

ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



SOICT

BÁO CÁO BÀI TẬP HỌC PHẦN
IT3920Q – PROJECT 2

Đề tài: Hệ thống quản lý văn bản hành chính

Giáo viên hướng dẫn: Ths. Nguyễn Duy Hiệp

Sinh Viên Thực Hiện:

Nguyễn Văn Linh : 20187180

Hà Nội, 7/2022

GIỚI THIỆU ĐỀ TÀI.....	3
1. Đặt vấn đề	3
2. Mục tiêu và phạm vi đề tài.....	3
3. Định hướng giải pháp.....	4
KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU.....	5
1. Khảo sát hiện trạng	5
2. Tổng quan chức năng.....	5
3. Đặc tả chức năng.....	9
4. Yêu cầu phi chức năng.....	31
Chương 4 Phát triển và triển khai ứng dụng	40
4.1 Thiết kế kiến trúc.....	40
4.2 Thiết kế chi tiết.....	45
4.3 Kết quả đạt được.....	59
4.4. Kiểm thử.....	70
Chương 5 Kết luận và hướng phát triển.....	74
4. 5. 1 Kết luận.....	74
5. 5.2 Hướng phát triển.....	75
6. Tài liệu tham khảo	75

GIỚI THIỆU ĐỀ TÀI

1. Đặt vấn đề

Để đạt đến mức độ phát triển như ngày nay, lịch sử Việt Nam đã trải qua rất nhiều cuộc cách mạng: cách mạng nông nghiệp, cách mạng công nghiệp. Và cuộc cách mạng đang diễn ra ngay lúc này, tác động từng giờ từng phút tới đời sống của nhân loại chính là cách mạng Công nghệ 4.0.

Thời đại Công nghệ 4.0 tập trung vào sự phát triển của công nghệ. Tức là tất cả những gì liên quan đến hệ thống vật lý không gian mạng Internet. Như chúng ta đều có thể cảm nhận được, công nghệ đang và sẽ tạo ảnh hưởng to lớn lên tất cả các ngành và lĩnh vực đời sống. Kỷ nguyên khác biệt này tạo ra tốc độ phát triển sản xuất, xã hội siêu nhanh chóng, phá bỏ các truyền thống trước đây.

Sự xuất hiện và ảnh hưởng quá lớn của công nghệ sẽ thay đổi bộ mặt toàn bộ các ngành, lĩnh vực. Để có thể bắt kịp với thời đại, công nghệ tin học đã được ứng dụng trong nhiều lĩnh vực nhằm phục vụ lợi ích cho con người. Tất cả các nước đều đang cố gắng làm chủ kiến thức và tìm cách áp dụng thành tựu của Công nghệ thông tin vào mọi ngành kinh tế - xã hội của nhà nước. Xã hội ngày càng phát triển, không chỉ các công ty, doanh nghiệp cần đến những phần mềm hỗ trợ quản lý mà các cơ quan nhà nước cũng đang sử dụng rất nhiều trong công việc.

Điển hình nhất là công tác văn thư lưu trữ có vai trò rất quan trọng đối với hoạt động của bất kỳ cơ quan, tổ chức nào vì nó đảm bảo thông tin dưới dạng văn bản được cập nhật, lưu trữ, phát hành và tìm kiếm một cách nhanh chóng, kịp thời và chính xác. Nhưng, hiện nay, hầu hết các cơ quan tổ chức vẫn lưu trữ văn thư dưới dạng văn bản giấy từ năm này qua năm khác. Do đó, theo thời gian, số lượng văn bản càng tăng lên dẫn đến việc lưu trữ và tìm kiếm vô cùng khó khăn. Với các lý do ấy, chúng tôi đã xây dựng **Phần mềm Quản lý Văn bản** để hỗ trợ cho việc lưu trữ và tra cứu văn thư một cách hiện đại, bảo mật, nhanh chóng và chính xác nhất.

2. Mục tiêu và phạm vi đề tài

Mục tiêu của đề tài là tạo ra phần mềm giúp các tổ chức, doanh nghiệp giảm tải công việc hành chính.

Phần mềm Quản lý Văn bản được thiết kế gắn liền với quy trình nghiệp vụ xử lý công việc của các cơ quan quản lý hành chính Nhà nước với các quy trình nghiệp vụ chính như sau:

Quản lý văn bản đến: Thực hiện tiếp nhận và xử lý văn bản đến;

Quản lý văn bản đi: Thực hiện các công việc liên quan trong quá trình dự thảo, ban hành và theo dõi xử lý văn bản đi của cơ quan;

Quản lý hồ sơ lưu trữ: Chức năng này cho phép người sử dụng có thể thiết lập các hồ sơ lưu trữ để quản lý văn bản và theo dõi tiến độ xử lý công việc; trong chức năng này người sử dụng có thể tạo lập và chỉnh sửa hồ sơ; đưa các văn bản vào trong hồ sơ, in ấn các thông tin liên quan trong hồ sơ.

Theo dõi xử lý văn bản: Tại đây người sử dụng có thể theo dõi tiến độ xử lý các văn bản đến, văn bản đi và các thông tin chỉ đạo điều hành khác của cơ quan;

Quản lý công việc: Theo vai trò của từng người sử dụng có thể thực hiện tạo lập công việc, phân công cho phòng ban thực hiện công việc, cập nhật tình hình kết quả thực hiện công việc và báo cáo tình hình xử lý công việc với lãnh đạo các cấp.

Gửi nhận văn bản: Chức năng này cho phép người sử dụng có thể truyền nhận văn bản

3. Định hướng giải pháp

Hệ thống được xây dựng cần đảm bảo đầy đủ chức năng để giúp khách hàng thao tác thuận tiện nhất.

KHẢO SÁT VÀ PHÂN TÍCH YÊU CẦU

1. Khảo sát hiện trạng

Quản lý văn bản, tài liệu là một trong những nhiệm vụ cơ bản của cơ quan nhà nước nhằm đảm bảo hiệu quả cải cách hành chính, chứng minh tính minh bạch và trách nhiệm giải trình các hoạt động bằng thông tin văn bản. Dưới sự tác động của khoa học công nghệ, bên cạnh tài liệu truyền thống, tài liệu điện tử ra đời với những tính năng và đặc điểm khác biệt. Do đó, việc nghiên cứu và làm rõ quan niệm và những đặc điểm của tài liệu điện tử cùng một số yêu cầu của quản lý tài liệu điện tử là việc làm cần thiết giúp bổ sung lý luận và nâng cao hiệu quả quản lý về văn thư và lưu trữ trong bối cảnh xây dựng và vận hành Chính phủ điện tử, Chính phủ số hiện nay.

Trên thị trường về ứng dụng quản lý văn bản trong cơ quan nhà nước được sử dụng rất phổ biến. Nhưng không phải một phần mềm đều phù hợp với tất cả các cơ quan, vậy nên nhóm đã triển khai phần mềm phù hợp với chức năng và kinh phí của cơ quan đó.

2. Tổng quan chức năng

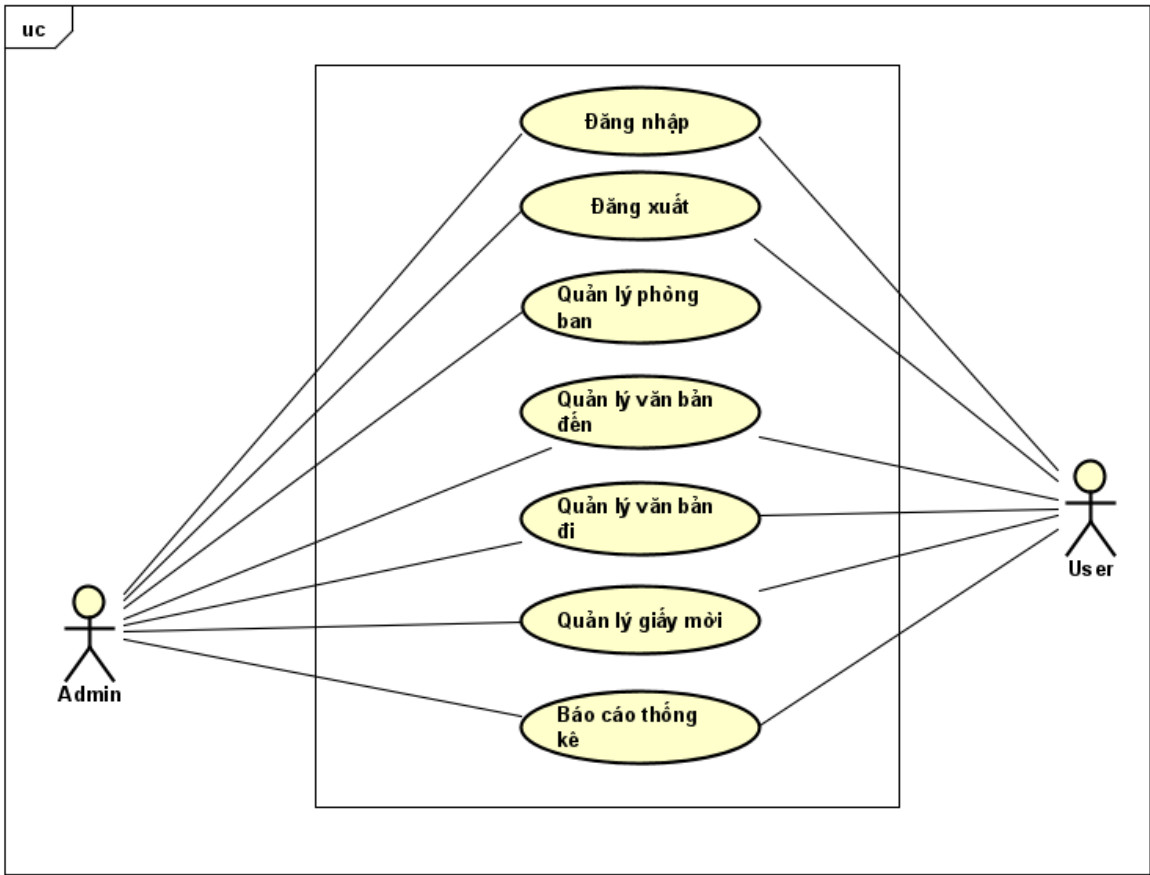
2.1. Biểu đồ usecase tổng quan

Hệ thống có 4 tác nhân chính: Admin, Lãnh đạo, Nhân viên và Văn thư.

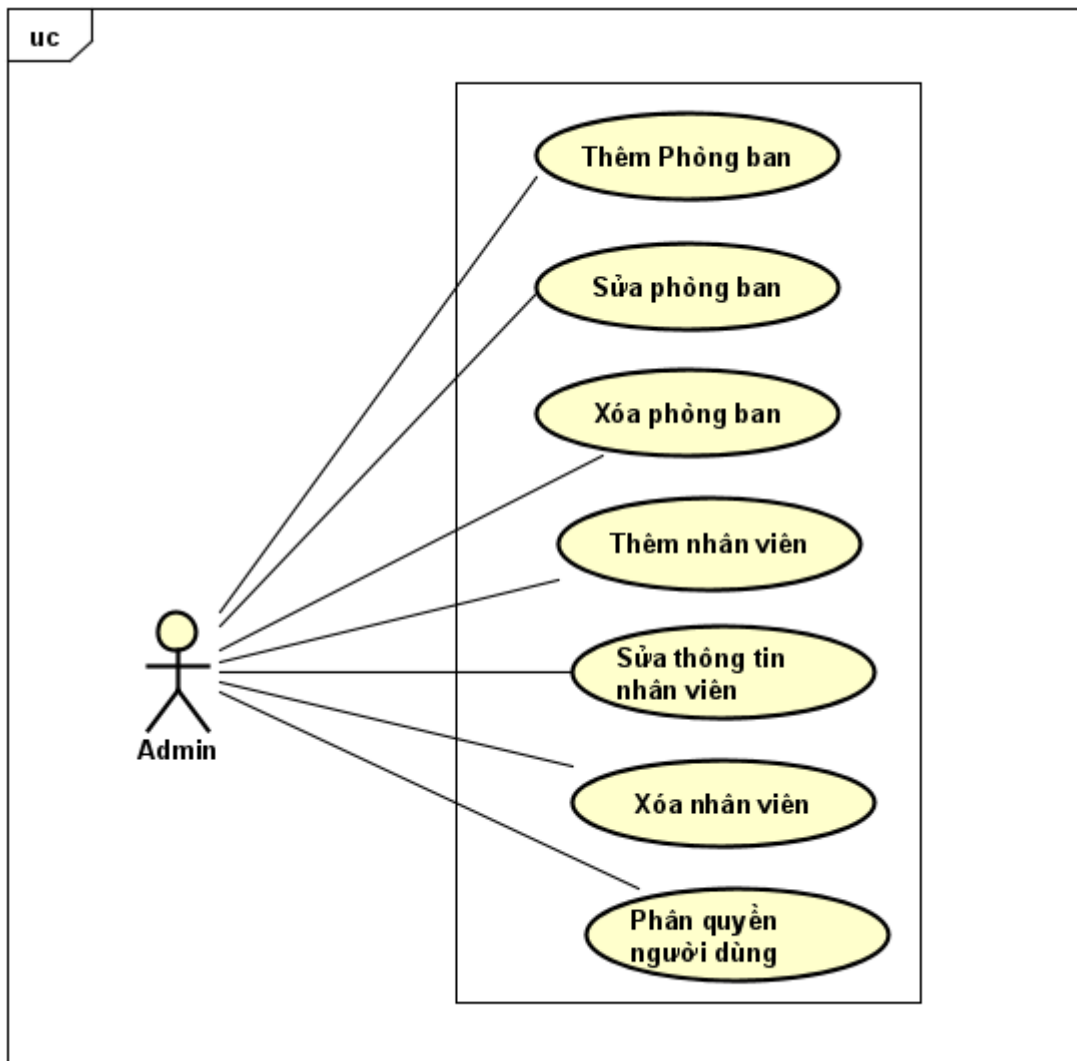
Mô tả sơ lược các tác nhân như sau

- Admin: Quản lý phòng ban, thêm sửa xóa nhân viên
- Lãnh đạo: Phân xử lý phòng ban, Phê duyệt hoặc từ chối văn bản
- Nhân viên: Xử lý văn bản
- Văn thư: Tiếp nhận văn bản

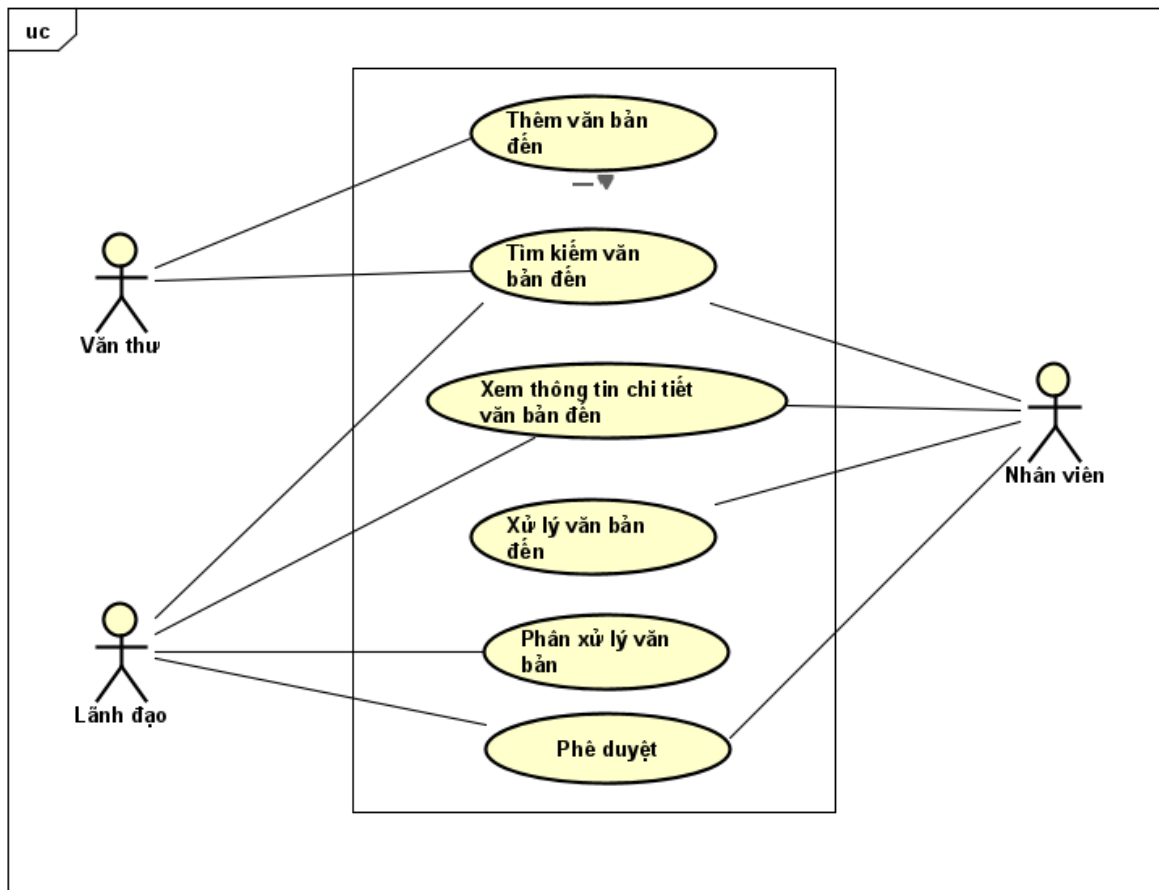
Mô tả sơ lược các tác nhân như sau:



