INDUSTRIAL TRAINING REPORT

ON

AWS AND DEVOPS

AT

SV GLOBAL SERVICES INDIA PRIVATE LIMITED

Submitted in the partial fulfillment of the

Requirement for the award of

IN

DIPLOMA IN COMPUTER SCIENCE AND ENGINEERING

Submitted by

GASIKANTI AKSHITHA

(22054-CS-004)

GOVERNMENT INSTITUTE OF ELECTRONICS

(Affiliated to SBTET-TS, HYD), EAST MARREDPALLY

SECUNDERABAD, 500026

GOVERNMENT INSTITUTE OF ELECTRONICS

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitle **"AMAZON WEB SERVICES (AWS) AND DEVOPS"** has been carried by **GASIKANTI AKSHITHA** bearing PIN **22054-CS-004** in partial fulfilment for the award of DIPLOMA IN COMPUTER SCIENCE ENGINEERING to the STATE BOARD OF TECHNICAL EDUCATION AND TRAINING, Hyderabad at GOVERNMENT INSTITUTE OF ELECTRONICS, East Marredpally , Hyderabad during the Academic year 2024-2025

INTERNAL EXAMINER **EXTERNAL EXAMINER**

HEAD OF DEPT **PRINCIPAL**

ACKNOWLEDGEMENT

The Immense satisfaction and delight that accompanies the successful completion this task would be incomplete without the mention of the people whose constant guidance and encouragement have crowned our efforts with success.

Our sincere regards to **MRS. P. ANNAPURNA [M.E], PRINCIPAL, GOVERNMENT INSTITUTE OF ELECTRONICS** for this encouragement and support during all the stages of the project.

We express our sincere gratitude to **MR S.P VENKAT REDDY [M TECH] , HEAD OF THE DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING, GOVERNMENT INSTITUTE OF ELECTRONICS** for his interminable support and encouragement and providing us with outstanding facilities for the successful completion of our project.

It is our privilege and pleasure to express my found sense of respect, gratitude and indebtedness to our guide **MRS. V. KALPANA [L/CSE]**, for her indefatigable inspiration, cogent discussion, constructive criticisms and encouragement throughout the dissertation work.

We are also thankful to other staff members for their support during the course of the project and in completing the project successfully.

We thank all those who directly and indirectly helped us in the successful completion of the project in time

DECLARATION:

We the undersigned declare that the project report entitled **"INDUSTRIAL TRAINING REPORT"** written and submitted by us in an original work done under the guidance of **Sri. T. NITESH.** The matter here in is not reproduced from any other source. I hereby declare that this project was outcome of efforts and is not been submitted to any other university for the award of any degree or diploma.

GASIKANTI AKSHITHA **22054-CS-004**

ABSTRACT

This industrial training report presents a comprehensive overview of the practical experience and technical knowledge gained during the six-month training period at SV Global Services India Pvt. Ltd., Hyderabad, focusing on the core areas of AWS and DevOps. The project emphasizes the significance of modern software development methodologies and deployment strategies through the implementation of DevOps tools such as Git, Jenkins, Maven, Docker, Kubernetes, and Selenium.

The report also explores the role of infrastructure as code (IAC), containerization, and monitoring in streamlining IT operations and improving development productivity. Through real-time application of tools and collaborative practices, the training enhanced the team's skills in project automation, deployment, and infrastructure management, aligning with current industry standards.

This project marks a significant step towards bridging academic knowledge with industrial practices, preparing students for real-world DevOps and cloud engineering roles.

INDEX

SNO	CONTENTS	PAGE .NO
1	INTRODUCTION	9-10
	1.1 DevOps	11
	1.2 DevOps Technologies	12
2	Introduction to Jenkins	13
	2.1 Key Features of Jenkins	14
	2.2 Jenkins Applications	15-16
3	Introduction to Maven	17
4	Installing Jenkins and Mavens	18-20
5	Building a Maven project in Jenkins	21-22
6	Output	23
7	Conclusion and References	24

