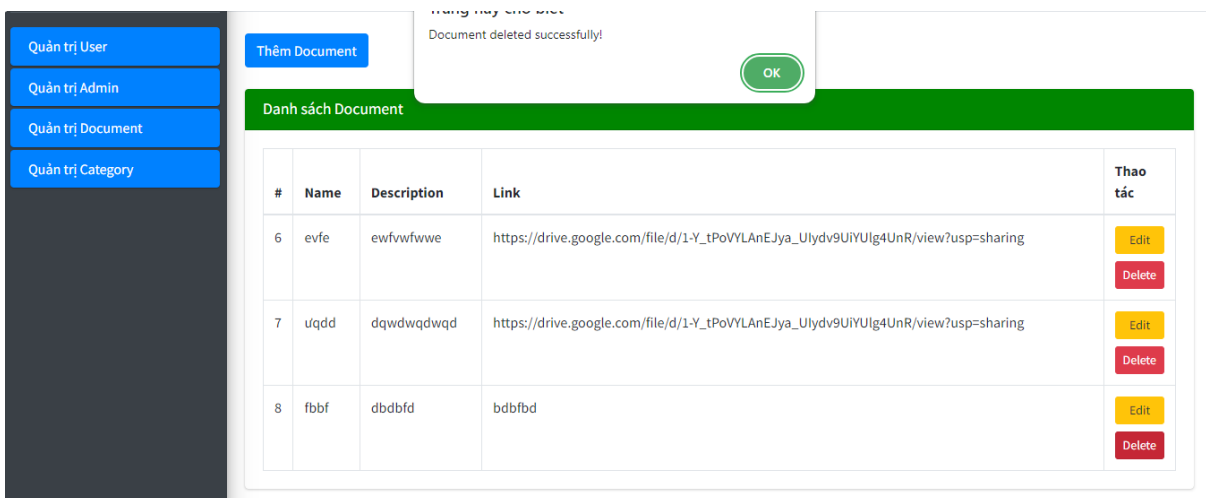
Hình 47: Giao diện danh sách document

#### - Sửa Document



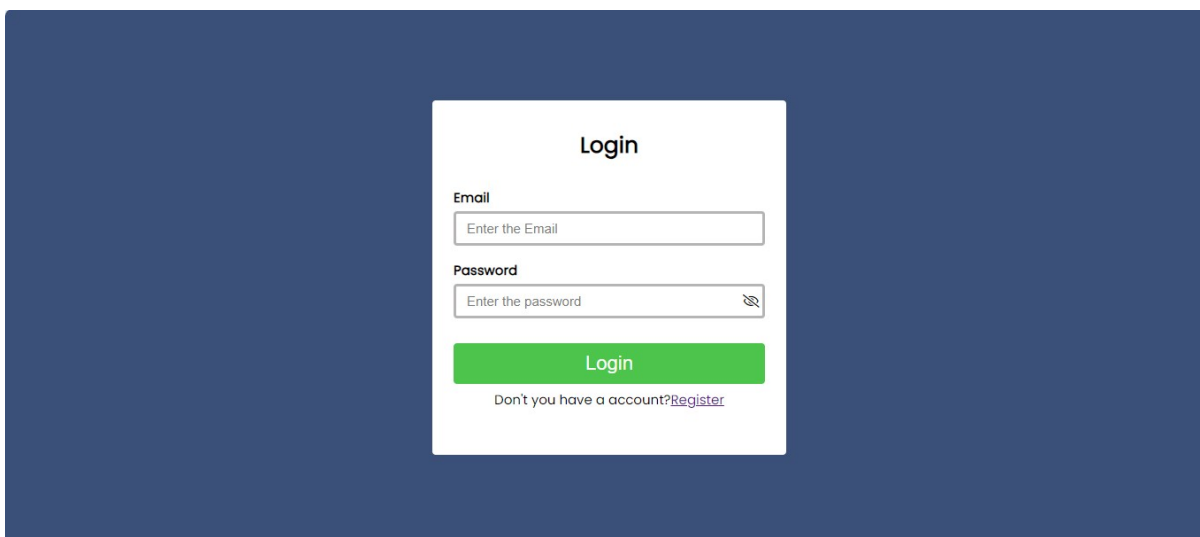
Hình 48: Giao diện sửa document

- Xóa Document