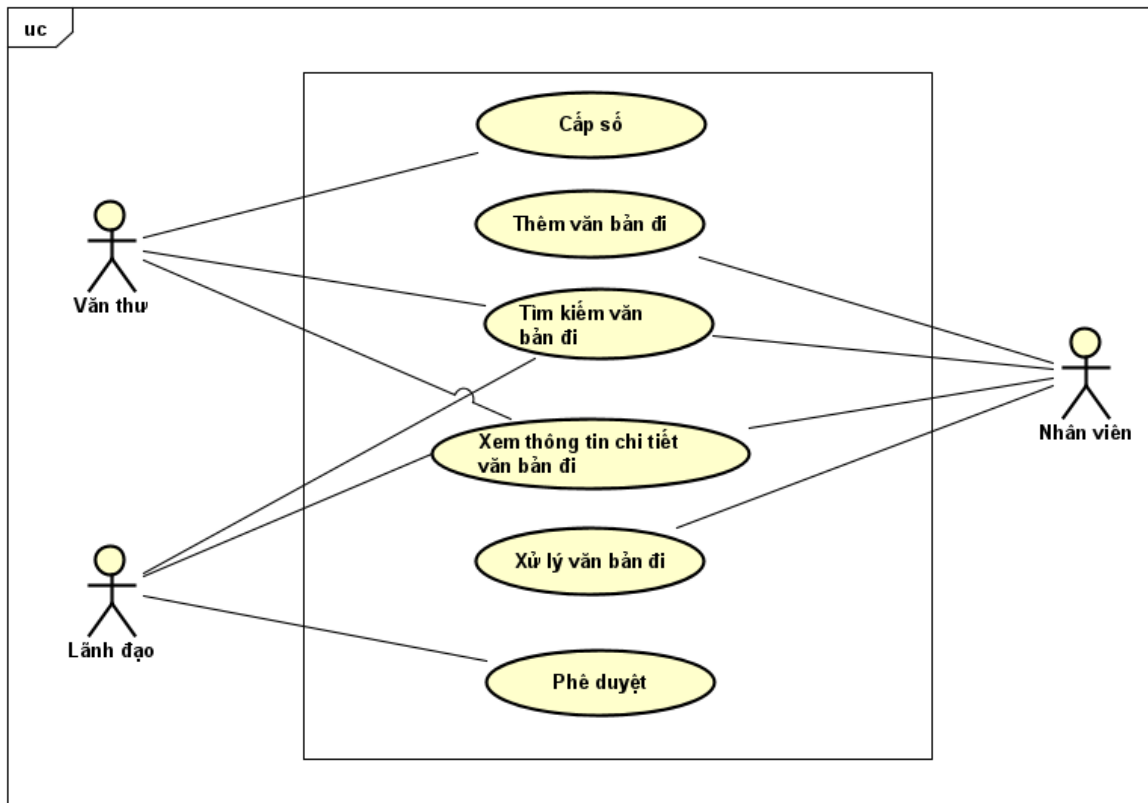
2.2. Phân rã use case Quản lý phòng ban



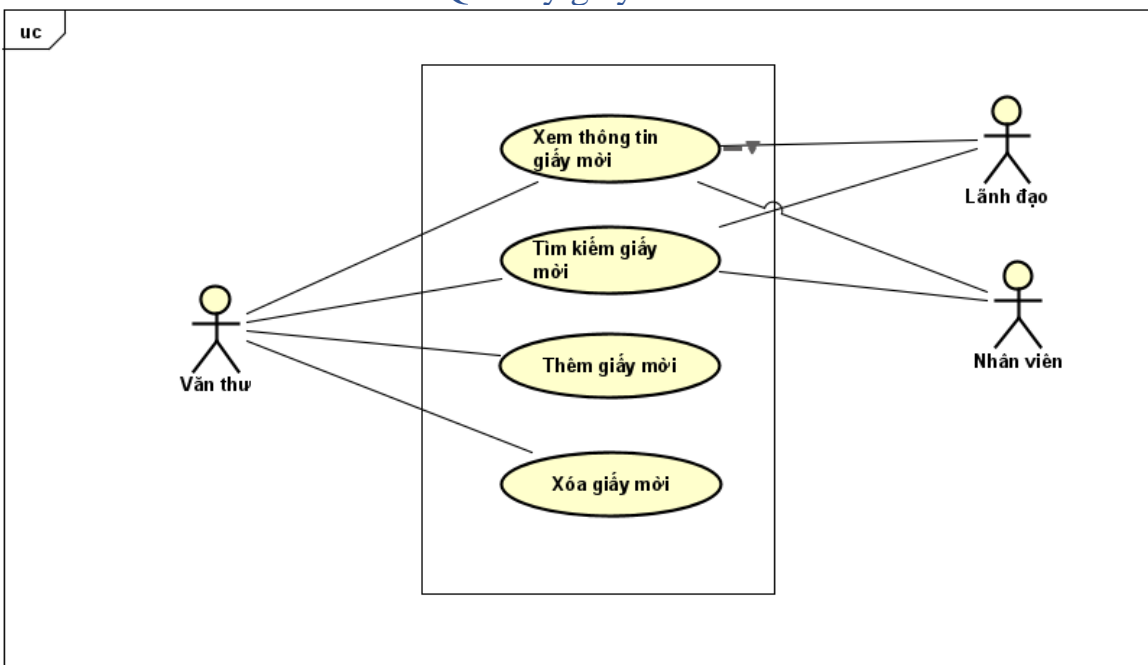
2.3. Phân rã use case Quản lý văn bản đến



2.4. Phân rã use case Quản lý văn bản đi



2.5. Phân rã use case Quản lý giấy mời



3. Đặc tả chức năng

3.1. Đặc tả use case “đăng nhập”

Tên UC	Đăng nhập
Mô tả	Usecase này được thực hiện người dùng để đăng nhập vào hệ thống
Actor	Admin, User
Precondition	
Main Flow	<ol style="list-style-type: none"> 1. Actor yêu cầu đăng nhập 2. Hệ thống hiển thị form đăng nhập 3. Actor nhập tài khoản 4. Actor xác nhận đăng nhập 5. Hệ thống hiển thị màn hình chính
Alternative Flow	<p>3.1. Admin nhập sai tài khoản</p> <p>5.1. Hệ thống hiển thị thông báo lỗi</p>
Post condition	Actor đăng nhập vào được hệ thống. Hệ thống sẽ hiển thị giao diện chính và actor có thể thực hiện chức năng theo đúng quyền hạn của mình.

Dữ liệu đầu vào của tài khoản:

STT	Trường dữ liệu	Mô tả	Định dạng hiển thị	Ví dụ
1.	Tên đăng nhập	Tên đăng nhập	dạng: abc@gmail.com	
2.	Mật khẩu	Mật khẩu	- Từ 6 kí tự trở lên	

3.2. Đặc tả use case “đăng xuất”

Tên UC	Đăng xuất
Mô tả	Usecase này được thực hiện để đăng xuất khỏi hệ thống
Actor	Admin, User
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> 1. Actor yêu cầu đăng xuất tài khoản 2. Hệ thống xác nhận đăng xuất
Alternative Flow	
Post condition	Actor đăng xuất khỏi hệ thống. Actor sẽ không được thực hiện các chức năng của hệ thống.

3.3. Đặc tả “thêm phòng ban”

Tên UC	Thêm phòng ban
Mô tả	Usecase này được thực hiện để thêm phòng ban mới
Actor	Admin
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> 1. Actor chọn chức năng Thêm phòng ban 2. Hệ thống hiển thị form thêm phòng ban 3. Actor nhập thông tin phòng ban 4. Actor xác nhận form phòng ban 5. Hệ thống kiểm tra tính hợp lệ của thông tin và lưu vào database
Alternative Flow	3.1. Actor nhập sai thông tin, nhập không đúng quy định

Post condition	Phòng ban mới được thêm vào hệ thống
-----------------------	--------------------------------------

3.4. Đặc tả use case “sửa phòng ban”

Tên UC	Sửa phòng ban
Mô tả	Usecase này được thực hiện để sửa thông tin phòng ban
Actor	Admin
Precondition	Admin phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> Admin chọn phòng ban cần chỉnh sửa Admin chọn Sửa phòng ban Admin sửa thông tin phòng ban Admin xác nhận chỉnh sửa Hệ thống kiểm tra tính hợp lệ của thông tin và lưu vào database
Alternative Flow	
Post condition	Thông tin phòng ban cập nhật vào hệ thống thành công

3.5. Đặc tả use case “xóa phòng ban”

Tên UC	Xóa phòng ban
Mô tả	Usecase này được thực hiện để xóa phòng ban
Actor	Admin
Precondition	Actor cần phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> Hệ thống hiển thị danh sách phòng ban

	<ol style="list-style-type: none"> 2. Actor chọn xóa phòng ban 3. Actor xác nhận xóa 4. Hệ thống xóa phòng ban khỏi hệ thống
Alternative Flow	
Post condition	Phòng ban được xóa khỏi hệ thống

3.6. Đặc tả usecase “thêm nhân viên”

Tên UC	Thêm nhân viên
Mô tả	Usecase này được thực hiện để thêm nhân viên mới
Actor	Admin
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> 1. Actor chọn phòng ban muốn thêm nhân viên mới 2. Actor chọn chức năng Thêm nhân viên 3. Hệ thống hiển thị form thêm nhân viên 4. Actor nhập thông tin nhân viên 5. Actor xác nhận form nhân viên 6. Hệ thống kiểm tra tính hợp lệ của thông tin và lưu vào database
Alternative Flow	4.1. Actor nhập sai thông tin, nhập không đúng quy định
Post condition	Nhân viên mới được thêm vào phòng ban đó

3.7. Đặc tả usecase “sửa thông tin nhân viên”

Tên UC	Sửa thông tin nhân viên
---------------	-------------------------

Mô tả	Usecase này được thực hiện để sửa thông tin nhân viên
Actor	Admin
Precondition	Admin phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> 1. Admin chọn phòng ban chứa nhân viên cần chỉnh sửa 2. Hệ thống hiển thị danh sách nhân viên 3. Admin chọn sửa thông tin nhân viên 4. Admin sửa thông tin nhân viên 5. Admin xác nhận chỉnh sửa 6. Hệ thống kiểm tra tính hợp lệ của thông tin và lưu vào database
Alternative Flow	
Post condition	Thông tin nhân viên cập nhật vào hệ thống thành công

3.8. Đặc tả usecase “xóa nhân viên”

Tên UC	Xóa nhân viên
Mô tả	Usecase này được thực hiện để xóa nhân viên
Actor	Admin
Precondition	Actor cần phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> 1. Hệ thống hiển thị danh sách nhân viên 2. Actor chọn xóa nhân viên 3. Actor xác nhận xóa 4. Hệ thống xóa nhân viên khỏi hệ thống

Alternative Flow	
Post condition	Nhân viên được xóa khỏi hệ thống

3.9. Đặc tả usecase “phân quyền người dùng”

Tên UC	Phân quyền người dùng
Mô tả	Usecase này được thực hiện để phân quyền người dùng
Actor	Admin
Precondition	Admin phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> Admin chọn Phân quyền người dùng Admin Phân quyền người dùng Admin xác nhận phân quyền Hệ thống hiển thị quyền của người dùng
Alternative Flow	
Post condition	Thông tin về quyền người dùng cập nhật vào hệ thống

3.10. Đặc tả usecase “thêm văn bản đến”

Tên UC	Thêm văn bản đến
Mô tả	Usecase này được thực hiện để thêm thông tin văn bản mới
Actor	Văn thư
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> Actor chọn chức năng Thêm văn bản đến

	8. Hệ thống hiển thị form thêm văn bản đến 9. Actor nhập thông tin văn bản đến 10. Actor xác nhận form văn bản đến 11. Hệ thống kiểm tra tính hợp lệ của thông tin và lưu vào database
Alternative Flow	3.1. Actor nhập sai thông tin, nhập không đúng quy định
Post condition	Thông tin văn bản đến được thêm vào hệ thống

Dữ liệu đầu vào của form thêm văn bản đến

STT	Trường dữ liệu	Mô tả	Định dạng hiển thị	Ví dụ
1.	Tên văn bản	Tên văn bản đến	Text	Quyết định
2.	Mã định danh cơ quan tổ chức phát hành	Mã định danh cơ quan tổ chức phát hành văn bản đến	Text	UBND
3.	Số văn bản	Số của văn bản đến	Số nguyên dương	123
4.	Kí hiệu văn bản	Mã loại văn bản đến	Text	QĐ
5.	Loại văn bản	Loại văn bản đến		Quyết định
6.	Ngày phát hành	Ngày phát hành văn bản đến	Hiển thị dưới dạng: DD/MM/YYYY DD: ngày, MM: tháng, YYYY: năm	24/01/2022
7.	Số trang	Số trang văn bản đến	Số nguyên dương	30

8.	Tài liệu	Các file đính kèm đi cùng văn bản đến	Là các file có dạng: doc, jpg, ...	QĐTC.doc
9.	Trích yếu nội dung	Tóm tắt nội dung văn bản đến	Text Editor	Quyết định thăng chức
10.	Độ mật	Độ mật		Mật
11.	Độ khẩn	Độ khẩn		Khẩn

3.11. Đặc tả usecase “tìm kiếm văn bản đến”

Tên UC	Tìm kiếm văn bản đến
Mô tả	Usecase này được thực hiện để tìm kiếm văn bản đến
Actor	Admin, Văn thư, lãnh đạo, nhân viên
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> Hệ thống hiển thị danh sách văn bản đến Actor nhập các thông tin của văn bản đến Actor xác nhận tìm kiếm văn bản đến Hệ thống hiển thị văn bản tìm kiếm
Alternative Flow	<ol style="list-style-type: none"> 1.1. Actor nhập không đúng trường dữ liệu 4.1. Hệ thống không tìm được văn bản
Post condition	Hệ thống hiển thị văn bản mà actor muốn tìm kiếm.

Dữ liệu đầu vào của tìm kiếm văn bản đến:

STT	Trường dữ liệu	Mô tả	Định dạng hiển thị	Ví dụ
1.	Tên văn bản	Tên văn bản đến	Text	Quyết định
2.	Loại văn bản	Loại văn bản đến	Text	Quyết định
3.	Ngày tạo	Số của văn bản	Hiển thị dưới	2022/01/24

		đến	dạng: YYYY/MM/DD DD: ngày, MM: tháng, YYYY: năm	
4.	Trạng thái	Trạng thái của văn bản đến	Text	Tiếp nhận

3.12. Đặc tả usecase “xem thông tin chi tiết văn bản đến”

Tên UC	Xem thông tin chi tiết văn bản đến
Mô tả	Usecase này được thực hiện để xem thông tin chi tiết văn bản
Actor	Admin, Văn thư, lãnh đạo, nhân viên
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> Hệ thống hiển thị danh sách văn bản đến Actor chọn văn bản muốn xem chi tiết Actor chọn chức năng xem chi tiết văn bản đến Hệ thống hiển thị form xem chi tiết văn bản
Alternative Flow	
Post condition	Hệ thống hiển thị thông tin chi tiết của văn bản đến.

Dữ liệu đầu ra của form chi tiết văn bản đến:

STT	Trường dữ liệu	Mô tả	Định dạng hiển thị	Ví dụ
1.	Tên văn bản	Tên văn bản đến	Text	Quyết định
2.	Mã định danh cơ quan tổ chức phát hành	Mã định danh cơ quan tổ chức phát hành văn bản đến	Text	UBND

3.	Số văn bản	Số của văn bản đến	Số nguyên dương	123
4.	Kí hiệu văn bản	Mã loại văn bản đến	Text	QĐ
5.	Loại văn bản	Loại văn bản đến		Quyết định
6.	Ngày phát hành	Ngày phát hành văn bản đến	Hiện thị dưới dạng: DD/MM/YYYY DD: ngày, MM: tháng, YYYY: năm	24/01/2022
7.	Thời hạn xử lý	Thời gian xử lý văn bản	Hiện thị dưới dạng: YYYY/MM/DD DD: ngày, MM: tháng, YYYY: năm	2022/01/24
8.	Hồ sơ gửi kèm	Các file đính kèm đi cùng văn bản đến	Là các file có dạng: doc, jpg, ...	QĐTC.doc
9.	Trích yếu nội dung	Tóm tắt nội dung văn bản đến	Text Editor	Quyết định thăng chức
10.	Độ mật	Độ mật		Mật
11.	Độ khẩn	Độ khẩn		Khẩn

3.13. Đặc tả usecase “xử lý văn bản đến”

Tên UC	Xử lý văn bản đến
Mô tả	Usecase này được thực hiện để xử lý văn bản đến
Actor	Nhân viên
Precondition	Actor cần phải hoàn thành usecase “xem thông tin chi tiết văn

	bản đến”
Main Flow	<ol style="list-style-type: none"> 1. Hệ thống hiển thị form xem thông tin chi tiết văn bản 2. Actor chỉnh sửa trên form 3. Actor xác nhận form 4. Hệ thống kiểm tra tính hợp lệ và lưu vào database
Alternative Flow	
Post condition	Actor xử lý văn bản đến

3.14. Đặc tả usecase “phân xử lý văn bản”

Tên UC	Phân xử lý văn bản
Mô tả	Usecase này được thực hiện để phân văn bản đến phòng ban xử lý
Actor	Lãnh đạo
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> 1. Actor chọn văn bản muốn phân xử lý 2. Actor chọn chức năng Phân xử lý 3. Hệ thống hiển thị form phân xử lý 4. Actor phân xử lý và xác nhận phân xử lý 5. Hệ thống xác nhận và lưu thông tin
Alternative Flow	
Post condition	Phân xử lý văn bản thành công.

3.15. Đặc tả usecase “phê duyệt”

Tên UC	Phê duyệt
Mô tả	Usecase này được thực hiện để phê duyệt văn bản
Actor	Lãnh đạo, Nhân viên
Precondition	Actor cần phải hoàn thành usecase “Thêm văn bản đến” hoặc “Xử lý văn bản đến”
Main Flow	<ol style="list-style-type: none">1. Hệ thống hiển thị form xem thông tin chi tiết văn bản2. Actor chọn Phê duyệt hoặc từ chối3. Hệ thống xác nhận và lưu thông tin
Alternative Flow	
Post condition	Hành động phê duyệt được thực hiện

3.16. Đặc tả usecase “thêm văn bản đi”

Tên UC	Thêm văn bản đi
Mô tả	Usecase này được thực hiện để thêm thông tin văn bản mới
Actor	Nhân viên
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none">1. Actor chọn chức năng Thêm văn bản đi2. Hệ thống hiển thị form thêm văn bản đi3. Actor nhập thông tin văn bản đi4. Actor xác nhận form văn bản đi5. Hệ thống kiểm tra tính hợp lệ của thông tin và lưu vào database

Alternative Flow	3.1. Actor nhập sai thông tin, nhập không đúng quy định
Post condition	Thông tin văn bản đi được thêm vào hệ thống

Dữ liệu đầu vào của form thêm văn bản đi

STT	Trường dữ liệu	Mô tả	Định dạng hiển thị	Ví dụ
1.	Tên văn bản	Tên văn bản đi	Text	Quyết định
2.	Mã định danh cơ quan tổ chức phát hành	Mã định danh cơ quan tổ chức phát hành văn bản đi	Text	UBND
3.	Số văn bản	Số của văn bản đi	Số nguyên dương	123
4.	Kí hiệu văn bản	Mã loại văn bản đi	Text	QĐ
5.	Loại văn bản	Loại văn bản đi		Quyết định
6.	Ngày phát hành	Ngày phát hành văn bản đi	Hiển thị dưới dạng: DD/MM/YYYY DD: ngày, MM: tháng, YYYY: năm	24/01/2022
7.	Số trang	Số trang văn bản đi	Số nguyên dương	30
8.	Tài liệu	Các file đính kèm đi cùng văn bản đến	Là các file có dạng: doc, jpg, ...	QĐTC.doc
9.	Trích yếu nội dung	Tóm tắt nội dung văn bản đến	Text Editor	Quyết định thăng chức
10.	Độ mật	Độ mật		Mật
11.	Độ khẩn	Độ khẩn		Khẩn

3.17. Đặc tả usecase “tìm kiếm văn bản đi”

Tên UC	Tìm kiếm văn bản đi
Mô tả	Usecase này được thực hiện để tìm kiếm văn bản đi
Actor	Admin, Văn thư, lãnh đạo, nhân viên
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	5. Hệ thống hiển thị danh sách văn bản đi 6. Actor nhập các thông tin của văn bản đi 7. Actor xác nhận tìm kiếm văn bản đi 8. Hệ thống hiển thị văn bản tìm kiếm
Alternative Flow	1.2. Actor nhập không đúng trường dữ liệu 4.2. Hệ thống không tìm được văn bản
Post condition	Hệ thống hiển thị văn bản mà actor muốn tìm kiếm.

