**PROJECT**

**Quản lí kho quần áo**

**MỤC LỤC**

[**I.** **Tổng quan** 2](#_Toc48651552)

[**1.** **Mục đích** 2](#_Toc48651553)

[**2.** **Phạm vi ảnh hưởng** 2](#_Toc48651554)

[**3.** **Ghi chú** 3](#_Toc48651555)

[3.1. Giải thích ký hiệu 3](#_Toc48651556)

[3.2. Thuật ngữ 3](#_Toc48651557)

[**II.** **Mô tả chi tiết** 4](#_Toc48651558)

[**1.** **Quy trình** 4](#_Toc48651559)

[1.1. Flow (BPMN) 4](#_Toc48651560)

[1.2. Mô tả ngắn 4](#_Toc48651561)

[**2.** **Nghiệp vụ chi tiết** 5](#_Toc48651562)

[2.1 Quy trình 1. 5](#_Toc48651563)

[2.2 Quy trình 2. 11](#_Toc48651564)

[2.3 Quy trình 3 11](#_Toc48651565)

[2.4 Quy trình 4 11](#_Toc48651566)

[2.5 Quy trình 5 11](#_Toc48651567)

[2.6 Quy trình 6 11](#_Toc48651568)

[**III.** **Tài liệu liên quan** 11](#_Toc48651569)

[**IV.** **Lịch sử chỉnh sửa** 11](#_Toc48651570)

# **Tổng quan**

## **Mục đích**

Ngày nay khoa học kỹ thuật phát triển với tốc độ một cách nhanh chóng, phát triển về cả chiều sâu lẫn chiều rộng. Nó góp phần vào sự phát triển của nền văn minh loài người. Con người đã biết áp dụng công nghệ thông tin phục vụ cho đời sống của mình. Mọi người mọi nhà dù cách xa bao nhiêu họ vẫn gần nhau nhờ Internet.

Việc thống kê hay kiểm soát các bản lưu trữ lớn trên giấy tờ hay các bản file exel trở nên khá khó khăn cho người quản lí, việc truy xuất dữ liệu, thêm sửa xóa dữ liệu và liên kết giữa các bản ghi lại càng khó khăn hơn. Vì thế, nhu cầu của người dùng là muốn có 1 hệ thống quản lí và kiểm soát tốt tất cả những điều trên, tránh sai sót, giao diện đơn giản dễ thao tác và dễ sử dụng nên chúng ta cần áp dụng mạnh mẽ khoa học công nghệ vào các yêu cầu thực tiễn và thiết thực để làm cho cuộc sống trở nên đơn giản và tiện ích hơn.

Một hệ thống quản lí sẽ giúp ích cho người sử dụng như:

+ Dễ dàng quản lí, kiểm soát việc thêm, sửa, xóa, cập nhật các dữ liệu cho kho hàng của mình, tránh mất mát hay sai sót dữ liệu.

+ Dễ dàng truy xuất dữ liệu và đưa ra các đánh giá để có biện pháp khắc phục và phát triển.

+ Dễ quản lí được, phân chia các công việc của từng cá nhân với các nhiệm vụ hay các quyền khác nhau khi quản lí dữ liệu.

+ Dễ dàng thống kê và đưa ra đánh giá, nhận xét hoặc khắc phúc những điểm yếu còn tồn tại (ví dụ: thống kê các mặt hàng khách thường mua và nhận xét của khách hàng để tổng hợp đánh giá)

## **Phạm vi ảnh hưởng**

Liệt kê các phần ảnh hưởng trên hệ thống như các module tính năng,

phân hệ phát triển:

* Quản lí nhập kho:

+ Gồm đơn nhập hàng.

+ thanh toán với nhà cung cấp.

+ Kiểm hàng, nhập kho.

* Quản lí nhà cung cấp:

+ Thêm nhà cung cấp khi nhà cung cấp ấy chưa có trên hệ thống.

+ Cập nhật thông tin nhà cung cấp khi có thay đổi.

+ Xóa nhà cung cấp khi không còn hợp tác hoặc nhà cung cấp phá sản.

* Quản lí sản phẩm:

+ Thêm sản phẩm mới khi sản phẩm ấy mới xuất hiện tại kho.

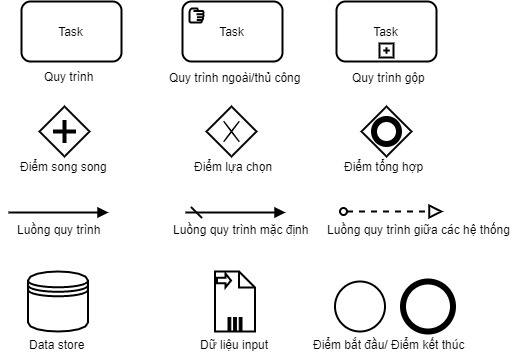
+ Cập nhật sản phẩm khi có thay đổi, cập nhật số lượng sản phẩm khi có hàng được nhập thêm vào kho từ nhà cung cấp.

+ Xóa sản phẩm khi sản phẩm ấy không còn được bán.

## **Ghi chú**

### Giải thích ký hiệu

* Liệt kê ra toàn bộ các icon ký hiệu sử dụng trên các biểu đồ

****

### Thuật ngữ

* Liệt kê các thuật ngữ hoặc từ viết tắt sử dụng trong tài liệu
* Từ t.anh, viết tắt, chuyên ngành

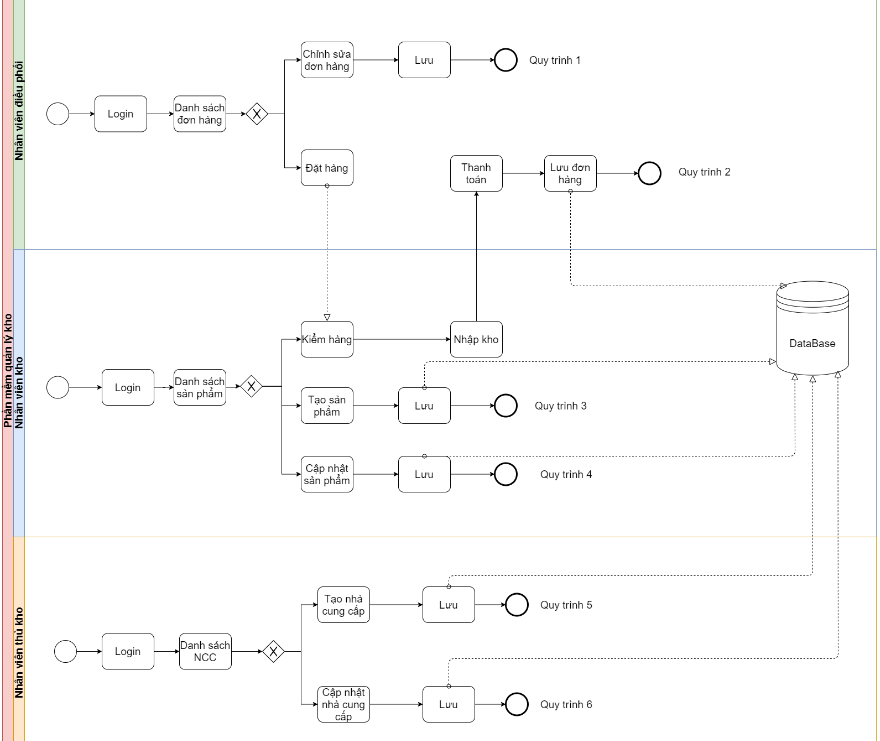
|  |  |
| --- | --- |
| **Tên thuật ngữ** | **Định nghĩa** |
| NCC | Nhà cung cấp |
| product | Sản phẩm |
| supplier | Nhà cung cấp |
| receipt | Đơn đặt hàng |
| DB | Database |