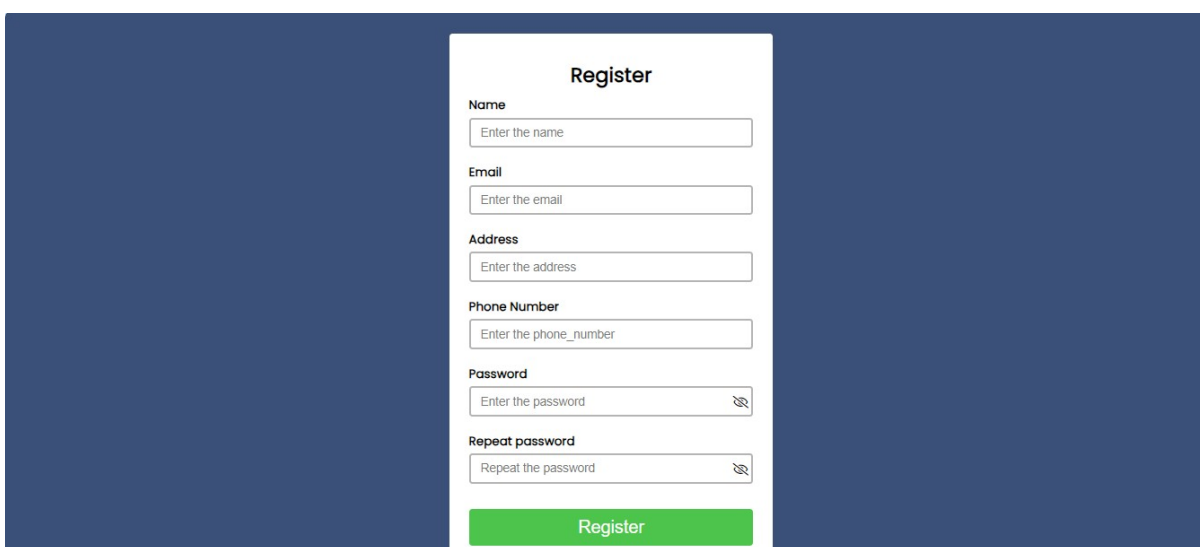
Hình 49: Giao diện xóa document

4. Giao diện trang login



*Hình 50: Giao diện Login*

## 5. Giao diện trang Register



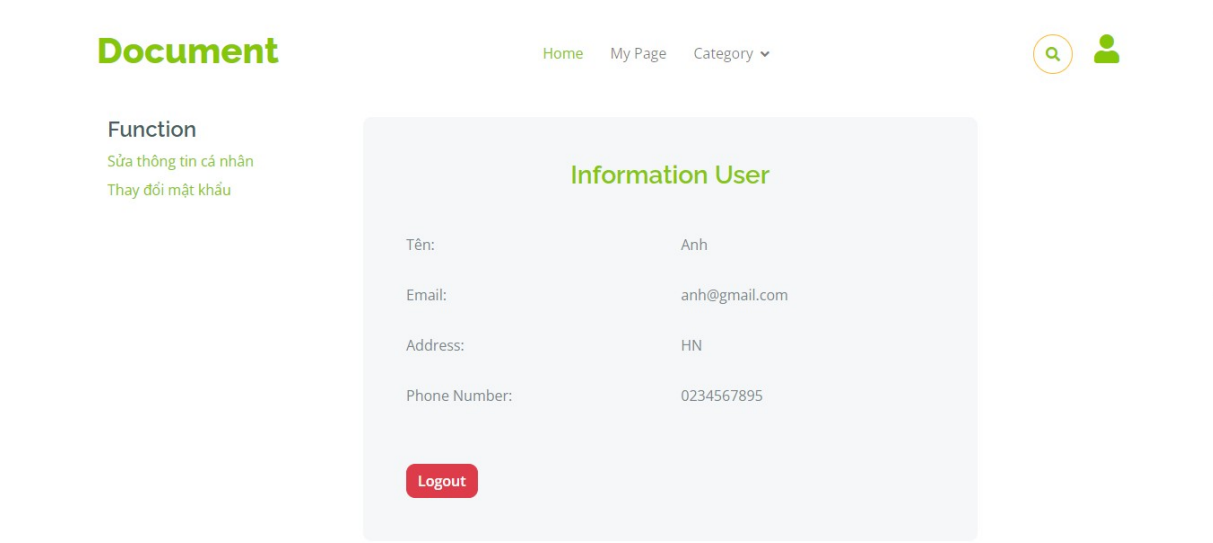