Dữ liệu đầu vào của tìm kiếm văn bản đi:

STT	Trường dữ liệu	Mô tả	Định dạng hiển thị	Ví dụ
1.	Tên văn bản	Tên văn bản đi	Text	Quyết định
2.	Loại văn bản	Loại văn bản đi	Text	Quyết định
3.	Ngày tạo	Số của văn bản đi	Hiển thị dưới dạng: YYYY/MM/DD DD: ngày, MM: tháng, YYYY: năm	2022/01/24
4.	Trạng thái	Trạng thái của văn bản đi	Text	Tiếp nhận

3.18. Đặc tả usecase “xem thông tin chi tiết văn bản đi”

Tên UC	Xem thông tin chi tiết văn bản đi
Mô tả	Usecase này được thực hiện để xem thông tin chi tiết văn bản
Actor	Admin, lãnh đạo, nhân viên
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> Hệ thống hiển thị danh sách văn bản đi Actor chọn văn bản muốn xem chi tiết Actor chọn chức năng xem chi tiết văn bản đi Hệ thống hiển thị form xem chi tiết văn bản
Alternative Flow	
Post condition	Hệ thống hiển thị thông tin chi tiết của văn bản đi

Dữ liệu đầu ra của form chi tiết văn bản đến:

STT	Trường dữ liệu	Mô tả	Định dạng hiển thị	Ví dụ
1.	Tên văn bản	Tên văn bản đi	Text	Quyết định
2.	Mã định danh cơ quan tổ chức phát hành	Mã định danh cơ quan tổ chức phát hành văn bản đi	Text	UBND
3.	Số văn bản	Số của văn bản đi	Số nguyên dương	123
4.	Kí hiệu văn bản	Mã loại văn bản đi	Text	QĐ
5.	Loại văn bản	Loại văn bản đi		Quyết định
6.	Ngày phát hành	Ngày phát hành văn bản đi	Hiện thị dưới dạng: DD/MM/YYYY DD: ngày, MM:	24/01/2022

			tháng, YYYY: năm	
7.	Thời hạn xử lý	Thời gian xử lý văn bản	Hiển thị dưới dạng: YYYY/MM/DD DD: ngày, MM: tháng, YYYY: năm	2022/01/24
8.	Hồ sơ gửi kèm	Các file đính kèm đi cùng văn bản đến	Là các file có dạng: doc, jpg, ...	QĐTC.doc
9.	Trích yếu nội dung	Tóm tắt nội dung văn bản đi	Text Editor	Quyết định thăng chức
10.	Độ mật	Độ mật		Mật
11.	Độ khẩn	Độ khẩn		Khẩn

3.19. Đặc tả usecase “xử lý văn bản đi”

Tên UC	Phân xử lý văn bản
Mô tả	Usecase này được thực hiện để phân văn bản đến phòng ban xử lý
Actor	Lãnh đạo
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> 1. Actor chọn văn bản muốn phân xử lý 2. Actor chọn chức năng Phân xử lý 3. Hệ thống hiển thị form phân xử lý 4. Actor phân xử lý và xác nhận phân xử lý 5. Hệ thống xác nhận và lưu thông tin
Alternative Flow	

Post condition	Phân xử lý văn bản thành công.
-----------------------	--------------------------------

3.20. Đặc tả usecase “cấp số”

Tên UC	Cấp số
Mô tả	Usecase này được thực hiện để cấp số văn bản đi
Actor	Văn thư
Precondition	Actor cần phải hoàn thành usecase “Thêm văn bản đi”
Main Flow	<ol style="list-style-type: none"> 1. Actor chọn văn bản cần cấp số 2. Actor chọn cấp số 3. Hệ thống hiển thị form cấp số 4. Actor cấp số cho văn bản 5. Hệ thống xác nhận và lưu thông tin
Alternative Flow	4.1. Cấp số bị lỗi – Số đã bị trùng lặp
Post condition	Hành động phê duyệt được thực hiện

3.21. Đặc tả usecase “phê duyệt”

Tên UC	Phê duyệt
Mô tả	Usecase này được thực hiện để phê duyệt văn bản đi
Actor	Lãnh đạo
Precondition	Actor cần phải hoàn thành usecase “Thêm văn bản đi” hoặc “Xử lý văn bản đi”
Main Flow	<ol style="list-style-type: none"> 1. Hệ thống hiển thị form xem thông tin chi tiết văn bản 2. Actor chọn Phê duyệt hoặc từ chối

	3. Hệ thống xác nhận và lưu thông tin
Alternative Flow	
Post condition	Hành động phê duyệt được thực hiện

3.22. Đặc tả usecase “thêm giấy mời”

Tên UC	Thêm giấy mời
Mô tả	Usecase này được thực hiện để thêm giấy mời
Actor	Văn thư
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> 1. Actor chọn chức năng Thêm giấy mời 2. Hệ thống hiển thị form thêm giấy mời 3. Actor nhập thông tin giấy mời 4. Actor xác nhận form 5. Hệ thống kiểm tra tính hợp lệ của thông tin và lưu vào database
Alternative Flow	3.1. Actor nhập sai thông tin, nhập không đúng quy định
Post condition	Giấy mời mới được thêm vào hệ thống

Dữ liệu đầu vào của form thêm giấy mời

STT	Trường dữ liệu	Mô tả	Định dạng hiển thị	Ví dụ
1.	Tên giấy mời	Tên giấy mời	Text	Quyết định
2.	Mã định danh cơ	Mã định danh cơ quan tổ chức phát	Text	UBND

	quan tổ chức phát hành	hành giấy mời		
3.	Số giấy mời	Số của giấy mời	Số nguyên dương	123
4.	Kí hiệu giấy mời	Mã loại giấy mời	Text	
5.	Loại giấy mời	Loại giấy mời		
6.	Ngày phát hành	Ngày phát hành giấy mời	Hiển thị dưới dạng: DD/MM/YYYY DD: ngày, MM: tháng, YYYY: năm	24/01/2022
7.	Số trang	Số trang giấy mời	Số nguyên dương	30
8.	Tài liệu	Các file giấy mời	Là các file có dạng: doc, jpg, ...	giaymoi.jpg
9.	Trích yếu nội dung	Tóm tắt nội dung giấy mời	Text Editor	Giấy mời tham gia hội nghị

3.23. Đặc tả usecase “tìm kiếm giấy mời”

Tên UC	Tìm kiếm giấy mời
Mô tả	Usecase này được thực hiện để tìm kiếm giấy mời
Actor	Admin, User
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> Hệ thống hiển thị danh sách giấy mời Actor nhập các thông tin của giấy mời Actor xác nhận tìm kiếm giấy mời Hệ thống hiển thị giấy mời tìm kiếm
Alternative	<ol style="list-style-type: none"> Actor nhập không đúng trường dữ liệu

Flow	4.1. Hệ thống không tìm được giấy mời
Post condition	Hệ thống hiển thị giấy mời mà actor muốn tìm kiếm.

Dữ liệu đầu vào của tìm kiếm văn bản đến:

STT	Trường dữ liệu	Mô tả	Định dạng hiển thị	Ví dụ
1.	Tên giấy mời	Tên giấy mời	Text	Quyết định
2.	Loại giấy mời	Loại giấy mời	Text	Quyết định
3.	Ngày tạo	Ngày thêm giấy mời vào hệ thống	Hiển thị dưới dạng: YYYY/MM/DD DD: ngày, MM: tháng, YYYY: năm	2022/01/24
4.	Trạng thái	Trạng thái của giấy mời	Text	Tiếp nhận

2.24. Đặc tả usecase “xem thông tin giấy mời”

Tên UC	Xem thông tin giấy mời
Mô tả	Usecase này được thực hiện để xem thông tin giấy mời
Actor	Admin, User
Precondition	Actor phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> Hệ thống hiển thị danh sách giấy mời Actor chọn giấy mời muốn xem chi tiết Actor chọn chức năng xem chi tiết giấy mời Hệ thống hiển thị form xem chi tiết giấy mời

Alternative Flow	
Post condition	Hệ thống hiển thị thông tin chi tiết của giấy mời.

Dữ liệu đầu ra của form chi tiết giấy mời:

STT	Trường dữ liệu	Mô tả	Định dạng hiển thị	Ví dụ
1.	Tên giấy mời	Tên giấy mời	Text	Quyết định
2.	Mã định danh cơ quan tổ chức phát hành	Mã định danh cơ quan tổ chức phát hành giấy mời	Text	UBND
3.	Số giấy mời	Số của giấy mời	Số nguyên dương	123
4.	Kí hiệu giấy mời	Mã loại giấy mời	Text	QĐ
5.	Loại giấy mời	Loại giấy mời		Quyết định
6.	Ngày phát hành	Ngày phát hành giấy mời	Hiển thị dưới dạng: DD/MM/YYYY DD: ngày, MM: tháng, YYYY: năm	24/01/2022
8.	Tài liệu	File giấy mời	Là các file có dạng: doc, jpg, ...	Giaymoi.jpg
9.	Trích yếu nội dung	Tóm tắt nội dung giấy mời	Text Editor	Quyết định thăng chức

3.25. Đặc tả usecase “xóa giấy mời”

Tên UC	Xóa giấy mời
---------------	--------------

Mô tả	Usecase này được thực hiện để xóa giấy mời
Actor	Văn thư
Precondition	Actor cần phải đăng nhập vào hệ thống
Main Flow	<ol style="list-style-type: none"> 1. Hệ thống hiển thị danh sách giấy mời 2. Actor chọn xóa giấy mời 3. Actor xác nhận xóa 4. Hệ thống xóa giấy mời khỏi hệ thống
Alternative Flow	
Post condition	Giấy mời được xóa khỏi hệ thống

4. Yêu cầu phi chức năng

4.1. Yêu cầu về giao diện người dùng

Hệ thống cần đáp ứng về các chuẩn thiết kế giao diện tạo hiệu ứng thị giác tốt như sau:

- Giao diện đơn giản, dễ nhìn rõ ràng.
- Thống nhất về màu sắc chủ đạo, font chữ các nút, form nhập thông tin có sự nhất quán cao.
- Thông báo rõ ràng trực quan không gây phiền toái cho người dùng.
- Định dạng ngày tháng theo chuẩn DD/MM/YYYY.

Hệ thống phải hỗ trợ gõ Tiếng Việt người hỗ trợ chuẩn hóa ngôn ngữ Unicode (TCVN 6909:2001). Người dùng có thể sử dụng các bộ gõ Tiếng Việt Unikey, Vietkey để soạn thảo nhập các thông tin mà không gặp trở ngại về ngôn ngữ

4.2. Yêu cầu về kỹ thuật

Sau khi thiết kế và xây dựng các chức năng hệ thống cần đáp ứng yêu cầu về mặt kỹ thuật như sau:

- Mật khẩu user được mã hóa theo chuẩn Bcrypt.
- Với các usecase ứng với từng vai trò người dùng phải đăng nhập để cấp quyền theo tài khoản.
- Phân quyền user trên cả giao diện và các RESTFul API
- Ký tự theo chuẩn UTF-8.
- Khả năng sao lưu dữ liệu để backup dữ liệu khi cần thiết.

- Server đáp ứng được hiệu năng cao khi nhiều người truy cập một lúc.
- Ứng dụng được thiết kế để dễ bảo trì và mở rộng sau này

4.3. Đáp ứng về xử lý logic với việc nhập liệu qua ô nhập liệu

Các ô nhập liệu trên giao diện cần đảm bảo các nguyên tắc:

- Các ô dữ liệu bắt buộc có ký hiệu thông báo cho người dùng.
- Các ô nhập có định dạng chuyên biệt cần có giao diện chọn. VD: chọn ngày tháng
- Dữ liệu được kiểm tra ngay thời điểm người dùng nhập liệu.
- Hiện thị thông báo ngay khi người dùng nhập dữ liệu không hợp lệ.
- Đối với các dữ liệu dạng cố định như danh mục, danh sách cụ thể cần hỗ trợ hiện thị danh sách chọn thay vì nhập liệu từ bàn phím.

Chương 3 Công nghệ sử dụng

Hệ thống được xây dựng dựa trên ngôn ngữ lập trình javascript sử dụng Framework Express, sử dụng cơ chế Server side rendering(SSR) toàn bộ phần Frontend sẽ được server xử lý rồi trả về cho người dùng dưới dạng html. Phần Frontend được viết trên template EJS kết hợp với một số thư viện như Bootstrap, JQuery, AdminLTE cùng với các kĩ thuật xử lý như AJAX,... kết hợp với ngôn ngữ truy vấn sql và cơ sở dữ liệu mysql để truy vấn cơ sở dữ liệu

3.1 Backend

3.1.1 : Javascript

JavaScript, theo phiên bản hiện hành, là một ngôn ngữ lập trình thông dịch được phát triển từ các ý niệm nguyên mẫu. Ngôn ngữ này được dùng rộng rãi cho các trang web (phía người dùng) cũng như phía máy chủ (với Nodejs). Nó vốn được phát triển bởi Brendan Eich tại Hãng truyền thông Netscape với cái tên đầu tiên Mocha, rồi sau đó đổi tên thành LiveScript, và cuối cùng thành JavaScript. Giống Java, JavaScript có cú pháp tương tự C, nhưng nó gần với Self hơn Java. .js là phần mở rộng thường được dùng cho tập tin mã nguồn JavaScript.

Ưu điểm của javascripts

- Bạn không cần một compiler vì web browser có thể biên dịch nó bằng HTML
- Nó dễ học hơn các ngôn ngữ lập trình khác;
- Lỗi dễ phát hiện hơn và vì vậy dễ sửa hơn;

- Nó có thể được gắn trên một số element của trang web hoặc event của trang web như là thông qua click chuột hoặc di chuột tới;
- JS hoạt động trên nhiều trình duyệt, nền tảng,
- Bạn có thể sử dụng JavaScript để kiểm tra input và giảm thiểu việc kiểm tra thủ công khi truy xuất qua database;
- Nó giúp website tương tác tốt hơn với khách truy cập;
- Nó nhanh hơn và nhẹ hơn các ngôn ngữ lập trình khác
- Giảm thiểu truyền tải dữ liệu với server: Các đầu vào dữ liệu được xác thực trước khi người dùng gửi đi, nếu dữ liệu không hợp lệ sẽ yêu cầu phía người dùng kiểm tra lại dữ liệu. Do đó tiết kiệm được lưu lượng tới máy chủ và giảm tải trên máy chủ.

Ngược điểm của javascripts

- Dễ bị khai thác;
- Có thể được dùng để thực thi mã độc trên máy tính của người dùng
- Nhiều khi không được hỗ trợ trên mọi trình duyệt
- JS code snippets lớn
- Có thể bị triển khai khác nhau tùy từng thiết bị dẫn đến việc không đồng nhất.

3.1.2.Nodejs

Nodejs là một nền tảng (Platform) phát triển độc lập được xây dựng trên V8 JavaScript Engine trình thông dịch thực thi mã JavaScript giúp chúng ta có thể xây dựng được các ứng dụng web như các trang video clip, các forum và đặc biệt là trang mạng xã hội phạm vi hẹp một cách nhanh chóng và dễ dàng mở rộng.

NodeJS có thể chạy trên nhiều nền tảng hệ điều hành khác nhau từ Window cho tới Linux, OS X nên đó cũng là một lợi thế. NodeJS cung cấp các thư viện phong phú ở dạng Javascript Module khác nhau giúp đơn giản hóa việc lập trình và giảm thời gian ở mức thấp nhất.

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

Đặc tính Nodejs

Qua phần tìm hiểu NodeJS là gì mình có giới thiệu một đặc tính rất quan trọng đó là Realtime, tuy nhiên vẫn còn khá nhiều đặc tính mà bạn cần phải biết trước khi học NodeJS.

Không đồng bộ: Tất cả các API của NodeJS đều không đồng bộ (non-blocking), nó chủ yếu dựa trên nền của NodeJS Server và chờ đợi Server trả dữ liệu về. Việc di chuyển máy chủ đến các API tiếp theo sau khi gọi và cơ chế thông báo các sự kiện của Node.js giúp máy chủ để có được một phản ứng từ các cuộc gọi API trước (Realtime).

Chạy rất nhanh: NodeJ được xây dựng dựa vào nền tảng V8 Javascript Engine nên việc thực thi chương trình rất nhanh.

Đơn luồng nhưng khả năng mở rộng cao: Node.js sử dụng một mô hình luồng duy nhất với sự kiện lặp. cơ chế tổ chức sự kiện giúp các máy chủ để đáp ứng một cách không ngăn chặn và làm cho máy chủ cao khả năng mở rộng như trái ngược với các máy chủ truyền thống mà tạo đề hạn chế để xử lý yêu cầu. Node.js sử dụng một chương trình đơn luồng và các chương trình tương tự có thể cung cấp dịch vụ cho một số lượng lớn hơn nhiều so với yêu cầu máy chủ truyền thống như Apache HTTP Server.

Không đệm: NodeJS không đệm bất kì một dữ liệu nào và các ứng dụng này chủ yếu là đầu ra dữ liệu.

Ưu điểm của Nodejs

- Có tốc độ xử lý nhanh nhờ cơ chế xử lý bất đồng bộ (non-blocking). Bạn có thể dễ dàng xử lý hàng ngàn kết nối trong khoảng thời gian ngắn nhất.
- Giúp bạn dễ dàng mở rộng khi có nhu cầu phát triển website.
- Nhận và xử lý nhiều kết nối chỉ với một single-thread. Nhờ đó, hệ thống xử lý sẽ sử dụng ít lượng RAM nhất và giúp quá trình xử Nodejs lý nhanh hơn rất nhiều.
- Có khả năng xử lý nhiều Request/s cùng một lúc trong thời gian ngắn nhất.
- Có khả năng xử lý hàng ngàn Process cho hiệu suất đạt mức tối ưu nhất.
- Phù hợp để xây dựng những ứng dụng thời gian thực như các ứng dụng chat, mạng xã hội ...

Nhược điểm của Nodejs

- Nodejs gây hao tốn tài nguyên và thời gian. Nodejs được viết bằng C++ và JavaScript nên khi xử lý cần phải trải qua một quá trình biên dịch. Nếu bạn cần xử lý những ứng dụng tốn tài nguyên CPU thì không nên sử dụng Nodejs.
- Nodejs so với các ngôn ngữ khác như PHP, Ruby và Python sẽ không có sự chênh lệch quá nhiều. Nodejs có thể sẽ phù hợp với việc phát triển ứng dụng mới. Tuy nhiên khi xây dựng và triển khai dự án quan trọng thì Nodejs không phải là sự lựa chọn hoàn hảo nhất.
- Bài viết trên đã giúp bạn giải đáp thắc mắc Nodejs là gì và đưa ra một số ưu điểm, nhược điểm của Nodejs. Hi vọng với những thông tin tìm hiểu về Nodejs hữu ích này sẽ giúp bạn có thể xác định được rõ có nên theo học Nodejs hay không. Chúc bạn sớm thành công.

3.1.3.Express

Express là 1 framework nodejs để tạo web application, bao gồm nhiều tính năng mạnh mẽ để xây dựng các ứng dụng web và mobile. Bản thân express framework khá tối giản, tuy nhiên các nhà phát triển đã tạo ra rất nhiều thư viện giúp giải quyết hầu hết các vấn đề trong lập trình web, bao gồm thư viện để làm việc với cookies, sessions, URL parameters, POST data, security headers..

2. Các thành phần chính

2.1 Route Handlers

Route Handlers quyết định ứng dụng sẽ đáp ứng các request từ phía client như thế nào. Route là 1 URI (hoặc đường dẫn) và method (POST, GET, PUT, ...), mỗi route có thể có 1 hoặc nhiều functions, các function này sẽ được thực thi khi route khớp. Về mặt cơ bản, tất cả route đều có thể được định nghĩa trong file js chính của application.

2.2 Middleware

Function Middleware là các function có thể truy xuất object request (req), object response (res), và function middleware kế tiếp trong chu kỳ request-response của ứng dụng.

