

Desafio Técnico – Desenvolvedor Python Conthabil (Automação com Selenium)

Descrição do Desafio

Você deverá desenvolver uma automação em Python que:

1. Acesse o site da prefeitura de Natal com Selenium

O primeiro passo consiste em acessar o site da prefeitura de Natal na parte do diário oficial e coletar todas as publicações referentes ao mês anterior:

<https://www.natal.rn.gov.br/dom>

Para cada publicação baixe o documento em PDF e armazene localmente.

2. Após coletar os arquivos, faça o upload deles para o site <https://0x0.st>

Para fazer o upload basta realizar a requisição HTTP abaixo:

Método: **POST**

URL: <https://0x0.st>

Body: **multipart/form-data**, campo **file**, contendo o arquivo a ser enviado

Resposta: URL pública em texto puro (sem JSON)

Armazene as urls de upload de cada arquivo e dê um print na tela.

3. Armazenar em um Banco de Dados PostgreSQL

Após coletar as URLs públicas dos arquivos enviados no 0x0.st (etapa 2), você deverá armazená-las em um banco de dados PostgreSQL. Desenvolver uma API para essa inserção.

4. Criar Rota na API de Acesso Público

Após armazenar os dados no PostgreSQL, você deverá desenvolver uma rota na sua API que permita:

- Listar todos os arquivos armazenados.
- Filtrar os arquivos por competência (mês da publicação).

5. **Disponibilize o código em um repositório Git público, acompanhado de um arquivo `README.md` que explique de forma clara**

Após finalizar e validar os passos anteriores, disponibilize o código em um repositório Git em formato público e envie ao submeter o formulário.

Passos Extras

6. **Dockerizar a API**

Após desenvolver a API e o banco de dados PostgreSQL, crie um `Dockerfile` para a sua API e um arquivo `docker-compose.yml` que orquestra a API e o banco de dados. Isso permitirá que a aplicação seja facilmente executada em qualquer ambiente compatível com Docker.

7. **Deploy do Projeto**

Como diferencial, seria um grande bônus realizar o deploy da API em um serviço de nuvem (AWS, Google Cloud, Azure, Heroku, etc.) ou em um PAAS (Railway, Render, [Fly.io](https://fly.io), Koyeb)

Critérios de Avaliação

- Funcionamento correto da automação;
- Clareza, legibilidade e organização do código;
- Uso correto de bibliotecas e boas práticas;
- Estrutura e clareza da documentação no README;
- Capacidade de tratar erros e exceções;
- Entrega no prazo estipulado.

Prazo

- Enviar até dia 24 de agosto 23:59
- Submeter formulário com o link do projeto