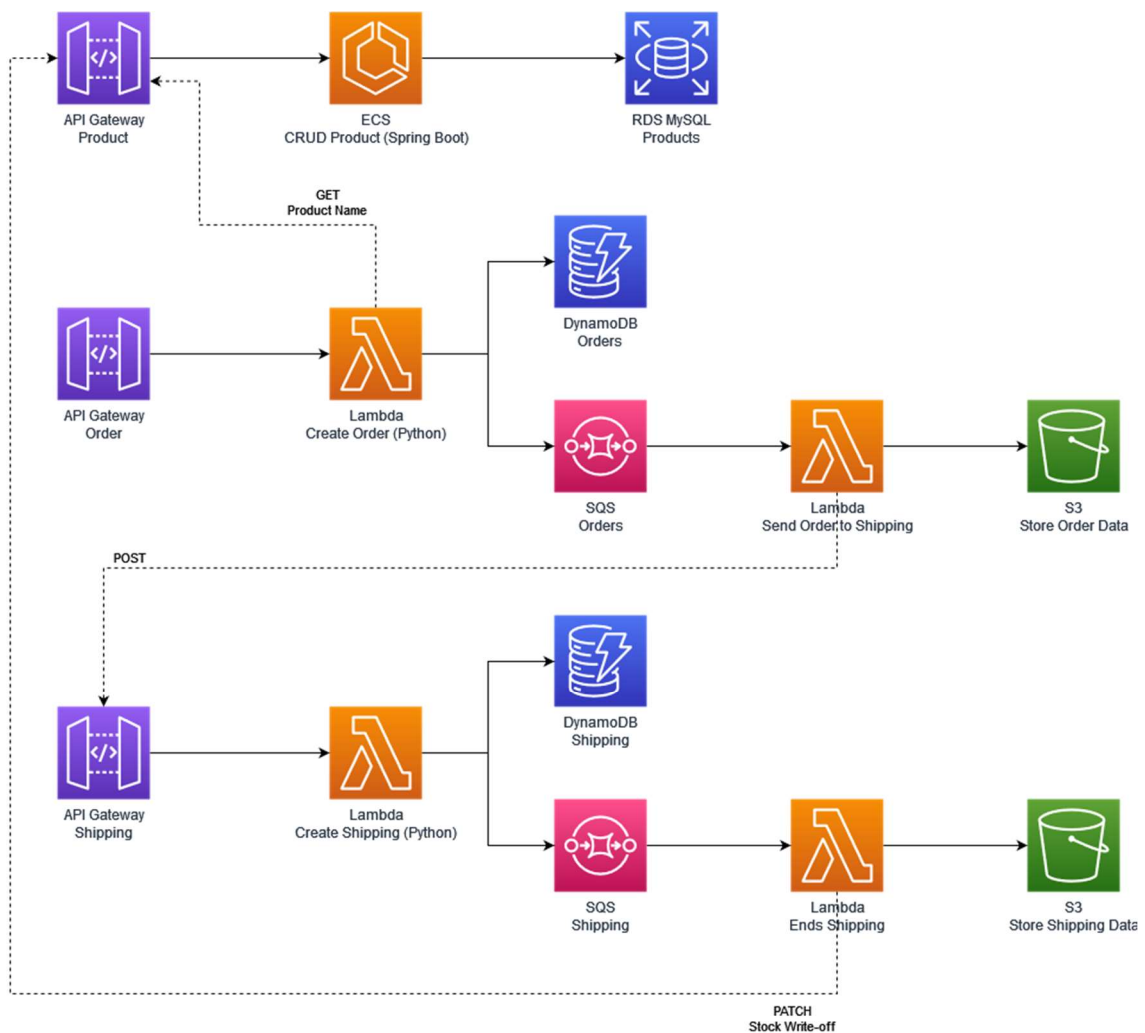


## DESAFIO FINAL

O desafio final consiste em criar uma aplicação de pedidos e envio de produtos, conforme diagrama abaixo.

O projeto é separado em 3 APIs: Produto, Pedido e Transporte. A API de produto será disponibilizada pelos professores, e será somente consumida.

Você deverá construir uma das outras APIs, de Pedido ou de Transporte, que será definido pelo instrutor. As instruções para cada API estão abaixo



## ENDPOINT Pedido

1. A sua aplicação deve ser capaz de receber uma solicitação via API (POST) de pedidos. O payload do pedido virá no formato json na seguinte forma:

```
{
  "produtos": [
    {
      "id_produto": "number",
      "quantidade": "number"
    },
    {
      "id_produto": "number",
      "quantidade": "number"
    }
  ],
  "url_transportadora": "string"
}
```

- a. Obter o nome do produto na API (GET) de produto. O retorno será no formato:
  - i. URL API Produto: [http://url.com/{id\\_produto}](http://url.com/{id_produto})
  - ii. Pedir a URL para o instrutor

```
{
  "id_produto": "string",
  "nome_produto": "string"
}
```

2. Salvar no DynamoDB essa solicitação, no formato que achar necessário
3. Criar fila no SQS e postar mensagem no formato que achar necessário, para disparar um novo lambda
  - a. Opcional: Customizar e otimizar a fila do jeito que achar necessário
4. Ler a mensagem do SQS e enviar a requisição de transporte para a API da transportadora. O endpoint a ser utilizado deve ser a url recebida no payload inicial.
  - a. Opcional: Para testar sua aplicação de forma integrada, pedir para outros alunos responsáveis pela API da transportadora a URL disponível
  - b. O formato da requisição, onde nome\_loja é o nome da sua loja, que você pode escolher

```
{
  "produtos": [
    {
      "id_produto": "number",
      "quantidade": "number"
    },
    {
      "id_produto": "number",
      "quantidade": "number"
    }
  ],
  "nome_loja": "string"
}
```

5. Salvar os dados do pedido e do envio no S3, no formato que achar necessário

#### ENDPOINT Transportadora

1. A sua aplicação deve ser capaz de receber uma solicitação via API (POST) de transportes. O payload do pedido virá no formato json na seguinte forma:

```
{
  "produtos": [
    {
      "id_produto": "number",
      "quantidade": "number"
    },
    {
      "id_produto": "number",
      "quantidade": "number"
    }
  ],
  "nome_loja": "string"
}
```

2. Salvar no DynamoDB essa solicitação, no formato que achar necessário
3. Criar fila no SQS e postar mensagem no formato que achar necessário, para disparar um novo lambda
  - a. Opcional: Customizar e otimizar a fila do jeito que achar necessário
4. Ler a mensagem do SQS para realizar a finalização do transporte.
  - a. Antes de finalizar o transporte, enviar uma solicitação para a API de produto para dar baixa no estoque
    - i. PATCH na URL : [http://url.com/estoque/{id\\_produto}](http://url.com/estoque/{id_produto})
    - ii. Pedir a URL para o instrutor
    - iii. O formato da requisição será

```
{
  "quantidade": "number"
}
```

5. Salvar os dados do envio no S3, no formato que achar necessário