**ĐẠI HỌC DUY TÂN**

**KHOA CÔNG NGHỆ THÔNG TIN**

**--------------🙖🙐✰🙖🙐-------------**

****

**Tên đề tài:**

**PHÁT TRIỂN HỆ THỐNG MẠNG XÃ HỘI LINKVERSE**

**TÍCH HỢP THÔNG MINH VÀ**

**CÁ NHÂN HÓA VỚI AI**

**∙•🙞🟏🙜•∙**

**TÀI LIỆU PRODUCT BACKLOG**

GVHD: ThS. Lương Thị Thu Phương

Thành viên:

1. Đặng Ngọc Dương – 0649
2. Trần Quốc Nguyên – 0875
3. Dương Hà Tuấn – 2658
4. Nguyễn Thị Thu Thủy – 1171
5. Nguyễn Trần Khánh Vinh – 8473

*Đà Nẵng, 2025*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **THÔNG TIN DỰ ÁN** | | | | | | |
| **Dự án viết tắt** | LINKVERSE | | | | | |
| **Tên dự án** | Phát Triển Hệ Thống Mạng Xã Hội LinkVerse Tích Hợp Gợi Ý Thông Minh Và Cá Nhân Hóa Với AI | | | | | |
| **Thời gian bắt đầu** | 17/03/2025 | **Thời gian kết thúc** | 23/05/2025 | | |
| **Lead Institution** | Khoa Công Nghệ Thông Tin, Đại Học Duy Tân | | | | | |
| **Giáo viên hướng dẫn** | ThS. Lương Thị Thu Phương  Email: ltphuongdtu@gmail.com  Phone: 033 7571 631 | | | | | |
| **Chủ sở hữu dự án & Chi tiết liên hệ** | ThS. Lương Thị Thu Phương  Email: ltphuongdtu@gmail.com  Phone: 033 7571 631 | | | | | |
| **Đối tác** | Duy Tan University | | | | | |
| **Quản lý dự án &Scrum Master** | Đặng Ngọc Dương | duongng.dn@gmail.com | | 0372682871 |
| **Thành Viên Nhóm** | Trần Quốc Nguyên | tranquocnguyeeen@gmail.com | | 0384003168 |
| Dương Hà Tuấn | hatuan1423@gmail.com | | 0901942192 |
| Nguyễn Thị Thu Thủy | nthuthuy99999@gmail.com | | 0789137207 |
| Nguyễn Trần Khánh Vinh | vinhtran0206003@gmail.com | | 0364045426 |

**THÔNG TIN TÀI LIỆU**

|  |  |
| --- | --- |
| **Tên dự án** | Phát Triển Hệ Thống Mạng Xã Hội LinkVerse Tích Hợp Gợi Ý Thông Minh Và Cá Nhân Hóa Với AI |
| **Tiêu đề tài liệu** | Product-Backlog Document |
| **Người thực hiện** | Nguyễn Trần Khánh Vinh |

**LỊCH SỬ CHỈNH CẬP NHẬT TÀI LIỆU**

|  |  |  |  |
| --- | --- | --- | --- |
| **Phiên bản** | **Người chỉnh sửa** | **Ngày** | **Ghi chú** |
| 1.0 | Nguyễn Trần Khánh Vinh | 17/03/2025 | Tạo tài liệu |

**PHÊ DUYỆT TÀI LIỆU**

|  |  |  |  |
| --- | --- | --- | --- |
| **Người hướng dẫn** | Lương Thị Thu Phương | **Chứ ký** |  |
| **Ngày** | ………/03/2025 |
| **Chủ sở hữu** | Lương Thị Thu Phương | **Chứ ký** |  |
| **Ngày** | ………/03/2025 |
| **Scrum Master** | Đặng Ngọc Dương | **Chứ ký** |  |
| **Ngày** | ………/03/2025 |
| **Thành viên** | Trần Quốc Nguyên | **Chứ ký** |  |
| **Ngày** | ………/03/2025 |
| Dương Hà Tuấn | **Chứ ký** |  |
| **Ngày** | ………/03/2025 |
| Nguyễn Thị Thu Thủy | **Chứ ký** |  |
| **Ngày** | ………/03/2025 |
| Nguyễn Trần Khánh Vinh | **Chứ ký** |  |
| **Ngày** | ………/03/2025 |

# 1. GIỚI THIỆU

Đây là tài liệu Product Backlog trong quy trình Scrum. Nó bao gồm danh sách tất cả các tính năng được ưu tiên và các mô tả cho từng tính năng của sản phẩm cần phải hoàn thành trong dự án. Thông thường, Product Backlog chứa các User Story, nhưng đôi khi cũng có thể có các yêu cầu chức năng, yêu cầu phi chức năng, bugs và một số vấn đề khác.

Product Backlog trong Scrum cũng chấp nhận việc thay đổi và tăng thêm trong quá trình dự án tùy theo sự thay đổi yêu cầu của Product Owner.

## 1.1. Mục đích.

Tài liệu này thể hiện các yêu cầu cấp cao trong quan điểm của người dùng cuối. Những yêu cầu của người dùng sẽ được chia thành nhiều nhiệm vụ để phân công cho nhóm phát triển. Ngoài ra, tài liệu này cũng có các tiêu chuẩn chấp nhận, hữu ích cho người kiểm tra để tạo kế hoạch kiểm tra và kiểm thử.

## 1.2. Phạm vi.

* Các user story trong dự án
* Các yêu cầu chức năng và phi chức năng
* Sơ đồ Use case
* Biểu đồ hoạt động

## 1.3. Tài liệu tham khảo.

Bảng 1: Tài liệu tham khảo

|  |  |  |
| --- | --- | --- |
| **No** | **Reference** | **Note** |
| 1 | <http://agilebench.com/blog/the-product-backlog-for-agile-teams> | Cách tạo Product Backlog |
| 2 | <http://www.mountaingoatsoftware.com/agile/scrum/product-backlog/example/> | Ví dụ về Product Backlog |
| 3 | KLTN-SE.01.Product Proposal.docx | Proposal Document |

