Automation Framework Documentation

Contents

[1. Introduction 3](#_Toc43656520)

[2. Framework Modules 3](#_Toc43656521)

[ pom.xml 3](#_Toc43656522)

[ testing.xml 3](#_Toc43656523)

[ environment.xml 3](#_Toc43656524)

[ src folder 3](#_Toc43656525)

[ testoutput folder 5](#_Toc43656526)

[3. Reporting 5](#_Toc43656527)

[4. Prerequisites 5](#_Toc43656528)

[5. Test Case Output 6](#_Toc43656529)

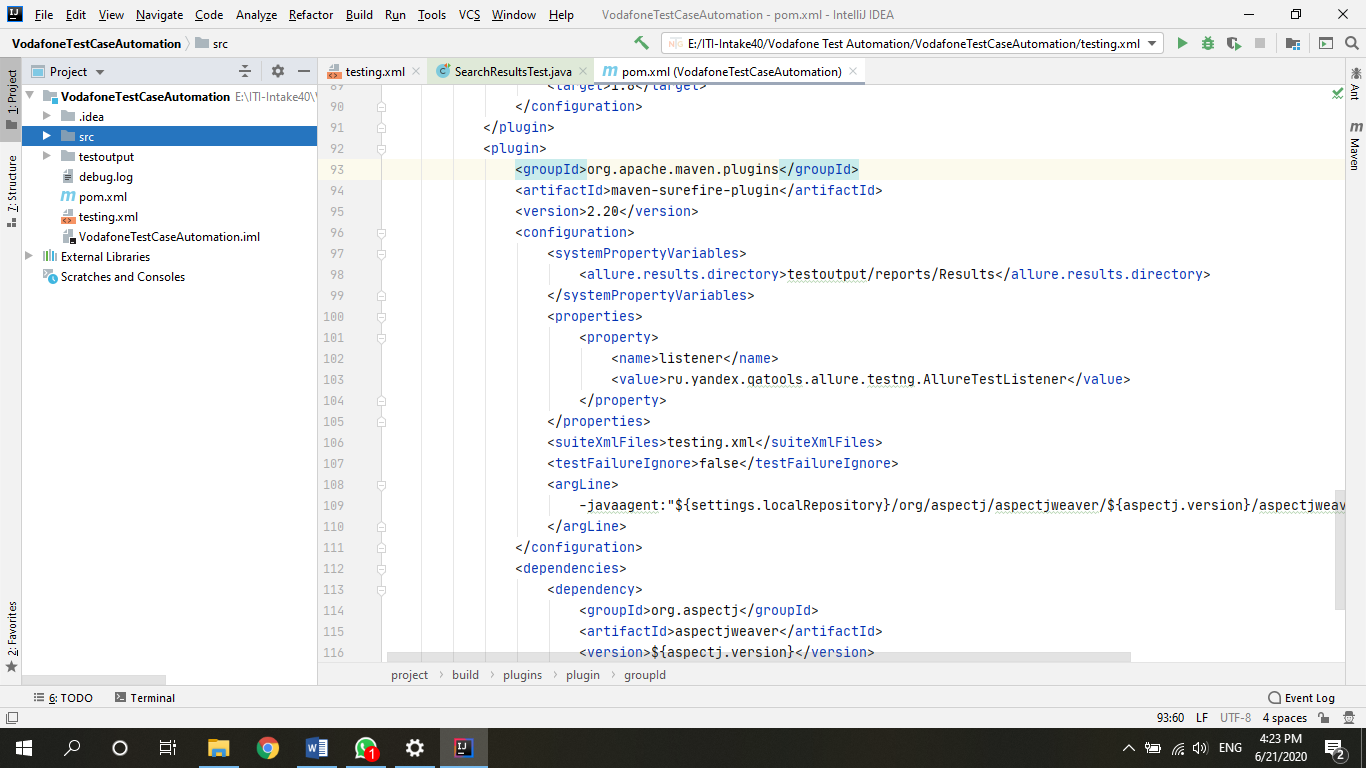
[6. How to run? 6](#_Toc43656531)

## Introduction

This is a documentation for my automation framework that has modules will help the test automation engineer with writing a test automation script, it is a maven project, and these framework modules will be declared below. And I apply it on the test case that was sent in the mail. This framework based on POM, Data provider, Object-Oriented Design, and cross-browser concepts.

