Experiment No : 07

Aim : To Setup and Run Selenium Tests in Jenkins Using Maven

Theory :

Jenkins is one of the popular tools for continuous integration, build management, and automation testing. Maven is a popular build automation tool that is widely used for Java projects. The combination of Selenium, Maven, Jenkins integration is the pinnacle of continuous integration and deployment.

As a part of automation testing, we're sure your team comes up with numerous Selenium test automation scripts every other day. As Selenium itself is not used for Continuous integration, Selenium, Maven, and Jenkins integration are leveraged for build management and continuous integration. In this article, we look at the usage of Maven and Jenkins with Selenium.

Maven is a popular build automation tool that is primarily used for Java projects. The advantage of using Maven is that Java libraries and plugins are downloaded on a dynamic basis from the Maven 2 Central Repository.

The dependencies and other essential build-related information are stored in a pom.xml file of the project. Once downloaded, the dependencies are stored in a local cache (.M2 repository), and the same is used for build generation. This simplifies the build process, as any new project dependency has to be added only in pom.xml, i.e., no manual downloading and installation packages are required.

Selenium is a widely used test automation framework for validating web applications across different combinations of browsers, platforms, and devices (or emulators). You can refer to our blogs on how Selenium Grid can be used for automated browser testing.

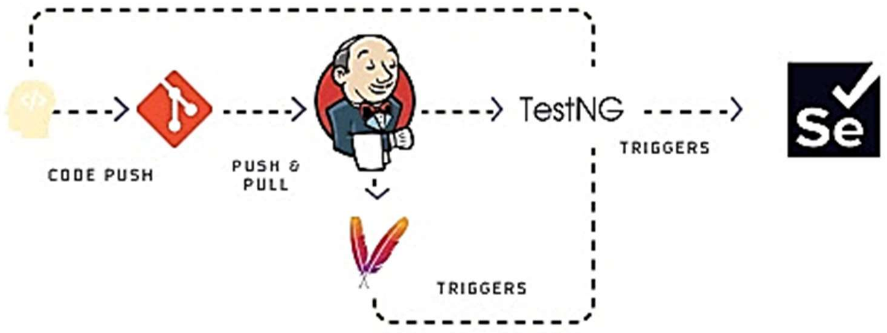
Jenkins is an open-source CI/CD tool that helps in the automation of activities related to build, test, and deployment. Jenkins has an extensive plugin ecosystem, is open-source, and a passionate community – factors that can be attributed to Jenkins' features.

In this Maven and Jenkins with Selenium blog, we would be using TestNG, which is a popular test framework that is inspired by JUnit. It is widely used for testing areas such as functional testing, end-to-end testing, and more.

These are the set of tools that we would be using in the demonstration:

- Maven – Project management tool in Java

- TestNG – Popular test automation framework
- Selenium WebDriver – Library primarily used for automation of browser interactions
- Jenkins – Tool for Continuous Integration (CI) and Continuous Deployment (CD)
- Here is the overall flow of information when Selenium, Maven, and Jenkins are integrated:



- Once the developer pushes code into the repository (e.g., GitHub), a Jenkins build is triggered.

- Maven downloads the dependent libraries & packages and starts performing the build. The information related to the test suite is available in testing.xml, and the same is also used in pom.xml.

- A build goal (e.g., install) for running the automated tests is set. The same is achieved through the maven-surefire-plugin.

- The maven-surefire-plugin tells TestNG to run the tests that are under the annotation @Test.

- Depending on the AUT (application under test) and the browser (& OS combination) on which cross browser tests are performed, the Selenium WebDriver invokes the corresponding browser instance and executes the same automation tests.

- Test results are published in HTML Reports, as the HTML Publisher plugin is used for report generation.

