

# Jenkins pipeline with roll-back functionality using nexus and Tomcat

## Pre-requisites

1. Jenkins server
2. Nexus repository (hosted in ec2 instance )
3. Tomcat server

## Plugins need to install in jenkins

Go to manage jenkins -> available-> search for publish over ssh plugin and install without restart.

Go to manage jenkins -> available-> search for nexus platform and install without restart.

Go to manage jenkins -> available-> search for parameterized trigger plugin and install without restart.

Now configure nexus and tomcat server from system configure in jenkins dashboard.

 SSH Server

Name?

tomcat-8

Hostname?

52.205.71.129

Username?

ec2-user

Remote Directory?

/opt/apache-tomcat-8.5.66/webapps

Success

 Advanced...

Test Configuration

## Nexus Repository Manager 3.x Server

Display Name

artifact-nexus

Server ID 

nexus-3

Server URL

http://3.85.211.94:8081

Credentials

admin/\*\*\*\*\*

 Add

Nexus Repository Manager 3.x connection succeeded

Test connection




NXRM OSS 3.29.2-02 found. Some operations require Nexus Repository Manager Professional server version 3.13.0 or newer; use of an incompatible server could result in failed builds.


Now go to jenkins dashboard -> new item -> name of CI-pipeline -> select freestyle project -> go.


Select git as a source code and provide git url and branch name with git credentials  
In my case i chose public git repo.

### Source Code Management


☐ None


☒ Git 

Repositories 

Repository URL 

https://github.com/kishancs2020/webAppExample

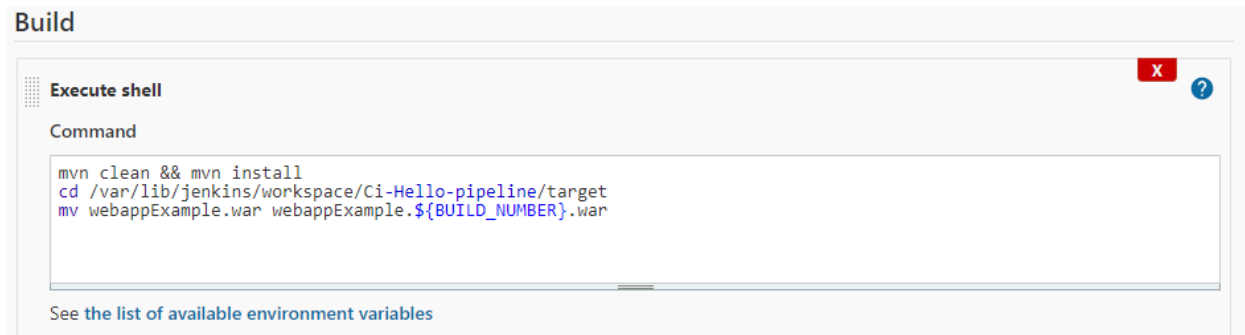
Credentials 

- none -  Add

Advanced...

Add Repository

Go to Build -> select execute shell open from the available plugin and execute maven commands to build package and rename the war name as per the server configuration. In my case, I have given build number as a version tag for war.



Once we are good with packaging source code to war file, now we need to push war file to artifact repository. In my case, I am using Nexus as my artifacts; I already have repository, so I will be pushing war file to Nexus repo as shown below.

Select Nexus artifacts publish manager plugin from build option and provide required parameters to push war to artifacts. As shown below

Nexus Repository Manager Publisher

X

### Nexus Repository Manager Publisher

Nexus Instance

artifact-nexus

▼

Nexus Repository ?

jenkins-hello-app

▼

Tag ?