*Hình 51: Giao diện Register*

## 6. Giao diện trang chủ



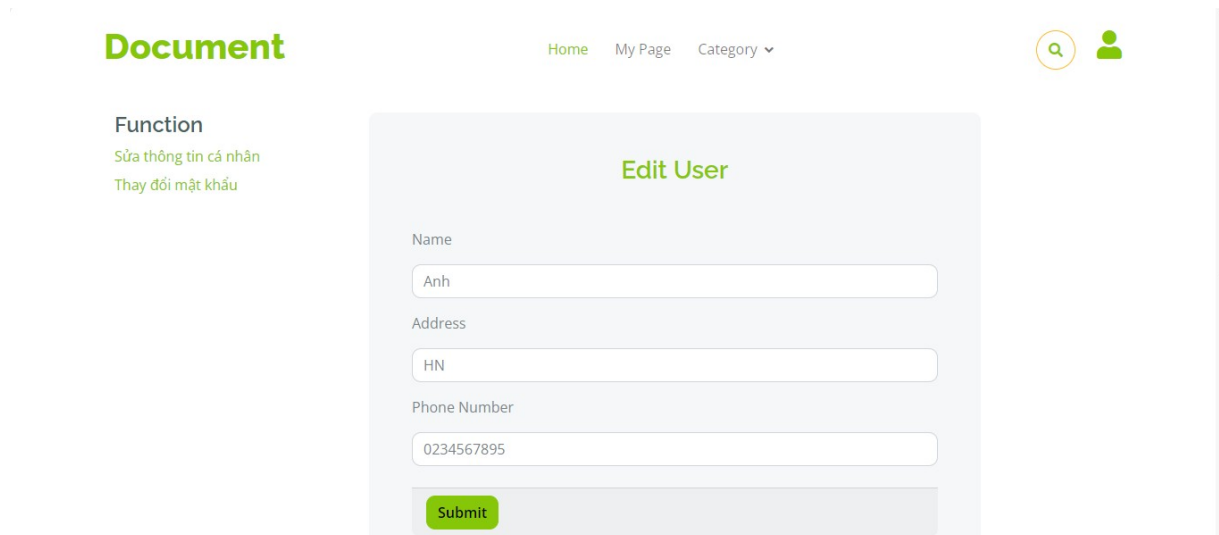
Hình 52: Giao diện trang chủ

## 7. Thông tin cá nhân