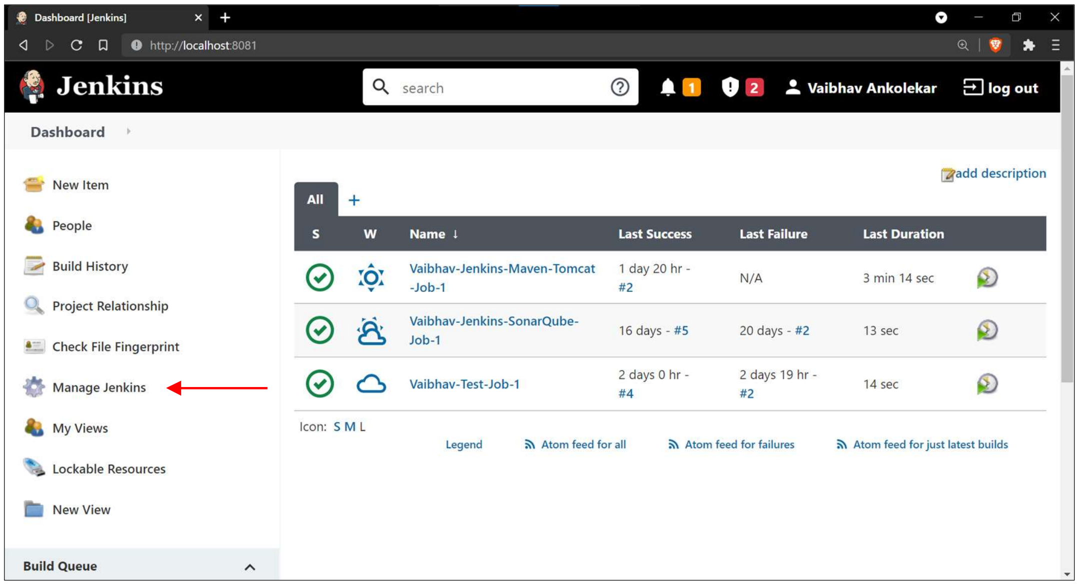
- Even if a single test case has failed, the complete test is marked with status 'Failed.'

Prerequisite :

- **Create a Maven Selenium Project and upload it to Github repository**
- **Install Jenkins and Maven in local machine**

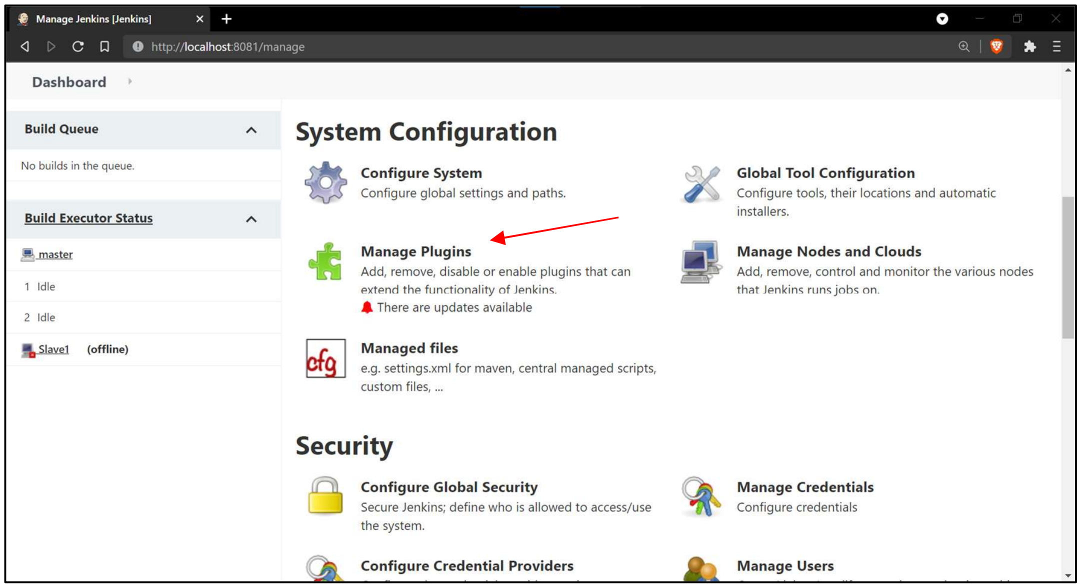
Procedure :

1) Start Jenkins and go to Manage Jenkins.



S	W	Name ↓	Last Success	Last Failure	Last Duration
✓	☀	Vaibhav-Jenkins-Maven-Tomcat-Job-1	1 day 20 hr - #2	N/A	3 min 14 sec
✓	⛅	Vaibhav-Jenkins-SonarQube-Job-1	16 days - #5	20 days - #2	13 sec
✓	☁	Vaibhav-Test-Job-1	2 days 0 hr - #4	2 days 19 hr - #2	14 sec

2) Go to Manage Plugins.



System Configuration

Configure System
Configure global settings and paths.

