ỨNG DỤNG THƯƠNG MẠI ĐIỆN TỬ

MUA BẢO HIỂM

Tính đến gày 12/7/2021

MỤC LỤC

1. Tổng quan về hệ thống 4
2. Mục tiêu của hệ thống: 4

2. Các định nghĩa: 4

2.1: Giới thiệu 4

2.2: Các định nghĩa 5

1. Thiết kế hệ thống 5
2. Các services 6
   1. User service 6
   2. Third party service 6
   3. Order service 6
   4. Product service 7
   5. Approval flow service 7
   6. Job management service 7
   7. Travel insurance service 7
   8. Authorization server 7
3. Các luồng hoạt động chính 7
   1. Đăng ký– đăng nhập 7
      1. Đăng ký 7
      2. Đăng nhập 8
   2. Tạo – hủy đơn hàng 8
      1. Tạo đơn hàng 8
      2. Hủy đơn hàng 8
   3. Tạo – cập nhật luồng phê duyệt 9
      1. Tạo luồng phê duyệt 9
      2. Cập nhật luồng phê duyệt 9
   4. Tạo – Cập nhật - Xóa sản phẩm 9
      1. Tạo sản phẩm 9
      2. Cập nhật sản phẩm 9
      3. Xóa sản phẩm 9
   5. Phê duyệt sản phẩm 9
      1. Phê duyệt khi tạo sản phẩm đã tạo 9
      2. Phê duyệt khi chỉnh sửa thông tin sản phẩm 10
      3. Phê duyệt khi xóa sản phẩm 10

NỘI DUNG

1. **Tổng quan về hệ thống:**
2. **Mục tiêu hệ thống:**

* Hệ thống sẽ được triển khai trên nền tảng website, cho phép người dùng truy cập từ các loại máy tính cũng như điện thoại thông minh. Những người sử dụng hệ thống bao gồm: admin hệ thống, khách hàng, admin của nhà cung cấp, user của nhà cung cấp. Họ có thể đăng nhập vào hệ thống 1 cách đồng thời. Sau khi đăng nhập vào hệ thống, họ có thể thực hiện các thao tác trong phạm vi cho phép phù hợp.
* Khách hàng có thể xem thông tin các hãng bảo hiểm cũng như các loại bảo hiểm khác nhau của hãng đó, cùng với đó là thực hiện mua bảo hiểm trực tuyến và những chức năng cơ bản khác.
* Admin hệ thống có thể thực hiện những thao tác cơ bản như xem thống kê số liệu, quản lý sản phẩm (bảo hiểm), quản lý khách hàng,...
* Admin của bên cung cấp bảo hiểm có thể thực hiện quản lý user của bên cung cấp bảo hiểm
* User của bên cung cấp bảo hiểm có hể phê duyệt sản phẩm, tạo sản phẩm tương ứng với nhà cung cấp, cập nhật luồng phê duyệt sản phẩm

**2. Các định nghĩa:**

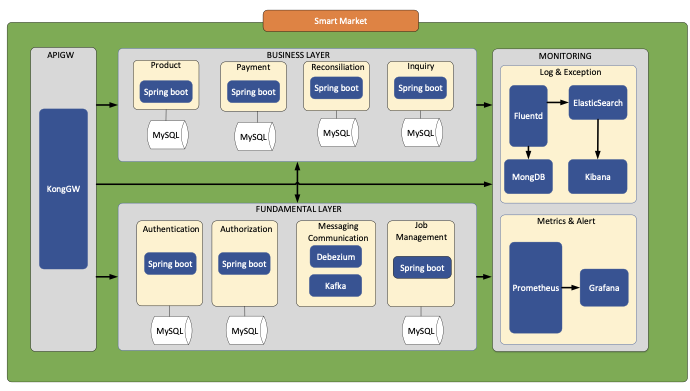
**2.1: Giới thiệu:**

- Tài liệu này được sử dụng để xác định thuật ngữ cụ thể cho lĩnh vực vấn đề, giải thích các thuật ngữ có thể không quen thuộc với người đọc mô tả ca sử dụng hoặc các tài liệu dự án khác. Thông thường, tài liệu này có thể được sử dụng như một từ điển dữ liệu không chính thức, nắm bắt các định nghĩa dữ liệu để mô tả use-case và các tài liệu dự án khác có thể tập trung vào những gì hệ thống phải làm với thông tin.