## 2.1. Mô tả Product Backlog

### 2.1.1 PB01 - Đăng nhập

|  |  |
| --- | --- |
| **ID** | PB01 |
| **Tiêu đề** | Đăng nhập |
| **Tác nhân** | Người dùng (Admin, User) |
| **Mô tả** | Người dùng đăng nhập vào hệ thống |
| **Các bước thực hiện** | 1. Thiết kế giao diện đăng nhập trên trang đăng nhập  2. Xử lý logic đăng nhập và gọi API đăng nhập.  3. Hiển thị thông báo sai Password hoặc Email hoặc thông báo đã gửi mã OTP.  4. Xử lý hệ thống gửi mã OTP bảo mật cho người dùng.  5. Hiển thị thông báo đăng nhập thành công hoặc không thành công. |
| **Điều kiện trước** | 1. Người dùng truy cập vào trang chủ hệ thống.  2. Chọn chức năng đăng nhập (Nhấn chọn [Đăng Nhập]).  3. Nhập Email/Username. (Nhập vào [Username/Email] TextBox).  4. Nhập Mật khẩu (Nhập vào [Password] TextBox).  5. Gửi yêu cầu đăng nhập (Nhấn chọn [Đăng Nhập] Button).  6. Kiểm tra đăng nhập:     6.1. Kiểm tra Username. Nếu nhập sai Username, hệ thống hiển thị thông báo “Nhập sai Username hoặc Password”.     6.2. Kiểm tra Mật khẩu. Nếu nhập sai Mật khẩu, hệ thống hiển thị thông báo “Nhập sai Username hoặc Password”.     6.3. Nếu nhập đúng Email/Username và Password, mã OTP sẽ được gửi vào gmail, người dùng nhập OTP, hệ thống hiển thị thông báo và chuyển sang trang thông tin người dùng. |
| **Điều kiện sau** | Trang chủ tải thành công và người dùng có thể nhìn thấy button Đăng nhập của hệ thống |
| **Công nghệ sử dụng** | Frontend: ReactJS (giao diện tìm kiếm, API)  * Backend: Spring Boot Microservices (Xử lý đăng nhập, gửi OTP, truy vấn cơ sở dữ liệu). * Database: MySQL (Lưu dữ liệu tài khoản người dùng) * API: RESTful API (gửi và nhận dữ liệu từ backend) |
| **Độ phức tạp** | Cao |
| **Điều kiện ràng buộc** | 1. Khi nhập sai password quá nhiều lần sẽ bị tạm khóa. 2. Nếu tài khoản bị vô hiệu hóa, cần xác minh danh tính trước khi đăng nhập lại. |

### 2.1.2 PB02 - Đăng xuất

|  |  |
| --- | --- |
| **ID** | PB02 |
| **Tiêu đề** | Đăng xuất |
| **Tác nhân** | Người dùng (Admin, User) |
| **Mô tả** | Người dùng đăng xuất khỏi tài khoản đang đăng nhập. |
| **Các bước thực hiện** | 1. Thiết kế giao diện dăng xuất trên trang chủ.  2. Hiển thị thông báo xác nhận đăng xuất.  3. Xử lý logic đăng xuất và gọi API đăng xuất.  3. Hiển thị thông báo đăng xuất thành công và trở về trang chủ chưa đăng nhập. |
| **Điều kiện trước** | 1. Người dùng đang đăng nhập vào hệ thống.  2. Chọn chức năng đăng xuất (Nhấn chọn [Đăng Xuất]).  3. Hệ thống hiển thị thông báo xác nhận    3.1. Xác nhận đăng xuất, hệ thống hiển thị thông báo đăng xuất thành công và đưa người dùng về trang chủ chưa đăng nhập.     6.2. Nếu không người dùng giữ nguyên giao diện và tác vụ đang thực hiện. |
| **Điều kiện sau** | Hiển thị trang chủ đã đăng nhập và có thể hiển thị button Đăng xuất |
| **Công nghệ sử dụng** | Frontend: ReactJS (giao diện tìm kiếm, API)  * Backend: Spring Boot Microservices (Xử lý đăng nhập, gửi OTP, truy vấn cơ sở dữ liệu). * Database: MySQL (Lưu dữ liệu tài khoản người dùng) * API: RESTful API (gửi và nhận dữ liệu từ backend) |
| **Độ phức tạp** | Cao |
| **Điều kiện ràng buộc** | 1. Khi nhập sai password quá nhiều lần sẽ bị tạm khóa. 2. Nếu tài khoản bị vô hiệu hóa, cần xác minh danh tính trước khi đăng nhập lại.  3.Tất cả các hoạt động hoặc thao tác chưa được lưu trước khi đăng xuất sẽ không được ghi nhận. |

### 2.1.3 PB03 – Quản lý trang cá nhân

|  |  |
| --- | --- |
| **ID** | PB03 |
| **Tiêu đề** | Quản lý trang cá nhân |
| **Tác nhân** | Người dùng (Admin, User) |
| **Mô tả** | Người dùng thay đổi thông tin hiển thị trên trang cá nhân của bản thân. |
| **Các bước thực hiện** | 1. Thiết kế giao diện Quản lý trang cá nhân.  2. Xử lý logic cập nhật các thông tin về trang cá nhân như: tên, ảnh dại diện, bài viết hiển thị, tiểu sử, … và gọi API update thông tin.  3. Hiển thị thông tin đã cập nhật |
| **Điều kiện trước** | 1. Người dùng đang đăng nhập vào hệ thống.  2. Truy cập vào Trang cá nhân, chọn Chỉnh sửa trang cá nhân.  3. Hệ thống hiển thị giao diện chỉnh sửa thông tin:    3.1. Hiển thị thông tin chỉnh sửa hợp lệ, hiển thị thông báo thành công và hiển thị thông tin vừa update.     3.2. Hiển thị thông báo cập nhật không thành công |
| **Điều kiện sau** | Trang cá nhân được tải thành công và người dùng có thể nhìn thấy chức năng Chỉnh sửa trang cá nhân. |
| **Công nghệ sử dụng** | Frontend: ReactJS (giao diện tìm kiếm, API)  * Backend: Spring Boot Microservices (Xử lý đăng nhập, gửi OTP, truy vấn cơ sở dữ liệu). * Database: MySQL (Lưu dữ liệu tài khoản người dùng) * API: RESTful API (gửi và nhận dữ liệu từ backend) |
| **Độ phức tạp** | Cao |
| **Điều kiện ràng buộc** | 1. Giới hạn 200 ký tự về thông tin tiểu sử. |