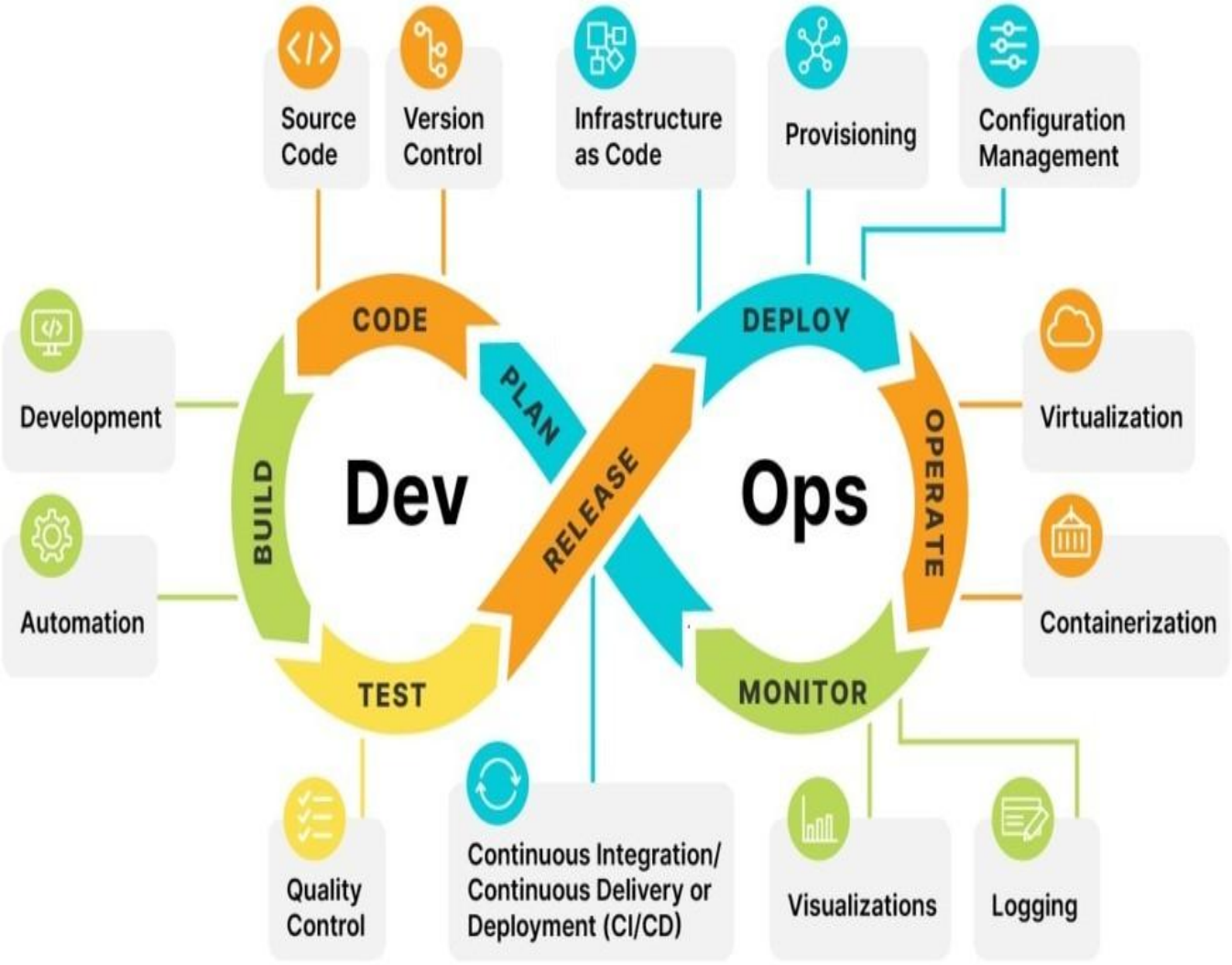
INTRODUCTION TO DEVOPS

Software is the backbone of modern technological advancement, powering everything from mobile applications to complex enterprise systems. In its essence, software refers to a set of instructions or programs that enable computers and other devices to perform specific tasks or functions. From simple utilities that organize daily tasks to sophisticated algorithms driving artificial intelligence, software encompasses a vast array of applications that shape our digital world.

As technology continues to evolve, so does the demand for innovative software solutions that streamline processes, enhance productivity, and enrich user experiences.

Understanding the principles of software development, including design, coding, testing, and maintenance, is essential for navigating today's digital landscape and driving meaningful change through technology.

Continuous growth in software is not merely a trend but a fundamental aspect of the industry's evolution. With technology advancing at an unprecedented pace, software development constantly expands its horizons to meet ever-changing demands and challenges. For this continuous growth of software DevOps also plays a crucial play. We use DevOps in almost every sector for better utilization and better productive


Source Code
Version Control
Infrastructure as Code
Provisioning
Configuration Management
CODE
PLAN
BUILD
Dev
RELEASE
DEPLOY
OPERATE
Ops
TEST
MONITOR
Development
Automation
Virtualization
Containerization
Quality Control
Continuous Integration/ Continuous Delivery or Deployment (CI/CD)
Visualizations
Logging

DEVOPS

DevOps an amalgamation of "development" and "operations," represents a transformative approach to software development and IT operations. It emphasizes collaboration, communication, and integration between traditionally siloed teams, aiming to streamline the software delivery process and improve overall organizational performance.

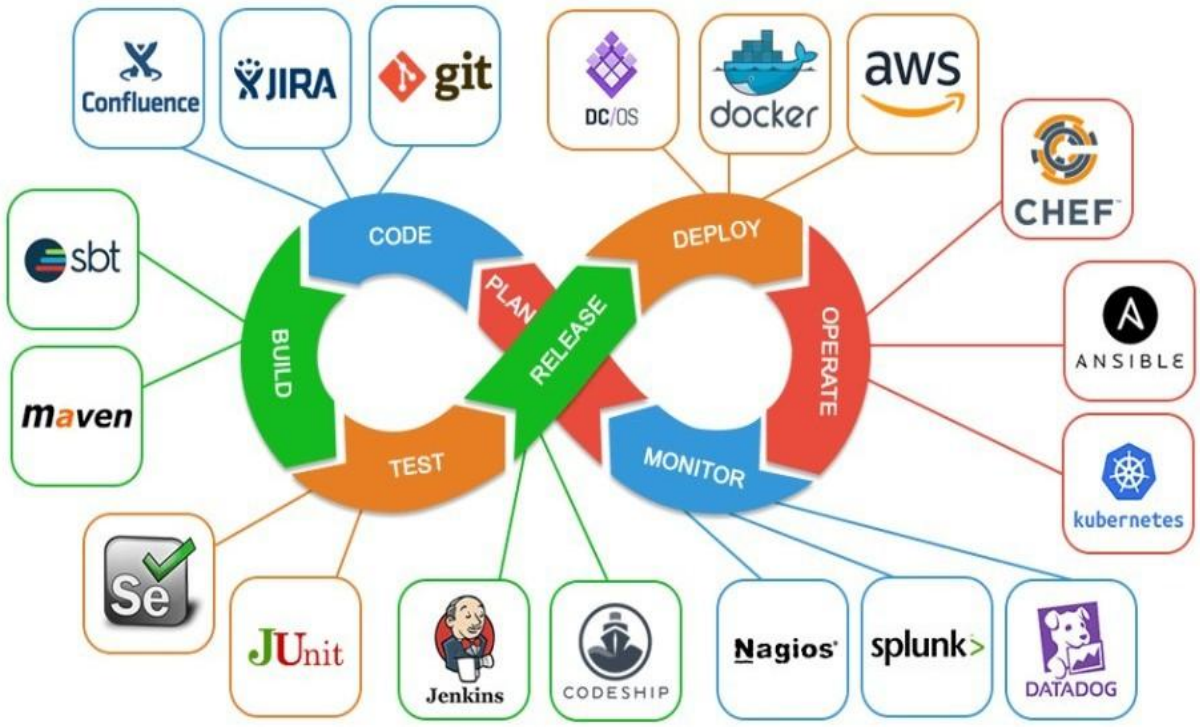
In today's fast-paced digital landscape, DevOps has emerged as a cornerstone of modern software development practices, empowering teams to innovate faster, respond to market changes more effectively, and drive business success.

Devops goes beyond continuous integration and continuous delivery to enable near-instantaneous deployment of products and services in the cloud.

