

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC BÁCH KHOA  
KHOA KHOA HỌC & KỸ THUẬT MÁY TÍNH



BÁO CÁO  
ĐỒ ÁN TỐT NGHIỆP

Phát triển và mở rộng  
nền tảng đánh giá quy trình nghiệp vụ tự động

**Giảng viên hướng dẫn:** TS. Trương Thị Thái Minh  
TS. Tôn Long Phước  
**Sinh viên thực hiện:** Lê Văn Vy 2010805  
Nguyễn Trí Hiếu 2013153



\_\_\_\_\_ Ngày: \_\_\_\_\_

**TS. Trương Thị Thái Minh (Giảng viên hướng dẫn)**

**Tiến sĩ**

**Khoa Khoa học và kỹ thuật máy tính**

# **Mục lục**

# **Danh sách hình vẽ**

# Danh sách bảng

# **Chương 1**

## **Giới thiệu đề tài**

### **1.1 Lý do chọn đề tài**

Quy trình nghiệp vụ là đối tượng chính trong hệ thống BPSky, một hệ thống hỗ trợ đánh giá chất lượng quy trình nghiệp vụ tự động, mô phỏng luồng thực thi và so sánh chất lượng giữa những quy trình khác nhau. Đối với tổ chức có quy mô nhỏ, việc thuê một chuyên gia phụ trách đánh giá chất lượng quy trình nghiệp vụ đôi khi sẽ tiêu hao nhiều tài lực; hệ thống BPSky đã ra đời và cung cấp công cụ giúp hỗ trợ đánh giá chất lượng quy trình một cách nhanh chóng và khách quan.

Tuy nhiên, khi quy mô của quy trình phát triển, điển hình là trong những công ty hoạt động hướng quy trình (process-centered way), nhiệm vụ quản lý nhiều quy trình nghiệp vụ chuyên biệt cho từng bộ phận đóng vai trò không thể thiếu trong việc vận hành bộ máy của tổ chức. Điều này đặt ra nhu cầu mở rộng phạm vi quản lý quy trình của tổ chức, cho phép phân cấp và chia sẻ quy trình giữa những bộ phận trong cùng một tổ chức.

Trong đề tài lần này với tiêu đề "Phát triển và mở rộng nền tảng đánh giá quy trình nghiệp vụ tự động", chúng tôi mong muốn kế thừa và mở rộng phạm vi quản lý của hệ thống. Đồng thời đưa ra những tính năng tích hợp vào bộ công cụ để tăng tính chính xác và khách quan của chức năng đánh giá chất lượng quy trình nghiệp vụ tự động.

### **1.2 Mục tiêu của đề tài**

Hệ thống hướng tới việc cải thiện phạm vi quản lý của không gian làm việc trong một tổ chức. Điều này giúp tăng cường khả năng phân cấp và chia sẻ quy trình giữa các bộ phận, từ đó tạo ra sự linh hoạt trong việc vận hành tổ chức. Hơn nữa, việc mở rộng phạm vi quản lý cũng sẽ hỗ trợ giám sát chất lượng quy trình tốt hơn.

Bên cạnh mục tiêu trên, hệ thống BPSky mong muốn bổ sung tính chính xác của đánh giá liên quan tới chất lượng quy trình. Đa dạng hóa góc nhìn khi đánh giá thông qua việc thu thập ý kiến đến từ phía khách hàng và người tham gia vận hành quy trình. Điều này nhằm cải thiện độ chính xác và tính khách



quan của đánh giá mà hệ thống đề xuất cho người dùng, từ đó tạo được độ tin cậy cao hơn.

Ngoài ra, hệ thống mong muốn hỗ trợ việc tổng hợp và giám sát tình trạng của quy trình đang hoạt động trong tổ chức bằng cách tạo ra một Danh mục quy trình nghiệp vụ (Process portfolio). Danh mục này đánh giá quy trình dựa trên tính khả thi khi vận hành quy trình, hiệu suất đem lại của nó và mức độ ảnh hưởng đến với chiến lược của tổ chức.

### 1.3 Phạm vi của đề tài

Trong phạm vi hệ thống hiện tại, chúng tôi sẽ tập trung phát triển những module chính như sau:

1. Cung cấp không gian làm việc cho phép người dùng quản lý được nhiều mô hình quy trình nghiệp vụ trong tổ chức, đồng thời chia sẻ và phân quyền truy cập đến các thành viên trong hệ thống.
2. Tích hợp chức năng khảo sát vào quá trình đánh giá chất lượng quy trình, khảo sát tập trung vào đối tượng là khách hàng và người tham gia vào quy trình. Điều này nhằm tăng tính khách quan và độ tin cậy của kết quả đánh giá quy trình tự động. Đồng thời cũng thu thập những góp ý và phản hồi của khách hàng và người tham gia quy trình về quá trình thực thi trong thực tế.
3. Tạo danh mục quy trình hoạt động trong không gian làm việc. Danh mục cung cấp góc nhìn tổng quát và trực quan của những quy trình. Từ đó giúp người dùng với vai trò quản lý không gian làm việc đưa ra các quyết định giám sát và cải tiến các quy trình có hiệu suất thấp và tính khả thi không cao.

### 1.4 Ý nghĩa của đề tài

Với mục tiêu phát triển hiện tại của hệ thống, BPSky sẽ trở thành công cụ hỗ trợ đắc lực của nhà quản lý quy trình nghiệp vụ trong việc tự động hóa quá trình đánh giá quy trình nghiệp vụ dựa trên mô hình Business Process Model and Notation (BPMN).

Bên cạnh đó, hệ thống nhằm tối mục tiêu mở rộng phạm vi quản lý mô hình quy trình nghiệp vụ; từ quản lý mô hình trong không gian làm việc cá nhân, mở rộng thành không gian làm việc chung có thể được truy cập và chia sẻ giữa nhiều người dùng; điều này giúp nhà quản lý tiếp cận hệ thống và thực hiện công việc quản lý dễ dàng hơn.

Ngoài ra, hệ thống còn tích hợp khảo sát vào tính năng đánh giá chất lượng quy trình nghiệp vụ nhằm thu thập trải nghiệm của khách hàng và người tham gia vào các tác vụ trong quy trình. Điều này nhằm đem lại góc nhìn đa chiều hơn trong việc đánh giá chất lượng quy trình nghiệp vụ; không chỉ từ góc nhìn của hệ thống và người quản lý quy trình, mà còn dựa trên góc nhìn của khách hàng trải nghiệm sản phẩm/dịch vụ của doanh nghiệp và những cá nhân tham gia thực thi quy trình nghiệp vụ. Yếu tố này sẽ giúp đánh giá chất lượng quy trình nghiệp vụ, vốn đã là một đánh giá có phần trừu tượng và bị ảnh hưởng bởi nhiều yếu tố, theo hướng khách quan và có độ tin cậy cao hơn.

Hơn nữa, hệ thống hỗ trợ người dùng có được góc nhìn tổng quát đối với các quy trình đang hoạt động trong không gian làm việc. Điều này đồng thời liên quan tới việc giám sát và đưa ra các quyết định đối với quy trình trong tổ chức.

## 1.5 Cấu trúc của đồ án

Cấu trúc của đồ án gồm:

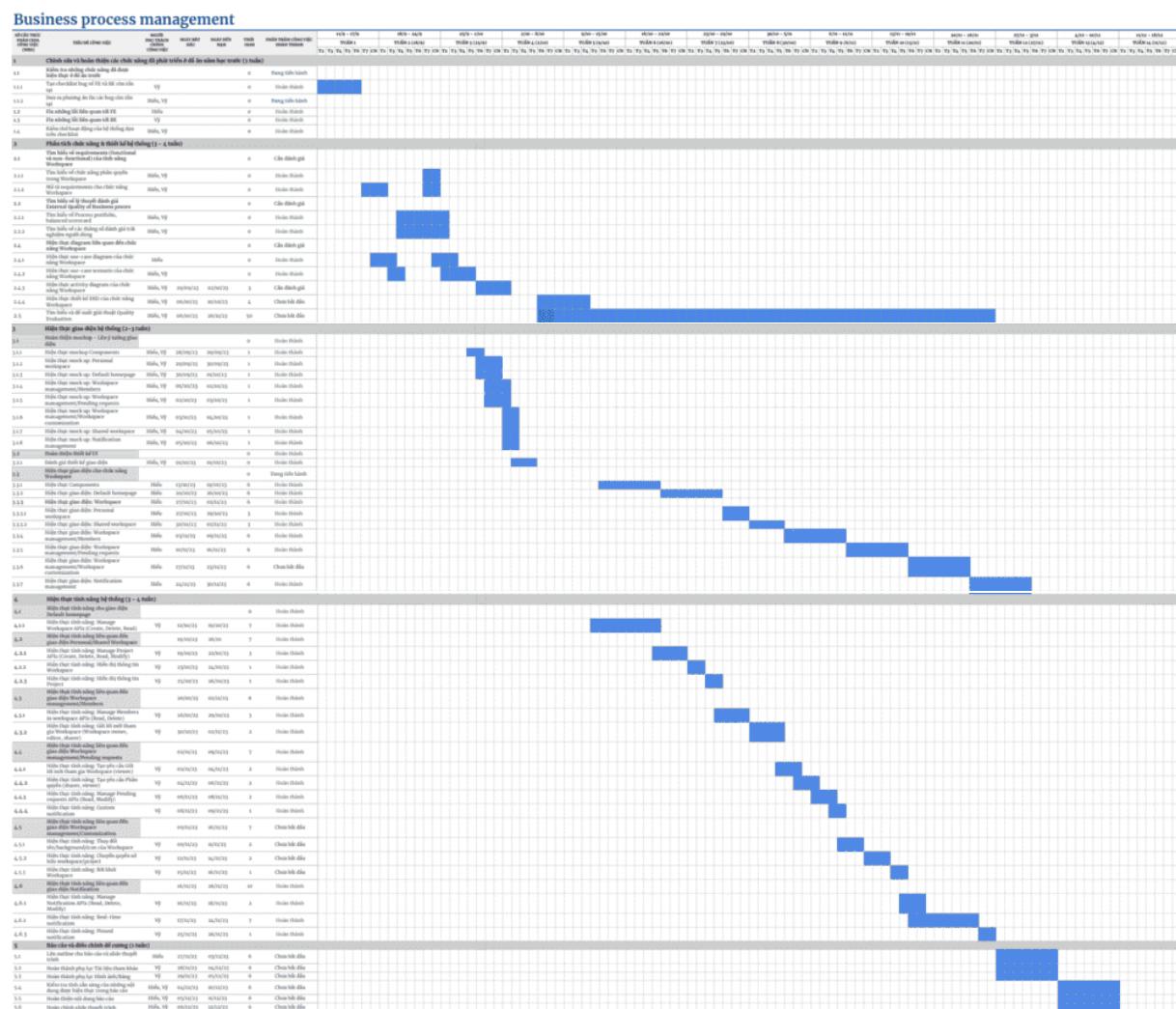
- **Chương 1:** Giới thiệu đề tài, mục đích chọn đề tài, ý nghĩa và phạm vi đề tài, cấu trúc đồ án và kế hoạch phân chia công việc chi tiết.
- **Chương 2:** Cơ sở lý thuyết cho việc đánh giá mô hình BPMN, lý thuyết thiết kế và tính toán kết quả khảo sát, lý thuyết khởi tạo danh mục quy trình.
- **Chương 3:** Phân tích và thiết kế hệ thống
  - Giới thiệu hệ thống: Đưa ra những yêu cầu người dùng cần để thực hiện hệ thống, bao gồm các yêu cầu chức năng và phi chức năng.
  - Use-case: lược đồ use-case của toàn bộ hệ thống và đặc tả use-case.
  - Sơ đồ hoạt động: bao gồm sơ đồ hoạt động của các tính năng chính trong hệ thống.
  - Kiến trúc phần mềm: mô tả kiến trúc hệ thống, so sánh kiến trúc hệ thống với các dạng kiến trúc khác nhau.
  - Sơ đồ lớp: vẽ và mô tả lớp theo mô hình phân lớp của hệ thống.
  - Thiết kế cơ sở dữ liệu: lựa chọn cơ sở dữ liệu và vẽ Entity Relationship Model (ERD), đặc tả các thực thể, mối quan hệ trong cơ sở dữ liệu.
- **Chương 4:** Hiện thực hệ thống
  - Công nghệ sử dụng: giới thiệu những công nghệ được sử dụng để hiện thực hệ thống bao gồm front-end, back-end, database.
  - Hiện thực giao diện người dùng: Giới thiệu các trang giao diện chính của phần mềm.
  - Triển khai hệ thống
- **Chương 5:** Kiểm thử và đánh giá hệ thống
  - Kiểm thử hệ thống: Sử dụng kỹ thuật use-case testing để kiểm thử các nhóm chức năng được hiện thực trong hệ thống.
  - Case study: Ứng dụng hệ thống vào quy trình của một doanh nghiệp thực tế.
  - Kiểm chứng: Kiểm chứng lại kết quả của chức năng mà hệ thống cung cấp so với tính toán lý thuyết.
- **Chương 6:** Kết luận

- Tổng kết kết quả đạt được, nhìn nhận những hạn chế còn tồn đọng.
- Hướng phát triển: Định hướng phát triển hệ thống trong tương lai.

## 1.6 Kế hoạch làm việc

Biểu đồ Gantt kế hoạch làm việc ở hai học kỳ HK231 và HK232 được thể hiện bao gồm nội dung đầu việc, thời gian thực hiện, thời gian tới hạn, tiến độ mỗi tuần và người phụ trách. Mô tả chi tiết phần việc trong quá trình hiện thực hệ thống sẽ được thể hiện ở Bảng phân công công việc chi tiết ở hai học kỳ HK231 và HK232 bên dưới.

### 1.6.1 Giai đoạn học kỳ HK231



Hình 1.1: Biểu đồ Gantt kế hoạch làm việc giai đoạn học kỳ HK231



Mô tả công việc	Phân công	Ngày bắt đầu	Ngày kết thúc
Hiện thực mockup Components	Hiếu, Vỹ	28/09/23	29/09/23
Hiện thực mock up: Personal workspace	Hiếu, Vỹ	29/09/23	30/09/23
Hiện thực mock up: Default homepage	Hiếu, Vỹ	30/09/23	01/10/23
Hiện thực mock up: Workspace management/Members	Hiếu, Vỹ	01/10/23	02/10/23
Hiện thực mock up: Workspace management/Pending requests	Hiếu, Vỹ	02/10/23	03/10/23
Hiện thực mock up: Workspace management/Workspace customization	Hiếu, Vỹ	03/10/23	04/10/23
Hiện thực mock up: Shared workspace	Hiếu, Vỹ	04/10/23	05/10/23
Hiện thực mock up: Notification management	Hiếu, Vỹ	05/10/23	06/10/23
Đánh giá thiết kế giao diện	Hiếu, Vỹ	01/10/23	01/10/23
Hiện thực Components	Hiếu	13/10/23	19/10/23
Hiện thực giao diện: Default homepage	Hiếu	20/10/23	26/10/23
Hiện thực giao diện: Personal workspace	Hiếu	27/10/23	29/10/23
Hiện thực giao diện: Shared workspace	Hiếu	30/10/23	02/11/23
Hiện thực giao diện: Workspace management/Members	Hiếu	03/11/23	09/11/23
Hiện thực giao diện: Workspace management/Pending requests	Hiếu	10/11/23	16/11/23
Hiện thực giao diện: Workspace management/Workspace customization	Hiếu	17/11/23	23/11/23
Hiện thực giao diện: Notification management	Hiếu	24/11/23	30/11/23
Hiện thực tính năng: Manage Workspace APIs (Create, Delete, Read)	Vỹ	12/10/23	19/10/23
Hiện thực tính năng: Manage Project APIs (Create, Delete, Read, Modify)	Vỹ	19/10/23	22/10/23
Hiện thực tính năng: Hiển thị thông tin Workspace	Vỹ	23/10/23	24/10/23
Hiện thực tính năng: Hiển thị thông tin Project	Vỹ	25/10/23	26/10/23
Hiện thực tính năng: Manage Members in workspace APIs (Read, Delete)	Vỹ	26/10/23	29/10/23
Hiện thực tính năng: Gửi lời mời tham gia Workspace (Workspace owner, editor, sharer)	Vỹ	30/10/23	02/11/23
Hiện thực tính năng: Tạo yêu cầu Gửi lời mời tham gia Workspace (viewer)	Vỹ	02/11/23	04/11/23
Hiện thực tính năng: Tạo yêu cầu Phân quyền (sharer, viewer)	Vỹ	04/11/23	06/11/23

Bảng 1.1: Bảng kê hoạch công việc học kỳ HK231 - phần 1

Hiện thực tính năng: Manage Pending requests APIs (Read, Modify)	Vỹ	06/11/23	08/11/23
Hiện thực tính năng: Custom notification	Vỹ	08/11/23	09/11/23
Hiện thực tính năng: Thay đổi tên/background/icon của Workspace	Vỹ	09/11/23	11/11/23
Hiện thực tính năng: Chuyển quyền sở hữu workspace/project	Vỹ	12/11/23	14/11/23
Hiện thực tính năng: Rời khỏi Workspace	Vỹ	15/11/23	16/11/23
Hiện thực tính năng: Manage Notification APIs (Read, Delete, Modify)	Vỹ	16/11/23	18/11/23
Hiện thực tính năng: Real-time notification	Vỹ	17/11/23	24/11/23
Hiện thực tính năng: Pinned notification	Vỹ	25/11/23	26/11/23
Lên outline cho báo cáo và slide thuyết trình	Hiếu	27/11/23	03/12/23
Hoàn thành phụ lục Tài liệu tham khảo	Vỹ	28/11/23	04/12/23
Hoàn thành phụ lục Hình ảnh/Bảng	Vỹ	29/11/23	05/12/23
Kiểm tra tính sẵn sàng của những nội dung được hiện thực trong báo cáo	Hiếu, Vỹ	04/12/23	10/12/23
Hoàn thiện nội dung báo cáo	Hiếu, Vỹ	05/12/23	11/12/23
Hoàn chỉnh slide thuyết trình	Hiếu, Vỹ	06/12/23	12/12/23

Bảng 1.2: Bảng kê hoạch công việc học kỳ HK231 - phần 2

## 1.6.2 Giai đoạn học kỳ HK232



Hình 1.2: Biểu đồ Gantt kế hoạch làm việc giai đoạn học kỳ HK232



Mô tả công việc	Phân công	Ngày bắt đầu	Ngày kết thúc
Tìm hiểu về requirements (functional và non-functional) của tính năng Survey	Hiếu, Vỹ	05/02/24	11/02/24
Tìm hiểu về requirements (functional và non-functional) của tính năng Process Portfolio	Hiếu, Vỹ	05/02/24	11/02/24
Thiết kế ERD, relational mapping cho tính năng Survey và Process portfolio	Hiếu, Vỹ	12/02/24	25/02/24
Hiện thực diagram liên quan đến chức năng Survey (Use-case diagram, use-case scenario, activity diagram)	Hiếu, Vỹ	05/02/24	11/02/24
Hiện thực diagram liên quan đến chức năng Process Portfolio (Use-case diagram, use-case scenario, activity diagram)	Hiếu, Vỹ	05/02/24	11/02/24
Chỉnh sửa class diagram của hệ thống	Hiếu, Vỹ	04/03/24	10/03/24
Hiện thực mockup Survey	Hiếu, Vỹ	12/02/24	18/02/24
Hiện thực mock up: Process Portfolio	Hiếu, Vỹ	19/02/24	25/02/24
Đánh giá, chỉnh sửa thiết kế giao diện, thiết kế ERD	Hiếu, Vỹ	26/02/24	03/03/24
Hiện thực giao diện Survey Builder	Hiếu	04/03/24	10/03/24
Hiện thực giao diện Survey Configuration	Hiếu	04/03/24	10/03/24
Hiện thực giao diện Survey Result	Hiếu	11/03/24	17/03/24
Hiện thực giao diện Survey Result (trên toàn bộ câu hỏi)	Hiếu	22/04/24	28/04/24
Hiện thực giao diện Survey Preview	Hiếu	11/03/24	17/03/24
Hiện thực giao diện Survey cho Người tham gia survey	Hiếu	11/03/24	17/03/24
Hiện thực giao diện đánh giá Feasibility	Hiếu	18/03/24	19/03/24
Hiện thực giao diện đánh giá Strategic importance	Hiếu	20/03/24	21/03/24
Hiện thực giao diện đánh giá Health	Hiếu	22/03/24	23/03/24
Hiện thực giao diện khởi tạo Process portfolio	Hiếu	23/03/24	24/03/24
Hiện thực tính năng Survey Builder	Vỹ	04/03/24	10/03/24
Hiện thực tính năng Survey Configuration	Vỹ	04/03/24	10/03/24
Hiện thực tính năng Survey Result	Vỹ	11/03/24	17/03/24
Hiện thực tính năng Survey Result (trên toàn bộ câu hỏi)	Vỹ	22/04/24	28/04/24

Bảng 1.3: Bảng kế hoạch công việc học kỳ HK232 - phần 1



Mô tả công việc	Phân công	Ngày bắt đầu	Ngày kết thúc
Hiện thực tính năng Survey Preview	Vỹ	11/03/24	17/03/24
Hiện thực tính năng Survey cho Người tham gia survey	Vỹ	11/03/24	17/03/24
Hiện thực tính năng đánh giá Feasibility	Vỹ	18/03/24	19/03/24
Hiện thực tính năng đánh giá Strategic importance	Vỹ	20/03/24	21/03/24
Hiện thực tính năng đánh giá Health	Vỹ	22/03/24	23/03/24
Hiện thực tính năng khởi tạo Process portfolio	Vỹ	23/03/24	24/03/24
Kiểm thử đơn vị với nhóm tính năng chỉnh sửa workspace	Hiếu, Vỹ	25/03/24	26/03/24
Kiểm thử đơn vị với nhóm tính năng quản lý thành viên trong workspace	Hiếu, Vỹ	27/03/24	28/03/24
Kiểm thử đơn vị với nhóm tính năng quản lý yêu cầu từ thành viên trong workspace	Hiếu, Vỹ	29/03/24	30/03/24
Kiểm thử: Nhóm tính năng project	Hiếu, Vỹ	30/03/24	31/03/24
Kiểm thử: Nhóm tính năng notification	Hiếu, Vỹ	30/03/24	31/03/24
Kiểm thử đơn vị với nhóm tính năng chỉnh sửa nội dung bảng khảo sát	Hiếu, Vỹ	01/04/24	02/04/24
Kiểm thử đơn vị với nhóm tính năng thực hiện bảng khảo sát (từ phía người dùng)	Hiếu, Vỹ	03/04/24	05/04/24
Kiểm thử đơn vị với nhóm tính năng đánh giá kết quả bảng khảo sát	Hiếu, Vỹ	05/04/24	07/04/24
Kiểm thử đơn vị với tính năng đánh giá tính khả thi của process	Hiếu, Vỹ	01/04/24	02/04/24
Kiểm thử đơn vị với tính năng đánh giá độ quan trọng chiến lược của process	Hiếu, Vỹ	03/04/24	04/04/24
Kiểm thử đơn vị với tính năng đánh giá hiệu suất của process	Hiếu, Vỹ	05/04/24	06/04/24
Kiểm thử đơn vị với tính năng khởi tạo process portfolio	Hiếu, Vỹ	07/04/24	07/04/24
Deploy source code FE với Vercel	Hiếu	08/04/24	14/04/24
Triển khai Docker containers với source code BE	Hiếu, Vỹ	08/04/24	14/04/24
Deploy source code BE với Render và Neon.tech	Vỹ	08/04/24	21/04/24
Lên outline cho báo cáo và slide thuyết trình	Hiếu, Vỹ	29/04/24	05/05/24
Hoàn thiện nội dung báo cáo	Hiếu, Vỹ	06/05/24	12/05/24
Hoàn thành phụ lục Hình ảnh, Bảng, Tài liệu tham khảo	Hiếu, Vỹ	10/05/24	12/05/24
Hoàn thiện slide thuyết trình	Hiếu, Vỹ	06/05/24	12/05/24
Đánh giá và chỉnh sửa phần trình bày/thuyết trình với GVHD	Hiếu, Vỹ	13/05/24	19/05/24

Bảng 1.4: Bảng kế hoạch công việc học kỳ HK232 - phần 2

## Chương 2

# Cơ sở lý thuyết

### 2.1 Đánh giá chất lượng quy trình nghiệp vụ

Chức năng đánh giá tự động quy trình nghiệp vụ được phát triển ở hệ thống BPSky trong giai đoạn trước đó; và ở giai đoạn này, chúng tôi mong muốn mở rộng khía cạnh chất lượng của quy trình nghiệp vụ thông qua việc thu thập thêm trải nghiệm của người dùng và người tham gia vào quy trình nhằm tăng tính chính xác và khách quan của kết quả.

#### 2.1.1 Thời gian chu kỳ

Thời gian chu kỳ của một quy trình được tính toán dựa trên thời gian chu kỳ của từng hoạt động trong quy trình đó. Thời gian được sử dụng ở đây là thời gian lý thuyết, không tính thời gian thực tế. Quy trình nghiệp vụ được chia thành các khối nhỏ hơn. Khi tính toán được thời gian chu kỳ của từng khối, có thể dễ dàng tính toán được chu kỳ của cả quy trình.

- Khối tuần tự (sequential fragment): thời gian chu kỳ của 1 khối tuần tự bằng với thời gian chu kỳ của các hoạt động trong khối đó:

$$CT = \sum_{i=1}^n T_i \quad [?]$$

- Khối XOR (Exclusive gateway): khối được bắt đầu và kết thúc bởi 2 cổng XOR, trong trường hợp này cần quan tâm tới xác suất rẽ nhánh của từng nhánh đi ra từ cổng XOR, ta ký hiệu là p:

$$CT = \sum_{i=1}^n p_i \times T_i \quad [?]$$

- Khối AND (parallel gateway): Khối được bắt đầu và kết thúc bởi 2 cổng AND, ta cần biết thời gian lớn nhất mà các nhánh cần để thực thi khối AND này.

$$CT = \text{Max}(T_1, T_2, \dots, T_n) \quad [?]$$



- Vòng lặp (rework pattern): Một chuỗi các hoạt động hoặc một hoạt động được lặp đi lặp lại với tần suất  $r$  nào đó, ta có công thức tính thời gian chu kỳ của vòng lặp:

$$CT = \frac{T}{1-r} \quad [?]$$

Với  $T$  là tổng thời gian chu kỳ của khối lặp này

- Khối OR (inclusive gateway): Khối được bắt đầu và kết thúc bởi 2 cổng OR, cho phép ta chọn được một hoặc nhiều hơn một nhánh được rẽ ra từ cổng này. Công thức tổng quát cho khối OR:

$$CT = \text{Max}(T_1, T_2, \dots, T_i) \quad [?]$$

Với  $T_1, T_2, \dots, T_i$  là những nhánh được xác định sẽ được chạy trong khối OR này và  $i \leq n$  với  $n$  là tất cả các nhánh trong khối OR. Các nhánh được chạy sẽ được xác định ngay từ đầu.

### 2.1.2 Chi phí

Trong đề tài này, tương tự như đề tài trước, chúng tôi vẫn dựa trên mô hình chi phí dựa trên hoạt động theo thời gian *Time-driven activity-based costing model* - chi phí được tính dựa trên thời gian cần thiết để thực hiện một tác vụ có trong quy trình và chi phí đơn vị (unit cost) của khả năng cung ứng (supplying capacity) theo thời gian. Công thức tính như sau:

$$C = UC \times T \quad [?]$$

Trong đó  $C$  là tổng chi phí của cả quy trình,  $T$  là thời gian chu kỳ của quy trình và  $UC$  (unit cost) là chi phí đơn vị của quy trình đó. Chi phí đơn vị được tính như sau:

$$UC = \frac{CTT}{PC} \quad [?]$$

Trong đó  $CTT$  là tổng số thời gian mà một đơn vị (phòng ban) trong tổ chức có thể bỏ ra trong một đơn vị thời gian.  $PC$  là chi phí có thể cung cấp cho một đơn vị của tổ chức trong một đơn vị thời gian.

### 2.1.3 Độ minh bạch

Tính minh bạch của một quy trình trong một khung nhìn (lane) cụ thể được xác định bởi số lượng tác vụ minh bạch trong khung nhìn tương ứng so với tổng số tác vụ minh bạch hiện có trong quy trình đó trong khung nhìn tổng quát. Một hoạt động được xem như là minh bạch khi nó không phải là một Quy trình phụ - (*sub process*) hay một hoạt động tham chiếu đến một quy trình bên ngoài, còn gọi là Hoạt động gọi - (*call activity*):

$$\text{Transparent Level (specific view)} = \frac{\text{Num of explicit tasks (specific view)}}{\text{Num of explicit tasks (full view)}} \quad [?]$$

### 2.1.4 Khả năng xử lý ngoại lệ

Ngoại lệ là những sự kiện khiến cho quy trình đi từ luồng thông thường sang tình huống không mong đợi và không thường xảy ra. Có thể đó là những lỗi trong quy trình nghiệp vụ hoặc lỗi về mặt kỹ thuật. Công thức tính độ xử lý ngoại lệ như sau:

$$\text{Exception handling level} = \frac{\text{Num of handled exceptions}}{\text{Num of handled exceptions} + \text{Num of unhandled exceptions}} \quad [?]$$

Chúng tôi cùng với đề tài trước quy định để biểu diễn một ngoại lệ, người dùng phải biểu diễn bằng các sự kiện lỗi (*error event*), nếu như sự kiện đó dẫn ra một hoạt động khác thì ngoại lệ đó được xem như là đã được xử lý. Còn lại các ngoại lệ khác sẽ được xem như chưa được xử lý.

### 2.1.5 Độ linh hoạt

Độ linh hoạt được biểu diễn bằng số lượng biến thể có thể có của một quy trình. Biến thể được tạo ra vì chúng có các khối hay các nhánh không phải thực thi. Độ linh hoạt của một quy trình được tính như sau:

$$F = \frac{\text{Num of optional tasks}}{\text{Num of total tasks}} \quad [?]$$

Với *Num of optional tasks* là số lượng tasks có trong khối XOR, và *Num of total tasks* là tổng các tasks có trong quy trình đó.

### 2.1.6 Chất lượng

Chất lượng là một khái niệm trừu tượng và khó để mô hình hoá cụ thể. Hệ thống ở giai đoạn trước, chất lượng của quy trình được đo lường dựa trên xác suất lặp lại  $r$  của các khối lặp, với công thức tính như sau:

$$Q = 1 - \frac{(r_1 + r_2 + \dots + r_n)}{n} \quad [?]$$

trong đó  $r_1, r_2, \dots, r_n$  là xác suất lặp lại của các vòng lặp trong quy trình,  $n$  là tổng số vòng lặp có trong quy trình.

Tuy nhiên, ở đề tài này, chúng tôi nhận thấy chất lượng của một quy trình không chỉ đến từ xác suất của các vòng lặp trong quy trình, mà nó còn có thể bị tác động bởi nhiều yếu tố khác. Chất lượng của một quy trình có thể được xem xét ở hai góc nhìn khác nhau: góc nhìn từ phía người thiết kế quy trình, và góc nhìn của người tham gia vận hành quy trình và sử dụng sản phẩm, dịch vụ của quy trình đó trên thực tế. Đây cũng có thể được xem lần lượt là *internal quality* (chất lượng từ tài nguyên bên trong hệ thống) và *external quality* (chất lượng từ tài nguyên bên ngoài hệ thống).

*Internal quality* của một quy trình nghiệp vụ trong ngữ cảnh của hệ thống ám chỉ chất lượng được tính toán thông qua mô hình quy trình mà người dùng cung cấp cho hệ thống. Ở đây, chúng tôi cho rằng *internal quality* gắn liền với số lượng và xác suất của các vòng lặp có trong quy trình, được biểu diễn và tính toán như trên. Mặt khác, *external quality* lại được đánh giá bởi trải nghiệm của những người tham

gia vận hành hay người sử dụng sản phẩm, dịch vụ của quy trình đó trong thực tế.

## 2.2 Danh mục quy trình

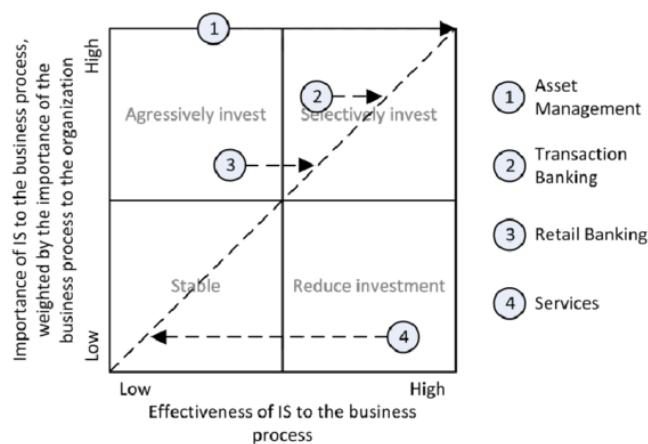
### 2.2.1 Khái niệm Danh mục quy trình

Nhìn vào thực tế, quy trình nghiệp vụ thường gắn liền với một tổ chức hoặc công ty, vì vậy việc quản lý quy trình nghiệp vụ liên quan đến một nhóm những quy trình có ảnh hưởng tới nhau.

Tuy nhiên, không phải tất cả những bên liên quan (stakeholders) phụ trách quản lý quy trình nghiệp vụ đều có được cái nhìn tổng quát về toàn bộ quy trình trong tổ chức của họ.

Việc đặt ra nhu cầu nắm bắt được bức tranh tổng quát về hệ thống quy trình trong tổ chức là cần thiết, từ đó chúng ta có thể đưa ra những quyết định liên quan tới việc giám sát hiệu suất của quy trình hoặc thay đổi, chỉnh sửa quy trình đó.

Với mong muốn hỗ trợ những người quản lý quy trình nghiệp vụ có được góc nhìn chính xác về việc xác định những quy trình nào thì cần được cải tiến, hệ thống BPSky hình thành nên Danh mục quy trình (hay còn gọi là Process portfolio).



Hình 2.1: Ví dụ về Danh mục quy trình (Process portfolio) [26]

Danh mục quy trình đề cập tới một lược đồ có nhiệm vụ trực quan hóa những quy trình nghiệp vụ thông qua những tiêu chí được đề ra. Những quy trình được biểu diễn dưới dạng những node, thông tin của từng node sẽ được cung cấp từ phía người dùng và cả từ những đánh giá của hệ thống về mô hình quy trình nghiệp vụ hiện tại. [14]

Danh mục quy trình giúp người quản lý hình dung được mức độ ưu tiên giữa những quy trình nghiệp vụ, đây là cơ sở để chúng ta có được những bước đi kế tiếp liên quan tới việc phân tích, đánh giá, tái thiết kế quy trình. [14]

## 2.2.2 Lựa chọn quy trình

Không phải toàn bộ những quy trình đều có được sự ảnh hưởng tương đương đối với việc vận hành tổ chức, mỗi một quy trình đều có những nhiệm vụ và mục tiêu khác nhau. Vì vậy, mức độ ưu tiên khi đặt hai quy trình khác nhau lên bàn cân không ít thì nhiều sẽ khác nhau. Mục đích của quá trình lựa chọn quy trình là nhằm xác định những tiêu chí để đánh giá được hiệu suất vận hành của một quy trình nghiệp vụ, từ đó giúp người quản lý rút ra những thông tin hỗ trợ cho việc giám sát, cải thiện hay loại trừ những quy trình không hiệu quả.

Việc lựa chọn quy trình sẽ ảnh hưởng bởi nhiều yếu tố, trong số có đây là những yếu tố thường được cân nhắc:

### 2.2.2.1 Độ quan trọng chiến lược

Mức độ quan trọng chiến lược liên quan tới sự ảnh hưởng hoặc giá trị của một quy trình đối với việc đạt được mục tiêu lâu dài của tổ chức. Những quy trình này có thể liên quan tới tính độc quyền thương hiệu, quy trình đóng góp vào việc tạo ra lợi thế cạnh tranh trong thị trường đối với các đối thủ khác hoặc quy trình đóng góp vào việc tạo ra giá trị cho khách hàng và đem lại nguồn lợi nhuận chính. [14]

Việc xác định được tầm quan trọng của quy trình đối với chiến lược của tổ chức sẽ là chìa khóa giúp chúng ta nhìn ra được những quy trình trọng tâm cần nhắm tới khi muốn phát triển bền vững và lâu dài.

### 2.2.2.2 Sức khỏe

Health, hay gọi là Công năng của quy trình, là tiêu chí nhắm vào việc đánh giá mức độ hiệu quả và hiệu suất hoạt động của quy trình nghiệp vụ. Nó đo lường khả năng tổ chức thực hiện quy trình một cách hiệu quả, tiết kiệm và linh hoạt xử lý các trường hợp ngoại lệ nhưng cũng đồng thời đảm bảo các tiêu chuẩn về đầu ra. [14]

Việc đánh giá tiêu chí này sẽ dễ dàng với những tổ chức hoạt động hướng quy trình - xác định nghiệp vụ dưới dạng những quy trình nghiệp vụ có thể mô hình hóa được; đây là một lợi thế cho phép doanh nghiệp thu thập được những thông tin cần thiết cho quá trình đánh giá hiệu suất. Cụ thể hơn, việc đánh giá hiệu suất quy trình nghiệp vụ sẽ được mô tả thông qua 4 tiêu chí: thời gian (time), chi phí (cost), chất lượng (quality) và độ linh hoạt (flexibility). [14]

- **Thời gian:** Thời gian là yếu tố phổ biến trong đánh giá hiệu suất quy trình nghiệp vụ. Cụ thể hơn thì đại lượng được tính toán phổ biến là thời gian chu kỳ (cycle time) - khoảng thời gian mà quy trình nghiệp vụ bắt đầu cho đến khi kết thúc. Mục tiêu của đa phần những quy trình nghiệp vụ là giảm thời gian chu kỳ thực tế, điều này có thể hiểu là giảm thời gian chờ đợi (waiting time) - khoảng thời gian mà quy trình chờ để nhận được phản hồi từ người dùng; hoặc thời gian phục vụ (serving time) - khoảng thời gian thực sự tiêu tốn để hoàn thành tác vụ quy định trong quy trình nghiệp vụ. Trong ngữ cảnh của hệ thống hiện tại, chúng ta vẫn sẽ tập trung vào đánh giá hiệu suất của quy trình dựa vào đại lượng thời gian chu kỳ (cycle time). [14]

- **Chi phí:** Khi chúng ta đề cập tới vấn đề chi phí, mục tiêu thường là giảm thiểu chi phí được sử dụng để vận hành quy trình nghiệp vụ. Chi phí này có thể là chi phí cố định (fixed cost) - chi phí không bị ảnh hưởng bởi cường độ và tần suất vận hành của quy trình nghiệp, thường là chi phí sử dụng cơ sở vật chất, chi phí bảo trì hệ thống,...; hoặc chi phí biến đổi (variable cost) - chi phí liên quan tới "lượng", ví dụ phụ thuộc vào số lượng sản phẩm, số lượng nhân viên,... Tiếp tục xem xét khía cạnh chi phí theo mô hình chi phí dựa trên hoạt động theo thời gian (Time-driven activity-based costing model); trong mô hình này chúng ta sẽ quan tâm tới chi phí đơn vị (unit cost) mà tổ chức chi trả cho quy trình nghiệp vụ để thực thi, vận hành trong một đơn vị thời gian. [14]
- **Chất lượng:** Như đã trình bày ở trên, chất lượng của quy trình có thể được nhìn nhận dưới hai góc nhìn chính: từ phía mô hình quy trình (internal quality) và từ phía trải nghiệm do người tham gia quy trình chia sẻ (external quality). Chất lượng quy trình là khái niệm có phần trừu tượng, định nghĩa chất lượng có thể được tổng hợp từ nhiều yếu tố nhỏ liên quan; tuy nhiên trong ngữ cảnh hệ thống hiện tại, chúng ta sẽ tập trung phản ánh chất lượng của quy trình thông qua tính dễ hiểu của quy trình đối với người tham gia quy trình (process's participants - phân biệt với khách hàng sử dụng sản phẩm/dịch vụ cung cấp bởi đầu ra của quy trình). Trong chức năng đánh giá chất lượng quy trình, thời gian của các tác vụ trong khối lặp trong quy trình ảnh hưởng theo một trọng số lớn đến tổng thời gian của cả quy trình. Vì vậy xác suất lặp của khối lặp càng lớn thì khả năng nó tác động đến tổng thời gian của quy trình càng cao, chất lượng bên trong của quy trình sẽ bị ảnh hưởng bởi xác suất lặp của các khối lặp. Bên cạnh đó, phản hồi từ phía người tham gia quy trình nghiệp vụ cũng đóng góp vào việc đánh giá chất lượng của quy trình. Phản hồi có thể được thu thập thông qua khảo sát hoặc những lời góp ý từ phía người tham gia quy trình. Hệ thống hiện tại sẽ tập trung vào việc khai thác phản hồi từ phía người dùng để hoàn thiện hơn quá trình đánh giá chất lượng của quy trình nghiệp vụ. [14]
- **Độ linh hoạt:** Độ linh hoạt của quy trình thể hiện thông qua khả năng thay đổi để thích ứng dưới những điều kiện khác nhau của quy trình nghiệp vụ. Tổ chức thường có mong muốn khiến quy trình nghiệp vụ của họ nhanh hơn, rẻ hơn và tốt hơn mà không chú ý đến yếu tố thay đổi của quy trình. Có thể dưới những hoàn cảnh khác nhau, tính ổn định vốn có của quy trình sẽ đánh mất hoàn toàn; khi đó, để so sánh hai quy trình với nhau thì quy trình có thể hoạt động ổn định trong đa số hoàn cảnh lại chiếm ưu thế hơn quy trình hoạt động tốt khi tải bình thường nhưng lại đình trệ khi tải vượt quá cao. Trong phạm vi hệ thống, định nghĩa tính linh hoạt sẽ xoay quanh Khả năng sinh ra luồng thực thi ứng với những điều kiện đầu vào khác nhau: Luồng thực thi khi này sẽ được xem xét như là một biến thể (variation) của quy trình nghiệp vụ. [14]

Những tiêu chí được trình bày bên trên đều được đánh giá dưới dạng các đại lượng định tính/định lượng, mỗi đại lượng sẽ được thể hiện thông qua những đơn vị và thang đo khác nhau.

