**Test Automation User Guide**

Revision History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Version Number | Release  Date | Prepared By | Reviewed By | Approved By | Summary of Changes |
| 0.1 | 20/2/2019 | Satheesh S Nair |  |  |  |

Contents

[Purpose …………………………………………………………………………………………………………..4](#_Toc510630400)

[Understanding Automation Framework 5](#_Toc510630401)

[Setting-Up Automation Environment 8](#_Toc510630402)

# Purpose

This document serves as a User guide to the Automation Engineer who is new to the Environment or to someone who wants to setup the environment and execute the Automation scripts for the Appian project, written using POM Framework. It details about:

* The Automation Environment setup
* The Automation Framework
* Configuring TestNG
* Input test data
* Reporting features using Extent Reports

# Understanding Automation Framework

Automation Framework is a methodology adopted to ease the process of Test Automation, and is the process to provide guidelines to automation engineer(s) to develop good quality scripts by following certain best practices.

The Automation Framework used by the Appian Project uses POM Framework to create, execute and Report the Test Scenarios associated with the features of the AUT. The framework majorly comprises of the following components as listed in the bulleted points below. Please refer the following figure to understand the components:

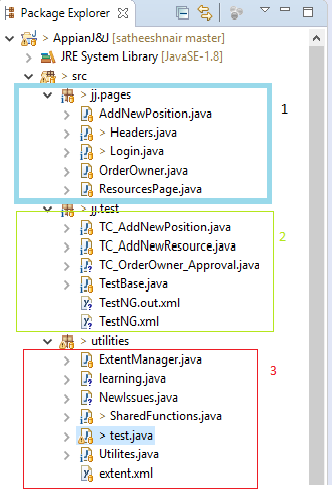


Figure 1

The corresponding folder structure of the Eclipse workspace is shown below:

1. src/jj.Pages 🡪 This package contains all the pages used by the Automation kit. This includes the locators and some of the unique functionalities the pages hold.
2. Src/jj.test 🡪 This package contains the Test cases to be executed. It also contains Test base page which will hold the initialization of browser and Extent Report Within this package there are three different components namely:
3. Src/utilities 🡪 This package contains Extent Report, Utilities, and Shared Functions. Utilities contains read and write to Excel, Share Functions contains common functionality that can be used across pages, Extent Report contains the reporting part and logs.

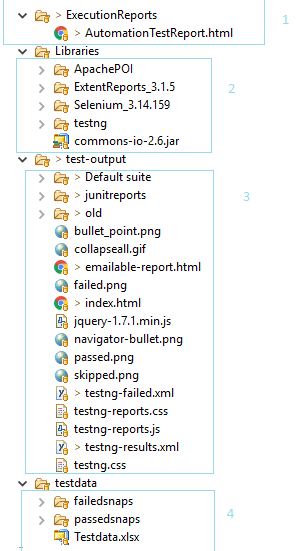


Figure 2

1. Some other folders and files in the Automation kit are as follows:
2. Execution Reports🡪 This folder contains the Reports created after the test cases are executed. It states whether the test case is pass or fail and step by step report
3. Libraries 🡪 This folder will have the Libraries for the entire framework
4. Test Output🡪This contains basic TestNG reports.
5. Testdate 🡪 This folder contains passed/failed snapshots, and test data which is needed for test execution

# Setting-Up Automation Environment

The following are required to be installed for the Cucumber Framework to be run:

1. Eclipse for Java Developers (Oxygen.3 release) – Download it from here if you don’t have it: <https://www.eclipse.org/downloads/packages/eclipse-ide-java-developers/oxygen3>. Make sure you download the same Eclipse architecture (32 - or 64 - bit) as your system/laptop. Unzip it to a folder in your machine, say C:\eclipse.
2. Java Development kit (JDK 8 version 32 or 64 bit according to your System/Laptop architecture) – Download it from here: <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
3. Selenium ChromeDriver – Download it from the site: <https://chromedriver.storage.googleapis.com/index.html?path=2.37/>
4. You need to add the PATH environment variables to your system. Based on the system you are running the tests (Windows 7, 8 or 10), go to the System Environment Variables (Control Panel > System > Advanced System Settings). If you don’t have Admin access to the system, please refer the Troubleshooting Section below.

