Project Setup Instructions:-

The Project has been created using the below software products,

a. Maven Project – is a tool to manage builds documentation, reporting, dependencies etc.

b. BDD-Cucumber – is a test automation tool written in Gherkin format to support Behavior driven development.

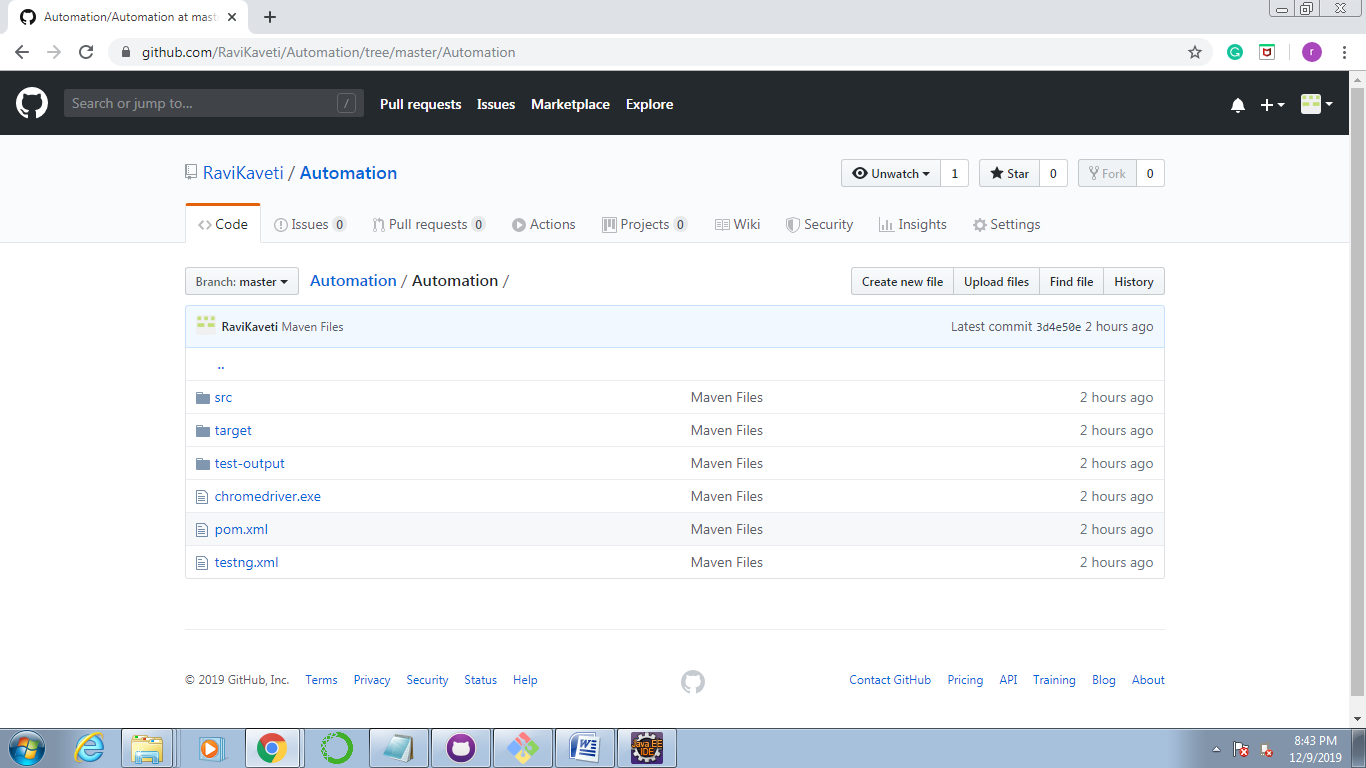
c. Selenium WebDriver – creates browser based robust automation test scripts.

d. TestNG – is an open source test automation framework for Java.

e. Java jars – to support the entire framework with an integration of java jar files.