Nhằm có được đánh giá tổng hợp về công năng của quy trình, chúng ta cần đưa những đại lượng này về cùng một hệ quy chiếu, thông qua việc chuyển đổi đại lượng về cùng một đơn vị đo lường. Để làm được điều này, hệ thống sử dụng lý thuyết của Horkoff liên quan tới việc đo lường mức hiệu năng

(performance level) của các chỉ số, mức hiệu năng này sẽ phản ánh mức độ đạt được mục tiêu của quy trình nghiệp vụ. Giá trị sẽ này được quy về khoảng giá trị từ (-1, 1).

Trước tiên, chúng ta biết rằng một đại lượng có thể là tích cực (positive), tiêu cực (negative) hoặc lưỡng cực (bidirectional); nghĩa là chúng ta cần tối đa hóa, tối thiểu hóa hoặc cân bằng giá trị của đại lượng đó để mức độ đạt được mục tiêu của đại lượng tăng. Để làm rõ mục tiêu của đại lượng thì chúng ta cần hiểu được lý do vì sao đại lượng đó là cần thiết. Ví dụ, chỉ số "Thời gian chu kỳ của quy trình" là cần thiết để đánh giá mục tiêu "Tối thiểu hóa thời gian chu kỳ của quy trình". [14]

Sau khi xác định được tính chất của đại lượng, chúng ta quan tâm đến bộ 3 giá trị lần lượt là Target value - Giá trị mục tiêu, Threshold value - Giá trị ngưỡng và Worst value - Giá trị xấu nhất; trong đó:

$$Target\ value \geq Threshold\ value \geq Worst\ value \quad (2.1)$$

Bộ ba giá trị sẽ giúp chúng ta xác định được mức hiệu năng của chỉ số Current value mà người dùng nhập vào, đang ở mức độ nào trong khoảng giá trị (-1, 1). [27]

$$pl(current\ v.) = \frac{|current\ v. - threshold\ v.|}{|target\ v. - threshold\ v.|} (current\ v. \geq threshold\ v.) \quad (2.2)$$

hoặc

$$pl(current\ v.) = -\frac{|current\ v. - threshold\ v.|}{|threshold\ v. - worst\ v.|} (current\ v. < threshold\ v.) \quad (2.3)$$

#### 2.2.2.3 Tính khả thi

Tính khả thi là tiêu chí thứ ba được cân nhắc trong process portfolio; trong phạm vi ngữ cảnh của hệ thống, tính khả thi của quy trình nghiệp vụ sẽ liên quan tới câu hỏi "Quy trình có khả năng mở rộng không?". Thay vì tập trung vào câu hỏi "Quy trình có khả năng hiện thực không?" thì khả năng mở rộng của quy trình trong tương lai sẽ ảnh hưởng nhiều hơn tới quyết định đầu tư tài nguyên và nhân lực của tổ chức. Tính khả thi thể hiện rõ ràng hơn trên con đường phát triển dài về sau; khi mà mục tiêu hiệu suất trở nên ngày càng cao thì doanh nghiệp sẽ phải nỗ lực hơn để liên tục thử nghiệm và thay đổi quy trình nghiệp vụ, quá trình này có thể kéo dài theo năm. Vì vậy để có được điểm khởi đầu tốt thì tính khả thi của quy trình nghiệp vụ là đáng được cân nhắc. [14]

## 2.3 Thiết kế bảng khảo sát

Đánh giá *external quality* đồng nghĩa với việc đánh giá được sự hài lòng của người dùng hoặc khách hàng đối với một quy trình nghiệp vụ. Để làm được như vậy, cần phải thu thập những ý kiến của người sử dụng hoặc theo dõi và tự động đánh giá quá trình người dùng thực thi quy trình nghiệp vụ. Ngày



nay, có một số công cụ dưới dạng các tiện ích bổ sung (add-on extensions) trên các trình duyệt web, cho phép tích hợp vào hệ thống để theo dõi các luồng thực thi của khách hàng trên một quy trình cụ thể, chẳng hạn như khách hàng có hoàn thành phiên hoạt động của mình hay không, khách hàng có rời vào những trường hợp ngoại lệ không, có khách hàng nào dừng quy trình giữa chừng hay không. Ứng dụng các công cụ trên vào đánh giá chất lượng bên ngoài của quy trình có thể đảm bảo tính chính xác, và được thực thi một cách tự động dựa trên việc theo dõi hành vi của người dùng, không tốn nhiều thời gian, công sức của người dùng. Tuy nhiên, không phải quy trình nghiệp vụ nào cũng có thể được tích hợp công cụ trên. Một số quy trình nghiệp vụ cần được thực thi ngoài thực tế, không phải trên các trang web hay hệ thống máy tính, dẫn tới không thể ứng dụng các công cụ theo dõi hành vi người dùng vào những quy trình nghiệp vụ này. Chúng tôi nhận thấy rằng, đối với những loại quy trình này nói riêng, hay cả những quy trình khác nói chung, đều có thể thu thập được sự hài lòng và trải nghiệm của người dùng thông qua các cuộc khảo sát (survey).

Khảo sát đã được ứng dụng từ lâu để lấy ý kiến khách hàng về một dịch vụ, sản phẩm,... hay thậm chí là cả một tổ chức, một doanh nghiệp. Ngày nay, khảo sát vẫn là một công cụ phổ biến và hiệu quả để thu thập thông tin từ người dùng, cho nên ta có thể sử dụng khảo sát, với các câu hỏi khai thác hợp lý, có thể đánh giá được thái độ của khách hàng đối với quy trình nghiệp vụ cụ thể như thế nào, từ đó có thể rút ra được *external quality* của quy trình. Khảo sát có thể được thực thi trên nhiều nền tảng khác nhau với nhiều hình thức khác nhau, chẳng hạn như có thể khảo sát khách hàng thông qua chatbot - là một phần mềm ứng dụng trí tuệ nhân tạo để mô phỏng lại các cuộc trò chuyện với người dùng, thông qua cuộc trò chuyện đó có thể lấy được thông tin từ người dùng. Hoặc có thể là một bảng khảo sát trực tuyến với các câu hỏi đa dạng hình thức được thiết kế sẵn để người dùng trả lời và gửi phản hồi về quy trình đó. Khảo sát trực tuyến có một số lợi ích như sau:

- Người dùng có thể thực hiện khảo sát bất cứ khi nào họ muốn.
- Người dùng có thể dành nhiều thời gian hơn để trả lời các câu hỏi, nói cách khác, họ không bị ràng buộc thời gian.
- Khảo sát trực tuyến cho phép người thiết kế đưa ra nhiều hình thức câu hỏi khác nhau để đánh giá nhiều khía cạnh khác nhau của vấn đề cần được khảo sát (câu hỏi hai đáp án, nhiều đáp án, câu hỏi dạng thang điểm,...).

Vì một số ưu điểm nêu trên, chúng tôi quyết định lựa chọn khảo sát trực tuyến như là phương tiện chính để đánh giá sự hài lòng của khách hàng đối với quy trình nghiệp vụ.

### 2.3.1 Nguyên tắc thiết kế bảng khảo sát

#### 2.3.1.1 Nguyên tắc

Một số nguyên tắc khi thiết kế bảng khảo sát được tổng hợp bên dưới như sau:

- Sử dụng từ ngữ đơn giản, quen thuộc, tránh sử dụng các từ ngữ chuyên môn, tiếng lóng, nhiều tầng lớp nghĩa, mơ hồ.

- Sử dụng cấu trúc câu đơn giản.
- Đưa ra các đáp án lựa chọn đầy đủ, toàn diện, và riêng biệt với nhau.
- Tránh những câu hỏi đồi, những câu hỏi phủ định hay phủ định của phủ định.
- Tránh những câu hỏi định hướng người dùng buộc phải chọn một đáp án một cách chủ ý.
- Những câu hỏi mở đầu nên là những câu hỏi dễ, từ đó xây dựng được sự kết nối giữa người thiết kế câu hỏi và người trả lời câu hỏi.
- Các câu hỏi thuộc cùng một chủ đề nên được nhóm lại với nhau.
- Các câu hỏi thuộc cùng một chủ đề nên đi từ chung đến riêng.

### 2.3.1.2 Thang điểm

Việc chọn lựa thang điểm đánh giá cũng là một yếu tố cần được xem xét khi thiết kế bảng câu hỏi khảo sát. Likert (1932) sử dụng thang điểm 5. Osgood, Suci và Tannenbaum (1957) sử dụng thang điểm 7, hay như Thurstone (1928) lại sử dụng thang điểm 11. Không có một quy chuẩn nào cho việc sử dụng thang điểm nào trong khảo sát, dẫu vậy, chúng tôi cho rằng một số thang điểm cụ thể nên được sử dụng để tối ưu hóa dữ liệu thu được từ người dùng. Người dùng khi quan sát thang điểm sẽ thực hiện việc đối chiếu các mức điểm với câu trả lời đưa ra và cố gắng tìm kiếm mức điểm trùng khớp nhất có thể. Vì vậy, có một số điều kiện nhất định cần phải được thỏa mãn.

- Thứ nhất, thang đo cần bao phủ các mức độ nhiều nhất có thể, không có ngoại lệ nào.
- Thứ hai, các mức điểm trong thang đo cần có sự khác biệt với nhau, ý nghĩa của chúng không đan xen trùng lặp nhau.
- Thứ ba, người dùng và người thiết kế câu hỏi phải hiểu được ý nghĩa của mỗi mức điểm và có cách hiểu giống nhau.

Nếu một vài trong những điều kiện trên không được đáp ứng có thể ảnh hưởng đến chất lượng của dữ liệu. Ví dụ, nếu câu trả lời người dùng rơi vào một trường hợp nào đó chưa được liệt kê trong thang điểm, nói cách khác, độ chi tiết của thang đo chưa cao dẫn tới chưa có đáp án thực sự phù hợp với câu trả lời của người dùng lúc đó, dẫn tới họ có thể chọn các đáp án khác nhau ở các thời điểm thực hiện khảo sát khác nhau. Ngoài ra, những đáp án có ý nghĩa tương tự nhau (chẳng hạn như “đôi khi” - “thỉnh thoảng”) sẽ khiến người dùng bối rối khi chọn lựa. Khi đó, giả sử như có nhiều người cùng thực hiện bài khảo sát này, họ có thể chọn những đáp án khác nhau vì họ có cách hiểu khác nhau đối với mỗi đáp án mặc dù câu trả lời của họ đưa ra trong đầu là giống nhau.

### 2.3.2 Yếu tố khảo sát

Một số độ đo độ hài lòng của khách hàng đối với một quy trình nghiệp vụ nói riêng và một dịch vụ, sản phẩm nói chung được áp dụng rộng rãi hiện nay có thể kể đến như: Customer Satisfaction Score (CSAT), Customer Effort Score (CES), Net Promoter Score (NPS).

#### 2.3.2.1 CES

CES được Corporate Executive Board (CEB) giới thiệu vào năm 2010. CES là dạng câu hỏi đánh giá sự hài lòng của người dùng bằng việc đo đạc nỗ lực của người dùng trong việc tương tác với sản phẩm hay dịch vụ, yêu cầu người dùng đánh giá nỗ lực của họ trên thang điểm từ 1 - 7, với 1 là giá trị cao nhất cho sự không đồng thuận với câu hỏi được đặt ra.

Một số công thức tính CES đã được đưa ra, tuy nhiên mỗi công thức sẽ phù hợp với một số những thang điểm nhất định. Ở đây, chúng tôi chọn thang điểm từ 1 - 7, công thức tính thường gặp sẽ lấy số lượng người đồng ý chia cho tổng số phản hồi từ người dùng và nhân với 10 hoặc 100. Người dùng được xem như là đồng ý khi số điểm họ đánh giá cho câu hỏi này nằm trong tập giá trị {5, 6, 7}. Miền giá trị của điểm CES là [0, 1].

$$\text{CES (\%)} = \frac{\text{Number of positive results}}{\text{Total of respondents}} \times 100 \quad [?]$$

Trong đó *Number of positive results* là số lượng người dùng đồng ý với câu hỏi, *Total of respondents* là số lượng người dùng tham gia trả lời câu hỏi. Ví dụ, nếu chúng tôi nhận được 100 phản hồi và 70 trong số đó là phản hồi tích cực, thì điểm CES sẽ là 70%.

#### 2.3.2.2 CSAT

CSAT là một yếu tố đo lường mức độ hài lòng của khách hàng đối với một trải nghiệm cụ thể. CSAT thường được đo trên thang điểm từ 1 - 7, với 1 là rất không hài lòng và 7 là tuyệt đối hài lòng. Điểm CSAT nằm trong khoảng từ 75% đến 85% được xem như là một giá trị tốt. Ta có thể tính điểm CSAT bằng cách lấy số lượng phản hồi hài lòng chia cho tổng số phản hồi từ người dùng và nhân với 100. Người dùng được xem như là hài lòng khi họ chọn đáp án thuộc tập giá trị {5, 6, 7}. Miền giá trị của điểm CSAT là [0, 1].

$$\text{CSAT (\%)} = \frac{\text{Number of satisfied responses}}{\text{Total of respondents}} \times 100 \quad [?]$$

Trong đó, *Number of satisfied responses* là số phản hồi hài lòng từ khách hàng, *Total of respondents* là tổng số phản hồi khảo sát từ người dùng. Ví dụ, nếu chúng tôi nhận được 100 phản hồi và 80 trong số đó là đánh giá hài lòng, điểm CSAT sẽ là 80%.

#### 2.3.2.3 NPS

NPS là câu hỏi được phát triển bởi Reichheld vào năm 2003, đánh giá tỉ lệ người dùng sẽ đề xuất sản phẩm hay dịch vụ cho người khác, trên thang điểm từ 0 - 10, với 0 là giá trị cho biết sản phẩm hay dịch



vụ chắc chắn sẽ không được đề xuất, và 10 là giá trị đảm bảo rằng sản phẩm hay dịch vụ sẽ được đề xuất cho người khác bởi người dùng.

Những phản hồi từ khảo sát NPS được phân loại thành ba nhóm chính: Promoters, Passives và Detractors. *Promoters* là những người dùng có lòng trung thành cao đã đánh giá trải nghiệm của họ từ 9 đến 10. *Passives* là những người trung lập và đánh giá ở mức 7 và 8. Trong khi đó, *Detractors* là những người không hài lòng với sản phẩm và dịch vụ, và chỉ cho số điểm từ 0 đến 6. Điểm NPS được tính bằng tỉ lệ giữa hiệu của số người thuộc nhóm Promoters trừ đi số người thuộc nhóm Detractors trên tổng số người dùng tham gia khảo sát.

$$NPS (\%) = \frac{\text{Promoters} - \text{Detractors}}{\text{Total of respondents}} \times 100 \quad [?]$$

Trong đó, *Promoters* là số người thuộc nhóm *Promoters*, *Detractors* là số người thuộc nhóm *Detractors*, *Total of respondents* là tổng số phản hồi khảo sát từ người dùng.

Như vậy có thể thấy, điểm NPS sẽ nằm trong khoảng giá trị từ -100% đến 100%, hay  $NPS \leq |1|$ . Theo Reichheld và Markey, giá trị NPS từ -100% đến 0 sẽ là không hài lòng, với -100% đến -50% là *deficient*, và từ -49% đến 0 là *insufficient*. *Deficient* nghĩa là chất lượng của dịch vụ được đánh giá là cực kỳ tệ. *Insufficient* cho thấy người dùng đánh giá chất lượng không tốt lắm. Ngược lại, kết quả rơi vào khoảng giá trị từ 0 đến 100% sẽ là mức độ hài lòng, với 0 đến 49% và *sufficient* và 50% đến 100% là *excellent*. *Sufficient* được hiểu rằng chất lượng của dịch vụ nhận được phản hồi khá tích cực, trong khi đó, *excellent* cho thấy người dùng đánh giá rất cao.

### 2.3.3 Phương pháp MUSA

Một vấn đề đặt ra, đó bảng khảo sát đo đặc giá trị CES, CSAT, NPS không chỉ trên một yếu tố của quy trình, mà còn trên nhiều yếu tố khác, chẳng hạn như khả năng xử lý các ngoại lệ và độ linh hoạt của quy trình, cần phải tổng hợp các giá trị đó thành một giá trị đo lường chung. Bên cạnh đó, tùy vào mỗi loại quy trình nghiệp vụ, các yếu tố được đánh giá có thể có những trọng số khác nhau, phụ thuộc vào người thiết kế quy trình ưu tiên yếu tố nào hơn. Chẳng hạn, đối với một quy trình A, người thiết kế xem trọng việc xử lý những ngoại lệ hơn hẳn tính linh hoạt của quy trình, dẫn đến việc trong bảng khảo sát, một điều tất yếu là điểm CES của xử lý ngoại lệ có độ ưu tiên cao hơn điểm CES của tính linh hoạt của quy trình. Điều này cũng sẽ ảnh hưởng đến điểm tổng CES của cả cuộc khảo sát.

Một bài nghiên cứu về đánh giá sự hài lòng của khách hàng trong lĩnh vực ngân hàng tư nhân của ngân hàng Thương mại Hy Lạp vào năm 1999, đánh giá trên nhiều tiêu chí khác nhau:

- Bộ phận nhân sự: Bao gồm các đặc trưng như: kỹ năng và kiến thức chuyên môn, tính trách nhiệm, khả năng giao tiếp và làm việc với khách hàng, sự thân thiện,...
- Sản phẩm: Tiêu chí này tập trung chủ yếu vào đặc điểm các sản phẩm cung cấp cho khách hàng, như độ đa dạng, khả năng bồi thường, giá cả, các dịch vụ đặc biệt,...

- Hình ảnh thương hiệu: Bao gồm tên tuổi, danh tiếng của ngân hàng, khả năng ứng dụng công nghệ và đáp ứng nhu cầu khách hàng trong tương lai.
- Dịch vụ: Liên quan tới những dịch vụ cung cấp cho khách hàng, như thời gian chờ đợi để được xử lý, độ phức tạp của các quy trình, thông tin cung cấp cho khách hàng,...
- Khả năng truy cập: Khả năng mở rộng mạng lưới của ngân hàng, vị trí các chi nhánh, khả năng xử lý những vấn đề có thể xảy ra với hệ thống (ATM bị lỗi).

Để tính toán được giá trị hài lòng cuối cùng (global satisfaction), trước đó cần biết được mức độ hài lòng của khách hàng trên các tiêu chí được liệt kê ở trên (partial satisfaction). Và mỗi tiêu chí lại phụ thuộc vào những đặc trưng bên trong chúng. Các tiêu chí hay đặc trưng của chúng có độ quan trọng khác nhau, đòi hỏi cần phải kết hợp các giá trị này lại với nhau thành một giá trị tổng hoà duy nhất. Tác giả đã đề xuất việc sử dụng phương pháp MUSA để xử lý vấn đề trên, và khi đổi chiều ngược trở về vấn đề xây dựng khảo sát đánh giá sự hài lòng của khách hàng đối với quy trình nghiệp vụ của chúng tôi, chúng tôi nhận thấy có sự tương đồng, bởi chúng tôi cũng khảo sát sự hài lòng trên nhiều yếu tố liên quan đến quy trình nghiệp vụ, chẳng hạn như khả năng xử lý ngoại lệ, thời gian, chi phí thực thi quy trình, sản phẩm đầu ra của quy trình. Chính vì thế, chúng tôi quyết định lựa chọn ứng dụng phương pháp MUSA vào việc tính toán mức độ hài lòng của khách hàng trên nhiều khía cạnh khác nhau của quy trình nghiệp vụ.

Multicriteria Satisfaction Analysis (MUSA) là phương pháp để đo lường và phân tích sự hài lòng của khách hàng. Phương pháp MUSA là một mô hình phân tách mức độ ưu tiên theo những nguyên tắc của phân tích hồi quy thứ tự. Phương pháp luận này sẽ đánh giá mức độ hài lòng của một tập những người dùng dựa trên giá trị của họ và những mức độ ưu tiên. Quá trình kết hợp - phân tách này sẽ được thực thi với ít khả năng xảy ra lỗi nhất. Ưu điểm của phương pháp MUSA là nó hoàn toàn xem xét chất lượng mức độ ưu tiên và đánh giá của người dùng.

Mục đích chính của phương pháp MUSA là kết hợp các đánh giá độc lập thành một hàm thu thập giá trị, giả sử như độ hài lòng tổng thể của người dùng sẽ phụ thuộc vào tập n tiêu chí hay biến số đại diện cho các yếu tố khác nhau của dịch vụ. Tập các tiêu chí này được ký hiệu  $X = (X_1, X_2, \dots, X_n)$  với mỗi tiêu chí cụ thể  $i$  được đại diện bằng một biến đơn  $X_i$ . Bằng cách này, việc đánh giá sự hài lòng của người dùng có thể được xem như một bài toán phân tích nhiều tiêu chí.

Phương pháp MUSA đánh giá độ hài lòng tổng thể và độ hài lòng ở từng yếu tố lần lượt  $Y^*$  và  $X_i^*$ . Cần chú ý rằng phương pháp này tuân theo những nguyên tắc của phân tích hồi quy tuần tự với một số ràng buộc, sử dụng các kỹ thuật quy hoạch tuyến tính (Jacquet-Lagreze and Siskos, 1982; Siskos and Yannacopoulos, 1985; Siskos, 1985). Công thức của phân tích hồi quy tuần tự trình bày như sau:

$$Y^* = \sum_{i=1}^n b_i X_i^* \quad [?]$$

$$\sum_{i=1}^n b_i = 1 \quad [?]$$

Với  $b_i$  là trọng số của tiêu chí thứ  $i$  và giá trị của  $Y^*$  và  $X_i^*$  đã được chuẩn hoá về miền giá trị  $[0, 1]$ . Các hàm dự đoán giá trị là những kết quả quan trọng nhất của phương pháp MUSA, chúng cho thấy giá trị thực trong miền giá trị  $[0, 1]$  là giá trị mà người dùng đánh giá cho mỗi mức độ hài lòng, từng yêu tố hay tổng thể.

### 2.3.4 Tính toán kết quả khảo sát

Như đã đề cập, bài khảo sát sẽ có những câu hỏi để tính toán các giá trị để đánh giá mức độ hài lòng của người dùng đối với một quy trình nghiệp vụ. Các độ đo ở những thang điểm khác nhau, cụ thể CES và CSAT tính trên thang điểm 1 - 7, NPS trên thang điểm 0 - 10. Vì thế, chúng tôi nhận thấy cần phải chuẩn hóa các độ đo về một miền giá trị duy nhất.

Hiện nay có nhiều phương pháp sử dụng để chuẩn hoá dữ liệu, và Min-Max là một trong số đó. Chuẩn hoá Min-Max biểu diễn sự biến đổi tuyến tính trên dữ liệu ban đầu. Biết rằng  $min_a$  và  $max_a$  lần lượt là giá trị nhỏ nhất và giá trị lớn nhất của thuộc tính A. Phương pháp chuẩn hoá Min-Max sẽ ánh xạ một giá trị  $v$  của A thành  $v'$  trong khoảng  $(new - min_a, new - max_a)$  bằng việc tính toán như công thức dưới đây:

$$v' = \left( \frac{v - min_a}{max_a - min_a} \right) \times ((new - max_a) - (new - min_a)) + new - min_a \quad [?]$$

Ở đây chúng tôi chọn một miền giá trị phổ biến là  $[0, 1]$ , và đưa ra công thức chuẩn hoá cụ thể cho từng độ đo như sau, vận dụng phương pháp chuẩn hoá Min-Max được trình bày ở trên:

#### Độ đo CES:

$$\text{Normalized CES (\%)} = \left( \frac{CES - CES_{Min}}{CES_{Max} - CES_{Min}} \right) \times 100$$

Trong đó, CES là giá trị CES tính được từ kết quả khảo sát,  $CES_{Min}$  và  $CES_{Max}$  lần lượt là giá trị nhỏ nhất và giá trị lớn nhất của điểm CES, hay cận dưới và cận trên của thang điểm của độ đo CES. Ví dụ, trên miền giá trị của điểm CES là  $[0, 1]$ , ta có  $CES_{Min} = 0$  và  $CES_{Max} = 1$

#### Độ đo CSAT:

$$\text{Normalized CSAT (\%)} = \left( \frac{CSAT - CSAT_{Min}}{CSAT_{Max} - CSAT_{Min}} \right) \times 100$$

Trong đó, CSAT là giá trị CSAT tính được từ kết quả khảo sát,  $CSAT_{Min}$  và  $CSAT_{Max}$  lần lượt là giá trị nhỏ nhất và giá trị lớn nhất của điểm CSAT, hay cận dưới và cận trên của thang điểm của độ đo CSAT. Ví dụ, trên miền giá trị của điểm CSAT là  $[0, 1]$ , ta có  $CSAT_{Min} = 0$  và  $CSAT_{Max} = 1$

#### Độ đo NPS:

$$\text{Normalized NPS (\%)} = \left( \frac{NPS - NPS_{Min}}{NPS_{Max} - NPS_{Min}} \right) \times 100$$

Trong đó, NPS là giá trị NPS tính được từ kết quả khảo sát,  $NPS_{Min}$  và  $NPS_{Max}$  lần lượt là giá trị nhỏ nhất và giá trị lớn nhất của điểm NPS, hay cận dưới và cận trên của thang điểm của độ đo NPS. Ví dụ, trên miền giá trị của điểm NPS là  $[-1, 1]$ , ta có  $NPS_{Min} = -1$  và  $NPS_{Max} = 1$

Tuỳ vào mỗi loại quy trình nghiệp vụ mà người thiết kế sẽ ưu tiên xem xét độ đo nào chiếm tỉ trọng lớn hơn trong giá trị chung của bảng khảo sát. Vì thế, ứng dụng phương pháp MUSA đã được trình bày



ở trên, chúng tôi đề xuất công thức tính giá trị chung như sau:

$$\text{Survey Score} = (w_{CES} \times \text{Normalized CES}) + (w_{NPS} \times \text{Normalized NPS}) + (w_{CSAT} \times \text{Normalized CSAT})$$

$$w_{CES} + w_{CSAT} + w_{NPS} = 1$$

Trong đó,  $w_{CES}$ ,  $w_{CSAT}$ ,  $w_{NPS}$  lần lượt là trọng số của giá trị CES, NPS và CSAT đã chuẩn hóa. Trọng số này có thể được quy định bởi người thiết kế bảng khảo sát, tuỳ thuộc vào độ ưu tiên của họ đối với độ đo nào cho bảng khảo sát hay quy trình nghiệp vụ.

### 2.3.5 Tính toán giá trị chất lượng của quy trình nghiệp vụ

Như đã đề cập, chất lượng của một quy trình nghiệp vụ bao gồm chất lượng bên ngoài và bên trong. Chúng tôi đã đề xuất phương pháp và công thức tính chất lượng bên ngoài chính là giá trị chung của bài khảo sát, và chất lượng bên trong chúng tôi giữ nguyên cách tính toán của đề tài trước. Người thiết kế quy trình nghiệp vụ vẫn có thể thiết lập mức độ quan trọng của từng loại chất lượng, và với phương pháp MUSA, chúng tôi đề xuất công thức tính giá trị Quality của cả quy trình nghiệp vụ như sau:

$$Q = w_{eq} \times \text{Survey Score} + w_{iq} \times \text{Internal Quality}$$

$$w_{eq} + w_{iq} = 1$$

Trong đó,  $w_{eq}$ ,  $w_{iq}$  lần lượt là trọng số của *external quality* và *internal quality*; Survey Score là điểm của bài khảo sát cũng như là giá trị của *external quality*. Vì 2 giá trị này có chung một miền giá trị  $[0, 1]$  nên không cần phải chuẩn hóa chúng trước khi đưa vào tính toán nữa. Trọng số này có thể được quy định bởi người thiết kế quy trình nghiệp vụ, tuỳ thuộc vào độ ưu tiên của họ đối với yếu tố nào của chất lượng để đánh giá chất lượng tổng quan của quy trình nghiệp vụ.

### 2.3.6 Đề xuất bảng khảo sát

#### 2.3.6.1 Đối tượng khảo sát

Đối tượng được nhắm đến để thực hiện khảo sát bao gồm những người tham gia vào vận hành, thực thi quy trình nghiệp vụ ở những vai trò khác nhau trong quy trình; và những người là khách hàng của quy trình. Như vậy, kết quả chất lượng của quy trình trở nên chính xác và khách quan hơn, vì không chỉ được đánh giá thông qua việc thực thi quy trình trên thực tế, mà còn được đánh giá qua sản phẩm đầu ra của quy trình đó.

#### 2.3.6.2 Nội dung khảo sát

Để đánh giá được toàn diện chất lượng bên ngoài của một quy trình nghiệp vụ, cần phải xem xét đến ba độ đo CES, CSAT, NPS đã được trình bày ở trên. Vì thế, các câu hỏi dùng để thu thập giá trị của những



độ đo này là bắt buộc và người thiết kế khảo sát không thể loại trừ chúng ra khỏi bộ câu hỏi. Bài khảo sát tập trung đánh giá trải nghiệm, sự hài lòng của người dùng nên ở đây, nội dung câu hỏi sẽ xoay quanh các yếu tố liên quan đến trải nghiệm của người dùng, đó là thời gian, chi phí, khả năng xử lý ngoại lệ và độ linh hoạt của quy trình.

Tuy nhiên, tuỳ vào loại quy trình mà có sự xuất hiện của đối tượng khách hàng của quy trình hay không. Chẳng hạn như, những quy trình đóng, phục vụ nội bộ thì có thể không xuất hiện đối tượng khách hàng, hay không có một sản phẩm đầu ra cụ thể nào phục vụ cho khách hàng bên ngoài. Vì vậy, người thiết kế bảng khảo sát cần xác định rõ đối tượng khảo sát để tùy chỉnh nội dung bảng khảo sát phù hợp, tránh xảy ra tình trạng khảo sát không đúng đối tượng, gây nhầm lẫn và sai lệch kết quả thu được.

Dựa vào những nguyên tắc thiết kế bảng câu hỏi đã được trình bày ở trên, chúng tôi quyết định chọn trình tự các câu hỏi như sau:

- Mở đầu bằng những câu hỏi rẽ nhánh, lấy thông tin cơ bản của người dùng, với mục đích lấy kết quả trả lời của người dùng ở những câu hỏi này để điều hướng tới các câu hỏi phù hợp. Thông tin cơ bản của người dùng chỉ bao gồm vai trò cụ thể của người dùng trong quy trình nghiệp vụ đó là gì, là một dạng thông tin công khai.
- Nhóm các câu hỏi chung chủ đề lại với nhau, bao gồm các nhóm câu hỏi về CES, CSAT, NPS tương ứng với ba độ đo đã đề cập, với thứ tự như trên, với mục đích sau khi đã tìm hiểu những nỗ lực của người dùng khi thực thi quy trình, sẽ đánh giá sự hài lòng tổng quan đối với quy trình, và cuối cùng là liệu người dùng có muốn đề xuất quy trình cho người khác hay không. Ở trong mỗi nhóm câu hỏi, sẽ đi từ những câu hỏi đánh giá tổng quan trước (được dùng để tính điểm độ đo), sau đó là những câu hỏi để lấy thêm thông tin xoay quanh lựa chọn của người dùng ở những câu hỏi đánh giá tổng quan đó.

### 2.3.6.3 Hình thức câu hỏi

Chúng tôi sẽ sử dụng một số hình thức câu hỏi khác nhau. Điều này cho phép chúng tôi có thể thu thập nhiều hơn thông tin từ người dùng. Dưới đây là bảng các hình thức và tên câu hỏi được sử dụng trong bài khảo sát:



Hình thức	Mô tả	Ký hiệu
Multiple choice (single select) question	Nhiều lựa chọn, nhưng một lúc chỉ chọn nhiều nhất một đáp án	MC - SS
Open question	Cho phép nhập câu trả lời tùy ý, dưới dạng một đoạn văn ngắn hoặc câu trả lời ngắn	OP
Likert Scale question	Dùng để đánh giá CES, NPS, CSAT với các mức độ khác nhau	LS
Dichotomous	Hai lựa chọn, cùng lúc chỉ chọn một đáp án	DM

Bảng 2.1: Các hình thức câu hỏi được sử dụng trong bảng khảo sát

Chúng tôi cũng đặt tên các câu hỏi theo mục đích của chúng.

Tên	Mô tả	Ký hiệu
CES question	Câu hỏi đánh giá điểm CES	CES
CSAT question	Câu hỏi đánh giá điểm CSAT	CSAT
CES insight question	Câu hỏi thu thập thêm thông tin từ việc đánh giá CES của người dùng	CES - IN
CSAT insight question	Câu hỏi thu thập thêm thông tin từ việc đánh giá CSAT của người dùng	CSAT - IN
NPS question	Câu hỏi đánh giá điểm NPS	NPS
User-information-collecting question	Câu hỏi thu thập thông tin cơ bản của người dùng	UIC
Branching question	Câu hỏi điều kiện, mục đích điều hướng người dùng tới những câu hỏi khác phù hợp với câu trả lời của họ ở câu hỏi này	BR

Bảng 2.2: Tên các câu hỏi được sử dụng trong bảng khảo sát

#### 2.3.6.4 Nội dung câu hỏi

##### Câu hỏi điều kiện:

- Do you use any products or services provided by this process?
- Are there any handled exceptions in your process?
- Variations refer to process' workflows in different conditions. Are there any variations in your process?



Các câu hỏi điều kiện được sử dụng để xác định chi tiết đối tượng của bài khảo sát, liệu trong phần quy trình của họ có xảy ra những ngoại lệ nào đã được xử lý hay không, hay có bất kỳ những biến thể của quy trình hay không. Với những thông tin cung cấp từ người dùng sẽ điều hướng họ tới những câu hỏi sau đó phù hợp với câu trả lời ở những câu hỏi này.

#### **Câu hỏi lấy thông tin cơ bản của người dùng:**

- What is your role in this process?

Câu hỏi này dùng để xác định xem vai trò cụ thể của người dùng (lane) trong quy trình nghiệp vụ là gì, sau khi người dùng được phân loại thuộc nhóm những người tham gia vận hành, thực thi quy trình.

#### **Câu hỏi CES**

- To what extent do you agree with this statement: "It is easy to handle exceptions in your process".
- To what extent do you agree with this statement: "It is easy to execute your process".

Hai câu hỏi CES tập trung đánh giá nỗ lực của người dùng trong việc thực thi quy trình nói chung, và xử lý những ngoại lệ xảy ra trong quá trình thực thi đó. Người dùng đồng ý với mệnh đề ở mức độ nào sẽ chọn giá trị tương ứng trong những lựa chọn được cung cấp sẵn. Ngoài ra, để thu thập thêm thông tin xoay quanh những ngoại lệ xảy ra trong quy trình, chúng tôi cung cấp thêm một số câu hỏi theo sau (follow-up question), nhưng những câu hỏi này không được sử dụng để tính điểm khảo sát:

- How frequently do exceptions happen in practice?
- Which handled exceptions do you think the way to handle them needs optimizing? Can you suggest some solutions to those?
- Can you suggest any exceptions that need handling and the way to handle them?

Các câu hỏi trên tìm hiểu nhiều hơn về việc người dùng nhận thấy những ngoại lệ trên thực tế xảy ra với tần suất như thế nào. Trong số những ngoại lệ đã được xử lý bởi người thiết kế quy trình, người dùng có cảm thấy cách xử lý nào chưa được tối ưu hay vẫn còn một số vấn đề như quá phức tạp, tốn nhiều thời gian không, và họ có thể đề xuất những giải pháp khác. Bên cạnh đó, người dùng có gặp phải những ngoại lệ mà chưa được xử lý không, và trong những trường hợp đó, người dùng đã làm gì để giải quyết tạm thời. Sau khi thu thập được những thông tin này, người thiết kế quy trình nghiệp vụ có thể đưa ra được các phương án để tối ưu quy trình và xử lý các ngoại lệ có thể có.

#### **Câu hỏi CSAT**

- How satisfied are you with the way exceptions are handled in your process?
- A flexible process means it has suitable variations for different conditions. How satisfied are you with those variations of the process?
- How satisfied are you with the amount of time you have to spend on this process?
- How satisfied are you with the cost of products or services of this process?



- How satisfied are you with the products or services provided by this process?
- How satisfied are you with the total time you have to spend on waiting for the output of this process?

Các câu hỏi CSAT tập trung đo lường mức độ hài lòng của người dùng ở cách các ngoại lệ đã được xử lý trong quy trình, việc quy trình có những biến thể để linh hoạt đáp ứng các điều kiện khác nhau xảy ra trong quá trình thực thi, thời gian, chi phí mà người dùng phải bỏ ra để thực thi quy trình, cũng như kết quả đầu ra của quy trình. Người dùng cảm thấy hài lòng tới mức độ nào thì có thể chọn giá trị tương ứng trong những lựa chọn được cung cấp sẵn. Ngoài ra, chúng tôi đề xuất thêm câu hỏi sau để tìm hiểu xem liệu người dùng có gặp vấn đề khi sử dụng sản phẩm hoặc dịch vụ mà quy trình nghiệp cung cấp hay không. Từ đó, người thiết kế quy trình nghiệp vụ có thể đưa ra những giải pháp để nâng cao chất lượng sản phẩm, dịch vụ của quy trình. Và câu hỏi này không được dùng để tính điểm CSAT:

- Do you encounter any problems while using products or services provided by this process?

### Câu hỏi NPS

- How likely is it that you would recommend this process to others?

Để chốt lại bảng khảo sát, câu hỏi NPS sẽ được sử dụng để đánh giá liệu người dùng có cảm thấy quy trình nghiệp vụ xứng đáng để được giới thiệu cho những người khác hay không. Người dùng có thể chọn câu trả lời phù hợp với đánh giá và thái độ của họ.