2.3 Template Engine

Template Engine cho phép ta sử dụng các file view template trong ứng dụng, tại thời điểm thực thi, template engine sẽ thay thế các biến trong file template bằng giá trị được truyền, và chuyển đổi template đó thành file HTML để trả về phía client. Các template engine được sử dụng phổ biến trong Express là Pug, Mustache, EJS, Jade, ... bài viết này sử dụng Pug để làm ví dụ cho việc sử dụng template engine. Khai báo template engine để sử dụng như sau:

Ưu điểm Express:

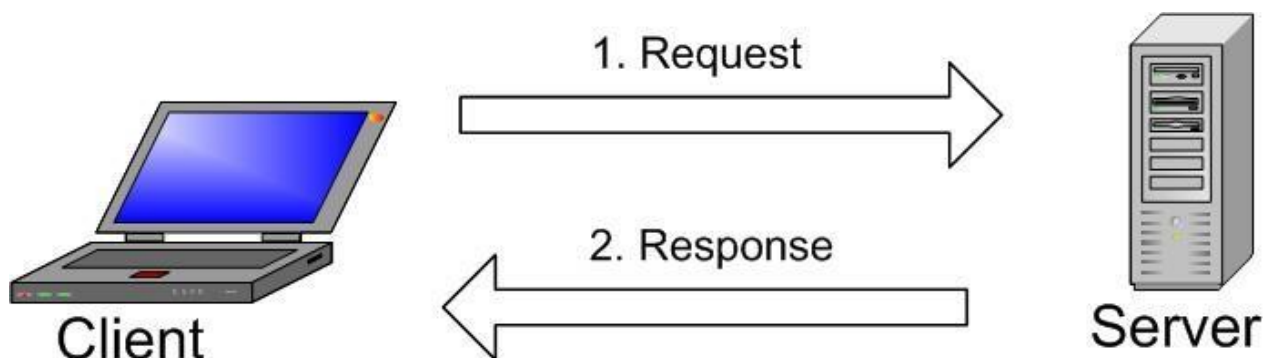
- Framework giúp cho việc phát triển ứng dụng được rút ngắn đi rất nhiều. Cũng như các framework dựa trên những ngôn ngữ khác như Rails (Ruby); Django (Python); Laravel, CakePHP (PHP)... Express được xây dựng dựa trên Node.js. Vậy nó có ưu điểm gì để ta lựa chọn cho việc phát triển ứng dụng.
- Express hỗ trợ việc phát triển ứng dụng theo mô hình MVC, mô hình phổ biến cho việc lập trình web hiện nay.
- Cho phép định nghĩa Middleware hỗ trợ cho việc tổ chức và tái sử dụng code.
- Định nghĩa routes và các request method đến server một cách dễ dàng.
- Hỗ trợ REST API.
- Và còn rất nhiều thứ mà framework này hỗ trợ, còn bây giờ hãy thử cài đặt và xây dựng một ứng dụng đơn giản nhé.

3.1.4. Mysql

MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (Relational Database Management System, viết tắt là RDBMS) hoạt động theo mô hình client-server. RDBMS là một phần mềm hay dịch vụ dùng để tạo và quản lý các cơ sở dữ liệu (Database) theo hình thức quản lý các mối liên hệ giữa chúng.

MySQL là một trong số các phần mềm RDBMS. RDBMS và MySQL thường được cho là một vì độ phổ biến quá lớn của MySQL. Các ứng dụng web lớn nhất như Facebook, Twitter, YouTube, Google, và Yahoo! đều dùng MySQL cho mục đích lưu trữ dữ liệu. Kể cả khi ban đầu nó chỉ được dùng rất hạn chế nhưng giờ nó đã tương thích với nhiều hạ tầng máy tính quan trọng như Linux, macOS, Microsoft Windows, và Ubuntu.

Các thức hoạt động MYSQL



Hình ảnh trên giải thích cấu trúc cơ bản về việc giao tiếp giữa client-server model. Một máy client sẽ liên lạc với máy server trong một mạng nhất định. Mỗi client có thể gửi một request từ giao diện người dùng (Graphical user interface – GUI) trên màn hình, và server sẽ trả về kết

quả như mong muốn. Miễn là cả hai hiểu nhau. Cách vận hành chính trong môi trường MySQL cũng như vậy:

- MySQL tạo ra bảng để lưu trữ dữ liệu, định nghĩa sự liên quan giữa các bảng đó.
- Client sẽ gửi yêu cầu SQL bằng một lệnh đặc biệt trên MySQL.
- Ứng dụng trên server sẽ phản hồi thông tin và trả về kết quả trên máy client.

Từ máy client, việc chọn GUI MySQL khá quan trọng. GUI càng nhẹ chừng nào, thì các thao tác quản lý data sẽ càng dễ dàng và nhanh chừng đó. MySQL GUI phổ biến nhất MySQL WorkBench, SequelPro, DBVisualizer, và Navicat DB Admin Tool. Một vài trong số chúng miễn phí, một vài bản thương mại, một vài bản chỉ chạy được trên macOS, và một vài ứng dụng chạy được hết trên các hệ điều hành phổ biến. Clients nên chọn GUI tùy vào nhu cầu của họ. Để quản lý web database, ví dụ như một trang web WordPress, rõ ràng nên chọn phpMyAdmin.

Ưu điểm của MySQL

- Sử dụng dễ dàng: MySQL là cơ sở dữ liệu tốc độ cao và ổn định, công cụ này dễ sử dụng và hoạt động trên nhiều hệ điều hành cung cấp hệ thống lớn các hàm tiện ích.
- Tính bảo mật cao: MySQL phù hợp với các ứng dụng có truy cập cơ sở dữ liệu trên internet vì nó sở hữu nhiều tính năng bảo mật, thậm chí là bảo mật cấp cao.
- Đa tính năng: MySQL có thể hỗ trợ hàng loạt các chức năng SQL từ hệ quản trị cơ sở dữ liệu quan hệ trực tiếp và cả gián tiếp.
- Khả năng mở rộng và mạnh mẽ: Công cụ MySQL có khả năng xử lý khối dữ liệu lớn và có thể mở rộng khi cần thiết.
- Tương thích trên nhiều hệ điều hành: MySQL tương thích để chạy trên nhiều hệ điều hành, như Novell NetWare, Windows * Linux *, nhiều loại UNIX * (như Sun * Solaris *, AIX và DEC * UNIX), OS / 2, FreeBSD *,..... MySQL cũng cung cấp phương tiện mà các máy khách có thể chạy trên cùng một máy tính với máy chủ hoặc trên một máy tính khác (giao tiếp qua mạng cục bộ hoặc Internet).
- Cho phép khôi phục: MySQL cho phép các transaction được khôi phục, cam kết và phục hồi sự cố.

Nhược điểm của MySQL

- MySQL bị hạn chế dung lượng, cụ thể, khi số bản ghi của người dùng lớn dần, sẽ gây khó khăn cho việc truy xuất dữ liệu, khiến người dùng cần áp dụng nhiều biện pháp để tăng tốc độ chia sẻ dữ liệu như chia tải database ra nhiều server, hoặc tạo cache MySQL.
- So với Microsoft SQL Server hay Oracle, độ bảo mật của MySQL chưa cao bằng. Và quá trình Restore cũng có phần chậm hơn. Tuy nhiên, những nhược điểm này không đáng kể với những hệ quản trị nội dung cỡ trung, bởi chỉ những hệ thống lớn hoặc rất lớn mới phát sinh các yêu cầu phức tạp hơn.

3.2 Front end

3.2.1 HTML và CSS

HTML (HyperText Markup Language) : là một ngôn ngữ đánh dấu được thiết kế ra để tạo nên các trang web, nghĩa là các mẫu thông tin được trình bày trên World Wide Web.

CSS (Cascading Style Sheets) : định nghĩa về cách hiển thị của một tài liệu HTML. CSS đặc biệt hữu ích trong việc thiết kế Web. Nó giúp cho người thiết kế dễ dàng áp dụng các phong cách đã được thiết kế lên bất kì page nào của website một cách nhanh chóng, đồng bộ

3.2.2 Teemplate Engine

Template Engine tạo ra các khuôn mẫu, cho phép bạn sử dụng các tệp mẫu tĩnh trong ứng dụng của mình. Trong thời gian chạy, Template Engine sẽ thay thế các biến trong tệp mẫu bằng các giá trị thực và chuyển đổi mẫu thành tệp HTML được gửi đến client. Sử dụng cách này sẽ giúp thiết kế trang HTML nhanh và tổ chức các component được nhanh hơn. Ngoài ra nó có thể đưa dữ liệu vào từ phía server và render ra đoạn HTML cuối cùng.

3.2.3 Bootstrap

Bootstrap cho phép quá trình thiết kế website diễn ra nhanh chóng và dễ dàng hơn dựa trên những thành tố cơ bản sẵn có như typography, forms, buttons, tables, grids, navigation, image carousels... Cùng Mất Bão tìm hiểu tính năng và lợi ích mang lại cho lập trình viên của Bootstrap là gì nhé!

Bootstrap là một bộ sưu tập miễn phí của các mã nguồn mở và công cụ dùng để tạo ra một mẫu website hoàn chỉnh. Với các thuộc tính về giao diện được quy định sẵn như kích thước, màu sắc, độ cao, độ rộng..., các designer có thể sáng tạo nhiều sản phẩm mới mẻ nhưng vẫn tiết kiệm thời gian khi làm việc với framework này trong quá trình thiết kế giao diện website.

Ưu điểm

- tạo giao diện nhanh chóng với component mà không cần lặp lại quá trình viết ra các đoạn mã HTML hay class CSS.
- Nền tảng tối ưu: Bootstrap được xây dựng với một thư viện lưu trữ mà bạn có thể dễ dàng sử dụng và tùy chỉnh theo mục đích của mình. Điều này giúp cho việc thiết kế website trở nên dễ dàng và nhanh chóng hơn với bạn có thể lựa chọn cho mình một mẫu phù hợp và thay đổi màu sắc, hình ảnh
- Tương tác tốt với smartphone
- Giao diện đầy đủ, sang trọng
- Dễ dàng tùy biến:

Ngược điểm

- Sản phẩm nặng, tốc độ tối ưu chưa cao
- Chưa hoàn thiện
- Quá nhiều code thừa
- Bootstrap không khuyến khích sáng tạo

3.2.4 JQuery

jQuery là thư viện javascript nổi bật nhất, phổ biến nhất trong lập trình web. Ra đời vào năm 2006 bởi John Resig. Nó đã trở thành một thành phần không thể thiếu trong các website có sử dụng Javascript. Với slogan “Write less – Do more” (viết ít hơn – làm nhiều hơn) nó đã giúp lập trình viên tiết kiệm được rất nhiều thời gian và công sức trong việc thiết kế website.

Ưu điểm:

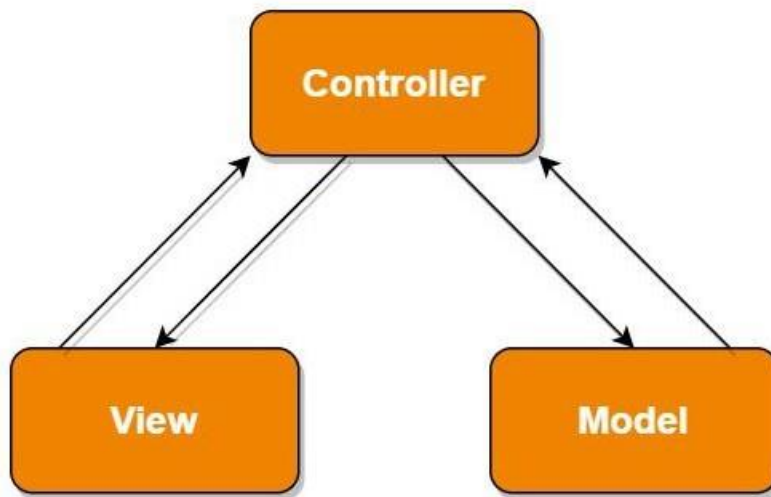
- Gọn nhẹ: jQuery là một thư viện khá gọn nhẹ – có kích cỡ khoảng 19KB (gzipped).
- Tương thích đa nền tảng: Nó tự động sửa lỗi và chạy được trên mọi trình duyệt phổ biến như Chrome, Firefox, Safari, MS Edge, IE, Android và iOS.
- Dễ dàng tạo Ajax: Nhờ thư viện jQuery, code được viết bởi Ajax có thể dễ dàng tương tác với server và cập nhật nội dung tự động mà không cần phải tải lại trang.
- Xử lý nhanh nhạy thao tác DOM: jQuery giúp lựa chọn các phần tử DOM để traverse (duyệt) một cách dễ dàng, và chỉnh sửa nội dung của chúng bằng cách sử dụng Selector mã nguồn mở, mà còn được gọi là Sizzle.
- Đơn giản hóa việc tạo hiệu ứng: Giống với code snippet có hiệu ứng animation, nó phủ các dòng code và bạn chỉ việc thêm biến/nội dung vào thôi.
- Hỗ trợ tốt phương thức sự kiện HTML: Xử lý sự kiện – jQuery xử lý các sự kiện đa dạng mà không làm cho HTML code trở nên lộn xộn với các Event Handler.

Chương 4 Phát triển và triển khai ứng dụng

4.1 Thiết kế kiến trúc

4.1.1 Lựa chọn kiến trúc phần mềm

MVC là viết tắt của Model – View – Controller. Là một mẫu kiến trúc phần mềm hay mô hình thiết kế được sử dụng trong kỹ thuật phần mềm để tạo lập giao diện người dùng trên máy tính. MVC chia ứng dụng thành ba phần (layer) riêng biệt, độc lập và có thể tương tác được với nhau



Hình 4.1 Mô hình MCV

Mô hình MVC chia ứng dụng thành ba phần bao gồm Model, View và Controller: Model

Là nơi chứa các logic, nghiệp vụ tương tác với dữ liệu hoặc hệ quản trị cơ sở dữ liệu (MySQL, SQL Server...), nó sẽ bao gồm các phương thức xử lý kết nối database, truy vấn dữ liệu. Là nơi lưu giữ các đối tượng mô tả dữ liệu, như là Class và các hàm xử lý get, set của Class...

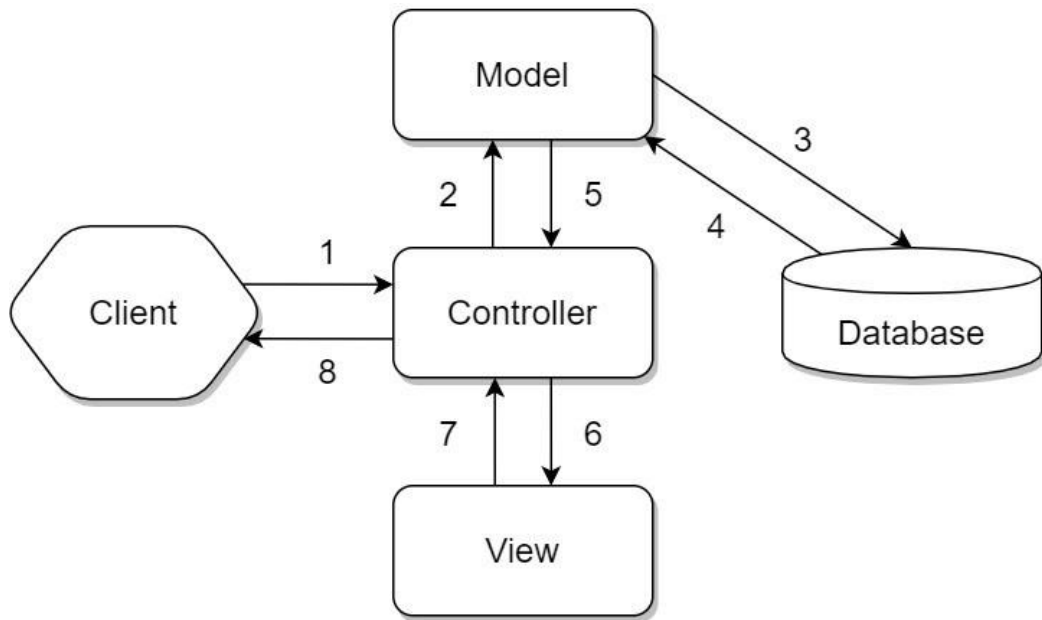
View

Đảm nhận việc hiển thị, trả về thông tin, dữ liệu cho end-user. Ví dụ như hiển thị UI/UX, hiển thị dữ liệu ra cho người dùng xem website, hoặc có thể là một đoạn XML hoặc JSON...

Controller

Giữ nhiệm vụ tiếp nhận, điều hướng yêu cầu từ end-user để gọi đúng phương thức xử lý, thao tác trực tiếp với Model và trả về dữ liệu cho View. Đây còn là nơi quản lý sự trao đổi dữ liệu và nguyên tắc nghiệp vụ trong các thao tác liên quan đến mô hình. Controller giữ vai trò trung gian giữa Model và View.

Luồng xử lý trong mô hình MVC



Hình 4.2 Luồng xử lý trong mô hình MVC

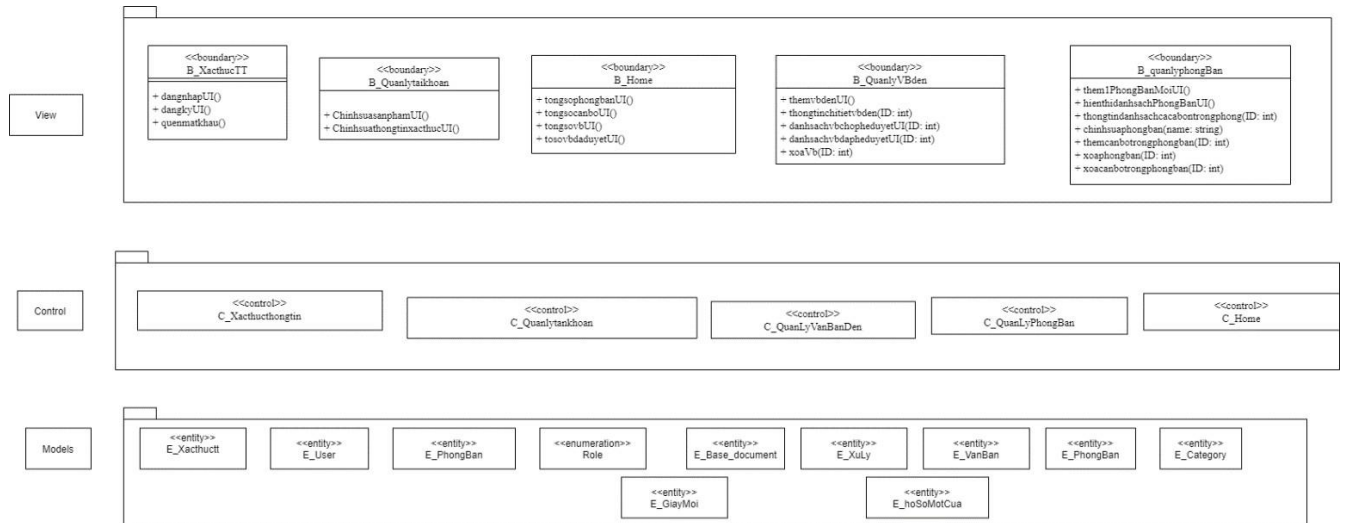
- Client gửi yêu cầu đến server thông qua Controller, Controller sẽ tiếp nhận yêu cầu.
- Controller sẽ xử lý dữ liệu đầu vào, và quyết định luồng đi tiếp theo của yêu cầu. Trả về kết quả hay tương tác với database để lấy dữ liệu. Nếu cần tương tác với Model để lấy dữ liệu, Controller sẽ gọi tới Model để lấy dữ liệu đầu ra. Nếu không, Controller sẽ trả về kết quả theo mũi tên số (8).
- Model tương tác với Database để truy xuất dữ liệu phù hợp với yêu cầu.
- Database trả về cho Model dữ liệu theo yêu cầu của Model.

- Model trả về dữ liệu cho Controller xử lý.
- Controller sẽ gọi đến View phù hợp với yêu cầu và kèm theo dữ liệu cho View. View chịu trách nhiệm hiển thị dữ liệu phù hợp với yêu cầu.

