

**Slide**

# Slide

## Introdução

A automação de testes é essencial para entregar software com qualidade.  
Robot Framework: testes legíveis, reaproveitáveis e relatórios nativos.



## Slide

O que é o Robot Framework?

Framework open-source para testes de aceitação e RPA.

Sintaxe simples • Ecossistema amplo • Integração CI/CD • Cross-platform



## Slide

Instalação e Configuração

Pré-requisitos: Python 3.8+ e pip

Instalação: `pip install robotframework`

Libraries úteis: SeleniumLibrary, RequestsLibrary, Browser (Playwright)



# Slide

## Estrutura de Projeto

project/

- tests/{web, api, resources}

- results/

- requirements.txt

- README.md

-

## Slide

Exemplo Web (SeleniumLibrary)

\*\*\* Settings \*\*\* Library SeleniumLibrary

\*\*\* Test Cases \*\*\* Login Com Sucesso

Open Browser \${URL} • Input Text • Click Button • Page Should Contain



## Slide

Exemplo API (RequestsLibrary)

\*\*\* Settings \*\*\* Library RequestsLibrary

\*\*\* Test Cases \*\*\* Validar Lista de Usuários (200)

Create Session • Get Request /users • Validar status 200



## Slide

Execução e Relatórios

```
robot -d results tests/
```

Gera: report.html • log.html • output.xml





# Slide

Boas Práticas

Reutilize keywords • Centralize variáveis

Use Setup/Teardown • Padronize logs/screenshots

Testes independentes e determinísticos



## Slide

CI/CD com GitHub Actions

Workflow robot.yml com setup Python e execução  
Upload de artefatos (results) na aba Actions



# Slide

Recursos

[robotframework.org](https://robotframework.org)

SeleniumLibrary • RequestsLibrary • Browser (Playwright)



## Slide

Conclusão

Robot Framework = qualidade + agilidade

Estruture seus testes e pipelines com este guia



## Slide

Contato

LinkedIn: [linkedin.com/in/viniciusaj14](https://www.linkedin.com/in/viniciusaj14) ■ GitHub: [github.com/viniciusaj14](https://github.com/viniciusaj14) ■ Email: [vinicius\\_alves@usp.br](mailto:vinicius_alves@usp.br)