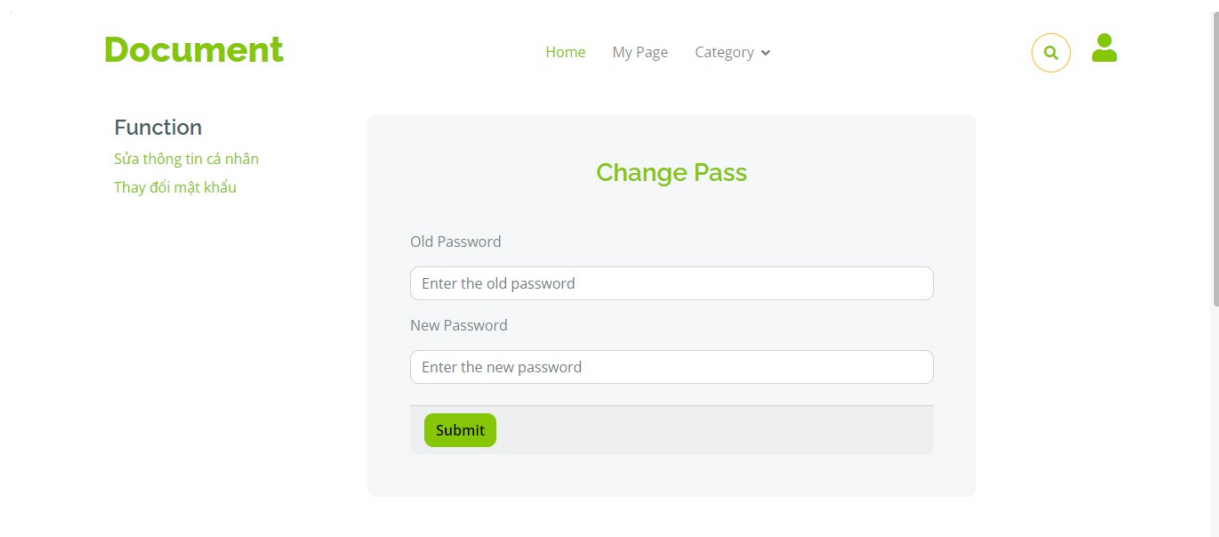
Hình 53: Giao diện thông tin cá nhân

## 8. Sửa thông tin cá nhân



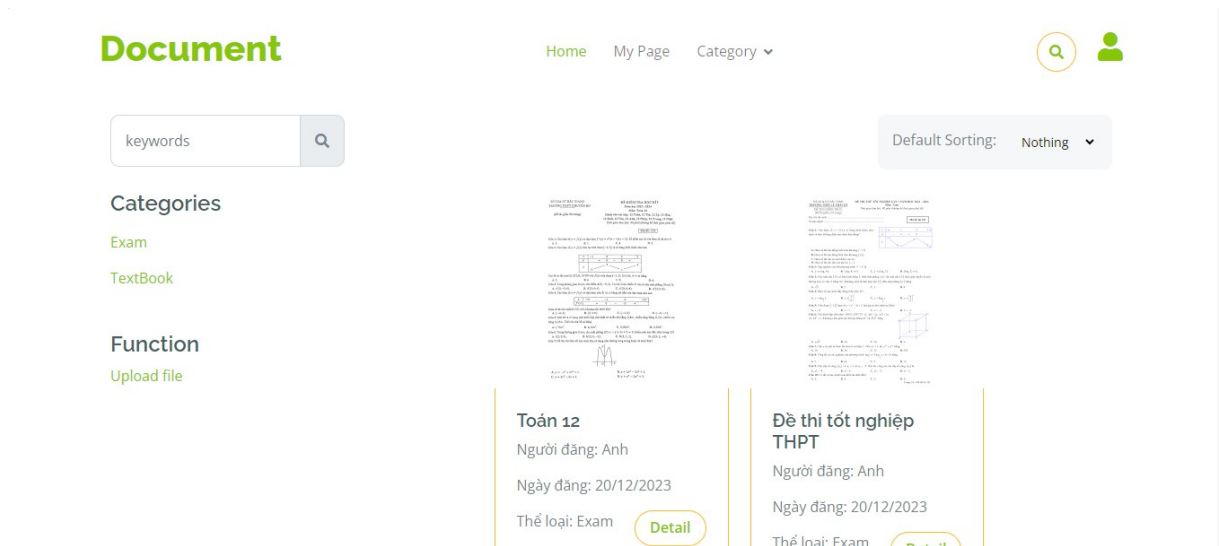
Hình 54: Giao diện sửa thông tin cá nhân