- Sau khi xử lý hiển thị dữ liệu, View trả về cho Controller kết quả (HTML, XML JSON...).
- Sau khi hoàn tất, Controller sẽ trả về kết quả cho Client.

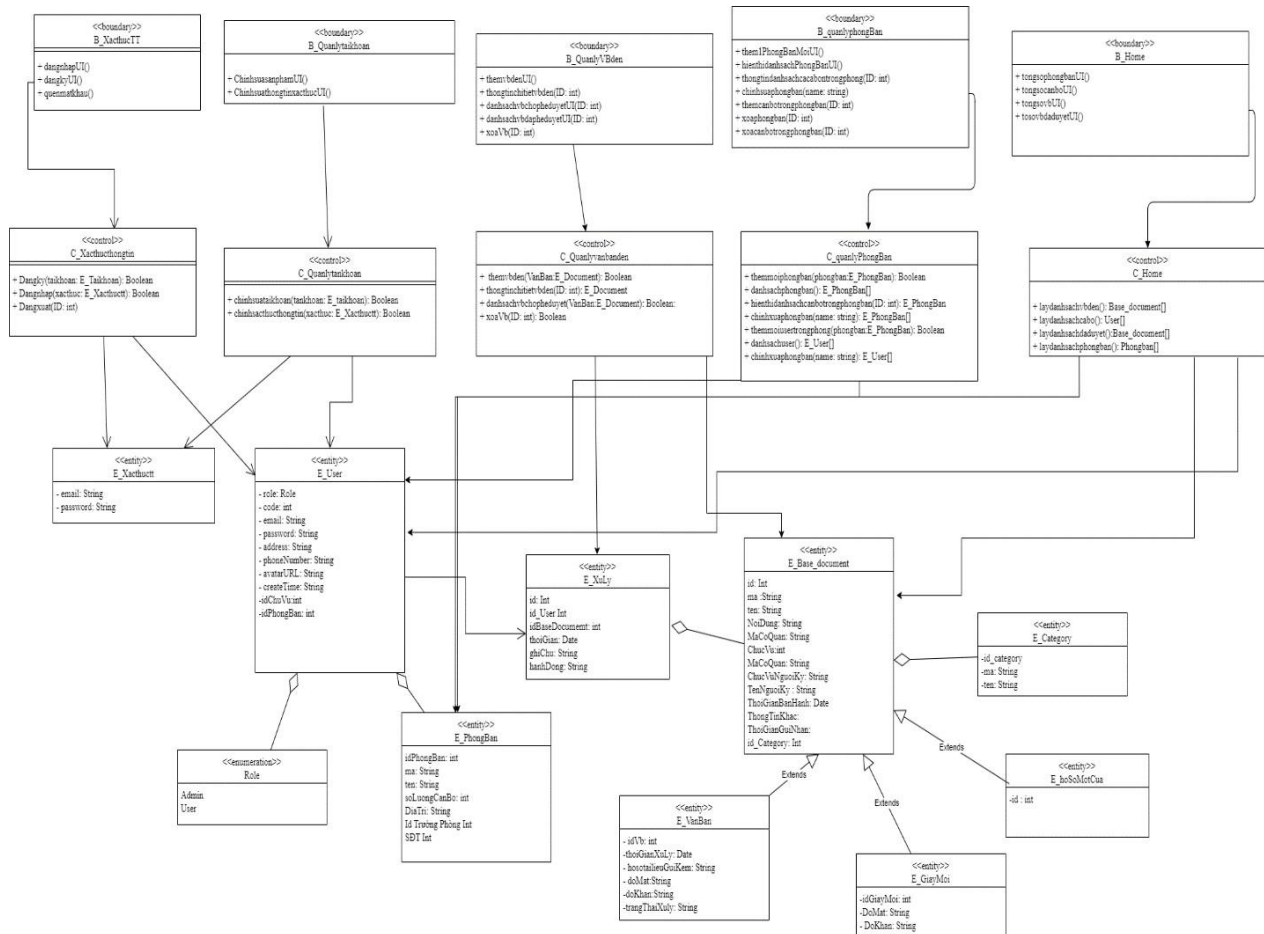
4.1.2 Thiết kế tổng quan các gói trong hệ thống

a) các gói trong chương trình được đóng theo mô hình MCV



Hình 4.3 Thiết kế các gói trong hệ thống

4.1.3 Thiết kế chi tiết gói



Hình 4.4 Thiết kế chi các lớp cho toàn bộ chương trình

4.2 Thiết kế chi tiết

4.2.1 Thiết kế giao diện

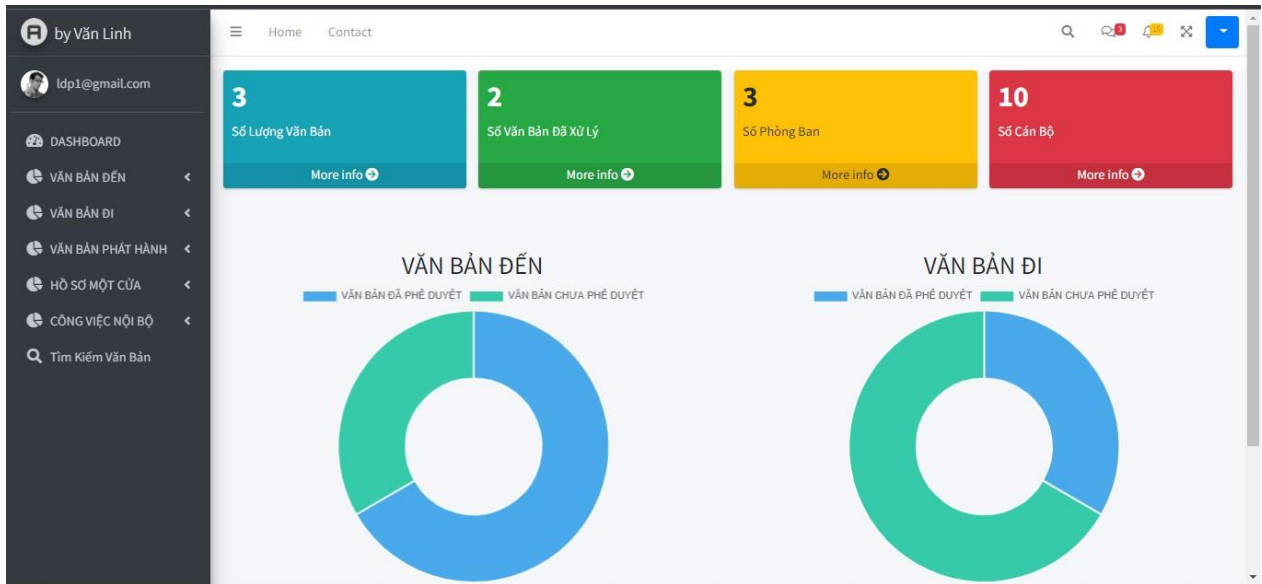
Giao diện của ứng dụng cần được thiết kế nhằm đảm bảo các nguyên tắc thiết kế, giúp tăng tính trải nghiệm cho người dùng. Giao diện màn phù hợp với các thiết bị có kích thước lớn như laptop, máy tính bảng... Đảm bảo tính responsive với từng kích thước màn hình. Về màu sắc, hệ thống sử dụng tối đa ba màu sắc trên một màn hình giao diện không kể các video, hình ảnh để tránh tạo cảm giác khó chịu về thị giác cho người dùng.

Các nút bấm của giao diện được thiết kế đồng nhất, tất cả đều phải là hình chữ nhật và được làm tròn cạnh ở các góc nút bấm. Vị trí hiển thị thông điệp phải ở trên đầu trang để người dùng dễ dàng quan sát.

Một số giao diện minh họa cho các chức năng của hệ thống:

a) Giao diện trang chủ của hệ thống

Giao diện trang chủ của hệ thống là giao diện cung cấp các chức năng chính cho việc hiển thị số lượng văn bản, tổng số lượng văn bản đến cũng như thông tin về số cán bộ và số phòng ban



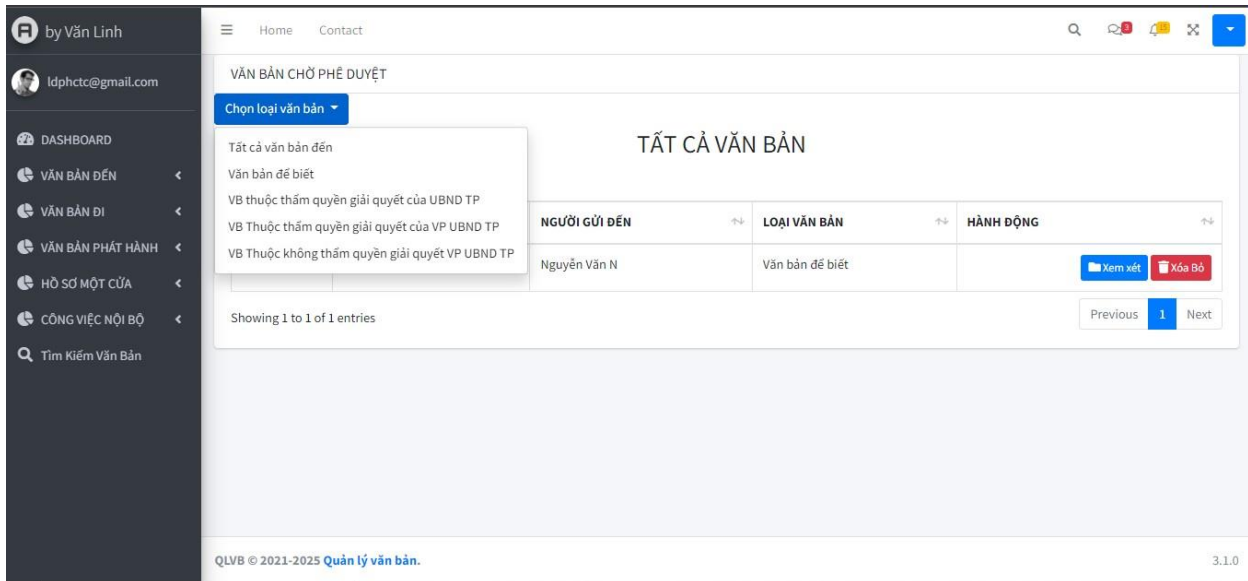
Hình 4.5 Thiết kết giao diện trang chủ

C) Giao diện chức năng hiển thị tất cả các văn bản chưa được phê duyệt của hệ thống

Giao diện này của hệ thống là giao diện cung cấp các chức năng chính cho việc hiển thị tất cả các văn bản chưa được phê duyệt của hệ thống

Và có thể tìm kiếm theo loại văn bản giúp người dùng có thể tìm kiếm chính xác loại văn bản mà mình muốn nhưng ở trong hình có các loại văn bản sau

- Văn bản để biết
- Văn bản thuộc thẩm quyền giải quyết của UBND TP
- Văn bản thuộc thẩm quyền giải quyết của VP UBND TP
- Văn bản không thuộc thẩm quyền giải quyết của VP UBND TP



Hình 4.6 Thiết kế giao diện trong danh sách văn bản chờ phê duyệt

D) Giao diện chức năng xem chi tiết văn bản

Giao diện này của hệ thống là giao Giao diện cung cấp các chức năng chính cho việc xem chi tiết văn bản đến và có thể hiển thị thông tin như:

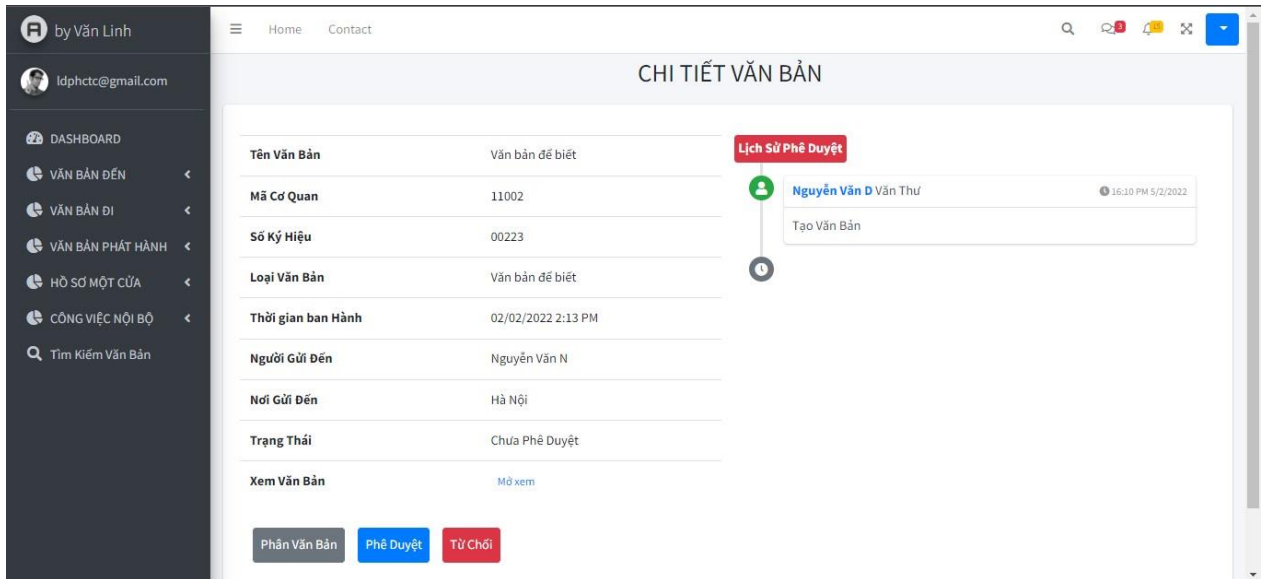
- Tên văn bản
- Mã cơ quan
- Loại văn bản
- Thời gian ban hành

Và có thể xem lịch sử phê duyệt của văn bản đề thuật tiện trong qua trình phê duyệt văn bản sẽ có thể hiển thị được

- Tên người phê duyệt
- Hành động thực hiện của người đó
- Thời gian thực hiện
- Chức vụ của người thực hiện

Hiển thị 1 số chức năng mà người dùng có thể làm với hệ thống

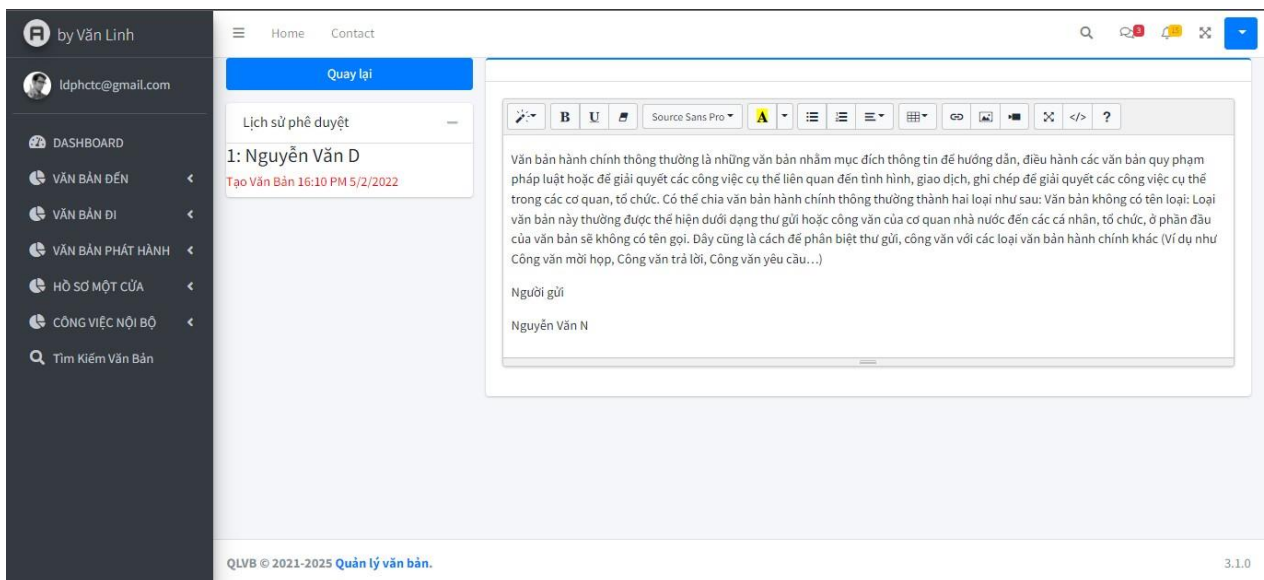
- Phân văn bản
- Phe duyệt văn bản
- Từ chối văn bản



Hình 4.7 Thiết kết giao diện xem chi tiết văn bản chờ phê duyệt

E) Giao diện chức năng xem chi tiết nội dung của văn bản gửi đến

khi người dùng ấn vào nút mở văn bản ở trên kia sẽ điều hướng người dùng sang 1 trang khác để đọc nội dung văn bản, và có thể hiển thị lịch sử phê duyệt văn bản



Hình 4.8 Thiết kết giao diện xem chi tiết nội dung của văn bản gửi đến

f) Giao diện chức năng tìm kiếm văn bản

Giao diện này cung cấp cho người dùng một số cách thức để cho thể tìm kiếm văn bản người dùng có thể tìm bằng một trong số thông tin sau:

- Tên Văn Bản
- Từ khóa tìm kiếm
- Mã cơ quan
- Số và kí hiệu
- Loại văn bản
- Nơi gửi đến

Khi tìm kiếm thì các văn bản có từ khóa liên quan sẽ hiện ở ngay dưới điều này giúp cho người dùng thuật tiện trong việc xem và tìm kiếm dễ dàng hơn người dùng có thể xem và xóa văn bản

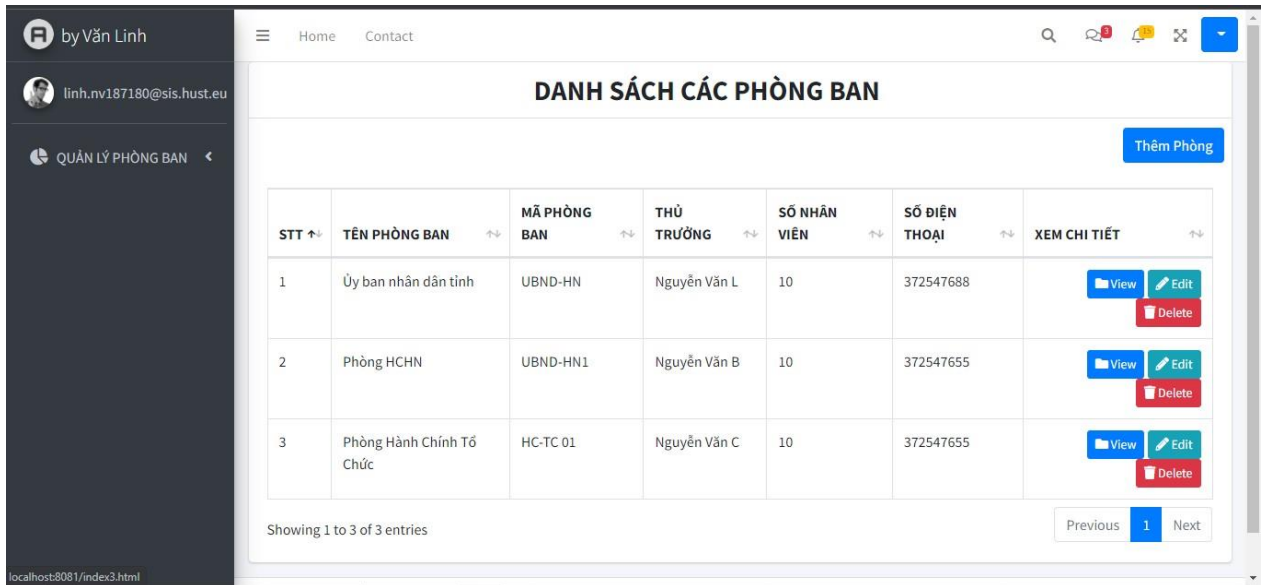
STT	TÊN VĂN BẢN	NGƯỜI GỬI ĐẾN	LOẠI VĂN BẢN	TRẠNG THÁI	HÀNH ĐỘNG
1	Văn bản đề biết	Nguyễn Văn N	Văn bản đề biết	Đã Duyệt	<button>Xem Văn Bản</button> <button>Xóa Bỏ</button>
2	Văn bản đề biết	Nguyễn Văn G	VB thuộc thẩm quyền giải quyết của UBND TP	Đã Duyệt	<button>Xem Văn Bản</button> <button>Xóa Bỏ</button>