## Framework Modules

My framework was designed as below



### pom.xml

This file includes all the dependencies that I needed to build my project and includes some comments that identify the reason that I need this dependency

The Main dependencies:

* WebDriver Manager: which needed for manage the drivers of Chrome, Firefox, and Edge browsers, make it easy when I didn’t need to download them, only download IE-driver for some issues I will clarify it in the next points
* TestNG: which needed to make test cases, and run it with java
* Selenium: which needed to help us interacting with our browsers
* JSON-Simple: needed to help us read from JSON file
* Allure Reports: needed to help us generate an Allure Reports

### testing.xml

This file contains the configuration of our test suit, test classes, cross browser, threads, and parallel tests

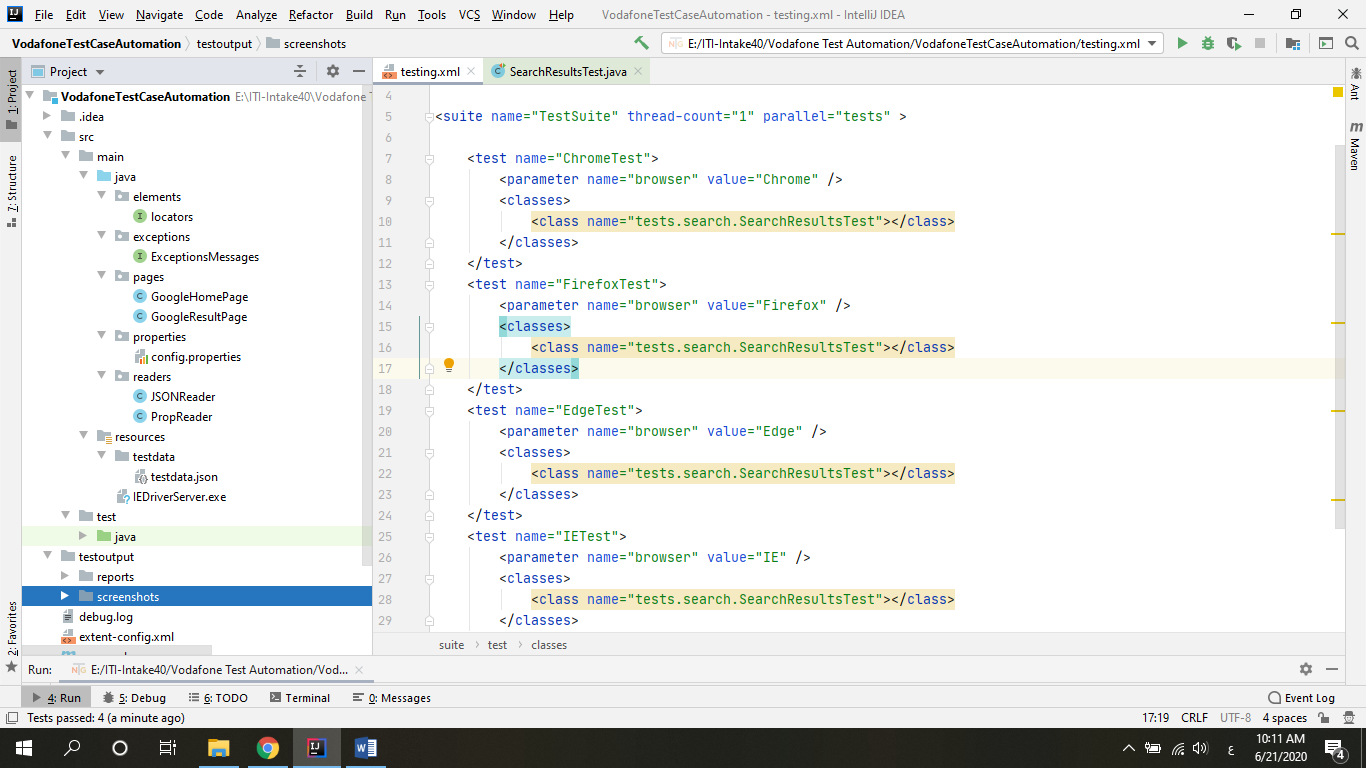
### environment.xml

This file contains the configuration of our Allure reports like report directory. It is generated with by code in the Result file