**2.2: Định nghĩa:**

* Bảng thuật ngữ chứa các định nghĩa làm việc cho các khái niệm chính trong hệ thống mua bán bảo hiểm.
  + 1. **User**: là những khách hàng mà đã đăng ký tài khoản trên hệ thống.
    2. **Guest**: là những khách hàng mà chưa đăng ký tài khoản trên hệ thống.
    3. **Admin hệ thống (admin)**: là người làm chủ có quyền kiểm soát mọi hoạt động của hệ thống. Người này được cấp username và password để đăng nhập hệ thống thực hiện chức năng của mình.
    4. **Bên cung cấp sản phẩm (product provider)**: là bên cung cấp API cho hệ thống để có thể thực hiện mua bảo hiểm của họ. Ví dụ: BIC, ...
    5. **Admin bên cung cấp bảo hiểm (admin provider)**: là người quản trị các tài khoản của bên cung cấp bảo hiểm. Mỗi bên cung cấp bảo hiểm được admin hệ thống tạo ra 1 account admin tương ứng.
    6. **User bên cung cấp bảo hiểm (provider i với i = 1,2,3,4,5)**: là những user được admin bên cung cấp bảo hiểm tạo ra để thực hiện những việc như tạo sản phẩm, phê duyệt sản phẩm, chỉnh sửa luồng phê duyệt sản phẩm. Hiện tại, sẽ có tối đa 5 user của bên cung cấp bảo hiểm, 1 user sẽ có 1 role tương ứng và chịu sự quản lý của admin bên cung cấp bảo hiểm.
    7. **Third party:** là bên thứ 3 đáng tin cậy mà user đã có tài khoản, cho phép dùng tài khoản bên thứ 3 đó để đăng nhập vào hệ thống. Ví dụ: google, facebook,...

