

Centro Universitário SENAC

Sistema de Controle de Estoque - Solução para Problemas de Gerenciamento  
Stock Control System - Solution to Management Problems

Anderson  
Gustavo  
Mateus  
Maximiliano

São Paulo, 22/06/2023

# Sumário

<b>Resumo.....</b>	<b>1</b>
<b>Abstract.....</b>	<b>2</b>
<b>Introdução.....</b>	<b>3</b>
<b>Objetivo.....</b>	<b>3</b>
<b>Metodologia.....</b>	<b>4</b>
Regras do negócio.....	4
Tecnologias.....	5
HTML.....	5
CSS.....	5
Javascript.....	5
Bootstrap.....	5
Java.....	5
Quarkus.....	6
Oracle Express Edition (XE).....	6
Arquitetura.....	6
Modelo C4.....	6
Modelo Entidade X Relacionamento.....	7
Objetos criados no banco de dados Oracle.....	8
Construção dos dados no banco.....	9
Funcionalidades.....	10
Cadastrar um produto com seu ponto de reposição.....	10
Realizar uma movimentação de cadastro de produto em um depósito com seu ponto de reposição.....	11
Registrar um produto para um fornecedor.....	11
Realizar uma movimentação alterando a quantidade de um produto em um depósito.	12
Listar a movimentação geral do estoque.....	12
Listar as movimentações de um produto.....	15
Buscar os produtos com quantidade abaixo do limite cadastrado para o estoque....	16
Buscar os produtos com quantidade abaixo do limite cadastrado para cada depósito.	17
Buscar os fornecedores que tem um produto específico.....	17
Listar uma relação de depósitos com produtos.....	18
<b>Conclusões.....</b>	<b>20</b>
Referente a proposta do projeto.....	20
Referente ao sistema desenvolvido.....	20
Referente a experiência em construir o projeto.....	20
<b>Referências Bibliográficas.....</b>	<b>22</b>
<b>Apêndice 1 - Exemplos de requisições dos serviços.....</b>	<b>23</b>
<b>Apêndice 2 - Comandos SQL para criar os dados necessários do projeto.....</b>	<b>39</b>

# Resumo

Empresas enfrentam desafios significativos no gerenciamento eficiente de estoques, incluindo falta de visibilidade, dificuldades no rastreamento de movimentações e falta de controle sobre localização física dos produtos. Esses problemas resultam em consequências negativas, como falta de produtos para atender à demanda dos clientes e perdas financeiras.

Para superar esses desafios, propomos o desenvolvimento de um sistema de controle de estoque eficiente e intuitivo. O sistema fornecerá visibilidade das quantidades de produtos disponíveis, permitindo um controle preciso e atualizado. Além disso, o sistema possibilitará o rastreamento de movimentações de entrada e saída, fornecendo informações sobre as transações realizadas.

Palavras chave: gerenciamento eficiente de estoques, rastreamento de movimentações de entrada e saída, controle sobre localização física dos produtos, informações sobre as transações realizadas.

# Abstract

Companies face significant challenges in effectively managing inventory, including lack of visibility, difficulties in tracking movements and lack of control over the physical location of products. These problems result in negative consequences such as lack of products to meet customer demand and financial losses.

To overcome these challenges, we propose the development of an efficient and intuitive inventory control system. The system will provide visibility of the quantities of products available, allowing accurate and up-to-date control. In addition, the system will enable the tracking of incoming and outgoing movements, providing information on the transactions carried out.

Keywords: efficient inventory management, tracking incoming and outgoing movements, control over the physical location of products, information on transactions carried out.

# Introdução

Muitas empresas enfrentam desafios significativos no gerenciamento eficiente de seus estoques. Os problemas comuns incluem:

- falta de visibilidade das quantidades de produtos disponíveis,
- dificuldade em rastrear movimentações de entrada e saída,
- falta de controle sobre localização física dos produtos no estoque
- e falta de análises precisas sobre o desempenho do estoque.

Essas questões podem levar a consequências negativas, como:

- falta de produtos para atender à demanda dos clientes,
- excesso de estoque resultando em capital imobilizado e custos adicionais,
- atrasos nas entregas e perdas financeiras devido a produtos vencidos ou obsoletos.

Um artigo publicado pelo mentor e conselheiro Pedro Janot ao Portal de Empreendedorismo da Endeavor em 11 de março de 2015 (e atualizado em 04 de maio de 2017) lista os (considerado por ele) "cinco pecados da gestão de estoque", que são:

- Perder de vista os 20% dos produtos que representam 80% da sua venda.
- Perder o pico da sazonalidade.
- Negociar prazos e descontos com fornecedores sem prever atrasos e reajustes.
- Não alimentar continuamente as lojas com itens básicos.
- Não se antecipar.

Esses "pecados" destacam a relevância e a necessidade de um sistema de controle de estoque eficiente para solucionar os problemas comuns enfrentados pelas empresas, além de melhorar sua eficiência operacional.

## Objetivo

O objetivo deste projeto é desenvolver um sistema de controle de estoque eficiente e intuitivo que permita acompanhar e gerenciar o estoque de uma empresa de forma automatizada. O sistema fornecerá informações sobre:

- os produtos disponíveis,
- suas quantidades,
- movimentações
- e localização.

Além de oferecer funcionalidades avançadas para otimizar o gerenciamento e evitar problemas como falta de estoque.

# Metodologia

## Regras do negócio

Nosso projeto aplica algumas regras e direcionamentos para garantir sua operabilidade:

- O usuário consegue cadastrar, editar, visualizar, e deletar departamentos, produtos, e fornecedores.
- Departamentos, produtos e fornecedores possuem um atributo chamado código cuja finalidade é auxiliar nas consultas, entretanto as movimentações são feitas usando o ID.
- Os produtos possuem um atributo chamado quantidade mínima, e sua função é a de indicar quando o usuário deve fazer uma "reposição".
- Os produtos possuem um atributo chamado média, este atributo é atualizado sempre que ocorre uma entrada de produto no estoque com nota fiscal.
- Um depósito pode conter vários produtos, assim como um produto pode estar em vários depósitos.
- Um produto pode ser fornecido por vários fornecedores, assim como um fornecedor pode fornecer vários produtos.
- O usuário pode relacionar os produtos com os fornecedores registrados no sistema.
- A adição e remoção de produtos nos depósitos é feita via "movimentações".
- O usuário consegue fazer 6 tipos de "movimentações" no sistema:
  - (1) Entrada por Nota Fiscal.
  - (2) Entrada por doação.
  - (3) Saída por Nota Fiscal.
  - (4) Saída por doação.
  - (5) Transferência entre depósitos (entrada).
  - (6) Transferência entre depósitos (saída).
- As movimentações podem:
  - Adicionar um produto em um depósito.
  - Aumentar a quantidade de um produto em um depósito.
  - Diminuir a quantidade de um produto em um depósito.
- Sempre que ocorrer a adição de um produto em um depósito a quantidade mínima deste produto deve ser informada junto, que é diferente da quantidade mínima do estoque inteiro (informado diretamente no produto).
- Se houver a necessidade de transferir produtos entre depósitos é necessário efetuar duas movimentações.
- Estas movimentações podem ou não ter valor monetário.
- Repetindo... Sempre que ocorrer uma movimentação de entrada por nota fiscal o sistema irá atualizar a média do produto.

## Tecnologias

A motivação do uso destas tecnologias é aplicar na prática o material apresentado nas disciplinas abaixo, promovidas pelo Centro Universitário Senac no curso de pós graduação Full Stack Developer:

- Desenvolvimento Back-end e Arquitetura de Software
- Padrões e Linguagens Front-End

### HTML

HTML é a linguagem de marcação padrão para páginas da Web.

### CSS

CSS é a linguagem que usamos para estilizar um documento HTML. CSS descreve como os elementos HTML devem ser exibidos.

### Javascript

JavaScript é uma linguagem de programação que permite a você implementar itens complexos em páginas web — toda vez que uma página da web faz mais do que simplesmente mostrar a você informação estática — mostrando conteúdo que se atualiza em um intervalo de tempo, mapas interativos ou gráficos 2D/3D animados, etc.

### Bootstrap

Bootstrap é um kit de ferramentas de front-end poderoso e repleto de recursos.

### Java

Java é uma linguagem de programação e plataforma de computação lançada pela primeira vez pela Sun Microsystems em 1995. Ela evoluiu de um começo humilde para alimentar uma grande parte do mundo digital de hoje, fornecendo uma plataforma confiável sobre a qual muitos serviços e aplicativos são construídos. Novos produtos inovadores e serviços digitais projetados para o futuro também continuam a depender do Java.

## Quarkus

Quarkus é uma estrutura Java nativa do Kubernetes adaptada para GraalVM e HotSpot, criada a partir das melhores bibliotecas e padrões Java. O objetivo é tornar o Java a plataforma líder em Kubernetes e ambientes sem servidor, oferecendo aos desenvolvedores uma estrutura para atender a uma ampla variedade de arquiteturas de aplicativos distribuídos.