Add the following variables to the “User Variables”:

* JAVA\_HOME 🡪 Path to the Java JDK Install Path
* PATH 🡪 Append the bin folder of the Java JDK path to the path already there, separated with a semicolon. Don’t remove the entries already there. They may be required to run some other applications in your system.

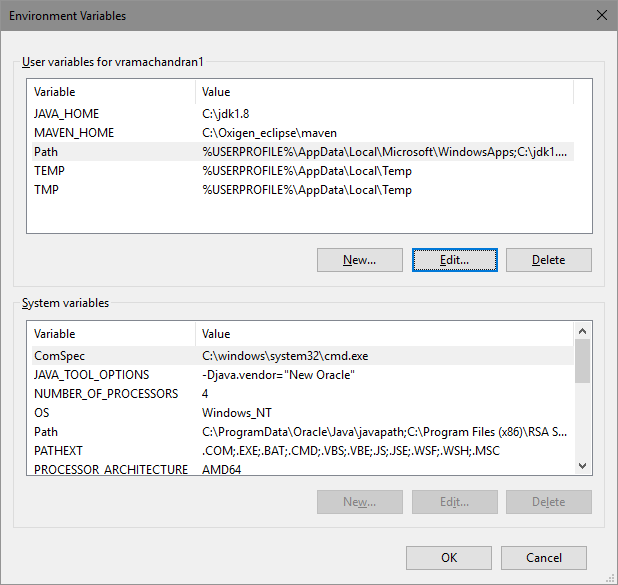


Figure 3

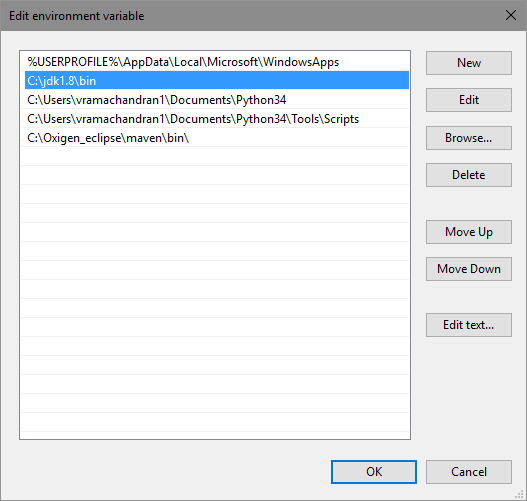


Figure 4

1. Open Eclipse; you will be presented to choose a Workspace for your project. Accept (and remember) the default workspace. If you want the workspace to be in a different place in your system, please choose it from the directory path chooser. Eclipse should open with the Java Perspective view by default.
2. After Installing and running Eclipse successfully, please follow the procedure below to install TestNG:
3. Go to Help > Install New Software. Click on the Add Button and Enter the URL (<http://beust.com/eclipse/>) at Work with field and click on "Add" button. Once you click on "Add", it will display the screen, enter the Name as "TestNG". After clicking on "OK", it will scan and display the software available with the URL which you have mentioned.
4. Importing Zipped Automation Project Kit: You would have received a zip file with all the Project files in it. Unzip it to a working folder (say C:\APPIAN\_JJ).
5. Now go to File > Import… You will be presented with the Eclipse Project Import window. Select the “Existing Project into Workspace” and click on Next.

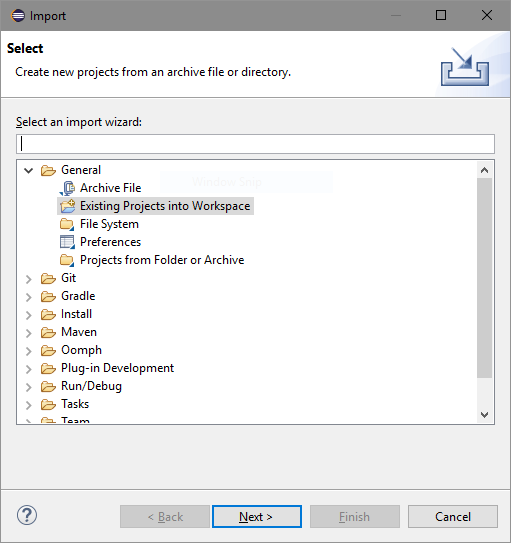


Figure 5

1. In the next screen, select the root directory where you have unzipped the files (C:\APPIAN\_JJ) in the “*Select root directory*”. Click on Finish.