DevOps Technologies

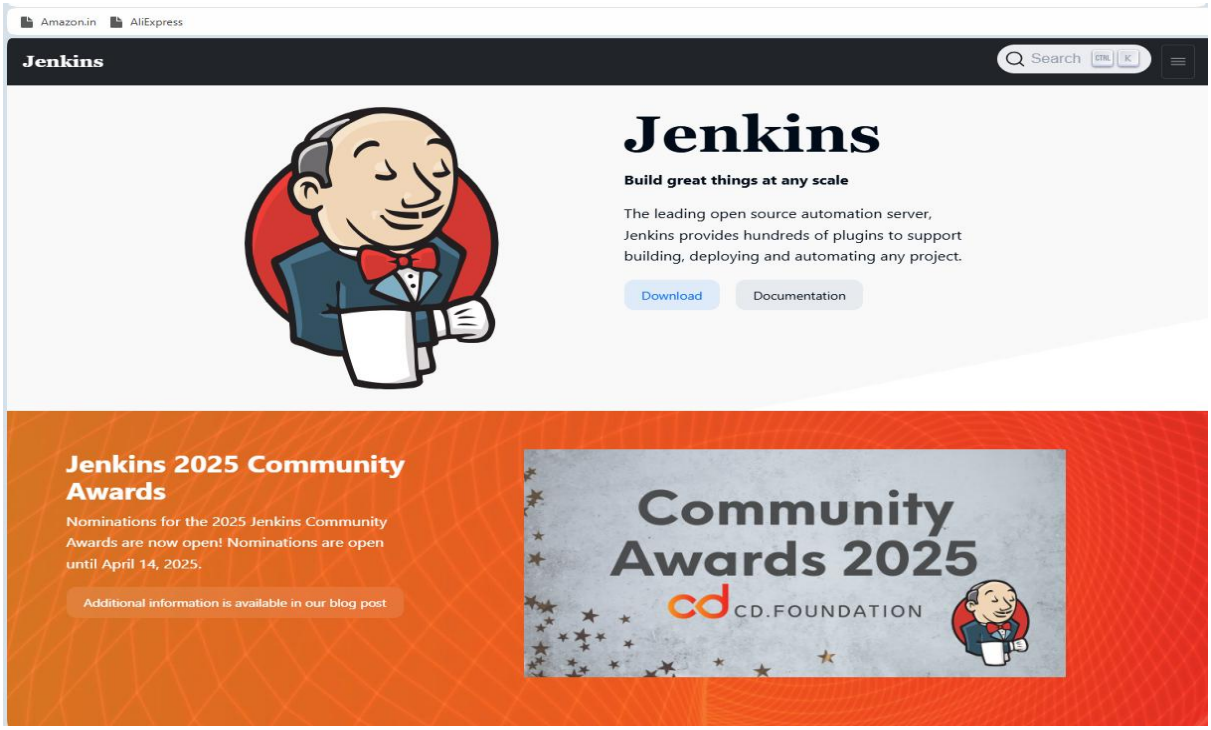
There are many technologies used in DevOps. Each technology is used for specific function.

Some of the technologies with their usage are:



- **GIT** – Used for writing code in DevOps.
- **MAVEN**- Used for build.
- **SELENIUM** – Used for test the project.
- **JENKINS** – Used for release of project.
- **DOCKER** – Used for deploy project.
- **KUBERNETES** – Used to operate project.
- **SPLUNK** – Used for monitoring project

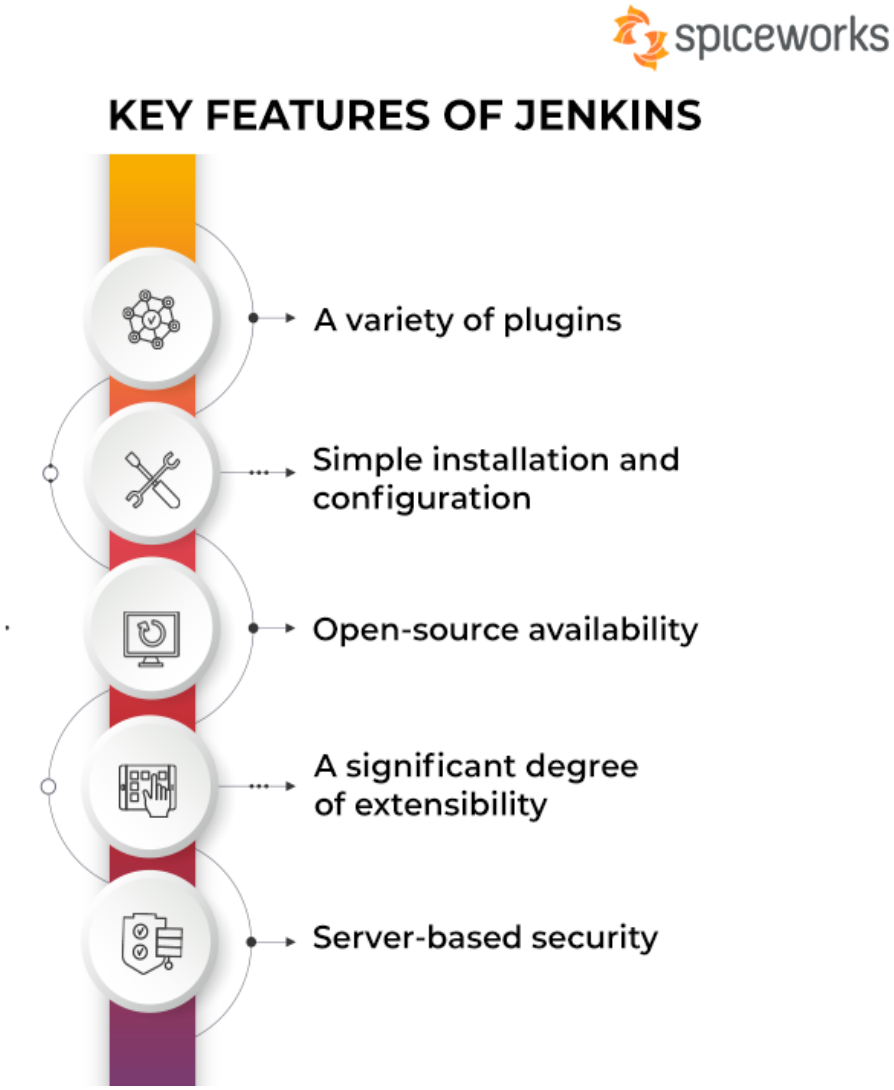
JENKINS

- Jenkins is an opensource **continuous integration /continuous delivery** and deployment (CI/CD) automation software
- **DevOps** tool written in the **Java** programming language. It is used to implement CI/CD workflows called pipelines.
- CI/CD pipelines automate testing and reporting on isolated changes in a larger codebase in real time. They also facilitate the integration of disparate branches of the code into a main branch.
- Pipelines rapidly detect defects in a codebase, build the software, automate testing of builds, prepare the codebase for deployment and delivery, and ultimately deploy code to containers and virtual machines
- 

Key Features of Jenkins

Jenkins is simple to set up and customize. Jenkins has many plugins that give it a lot of versatility. It delivers code instantaneously, generates a report after deployment, highlights errors in code or tests, and detects and resolves various issues in near real-time. It's also ideal for integration because it's all done automatically.