Để có cái nhìn tổng quan nhất về những câu hỏi trong bảng khảo sát, chúng tôi xin cung cấp bảng tổng hợp các câu hỏi cũng như trình tự các câu hỏi được thể hiện dưới dạng biểu đồ như sau:

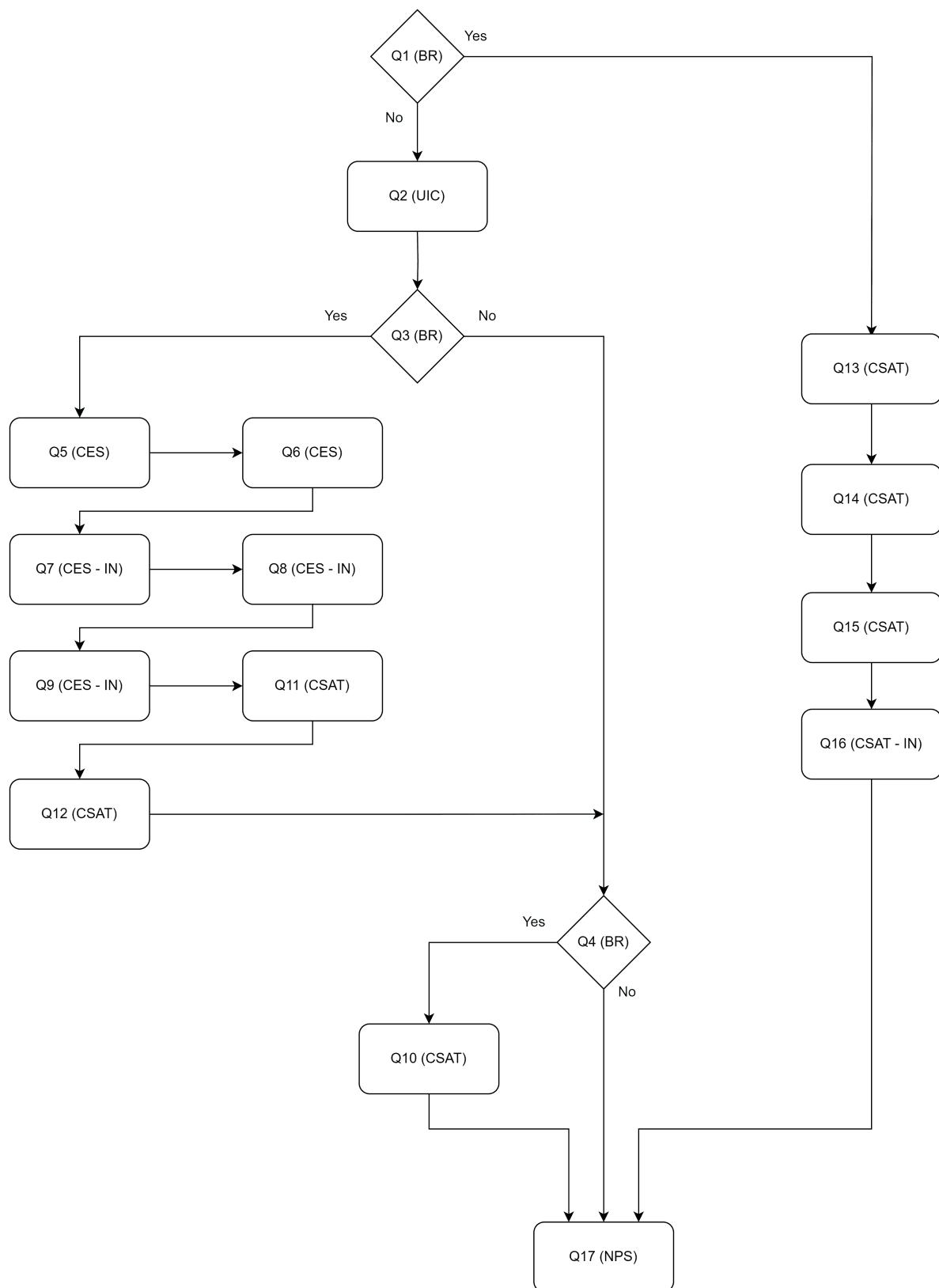


Câu hỏi	Hình thức	Thang điểm	Tên	Nội dung	Bản dịch (tiếng Việt)
Q1	DM		BR	Do you use any products or services provided by this process?	Bạn có sử dụng sản phẩm hay dịch vụ nào được cung cấp bởi quy trình này không?
Q2	MC - SS		UIC	What is your role in this process?	Vai trò của bạn trong quy trình này là gì?
Q3	DM		BR	Are there any handled exceptions in your process?	Trong quy trình của bạn có xảy ra ngoại lệ nào đã được xử lý không?
Q4	DM		BR	Variations refer to process' workflows in different conditions. Are there any variations in your process?	Các biến thể là các luồng thực thi ở những điều kiện khác nhau của quy trình. Trong quy trình của bạn, có xảy ra biến thể nào của quy trình hay không?
Q5	LS	1 - 7	CES	To what extent do you agree with this statement: "It is easy to handle exceptions in your process".	Bạn đồng ý với mệnh đề này tới mức độ nào: "Bạn cảm thấy dễ dàng để xử lý các ngoại lệ trong quy trình nghiệp vụ".
Q6	LS	1 - 7	CES	To what extent do you agree with this statement: "It is easy to execute your process."	Bạn đồng ý với mệnh đề này tới mức độ nào: "Bạn cảm thấy dễ dàng thực thi quy trình nghiệp vụ".
Q7	MC - SS		CES - IN	How frequently do exceptions happen in practice?	Hãy chọn mức độ thường xuyên xảy ra ngoại lệ trên thực tế theo đánh giá của bạn.
Q8	OP		CES - IN	Which handled exceptions do you think the way to handle them needs optimizing? Can you suggest some solutions to those?	Trong số những ngoại lệ đã được xử lý, bạn nghĩ cái nào cách xử lý của nó vẫn chưa được tối ưu? Bạn có thể đề xuất một số cách tối ưu của bạn cho ngoại lệ đó được không?
Q9	OP		CES - IN	Can you suggest any exceptions that need handling and the way to handle them?	Bạn có thể đề xuất một số những ngoại lệ cần phải được xử lý và cách xử lý chúng không?



Q10	LS	1 - 7	CSAT	A flexible process means it has variations for different conditions. How satisfied are you with those variations of the process?	Một quy trình được xem là linh hoạt khi nó có nhiều biến thể cho nhiều điều kiện khác nhau. Bạn hài lòng như thế nào với những biến thể này của quy trình?
Q11	LS	1 - 7	CSAT	How satisfied are you with the way exceptions are handled in your process?	Bạn hài lòng như thế nào với cách mà quy trình nghiệp vụ xử lý các ngoại lệ?
Q12	LS	1 - 7	CSAT	How satisfied are you with the amount of time you have to spend on this process?	Bạn hài lòng như thế nào với tổng thời gian bạn dành cho quy trình?
Q13	LS	1 - 7	CSAT	How satisfied are you with the cost of products or services of this process?	Bạn hài lòng như thế nào với chi phí của sản phẩm, dịch vụ của quy trình?
Q14	LS	1 - 7	CSAT	How satisfied are you with the products or services provided by this process?	Bạn hài lòng như thế nào với sản phẩm, dịch vụ quy trình cung cấp?
Q15	LS	1 - 7	CSAT	How satisfied are you with the total time you have to spend on waiting for the output of this process?	Bạn hài lòng như thế nào với thời gian chờ sản phẩm hay dịch vụ của quy trình?
Q16	OP		CSAT - IN	Do you encounter any problems while using products or services provided by this process?	Bạn có gặp vấn đề gì trong quá trình sử dụng sản phẩm, dịch vụ của quy trình không?
Q17	LS	0 - 10	NPS	How likely is it that you would recommend this process to others?	Bạn có muốn đề xuất quy trình nghiệp vụ này cho người khác không?

**Bảng 2.3:** Tổng hợp các câu hỏi trong bảng khảo sát



**Hình 2.2:** Lược đồ trình tự câu hỏi trong bảng khảo sát

## Chương 3

# Phân tích và thiết kế hệ thống

### 3.1 Giới thiệu hệ thống

Hệ thống BPSky trong giai đoạn trước đã thành công hiện thực bộ công cụ cho phép người dùng thiết kế quy trình dưới dạng mô hình thông qua ngôn ngữ mô hình hóa BPMN và đánh giá chất lượng quy trình một cách tự động. Bên cạnh đó, bộ công cụ trong hệ thống BPSky cũng cung cấp những chức năng khác như hỗ trợ nhập xuất file định dạng .bpmn, ghi chú trên các thành phần và mô hình, các tính năng undo, redo, zoom-in, zoom-out,... Trong quá trình thiết kế, người dùng cũng có thể xác thực mô hình BPMN về mặt ngữ nghĩa cũng như so sánh các phiên bản khác nhau của mô hình.

Trong giai đoạn này, hệ thống sẽ được tập trung cải tiến về phạm vi quản lý và lưu trữ của quy trình nghiệp vụ. Các dự án của người dùng sẽ được mở rộng từ không gian cá nhân thành không gian làm việc chung, cho phép người dùng chia sẻ đến với các thành viên khác trong tổ chức. Khi mở rộng không gian làm việc thì hệ thống cũng sẽ đáp ứng những nhu cầu về quản lý thành viên, quản lý yêu cầu từ thành viên và vấn đề giám sát hiệu suất của quy trình.

Ngoài ra hệ thống cũng mong muốn bổ sung vào tính năng đánh giá chất lượng quy trình bằng việc tích hợp tính năng tạo khảo sát vào bộ công cụ của hệ thống, không chỉ đánh giá chất lượng thông qua mô hình của quy trình mà còn bổ sung những đánh giá từ phía người tham gia vận hành quy trình và sử dụng sản phẩm của quy trình.

Những tính năng trên sẽ được cụ thể hóa thông qua các nhóm chức năng sau:

- Quản lý không gian làm việc chung: Cho phép người dùng tạo không gian làm việc và chia sẻ đến với các người dùng khác trong hệ thống.
- Quản lý thành viên trong không gian làm việc: Quản lý phân quyền và thông tin của những thành viên được mời vào workspace.
- Quản lý yêu cầu được gửi từ thành viên: Các yêu cầu liên quan tới việc phân quyền và mời người dùng khác vào không gian làm việc sẽ được người sở hữu workspace xem xét và phản hồi.

4. Quản lý danh mục quy trình: Cho phép người dùng chỉnh sửa các thông số của quy trình liên quan tới tính khả thi, mức độ ảnh hưởng tới chiến lược tổ chức và hiệu suất của quy trình; từ đó tạo danh mục quy trình cho không gian làm việc để hỗ trợ khả năng giám sát.
5. Quản lý thông báo cá nhân: Cho phép người dùng nhận lời mời thông qua hòm thông báo của người dùng. Ngoài ra cũng có thể nhận các thông báo từ hệ thống theo thời gian thực.
6. Tích hợp khảo sát vào đánh giá chất lượng quy trình: Bên cạnh chất lượng được hệ thống tính toán dựa trên mô hình quy trình nghiệp vụ, người dùng có thể tạo khảo sát cho từng quy trình để thu thập ý kiến của người dùng khác.

Các yêu cầu phi chức năng cũng là yếu tố quan trọng cần được đánh giá, xem xét. Các yêu cầu phi chức năng dưới đây được chúng tôi đảm bảo hệ thống có thể thực hiện được nhằm tối ưu hoá trải nghiệm của người dùng, bên cạnh các yêu cầu phi chức năng kế thừa từ đề tài trước đó:

Hiệu suất	<ul style="list-style-type: none"><li>- Đổi với màn hình input: tối đa 5 trường dữ liệu, không tính toán dữ liệu phức tạp, không tương tác với hệ thống ngoài, có thể lưu trữ dữ liệu trực tiếp ngay xuống database, không lưu trữ các tệp nội dung lớn như hình ảnh, video, tập tin quá 3 MB.</li><li>- Đổi với màn hình output: dữ liệu được truy vấn trực tiếp từ database, hạn chế những câu lệnh truy vấn phức tạp, những truy vấn từ hệ thống ngoài. Hiển thị tối đa 10 dòng dữ liệu, có độ dài nhỏ hơn 100 ký tự</li></ul>
Tính khả dụng	<ul style="list-style-type: none"><li>- Tất cả thông tin quan trọng phải được hiển thị trong 1 màn hình (không cần phải thực hiện thêm thao tác cuộn).</li><li>- Giao diện của hệ thống cần có sự nhất quán, về mặt hình ảnh biểu tượng cũng như vị trí các đối tượng trên màn hình để người dùng làm quen dễ dàng hơn.</li><li>- Người dùng có thể thành thạo các thao tác trên màn hình trong 15 phút sử dụng.</li></ul>
Khả năng hỗ trợ trình duyệt	<ul style="list-style-type: none"><li>- Hệ thống có thể được sử dụng hiệu quả trên các trình duyệt web (Opera, UC Browser, Safari, Microsoft Edge, Google Chrome, Samsung Internet Browser).</li><li>- Hệ thống không chạy trên các phiên bản trình duyệt quá cũ.</li></ul>
Khả năng mở rộng	<ul style="list-style-type: none"><li>- Triển khai cơ sở dữ liệu trên một máy chủ riêng và phần backend của hệ thống trên một máy chủ riêng</li></ul>

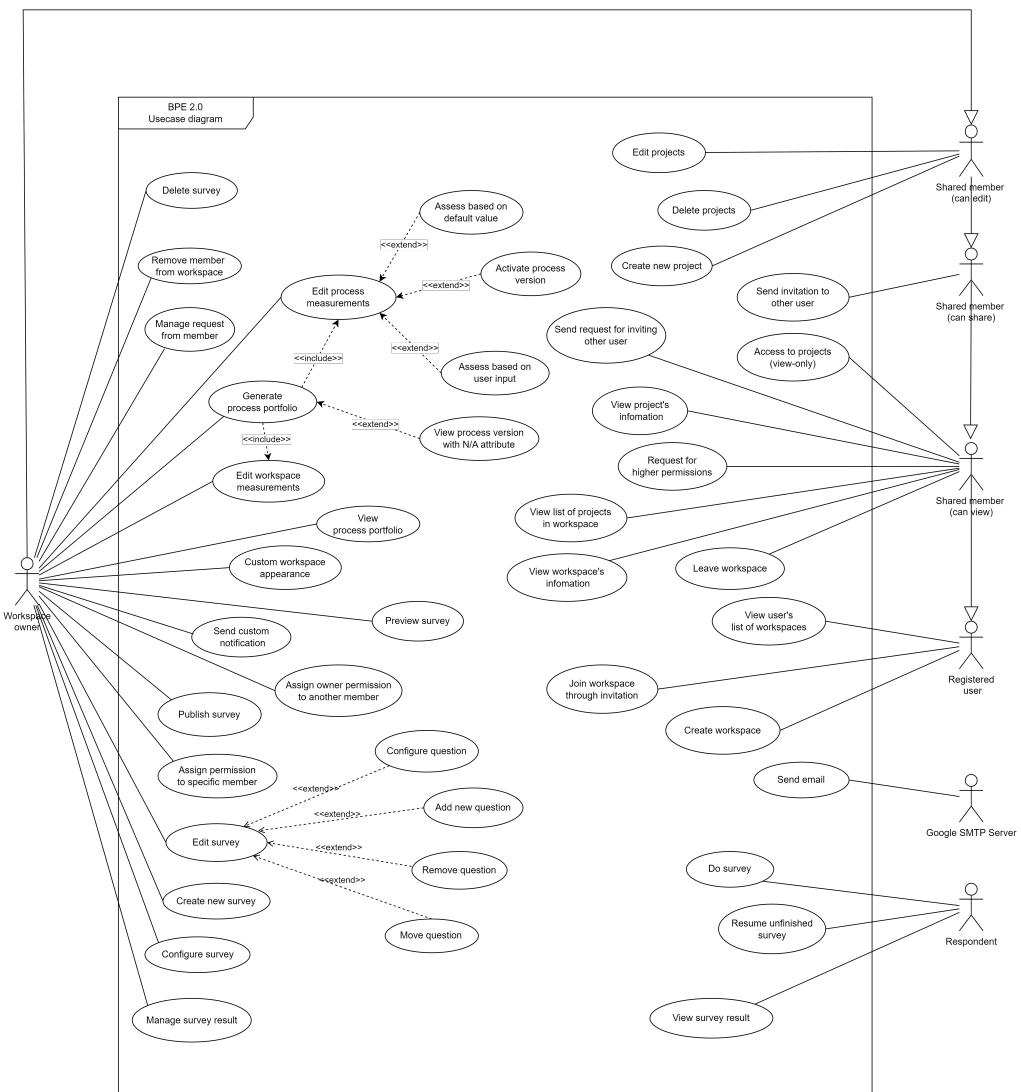
Bảng 3.1: Bảng liệt kê tiêu chí về yêu cầu phi chức năng của hệ thống

## 3.2 Use-case

### 3.2.1 Lược đồ use-case của hệ thống

## Chi tiết về lược đồ use-case của hệ thống:

## Tổng hợp diagram hệ thống BPSky



**Hình 3.1:** Lược đồ use-case hệ thống



### 3.2.2 Gửi lời mời đến người dùng khác vào workspace

<b>Use-case name</b>	Gửi lời mời đến người dùng khác vào workspace
<b>Actor</b>	Thành viên đã tham gia vào workspace
<b>Description</b>	Người dùng gửi lời mời tham gia workspace cho người dùng trong hệ thống.
<b>Trigger</b>	Không
<b>Pre-conditions</b>	Người dùng đã đăng nhập vào hệ thống thành công
<b>Post-conditions</b>	Gửi lời mời người dùng trong hệ thống vào workspace thành công.
<b>Normal Flow</b>	<ol style="list-style-type: none"><li>Người dùng ở trang chủ, hệ thống hiển thị danh sách những workspace mà người dùng tham gia/sở hữu</li><li>Người dùng chọn icon "Menu" ở workspace mà người dùng tham gia với quyền hạn của viewer để hiển thị dropdown menu</li><li>Người dùng chọn nút "Share" để hiện modal chia sẻ workspace</li><li>Người dùng nhập email của người được mời để tìm kiếm người dùng trong hệ thống</li><li>Người dùng chọn quyền hạn của người dùng được mời vào workspace</li><li>Người dùng chọn "Send" để gửi lời mời tới người dùng</li></ol>
<b>Alternative flows</b>	<p>4a. Email của người được mời không tồn tại trên hệ thống</p> <p>4a.1. Hệ thống hiển thị không tìm thấy kết quả phù hợp</p> <p>5a. Người dùng gán quyền hạn cho người được mời vào workspace vượt quá quyền hạn hiện có trong workspace</p> <p>5a.1. Hệ thống thông báo lỗi yêu cầu người dùng gán quyền cho người được mời không vượt quá quyền hạn hiện có (Editor &gt; Sharer &gt; Viewer)</p> <p>5a.2. Tiếp tục ở bước 5</p>
<b>Exceptions</b>	5a. Người dùng chọn "Cancel" để hủy yêu cầu và đóng modal

Bảng 3.2: Use-case scenario cho use-case Gửi lời mời đến người dùng khác vào workspace

### 3.2.3 Gửi yêu cầu mời người dùng khác vào workspace

<b>Use-case name</b>	Gửi yêu cầu mời người dùng khác vào workspace
<b>Actor</b>	Workspace member
<b>Description</b>	Gửi yêu cầu “Chia sẻ workspace đến người khác” tới workspace owner để được xét duyệt
<b>Trigger</b>	Không
<b>Pre-conditions</b>	Người dùng đã đăng ký vào hệ thống và đăng nhập thành công
<b>Post-conditions</b>	Người dùng gửi thành công yêu cầu “Chia sẻ workspace đến người khác” tới workspace owner để được xét duyệt
<b>Normal flow</b>	<ol style="list-style-type: none"> <li>1. Người dùng ở giao diện trang chủ, hệ thống hiển thị danh sách những workspace mà người dùng tham gia/sở hữu</li> <li>2. Người dùng chọn icon "Menu" ở workspace mà người dùng tham gia với quyền hạn của viewer để hiển thị dropdown menu</li> <li>3. Người dùng chọn nút “Share” để hiện modal chia sẻ workspace</li> <li>4. Người dùng nhập email của người được mời để tìm kiếm người dùng trong hệ thống</li> <li>5. Người dùng chọn quyền hạn của người dùng được mời vào workspace</li> <li>6. Người dùng chọn “Send” để gửi yêu cầu đến workspace owner</li> </ol>
<b>Alternative flows</b>	<p>4a. Email của người được mời không tồn tại trên hệ thống</p> <p>4a.1. Hệ thống hiển thị không tìm thấy kết quả phù hợp</p> <p>5a. Người dùng gán quyền hạn cho người được mời vào workspace vượt quá quyền hạn hiện có trong workspace</p> <p>5a.1. Hệ thống thông báo lỗi yêu cầu người dùng gán quyền cho người được mời không vượt quá quyền hạn hiện có (Editor &gt; Sharer &gt; Viewer)</p> <p>5a.2. Tiếp tục ở bước 5</p>
<b>Exceptions</b>	5a. Người dùng chọn “Cancel” để hủy yêu cầu và đóng modal

Bảng 3.3: Use-case scenario cho use-case Gửi yêu cầu mời người dùng khác vào workspace

### 3.2.4 Gửi yêu cầu điều chỉnh quyền hạn trong workspace

<b>Use-case name</b>	Gửi yêu cầu điều chỉnh quyền hạn trong workspace
<b>Actor</b>	Thành viên đã tham gia vào workspace
<b>Description</b>	Gửi yêu cầu “Cung cấp thêm quyền” tới workspace owner để được xét duyệt
<b>Trigger</b>	Không
<b>Pre-conditions</b>	Người dùng đã đăng ký vào hệ thống và đăng nhập thành công
<b>Post-conditions</b>	Người dùng gửi thành công yêu cầu “Cung cấp điều chỉnh quyền” tới workspace owner để được xét duyệt
<b>Normal flow</b>	<ol style="list-style-type: none"> <li>1. Người dùng ở giao diện của workspace đã chọn và đang tham gia vào workspace dưới quyền hạn của viewer hoặc sharer</li> <li>2. Người dùng chọn nút “Create new project”, hệ thống hiện modal thông báo:             <ol style="list-style-type: none"> <li>(a) Người dùng không có quyền hạn để chỉnh sửa nội dung workspace</li> <li>(b) Gửi yêu cầu đến workspace owner để điều chỉnh quyền trong workspace</li> </ol> </li> <li>3. Người dùng chọn nút “Send” để gửi yêu cầu</li> </ol>
<b>Alternative flows</b>	Không
<b>Exceptions</b>	3a. Người dùng chọn “Cancel” để đóng modal

Bảng 3.4: Use-case scenario cho use-case Gửi yêu cầu điều chỉnh quyền hạn trong workspace

### 3.2.5 Tham gia workspace thông qua lời mời

<b>Use-case name</b>	Tham gia workspace thông qua lời mời
<b>Actor</b>	Người dùng có tài khoản trong hệ thống
<b>Description</b>	Tham gia workspace thông qua “Lời mời tham gia workspace”
<b>Trigger</b>	<ol style="list-style-type: none"> <li>1. Thành viên trong workspace với quyền Chia sẻ hoặc cao hơn gửi lời mời trực tiếp đến người dùng trên hệ thống và chưa tồn tại trong workspace</li> <li>2. Workspace owner đồng ý xét duyệt yêu cầu chia sẻ của thành viên không có quyền Chia sẻ trong workspace đến người dùng trên hệ thống</li> <li>3. Workspace owner gửi lời mời tham gia trực tiếp đến người dùng trên hệ thống</li> </ol>
<b>Pre-conditions</b>	Người dùng đã đăng ký vào hệ thống và đăng nhập thành công
<b>Post-conditions</b>	Người dùng thành công tham gia vào workspace
<b>Normal flow</b>	<ol style="list-style-type: none"> <li>1. Người dùng chọn icon “Notification” trên thanh điều hướng</li> <li>2. Hệ thống chuyển hướng đến trang Notification (thông báo) của người dùng</li> <li>3. Hệ thống hiển thị danh sách những thông báo của người dùng</li> <li>4. Người dùng chọn vào thông báo “Lời mời tham gia workspace” để mở modal hiển thị thông tin chi tiết của Lời mời</li> <li>5. Người dùng chọn “Accept” để chấp nhận lời mời tham gia vào workspace</li> </ol>
<b>Alternative flows</b>	6a. Người dùng chọn “Decline” để từ chối lời mời tham gia workspace
<b>Exceptions</b>	6a. Người dùng chọn “Accept” với những lời mời đã hết hiệu lực <ul style="list-style-type: none"> <li>6a.1. Hệ thống hiển thị lỗi khi xử lý yêu cầu của người dùng</li> <li>6a.2. Hệ thống đóng modal hiển thị thông tin Lời mời vào workspace</li> </ul>

Bảng 3.5: Use-case scenario cho use-case Tham gia workspace thông qua lời mời

### 3.2.6 Xét duyệt yêu cầu từ phía người dùng

<b>Use-case name</b>	Xét duyệt yêu cầu từ phía người dùng
<b>Actor</b>	Người sở hữu workspace
<b>Description</b>	Người sở hữu workspace xét duyệt các yêu cầu từ thành viên trong Workspace
<b>Trigger</b>	Không
<b>Pre-conditions</b>	Người dùng đã đăng nhập vào hệ thống, người dùng hiện tại đang ở giao diện workspace mà người dùng sở hữu
<b>Post-conditions</b>	Thành công xét duyệt các yêu cầu từ người dùng trong workspace
<b>Normal Flow</b>	<ol style="list-style-type: none"> <li>1. Người dùng chọn icon hình bánh răng bên cạnh tiêu đề workspace, hệ thống chuyển hướng đến trang quản lý workspace. Mặc định ở giao diện "Members management"</li> <li>2. Người dùng chọn "Requests management" ở thanh sidebar để hệ thống chuyển hướng tới trang quản lý yêu cầu của workspace.</li> <li>3. Hệ thống hiển thị danh sách những yêu cầu từ người dùng trong workspace.</li> <li>4. Người dùng chọn vào yêu cầu để mở modal hiển thị những thông tin của yêu cầu.</li> <li>5. Người dùng chọn nút "Approve" trong modal để chấp nhận yêu cầu tương ứng.</li> </ol>
<b>Alternative Flow</b>	5a. Người dùng chọn "Decline" trong modal để từ chối yêu cầu tương ứng.
<b>Exceptions</b>	5a. Người dùng chọn "Cancel" để hủy thao tác, hệ thống tắt modal xác nhận

Bảng 3.6: Use-case scenario cho use-case Xét duyệt yêu cầu từ phía người dùng

### 3.2.7 Quản lý kết quả thực hiện bảng khảo sát

<b>Use-case name</b>	Quản lý kết quả thực hiện bảng khảo sát
<b>Actor</b>	Người sở hữu Project
<b>Description</b>	Người sở hữu Project quản lý kết quả bảng khảo sát
<b>Trigger</b>	Không
<b>Pre-conditions</b>	Người dùng đã đăng ký vào hệ thống và đăng nhập thành công, hiện tại đang đứng ở giao diện <b>Survey Builder</b>
<b>Post-conditions</b>	Người dùng quản lý kết quả bảng khảo sát thành công
<b>Normal flow</b>	<ol style="list-style-type: none"><li>Người dùng chọn tab <b>Result</b>.</li><li>Hệ thống hiển thị trang kết quả survey.</li><li>Người dùng xem các điểm của survey, số lượng người đã tham gia làm khảo sát, thống kê trả lời mỗi câu hỏi.</li></ol>
<b>Alternative flows</b>	Không
<b>Exceptions</b>	Không

Bảng 3.7: Use-case scenario cho use-case Quản lý kết quả thực hiện bảng khảo sát

### 3.2.8 Công bố bảng khảo sát

<b>Use-case name</b>	Công bố bảng khảo sát
<b>Actor</b>	Người sở hữu Project
<b>Description</b>	Người sở hữu Project công bố bảng khảo sát
<b>Trigger</b>	Không
<b>Pre-conditions</b>	Người dùng đã đăng ký vào hệ thống và đăng nhập thành công, hiện tại đang đứng ở giao diện <b>Survey Builder</b>
<b>Post-conditions</b>	Người dùng công bố bảng khảo sát thành công
<b>Normal flow</b>	<ol style="list-style-type: none"> <li>1. Người dùng chọn nút <b>Publish</b>.</li> <li>2. Hệ thống hiển thị modal.</li> <li>3. Người dùng nhập email cần được gửi bảng khảo sát tới.</li> <li>4. Người dùng chọn biểu tượng dấu cộng để thêm email vào danh sách cần gửi.</li> <li>5. Hệ thống hiển thị những email đã được thêm vào ở bên dưới ô nhập email.</li> <li>6. Người dùng chọn ngày giờ bắt đầu và kết thúc bảng khảo sát.</li> <li>7. Người dùng chọn nút <b>Publish</b>.</li> <li>8. Survey được chuyển sang trạng thái pending. Dựa vào thời gian được thiết lập, hệ thống gửi email tới những người có trong danh sách, sau đó survey được chuyển sang trạng thái published.</li> </ol>
<b>Alternative flows</b>	<p>Người dùng công bố ngay lập tức.</p> <p>6a. Người dùng chọn nút Publish sau khi modal được hiển thị.</p> <p>7a. Hệ thống gửi email ngay lập tức nếu có, survey được chuyển sang trạng thái published.</p>
<b>Exceptions</b>	<p>3c. Người dùng chọn nút Cancel.</p> <p>4c. Hệ thống tắt cửa sổ popup.</p> <p>Usecase kết thúc.</p>

Bảng 3.8: Use-case scenario cho use-case Công bố bảng khảo sát



### 3.2.9 Di chuyển câu hỏi trong bảng khảo sát

<b>Use-case name</b>	Di chuyển câu hỏi trong bảng khảo sát
<b>Actor</b>	Người sở hữu Project
<b>Description</b>	Người sở hữu Project di chuyển câu hỏi trong phần của bảng khảo sát
<b>Trigger</b>	Không
<b>Pre-conditions</b>	Người dùng đã đăng ký vào hệ thống và đăng nhập thành công, hiện tại đang đứng ở giao diện <b>Survey Builder</b>
<b>Post-conditions</b>	Người dùng di chuyển câu hỏi thành công
<b>Normal flow</b>	<ol style="list-style-type: none"><li>Người dùng nhấn giữ 1 câu hỏi trong bảng khảo sát.</li><li>Người dùng kéo thả câu hỏi tới vị trí mới.</li><li>Hệ thống xác nhận thao tác và đặt câu hỏi ở vị trí mới.</li></ol>
<b>Alternative flows</b>	<ol style="list-style-type: none"><li>a. Người dùng chọn 1 câu hỏi trong bảng khảo sát.</li><li>a. Hệ thống hiển thị chi tiết câu hỏi ở sidebar bên cạnh.</li><li>a. Người dùng chọn biểu tượng dấu cộng hoặc trừ ở mục <b>Position in section</b> để di chuyển câu hỏi lên xuống.</li><li>a. Người dùng chọn Apply để xác nhận thay đổi. Tiếp tục Normal Flow ở bước 3.</li></ol>
<b>Exceptions</b>	Không

Bảng 3.9: Use-case scenario cho use-case Di chuyển câu hỏi trong bảng khảo sát



### 3.2.10 Chính sửa thiết lập câu hỏi trong bảng khảo sát

<b>Use-case name</b>	Configure question
<b>Actor</b>	Người sở hữu Project
<b>Description</b>	Người sở hữu Project chỉnh sửa chi tiết câu hỏi trong bảng khảo sát
<b>Trigger</b>	Không
<b>Pre-conditions</b>	Người dùng đã đăng ký vào hệ thống và đăng nhập thành công, hiện tại đang đứng ở giao diện Survey Builder
<b>Post-conditions</b>	Người dùng chỉnh sửa thiết lập câu hỏi thành công
<b>Normal flow</b>	<ol style="list-style-type: none"><li>Người dùng chọn một câu hỏi trong bảng khảo sát.</li><li>Hệ thống hiển thị thông tin câu hỏi ở sidebar bên cạnh.</li><li>Người dùng thay đổi các thông tin trong cột sidebar: Loại câu hỏi, Nội dung câu hỏi, Câu hỏi có bắt buộc hay không,...</li><li>Người dùng chọn Apply.</li><li>Hệ thống lưu lại thay đổi của người dùng.</li></ol>
<b>Alternative flows</b>	Không
<b>Exceptions</b>	Không

Bảng 3.10: Use-case scenario cho use-case Chính sửa thiết lập câu hỏi trong bảng khảo sát



### 3.2.11 Thực hiện bài khảo sát

<b>Use-case name</b>	Thực hiện bài khảo sát
<b>Actor</b>	Khách
<b>Description</b>	Khách thực hiện bài khảo sát
<b>Trigger</b>	Không
<b>Pre-conditions</b>	Không
<b>Post-conditions</b>	Khách thực hiện bài khảo sát thành công
<b>Normal flow</b>	<ol style="list-style-type: none"><li>1. Khách truy cập đường dẫn bài khảo sát.</li><li>2. Hệ thống hiển thị nội dung bài khảo sát.</li><li>3. Khách trả lời các câu hỏi trong bài khảo sát và điền thông tin cá nhân.</li><li>4. Khách chọn nút <b>Submit</b>.</li><li>5. Hệ thống hiển thị thông báo bài khảo sát đã được gửi thành công, đồng thời tính lại điểm hiện tại của bài khảo sát.</li></ol>
<b>Alternative flows</b>	Không
<b>Exceptions</b>	Không

Bảng 3.11: Use-case scenario cho use-case Thực hiện bài khảo sát

### 3.2.12 Chính sửa thông tin của process version

<b>Use-case name</b>	Chỉnh sửa thông tin của process version
<b>Actor</b>	Chủ sở hữu workspace
<b>Description</b>	Thay đổi thông tin của process version như các giá trị health/feasibility/strategic importance của process version
<b>Trigger</b>	Không
<b>Pre-conditions</b>	Người dùng đã đăng nhập vào hệ thống. Người dùng sở hữu một workspace và đang truy cập vào giao diện quản lý workspace (workspace management)
<b>Post-conditions</b>	Người dùng thành công thay đổi giá trị thay đổi giá trị health/feasibility/strategic importance của process version, những giá trị này sẽ được sử dụng để khởi tạo process portfolio.
<b>Normal flow</b>	<ol style="list-style-type: none"> <li>1. Người dùng chọn "Process portfolio" trên sidebar, hệ thống điều hướng tới giao diện quản lý process portfolio.</li> <li>2. Hệ thống hiển thị danh sách project/process/process version có trong workspace.</li> <li>3. Người dùng chọn icon "ba dấu chấm" và chọn "Edit", hệ thống mở modal cho phép người dùng chỉnh sửa giá trị health/feasibility/strategic importance của process version tương ứng.</li> <li>4. Người dùng nhập thông tin được yêu cầu.</li> <li>5. Người dùng chọn nút "Save" để lưu thông tin của process version, hệ thống đóng modal.</li> </ol>
<b>Alternative flows</b>	<ol style="list-style-type: none"> <li>4. Người dùng không nhập thông tin. Người dùng chọn nút "Save" để lưu thông tin về thang đo, hệ thống đóng modal.</li> </ol>
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>3. Người dùng nhập kiểu dữ liệu không hợp lệ             <ol style="list-style-type: none"> <li>a) Người dùng chọn nút "Save" để lưu lại thông tin về thang đo, hệ thống hiện cảnh báo</li> <li>b) Quay lại bước 3</li> </ol> </li> </ol>

Bảng 3.12: Use-case scenario cho use-case Chính sửa thông tin của process version

Tên trường	Mô tả
Current cycle time	Giá trị độ đo thời gian của quy trình hiện tại được người dùng cung cấp
Current cost	Giá trị độ đo chi phí của quy trình hiện tại được người dùng cung cấp
Current quality	Giá trị độ đo chất lượng của quy trình hiện tại được người dùng cung cấp
Current flexibility	Giá trị độ đo linh hoạt của quy trình hiện tại được người dùng cung cấp
Feasibility	Đánh giá tính khả thi của quy trình hiện tại được người dùng cung cấp
Strategic importance	Đánh giá mức độ quan trọng chiến lược của quy trình hiện tại được người dùng cung cấp
Target flexibility	Giá trị độ đo linh hoạt người dùng mong muốn đạt được
Worst flexibility	Giá trị độ đo linh hoạt người dùng tối thiểu cho phép đạt được

Bảng 3.13: Thông tin về yêu cầu đầu vào của use-case *Chỉnh sửa thông tin của process version*

Tên trường	Mô tả
Evaluated cycle time	Giá trị độ đo thời gian của quy trình hiện tại được hệ thống tính toán dựa trên mô hình của quy trình
Evaluated cost	Giá trị độ đo chi phí của quy trình hiện tại được hệ thống tính toán dựa trên mô hình của quy trình
Evaluated quality	Giá trị độ đo chất lượng của quy trình hiện tại được hệ thống tính toán dựa trên mô hình của quy trình
Evaluated flexibility	Giá trị độ đo linh hoạt của quy trình hiện tại được hệ thống tính toán dựa trên mô hình của quy trình

Bảng 3.14: Thông tin hệ thống cung cấp của use-case *Chỉnh sửa thông tin của process version*



### 3.2.13 Chính sửa giá trị thang đo của workspace

<b>Use-case name</b>	Chỉnh sửa giá trị thang đo của workspace
<b>Actor</b>	Chủ sở hữu workspace
<b>Description</b>	Thay đổi giá trị thang đo của workspace
<b>Trigger</b>	Không
<b>Pre-conditions</b>	Người dùng đã đăng nhập vào hệ thống. Người dùng sở hữu một workspace và đang truy cập vào giao diện quản lý workspace (workspace management)
<b>Post-conditions</b>	Người dùng thành công thay đổi giá trị thang đo của workspace, thang đo này áp dụng cho toàn bộ quy trình trong workspace.
<b>Normal flow</b>	<ol style="list-style-type: none"><li>Người dùng chọn "Process portfolio" trên sidebar, hệ thống điều hướng tới giao diện quản lý process portfolio.</li><li>Người dùng chọn nút "Performance level", hệ thống mở modal "Configure performance level".</li><li>Người dùng nhập thông tin được yêu cầu.</li><li>Người dùng chọn nút "Save" để lưu thông tin về thang đo, hệ thống đóng modal.</li></ol>
<b>Alternative flows</b>	<ol style="list-style-type: none"><li>Người dùng không nhập thông tin, chọn nút "Save" để lưu thông tin về thang đo, hệ thống đóng modal.</li></ol>
<b>Exceptions</b>	<ol style="list-style-type: none"><li>Người dùng nhập kiểu dữ liệu không hợp lệ<ol style="list-style-type: none"><li>Người dùng chọn nút "Save" để lưu lại thông tin về thang đo, hệ thống hiện cảnh báo</li><li>Quay lại bước 2</li></ol></li><li>Người dùng nhập giá trị Target cycle time/cost lớn hơn giá trị Worst cycle time/cost hoặc nhập giá trị Target quality/flexibility nhỏ hơn Worst quality/flexibility<ol style="list-style-type: none"><li>Người dùng chọn nút "Save" để lưu lại thông tin về thang đo, hệ thống hiện cảnh báo</li><li>Quay lại bước 2</li></ol></li></ol>

Bảng 3.15: Use-case scenario cho use-case Chính sửa giá trị thang đo của workspace

Tên trường	Mô tả
Target cycle time	Giá trị độ đo thời gian người dùng mong muốn đạt được
Worst cycle time	Giá trị độ đo thời gian người dùng tối đa cho phép đạt được
Target cost	Giá trị độ đo chi phí người dùng mong muốn đạt được
Worst cost	Giá trị độ đo chi phí người dùng tối đa cho phép đạt được
Target quality	Giá trị độ đo chất lượng người dùng mong muốn đạt được
Worst quality	Giá trị độ đo chất lượng người dùng tối thiểu cho phép đạt được
Target flexibility	Giá trị độ đo linh hoạt người dùng mong muốn đạt được
Worst flexibility	Giá trị độ đo linh hoạt người dùng tối thiểu cho phép đạt được

**Bảng 3.16:** Thông tin về yêu cầu đầu vào của use-case Chính sửa giá trị thang đo của workspace



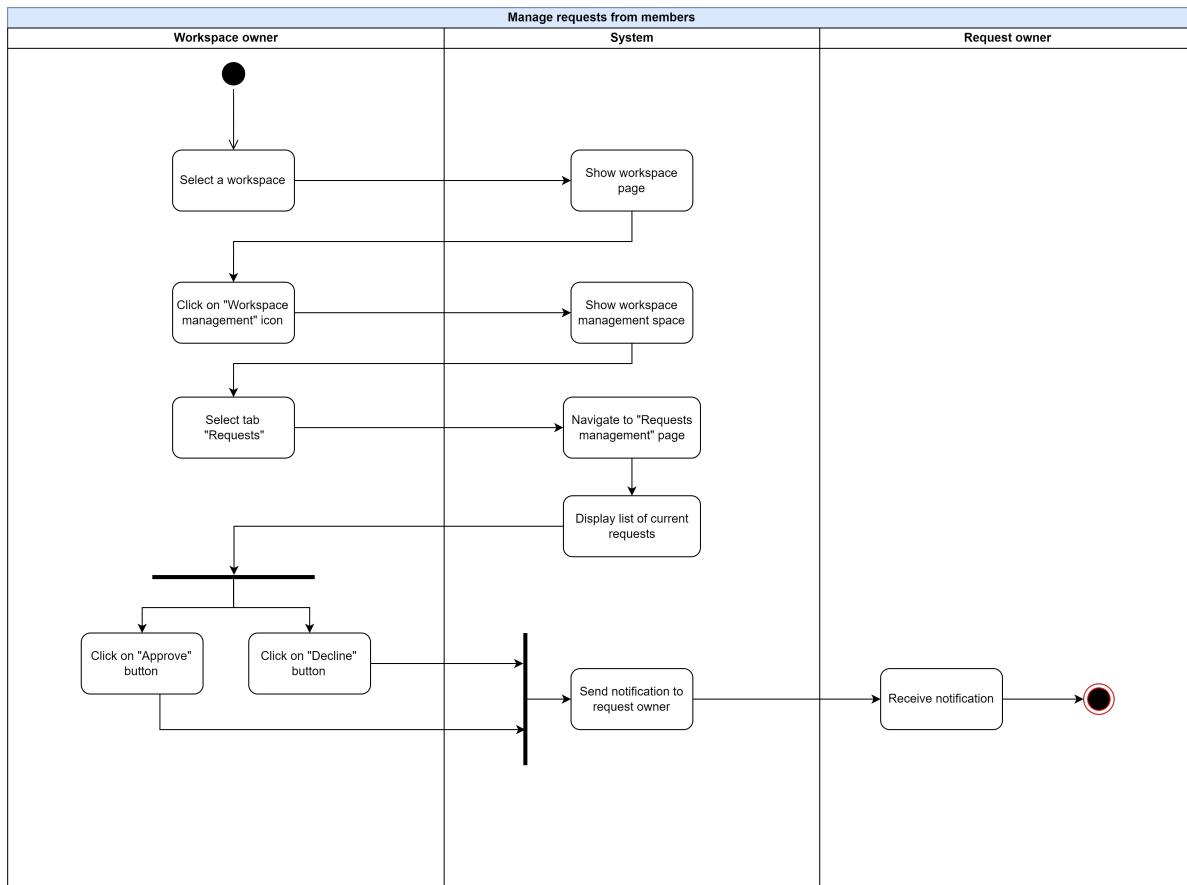
### 3.2.14 Khởi tạo process portfolio

<b>Use-case name</b>	Khởi tạo process portfolio
<b>Actor</b>	Chủ sở hữu workspace
<b>Description</b>	Khởi tạo process portfolio cho workspace
<b>Trigger</b>	Không
<b>Pre-conditions</b>	Người dùng đã đăng nhập vào hệ thống. Người dùng sở hữu một workspace và đang truy cập vào giao diện quản lý workspace (workspace management)
<b>Post-conditions</b>	Người dùng thành công khởi tạo biểu đồ process portfolio của workspace
<b>Normal flow</b>	<ol style="list-style-type: none"><li>Người dùng chọn “Process portfolio” trên sidebar, hệ thống điều hướng tới giao diện quản lý process portfolio.</li><li>Người dùng chọn nút “Process portfolio”.</li><li>Hệ thống mở modal và khởi tạo process portfolio.</li><li>Người dùng chọn “Close” để đóng modal</li></ol>
<b>Alternative flows</b>	<ol style="list-style-type: none"><li>Trong workspace còn có những active version còn bị khuyết giá trị<ol style="list-style-type: none"><li>Hệ thống hiển thị danh sách những active version còn bị khuyết giá trị (feasibility/health/strategic importance)</li><li>Người dùng chọn icon “ba dấu chấm” và chọn “Edit”, hệ thống mở modal cho phép người dùng</li><li>chỉnh sửa giá trị health/feasibility/strategic importance của process version tương ứng.</li><li>Người dùng nhập thông tin được yêu cầu.</li><li>Người dùng chọn nút “Save” để lưu thông tin của process version, hệ thống đóng modal Edit.</li><li>Quay lại bước 3</li></ol></li></ol>
<b>Exceptions</b>	Không

