

DESAFIO 03 – Desenvolvimento de uma aplicação para um Banco utilizando Spring Boot.

O projeto consiste no desenvolvimento de uma estrutura “Backend” que compreenda em uma aplicação para o funcionamento de um ecommerce, utilizando as tecnologias e conhecimentos previamente vistos nas trilhas da bolsa.

Entrega:

O desafio será em individual e o código deverá ser disponibilizado no GitHub e o link do repositório deve ser enviado por e-mail.

Envio do e-mail:

Assunto:

- [PB-MAI24] Desafio 3 - [nome do Bolsista]

Envie o e-mail para os seguintes destinatários:

Para:

- franciele.ciostek@compasso.com.br
- diego.bonetti@compasso.com.br
- gabryel.melo@compasso.com.br
- gladson.ramos@compasso.com.br
- angelo.fabris@compasso.com.br
- jean.lima@compasso.com.br

Apresentação:

As apresentações serão realizadas a partir do dia **09/09/2024** com sorteio da ordem de apresentação a ser realizada de forma antecipada em horário ao que seria a Daily, sendo importante a presença de todos os colegas. No dia da apresentação o bolsista **deverá** clonar ou realizar o “pull” da última versão válida (**até a data limite de entrega**) do repositório para apresentar.

Notas:

Nota da Implementação: Os Instrutores avaliarão o desenvolvimento e a utilização dos itens citados na seção “Regras de negócio da aplicação” e o uso de boas práticas de programação aprendidos durante os cursos da trilha.

Nota da Apresentação: Iremos avaliar o domínio com relação ao próprio código e a postura durante a apresentação.

Agenda:

Data	O que faremos	Observação
22/08/2024	Início do Desafio	Lançamento do Desafio durante a Daily.
22/08 a 06/09	Daily com Tira Dúvidas	A reunião da Daily será reservada para consulta de dúvidas.
05/09/2024	Sorteio da ordem de Apresentações	Todos deverão participar do sorteio
06/09/2024	Entrega do Desafio	Prazo máximo 12h.
09/09/2024	Início das Apresentações conforme sorteio realizado.	As apresentações serão realizadas das 10:30h às 11:30h na data e nos dias seguintes até que todos os bolsistas realizem suas apresentações;

Regras de negócio da aplicação

Visão geral da aplicação:

Nesta avaliação técnica deverá ser implementado uma aplicação de ecommerce que seja capaz de criar e gerenciar produtos, estoque dos produtos, vendas e usuários, conforme descrito abaixo.

Requisitos básicos:

- **Tecnologias:** Java 17+
- **Framework:** Spring Boot
- **Diagrama:** Entidade-Relacionamento (ER) ou Unified Modeling Language (UML).

- **Banco de dados:** O sistema deve utilizar um banco de dados **RELACIONAL** (PostgreSQL, MySQL...).
- **Arquitetura de API:** Escolher uma arquitetura de API para organizar o projeto (Layered Architecture, Hexagonal Architecture, Clean Architecture, etc...).
- **Boas práticas:** O código deve ser desenvolvido seguindo as boas práticas de desenvolvimento de software, para garantir a qualidade e a manutenibilidade.
- **Versionamento:** O código fonte deve estar no repositório do GitHub.

Requisitos funcionais:

- 1) Permitir que os usuários **criem, leiam, atualizem e excluam** produtos. **PESO (3)**
- 2) Criar validações que façam sentido para os dados de entrada, por exemplo: Preço do produto só pode ser positivo. **PESO (2)**
- 3) Um produto não pode ser **DELETADO** após ser incluso em uma venda, porém deve ter alguma maneira de **INATIVAR** ele. **PESO (2)**
- 4) Controlar o estoque do produto de forma que ele **não possa ser vendido** caso o estoque seja menor do que a quantidade necessária para a venda ou igual a zero. **PESO (3)**
- 5) Permitir que os usuários **criem, leiam, atualizem e excluam** vendas (Uma venda tem que ter **no mínimo 1** produto para ser concluída). **PESO (3)**
- 6) Criar métodos de relatório de vendas por data (**informada pelo cliente**), por mês e pela semana atual (**considerar dias úteis**). **PESO (1)**
- 7) Os métodos “**GET ALL**” de Produtos e Vendas devem salvar as informações no **CACHE** da aplicação, para que as próximas buscas sejam mais rápidas. Deve ser feito um bom gerenciamento do cache, por exemplo: ao inserir uma nova venda, deletar o cache, para que a informação seja buscada no banco de dados diretamente e venha atualizada. **PESO (1)**
- 8) Todas as **EXCEÇÕES** devem ser tratadas e seguir o mesmo padrão de resposta. **PESO (1)**
- 9) Todos os campos data, devem seguir o padrão **ISO 8601** (exemplo: 2023-07-20T12:00:00Z). **PESO (1)**
- 10) Implementar **Autenticação** via Token **JWT**. **PESO (3)**
- 11) Implementar **Autorização** para os usuários autenticados. **PESO (3)**
- 12) Implementar um método para **resetar** a senha do usuário. **PESO (3)**
- 13) **Somente** usuários com permissão de **ADMIN** podem **deletar** informações do sistema. **PESO (3)**
- 14) **Somente** usuários com permissão de **ADMIN** podem **cadastrar e atualizar** novos produtos e usuários. **PESO (2)**

Legenda:

- Os pesos informados são para fins exclusivamente avaliativos, e são relevantes somente para os mentores.