# **Mô tả chi tiết**

## **Quy trình**

### Flow (BPMN)

* Vẽ một biểu đồ tổng quan quy trình của tính năng



### Mô tả ngắn

* Liệt kê ra các bước thực hiện trên quy trình.

|  |  |  |
| --- | --- | --- |
| **Quy trình** | **Đối tượng thực hiện** | **Mô tả** |
| Quy trình 1 | Nhân viên điều phối | + Xem danh sách đơn hàng.  + Chỉnh sửa đơn hàng cụ thể. |
| Quy trình 2 | Nhân viên điều phối  Nhân viên kho | + Xem danh sách đơn hàng.  + Tạo đơn hàng.  + Kiểm hàng.  + Nhập kho.  + Thanh toán |
| Quy trình 3 | Nhân viên kho | + Xem danh sách sản phẩm.  + Tạo sản phẩm mới khi có hàng về. |
| Quy trình 4 | Nhân viên kho | + Xem danh sách sản phẩm.  + Cập nhật sản phẩm, xóa sản phẩm. |
| Quy trình 5 | Nhân viên thủ kho | + Xem danh sách NCC.  + Tạo mới NCC. |
| Quy trình 6 | Nhân viên thủ kho | + Xem danh sách NCC.  + Cập nhật NCC, xóa NCC. |

* Cấu trúc API.

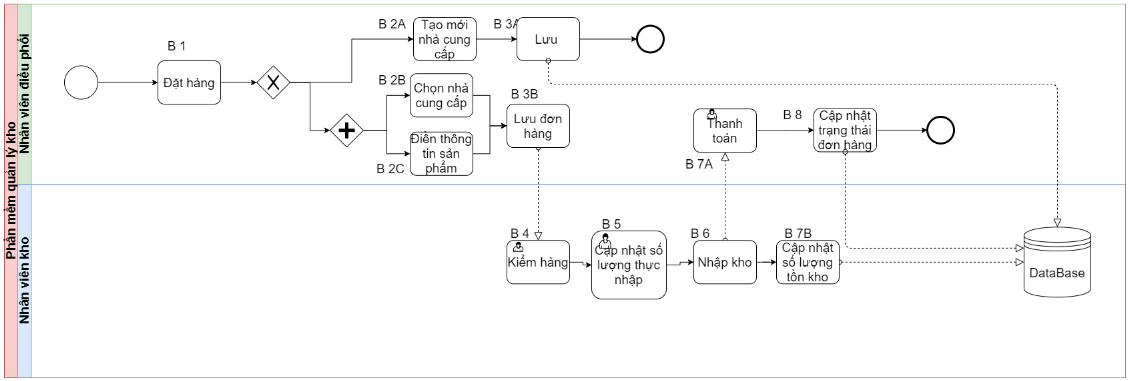
|  |  |  |  |
| --- | --- | --- | --- |
|  | method | url | description |
| /receipt | POST | / | Thêm mới đơn nhập hàng |
|  | GET | ?limit=?&page=? | Lấy danh sách đơn nhập hàng và phân trang |
|  | GET | /id | Lấy chi tiết đơn nhập hàng |
|  | PUT | /id | Cập nhật đơn hàng |
|  | DELETE | /id | Xóa đơn nhập hàng |
|  | GET | ? limit=?&page=?&search=?&sort=? | Tìm kiếm theo tên, sắp xếp. |
| /product | POST | / | Thêm mới sản phẩm |
|  | GET | ?limit=?&page=? | Lấy danh sách sản phẩm và phân trang |
|  | GET | /id | Lấy chi tiết sản phẩm |
|  | PUT | /id | Cập nhật sản phẩm |
|  | DELETE | /id | Xóa sản phẩm theo id |
|  | GET | ? limit=?&page=?&search=?&sort=? | Tìm kiếm theo tên, sắp xếp. |
| /supplier | POST | / | Thêm mới NCC |
|  | GET | ?limit=?&page=? | Lấy danh sách NCC và phân trang |
|  | GET | /id | Lấy chi tiết NCC |
|  | PUT | /id | Cập nhật NCC |
|  | DELETE | /id | Xóa NCC |
|  | GET | ? limit=?&page=?&search=?&sort=? | Tìm kiếm theo tên, sắp xếp. |

## **Nghiệp vụ chi tiết**

### 2.1 Quy trình 1.

2.1.1 Mã màn hình, tên màn hình.

- Flow:

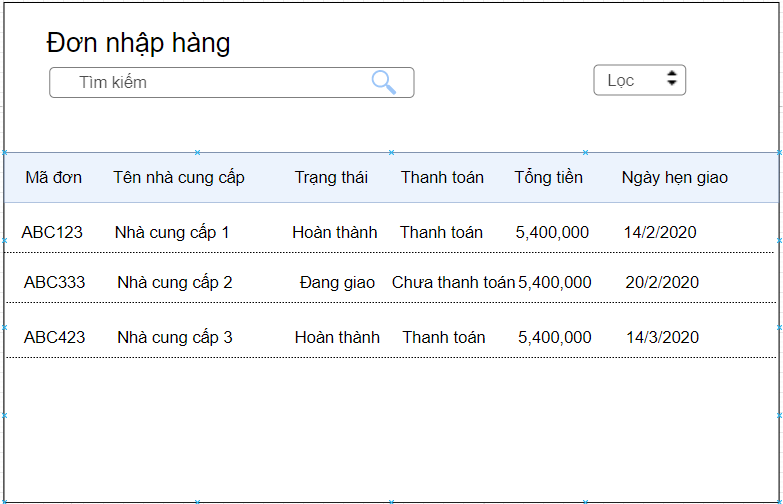


- Các bước thực hiện:

|  |  |  |
| --- | --- | --- |
| **Bước** | **Đối tượng thực hiện** | **Mô tả** |
| B 1 | Nhân viên điều phối | Xem danh sách đơn đặt hàng |
| B 2A | Nếu NCC chưa tồn tại thì tạo mới NCC |
| B 3A | Lưu NCC mới vào DB |

2.1.2 Mô tả nghiệp vụ.

* Danh sách đơn đặt hàng:
* Màn hình giao diện.



* Mô tả: Danh sách sẽ hiển thị các thông tin liên quan đến các đơn đặt hàng với các nhà cung cấp đã được tạo gồm có các trường như:

+ Mã đơn.

+ Tên nhà cung cấp.

+ Trạng thái.

+ Thanh toán.

+ Tổng tiền.

+ Ngày hẹn giao.

Và có thể tìm kiếm theo tên NCC hoặc mã đơn, lọc theo tên từ A-Z, Z-A, theo trạng thái và theo thanh toán.

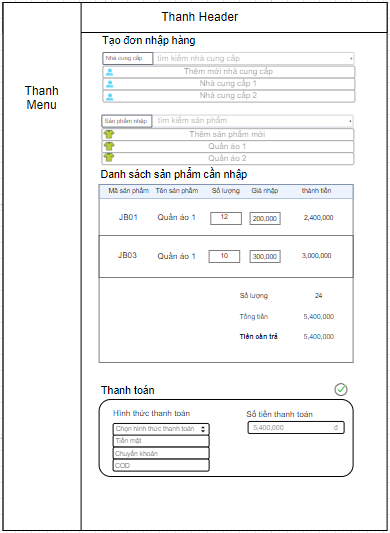
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Đơn nhập hàng | GET: /api/receipt?limit=?&page=? |  |  |

* Logic xử lí:

+ Khi hiển thị danh sách đơn nhập hàng, hệ thống sẽ gọi đến server để lấy dữ liệu từ database về.