### 2.1.4 PB04 – Quản lý tài khoản

|  |  |
| --- | --- |
| **ID** | PB04 |
| **Tiêu đề** | Quản lý tài khoản |
| **Tác nhân** | Người dùng (Admin, User) |
| **Mô tả** | Người dùng đăng xuất khỏi tài khoản đang đăng nhập. |
| **Các bước thực hiện** | 1. Thiết kế giao diện Quản lý tài khoản  2. Xử lý logic cập nhật các thông tin về tài khoản cá nhân như: Thay đổi thông tin tài khoản (email, số điện thoại, …), Kiểm tra hoạt động đăng nhập, Xem lịch sử hoạt động, tạm ngưng hoặc xóa tài khoản, và gọi API phù hợp cho từng tác vụ khác nhau của hệ thống.  3. Hiển thị thông tin và thực hiện các thay đổi theo yêu cầu của người dùng. |
| **Điều kiện trước** | 1. Người dùng đang đăng nhập vào hệ thống.  2. Truy cập vào Trang cá nhân, chọn chức năng Quản lý tài khoản  3. Hệ thống hiển thị thông tin của tài khoản và các chức năng có thể thực hiện:    3.1. Nếu người dùng thực hiện các thay đổi về thông tin tài khoản, hệ thống thông báo thành công hoặc thất bại và hiển thị thông tin đã được sửa đổi.     6.2. Nếu người dùng thực hiện các tác vụ khác như xem lịch sử, tạm ngưng hoặc xóa tài khoản, hệ thống hiển thị thông tin đã xử lý cần thiết cho người dùng. |
| **Điều kiện sau** | Trang Quản lý tài khoản tải thành công và có thể nhìn thấy các button để sử dụng các chức năng bên trong Quản lý tài khoản. |
| **Công nghệ sử dụng** | Frontend: ReactJS (giao diện tìm kiếm, API)  * Backend: Spring Boot Microservices (Xử lý đăng nhập, gửi OTP, truy vấn cơ sở dữ liệu). * Database: MySQL (Lưu dữ liệu tài khoản người dùng) * API: RESTful API (gửi và nhận dữ liệu từ backend) |
| **Độ phức tạp** | Cao |
| **Điều kiện ràng buộc** | 1.Email chỉ sử dụng cho duy nhất 1 tài khoản.  2.Số điện thoại có thể sử dụng tối đa cho 3 tài khoản.  3. Khi tạm ngưng tài khoản, đăng nhập lại sẽ phải thực hiện xác thực người dùng. |

### 2.1.5 PB05 – Xem bảng tin – Stories

|  |  |
| --- | --- |
| **ID** | PB05 |
| **Tiêu đề** | Xem bảng tin – Stories |
| **Tác nhân** | Người dùng (Admin, User) |
| **Mô tả** | Người dùng có thể xem bảng tin chưa các bài viết và story (short video) trên trang chủ trang cá nhân của mình. |
| **Các bước thực hiện** | 1. Thiết kế giao diện hiển thị Xem bảng tin – Stories.  2. Xử lý logic và gọi API lấy các bài viết và story từ hệ thống.  3. Hiển thị bảng tin. |
| **Điều kiện trước** | 1. Người dùng đang đăng nhập vào hệ thống.  2. Truy cập vào trang chủ Trang cá nhân, chọn bài viết hoặc story muốn xem.  3. Hệ thống hiển thị giao diện Trang cá nhân hiển thị nội dung đầy đủ nội dung của bài viết và story |
| **Điều kiện sau** | Trang chủ được tải thành công và người dùng có thể thấy được các bài viết và story của các người dùng khác. |
| **Công nghệ sử dụng** | Frontend: ReactJS (giao diện tìm kiếm, API)  * Backend: Spring Boot Microservices (Xử lý đăng nhập, gửi OTP, truy vấn cơ sở dữ liệu). * Database: MySQL (Lưu dữ liệu tài khoản người dùng) * API: RESTful API (gửi và nhận dữ liệu từ backend) |
| **Độ phức tạp** | Cao |
| **Điều kiện ràng buộc** | Không |

### 2.1.6 PB06 – Đổi mật khẩu

|  |  |
| --- | --- |
| **ID** | PB06 |
| **Tiêu đề** | Đổi mật khẩu |
| **Tác nhân** | Người dùng (Admin, User) |
| **Mô tả** | Người dùng có thể thay đổi mật khẩu của tài khoản. |
| **Các bước thực hiện** | 1. Thiết kế giao diện Đổi mật khẩu.  2. Xử lý logic và gọi API đổi mật khẩu.  3. Hiển thị thông báo khi thành công hoặc không hợp lệ. |
| **Điều kiện trước** | 1. Người dùng đang đăng nhập vào hệ thống.  2. Truy cập vào Đổi mật khẩu, nhập thông tin mật khẩu cũ và mới.  3. Hệ thống hiển thị thông báo đổi mật khẩu thành công hoặc thông báo các vi phạm trong đổi mật khẩu. |
| **Điều kiện sau** | Trang cá nhân được tải thành công và người dùng có thể nhìn thấy button để sử dụng chức năng đổi mật khẩu. |
| **Công nghệ sử dụng** | Frontend: ReactJS (giao diện tìm kiếm, API)  * Backend: Spring Boot Microservices (Xử lý đăng nhập, gửi OTP, truy vấn cơ sở dữ liệu). * Database: MySQL (Lưu dữ liệu tài khoản người dùng) * API: RESTful API (gửi và nhận dữ liệu từ backend) |
| **Độ phức tạp** | Cao |
| **Điều kiện ràng buộc** | 1. Mật khẩu mới không được trùng với mật khẩu hiện tại hoặc các mật khẩu đã sử dụng trước đó (nếu áp dụng lịch sử mật khẩu).  2. Mật mới phải đạt đủ điều kiện chứa ít nhất 1 chữ hoa, 1 chữ thường, 1 ký tự số và 1 ký tự đặc biệt. |

### 2.1.7 PB07 – Nhắn tin (Chat)

|  |  |
| --- | --- |
| **ID** | PB07 |
| **Tiêu đề** | Nhắn tin (Chat) |
| **Tác nhân** | Người dùng (Admin, User) |
| **Mô tả** | Người dùng có nhắn tin giữa các người dùng với nhau trong hệ thống. |
| **Các bước thực hiện** | 1. Thiết kế giao diện Nhắn tin.  2. Xử lý logic và gọi API nhắn tin, ở đây sẽ là tin nhắn real time (người dùng và người nhận sẽ gửi và nhận tin nhắn tương xứng với thời gian thực).  3. Hiển thị thông báo khi có thông báo tin nhắn mới và hiển thị thời gian mà tin nhắn được gửi. |
| **Điều kiện trước** | 1. Người dùng đang đăng nhập vào hệ thống.  2. Truy cập vào Nhắn tin, chọn người dùng muốn gửi tin nhắn.  3. Hệ thống hiển thị thông khi có tin nhắn và hiển thị thời gian mà tin nhắn được gửi đi. |
| **Điều kiện sau** | Trang cá nhân hiển thị và hiển thị button cho chức năng nhắn tin (chat). |
| **Công nghệ sử dụng** | Frontend: ReactJS (giao diện tìm kiếm, API)  * Backend: Spring Boot Microservices (Xử lý đăng nhập, gửi OTP, truy vấn cơ sở dữ liệu). * Database: MySQL (Lưu dữ liệu tài khoản người dùng) * API: RESTful API (gửi và nhận dữ liệu từ backend) |
| **Độ phức tạp** | Cao |
| **Điều kiện ràng buộc** | Không |