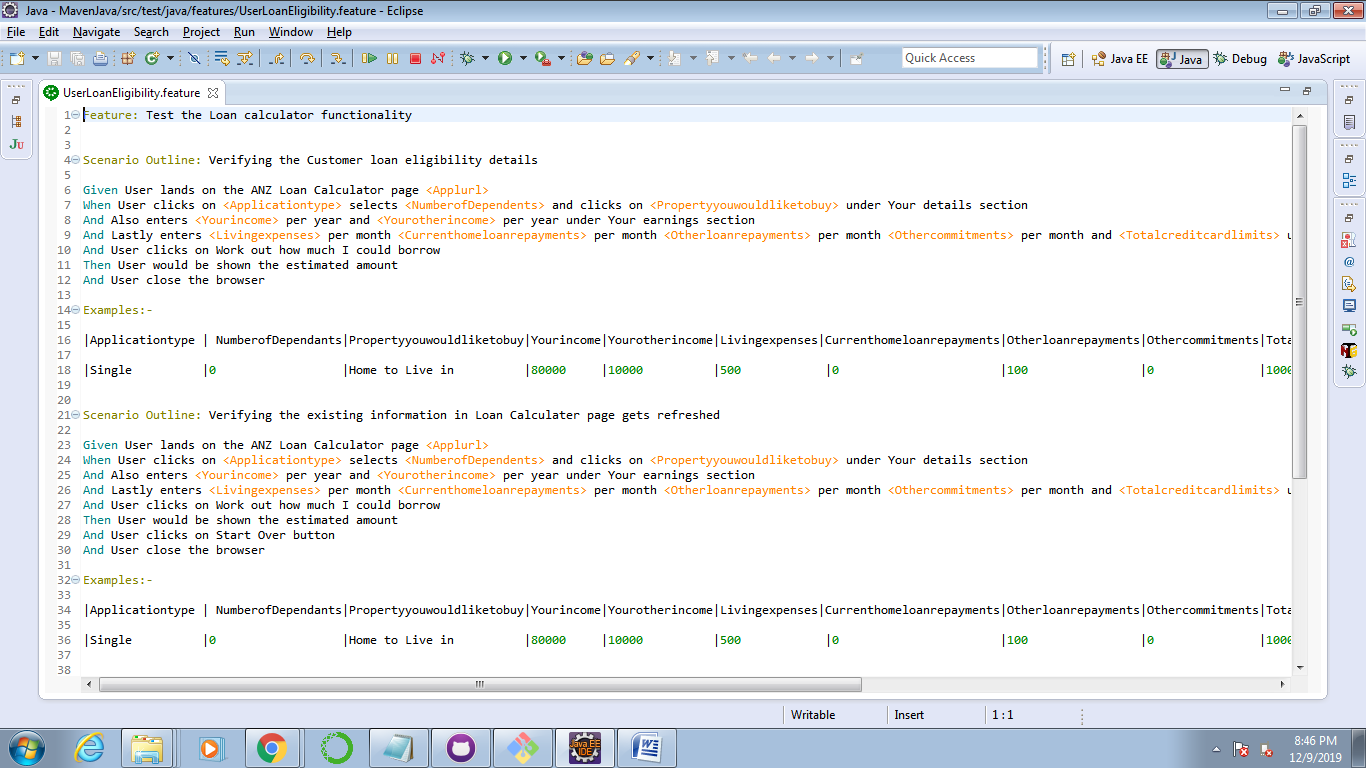
f. GitHub – is an open source version control system to maintain the project repositories.

Github repo link to clone the project - https://github.com/RaviKaveti/Automation.git