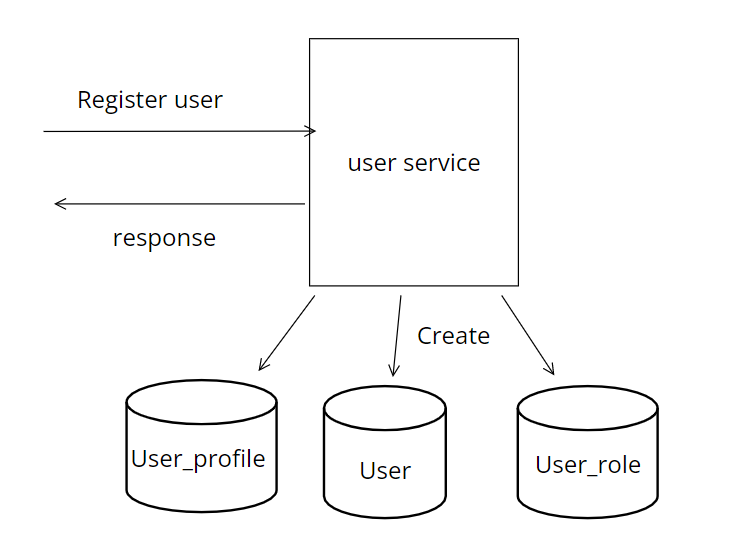
1. **Thiết kế hệ thống:**

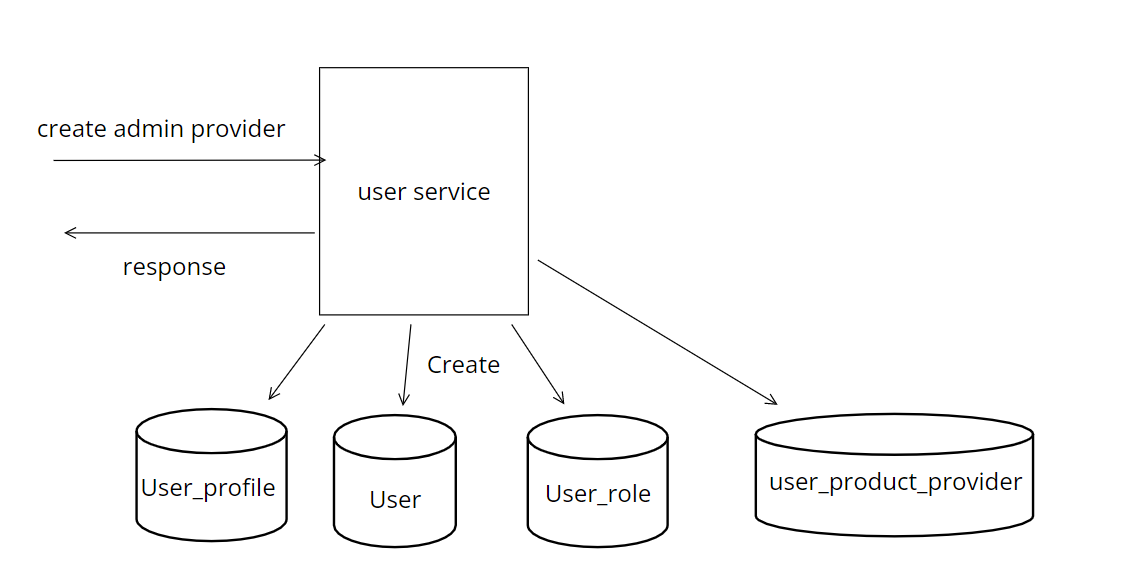
****

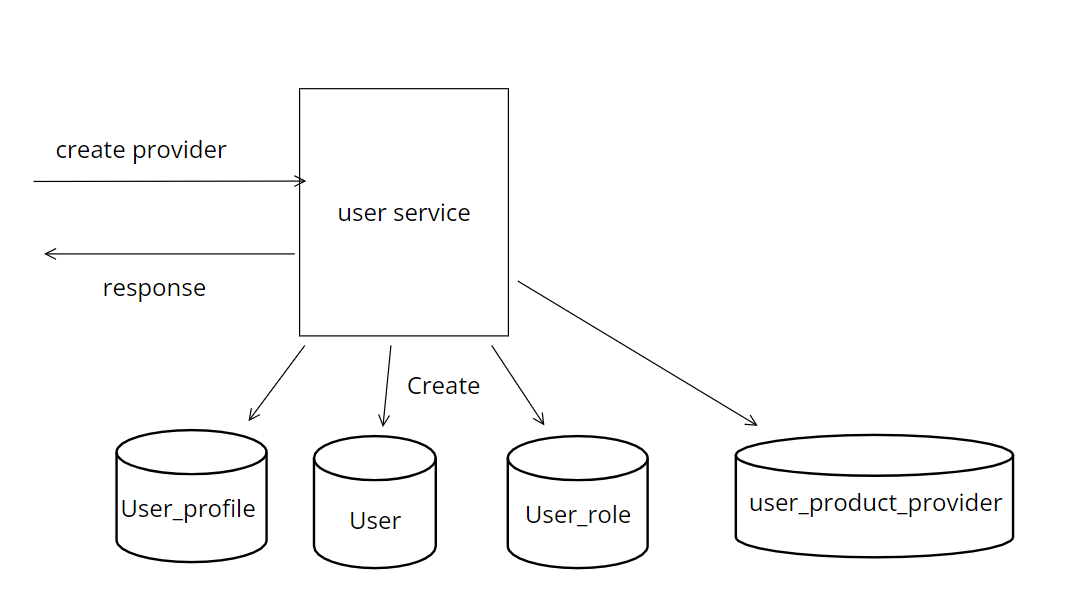
1. **Các services:**
   1. **User service:**

