An application's stability and reliability depend on the depth of the test
the last couple of decades, automated testing, coupled with cloud-nati
for faster and more extensive testing.

As an integral part of the Agile pipeline, automation testing is essential
In particular, CI/CD (Continuous Integration / Continuous Deployment)
and testers to speed up development and testing so that their product
driven.

The implementation of CI/CD requires several tools, among which Jenkins is the most popular. This article will explore Jenkins, its features, and how it can facilitate test automation.

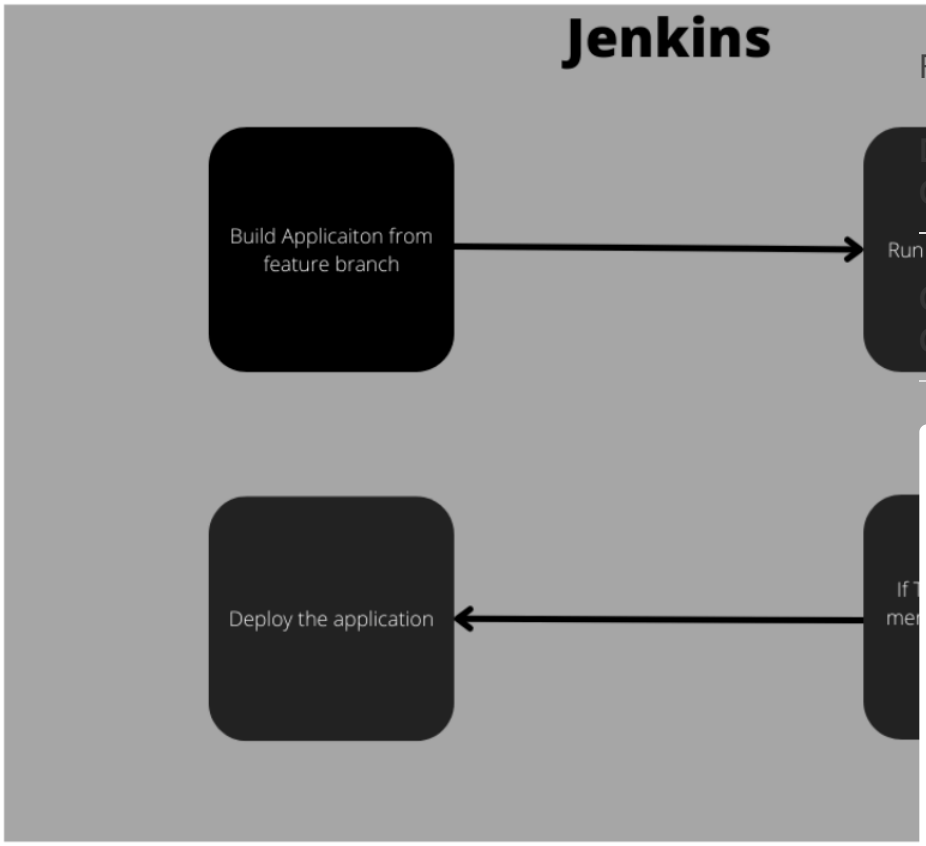
Table of Contents

What is Jenkins?

CI/CD is the way to continuou
any change is made using uni
and deploy the successful bui

To achieve this CI/CD, Jenkins
that is self-contained and con
automate the build process.

Jenkins can be used to automate building an application, running test
can be easily installed through native system packages, Docker, or eve
a Java Runtime Environment (JRE) installed.



Why use Jenkins for Test Automati

Jenkins is a popular CI orchestration tool. It provides numerous plugin [automation tools and frameworks] into the test pipeline. When it comes to Test Automation, Jenkins provides plugins that help run test suites, gather dashboard results, and provide details on failures.

- **Runs Automated Test Suites**: Jenkins provides plugins for various test frameworks like Selenium, Cucumber, Appium, etc. These can be integrated into CI pipelines to run automated tests for every build.
- **Summarizes the results**: Most plugins also summarize the test results and display them as an HTML page.
- **Provides Trends**: Jenkins keeps track of results and displays them as a trend graph. This offers a better view of how the tests have fared in the past.
- **Display details on Test Failures:** Test results are tabulated, and failures are logged with the test results.

Advantages of using Jenkins

Jenkins is a very popular open-source CI/CD build tool, with a wide variety of Plugins available. Here are some of its core advantages:

- Jenkins can be used to build pipeline/workflows
- It can be integrated with any source management tool
- It can be Integrated with Chat tools like Teams, and Slack for notifying the status of build jobs
- Jenkins can be used to Schedule builds periodically or trigger ba Source code repository
- Jenkins can be used to show the HTML report of AutomatedTest
- Jenkins can be used to send emails with Test Execution Report

Limitations of using Jenkins

Some of the limitations faced while using Jenkins for Test Automation

- Jenkins, being a community-driven tool has a large number of co single tool integration can make the options unclear for users.
- Although there are plugins supporting Docker integration, Jenkins still has a long way to go in this domain.
- Jenkins does not itself host a cloud-based service. Users need to rely on Jenkins provided as a service by cloud service providers like AWS, GCP, Azure, Cloudbees, etc.
- Jenkins involves a steep learning curve as the configuration is not straightforward. Too many options and configurations can be complicated for newbies.
- Lesser flexibility in using plugins as they are not customizable.
- Lack of authentication and authorization rules and options.
- Most community-driven plugins do not have clear documentation on how to use them.

- Test Automation forms the crux of the CI/CD process. Despite its issue, Jenkins is the most effective tool for implementing automation testing in CI/CD pipelines. The plethora of plugins that Jenkins offers and the strong community support makes it a powerful ally in automation efforts, no matter the software being tested.

When to use Jenkins for Test Automation?

