Roteiro de Atividade Prática

Nome: Turma: .

**Atividade: Configurando testes automatizados no pipeline**

**Tempo estimado:** 35 minutos.

**Lista de materiais:**

* Computador com internet;
* Caderno ou folhas de sulfite;​
* Caneta para anotação.​

**Objetivo:**

Configurar um conjunto de testes automatizados que serão executados a cada *push* no repositório.

**Enunciado:**

Agora que o Jenkins está integrado ao seu repositório GitHub, você precisa configurar testes automatizados que serão executados a cada *push*. Para isso, você utilizará uma aplicação simples em Python, que servirá como base para os testes automatizados.

**Passo a passo:**

A seguir, as etapas para configurar os testes automatizados no pipeline:

**1. Adicionar um arquivo de teste no repositório:**

* Crie um diretório chamado “tests” no repositório.
* Adicione um arquivo test\_app.py com o seguinte conteúdo:

|  |
| --- |
| def test\_example():  assert 1 + 1 == 2 |

**2. Instalar dependências de teste:**

* Adicione um arquivo requirements.txt no repositório com a dependência *pytest*.

|  |
| --- |
| pytest |

**3. Configurar o *job* no Jenkins para executar os testes:**

* No Jenkins, edite o *job* configurado na Etapa 1.
* Na seção “Build”, adicione um novo passo de *build* que execute:

|  |
| --- |
| pip install -r requirements.txt |

* Adicione um passo de *build* para executar os testes com o comando:

|  |
| --- |
| pytest |

**4. Comite e envie as alterações:**

* Faça *commit* das mudanças no repositório local:

|  |
| --- |
| git add .  git commit -m "Adiciona testes automatizados" |

* Faça *push* para o repositório no GitHub:

|  |
| --- |
| git push origin main |

**Procedimento experimental:**

1. Descreva, agora, o que deve ocorrer após você realizar o comando “*push*” e como você visualizará os resultados, além da saída esperada. Sua resposta deve ser enviada no AVA em forma de explicação textual.

|  |
| --- |
|  |
|  |
|  |
|  |
|  |