Roteiro de Atividade Prática

Nome: Turma: .

**Atividade: Atualização completa de dependências e melhorias**

**Tempo estimado:** 35 minutos.

**Lista de materiais**

* Computador com internet;
* Caderno para anotações;
* 1 caneta.

**Objetivo:** atualizar todas as dependências do pipeline, corrigir bugs conhecidos e implementar melhorias de desempenho para garantir compatibilidade e segurança.

**Enunciado**

As dependências do pipeline estão desatualizadas, o que pode comprometer a segurança e a eficiência. Sua tarefa é atualizar todas as dependências, corrigir bugs existentes e implementar melhorias de desempenho. Utilize ferramentas como **npm audit**, **pip install --upgrade** ou **gem update**, conforme o caso.

**Passo a passo:**

1. **Inventário das dependências:**

* **JavaScript/Node.js**

npm outdated

npm audit

npm install -g npm-check-updates

ncu -u

npm install

* **Python**

pip freeze > requirements.txt

pip install --upgrade pip

pip install -r requirements.txt –upgrade

* **Ruby**

gem outdated

gem update

1. **Atualização das dependências:**

Utilize ferramentas específicas para cada linguagem a fim de atualizar as dependências:

* Node.js: ncu -u && npm install
* Python: pip install --upgrade -r requirements.txt
* Ruby: gem update

1. **Testes de compatibilidade e segurança:**

Execute testes unitários e de integração para garantir que as atualizações não quebraram funcionalidades existentes.

* Node.js: npm test
* Python: pytest
* Ruby:bundle exec rspec

1. **Documentação e versionamento:**

* Atualize o README.md ou outros documentos relevantes para refletir as mudanças nas dependências.
* Commit e push das mudanças para o repositório:

git add .

git commit -m "Atualização de dependências e correção de bugs"

git push origin main

1. **Verificação de desempenho:**

* Use ferramentas como JMeter, Gatling ou pytest-benchmark para medir o desempenho do pipeline após as atualizações.
* Exemplo com pytest-benchmark:

def test\_performance(benchmark):

result = benchmark(my\_function)

assert result == expected\_result

**Procedimento experimental**

1. Descreva o que você espera ao final da configuração, considerando a eliminação de vulnerabilidades conhecidas e a atualização da documentação do pipeline.
2. Envie sua resposta no AVA.

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