Nikardo Adams  
QA Engineering Problem  
Insightly, San Francisco  
2/11/2016

**Overview:**The testing framework has been created as a Maven project using TestNG and DataProvider with Excel. It consists of one test class EbayListingsTest.java and one test suite EbayListingsTestSuite.xml. The test suite is configured to run 1 test scenario verifySearchListingsCount(). The suite is setup to run in parallel with 2 threads implementing cross browser testing against Firefox and Chrome. The test scenarios themselves utilize a data driven approach with the test values stored in EbayListingsTests.xls. More details on each test method are specified at the top of the testcase class.

**Running:**The project can be cloned from https://github.com/nikadams/Ebay.Insightly.git. To run as a maven build from the Command Line issue the command “mvn clean test” from the root of the project directory where the pom.xml is located. To run in Eclipse, copy the cloned files into the new Maven project. The pom.xml should include all dependencies needed. EbayListingsTestSuite.xml located in root directory can be executed as TestNG Suite. The Chrome/Firefox driver are included in the project in the test\resource\webdrivers folder and referenced relative (Windows OS) to their location in the project so no further configuration should be required. EbayListingsTest.java can also be run as TestNG test in Eclipse. There are 4 Suite tests where 2 are expected to fail.

**Project Improvements:**As this is a basic project with time constraints there are certainly areas that can receive improvements. The DataProvider algorithm used includes a basic method to retrieve the items between markers and return the entire array. Another approach which I prefer is to store the values of the array into a Map<key,value>. This allows you to return only a Map object into your test method and you can use Map.get(column\_name) to retrieve its value, rather than explicitly passing in each variable into the test case as there could be many variables in the spreadsheet to keep track of.

Also use of the logger along with screenshots for end of tests and failed test will certainly help to improve debugging, while dynamically outputting to a test-output location.

Finally, there is currently no dynamic web driver manager used in this assignment. The specific Web drivers needed are included in the project however there are open source webdriver managers which can detect the system architecture and download the latest driver and set the path dynamically. This is especially useful when running across many different environments.