**Objective:**

To automate a safe but thrifty shopper’s shopping action.

**Purpose:**

To achieve the automation for Shopper’s shopping action scenarios using Keyword Driven Framework. Purpose of choosing Keyword Design Framework, is suitable for to analyse easily for Test Script execution not only for Automation testers but also for Manual testers as well.

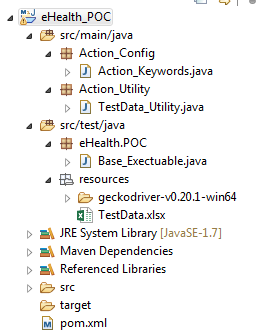
**Framework Design:**

Designed Keyword Driven framework, in Eclipse IDE using Selenium as automation tool.

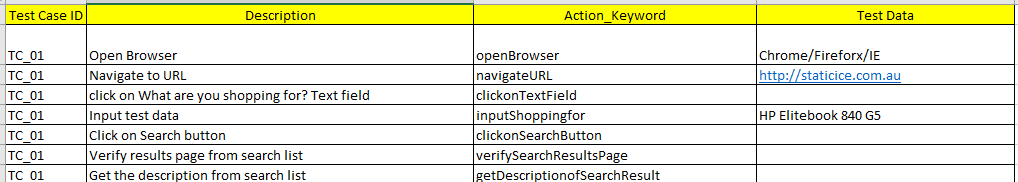
It contains various software’s and library files like Java, JDE, Selenium, Maven, Apache POI, TestNG, Extent reports and Junit dependencies.

Folder Structure:

Below screenshot, represents the folder structure of Maven project for the Keyword Driven framework as part of eHealth POC (Proof of concept).



1. **eHealth.POC/Base\_Executable.java** file contains high level of executable scripts, which call each methods from Action\_Keywords.java. Can design and write code more effective by using annotations, reporting etc.,
2. **Action\_Config/Action\_Keywords.java** file contains the list of methods which are mentioned as per in TestData.xlsx under Action\_Keywords column. Each method has designed for each action keyword. Major advantage of this action keyword is reusability of methods across the project. Can design more effectively by using Object repositories for each element. These object repositories can be Local or Shared.
3. **Resources/TestData\_Utility.java** file contains the read and write methods for using test data from TestData.xlsx.
4. **TestData.xlsx** is the Test data file contains detailed test steps to execute the shopping cart scenario. Can use for both Manual (as Testcase Template) and Automation testing.

****

TestCaseID is the Test Case ID for above scenario.

Description: Describes each step with detail information of execution

Action\_Keyword: Contains the keywords for manual testing but also use as keyword methods for automation testing.

TestData: Is the Test data used for test steps execution.