

# **Continuous Integration & Continuous Delivery Lab File**

## **Submitted By**

Rohin Mehrotra

SAPID: 500095605

Enrollment No: R2142210656

Batch: 03

Semester V

BTech CSE DevOps



SCHOOL OF COMPUTER SCIENCE  
UNIVERSITY OF PETROLEUM & ENERGY  
STUDIES

Dehradun-248007

2023-24

# INDEX

[illegible]

# Experiment 1

## Introduction and setting up Jenkins

Jenkins is an open-source automation server that facilitates continuous integration (CI) and continuous delivery (CD) of software projects. It helps development teams streamline the building, testing, and deployment processes, leading to faster and more reliable software development. Jenkins is highly extensible and customizable, making it a popular choice for automating various aspects of the software development lifecycle.

Here are its key features in bullet points:

- **Continuous Integration (CI):** Automates the process of integrating code changes from multiple developers into a shared repository.
- **Continuous Delivery/Deployment (CD):** Supports automating the deployment process, enabling teams to deliver software updates to production or staging environments efficiently and reliably.
- **Extensibility:** Offers a vast library of plugins that extend its functionality, allowing integration with various tools, technologies, and services.
- **Customizable Workflows:** Enables the creation of complex, multi-step build and deployment pipelines using a simple and flexible scripting language (usually Groovy-based DSL).
- **Distributed Builds:** Can distribute build and test tasks across multiple machines, improving overall efficiency and reducing build times.
- **Easy Setup and Configuration:** Provides a web-based interface for managing jobs, configurations, and plugins, making it user-friendly for both developers and administrators.
- **Monitoring and Reporting:** Offers detailed build logs, reports, and dashboards, helping to identify issues and monitor the status of builds and deployments.
- **Scalability:** Supports horizontal scaling by distributing build agents across different nodes, allowing for handling larger workloads.

- **Security:** Offers user authentication and authorization controls to ensure that only authorized users have access to specific jobs and resources.
- **Integration:** Integrates with version control systems (e.g., Git, SVN), build tools (e.g., Maven, Gradle), testing frameworks, and deployment platforms.
- **Community and Support:** Being open-source, Jenkins benefits from a large and active community that contributes to its development, provides plugins, and offers support through forums and documentation

## Prerequisites:-

- Any small java project
- Java
- Maven
- Jenkins

## Setting up things

1. `sudo apt install openjdk-17.jdk`

```
+ ▾
rohin@victus:~$ java --version
openjdk 11.0.20 2023-07-18
OpenJDK Runtime Environment (build 11.0.20+8-post-Ubuntu-1ubuntu122.04)
OpenJDK 64-Bit Server VM (build 11.0.20+8-post-Ubuntu-1ubuntu122.04, mixed mode, sharing)
rohin@victus:~$
```

2. `sudo apt install git`

```
+ ▾
rohin@victus:~$ git --version
git version 2.34.1
rohin@victus:~$
```

### 3. sudo apt install maven

```
+ ▾
rohin@victus:~$ mvn --version
Apache Maven 3.6.3
Maven home: /usr/share/maven
Java version: 11.0.20, vendor: Ubuntu, runtime: /usr/lib/jvm/java-11-openjdk-amd64
Default locale: en_IN, platform encoding: UTF-8
OS name: "linux", version: "6.2.0-26-generic", arch: "amd64", family: "unix"
rohin@victus:~$
```

### 4. wget -q -O - <https://pkg.jenkins.io/debian-stable/jenkins.io.key> |sudo gpg --dearmor -o /usr/share/keyrings/jenkins.gpg

### 5. sudo sh -c 'echo deb [signed-by=/usr/share/keyrings/jenkins.gpg] <https://pkg.jenkins.io/debian-stable> binary/ > /etc/apt/sources.list.d/jenkins.list'

### 6. sudo apt update

### 7. sudo apt install jenkins

```
+ ▾
rohin@victus:~$ jenkins --version
2.420
rohin@victus:~$
```

### 8. sudo systemctl status jenkins

```
+ ▾ rohin@victus: ~
rohin@victus:~$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/lib/systemd/system/jenkins.service; enabled; vendor preset: enabled)
   Active: active (running) since Sat 2023-08-26 16:42:05 IST; 13s ago
     Main PID: 2062460 (java)
       Tasks: 63 (limit: 18282)
      Memory: 487.6M
         CPU: 20.250s
    CGroup: /system.slice/jenkins.service
            └─2062460 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=
```