* User service là service thực hiện những nhiệm vụ như thêm, sửa, xóa, truy vấn user và role.
* Hiện tại, các service order, product, approval flow và authorizaition server đều đồng bộ kafka các bảng user, user\_role từ user service về, ngoài ra các product, approval flow service còn đồng bộ thêm bảng user\_product\_provider.
  1. **Third party service:**
* Third party service là service thực hiện nhiệm vụ đăng nhập bằng bên thứ 3 như google, facebook.
  1. **Order service:**
* Order service là service thực hiện những nhiệm vụ như thêm, sửa, xóa, truy vấn đơn hàng.
  1. **Product service:**
* Product service là service thực hiện những nhiệm vụ như thêm, sửa, xóa, truy vấn product và product provider (bên cung cấp sản phẩm).
* Hiệnt tại, order service sẽ đồng bộ bảng product và product provider, user service + approval flow service sẽ đồng bộ bảng product provider từ product service
  1. **Approval flow service:**
* Approval flow service là service thực hiện những nhiệm vụ như thêm, sửa, xóa, truy vấn luồng phê duyệt sản phẩm.
* Hiện tại, product service sẽ đồng bộ bảng approval flow về
  1. **Travel insurance service:**
* Travel insurance service là service thực hiện những nhiệm vụ như tạo sản phẩm, hủy sản phẩm, truy vấn sản phẩm bằng cách gọi api tương ứng sang bên nhà cung cấp sản phẩm.
* Hiện tại job service sẽ đồng bộ bảng pending bic transaction về.
  1. **Job management service:**
* Job management service là service thực hiện những nhiệm vụ như định kỳ 1 khoảng thời gian yêu cầu Travel insurance service truy vấn lại những sản phẩm bị pending khi gọi api tạo hoặc hủy sang bên nhà cung cấp.
  1. **Authorization server:**
* Authorization server thực hiện nhiệm vụ cấp token cho user sau khi đăng nhập thành công.

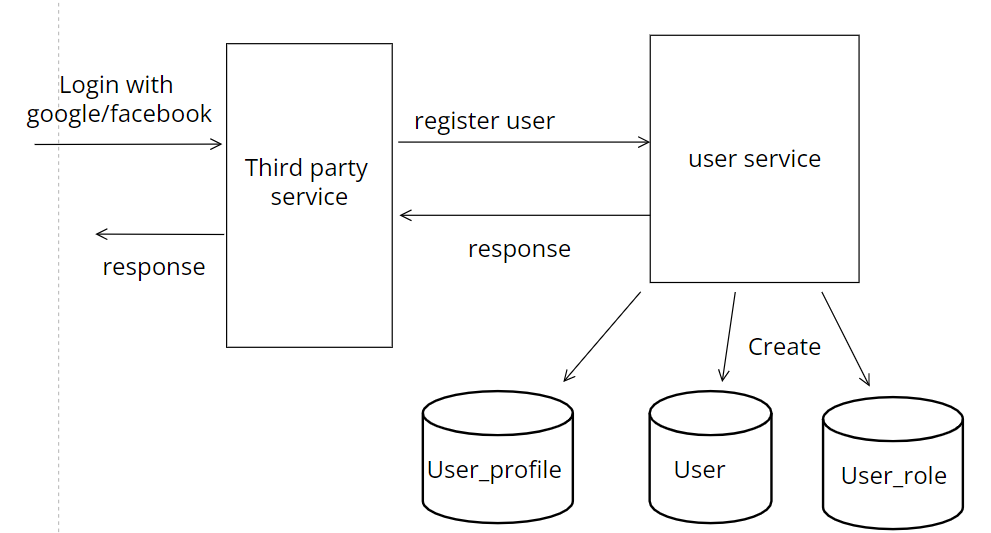
1. **Các luồng hoạt động chính:**
   1. **Đăng ký– Đăng nhập:**
      1. **Đăng ký:**