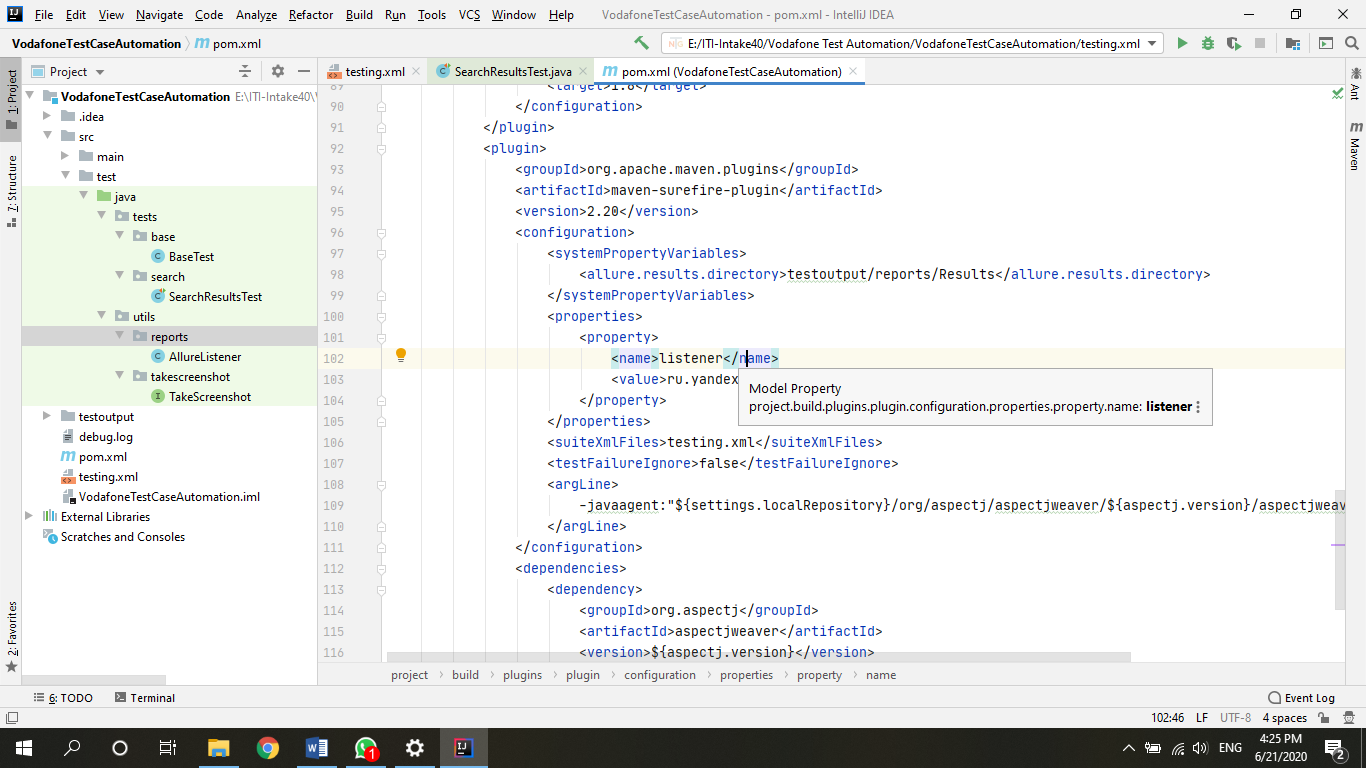
### src folder

This file contains my code and those things that may be support it like:

1. main folder: This folder contains all java code that will support our tests and the main resources like test data

