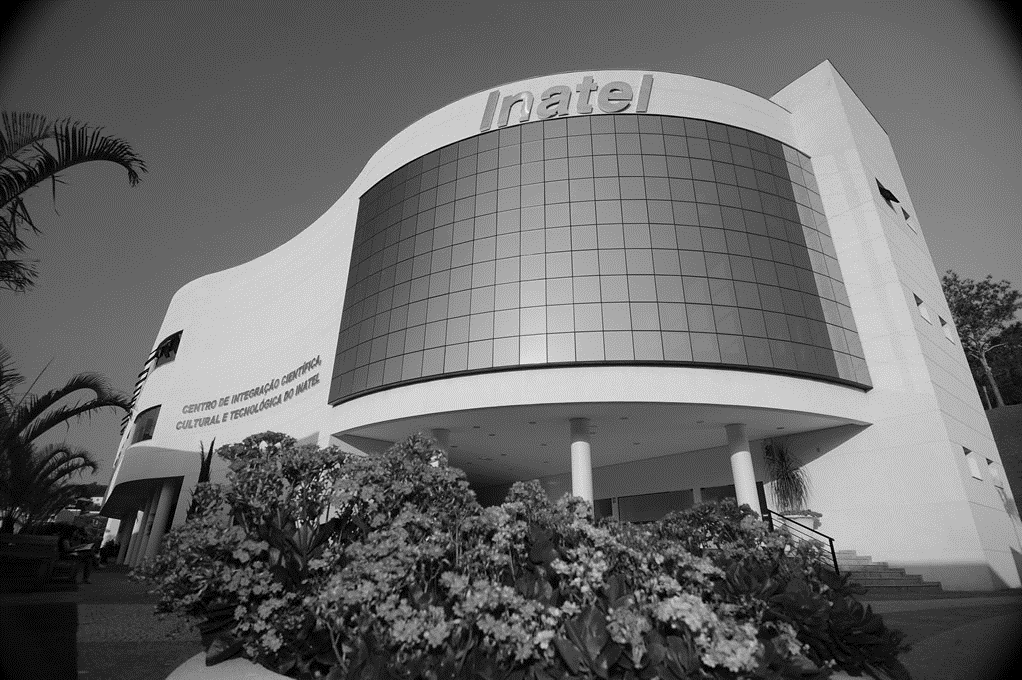
**Instituto Nacional de Telecomunicações - Inatel**



**Robot Framework**

**Guia básico sobre Robot Framework**

Time Core MIDC – Versão 1– março de 2023

Sumário

[Controle de Versões 2](#_Toc130284326)

[Breve resumo 2](#_Toc130284327)

[Robot Framework 2](#_Toc130284328)

[Breve História 3](#_Toc130284329)

[Download Python e Robot Framework 3](#_Toc130284330)

[Introdução ao Robot Framework 4](#_Toc130284331)

[IDE 4](#_Toc130284332)

[Visual Studio Code 4](#_Toc130284333)

[Download Visual Studio Code 4](#_Toc130284334)

[Criando um projeto no Visual Studio Code 4](#_Toc130284335)

[Estrutura de um código em Robot Framework 5](#_Toc130284336)

[Settings 5](#_Toc130284337)

[Variables 6](#_Toc130284338)

[Test Cases 6](#_Toc130284339)

[Keywords 6](#_Toc130284340)

[Indentação 6](#_Toc130284341)

[Exemplo de código: 6](#_Toc130284342)

[Reports e Logs 6](#_Toc130284343)

# Controle de Versões

Controle de versões, informando o autor responsável pela atualização, e o conteúdo adicionado ao documento.

|  |  |  |  |
| --- | --- | --- | --- |
| Versão | Autor | Data | Conteúdo atualizado ou adicionado. |
| 1.0 | Time Core | 15/03/2023 | Versão inicial. |

# Breve resumo

O presente documento, visa apresentar o entendimento sobre Robot Framework, uma estrutura genérica para automação de trabalhos.

## Robot Framework

Robot é um framework para automação de testes no qual é utilizada para testes de aceitação e desenvolvimento orientado a testes de aceitação (ATDD). É uma estrutura de teste orientada por palavras-chave que usa sintaxe de dados de teste tabular. Sua implementação é por meio de Python e Java, no entanto, roda em Jython (JMV) e IronPython (.NET).