## Oracle Express Edition (XE)

Banco de Dados Oracle (Oracle Database) é um Sistema Gerenciador de Banco de Dados (SGDB) - isso é, um software que permite a administração de dados computacionais - multiplataforma e de natureza objeto-relacional comercializado pela Oracle Corporation.

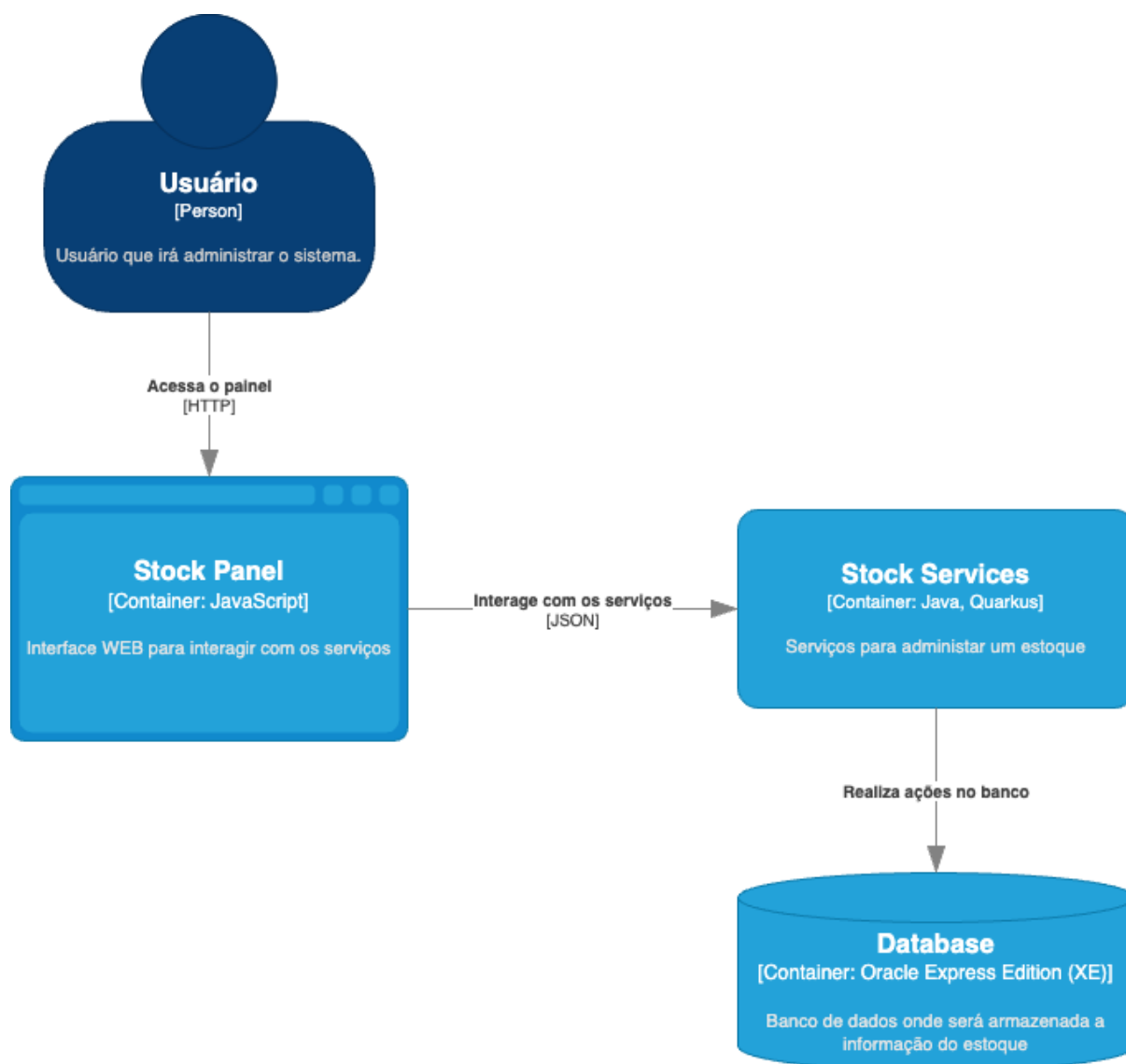
## Arquitetura

Toda a regra de negócio está sendo aplicada no banco de dados Oracle, os serviços e o frontend são apenas uma "casca" deste processo.

O frontend não opera todas as funcionalidades disponíveis pelo backend, este no momento opera apenas as possíveis movimentações em produtos (listagem, edição e remoção) mais a listagem das movimentações.

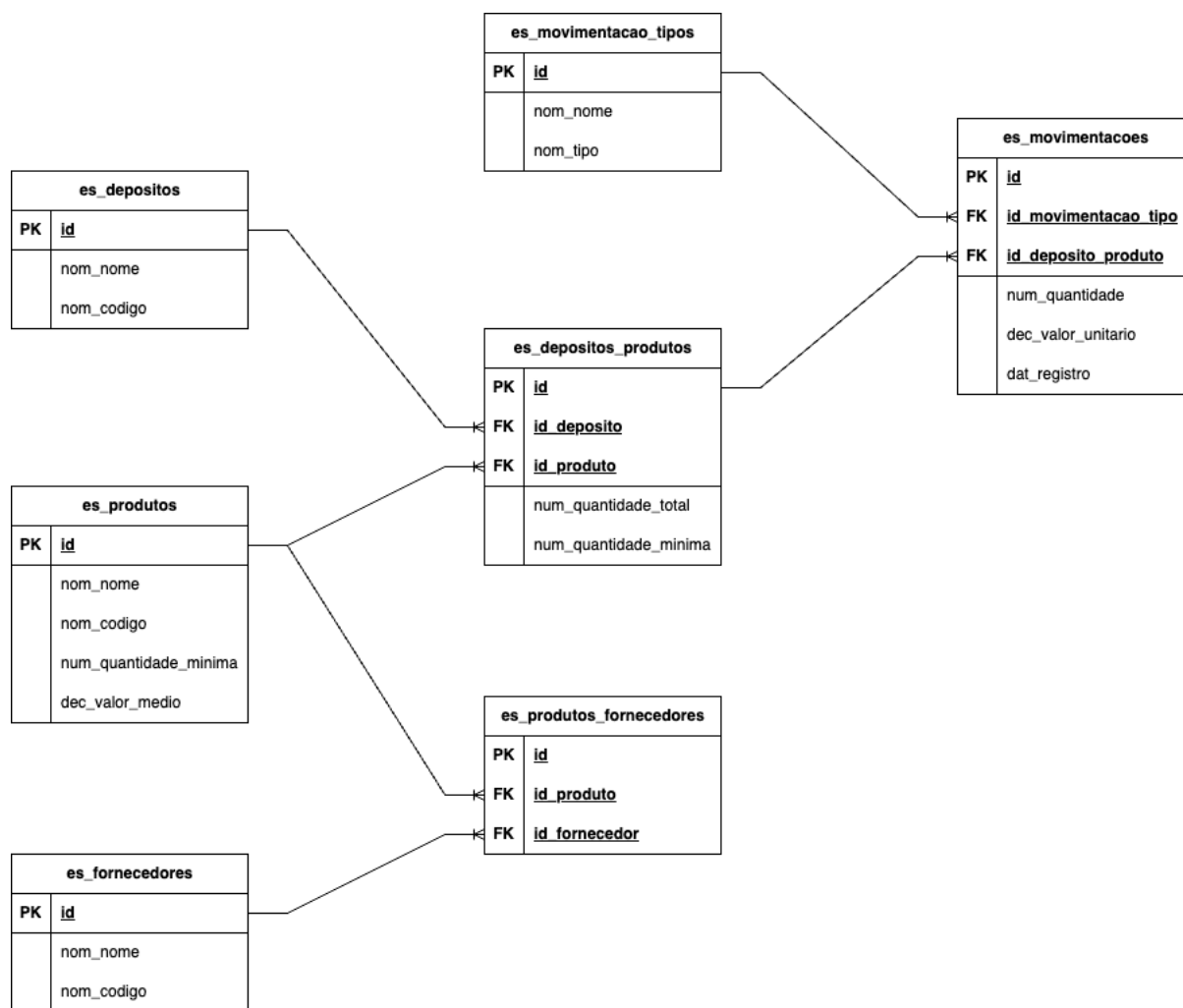
## Modelo C4





Modelo C4 do sistema de Estoque

## Modelo Entidade X Relacionamento



Modelo Entidade Relacionamento (MER)