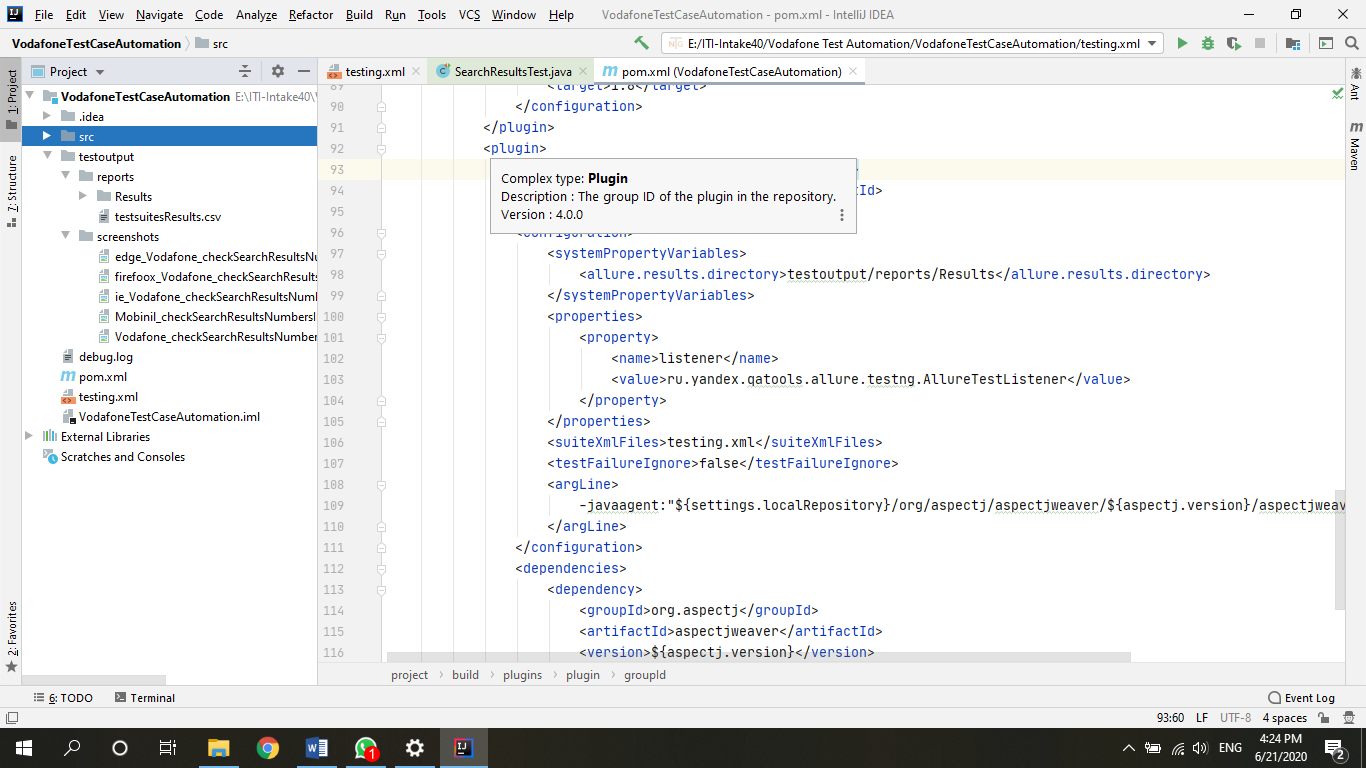


1. java
2. elements: this package contains an interface to make it easier to read the locator from the property file using PropReader class and find the element
3. exceptions: this package contains an interface to make it easier to print an error message for the test automation engineer to make the error understandable
4. pages: this package contains pages that I will interact with in my test case with all locators and methods needed (POM)
5. properties: this package contains the config.properties file, this file contains all locators needed with a descriptive name to make it easier understand and refactor the code for test automation engineer
6. readers: this package contains all readers I needed to read data from files in my test case
7. resources
8. testdata: this folder contains the JSON file that contains all test data needed
9. test folder: This file contains all java code to make out tests and the main utilities for test like reports, and take screenshot



1. java
2. tests package
3. base: this package contains the main base test methods like @BeforeTest, @BeforeClass, @BeforeMethod, @AfterMethod, @AfterClass, and @AfterTest.
4. search: this package contains our test case scenario class which extends from the BaseTest class and implements from EceptionsMessages interface.
5. utils
6. reports: this package contains the classes which help us to make a test report
7. takescreenshot: this package contains an interface which help us to take screenshot as a file at failure

### testoutput folder



This file contains all test output data like screenshots for each browser if test failed, and reports

## Reporting

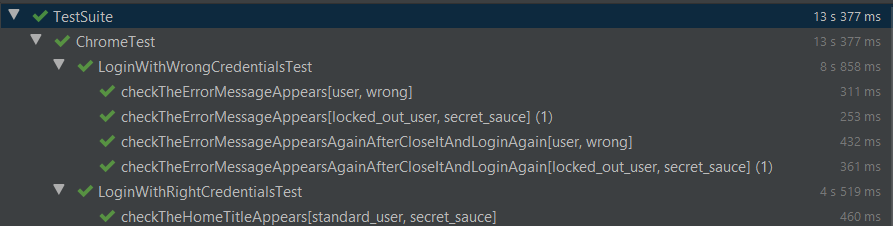
I have used 2 frameworks for generating Reports (Extent Report, Allure Report) and both give me a good result but I thought the Allure can make a lot of things comparing with Extent Report.

## Prerequisites

* Add Allure to System path -> the Jar file in the resources folder
* The Environment that this test case ran well on it (my environment)
* OS: Windows 10 Pro amd64
* Java Version: 11
* The IDE used: IntelliJ IDEA Community Edition
* Apache Maven
* Browsers:

1. Chrome 83.0.4103.106 (Official Build) (64-bit)

## Test Case Output



## How to run?

* You can run from the testing.xml
* To Generate the Report, you need to run “allure serve Results” in the “reports” directory.

## Project URL

Link of my project: <https://github.com/HebaaAshraaf/DellSauceTestAutomation.git>