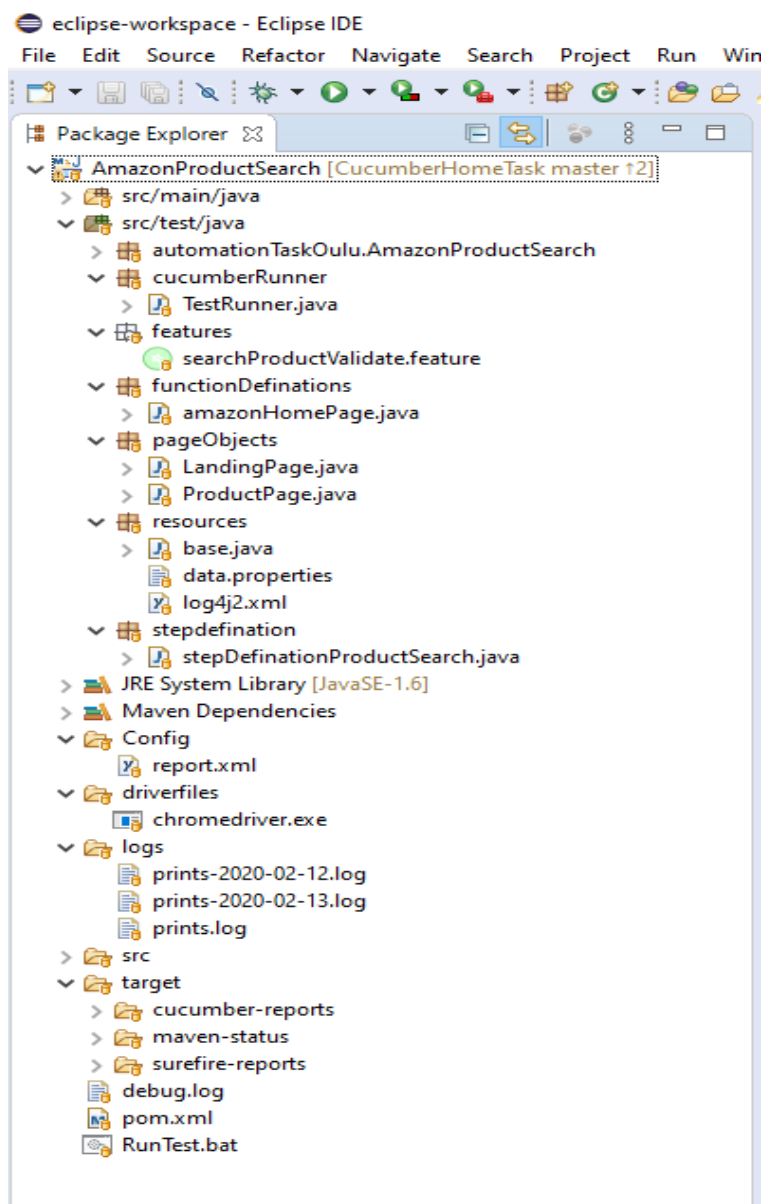


RUN BOOK

Pre-Requisite Setup to run framework and cucumber features

- Maven-Set Environment variable
- Java JDK-Set Environment variable
- Eclipse IDE
- Git
- Setup Cucumber Framework and adding data. Properties values as per needed
- Clone the project in C: Drive
- Open CMD and navigate to project and run maven Test or Run Batch file-RunTest.bat

Selenium-Cucumber Maven Framework Structure



This is the basis selenium-cucumber maven framework structure contains various packages.