Hình 4.9 Thiết kết giao diện xem chi tiết nội dung của văn bản gửi đến

g) Thiết kết giao diện xem danh sách các phòng ban

Giao diện này là một trong số các giao diện quan trọng nhất trong chương trình người dùng sẽ xem được danh sách phòng ban có trong hệ thống và có thể thêm 1 phòng ban mới và thể thể hiện thị một số thông tin chính về phòng ban

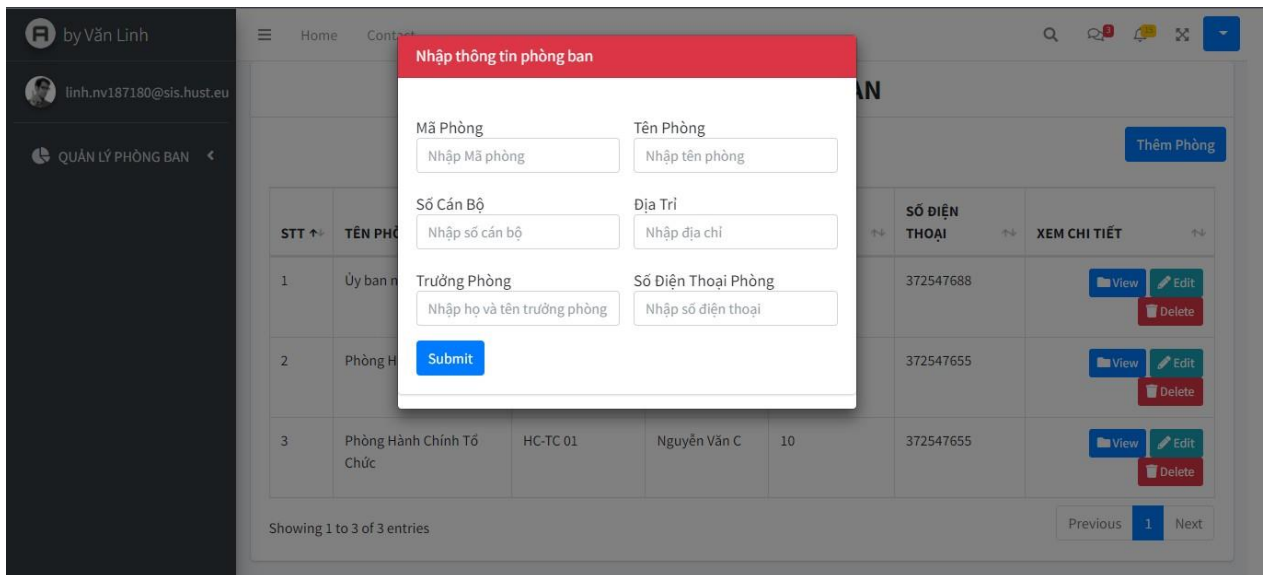
- Tên phòng
- Thủ trưởng
- Số nhân viên trong phòng ban
- Số điện thoại của phòng
- Và có thể xem chi tiết cũng như Admin có thể xóa phòng ban



Hình 4.10 Thiết kết giao diện xem danh sách các phòng ban

h) giao diện thêm một phòng ban mới vào hệ thống

Thêm mới một phòng ban khi mà yêu cầu của hệ thống cần thêm một phòng ban mới thì người dùng sẽ nhập một số thông tin quan trọng như hình ở dưới.



Hình 4.11 Thiết kết giao diện thêm một phòng ban mới vào hệ thống

k) Thiết kết giao diện thêm một phòng ban mới vào hệ thống

Tương tự với thêm phòng ban thì các bộ trong một phòng sẽ thay đổi liên tục từ đó phát sinh ra yêu cầu thêm mới 1 người vào trong phòng ban Admin sẽ cần phải nhập đúng các thông tin sau:

- Họ và tên
- Email
- Địa chỉ
- Ngày sinh
- Mật khẩu
- Giới tính
- Chức vụ

The screenshot displays a web interface for managing staff. A modal form titled 'Thông Tin Cán Bộ' is open, allowing the addition of new staff members. The form contains the following fields:

- Họ Và Tên: Nhập họ và tên
- Địa Trĩ: Nhập địa chỉ
- Email: Nhập Email
- Mật khẩu: Nhập mật khẩu
- Ngày Sinh: Nhập ngày sinh
- Giới Tính: Nam (dropdown menu)
- Số Điện Thoại: Nhập số điện thoại
- Chức Vụ: Văn Thư (dropdown menu)

A 'Submit' button is located at the bottom of the form. In the background, a table lists existing staff members with columns for STT, HỌ TÊN, NGÀY SINH, GIỚI TÍNH, CHỨC VỤ, and HÀNH ĐỘNG (Edit/Delete).

Hình 4.12 Thiết kết giao diện thêm một phòng ban mới vào hệ thống

4.2.2 Thiết kế cơ sở dữ liệu

a) Xây dựng biểu đồ thực thể liên kết

Các thực thể chính trong hệ thống gồm có:

User: là các bộ hoặc Admin có các thông tin cơ bản như tên, ngày sinh, giới tính, địa chỉ, chức vụ, password

Phòng Ban: tên phòng, mã phòng, id của trưởng phòng, số điện thoại phòng, email phòng

Document: Tên văn bản, nội dung mã cơ quan, thời gian gửi, thời gian nhận

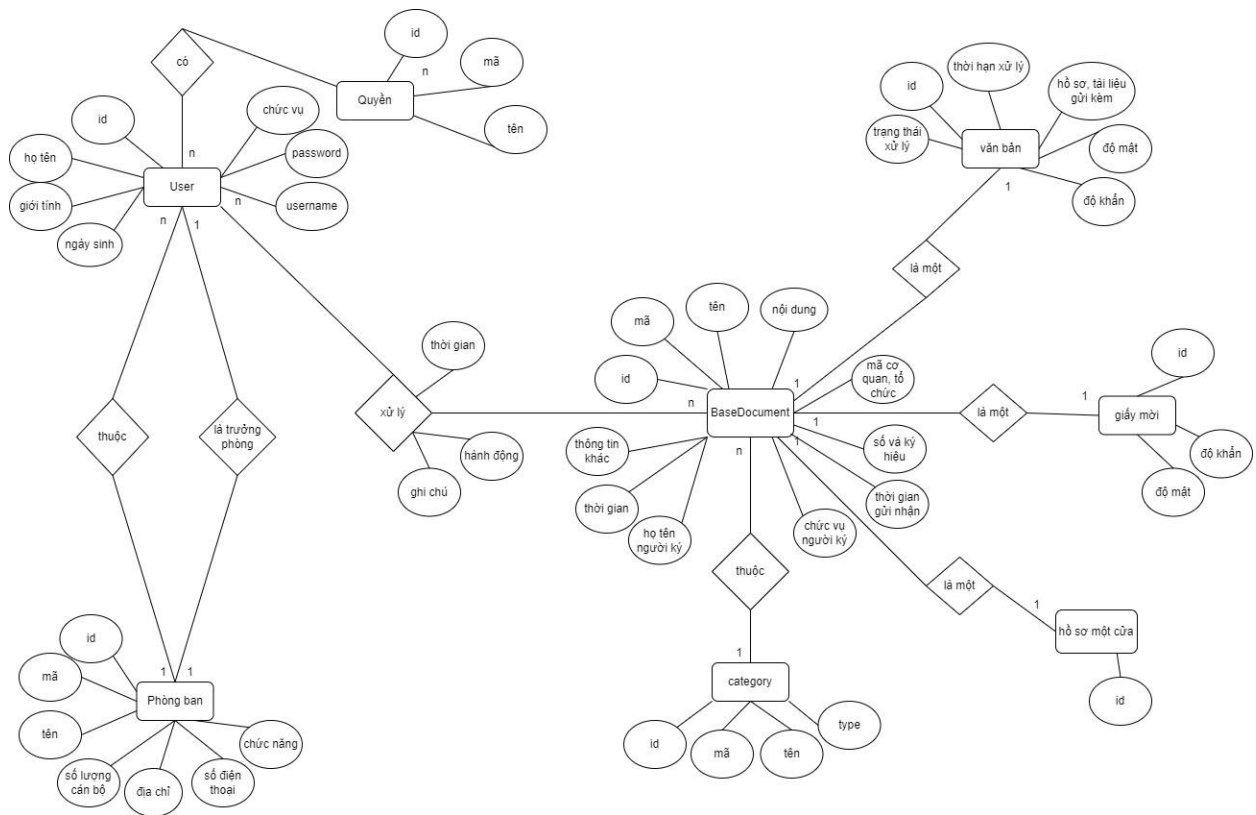
Văn Bản: id, thời gian xử lý, hồ sơ tài liệu đi kèm, độ mật, độ khẩn, trạng thái xử lý Giấy

Mời: độ khẩn, độ mật

Hồ Sơ Một Cửa: id

Category: là loại văn bản

Xử lý: Thời gian xử lý, hoành động, user xử lý, ghi chú



b) Xây dựng biểu đồ cơ sở dữ liệu

Từ bản thiết kế biểu đồ thực thể liên kết, em xây dựng được biểu đồ cơ sở dữ liệu hoàn chỉnh như hình dưới:

#	PK	FK	Column Name	Data type	Mandatory	Description
3.			Gioi_tinh	boolean	Yes	No (nam) / Yes (nữ)
4.			Ngay_sinh	DATETIME(45)	Yes	11/11/1999
5.			username	VARCHAR(45)	Yes	Anh.nv
6.			password	VARCHAR(45)	Yes	123456
7.			Chuc_vu	VARCHAR(45)	Yes	Thủ trưởng
8.		FK	ID_phong_ban	Interger	yes	012

▪ Role

#	PK	FK	Column Name	Data type	Mandatory	Description
9.	x		id	Integer	Yes	ID, auto increment
10.			Mã_quyền	VARCHAR(45)	Yes	123
11.			Ten_quyen	VARCHAR(45)	Yes	admin

▪ User_role

#	PK	FK	Column Name	Data type	Mandatory	Description
12.	x	x	Id_user	interger	Yes	123
13.	x	x	Id_quyen	interger	Yes	123

Phòng ban

#	PK	FK	ColumnName	Data type	Mandatory	Description
1			Id_phong ban	Interger	Yes	ID phòng ban
2			Mã	VARCHAR(10)	Yes	Mã phòng ban
3			Tên	VARCHAR(45)	Yes	Tên phòng ban

4			Số lượng cán bộ	Interger	Yes	Số lượng cán bộ có trong phòng ban.
---	--	--	--------------------	----------	-----	--

5			Địa chỉ	VARCHAR(200)	Yes	Địa chỉ phòng ban
6		x	ID trưởng phòng	Interger	Yes	ID trưởng phòng ban
7			SĐT	Interger	Yes	Số điện thoại của phòng ban

Base_document

#	PK	FK	ColumnName	Data type	Mandatory	Description
1			id	Interger	Yes	ID văn bản/giấy mời/hồ sơ
2			Mã	VARCHAR(10)	Yes	Mã văn bản/giấy mời/hồ sơ
3			Tên	VARCHAR(45)	Yes	Tên văn bản/giấy mời/hồ sơ
4			Nội dung	VARCHAR(200)	Yes	Nội dung văn bản/giấy mời/hồ sơ
5			Mã cơ quan	VARCHAR(10)	Yes	Mã cơ quan tổ chức được ghi trong văn bản/ giấy mời/ hồ sơ.
6			Số và kí hiệu	VARCHAR(20)	Yes	Số và kí hiệu văn bản/giấy mời/hồ sơ
7			Chức vụ người ký	VARCHAR(45)	Yes	Chức vụ người ký trong văn bản/giấy mời/hồ sơ
8			Tên người ký	VARCHAR(45)		Tên người ký trong văn bản/giấy mời/hồ sơ
9			Thời gian ban hành	DATETIME		Thời gian ban hành văn bản/giấy mời/hồ sơ
10			Thông tin khác	VARCHAR(200)		
11			Thời gian gửi nhận	DATETIME		Thời gian gửi nhận văn bản/giấy mời/hồ sơ
12		x	Id_Category	Interger		

Category

#	PK	FK	ColumnName	Data type	Mandatory	Description
1		x	Id_category	Interger	Yes	ID category
2			Mã	VARCHAR(10)	Yes	Mã catedory
3			Tên	VARCHAR(45)	Yes	Tên catedory
4			Type	VARCHAR(10)	Yes	Kiểu văn bản/hồ sơ/giấy mời

Văn bản

#	PK	FK	ColumnName	Data type	Mandatory	Description
1		x	Id_van ban	Interger	yes	Id của văn bản
2			Thời hạn xử lý	DATETIME	yes	Thời gian xử lý văn bản
3			Hồ sơ tài liệu gửi kèm	VARCHAR(200)		Hồ sơ gửi kèm theo văn bản (nếu có)
4			Độ mật	VARCHAR(45)	yes	Độ bảo mật của văn bản -
5			Độ khẩn	VARCHAR(45)	yes	Độ khẩn của văn bản -Khẩn -Thượng khẩn -Hỏa tốc
6			Trạng thái xử lý	VARCHAR(45)	yes	

Giấy mời

#	PK	FK	ColumnName	Data type	Mandatory	Description
1		x	Id_giay moi	Interger	yes	Id của giấy mời
2			Độ mật	VARCHAR(45)		Độ bảo mật của văn bản -
3			Độ khẩn	VARCHAR(45)		Độ khẩn của văn bản -Khẩn -Thượng khẩn -Hỏa tốc

Hồ sơ một cửa

#	PK	FK	ColumnName	Data type	Mandatory	Description
1		x	Id_ho so mot cua	Interger	yes	Id của hồ sơ một cửa

Xử lý

#	PK	FK	ColumnName	Data type	Mandatory	Description
1			Id	Interger	yes	Id của giấy mời
2		x	Id user	Interger	yes	Id của user
3		x	Id base_document	Interger	yes	Id của base_document
4			Thời gian	DATENAME		Thời gian xử lý
5			Ghi chú	VARCHAR(45)		Nội dung cần ghi chú
6			Hành động	VARCHAR(45)		Hành động sửa thêm xóa của văn bản

4.3 Kết quả đạt được

Hệ thống Quản Lý Văn Bản Hành Chính đã được được hầu hết các yêu cầu của ban lãnh đạo và các cán bộ trong cơ quan bao gồm việc quản lý văn bản cũng như quản lý phòng ban và quản lý nhân viên trong cơ quan.

- Quản lý văn bản
 - Đã có thể upload được văn bản
 - Xem được nội dung văn bản trên nền web
 - Xem chi tiết văn bản cũng như trạng thái văn bản
 - Phê duyệt và phân loại văn bản
 - Xóa được văn bản
 - Tìm kiếm văn bản
- Quản lý phòng ban
 - Thêm được phòng ban mới
 - Xem danh sách các phòng ban
 - Xóa được phòng ban
 - Xem danh sách các cán bộ trong phòng ban
 - Thêm được mới 1 cán bộ vào trong phòng ban
- Quản lý tổng quát
 - Xem tổng các cán bộ trong hệ thống
 - Xem tổng các văn bản xử lý
 - Xem tổng các văn bản đã xử lý

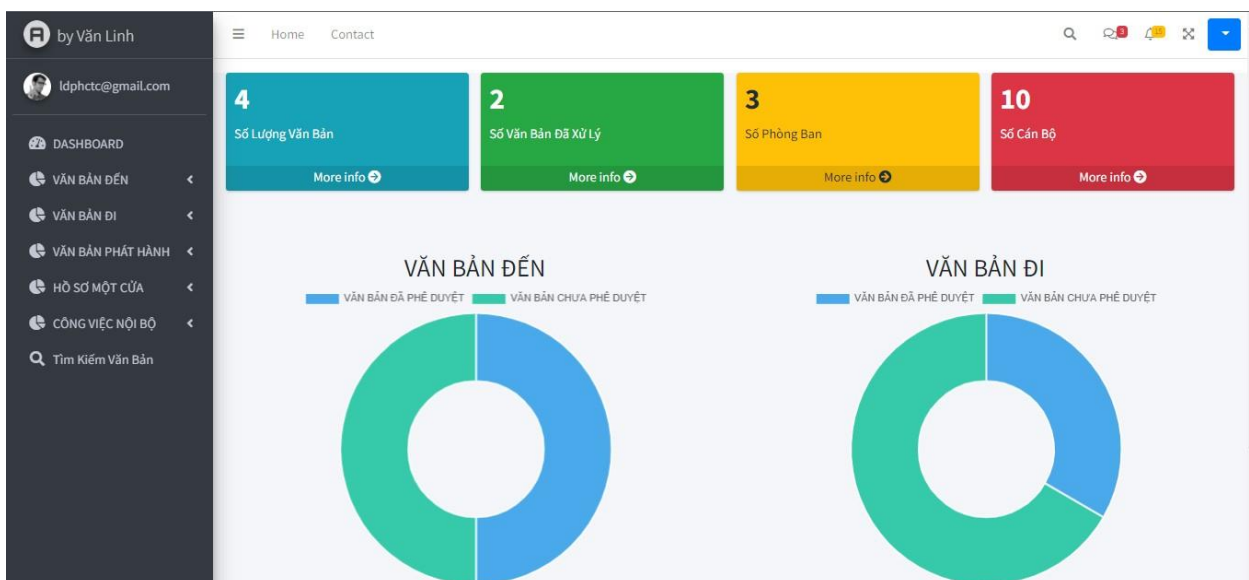
Các thông số thống kê về hệ thống đã xây dựng:

Thông số hệ thống:

Thông tin	Thống kê
Số lượng bảng trong cơ sở dữ liệu	8
Dung lượng mã nguồn Backend	2 MB
Dung lượng mã nguồn Frontend	79.2 MB
Số lượng file JavaScripts	80 file