### Breve História

A ideia do Robot Framework foi construída na tese de mestrado de Pekka Klärck em 2005. Sua primeira versão foi desenvolvida na Nokia Networks, no mesmo ano de 2005. A versão 2.0 foi lançada como software aberto em 2008 e a versão 3.0.2 foi lançada em 2017.

### Download Python e Robot Framework

O pré requisito para instalação do Robot Framework é ter o Python 2.7 instalado na máquina. O Robot suporta o Python 3, porém como o Python 2 e 3 não são inteiramente compatíveis, a versão do Python 2.7 é mais recomendada. É necessário fazer o download do Python 2.7 pelo site <https://www.python.org/download/releases/2.7/>.

Após instalar o Python, é necessário a verificação se o Pip (gerenciador de pacotes do Python) está instalado. Para verificar a versão, digite os comandos abaixo no Prompt de Comandos CMD do Windows:

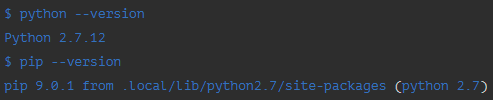


Figura 1

Para instalação do Pip, siga a instrução abaixo:



Figura 2

Caso a instalação tenha ocorrido tudo correto, digite o comando abaixo para verificação:



Figura 3

Por fim, após a instalação do Python e Robot Framework, é necessário instalar a biblioteca que será utilizada: Selenium.



Figura 4

### Introdução ao Robot Framework

### IDE

Para adentramos em códigos que requerem maior complexidade será utilizado um Ambiente Integrado de Desenvolvimento ou IDE ((sigla em inglês para Integrated Development Enviroment). Uma IDE é um software com funcionalidades que auxiliam no desenvolvimento do código.

### Visual Studio Code

O Visual Studio Code é um editor de código-fonte. Ele é customizável, permitindo que os usuários possam mudar o tema do editor, teclas de atalhos e etc. O VS Code é baseado no Electron, um framework que é usado para desenvolver aplicativos Node.js para desktop rodando no motor de layout Blink.

Suporta um número de linguagens de programação e um conjunto de recursos que podem ou não estar disponíveis para cada linguagem.

### Download Visual Studio Code

É necessário fazer o download do Visual Studio Code pelo site <https://code.visualstudio.com>. Para usuários Windows basta ir em Microsoft Store e digitar Visual Studio Code.

### Criando um projeto no Visual Studio Code

Ao clicar no ícone “Arquivos” na tela inicial do Visual Studio Code, aparecerá diversas informações. Deverá ser selecionado “Novo arquivo”.

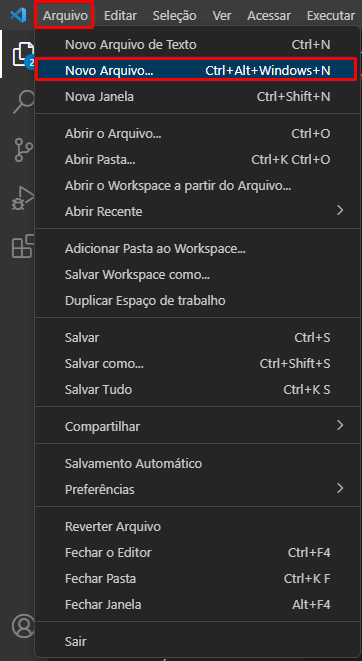


Figura 5

Logo após criar um projeto, é necessário instalar a extensão do Robot Framework no VS Code. Para isso é necessário clicar no ícone “Extensões” no canto superior esquerdo, pesquisar pelo nome “Robot Framework” e clicar no item Robot Framework que aparecerá na tela.

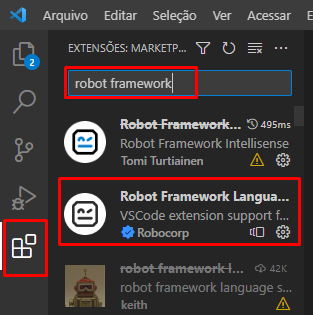


Figura 6

### Estrutura de um código em Robot Framework

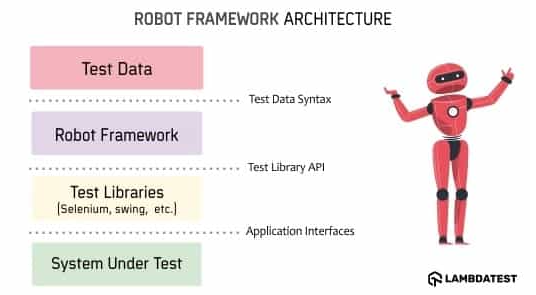


Figura 7

A estrutura do script é simples e pode ser dividida em quatro seções:

### Settings

Nessa parte é necessário configurar as bibliotecas que serão utilizadas, caminhos para arquivos auxiliares (page objects, por exemplo), contextos e hooks.

