# **Empirix Automation Framework**

**Problem Statement:**

Create an **automation test suite** to login **Empirix Cloud Platform** Website(<https://services.empirix.com/>) and verify the following test cases-

* + - The user is able to login with the correct username/password.
    - The user is able to switch the language to Japanese from the profile dropdown and visa Versa.
    - The user is able to see the required tabs and is able to access all of them in English as well as Japanese. i.e. Dashboard, Alert, Test, Variable, Notification.
    - The user is able to go the client tab from the profile dropdown and verify the information is coming.

**Approach:**

The approach for the above-mentioned problem statement is:

In this, **Page object pattern** is used in which for each web page a page class is created. This Page class will identify the Web Elements of that web page and also contains Page methods which perform operations on those Web Elements. **Maven** a build automation tool used to build and manage projects and is very helpful for a project while updating central repository of JARs and other dependencies. Continuous builds, integration, and testing can be easily handled by using maven. **TestNG** framework is implemented for maintaining test cases with various TestNG annotations which helps in providing full control over the test cases and the execution of the test cases. **ExtentReports** is a reporting library for test automation which can be easily integrated with TestNG. **WebEventListener** “listen” to the event defined in the selenium script, the main purpose of using listeners is to create logs. **Java language** is used to write scripts for automation. **Git** is used as a distributed version-control system for tracking changes in source code during development. **Jenkins** is continuous Integration server capable of orchestrating a chain of actions that help to achieve the Continuous Integration process in an automated fashion. It is a powerful instrument that consists of a set of tools designed to host, monitor, compile and test code.