+ Khi có dữ liệu thì phần fontend sẽ tiến hàng xử lí và hiển thị dữ liệu ra bảng cho người dùng xem.

* Đơn đặt hàng:
* Màn hình giao diện đơn nhập hàng.



* Mô tả: Đơn nhập hàng gồm có:

+ Chọn nhà cung cấp(nếu nhà cung cấp chưa tồn tại thì thêm mới nhà cung cấp).

+ Chọn sản phẩm cần nhập( nếu sản phẩm ấy chưa tồn tại thì thêm mới sản phẩm ấy).

+ Hiển thị danh sách các sản phẩm đã chọn và số tiền cần thanh toán.

+ Tích xanh nếu đã thanh toán với nhà cung cấp.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Đơn đặt hàng | + POST: api/receipt  + POST: api/supplier  + POST: api/product | + Nhà cung cấp: bắt buộc chọn, chỉ được chọn 1.  + Sản phẩm: bắt buộc chọn, có thể chọn nhiều, số lượng phải lớn hơn 10.  + Hình thức thanh toán phải chọn. |  |

* Logic xử lí bên trong:

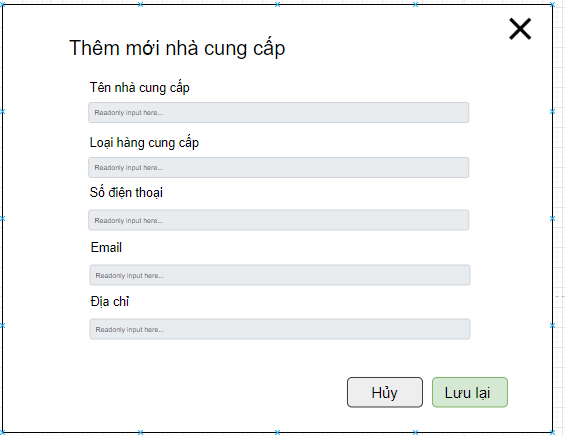
+ Khi tạo đơn nhập hàng, phải chọn nhà cung cấp và các sản phẩm cần nhập về từ nhà cung cấp ấy( 2 trường này là bắt buộc).

+ Sau đấy, hệ thống sẽ lấy thông tin thu được từ lúc người dùng chọn NCC và sản phẩm ở trên để hiện thị thông tin nhà cung cấp và thông tin các sản phẩm được chọn( có thể thay đổi số lượng và giá nhập cho các sản phẩm trong danh sách chọn).

+ Nếu chưa thanh toán thì sẽ không tích xanh vào ô thanh toán. Chọn hình thức thanh toán và số tiền cần trả cho NCC.

+ Lưu đơn nhập vào database.

* Tạo mới nhà cung cấp NCC:
* Màn hình giao diện:



* Mô tả: Thêm mới nhà cung cấp sẽ gồm các thông tin như:

+ Tên nhà cung cấp.

+ Loại hàng cung cấp.

+ Số điện thoại.

+ Email.

+ Địa chỉ nhà cung cấp.

Cập nhật NCC sẽ tương tự như trên, nhưng các ô input phải có sẵn các thông tin cũ để có thể cập nhật dễ dàng.

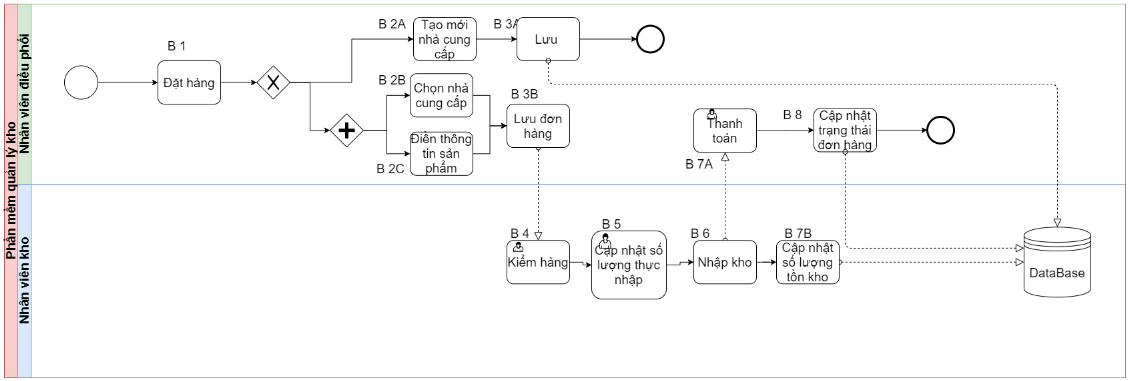
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Thêm mới NCC | POST: /api/supplier | + Tên nhà cung cấp: phải có, không quá 255 kí tự  + Loại hàng cung cấp: tối đa 50 kí tự.  + Số điện thoại: phải có, kiểu dữ liệu number, tối đa 12 kí tự.  + Email: phải có, phải tuân theo regular expression.  + Địa chỉ nhà cung cấp: phải có, giới hạn 255 kí tự |  |

* Logic xử lí: Khi thêm mới NCC, hệ thống sẽ hiển thị form nhập các thông tin cần thiết cho NCC đấy. Sau khi đã nhập đủ và đúng theo yêu cầu thì hệ thống sẽ gửi các thông tin của sản phẩm vừa nhập đến server với method là POST và url là /api/supplier để thêm NCC mới vào database.

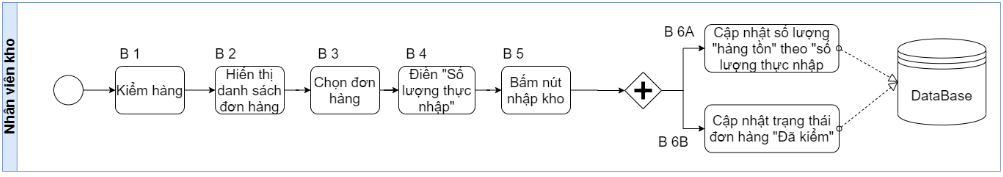
### 2.2 Quy trình 2.

2.2.1 Mã màn hình, tên màn hình.

- Flow:



Hình a



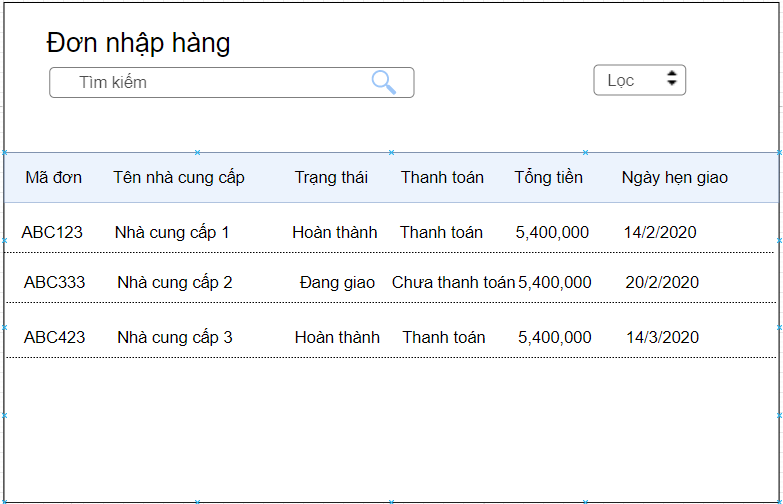
Hình b

- Các bước thực hiện (theo hình a):

|  |  |  |
| --- | --- | --- |
| **Bước** | **Đối tượng thực hiện** | **Mô tả** |
| B 2B | Nhân viên điều phối | Chọn NCC cần đặt hàng. |
| B 2C | Chọn các sản phẩm cần đặt hàng |
| B 3B | Lưu đơn đặt hàng vào DB |
| B 4 | Nhân viên kho | Khi hàng về thì kiểm hàng theo đơn hàng |
| B 5 | Cập nhật số lượng thực nhập sau khi kiểm hàng |
| B 6 | Sau khi kiểm hàng thì tiến hành nhập hàng vào kho |
| B 7B | Cập nhật số lượng trong kho của các mặt hàng vừa về |
| B 7A | Nhân viên điều phối | Thanh toán với NCC |
| B 8 | Sau khi hoàn thành tất cả các bước thì cập nhật trạng thái đơn hàng thành đã hoàn thành. |

2.2.2 Mô tả chi tiết.

* Danh sách đơn đặt hàng:
* Màn hình giao diện.



