### Overview of Automation Suite

The automation script is written in java using selenium web driver. We used BDD framework. Cucumber feature file is driving the test from front end and cucumber runner class is internally drives the testing using step definition classes. We used POM (Page object method) with Page Factory concept so each page from main page till kiwi saver retirement calculator page we have a separate page object.

*Note: Westpac NZ Limited website is developed in Angular js, based on my knowledge, using protractor, BDD framework, Jasmine (BDD for protractor) is the best suitable automation tool but since I don’t have much exposure in protractor, I have used Selenium, java, cucumber, etc.*

Please find below attached document for test requirements.



Based on the stories, test suite has two feature files each story has been converted as a feature file. The scenario has been driven mostly from acceptance criteria few from story point itself e.g. scenario number one on feature file tagged as @Stroy1 contains the input data to bring all the info icon to visibility.

We have utilised the cucumber tagging feature so we can run the test individually from maven cucumber command from windows command prompt or eclipse ide using cucumber runner class tag option.

Used dynamic field identification methods in few page classes to cover higher volume of elements with similar attributes

Take screenshot after validation or input entered and save it in project/target/Screenshot folder, in case of failed scenario (Most of the scenario) project/target/Screens/FailerScenario folder.

Planned vs Delivered:

Test data file:

The actual plan to keep the input data is in a file format (excel or xml) but having minimal requirement and the usability of the test suite (only for interview purpose and time crunch due to production issue testing in parallel) we kept the input data in feature file itself.

Log4J console + log file print:

Due to resent crash of my laptop, I have lost my previous setup, I tried to bring by log4j console + logfile print but unfortunately, I couldn’t due to time crunch. However, I managed to bring the console print but still I haven’t used in the automation suite

System Requirements (or Prerequisites)

1. Basic system requirement to run a simple java application
2. Java IDE (Eclipse/ IntelliJ) - Environment that you will use to write your code in. We used Eclipse to develop this automation suite.
3. Selenium Web-driver libraries - Libraries which allows you use all the Selenium functions and classes.
4. Browser drivers (Chrome/ Firefox) - The driver which Selenium WebDriver communicates. Chrome driver comes with the project folder so no need to download separately.
5. Apache Maven
6. Please refer below links for related download and installation guides

<https://www.java.com/en/download/help/download_options.xml> (java version "1.8.0\_161" or above)

<https://www.eclipse.org/downloads/packages/installer>

<https://cucumber.io/docs/installation/>

<https://maven.apache.org/install.html>

### Set-Up Project/test suite

Once completed the pre-requisite, please download the project/test suite from given git location and import the project using eclipses ide.

Project location: <https://github.com/shyam239/WestPackInterview>

Please click the below link for more details.

<https://help.eclipse.org/kepler/index.jsp?topic=%2Forg.eclipse.platform.doc.user%2Ftasks%2Ftasks-importproject.htm>

how to run the project:

Once pre-requisite and Setup has been completed, we can run the project multiple ways few easy ways are below.

1. Run the .bat files in project location (*‘YourWorkSpace’*\WestPacInterview\interview)
2. Run from eclipse IDE (Right click on project in eclipse ide->Goto ‘Run as’->Select Maven Clean and Right click on project in eclipse ide->Goto ‘Run as’->Select Maven Install)
3. In any class file of the test suite press right click-> select ‘run as’->double click TestNg->give project name as ‘WestPacKiwiSavRetCalc’, project as ‘interview’ (best to browse and select), Class as ‘cucumber.CucumberRunner’ and click run to start execution. Please change the tag name in cucumber.CucumberRunner.class file to execute appropriate test case.