

Roll No: 67

NAME: HERAMB R. PAWAR

SUB: SEPM

CLASS: D11AD

### PRACTICAL 5

**Aim:** To Build the Pipeline of Job using Maven / Gradle / Ant in Jenkins, create a pipeline script to Test and deploy an application over a server

**Problem Statement:** To understand the importance of Jenkins to build and deploy Software Applications on Server Environment

**Theory:**

Jenkins Pipeline is a combination of plugin that supports integration and implementation of continuous delivery pipelines. It has an extensible automation server to create simple and complex delivery pipelines as code via pipeline DSL. A pipeline is a group of events interlinked with each other in sequence.

**Continuous Delivery Pipeline:**

In a Jenkins Pipeline, every job or event has some sort of dependency on at least one or more events. It contains a group of states called build, deploy, test and release. These events are interlinked with each other. Every state has its events, which



work in sequence called a Continuous delivery pipeline.

A Continuous Delivery pipeline is an automated expression to display your process for getting software for version control. Thus every change made in your software goes through a number of complex processes on its way of being released. It also invokes developing the software in a reliable and repeatable manner and progression of the built software through multiple stages of testing and deployment.

#### Purpose of Pipelining:

Jenkins is fundamentally an automation engine which supports a number of automation patterns. Pipeline adds a powerful set of automation ~~tools~~ tool onto Jenkins, supporting use cases that span from simple continuous integration to comprehensive CD pipelines. By modelling a series of related tasks, users can take many advantage of the many features of Pipeline.

- a) Durable
- b) ~~Pause~~ Pausable
- c) Versatile
- d) Extensible

NAME: \_\_\_\_\_ STD.: \_\_\_\_\_ DIV.: \_\_\_\_\_

DATE :

PAGE :

Conclusion:

In this experiment, we installed Jenkins along with its prerequisites on our Ubuntu machine and created our first Freestyle build job on Jenkins.

Ea

**Name:** Heramb Pawar


**Subject:**SEPM

**Roll:** 67

## Practical Output:

**Pre-requisites:** Tomcat Server


## Building First Pipeline



### Welcome to Jenkins!


Please sign in below or [create an account](#).


Sign in


 **Jenkins**


Dashboard >


+ New Item


 People


 Build History


 Manage Jenkins

 My Views

 **Jenkins**

 1


 2

 Heramb Ramakant Pawar


Dashboard > All >

Enter an item name


» 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.

 **Pipeline**

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

 **Multi-configuration project**

Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

 **Folder**

Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a new namespace, so you can have multiple things of the same name as long as they are in different folders.

[View Multibranch Pipeline](#)

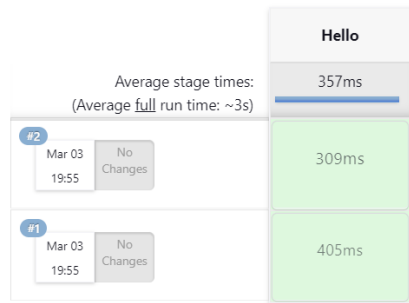
## Pipeline Heramb's First Pipeline

Hi, this is My First Pipeline !!!