Bảng 3.17: Use-case scenario cho use-case Khởi tạo process portfolio

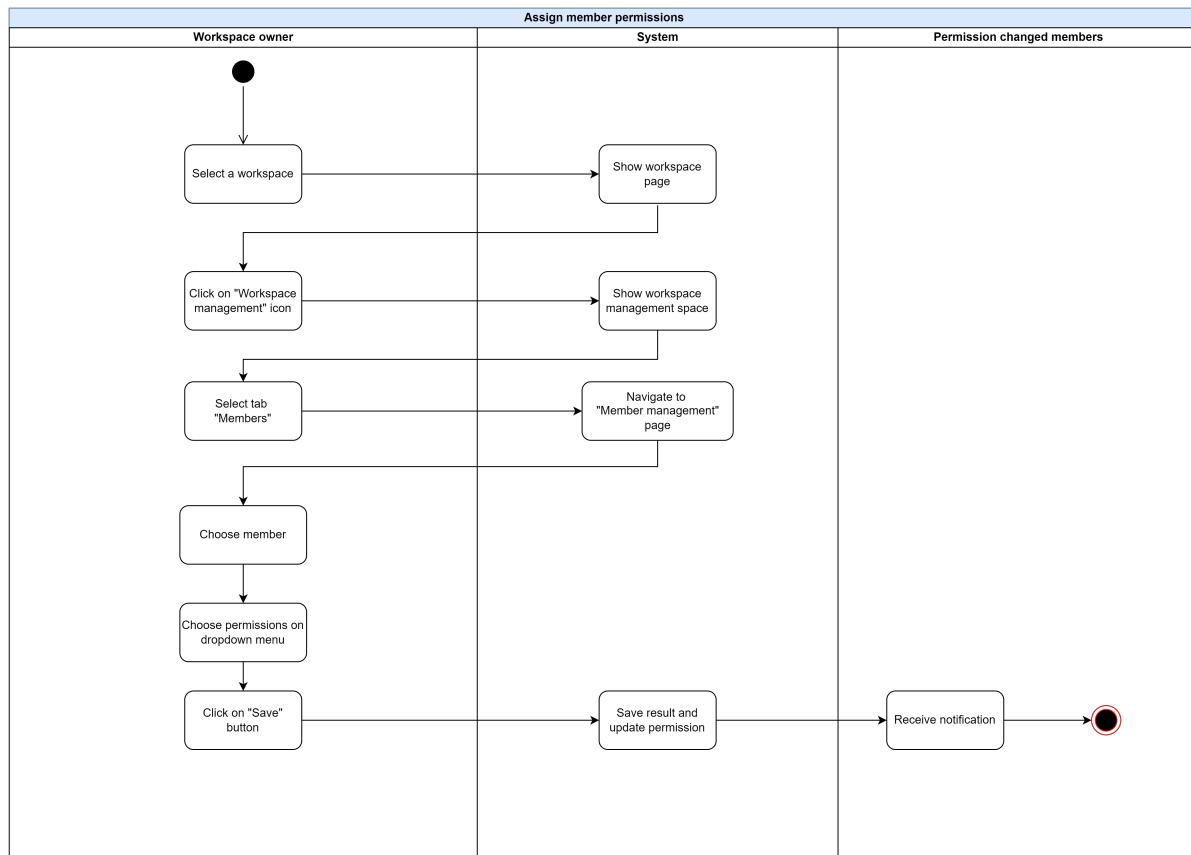
### 3.3 Sơ đồ hoạt động

#### 3.3.1 Quản lý requests trong Workspace



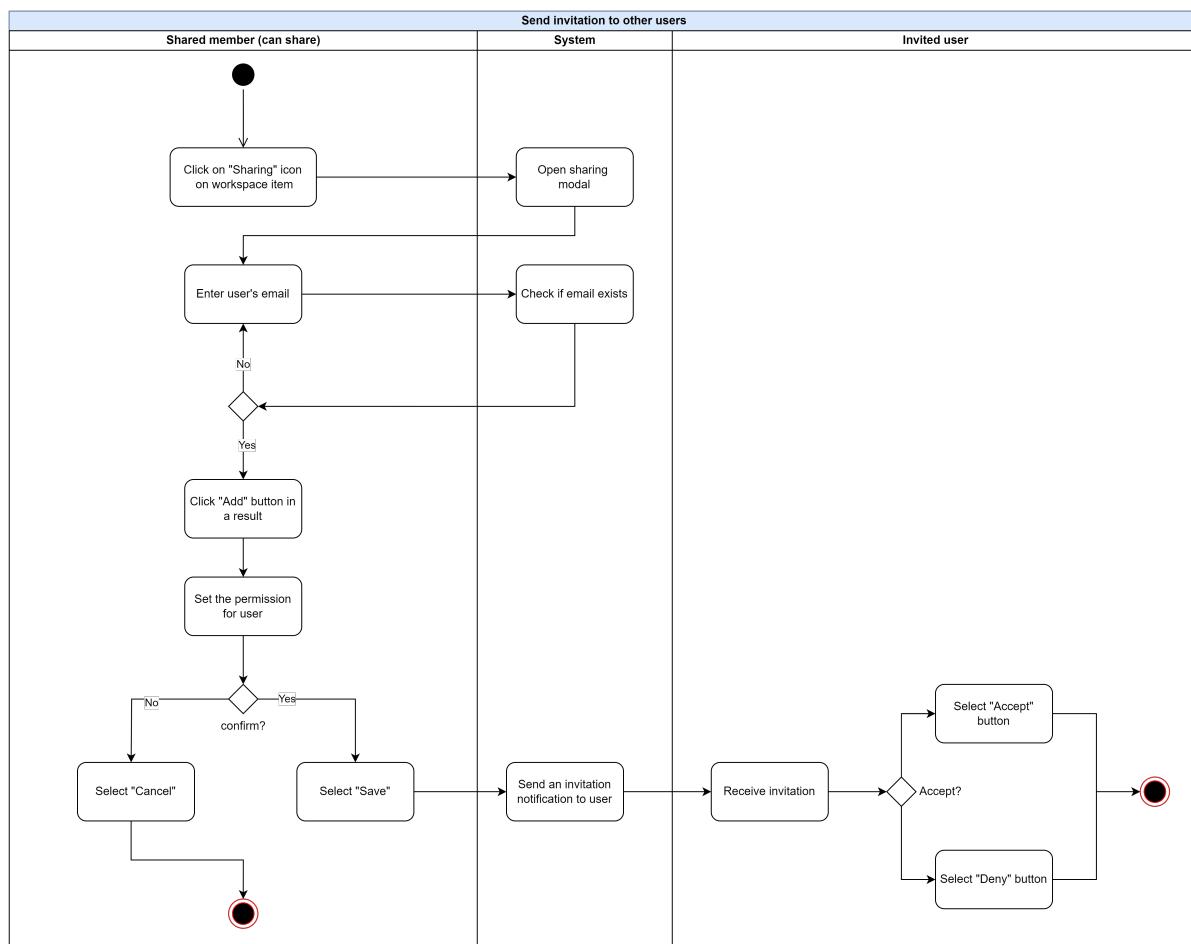
**Hình 3.2: Quản lý requests trong Workspace**

### 3.3.2 Thay đổi quyền của thành viên trong Workspace



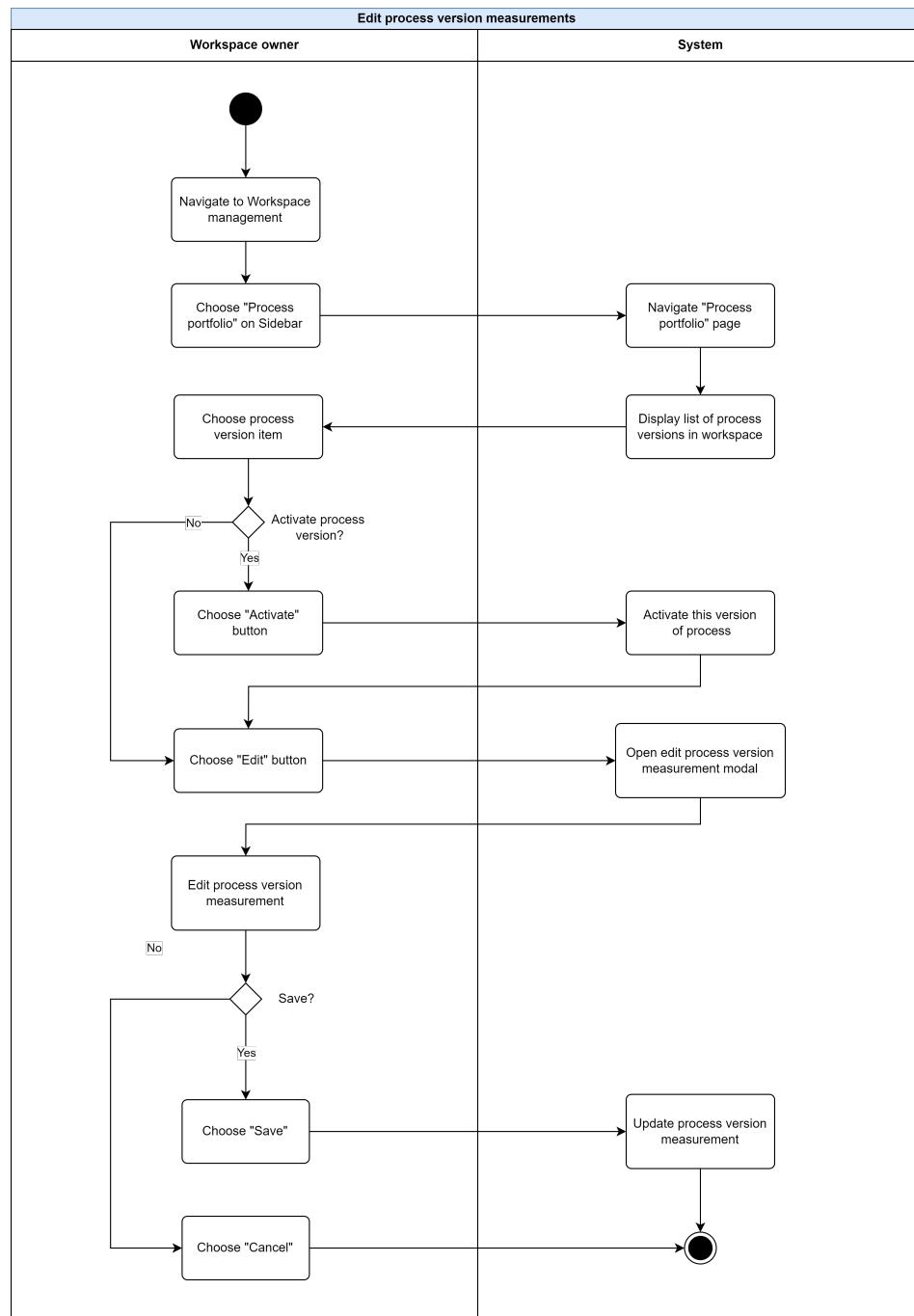
**Hình 3.3: Thay đổi quyền của thành viên trong Workspace**

### 3.3.3 Gửi lời mời người dùng tham gia Workspace



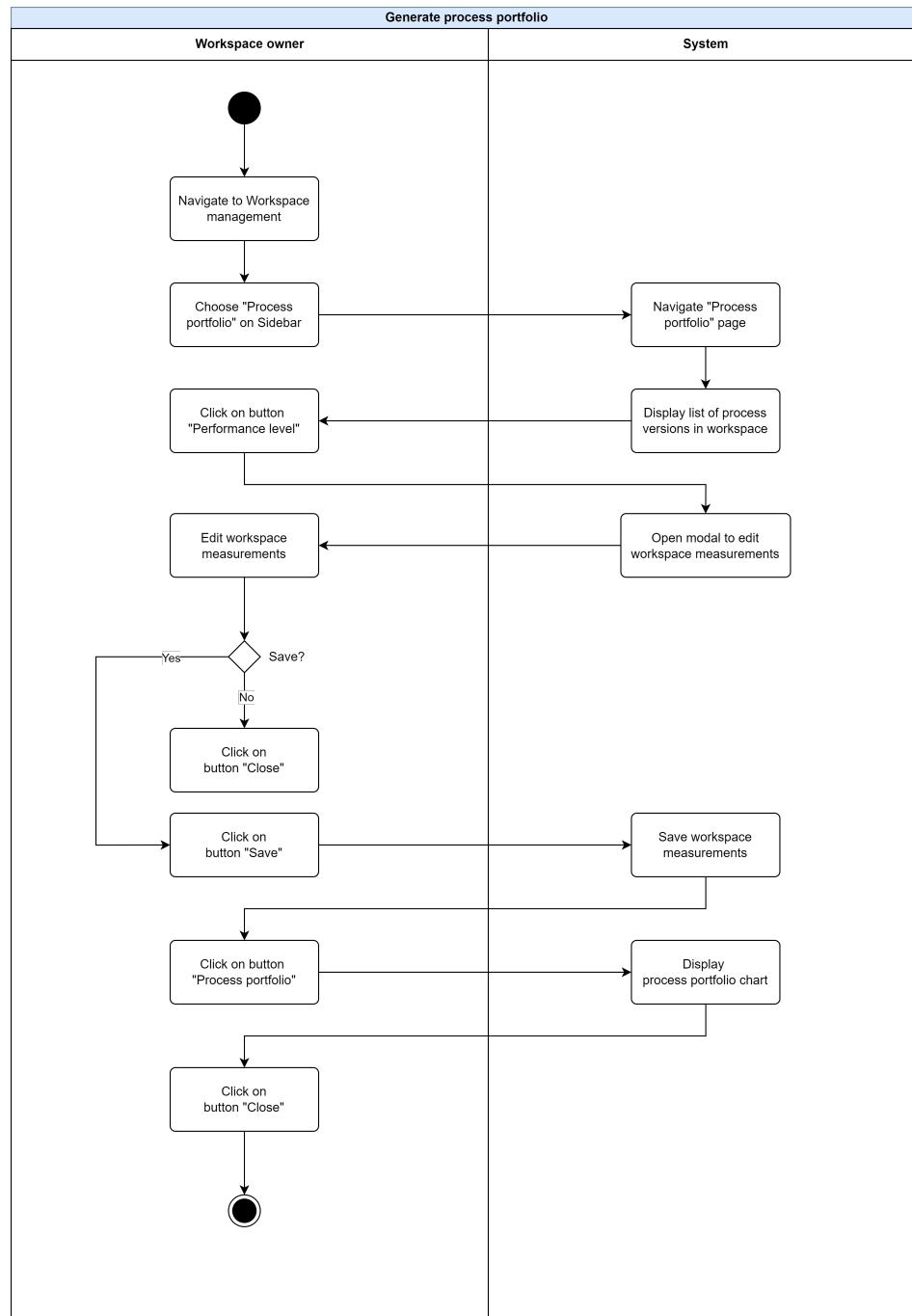
Hình 3.4: Gửi lời mời người dùng tham gia Workspace

### 3.3.4 Chính sửa giá trị của process version



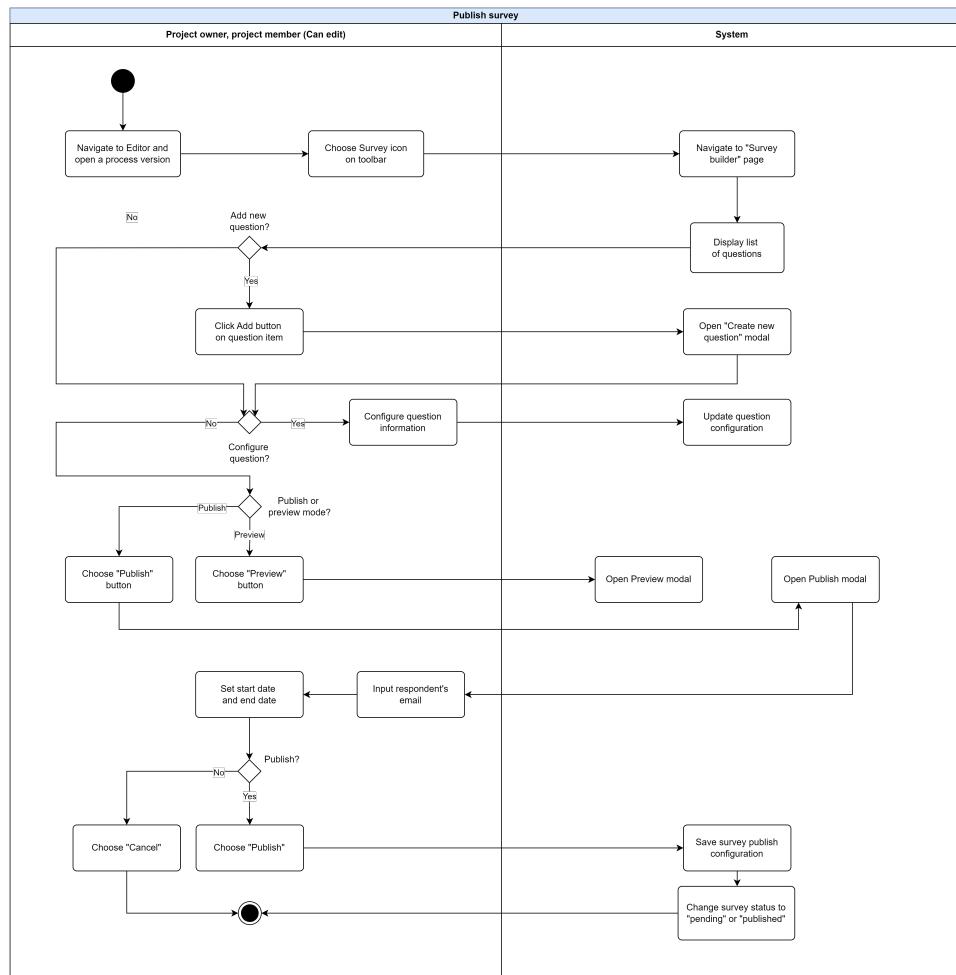
Hình 3.5: Chính sửa giá trị của process version

### 3.3.5 Khởi tạo process portfolio



**Hình 3.6:** Khởi tạo process portfolio

### 3.3.6 Công bố bảng khảo sát



Hình 3.7: Công bố bảng khảo sát

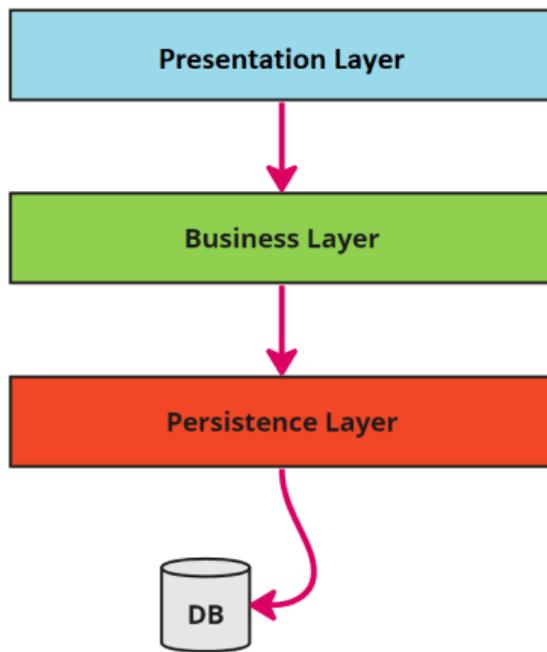
## 3.4 Kiến trúc phần mềm

Ngày nay, có rất nhiều mô hình kiến trúc hệ thống để cho chúng tôi lựa chọn. Tuy nhiên, chúng tôi ưu tiên lựa chọn mô hình kiến trúc phần mềm có khả năng tương thích với hệ thống cũ, độ tin cậy cao, độ mở rộng cao và đơn giản, phù hợp với kích thước của hệ thống. Chúng tôi xem xét hai loại kiến trúc rất phổ biến hiện nay, đó là kiến trúc đa lớp và kiến trúc microservices.

Kiến trúc đa lớp là một trong những kiến trúc phổ biến trong các loại kiến trúc phần mềm. Kiến trúc này ra đời nhằm phân chia các thành phần trong hệ thống, các thành phần cùng chức năng sẽ được nhóm lại với nhau và phân chia công việc cho từng nhóm để dữ liệu không bị chồng chéo và lộn xộn. Kiến trúc này phát huy hiệu quả nhất ở cả hệ thống nhỏ và lớn, giúp cho việc quản lý code và xử lý lỗi dễ dàng hơn.

Mặc dù không có quy định cụ thể về số lượng hay kiểu của các lớp, ở những hệ thống phức tạp hơn có thể có nhiều lớp hơn, đa số các kiến trúc đa lớp gồm có ba lớp chuẩn (3-layer): *Presentation Layer*, *Business Layer*, *Persistence Layer*. *Presentation layer* có nhiệm vụ chính giao tiếp với người dùng, gồm các

thành phần giao diện và thực hiện các công việc như nhập liệu, hiển thị dữ liệu, kiểm tra tính đúng đắn của dữ liệu trước khi gọi lớp *Business layer*. *Business layer* phân ra thành hai nhiệm vụ: thứ nhất, đây là nơi đáp ứng các yêu cầu thao tác dữ liệu của *Presentation layer*, xử lý chính nguồn dữ liệu từ *Presentation layer* trước khi truyền xuống *Persistence layer* và lưu xuống hệ quản trị cơ sở dữ liệu. *Persistence layer* có chức năng giao tiếp với hệ quản trị cơ sở dữ liệu như thực hiện các công việc liên quan đến lưu trữ và truy vấn dữ liệu (tìm kiếm, thêm, xóa, sửa,...).

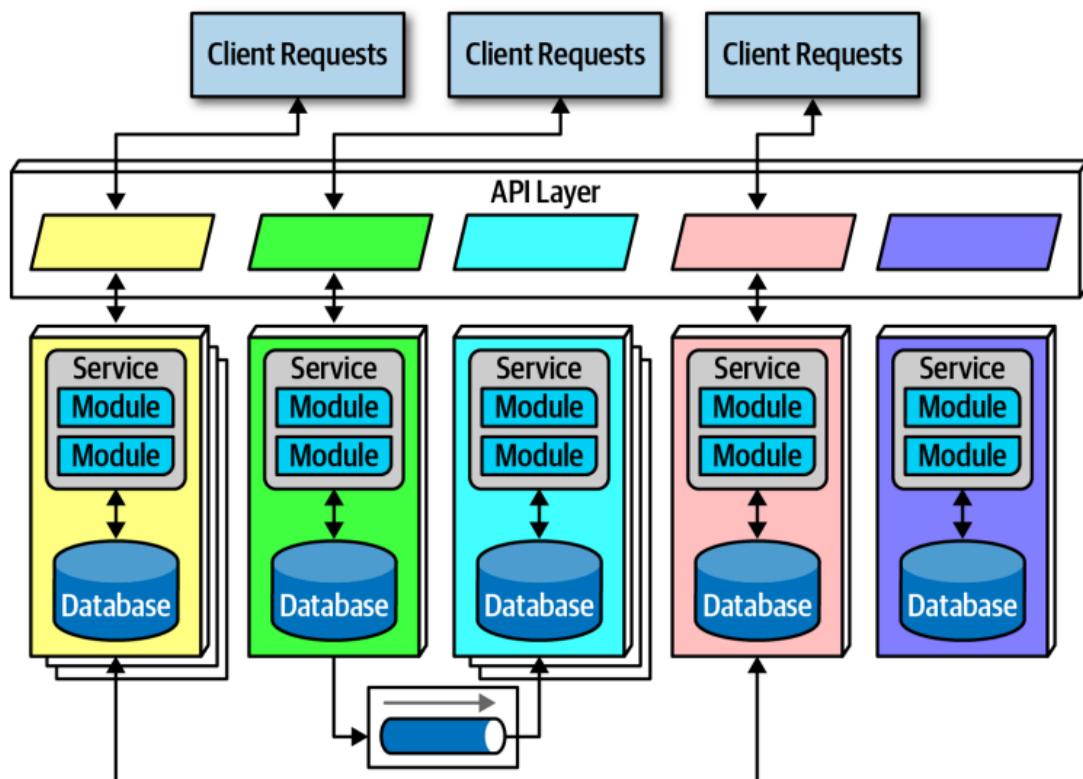


Hình 3.8: Mô hình kiến trúc đa lớp [6]

Kiến trúc này có một số ưu điểm như sau:

- Việc phân chia thành từng lớp giúp cho code được tường minh hơn. Nhờ vào việc chia ra từng lớp đảm nhận các chức năng khác nhau và riêng biệt như giao diện, xử lý, truy vấn thay vì để tất cả lại một chỗ, nhằm giảm sự kết dính.
- Dễ bảo trì khi được phân chia, thì một thành phần của hệ thống sẽ dễ thay đổi. Việc thay đổi này có thể được cài đặt trong 1 lớp, hoặc ảnh hưởng đến lớp gần nhất mà không ảnh hưởng đến cả chương trình.
- Dễ phát triển, tái sử dụng: khi chúng ta muốn thêm một chức năng nào đó thì việc lập trình theo một mô hình sẽ dễ dàng hơn vì chúng ta đã có chuẩn để tuân theo. Và việc sử dụng lại khi có sự thay đổi giữa hai môi trường thì chỉ việc thay đổi lại *Presentation layer*.
- Phù hợp với các hệ thống vừa và nhỏ, với sự tăng trưởng nhỏ và có thể dự đoán được.

Ngoài kiến trúc đa lớp này, chúng tôi còn tìm hiểu thêm một loại kiến trúc cũng rất phổ biến hiện nay, đó là Microservices. Theo kiến trúc này, một ứng dụng được chia thành một bộ các microservice, mỗi microservice thực chất là một service có thể được triển khai và chạy độc lập. Chúng tách biệt về mặt mã nguồn, về hoạt động và dữ liệu. Mỗi microservice có nơi chứa dữ liệu của riêng của nó và chỉ có nó có quyền truy cập vào vùng dữ liệu này. Do các microservice là độc lập, chúng không giao tiếp trực tiếp với nhau mà qua một thành phần trung gian được gọi là API gateway. Có thể thấy vai trò của API gateway rất quan trọng trong mô hình microservice. Nó là điểm đến và đi của mọi yêu cầu hay phản hồi.



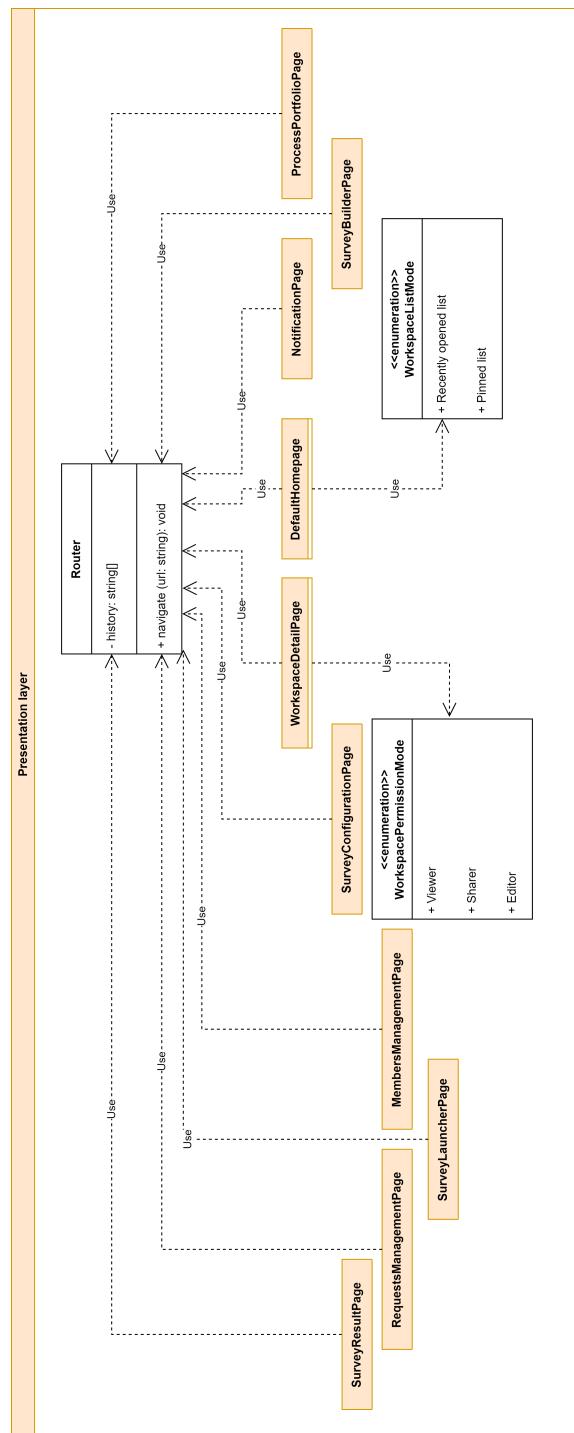
Hình 3.9: Mô hình kiến trúc Microservices [10]

Tính phân tán là đặc trưng của kiến trúc này, vì vậy việc xác định mức độ chi tiết của từng dịch vụ là chìa khóa để xây dựng một hệ thống tốt. Đó là điều khó đạt được đối với hệ thống đang phát triển của chúng tôi. Một vấn đề khác với hướng tiếp cận theo kiến trúc này đó là việc các nghiệp vụ trong hệ thống chúng tôi liên quan khá chặt chẽ, vì vậy kiến trúc Microservices tỏ ra không phù hợp với hệ thống của chúng tôi.

Với những phân tích trên về hai loại kiến trúc, chúng tôi quyết định sử dụng kiến trúc đa lớp cho hệ thống của mình, không chỉ vì tính đơn giản, dễ hiểu mà còn vì nó còn tương thích tốt hơn với hệ thống cũ cũng đang sử dụng loại hình kiến trúc này.

## 3.5 Sơ đồ lớp

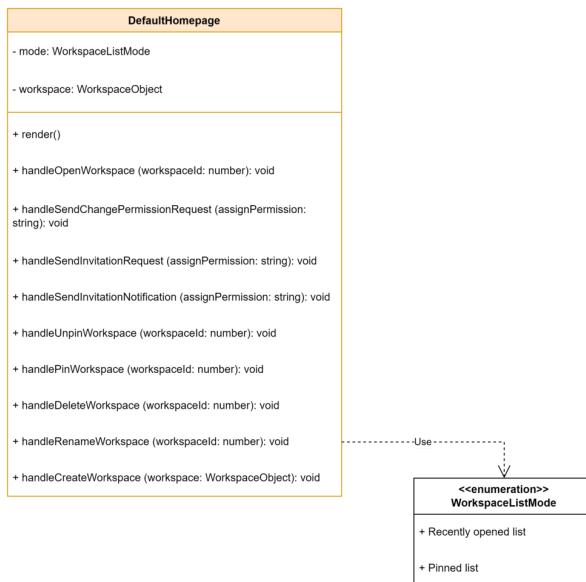
### 3.5.1 Presentation Layer



Hình 3.10: Tổng quan Presentation Layer

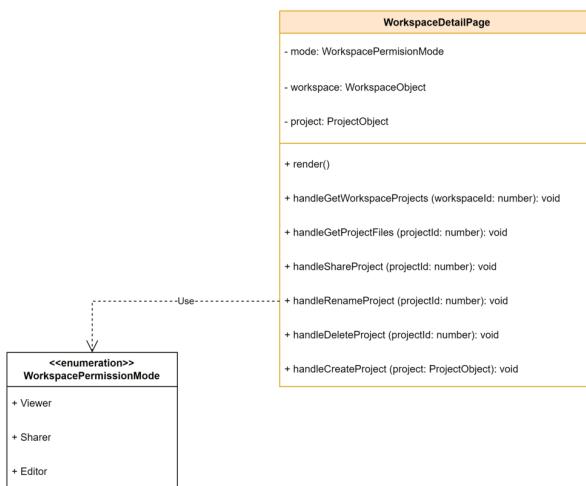
Tầng presentation biểu diễn những giao diện được hiện thực, nó sẽ bao gồm cả những trạng thái (mode) của giao diện cũng như những hàm hiện thực chức năng logic của giao diện tương ứng. Chi tiết

cụ thể hơn sẽ được trình bày trong phần dưới theo từng class được hiện thực.



**Hình 3.11:** Class `DefaultHomepage` trong Presentation layer

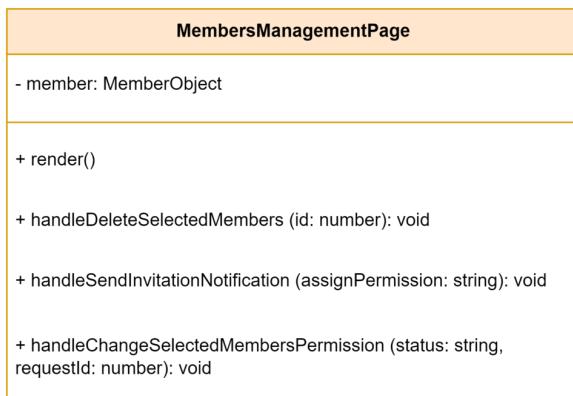
Ở giao diện mặc định khi người dùng được chuyển hướng tới sau khi hoàn tất thủ tục xác thực với hệ thống, chúng ta sẽ tập trung vào những chức năng liên quan tới việc hiển thị và tương tác với workspace mà người dùng tham gia (create, open, pin, delete, rename). Những chức năng đáng chú ý là nhóm chức năng gửi lời mời vào workspace/gửi yêu cầu người dùng khác vào workspace, nhóm chức năng này được xử lý real-time (thời gian thực) để nhận và hiển thị dữ liệu ở những giao diện khác. Giao diện sẽ có hai hình thức hiển thị danh sách workspace là theo thời gian mở workspace gần nhất - "Recently opened workspace" và những workspace được người dùng ghim thủ công - "Pinned workspace".



**Hình 3.12:** Class `WorkspaceDetailPage` trong Presentation layer

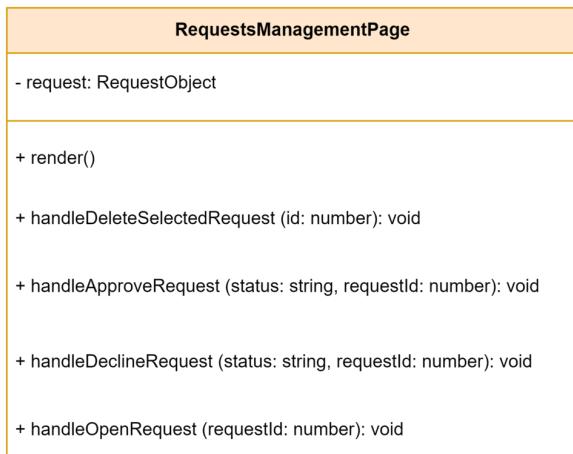
Người dùng sẽ được chuyển hướng tới giao diện hiển thị nội dung bên trong workspace khi sử dụng chức năng "Open workspace" hoặc chọn item workspace tương ứng, nội dung bên trong workspace sẽ được hiển thị dưới dạng danh sách những project có bên trong workspace. Mỗi project sẽ là bao gồm

nhiều process/version của process và document bên trong. Người dùng sẽ được phân quyền hạn khi tham gia vào workspace: viewer, sharer và editor.



**Hình 3.13:** Class *MemberManagementPage* trong *Presentation layer*

Giao diện quản lý thành viên trong workspace là mục mặc định được điều hướng khi người sở hữu workspace chuyển hướng tới giao diện quản lý. Class này sẽ tập trung hiện thực những chức năng liên quan tới thêm/xóa thành viên trong workspace và điều chỉnh quyền hạn của thành viên tương ứng trong workspace. Ngoài ra còn có chức năng gửi lời mời vào workspace cho người dùng khác, chức năng này cũng sẽ được xử lý real-time để hiển thị dữ liệu ở giao diện thông báo cá nhân khi người dùng khác nhận được lời mời.



**Hình 3.14:** Class *RequestManagementPage* trong *Presentation layer*

Giao diện quản lý yêu cầu sẽ hiển thị những yêu cầu nhận được từ thành viên bên trong workspace. Những yêu cầu này sẽ được hiển thị dưới dạng danh sách và được phân loại theo loại yêu cầu và trạng thái xử lý của chúng (pending - chờ xử lý, approved - đồng ý hay declined - từ chối).



NotificationPage	
-	notification: NotificationObject
+	render()
+	handleOpenNotification (notificationId: number): void
+	handleDeleteNotification (notificationId: number): void
+	handleApproveInvitation (notificationId: number, status: string): void
+	handleDeclineInvitation (notificationId: number, status: string): void
+	handleReceiveRealtimeNotification (): void
+	handleSeenNotification (): void
+	handleStarNotification (notificationId: number): void
+	handleUnstarNotification (notificationId: number): void

Hình 3.15: Class NotificationPage trong Presentation layer

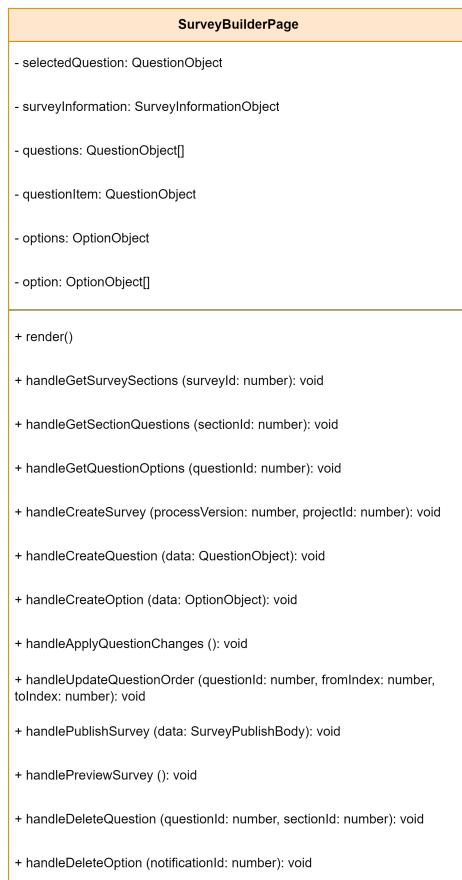
Giao diện thông báo cá nhân là giao diện phụ trách việc hiển thị và giúp người dùng phản hồi những thông báo gửi đến họ. Sẽ có nhiều loại thông báo nhưng trong đó sẽ có thông báo yêu cầu xác nhận từ người dùng (lời mời vào workspace của thành viên trong workspace), hệ thống còn ghi nhận những thông báo đã được đọc và trạng thái của chúng để giúp người dùng có trải nghiệm tốt hơn khi truy xuất những thông báo cần xử lý.