There is also a fantastic support community. These features in more detail:



Jenkins Applications

Jenkins is a popular open-source automation server that provides a robust platform for implementing continuous integration and continuous delivery (CI/CD) pipelines. Its flexibility and extensive plugin ecosystem allow it to fit into nearly any development workflow, supporting a wide range of use cases across software development and deployment. Here are some detailed applications of Jenkins:

1. **Continuous Integration (CI)**

Jenkins is primarily used to automate the process of continuous integration, where it builds and tests code every time a change is committed to a version control system. This helps developers detect issues early in the development cycle, improving code quality and reducing the time needed to validate and release new software updates.

- **Automated Builds:** Jenkins can compile and build code from various environments and languages.
- **Automated Testing:** It can run a suite of tests (unit, integration, system) on new code to ensure it doesn't break anything.

2. Continuous Delivery (CD)

Beyond continuous integration, Jenkins can automate steps in software delivery, making it easier to deploy and release new versions. It allows for the automation of the deployment process, making sure that you can release reliably at any time.

- **Automated Deployment**: Jenkins can automate the deployment of applications to various environments, including testing, staging, and production.
- **Rollbacks**: It can also automate the rollback of a deployment if the deployment fails, ensuring quick recovery from errors.

3. Infrastructure as Code (IAC)

Jenkins is used to implement Infrastructure as Code practices, which involve managing and provisioning infrastructure through code instead of through manual processes.

- Configuration Management: Jenkins can integrate with tools like Ansible, Chef, and Puppet to automate the configuration of servers.

- Server Provisioning: It can also use scripts or templates to create or update servers.

4. Monitoring and Reporting

Jenkins can be configured to monitor its own performance and to generate reports on various aspects of the development process.

- Build Monitoring: Jenkins can keep track of build success rates and notify developers of failures.
- Performance Trends: It can generate reports that track performance metrics over time, helping teams understand trends.

5. DevOps and Multibranch Pipeline

Jenkins supports DevOps practices by enabling teams to implement multibranch pipelines, where each branch of the version control system can have its own tailored CI/CD pipeline.

- Pipeline as Code: Jenkins Pipelines allow defining build, test, and deploy stages that are stored in a Jenkins file and versioned along with the code.
- Parallel Execution: Jenkins can execute jobs in parallel, reducing the time required for builds and tests.

6. Containerization Support

Jenkins has robust support for Docker and Kubernetes, allowing teams to use containers for builds, tests, and deployments.

- **Docker Integration**: Jenkins can build Docker images and push them to Docker registries.
- **Kubernetes Integration**: It can manage Kubernetes pods, enabling dynamic provisioning of agents for builds and tests.

7. Third-Party Integration

Jenkins' extensive plugin ecosystem allows it to integrate with virtually any tool used in software development, from version control systems like Git to issue tracking systems like JIRA, and artifact repositories like Artifactory.

8. Security and Compliance

Jenkins can help enforce security policies and compliance standards by automating security scans and compliance checks.

- **Static Code Analysis**: Plugins can scan source code for security vulnerabilities.

MAVEN

- Maven is a build tool that uses a **POM (project object model)** to help build processes through plugins. It's not unlike how MS Build helps with C#, or MAKE with C/C++, or even npm /Grunt and JavaScript.
- A POM is the core of a project's configuration in Maven. It's an XML file containing info about the project, configuration details, and default values for most projects.
- Maven helps developers maintain Java-based applications through projects that organize code files and build scripts to run compiler tools, version numbers for compiled code, and dependency management that lets one project reference a version of another project.
- Jenkins allows you to run Maven, and decide when to call a POM file, what condition to call, and what to do with the outcome. And since Jenkins can listen to different events, e.g svn commit, the current time is 12:00 AM, etc. Jenkins and Maven can become quite a powerful duo. For example, you can ask Jenkins to trigger a build or run through all JUnit tests whenever a new code is committed and then, if the unit tests are passed, deploy on a target machine.

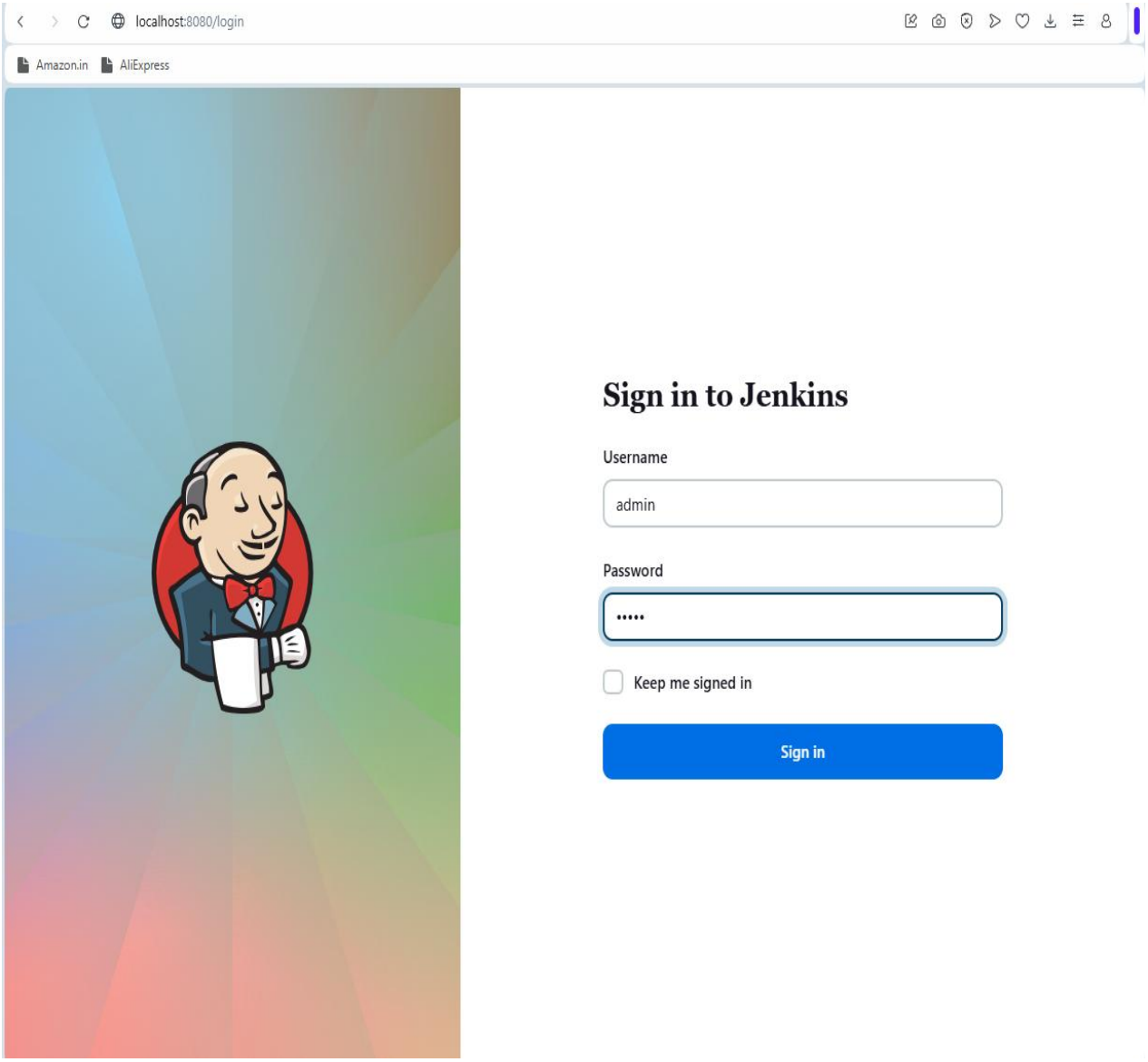
- Maven projects are then published to a **Maven Repository,** which is essentially like a web-based file share. There are index files in Maven repositories that list what projects and versions are stored in the repository, as well as metadata files that describe what each project is.

