**Automation** **Framework Description**

**1.** **Purpose**

This document describes the architecture of the Automation framework components and explain the operational details.

**2.** **Highlights**

1. **Technology**: JAVA 8
2. **BDD Framework**: Cucumber
3. **Automation Framework:** Selenium 3.14
4. **Project Management:** Maven

**3.** **Framework Architecture**

The Framework consists of the following components

1. **src/main/java:** It consist of packages containing all test utility, constant files, selenium action events.
2. **src/test/java:** It consist of packages containing Test Runner, step definition and business utility class for verification
3. **src/test/resources:** It contains all the feature files and properties file.

**4. Overview of Automation Components**

I have used below packages and classes to create the framework. Please find below the components:

**5. Test Execution Step**

|  |  |
| --- | --- |
| **Number** | **Description** |
| src/test/resources/FeatureFiles | Contains all the feature files according to the features. All the feature files will have scenarios in Cucumber BDD syntax. |
| com.test.stepDefinitions | Contains separate java class file for each feature file. In class file, we will map the BDD statements. |
| com.test.runner | Contains the Cucumber Runner java class. This will be the entry point for our framework |
| com.test.pageObjects | This package contains the classes which basically acts as an object repository |
| com.test.businesslogic | This package contains the classes which handled the business specific actions particular to the page |
| com.test.webdriver | This package contains the Driver classes which initializes the Webdriver instance as per browser parameter provided |
| com.test.util | This package will contain the utility classes like wait time, file reading etc. |
| com.selecnium.actions | This package contains the SeleniumActions class which is basically a wrapper methods for selenium actions and events |
| com.config.properties | This package has LoadProperties class which reads all the config properties defined for framework and user configurations |
| src/test/resources/config.properties | This properties file will contain all the framework configurable properties and test data |

* Take clone of the project from git repository
* Import as maven project
* Updated the project to download all the necessary dependencies
* Make sure Developer and USB Debugging is enabled in Android device
* Install Appium
* If Running on iOS device install XCode along with Appium
* Make sure Appium is started
* For Android fill the below details in config.properties file:

1. Udid
2. androidVersion
3. remoteURL

* For iOS fill the below details in config.properties file:

1. Udid
2. iOSVersion
3. remoteURL

* Fill the below user details in config.properties file

1. emailId
2. password
3. username

* Please update the user name and password for or else test case will get failed. I have provided the dummy user for now
* Navigate to the Projects POM location
* Run the below command:

1. **For ANDROID:** mvn clean install -DbrowserName=chrome\_android
2. **For Safari:** mvn clean install -DbrowserName=safari\_ios

**6. Results**

* Once the execution is completed Results are generated in the target /cucumber-report-html/cucumber-html-reports/ overview-features.html