With numerous plugins on offer, Jenkins meets all testing automation needs in most cases. Most popular testing tools can be easily integrated with Jenkins.

For simple workflows containing a streamlined CI process, Jenkins wo automation. Test scripts can be called, and reports can be generated fr

It would be best to use Jenkins when the plugins are from a trusted so of the testing tool itself. This would avoid unruly behavior of the jobs a make integrations much more seamless.

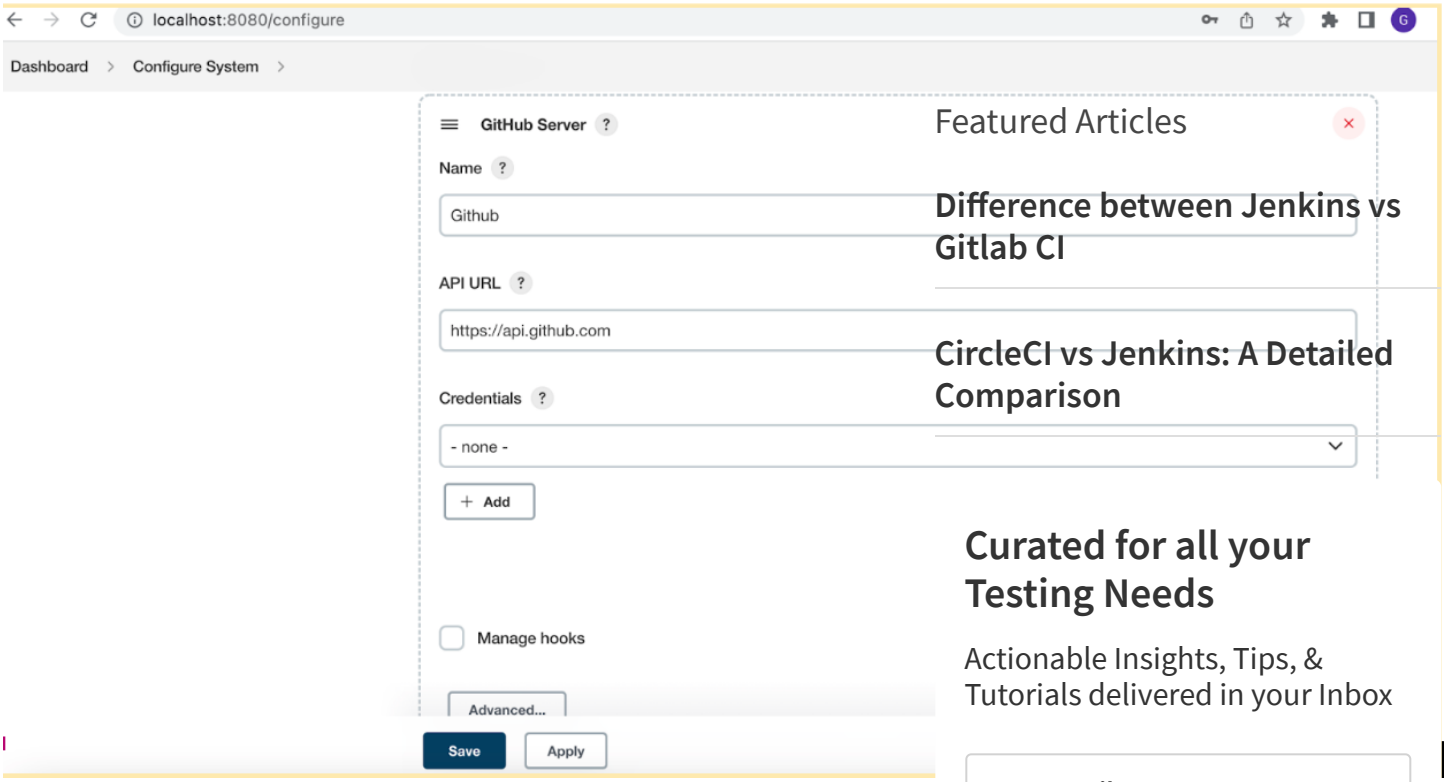
Integrating Jenkins with Git

In order for Jenkins to build/run an application, Jenkins needs to be c management system like Git (other systems include Azure, Bitbucket, e

> **Note:** Refer to this GitHub demo repository, which can be cloned/forked for practice purposes.