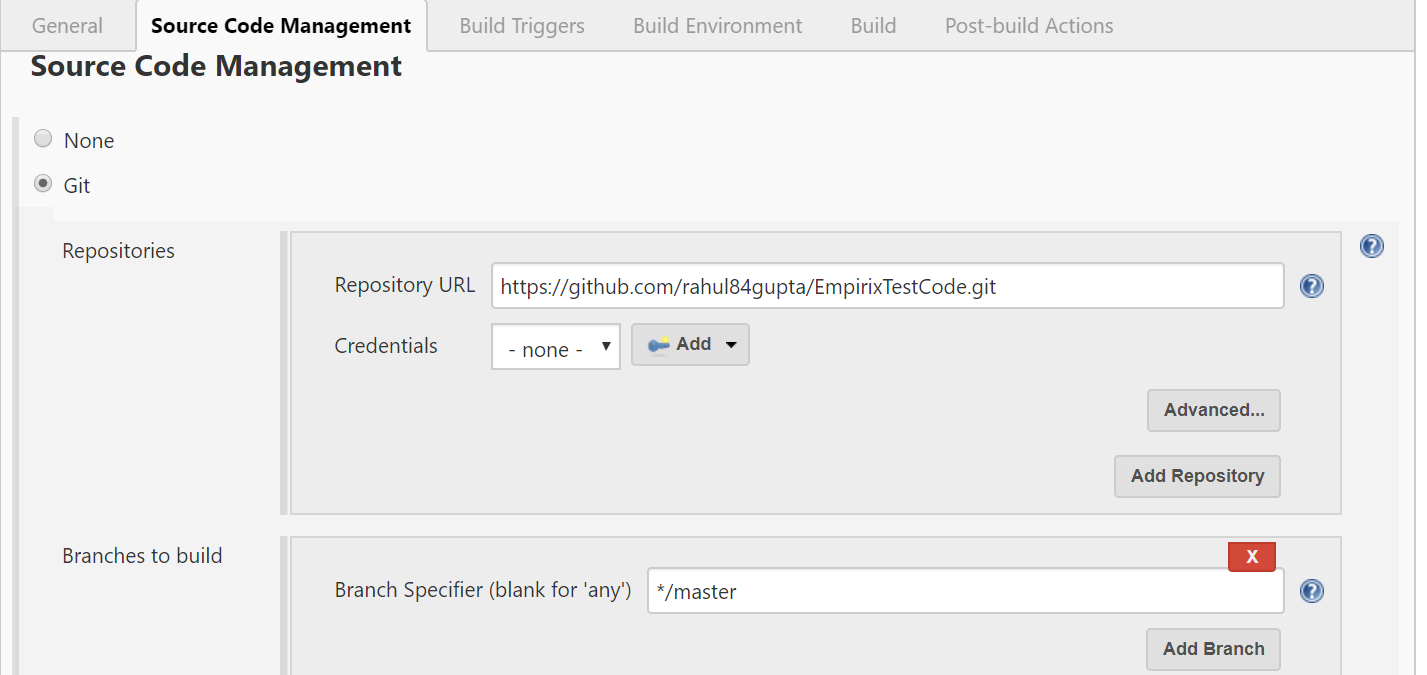
**Jenkins Configuration:**

**Description:** Add project description.

**Source Code Management Tab:**

Repository URL: <https://github.com/rahul84gupta/EmpirixTestCode.git>

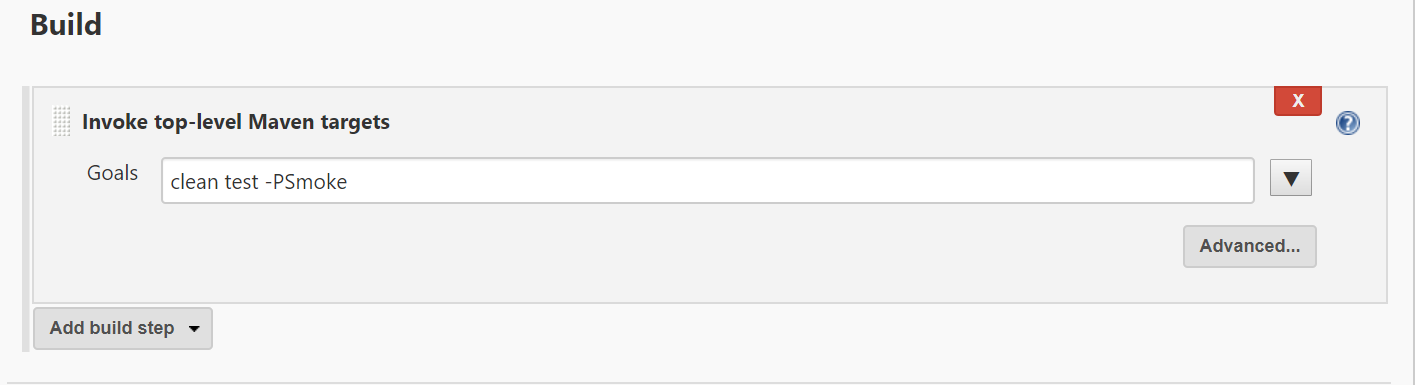
Branch Specifier (blank for 'any'): \*/master



**Build:**

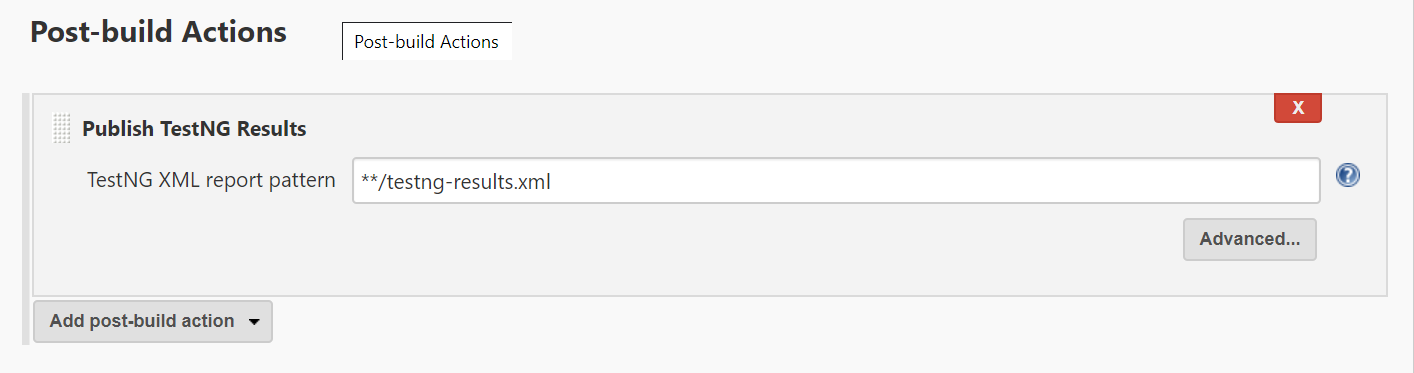
**Select “Invoke top-level Maven targets”** under **“Add build steps”** drop down

Under Goals: **clean test -PSmoke**



**Post Build Actions:**

Select **“Publish TestNG results”** from **“Add Post-build action”** drop down



**How to Run:**

After completing the Jenkins configurations as per mentioned above, click on Build now link present under project details.

For running the test cases parallelly in different browsers, in testng.xml file there is a suite containing two tests, one for each browser i.e. Chrome and Firefox as of now. Under the test tag, Parameters are defined which is consumed in class files(code) & multiple Classes are also defined in which we can define class tag with test class file names for which test cases need to be executed. Testng will run test cases of test class file name which are mentioned under classes tag, we can add/remove class file for which we don’t want to execute the test cases.