* Mô tả: Danh sách sẽ hiển thị các thông tin liên quan đến các đơn đặt hàng với các nhà cung cấp đã được tạo gồm có các trường như:

+ Mã đơn.

+ Tên nhà cung cấp.

+ Trạng thái.

+ Thanh toán.

+ Tổng tiền.

+ Ngày hẹn giao.

Và có thể tìm kiếm theo tên NCC hoặc mã đơn, lọc theo tên từ A-Z, Z-A, theo trạng thái và theo thanh toán.

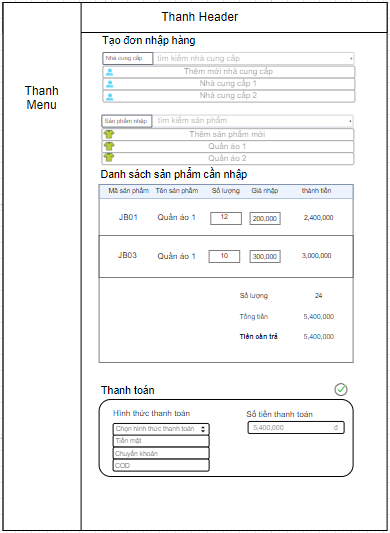
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Đơn nhập hàng | GET: /api/receipt?limit=?&page=? |  |  |

* Logic xử lí:

+ Khi hiển thị danh sách đơn nhập hàng, hệ thống sẽ gọi đến server để lấy dữ liệu từ database về.

+ Khi có dữ liệu thì phần fontend sẽ tiến hàng xử lí và hiển thị dữ liệu ra bảng cho người dùng xem.

* Đơn đặt hàng:
* Màn hình giao diện đơn nhập hàng.



* Mô tả: Đơn nhập hàng gồm có:

+ Chọn nhà cung cấp(nếu nhà cung cấp chưa tồn tại thì thêm mới nhà cung cấp).

+ Chọn sản phẩm cần nhập( nếu sản phẩm ấy chưa tồn tại thì thêm mới sản phẩm ấy).

+ Hiển thị danh sách các sản phẩm đã chọn và số tiền cần thanh toán.

+ Tích xanh nếu đã thanh toán với nhà cung cấp.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Đơn đặt hàng | + POST: api/receipt  + POST: api/supplier  + POST: api/product | + Nhà cung cấp: bắt buộc chọn, chỉ được chọn 1.  + Sản phẩm: bắt buộc chọn, có thể chọn nhiều, số lượng phải lớn hơn 10.  + Hình thức thanh toán phải chọn. |  |

* Logic xử lí bên trong:

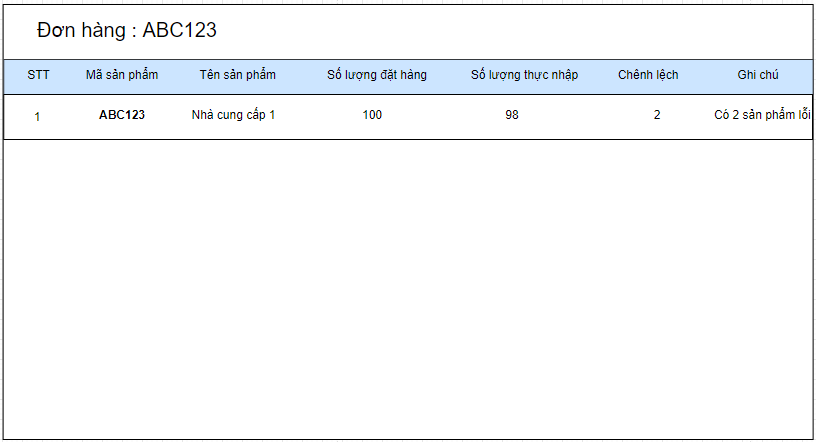
+ Khi tạo đơn nhập hàng, phải chọn nhà cung cấp và các sản phẩm cần nhập về từ nhà cung cấp ấy( 2 trường này là bắt buộc).

+ Sau đấy, hệ thống sẽ lấy thông tin thu được từ lúc người dùng chọn NCC và sản phẩm ở trên để hiện thị thông tin nhà cung cấp và thông tin các sản phẩm được chọn( có thể thay đổi số lượng và giá nhập cho các sản phẩm trong danh sách chọn).

+ Nếu chưa thanh toán thì sẽ không tích xanh vào ô thanh toán. Chọn hình thức thanh toán và số tiền cần trả cho NCC.

+ Lưu đơn nhập vào database.

* Kiểm hàng từ NCC:
* Màn hình giao diện:



* Mô tả: Nhân viên sẽ kiểm hàng lúc sản phẩm đc giao từ NCC về kho, nhân viên sẽ lấy đơn nhập hàng để đối chiếu với sản phẩm nhận đc, nếu sản phẩm thiếu, thừa hay lỗi hoặc 1 số nguyên nhân khác thì sẽ ghi lại và tiến hành xử lí và thanh toán.

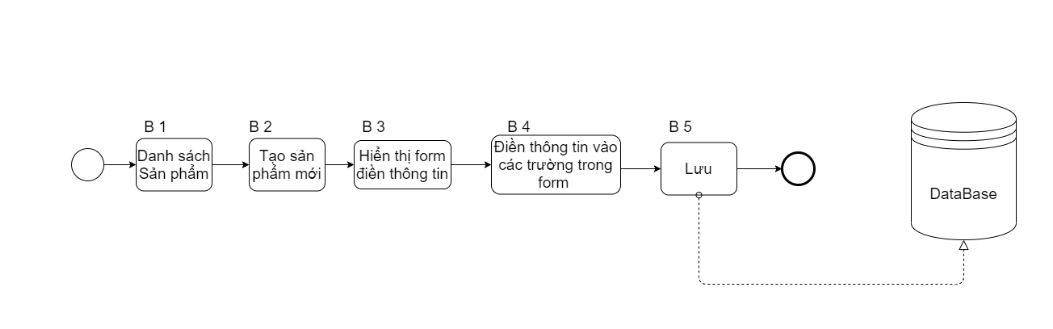
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Kiểm hàng | GET: /api/receipt/id  PUT: /api/receipt/id | Chỉ thay đổi số lượng chênh leenchj và ghi chú |  |

* Logic xử lí: Khi lấy chi tiết đơn hàng từ server về, người dùng có thể thay đổi trường chênh lệch , số lượng thực nhập và ghi lại lí do tại ghi chú( lí do ví dụ như hàng lỗi, sai hàng, thiếu hàng, …) sau đó gửi cập nhật lên server.