Global Tool Configuration
Configure tools, their locations and automatic installers.

Manage Plugins
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
There are updates available

Manage Nodes and Clouds
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

Managed files
e.g. settings.xml for maven, central managed scripts, custom files, ...

Security

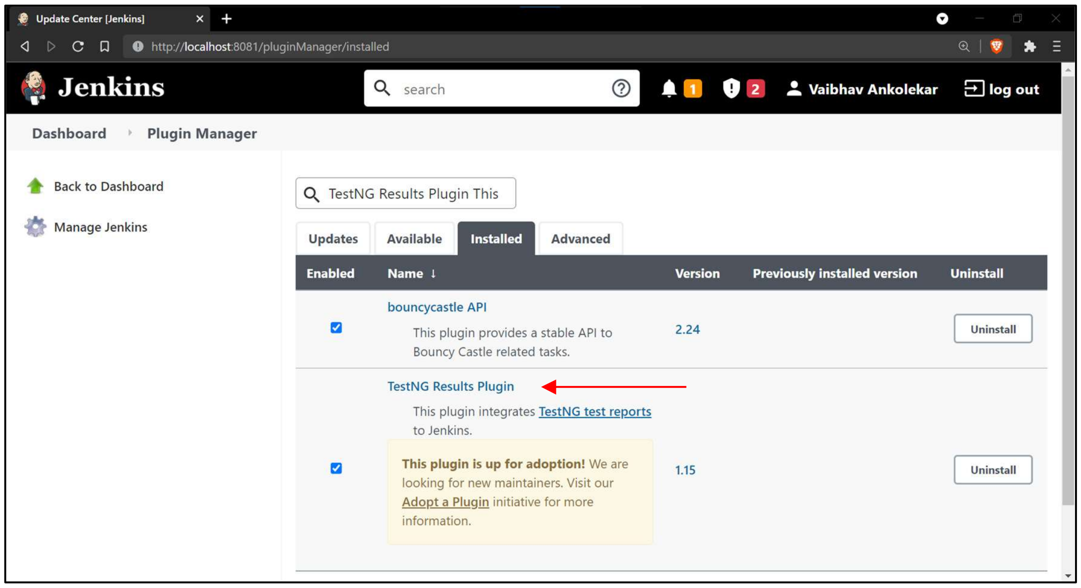
Configure Global Security
Secure Jenkins; define who is allowed to access/use the system.

Manage Credentials
Configure credentials

Configure Credential Providers

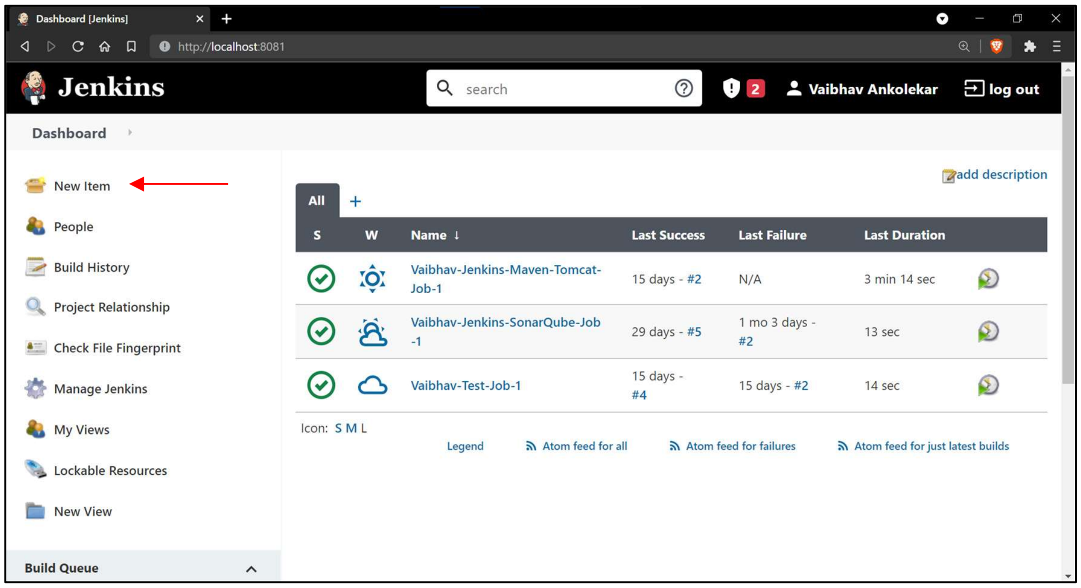
Manage Users

3) Search for TestNG Result plugin and install it.