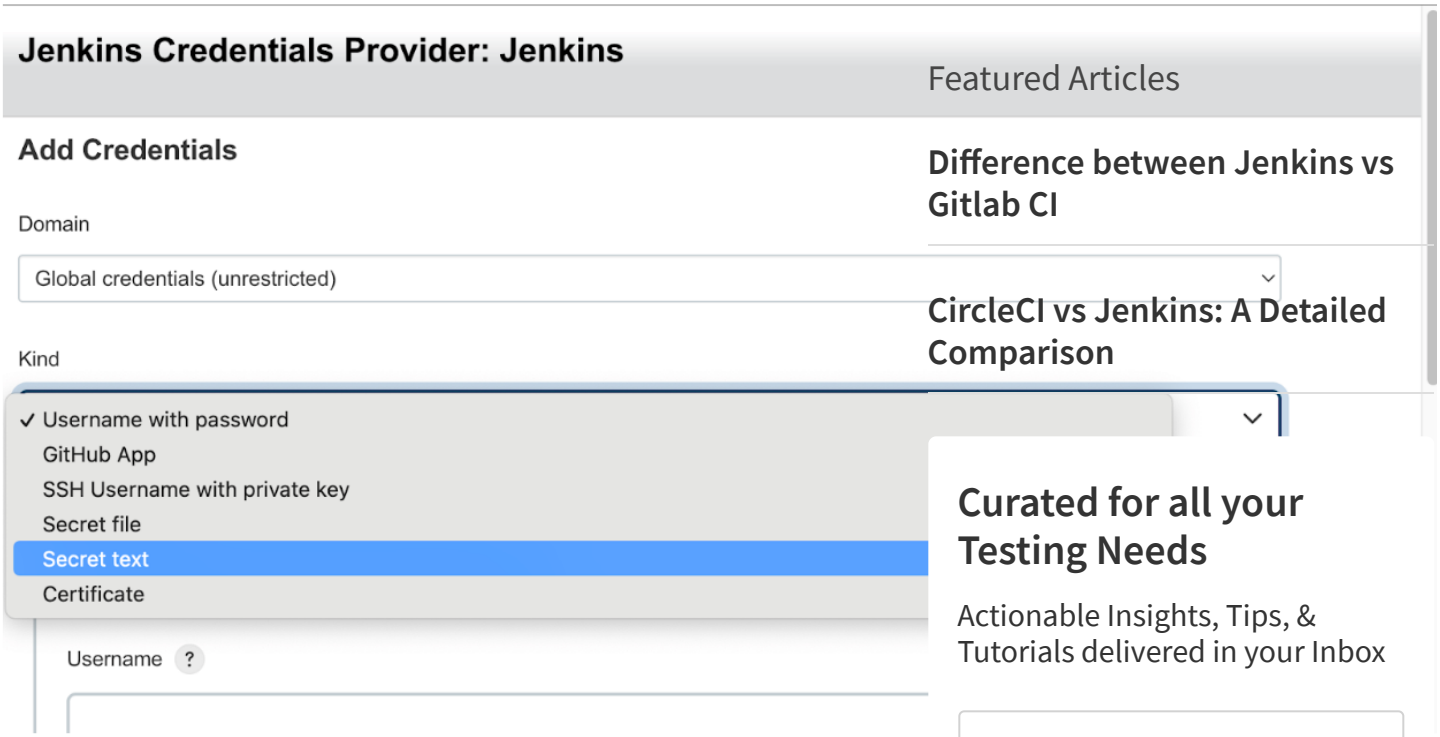
Step 1 – Navigate to Configure Systems in the Jenkins using the below steps:

- Navigate to the GitHub Server section
- Enter the Name and API URL as shown below
- Click on Add button under Credentials



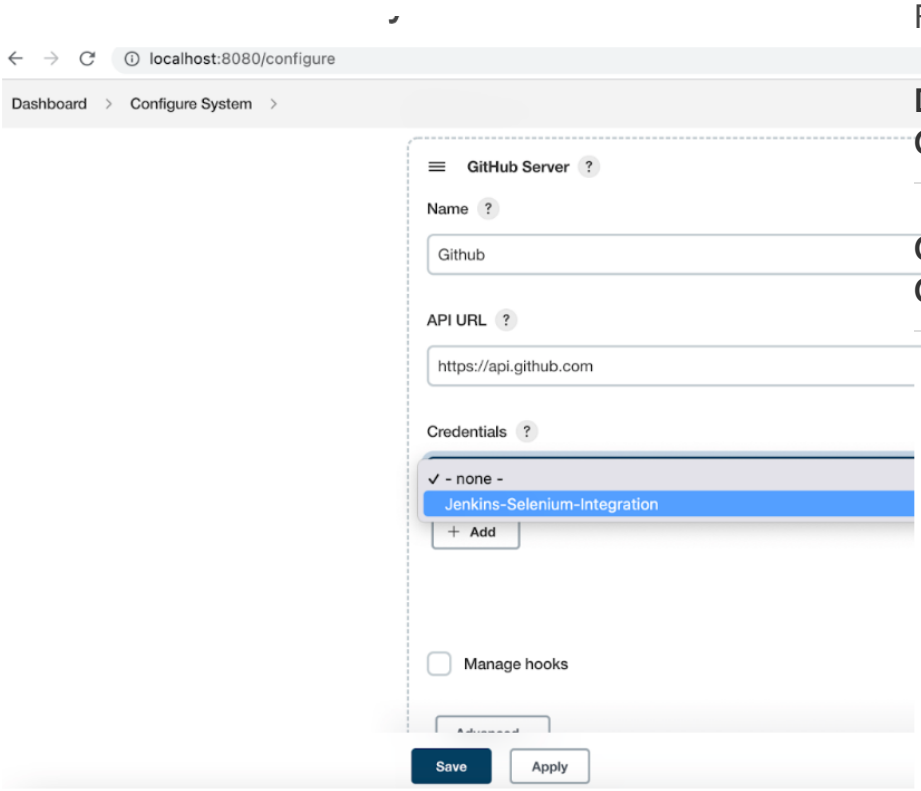
Step 2 – Select the Secret Text under Kind drop-down as seen in the ste
what kind of authentication you need Jenkins to perform while accessi

- You need to get the Personal access token from Git and Enter tha
follow this guide to get a Personal access token in Git.



- Enter the secret key (Git personal access token) and ID as shown
- Click on Add button