### 2.3 Quy trình 3

2.3.1 Mã màn hình, tên màn hình.

- Flow:

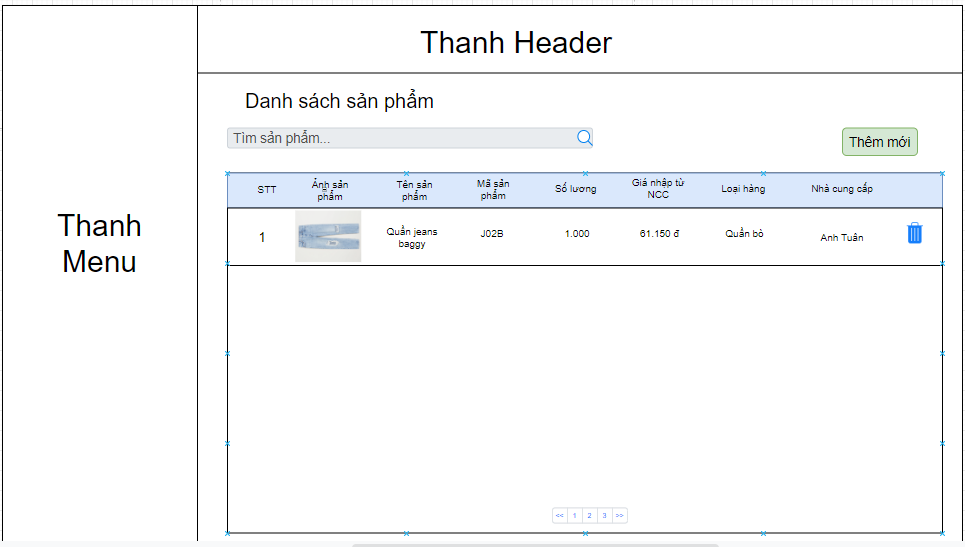


- Các bước thực hiện (theo hình a):

|  |  |  |
| --- | --- | --- |
| **Bước** | **Đối tượng thực hiện** | **Mô tả** |
| B 1 | Nhân viên kho | Hiển thị danh sách sản phẩm |
| B 2 | Tạo sản phẩm mới |
| B 3 | Hiển thị form nhập sản phẩm mới |
| B 4 | Điền các thông tin và các trường để tạo sản phẩm mới( có validate ) |
| B 5 | Lưu sản phẩm mới vào DB |

2.3.2 Mô tả chi tiết.

* Danh sách sản phẩm trong kho:
* Màn hình giao diện:



* Mô tả: Danh sách sản phẩm trong kho sẽ gồm các trường như:

+ STT.

+ Ảnh sản phẩm.

+ Tên sản phẩm.

+ Số lượng trong kho.

+ Giá nhập từ NCC.

+ Loại hàng.

+ Nhà cung cấp sản phẩm đấy.

Và có thể tìm kiếm theo tên. Có nút thêm mới để thêm các sản phẩm mới chưa tồn tại trên hệ thống. Tại mỗi sản phẩm sẽ có thêm nút xóa.

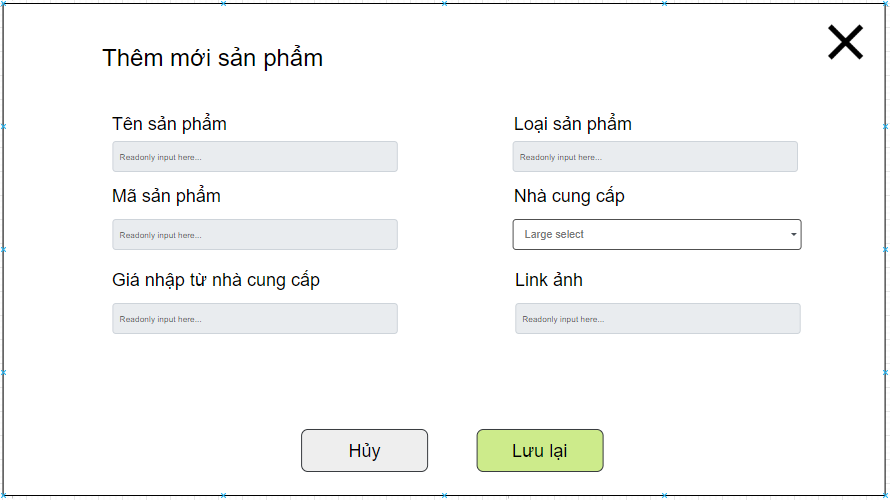
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Danh sách sản phẩm | GET: /api/product?limit=?&page=?  GET:  /api/product? limit=?&page=?&search=?&sort=? |  |  |

* Logic xử lí:

+ Khi hiển thị danh sách sản phẩm, hệ thống sẽ gọi đến server để lấy dữ liệu từ database về.

+ Khi có dữ liệu thì phần fontend sẽ tiến hàng xử lí và hiển thị dữ liệu ra bảng cho người dùng xem.

* Thêm mới sản phẩm:
* Màn hình giao diện.



* Mô tả: Thêm mới sản phẩm sẽ gồm các thông tin như sau:

+ Tên sản phẩm.

+ Mã sản phẩm:

+ Giá nhập vào.

+ Loại sản phẩm.

+ Nhà cung cấp.

+ Link ảnh của sản phẩm.

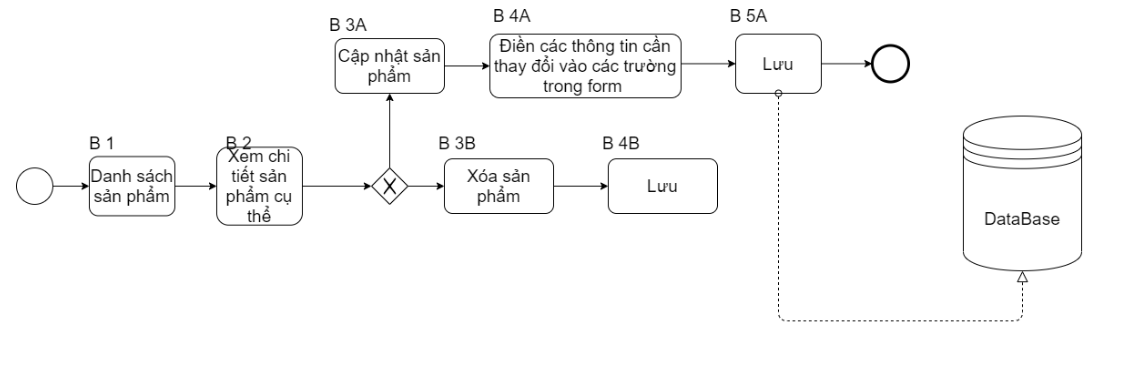
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Thêm mới sản phẩm | POST: /api/product | + Tên sản phẩm: Phải có, tối đa 255 kí tự  + Mã sản phẩm: phải có, tối đa 10 kí tự.  + Giá nhập vào: phải có, number, tối đa 15 kí tự  + Loại sản phẩm: phải có.  + Nhà cung cấp: phải có, chọn trong list các nhà cung cấp đã có.  + Link ảnh của sản phẩm: phải có. |  |

* Logic xử lí: Khi thêm mới sản phẩm, hệ thống sẽ hiển thị form nhập các thông tin cần thiết cho sản phẩm đấy. Sau khi đã nhập đủ và đúng theo yêu cầu thì hệ thống sẽ gửi các thông tin của sản phẩm vừa nhập đến server với method là POST và url là /api/product.