### 2.1.8 PB08 – Xem thông báo

|  |  |
| --- | --- |
| **ID** | PB08 |
| **Tiêu đề** | Xem thông báo |
| **Tác nhân** | Người dùng (Admin, User) |
| **Mô tả** | Người dùng có thể xem thông báo về bài viết, tương tác, tin nhắn, … của tài khoản cá nhân. |
| **Các bước thực hiện** | 1. Thiết kế giao diện Thông báo.  2. Xử lý logic và gọi API thông báo của hệ thống  3. Hiển thị thông báo khi có tin nhắn, tương tác, bài viết mới hoặc có sự thay đổi. |
| **Điều kiện trước** | 1. Người dùng đang đăng nhập vào hệ thống.  2. Truy cập vào Trang cá nhân, nhấn xem thông báo.  3. Hệ thống hiển thị các thông báo về các thay đổi và các thông tin mới. |
| **Điều kiện sau** | Trang chủ đã được tải thành công và người dùng có thể nhìn thấy button để xem thông báo. |
| **Công nghệ sử dụng** | Frontend: ReactJS (giao diện tìm kiếm, API)  * Backend: Spring Boot Microservices (Xử lý đăng nhập, gửi OTP, truy vấn cơ sở dữ liệu). * Database: MySQL (Lưu dữ liệu tài khoản người dùng) * API: RESTful API (gửi và nhận dữ liệu từ backend) |
| **Độ phức tạp** | Cao |
| **Điều kiện ràng buộc** | Không |

### 2.1.9 PB09 – Quên mật khẩu

|  |  |
| --- | --- |
| **ID** | PB08 |
| **Tiêu đề** | Quên mật khẩu |
| **Tác nhân** | Người dùng (Admin, User) |
| **Mô tả** | Người dùng có thể tạo mật khẩu mới khi quên mật khẩu và không thể đăng nhập. |
| **Các bước thực hiện** | 1. Thiết kế giao diện Quên mật khẩu  2. Xử lý logic và gọi API thay đổi mật khẩu.  3. Hệ thống gọi API gửi liên kết qua email đến người dùng.  4. Thông báo đổi thành công hoặc thất bại. |
| **Điều kiện trước** | 1. Người dùng vào trang đăng nhập và chọn Quên mật khẩu  2. Người dùng truy cập email, thực hiện đặt lại mật khẩu.  3. Sau khi người dùng thực hiện đổi  3.1. Thành công: Hệ thống thông báo thành công và quay về trang đăng nhập  3.2. Thất bại: Hệ thống thông báo thất bại. |
| **Điều kiện sau** | Trang đăng nhập hiẻn thị thành công và người dùng có thể nhìn thấy chức năng Quên mật khẩu trên trang đăng nhập. |
| **Công nghệ sử dụng** | Frontend: ReactJS (giao diện tìm kiếm, API)  * Backend: Spring Boot Microservices (Xử lý đăng nhập, gửi OTP, truy vấn cơ sở dữ liệu). * Database: MySQL (Lưu dữ liệu tài khoản người dùng) * API: RESTful API (gửi và nhận dữ liệu từ backend) |
| **Độ phức tạp** | Cao |
| **Điều kiện ràng buộc** | 1. Mật khẩu đạt điều kiện chứa ít nhất 1 ký tự in hoa, 1 ký tự in thường, 1 ký tự số và 1 ký tự đặc biệt.  2. Giới hạn số lần yêu cầu (tránh spam) |

### 2.1.10 PB10 - Quản lý danh sách Friends - Follower

|  |  |
| --- | --- |
| **ID** | PB10 |
| **Tiêu đề** | Quản lý danh sách Friends - Follower |
| **Tác nhân** | Người dùng (Admin, User) |
| **Mô tả** | Người dùng có thể thêm, chấp nhận, hủy kết bạn với các người dùng khác hoặc có thể xem được danh sách bạn bè, người theo dõi. Ngoài ra có thể chặn hoặc bỏ chặn người dùng. |
| **Các bước thực hiện** | 1. Thiết kế giao diện Tìm kiếm trên trang chủ.  2. Xử lý logic và gọi API xử lý các chức năng liên quan đến Quản lý danh sách Friends - Follower.  3. Hiển thị kết danh sách bạn bè vè đã chặn, hiển thị các mục cần thiết cho chức năng Quản lý Friends – Follower. |
| **Điều kiện trước** | 1. Người dùng đang đăng nhập vào hệ thống.  2. Truy cập vào Trang chủ của tài khoản, nhấn chọn quản lý danh sách bạn bè.  3. Hệ thống đưa thông báo khi thực hiện thành công tác vụ như: đã kết bạn thành công. |
| **Điều kiện sau** | Trang chủ tải thành công và người dùng có thể thấy mục bạn bè. |
| **Công nghệ sử dụng** | Frontend: ReactJS (giao diện tìm kiếm, API)  * Backend: Spring Boot Microservices (Xử lý đăng nhập, gửi OTP, truy vấn cơ sở dữ liệu). * Database: MySQL (Lưu dữ liệu tài khoản người dùng) * API: RESTful API (gửi và nhận dữ liệu từ backend) |
| **Độ phức tạp** | Cao |
| **Điều kiện ràng buộc** | Không |

### 2.1.11 PB11 – Tìm kiếm

|  |  |
| --- | --- |
| **ID** | PB09 |
| **Tiêu đề** | Tìm kiếm |
| **Tác nhân** | Người dùng (Admin, User) |
| **Mô tả** | Người dùng có thể tìm kiếm thông qua tên người dùng, từ khóa bài viết hoặc hagtag. |
| **Các bước thực hiện** | 1. Thiết kế giao diện Tìm kiếm trên trang chủ.  2. Xử lý logic và gọi API tìm kiếm.  3. Hiển thị kết quả tìm kiếm hoặc thông báo không tồn tại. |
| **Điều kiện trước** | 1. Người dùng đang đăng nhập vào hệ thống.  2. Truy cập vào Trang chủ của tài khoản, nhấn tìm kiếm và nhập từ khóa cần tìm kiếm.  3. Hệ thống hiển thị các kết quả phù hợp với từ khóa tìm kiếm hoặc thông báo không có kết quả phù hợp. |
| **Điều kiện sau** | Trang chủ tải thành công và ngươid dùng có thể thấy ô tìm kiếm. |
| **Công nghệ sử dụng** | Frontend: ReactJS (giao diện tìm kiếm, API)  * Backend: Spring Boot Microservices (Xử lý đăng nhập, gửi OTP, truy vấn cơ sở dữ liệu). * Database: MySQL (Lưu dữ liệu tài khoản người dùng) * API: RESTful API (gửi và nhận dữ liệu từ backend) |
| **Độ phức tạp** | Cao |
| **Điều kiện ràng buộc** | Không |