Some highlights on framework

- This framework contains 2 different approaches which can be executed as a simple selenium j-unit test and another approach is through cucumber by defining feature files and stepdefinition using j-unit or testng and this test is Test Runner and we can configured as per the user or project requirement.
- POM.XML is heart of this framework which has all the dependencies added prior to defining framework structure
- Framework has excellent cucumber html report and added logs to get in detail information about test.
- It contain base file for driver initialize which is parents class which can be inherited into different java files. For more info please go through the code.
- It also contains data. Properties file which has browser, url, searchkeywords is used to avoid hard code. These values can be put in any source files. Ex-Excel, XML, CSV, config.ini, database...etc.
- Framework contain separate page Objects package so that it will be easy to maintain the web page object for bigger test.
- I have added Function headers and many comments in the code so that reviewer can read the code.
- No hard coded wait added in the framework as this is very small test and for bigger test we can create another package and define wait (Explicit or Implicit) for the objects to get loaded in webpages.
- Used Log4j2.xml to generate log and configure it in POM.XML. Log4j2 is heart of loggerManager.getLogger of base class.
- Feature file added as below.

```
1 #Author: NILESH PATIL
2 #Feature Reviewer: JANNE AND TIMO
3 #Date: 2020-02-12
4 #Keywords Summary : Create Selenium Web browser test using cucumber scenarios for amazon.com
5 #Feature: Search Nikon products on Amazon Website
6 #Scenario: Search the product and verify the availability of expected product
7 #Given: Amazon.com
8 #When: Search Nikon and sort results from highest price to lowest.
9 #Then: Select second product and click it for details.
10 #And,But: From details check (verify with assert) that product topic contains text "Nikon D3X"
11 #Background: User is on Amazon home page
12 @SearchProductValidate
13 Feature: Open Amazon Shopping Website and Search Nikon Products
14   As a user of Amazon
15   I want to be able to search for product using keywords
16
17 Scenario Outline: Search Product Details which is present
18   Given Initialize the browser with chrome
19   And Navigate to "http://www.amazon.com" Site
20   Then I should see Search box
21   When I fill in search input as <searchkeyword> and press Search button
22   And I get List of <searchkeyword> Products
23   And I Select Option 'Price: High to Low'
24   And I Select Second Product in List
25   Then I can see Product Title Containing 'Nikon D3X'
26   And close browsers
27
28 Examples:
29 |searchkeyword|
30 |Nikon|
```

- As given in feature file which is defined as SearchproductValidate.
- In this Scenario we can either hard code the values or can be parameterized as per the project requirement as given the code. Kept both option to make review understand.
- And last Feature header is very important to defined to make understand other test or coders

Execute test

- Pull the code from Github <https://github.com/TechNileshPatil/CucumberHomeTask> and save it on Local C: Drive
- Now you have 2 option either you can open through eclipse IDE by import existing maven project or double click on RunTest.bat which will run the test.
- If there are some failure or configuration issue, request you to look the error. As this framework is developed in another computer and there might be chance for some configuration issue. I am pretty sure you will not get any error and test will run successfully.
- Now this is must important point because is have configured my final test in j-unit and not using Testng. You will get error while running RunTest.bat that Testng is not configured, this is because I have set 2 different options to run the test as given above and my test will run in j-unit so please ignore it and test will run in 1 sec.
- Below are the test run screens through command prompt (cmd).

Command Prompt

```
C:\>cd CucumberHomeTask

C:\CucumberHomeTask>mvn clean test
[INFO] Scanning for projects...
[INFO]
[INFO] -----< automationTaskOulu:AmazonProductSearch >-----
[INFO] Building AmazonProductSearch 0.0.1-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- maven-clean-plugin:3.1.0:clean (default-clean) @ AmazonProductSearch ---
[INFO] Deleting C:\CucumberHomeTask\target
[INFO]
[INFO] --- maven-resources-plugin:3.0.2:resources (default-resources) @ AmazonProductSearch
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] Copying 3 resources
[INFO]
[INFO] --- maven-compiler-plugin:3.8.0:compile (default-compile) @ AmazonProductSearch ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to C:\CucumberHomeTask\target\classes
[INFO]
[INFO] --- maven-resources-plugin:3.0.2:testResources (default-testResources) @ AmazonProductSearch
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory C:\CucumberHomeTask\src\test\resources
[INFO]
[INFO] --- maven-compiler-plugin:3.8.0:testCompile (default-testCompile) @ AmazonProductSearch ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 7 source files to C:\CucumberHomeTask\target\test-classes
[INFO]
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ AmazonProductSearch ---
```