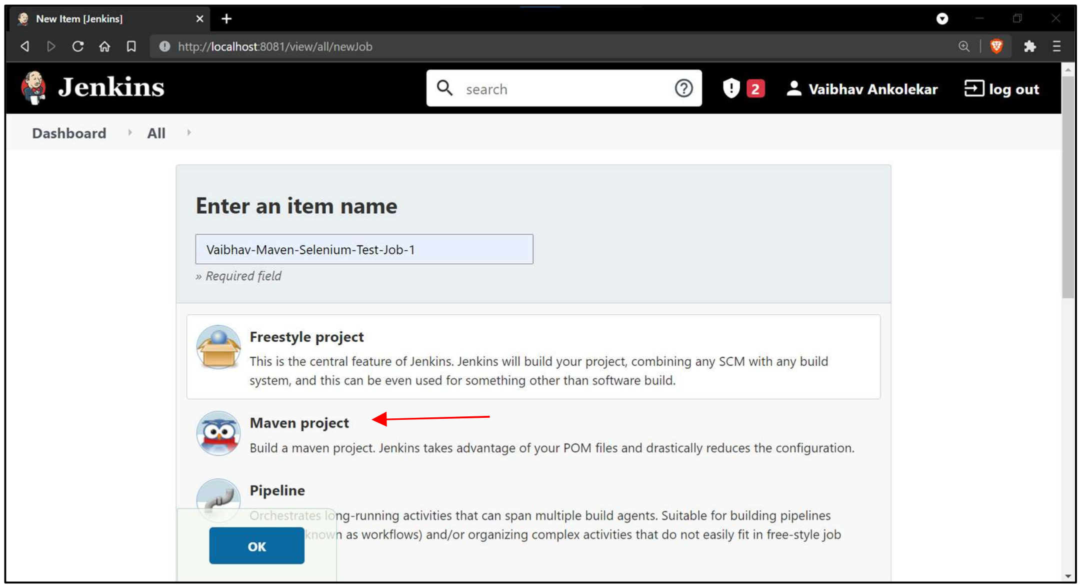
Enabled	Name ↓	Version	Previously installed version	Uninstall
✓	bouncycastle API This plugin provides a stable API to Bouncy Castle related tasks.	2.24		Uninstall
✓	TestNG Results Plugin This plugin integrates TestNG test reports to Jenkins. **This plugin is up for adoption!** We are looking for new maintainers. Visit our Adopt a Plugin initiative for more information.	1.15		Uninstall

4) Now create a new job item.



S	W	Name ↓	Last Success	Last Failure	Last Duration
✓	☀	Vaibhav-Jenkins-Maven-Tomcat-Job-1	15 days - #2	N/A	3 min 14 sec
✓	⛅	Vaibhav-Jenkins-SonarQube-Job-1	29 days - #5	1 mo 3 days - #2	13 sec
✓	☁	Vaibhav-Test-Job-1	15 days - #4	15 days - #2	14 sec

5) **Give it a name and select Maven project.**



New Item [Jenkins]