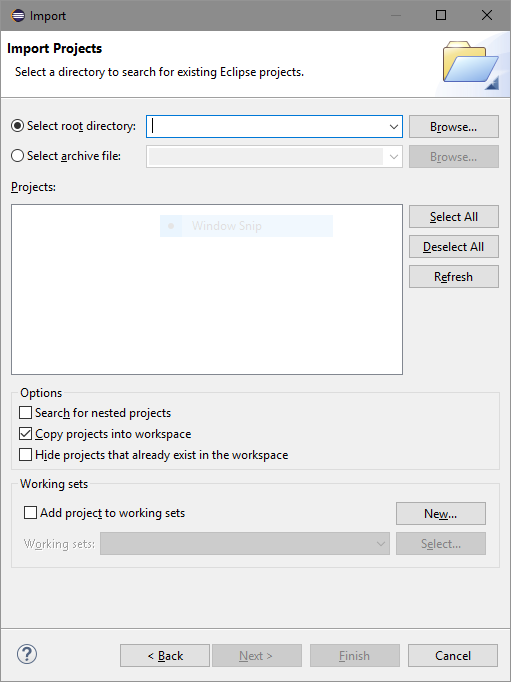


Figure 6

The Project will be imported into the workspace.

1. The next step is to download the dependencies for the project. Once dependencies are downloaded we need to add then to our project. Download all the jars and hold it in a single folder.
2. Right click on project and select Build path ->Configure Build path

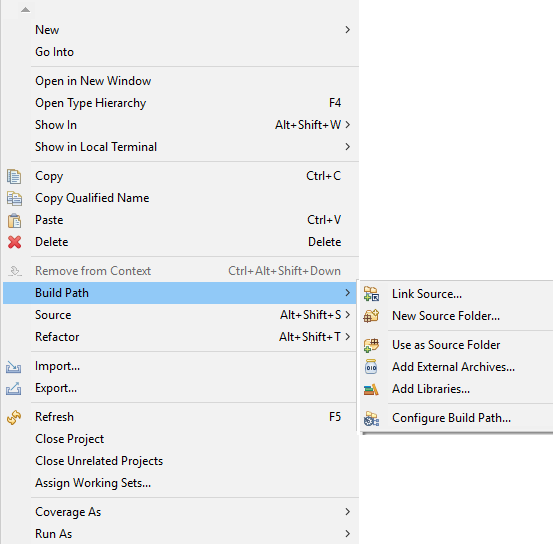


Figure 7

1. A new window with title “Properties for XXXX project” will be displayed. Select the libraries option from the tabs.

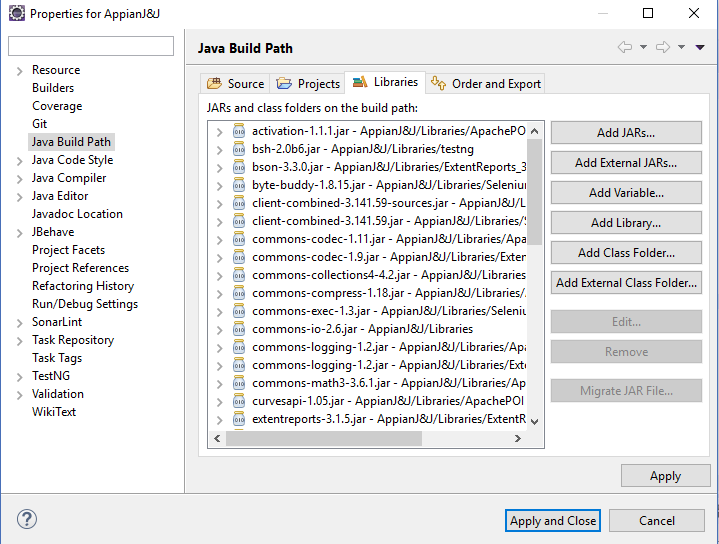


Figure 8

1. Now click on “Add External Jars” and select all the Jars you downloaded and click on “Apply and Close”

If all of the steps above gets completed, without any fail, Congratulations! You are good to go for running the Automation Test Suite.