- Adding Maven to your Jenkins is just as simple as adding any other plugin. Simply install the Maven plugin in Jenkins. This will add a “Build” section to your projects where you can specify exactly what to execute.

Installing Jenkins and Maven

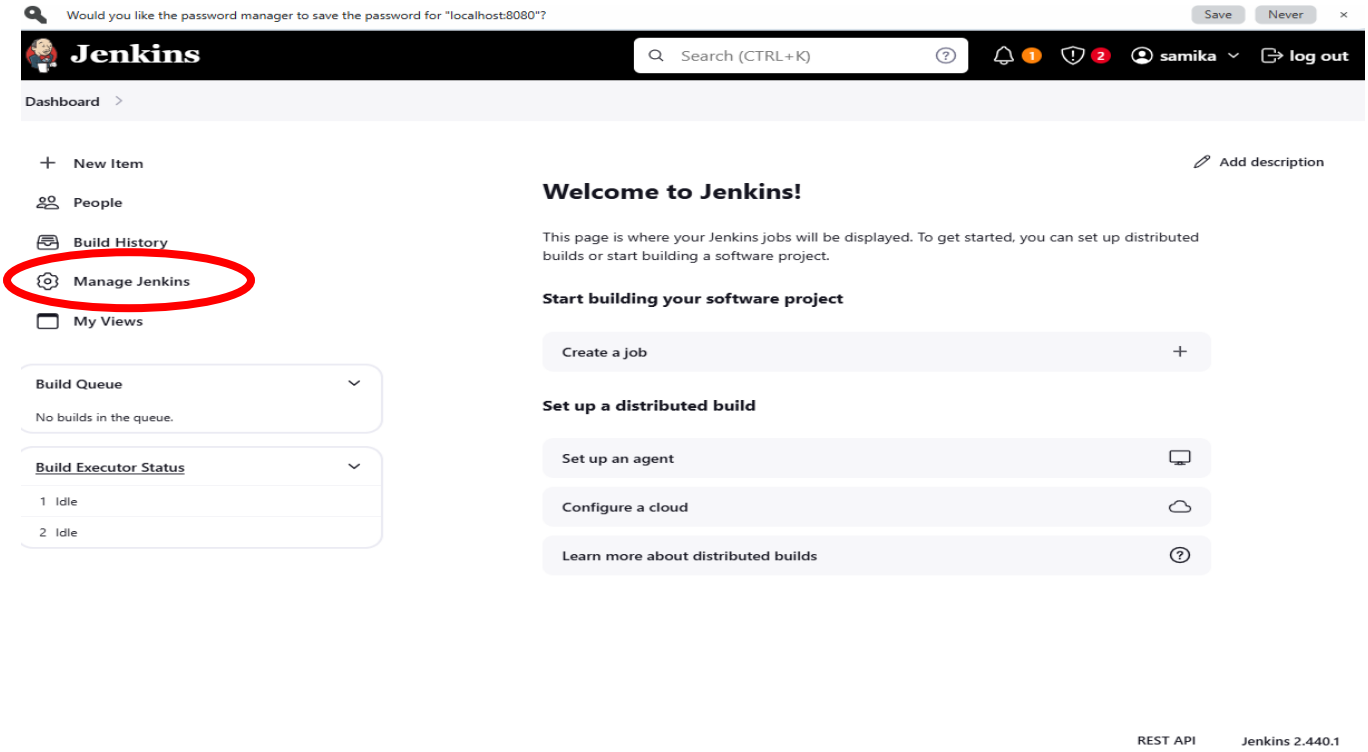
Step 1: Installing Jenkins

- Install Jenkins in our local desktop