http://localhost:8081/view/all/newJob

Jenkins | search | Vaibhav Ankolekar | log out

Dashboard › All ›

Enter an item name

Vaibhav-Maven-Selenium-Test-Job-1

» Required field

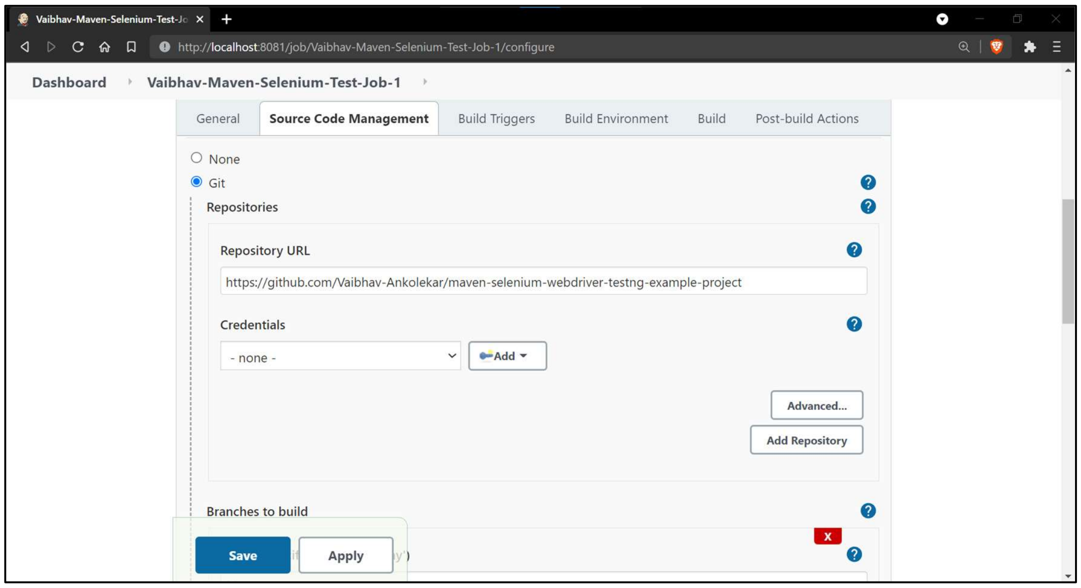
Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines known as workflows) and/or organizing complex activities that do not easily fit in free-style job

OK

6) **In Source Code Management, select Git and add your Github Repo.**



http://localhost:8081/job/Vaibhav-Maven-Selenium-Test-Job-1/configure

Dashboard › Vaibhav-Maven-Selenium-Test-Job-1 ›

General | Source Code Management | Build Triggers | Build Environment | Build | Post-build Actions

None
Git [✓]

Repositories

Repository URL
https://github.com/Vaibhav-Ankolekar/maven-selenium-webdriver-testng-example-project

Credentials
- none -   Add

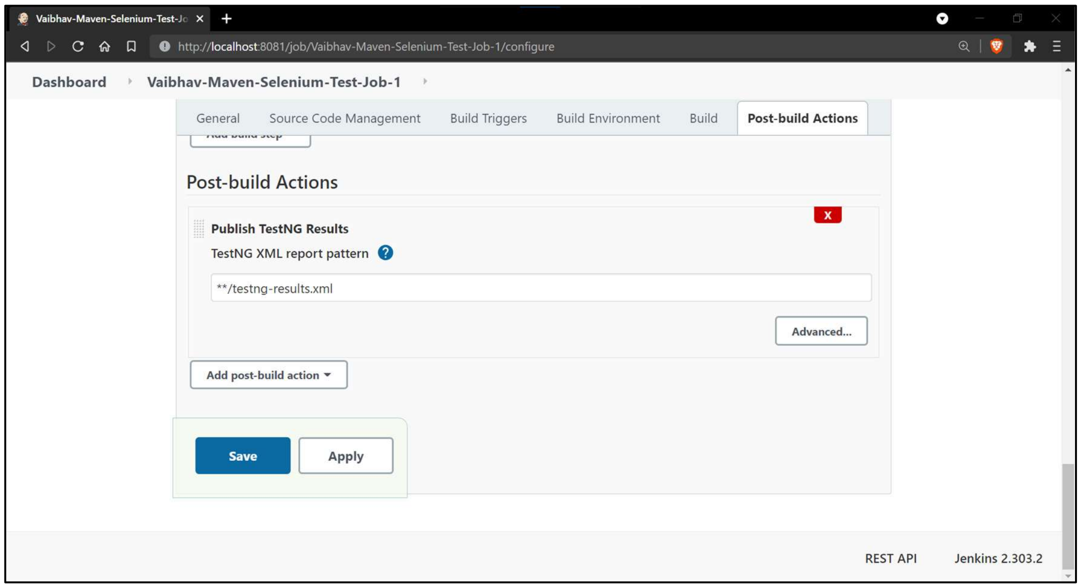
Advanced...
Add Repository

Branches to build

Save   Apply

7) Leave everything as default.

8) In Post Build Actions, add Publish TestNG Results. Click Apply and Save.