### 2.4 Quy trình 4

2.4.1 Mã màn hình, tên màn hình.

- Flow:

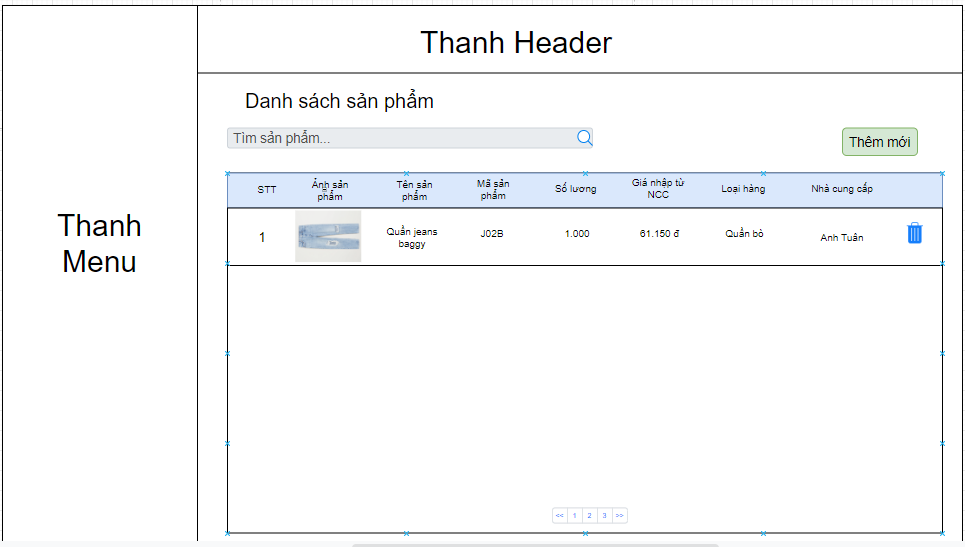


- Các bước thực hiện (theo hình a):

|  |  |  |
| --- | --- | --- |
| **Bước** | **Đối tượng thực hiện** | **Mô tả** |
| B 1 | Nhân viên kho | Hiển thị danh sách sản phẩm |
| B 2 | Xem chi tiết 1 sản phẩm |
| B 3A | Chọn cập nhật sản phẩm đang xem chi tiết |
| B 4A | Điền các thông tin cần thay đổi vào các trường để cập nhật sản phẩm ( có validate ) |
| B 5A | Lưu sản phẩm đã cập nhật vào DB |
| B 3B | Chọn xóa sản phẩm |
| B 4B | Đổi status sản phẩm đã chọn xóa vào DB để ẩn sản phẩm đấy đi |

2.4.2 Mô tả chi tiết.

* Danh sách sản phẩm trong kho:
* Màn hình giao diện:



* Mô tả: Danh sách sản phẩm trong kho sẽ gồm các trường như:

+ STT.

+ Ảnh sản phẩm.

+ Tên sản phẩm.

+ Số lượng trong kho.

+ Giá nhập từ NCC.

+ Loại hàng.

+ Nhà cung cấp sản phẩm đấy.

Và có thể tìm kiếm theo tên. Có nút thêm mới để thêm các sản phẩm mới chưa tồn tại trên hệ thống. Tại mỗi sản phẩm sẽ có thêm nút xóa.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Danh sách sản phẩm | GET: /api/product?limit=?&page=?  GET:  /api/product? limit=?&page=?&search=?&sort=? |  |  |

* Logic xử lí:

+ Khi hiển thị danh sách sản phẩm, hệ thống sẽ gọi đến server để lấy dữ liệu từ database về.

+ Khi có dữ liệu thì phần fontend sẽ tiến hàng xử lí và hiển thị dữ liệu ra bảng cho người dùng xem.

* Chi tiết sản phẩm
* Màn hình giao diện:



* Mô tả: Chi tiết sản phẩm sẽ hiển thị các trường như:

+ Tên sản phẩm.

+ Mã sản phẩm:

+ Giá nhập vào.

+ Loại sản phẩm.

+ Nhà cung cấp.

+ Ảnh của sản phẩm.

+ Số lượng

+ Ngày tạo và ngày sửa.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Chi tiết sản phẩm | GET: /api/product/id |  |  |

* Logic bên trong:

+ Khi hiển thi danh sách sản phẩm, thì người dùng bấm vào sản phẩm sẽ xem được chi tiết sản phẩm đó.

+ Hệ thống sẽ gửi 1 request đến server với method là GET và url là /api/product/id để lấy dữ liệu theo id, và server sẽ gửi dữ liệu chi tiết sản phẩm cần lấy về cho fontend.

+ Sau đấy bên fontend sẽ lấy dữ liệu nhận được và tiến hành hiển thi ra màn hình giao diện cho người dùng.

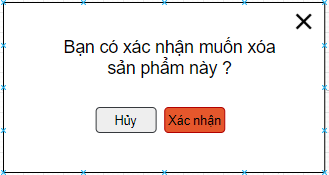
* Cập nhật sản phẩm
* Màn hình giao diện:



* Mô tả: Khi cập nhật sản phẩm thì sẽ có thể sửa đổi các trường của sản phẩm và lưu lại.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Cập nhật sản phẩm | GET: /api/product/id  PUT: /api/product/id | + Tên sản phẩm: Phải có, tối đa 255 kí tự  + Mã sản phẩm: phải có, tối đa 10 kí tự.  + Giá nhập vào: phải có, number, tối đa 15 kí tự  + Loại sản phẩm: phải có.  + Nhà cung cấp: phải có, chọn trong list các nhà cung cấp đã có.  + Link ảnh của sản phẩm: phải có. |  |

* Logic xử lí: Khi xem chi tiết 1 sản phẩm sẽ có 1 nút bấm để có thể cập nhật sản phẩm đó. Trong form cập nhật sẽ điền các thông tin cần thay đổi sau đó bấm xác nhận cập nhật thì hệ thống sẽ gửi 1 request với method là PUT và url là /api/product/id.
* Xóa sản phẩm:
* Màn hình giao diện



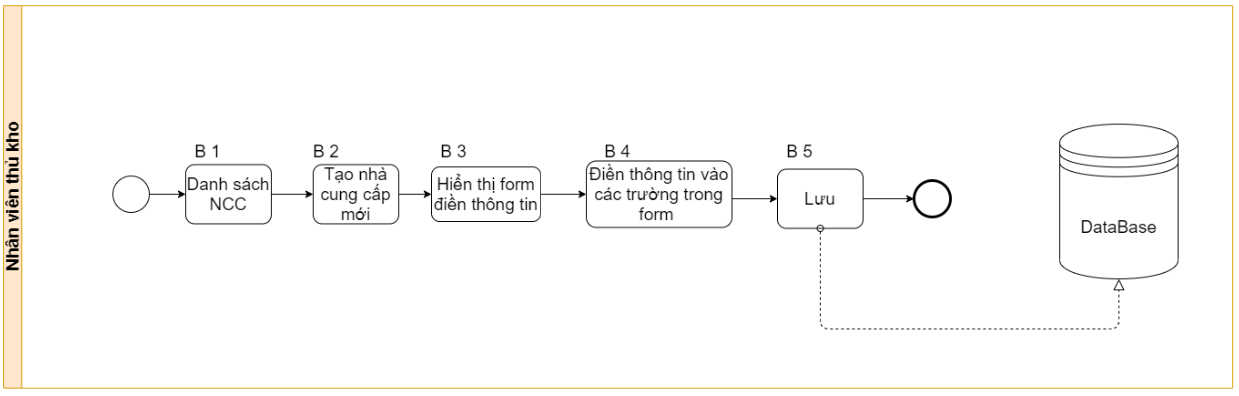
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Xóa NCC | DELETE: /api/product |  |  |

* Logic xử lí: khi người dùng bấm xác nhận thì hệ thống sẽ gửi 1 request với method là DELETE với url /api/product/id đến server để đổi status của sản phẩm tương ứng và ẩn nó đi khỏi giao diện.