****

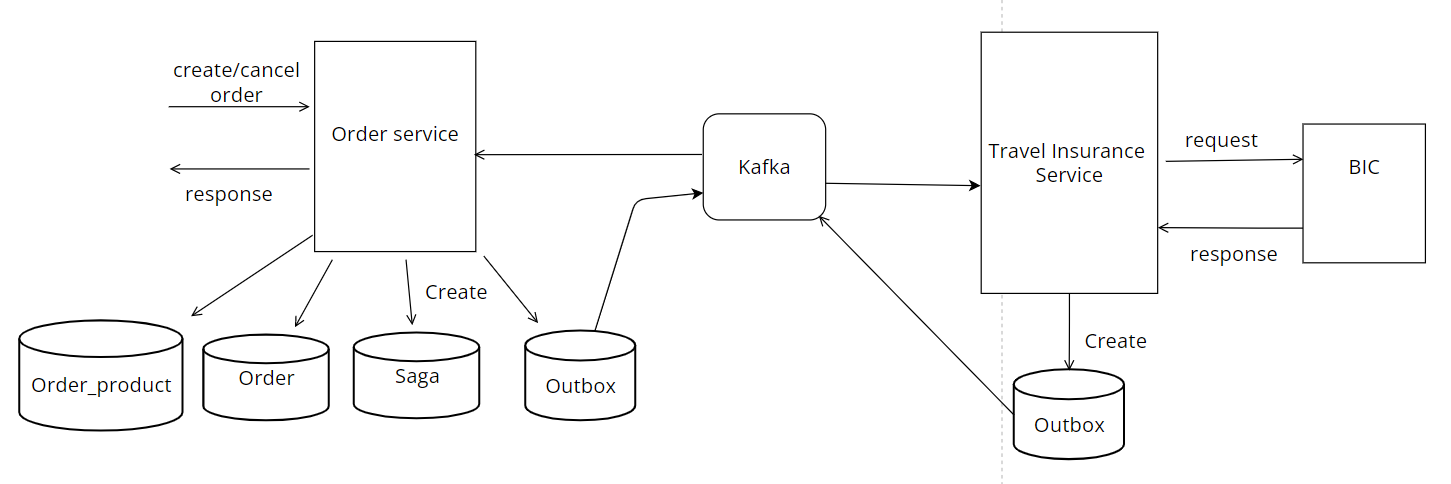
****

****

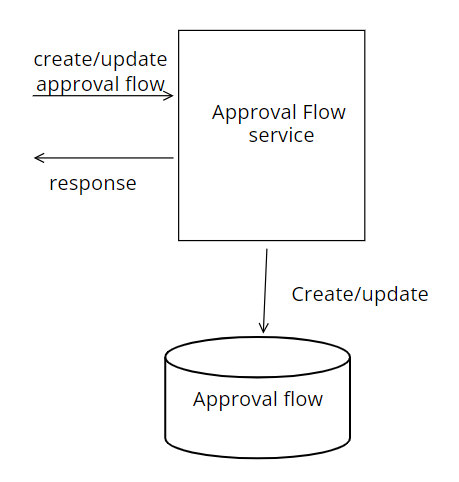
* Khi user muốn đăng ký thì sẽ gọi vào api register-user (1 user không thuộc provider nào thì trường email là unique)
* Khi admin muốn đăng ký tài khoản cho admin provider (sau khi đã tạo thông tin product provder ở product servce) thì sẽ gọi vào api create-provider-admin-user (1 admin provider ứng với 1 product provider)
* Khi admin provider muốn tạo provider thì sẽ gọi vào api create-provider-user (tối đa 5 provider ứng với 1 product provider)
* Những tài khoản mà thuộc 1 product provider sẽ được tạo thêm 1 row trong bảng user\_product\_provider gồm username + product\_provider\_id.
  + 1. **Đăng nhập: (hiện tại chưa đăng nhập bằng google được trên fis)**