Step 3 – Select the newly added credential



Step 4 – As the final step, click on Save on the Configure System page t setup

Setup Job to Run Automated Test with Jenkins

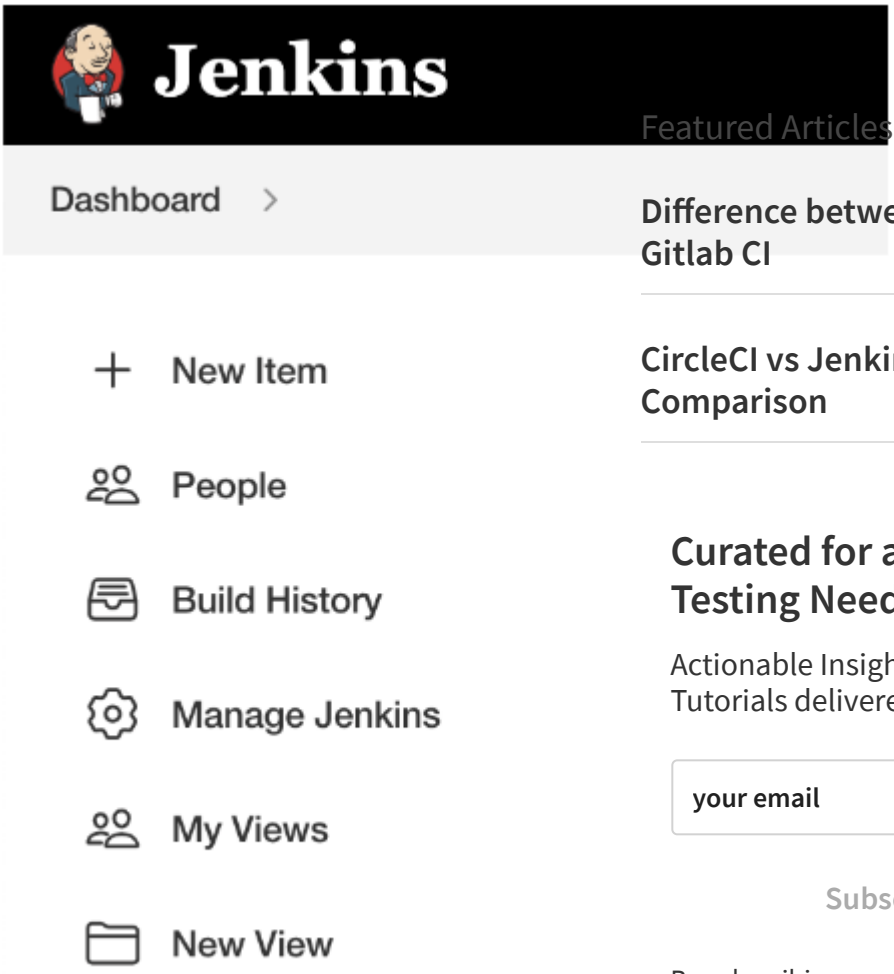
To perform a build deployment or to run automated tests, you would need to create a job in Jenkins.

The job typically contains

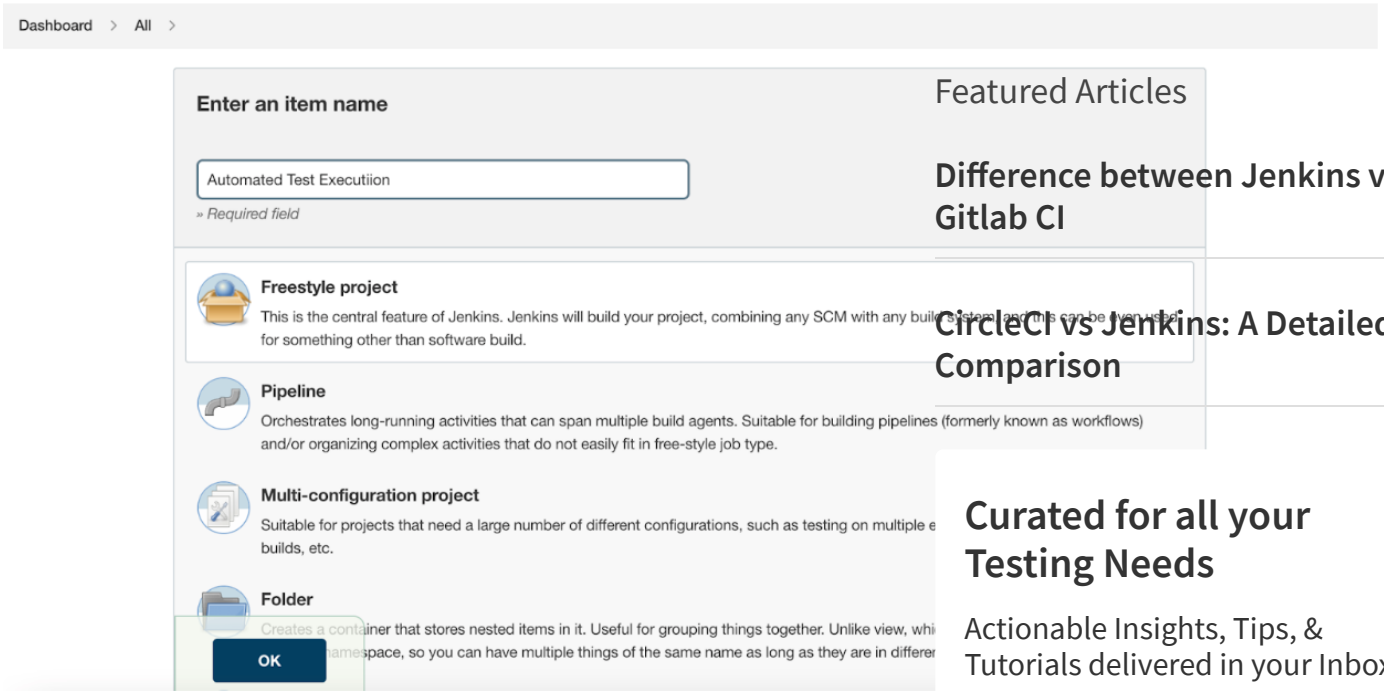
- Source code repository link to the checkout
- Command to run
- Any post-run actions like collecting test reports or publishing build artifacts

