Page Object Model (POM) Architecture

Advantages of POM: Code Reusability, Code Maintainability, Code Readability, Object Repository

FRAMEWORK COMPONENTS (WEB/DESKTOP/MOBILE APPS)

DevOps

Base Class / TestBase.java (parent)
FileInputStream, Initializing Webdriver;*
Properties prop (), EventFiringWebDriver
Maximize window, deleteAllCookies
pageLoadTimeout, implicitlyWait

TestNG.xml
Annotations are easier to understand, test cases can be grouped
more easily, parallel testing is possible

Page Layer

Test Layer

Config.Properties
Url
Login Credentials

LoginPage.java
WebElements – Initialize Page
Objects Factory
Initialize LoginPage
constructor()
Actions () – different features of
the page

LoginPageTest.java
@Before class/method
@Test - @dataprovider,
Assertions
@After class/mothod

XlsReader.java
getRowCount();
getColumnCount();
getCellData(); setCellData();
addColumn();
removeColumn();
isSheetExist();
addSheet(); removeSheet();

Logs/Reports
ExtentReport.java
Log4j.properties
WebDriver logs
HTML Report

Maven (POM.xml) - Build Automation
Support dependency management, easy to maintain, supports a multi-project structure

Git & GitHub Repository

Jenkins - Continuous Integration

Home Page

Home Page Test

Test Data
Controller.xlsx

WebEventListener
Class

Product Page

Product Page
Test

Common Utilities
EventHandler.java
CommonUlti.java
Xls_Reader.java
JSExcecutor.java
IframeUlti.java
TestUlti.java

Screen Shots

Logout Page

Logout Page
Test

Designed by John Dang (JD)