****

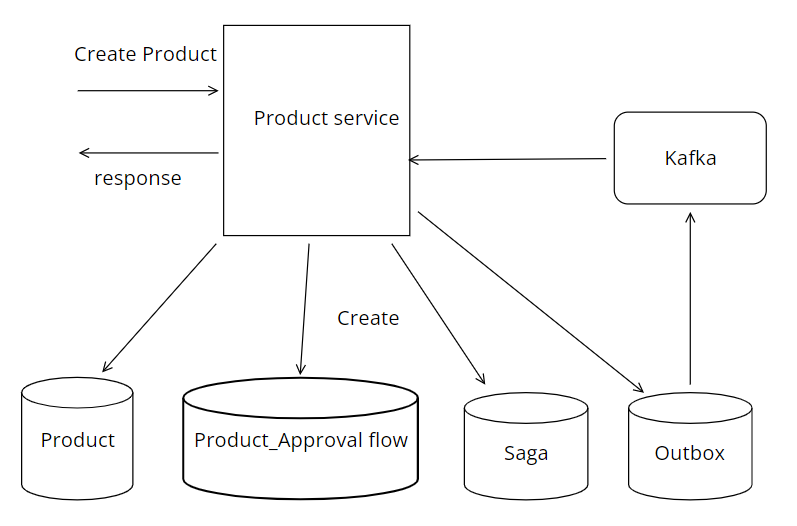
* Khi user muốn đăng nhập thì có thể đăng nhập bằng username + password
* Nếu user chọn đăng nhập bằng third party thì khi đăng nhập lần đầu với third party, third party service sẽ lấy 1 số thông tin cơ bản của người dùng và tạo ra 1 password sẵn rồi gọi vào api register user của user service để tạo 1 tài khoản mới.
* Admin, admin provider, provider-i chỉ được đăng nhập bằng username, password thông thường chứ không được dùng tài khoản của third party.
  1. **Tạo – Hủy đơn hàng:**

****

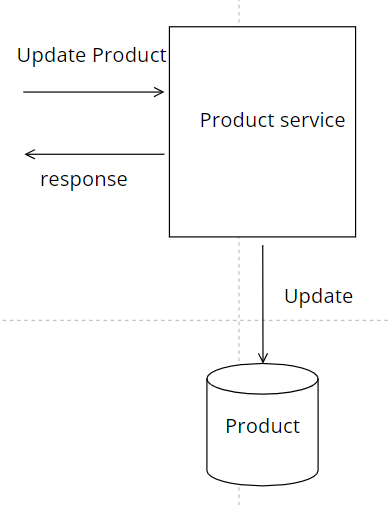
* + 1. **Tạo đơn hàng:**
* Khi gọi api create-order, order service sẽ tạo 1 bảng saga lưu thông tin giao dịch đó và đồng thời tạo tuần tự 1 outbox 1 lần đến các service của sản phẩm tương ứng
  + Ví dụ: travel insurance service cho item là bảo hiểm du lịch của BIC.
* Outbox liền trước trả về thành công thì mới tiếp tục tạo outbox sản phẩm liền sau.
* Service ứng với mỗi item khi lắng nghe outbox từ kafka sẽ lấy những thông tin cần thiết trong outbox và thực hiện gọi api sang bên nhà cung cấp sản phẩm để tạo sản phẩm bên đó, kết quả sẽ được thêm vào outbox trả lại cho order service để xử lý
* Nếu item đầu tiên trong order tạo thất bại thì hủy tạo cả order, trạng thái order là aborted.
* Nếu có 1 item ở giữa thất bại thì thực hiện aborting các item đã tạo thành công ở trước. Khi aborting thì cũng lần lượt tạo 1 outbox 1 lần cho lần lượt các item ở trước item thất bại (bắt đầu từ item liền trước item tất bại).
* Nếu aborting 1 item thành công thì mới aborting tiếp item khác, nếu không thành công thì trạng thái order là error.
* Nếu tất cả item tạo thành công thì coi như order được tạo thành công, trạng thái là succeeded.
* Bất kỳ item nào trong order, sau khi order service nhận được phản hồi sẽ đều tạo 1 row trong bảng order\_product và cập nhật trạng thái tương ứng với item đó ví dụ: Succeeded, Cancled, Aborted,...
  + 1. **Hủy đơn hàng:**
* Khi truy vấn ra đơn hàng cần hủy, hệ thống sẽ thực hiện lock đơn hàng đó trong DB để nếu có 1 request cũng là hủy đơn hàng đó đến gần như đồng thời thì phải đợi đến khi request đầu tiên commit transaction mới truy vấn ra order được (tránh trường hợp request 1 đang chuyển trạng thái order từ succeeded 🡪 cancling chưa xong mà request 2 lại truy vấn ra thì vẫn là succeeded thì sẽ lại tạo 1 luồng hủy nữa).
* Khi gọi api cancel-order, order service sẽ kiểm tra trạng thái order , nếu order đã được tạo thành công (Succeeded) thì mới tiếp tục thực hiện tiếp.
* Nếu order đã được tạo thành công, tạo 1 bảng saga lưu thông tin giao dịch đó và đồng thời tạo tuần tự tuần tự 1 outbox 1 lần đến các service của sản phẩm tương ứng
  + Ví dụ: travel insurance service cho item là bảo hiểm du lịch của BIC.
* Service ứng với mỗi item khi lắng nghe outbox từ kafka sẽ lấy những thông tin cần thiết trong outbox và thực hiện gọi api sang bên nhà cung cấp sản phẩm để truy vấn thông tin sản phẩm bên đó, sau đó lấy kết quả truy vấn để tạo request body cho api hủy sản phẩm và gọi api hủy sang bên nhà cung cấp. Kết quả sẽ được thêm vào outbox trả lại cho order service để xử lý
* Nếu cancel thành công hết thì trạng thái order là cancled.
* Nếu order chỉ cần có 1 item cancel thất bại thì trạng thái order là error.
* Nếu có item khi cancel thất bại thì order service vẫn tiếp tục tạo outbox để cancel các item tiếp theo. Item nào cancel thất bại thì sẽ lưu trạng thái item đó là error, item nào thành công thì trạng thái là canceled.
* Bất kỳ item nào trong order, sau khi order service nhận được phản hồi sẽ đều tạo 1 row trong bảng order\_product và cập nhật trạng thái tương ứng với item đó ví dụ: Succeeded, Cancled, Aborted,...
  1. **Tạo - Cập nhật luồng phê duyệt:**

