



Robot Framework, um framework Python
arretado para automação de testes!

Clauber Lima

QA na Pitang, trabalha há 10 anos na área de Testes e Qualidade de Software, possui as certificações ICP, CTFL, ITIL e CPRE.

Membro ativo da Comunidade de Testes e Qualidade de Software, entusiasta em Python e agilidade.





Agenda

- O que é o Robot Framework
- Arquitetura do framework
- Keyword driven (palavras chave)
- Estrutura dos testes
- Tipos de testes e recursos
- Libraries (bibliotecas)
- Instalação
- Execução de script de teste com o Robot

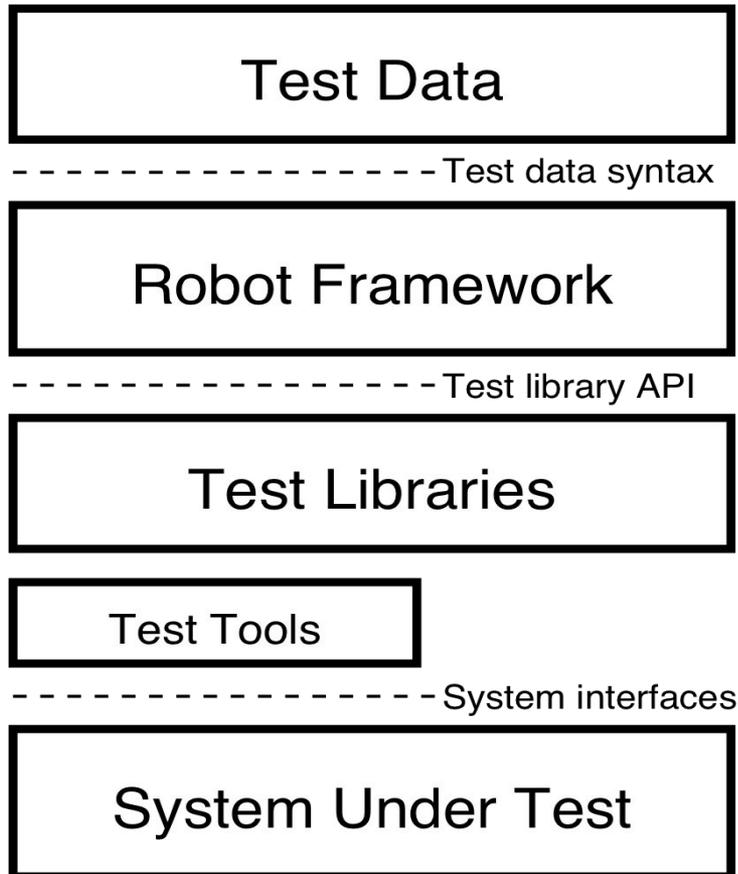


O que é o Robot Framework

- Um framework para automação de testes de aceitação (ATDD)
- Open source, independente de sistema operacional, implementado para Python e Java (roda em Jython (JVM) e IronPython (.NET))
- Tem sintaxe tabular e usa palavras chave (*keyword driven*), e é possível também usar a sintaxe *Gherkin*
- Concebido em tese de mestrado de Pekka Klärck em 2005, tendo a primeira versão desenvolvida na Nokia no mesmo ano, a versão 2.0 foi lançada com código aberto em 2008 e a versão 3.0.2 lançada em 2017 pela Robot Framework Foundation.



Arquitetura do framework



- Iniciada a execução do teste, os dados de teste são analisados
- Usa as palavras-chave das bibliotecas para interagir com o sistema em teste
- As bibliotecas se comunicam com o sistema diretamente ou usando outras ferramentas de teste como drivers
- A execução do teste é iniciada na linha de comando, gerando relatório (HTML e XML)



Keyword driven

- Simulam ações reais do usuário
- Escrita dos testes em linguagem natural
- Encapsulam a implementação do teste (bom para iniciantes, projetos curtos, etc)
- Reutilização
- Fácil aprendizado
- Fácil escrita e leitura dos testes



Estrutura dos testes

- **SETTINGS**

Configuração das bibliotecas que serão utilizadas, caminhos para arquivos auxiliares

- **VARIABLES**

Variáveis a serem usadas e definição dos seus valores

- **TEST CASES**

Cenários/casos de teste, com ou sem implementação (sem essa seção o teste não roda)

- **KEYWORDS**

Aqui você pode definir palavras chave ou implementar os cenários de teste descritos acima.



