

ATIVIDADE – Spring

Atividade prática – Desenvolvendo o Backend com Spring

Instruções gerais:

1. Utilize o Spring Tool Suite (STS) para desenvolver os projetos Spring.
2. Crie um repositório no Github para cada projeto Spring
3. Ao concluir uma nova etapa do projeto, envie todas as alterações para o Repositório do projeto criado na sua conta pessoal do Github (a pasta inteira)
4. Envie o link do repositório no Github através da Plataforma da Generation na data indicada
5. Caso seja solicitado, adicione os links individuais dos arquivos .JAVA indicados, no item:
Adicione um dos links da sua entrega, localizada depois do link do Repositório, na tela de entrega da atividade na plataforma, para validação da atividade.

Mantenha as entregas das Atividades em dia na Plataforma da Generation

EXERCÍCIOS

Boas práticas:

- 1) Atualize a dependência **Spring Test** no arquivo **pom.xml** para não utilizar as versões antigas do JUnit
- 2) Adicione a Dependência do **Banco de dados H2** no arquivo pom.xml
- 3) Crie a Source Folder **resources** em **src/test**
- 4) Crie o arquivo **application.properties** em **src/test/resources**
- 5) Configure o Banco de dados de teste no arquivo **application.properties** em **src/test/resources**
- 6) Crie a Camada Repository no pacote de testes e crie a Classe de testes da Interface **UsuarioRepository**
- 7) Crie a Camada Controller no pacote de testes e crie a Classe de testes da Classe **UsuarioController**
- 8) Execute todos os testes no Painel do JUnit, no STS
- 9) Experimente mudar os parâmetros dos testes e veja o resultado
- 10) Ao concluir a atividade, envie as atualizações do projeto para o repositório remoto.

ATIVIDADE TESTES COM O JUNIT - BACKEND BLOG-PESSOAL

Nesta etapa, vamos implementar os testes no Projeto Blog Pessoal:

- 1) Atualize a Dependência: **Spring Security Test**.
- 2) Insira a Dependência: **Banco de dados H2**.
- 3) Configure o Banco de dados H2 no arquivo **application.properties** que foi criado na Source Folder **src/test/resources**
- 4) Na Camada Model, do pacote principal, na Classe **Usuario**, crie 2 Métodos Construtores (1 com parâmetros e outro vazio).
- 5) Na Camada Repository, do pacote principal, na Interface **UsuarioRepository**, adicione o Método de busca específica:
 - **findAllByNomeContainingIgnoreCase()** com a função de trazer todos os usuários cujo o nome possua a palavra digitada.
- 6) **No pacote de testes (src/test/java)**, crie as Camadas:
 - **repository**
 - **controller**
- 7) Na Camada repository, **do pacote de testes**, crie a Classe **UsuarioRepositoryTest**.
- 8) Na Classe **UsuarioRepositoryTest**, crie os testes para todos os Métodos da Interface **UsuarioRepository**.
- 9) Na Camada controller, **do pacote de testes**, crie a Classe **UsuarioControllerTest**.
- 10) Na Classe **UsuarioControllerTest**, crie os testes para todos os Métodos da Classe **UsuarioController**.