****

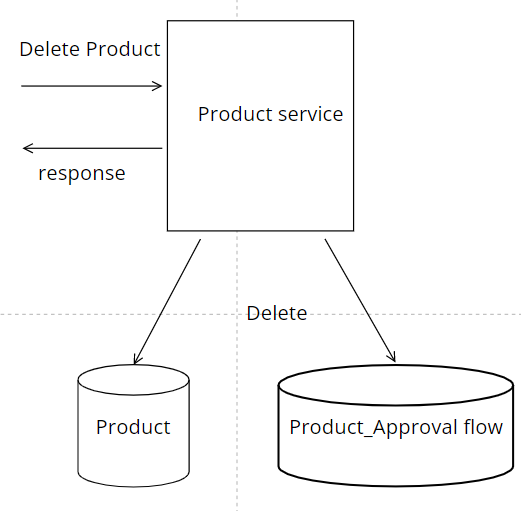
* + 1. **Tạo luồng phê duyệt:**
* Admin sẽ tạo luồng phê duyệt mặc định (tức là không cần phải phê duyệt) cho mỗi product provider trước khi admin provider tạo sản phẩm đó.
  + 1. **Cập nhật luồng phê duyệt:**
* Admin provider cập nhật lại luồng phê duyệt mặc định, luồng phê duyệt sẽ có tối đa 5 step, mỗi step ứng với 1 provider i (i =1 🡪5) (không cần phải theo thứ tự i tăng dần với mỗi step)
  1. **Tạo– Cập nhật – Xóa sản phẩm:**
     1. **Tạo sản phẩm:**