4.3.2 Minh họa các chức năng chính

a) Khi người dùng đăng nhập với quyền là cán bộ



Hình 4.3.1 Giao diện trang home của hệ thống quản lý văn bản

Văn Bản Đến

Home / Văn bản đến / Thêm Văn bản đến

Nhập thông tin văn bản

Tên văn bản: Văn bản đề biết

Người gửi đến: Nguyễn Văn N

Nơi gửi đến: Hà Nội

Ngày Ban Hành: 02/02/2022 2:13 PM

Mã cơ quan: 11002

Số và kí hiệu: 00223

Độ Mật: Mật

Độ Khẩn: Khẩn

Loại văn bản: Văn bản đề biết

File input

Văn bản đề biết.docx

Browse Upload

Thêm Văn Bản

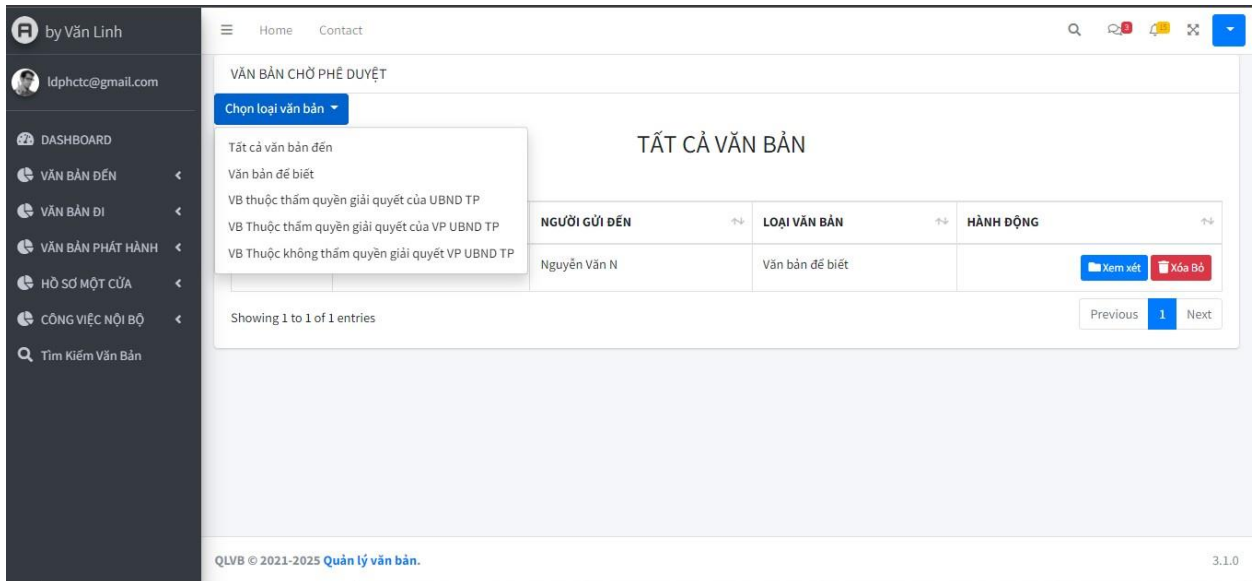
Hình 4.3.2 Giao diện trang thêm văn bản đến

THÔNG BÁO

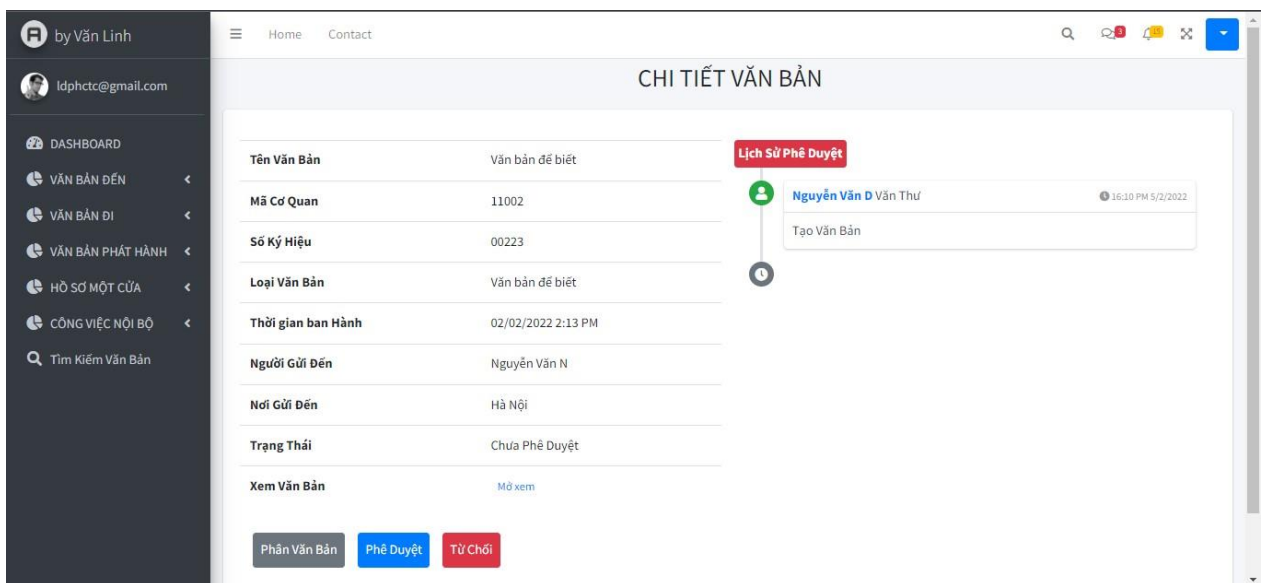
thêm văn bản thành công

Save

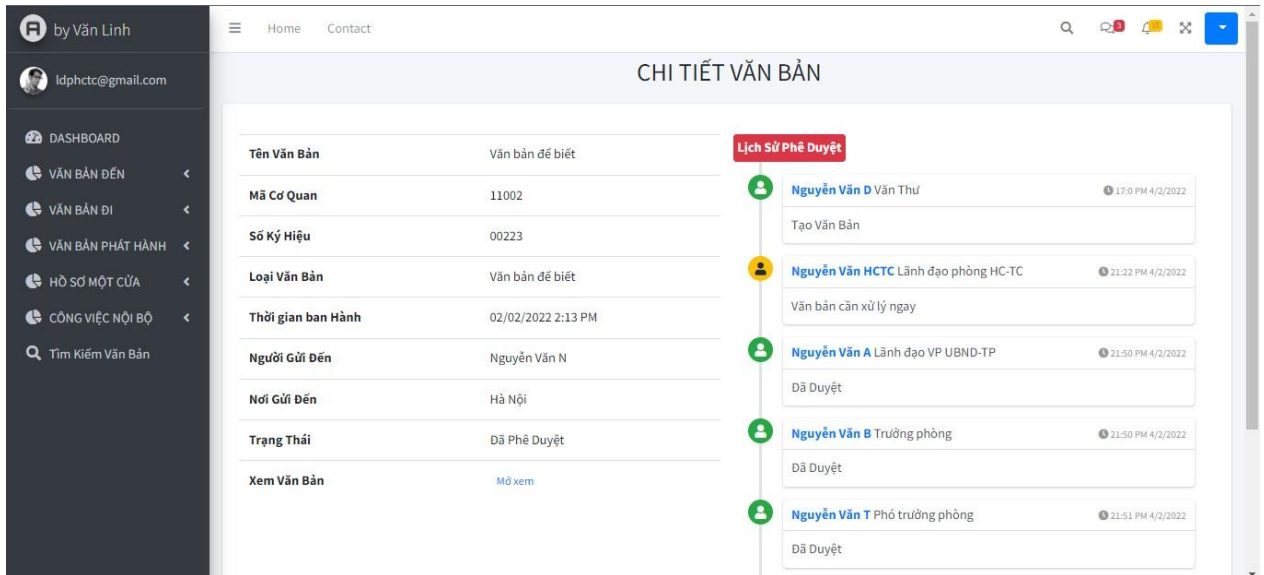
Hình 4.3.3 Giao diện khi thêm thành công vào trong hệ thống



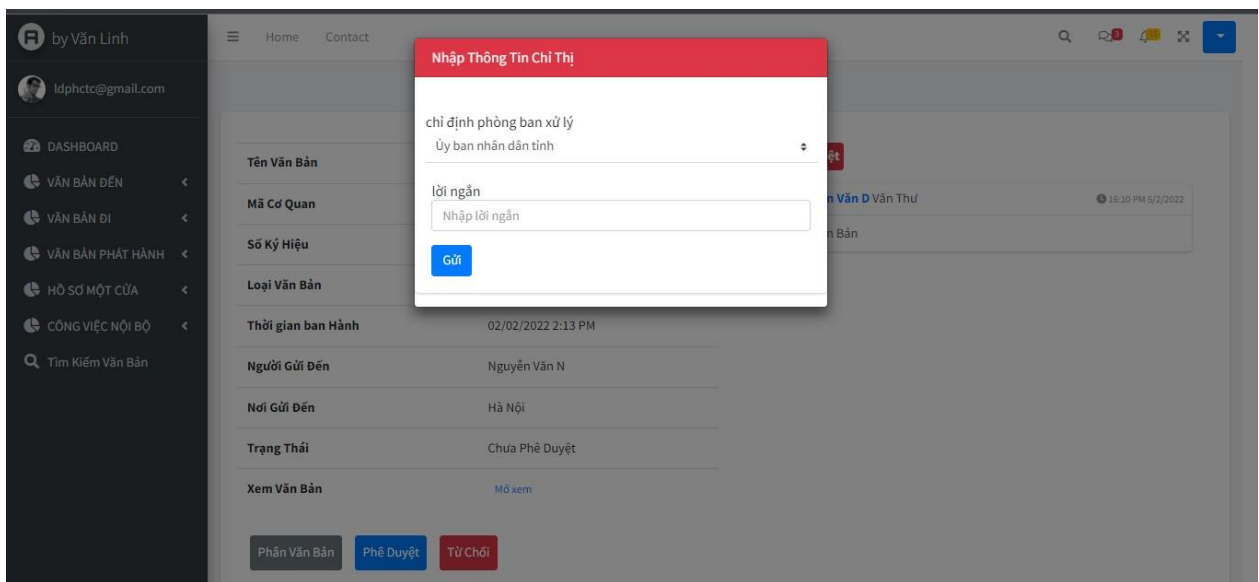
Hình 4.3.4 Giao diện hiện thị tất cả các văn bản đến chưa được duyệt của hệ thống



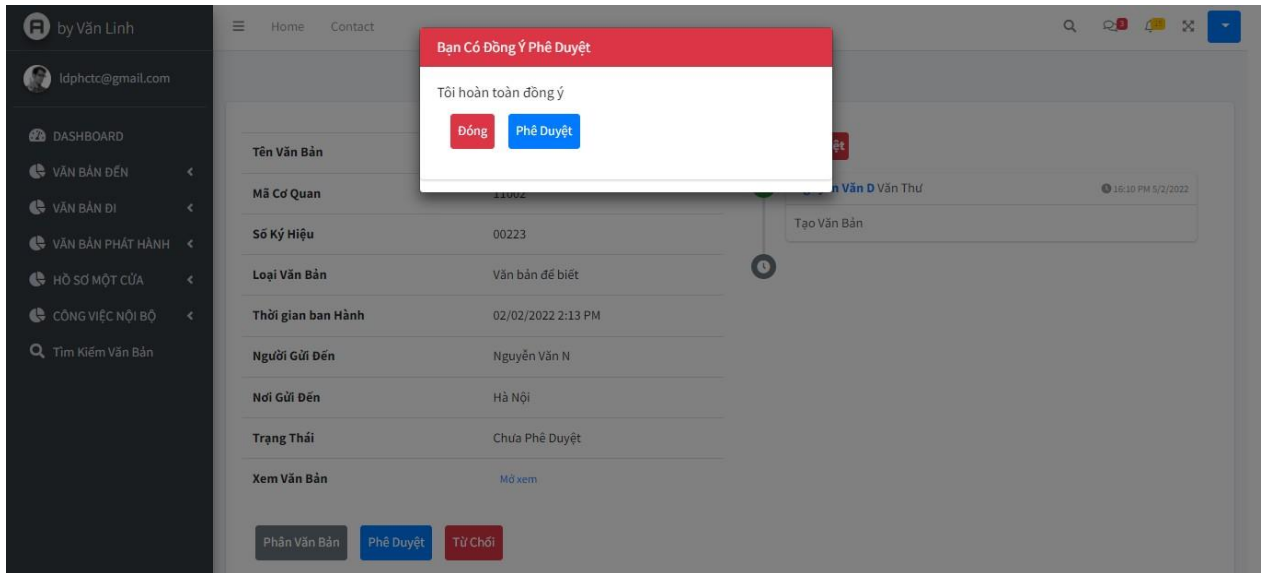
Hình 4.3.5 Giao diện xem chi tiết văn bản đến của hệ thống



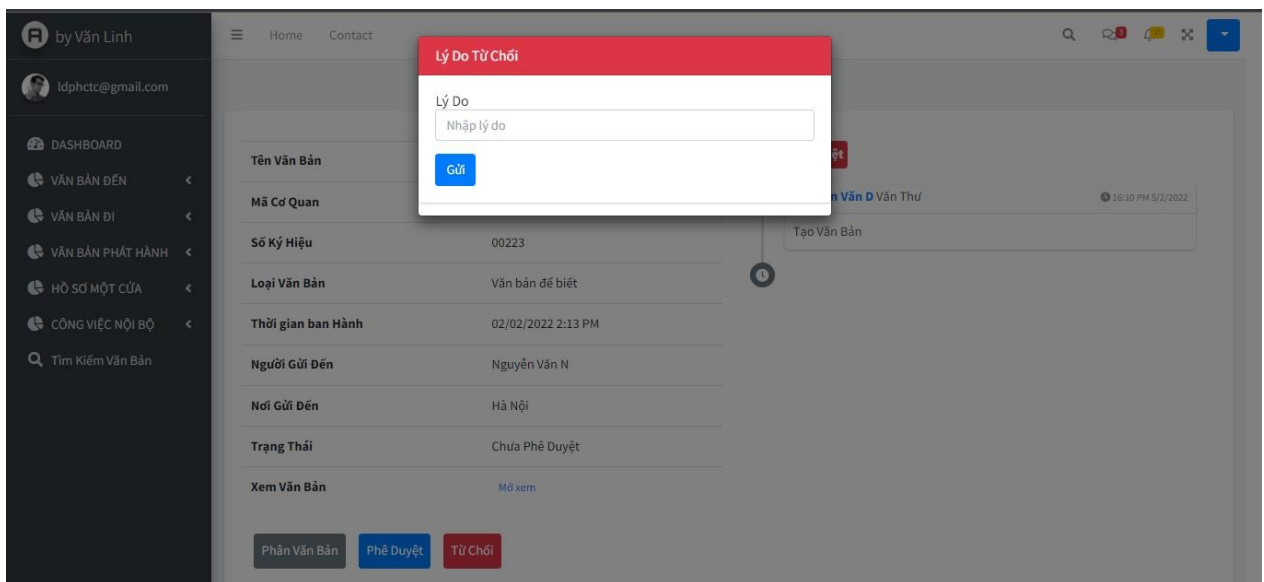
Hình 4.3.6 Giao diện xem chi tiết văn bản đến và lịch sử khi đã hoàn thành phê duyệt



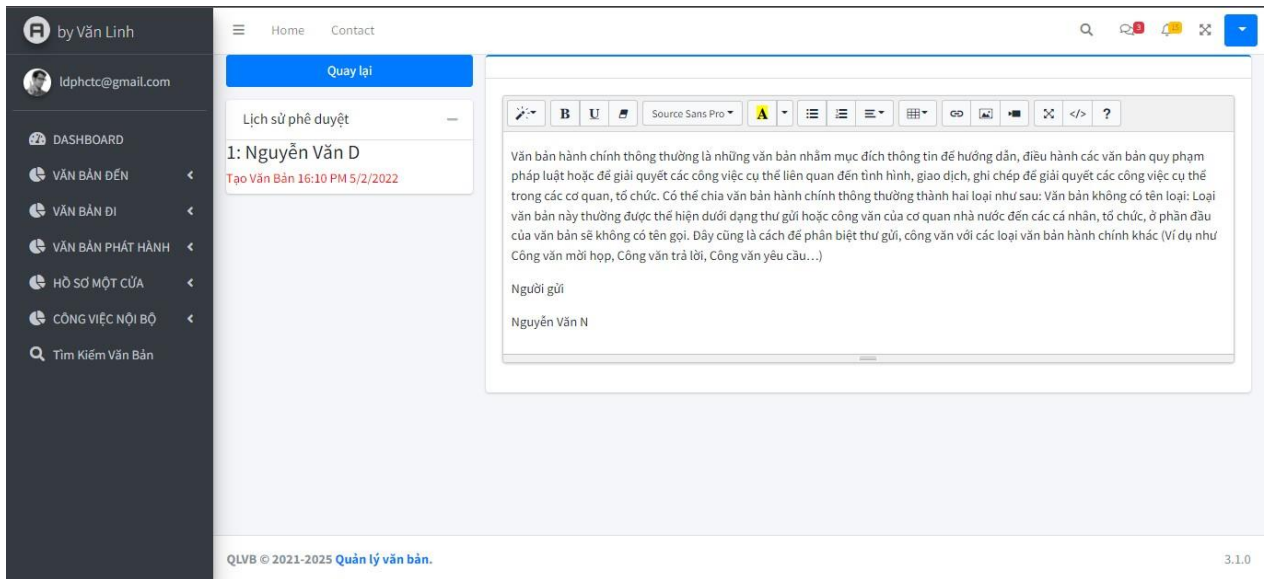
Hình 4.3.7 Giao diện khi lãnh đạo muốn chỉ thị một phòng ban xử lý



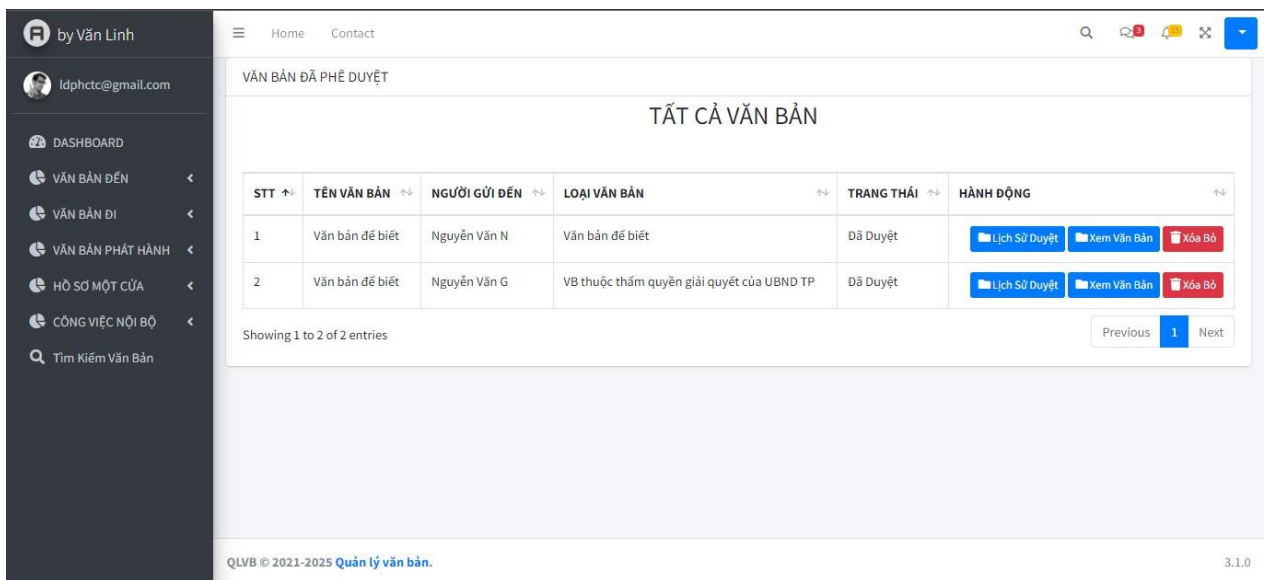
Hình 4.3.8 Giao diện khi cán bộ muốn duyệt văn bản



