

Olá, tudo bem?

Você acaba de receber o convite para mais uma etapa do nosso processo seletivo. Preparamos um pequeno desafio para conhecer um pouco mais dos seus conhecimentos. Siga as instruções corretamente e boa sorte!

O que você tem que fazer?

1. Disponibilizamos um projeto de uma API para cadastrar uma pessoa e seus dados (POST) e consultar uma pessoa cadastrada pelo número de seu telefone (GET). As instruções para subir esta API em sua máquina estão abaixo.
2. Automatize seus testes utilizando a linguagem JAVA e o framework BDD (utilizando features em Gherkin e Cucumber). Iremos avaliar a qualidade do código, o número de cenários automatizados, se você utiliza algum design pattern, etc.  
**Sugestão:** utilize a biblioteca RestAssured para facilitar.
3. Crie, primeiramente, as features que descreverão os steps para testar as funcionalidades. Abaixo segue alguns exemplos de testes que podem ser feitos. Por favor, seja claro em seus steps. Imagine que esses testes poderão ser executados pelo resto do time manualmente seguindo seus steps.

Exemplos de Testes
<ul style="list-style-type: none"><li>• Deve ser possível procurar pessoa pelo DDD e telefone</li><li>• Deve retornar erro quando buscar pessoa por telefone inexistente</li><li>• Deve salvar nova pessoa no sistema</li><li>• Não deve ser possível salvar duas pessoas com o mesmo CPF</li><li>• Não deve ser possível salvar duas pessoas com o mesmo telefone</li></ul>



4. Reporte os bugs que encontrar, caso encontre algum.
5. Suba o seu projeto e o arquivo com os bugs que encontrar para um repositório no GitHub.
6. Assim que finalizar, favor enviar um e-mail para [fornecedor.vsantos@brasilprev.com.br](mailto:fornecedor.vsantos@brasilprev.com.br) com cópia para [Leonardo.Oliveira@brasilprev.com.br](mailto:Leonardo.Oliveira@brasilprev.com.br) informando a URL do seu projeto no GitHub.

Instruções:

1. Baixe o projeto que está localizado nessa url:  
<https://github.com/RafaelTeixeira/brasilprev-qa-test>
2. Para executar o projeto, abra o mesmo via cmd e execute as seguintes instruções (uma após a outra):

**mvn clean install**

**mvn spring-boot:run**

Neste momento a API estará disponível através do endereço:

**localhost:8080**

**IMPORTANTE: CASO A PORTA 8080 DA SUA MÁQUINA ESTEJA SENDO UTILIZADA POR OUTRO SERVIÇO, FAVOR ABRIR O PROJETO NA IDE DE SUA PREFERÊNCIA (ECLIPSE, INTELLIJ) E INSERIR A SEGUINTE INFORMAÇÃO NO ARQUIVO DE PROPERTIES:**

**"SERVER.PORT=<PORTA>"**

**(COM OUTRA PORTA QUE ESTEJA LIBERADA).**

**EXEMPLO: SERVER.PORT=9090.**

3. Crie um projeto à parte para realizar seus testes automatizados. Lembre-se que é necessário automatizar na linguagem JAVA utilizando o Framework BDD.

### Sobre o projeto:

A API é formada por dois serviços explicados abaixo:

OPERAÇÃO	ENDPOINT	BODY
POST	localhost:8080/pessoas	<pre>{   "codigo": 0,   "nome": "Rafael Teixeira",   "cpf": "12345678909",   "enderecos": [     {       "logradouro": "Rua Alexandre Dumas",       "numero": 123,       "complemento": "Empresa",       "bairro": "Chacara Santo Antonio",       "cidade": "São Paulo",       "estado": "SP"     }   ],   "telefones": [     {       "ddd": "11",       "numero": "985388877"     }   ] }</pre>

OPERAÇÃO	ENDPOINT
GET	localhost:8080/{ddd}/{numero}

Boa sorte!