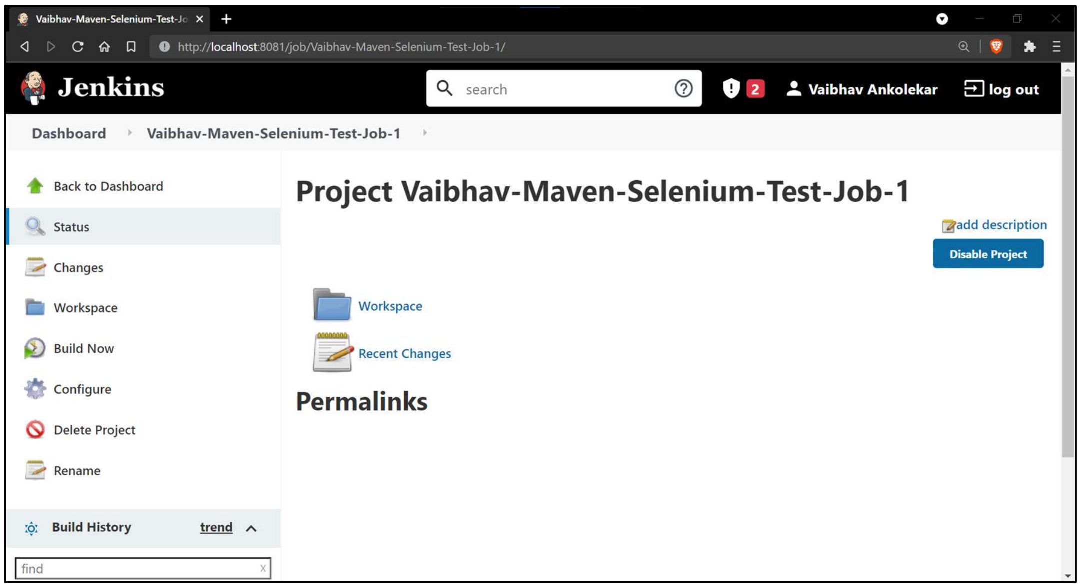
Post-build Actions

Publish TestNG Results

TestNG XML report pattern

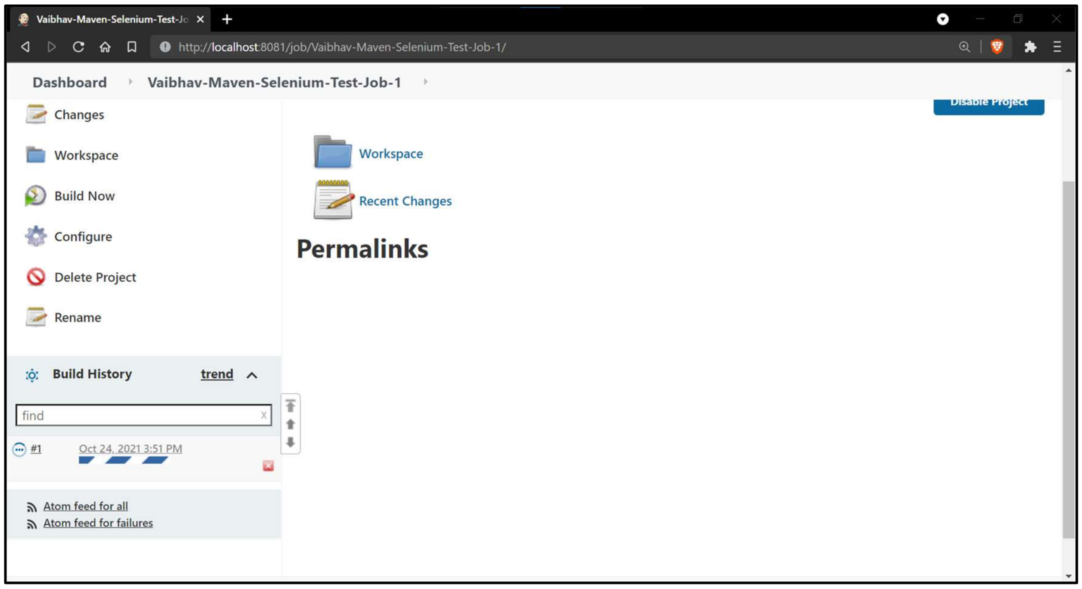
**/testng-results.xml

9) Click on Build Now.