### 2.1.12 PB12 – Đăng & Quản lý bài viết

|  |  |
| --- | --- |
| **ID** | PB09 |
| **Tiêu đề** | Tìm kiếm |
| **Tác nhân** | Người dùng (Admin, User) |
| **Mô tả** | Người dùng có thể tìm kiếm thông qua tên người dùng, từ khóa bài viết hoặc hagtag. |
| **Các bước thực hiện** | 1. Thiết kế giao diện Tìm kiếm trên trang chủ.  2. Xử lý logic và gọi API tìm kiếm.  3. Hiển thị kết quả tìm kiếm hoặc thông báo không tồn tại. |
| **Điều kiện trước** | 1. Người dùng đang đăng nhập vào hệ thống.  2. Truy cập vào Trang chủ của tài khoản, nhấn tìm kiếm và nhập từ khóa cần tìm kiếm.  3. Hệ thống hiển thị các kết quả phù hợp với từ khóa tìm kiếm hoặc thông báo không có kết quả phù hợp. |
| **Điều kiện sau** | Trang chủ tải thành công và ngươid dùng có thể thấy ô tìm kiếm. |
| **Công nghệ sử dụng** | Frontend: ReactJS (giao diện tìm kiếm, API)  * Backend: Spring Boot Microservices (Xử lý đăng nhập, gửi OTP, truy vấn cơ sở dữ liệu). * Database: MySQL (Lưu dữ liệu tài khoản người dùng) * API: RESTful API (gửi và nhận dữ liệu từ backend) |
| **Độ phức tạp** | Cao |
| **Điều kiện ràng buộc** | Không |

### 2.1.13 PB13 - Quản lý người dùng

|  |  |
| --- | --- |
| **ID** | PB13 |
| **Tác nhân** | Admin |
| **Mô tả** | |  | | --- | | Thêm, cập nhật, xóa và phân quyền người dùng trong hệ thống | |
| **Tiêu chí chấp nhận** | 1. Người dùng truy cập mục **"Quản lý người dùng"**.  2. **Thêm mới người dùng**:  • Chọn **Tạo mới** và nhập thông tin người dùng (username, mật khẩu, quyền hạn, v.v.).  • Nhấn **"Lưu"** để tạo người dùng mới.  3. **Cập nhật thông tin người dùng**:  • Chọn người dùng cần chỉnh sửa thông tin.  • Cập nhật thông tin người dùng và nhấn **"Lưu"**.  4. **Xóa người dùng**:  • Chọn người dùng cần xóa.  • Hệ thống yêu cầu xác nhận xóa.  • Nếu người dùng đồng ý, hệ thống sẽ xóa người dùng khỏi hệ thống.  5. **Phân quyền người dùng**:  • Chọn người dùng cần phân quyền.  • Chọn quyền hạn phù hợp và lưu lại.  6. Hệ thống xử lý và cập nhật thông tin người dùng trên hệ thống sau khi thêm, cập nhật, xóa, hoặc phân quyền. |
| **Điều kiện trước** | Người dùng có quyền quản lý chức vụ |
| **Điều kiện ràng buộc** | Người dùng chỉ có thể cập nhật hoặc xóa tài khoản khi tài khoản đó đã tồn tại trong hệ thống. |

### 2.1.14 PB14 - Quản lý phân quyền

|  |  |
| --- | --- |
| **ID** | PB14 |
| **Tác nhân** | Admin |
| **Mô tả** |  |
| **Tiêu chí chấp nhận** | 1. Người dùng đăng nhập thành công vào hệ thống và truy cập mục **Quản lý Người dùng**. 2. Danh sách người dùng hiển thị đầy đủ thông tin (Tên, chức vụ, Vai trò, v.v.) cùng với nút thao tác **Phân quyền**. 3. Admin nhấn vào nút **Phân quyền** tương ứng với một người dùng. 4. Hệ thống hiển thị giao diện phân quyền, bao gồm danh sách các quyền hiện có. 5. Admin chọn các quyền cần cấp cho người dùng từ danh sách. 6. Admin nhấn nút **Thêm quyền** để hoàn tất quá trình phân quyền. 7. Sau khi hoàn tất, hệ thống hiển thị thông báo "Phân quyền thành công" và cập nhật danh sách vai trò của người dùng. 8. Nếu Admin không thực hiện phân quyền và thoát giao diện, không có thay đổi nào được lưu. |
| **Điều kiện trước** | Người dùng đã đăng nhập thành công vào hệ thống với tư cách là Admin |
| **Điều kiện ràng buộc** | Tài khoản người dùng cần phân quyền đã tồn tại trong hệ thống |

### 2.1.15 PB15 - Quản lý mẫu văn bản

|  |  |
| --- | --- |
| **ID** | PB15 |
| **Tác nhân** | Admin, Trưởng đơn vị |
| **Mô tả** | Tạo mẫu văn bản mới dựa trên loại văn bản và đơn vị ban hành. |
| **Tiêu chí chấp nhận** | 1. Người dùng truy cập mục **"Quản lý Văn Bản Mẫu"**.  2. Thực hiện các thao tác sau:  • **Thêm mới:** Người dùng chọn **Tạo mới**, nhập thông tin (Tên văn bản mẫu , Mô tả, v.v.), và nhấn **Lưu**.  • **Cập nhật:** Người dùng chọn văn bản mẫu cần chỉnh sửa, cập nhật thông tin và nhấn **Lưu**.  • **Xóa:** Người dùng chọn văn bản mẫu cần xóa và nhấn **Xóa**, hệ thống hiển thị thông báo xác nhận. Nếu người dùng đồng ý, văn bản mẫu sẽ bị xóa khỏi hệ thống.  3. Đính kèm file (nếu có).  4. Nhấn nút **Lưu** để tạo mẫu văn bản.  5. Hệ thống lưu mẫu văn bản và hiển thị thông báo thành công. |
| **Điều kiện trước** | Người dùng có quyền Admin, trưởng đơn vị |
| **Điều kiện ràng buộc** | Văn bản mẫu chỉ có thể xóa khi không có liên kết đến dữ liệu khác trong hệ thống |

### 2.1.16 PB16 - Quản lý chữ ký số

|  |  |
| --- | --- |
| **ID** | PB16 |
| **Tác nhân** | Admin |
| **Mô tả** | Quản lý và cấp chữ ký số cho các cá nhân trong hệ thống. |
| **Tiêu chí chấp nhận** | 1.Người dùng (Admin) đăng nhập thành công vào hệ thống. 2. Truy cập vào mục **Quản lý chữ ký số**. 3. Hệ thống hiển thị danh sách người dùng đủ điều kiện cấp chữ ký số, bao gồm các thông tin như: Tên, Email, Trạng thái chữ ký số (đã cấp/chưa cấp). 4. Admin thực hiện các bước sau:   * Đánh dấu chọn người cần cấp chữ ký số. * Nhấn nút **Sinh khóa**, hệ thống sẽ tự động tạo khóa công khai và khóa bí mật. * Nhấn nút **Tạo khóa**, hệ thống lưu trữ khóa vừa tạo vào cơ sở dữ liệu   5. Sau khi lưu khóa, hệ thống tự động thay đổi trạng thái chữ ký số của người dùng từ "Chưa cấp" sang "Đã cấp". 6. Hiển thị thông báo "Cấp chữ ký số thành công" nếu thực hiện thành công. 7. Nếu người dùng thoát giao diện hoặc không thực hiện đủ các bước, không có thay đổi nào được lưu. |
| **Điều kiện trước** | 1. Người dùng đã đăng nhập thành công vào hệ thống với tư cách là Admin 2. Người dùng được cấp chữ ký số đã tồn tại trong hệ thống và đủ điều kiện (không bị khóa tài khoản, đúng vai trò được chỉ định). |
| **Điều kiện ràng buộc** | 1. Người dùng cần có tài khoản hợp lệ trong hệ thống và không bị khóa hoặc bị hạn chế quyền. 2. Người dùng chưa được cấp chữ ký số |