### 2.5 Quy trình 5

2.5.1 Mã màn hình, tên màn hình.

- Flow:

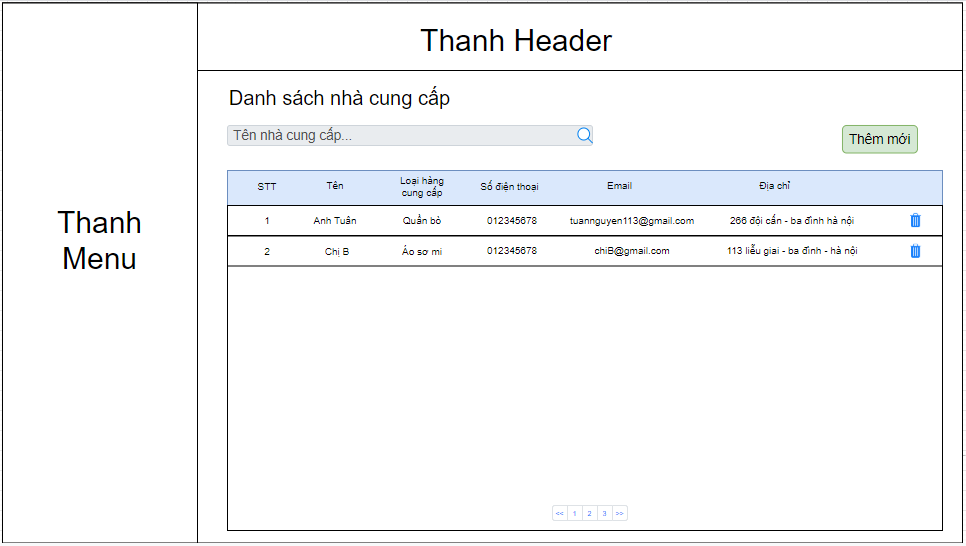


- Các bước thực hiện

|  |  |  |
| --- | --- | --- |
| **Bước** | **Đối tượng thực hiện** | **Mô tả** |
| B 1 | Quản lí kho | Hiển thị danh sách NCC |
| B 2 | Tạo NCC mới |
| B 3 | Hiển thị form nhập NCC mới |
| B 4 | Điền các thông tin và các trường để tạo NCC mới( có validate ) |
| B 5 | Lưu NCC mới vào DB |

2.5.2 Mô tả chi tiết.

* Danh sách NCC:
* Màn hình giao diện:



* Mô tả: Danh sách NCC sẽ hiển thị 1 danh sách các nhà cung cấp sản phẩm đang hợp tác với kho. Danh sách sẽ gồm các trường như:

+ STT

+ Tên.

+ Loại hàng cung cấp.

+ Số điện thoại.

+ Email.

+ Địa chỉ.

Và có thể tìm kiếm theo tên, số điện thoại, email. Sẽ có nút thêm mới để thêm các nhà cung cấp chưa tồn tại trên hệ thống. Tại mỗi NCC sẽ có thêm nút xóa.

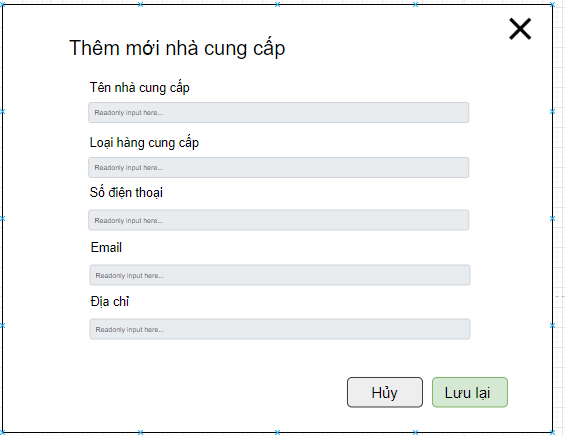
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Danh sách NCC | GET: /api/supplier/?limit=?&page=?  GET: /api/supplier/? limit=?&page=?&search=?&sort=? |  |  |

* Logic xử lí:

+ Khi hiển thị danh sách đơn nhập hàng, hệ thống sẽ gọi đến server để lấy dữ liệu từ database về.

+ Khi có dữ liệu thì phần fontend sẽ tiến hàng xử lí và hiển thị dữ liệu ra bảng cho người dùng xem.

* Tạo mới nhà cung cấp NCC:
* Màn hình giao diện:



* Mô tả: Thêm mới nhà cung cấp sẽ gồm các thông tin như:

+ Tên nhà cung cấp.

+ Loại hàng cung cấp.

+ Số điện thoại.

+ Email.

+ Địa chỉ nhà cung cấp.

Cập nhật NCC sẽ tương tự như trên, nhưng các ô input phải có sẵn các thông tin cũ để có thể cập nhật dễ dàng.

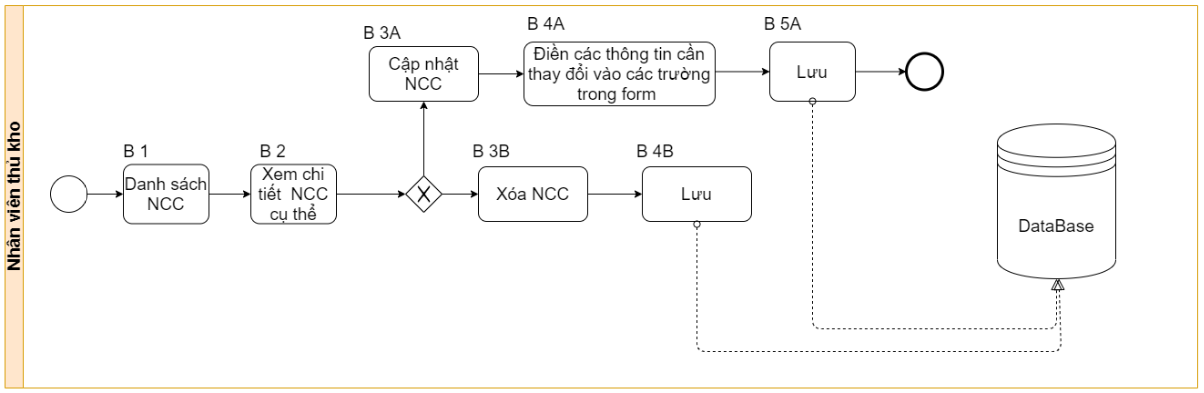
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Thêm mới NCC | POST: /api/supplier | + Tên nhà cung cấp: phải có, không quá 255 kí tự  + Loại hàng cung cấp: tối đa 50 kí tự.  + Số điện thoại: phải có, kiểu dữ liệu number, tối đa 12 kí tự.  + Email: phải có, phải tuân theo regular expression.  + Địa chỉ nhà cung cấp: phải có, giới hạn 255 kí tự |  |

* Logic xử lí: Khi thêm mới NCC, hệ thống sẽ hiển thị form nhập các thông tin cần thiết cho NCC đấy. Sau khi đã nhập đủ và đúng theo yêu cầu thì hệ thống sẽ gửi các thông tin của sản phẩm vừa nhập đến server với method là POST và url là /api/supplier để thêm NCC mới vào database.

