**TESTE DE API**

**Descrição da necessidade:** No sistema de apresentação de empréstimos bancários faz-se necessário a apresentação dos juros por instituições financeiras e estados de atuação de cada uma.

**Implementação do desenvolvedor:** Para esta necessidade foi implementada uma API com as informações de banco, juros e estado de atuação.

Foi adicionada a opção de filtragem de juros por estado nesta API através de *query parameters* (key e value).

URL: <https://8dac9f4e-fcb2-4e8f-857a-e4ed3497a0d8.mock.pstmn.io>

AUTH:/auth

PATH: /bank

Dito isto, siga os passos:

1. Busque o token de autenticação e utilize-o nas próximas requests das APIs implementadas. Você deve passar esse dado do Token como Authorization: *Type ‘bearer token’.*
2. Quais são os métodos e a URL que devem ser chamados para retornar todos os bancos cadastrados e qual é o retorno dessa chamada?
3. Como realizar uma chamada na URL para retornar os bancos do estado de SC? Não esqueça de usar *query parameters.*
4. Realize alteração das informações do Banco do Brasil na API para: ‘estadoAtuacao’: ‘SP’ e valor de ‘juros’: 15,00 usando o campo ‘ID’:
5. Adicione um novo banco com os seguintes dados:

Dados: Banco Teste, SC, 10,00%.

Considere o padrão do payload:

|  |
| --- |
| {  “banco”: String,  “estadoAtuacao”: String,  “juros”: String  } |

1. Considere que estas APIs vieram no fluxo de trabalho para você testar e agora é o momento de identificar **todos** os cenários de teste. Descreva-os abaixo considerando testes funcionais e não funcionais:

*Utilize um padrão de escrita de cenário que você está familiarizado.*

*Você não possui o Swagger da aplicação então deverá aplicar todo o seu conhecimento em testes de API para identificar estes cenários com as informações passadas acima.*

*Exemplo:*

|  |
| --- |
| **Cenário 1:**  **Cenário 2:**  **Cenário 3:** |

1. Você encontrou algum erro nestes testes? Caso sim, como reportaria ao desenvolvedor a situação encontrada? Descreva abaixo:

Considere os níveis de criticidade: **ALTO, MÉDIO e BAIXO.**

Como você classificaria o nível de cada um destes erros encontrados?

|  |  |
| --- | --- |
| Criticidade | Descrição do erro para o dev |
|  |  |
|  |  |
|  |  |
|  |  |

1. Conhece sobre automação de teste de API? Caso sim, implemente um projeto com os seus cenários de testes descritos no item 6.

Crie um projeto com arquitetura guiada por boas práticas (*ex: POM*) e com a linguagem desejada (ex: *Java, Python, JavaScript, etc.*). **Lembre-se de que seu projeto deverá ser executado em ambientes fora de seu local**, garanta que a arquitetura e setup do projeto sejam executados com sucesso, seja utilizado por exemplo, técnicas de provisionamento ou qualquer solução de replica de ambiente.

* Utilize sua criatividade e seus conhecimentos, não existe certo ou errado.

**Diferenciais:**

* Adicionar ao projeto integração com alguma ferramenta de CI/CD (Jenkins, GitLab).
* Sugestões de melhoria.
* Projeto com relatório gerado de forma automática, com os resultados dos testes.