Let’s set up a job by following the below steps

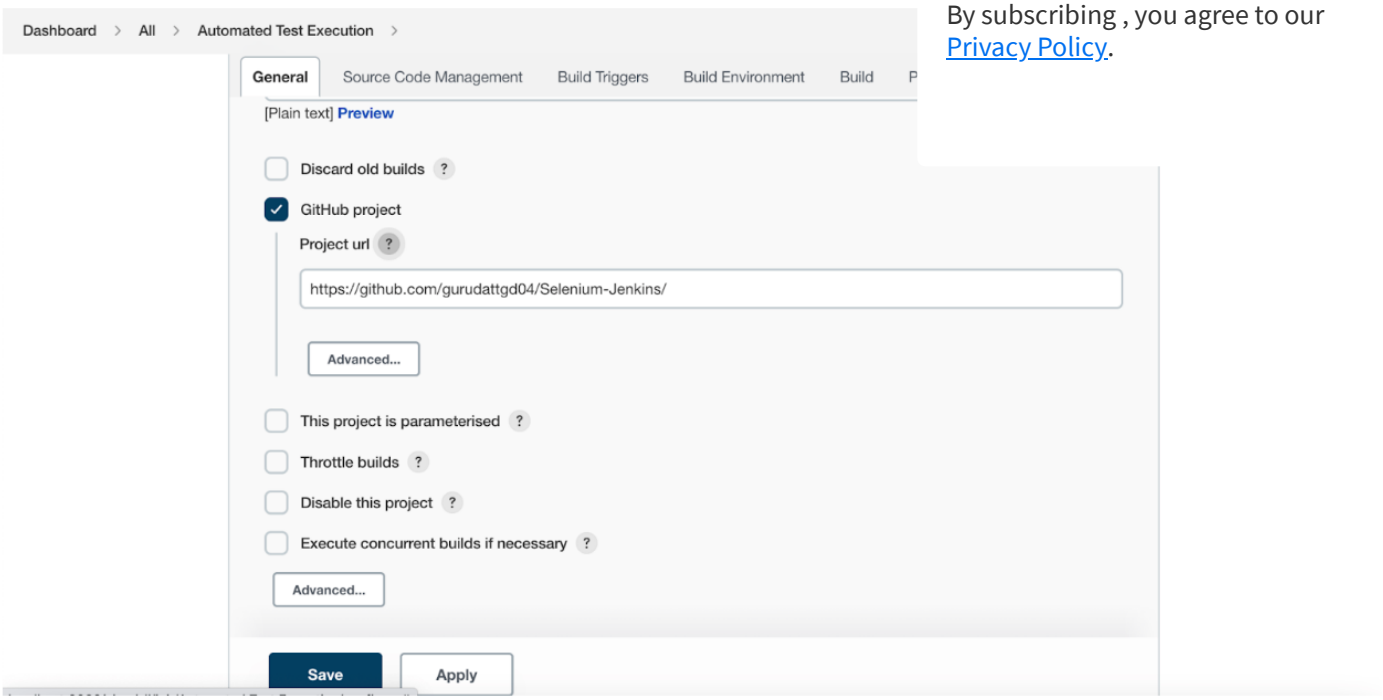
Step 1 – Click on New Item in the Jenkins dashboard



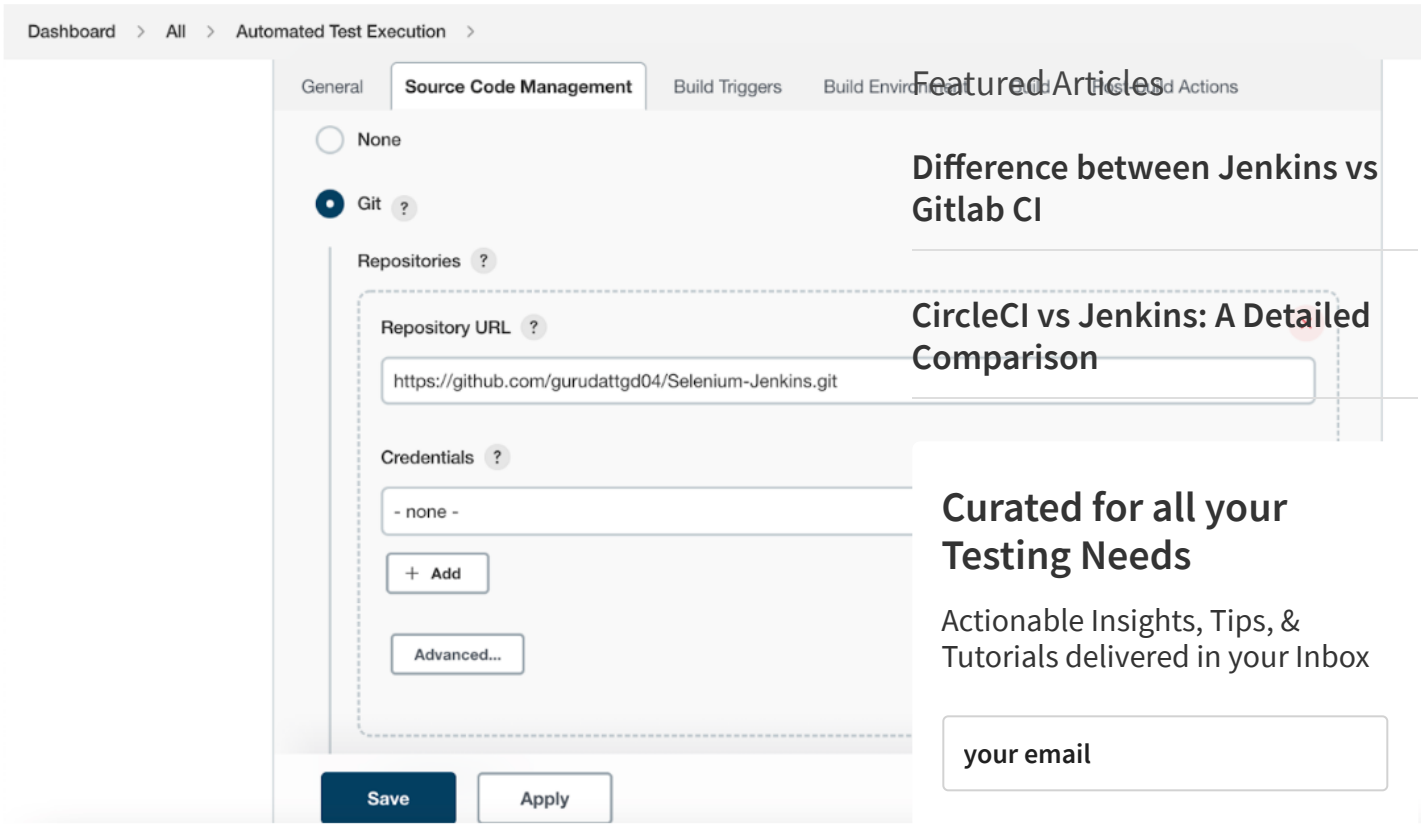
Step 2 – Next enter the Job name, Select Job type as Freestyle project, and click on OK button