 Edit description

**Disable Project**


### Stage View



### ✓ Console Output


```
Started by user Heramb Ramakant Pawar
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/Heramb's First Pipeline@2
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Hello)
[Pipeline] echo
Hello Heramb !!
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```


## Running Application on Tomcat:


**Jenkins**


Search (CTRL+K)


Dashboard > Manage Jenkins > Plugin Manager

 Updates

 Available plugins

 Installed plugins


 Advanced settings


 Download progress

### Download progress

Preparation

- Checking internet connectivity
- Checking update center connectivity
- Success

Deploy to container  Installing

Loading plugin extensions  Pending

→ [Go back to the top page](#)  
(you can start using the installed plugins right away)

→ ☐ Restart Jenkins when installation is complete and no jobs are running





Dashboard > All >


Enter an item name

SEPM\_EXP\_5

» 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**  
Organizes long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) organizing complex activities that do not easily fit in free-style job type.

OK

Source Code Management

☐ None

☒ Git ?

Repositories ?

Repository URL ?

https://github.com/Heramb123/webapp.git

X

Credentials ?

/\*\*\*\*\*


▼


Add ▼


Advanced ▼


Dashboard > SEPM\_EXP\_5 > Configuration


Configure


 General

 Source Code Management

 Build Triggers

 Build Environment

 Build Steps

 Post-build Actions

Build Steps

Execute Windows batch command ?

X

Command

See [the list of available environment variables](#)

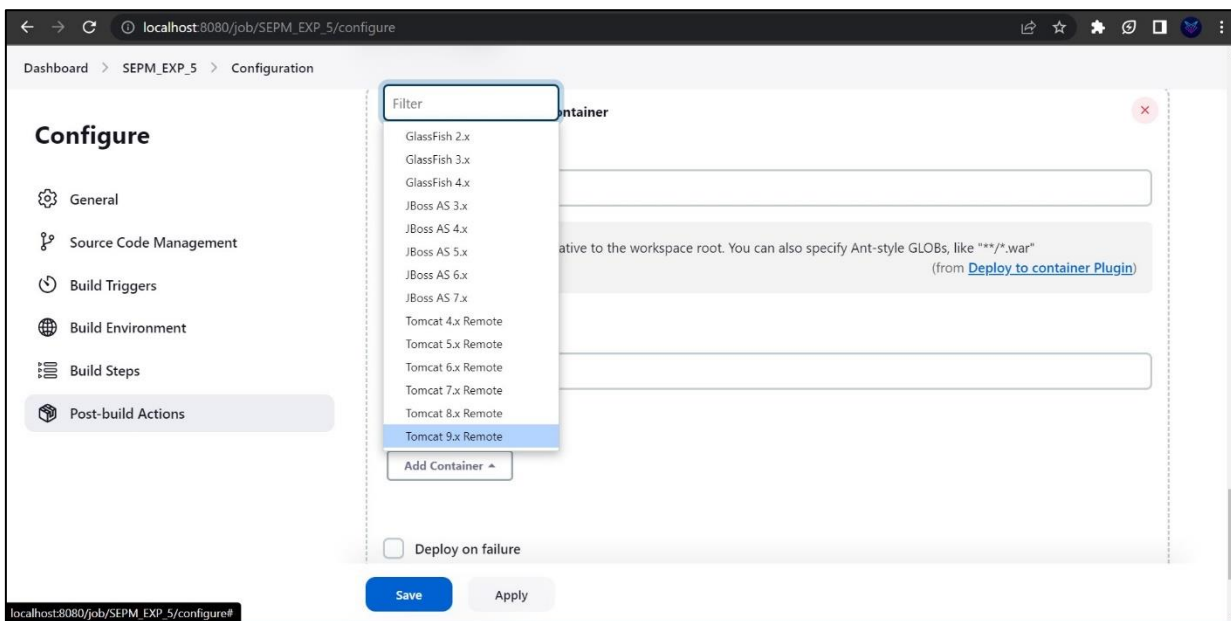
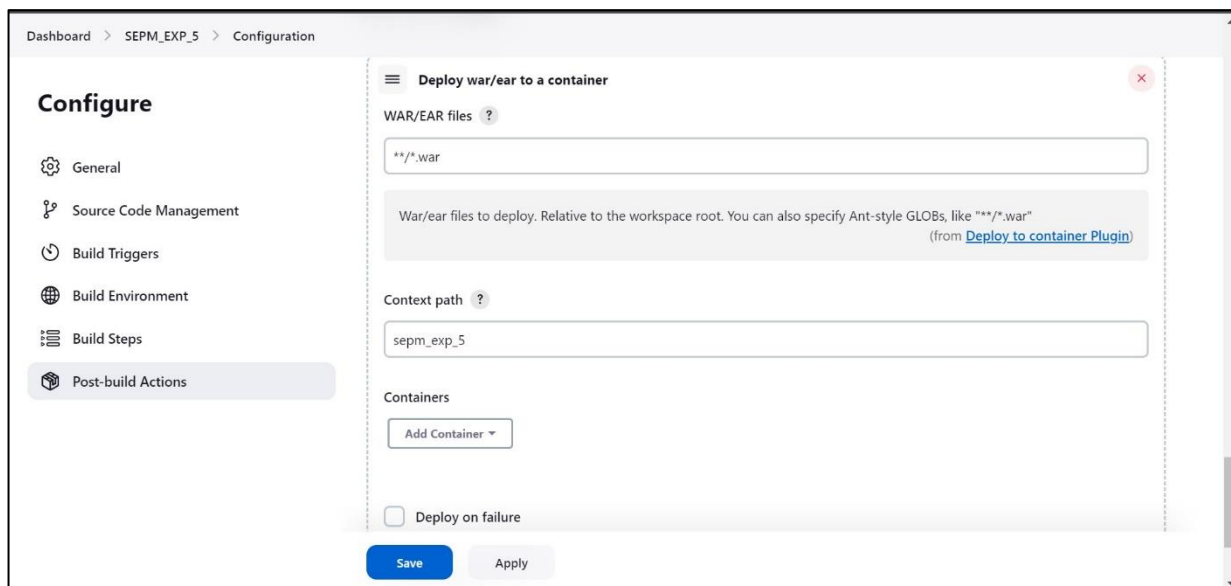
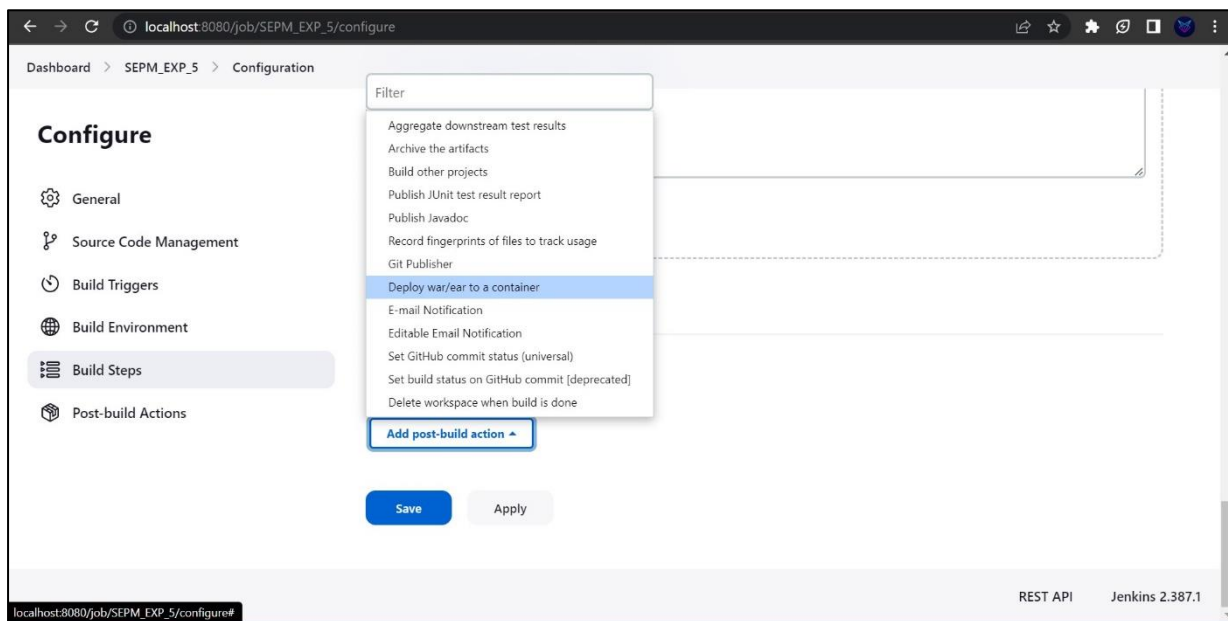
```
mvn package
mvn test
```

Advanced ▼