### 2.1.17 PB17 - Thống kê

|  |  |
| --- | --- |
| **ID** | PB17 |
| **Tác nhân** | Admin |
| **Mô tả** | Quản lý thống kê. |
| **Tiêu chí chấp nhận** | 1. Người dùng (Admin) đăng nhập thành công vào hệ thống. 2. Truy cập mục **Thống kê văn bản** trên giao diện chính. 3. Giao diện hiển thị tùy chọn thống kê, bao gồm:   * **Theo loại văn bản**: Người dùng chọn một hoặc nhiều loại văn bản (ví dụ: Công văn, Quyết định, Báo cáo, v.v.). * **Theo đơn vị ban hành**: Người dùng chọn một hoặc nhiều đơn vị ban hành cụ thể.   4. Chọn phạm vi thời gian thống kê, bao gồm các tùy chọn:   * Tuần trước. * Tháng trước. * Tháng này. * Khoảng thời gian bất kỳ (Admin nhập ngày bắt đầu và ngày kết thúc).   5. Nhấn nút **Thống kê**, hệ thống hiển thị danh sách văn bản gồm các thông tin:   * Tên văn bản. * Số hiệu văn bản. * Ngày ban hành. * Đơn vị ban hành. * Tình trạng văn bản (Đã duyệt/Đang chờ duyệt)   6. Hệ thống hiển thị biểu đồ thống kê tổng quan (nếu có), ví dụ: số lượng văn bản theo loại hoặc theo đơn vị ban hành. |
| **Điều kiện trước** | 1. Người dùng đã đăng nhập thành công vào hệ thống với tư cách là Admin 2. Dữ liệu văn bản đã tồn tại trong hệ thống và được phân loại chính xác. |

### 2.1.18 PB18 - Xem danh sách văn bản đến

|  |  |
| --- | --- |
| **ID** | PB18 |
| **Tác nhân** | Người dùng (Admin, trưởng đơn vị, nhân viên) |
| **Mô tả** | Người dùng có thể xem các văn bản được gửi đến. |
| **Tiêu chí chấp nhận** | 1. Sau khi đăng nhập thành công vào hệ thống, người dùng chọn chức năng “Xem văn danh sách văn bản đến”.  2. Để tìm kiếm văn bản, có thể chọn mục tìm kiếm:   * Hệ thống hiển thị các hình thức tìm kiếm.   +Theo số văn bản. Người dùng chọn số văn bản trên hộp danh sách thả xuống.  +Theo loại: Người dùng chọn các thông tin liên quan như: Loại văn bản, theo thời gian đơn vị ban hành trên hộp danh sách thả xuống.   * Trường hợp văn bản có trong hệ thống sẽ hiển thị danh sách văn bản. Ngược lại, thông báo ” Văn bản trên không tồn tại”. |
| **Điều kiện trước** | Người dùng đã đăng nhập thành công vào hệ thống. |
| **Điều kiện ràng buộc** | Dữ liệu văn bản phải có thông tin đầy đủ về tên, số hiệu, ngày ban hành, đơn vị ban hành, và tình trạng văn bản. |

### 2.1.19 PB19 - Xem chi tiết văn bản đến

|  |  |
| --- | --- |
| **ID** | PB19 |
| **Tác nhân** | Người dùng (Admin, trưởng đơn vị, nhân viên) |
| **Mô tả** | Người dùng có thể xem các văn bản được gửi đến. |
| **Tiêu chí chấp nhận** | 1. Người dùng truy cập vào màn hình **Xem văn bản đến**.  2. Hệ thống hiển thị danh sách các văn bản hiện có.  3. Người dùng chọn một văn bản cụ thể để xem chi tiết.  4. Hệ thống hiển thị các thông tin chi tiết của văn bản đó như:   * Tiêu đề văn bản * Số hiệu văn bản * Ngày ban hành * Nội dung chi tiết   5. Nếu văn bản có file đính kèm, người dùng có thể tải xuống hoặc mở trực tiếp.  6. Người dùng có thể sử dụng các nút điều hướng để quay lại danh sách hoặc tiếp tục xem văn bản khác. |
| **Điều kiện trước** | Người dùng đã đăng nhập thành công vào hệ thống. |
| **Điều kiện ràng buộc** | Văn bản phải có thông tin đầy đủ như tiêu đề, số hiệu, ngày ban hành và nội dung chi tiết. |

### 2.1.20 PB20 - Tìm kiếm văn bản đến

|  |  |
| --- | --- |
| **ID** | PB20 |
| **Tác nhân** | Ngưới dùng (Admin, Trưởng các đơn vi, nhân viên thuộc đơn vị đó) |
| **Mô tả** | Tìm kiếm và lọc các văn bản theo tiêu chí cụ thể |
| **Tiêu chí chấp nhận** | 1. Người dùng truy cập mục "Tìm kiếm văn bản". 2. Nhập các tiêu chí tìm kiếm/lọc:   * **Lọc theo số hiệu**: Nhập số hiệu văn bản cần tìm * **Lọc theo loại văn bản**: Chọn loại văn bản từ danh sách có sẵn. * **Lọc theo ngày**: Nhập ngày hoặc khoảng thời gian cụ thể. * **Tìm kiếm từ khóa**: Nhập từ khóa cần tìm kiếm.   3. Nhấn nút "Tìm kiếm". 4. Danh sách kết quả hiển thị chính xác theo tiêu chí đã chọn. |
| **Điều kiện trước** | Người dùng đã đăng nhập và có quyền tìm kiếm văn bản |

### 2.1.21 PB21 - Xem danh sách văn bản đi

|  |  |
| --- | --- |
| **ID** | PB21 |
| **Tác nhân** | Ngưới dùng (Admin, Trưởng các đơn vi, nhân viên thuộc đơn vị đó) |
| **Mô tả** | Người dùng có thể xem các văn bản đã gửi đi. |
| **Tiêu chí chấp nhận** | 1. Người dùng truy cập vào màn hình **Xem văn bản đi**.  2. Hệ thống hiển thị danh sách các văn bản hiện có.  3. Người dùng chọn một văn bản cụ thể để xem chi tiết.  4. Hệ thống hiển thị các thông tin chi tiết của văn bản đó như:   * Tiêu đề văn bản * Số hiệu văn bản * Ngày ban hành * Nội dung chi tiết   5. Nếu văn bản có file đính kèm, người dùng có thể tải xuống hoặc mở trực tiếp.  6. Người dùng có thể sử dụng các nút điều hướng để quay lại danh sách hoặc tiếp tục xem văn bản khác. |
| **Điều kiện trước** | Người dùng đã đăng nhập thành công vào hệ thống. |
| **Điều kiện ràng buộc** | 1. Người dùng phải có quyền truy cập vào màn hình "Xem văn bản đi" và phải thuộc đơn vị liên quan.  2. Văn bản phải có thông tin đầy đủ như tiêu đề, số hiệu, ngày ban hành và nội dung chi tiết. |