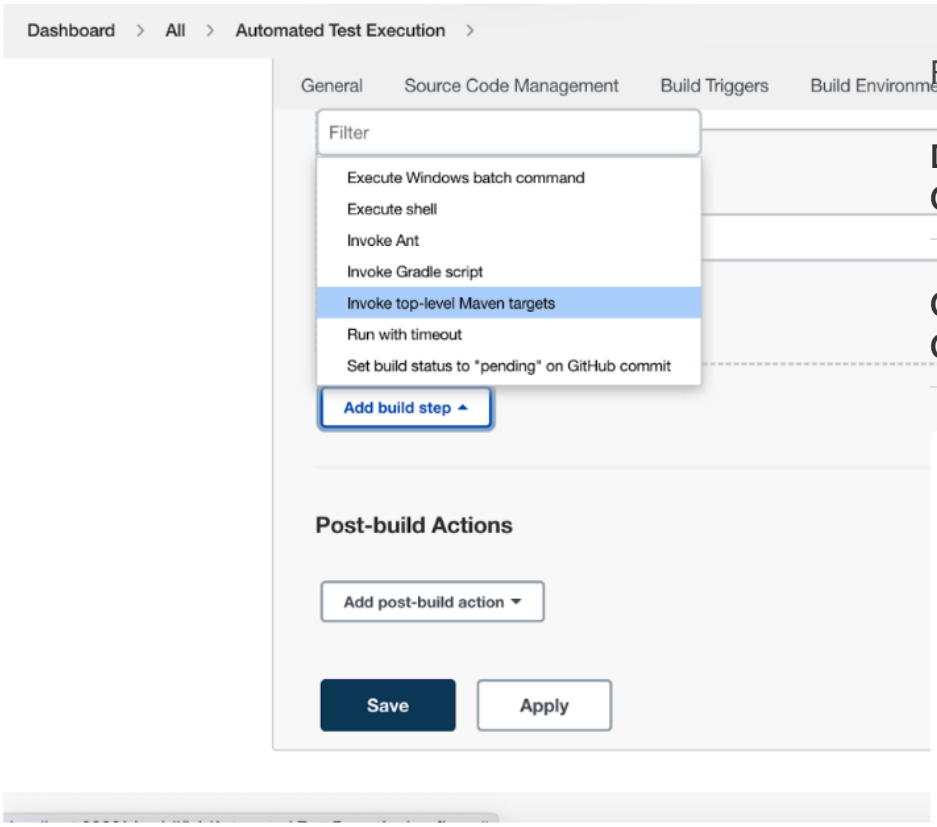
Step 3 – The new Job is created which needs to be configured now by S
>> Entering the root URL of the repository



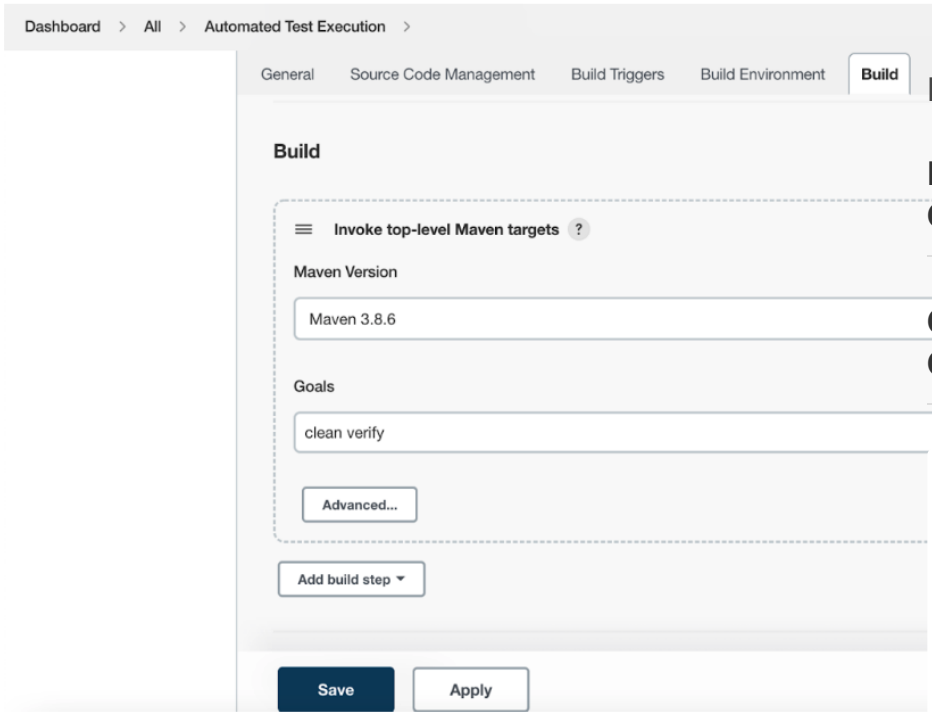
Step 4 – You need to specify the git repo URL by selecting the Git radio button under the Source Code Management section



Step 5 – Next step is to add Build Command and since the project used hence selecting the Invoke top-level Maven targets option.