Select Command Prompt

```
at org.apache.maven.surefire.booter.ForkedBooter.execute(ForkedBooter.java:126)
at org.apache.maven.surefire.booter.ForkedBooter.main(ForkedBooter.java:418)
Chrome
Starting ChromeDriver 79.0.3945.36 (3582db32b33893869b8c1339e8f4d9ed1816f143-refs/branch-heads/3945@#614
48523
Only local connections are allowed.
Please protect ports used by ChromeDriver and related test frameworks to prevent access by malicious code.
1581547315.835][WARNING]: Timed out connecting to Chrome, retrying...
Feb 12, 2020 11:41:58 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected dialect: W3C
1581547320.201][WARNING]: Timed out connecting to Chrome, retrying...

1 Scenarios (1 passed)
0 Steps (9 passed)
43.505s

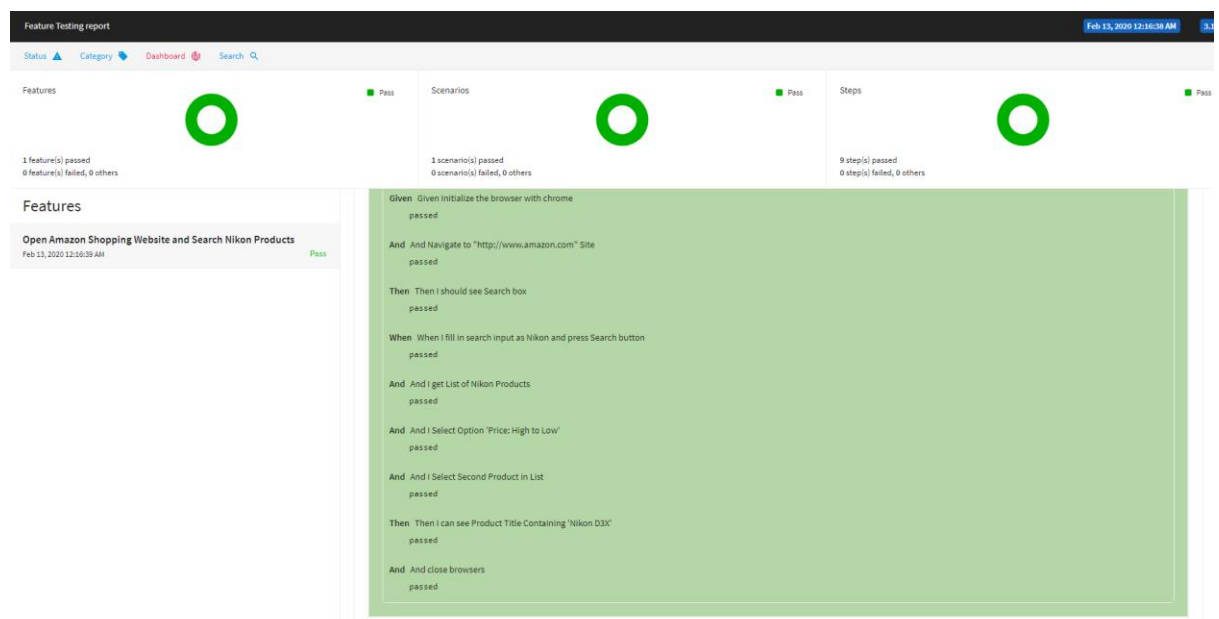
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 49.519 s - in TestSuite
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 01:04 min
[INFO] Finished at: 2020-02-12T23:42:30+01:00
[INFO] -----

C:\CucumberHomeTask>mvn clean test
```

```
Command Prompt
[INFO] Compiling 7 source files to C:\CucumberHomeTask\target\test-classes
[INFO]
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ AmazonProductSearch ---
[INFO]
[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO] Running TestSuite
TestNG [ERROR] No test suite found. Nothing to run
Usage: <main class> [options] The XML suite files to run
Options:
  -configfailurepolicy
    Configuration failure policy (skip or continue)
  -d
    Output directory
  -dataproviderthreadcount
    Number of threads to use when running data providers
  -excludegroups
    Comma-separated list of group names to exclude
  -groups
    Comma-separated list of group names to be run
  -junit
    JUnit mode
    Default: false
  -listener
    List of .class files or list of class names implementing ITestListener
    or ISuiteListener
  -methods
    Comma separated of test methods
    Default: []
```

Test Result- Cucumber HTML and Logs

- After the test execution you can get this result from <C:\CucumberHomeTask\target\cucumber-reports\report.html>
- And for logs <C:\CucumberHomeTask\logs>



Dashboard

Features

■ Pass

1 feature(s) passed
0 feature(s) failed, 0 others

Scenarios

■ Pass

1 scenario(s) passed
0 scenario(s) failed, 0 others

Steps

■ Pass

9 step(s) passed
0 step(s) failed, 0 others

Features

1

Scenarios

1

Steps

9

Start

Feb 13, 2020 12:16:38 AM

End

Feb 13, 2020 12:17:21 AM

Time Taken

42,248ms

Categories

Name	Passed	Failed	Others	Passed %
@searchProductValidate	2	0	0	100%

Post Execution Test

- Validate the reports and logs if any defect found and report it to defect management tool and send this reports to repository to accessible to team or project
- Also we can implement hook to send an email the test results to the test manager after execution. It mostly depends on project and team.
- And last for CI/CD we can create a job in Jenkins to trigger this test automatically the whenever new release is planned.

Test Data for Test cases execution

- In most of the organization and the project this is very important to have correct test data for execution and to test the functionality or business critical flow. And we can create separate team as TDM to perform this task which generates test data as per the scenario or data profiles based on bcfs or functionality.
- Test data should be consistence, coherent and compliant as per the EU GDPR rule.