## 9. Đổi password



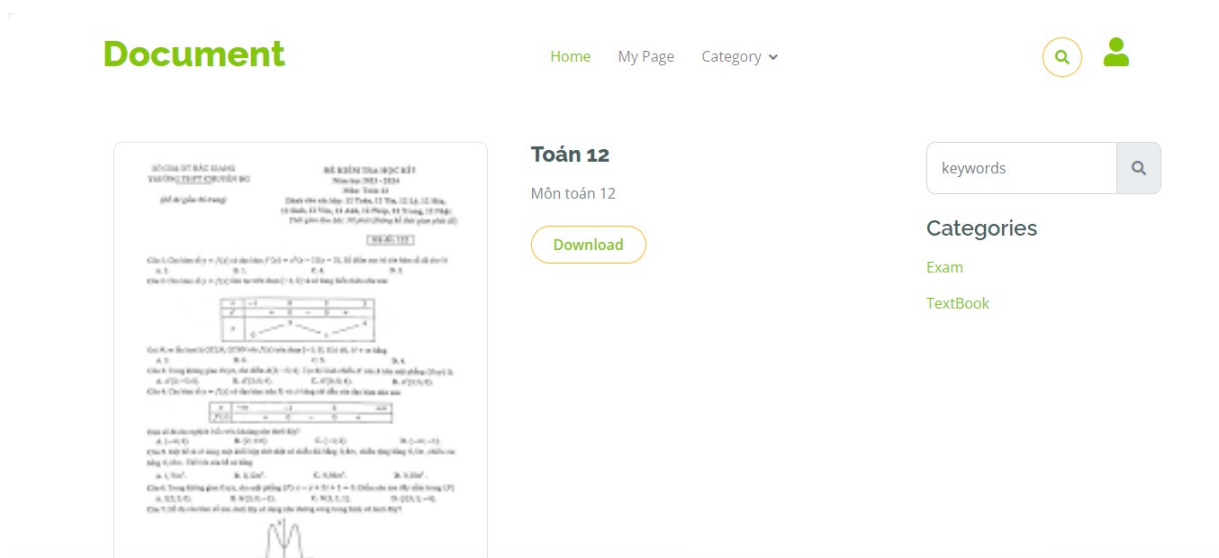
Hình 55: Giao diện đổi password

## 10. Danh sách tài liệu theo danh mục



Hình 56: Giao diện danh sách tài liệu theo danh mục

11. Chi tiết tài liệu



Hình 57: Giao diện chi tiết tài liệu

12. Upload file từ google drive



## Document

Home My Page Category ▾



### Function

From Google Drive

### Choose File

Name\*

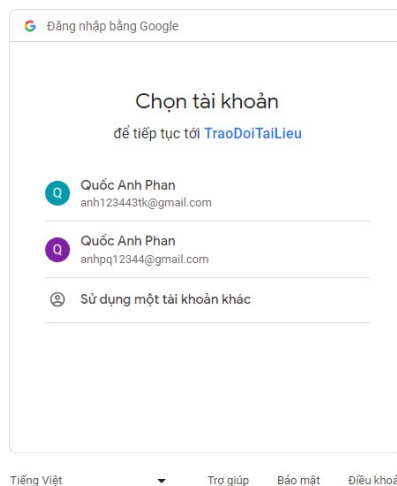
Description\*

Choose file from Google Drive:

Category\* Exam ▾

Hình 58: Giao diện upload file từ google drive

## 13. Chọn tài khoản



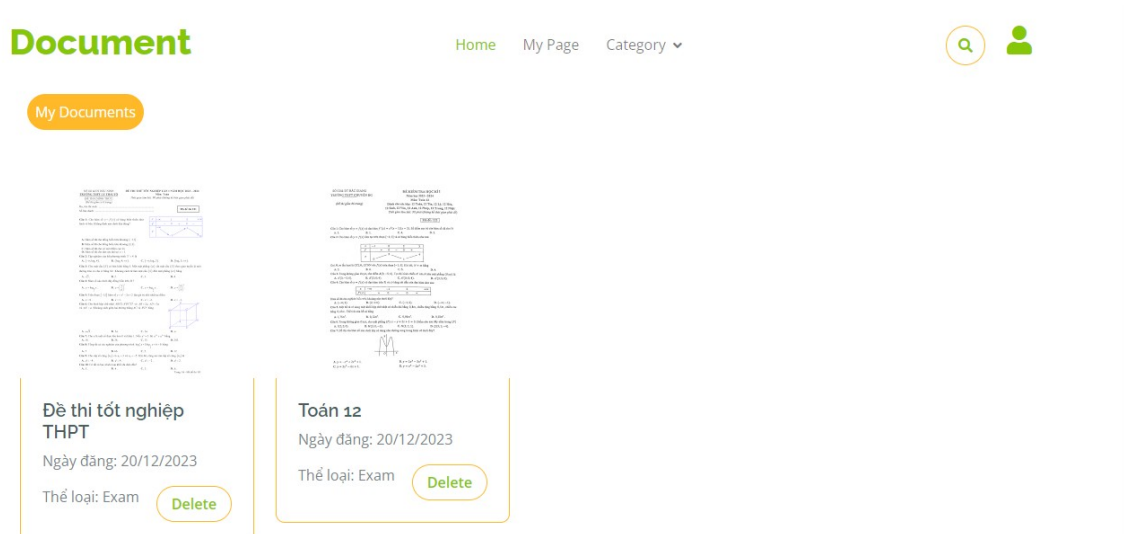
Hình 59: Giao diện chọn tài khoản

## 14. Danh sách các tệp trong google drive



Hình 60: Giao diện danh sách các tệp trong google drive

## 15. Trang xem danh sách tài liệu của tôi



Hình 61: Giao diện trang xem danh sách tài liệu của tôi

### 3.3 Kết luận chương 3

Trong chương này ta đã trình bày được về môi trường cài đặt hệ thống cũng như trình bày được quá trình cài đặt, các giao diện, chức năng, các dịch vụ của hệ thống chia sẻ tài liệu theo kiến trúc hướng dịch vụ.

## **KẾT LUẬN**

### **1. Kết quả đạt được**

Đồ án đã trình bày được các nội dung:

- Giới thiệu về dự án
- Trình bày được kiến trúc Microservices và áp dụng vào dự án
- Làm được các dịch vụ chọn file trên google drive,.....

Trong quá trình làm đề tài, em tích lũy được nhiều kinh nghiệm bổ ích, nắm rõ về cách phát triển một dự án Microservices cơ chế xác thực Oauth, từ đó áp dụng các kiến thức đã học vào dự án

### 2. Ưu và nhược điểm

#### a) Ưu điểm

- Tài liệu đã cung cấp các kiến thức cơ bản tổng quan về Micorservice, kiến trúc, đặc điểm, lợi ích của nó.
- Các chức năng hoạt động ổn định đúng theo mô tả trong tài liệu.
- Về code, cách thức tổ chức code rõ ràng mạch lạc.

#### b) Nhược điểm

- Các chức năng của hệ thống còn ít, mới chỉ có vài chức năng cơ bản do giới hạn về thời gian.
- Phần giao tiếp giữa các service chưa được tốt và chưa tối ưu do chưa có kinh nghiệm làm việc thực tế với loại hình này.

### 3. Hướng phát triển

Do thời gian hạn hẹp cũng như kiến thức của bản thân vẫn còn hạn chế, dự án chỉ phát triển được ở mức cơ bản, còn thiếu nhiều chức năng, giao diện chưa đẹp, em gợi ý một vài hướng phát triển sau:

1. Cập nhật công nghệ mới, tối ưu code
2. Làm giao diện đẹp hơn
3. Trong chức năng chọn file nên chọn file được trên nhiều nền tảng khác nhau
4. Tăng kích thước file chọn
5. Làm thêm chức năng về hiệu năng, số lượng truy cập

## TÀI LIỆU THAM KHẢO

Các trang web tham khảo

[1]. <https://topdev.vn/blog/phat-trien-phan-mem-theo-kien-truc-microservice/>

- [2]. <https://fptcloud.com/git-la-gi/>
- [3]. <https://stanford.com.vn/kien-thuc-lap-trinh/tin-chi-tiet/cagId/27/id/23576/spring-boot-la-gi-gioi-thieu-spring-boot-framework>
- [4]. <https://vn.got-it.ai/blog/mysql-la-gi-tim-hieu-he-quan-tri-co-so-du-lieu-mysql>
- [5]. <https://itnavi.com.vn/blog/microservices-la-gi>
- [6]. <https://www.atatus.com/blog/microservices-architecture/>