- Sign into the Jenkins

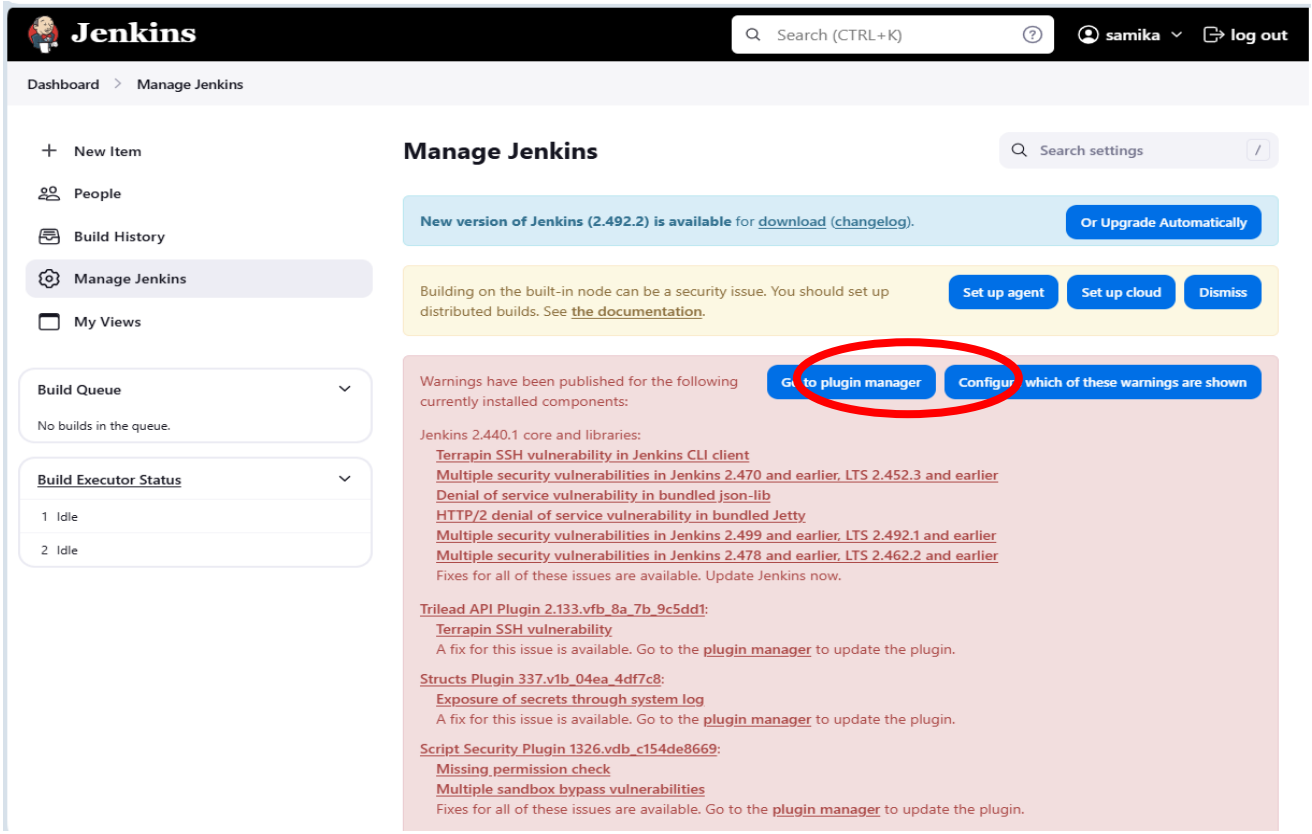


Step 2: Installing Maven

- Go to Manage Jenkins



- Go to the Plugin manager

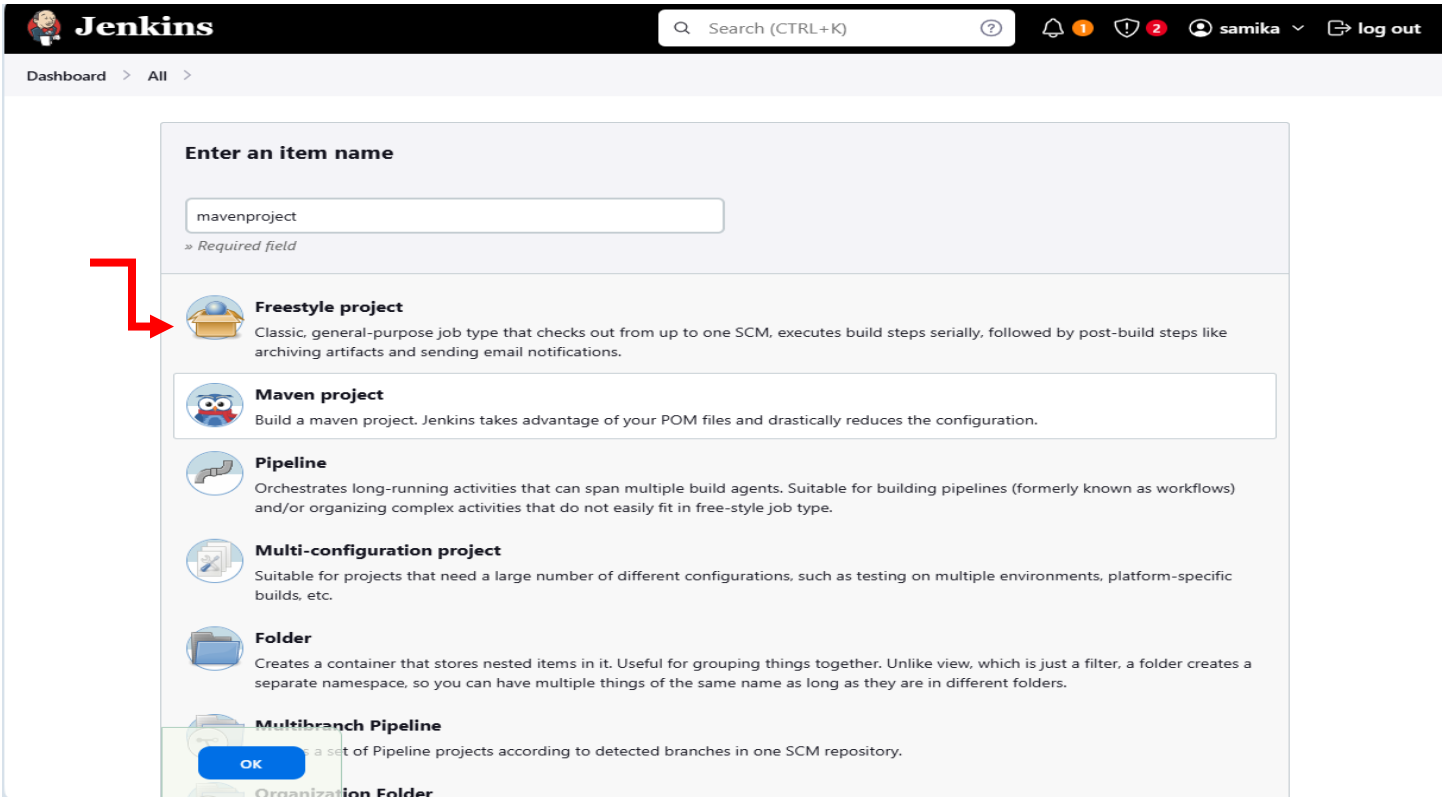


- Go to the Available Plugins
- Install the Maven by searching in search engine

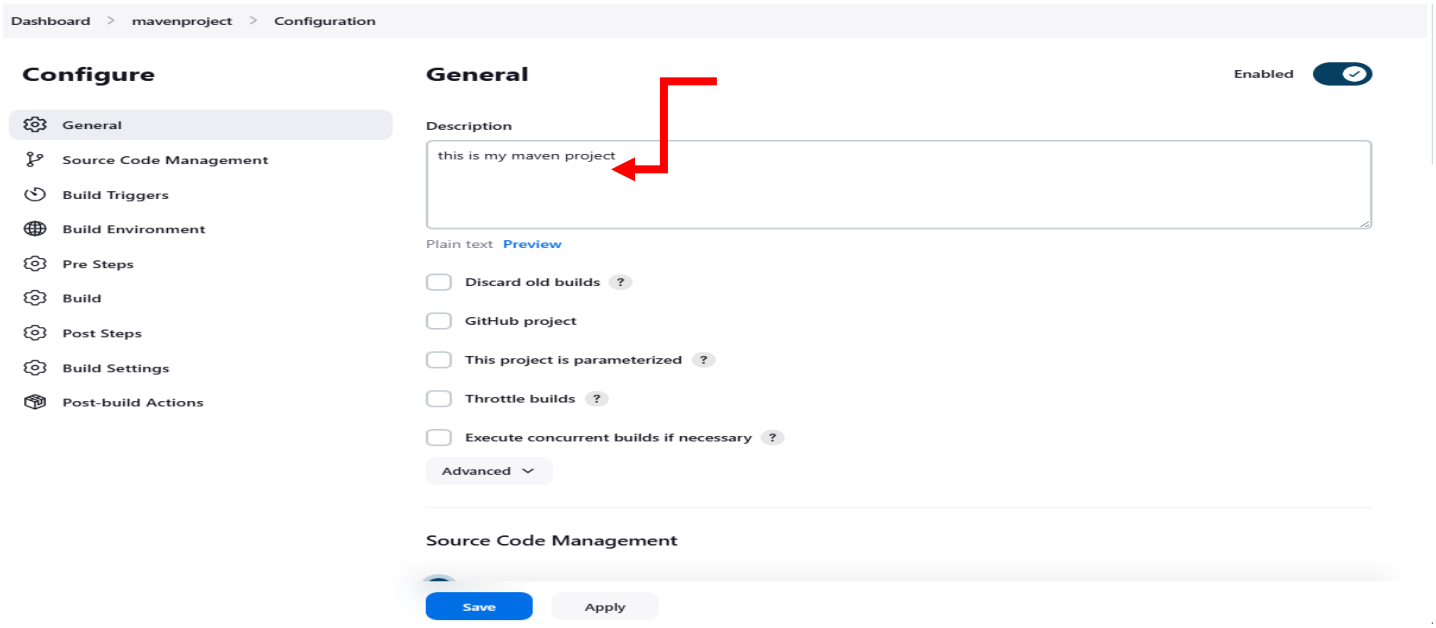