Packages

Group

X

Group

X

Group

webapp-group

Artifact

webappExample

Version

\${BUILD\_NUMBER}

Packaging

war

Artifacts

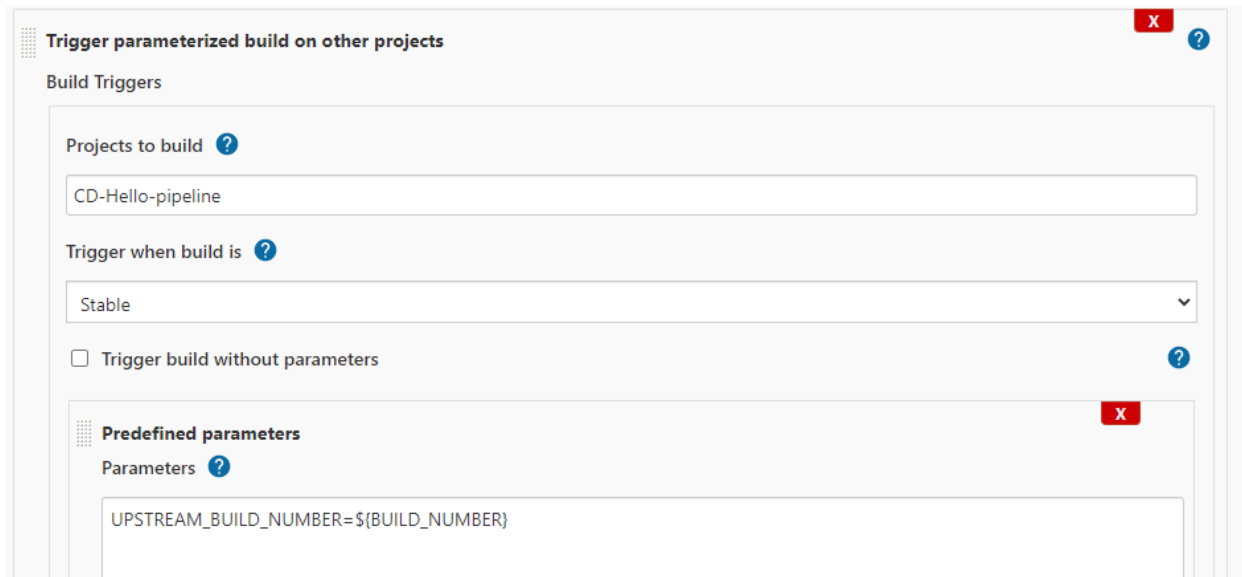
Maven Artifact

X

File Path

/var/lib/jenkins/workspace/Ci-Hello-pipeline/target/webappExample.\${BUILD\_NUMBER}.war

Now choose trigger parameterized build on other projects plugin to pass BUILD\_NUMBER of CI pipeline to pass cd pipeline so CD pipeline with trigger only if CI pipeline is succeeded or stable that is called as a downstream job.



The screenshot shows the configuration window for the 'Trigger parameterized build on other projects' plugin. The window has a title bar with a red 'X' and a question mark icon. The main content area is titled 'Build Triggers' and contains the following fields:

- Projects to build**: A text input field containing 'CD-Hello-pipeline'.
- Trigger when build is**: A dropdown menu with 'Stable' selected.
- Trigger build without parameters**: An unchecked checkbox.
- Predefined parameters**: A section with a red 'X' icon and a question mark icon, containing a text input field with the value 'UPSTREAM\_BUILD\_NUMBER=\${BUILD\_NUMBER}'.

Under general features select this build as a parameterized and select which parameter to execute. In our case we are passing upstream job build number as a parameter.



The screenshot shows the configuration window for the 'String Parameter' plugin. The window has a title bar with a red 'X' and a question mark icon. The main content area is titled 'String Parameter' and contains the following fields:

- Name**: A text input field containing 'UPSTREAM\_BUILD\_NUMBER'.
- Default Value**: An empty text input field.
- Description**: A large text area for describing the parameter.

At the bottom of the window, there is a checkbox labeled 'This project is parameterized' which is checked, and a 'Preview' button.

Download war file from nexus repo using wget command.

**Build**

**Execute shell** X ?

Command

```
set VNAME=$UPSTREAM_BUILD_NUMBER
wget --user=admin --password=| "http://3.85.211.94:8081/repository/jenkins-hello-app/webapp-group/webappExample/${U
```

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

Deploy war file to webapps folder in tomcat server and execute required commands

**Transfers**

**Transfer Set**

**Source files** ?

`**/webappExample-${UPSTREAM_BUILD_NUMBER}.war`

**Remove prefix** ?

**Remote directory** ?

`/`

**Exec command** ?

`sudo systemctl restart tomcat`

All of the transfer fields (except for Exec timeout) support substitution of [Jenkins environment variables](#)

[Advanced...](#)

Console output:

2021-06-05 17:02:51 (330 MB/s) - 'webappExample-11.war' saved [2517/2517]

SSH: Connecting from host [ip-172-31-86-58]  
SSH: Connecting with configuration [tomcat-8] ...  
SSH: EXEC: completed after 201 ms  
SSH: Disconnecting configuration [tomcat-8] ...  
SSH: Transferred 1 file(s)  
Finished: SUCCESS

← → ↻ ⚠ Not secure | 52.205.71.129:8080/webappExample-11

**Welcome to KK JavaTutorials..**