### 2.1.22 PB22 - Xem chi tiết văn bản đi

|  |  |
| --- | --- |
| **ID** | PB22 |
| **Tác nhân** | Admin, Trưởng đơn vị, hoặc nhân viên được cấp quyền |
| **Mô tả** | Người dùng có thể xem chi tiết các văn bản đã gửi đi. |
| **Tiêu chí chấp nhận** | 1. Người dùng (Admin, Trưởng đơn vị, hoặc Nhân viên được cấp quyền) đăng nhập thành công vào hệ thống. 2. Truy cập mục **Văn bản đi** trên giao diện chính. 3. Giao diện hiển thị danh sách các văn bản đi với các chức năng hỗ trợ như:   * **Ô tìm kiếm**: Người dùng nhập từ khóa (ví dụ: tên văn bản, số hiệu) để tìm kiếm văn bản đi cụ thể. * **Ô lọc văn bản đi**: Lọc văn bản theo tiêu chí (ví dụ: đơn vị ban hành, trạng thái duyệt, hoặc ngày gửi). * Hiển thị danh sách các văn bản đi hiện có bao gồm các thông tin cơ bản: Tên văn bản, Số hiệu, Ngày ban hành, Trạng thái (đã duyệt/đang chờ duyệt).   4. Nhấn vào văn bản đi muốn xem chi tiết trong danh sách. 5. Hệ thống hiển thị đầy đủ thông tin chi tiết của văn bản đi, bao gồm:   * Loại văn bản. * Số hiệu văn bản. * Đơn vị ban hành.Tuần trước. * Ngày ban hành và ngày gửi. * Trích yếu nội dung. * Ghi chú (nếu có). * Người gửi * Danh sách người nhận văn bản (gửi đến). * File đính kèm (nếu có). |
| **Điều kiện trước** | 1. Người dùng đã đăng nhập thành công vào hệ thống. 2. Dữ liệu văn bản đi đã tồn tại trong hệ thống và có trạng thái cho phép xem. |
| **Điều kiện ràng buộc** | 1. Người dùng phải có quyền truy cập vào màn hình "Xem văn bản đi" và phải thuộc đơn vị liên quan.  2. Văn bản phải có thông tin đầy đủ như tiêu đề, số hiệu, ngày ban hành và nội dung chi tiết. |

### 2.1.23 PB23 - Tạo văn bản đi

|  |  |
| --- | --- |
| **ID** | PB23 |
| **Tác nhân** | Admin, Trưởng đơn vị, hoặc nhân viên được cấp quyền |
| **Mô tả** | Người dùng có thể tạo văn bản đi |
| **Tiêu chí chấp nhận** | 1. Sau khi đăng nhập thành công vào hệ thống, người dùng chọn văn bản đi, chọn tiếp mục *Thêm* để tạo văn bản đi.  2. Hiển thị các danh mục gồm:   * Phân loại công văn (bắt buộc) * Đơn vị ban hành (bắt buộc) * Số văn bản (bắt buộc) * Ngày ban hành, ngày gửi (bắt buộc) * Đánh số công văn (bắt buộc) * File đính kèm (Không bắt buộc) * Xác nhận chữ ký số (bắt buộc)   + Người dùng click chọn “xác nhận chữ ký số” nếu người dùng không được phép cấp chữ ký số thì hệ thống thông báo “bạn không được cấp chữ ký số”. Ngược lại:  + Hệ thống sẽ tìm chữ ký số và đính kèm vào văn bản thông báo “văn bản đã được xác nhận chữ ký số” trước khi lưu   * Lưu trữ (Bắt buộc)   + Hệ thống kiểm tra:   * Nếu người dùng chưa có chữ ký điện tử hệ thống thông báo “lưu không thành công”. Ngược lại: * Hệ thống thông báo lưu thành công   3. Nút "Gửi" để gửi văn bản sau khi nhập đủ thông tin.  4. Hiển thị thông báo xác nhận khi lưu hoặc gửi thành công. |
| **Điều kiện trước** | 1. Người dùng đã đăng nhập thành công vào hệ thống. 2. Dữ liệu các danh mục liên quan (phân loại công văn, đơn vị ban hành, ...) đã tồn tại trong hệ thống |
| **Điều kiện ràng buộc** | 1. Người dùng phải có quyền tạo văn bản đi. 2. Người dùng phải có quyền cấp chữ ký số, và nếu không có chữ ký số hợp lệ, không thể lưu văn bản đi.  3. Các thông tin như loại công văn, đơn vị ban hành, số văn bản, ngày ban hành, ngày gửi và chữ ký số phải có sẵn trong hệ thống và được xác nhận chính xác. |

### 2.1.24 PB24 - Xóa văn bản đi

|  |  |
| --- | --- |
| **ID** | PB24 |
| **Tác nhân** | Admin, Trưởng đơn vị, hoặc nhân viên được cấp quyền |
| **Mô tả** | Người dùng có thể xóa văn bản đi |
| **Tiêu chí chấp nhận** | 1. Người dùng đăng nhập thành công vào hệ thống.  2. Truy cập mục **Văn bản đi** trên giao diện chính.  3. Hệ thống hiển thị danh sách văn bản đi hiện có.  4. Người dùng chọn văn bản đi cần xóa và nhấn nút **Xóa**.  5. Nếu văn bản đã được gửi đi:   * Hệ thống yêu cầu xác nhận bằng chữ ký số trước khi xóa. * Nếu xác nhận chữ ký số thành công, văn bản được xóa khỏi danh sách. * Nếu không xác nhận chữ ký số, hiển thị thông báo: *“Không thể xóa vì chưa xác nhận chữ ký số”*.   6. Nếu văn bản chưa được gửi đi:   * Hiển thị thông báo xác nhận xóa văn bản: “Bạn có chắc chắn muốn xóa văn bản này?”. * Nếu người dùng đồng ý, văn bản được xóa khỏi hệ thống. |
| **Điều kiện trước** | 1. Người dùng đã đăng nhập thành công vào hệ thống và có được cấp quyền xóa văn bản2. Văn bản đi tồn tại trong hệ thống. |