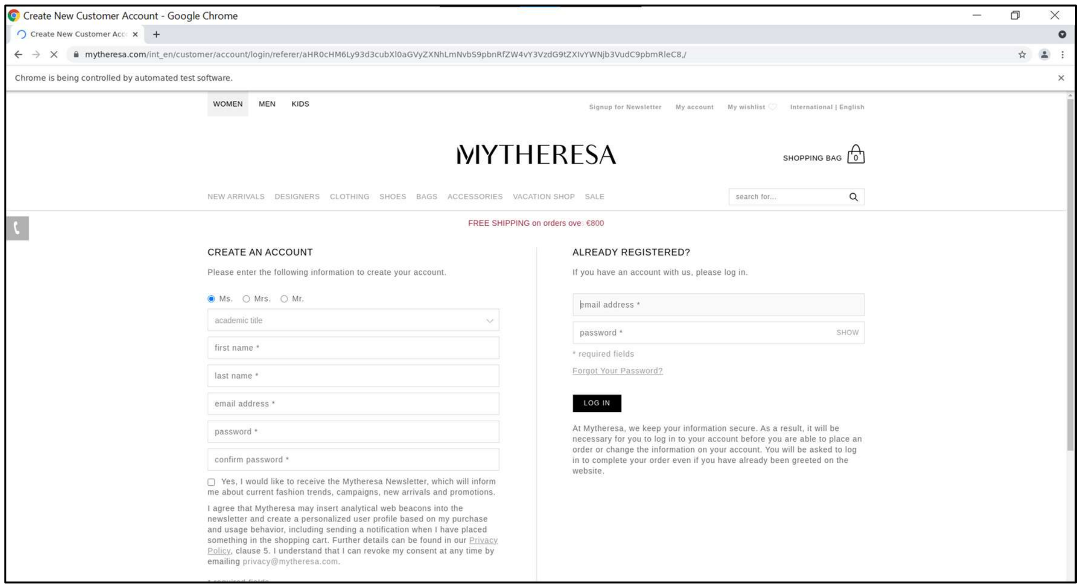


Project Vaibhav-Maven-Selenium-Test-Job-1

10) Your Job will start. Click on Build Number.



11) Now the Selenium will trigger chromedriver. Chromedriver will open new Chrome window and start the selenium testing.



Chrome is being controlled by automated test software.

FREE SHIPPING on orders ove €800

CREATE AN ACCOUNT

Please enter the following information to create your account.

[✓ Ms.] Mrs. Mr.

academic title

first name *

last name *

email address *

password *

confirm password *

Yes, I would like to receive the Mytheresa Newsletter, which will inform me about current fashion trends, campaigns, new arrivals and promotions.

I agree that Mytheresa may insert analytical web beacons into the newsletter and create a personalized user profile based on my purchase and usage behavior, including sending a notification when I have placed something in the shopping cart. Further details can be found in our Privacy Policy, clause 5. I understand that I can revoke my consent at any time by emailing privacy@mytheresa.com.

ALREADY REGISTERED?

If you have an account with us, please log in.

email address *

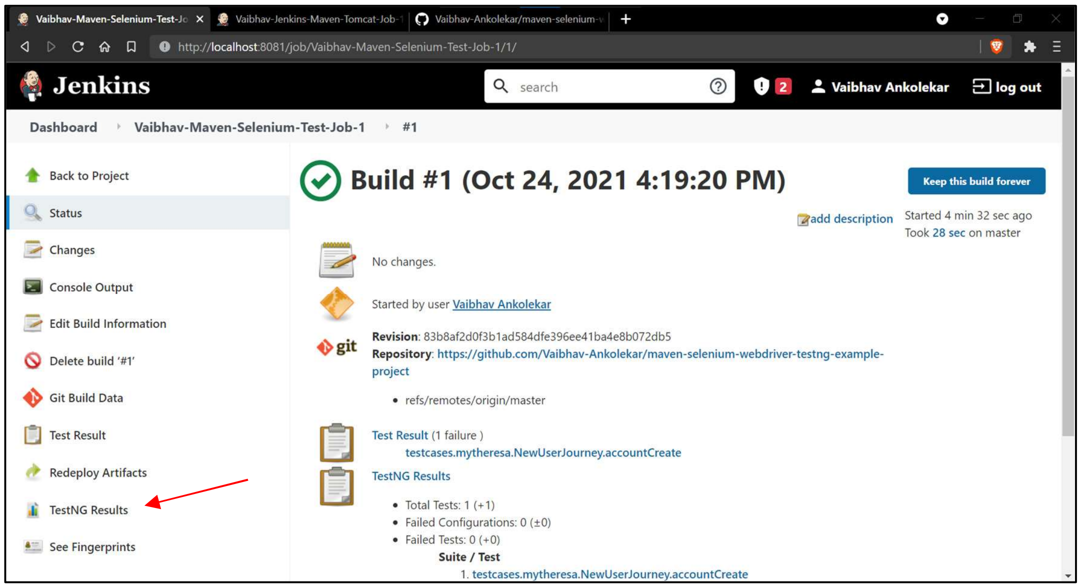
password *

* required fields

Forgot Your Password?