Add build step ▼

Save

Apply



Dashboard > SEPM\_EXP\_5 >

Status

Changes

Workspace

Build Now

Configure

Delete Project

Rename

## Project SEPM\_EXP\_5

This is experiment 5 of sepm amd aim is to build the pipeline of jobs using Maven in Jenkins, create a pipeline script to Test and deploy an application over a server.

Edit description

Disable Project

### Permalinks

Build History

trend

Filter builds...

#1

Mar 17, 2023, 1:59 PM

Atom feed for all

Atom feed for failures

localhost:8080/job/SEPM\_EXP\_5/build?delay=0sec

REST API Jenkins 2.387.1

Dashboard > SEPM\_EXP\_5 > #3 > Console Output

Status

Changes

Console Output

View as plain text

Edit Build Information

Git Build Data

Previous Build

✓ Console Output

Started by user  
Running as SYSTEM [Heramb Ramakant Pawar](#)  
Building in workspace C:\ProgramData\Jenkins\jenkins\workspace\SEPM\_EXP\_5  
The recommended git tool is: NONE  
using credential 67068613-6849-41b0-9f4e-5da80b0d61df  
> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\jenkins\workspace\SEPM\_EXP\_5\.git # timeout=10  
Fetching changes from the remote Git repository  
> git.exe config remote.origin.url <https://github.com/Heramb123/webapp.git> # timeout=10  
Fetching upstream changes from  
> git.exe --version # timeout=10  
> git --version # 'git version 2.37.0.windows.1'  
using GIT\_ASKPASS to set credentials  
> git.exe fetch --tags --force --progress -- <https://github.com/Heramb123/webapp.git>  
+refs/heads/\*:refs/remotes/origin/\* # timeout=10  
> git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10  
Checking out Revision c0296cd1dd2a4296f81d7fe0ca368563f84522bb (refs/remotes/origin/master)  
> git.exe config core.sparsecheckout # timeout=10

← → ↺ ⓘ localhost:8081/sepm\_exp\_5/

# Hello World!