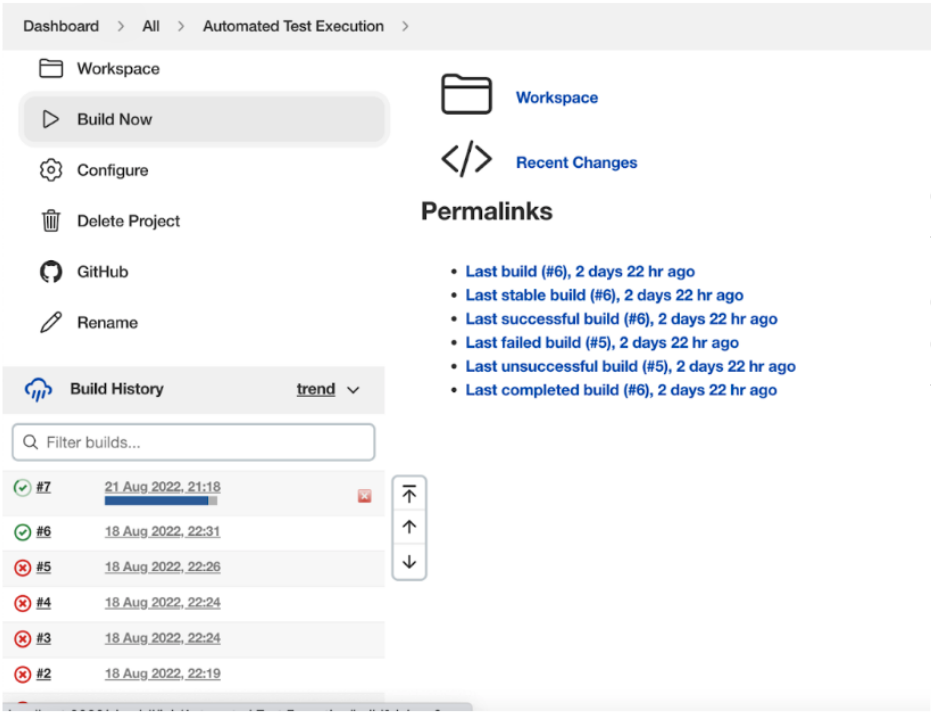
Step 6 – Select the Maven version and enter the Maven command unde **verify.**



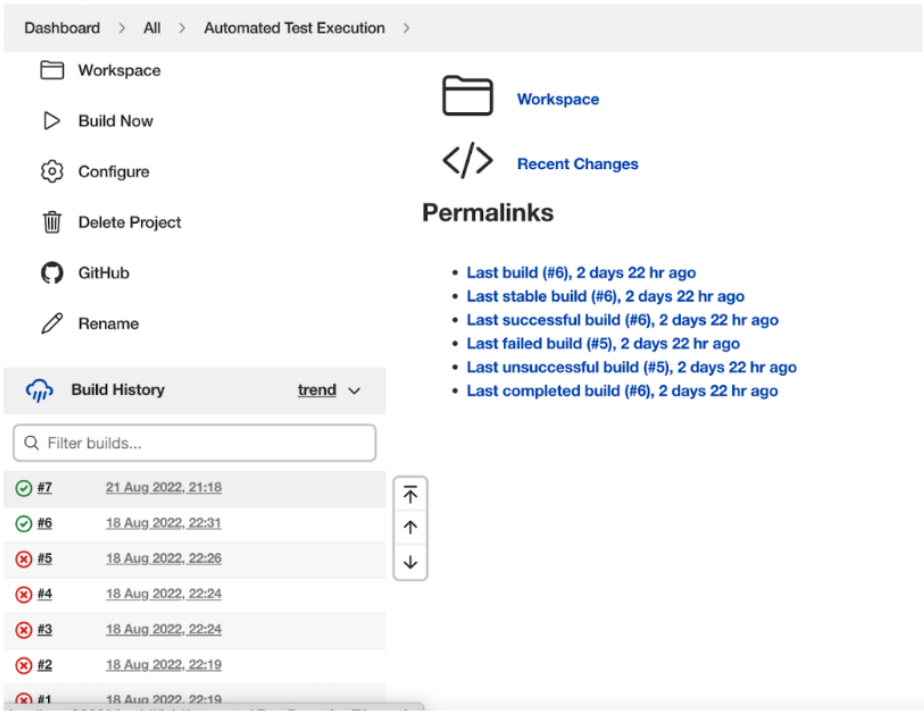
Step 7 – Save the Job configuration by clicking on the Save button