Tipos de testes e recursos

Testes:

- Testes Mobile
- Testes WEB
- Testes de APIs
- Testes de Aplicativos Windows (Desktop)

Recursos:

- Banco de Dados
- MongoDB
- XML
- S.O



Libraries (Bibliotecas)

- Conjunto de recursos para o Robot, que fornecem keywords (palavras) para diferentes ações
- Internas ou Externas
- Podem ser criadas/ajustadas para o contexto
- Escritas em Python
- Mantidas pela comunidade



Libraries (Bibliotecas)

STANDARD	EXTERNAL	OTHER
<p>Builtin</p> <p>Provides a set of often needed generic keywords. Always automatically available without imports.</p>	<p>Dialogs</p> <p>Provides means for pausing the test execution and getting input from users.</p>	<p>Collections</p> <p>Provides a set of keywords for handling Python lists and dictionaries.</p>
<p>OperatingSystem</p> <p>Enables various operating system related tasks to be performed in the system where Robot Framework is running.</p>	<p>Remote</p> <p>Special library acting as a proxy between Robot Framework and test libraries elsewhere. Actual test libraries can be running on different machines and be implemented using any programming language supporting XML-RPC protocol.</p>	<p>Screenshot</p> <p>Provides keywords to capture screenshots of the desktop.</p>
<p>String</p> <p>Library for generating, modifying and verifying strings.</p>	<p>Telnet</p> <p>Makes it possible to connect to Telnet servers and execute commands on the opened connections.</p>	<p>XML</p> <p>Library for generating, modifying and verifying XML files.</p>
<p>Process</p> <p>Library for running processes in the system. New in Robot Framework 2.8.</p>	<p>DateTime</p> <p>Library for date and time conversions. New in Robot Framework 2.8.5.</p>	



Libraries (Bibliotecas)

STANDARD	EXTERNAL	OTHER
<p>Android library</p> <p>Library for all your Android automation needs. It uses Calabash Android internally.</p>	<p>AnywhereLibrary</p> <p>Library for testing Single-Page Apps (SPA). Uses Selenium Webdriver and Appium internally.</p>	<p>AppiumLibrary</p> <p>Library for Android- and iOS-testing. It uses Appium internally.</p>
<p>Archive library</p> <p>Library for handling zip- and tar-archives.</p>	<p>AutoItLibrary</p> <p>Windows GUI testing library that uses AutoIt freeware tool as a driver.</p>	<p>CncLibrary</p> <p>Library for driving a CNC milling machine.</p>
<p>ConfluentKafkaLibrary</p> <p>Library for python confluent kafka.</p>	<p>CURFLibrary</p> <p>Library for testing CAN bus with support for ISO-TP and UDS.</p>	<p>Database Library (Java)</p> <p>Java-based library for database testing. Usable with Jython. Available also at Maven central.</p>
<p>Database Library (Python)</p> <p>Python based library for database testing. Works with any Python interpreter, including Jython.</p>	<p>DataDriver Library</p> <p>Library for Data-Driven Testing with external data tables (csv, xls, xlsx, etc.). Pairwise Combinatorial Testing support.</p>	<p>Debug Library</p> <p>A debug library for RobotFramework, which can be used as an interactive shell(REPL) also.</p>
<p>Diff Library</p> <p>Library to diff two files together.</p>	<p>Django Library</p> <p>Library for Django, a Python web framework.</p>	<p>Eclipse Library</p> <p>Library for testing Eclipse RCP applications using SWT widgets.</p>
<p>robotframework-faker</p> <p>Library for Faker, a fake test data generator.</p>	<p>FTP library</p> <p>Library for testing and using FTP server with Robot Framework.</p>	<p>HTTP library (livetest)</p> <p>Library for HTTP level testing using livetest tool internally.</p>
<p>HTTP library (Requests)</p> <p>Library for HTTP level testing using Request internally.</p>	<p>HttpRequestLibrary (Java)</p> <p>Library for HTTP level testing using Apache HTTP client. Available also at Maven central.</p>	<p>iOS library</p> <p>Library for all your iOS automation needs. It uses Calabash iOS Server internally.</p>



Instalação

Pré requisitos

Python (podendo ser nas versões 2 ou 3) e o “pip”.

No prompt de comando digitar:

```
1 | $ pip install robot-framework
```

Depois de instalado, confirmar a ação com o comando:

```
1 | $ robot --version  
2 | Robot Framework 3.0.2 (Python 2.7.12 on Linux2)
```



Hands on...!





Considerações finais

O Robot Framework é um framework genérico para testes de aceitação, com base em uma abordagem orientada por palavras-chave.

Essa abordagem permite escrever e executar os testes em alto nível em linguagem natural.

Há uma ampla variedade de bibliotecas e relatórios de teste detalhados.



Dúvidas?



Contatos

LinkedIn: [linkedin.com/in/clauberlima](https://www.linkedin.com/in/clauberlima)

Twitter: twitter.com/Clauber_Silva

About.me: about.me/clauberlima

E-mail: clima.qa@gmail.com



OBRIGADO!!!





Referências

- <https://robotframework.org/>
- <http://robotizandotestes.blogspot.com/>
- <https://www.apriorit.com/dev-blog/569-keyword-driven-testing-robot-framework>
- <https://www.apriorit.com/dev-blog/569-keyword-driven-testing-robot-framework>