### 2.1.25 PB25 - Chỉnh sửa văn văn bản đi

|  |  |
| --- | --- |
| **ID** | PB23 |
| **Tác nhân** | Trưởng đơn vị, hoặc nhân viên được cấp quyền |
| **Mô tả** | Người dùng có thể tạo văn bản đi |
| **Tiêu chí chấp nhận** | 1. Sau khi đăng nhập thành công vào hệ thống, người dùng truy cập vào mục "Văn bản đi".  2. Người dùng chọn văn bản đi cần chỉnh sửa.  3. Người dùng nhấn nút "Sửa" để chỉnh sửa văn bản đi.  4. Người dùng nhập các thông tin cần chỉnh sửa:  • **Tiêu đề văn bản**: Chỉnh sửa tiêu đề văn bản nếu cần thiết.  • **Số hiệu văn bản**: Cập nhật số hiệu văn bản nếu có thay đổi.  • **Ngày ban hành**: Điều chỉnh ngày ban hành nếu cần.  • **Ngày gửi**: Cập nhật ngày gửi văn bản.  • **Phân loại công văn**: Thay đổi phân loại nếu cần thiết.  • **Đơn vị ban hành**: Chỉnh sửa đơn vị ban hành nếu cần.  • **Đánh số công văn**: Cập nhật số công văn nếu có thay đổi.  • **File đính kèm**: Thêm, thay đổi hoặc xóa file đính kèm (nếu có). 5. Người dùng nhấn "Lưu" để hoàn thành chỉnh sửa.  6. Hệ thống hiển thị thông báo "Chỉnh sửa thành công" sau khi hoàn tất. |
| **Điều kiện trước** | Người dùng đã đăng nhập thành công vào hệ thống. |
| **Điều kiện ràng buộc** | 1. Người dùng cần có quyền chỉnh sửa văn bản đi trong hệ thống. 2. Văn bản đi cần phải tồn tại trong hệ thống và có thông tin đầy đủ để chỉnh sửa. |

### 2.1.26 PB26 - Tìm kiếm văn bản đi

|  |  |
| --- | --- |
| **ID** | PB26 |
| **Tác nhân** | Trưởng đơn vị, hoặc nhân viên được cấp quyền |
| **Mô tả** | Người dùng có thể tạo văn bản đi |
| **Tiêu chí chấp nhận** | 1. Người dùng truy cập mục "Tìm kiếm văn bản". 2. Nhập các tiêu chí tìm kiếm/lọc:   * **Lọc theo số hiệu**: Nhập số hiệu văn bản cần tìm * **Lọc theo loại văn bản**: Chọn loại văn bản từ danh sách có sẵn. * **Lọc theo ngày**: Nhập ngày hoặc khoảng thời gian cụ thể. * **Tìm kiếm từ khóa**: Nhập từ khóa cần tìm kiếm.   3. Nhấn nút "Tìm kiếm". 4. Danh sách kết quả hiển thị chính xác theo tiêu chí đã chọn. |
| **Điều kiện trước** | Người dùng đăng nhập thành công vào hệ thống |
| **Điều kiện ràng buộc** | Văn bản cần tìm phải tồn tại trong hệ thống và có thông tin đầy đủ. |

### 2.1.27 PB27 - Tải tài liệu, mẫu văn bản

|  |  |
| --- | --- |
| **ID** | PB27 |
| **Tác nhân** | Người dùng (Admin, trưởng các đơn vị, nhân viên) |
| **Mô tả** | Người dùng có thể tải về hoặc lưu mẫu các loại văn bản trong hệ thống. |
| **Tiêu chí chấp nhận** | 1. Người dùng đăng nhập thành công vào hệ thống.  2. Truy cập mục **Văn bản mẫu**  3. Hệ thống hiển thị danh sách các loại tài liệu hoặc văn bản mẫu hiện có.  4. Người dùng có thể tìm kiếm tài liệu mẫu theo tiêu chí:   * Loại văn bản. * Đơn vị ban hành.   5. Chọn mẫu văn bản cần tải về từ danh sách kết quả tìm kiếm hoặc danh sách gốc.  6. Nhấn nút **Download** để tải về tài liệu mẫu.  7. Hệ thống tải về thành công tài liệu mẫu dưới định dạng gốc. |
| **Điều kiện trước** | Người dùng đã đăng nhập thành công vào hệ thống |
| **Điều kiện ràng buộc** | Các tài liệu mẫu phải tồn tại trong hệ thống và được phân loại chính xác. |

## 2.2.Ưu tiên và ước tính

*Bảng xx: Bảng mức độ ưu tiên*

|  |  |  |
| --- | --- | --- |
| **Mức độ ưu tiên** | **Mức độ ảnh hưởng** | **Ký hiệu** |
| Rất cao | Ảnh hưởng nghiêm trọng | 1 |
| Cao | Ảnh hưởng nghiêm trọng | 2 |
| Trung bình | Có ảnh hưởng | 3 |
| Thấp | Không ảnh hưởng | 4 |

|  |  |  |
| --- | --- | --- |
| **ID** | **Chủ đề** | **Sự ưu tiên** |
| PB01 | Đăng nhập | 2 |
| PB02 | Đăng xuất | 2 |
| PB03 | Đổi mật khẩu | 2 |
| PB04 | Quản lý loại văn bản | 1 |
| PB05 | Quản lý đơn vị | 1 |
| PB06 | Quản lý phòng ban | 1 |
| PB07 | Quản lý khối | 1 |
| PB08 | Quản lý phòng | 1 |
| PB09 | Quản lý ngành | 1 |
| PB10 | Quản lý chuyên ngành | 1 |
| PB11 | Quản lý group | 1 |
| PB12 | Quản lý chức vụ | 1 |
| PB13 | Quản lý người dùng | 1 |
| PB14 | Quản lý phân quyền | 1 |
| PB15 | |  | | --- | |  |  |  | | --- | | Quản lý mẫu văn bản | | 3 |
| PB16 | Quản lý chữ ký số | 1 |
| PB17 | Thống kê | 4 |
| PB18 | Xem danh sách văn bản đến | 2 |
| PB19 | Xem chi tiết văn bản đến | 2 |
| PB20 | Tìm kiếm văn bản đến | 2 |
| PB21 | Xem danh sách văn bản đi | 2 |
| PB22 | Xem chi tiết văn bản đi | 2 |
| PB23 | Tạo văn bản đi | 1 |
| PB24 | Xóa văn bản đi | 2 |
| PB25 | Chỉnh sửa văn bản đi | 2 |
| PB26 | Tìm kiếm văn bản đi | 2 |
| PB27 | Tải tài liệu, mẫu văn bản | 3 |

## 2.3. Chia theo từng Sprirnt

|  |  |  |  |
| --- | --- | --- | --- |
| **Sprint** | **Function** | **Started Date** | **Due Date** |
| Sprint 1 | PB01, PB02, PB03, PB04, PB05, PB09, PB13, PB14 | 06/10/2024 | 28/10/2024 |
| Sprirnt 2 | PB06, PB07, PB08, PB10, PB11, PB18, PB19, PB21, PB22, | 29/10/2024 | 18/11/2024 |
| Sprirnt 3 | PB12, PB15, PB16, PB17, PB20, PB23, PB24, PB25, PB26, PB27, | 19/11/2024 | 08/12/2024 |