Executing the Job

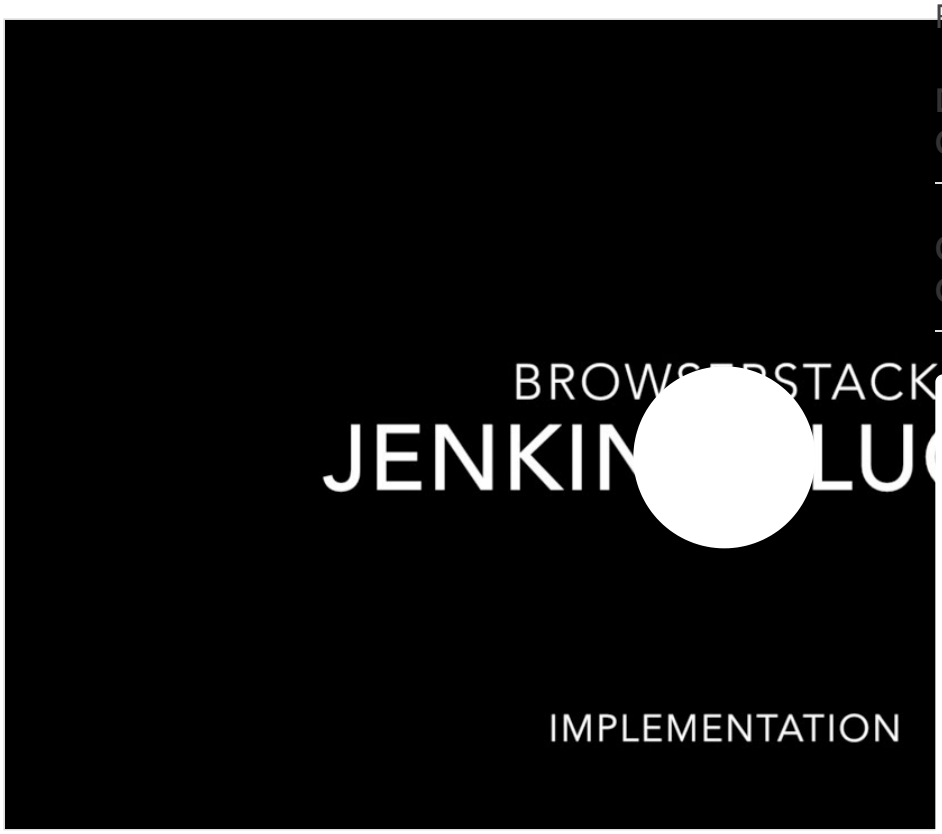
To execute the job Click on Build Now, which will trigger a new build as seen below.



Once the build is completed, you can see the console output or the HT on the Build number (In this case its **#7**)



Read More: Learn how to integrate BrowserStack Automate with Jenkins



Popular plugins for test automation:

- **Selenium Plugin:** This plugin helps turn the Jenkins cluster into a Selenium Grid3 cluster and carry out tests on the heterogeneous Jenkins clusters.
- **Cucumber-reports:** This plugin publishes the reports of Cucumber runs in pretty HTML format.
- **Browserstack Plugin**: Test suites can be integrated and run from the CI server on the Browserstack real device cloud.
- **TestComplete**: TestComplete tests can be integrated with Jenkins Freestyle jobs and pipelines with this plugin.
- **Kalaton TestOps:** This plugin helps to run Kalaton Studio on Jenkins. It also automatically handles the download and deployment of Kalaton Studio.

Apart from these, many tools help with the analysis and reporting of tests. Some of these are **Tests Analysis Plugin, Test Results Analyzer, bootstrapper-multi-test-reports-plugin, Junit publisher plugin**, etc.

Try Continous Testing with Jenkins

BrowserStack's cloud Selenium grid of 3000+ real browsers and devices offers integrations with multiple CI/CD tools, including Jenkins. **Sign up for free** and start running automated tests on the latest devices and browsers on the cloud.

Automation Testing | CI CD Tools | DevOps

Was this post useful? Yes, Thanks

Featured Articles

Difference between Jenkins vs Gitlab CI

CircleCI vs Jenkins: A Detailed Comparison

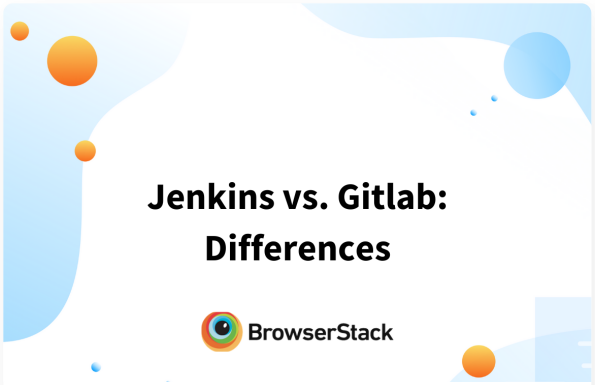
Curated for all your Testing Needs

Actionable Insights, Tips, & Tutorials delivered in your Inbox

your email

Subscribe

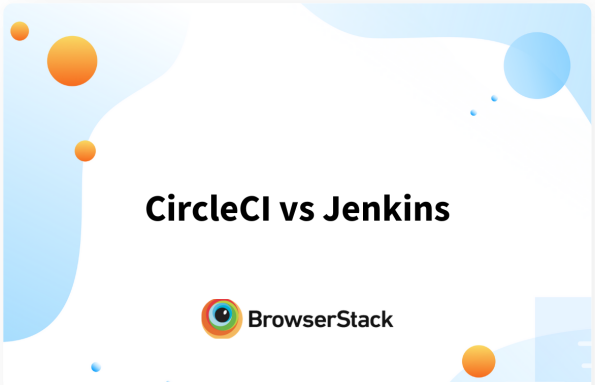
Related Articles



Difference between Jenkins vs Gitlab CI

Evaluating Jenkins vs GitLab Ci? Can GitLab replace Jenkins or vice versa? Read this article to unde...

Learn More



CircleCI vs Jenkins: A Detailed Comparison

Check the differences between CircleCI and Jenkins - two widely used CI/CD integration tools and the...

Learn More

Continuous Integration with Jenkins

With this step-by-step tutorial, understand the key benefits of Jenkins CI and how to create a conti...

Learn More

Contact Us

Curated for all your Testing Needs

Actionable Insights, Tips, & Tutorials delivered in your Inbox

your email

Subscribe

By subscribing , you agree to our Privacy Policy.