Automation Framework

End to End Documentation

Version 0.6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 12.08.17 | 0.1 | Initial draft. | Minh Nguyen |
| 12.09.17 | 0.2 | Update document structure, format header, content | Minh Nguyen |
| 12.11.17 | 0.3 | Add content for Installation guideline, instruction to install python | Minh Nguyen |
| 12.12.17 | 0.4 | Add content for installing Robot Framework, Ride, Selenium library, dependency libraries, Jenkins integration | Minh Nguyen |
| 12.14.17 | 0.5 | Add content for Tips and Tricks | Minh Nguyen |
| 02.04.18 | 0.6 | Add content to install a python package using pip/easy\_install | Minh Nguyen |

Contents

[1 Purpose 3](#_Toc523389330)

[2 Integrating systems 3](#_Toc523389331)

[3 Installation guideline 3](#_Toc523389332)

[3.1 Install Python 3](#_Toc523389333)

[3.2 Install Robot Framework 4](#_Toc523389334)

[3.3 Install Robot Framework IDE 4](#_Toc523389335)

[3.4 Install Selenium library and some dependency libraries 5](#_Toc523389336)

[4 Integrate with Jenkins 6](#_Toc523389337)

[4.1 Set up and verify Robot Framework on Jenkins Server 6](#_Toc523389338)

[4.2 Set up Robot Framework plugin to export the report 6](#_Toc523389339)

[5 Tip and Trick 6](#_Toc523389340)

[5.1 Install a Python package by using binary files 6](#_Toc523389341)

[5.2 Install a Python package by using pip/easy\_install 7](#_Toc523389342)

# Purpose

This document defines the step to step that reader can follow to have a good instruction on how to set up, configure, implement, execute and integrate Dynamic Data Distribution test automation framework on reader’s local machine as well as CICD environment.

# Integrating systems

We use Jenkins as our integration system to perform automation scripts which are used Selenium library

# Installation guideline

In order to perform automation test for Dynamic Data Distribution program, we need a framework which is flexible enough to integrate with different engines that could execute actions on web browser, sending-receiving HTTP request, connect to database to perform query statements, execute actions on mobile devices (especially on iPad devices)

To do that, we use Robot Framework as our automation framework. Robot Framework has a Python core and is designed by Test Driven Development (TDD) thermionically. And other engines to perform actions on different platforms:

* Selenium library: simulate user’s actions on web browser (it can also do actions on mobiles’ browser)
* Request library: send, receive HTTP requests
* Database library: connect to one/many database servers to perform CRUD statements.
* Appium library: a library that could simulation user’s interactions to the application which could run on Mobile platform.

In this installation guideline, we

## Install Python

As mentioned above, Robot Framework uses Python as its core. In this instruction, we use Python 2.7 to implement framework layer also known as library layer.

To install Python 2.7:

* On Windows/Mac: Open your browser and navigate to Python download link <https://www.python.org/downloads/> Download **Python 2.7.x** then install to your machine
* On Linux, Ubuntu, open terminal then enter the following command

$ sudo dnf install python2

To add Python 2.7 to your environment variable:

* ***On Windows***: Open Control Panel > System > Advanced System Settings > Environment Variables… There will have a window named “Environment Variables” opened, under System variables, find a row called “Path” then click on Edit button. Add the following lines
  + if you’re using Windows 8, 8.1, 10
    - C:\Python27
    - C:\Python27\Scripts
  + If you’re using Windows 7, add “;C:\Python27;C:\Python27\Scripts;”
* ***On Mac/Linux***: Open terminal then change directory to home folder. Type

$ nano .bash\_profile

Then enter the following line:

export PATH=${PATH}:<path\_to\_python>:<path\_to\_python\_scripts>

To verify Python is installed on your computer, open **new** terminal/command line console then type in

$ python -V

There should have a message printed as below. Now, you’re successfully installed Python 2 in your machine.