ProcessPortfolioPage	
-	processVersions: ProcessVersionObject[]
-	processVersion: ProcessVersionObject
-	processPortfolio: ProcessPortfolioObject
-	workspaceMeasurement: WorkspaceMeasurementObject
-	processVersionMeasurement: ProcessVersionMeasurementObject
+	render()
+	handleGetProcessVersion (workspaceId: number): void
+	handleActivateProcessVersion (processVersion: string): void
+	handleEditProcessVersionMeasurement (data: processVersionMeasurementObject): void
+	handleEditWorkspaceMeasurement (data: workspaceMeasurement): void
+	handleGetProcessPortfolio (): void
+	handleEditNAProcessVersion (): void
+	handleGetWorkspaceMeasurement (workspaceId: number): void
+	handleGetProcessVersionMeasurement (processVersion: string): void

Hình 3.16: Class ProcessPortfolioPage trong Presentation layer

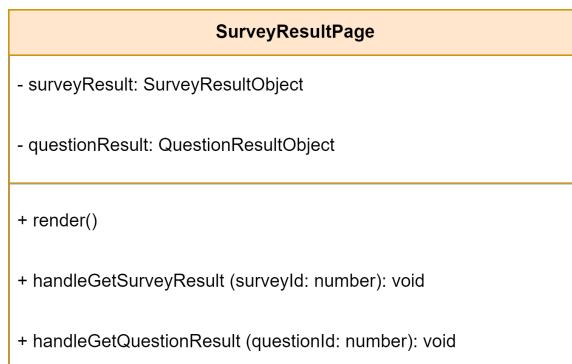
Giao diện quản lý danh mục quy trình (process portfolio), người dùng có thể xem danh sách của những phiên bản quy trình đang hoạt động trong quy trình. Để hệ thống tạo danh mục quy trình, người dùng cần cung cấp đủ hai thông tin: thông tin về thang đo được dùng cho quá trình đánh giá hiệu suất của quy trình và thông tin của quy trình thực tế. Thang đo hỗ trợ hệ thống quy đổi các đại lượng khác nhau về cùng một thang đo chung khi đánh giá hiệu suất của quy trình, ngoài ra cũng cần cung cấp các

thông tin của quy trình về tính khả thi khi vận hành quy trình và mức độ ảnh hưởng đến chiến lược hoạt động của doanh nghiệp.



**Hình 3.17:** Class SurveyBuilderPage trong Presentation layer

Giao diện chỉnh sửa khảo sát sẽ bao gồm 2 phần là khu vực để hiển thị danh sách những câu hỏi có trong khảo sát và khu vực để chỉnh sửa thuộc tính của câu hỏi, bao gồm nội dung câu hỏi, các lựa chọn (nếu câu hỏi thuộc loại multiple choice), cài đặt yêu cầu người dùng bắt buộc phải trả lời câu hỏi hay không, cài đặt trọng số của câu hỏi. Ngoài ra, người dùng có thể thay đổi vị trí của hỏi trong khảo sát thông qua thao tác kéo thả.



**Hình 3.18:** Class SurveyResultPage trong Presentation layer



Giao diện hiển thị kết quả khảo sát sẽ hiển thị kết quả tổng hợp của khảo sát và kết quả của từng câu hỏi. Kết quả sẽ được biểu diễn thông qua biểu đồ cột và tròn tương ứng với loại câu hỏi.

SurveyConfigurationPage	
- responseConfiguration:	SurveyResponseConfigurationObject
- generalConfiguration:	SurveyGeneralConfigurationObject
+ render()	
+ handleGetSurveyConfiguration	(surveyId: number): void
+ handleUpdateSurveyResponseConfiguration	(surveyId: number): void
+ handleUpdateSurveyGeneralConfiguration	(surveyId: number): void
+ handleCloseSurvey	(surveyId: number): void
+ handleDeleteSurvey	(surveyId): void
+ handleChangeSurveyStartDate	(surveyId: number, startDate: Date): void
+ handleChangeSurveyEndDate	(surveyId: number, endDate: Date): void

Hình 3.19: Class SurveyConfigurationPage trong Presentation layer

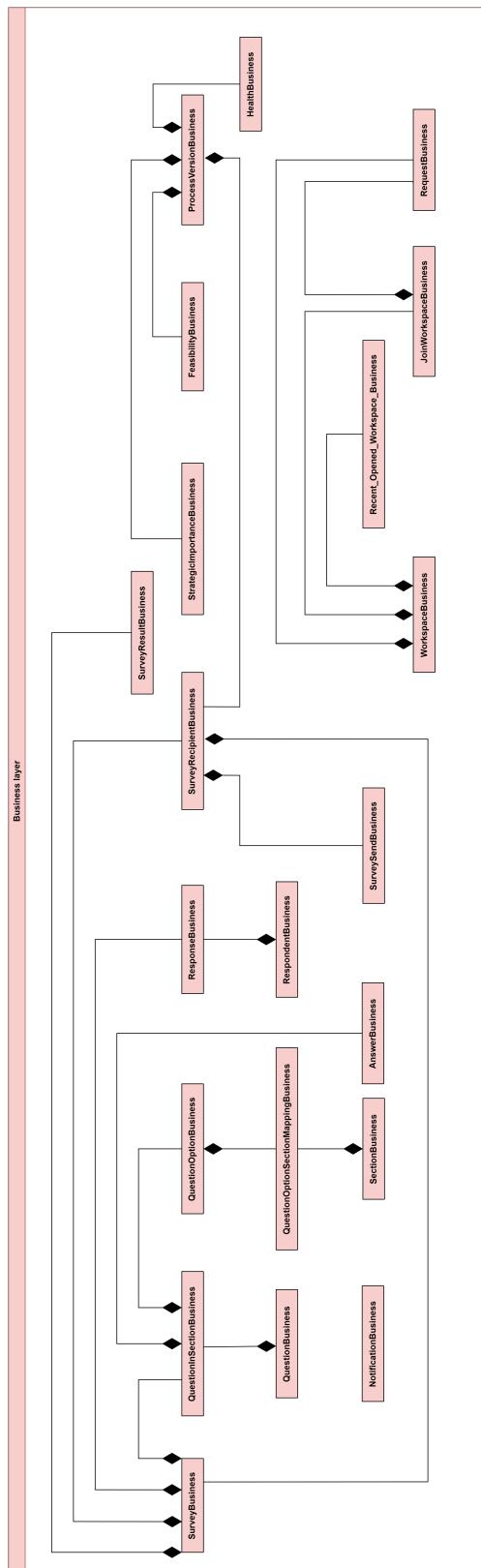
Giao diện chỉnh sửa cấu hình của khảo sát cho phép người dùng cài đặt thời gian công bố/đóng khảo sát, cũng như cài đặt cho phép người dùng thực hiện khảo sát nhiều lần. Ngoài ra, cũng có thể chỉnh sửa tên và mô tả của khảo sát tại đây.

SurveyLauncherPage	
- survey:	SurveyObject
- questions:	QuestionObject[]
- question:	QuestionObject
- options:	OptionObject[]
- option:	OptionObject
- step:	number
+ render()	
+ handleNextStep	(currentStep: number): void
+ handleBackStep	(currentStep: number): void
+ handleSubmitResponse	(surveyId: number, data: OptionObject[], email: string): void

Hình 3.20: Class SurveyLauncherPage trong Presentation layer

Giao diện này cho phép người dùng gửi phản hồi cho khảo sát, người dùng có thể truy cập vào khảo sát thông qua đường dẫn được người tạo khảo sát gửi trực tiếp hoặc thông qua hòm thư điện tử. Khảo sát sẽ chia thành nhiều phần khác nhau, người dùng có thể chọn đến phần tiếp theo hoặc quay trở lại phần trước đó để chỉnh sửa phản hồi của mình. Để gửi khảo sát người dùng cần cung cấp email và tên người dùng để nhận kết quả sau khi khảo sát tổng kết kết quả.

### 3.5.2 Business Layer



**Hình 3.21:** Tổng quan Business Layer



Nhiệm vụ của tầng Business là phân tích những yêu cầu được gửi từ tầng Presentation, xử lý logic nghiệp vụ và gọi tới các đối tượng ở tầng Persistence để tương tác với cơ sở dữ liệu.

WorkspaceBusiness
<ul style="list-style-type: none"><li>- getWorkspace (workspaceId: string) : Workspace</li><li>- getPinned Workspace (userId: string) : list[Workspace]</li><li>- checkWorkspaceOwner (workspaceId: string, ownerId: string): boolean</li><li>- deleteWorkspace (workspaceId: string, deletedAt: timestamp): void</li><li>- getAllWorkspacesByUser(userId: string, page: integer, limit: integer, openedAt : timestamp, ownerId: integer, keyword: string, pinned: boolean): list[Workspace]</li><li>- updateWorkspaceDescription (workspaceId: string, description: string): void</li><li>- updateWorkspaceName (workspaceId: string, name: string): void</li><li>- changeOwnerShip (workspaceId: string, newOwnerId: string): void</li><li>- updateWorkspaceIcon (workspaceId: string, newIcon: File)</li><li>- insertOpenedWorkspace (workspaceId: string, userId: string, openedAt: timestamp, isDeleted: boolean): Workspace</li><li>- createNewWorkspace (name: string, description: string, createdAt: timestamp, ownerId: string, background: string, icon: string, isPersonal: boolean, isDeleted: boolean): Workspace</li><li>- updateWorkspaceBackground(workspaceId: string, newBackground: File)</li><li>- get_workspace_measurements(workspace_id: int)</li><li>- edit_workspace_measurements(workspace_id: int, targeted_measurements: object[])</li><li>- check_target_worst_values(targeted_measurements: object[])</li></ul>

**Hình 3.22:** Class WorkspaceBusiness trong Business layer

Class WorkspaceBusiness là class chịu trách nhiệm xử lý logic nghiệp vụ liên quan đến Workspace. Class này sẽ thừa kế các class khác như: Workspace\_Get\_Business, Workspace\_Delete\_Business, Workspace\_Update\_Business, Workspace\_Insert\_Business. Những class này sẽ thực hiện các chức năng tương ứng với tên của chúng, như vậy, bằng cách này sẽ giúp cho việc quản lý code dễ dàng hơn, và khi có sự thay đổi trong một chức năng nào đó thì việc thay đổi này sẽ không ảnh hưởng đến các chức năng khác. Ngoài ra, class Workspace\_Insert\_Business còn thừa kế class Recent\_Opened\_Workspace\_Insert\_Business để thực hiện chức năng thêm mới vào bảng Recent\_Opened\_Workspace.



JoinWorkspaceBusiness
- undoDeleteMember (workspaceId: string, memberIdList: list[string]): void
- updateMemberPermission (workspaceId: string, memberIdList: list[string], permission: string): void
- getAllMembers (workspaceId: string, page: integer, limit: integer, permission: string, keyword: string): list[Member]
- checkMemberPermission(workspaceId: string, userId: string, permission: string) : boolean
- getMember (workspaceId: string, memberId: string): Member
- insertNewMember (memberId: string, workspaceId: string, joinedAt: timestamp, permission: string): Member
- deleteMember (workspaceId: string, memberIdList: list[string], deletedAt: timestamp): void

**Hình 3.23:** Class JoinWorkspaceBusiness trong Business layer

Class Join\_Workspace\_Business là class chịu trách nhiệm xử lý logic nghiệp vụ liên quan đến việc người dùng tham gia vào một Workspace. Class này sẽ thừa kế các class khác như: Join\_Workspace\_Get\_Business, Join\_Workspace\_Delete\_Business, Join\_Workspace\_Update\_Business. Những class này sẽ thực hiện các chức năng tương ứng với tên của chúng. Ngoài ra, class Join\_Workspace\_Get\_Business còn kế thừa từ class CheckMemberPermissionBusiness để thực hiện chức năng kiểm tra quyền của người dùng trong một Workspace.

Recent_Opened_Workspace_Business
- pinWorkspace (workspaceId: string, userId: string): void
- getPinned Workspace (userId: string) : list[Workspace]
- openWorkspace (workspaceId: string, userId: string, openedAt: timestamp): void

**Hình 3.24:** Class RecentOpenedWorkspaceBusiness trong Business layer

Class Recent\_Opened\_Workspace\_Business là class chịu trách nhiệm xử lý logic cho nghiệp vụ liên quan đến những workspace được mở gần đây của người dùng. Class này chỉ kế thừa hai class khác: Recent\_Opened\_Workspace\_Get\_Business và Recent\_Opened\_Workspace\_Update\_Business cho những xử lý logic khi lấy danh sách Workspace được đánh dấu và logic mở cũng như đánh dấu một Workspace.



RequestBusiness
<ul style="list-style-type: none"><li>- deleteRequestWhenDeletingUser (workspaceId: string, newMemberList: list[string], leftAt: timestamp): void</li><li>- deleteRequest (workspaceId: string, requestIdList: list[string], deletedAt: timestamp): void</li><li>- approveRequest (workspaceId: string, requestIdList: list[string], handlerId: string): void</li><li>- declineRequest (workspaceId: string, requestIdList: list[string], handlerId: string): list[Request]</li><li>- insertNewRequest (requestType: string, content: string, createdAt: timestamp, status: string, workspaceId: string, senderId: string, recipientId: string, handlerId: string, fr_permission: string, to_permission: string, rcp_permission: string): Request</li><li>- getAllRequests (workspaceId: string, page: integer, limit: integer, status: string, type: string, keyword: string): list[Request]</li><li>- adjustPermission (approvedRequest: Request, createdAt: timestamp): void</li><li>- invitation (approvedRequest: Request, createdAt: timestamp): void</li></ul>

**Hình 3.25:** Class RequestBusiness trong Business layer

Class RequestBusiness là class chịu trách nhiệm xử lý logic nghiệp vụ liên quan đến những yêu cầu trong một Workspace. Class này sẽ thừa kế các class khác như: Request\_Get\_Business, Request\_Delete\_Business, Request\_Update\_Business, Request\_Insert\_Business. Những class này sẽ thực hiện các chức năng tương ứng với tên của chúng. Ngoài ra, class Request\_Update\_Business còn kế thừa từ class Request\_Handling\_Business, class này sẽ thực hiện chức năng xử lý những yêu cầu.

NotificationBusiness
<ul style="list-style-type: none"><li>- insertNewNotification (userId: string, workspaceId: string, createdAt: timestamp, permission: string, content: string, isRead: boolean, isDeleted: boolean, notificationType: string, status: string): Notification</li><li>- getAllNotifications (userId: string, page: integer, limit: integer, notificationType: string, keyword: string, isStarred: boolean): list[Notification]</li><li>- starNotification(notificationId: string, isStarred: boolean): Notification</li><li>- readNotification (notificationId: string): Notification</li><li>- approveNotification (userId: string, status: string, workspaceId: string, joinedAt: timestamp, permission: string, notificationId: string): Notification</li><li>- updateNotificationstatus (notificationId: string, status: string): Notification</li><li>- declineNotification (status: string, workspaceId: string): Notification</li></ul>

**Hình 3.26:** Class NotificationBusiness trong Business layer

Class NotificationBusiness là class chịu trách nhiệm xử lý logic nghiệp vụ liên quan đến những thông báo trong một Workspace. Tương tự như những class trên, class này sẽ thừa kế các class khác như: Notification\_Get\_Business, Notification\_Delete\_Business, Notification\_Update\_Business, Notification\_Insert\_Business.



SurveyBusiness
<ul style="list-style-type: none"><li>- validate_start_date_end_date(start_date: timestamp, end_date: timestamp)</li><li>- get_survey_detail(process_version: string)</li><li>- create_new_survey(project_id: int, user_id: int, process_version: string)</li><li>- get_survey_content(user_id: int, project_id: int, process_version: string)</li><li>- delete_survey(survey_id: int)</li><li>- config_survey_general(survey_id: int, survey_name: string, survey_description: string, nps_weight: float, ces_weight: float, csat_weight: float)</li><li>- config_survey_response(survey_id: int, incomplete_survey_action: string, allow_duplicate_respondent: boolean, send_result_to_respondent: boolean, start_date: timestamp, end_date_timestamp)</li><li>- check_if_survey_exists(process_version: string)</li><li>- get_survey_general_config(survey_id: int)</li><li>- get_survey_response_config(survey_id: int)</li><li>- get_survey_questions_by_section_id(section_id: int)</li><li>- get_section_in_survey(process_version: string, mode: string)</li><li>- get_survey_response_config_some(survey_id: int)</li><li>- publish_survey(process_version: string, survey_url: string, email_list: string[], start_date: timestamp, end_date: timestamp)</li><li>- send_survey_url(email: string, survey_url: string, start_date: timestamp, end_date: timestamp)</li><li>- close_publish_survey(process_version: string)</li><li>- save_recipient_email(survey_id: int, email_list: string[])</li><li>- get_publish_info(process_version: string)</li><li>- set_survey_published(survey_id: int)</li><li>- get_unpublished_survey()</li><li>- set_survey_closed(survey_id: int)</li><li>- get_published_surveys()</li><li>- reset_dates(survey_id: int)</li></ul>

Hình 3.27: Class SurveyBusiness trong Business layer

Class SurveyBusiness chịu trách nhiệm xử lý các logic liên quan đến bảng khảo sát. Một số phương thức xử lý chẳng hạn như, lấy toàn bộ nội dung của bảng khảo sát, gửi email đường dẫn bảng khảo sát tới một số người dùng, công bố bảng khảo sát, chỉnh sửa những thiết lập liên quan đến bảng khảo sát.

SurveyRecipientBusiness
<ul style="list-style-type: none"><li>- save_recipient_email(email_list: string[])</li><li>- check_if_email_exists(email: string)</li></ul>

Hình 3.28: Class SurveyRecipientBusiness trong Business layer

Class SurveyRecipientBusiness dùng để quản lý các logic liên quan đến những người dùng sẽ nhận email bài khảo sát được gửi từ hệ thống. Một số phương thức xử lý như lưu lại tạm thời email của người nhận, kiểm tra xem email đã tồn tại hay chưa.



QuestionInSectionBusiness
- create_sample_questions (section_list_in_survey: object[])
- get_questions_in_survey(section_list_in_survey: object[])
- get_question_detail_in_survey(question_in_section_id: int)
- update_question_detail_in_survey(user_id: int, project_id: int, section_id: int, question_in_section_id: int, question_type: string, is_required: boolean, order_in_section: int, weight: float, content: string, question_options: object[])
- handle_question_options(question_in_section_id: int, new_question_options: object[])
- reorder_question_in_section_when_delete_question(section_id: int, question_in_section_id: int)
- reorder_questions_in_section_when_change_position(section_id: int, question_in_section_id: int, new_order: int)
- delete_question_in_survey(section_id: int, question_in_section_id: int)
- change_question_type(question_in_section_id: int, new_question_type: string)
- add_new_question_to_section(section_id: int, content: string, order_in_section: int, weight: float, is_required: boolean, question_type: string, question_options: object[])
- reorder_questions_in_section_when_add_new_question(section_id: int, new_question_in_section_id: int, order_in_section: int)
- get_questions_in_section(section_id: int)
- delete_questions_in_section(section_id: int)
- get_list_of_questions_in_survey(survey_id)

Hình 3.29: Class QuestionInSectionBusiness trong Business layer

Class QuestionInSectionBusiness sẽ xử lý những logic liên quan đến các câu hỏi có trong bảng khảo sát, chẳng hạn như lấy nội dung của câu hỏi, xóa câu hỏi được chọn và sắp xếp thứ tự những câu hỏi còn lại, thêm câu hỏi mới, thay đổi nội dung, thiết lập của câu hỏi.

QuestionBusiness
- check_if_sample_questions_initialized()
- initialize_sample_questions()
- get_all_questions(project_id: int, user_id: int)

Hình 3.30: Class QuestionBusiness trong Business layer

Class QuestionBusiness xử lý những logic liên quan đến những câu hỏi có sẵn trong hệ thống. Khi survey được khởi tạo, các câu hỏi mẫu có sẵn trong survey sẽ được lấy từ câu hỏi đã có trong hệ thống. Một số phương thức xử lý như khởi tạo câu hỏi mẫu, lấy toàn bộ các câu hỏi, hay kiểm tra xem câu hỏi đã được khởi tạo hay chưa.

QuestionOptionBusiness
- create_sample_question_option(question_list_in_survey: object[])
- create_question_options(question_in_section: object, question_options: object[])
- update_question_option(question_option: object)
- add_new_question_option(question_in_section_id: int, question_option: object)
- delete_question_option(question_option: object)
- delete_question_options(question_in_section_id: int)
- get_question_options_in_question_in_section(question_in_section_id: int)

Hình 3.31: Class QuestionOptionBusiness trong Business layer

Class QuestionOptionBusiness sẽ xử lý những logic liên quan đến các lựa chọn trong mỗi câu hỏi, ứng với các câu hỏi dạng multiple choice. Một số phương thức xử lý như: khởi tạo các lựa chọn, thêm, sửa, xóa các lựa chọn, thay đổi thứ tự các lựa chọn khi người dùng kéo thả.

QuestionOptionSectionMappingBusiness
- create_sample_question_option_section_mapping(question_options: object[], section_lists_in_survey: object[])
- delete_question_option_section_mapping(question_in_section_id: int)

Hình 3.32: Class QuestionOptionSectionMappingBusiness trong Business layer

Class QuestionOptionSectionMappingBusiness sẽ tập trung xử lý những logic liên quan tới việc khi người dùng chọn các đáp án khác nhau sẽ được điều hướng tới những phần khác nhau trong bảng khảo sát. Ở đây có một số phương thức xử lý như khởi tạo việc tham chiếu lựa chọn này tới một phần nào đó trong bảng khảo sát, hay xóa đi mối quan hệ tham chiếu đó.

SectionBusiness
- create_section(survey_id: int, name: string, order_in_survey: int)
- create_sample_sections(survey_id: int)
- get_sections_in_survey(survey_id: int)
- delete_sections(survey_id: int)

Hình 3.33: Class SectionBusiness trong Business layer

Class SectionBusiness sẽ quản lý việc xử lý logic đối với các phần trong bảng khảo sát. Bảng khảo sát được chia thành nhiều phần khác nhau để phân loại các nhóm câu hỏi, hướng tới những nhóm người dùng khác nhau cần được khảo sát. Các phương thức xử lý chẳng hạn như lấy các phần trong bảng khảo sát, xóa phần hay tạo mới phần.



ResponseBusiness
- submit_survey_form(answers: object[], email: string, full_name: string, process_version: string)
- create_response(survey_id: int, respondent_id: int, end_date: timestamp, start_date: timestamp)
- get_number_of_responses(survey_id: int)
- get_survey_response(response_id: int)
- delete_responses(survey_id: int)

**Hình 3.34:** Class ResponseBusiness trong Business layer

Class ResponseBusiness phụ trách những logic liên quan đến việc người dùng thực hiện và gửi bài làm khảo sát về cho hệ thống. Một số hàm xử lý logic chẵng hạn như làm và nộp bài làm khảo sát, lấy số lượng các bài làm hiện có của bài khảo sát,...

RespondentBusiness
- create_respondent(email: string, full_name: string)

**Hình 3.35:** Class RespondentBusiness trong Business layer

Class RespondentBusiness bao gồm logic cho việc tạo ra dữ liệu của người dùng tham gia làm bài khảo sát. Class này có một phương thức xử lý là khởi tạo dữ liệu của người dùng, bao gồm 2 tham số là email và tên đầy đủ của người dùng.

AnswerBusiness
- add_answer(response_id: int, question_id: int, value: string)
- get_all_answers(response_id: int)
- delete_answers(response_id: int)
- get_list_of_answers_for_question(question_id: int)

**Hình 3.36:** Class AnswerBusiness trong Business layer

Class AnswerBusiness gồm những phương thức xử lý logic liên quan đến những câu trả lời của người dùng tương ứng với các câu hỏi trong bảng khảo sát. Một số phương thức xử lý chẵng hạn như lấy toàn bộ các câu trả lời, xóa câu trả lời,...

SurveySendBusiness
- save_survey_recipient_email(survey_id: int, recipient_id: int)
- get_survey_recipient_email(survey_id: int)
- delete_survey_recipient_email(survey_id: int, recipient_id: int)
- get_all_emails_dates_url_of_not_published_survey()
- delete_survey_recipient_emails(survey_id: int)

Hình 3.37: Class SurveySendBusiness trong Business layer

Class SurveySendBusiness sẽ quản lý những logic xử lý về việc gửi bài khảo sát tới những người dùng cụ thể, và một số phương thức để phục vụ cho việc kiểm tra bài khảo sát đã được công bố hay chưa (tính năng tự động công bố hoặc đóng survey theo thời gian đã thiết lập).

SurveyResultBusiness
- create_sample_questions (section_list_in_survey: object[])
- calculate_ces_or_csat_score(survey_id: int, current_number_of_responses: int, question_type: string)
- calculate_ces_score(survey_id: int, current_number_of_responses: int)
- calculate_nps_score(survey_id: int, current_number_of_responses: int)
- calculate_csat_score(survey_id: int, current_number_of_responses: int)
- get_list_of_weight_and_answers_of_question_in_survey(survey_id: int, question_type: string)
- get_survey_result(process_version: string)
- update_scores(survey_id: int, ces_score: float, csat_score: float, nps_score: float, total_score: float)
- create_survey_result(survey_id: int)
- check_if_survey_result_exists(survey_id: int)
- get_answer_details_for_multiple_questions(question_in_section_id: int)
- get_answer_details(process_version: string)
- get_answer_details_for_csat_or_ces_question(question_in_section_id: int)
- get_answer_details_for_nps_question(question_id: int)
- get_answer_details_for_open_question(question_id: int)

Hình 3.38: Class SurveyResultBusiness trong Business layer

Class SurveyResultBusiness gồm những logic xử lý liên quan đến kết quả của bài khảo sát, chẳng hạn như tính toán các độ đo, điểm tổng thể của bài khảo sát, lấy thông tin chi tiết về việc trả lời mỗi câu hỏi của người dùng,...

<b>ProcessVersionBusiness</b>
<ul style="list-style-type: none"> <li>- delete_oldest_version (user_id: int, project_id: int, process_id: int)</li> <li>- delete_version(user_id: int, project_id: int, process_id: int, version: string)</li> <li>- delete_comment(user_id: int, project_id: int, process_id: int, xml_file_link: string, id: int)</li> <li>- get_by_version(user_id: int, project_id: int, process_id: int, version: string)</li> <li>- get_content_by_version(user_id: int, project_id: int, process_id: int, version: string)</li> <li>- get_by_process(user_id: int, project_id: int, process_id: int)</li> <li>- get_comment_by.bpmn_file(user_id: int, project_id: int, process_id: int, xml_file_link: string)</li> <li>- get_comment_by_user(user_id: int)</li> <li>- get_all_active_process_versions_in_workspace(workspace_id: int)</li> <li>- get_all_processes_versions_in_process(workspace_id: int, process_id: int, user_id: int)</li> <li>- save(xml_file_link: string, file: file, user_id: int, project_id: int, process_id: int, version: string)</li> <li>- bulk_save(user_id: int, files: file[], data: object)</li> <li>- edit_comment(user_id: int, project_id: int, process_id: int, xml_file_link: string, id: int, content: string)</li> <li>- activate_process_version(user_id, workspace_id, process_versionL:)</li> <li>- create_default(project_id: int, process_id: int)</li> <li>- create_default_values_measurements(version: string)</li> <li>- create_new_version(user_id: int, file: file, project_id: int, process_id: int)</li> <li>- create_new_version_permanently(user_id: int, xml_file_link: string, project_id: int, process_id: int)</li> <li>- comment(user_id: int, project_id: int, process_id: int, xml_file_link: string, content: string)</li> </ul>

**Hình 3.39:** Class ProcessVersionBusiness trong Business layer

Class ProcessVersionBusiness thừa kế lại những logic xử lý liên quan đến phiên bản của quy trình nghiệp vụ của đề tài trước đó, và phát triển thêm các hàm xử lý khác phục vụ cho đề tài này, chẳng hạn như kích hoạt các phiên bản và thay đổi phiên bản được kích hoạt nhằm phục vụ cho quá trình khởi tạo process portfolio cho toàn bộ quy trình nghiệp vụ có trong Workspace.

<b>StrategicImportanceBusiness</b>
<ul style="list-style-type: none"> <li>- edit_strategic_importance_of_process_version(workspace_id: int, version: string, user_id: int, total_score: float)</li> <li>- check_if_process_version_exists_in_strategic_importance(version: string)</li> <li>- add_strategic_importance_of_process_version(version: string, total_score: float)</li> <li>- update_strategic_importance_of_process_version(version: string, total_score: float)</li> <li>- get_strategic_importance_of_process_version(version: string)</li> </ul>

**Hình 3.40:** Class StrategicImportanceBusiness trong Business layer

Class StrategicImportanceBusiness quản lý logic liên quan đến độ đo độ quan trọng chiến lược của quy trình nghiệp vụ, chẳng hạn như cập nhật thay đổi của độ đo, lấy thông tin của độ đo để phục vụ

việc hiển thị process portfolio của toàn bộ Workspace.

FeasibilityBusiness
- edit_feasibility_of_process_version(version: string, total_score: float)
- check_if_process_version_exists_in_feasibility(version: string)
- update_feasibility_of_process_version(version: string, total_score: float)
- add_feasibility_of_process_version(version: string, total_score: float)
- get_feasibility_of_process_version(version: string)

Hình 3.41: Class FeasibilityBusiness trong Business layer

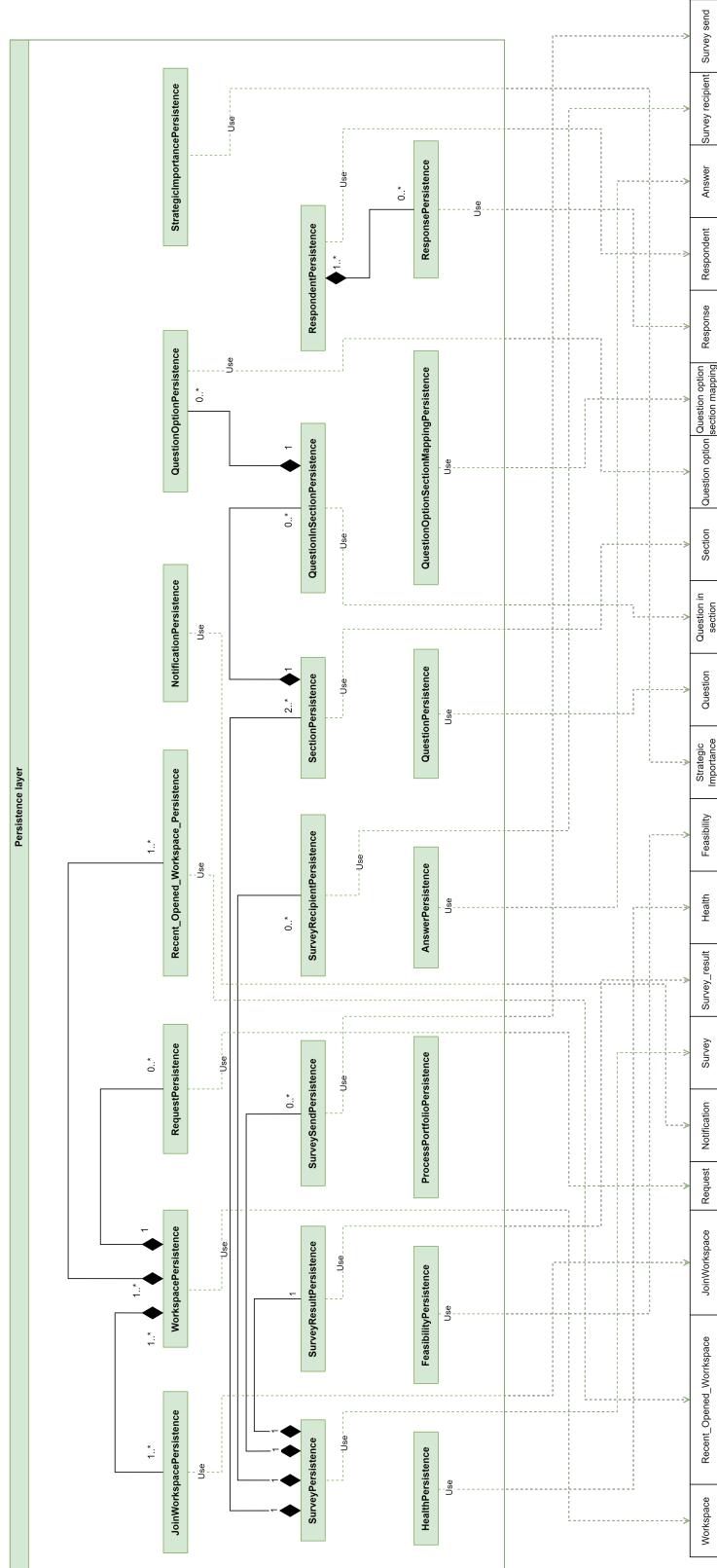
Class FeasibilityBusiness quản lý logic liên quan đến độ đo tính khả thi của quy trình nghiệp vụ, chẳng hạn như thay đổi giá trị của tính khả thi và lấy thông tin chi tiết về độ đo này.

HealthBusiness
- get_health_of_active_process_versions_in_workspace(workspace_id: int, user_id: int)
- edit_health_of_process_version(version: string, current_cycle_time: float, current_cost: float, current_quality: float, current_flexibility: float)
- check_if_process_version_exists_in_health(version: string)
- update_health_of_process_version(version: string, current_cycle_time: float, current_cost: float, current_quality: float, current_flexibility: float)
- add_health_of_process_version(version: string, current_cycle_time: float, current_cost: float, current_quality: float, current_flexibility: float)
- get_health_of_process_version(version: string)
- calculate_total_score(workspace_id: int, user_id: int, version: string)
- pl_method(measurement_values: object[], measurement_type: string)
- get_evaluation_result_of_process_version(version: string)
- get_workspace_measurements(workspace_id: int, user_id: int)
- check_values(evaluation_result: object, workspace_measurements: object, process_version_measurements: object)
- save_total_score(version: string, total_score: float)
- get_external_quality_of_process_version(version: string)
- delete_health_values(version: string)
- check_current_values(cycle_time_values: object, cost_values: object, quality_values: object, flexibility_values: object)

Hình 3.42: Class HealthBusiness trong Business layer

Class HealthBusiness bao gồm những logic xử lý về độ đo sức khỏe của quy trình nghiệp vụ, bao gồm những hàm xử lý như tính toán giá trị của độ đo này, lấy những tham số liên quan để tính toán được giá trị cuối cùng của độ đo,...

### 3.5.3 Persistence Layer

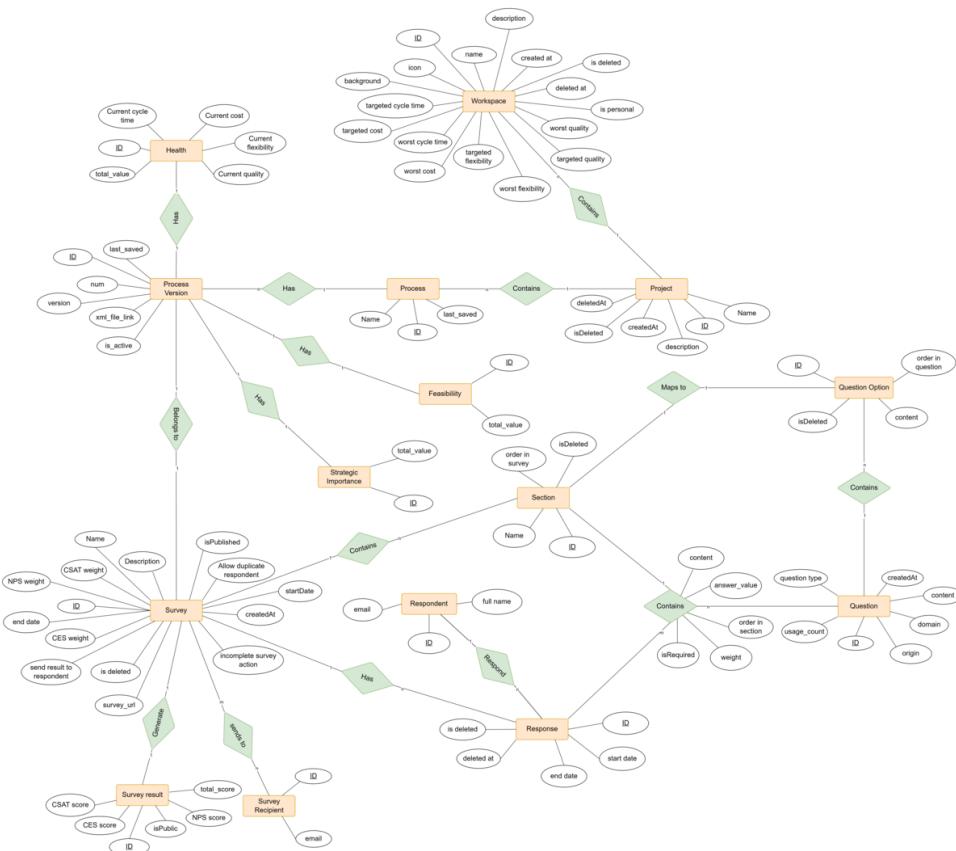


**Hình 3.43:** Tổng quan Persistence Layer

Tầng Persistence sẽ nhận dữ liệu, yêu cầu từ tầng Business để thực hiện các thao tác tương tác với cơ sở dữ liệu, sau đó trả về kết quả cho tầng Business. Các class ở tầng Business sẽ có các class ở tầng Persistence tương ứng để thực hiện xử lý dữ liệu.

## 3.6 Thiết kế cơ sở dữ liệu

### 3.6.1 Lược đồ ERD



Hình 3.44: Lược đồ ERD

Lược đồ ERD của hệ thống bao gồm các thực thể sau: Workspace, Request, User, Notification, Project, Process, Process Version, Survey, Survey result, Health, Strategic importance, Feasibility, Question, Question in survey, Question option, Section, Response, Answer, Respondent, Survey recipient với những thuộc tính như đã được mô tả ở trên. Các thực thể này có một số quan hệ như sau, và một số quan hệ cũng có những thuộc tính riêng nó:

- Một người dùng có thể tham gia nhiều workspace, một workspace có thể có nhiều người dùng tham gia.
- Một người dùng có thể tạo nhiều request, một request chỉ được tạo bởi một người dùng.
- Một workspace có thể quản lý nhiều request, một request chỉ thuộc về một workspace.

- Một workspace có thể có nhiều project, một project chỉ thuộc về một workspace.
- Một project có thể có nhiều process, một process chỉ thuộc về một project.
- Một process có thể có nhiều phiên bản, một phiên bản chỉ thuộc về một process.
- Một phiên bản process chỉ có thể có một survey.
- Một survey có thể có nhiều response, một response chỉ thuộc về một survey.
- Một survey chỉ có thể có một survey result.
- Một survey có thể có nhiều question, một question chỉ thuộc về một survey.
- Một question có thể có nhiều question option, một question option chỉ thuộc về một question.

