

# Jenkins Mini-project

## Description

### Jenkins Mini-project1:

Deploy Jenkins server on AWS EC2.

<https://www.jenkins.io/doc/tutorials/tutorial-for-installing-jenkins-on-AWS/>

### Jenkins Mini-project 2:

Build a Jenkins pipeline using a plugin to automatically build and publish our apache image to **Dockerhub**. The **Dockerfile** would be sourced from GITHUB.

### Submission:

Screenshot of your pipeline job console result.

Screenshot of your Docker image published in your Dockerhub.

## I- Jenkins server deployed on AWS EC2

The screenshot displays the AWS Management Console interface for an EC2 instance. At the top, the 'Instances (1/1)' section shows a table with one instance: 'jenkins-ubuntu-server' (ID: i-0d1d05a9b12ab28cd), which is in a 'Running' state. Below this, the 'Details' tab for the instance is open, showing a summary of its configuration. The instance is running on an Ubuntu platform (AMI ID: ami-04a81a99f5ec58529) with a t2.micro instance type. Monitoring is disabled, and termination protection is also disabled.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
jenkins-ubuntu-server	i-0d1d05a9b12ab28cd	Running	t2.micro	2/2 checks passed	View alarms	us-east-1a

  

i-0d1d05a9b12ab28cd (jenkins-ubuntu-server)		
Platform	AMI ID	Monitoring
Ubuntu (Inferred)	ami-04a81a99f5ec58529	disabled
Platform details	AMI name	Termination protection
Linux/UNIX		Disabled

## II- A Jenkins pipeline using a plugin automatically built and published the image to Dockerhub. The Dockerfile is sourced from GITHUB

### 1- Screenshot of the terminal successfully built

```
Operating System: Ubuntu 24.04 LTS
OSType: linux
Architecture: x86_64
CPUs: 1
Total Memory: 957.4MiB
Name: ip-10-0-12-136
ID: 080d9dc9-a609-41ce-89b7-747c1ef74518
Docker Root Dir: /var/lib/docker
Debug Mode: false
Experimental: false
Insecure Registries:
 127.0.0.0/8
Live Restore Enabled: false

jenkins@ip-10-0-12-136:~$ exit
logout
ubuntu@ip-10-0-12-136:~$ sudo su - jenkins
jenkins@ip-10-0-12-136:~$ docker login
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com/ to create one.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organizations using SSO. Learn more at https://docs.docker.com/go/access-tokens/

Username: etachenko.
Password:
Error response from daemon: Get "https://registry-1.docker.io/v2/": unauthorized: incorrect username or password
jenkins@ip-10-0-12-136:~$ login
login: Cannot possibly work without effective root
jenkins@ip-10-0-12-136:~$ docker login
Log in with your Docker ID or email address to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com/ to create one.
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope PAT grants better security and is required for organizations using SSO. Learn more at https://docs.docker.com/go/access-tokens/

Username: etachenko
Password:
WARNING! Your password will be stored unencrypted in /var/lib/jenkins/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credential-stores

Login Succeeded
jenkins@ip-10-0-12-136:~$
```

### 2- Screenshot of the pipeline console output

The screenshot shows the Jenkins Dashboard with a search bar and user controls at the top. The left sidebar contains navigation links: New Item, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins, and My Views. The main area displays a table of build history for two pipelines: 'docker-jenkins-ubuntu-pipeline' and 'elie-jenkins-pipeline'. The table includes columns for status (S), warnings (W), name, last success, last failure, and last duration. The 'docker-jenkins-ubuntu-pipeline' shows a successful build #9 and a failed build #6. The 'elie-jenkins-pipeline' shows a successful build #15 and a failed build #8. Below the table, there are filters for icons (S, M, L) and a '000' indicator.

S	W	Name	Last Success	Last Failure	Last Duration
✓	☁	docker-jenkins-ubuntu-pipeline	1 min 0 sec #9 etachenko/docker-jenkins-pipeline:v1.2 etachenko/docker-jenkins-pipeline:latest	12 min #6 etachenko/docker-jenkins-pipeline:v1.2 etachenko/docker-jenkins-pipeline:latest	1.5 sec
✓	☀	elie-jenkins-pipeline	1 day 18 hr #15	1 day 18 hr #8	17 ms

## ✓ Console Output

[Download](#)[Copy](#)[View as plain text](#)

```
Started by user admin
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/docker-jenkins-ubuntu-pipeline
The recommended git tool is: NONE
using credential Githubcreds
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/docker-jenkins-ubuntu-pipeline/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/etachenko/docker-jenkins-pipeline.git # timeout=10
Fetching upstream changes from https://github.com/etachenko/docker-jenkins-pipeline.git
> git --version # timeout=10
> git --version # 'git version 2.43.0'
using GIT_ASKPASS to set credentials my gihub credentials
> git fetch --tags --force --progress -- https://github.com/etachenko/docker-jenkins-pipeline.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 6f6b375dfb9957e785c1fa07c28922ebe5f3ab42 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 6f6b375dfb9957e785c1fa07c28922ebe5f3ab42 # timeout=10
Commit message: "docker-jenkins-pipeline"
> git rev-list --no-walk 6f6b375dfb9957e785c1fa07c28922ebe5f3ab42 # timeout=10
[docker-jenkins-ubuntu-pipeline] $ docker build -t etachenko/docker-jenkins-pipeline:v1.2 --pull=true /var/lib/jenkins/workspace/docker-jenkins-ubuntu-pipeline
#0 building with "default" instance using docker driver

#1 [internal] load build definition from Dockerfile
```

```
#1 transferring dockerfile: 587B done
#1 WARN: ConsistentInstructionCasing: Command 'From' should match the case of the command majority (uppercase) (line 2)
#1 WARN: ConsistentInstructionCasing: Command 'Label' should match the case of the command majority (uppercase) (line 5)
#1 WARN: ConsistentInstructionCasing: Command 'Label' should match the case of the command majority (uppercase) (line 6)
#1 WARN: ConsistentInstructionCasing: Command 'Run' should match the case of the command majority (uppercase) (line 9)
#1 WARN: ConsistentInstructionCasing: Command 'Run' should match the case of the command majority (uppercase) (line 11)
#1 WARN: ConsistentInstructionCasing: Command 'Copy' should match the case of the command majority (uppercase) (line 14)
#1 WARN: LegacyKeyValueFormat: "Label key=value" should be used instead of legacy "Label key value" format (line 5)
#1 WARN: LegacyKeyValueFormat: "Label key=value" should be used instead of legacy "Label key value" format (line 6)
```

### 3- Screenshot of docker hub

The screenshot shows the Docker Hub interface for the repository `etachenko/docker-jenkins-pipeline`. The 'Tags' tab is selected, displaying a list of tags. Two tags are visible: `latest` and `v1.2`. Both tags have the same digest `d626ea4b46ba` and are for the `linux/amd64` architecture. The 'Last pushed' time for both is '20 hours ago' by user `etachenko`. The 'Compressed Size' for both is '194.53 MB'. The 'Last pull' time for both is '19 hours ago'. The page also shows the repository name, search bar, and navigation tabs.

Tag	Digest	OS/ARCH	Last pull	Compressed Size
<code>latest</code>	<code>d626ea4b46ba</code>	<code>linux/amd64</code>	19 hours ago	194.53 MB
<code>v1.2</code>	<code>d626ea4b46ba</code>	<code>linux/amd64</code>	19 hours ago	194.53 MB