****

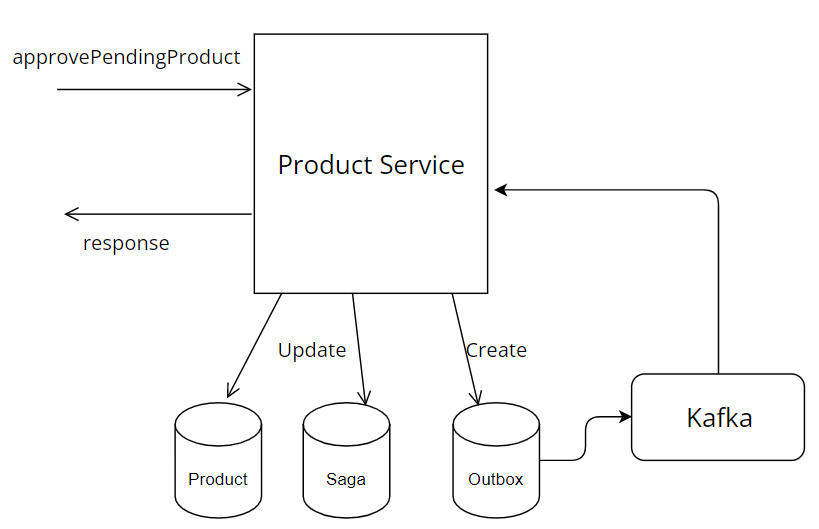
* Nếu muốn tạo sản phẩm thì 1 provider-i của product provider tương ứng sẽ phải gọi api tạo sản phẩm (sau khi admin đã tạo luồng phê duyệt cho sản phẩm đó).
* Khi tạo product mà luồng phê duyệt tạo vẫn là mặc định thì product sẽ có trạng thái là succeeded luôn, nếu không thì trạng thái là pending.
* Khi tạo product thì product service tạo ra thêm 1 row trong bảng product\_approval\_flow để lưu lại approval\_flow của product đó, tạo ra 1 row trong bảng saga để lưu lại trạng thái giao dịch phê duyệt từ lúc product được tạo đến khi phê duyệt tạo kết thúc và tạo 1 outbox type là watiting approve.
  + 1. **Cập nhật sản phẩm:**

****

* Nếu muốn cập nhật sản phẩm thì 1 provider của product provider tương ứng sẽ phải gọi api cập nhật sản phẩm (sau khi admin đã tạo luồng phê duyệt cho sản phẩm đó)
* Sau khi truy vấn được product, nếu product có trạng thái là approved thì mới được thực hiện cập nhật product.
  + 1. **Xóa sản phẩm:**

****

* Nếu muốn xóa sản phẩm thì 1 provider của product provider tương ứng sẽ phải gọi api xóa sản phẩm (sau khi admin đã tạo luồng phê duyệt cho sản phẩm đó).
* Sau khi truy vấn được product, nếu product có trạng thái là approved thì mới được thực hiện cập nhật product.
  1. **Phê duyệt sản phẩm:**
     1. **Truy vấn các product đang chờ user phê duyệt:**
* Provider-i gọi api getListPendingProduct để truy vấn các product đang chờ user đó phê duyệt.
  + 1. **Phê duyệt khi tạo sản phẩm đã tạo:**

****

* Sản phẩm đã tạo mà chưa được phê duyệt luồng tạo sẽ có trạng thái là pending.
* Nếu như sản phẩm đã được phê duyệt xong hết sẽ có trạng thái là approved.
* Khi thực hiện api phê duyệt product thì product service sẽ thực hiện kiểm tra xem role của user hiện tại có đúng với step phê duyệt hiện tại không, nếu có mới thực hiện tiếp.
* Sau khi kiểm tra role và pass thì product service tạo 1 outbox type là appprove created product và tự lắng nghe nó và xử lý cập nhật lại trạng thái product và cập nhật lại saga.
* Nếu xử lý outbox thành công và product được approve ở bước hiện tại, product service sẽ tạo tiếp 1 outbox waiting approve (nếu vẫn còn step sau) và cập nhật lại saga
* Nếu bị exception hoặc bị product bị disapprove thì kết thúc luồng phê duyệt và cập nhật lại trạng thái product + saga là disapproved hoặc error (nếu exception).
  + 1. **Phê duyệt khi chỉnh sửa thông tin sản phẩm: (hiện tại chưa làm)**
* Sản phẩm đã tạo, khi có yêu cầu cập nhật lại thông tin thì phải chờ hoàn thành hết các bước phê duyệt mới được cập nhật lại vào db. Trạng thái sau khi phê duyệt xong là updated.
  + 1. **Phê duyệt khi xóa sản phẩm: (hiện tại chưa làm)**
* Sản phẩm đã tạo, khi có yêu cầu xóa sản phẩm thì thì phải chờ hoàn thành hết các bước phê duyệt mới được xóa sản phẩm trong db.