### 3.6.2 Ánh xạ ERD và mô tả chi tiết thực thể

#### 3.6.2.1 User: Người dùng của hệ thống

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của người dùng. Khoá chính.
email	varchar	Email của người dùng
password	varchar	Mật khẩu của người dùng
phoneNumber	varchar	Số điện thoại của người dùng
name	text	tên người dùng
avatar	text	Đường dẫn tới ảnh đại diện của người dùng
verified	boolean	True nếu tài khoản đã xác thực, False nếu tài khoản chưa xác thực

Bảng 3.18: Thực thể User

### 3.6.2.2 Project: Dự án trong workspace

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của project. Khoá chính.
projectName	text	Tên của project
description	text	Mô tả của project
isDelete	boolean	Đánh dấu project này đã bị xoá hay chưa
createdAt	timestamp	Thời gian project được tạo
ownerId	integer	ID của người sở hữu project. Khoá ngoại tham chiếu tới trường Id trong bảng User
workspaceId	integer	ID của workspace nơi mà project được tạo. Khoá ngoại tham chiếu tới trường Id trong bảng Workspace
deletedAt	timestamp	Thời gian project được xoá khỏi Workspace
isWorkspaceDeleted	boolean	Đánh dấu workspace đã bị xoá hay chưa

Bảng 3.19: Thực thể Project

### 3.6.2.3 Join Workspace: Quan hệ tham gia giữa người dùng và workspace

Thuộc tính	Kiểu	Mô tả
<u>memberId</u>	integer	ID của member tham gia Workspace. Khoá ngoại tham chiếu tới trường Id trong User.
<u>workspaceId</u>	integer	ID của Workspace. Khoá ngoại tham chiếu tới trường Id trong Workspace.
joinedAt	timestamp	Thời gian người dùng tham gia Workspace
isDeleted	boolean	Đánh dấu người dùng đã rời khỏi Workspace hay chưa
permission	text	Quyền hạn của người dùng khi tham gia Workspace
leftAt	timestamp	Thời gian người dùng rời khỏi Workspace
isWorkspaceDeleted	boolean	Đánh dấu workspace đã bị xoá hay chưa

Bảng 3.20: Thực thể Join Workspace

### 3.6.2.4 Workspace: Không gian làm việc trong hệ thống

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của workspace. Khoá chính.
name	text	Tên của workspace
description	text	Mô tả của workspace
isDeleted	boolean	Đánh dấu workspace đã bị xoá hay chưa
createdAt	timestamp	Thời gian workspace được tạo
ownerId	integer	ID của người sở hữu workspace. Khoá ngoại tham chiếu tới trường Id trong bảng User
background	text	Đường link dẫn tới background của workspace
icon	text	Đường link dẫn tới icon của workspace
deletedAt	timestamp	Thời gian Workspace được xoá khỏi hệ thống
isPersonal	boolean	True nếu đây là workspace cá nhân. False nếu là workspace khác
targeted_cost	double precision	giá trị mục tiêu của độ đo chi phí
worst_cost	double precision	giá trị tệ nhất của độ đo chi phí
targeted_cycle_time	double precision	giá trị mục tiêu của độ đo thời gian chu kỳ
worst_cycle_time	double precision	giá trị tệ nhất của độ đo thời gian chu kỳ
targeted_quality	double precision	giá trị mục tiêu của độ đo chất lượng
worst_quality	double precision	giá trị tệ nhất của độ đo chất lượng
targeted_flexibility	double precision	giá trị mục tiêu của độ đo sự linh hoạt
worst_flexibility	double precision	giá trị tệ nhất của độ đo sự linh hoạt

Bảng 3.21: Thực thể Workspace



### 3.6.2.5 Recent Opened Workspace: Quan hệ chỉnh sửa giữa người dùng và workspace

Thuộc tính	Kiểu	Mô tả
<u>userId</u>	integer	ID của member tham gia Workspace. Khoá ngoại tham chiếu tới trường Id trong User.
<u>workspaceId</u>	integer	ID của Workspace. Khoá ngoại tham chiếu tới trường Id trong Workspace.
openedAt	timestamp	Thời gian người dùng mở Workspace lần cuối.
isHided	boolean	Đánh dấu người dùng đã ẩn Workspace khỏi danh sách của mình hay chưa
isPinned	boolean	Đánh dấu người dùng đã pin Workspace vào danh sách ưu tiên hay chưa
isWorkspaceDeleted	boolean	Đánh dấu workspace đã bị xoá hay chưa

Bảng 3.22: Thực thể Recent Opened Workspace

### 3.6.2.6 Notification: Các thông báo của người dùng

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của thông báo. Khoá chính
notificationType	varchar	Kiểu thông báo
createdAt	timestamp	Thời gian người dùng nhận được thông báo.
status	text	Trạng thái của thông báo, đã được chấp thuận hay đồng ý hoặc từ chối lời mời.
isDeleted	boolean	Đánh dấu thông báo bị xoá hay chưa
isStarred	boolean	Đánh dấu thông báo đã được đánh dấu ưu tiên hay chưa
workspaceId	integer	ID của workspace gửi thông báo đó. Khoá ngoại tham chiếu tới trường Id trong Workspace
isRead	boolean	Đánh dấu thông báo đã được đọc hay chưa
permission	text	permission của người nhận.
deletedAt	timestamp	Thời gian xoá request.

Bảng 3.23: Thực thể Notification



### 3.6.2.7 Request: Các yêu cầu trong hệ thống

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của request. Khoá chính
type	varchar	Kiểu request
createdAt	timestamp	Thời gian người dùng tạo request.
status	text	Trạng thái của request, đang được xử lý hay đã được chấp thuận, từ chối.
isDeleted	boolean	Đánh dấu request bị xoá hay chưa
isWorkspaceDeleted	boolean	Đánh dấu workspace đã bị xoá hay chưa
workspaceId	integer	ID của workspace chứa request đó. Khoá ngoại tham chiếu tới trường Id trong Workspace
senderId	integer	ID của người gửi request. Khoá ngoại tham chiếu tới trường Id trong User
recipientId	integer	ID của người nhận request đã xử lý. Khoá ngoại tham chiếu tới trường Id trong User
handlerId	integer	ID của người xử lý request. Khoá ngoại tham chiếu tới trường Id trong User
frPermission	text	permission hiện tại của người gửi.
toPermission	text	permission mong muốn của người gửi.
rcpPermission	text	permission của người sẽ nhận được lời mời tham gia workspace.
deletedAt	timestamp	Thời gian xoá request.

Bảng 3.24: Thực thể Request

### 3.6.2.8 Feasibility: Mức độ khả thi của quy trình

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của mức độ khả thi. Khoá chính
process_version_version	integer	phiên bản của process chứa mức độ khả thi. Khoá ngoại tham chiếu tới trường version trong Process version
total_value	double precision	Giá trị mức độ khả thi

Bảng 3.25: Thực thể Feasibility



### 3.6.2.9 Strategic Importance: Mức độ quan trọng chiến lược của quy trình

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của mức độ quan trọng chiến lược. Khoá chính
process_version_version	integer	phiên bản của process chứa mức độ quan trọng chiến lược. Khoá ngoại tham chiếu tới trường version trong Process version
total_value	double precision	Giá trị mức độ quan trọng chiến lược

Bảng 3.26: Thực thể Strategic Importance

### 3.6.2.10 Process version: Phiên bản của quy trình

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của phiên bản quy trình. Khoá chính
version	integer	Số phiên bản của quy trình
xml_file_link	varchar	Đường dẫn link tới file BPMN.
isDeleted	boolean	Đánh dấu phiên bản quy trình bị xoá hay chưa
process_id	integer	ID của quy trình. Khoá ngoại tham chiếu tới trường Id trong Process
project_id	integer	ID của project. Khoá ngoại tham chiếu tới trường Id trong Project
deleted_at	timestamp	Thời gian xoá phiên bản quy trình.
last_saved	timestamp	Thời gian lưu phiên bản quy trình cuối cùng.
is_active	boolean	Đánh dấu phiên bản quy trình đang hoạt động hay không.

Bảng 3.27: Thực thể Process Version

### 3.6.2.11 Health: Sức khoẻ của quy trình

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của sức khoẻ. Khoá chính
process_version_version	integer	phiên bản của process chứa sức khoẻ. Khoá ngoại tham chiếu tới trường version trong Process version
current_cycle_time	double precision	Giá trị hiện tại của độ đo thời gian
current_cost	double precision	Giá trị hiện tại của độ đo chi phí
current_quality	double precision	Giá trị hiện tại của độ đo chất lượng
current_flexibility	double precision	Giá trị hiện tại của độ đo sự linh hoạt

Bảng 3.28: Thực thể Health

### 3.6.2.12 Survey result: Kết quả khảo sát

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của kết quả khảo sát. Khoá chính
survey_id	integer	ID của bài khảo sát. Khoá ngoại tham chiếu tới trường ID trong Survey
ces_score	double precision	Giá trị độ đo CES
nps_score	double precision	Giá trị độ đo NPS
csat_score	double precision	Giá trị độ đo CSAT
total_score	double precision	Giá trị tổng của bài khảo sát
visibility	boolean	Đánh dấu kết quả survey có thể xem bởi những người dùng khác ngoài project owner không

Bảng 3.29: Thực thể Survey Result

### 3.6.2.13 Survey: Bài khảo sát

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của bài khảo sát. Khoá chính
name	text	Tên của bài khảo sát
description	text	Mô tả của bài khảo sát
is_deleted	boolean	Đánh dấu bài khảo sát bị xoá hay chưa
created_at	timestamp	Thời gian bài khảo sát được tạo
start_date	timestamp	Thời gian bài khảo sát được publish
process_version_version	varchar	phiên bản của process chứa bài khảo sát. Khoá ngoại tham chiếu tới trường version trong Process version
deleted_at	timestamp	Thời gian bài khảo sát được xoá khỏi hệ thống
is_published	boolean	Đánh dấu bài khảo sát đã được publish hay chưa
end_date	timestamp	Thời gian kết thúc bài khảo sát
allow_duplicate_respondent	boolean	Cho phép người dùng trả lời bài khảo sát nhiều lần hay không
send_result_to_respondent	boolean	Gửi kết quả khảo sát cho người trả lời hay không
survey_url	varchar	Đường dẫn tới bài khảo sát
ces_weight	double precision	Trọng số của độ đo CES
nps_weight	double precision	Trọng số của độ đo NPS
csat_weight	double precision	Trọng số của độ đo CSAT
domain	varchar	Lĩnh vực của bài khảo sát
last_saved	timestamp	Thời gian lưu bài khảo sát cuối cùng
incomplete_response_action	varchar	Hành động đối với những bài khảo sát đang làm dở của người dùng

Bảng 3.30: Thực thể Survey



### 3.6.2.14 Response: Phản hồi của người dùng

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của phản hồi. Khoá chính
survey_id	integer	ID của bài khảo sát. Khoá ngoại tham chiếu tới trường ID trong Survey
respondent_id	integer	ID của người trả lời. Khoá ngoại tham chiếu tới trường ID trong Respondent
start_date	timestamp	Thời điểm người dùng thực hiện bài làm khảo sát
end_date	timestamp	Thời điểm người dùng submit bài làm khảo sát
is_deleted	boolean	Đánh dấu phản hồi bị xoá hay chưa

Bảng 3.31: Thực thể Response

### 3.6.2.15 Respondent: Người trả lời khảo sát

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của người trả lời. Khoá chính
email	varchar	Email của người trả lời
full_name	text	Tên của người trả lời

Bảng 3.32: Thực thể Respondent

### 3.6.2.16 Answer: Câu trả lời của người dùng trong khảo sát

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của câu trả lời. Khoá chính
response_id	integer	ID của phản hồi. Khoá ngoại tham chiếu tới trường ID trong Response
question_id	integer	ID của câu hỏi. Khoá ngoại tham chiếu tới trường ID trong Question
value	varchar	Câu trả lời của người dùng

Bảng 3.33: Thực thể Answer



### 3.6.2.17 Question in section: Câu hỏi trong một phần của khảo sát

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của câu hỏi. Khoá chính
content	varchar	Nội dung của câu hỏi
section_id	integer	ID của phần. Khoá ngoại tham chiếu tới trường ID trong Section
question_id	integer	ID của câu hỏi. Khoá ngoại tham chiếu tới trường ID trong Question
order_in_section	integer	Thứ tự của câu hỏi trong phần
weight	double precision	Trọng số của câu hỏi
is_required	boolean	Đánh dấu câu hỏi có bắt buộc hay không
question_type	varchar	Loại câu hỏi

Bảng 3.34: Thực thể Question in section

### 3.6.2.18 Section in survey: Phần của khảo sát

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của phần. Khoá chính
name	varchar	Tên của phần
survey_id	integer	ID của bài khảo sát. Khoá ngoại tham chiếu tới trường ID trong Survey
order_in_survey	integer	Thứ tự của phần trong bài khảo sát
is_deleted	boolean	Đánh dấu phần bị xoá hay chưa

Bảng 3.35: Thực thể Section in survey

### 3.6.2.19 Question option: Câu trả lời của câu hỏi trong khảo sát

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của câu trả lời. Khoá chính
question_id	integer	ID của câu hỏi. Khoá ngoại tham chiếu tới trường ID trong Question
value	varchar	Nội dung của câu trả lời
is_deleted	boolean	Đánh dấu câu trả lời bị xoá hay không
question_in_section_id	integer	ID của câu hỏi trong phần. Khoá ngoại tham chiếu tới trường ID trong Question in section
order	integer	Thứ tự của câu trả lời trong câu hỏi

Bảng 3.36: Thực thể Question option

### 3.6.2.20 Question option Section mapping: Liên kết giữa câu hỏi và phần của khảo sát với câu trả lời của câu hỏi đó

Thuộc tính	Kiểu	Mô tả
<u>section_id</u>	integer	ID của phần trong khảo sát. Khoá ngoại tham chiếu tới trường ID trong Section
<u>question_option_id</u>	integer	ID của câu trả lời. Khoá ngoại tham chiếu tới trường ID trong Question option
is_deleted	boolean	Đánh dấu liên kết bị xoá hay không

Bảng 3.37: Thực thể Question option Section mapping

### 3.6.2.21 Survey recipient: Người nhận khảo sát

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của người nhận khảo sát. Khoá chính
email	varchar	Email của người nhận

Bảng 3.38: Thực thể Survey Recipient



### 3.6.2.22 Question: Câu hỏi trong hệ thống

Thuộc tính	Kiểu	Mô tả
<u>ID</u>	integer	ID của câu hỏi. Khoá chính
content	varchar	Nội dung của câu hỏi
is_deleted	boolean	Đánh dấu câu hỏi bị xoá hay không
question_type	varchar	Loại câu hỏi
created_at	timestamp	Thời gian câu hỏi được tạo
origin	varchar	Nguồn của câu hỏi
domain	varchar	Lĩnh vực của câu hỏi
contributor_id	integer	ID của người đóng góp câu hỏi. Khoá ngoại tham chiếu tới trường ID trong User
usage_count	integer	Số lần câu hỏi được sử dụng

Bảng 3.39: Thực thể Question

### 3.6.2.23 Send survey: Gửi khảo sát cho người nhận khảo sát

Thuộc tính	Kiểu	Mô tả
<u>survey_id</u>	integer	ID của bài khảo sát. Khoá ngoại tham chiếu tới trường ID trong Survey
<u>recipient_id</u>	integer	ID của người nhận khảo sát. Khoá ngoại tham chiếu tới trường ID trong Survey Recipient

Bảng 3.40: Thực thể Send Survey

# Chương 4

## Hiện thực hệ thống

### 4.1 Công nghệ sử dụng

#### 4.1.1 ReactJS

React là một thư viện Javascript mã nguồn mở, được phát triển và hỗ trợ bởi Meta (trước đây là Facebook), ra mắt vào năm 2013 với mục đích xây dựng giao diện người dùng. React được sử dụng rộng rãi để xây dựng các trang web Single Page Application (SPA) và các ứng dụng trên nền tảng di động. React tổ chức các thành phần của trang web thành các component, cho phép chúng ta nhóm các logic xử lý của các phần nhỏ của giao diện lại với nhau, từ đó có thể tái sử dụng nhiều lần và mở rộng thêm chức năng cho mỗi thành phần đó. Mỗi component có thể tự quản lý trạng thái (state) của riêng mình, cũng như nhận vào các thuộc tính (props) được truyền từ component cha. React tối ưu việc render trang web bằng cách chỉ cho phép render khi phát hiện có sự thay đổi trong state và props, từ đó giảm thiểu số lần phải re - render. React còn nổi tiếng với việc sử dụng Virtual DOM để cải thiện hiệu năng, khi so sánh DOM gốc của trang web và Virtual DOM để tìm ra những sự khác nhau, và chỉ render lại những điểm khác nhau đó, thay vì re - render cả cây DOM, điều mà sẽ làm hiệu năng của trang web giảm đi đáng kể.

Một số ưu điểm của React có thể kể đến như sau:

- Cập nhật hiệu quả. Các thuật toán của ReactJS đảm bảo rằng các cập nhật, thay đổi UI được thực thi hiệu quả. Chỉ những components bị ảnh hưởng bởi một thay đổi nào đó mới được re - render, giúp tối ưu hóa tổng quan hiệu năng của ứng dụng.
- Luồng dữ liệu một chiều. ReactJS chỉ cung cấp luồng dữ liệu đi theo một chiều, giúp cho lập trình viên dễ dàng hiểu cách dữ liệu hoạt động, thay đổi như thế nào trong cây phân cấp component, cũng như dự đoán trước trạng thái của component.
- Cộng đồng lớn. ReactJS là thư viện front-end Javascript nổi tiếng, do đó sự hỗ trợ từ cộng đồng là rất tốt khi ta có vấn đề trong việc lập trình. Ngoài ra, các thư viện, components của bên thứ ba, các hooks được phát triển và duy trì bởi hàng triệu lập trình viên trên toàn thế giới, giúp ta có thể tìm thấy thư viện mong muốn, phục vụ cho các chức năng khác nhau của hệ thống. [5]

#### 4.1.2 Python

Kế thừa từ đề tài trước, ở đề tài này, nhóm tiếp tục sử dụng Python làm ngôn ngữ lập trình chính cho phần server của hệ thống và áp dụng framework Flask.

Ở Flask có một số ưu điểm nổi bật để chúng tôi quyết định lựa chọn nó [20]:

- Flask là một micro web framework được viết bằng Python, không yêu cầu tool hay thư viện cụ thể nào. “Micro” không có nghĩa là thiếu chức năng mà “micro” theo triết lý thiết kế là cung cấp một lõi chức năng “súc tích” nhất cho ứng dụng web nhưng người dùng có thể mở rộng bất cứ lúc nào. Flask luôn hỗ trợ các thành phần tiện ích mở rộng cho ứng dụng như tích hợp cơ sở dữ liệu, xác thực biểu mẫu, xử lý upload, các công nghệ xác thực, template, email, RESTful... Người dùng có thể tập trung xây dựng web application ngay từ đầu trong một khoảng thời gian rất ngắn và có thể mở rộng quy mô của ứng dụng tùy theo yêu cầu.
- Chính nguyên lý và áp dụng microframework đã giúp cho Flask Python có thể dễ dàng mở rộng nếu có nhu cầu từ ứng dụng web. Do phần core chạy độc lập và ít phụ thuộc, nên cho dù mở rộng ở mức quy mô thì ứng dụng web sử dụng Flask Python vẫn đáp ứng được.
- Sự linh hoạt là tính năng cốt lõi và cũng là ưu điểm của Flask Python. Chính vì giữ cho core và các thành phần khác đơn giản nên ít bị phụ thuộc vào nhau, sự đơn giản này giúp ta có thể chuyển hướng ứng dụng web theo business owner dễ dàng hơn. Ngoài ra do ít bị phụ thuộc lẫn nhau nên một thành phần nào đó bị sập cũng khó mà kéo theo cả hệ thống bị sập. [20]

#### 4.1.3 Websocket và Socket.IO

Websocket là một công nghệ theo thời gian thực, cung cấp một giao tiếp hai chiều, song công toàn phần giữa máy khách và máy chủ trên một kết nối đơn socket, bền vững. Trong chế độ truyền song công toàn phần, việc giao tiếp giữa bên gửi và bên nhận có thể diễn ra đồng thời, bên gửi và bên nhận có thể truyền và nhận tín hiệu cùng một lúc. Chế độ truyền song công toàn phần giống như con đường hai chiều, trong đó các phương tiện có thể lưu chuyển theo cả hai hướng cùng một lúc. Ví dụ, trong một cuộc trò chuyện qua điện thoại, hai người giao tiếp và cả hai có thể tự do nói và nghe cùng một lúc. Một kết nối Websocket sẽ bắt đầu với một HTTP request/response handshake. Nếu handshake khởi tạo thành công, máy chủ và máy khách đồng ý sử dụng kết nối TCP có sẵn đã được thiết lập cho một kết nối Websocket. Kết nối này sẽ tồn tại lâu nhất có thể (theo lý thuyết có thể tồn tại mãi mãi), cho phép máy chủ và máy khách có thể độc lập gửi dữ liệu cho nhau.

Một số ưu điểm của Websocket:

- Trước khi có Websocket, các kỹ thuật HTTP như AJAX long polling và Comet là những tiêu chuẩn để xây dựng các ứng dụng thời gian thực. Tuy nhiên, ở Websocket đã lược bỏ đi việc thiết lập kết nối mỗi khi có request, và giảm kích thước của mỗi message vì không còn HTTP headers. Điều này giúp tiết kiệm băng thông, cải thiện độ trễ.



- Độ linh hoạt đã ăn sâu vào thiết kế của công nghệ Websocket, cho phép việc hiện thực các giao thức ở tầng ứng dụng và các mở rộng cho những chức năng khác (ví dụ như pub/sub messaging).
- Là một công nghệ hướng sự kiện (event - driven), Websocket cho phép dữ liệu được truyền đi mà không cần máy khách phải gửi request. Cơ chế này sẽ cực kỳ hữu ích trong ngữ cảnh máy khách cần phản ứng nhanh chóng với một sự kiện nào đó (đặc biệt là những sự kiện không thể dự đoán trước).

Dẫu vậy, Websocket vẫn có một số những nhược điểm như sau:

- Không như HTTP, Websocket là có trạng thái (stateful). Trong một số trường hợp sẽ khá khó khăn khi phải xử lý điều này, đặc biệt là lúc mở rộng hệ thống, bởi vì nó yêu cầu phía server phải theo dõi mỗi kết nối Websocket đơn lẻ và duy trì các thông tin trạng thái.
- Websocket không tự động khôi phục khi các kết nối bị huỷ - và để tự động khôi phục ta cần phải tự hiện thực tính năng này. Đây cũng là lý do vì sao có nhiều thư viện hỗ trợ Websocket phía máy khách.
- Một số môi trường (ví dụ như môi trường trong doanh nghiệp với các proxy servers) sẽ chặn các kết nối Websocket.

Chính vì vậy, chúng tôi sử dụng thêm một thư viện để hỗ trợ trong việc trao đổi dữ liệu qua giao thức Websocket. Socket.IO là một thư viện theo thời gian thực, giúp giảm thiểu độ trễ và giao tiếp hai chiều giữa máy khách và máy chủ. Socket.IO được xây dựng trên nền tảng giao thức Websocket, và cung cấp thêm một số tính năng khác như tự động kết nối lại, hỗ trợ broadcast,...

Một số ưu điểm của Socket.IO:

- Socket.IO hỗ trợ multiplexing thông qua namespaces. Việc tận dụng namespaces giúp ta có thể tối giản hóa số lượng kết nối TCP đã sử dụng, và lưu các cổng socket trên máy chủ.
- Socket.IO cho phép phía máy chủ có thể linh hoạt broadcast các sự kiện đến tất cả máy khách đã kết nối. Ta cũng có thể broadcast các sự kiện tới một tập nhỏ các máy khách thông qua tính năng room.
- Socket.IO hỗ trợ HTTP long polling như một phương án dự phòng, sẽ hữu ích trong các môi trường không hỗ trợ Websocket.
- Socket.IO cung cấp cơ chế Ping/Pong, cho phép phát hiện một kết nối có đang hoạt động hay không. Ngoài ra, nếu một máy khách bị ngắt kết nối, nó sẽ tự động kết nối lại.

Tuy nhiên, thư viện này cũng bộc lộ một số nhược điểm như sau:

- Socket.IO có giới hạn trong các tính năng bảo mật. Chẳng hạn như, nó không cung cấp mã hoá đầu cuối, hay cơ chế nào để tạo ra và làm mới các tokens phục vụ cho việc xác thực người dùng.



- Không tương thích với những hiện thực Websocket khác. Ta không thể sử dụng một máy khách Websocket cơ bản với một máy chủ Socket.IO.
- Socket.IO được thiết kế để làm việc trong một vùng đơn, hơn là kiểu kiến trúc đa vùng. Điều này có thể làm gia tăng độ trễ nếu các người dùng ở các vùng khác nhau.

Mặc dù Socket.IO có một số nhược điểm như trên, chúng tôi nhận thấy rằng những nhược điểm đó không ảnh hưởng nhiều đến hệ thống, và những ưu điểm của Socket.IO là đủ để chúng tôi quyết định sử dụng nó. [1]

#### 4.1.4 PostgreSQL

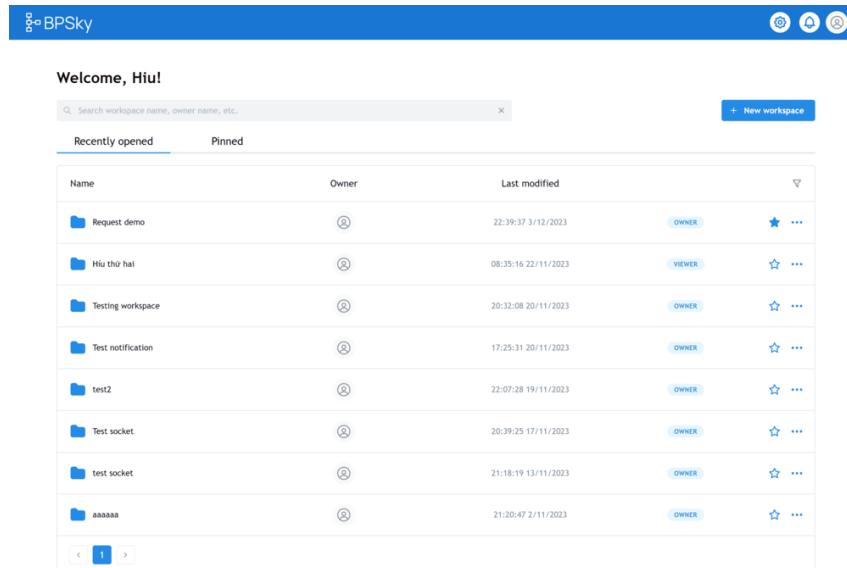
Để thiết kế database cho mục đích lưu trữ dữ liệu, nhóm quyết định chọn cơ sở dữ liệu quan hệ (relational database) để hiện thực, và hệ quản trị cơ sở dữ liệu được chọn là PostgreSQL. Những lợi ích của việc sử dụng hệ cơ sở dữ liệu có quan hệ như sau:

- Bảo đảm tính toàn vẹn dữ liệu. Cơ sở dữ liệu quan hệ bắt buộc cần phải bảo đảm tính toàn vẹn dữ liệu, bằng việc sử dụng các ràng buộc như khoá chính, khoá ngoại, các ràng buộc duy nhất. Những ràng buộc này giúp cho dữ liệu được chính xác, nhất quán và tuân theo các quy định đã được định nghĩa từ trước.
- Nhờ việc sử dụng kỹ thuật chuẩn hoá dữ liệu, mà cơ sở dữ liệu quan hệ có thể tái tổ chức lại dữ liệu, lược bỏ những dư thừa hay các vấn đề liên quan đến nhất quán dữ liệu khi tiến hành chỉnh sửa, thêm, xoá dữ liệu.
- Hỗ trợ các công cụ cũng như tính năng nâng cao giúp người dùng dễ dàng xử lý trên dữ liệu một cách hiệu quả như indexing, transaction, cơ chế query optimization.
- Hệ quản trị cơ sở dữ liệu quan hệ cho phép lưu trữ và phục hồi nhờ việc cung cấp các cơ chế có sẵn cho việc phục hồi. Điều này đảm bảo dữ liệu có thể được lưu trữ một cách an toàn, giải quyết các vấn đề liên quan tới mất mát dữ liệu.

So sánh với các loại cơ sở dữ liệu không quan hệ phổ biến hiện nay, ưu điểm lớn nhất của việc sử dụng cơ sở dữ liệu quan hệ so với cơ sở dữ liệu không quan hệ trong phạm vi đề tài đó là việc sử dụng cấu trúc có quan hệ giúp quản lý dữ liệu tốt hơn, tính nhất quán dữ liệu được đảm bảo hơn. Ngoài ra, do đề tài này được phát triển dựa trên đề tài gốc vốn sử dụng cơ sở dữ liệu quan hệ, để đảm bảo tính thống nhất về mặt chức năng của hệ thống, nhóm nhận thấy tiếp tục loại hình cơ sở dữ liệu này là một điều đúng đắn. [19]

## 4.2 Hiện thực giao diện người dùng

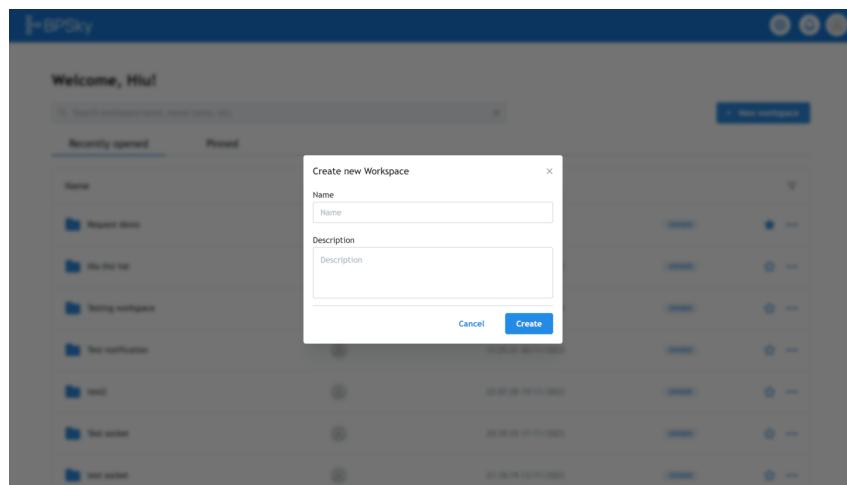
### 4.2.1 Giao diện Trang chủ BPSky



Hình 4.1: Giao diện trang default homepage

Người dùng sau khi đăng nhập hệ thống thành công sẽ được chuyển hướng tới giao diện Default homepage. Tại giao diện mặc định này, hệ thống sẽ hiển thị hai danh sách "Recently opened workspace" - hiển thị danh sách những workspace người dùng tham gia/sở hữu theo thứ tự thời gian mở và "Pinned workspace" - hiển thị danh sách những workspace người dùng pin (đánh dấu). Người dùng có thể chọn icon "ba chấm" ở bên phải cùng của mỗi item để mở menu thao tác với mỗi item workspace.

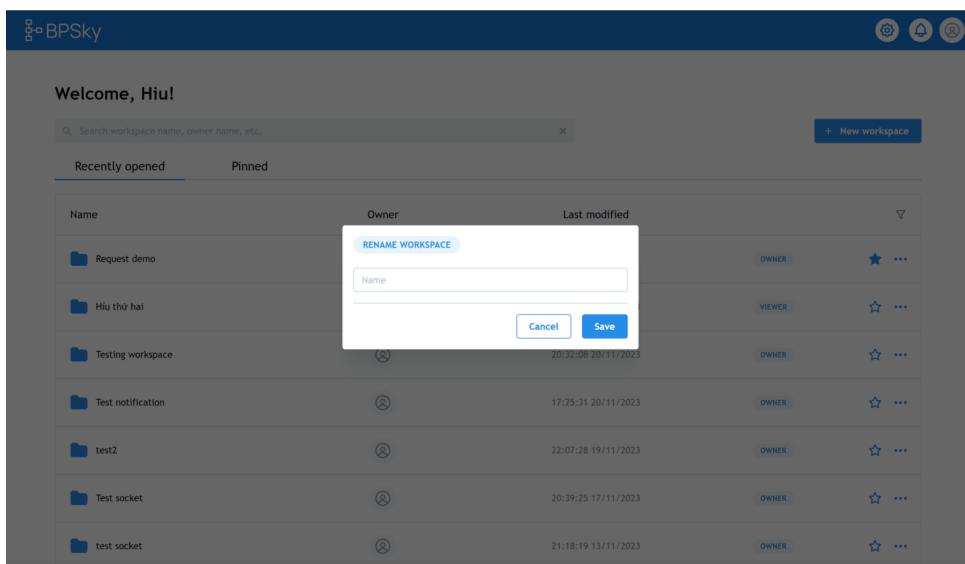
### 4.2.2 Giao diện tạo mới workspace



Hình 4.2: Giao diện create modal khi người dùng tạo mới workspace

Người dùng có thể tạo mới workspace bằng cách chọn "Create new workspace" từ dropdown menu. Sau khi chọn, hệ thống sẽ hiển thị giao diện create modal như hình ???. Tại đây, người dùng có thể nhập tên và mô tả cho workspace. Sau khi hoàn tất, người dùng có thể chọn "Create" để tạo mới workspace hoặc "Cancel" để hủy thao tác.

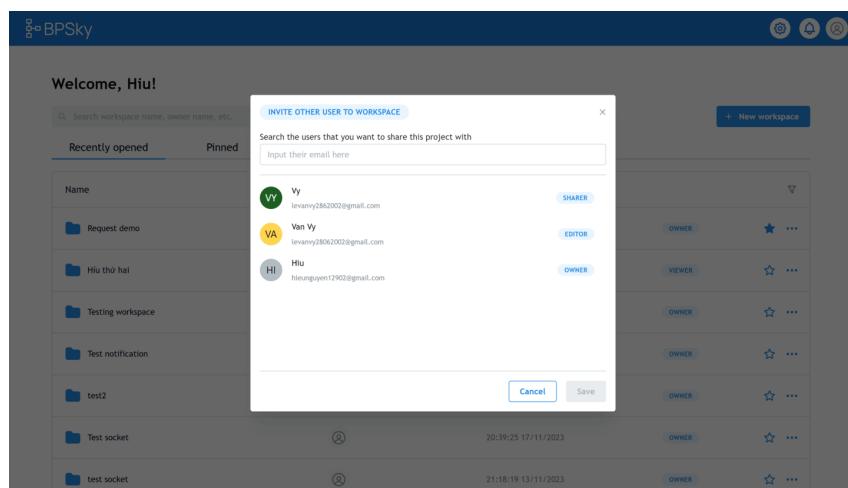
#### 4.2.3 Giao diện đổi tên workspace



Hình 4.3: Giao diện rename modal khi người dùng đổi tên workspace

Người dùng có thể đổi tên workspace bằng cách chọn "Rename" từ dropdown menu. Sau khi chọn, hệ thống sẽ hiển thị giao diện rename modal như hình ???. Tại đây, người dùng có thể nhập tên mới cho workspace. Sau khi hoàn tất, người dùng chọn "Save" để đổi tên workspace.

#### 4.2.4 Giao diện chia sẻ workspace



Hình 4.4: Giao diện share modal khi người dùng muốn mời/gửi yêu cầu mời người dùng khác vào workspace

Người dùng có thể mời người dùng khác vào workspace bằng cách chọn "Share" từ dropdown menu. Sau khi chọn, hệ thống sẽ hiển thị giao diện share modal như hình ???. Tại đây, người dùng có thể nhập tên người dùng mà mình muốn mời vào workspace, sau đó chọn quyền hạn của người được mời vào - quyền hạn của người này không được phép cao hơn người gửi lời mời (owner > editor > sharer > viewer). Sau khi hoàn tất, người dùng có thể chọn "Share" để mời người dùng vào workspace hoặc "Cancel" để hủy thao tác.

#### 4.2.5 Giao diện thông tin chi tiết workspace

The screenshot shows the 'test socket' workspace detail page. At the top, there is a search bar with placeholder text 'Search project name, etc.' and three icons: a person icon, a gear icon, and a refresh icon. To the right of the search bar are buttons for 'Open Editor' and '+ Create New Project'. Below the header is a table listing ten projects:

Name	Owner	Created at	Actions
Create 'Adjust permission' requests	⋮	14:54:12 12/11/2023	OWNER ⋮
test create new_renamed	⋮	12:29:43 9/11/2023	OWNER ⋮
téc nab	⋮	22:39:59 29/10/2023	OWNER ⋮
In o long	⋮	22:31:42 29/10/2023	OWNER ⋮
112	⋮	22:31:30 29/10/2023	OWNER ⋮
14	⋮	22:31:02 29/10/2023	OWNER ⋮
13	⋮	22:30:54 29/10/2023	OWNER ⋮
12	⋮	22:30:40 29/10/2023	OWNER ⋮
11	⋮	22:29:01 29/10/2023	OWNER ⋮

At the bottom of the table, there are navigation buttons for 'Previous', 'Next', and a page number '2'.

Hình 4.5: Giao diện trang Workspace detail

Người dùng có thể truy cập vào trang Workspace detail bằng cách chọn vào tên của workspace từ danh sách các workspace mà người dùng đang tham gia/sở hữu. Tại đây, người dùng có thể xem danh sách những project có trong workspace. Tương tự với danh sách workspace, mỗi item project cũng sẽ có icon "ba chấm" ở phải cùng, người dùng có thể mở dropdown menu để thao tác với project.

#### 4.2.6 Giao diện thông báo của người dùng

Content	Created at	Status	Pin
User Van Vy has invited you to join workspace Request demo with permission sharer	23/11/2023, 14:20:51	pending	
Van Vy has invited you to join Request demo with permission viewer	23/11/2023, 14:13:25	accepted	
Van Vy has invited you to join Request demo with permission viewer	23/11/2023, 14:12:14	declined	
Van Vy has invited you to join Request demo with permission viewer	23/11/2023, 12:33:57	accepted	
User Hiu has invited you to join workspace Testing workspace with permission editor	20/11/2023, 20:05:51	pending	
User Hiu has invited you to join workspace Testing workspace with permission editor	20/11/2023, 20:03:09	declined	
User Hiu has invited you to join workspace Testing workspace with permission viewer	20/11/2023, 18:42:46	accepted	
User Hiu has invited you to join workspace Test notification with permission viewer	20/11/2023, 18:09:59	declined	
User Hiu has invited you to join workspace Test notification with permission viewer	20/11/2023, 18:08:26	declined	
User Hiu has invited you to join workspace Test notification with permission viewer	20/11/2023, 18:08:07	declined	

Hình 4.6: Giao diện trang thông báo của người dùng

Người dùng có thể truy cập vào trang thông báo bằng cách chọn icon hình chuông ở phía trên thanh điều hướng. Tại đây, người dùng có thể xem danh sách những thông báo mà hệ thống gửi tới người dùng. Mỗi item thông báo sẽ tùy thuộc vào loại thông báo tương ứng mà hiển thị trạng thái khác nhau (bao gồm "Approved" - chấp nhận, "Declined" - từ chối, "Pending" - đang chờ phản hồi). Người dùng có thể đánh dấu thông báo quan trọng bằng cách nhấn vào icon ngôi sao ở mỗi item thông báo, thông báo quan trọng có thể được lọc thông qua công cụ filter.

#### 4.2.7 Giao diện chi tiết thông báo của người dùng

Content	Created at	Status	Pin
User Van Vy has invited you to join workspace Request demo with permission sharer	23/11/2023, 14:20:51	pending	
Van Vy has invited you to join Request demo with permission viewer	23/11/2023, 14:13:25	accepted	
Van Vy has invited you to join Request demo with permission viewer	23/11/2023, 14:12:14	declined	
Van Vy has invited you to join Request demo with permission viewer	23/11/2023, 12:33:57	accepted	
User Hiu has invited you to join workspace Testing workspace with permission editor	20/11/2023, 20:05:51	pending	
User Hiu has invited you to join workspace Testing workspace with permission editor	20/11/2023, 20:03:09	declined	
User Hiu has invited you to join workspace Testing workspace with permission viewer	20/11/2023, 18:42:46	accepted	
User Hiu has invited you to join workspace Test notification with permission viewer	20/11/2023, 18:09:59	declined	
User Hiu has invited you to join workspace Test notification with permission viewer	20/11/2023, 18:08:26	declined	
User Hiu has invited you to join workspace Test notification with permission viewer	20/11/2023, 18:08:07	declined	

Hình 4.7: Giao diện modal hiển thị thông tin chi tiết của thông báo

Người dùng có thể xem thông tin chi tiết của mỗi thông báo bằng cách nhấp vào item thông báo tương ứng. Hệ thống sẽ hiển thị giao diện modal như hình ???. Tại đây, người dùng có thể xem thông tin chi tiết của thông báo, tùy thuộc vào loại thông báo mà người dùng sẽ có thể tương tác "Approve" - chấp nhận hoặc "Decline" từ chối lời mời được đính kèm trong thông báo. Nếu người dùng muốn đóng modal, người dùng có thể nhấp vào nút "Cancel".