To check whether the Maven is installed, go to installed plugins and search Maven.

Building a Maven Project in Jenkins

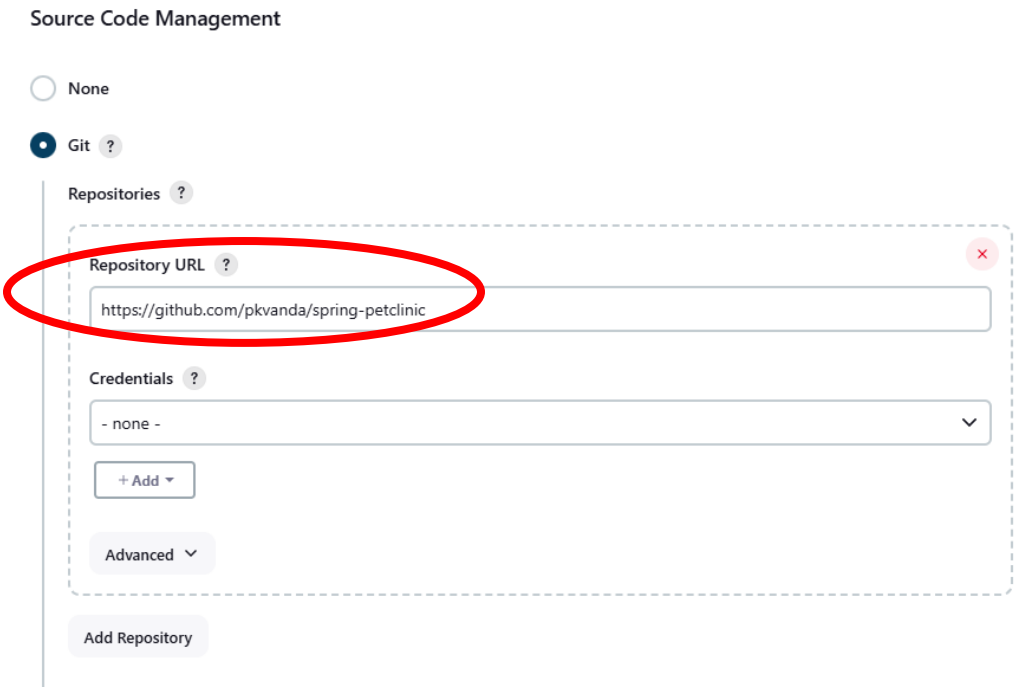
- Go to the Jenkins dashboard and click on new item.
- Now enter the item name and select freestyle project



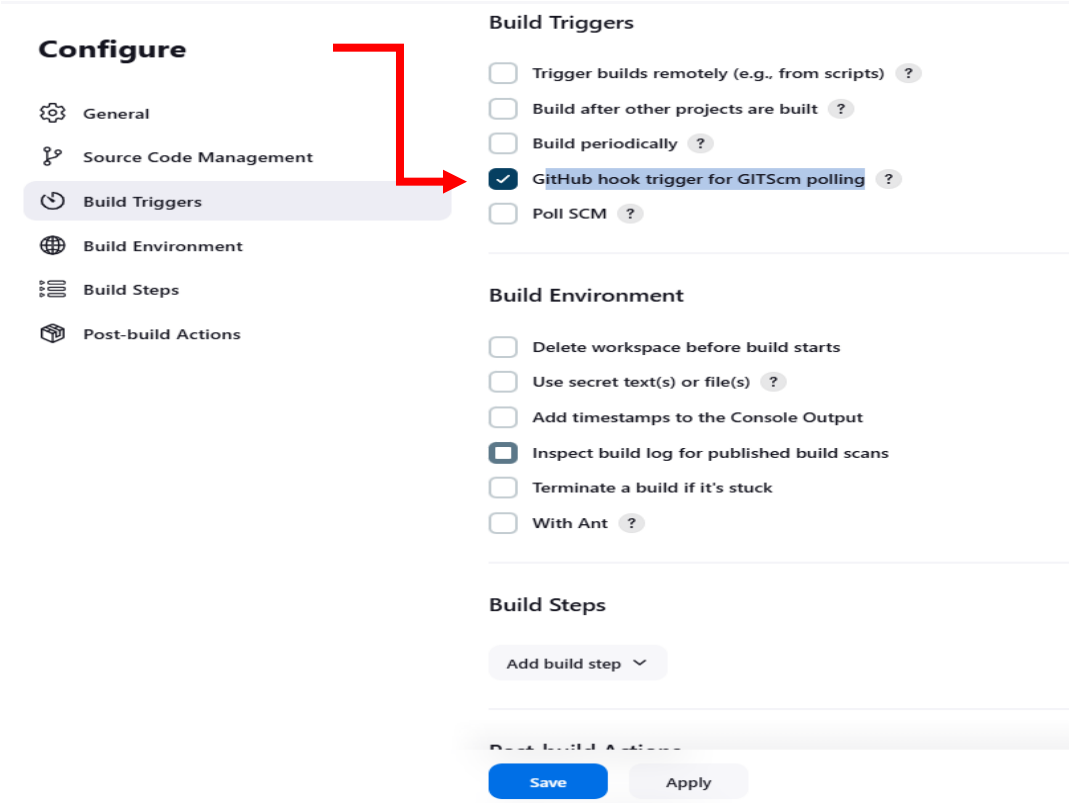
- Now write the description of project (optional)