Hình 4.3.9 Giao diện khi cán bộ muốn từ chối văn bản



Hình 4.3.10 Giao diện khi xem nội dung của văn bản



Hình 4.3.11 Giao diện các văn bản đã phê duyệt

by Văn Linh

ldphctc@gmail.com

DASHBOARD

VĂN BẢN ĐẾN

VĂN BẢN ĐI

VĂN BẢN PHÁT HÀNH

HỒ SƠ MỘT CỬA

CÔNG VIỆC NỘI BỘ

Tìm Kiếm Văn Bản

Home

Contact

🔍

🗨️

📧

⌘

⌵

NHẬP THÔNG TIN TÌM KIẾM

Tên văn bản:

Nơi gửi đến:

Mã cơ quan:

Số và kí hiệu:

Từ khóa tìm kiếm:

Loại văn bản:

Search

TẤT CẢ VĂN BẢN

STT	TÊN VĂN BẢN	NGƯỜI GỬI ĐẾN	LOẠI VĂN BẢN	TRẠNG THÁI	HÀNH ĐỘNG
1	Văn bản đề biết	Nguyễn Văn N	Văn bản đề biết	Đã Duyệt	<div>Xem Văn BảnXóa Bỏ</div>
2	Văn bản đề biết	Nguyễn Văn G	VB thuộc thẩm quyền giải quyết của UBND TP	Đã Duyệt	<div>Xem Văn BảnXóa Bỏ</div>

Hình 4.3.12 Giao diện tìm kiếm tất cả các văn bản có trong hệ thống

by Văn Linh

ldphctc@gmail.com

DASHBOARD

VĂN BẢN ĐẾN

VĂN BẢN ĐI

VĂN BẢN PHÁT HÀNH

HỒ SƠ MỘT CỬA

CÔNG VIỆC NỘI BỘ

Tìm Kiếm Văn Bản

Home

Contact

thông tin

Nguyễn Văn A

Chức Vụ: Lãnh đạo VP UBND-TP

Phòng: Ủy ban nhân dân tỉnh

Email: ldp1@gmail.com

View Profile

thông tin

Nguyễn Văn B

Chức Vụ: Trưởng phòng

Phòng: Ủy ban nhân dân tỉnh

Email: tpt1@gmail.com

View Profile

thông tin

Nguyễn Văn T

Chức Vụ: Phó trưởng phòng

Phòng: Ủy ban nhân dân tỉnh

Email: ptp1@gmail.com

View Profile

thông tin

Nguyễn Văn C

Chức Vụ: Chuyên viên

Phòng: Ủy ban nhân dân tỉnh

Email: cvp1@gmail.com

View Profile

thông tin

Nguyễn Văn D

Chức Vụ: Văn Thư

Phòng: Ủy ban nhân dân tỉnh

Email: vtp1@gmail.com

View Profile

thông tin

Nguyễn Văn A

Chức Vụ: Chánh văn phòng

Phòng: Phòng HCHN

Email: cvpp2@gmail.com

View Profile

thông tin

Nguyễn Văn T

Chức Vụ: Văn Thư

Phòng: Ủy ban nhân dân tỉnh

Email: vtp1@gmail.com

View Profile

thông tin

Nguyễn Văn HCTC

Chức Vụ: Chuyên viên

Phòng: Ủy ban nhân dân tỉnh

Email: hctc1@gmail.com

View Profile

thông tin

Nguyễn Văn A

Chức Vụ: Phó trưởng phòng

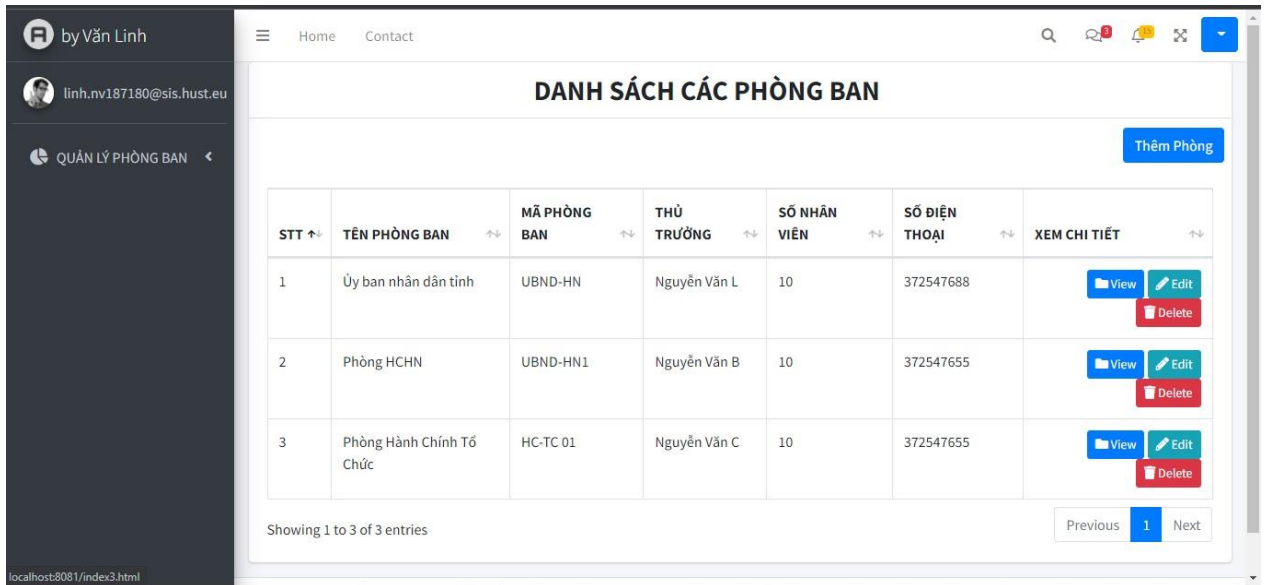
Phòng: Ủy ban nhân dân tỉnh

Email: ptp1@gmail.com

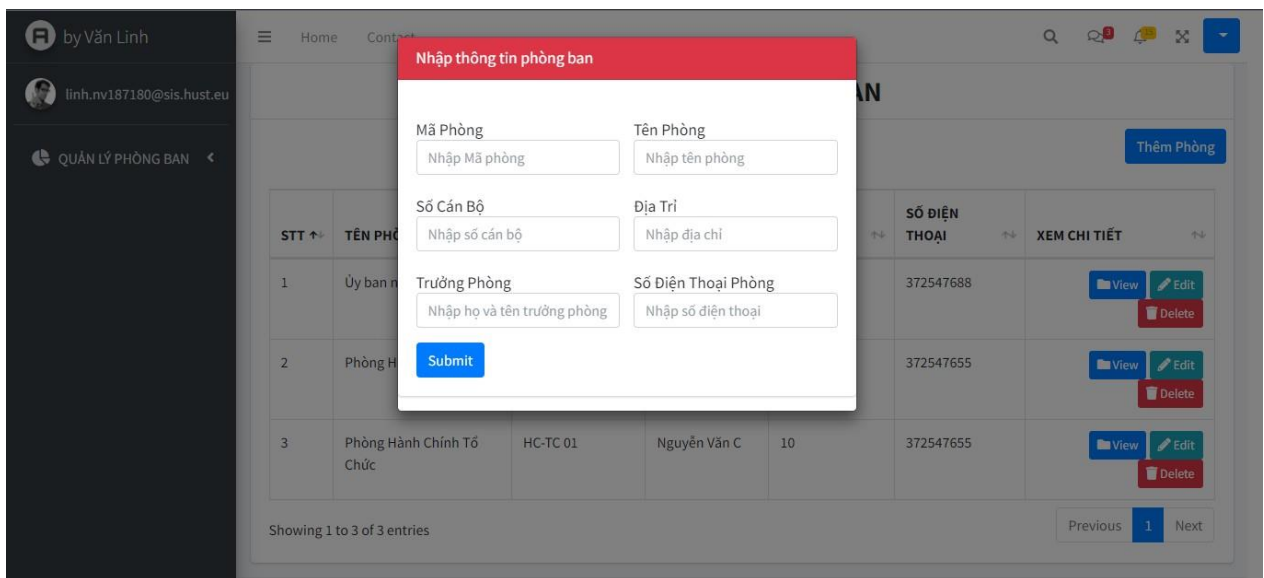
View Profile

Hình 4.3.13 Giao diện hiển thị danh sách các cán bộ

b) Khi đăng nhập với quyền là admin



Hình 4.3.14 Giao diện hiển thị danh sách các phòng ban



Hình 4.3.15 Giao diện nhập thông tin phòng ban mới

Thông Tin Cán Bộ

Họ Và Tên:

Địa Trĩ:

Email:

Mật khẩu:

Ngày Sinh:

Giới Tính:

Số Điện Thoại:

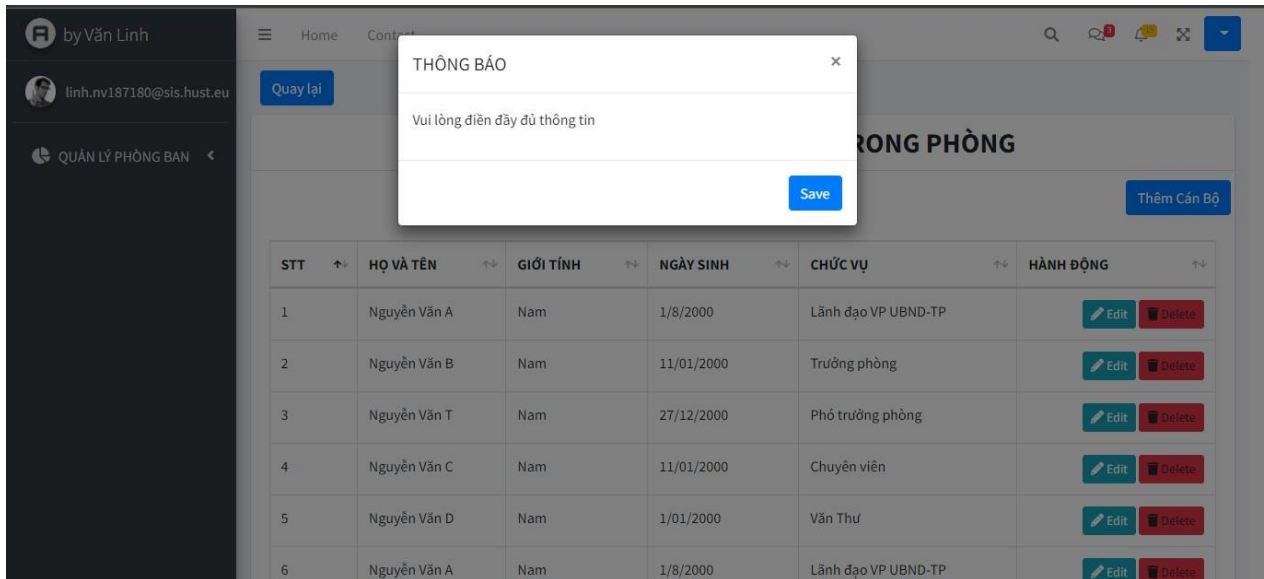
Chức Vụ:

Hình 4.3.15 Giao diện nhập thông tin cán bộ trong phòng ban

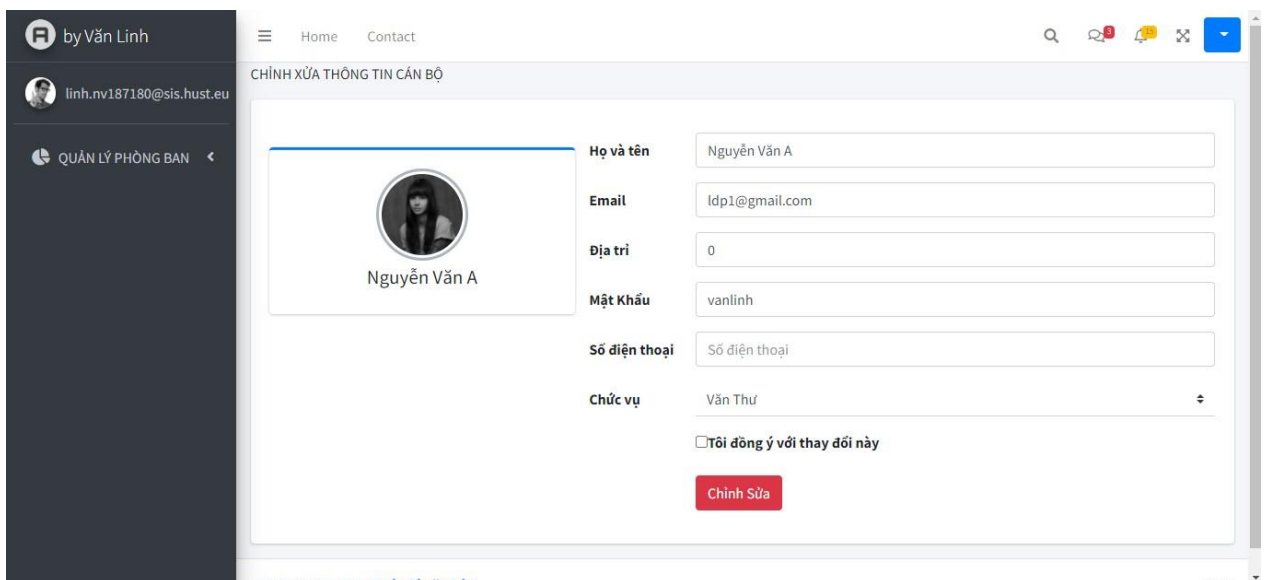
DANH SÁCH CÁC CÁN BỘ TRONG PHÒNG

STT	HỌ VÀ TÊN	GIỚI TÍNH	NGÀY SINH	CHỨC VỤ	HÀNH ĐỘNG
1	Nguyễn Văn A	Nam	1/8/2000	Lãnh đạo VP UBND-TP	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
2	Nguyễn Văn B	Nam	11/01/2000	Trưởng phòng	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
3	Nguyễn Văn T	Nam	27/12/2000	Phó trưởng phòng	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
4	Nguyễn Văn C	Nam	11/01/2000	Chuyên viên	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
5	Nguyễn Văn D	Nam	1/01/2000	Văn Thư	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
6	Nguyễn Văn A	Nam	1/8/2000	Lãnh đạo VP UBND-TP	<input type="button" value="Edit"/> <input type="button" value="Delete"/>

Hình 4.3.15 Giao diện danh sách các cán bộ trong phòng ban



Hình 4.3.16 Giao diện thông báo lỗi khi không nhập đầy đủ thông tin



Hình 4.3.17 Giao diện chỉnh sửa thông tin

4.4. Kiểm thử

4.4.1 Kiểm thử chức năng thêm mới văn bản đến

Bảng kiểm thử chức năng thêm mới văn bản đến

	Mô tả	Bước thực hiện	Kết quả kì vọng	Kết quả thực tế	Kết luận
	Xem các trường thông tin cần nhập	Chọn chức năng thêm mới văn bản	Hiện đầy đủ các trường	Hiện đầy đủ các trường	Đạt
	Validate các trường thông tin	Nhập thông tin các trường	Các trường thông tin được validate đầy đủ, không được bỏ trống	Các trường thông tin bị điền sai thì đều có cảnh báo, và nếu bỏ trống thì không gửi được form	Đạt
	Gửi form thêm mới văn bản	Sau khi điền thông tin, nhấn thêm mới văn bản đến	Văn bản được thêm vào bảng các văn bản đến, trạng thái là tiếp nhận	Văn bản được thêm vào bảng các văn bản đến, trạng thái là tiếp nhận	Đạt
	Quyền xem văn bản được tạo mới		Nhân viên không được xem thông tin của văn bản mới tiếp nhận. Sau khi được lãnh đạo chỉ định thì nhân viên mới được xem.	Nhân viên không được xem thông tin của văn bản mới tiếp nhận. Sau khi được lãnh đạo chỉ định thì nhân viên mới được xem.	Đạt

Bảng kiểm thử chức năng thêm mới 1 phòng ban trong hệ thống

STT	Mô tả	Bước thực hiện	Kết quả kì vọng	Kết quả thực tế	Kết luận
1	Xem các trường thông tin cần nhập	Chọn chức năng thêm mới văn bản	Hiện đầy đủ các trường	Hiện đầy đủ các trường	Đạt
2	Validate các trường thông tin	Nhập thông tin các trường	Các trường thông tin được validate đầy đủ, không được bỏ trống	Các trường thông tin bị điền sai thì đều có cảnh báo, và nếu bỏ trống thì không gửi được form	Đạt
3	Kiểm tra xem email đã được đăng ký trước đó chưa	Nhập vào 1 email trùng với Email đã tồn tại trong hệ thống	Bão lỗi ra cho người dùng là Email này đã được đăng ký trước rồi	Đã có lỗi log ra cho người dùng biết	Đạt
4	Kiểm tra xem chức vụ này có trong phòng ban này hay chưa	Nhập vào 1 chức vụ đã tồn tại trong phòng ban lọc ra báo lỗi	Bão lỗi ra cho người dùng là chức vụ này đã được có trong phòng ban rồi	Đã có lỗi báo cho admin biết	Đạt

4.5. Triển khai

Hệ thống được triển khai thử nghiệm trên môi trường localhost, với thông số Server thử nghiệm được yêu cầu như bảng sau

Bảng thông số cấu hình Server

Thông số	Chi tiết
RAM	8GB
Hệ điều hành	Window 10
Ổ cứng	SSD 128GB
CPU	Inter Core i7 6300U

Chương 5 Kết luận và hướng phát triển

4. 5. 1 Kết luận

Hệ thống sau khi hoàn thiện sẽ cung cấp các chức năng hỗ trợ các cán bộ và lãnh đạo có thể xử lý và phê duyệt văn bản một cách nhanh chóng từ đó giúp giảm thiểu thời gian hơn nữa còn thống kê được số lượng văn bản có trong hệ thống và số lượng văn bản đã xử lý giúp lãnh đạo có thể điều chỉnh được tiến độ xử lý văn bản của các cán bộ trong phòng ban. Còn có chức năng giúp người dùng có thể tìm kiếm văn bản một cách nhanh chóng.

Thứ hai là hệ thống có thể quản lý được số phòng ban và số cán bộ có trong phòng và có thể thao tác một số chức năng như thêm xóa xóa .

Do các kỹ năng của em còn hạn chế, trong quá trình làm Project 3 có gặp nhiều khó khăn và sai sót. Với sự hướng dẫn của PGS .TS . Nguyễn Khanh Văn cùng các anh chị trong công ty VKX đã giúp em hoàn thiện Project 3 này .Từ đó em tích lũy được rất nhiều kiến thức chuyên môn như khả năng code, cách thiết kế hệ thống sao cho phù hợp đảm bảo hệ thống phát triển và hoạt động tốt, giúp em nắm vững được các cơ sở lý thuyết của từng kỹ thuật chuyên môn. Giúp em hiểu thế nào là một hệ thống tốt, phù hợp với nhu cầu hiện tại của mọi người.

Một số hạn chế của hệ thống và những tính năng em chưa xây được và chưa hoàn thiện là: hệ thống sẽ phát triển thêm các tính năng còn thiếu như xử lý công việc nội bộ trong công ty, quản lý giấy mời,..

5. 5.2 Hướng phát triển

Hệ thống em phát triển còn nhiều hạn chế, để hoàn thiện hơn hệ thống cần phát triển thêm một số chức năng như:

- Quản lý công việc nội bộ
- Quản lý hồ sơ một cửa
- Có thể xem bất kỳ tài liệu word nào trên nền web một cách bảo mật nhất
- Hệ thống có thể hiển thị thông báo về cho số điện thoại của lãnh đạo khi có văn thư hỏa tốc cần xử lý ngay

6. Tài liệu tham khảo

[1] <http://expressjs.com/en/starter/installing.html>

[2] <https://nodejs.vn/category/4/h%E1%BB%8Fi-%C4%91%C3%A1p>

[3] <https://docs.npmjs.com/about-npm>