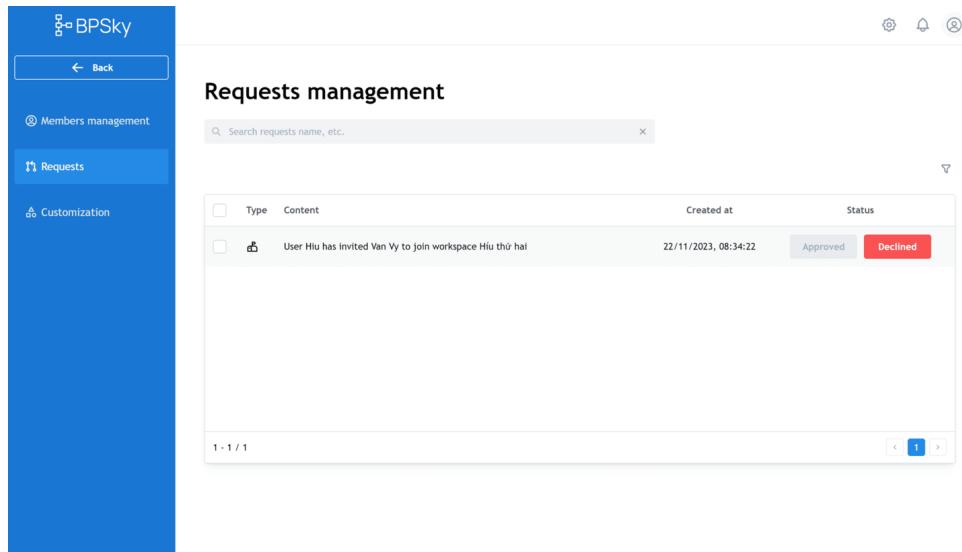
#### 4.2.8 Giao diện quản lý thành viên trong workspace

Name	Email	Permission	Joined at
Vy	levanvy2862002@gmail.com	Can view	17/11/2023, 11:27:23
Vy	vy.leforwork@hcmut.edu.vn	Can view	09/11/2023, 14:17:53
Hlu	hieu.nguyen12902@gmail.com	Owner	02/11/2023, 21:59:53

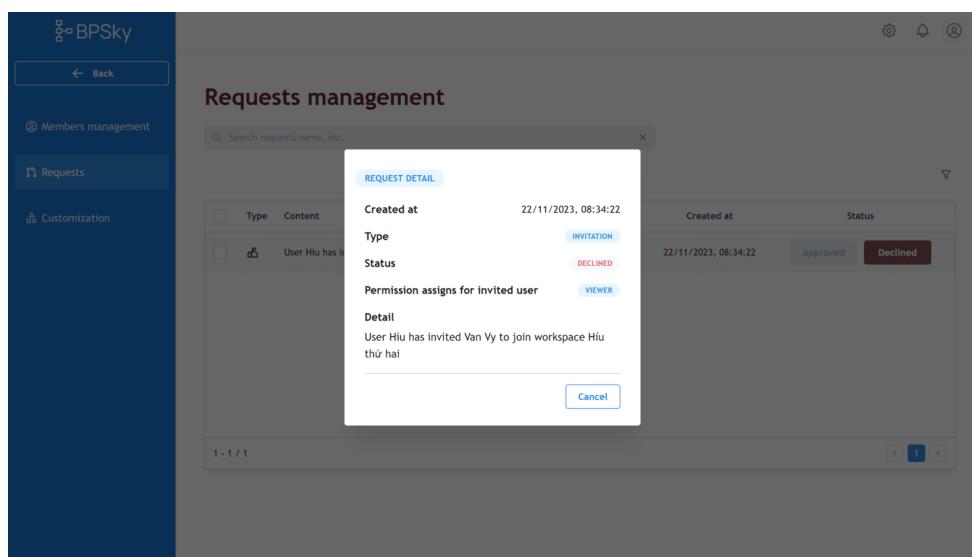
Hình 4.8: Giao diện trang quản lý thành viên trong Workspace

Nếu người dùng là người sở hữu workspace (workspace owner) thì có thể chọn icon bánh răng bên cạnh tên của workspace tại trang Workspace Detail để điều hướng tới giao diện quản lý workspace. Người dùng sẽ được điều hướng mặc định tới trang quản lý thành viên trong workspace. Hệ thống sẽ hiển thị danh sách những thành viên có trong workspace cùng với quyền hạn và thời gian tham gia workspace của họ, chúng ta có thể lọc người dùng theo vai trò trong hệ thống. Nếu người sở hữu workspace muốn mời thêm thành viên trong hệ thống vào workspace thì có thể chọn nút "Invite", hệ thống sẽ mở Share modal - Share modal này sẽ gửi lời mời trực tiếp đến người dùng thông qua hộp thông báo.

#### 4.2.9 Giao diện quản lý yêu cầu từ thành viên trong workspace



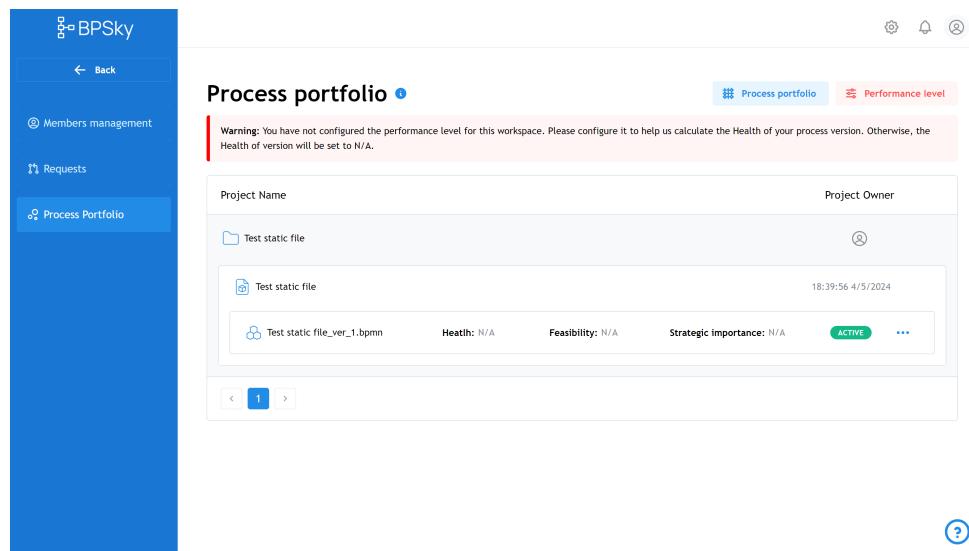
Hình 4.9: Giao diện trang quản lý yêu cầu từ thành viên trong workspace



Hình 4.10: Giao diện modal hiển thị thông tin chi tiết của yêu cầu

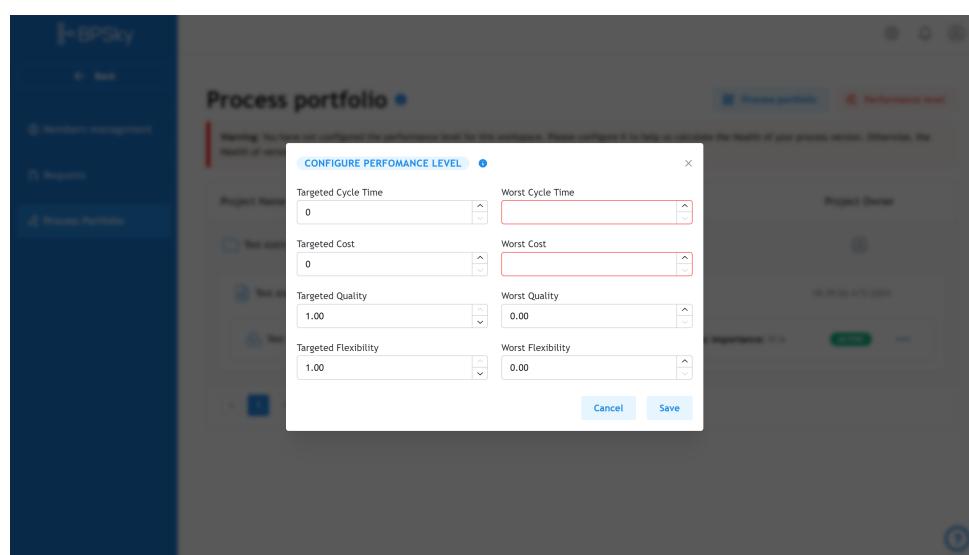
Khi người dùng chọn "Request management" ở thanh sidebar thì hệ thống sẽ chuyển hướng người dùng tới trang quản lý yêu cầu của workspace. Tại đây, hệ thống sẽ hiển thị danh sách những yêu cầu được gửi từ phía thành viên bên trong workspace. Người dùng có thể chọn "Approve" hoặc "Decline" để phản hồi yêu cầu tương ứng. Nếu người dùng muốn xem thông tin chi tiết của yêu cầu, người dùng có thể nhấp vào item yêu cầu tương ứng, hệ thống sẽ hiển thị modal như hình ??.

#### 4.2.10 Giao diện quản lý process portfolio của workspace



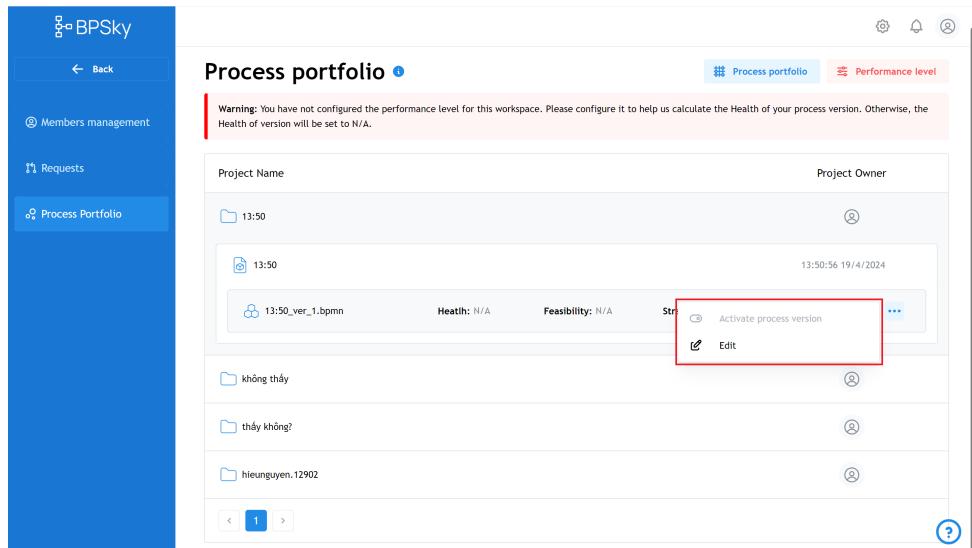
Hình 4.11: Giao diện quản lý process portfolio của workspace

Sau khi người dùng chọn Process portfolio trên thanh điều hướng, hệ thống sẽ chuyển hướng người dùng tới trang quản lý/khởi tạo process portfolio cho workspace. Người dùng có thể thấy được danh sách những process version/project có trong workspace. Người dùng có thể chọn nút Process portfolio để xem biểu đồ process portfolio của workspace hoặc nút Performance level để chỉnh sửa các thông tin liên quan tới thang đo được sử dụng chung cho toàn bộ workspace.



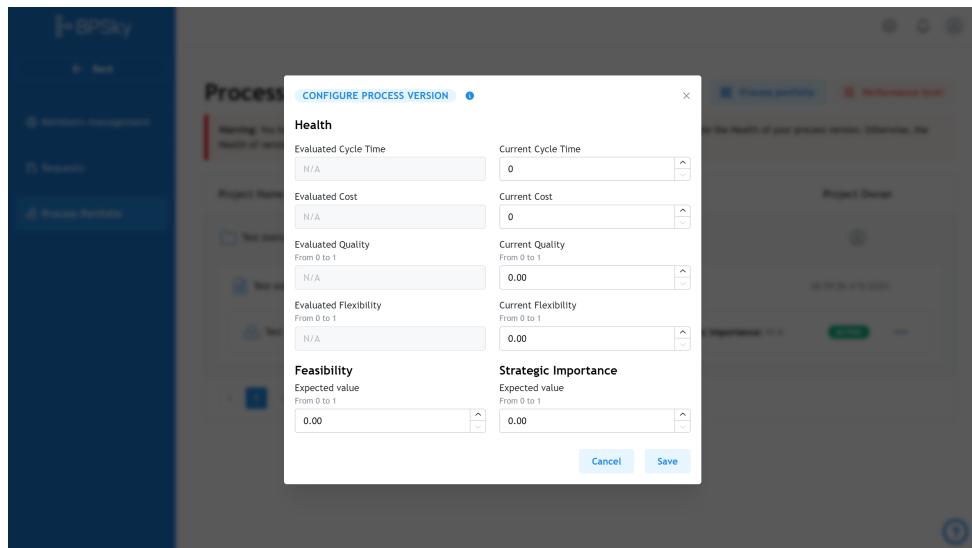
Hình 4.12: Giao diện chỉnh sửa giá trị thang đo cho process portfolio của workspace

Người dùng sẽ được yêu cầu nhập thông tin liên quan tới thang đo cho process portfolio của workspace. Sau khi nhập thông tin, người dùng có thể chọn nút Save để lưu thông tin hoặc nút Cancel để hủy bỏ thao tác. Thông tin này sẽ được sử dụng để xác định thang đo (giá trị lớn nhất, giá trị nhỏ nhất) khi tính toán những thông số liên quan tới hiệu suất của process version (Health).



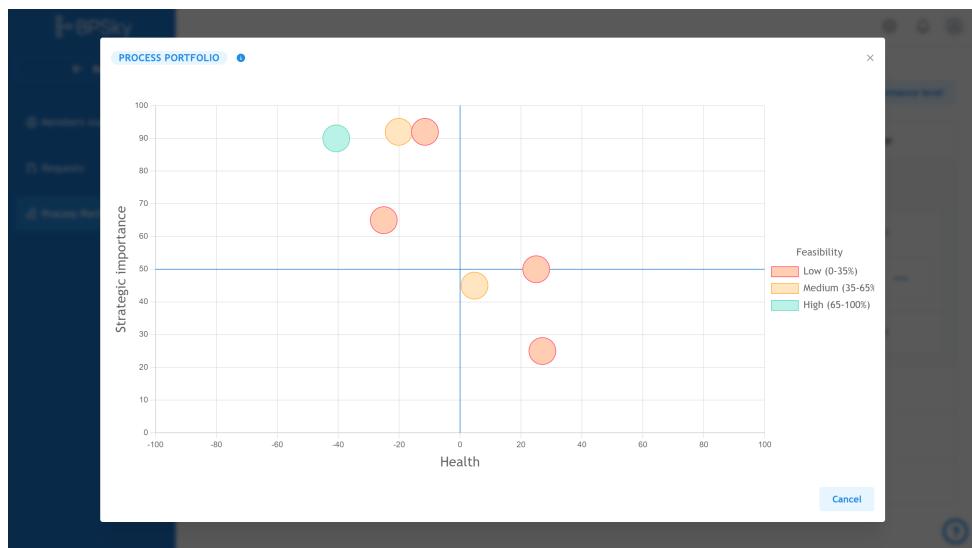
Hình 4.13: Giao diện menu của process version

Người dùng có thể chọn nút Edit trong menu trên item tương ứng để chỉnh sửa thông tin liên quan tới thang đo của process version. Sau khi chỉnh sửa xong, người dùng có thể chọn nút Save để lưu thông tin hoặc nút Cancel để hủy bỏ thao tác. Hoặc chọn Activate để kích hoạt process version tương ứng, mỗi một process chỉ có một version được đặt trạng thái active. Version này sẽ được sử dụng để khởi tạo biểu đồ process portfolio.



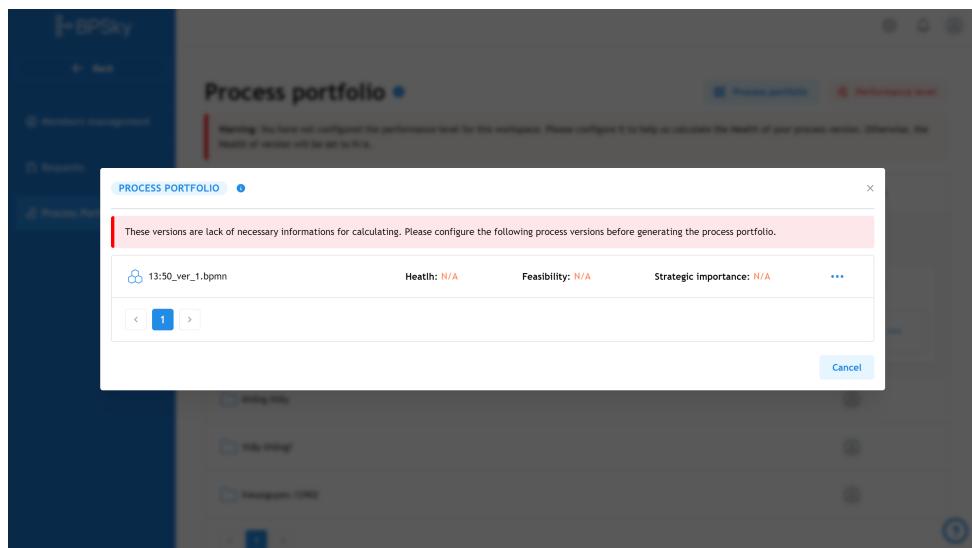
Hình 4.14: Giao diện chỉnh sửa giá trị của process version trong workspace

Nếu người dùng chọn Edit trong menu thì người dùng sẽ được yêu cầu nhập thông tin liên quan tới process version tương ứng. Hệ thống sẽ mở modal bao gồm thông tin về các giá trị Strategic importance (Mức độ quan trọng chiến lược), Feasibility (Tính khả thi) và Health (Hiệu suất) của version đó. Sau khi nhập thông tin, người dùng có thể chọn nút Save để lưu thông tin hoặc nút Cancel để hủy bỏ thao tác.



Hình 4.15: Giao diện biểu đồ process portfolio của workspace

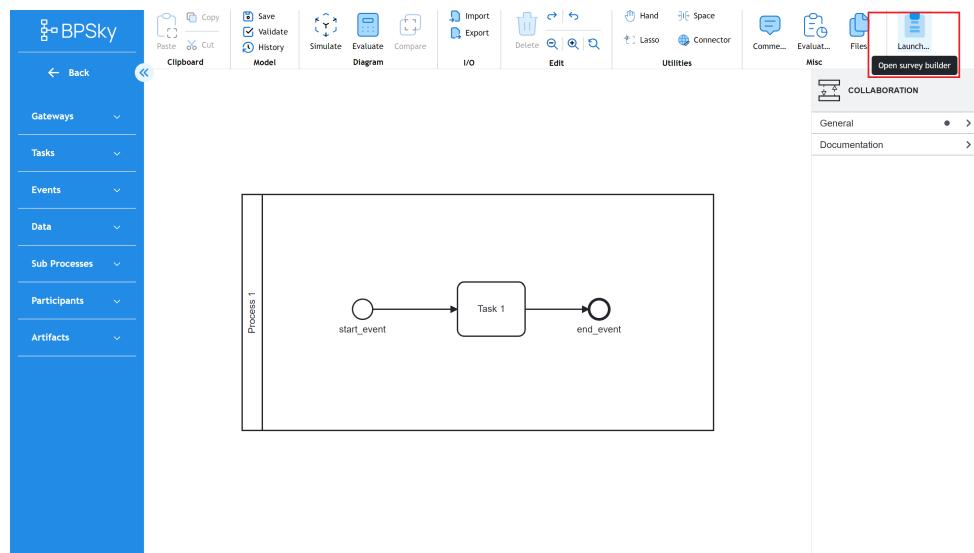
Người dùng sau khi chọn nút Process portfolio thì hệ thống sẽ mở biểu đồ process portfolio của workspace. Biểu đồ này sẽ hiển thị các process version/project có trong workspace. Mỗi node sẽ hiển thị thông tin liên quan tới process version tương ứng. Màu sắc của node sẽ thể hiện tính khả thi (Feasibility) của process version. Trục tung và trục hoành sẽ lần lượt là giá trị của hiệu suất (Health) và mức độ quan trọng chiến lược (Strategic importance). Người dùng có thể di chuyển chuột qua node để xem thông tin chi tiết của process version tương ứng.



Hình 4.16: Giao diện hiển thị danh sách process version không có/thiếu dữ liệu

Nếu có tồn tại process version nào trong workspace bị thiếu dữ liệu liên quan tới hiệu suất, tính khả thi hoặc độ quan trọng chiến lược thì hệ thống sẽ hiển thị danh sách những version đó. Người dùng cần bổ sung đầy đủ thông tin thì process portfolio mới được khởi tạo thành công.

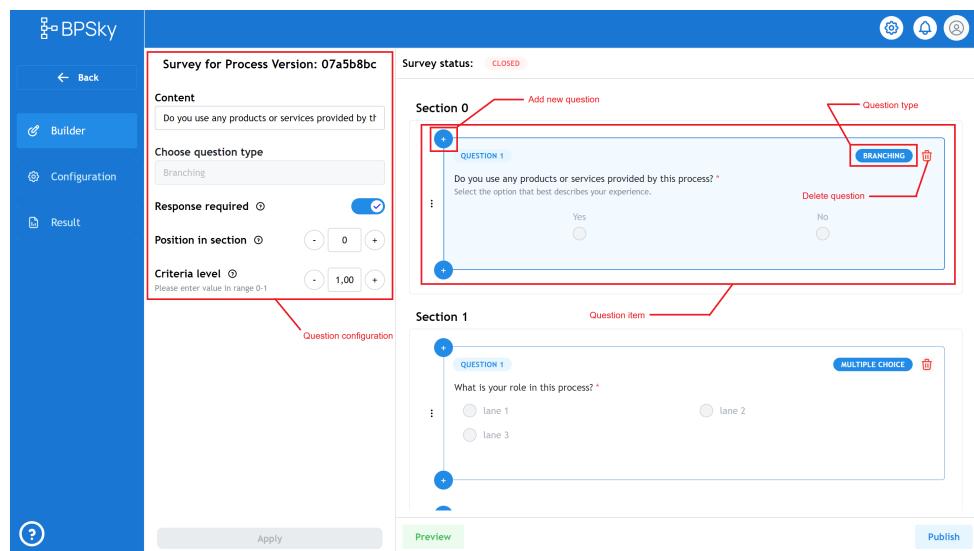
#### 4.2.11 Giao diện khởi tạo khảo sát cho quy trình nghiệp vụ



Hình 4.17: Giao diện chỉnh sửa mô hình quy trình nghiệp vụ

Khi người dùng muốn khởi tạo khảo sát cho quy trình đang làm việc hiện tại, người dùng cần mở quy trình tương ứng trong giao diện Editor, trên thanh công cụ (toolbar) người dùng chọn biểu tượng khảo sát (Launch Survey) để mở giao diện khởi tạo khảo sát.

#### 4.2.12 Giao diện chỉnh sửa nội dung khảo sát



Hình 4.18: Giao diện chỉnh sửa nội dung khảo sát

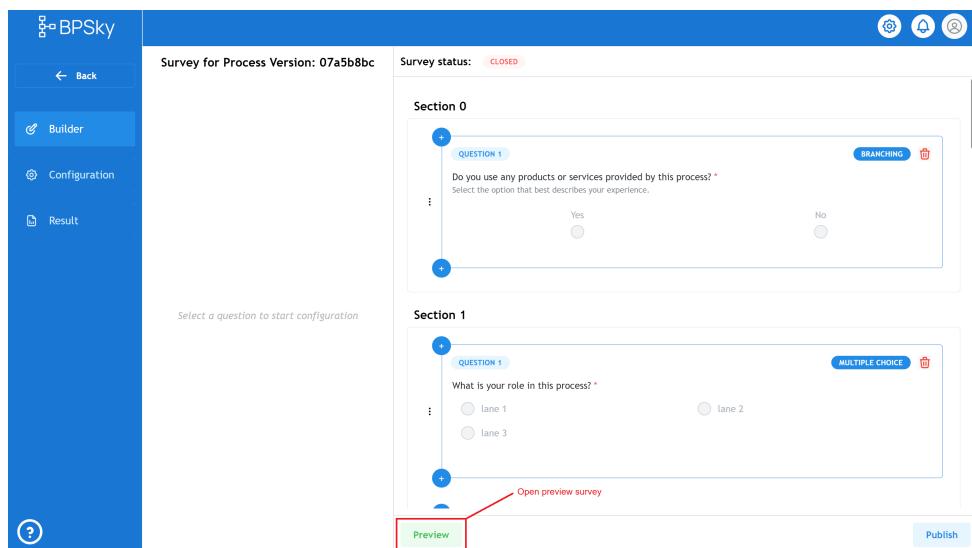
Sau khi chọn biểu tượng khảo sát trên thanh công cụ thì người dùng sẽ được chuyển hướng tới trang giao diện chỉnh sửa nội dung bảng khảo sát. Giao diện chỉnh sửa nội dung khảo sát bao gồm các thành phần sau:

- **Question configuration:** Người dùng có thể chỉnh sửa các thông tin của câu hỏi như tên câu hỏi,

loại câu hỏi, câu trả lời (đối với loại câu hỏi multiple question), ...

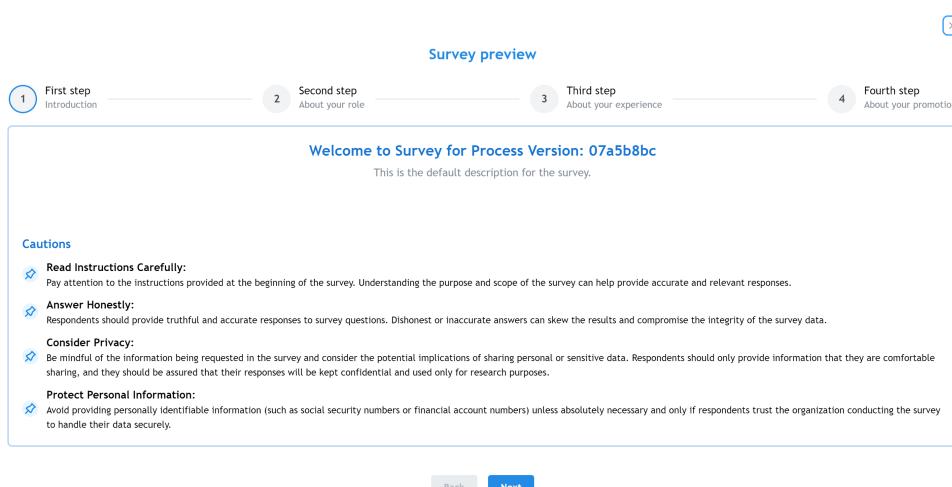
- **Question item:** Hiển thị nội dung của câu hỏi và thứ tự của câu hỏi trong bảng khảo sát.
- **Question type:** Hiển thị phân loại của câu hỏi, trong phạm vi hệ thống có các loại câu hỏi như NPS, CES, CSAT, CES-IN và CES-OUT (dùng thu thập ý kiến của người dùng), Multiple choice và Branching.
- **Delete question:** Dùng để xóa câu hỏi khỏi bảng khảo sát.

#### 4.2.13 Giao diện xem trước bảng khảo sát



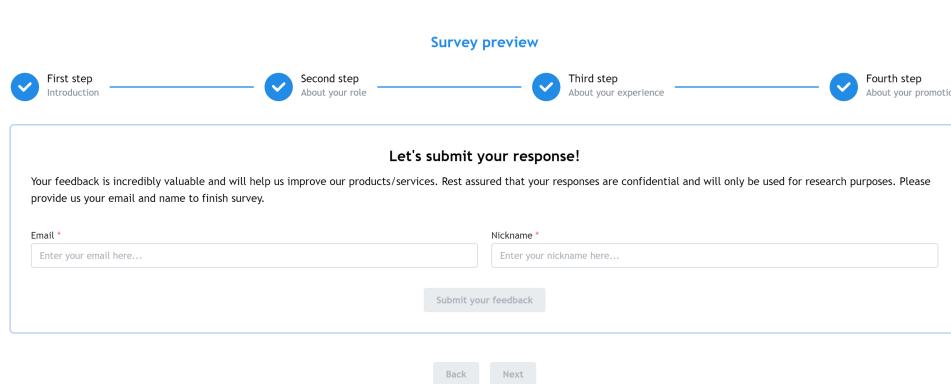
Hình 4.19: Giao diện xem trước bảng khảo sát

Người dùng có thể chọn nút Preview để mở phiên bản preview của bảng khảo sát.



Hình 4.20: Giao diện xem trước bảng khảo sát - trang mở đầu

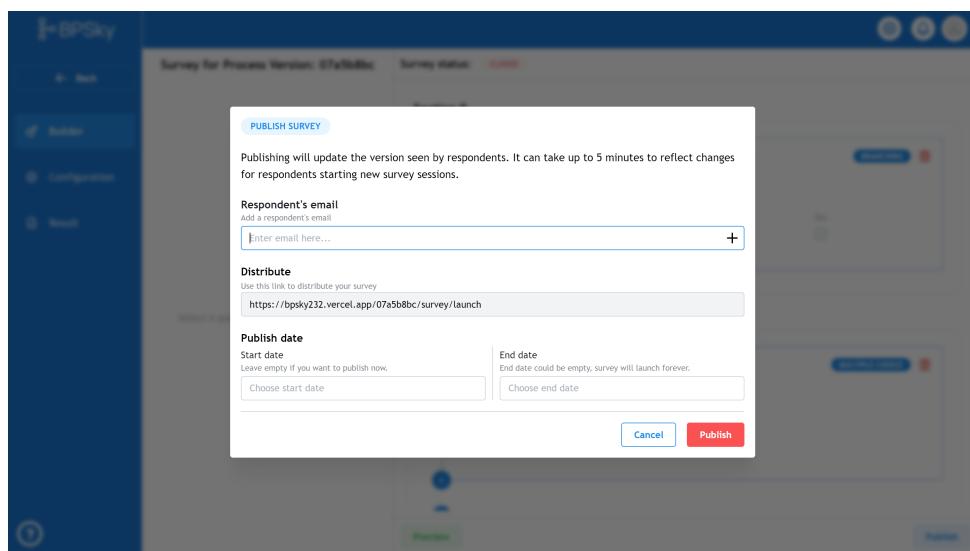
Trang mở đầu của bảng khảo sát sẽ hiển thị thông tin chung của bảng khảo sát như tên, mô tả, những cảnh báo đối với người làm khảo sát. Để chuyển đến phần tiếp theo người dùng chọn nút Next, tương tự nếu muốn quay trở lại phần trước đó thì có thể chọn Back.



**Hình 4.21:** Giao diện xem trước bảng khảo sát - trang kết thúc

Trang kết thúc của bảng khảo sát sẽ hiển thị lời cảm ơn và yêu cầu người dùng nhập thông tin như tên và email để nhận được kết quả của bảng khảo sát sau khi khảo sát đóng. Sau khi nhập đầy đủ thông tin thì người dùng chọn nút Submit để gửi đi phản hồi bảng khảo sát.

#### 4.2.14 Giao diện công bố khảo sát



**Hình 4.22:** Giao diện công bố khảo sát

Sau khi người dùng hoàn thành việc cấu hình bảng khảo sát, người dùng có thể chọn mục Publish để công bố bảng khảo sát. Tại đây, người dùng có thể cấu hình các thông tin liên quan tới việc công bố bảng khảo sát như thời gian mở khảo sát, thời gian đóng khảo sát và email của người dùng được gửi

thông báo khi khảo sát được mở. Ngoài ra người dùng có thể sử dụng url được tạo sẵn để mời người dùng khác tham gia khảo sát.

#### 4.2.15 Giao diện cấu hình khảo sát

General  
Title, description

Survey response  
Survey expiration, incomplete responses

Display name  
Enter a survey name to show in search results, social media posts, and on browser tabs.  
Survey for Process Version: 07a5b8bc

Description  
Enter a survey description to show in search results and on social media posts.  
This is the default description for the survey.

Criteria level for each measurement metric  
Select criteria level for each measurement in this survey. Depending on the aim, properties of survey, the measurements' level may vary. Range value: 0 - 1.

CSAT | 1.00 | CES | 1.00 | NPS | 1.00 |

Save changes

Hình 4.23: Giao diện cấu hình thông tin chung của bảng khảo sát

Người dùng sau khi chọn Configuration trên thanh điều hướng thì hệ thống sẽ chuyển hướng người dùng tới giao diện cấu hình thông tin chung của bảng khảo sát. Tại đây, người dùng có thể thay đổi các cấu hình liên quan tới bảng khảo sát như tên, mô tả và các trọng số của các loại câu hỏi trong bảng khảo sát. Các cấu hình này sẽ ảnh hưởng đến việc hiển thị và tính toán kết quả của bảng khảo sát.

General  
Title, description

Survey response  
Survey expiration, incomplete responses

Allow respondents to send another response  
Let respondents send another response when they have finished the current one.  
 No

Send survey results to respondents  
Send a summary of survey results to respondents after the survey closes.  
 No

Survey availability

Start date  
Choose start date

End date  
Choose end date

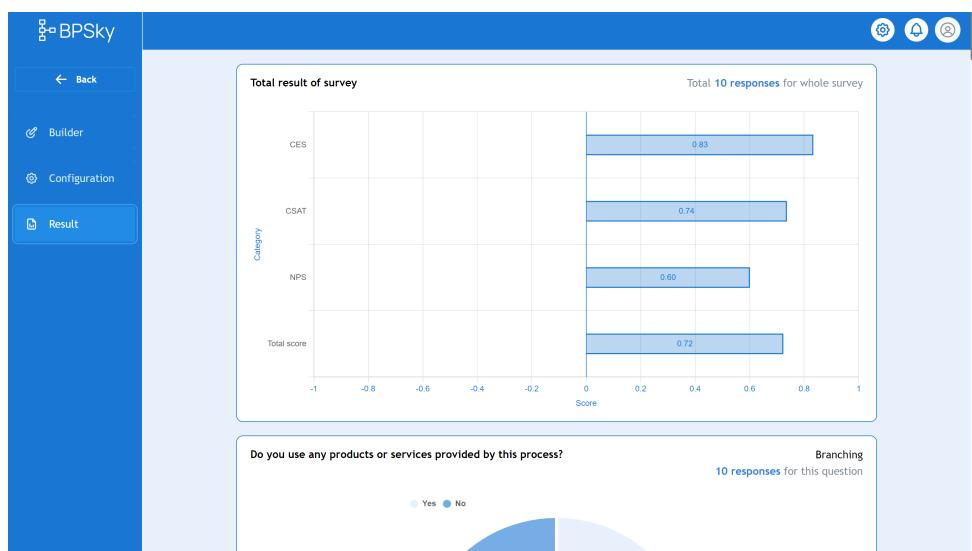
Delete survey Close survey Save changes

Hình 4.24: Giao diện cấu hình phản hồi của bảng khảo sát

Ngoài ra, người dùng có thể chọn mục Survey response để chuyển hướng tới giao diện cấu hình phản hồi của bảng khảo sát. Tại đây, người dùng có thể cấu hình các thông tin liên quan tới phản hồi của bảng

khảo sát như thông kết quả sau khi khảo sát hoàn thành qua hòm thư, cho phép người dùng thực hiện một/nhiều lần khảo sát và thời gian đóng/mở khảo sát.

#### 4.2.16 Giao diện kết quả bảng khảo sát



Hình 4.25: Giao diện kết quả bảng khảo sát

Sau khi người dùng chọn biểu tượng kết quả trên thanh công cụ thì hệ thống sẽ chuyển hướng người dùng tới trang giao diện kết quả bảng khảo sát. Giao diện này bao gồm kết quả thống kê chi tiết lựa chọn của từng câu hỏi trong bảng khảo sát và kết quả tổng hợp của bảng khảo sát. Tùy thuộc vào loại câu hỏi mà sẽ được thể hiện thông qua biểu đồ khác nhau:

- Loại câu hỏi rẽ nhánh - Branching: Sử dụng biểu đồ tròn để thể hiện phần trăm số lượng câu trả lời cho từng lựa chọn Có/Không.
- Loại câu hỏi Multiple choice/CES/CSAT/NPS: Sử dụng biểu đồ cột để thể hiện phần trăm số lượng câu trả lời cho từng lựa chọn. Bao gồm có 7 lựa chọn (tương ứng từ 1 đến 7) đối với loại câu hỏi CES và CSAT, và 10 lựa chọn (tương ứng từ 0 đến 10) đối với loại câu hỏi NPS.
- Loại câu hỏi CES-IN/CSAT-IN: Sử dụng dạng danh sách câu trả lời ngắn để liệt kê những ý kiến của người làm khảo sát.

### 4.3 Triển khai hệ thống

Hiện nay có rất nhiều cách để triển khai của hệ thống phần mềm cho người dùng, sau đây là một số cách triển khai:

1. Tự triển khai server: đây là cách thủ công nhất để triển khai ứng dụng web, lập trình viên sẽ cài đặt các công cụ như web server (Apache, Nginx, IIS,...), database server (MySQL, PostgreSQL,...) và triển khai ứng dụng lên server của mình. Phương pháp này đòi hỏi kiến thức về quản trị hệ thống

và bảo mật, tuy nhiên nó cung cấp cho lập trình viên sự linh hoạt cao để tuỳ chỉnh và điều khiển ứng dụng của mình.

2. Platform-as-a-service (PaaS): Đây là phương pháp triển khai ứng dụng web trên một nền tảng được quản lý bởi một nhà cung cấp dịch vụ. Lập trình viên chỉ cần đưa mã nguồn của ứng dụng lên nền tảng PaaS và các công cụ và hạ tầng sẽ được quản lý bởi nhà cung cấp dịch vụ. Các ví dụ về PaaS và Heroku, Google App Engine, Microsoft Azure, Amazon Web Services.
3. Infrastructure-as-a-service (IaaS): Đây là phương pháp triển khai ứng dụng web trên một hạ tầng đám mây được cung cấp bởi một nhà cung cấp dịch vụ. Lập trình viên có thể tự cấu hình môi trường của mình trên hạ tầng đám mây, bao gồm cả máy chủ ảo, lưu trữ, mạng và các công cụ khác. Các ví dụ về IaaS là Amazon Web Services, Google Cloud Platform và Microsoft Azure.
4. Docker containers: Đây là phương pháp triển khai ứng dụng web trên các container Docker. Docker cho phép lập trình viên đóng gói ứng dụng và các phụ thuộc của nó trong một container, và triển khai các container này trên bất kỳ máy chủ nào. Phương pháp này đảm bảo tính di động và tương thích với mọi môi trường.

Sau khi cân nhắc thì chúng tôi đã chọn cách tiếp cận là sử dụng PaaS để triển khai ứng dụng. Đây là một số ưu điểm của cách tiếp cận này:

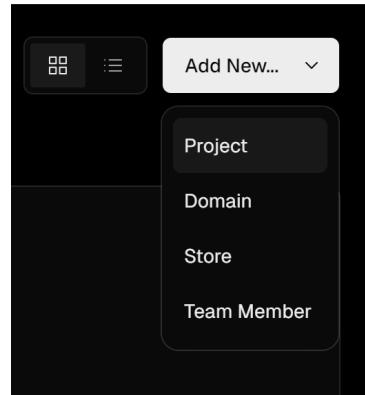
1. Tiết kiệm thời gian và chi phí: Không cần phải chi một khoản tiền lớn cho việc xây dựng, cài đặt phần cứng hay chi phí phát sinh trong khi hệ thống tạm ngừng hoạt động, và không phải tốn thời gian để thiết lập cũng như bảo trì phần lõi.
2. Tăng tốc độ phát triển: Thiết lập và triển khai đơn giản, nhanh chóng đưa ứng dụng lên giai đoạn production.
3. Mở rộng dễ dàng: Có thể mở rộng thêm về khả năng lưu trữ cũng như khả năng xử lý khi cần.
4. Tính linh hoạt cao: Cho phép các nhà phát triển có thể tham gia và làm việc trên các ứng dụng ở bất cứ nơi đâu.

Tuy nhiên, PaaS cũng có một số hạn chế, như:

1. Phụ thuộc vào nhà cung cấp dịch vụ: Việc triển khai hệ thống trên các nền tảng này sẽ phụ thuộc khá nhiều vào khả năng của nhà cung cấp dịch vụ, chẳng hạn như băng thông, dung lượng,... Chỉ có thể chọn lựa những giải pháp có sẵn mà khó lòng tự điều chỉnh theo ý muốn và nhu cầu của nhà phát triển.
2. Nguy cơ về bảo mật: Mặc dù các nhà cung cấp dịch vụ luôn cố gắng nâng cao và phát triển việc bảo mật nền tảng và cơ sở hạ tầng, vấn đề bảo mật của ứng dụng lại phụ thuộc khá nhiều vào mức độ bảo mật của nền tảng đó.

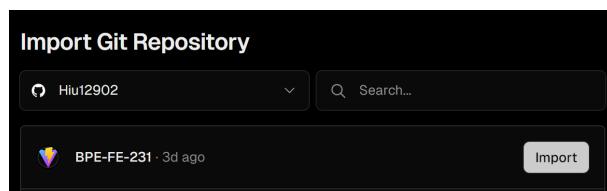
### 4.3.1 Triển khai phần Frontend của hệ thống

Hệ thống BPSky 2.0 được triển khai giao diện thông qua Vercel với đường dẫn (url) <https://bpsky232.vercel.app/>. Sau khi tạo tài khoản Vercel thành công chúng ta sẽ đến với giao diện quản lý những project cá nhân. Chọn nút "Add new" để mở menu lựa chọn, chúng ta sẽ chọn "Project" để tạo mới project.



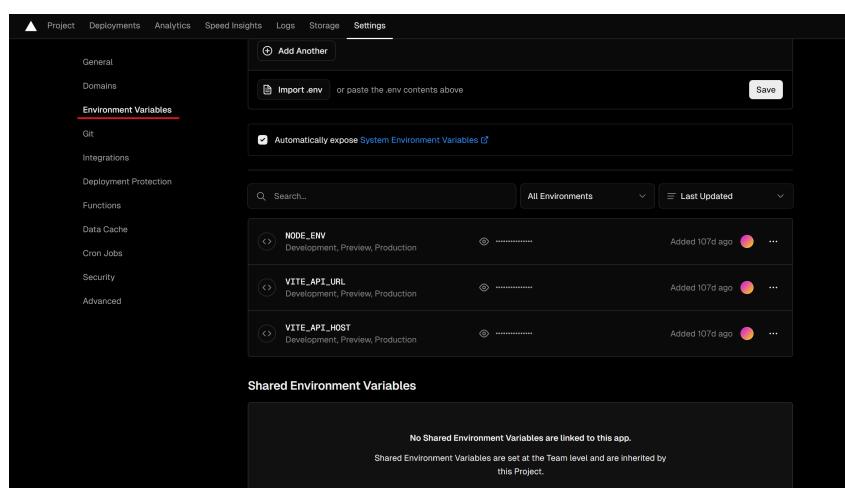
Hình 4.26: Giao diện tạo project mới trong Vercel

Sau khi liên kết với tài khoản Github thì chúng ta có thể kết nối với những repository đã tạo trước đó. Chọn "Import Git Repository" và chọn repository cần triển khai. Ở đây là repository "BPE-FE-231".