- Now select the source code management as GIT, now paste the GIT repository URL

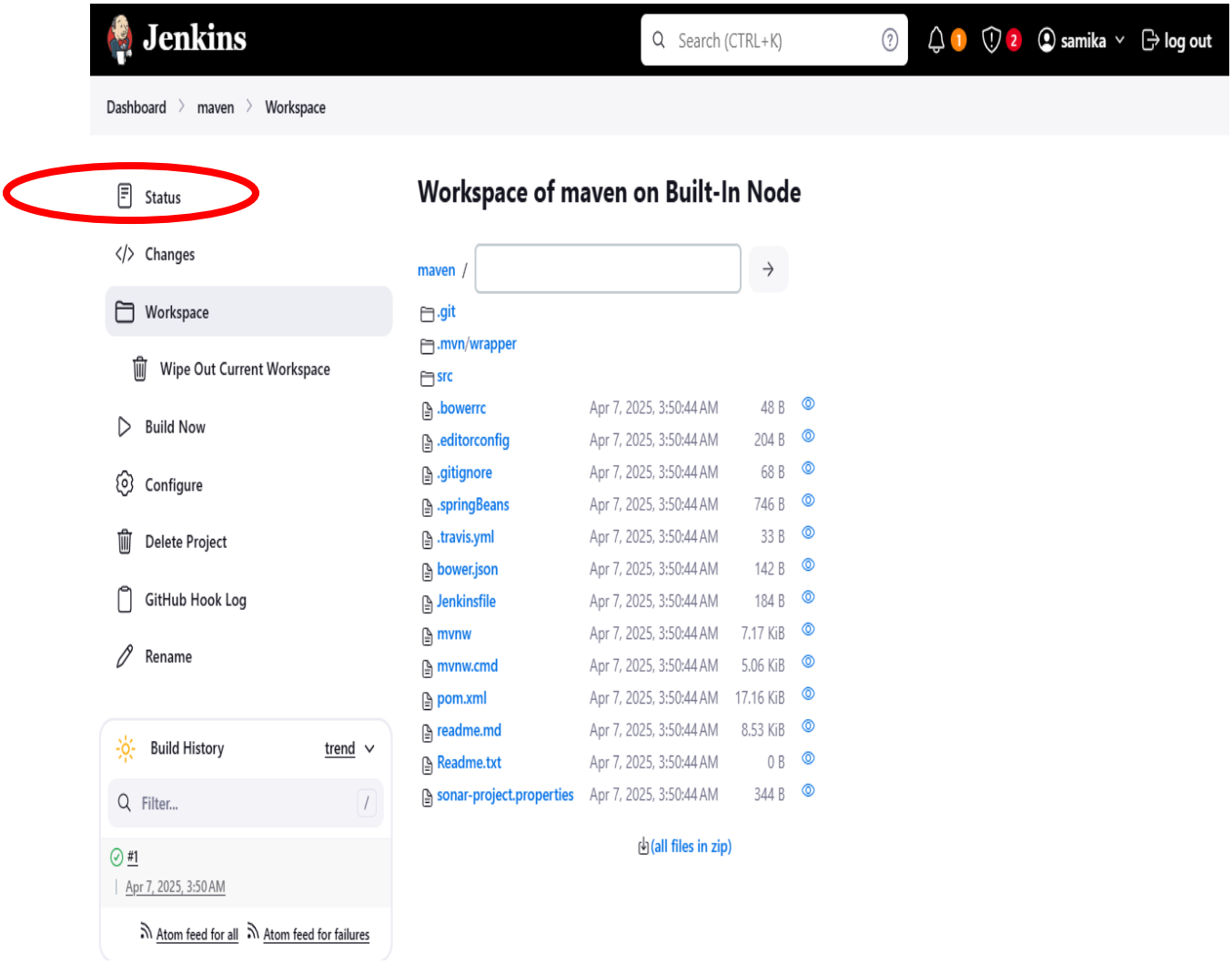


- Now select the build trigger (i.e GitHub hook trigger for GIT Scm polling)



- Now click on the Build now
- To check whether the project is build, go to the workspace then you will find the project output

Output


Jenkins
Search (CTRL+K)
samika
log out
Dashboard
maven
Workspace
Status
Changes
Workspace
Wipe Out Current Workspace
Build Now
Configure
Delete Project
GitHub Hook Log
Rename
Build History
trend
Filter...
#1
Apr 7, 2025, 3:50 AM
Atom feed for all
Atom feed for failures
Workspace of maven on Built-In Node
maven /
.git
.mvn/wrapper
src
.bowerrc Apr 7, 2025, 3:50:44 AM 48 B
.editorconfig Apr 7, 2025, 3:50:44 AM 204 B
.gitignore Apr 7, 2025, 3:50:44 AM 68 B
.springBeans Apr 7, 2025, 3:50:44 AM 746 B
.travis.yml Apr 7, 2025, 3:50:44 AM 33 B
bower.json Apr 7, 2025, 3:50:44 AM 142 B
Jenkinsfile Apr 7, 2025, 3:50:44 AM 184 B
mvnw Apr 7, 2025, 3:50:44 AM 7.17 KiB
mvnw.cmd Apr 7, 2025, 3:50:44 AM 5.06 KiB
pom.xml Apr 7, 2025, 3:50:44 AM 17.16 KiB
readme.md Apr 7, 2025, 3:50:44 AM 8.53 KiB
Readme.txt Apr 7, 2025, 3:50:44 AM 0 B
sonar-project.properties Apr 7, 2025, 3:50:44 AM 344 B
(all files in zip)

CONCLUSION

Hence we build a Maven Project in Jenkins & Using build triggers.

References

- https://www.simplilearn.com/tutorials/jenkins-tutorial/what-is-jenkins
- https://maven.apache.org/what-is-maven.html
- https://www.techtarget.com/searchitoperations/definition/GitHub