**Robot Framework basics**

1. Create a new file with **.robot** extension

2. Add the below Modules: These are all the different Modules that the Test File supports

\*\*\* Settings \*\*\*

Documentation (Built In-Keyword) Description of the Testcase

Library Library we intend to use (ex-SeleniumLibrary)

Resource External.robot files to use in the Test

\*\*\* Test Cases \*\*\*

(Testcase Name) **To Validate Unsuccessful Login**

\*In Robot Framework, Testcase is just collection of keywords.

(Business Keywords)

Login to the URL

Fill up the Form

Wait until it checks and displays error message

Verify Error message is correct

\*\*\* Keywords \*\*\*

\*In this module, we need to wrap the Business keywords to the actual Keywords (Selenium Keywords in this example)

Login to the URL

Create Webdriver Chrome executable\_path=/driverbrowserpath

Go To http://www.TestURL.com

Fill up the Form

Input Text id:Locator\_id “Input Text”

Input Password id:Locator\_id “Input Password”

Click Button id:Locator\_id

Wait until it checks and displays error message

Wait Until Element is Visible css:Locator\_id

Verify Error message is correct (Using Built-In Keywords)

${result}= Get Text css:Locator\_id

Should Be Equal As Strings ${result} “expectedText”

\*\*\* Variables \*\*\*

3. Convert Manual Steps to Automation Scripts

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pre Conditions | | | | | | | | |
| 1 | | Browser is Open and Website is Launched | |  | |  | |  |
|  | |  | |  | |  | |  |
| Step No | | **Test Step** | | **Expected Result** | | **Actual Result** | | **Test Data** |
| 1 | | User Clicks on Search Text Box and Types in Search Text | | User should be able to type the search text successfully | |  | | “Anything” |
| 2 | | Click on Search Button | | Search result is displayed | |  | |  |
| 3 | | Verify results on the search page | | Search page should display correct results | |  | |  |
| Post Conditions | | | | | | | | |
| 1 | Browser is Closed | |  | |  | |  | |

Text

Description automatically generated

To Run a Robot Script:

saeemkhan$ **robot** -d Results Tests/Google/FunctionalTestSuite/BasicSearch.robot

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

4. Test Report

Graphical user interface, text, application, email

Description automatically generated

4. Better Approach - Move User Defined Keywords to **Resource File** (added Step 4 and 5 in this example)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pre Conditions | | | | | | | | |
| 1 | | Browser is Open and Website is Launched | |  | |  | |  |
|  | |  | |  | |  | |  |
| Step No | | **Test Step** | | **Expected Result** | | **Actual Result** | | **Test Data** |
| 1 | | User Clicks on Search Text Box and Types in Search Text | | User should be able to type the search text successfully | |  | | “Anything” |
| 2 | | Click on Search Button | | Search result is displayed | |  | |  |
| 3 | | Verify results on the search page | | Search page should display correct results | |  | |  |
| 4 | | Filter Search result by condition | | User able to filter results | |  | | “New” |
| 5 | | Verify filter results are displayed as per the selected criteria | | Filtered results should match the selected criteria | |  | |  |
| Post Conditions | | | | | | | | |
| 1 | Browser is Closed | |  | |  | |  | |

Text

Description automatically generated

CommonSteps.robot

A screenshot of a computer

Description automatically generated

eBay\_UserDefinedKeywords.robot

Text

Description automatically generated

5. “Test Setup” and “Test Teardown” Settings. This will allow to Launch and Close the Browser every time, even if the test fails.

Text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

6. Page Object Model (POM) Design Pattern in Robot Framework

We can maintain the page elements in separate files. We have multiple pages in our application and every page has a different file or we can specify elements from different pages into one file. So, the test cases will be divided into different parts.

* Easy to maintain Test scripts
* Reduction in code duplication

To implement POM, we first need to analyse the application

Header Page

Graphical user interface, application, website

Description automatically generated

1\_HeaderPage.robot

Graphical user interface

Description automatically generated

Search Result Page

Graphical user interface, application

Description automatically generated

2\_SearchResultPage.robot

Graphical user interface, text

Description automatically generated

eBay\_Verify\_SearchFunctionality.robot

Graphical user interface, text

Description automatically generated

saeemkhan$ robot -d Results Tests/Google/FunctionalTestSuite/eBay\_Verify\_SearchFunctionality.robot

Adding **Variables** to the same robot files:

Graphical user interface, text

Description automatically generated

Graphical user interface, text

Description automatically generated

7. Data Driven Testing in Robot Framework

What is Data Driven Testing?

* Running Automated Tests with different input data
* Data driven testing helps to achieve better test coverage
* Data driven tests will read data from different data storages (Built-In, External file (.csv/.xls) or Database)

**Testing without Data Driven – Using just Resource File (Not very efficient way to script, very repetitive)**

Graphical user interface, text, website

Description automatically generated

CommonSteps-LoginPage.robot

Graphical user interface, text

Description automatically generated

Text

Description automatically generated

**Data Driven Testing – Built-in approach**

**Graphical user interface, text

Description automatically generated**

**Graphical user interface, text

Description automatically generated**

**Text

Description automatically generated**

**Data Driven Testing – External File CSV**

Install robotframework-datadriver Package

**Graphical user interface, text

Description automatically generated**

Graphical user interface

Description automatically generated

LoginTest-DataDriven-CSV.robot

Graphical user interface, text

Description automatically generated

TestData.csv

Graphical user interface, application, table, Excel

Description automatically generated