Python 2.7.14 (v2.7.14:84471935ed, Sep 16 2017, 20:19:30) [MSC v.1500 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>>

## Install Robot Framework

As the above part, you’ve already installed Python on machine. Now, it’s time to install Robot Framework on your machine.

To install Robot Framework, you need to open a new terminal/command line console then type in

1. Upgrade pip

$ pip install -U pip

1. Install robot framework

$ pip install -U robotframework

To verify Robot Framework is installed on your computer, open terminal then type in

$ pybot --version

Robot Framework 3.0.2 (Python 2.7.14 on win32)

It means Robot Framework has been installed on your computer.

## Install Robot Framework IDE

Each operating system has its own IDE which fits user’s expectation. I recommend using Robot Framework IDE (RIDE) for beginner users.

To install RIDE, you need to install wxPython (a cross platform GUI API). The wxPython must be compatible with Python version that installed on your computer. You have to install wxPython version 2.8.12.1 because you’ve installed Python 2.7 previously.

Navigate to the following link then download the corresponding wxpython 32/64 bit. It means if you installed Python 2.7 32 bit, you must download and install wxPython for Python 2.7, 32 bit. Otherwise, it won’t work!

https://sourceforge.net/projects/wxpython/files/wxPython/2.8.12.1/

After wxPython is installed, now you can install RIDE. Open terminal then type in

$ pip install -U robotframework-ride

To verify RIDE is installed, just open new terminal then type in

$ ride.py

RIDE will be opened and now you can interact with Robot Framework IDE.

If you’ve already known Robot Framework, you got to use with Robot Framework already. It’s time to move on. You can [download Pycharm](https://www.jetbrains.com/pycharm/) then install Robot Framework plugin. Now, everything can be explained by codes. However, Pycharm is a license application, you have 30 days trial then you must pay for a longer experience.

## Install Selenium library and some dependency libraries

As mentioned above, we use many engines which will be used to interact with different applications on different platforms. Here are the detailed steps to install those engines.

* Robot Framework Selenium library: this is an open source library to simulate user’s interactions with different browser on different platforms. It can perform those actions on Windows, Linux, Mac OS machine. It can also open an instance of browser such as Mozilla Firefox, Google Chrome, Safari, etc. It doesn’t depend on browser’s version because it has the advantaged use of webdriver. To install Selenium Library, open new terminal then type in

$ pip install -U robotframework-seleniumlibrary

* Web-drivers: As mentioned above, Selenium uses web-driver to create an instance of your favorite browser then performs actions there. You need to download the executable web-driver then pass the path to your Environment Variables.
  + For Google Chrome, go to the following link to download:

<https://sites.google.com/a/chromium.org/chromedriver/downloads>

* + For Mozilla Firefox: **TBD**
  + Internet Explorer
  + Microsoft Edge
  + Safari
  + Opera
  + Headless

***TBD***

* Request library:
* Database library:
* Appium library:

# Integrate with Jenkins

You may or may not know that Dynamic Data Distribution is using Jenkins as a continuous integration/continuous development system. To adapt to the expectation of project, this is the instruction on how to integrate Robot Framework with Jenkins.

## Set up and verify Robot Framework on Jenkins Server

TBD

## Set up Robot Framework plugin to export the report

TBD

# Tip and Trick

## Install a Python package by using binary files

When you can’t install a Python library using pip, you can try this workaround to install by your own. I’m using requests library for my example.

First, you need to download the request library (binary package) for python.

After extracted, the library folder contains a setup.py file. Open your command line tool then type:

$ python setup.py install

To verify the package is installed in your computer:

$ Python 2.7.14 (v2.7.14:84471935ed, Sep 16 2017, 20:19:30) [MSC v.1500 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>> import <package\_name>

If there is no error produced, you just installed a python library from a binary package.

## Install a Python package by using pip/easy\_install

In order to install a Python package, the fastest way you can do is to install by using pip/easy\_install.

For example, if you want to install the package called: “psutil” – a python package that helps you to measure/collect local cpu/memory usage, just open command prompt or terminal then type in:

$ pip install –U psutil

or

$ easy\_install psutil