Objetos criados no banco de dados Oracle

<b>ES_DEPOSITOS</b> <b>Table</b> Status: Valid Created 6 seconds ago	<b>ES_DEPOSITOS_PRODUTOS</b> <b>Table</b> Status: Valid Created 6 seconds ago	<b>ES_FORNECEDORES</b> <b>Table</b> Status: Valid Created 6 seconds ago
<b>ES_MOVIMENTACAO_TIPOS</b> <b>Table</b> Status: Valid Created 6 seconds ago	<b>ES_MOVIMENTACOES</b> <b>Table</b> Status: Valid Created 6 seconds ago	<b>ES_PRODUTOS</b> <b>Table</b> Status: Valid Created 6 seconds ago
<b>ES_PRODUTOS_FORNECEDORES</b> <b>Table</b> Status: Valid Created 6 seconds ago	<b>ISEQ\$\$_180375573</b> <b>Sequence</b> Status: Valid Created 6 seconds ago	<b>ISEQ\$\$_180375578</b> <b>Sequence</b> Status: Valid Created 6 seconds ago
<b>ISEQ\$\$_180375582</b> <b>Sequence</b> Status: Valid Created 6 seconds ago	<b>ISEQ\$\$_180375586</b> <b>Sequence</b> Status: Valid Created 6 seconds ago	<b>ISEQ\$\$_180375590</b> <b>Sequence</b> Status: Valid Created 6 seconds ago
<b>ISEQ\$\$_180375594</b> <b>Sequence</b> Status: Valid Created 6 seconds ago	<b>PRC_ADICIONAR_PRODUTO_DEPOSITO</b> <b>Procedure</b> Status: Valid Created 5 seconds ago	<b>PRC_ATUALIZAR_QUANTIDADE_PRODUTO_DEPOSITO</b> <b>Procedure</b> Status: Valid Created 5 seconds ago
<b>PRC_REGISTRAR_PRODUTO_FORNECEDOR</b> <b>Procedure</b> Status: Valid Created 5 seconds ago	<b>VW_LISTAR_DEPOSITOS_PRODUTOS</b> <b>View</b> Status: Valid Created 5 seconds ago	<b>VW_LISTAR_MEDIAS</b> <b>View</b> Status: Valid Created 5 seconds ago
<b>VW_LISTAR_MOVIMENTACOES</b> <b>View</b> Status: Valid Created 6 seconds ago	<b>VW_LISTAR_PRODUTOS</b> <b>View</b> Status: Valid Created 5 seconds ago	<b>VW_LISTAR_PRODUTOS_FORNECEDORES</b> <b>View</b> Status: Valid Created 5 seconds ago

## Construção dos dados no banco

Todas as tabelas possuem o prefixo "es", que são as iniciais de "estoque".

Com exceção da tabela "es\_movimentacao\_tipos", todos os IDs das tabelas possuem a configuração "number generated always as identity". Isso faz com que o Oracle crie e administre uma "sequence".

A tabela "es\_movimentacao\_tipos" já deve estar preenchida com os tipos de movimentações possíveis, e o ID delas é muito importante para as movimentações fornecidas pelo sistema.

Os tipos de movimentações possíveis são:

ID	nom_nome	nom_tipo
1	Entrada por Nota Fiscal	e
2	Entrada por Doação	e

3	Saída por Nota Fiscal	s
4	Saída por Doação	s
5	Transferência entre depósitos (entrada)	e
6	Transferência entre depósitos (saída)	s

Todos os atributos das tabelas possuem o seu tipo como prefixo.

Todas as chaves estrangeiras possuem índice.

Na tabela "es\_movimentacao\_tipos" o atributo "nom\_tipo" possui a constraint ""check (nom\_tipo in('e', 's'))". Isso é para que as "procedures de movimentação" saibam somar ou subtrair a quantidade de produtos em um depósito.

Nas tabelas "es\_depositos", "es\_produtos", e "es\_fornecedores" o atributo "nom\_codigo" possui a constraint "unique". O Oracle cria um índice para constraint "unique".

Os atributos "id\_deposito" e "id\_produto" da tabela "es\_depositos\_produtos" possuem a constraint "unique (id\_deposito, id\_produto)". Isso é para garantir que não tenhamos o mesmo produto mais de uma vez em um depósito. O Oracle cria um índice para constraint "unique".

Os atributos "id\_produto" e "id\_fornecedor" da tabela "es\_produtos\_fornecedores" possuem a constraint "unique (id\_produto, id\_fornecedor)". Isso é para garantir que não tenhamos o mesmo produto mais de uma vez com o mesmo fornecedor. O Oracle cria um índice para constraint "unique".

Os atributos "num\_quantidade\_total" da tabela "es\_depositos\_produtos" possuem a constraint "check (num\_quantidade\_total >= 0)". Isso é para garantir que não seja possível retirar mais produtos do que o existente no depósito.

Para adicionar produtos a depósitos temos a procedure "prc\_adicionar\_produto\_deposito". Esta procedure também atualiza a média do produto se a movimentação for "entrada por nota fiscal (1)".

Para atualizar a quantidade de produtos em depósitos temos a procedure "prc\_atualizar\_quantidade\_produto\_deposito".

Esta procedure também atualiza a média do produto se a movimentação for "entrada por nota fiscal (1)".

Temos várias views para facilitar a consulta de dados, especialmente quando existe um relacionamento N:N.

Todos os comandos SQL implementados estão disponíveis no "Apêndice 2 - Comandos SQL para criar os dados necessários do projeto".

## Funcionalidades

O projeto possui "premissas" necessárias para sua entrega.

Estas premissas serão descritas abaixo, entretanto todas as requisições implementadas nos serviços estão no "Apêndice 1 - Exemplos de requisições dos serviços".

### Cadastrar um produto com seu ponto de reposição

- POST /estoque/produtos

```
curl -X 'POST' \  
  'http://localhost:9090/estoque/produtos' \  
  -H 'accept: application/json' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "codigo": "sand_0001",  
    "nome": "Sandalias Havaianas",  
    "quantidadeMinima": 100  
  }'
```

```
exec prc_registrar_produto_fornecedor({codigoProduto}, {codigoFornecedor});
```

Realizar uma movimentação de cadastro de produto em um depósito com seu ponto de reposição

- POST /estoque/depositos/{codigoDeposito}/movimentacoes

```
curl -X 'POST' \  
  'http://localhost:9090/estoque/depositos/riog_0001/movimentacoes' \  
  -H 'accept: application/json' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "codigo": 1,  
    "codigoProduto": "teni_0001",  
    "quantidade": 826,  
    "quantidadeMinima": 30,  
  }'
```

```
"valorUnitario": 956.87
}'
```

```
exec prc_adicionar_produto_deposito({codigoProduto}, {quantidade}, {quantidadeMinima},
{codigo}, {valorUnitario});
```

## Registrar um produto para um fornecedor

- POST /estoque/produtos\_fornecedores

```
curl -X 'POST' \
'http://localhost:9090/estoque/produtos_fornecedores' \
-H 'accept: application/json' \
-H 'Content-Type: application/json' \
-d '{
  "codigoProduto": "teni_0001",
  "codigoFornecedor": "acme_0001"
}'
```

```
exec prc_registrar_produto_fornecedor({codigoProduto}, {codigoFornecedor});
```

## Realizar uma movimentação alterando a quantidade de um produto em um depósito

- POST  
/estoque/depositos/{codigoDeposito}/produtos/{codigoProduto}/movimentacoes

```
curl -X 'POST' \
'http://localhost:9090/estoque/depositos/saop_0001/produtos/teni_0001/movimentacoes' \
-H 'accept: application/json' \
-H 'Content-Type: application/json' \
-d '{
  "codigo": 3,
  "quantidade": 5
}'
```

```
exec prc_atualizar_quantidade_produto_deposito({codigoDeposito}, {codigoProduto},
{quantidade}, {codigo});
```