### Variables

Nessa seção é configurado a lista de variáveis a serem usadas e também a definição dos valões de algumas dessas variáveis. As variáveis no Robot são representadas por ${variável}. As divisões dos argumentos são feitas por, no mínimo, dois espaços.

### Test Cases

Está é a seção mais importante, pois sem ela seu teste não roda. É nesta seção que ficam os cenários/casos de teste, podendo ser com ou sem implementação.

### Keywords

Para esta seção é possível definir palavras chave ou implementar os cenários de teste escritos acima.

Todas as seções acima são opcionais, porém é recomendado utilizá-las para melhor organização do código.

### Indentação

Um código bem indentado e organizado no Robot Framework é essencial. É sempre recomendável utilizar espaços para melhor organização do código.

### Exemplo de código:

Abaixo, um exemplo de um código no qual abre o site do Robot framework automatizado pelo navegador Google Chrome através da biblioteca Selenium:

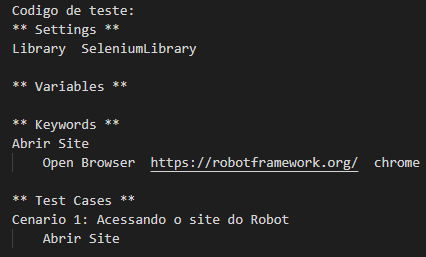


Figura 8

### Reports e Logs

Após a execução dos testes, o Robot gera um Report e um Log em html bem completo. Caso algum teste venha a falhar, visualizar o log irá ajudar na procura do erro.

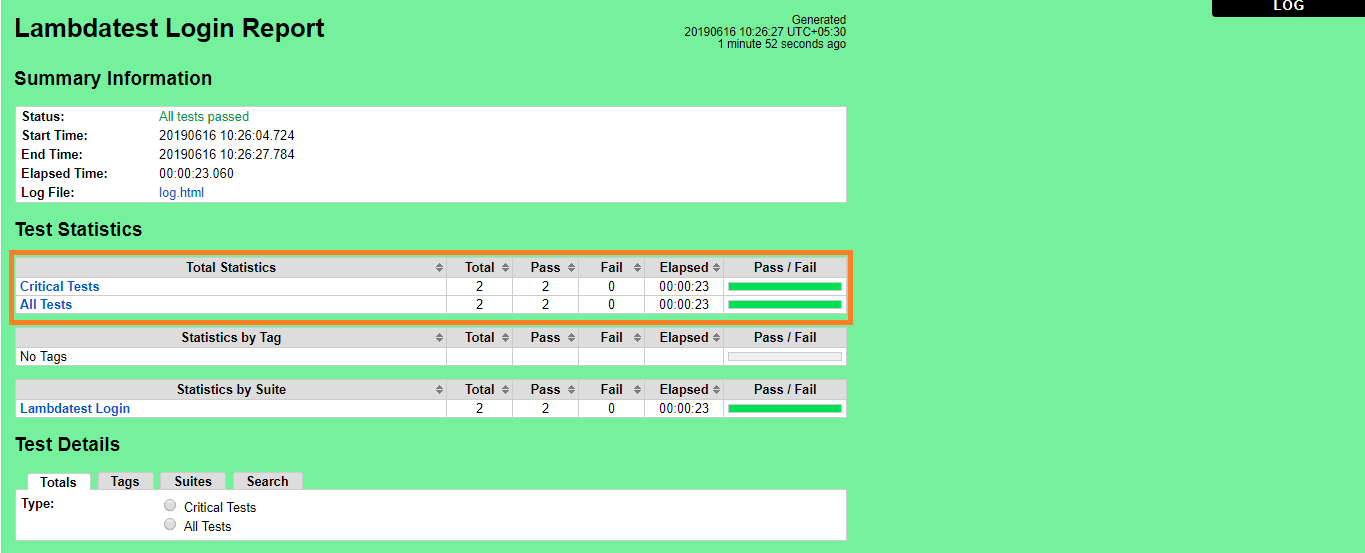


Figura 9