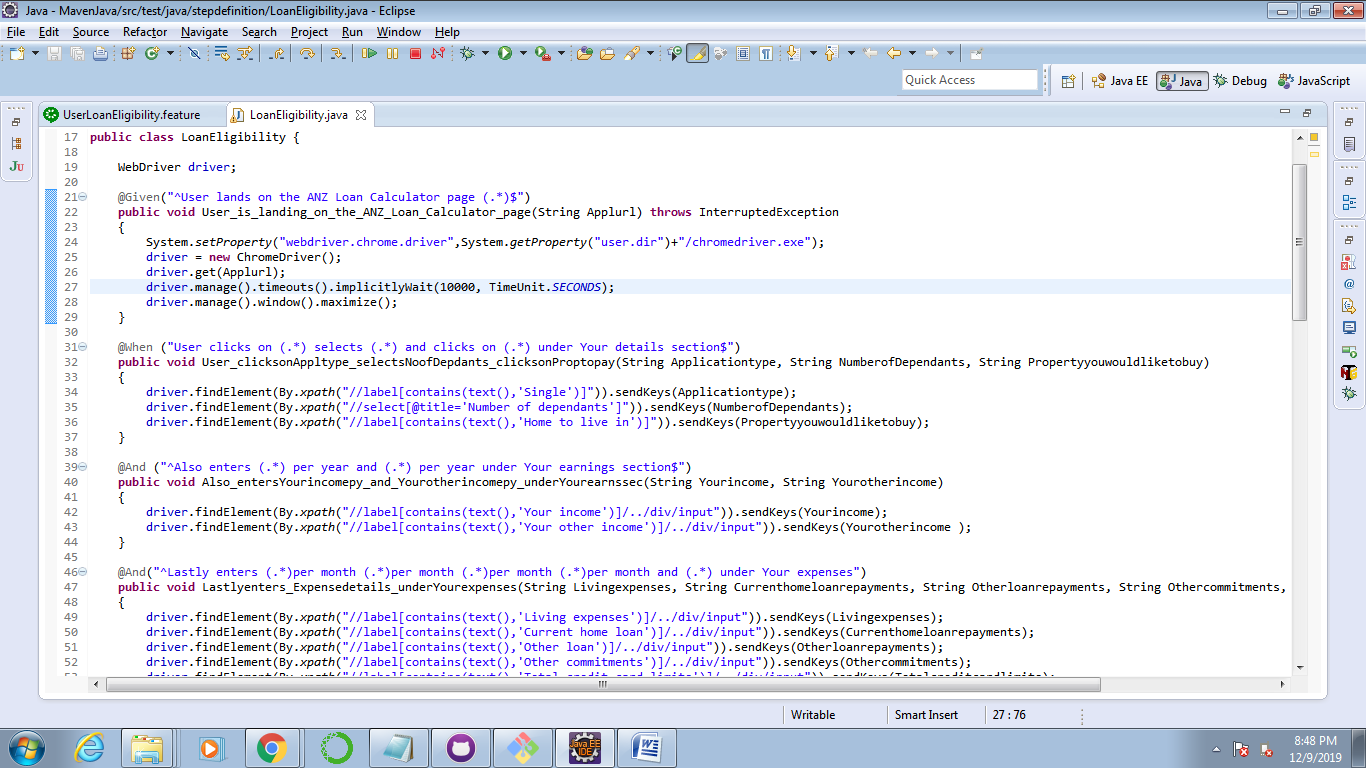


BDD-Cucumber:-

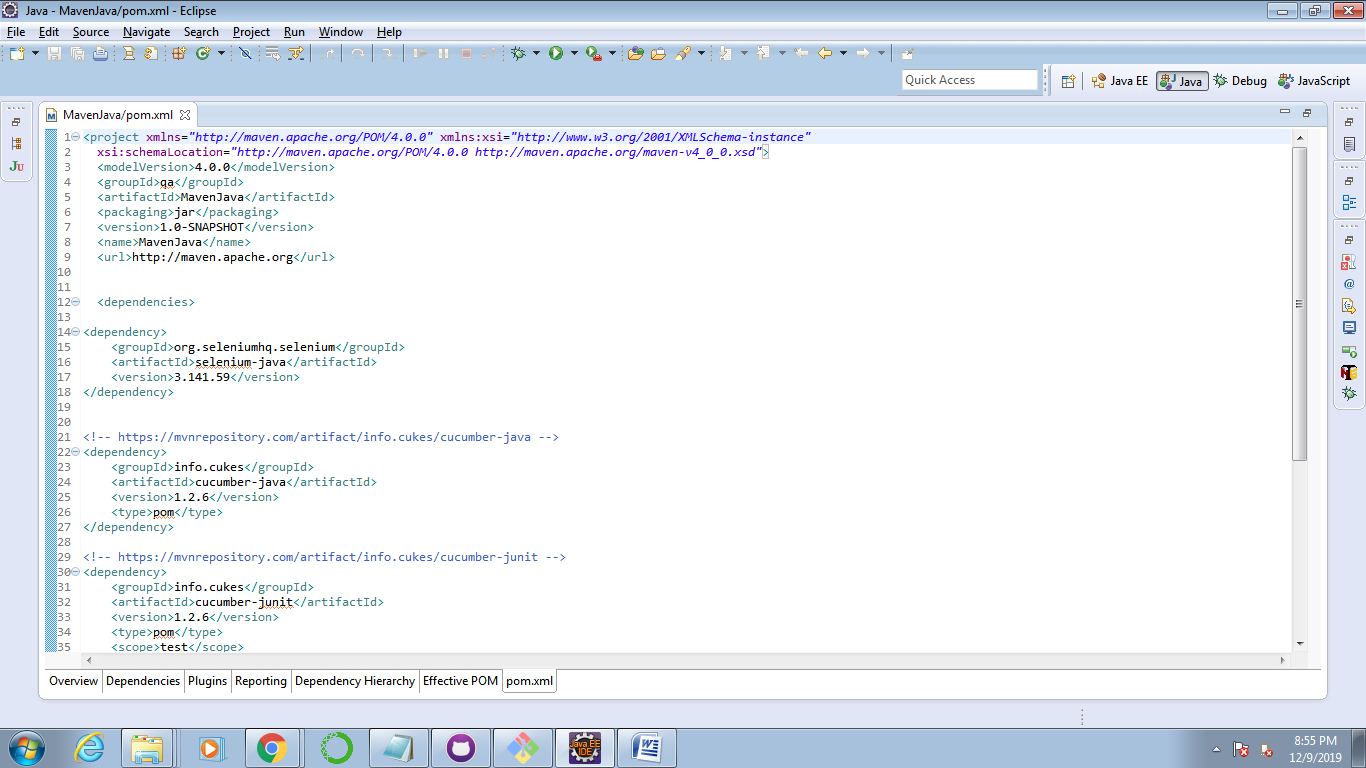
Feature file



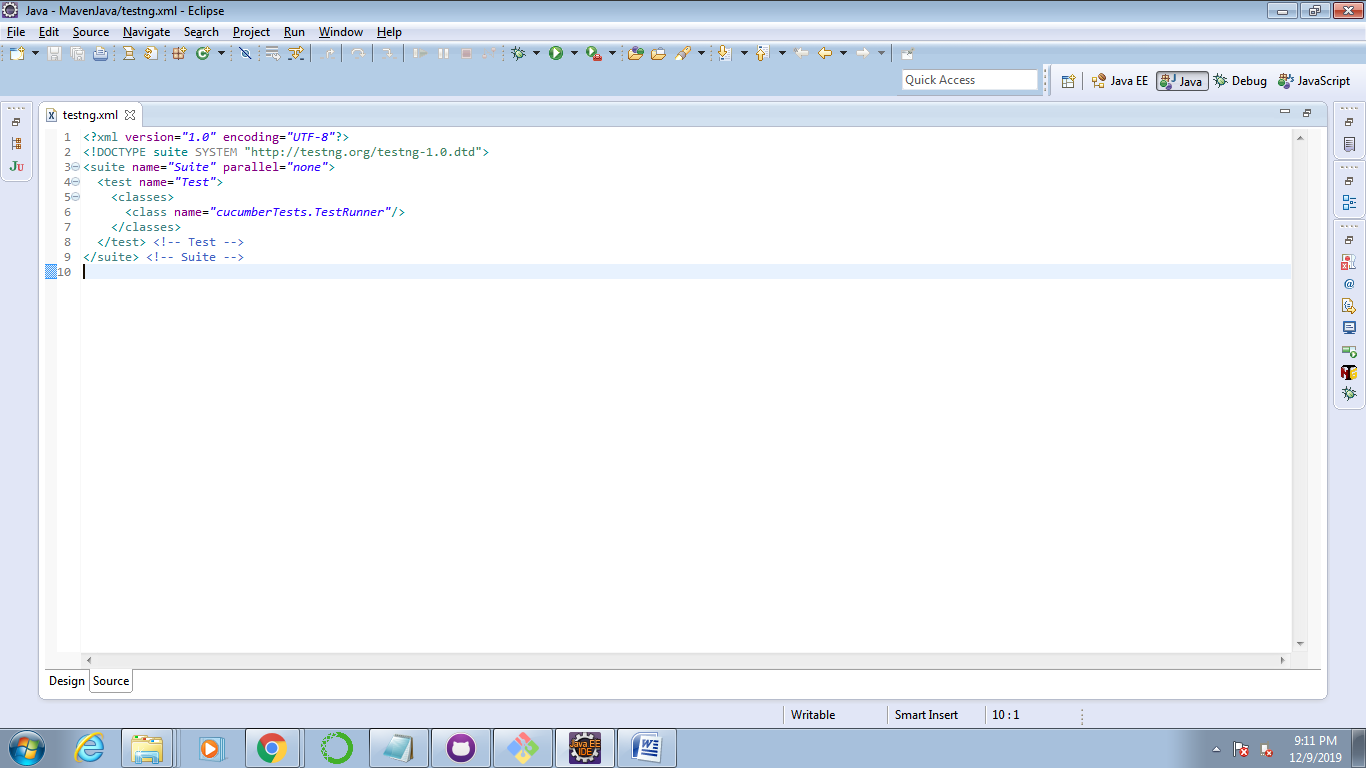
Step Definition file



POM file



testng.xml



MVN commands:-

MVN CLEAR

MVN COMPILE

MVN TEST

POM file dependencies:-

Selenium-java

Cucumber-java

Cucumber-testng

Test script execution:-

Right click on testng.xml file and choose Run->TestNG option.

Reports:-

test-output->emailable-report.html, index.html – generattes reports in html format.

Miscellaneous Questions:-

1. What other tests would you suggest could be written? You do not need to write them out in detail or code them.

There are certain validations which we can add here,

i. Text level validations.

ii. Field level validations.

iii. Locators consistency etc.

2. If this test was part of a much larger test set, how would you speed it up?

i. By setting up the Jenkins job, this could be easier to run the entire test set by scheduling at non-working hours.

ii. Going through the Selenium Grid, we can run the larger test sets by pointing to multiple machines.

iii. Using Docker - By creating multiple containers, you can run any volume of test set in the same machine.

3. Sometimes UI tests can fail unpredictably. For example, the page may not have fully loaded before test automation attempts to click a button on a webpage.

How would you improve reliability of these tests?

i. By adding necessary Implicit/Explicit waits in the scripts, the objects which are loading slowly on the page can be handled easily.

ii. Ensure the Test Environment/Application are up and compatible to the scripts to execute, which it avoids the unpredictable test script failures.