At Mytheresa, we keep your information secure. As a result, it will be necessary for you to log in to your account before you are able to place an order or change the information on your account. You will be asked to log in to complete your order even if you have already been greeted on the website.

12) In your Jenkins job you will see an option of TestNG result, click on it.



Build #1 (Oct 24, 2021 4:19:20 PM)

Started 4 min 32 sec ago
Took 28 sec on master

No changes.

Started by user Vaibhav Ankolekar

Revision: 83b8af2d0f3b1ad584dfe396ee41ba4e8b072db5
Repository: https://github.com/Vaibhav-Ankolekar/maven-selenium-webdriver-testng-example-project

- refs/remotes/origin/master

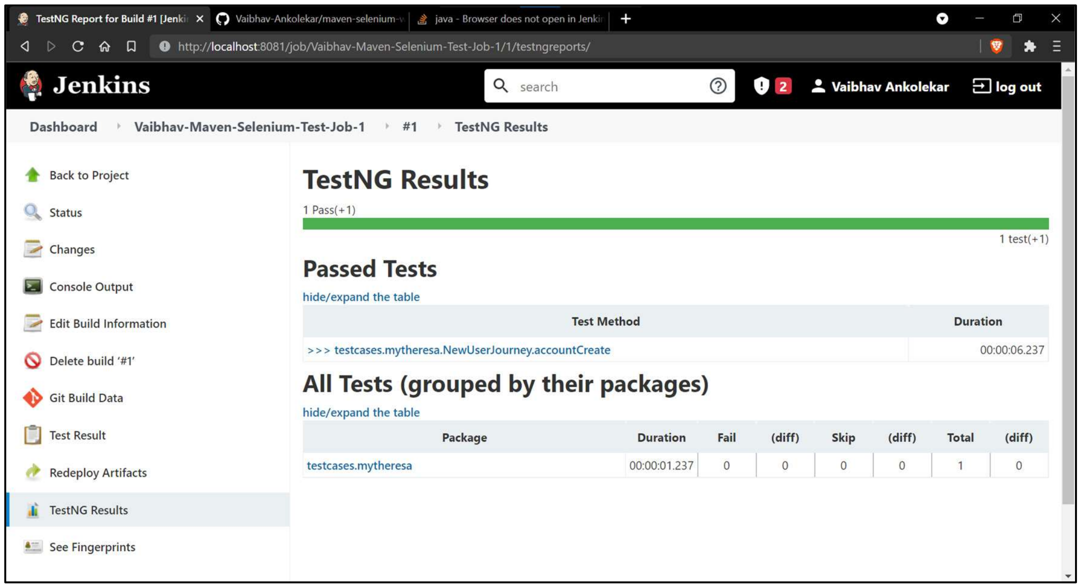
Test Result (1 failure)
testcases.mytheresa.NewUserJourney.accountCreate

TestNG Results

- Total Tests: 1 (+1)
- Failed Configurations: 0 (±0)
- Failed Tests: 0 (+0)

Suite / Test
1. testcases.mytheresa.NewUserJourney.accountCreate

13) You Selenium Test Result is published.



TestNG Results

1 Pass(+1)

1 test(+1)

Passed Tests

hide/expand the table

Test Method	Duration
>>> testcases.mytheresa.NewUserJourney.accountCreate	00:00:06.237

All Tests (grouped by their packages)

hide/expand the table

Package	Duration	Fail	(diff)	Skip	(diff)	Total	(diff)
testcases.mytheresa	00:00:01.237	0	0	0	0	1	0

Conclusion :

Running Selenium tests in Jenkins allows you to run your tests every time your software changes and deploy the software to a new environment when the tests pass. Jenkins can schedule your tests to run at specific time. You can save the execution history and Test Reports. Thus, we have successfully setup and run Selenium Tests in Jenkins Using Maven