## Listar a movimentação geral do estoque

- GET /estoque/movimentacoes

```
curl -X 'GET' \  
  'http://localhost:9090/estoque/movimentacoes' \  
  -H 'accept: application/json'
```

```
select  
  num_codigo,  
  nom_movimentacao,  
  nom_codigo_deposito,  
  nom_deposito,  
  nom_codigo_produto,  
  nom_produto,  
  num_quantidade,  
  dec_valor_unitario,  
  dec_valor_total,  
  dat_registro "  
from  
  vw_listar_movimentacoes;
```

```
[  
  {  
    "codigo": "1",  
    "codigoDeposito": "salv_0001",  
    "codigoProduto": "smar_0001",  
    "deposito": "Salvador",  
    "movimentacao": "Entrada por Nota Fiscal",  
    "produto": "Smartphone",  
    "quantidade": "800",  
    "registro": "2023-06-23 20:41:34",  
    "valorTotal": "840000",  
    "valorUnitario": 1050  
  },  
  {  
    "codigo": "3",  
    "codigoDeposito": "salv_0001",  
    "codigoProduto": "smar_0001",  
    "deposito": "Salvador",  
    "movimentacao": "Saída por Nota Fiscal",  
    "produto": "Smartphone",  
    "quantidade": "50",  
    "registro": "2023-06-23 20:41:34",  
    "valorUnitario": 0  
  },  
]
```

```

{
  "codigo": "2",
  "codigoDeposito": "salv_0001",
  "codigoProduto": "smar_0001",
  "deposito": "Salvador",
  "movimentacao": "Entrada por Doação",
  "produto": "Smartphone",
  "quantidade": "100",
  "registro": "2023-06-23 20:41:35",
  "valorUnitario": 0
},
{
  "codigo": "1",
  "codigoDeposito": "salv_0001",
  "codigoProduto": "smar_0001",
  "deposito": "Salvador",
  "movimentacao": "Entrada por Nota Fiscal",
  "produto": "Smartphone",
  "quantidade": "55",
  "registro": "2023-06-23 20:41:35",
  "valorTotal": "110027.5",
  "valorUnitario": 2000.5
},
{
  "codigo": "6",
  "codigoDeposito": "salv_0001",
  "codigoProduto": "smar_0001",
  "deposito": "Salvador",
  "movimentacao": "Transferência entre depósitos (saída)",
  "produto": "Smartphone",
  "quantidade": "110",
  "registro": "2023-06-23 20:41:35",
  "valorUnitario": 0
},
{
  "codigo": "5",
  "codigoDeposito": "saop_0001",
  "codigoProduto": "smar_0001",
  "deposito": "São Paulo",
  "movimentacao": "Transferência entre depósitos (entrada)",
  "produto": "Smartphone",
  "quantidade": "110",
  "registro": "2023-06-23 20:41:35",
  "valorUnitario": 0
},
{
  "codigo": "1",
  "codigoDeposito": "saop_0001",
  "codigoProduto": "teni_0001",
  "deposito": "São Paulo",
  "movimentacao": "Entrada por Nota Fiscal",
  "produto": "Tênis esportivo",
  "quantidade": "433",
  "registro": "2023-06-23 20:41:35",

```



```

    "valorTotal": "385759.7",
    "valorUnitario": 890.9
  },
  {
    "codigo": "1",
    "codigoDeposito": "riog_0001",
    "codigoProduto": "teni_0001",
    "deposito": "Rio Grande do Norte",
    "movimentacao": "Entrada por Nota Fiscal",
    "produto": "Tênis esportivo",
    "quantidade": "826",
    "registro": "2023-06-23 21:17:09",
    "valorTotal": "790374.62",
    "valorUnitario": 956.87
  },
  {
    "codigo": "3",
    "codigoDeposito": "saop_0001",
    "codigoProduto": "teni_0001",
    "deposito": "São Paulo",
    "movimentacao": "Saída por Nota Fiscal",
    "produto": "Tênis esportivo",
    "quantidade": "5",
    "registro": "2023-06-23 21:32:45",
    "valorTotal": "0",
    "valorUnitario": 0
  }
]

```

## Listar as movimentações de um produto

- GET /estoque/produtos/{codigoProduto}/movimentacoes

```

curl -X 'GET' \
  'http://localhost:9090/estoque/produtos/teni_0001/movimentacoes' \
  -H 'accept: application/json'

```

```

select
  num_codigo,
  nom_movimentacao,
  nom_codigo_deposito,
  nom_deposito,
  nom_codigo_produto,
  nom_produto,
  num_quantidade,
  dec_valor_unitario,
  dec_valor_total, dat_registro
from

```

```
vw_listar_movimentacoes
where
  nom_codigo_produto = {codigoProduto}
```

```
[
  {
    "codigo": "1",
    "codigoDeposito": "saop_0001",
    "codigoProduto": "teni_0001",
    "deposito": "São Paulo",
    "movimentacao": "Entrada por Nota Fiscal",
    "produto": "Tênis esportivo",
    "quantidade": "433",
    "registro": "2023-06-24 21:48:10",
    "valorTotal": "385759.7",
    "valorUnitario": 890.9
  },
  {
    "codigo": "1",
    "codigoDeposito": "salv_0001",
    "codigoProduto": "teni_0001",
    "deposito": "Salvador",
    "movimentacao": "Entrada por Nota Fiscal",
    "produto": "Tênis esportivo",
    "quantidade": "200",
    "registro": "2023-06-24 22:08:13",
    "valorTotal": "179960",
    "valorUnitario": 899.8
  }
]
```

Buscar os produtos com quantidade abaixo do limite cadastrado para o estoque

- GET /estoque/produtos/para\_repor

```
curl -X 'GET' \
  'http://localhost:9090/estoque/produtos/para_repor' \
  -H 'accept: application/json'
```

```
select
  id,
  nom_nome,
  nom_codigo,
```

```

    num_quantidade_minima,
    dec_valor_medio,
    num_quantidade
from
    vw_listar_produtos
where
    num_quantidade < num_quantidade_minima;

```

```

[
  {
    "codigo": "smar_0001",
    "id": "1",
    "nome": "Smartphone",
    "quantidade": "905",
    "quantidadeMinima": "3000",
    "valorMedio": "1111.14"
  },
  {
    "codigo": "relo_0001",
    "id": "3",
    "nome": "Relógio de pulso",
    "quantidade": "0",
    "quantidadeMinima": "800",
    "valorMedio": "0"
  }
]

```

Buscar os produtos com quantidade abaixo do limite cadastrado para cada depósito

- GET /estoque/depositos\_produtos/para\_repor

```

curl -X 'GET' \
  'http://localhost:9090/estoque/depositos_produtos/para_repor' \
  -H 'accept: application/json'

```

```

select
    nom_codigo_desposito,
    nom_desposito,
    nom_codigo_produto,
    nom_produto,
    num_quantidade,
    num_quantidade_minima

```

```

from
  vw_listar_depositos_produtos
where
  num_quantidade < num_quantidade_minima;

```

```

[
  {
    "codigoDeposito": "salv_0001",
    "codigoProduto": "teni_0001",
    "deposito": "Salvador",
    "produto": "Tênis esportivo",
    "quantidade": "200",
    "quantidadeMinima": "300"
  }
]

```

Buscar os fornecedores que tem um produto específico

- GET /estoque/produtos/{codigoProduto}/fornecedores

```

curl -X 'GET' \
  'http://localhost:9090/estoque/produtos/smar_0001/fornecedores' \
  -H 'accept: application/json'

```

```

select
  nom_produto,
  nom_codigo_produto,
  nom_fornecedor,
  nom_codigo_fornecedor
from
  vw_listar_produtos_fornecedores
where
  nom_codigo_produto = {codigo};

```

```

[
  {
    "codigoFornecedor": "glob_0001",
    "codigoProduto": "smar_0001",
    "fornecedor": "Global Solutions",
    "produto": "Smartphone"
  },
  {
    "codigoFornecedor": "acme_0001",

```

```

    "codigoProduto": "smar_0001",
    "fornecedor": "Acme Corporation",
    "produto": "Smartphone"
  }
]

```

## Listar uma relação de depósitos com produtos

- GET /estoque/depositos\_produtos

```

curl -X 'GET' \
  'http://localhost:9090/estoque/depositos_produtos' \
  -H 'accept: application/json'

```

```

select
  nom_codigo_desposito,
  nom_desposito,
  nom_codigo_produto,
  nom_produto,
  num_quantidade,
  num_quantidade_minima
from
  vw_listar_depositos_produtos;

```

```

[
  {
    "codigoDeposito": "riog_0001",
    "codigoProduto": "teni_0001",
    "deposito": "Rio Grande do Norte",
    "produto": "Tênis esportivo",
    "quantidade": "826",
    "quantidadeMinima": "30"
  },
  {
    "codigoDeposito": "salv_0001",
    "codigoProduto": "smar_0001",
    "deposito": "Salvador",
    "produto": "Smartphone",
    "quantidade": "795",
    "quantidadeMinima": "100"
  },
  {
    "codigoDeposito": "saop_0001",
    "codigoProduto": "teni_0001",

```

```
"deposito": "São Paulo",  
"produto": "Tênis esportivo",  
"quantidade": "428",  
"quantidadeMinima": "300"  
},  
{  
  "codigoDeposito": "saop_0001",  
  "codigoProduto": "smar_0001",  
  "deposito": "São Paulo",  
  "produto": "Smartphone",  
  "quantidade": "110",  
  "quantidadeMinima": "80"  
}  
]
```

# Conclusões

## Referente a proposta do projeto

Ao implementar um sistema de controle de estoque as empresas podem obter maior visibilidade, controle preciso das quantidades e melhor tomada de decisões.

Mas, apesar dos benefícios que um sistema de controle de estoque eficiente pode proporcionar, existem desafios e limitações que devem ser considerados:

- Implementação inicial
- Manutenção de dados precisos
- Integração com outros sistemas

Entretanto, investir em treinamento e conscientização da equipe, além de promover uma cultura de controle de estoque, são passos fundamentais para o sucesso do sistema.

Ou seja, o sistema de controle de estoque é uma solução essencial para enfrentar os desafios do gerenciamento eficiente de estoques. Embora existam desafios e limitações a serem considerados, é possível superar essas questões com as sugestões adequadas.

## Referente ao sistema desenvolvido

As funcionalidades apresentadas pelo sistema garantem um bom controle de um estoque:

- Cadastro de produtos, depósitos, fornecedores
- Gerência de estoque mínimos tanto para o estoque quanto para os depósitos
- Relatórios de movimentações

Mas durante o desenvolvimento foi percebido a necessidade de novas features para um controle mais preciso:

- Possibilidade de modificar a quantidade mínima de um produto em um depósito
- Possibilidade de remover um produto cadastrado para um fornecedor
- Registrar uma relação de transferência entre depósitos
- Responder o status http correto quando ocorre cenários diferentes de sucesso.

## Referente a experiência em construir o projeto

Foi um desafio para o time implementar este sistema, especialmente porque ele foi trabalhado com muitas tecnologias que não temos o costume de trabalhar: Java, Quarkus, e Oracle.

Entretanto, em nenhum momento estas tecnologias foram impostas ao projeto, estas tecnologias foram as apresentadas durante o curso que propôs o projeto.

Ou seja, foi uma decisão do time "nos desafiar" a usar essas tecnologias (foi desafiador porque queríamos que fosse).

E foi muito gratificante ver o sistema rodando, especialmente porque através dele conseguimos visualizar toda a trajetória de conhecimento adquirido por meio de pesquisas e "colocando em prática" tudo o que foi apresentado em aulas.



## Referências Bibliográficas

- <https://endeavor.org.br/operacoes/gestao-de-estoque/>
- <https://www.w3schools.com/html/>
- <https://www.w3schools.com/css/>
- [https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First\\_steps/What is Java Script](https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_Java_Script)
- <https://getbootstrap.com/docs/5.3/getting-started/introduction/>
- [https://www.java.com/en/download/help/whatis\\_java.html](https://www.java.com/en/download/help/whatis_java.html)
- <https://quarkus.io/about/>
- <https://www.portalgsti.com.br/oracle-database/sobre/>
- <https://www.oracle.com/br/database/technologies/appdev/xe.html>

## Apêndice 1 - Exemplos de requisições dos serviços

```
# Exemplos de requisições
```

```
## Buscar os depósitos
```

```
* GET /estoque/depositos
```

```
```sh
curl -X 'GET' \
  'http://localhost:9090/estoque/depositos' \
  -H 'accept: application/json'
```
```

```
```json
[
  {
    "codigo": "salv_0001",
    "id": "1",
    "nome": "Salvador"
  },
  {
    "codigo": "saop_0001",
    "id": "2",
    "nome": "São Paulo"
  },
  {
    "codigo": "riog_0001",
    "id": "3",
    "nome": "Rio Grande do Sul"
  }
]
```
```

```
## Criar um deposito
```

```
* POST /estoque/depositos
```

```
```sh
curl -X 'POST' \
  'http://localhost:9090/estoque/depositos' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "codigo": "mato_0001",
    "nome": "Mato Grosso do Sul"
  }'
```
```

```

```json
{
  "mensagem": "Deposito cadastrado!"
}
...

## Buscar um depósito

* GET /estoque/depositos/{codigoDeposito}

```sh
curl -X 'GET' \
  'http://localhost:9090/estoque/depositos/saop_0001' \
  -H 'accept: application/json'
...

```json
{
  "codigo": "saop_0001",
  "id": "2",
  "nome": "São Paulo"
}
...

## Criar uma movimentação adicionando um produto a um depósito

* /estoque/depositos/{codigoDeposito}/movimentacoes

```sh
curl -X 'POST' \
  'http://localhost:9090/estoque/depositos/riog_0001/movimentacoes' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "codigo": 1,
    "codigoProduto": "teni_0001",
    "quantidade": 826,
    "quantidadeMinima": 30,
    "valorUnitario": 956.87
  }'
...

```json
{
  "mensagem": "Movimentação registrada!"
}
...

## Buscar os produtos de um depósito

* GET /estoque/depositos/{codigoDeposito}/produtos

```sh
curl -X 'GET' \

```

```
'http://localhost:9090/estoque/depositos/saop_0001/produtos' \
-H 'accept: application/json'
...

```json
[
  {
    "codigoDeposito": "saop_0001",
    "codigoProduto": "teni_0001",
    "deposito": "São Paulo",
    "produto": "Tênis esportivo",
    "quantidade": "433",
    "quantidadeMinima": "300"
  },
  {
    "codigoDeposito": "saop_0001",
    "codigoProduto": "smar_0001",
    "deposito": "São Paulo",
    "produto": "Smartphone",
    "quantidade": "110",
    "quantidadeMinima": "80"
  }
]
...

## Buscar um produto de um depósito

* GET /estoque/depositos/{codigoDeposito}/produtos/{codigoProduto}

```sh
curl -X 'GET' \
'http://localhost:9090/estoque/depositos/saop_0001/produtos/teni_0001' \
-H 'accept: application/json'
...

```json
{
  "codigoDeposito": "saop_0001",
  "codigoProduto": "teni_0001",
  "deposito": "São Paulo",
  "produto": "Tênis esportivo",
  "quantidade": "433",
  "quantidadeMinima": "300"
}
...

## Criar uma movimentação atualizando a quantidade de um produto em um depósito

* POST /estoque/depositos/{codigoDeposito}/produtos/{codigoProduto}/movimentacoes

```sh
curl -X 'POST' \
'http://localhost:9090/estoque/depositos/saop_0001/produtos/teni_0001/movimentacoes' \

```

```

-H 'accept: application/json' \
-H 'Content-Type: application/json' \
-d '{
  "codigo": 3,
  "quantidade": 5,
  "valorUnitario": 0
}'
...

```json
{
  "mensagem": "Movimentação registrada!"
}
...

## Editar um depósito

* PUT /estoque/depositos/{id}

```sh
curl -X 'PUT' \
  'http://localhost:9090/estoque/depositos/3' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "nome": "Rio Grande do Norte"
  }'
...

```json
{
  "mensagem": "Deposito alterado!"
}
...

## Deletar um depósito

* DELETE /estoque/depositos/{id}

```sh
curl -X 'DELETE' \
  'http://localhost:9090/estoque/depositos/4' \
  -H 'accept: application/json'
...

```json
{
  "mensagem": "Depósito apagado!"
}
...

## Buscar relação de depósitos com produtos

* GET /estoque/depositos_produtos

```

```
```sh
curl -X 'GET' \
  'http://localhost:9090/estoque/depositos_produtos' \
  -H 'accept: application/json'
```
```

```
```json
[
  {
    "codigoDeposito": "riog_0001",
    "codigoProduto": "teni_0001",
    "deposito": "Rio Grande do Norte",
    "produto": "Tênis esportivo",
    "quantidade": "826",
    "quantidadeMinima": "30"
  },
  {
    "codigoDeposito": "salv_0001",
    "codigoProduto": "smar_0001",
    "deposito": "Salvador",
    "produto": "Smartphone",
    "quantidade": "795",
    "quantidadeMinima": "100"
  },
  {
    "codigoDeposito": "saop_0001",
    "codigoProduto": "teni_0001",
    "deposito": "São Paulo",
    "produto": "Tênis esportivo",
    "quantidade": "428",
    "quantidadeMinima": "300"
  },
  {
    "codigoDeposito": "saop_0001",
    "codigoProduto": "smar_0001",
    "deposito": "São Paulo",
    "produto": "Smartphone",
    "quantidade": "110",
    "quantidadeMinima": "80"
  }
]
```
```

## Buscar a relação de depósitos com produtos que estão com estoque baixo

\* GET /estoque/depositos\_produtos/para\_repor

```
```sh
curl -X 'GET' \
  'http://localhost:9090/estoque/depositos_produtos/para_repor' \
  -H 'accept: application/json'
```
```

```

```json
[
  {
    "codigoDeposito": "salv_0001",
    "codigoProduto": "teni_0001",
    "deposito": "Salvador",
    "produto": "Tênis esportivo",
    "quantidade": "200",
    "quantidadeMinima": "300"
  }
]
...

## Buscar os fornecedores

* GET /estoque/fornecedores

```sh
curl -X 'GET' \
  'http://localhost:9090/estoque/fornecedores' \
  -H 'accept: application/json'
...

```json
[
  {
    "codigo": "acme_0001",
    "id": "1",
    "nome": "Acme Corporation"
  },
  {
    "codigo": "glob_0001",
    "id": "2",
    "nome": "Global Solutions"
  },
  {
    "codigo": "prim_0001",
    "id": "3",
    "nome": "Prime Industries"
  }
]
...

## Criar um fornecedor

* POST /estoque/fornecedores

```sh
curl -X 'POST' \
  'http://localhost:9090/estoque/fornecedores' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "nome": "Amazon Brasil",

```

```

    "codigo": "amaz_0001"
  }
  ...

  ```json
  {
    "mensagem": "Fornecedor cadastrado!"
  }
  ...

  ## Buscar um fornecedor

  * GET /estoque/fornecedores/{codigoFornecedor}

  ```sh
  curl -X 'GET' \
    'http://localhost:9090/estoque/fornecedores/acme_0001' \
    -H 'accept: application/json'
  ...

  ```json
  {
    "codigo": "acme_0001",
    "id": "1",
    "nome": "Acme Corporation"
  }
  ...

  ## Buscar os produtos de um fornecedor

  * GET /estoque/fornecedores/{codigoFornecedor}/produtos

  ```sh
  curl -X 'GET' \
    'http://localhost:9090/estoque/fornecedores/glob_0001/produtos' \
    -H 'accept: application/json'
  ...

  ```json
  [
    {
      "codigoFornecedor": "glob_0001",
      "codigoProduto": "smar_0001",
      "fornecedor": "Global Solutions",
      "produto": "Smartphone"
    },
    {
      "codigoFornecedor": "glob_0001",
      "codigoProduto": "teni_0001",
      "fornecedor": "Global Solutions",
      "produto": "Tênis esportivo"
    }
  ]
  ...

```



### ## Editar um fornecedor

\* PUT /estoque/fornecedores/{id}

```
```sh
curl -X 'PUT' \
  'http://localhost:9090/estoque/fornecedores/3' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "nome": "Amazon EUA",
    "codigo": "amaz_0002"
  }'
...`
```

```
```json
{
  "mensagem": "Fornecedor alterado!"
}
...`
```

### ## Deletar um fornecedor

\* DELETE /estoque/fornecedores/{id}

```
```sh
curl -X 'DELETE' \
  'http://localhost:9090/estoque/fornecedores/3' \
  -H 'accept: application/json'
...`
```

```
```json
{
  "mensagem": "Fornecedor apagado!"
}
...`
```

### ## Buscar as movimentações

\* GET /estoque/movimentacoes

```
```sh
curl -X 'GET' \
  'http://localhost:9090/estoque/movimentacoes' \
  -H 'accept: application/json'
...`
```

```
```json
[
  {
    "codigo": "1",
    "codigoDeposito": "salv_0001",
    "codigoProduto": "smar_0001",
  }
]
...`
```

```

"deposito": "Salvador",
"movimentacao": "Entrada por Nota Fiscal",
"produto": "Smartphone",
"quantidade": "800",
"registro": "2023-06-23 20:41:34",
"valorTotal": "840000",
"valorUnitario": 1050
},
{
  "codigo": "3",
  "codigoDeposito": "salv_0001",
  "codigoProduto": "smar_0001",
  "deposito": "Salvador",
  "movimentacao": "Saída por Nota Fiscal",
  "produto": "Smartphone",
  "quantidade": "50",
  "registro": "2023-06-23 20:41:34",
  "valorUnitario": 0
},
{
  "codigo": "2",
  "codigoDeposito": "salv_0001",
  "codigoProduto": "smar_0001",
  "deposito": "Salvador",
  "movimentacao": "Entrada por Doação",
  "produto": "Smartphone",
  "quantidade": "100",
  "registro": "2023-06-23 20:41:35",
  "valorUnitario": 0
},
{
  "codigo": "1",
  "codigoDeposito": "salv_0001",
  "codigoProduto": "smar_0001",
  "deposito": "Salvador",
  "movimentacao": "Entrada por Nota Fiscal",
  "produto": "Smartphone",
  "quantidade": "55",
  "registro": "2023-06-23 20:41:35",
  "valorTotal": "110027.5",
  "valorUnitario": 2000.5
},
{
  "codigo": "6",
  "codigoDeposito": "salv_0001",
  "codigoProduto": "smar_0001",
  "deposito": "Salvador",
  "movimentacao": "Transferência entre depósitos (saída)",
  "produto": "Smartphone",
  "quantidade": "110",
  "registro": "2023-06-23 20:41:35",
  "valorUnitario": 0
},
{

```

```

"codigo": "5",
"codigoDeposito": "saop_0001",
"codigoProduto": "smar_0001",
"deposito": "São Paulo",
"movimentacao": "Transferência entre depósitos (entrada)",
"produto": "Smartphone",
"quantidade": "110",
"registro": "2023-06-23 20:41:35",
"valorUnitario": 0
},
{
"codigo": "1",
"codigoDeposito": "saop_0001",
"codigoProduto": "teni_0001",
"deposito": "São Paulo",
"movimentacao": "Entrada por Nota Fiscal",
"produto": "Tênis esportivo",
"quantidade": "433",
"registro": "2023-06-23 20:41:35",
"valorTotal": "385759.7",
"valorUnitario": 890.9
},
{
"codigo": "1",
"codigoDeposito": "riog_0001",
"codigoProduto": "teni_0001",
"deposito": "Rio Grande do Norte",
"movimentacao": "Entrada por Nota Fiscal",
"produto": "Tênis esportivo",
"quantidade": "826",
"registro": "2023-06-23 21:17:09",
"valorTotal": "790374.62",
"valorUnitario": 956.87
},
{
"codigo": "3",
"codigoDeposito": "saop_0001",
"codigoProduto": "teni_0001",
"deposito": "São Paulo",
"movimentacao": "Saída por Nota Fiscal",
"produto": "Tênis esportivo",
"quantidade": "5",
"registro": "2023-06-23 21:32:45",
"valorTotal": "0",
"valorUnitario": 0
}
]
...

```

## Buscar os produtos

\* GET /estoque/produtos

```sh

```
curl -X 'GET' \
  'http://localhost:9090/estoque/produtos' \
  -H 'accept: application/json'
...
```

```
```json
[
  {
    "codigo": "smar_0001",
    "id": "1",
    "nome": "Smartphone",
    "quantidade": "905",
    "quantidadeMinima": "3000",
    "valorMedio": "1111.14"
  },
  {
    "codigo": "teni_0001",
    "id": "2",
    "nome": "Tênis esportivo",
    "quantidade": "1254",
    "quantidadeMinima": "500",
    "valorMedio": "934.18"
  },
  {
    "codigo": "relo_0001",
    "id": "3",
    "nome": "Relógio de pulso",
    "quantidade": "0",
    "quantidadeMinima": "800",
    "valorMedio": "0"
  }
]
...
```

## Buscar produtos com estoque baixo

\* GET /estoque/produtos/para\_repor

```
```sh
curl -X 'GET' \
  'http://localhost:9090/estoque/produtos/para_repor' \
  -H 'accept: application/json'
...
```

```
```json
[
  {
    "codigo": "smar_0001",
    "id": "1",
    "nome": "Smartphone",
    "quantidade": "905",
    "quantidadeMinima": "3000",
    "valorMedio": "1111.14"
  },
  {
    "codigo": "teni_0001",
    "id": "2",
    "nome": "Tênis esportivo",
    "quantidade": "1254",
    "quantidadeMinima": "500",
    "valorMedio": "934.18"
  },
  {
    "codigo": "relo_0001",
    "id": "3",
    "nome": "Relógio de pulso",
    "quantidade": "0",
    "quantidadeMinima": "800",
    "valorMedio": "0"
  }
]
...
```

```
{
  "codigo": "relo_0001",
  "id": "3",
  "nome": "Relógio de pulso",
  "quantidade": "0",
  "quantidadeMinima": "800",
  "valorMedio": "0"
}
]
...
```

## ## Criar um produto

\* POST /estoque/produtos

```
```sh
curl -X 'POST' \
  'http://localhost:9090/estoque/produtos' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "codigo": "sand_0001",
    "nome": "Sandalias Havaianas",
    "quantidadeMinima": 100
  }'
...
```
```

```
```json
{
  "mensagem": "Produto cadastrado!"
}
...
```
```

## ## Buscar um produto

\* GET /estoque/produtos/{codigoProduto}

```
```sh
curl -X 'GET' \
  'http://localhost:9090/estoque/produtos/teni_0001' \
  -H 'accept: application/json'
...
```
```

```
```json
{
  "codigo": "teni_0001",
  "id": "2",
  "nome": "Tênis esportivo",
  "quantidade": "1254",
  "quantidadeMinima": "500",
  "valorMedio": "934.18"
}
...
```
```

```
## Buscar os depósitos que tem um produto
```

```
* GET /estoque/produtos/{codigoProduto}/depositos
```

```
```sh
curl -X 'GET' \
  'http://localhost:9090/estoque/produtos/teni_0001/depositos' \
  -H 'accept: application/json'
```
```

```
```json
[
  {
    "codigoDeposito": "riog_0001",
    "codigoProduto": "teni_0001",
    "deposito": "Rio Grande do Norte",
    "produto": "Tênis esportivo",
    "quantidade": "826",
    "quantidadeMinima": "30"
  },
  {
    "codigoDeposito": "saop_0001",
    "codigoProduto": "teni_0001",
    "deposito": "São Paulo",
    "produto": "Tênis esportivo",
    "quantidade": "428",
    "quantidadeMinima": "300"
  }
]
```
```

```
## Buscar os fornecedores que tem um produto
```

```
* GET /estoque/produtos/{codigoProduto}/fornecedores
```

```
```sh
curl -X 'GET' \
  'http://localhost:9090/estoque/produtos/smar_0001/fornecedores' \
  -H 'accept: application/json'
```
```

```
```json
[
  {
    "codigoFornecedor": "glob_0001",
    "codigoProduto": "smar_0001",
    "fornecedor": "Global Solutions",
    "produto": "Smartphone"
  },
  {
    "codigoFornecedor": "acme_0001",
    "codigoProduto": "smar_0001",
    "fornecedor": "Acme Corporation",
    "produto": "Smartphone"
  }
]
```

```
}
]
```

## Buscar as movimentações de um produto

\* GET /estoque/produtos/{codigoProduto}/movimentacoes

```
```sh
```

```
curl -X 'GET' \
  'http://localhost:9090/estoque/produtos/teni_0001/movimentacoes' \
  -H 'accept: application/json'
```

```
```
```

```
```json
```

```
[
  {
    "codigo": "1",
    "codigoDeposito": "saop_0001",
    "codigoProduto": "teni_0001",
    "deposito": "São Paulo",
    "movimentacao": "Entrada por Nota Fiscal",
    "produto": "Tênis esportivo",
    "quantidade": "433",
    "registro": "2023-06-24 21:48:10",
    "valorTotal": "385759.7",
    "valorUnitario": 890.9
  },
  {
    "codigo": "1",
    "codigoDeposito": "salv_0001",
    "codigoProduto": "teni_0001",
    "deposito": "Salvador",
    "movimentacao": "Entrada por Nota Fiscal",
    "produto": "Tênis esportivo",
    "quantidade": "200",
    "registro": "2023-06-24 22:08:13",
    "valorTotal": "179960",
    "valorUnitario": 899.8
  }
]
```

```
```
```

## Editar um produto

\* PUT /estoque/produtos/{id}

```
```sh
```

```
curl -X 'PUT' \
  'http://localhost:9090/estoque/produtos/21' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "nome": "Sandálias Havaianas Femininas",
```

```

    "quantidadeMinima": 200
  }
  ...

  ```json
  {
    "mensagem": "Produto alterado!"
  }
  ...

## Deletar um produto

* DELETE /estoque/produtos/{id}

```sh
curl -X 'DELETE' \
  'http://localhost:9090/estoque/produtos/21' \
  -H 'accept: application/json'
```

```json
{
  "mensagem": "Produto apagado!"
}
...

## Buscar relação de produtos com fornecedores

* GET /estoque/produtos_fornecedores

```sh
curl -X 'GET' \
  'http://localhost:9090/estoque/produtos_fornecedores' \
  -H 'accept: application/json'
```

```json
[
  {
    "codigoFornecedor": "acme_0001",
    "codigoProduto": "smar_0001",
    "fornecedor": "Acme Corporation",
    "produto": "Smartphone"
  },
  {
    "codigoFornecedor": "glob_0001",
    "codigoProduto": "smar_0001",
    "fornecedor": "Global Solutions",
    "produto": "Smartphone"
  },
  {
    "codigoFornecedor": "glob_0001",
    "codigoProduto": "teni_0001",
    "fornecedor": "Global Solutions",

```



```
"produto": "Tênis esportivo"
}
]
...

## Registrar um produto com um fornecedor

* POST /estoque/produtos_fornecedores

```sh
curl -X 'POST' \
  'http://localhost:9090/estoque/produtos_fornecedores' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "codigoProduto": "teni_0001",
    "codigoFornecedor": "acme_0001"
  }'
...

```json
{
  "mensagem": "Produto e fornecedor cadastrado!"
}
...
```

## Apêndice 2 - Comandos SQL para criar os dados necessários do projeto

-- Tabelas

```
create table es_movimentacao_tipos(  
    id number not null,  
    nom_nome varchar2(100) not null,  
    nom_tipo varchar2(1) not null,  
    primary key (id),  
    constraint tipo_ck check (nom_tipo in('e', 's'))  
);
```

```
create table es_depositos(  
    id number generated always as identity,  
    nom_nome varchar2(100) not null,  
    nom_codigo varchar2(100) not null unique,  
    primary key (id)  
);
```

```
create table es_produtos(  
    id number generated always as identity,  
    nom_nome varchar2(100) not null,  
    nom_codigo varchar2(100) not null unique,  
    num_quantidade_minima number not null,  
    dec_valor_medio decimal(10,2) default 0 not null,  
    primary key (id)  
);
```

```
create table es_fornecedores(  
    id number generated always as identity,  
    nom_nome varchar2(100) not null,  
    nom_codigo varchar2(100) not null unique,  
    primary key (id)  
);
```

```
create table es_depositos_produtos(  
    id number generated always as identity,  
    id_deposito number not null,  
    id_produto number not null,  
    num_quantidade_total number not null,  
    num_quantidade_minima number not null,  
    primary key (id),  
    constraint depoprod_depo_fk foreign key (id_deposito) references es_depositos(id),  
    constraint depoprod_prod_fk foreign key (id_produto) references es_produtos(id),  
    constraint deposito_produto_uk unique (id_deposito, id_produto),  
    constraint num_quantidade_total_ck check (num_quantidade_total >= 0)  
);
```

```
create table es_produtos_fornecedores(  
    id number generated always as identity,  
    id_produto number not null,  
    id_fornecedor number not null,  
    num_quantidade number not null,  
    primary key (id),  
    constraint prod_fornecedor_fk foreign key (id_produto, id_fornecedor) references es_produtos(id),  
    constraint prod_fornecedor_uk unique (id_produto, id_fornecedor)
```

```

id number generated always as identity,
id_produto number not null,
id_fornecedor number not null,
primary key (id),
constraint prodform_prod_fk foreign key (id_produto) references es_produtos(id),
constraint prodform_forn_fk foreign key (id_fornecedor) references es_fornecedores(id),
constraint produto_fornecedor_uk unique (id_produto, id_fornecedor)
);

create table es_movimentacoes(
id number generated always as identity,
id_movimentacao_tipo number not null,
id_deposito_produto number not null,
num_quantidade number not null,
dec_valor_unitario decimal(10,2),
dat_registro timestamp,
primary key (id),
constraint movi_movitipo_fk foreign key (id_movimentacao_tipo) references
es_movimentacao_tipos(id),
constraint movi_depoproduct_fk foreign key (id_deposito_produto) references
es_depositos_produtos(id)
);

create index depoproduct_depo_fk_i on es_depositos_produtos (id_deposito);
create index depoproduct_prod_fk_i on es_depositos_produtos (id_produto);
create index prodform_prod_fk_i on es_produtos_fornecedores (id_produto);
create index prodform_forn_fk_i on es_produtos_fornecedores (id_fornecedor);
create index movi_movitipo_fk_i on es_movimentacoes (id_movimentacao_tipo);
create index movi_depoproduct_fk_i on es_movimentacoes (id_deposito_produto);

-- Views

create view vw_listar_movimentacoes as
select
es_movimentacao_tipos.id as num_codigo,
es_movimentacao_tipos.nom_nome as nom_movimentacao,
es_depositos.nom_codigo as nom_codigo_deposito,
es_depositos.nom_nome as nom_deposito,
es_produtos.nom_codigo as nom_codigo_produto,
es_produtos.nom_nome as nom_produto,
es_movimentacoes.num_quantidade as num_quantidade,
es_movimentacoes.dec_valor_unitario as dec_valor_unitario,
(es_movimentacoes.num_quantidade * es_movimentacoes.dec_valor_unitario) as
dec_valor_total,
to_char(es_movimentacoes.dat_registro, 'yyyy-mm-dd hh24:mi:ss') as dat_registro
from
es_movimentacoes,
es_movimentacao_tipos,
es_depositos_produtos,
es_depositos,
es_produtos
where
es_movimentacao_tipos.id = es_movimentacoes.id_movimentacao_tipo
and es_depositos.id = es_depositos_produtos.id_deposito

```

```

    and es_produtos.id = es_depositos_produtos.id_produto
    and es_depositos_produtos.id = es_movimentacoes.id_deposito_produto
order by
    es_movimentacoes.dat_registro;

create view vw_listar_medias as
select
    nom_codigo_produto,
    nom_produto,
    sum(dec_valor_total) as dec_valor_total,
    sum(num_quantidade) as num_quantidade,
    round((sum(dec_valor_total) / sum(num_quantidade)), 2) as dec_media
from vw_listar_movimentacoes
where num_codigo = 1
group by nom_codigo_produto, nom_produto;

create view vw_listar_depositos_produtos as
select
    es_depositos.nom_codigo as nom_codigo_desposito,
    es_depositos.nom_nome as nom_desposito,
    es_produtos.nom_codigo as nom_codigo_produto,
    es_produtos.nom_nome as nom_produto,
    es_depositos_produtos.num_quantidade_total as num_quantidade,
    es_depositos_produtos.num_quantidade_minima as num_quantidade_minima
from
    es_depositos_produtos,
    es_produtos,
    es_depositos
where
    es_depositos.id = es_depositos_produtos.id_deposito
    and es_produtos.id = es_depositos_produtos.id_produto
order by
    es_depositos.nom_codigo;

create view vw_listar_produtos_fornecedores as
select
    es_produtos.nom_nome as nom_produto,
    es_produtos.nom_codigo as nom_codigo_produto,
    es_fornecedores.nom_nome as nom_fornecedor,
    es_fornecedores.nom_codigo as nom_codigo_fornecedor
from
    es_produtos_fornecedores,
    es_fornecedores,
    es_produtos
where
    es_produtos.id = es_produtos_fornecedores.id_produto
    and es_fornecedores.id = es_produtos_fornecedores.id_fornecedor
order by
    es_produtos.nom_nome;

create view vw_listar_produtos as
select
    es_produtos.id as id,
    es_produtos.nom_nome as nom_nome,

```

```

    es_produtos.nom_codigo as nom_codigo,
    es_produtos.num_quantidade_minima as num_quantidade_minima,
    es_produtos.dec_valor_medio as dec_valor_medio,
    nvl(sum(vw_listar_depositos_produtos.num_quantidade), 0) as num_quantidade
from
    es_produtos
left join
    vw_listar_depositos_produtos
on
    vw_listar_depositos_produtos.nom_codigo_produto = es_produtos.nom_codigo
group by
    es_produtos.id,
    es_produtos.nom_nome,
    es_produtos.nom_codigo,
    es_produtos.num_quantidade_minima,
    es_produtos.dec_valor_medio
order by
    es_produtos.id;

-- Procedures

create or replace procedure prc_adicionar_produto_deposito(
    p_codigo_deposito in varchar2, p_codigo_produto in varchar2, p_quantidade in
number, p_quantidade_minima in number, p_movimentacao in number, p_valor_unitario in
number default null
) is
begin
    insert into es_depositos_produtos (id_deposito, id_produto, num_quantidade_total,
num_quantidade_minima)
    values (
        (select id from es_depositos where nom_codigo = p_codigo_deposito),
        (select id from es_produtos where nom_codigo = p_codigo_produto),
        p_quantidade,
        p_quantidade_minima
    );

    insert into es_movimentacoes (id_movimentacao_tipo, id_deposito_produto,
num_quantidade, dec_valor_unitario, dat_registro)
    values (
        p_movimentacao,
        (
            select id from es_depositos_produtos
            where
                id_deposito = (select id from es_depositos where nom_codigo =
p_codigo_deposito)
                and id_produto = (select id from es_produtos where nom_codigo =
p_codigo_produto)
        ),
        p_quantidade,
        p_valor_unitario,
        systimestamp
    );

    if p_movimentacao = 1 then

```

```

        update es_produtos set dec_valor_medio = (select dec_media from vw_listar_medias
where nom_codigo_produto = p_codigo_produto)
        where nom_codigo = p_codigo_produto;
    end if;
    commit;
end;
/

create or replace procedure prc_atualizar_quantidade_produto_deposito(
    p_codigo_deposito in varchar2, p_codigo_produto in varchar2, p_quantidade in
number, p_movimentacao in number, p_valor_unitario in number default null
) is
    v_tipo varchar2(1);
begin
    select nom_tipo into v_tipo from es_movimentacao_tipos where id = p_movimentacao;

    if v_tipo = 'e' then
        update es_depositos_produtos set num_quantidade_total = (num_quantidade_total +
p_quantidade)
        where
            id_deposito = (select id from es_depositos where nom_codigo =
p_codigo_deposito)
            and id_produto = (select id from es_produtos where nom_codigo =
p_codigo_produto);
    else
        update es_depositos_produtos set num_quantidade_total = (num_quantidade_total -
p_quantidade)
        where
            id_deposito = (select id from es_depositos where nom_codigo =
p_codigo_deposito)
            and id_produto = (select id from es_produtos where nom_codigo =
p_codigo_produto);
    end if;

    insert into es_movimentacoes (id_movimentacao_tipo, id_deposito_produto,
num_quantidade, dec_valor_unitario, dat_registro)
    values (
        p_movimentacao,
        (
            select id from es_depositos_produtos
            where
                id_deposito = (select id from es_depositos where nom_codigo =
p_codigo_deposito)
                and id_produto = (select id from es_produtos where nom_codigo =
p_codigo_produto)
        ),
        p_quantidade,
        p_valor_unitario,
        systimestamp
    );

    if p_movimentacao = 1 then
        update es_produtos set dec_valor_medio = (select dec_media from vw_listar_medias
where nom_codigo_produto = p_codigo_produto)

```

```

        where nom_codigo = p_codigo_produto;
    end if;
    commit;
end;
/

create or replace procedure prc_registrar_produto_fornecedor(
    p_codigo_produto in varchar2, p_codigo_fornecedor in varchar2
) is
begin
    insert into es_produtos_fornecedores (id_produto, id_fornecedor)
    values (
        (select id from es_produtos where nom_codigo = p_codigo_produto),
        (select id from es_fornecedores where nom_codigo = p_codigo_fornecedor)
    );
    commit;
end;
/

-- Dados

insert into es_movimentacao_tipos (id, nom_nome, nom_tipo) values (1, 'Entrada por Nota
Fiscal', 'e');
insert into es_movimentacao_tipos (id, nom_nome, nom_tipo) values (2, 'Entrada por
Doação', 'e');
insert into es_movimentacao_tipos (id, nom_nome, nom_tipo) values (3, 'Saída por Nota
Fiscal', 's');
insert into es_movimentacao_tipos (id, nom_nome, nom_tipo) values (4, 'Saída por
Doação', 's');
insert into es_movimentacao_tipos (id, nom_nome, nom_tipo) values (5, 'Transferência
entre depósitos (entrada)', 'e');
insert into es_movimentacao_tipos (id, nom_nome, nom_tipo) values (6, 'Transferência
entre depósitos (saída)', 's');

insert into es_depositos (nom_nome, nom_codigo) values ('Salvador', 'salv_0001');
insert into es_depositos (nom_nome, nom_codigo) values ('São Paulo', 'saop_0001');
insert into es_depositos (nom_nome, nom_codigo) values ('Rio Grande do Sul',
'riog_0001');

insert into es_produtos (nom_nome, nom_codigo, num_quantidade_minima) values
('Smartphone', 'smar_0001', 3000);
insert into es_produtos (nom_nome, nom_codigo, num_quantidade_minima) values
('Tênis esportivo', 'teni_0001', 500);
insert into es_produtos (nom_nome, nom_codigo, num_quantidade_minima) values
('Relógio de pulso', 'relo_0001', 800);

insert into es_fornecedores (nom_nome, nom_codigo) values ('Acme Corporation',
'acme_0001');
insert into es_fornecedores (nom_nome, nom_codigo) values ('Global Solutions',
'glob_0001');
insert into es_fornecedores (nom_nome, nom_codigo) values ('Prime Industries',
'prim_0001');

exec prc_registrar_produto_fornecedor('smar_0001', 'acme_0001');

```

```
exec prc_registrar_produto_fornecedor('smar_0001', 'glob_0001');  
exec prc_registrar_produto_fornecedor('teni_0001', 'glob_0001');  
  
exec prc_adicionar_produto_deposito('salv_0001', 'smar_0001', 800, 100, 1, 1050);  
exec prc_atualizar_quantidade_produto_deposito('salv_0001', 'smar_0001', 50, 3);  
exec prc_atualizar_quantidade_produto_deposito('salv_0001', 'smar_0001', 100, 2);  
exec prc_atualizar_quantidade_produto_deposito('salv_0001', 'smar_0001', 55, 1,  
2000.5);  
exec prc_atualizar_quantidade_produto_deposito('salv_0001', 'smar_0001', 110, 6);  
exec prc_adicionar_produto_deposito('saop_0001', 'smar_0001', 110, 80, 5);  
exec prc_adicionar_produto_deposito('saop_0001', 'teni_0001', 433, 300, 1, 890.9);
```