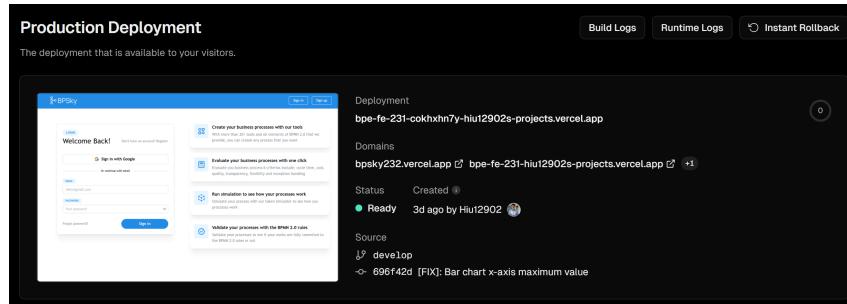
Hình 4.27: Giao diện import repository vào Vercel

Trước khi triển khai, chúng ta cần thiết lập môi trường cho project. Chọn "Environment Variables" để thêm các biến môi trường cần thiết.



Hình 4.28: Giao diện chỉnh sửa biến môi trường trong Vercel

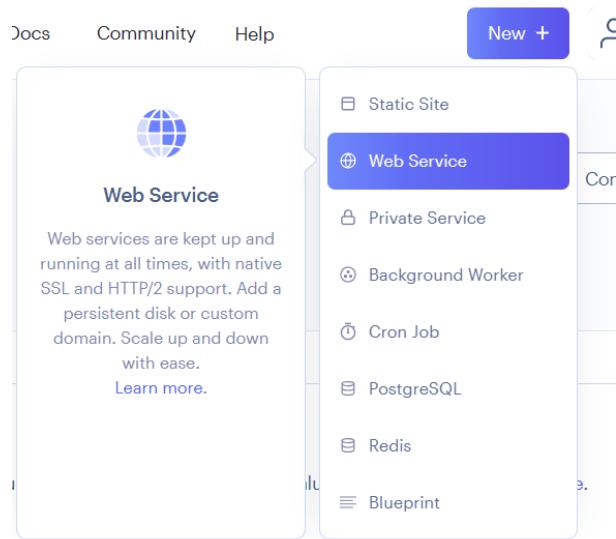
Cuối cùng, chúng ta chọn "Deploy" để triển khai project. Sau khi triển khai thành công, chúng ta sẽ nhận được đường dẫn truy cập vào project. Hệ thống BPSky 2.0 được triển khai thông qua URL: <https://bpsky232.vercel.app/>



Hình 4.29: Giao diện deploy thành công trên Vercel

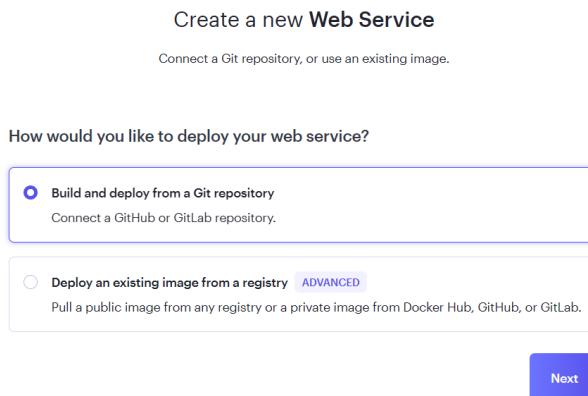
### 4.3.2 Triển khai phần Backend của hệ thống

Việc triển khai ở phía backend của ứng dụng sử dụng nền tảng Render như sau:  
Sau khi hoàn tất việc tạo tài khoản hoặc đăng nhập vào Render, chúng ta sẽ tạo một Web Service ở trên nền tảng này:



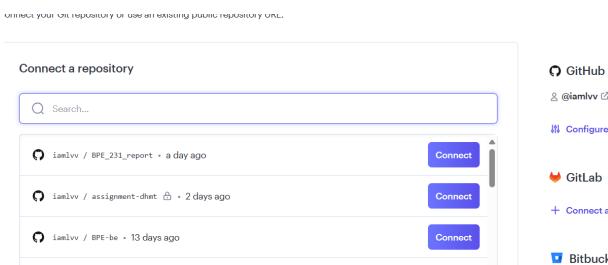
Hình 4.30: Giao diện tạo Web service trên Render

Sau đó, ta thiết lập kết nối giữa service trên Render với nơi chứa source code. Ở đây, Render cung cấp 2 lựa chọn là kết nối với Repository trên Github hoặc một Image trên các nền tảng khác. Nhóm chúng tôi lựa chọn phương án đầu tiên.



Hình 4.31: Chọn kết nối với Repository

Sau đó, chúng ta sẽ chọn kết nối đến một Repository cụ thể. Ở đây, chúng tôi chọn kết nối đến Repository BPE - be tương ứng với phần backend của hệ thống:



Hình 4.32: Kết nối với Repository của Backend

Sau đó, chúng ta cần điều chỉnh một số những thiết lập để service có thể tự triển khai trên source code của mình. Cụ thể, ta cần chọn nơi đặt server để chạy web service, hay nhánh (branch) trên repository được chọn để triển khai, thiết lập các biến môi trường cũng như câu lệnh để cài đặt các dependencies và khởi chạy server.

Name: BPE-be

Region: Singapore (Southeast Asia)

Branch: main

Root Directory: Optional (e.g., src)

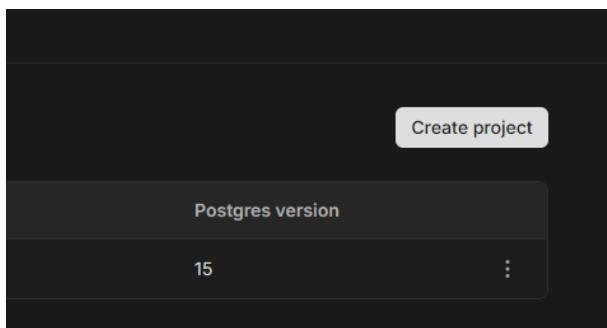
Hình 4.33: Thiết lập service

Render cũng cung cấp thêm một số những phiên bản trả phí. Với nhu cầu của chúng tôi đó là một máy chủ luôn trong trạng thái sẵn sàng cho các yêu cầu từ người dùng, và có khả năng lưu trữ file lâu dài, chúng tôi chọn sử dụng phiên bản trả phí Starter - đáp ứng được những nhu cầu trên.

Như vậy, chúng tôi đã triển khai phần backend của hệ thống lên một PaaS là Render với url <https://bpe.onrender.com>. Sau khi triển khai, chúng tôi có thể theo dõi những yêu cầu từ người dùng, cũng như những độ đo khác của hệ thống như băng thông, mức sử dụng CPU, RAM và một số những trạng thái khác.

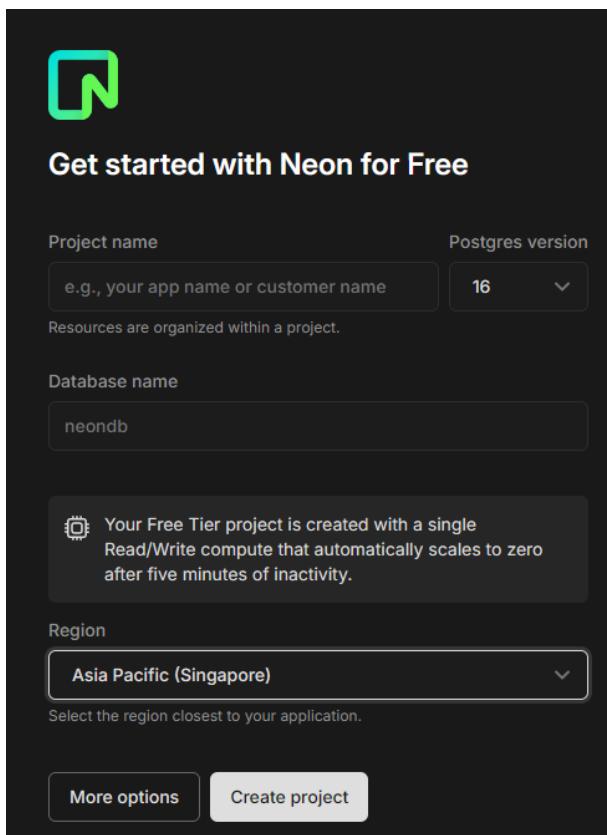
#### 4.3.3 Triển khai phần Database của hệ thống

Việc triển khai cơ sở dữ liệu của hệ thống được thực hiện trên nền tảng Neon.tech. Sau khi hoàn tất việc tạo tài khoản hoặc đăng nhập vào Neon.tech, chúng ta sẽ tạo một Database ở trên nền tảng này:



Hình 4.34: Giao diện tạo Database trên Neon.tech

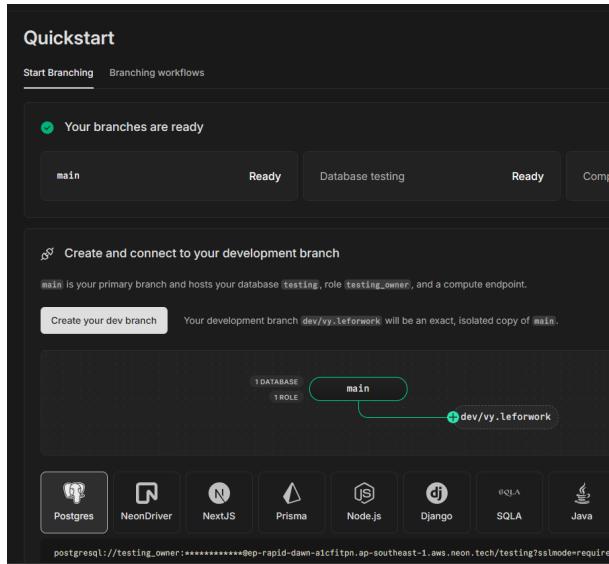
Sau đó, ta sẽ điều chỉnh một số thông tin về database, chẳng hạn như tên, server lưu trữ:



Hình 4.35: Điều chỉnh thông tin Database trên Neon.tech

Cuối cùng, việc tạo database đã hoàn tất. Neon.tech sẽ tạo ra đường dẫn để ta có thể kết nối tới

database này. Ngoài ra, Neon.tech còn hỗ trợ viết câu lệnh truy vấn, giúp ta có thể thao tác với database một cách dễ dàng.



Hình 4.36: Khởi tạo Database thành công

# Chương 5

## Kiểm thử và đánh giá hệ thống

### 5.1 Kiểm thử hệ thống

#### 5.1.1 Tổng quan các phương pháp kiểm thử

##### 5.1.1.1 Usecase testing

Kiểm thử phần mềm tập trung vào việc kiểm tra các tình huống hoặc trường hợp sử dụng cụ thể của một phần mềm, được quy định trong usecase scenario.

- **Ưu điểm**

Usecase testing thường dựa trên các kịch bản sử dụng đơn giản và dễ hiểu, giúp cho quá trình kiểm thử trở nên dễ dàng hơn cho cả nhóm phát triển và nhóm kiểm thử.

Unit testing là bước đệm giúp chúng ta thực hiện integration test. Phát hiện lỗi sớm sẽ giúp tiết kiệm thời gian và chi phí của cả quá trình kiểm thử về sau.

Hỗ trợ việc tái sử dụng testcase.

- **Nhược điểm**

Nếu usecase scenario không được viết rõ ràng hoặc không đầy đủ, việc thực hiện usecase testing có thể trở nên khó khăn và không chính xác.

Usecase cũng tập trung xác định những trường hợp được lên kế hoạch trước, sẽ khó khăn trong việc phát hiện các lỗi ẩn, nằm ngoài kế hoạch.

##### 5.1.1.2 Unit testing

Kiểm thử đơn vị là kiểm thử ở mức độ cơ bản, được thực hiện đối với từng module nhỏ trong hệ thống (hàm, lớp, phương thức). Kiểm thử đơn vị thông thường được thực hiện trực tiếp bởi đội ngũ lập trình viên mà không cần lưu lại hay quản lý ở các mức độ kiểm thử khác.



#### - Ưu điểm

Giúp chúng ta chỉnh sửa và tối ưu code dễ dàng hơn và an toàn hơn, việc có unit test cho từng module sẽ đảm bảo chức năng của các module không bị ảnh hưởng bởi thay đổi vì sẽ luôn được rà soát và kiểm tra.

Unit testing là bước đệm giúp chúng ta thực hiện integration test. Phát hiện lỗi sớm sẽ giúp tiết kiệm thời gian và chi phí của cả quá trình kiểm thử về sau.

Hỗ trợ việc tái sử dụng testcase.

#### - Nhược điểm

Không hỗ trợ phát hiện các lỗi liên quan tới việc tích hợp giữa nhiều module khác nhau hoặc việc tích hợp những logic từ ngữ cảnh vào.

### 5.1.1.3 Integration testing

Kiểm thử tích hợp là loại kiểm thử phần mềm hoặc chức năng được tích hợp logic và kiểm tra theo nhóm chung với nhau. Mức độ này được thực hiện bởi tester. Đây là loại kiểm thử tập trung vào việc kiểm tra giao diện (interface) giữa các thành phần trong hệ thống với nhau.

#### - Ưu điểm

Đảm bảo được tính tương thích hoạt động giữa những thành phần trong hệ thống.

Giúp phát hiện được những lỗi khi các module trong hệ thống hoạt động kết hợp với nhau, việc tích hợp logic vào kiểm thử cũng sẽ hỗ trợ bao phủ toàn bộ các trường hợp có thể xảy ra trên hệ thống.

#### - Nhược điểm

Chi phí dành cho kiểm thử tích hợp sẽ lớn hơn kiểm thử đơn vị.

Các lỗi xuất hiện trong giai đoạn tích hợp logic giữa nhiều module có thể khó khăn để phân tích và cần sự kết hợp giữa tester và developer.

### 5.1.1.4 System testing

Kiểm thử hệ thống là kiểm thử toàn bộ chức năng và giao diện của hệ thống. Sự khác nhau giữa system test và integration test là system test tập trung vào các hành vi và lỗi trên toàn bộ hệ thống, còn integration test thì chú trọng vào sự giao tiếp giữa các đơn thể hoặc đối tượng khi chúng hoạt động cùng với nhau. Chúng ta cần thông qua unit và integration test để đảm bảo mọi thành phần và sự tương tác giữa chúng hoạt động linh hoạt trước khi thực hiện system test.

#### - Ưu điểm

Đảm bảo tính toàn vẹn trong hoạt động của hệ thống từ những hành vi, chức năng đến các trường hợp gây ra ngoại lệ và lỗi.



#### - Nhược điểm

Chi phí để thực hiện system test sẽ nhiều hơn integration test, hơn nữa system test cần nên được thực hiện sau khi đã thông qua integration test và unit test để đảm bảo được hoạt động trên hệ thống ổn định và linh hoạt với các logic đầu vào khác nhau.

#### 5.1.1.5 Acceptance testing

Tương tự như system test nhưng acceptance testing thường được khách hàng trực tiếp thực hiện, mục đích là xem phần mềm có đáp ứng được yêu cầu khách hàng hay chưa.

Gồm 2 loại:

1. Alpha test được thực hiện bởi các thành viên quản lý sản phẩm (không liên quan trực tiếp đến dự án), một hình thức kiểm thử nội bộ trước khi tiến hành kiểm thử Beta.
2. Beta test là kiểm thử được thực hiện bởi người dùng cuối (thường là khách hàng), được thực hiện tại địa điểm của khách hàng, môi trường của riêng họ.

#### - Ưu điểm

Đảm bảo được những mục tiêu chính của sản phẩm đối với khách hàng, đây sẽ là bước cuối trước khi chính thức đưa sản phẩm đến tay của người dùng cuối.

Người dùng cuối sẽ đưa ra được nhiều góc nhìn mà người kiểm thử, người phát triển không thấy được.

#### - Nhược điểm

Acceptance test sẽ được thực hiện kiểm thử bởi người dùng cuối, vì vậy kết quả cuối cùng sẽ là thứ xoay quanh nhu cầu của họ. Điều đó đồng nghĩa sẽ có nhiều flow bị bỏ qua.

Việc thực hiện acceptance test ảnh hưởng nhiều bởi cá nhân người kiểm thử, QA hay PM biết quá nhiều về cách hoạt động của dự án, khó để đảm bảo background của người kiểm thử tương ứng với một khách hàng thật sự.

### 5.1.2 Thực hiện kiểm thử

Sau khi cân nhắc những ưu, nhược điểm cũng như yêu cầu về thời gian và nhân lực của các phương pháp kiểm thử, nhóm quyết định sử dụng phương pháp kiểm thử Usecase testing để kiểm thử hệ thống.

Dưới đây là tổng hợp các trường hợp kiểm thử dựa trên usecase, nội dung chi tiết được tổng hợp tại đường dẫn [https://docs.google.com/spreadsheets/d/1yDyn8-fa67z1SFjvmJKY9pEwQEnEU2gvkhm\\_gCw2BHo/edit#gid=0](https://docs.google.com/spreadsheets/d/1yDyn8-fa67z1SFjvmJKY9pEwQEnEU2gvkhm_gCw2BHo/edit#gid=0).



STT	Tên usecase	STT	Tên usecase
1	Truy cập trang quản lý workspace	26	Kích hoạt process version
2	Truy cập workspace	27	Khởi tạo process portfolio
3	Tạo mới workspace	28	Chỉnh sửa thông tin của process version bị khuyết giá trị
4	Xóa workspace	29	Tạo mới bảng khảo sát
5	Sử dụng bộ lọc tìm kiếm workspace	30	Đóng bảng khảo sát
6	Ghim workspace	31	Xóa bảng khảo sát
7	Thay đổi tên workspace	32	Cấu hình bảng khảo sát
8	Tìm kiếm workspace	33	Cấu hình câu hỏi
9	Chia sẻ workspace	34	Tạo mới câu hỏi
10	Tìm kiếm project trong workspace	35	Xóa câu hỏi
11	Loại thành viên khỏi workspace	36	Thực hiện bảng khảo sát
12	Thay đổi quyền truy cập của thành viên	37	Quản lý kết quả bảng khảo sát
13	Sử dụng bộ lọc tìm kiếm thành viên	38	Di chuyển vị trí câu hỏi
14	Mời người dùng khác vào workspace	39	Chế độ xem trước bảng khảo sát
15	Quản lý thành viên trong workspace	40	Công bố bảng khảo sát
16	Tìm kiếm thành viên trong workspace	41	Chấp nhận lời mời vào workspace
17	Sử dụng bộ lọc tìm kiếm yêu cầu	42	Xem nội dung chi tiết thông báo
18	Truy cập trang quản lý yêu cầu trong workspace	43	Xem thông báo
19	Chấp nhận yêu cầu từ thành viên trong workspace	44	Từ chối lời mời vào workspace
20	Từ chối yêu cầu từ thành viên trong workspace	45	Lọc thông báo
21	Xóa yêu cầu từ thành viên trong workspace	46	Ghim thông báo
22	Nhận yêu cầu từ thành viên trong workspace	47	Nhận thông báo từ hệ thống
23	Tìm kiếm yêu cầu trong workspace	48	Tìm kiếm thông báo
24	Chỉnh sửa giá trị thang đo của workspace	49	Xóa thông báo
25	Chỉnh sửa thông tin của process version		

Bảng 5.1: Tổng hợp các trường hợp kiểm thử usecase

## 5.2 Case study

Nhóm quyết định chọn quy trình đổi vé máy bay để làm case study cũng như kiểm chứng kết quả tính toán được sử dụng cơ sở lý thuyết được trình bày ở trên, so với kết quả thực tế của hệ thống.

### 5.2.1 Thu thập dữ liệu khảo sát

STT	Câu hỏi
1	Do you use any services or products provided by this process?
2	What is your role in this process?
3	To what extent do you agree with this statement: "It is easy to handle exceptions in your process".
4	To what extent do you agree with this statement: "It is easy to execute your process."
5	How frequently do exceptions happen in practice?
6	Which handled exceptions do you think the way to handle them needs optimizing? Can you suggest some solutions to those?
7	Can you suggest any exceptions that need handling and the way to handle them?
8	How satisfied are you with the way exceptions are handled in your process?
9	How satisfied are you with the amount of time you have to spend on this process?
10	A flexible process means it has variations for different conditions. How satisfied are you with those variations of the process?
11	How satisfied are you with the cost of products or services of this process?
12	How satisfied are you with the products or services provided by this process?
13	How satisfied are you with the total time you have to spend on waiting for the output of this process?
14	Do you encounter any problems while using products or services provided by this process?
15	How likely is it that you would recommend this process to others?

Bảng 5.2: Bảng câu hỏi khảo sát

Đây là tất cả câu hỏi mẫu có sẵn trong bảng khảo sát, được thiết kế để có thể được sử dụng ở nhiều loại quy trình tương ứng với những nghiệp vụ khác nhau. Ở đây, chúng tôi sẽ kiểm chứng kết quả đánh giá của hệ thống, so với kết quả chúng tôi tự tính toán được bằng tay. Trong trường hợp này, chúng tôi sẽ lấy dữ liệu từ 10 người dùng khác nhau. Và để kiểm chứng kết quả, chúng tôi sẽ chỉ lấy dữ liệu ở những loại câu hỏi được sử dụng để tính giá trị các độ đo cũng như kết quả cuối cùng của bài khảo sát. Cụ thể, chúng tôi sẽ quan tâm dữ liệu ở các câu:

- Loại câu hỏi rẽ nhánh: câu 1
- Loại câu hỏi CES: câu 3, câu 4
- Loại câu hỏi CSAT: câu 8, câu 9, câu 10, câu 11, câu 12, câu 13



#### 4. Loại câu hỏi NPS: câu 15

Ở đây, để dễ dàng cho việc tính toán bằng tay, chúng tôi thiết lập trọng số cho mỗi câu hỏi trong mỗi nhóm độ đo là như nhau. Bên dưới là dữ liệu thu thập được từ bảng khảo sát:

Người dùng	Nhóm	Câu 3	Câu 4	Câu 8	Câu 9	Câu 10	Câu 11	Câu 12	Câu 13	Câu 15
1	user	6	5	7	4	3				8
2	guest						7	6	4	9
3	guest						6	7	5	6
4	user	7	4	2	4	5				5
5	guest						5	7	7	10
6	user	7	7	5	6	4				9
7	user	6	6	7	2	7				8
8	user	5	4	5	3	5				7
9	user	6	7	6	7	4				9
10	guest						7	5	6	8

Bảng 5.3: Bảng chi tiết lựa chọn của người dùng cho mỗi câu hỏi

#### 5.2.2 Kiểm chứng dữ liệu khảo sát

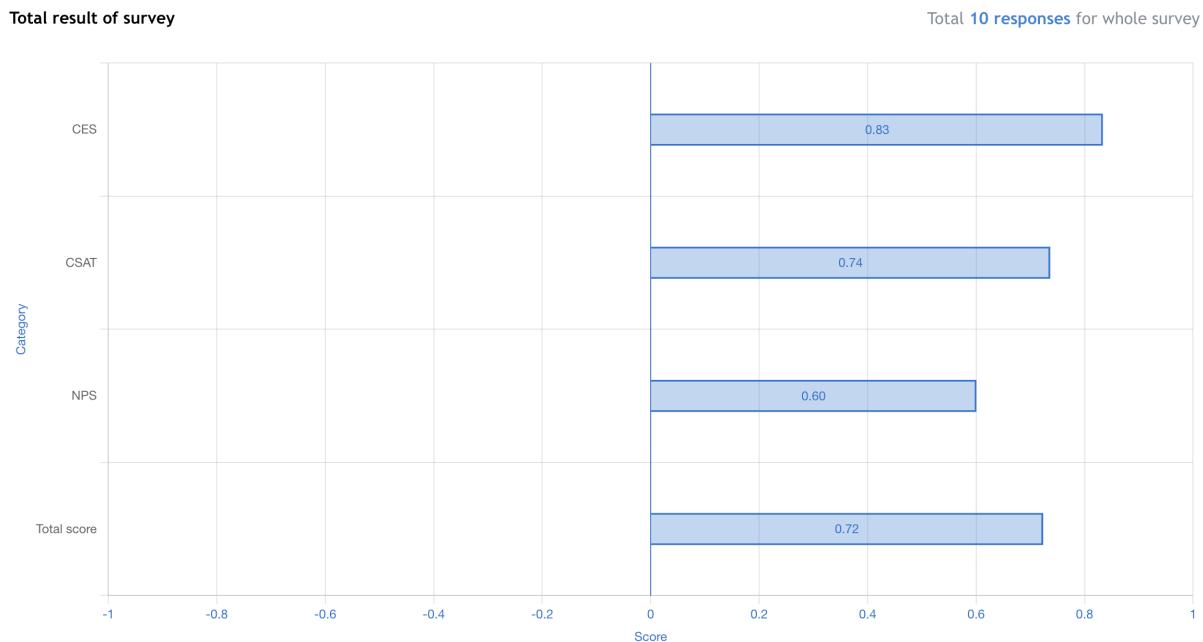
Như vậy, theo như lý thuyết về các độ đo trong bảng khảo sát, ta có thể tính toán được các biến số cần để tính ra giá trị cuối cùng của mỗi độ đo. Nhắc lại, số người đồng ý với câu hỏi (CES) hay số người hài lòng (CSAT) được tính khi người đó cho số điểm nằm trong khoảng [5, 7] cho câu hỏi CES hoặc CSAT. Số người thuộc nhóm Promoters khi họ cho điểm cho câu hỏi NPS thuộc tập giá trị 9, 10 và thuộc nhóm Detractors khi họ cho điểm cho câu hỏi NPS thuộc tập giá trị {0, 6}.

Câu hỏi	Phân loại	Số người đồng ý với câu hỏi (CES)	Số người hài lòng (CSAT)	Số người thuộc nhóm Promoters	Số người thuộc nhóm Detractors	Điểm
3	CES	6				0.6
4	CES	4				0.4
8	CSAT		5			0.5
9	CSAT		2			0.2
10	CSAT		3			0.3
11	CSAT		4			0.4
12	CSAT		4			0.4
13	CSAT		3			0.3
15	NPS			4	2	0.2

Bảng 5.4: Bảng kết quả tính toán mỗi câu hỏi

Mỗi câu hỏi đều có trọng số như nhau, và mỗi độ đo đều có trọng số như nhau trong bảng khảo sát,

như vậy ta có điểm số của từng độ đo và của bài khảo sát sau khi đã chuẩn hóa về cùng miền giá trị [0, 1]: CES = 0.833, CSAT = 0.736, NPS = 0.6. Từ đó, ta tính được điểm tổng của bài khảo sát là 0.72. Đôi chiểu với kết quả tính được từ hệ thống như sau:



**Hình 5.1:** Kết quả bài khảo sát cho quy trình đổi vé máy bay

Từ kết quả trên, ta thấy rằng kết quả của hệ thống và kết quả tính toán bằng tay khá giống nhau, với điểm số của hệ thống là 0.72, và điểm số tính bằng tay của bài khảo sát là 0.72. Điểm tổng bài khảo sát cũng là giá trị External Quality của quy trình nghiệp vụ. Giả sử 2 loại quality đều có trọng số như nhau, ta có giá trị trung bình của độ đo chất lượng của quy trình nghiệp vụ: 0.711.

### Lane's evaluated result

Start from Lane	Cycle time	Quality
Customer	31.00	<a href="#">More detail</a>

### Quality detail

Total quality	Internal Quality	External Quality
0.7115	0.7	0.723

**Hình 5.2:** Kết quả đánh giá chất lượng của quy trình đổi vé máy bay

Như vậy, ta thấy rằng kết quả đánh giá chất lượng của hệ thống và kết quả tính toán bằng tay khá

giống nhau, với điểm số của hệ thống là 0.7115, và điểm số của tính toán bằng tay là 0.711.

### 5.2.3 Kiểm chứng process portfolio

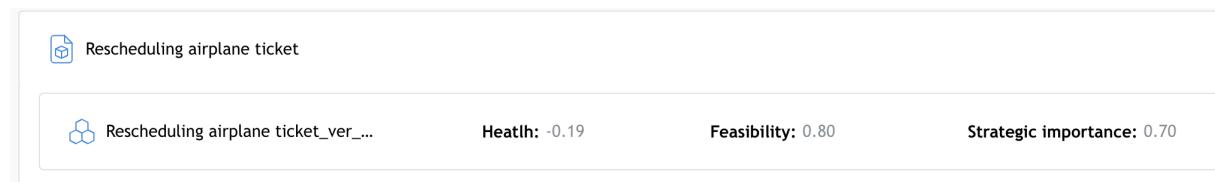
Để tạo ra được process portfolio cho các quy trình nghiệp vụ đang hoạt động trong Workspace, ta cần một số những dữ liệu sau:

1. Các giá trị mục tiêu, tệ nhất của các độ đo của Workspace, bao gồm targeted cycle time, worst cycle time, targeted cost, worst cost, targeted quality, worst quality, targeted flexibility, worst flexibility, giá trị của các độ đo feasibility, strategic importance
2. Các giá trị thực tế của các yếu tố trong độ đo health, bao gồm cycle time, cost, quality, flexibility.

Yếu tố khảo sát	Giá trị
Targeted cycle time	25
Worst cycle time	50
Targeted cost	20
Worst cost	30
Targeted quality	0.8
Worst quality	0.4
Targeted Flexibility	0.95
Worst Flexibility	0.65
Feasibility	0.8
Strategic Importance	0.7
Current cycle time	30.5
Current cost	28
Current quality	0.7
Current flexibility	0.7

Bảng 5.5: Bảng các giá trị mẫu của các yếu tố trong process portfolio

Vận dụng cơ sở lý thuyết được trình bày ở trên, ta tính được các độ đo của quy trình nghiệp vụ để xây dựng process portfolio: Health = -0.19, Feasibility = 0.8, Strategic Importance = 0.7.



Hình 5.3: Kết quả đánh giá process portfolio của Workspace

Như vậy, ta thấy rằng kết quả đánh giá process portfolio của hệ thống và kết quả tính toán bằng tay khá giống nhau, với điểm số của hệ thống là -0.19, và điểm số của tính toán bằng tay là -0.19. Với những



kết quả trên, ta có thể kết luận rằng hệ thống đánh giá chất lượng của quy trình nghiệp vụ và process portfolio của Workspace hoạt động hiệu quả và chính xác.

## **Chương 6**

# **Kết luận**

### **6.1 Kết quả đạt được và điểm hạn chế**

Hệ thống BPSky đã trải qua hai giai đoạn phát triển chính. Trong giai đoạn trước đó, hệ thống tập trung vào cung cấp cho người dùng bộ công cụ cho phép chỉnh sửa, thiết kế quy trình nghiệp vụ dưới dạng mô hình thông qua ngôn ngữ mô hình hóa BPMN. Ngoài ra, hệ thống cũng đã thành công phát triển khung giải thuật và hiện thực tính năng cho phép người dùng đánh giá chất lượng quy trình một cách tự động. Tuy nhiên, những yếu tố như phạm vi quản lý vẫn có thể được cải tiến thêm, thay vì chỉ quản lý quy trình trong không gian làm việc của từng cá nhân thì chúng tôi mong muốn tạo một không gian chung cho phép nhiều người tham gia chia sẻ; ngoài ra tính khách quan của kết quả đánh giá chất lượng vẫn còn có phần hạn chế, hệ thống tính toán dựa trên mô hình được thiết kế nên sẽ là thiếu sót nếu quyết định chất lượng của quy trình là tốt hay chưa. Vì vậy trong giai đoạn phát triển này, hệ thống đã tập trung cải tiến 3 yếu tố chính:

1. Chúng tôi đã hiện thực mở rộng không gian quản lý các dự án và các quy trình nghiệp vụ, cho phép người dùng chia sẻ không gian để mở rộng và cộng tác với nhiều thành viên khác trong hệ thống. Chủ sở hữu không gian làm việc chung giờ đây có thể quản lý thành viên cũng như xử lý được các yêu cầu của thành viên gửi tới liên quan tới phân quyền và chia sẻ đến thành viên khác ngoài không gian chung.
2. Chúng tôi đã hiện thực và tích hợp tính năng khảo sát vào việc thiết kế mô hình quy trình nghiệp vụ, bây giờ người dùng không chỉ xây dựng mô hình quy trình nghiệp vụ mà còn có thể tham gia tùy biến khảo sát để thu thập thêm góc nhìn từ người tham gia quy trình hoặc người sử dụng sản phẩm của quy trình tạo ra.
3. Chúng tôi đã hiện thực danh mục quy trình nghiệp vụ (process portfolio) cho mỗi không gian làm việc (workspace), danh mục quy trình cho phép người dùng thu thập cũng như điều chỉnh thông tin một cách tổng quan về toàn bộ những phiên bản quy trình nghiệp vụ đang hoạt động trong hệ thống; danh mục hỗ trợ lược đồ trực quan hóa mức độ ưu tiên giữa các quy trình dựa trên những



tiêu chí mà hệ thống đề xuất.

Qua đó, chúng tôi đã cung cấp một hệ thống quản lý quy trình nghiệp vụ mở rộng hơn về phạm vi quản lý, phạm vi cộng tác, cũng như cung cấp thêm những công cụ hỗ trợ người dùng trong việc đánh giá quy trình nghiệp vụ.

Tuy nhiên, đồ án tốt nghiệp vẫn còn một số hạn chế. Thứ nhất, hệ thống thông báo chỉ mới có hai loại thông báo cơ bản là thông báo về lời mời tham gia Workspace và thông báo trạng thái của yêu cầu thay đổi quyền hạn của bản thân người dùng trong Workspace, và chúng tôi cho rằng cần phải mở rộng hệ thống thông báo để người dùng có thể nhận được thông báo về những hoạt động quan trọng khác trong Workspace cũng như trong hệ thống. Thứ hai, chúng tôi chưa xử lý hết những trường hợp phân quyền phức tạp hơn, liên quan đến quyền hạn trong Workspace và Project. Thứ ba, tính năng thiết kế bảng khảo sát vẫn còn một số hạn chế, chẳng hạn như chưa linh hoạt cho phép người dùng tạo thêm các section, hay di chuyển câu hỏi giữa các section. Thứ tư, tính năng quản lý kết quả bảng khảo sát chưa có thêm những chức năng như xuất file báo cáo theo các định dạng pdf, csv, bộ lọc chi tiết kết quả thực hiện.

## 6.2 Hướng phát triển

- Tiếp tục mở rộng hệ thống, hướng đến việc phân loại các quy trình nghiệp vụ theo từng lĩnh vực cụ thể.
- Nghiên cứu, phát triển các tính năng liên quan đến việc đề xuất các mẫu quy trình thuộc lĩnh vực liên quan đến quy trình hiện tại hoặc nhu cầu của người dùng.
- Mở rộng thêm các tính năng đóng góp, cho phép người dùng đóng góp những quy trình, góp phần xây dựng cộng đồng phát triển quy trình nghiệp vụ.
- Tổ chức Workspace, Project chặt chẽ hơn, phát triển thêm những tính năng quản trị.
- Nghiên cứu, mở rộng thêm các hướng tiếp cận để đánh giá quy trình nghiệp vụ khách quan và chính xác hơn.
- Cho phép tích hợp thêm các công cụ mở rộng có sẵn trên thị trường để phục vụ cho quá trình đánh giá quy trình nghiệp vụ.

# Tài liệu tham khảo

- [1] Ably, *Socket.IO vs. WebSocket: Key differences and which to use*, [Online, URL: <https://ably.com/topic/socketio-vs-websocket>, truy cập ngày 12-10-2023].
- [2] Akanbi, O. A., Amiri, I. S., & Fazeldehkordi, E. (2015), *Feature Extraction. A Machine-Learning Approach to Phishing Detection and Defense*.
- [3] Asier Baquero (2022), *Net Promoter Score (NPS) and Customer Satisfaction: Relationship and Efficient Management*, Social Sciences, Law and Business Administration Department, Catholic University of Murcia (UCAM), 30107 Murcia, Spain.
- [4] Cao, X. H., Stojkovic, I., & Obradovic, Z. (2016), *A robust data scaling algorithm to improve classification accuracies in biomedical data*, BMC Bioinformatics.
- [5] Facebook Inc., Reactjs [Online, URL: <https://react.dev/>, truy cập vào ngày 15-12-2023].
- [6] Grigore Raileanu, *NPS, CSAT and CES - Customer Satisfaction Metrics to Track in 2024*, [Online, URL: <https://www.retently.com/blog/customer-satisfaction-metrics/>, truy cập ngày 9-2-2024].
- [7] Grigore Raileanu, *CSAT: Definition, Calculation & 2023 Benchmarks*, [Online, URL: <https://www.retently.com/blog/customer-satisfaction-score-csat/>, truy cập ngày 1-10-2023].
- [8] Grigoroudis, E., & Siskos, Y. (2002), *Preference disaggregation for measuring and analysing customer satisfaction: The MUSA method*, European Journal of Operational Research.
- [9] Jérôme Navez (2023), *Make your microservices tastier by using the Onion architecture*, [Online, URL: <https://dev.to/jnavez/make-your-microservices-tastier-by-cooking-them-with-a-sweet-onion-34n2>, truy cập ngày 24-10-2023].
- [10] Jon A. Krosnick and Stanley Presser (2010), *Question and Questionnaire Design*, Handbook of Survey Research, Second Edition.
- [11] Kim, S., Lee, J., & Gweon, G. (2019), *Comparing Data from Chatbot and Web Surveys*.
- [12] Krol, M. W., de Boer, D., Delnoij, D. M., & Rademakers, J. J. D. J. M. (2014), *The Net Promoter Score - an asset to patient experience surveys?*, Health Expectations.

- [13] Mark Richards, Neal Ford, (2020), *Fundamentals of Software Architecture - An engineering approach*, O'reilly Media, Inc.
- [14] Marlon Dumas, Marcello La Rosa, Jan Mendling, Hajo A. Reijers, (2018), *Fundamentals of Business Process Management*, Springer.
- [15] Microsoft, *N-tier architecture style*, [Online, URL: <https://learn.microsoft.com/en-us/azure/architecture/guide/architecture-styles/n-tier>, truy cập ngày 15-09-2023].
- [16] Mihelis, G., Grigoroudis, E., Siskos, Y., Politis, Y., & Malandrakis, Y. (2001), *Customer satisfaction measurement in the private bank sector*, European Journal of Operational Research, 130(2).
- [17] Moira Clark, Andrew Bryan (2013), *Customer Effort: Help or hype?*, Henley Business School.
- [18] Nguyễn Cảnh Hoàng, Nguyễn Ngọc Hiển, Nguyễn Việt Anh (2023), *Ứng dụng hỗ trợ đánh giá quy trình nghiệp vụ*, Đại học Bách Khoa Đại học Quốc gia thành phố Hồ Chí Minh.
- [19] PostgreSQL, *PostgreSQL 14.1 Documentation*, [Online, URL: <https://www.postgresql.org/>, truy cập ngày 12-09-2023].
- [20] Python, *Python 3.10.2 Documentation*, [Online, URL: <https://www.python.org/>, truy cập ngày 12-09-2023].
- [21] Qualtrics, *What is a Net Promoter Score?*, [Online, URL: <https://www.qualtrics.com/au/experience-management/customer/measure-nps/>, truy cập ngày 21-09-2023].
- [22] Rallis, I., Markoulidakis, I., Georgoulas, I., & Kopsiaftis, G. (2020), *A novel classification method for customer experience survey analysis*.
- [23] Rasmus, GD, Svensson, *Contextualizing Customer Feedback: A Research-through-Design Approach*, Uppsala University, Department of Informatics and Media.
- [24] Regmi, P. R., Waithaka, E., Paudyal, A., Simkhada, P., & Van Teijlingen, E. (2017), *Guide to the design and application of online questionnaire surveys*, Nepal Journal of Epidemiology.
- [25] Rotella, P., & Chulani, S. (2011), *Implementing quality metrics and goals at the corporate level*.
- [26] Peter Schuurman, Egon Berghout, & Philip Powell (2008), *Calculating the Importance of Information Systems: The Method of Bedell Revisited*.
- [27] Jennifer Horkoff, Daniele Barone, Lei Jiang, Eric Yu, Daniel Amyot, Alex Borgida, & John Mylopoulos (2012), *Strategic business modeling: representation and reasoning*.