

Desafio – Teste Prático para Desenvolvedor Java

O que eu preciso fazer?

1. Faça o download do *Teste-Pratico-Desenvolvedor-Java.zip* com o projeto base para seu desenvolvimento.
2. Implemente o desafio conforme abaixo.
3. Quando finalizar seu projeto, faça um push para um repositório de sua conta no GitHub.
4. Grave um vídeo explicando sobre as decisões que você tomou para implementação deste desafio.
5. Envie um e-mail avisando que finalizou o desafio com a url do seu repositório, link do vídeo que foi gravado e aguarde nosso contato.

Sobre o desafio:

Controle de Agendamentos de Vagas:

Deverá ser criada uma aplicação para registro de cadastros e implementação de validações. Considere que sua aplicação receberá a média de 300 agendamentos por dia de vários solicitantes. Desta forma, para implementar suas consultas observe critérios que possam garantir boa performance e agilidade nas respostas aos usuários.

Back-end:

Aplicação em Java com Spring Boot, JPA / Hibernate, conforme projeto base baixado.

Front-end:

JSF com Primefaces.

Requisitos para a aplicação:

Permitir cadastro de vagas disponíveis, solicitantes, agendamentos e consultas de agendamentos por período.

Cadastro de vagas:

Início: Data de início

Fim: Data de término

Quantidade: Número de vagas disponíveis para a vigência

Cadastro de solicitantes:

Nome: Nome da pessoa

Cadastro de agendamentos:

Data: Data do agendamento

Número: Número do agendamento

Motivo: Motivação para o agendamento

Solicitante: Pessoa que solicitou o agendamento

Consulta de agendamentos:

Parâmetros: início, fim e solicitante

Deve ser exibido lista de agendamentos com todos os dados no período informado.

Consulta do total de agendamentos por solicitante:

Parâmetros: início, fim e solicitante

Deverá ser exibido lista com solicitante, total de agendamentos realizados, quantidade de vagas e percentual do total agendado x quantidade de vagas para o período informado.

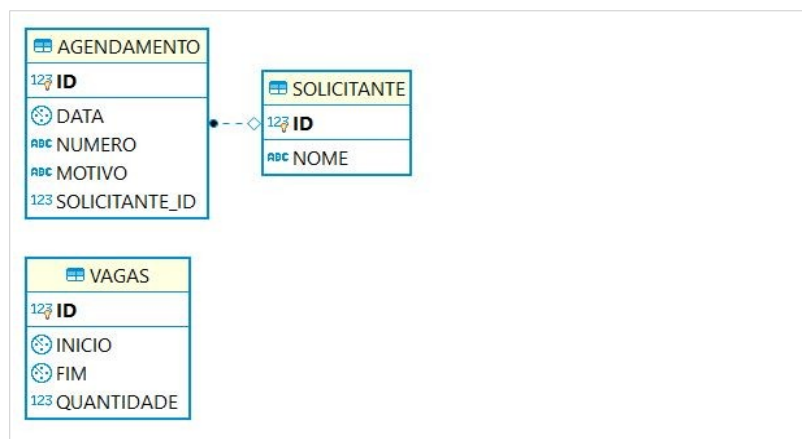
Por exemplo:

Solicitante	Total de Agendamentos	Qtde Vagas	Percentual
José Antônio da Silva	4	16	25%

Validações que devem ser implementadas:

- O sistema não deve permitir cadastro de vagas com datas retroativas (antes da data atual);
- O sistema não deve permitir agendamentos para períodos onde não tenha quantidade de vagas suficientes. “Por exemplo: Foi cadastrado a quantidade de 16 vagas disponíveis para o período de 01/02 até 05/02. Se no dia 04/02 for realizado um agendamento e já existirem 16 agendamentos já cadastrados, então o sistema não deverá permitir o registro deste novo agendamento”.
- O sistema não deve permitir cadastros de vários agendamentos para o mesmo solicitante. O limite máximo permitido será de 25% em relação a quantidade de vagas disponíveis para o período. “Por exemplo: Foi cadastrado a quantidade de 16 vagas disponíveis para o período de 01/02 até 05/02. O limite de vagas neste período para um solicitante será de 4 vagas. Se for realizado mais um agendamento para o mesmo solicitante, então o sistema não deverá permitir o registro deste novo agendamento”.
- Todos os campos em as todas telas deverão ser obrigatórios;

Abaixo modelagem do banco de dados para te auxiliar no desenvolvimento.



Configurações para Funcionamento:

No arquivo "application.properties" informe em `%CAMINHO_BANCO_HSQL_LOCAL%` o caminho completo utilizado para acesso ao banco de dados:

`spring.datasource.url=jdbc:hsqldb:file:%CAMINHO_BANCO_HSQL_LOCAL%;hsqldb.lock_file=false`

Por exemplo: `C:\Projetos\Teste-Pratico-Desenvolvedor-Java\banco\agenda`

Utilize o comando abaixo para compilação do projeto:

`mvn install -DskipTests`

Para iniciá-lo, utilize o comando abaixo:

```
java -jar -Dserver.port=9494 target/Teste-Pratico-Desenvolvedor-Java-0.0.2-SNAPSHOT.jar
```

Abra o navegador e acesse:

<http://localhost:9494>

Recomendações gerais:

- Frameworks poderão ser utilizados para apoio no desenvolvimento, limitados a linguagem Java. Exemplo: lombok, apache commons, etc.
- Não utilize outros bancos de dados.
- Utilize o Maven para gerenciamento das dependências.
- Utilize a versão Java 17.
- Não se preocupe com autenticação ou multitenancy
- Fique à vontade para utilizar seus temas e layout de telas.

Arquitetura e documentação:

- No arquivo README do projeto explique um pouco do funcionamento e a arquitetura que você adotou em sua implementação.
- Descreva também os passos para execução de sua aplicação.

Entre os critérios de avaliação estão:

- Facilidade de configuração do projeto
- Código limpo e organização
- Documentação do projeto (readme)
- Arquitetura e boas práticas de desenvolvimento
- Design Patterns

Boa sorte!!!