### 2.6 Quy trình 6

2.6.1 Mã màn hình, tên màn hình.

- Flow:

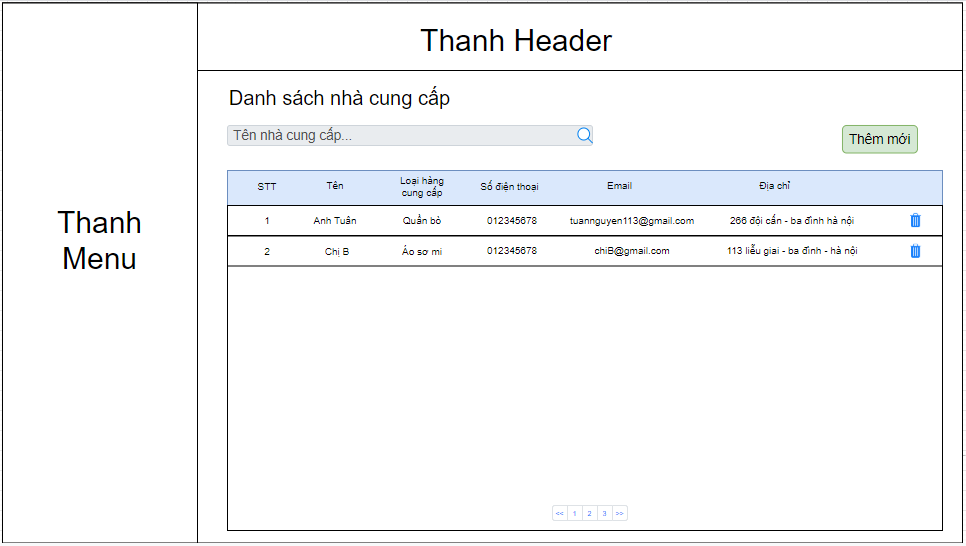


- Các bước thực hiện

|  |  |  |
| --- | --- | --- |
| **Bước** | **Đối tượng thực hiện** | **Mô tả** |
| B 1 | Thủ kho | Hiển thị danh sách NCC |
| B 2 | Xem chi tiết 1 NCC |
| B 3A | Chọn cập nhật NCC đang xem chi tiết |
| B 4A | Điền các thông tin cần thay đổi vào các trường để cập nhật NCC ( có validate ) |
| B 5A | Lưu NCC đã cập nhật vào DB |
| B 3B | Chọn xóa NCC |
| B 4B | Đổi status NCC đã chọn xóa vào DB để ẩn NCC đấy đi |

2.6.2 Mô tả chi tiết.

* Danh sách NCC:
* Màn hình giao diện:



* Mô tả: Danh sách NCC sẽ hiển thị 1 danh sách các nhà cung cấp sản phẩm đang hợp tác với kho. Danh sách sẽ gồm các trường như:

+ STT

+ Tên.

+ Loại hàng cung cấp.

+ Số điện thoại.

+ Email.

+ Địa chỉ.

Và có thể tìm kiếm theo tên, số điện thoại, email. Sẽ có nút thêm mới để thêm các nhà cung cấp chưa tồn tại trên hệ thống. Tại mỗi NCC sẽ có thêm nút xóa.

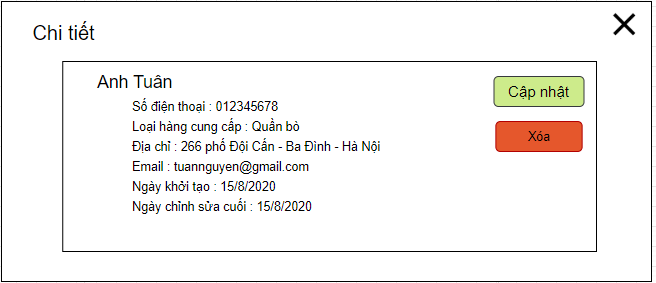
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Danh sách NCC | GET: /api/supplier/?limit=?&page=?  GET: /api/supplier/? limit=?&page=?&search=?&sort=? |  |  |

* Logic xử lí:

+ Khi hiển thị danh sách đơn nhập hàng, hệ thống sẽ gọi đến server để lấy dữ liệu từ database về.

+ Khi có dữ liệu thì phần fontend sẽ tiến hàng xử lí và hiển thị dữ liệu ra bảng cho người dùng xem.

* Chi tiết NCC
* Màn hình giao diện.



* Mô tả: Chi tiết NCC sẽ hiển thị:

+ Tên NCC.

+ Số điện thoại.

+ Loại hàng cung cấp.

+ Địa chỉ NCC.

+ Email.

+ Ngày khởi tạo và ngày sửa.

Và có nút cập nhật NCC để sửa các thông tin nếu cần thiết.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Chi tiết | GET: /api/supplier/id |  |  |

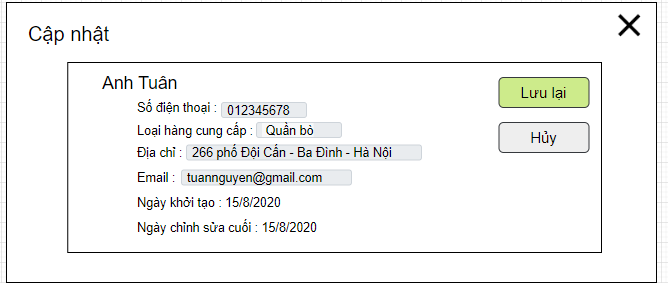
Logic bên trong:

+ Khi hiển thi danh sách NCC, thì người dùng bấm vào NCC sẽ xem được chi tiết NCC đó.

+ Hệ thống sẽ gửi 1 request đến server với method là GET và url là /api/supplier/id để lấy dữ liệu theo id, và server sẽ gửi dữ liệu chi tiết NCC cần lấy về cho fontend.

+ Sau đấy bên fontend sẽ lấy dữ liệu nhận được và tiến hành hiển thi ra màn hình giao diện cho người dùng.

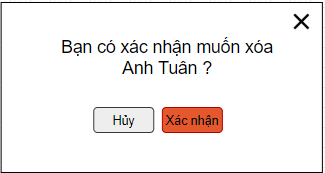
* Cập nhật nhà cung cấp:
* Màn hình giao diện:



* Mô tả: Khi cập nhật thì có thể thay đổi số điện thoại, loại hàng, địa chỉ và email.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Cập nhật NCC | GET: /api/supplier/id  POST: /api/supplier/id | + Tên nhà cung cấp: phải có, không quá 255 kí tự  + Số điện thoại: phải có, kiểu dữ liệu number, tối đa 12 kí tự.  + Email: phải có, phải tuân theo regular expression.  + Địa chỉ nhà cung cấp: phải có, giới hạn 255 kí tự |  |

* Logic xử lí: Khi xem chi tiết 1 NCC sẽ có 1 nút bấm để có thể cập nhật NCC đó. Trong form cập nhật sẽ điền các thông tin cần thay đổi sau đó bấm xác nhận cập nhật thì hệ thống sẽ gửi 1 request với method là PUT và url là /api/supplier /id và server sẽ lưu những cập nhật mới vào database.
* Xóa NCC
* Màn hình giao diện:



* Mô tả: Sẽ xóa NCC khi không còn hợp tác hoặc dư thừa.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên giao diện** | **API liên quan** | **Ràng buộc** | **Mô tả** |
| Xóa NCC | DELETE: /api/supplier |  |  |

- Logic xử lí: Khi xóa 1 NCC trong danh sách NCC, thì hệ thống sẽ gửi đến server 1 request với method là DELETE và url là /api/supplier để đổi status NCC ấy và không còn hiển thị NCC ấy ra giao diện nữa.

# **Tài liệu liên quan**

Link figma, link jira, doc API,...

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Mã tài liệu** | **Tên tài liệu** | **Link** |
| 1 |  | Sapo | <https://www.sapo.vn/phan-mem-quan-ly-ban-hang.html?campaign=search_cc_thuonghieupos_vn&utm_campaign=nhu_cau_pos_180320_cc_search&utm_source=coccoc.com&utm_medium=cpc&utm_content=thuong_hieu&utm_term=sapo> |
|  |  |  |  |

# **Lịch sử chỉnh sửa**

|  |  |  |
| --- | --- | --- |
| **Thời gian** | **Người chỉnh sửa** | **Nội dung chỉnh sửa** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |