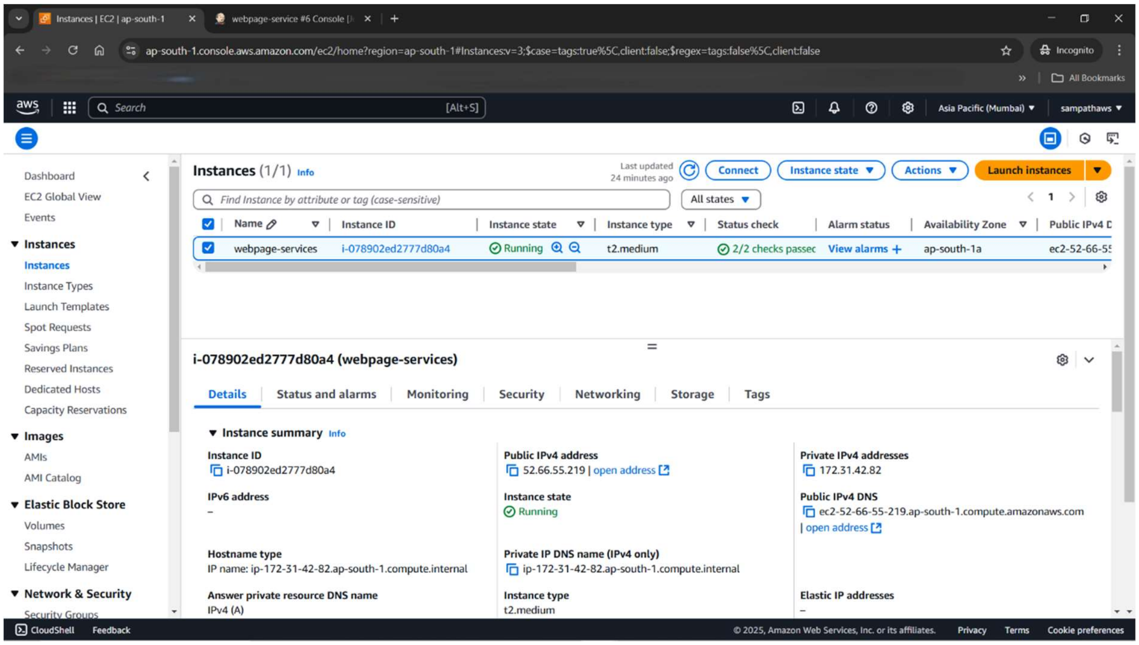
Hosting the Same Webpage in EC2 using Jenkins CI/CD and Docker

Hosting a webpage on AWS EC2 using **Jenkins for CI/CD and Docker**. To automate the process, ensuring that every code update triggers a build and deployment to your EC2 instance.

Requirements:-

1. **AWS Account** – Ensure you have an **EC2 instance** running **Ubuntu 22.04**.
2. **Jenkins Installed** on EC2.
3. **Docker Installed** on EC2.
4. **GitHub Repository** containing your webpage code.

Step 1: Configure Your EC2 Instance



ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Instances:v=3;$case=tags:true%5C,client:false;$regex=tags:false%5C,client:false

Instances (1/1)

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 D
☑	webpage-services	i-078902ed2777d80a4	Running	t2.medium	2/2 checks passec	View alarms +	ap-south-1a	ec2-52-66-5!

i-078902ed2777d80a4 (webpage-services)

Instance summary

Instance ID: i-078902ed2777d80a4

Public IPv4 address: 52.66.55.219 | open address

Private IPv4 addresses: 172.31.42.82

IPv6 address: –

Instance state: Running

Public IPv4 DNS: ec2-52-66-55-219.ap-south-1.compute.amazonaws.com | open address

Hostname type: IP name: ip-172-31-42-82.ap-south-1.compute.internal

Private IP DNS name (IPv4 only): ip-172-31-42-82.ap-south-1.compute.internal

Answer private resource DNS name: IPv4 (A)

Instance type: t2.medium

Elastic IP addresses: –

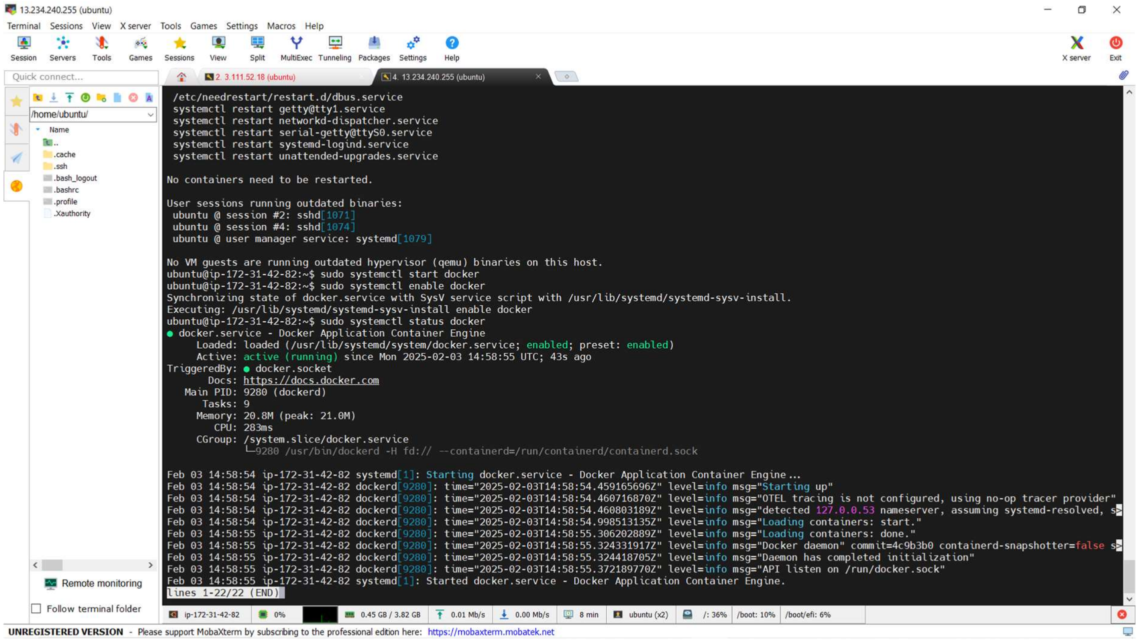
Install Necessary Packages

- sudo apt update
- sudo apt upgrade -y
- curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg

- echo "deb [arch=amd64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
- sudo apt update
- sudo apt install -y docker-ce docker-ce-cli containerd.io

Start and Enable Docker

- sudo systemctl start docker
- sudo systemctl enable docker
- sudo systemctl status docker
- sudo systemctl enable docker

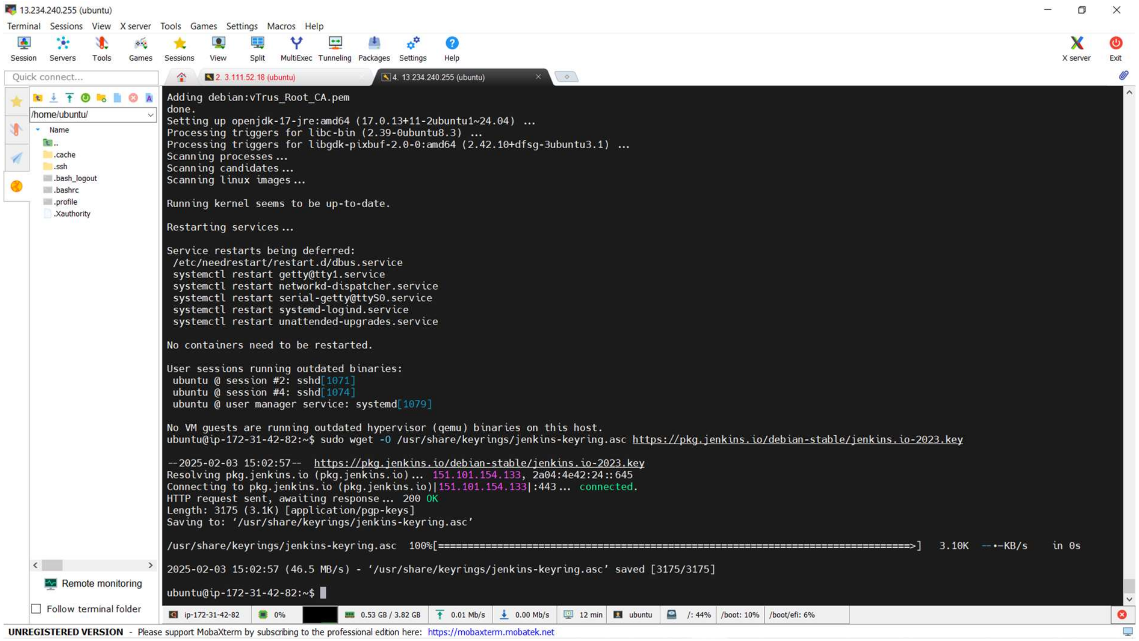

13.234.240.255 (ubuntu)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
2. 3.111.52.18 (ubuntu)
4. 13.234.240.255 (ubuntu)
/home/ubuntu/
Name
..
.cache
.ssh
.bash_logout
.bashrc
.profile
.Xauthority
Remote monitoring
Follow terminal folder
ip-172-31-42-82
0%
0.45 GB / 3.82 GB
0.01 Mb/s
0.00 Mb/s
8 min
ubuntu (x2)
/: 36%
/boot: 10%
/boot/efi: 6%
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

```
 /etc/needrestart/restart.d/dbus.service
 systemctl restart getty@tty1.service
 systemctl restart networkd-dispatcher.service
 systemctl restart serial-getty@ttyS0.service
 systemctl restart systemd-logind.service
 systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
 ubuntu @ session #2: sshd[1071]
 ubuntu @ session #4: sshd[1074]
 ubuntu @ user manager service: systemd[1079]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-42-82:~$ sudo systemctl start docker
ubuntu@ip-172-31-42-82:~$ sudo systemctl enable docker
Synchronizing state of docker.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable docker
ubuntu@ip-172-31-42-82:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
     Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)
     Active: active (running) since Mon 2025-02-03 14:58:55 UTC; 43s ago
TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
   Main PID: 9280 (dockerd)
      Tasks: 9
     Memory: 20.8M (peak: 21.0M)
        CPU: 283ms
     CGroup: /system.slice/docker.service
             └─9280 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

Feb 03 14:58:54 ip-172-31-42-82 systemd[1]: Starting docker.service - Docker Application Container Engine...
Feb 03 14:58:54 ip-172-31-42-82 dockerd[9280]: time="2025-02-03T14:58:54.459165696Z" level=info msg="Starting up"
Feb 03 14:58:54 ip-172-31-42-82 dockerd[9280]: time="2025-02-03T14:58:54.460716870Z" level=info msg="OTEL tracing is not configured, using no-op tracer provider"
Feb 03 14:58:54 ip-172-31-42-82 dockerd[9280]: time="2025-02-03T14:58:54.460803189Z" level=info msg="detected 127.0.0.53 nameserver, assuming systemd-resolved, s>
Feb 03 14:58:54 ip-172-31-42-82 dockerd[9280]: time="2025-02-03T14:58:54.998513135Z" level=info msg="Loading containers: start."
Feb 03 14:58:55 ip-172-31-42-82 dockerd[9280]: time="2025-02-03T14:58:55.306202889Z" level=info msg="Loading containers: done."
Feb 03 14:58:55 ip-172-31-42-82 dockerd[9280]: time="2025-02-03T14:58:55.324331917Z" level=info msg="Docker daemon" commit=4c9b3b0 containerd-snapshotter=false s>
Feb 03 14:58:55 ip-172-31-42-82 dockerd[9280]: time="2025-02-03T14:58:55.324418705Z" level=info msg="Daemon has completed initialization"
Feb 03 14:58:55 ip-172-31-42-82 dockerd[9280]: time="2025-02-03T14:58:55.372189770Z" level=info msg="API listen on /run/docker.sock"
Feb 03 14:58:55 ip-172-31-42-82 systemd[1]: Started docker.service - Docker Application Container Engine.
lines 1-22/22 (END)
```

Install and Configure Jenkins

Install Jenkins

- sudo apt update
- sudo apt upgrade -y
- sudo apt install fontconfig openjdk-17-jre -y
- java -version
- sudo wget -O /usr/share/keyrings/jenkins-keyring.asc https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key

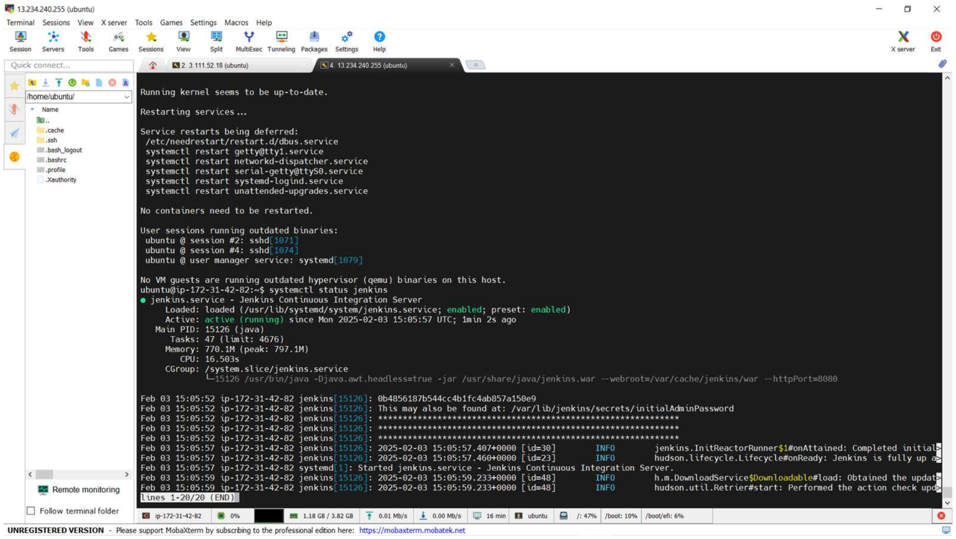

13.234.240.255 (ubuntu)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
2. 3.111.52.18 (ubuntu)
4. 13.234.240.255 (ubuntu)
/home/ubuntu/
Name
..
.cache
.ssh
.bash_logout
.bashrc
.profile
.Xauthority
Remote monitoring
Follow terminal folder
ip-172-31-42-82 0% 0.53 GB / 3.82 GB 0.01 Mb/s 0.00 Mb/s 12 min ubuntu /: 44% /boot: 10% /boot/efi: 6%
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

```
Adding debian:vTrus_Root_CA.pem
done.
Setting up openjdk-17-jre:amd64 (17.0.13+11-2ubuntu1~24.04) ...
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...
Processing triggers for libgdk-pixbuf-2.0-0:amd64 (2.42.10+dfsg-3ubuntu3.1) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Running kernel seems to be up-to-date.

Restarting services...

Service restarts being deferred:
 /etc/needrestart/restart.d/dbus.service
 systemctl restart getty@tty1.service
 systemctl restart networkd-dispatcher.service
 systemctl restart serial-getty@ttyS0.service
 systemctl restart systemd-logind.service
 systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
 ubuntu @ session #2: sshd[1071]
 ubuntu @ session #4: sshd[1074]
 ubuntu @ user manager service: systemd[1079]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-42-82:~$ sudo wget -O /usr/share/keyrings/jenkins-keyring.asc https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key

--2025-02-03 15:02:57--  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
Resolving pkg.jenkins.io (pkg.jenkins.io)... 151.101.154.133, 2a04:4e42:24::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)|151.101.154.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3175 (3.1K) [application/pgp-keys]
Saving to: '/usr/share/keyrings/jenkins-keyring.asc'

/usr/share/keyrings/jenkins-keyring.asc  100%[===========================================================================>]   3.10K  --.-KB/s    in 0s

2025-02-03 15:02:57 (46.5 MB/s) - '/usr/share/keyrings/jenkins-keyring.asc' saved [3175/3175]

ubuntu@ip-172-31-42-82:~$
```

- curl -fsSL https://pkg.jenkins.io/debian/jenkins.io-2023.key | sudo tee /usr/share/keyrings/jenkins-keyring.asc > /dev/null
- echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] https://pkg.jenkins.io/debian-stable binary/ | sudo tee /etc/apt/sources.list.d/jenkins.list > /dev/null
- sudo apt update
- sudo apt install jenkins -y

Start Jenkins

- systemctl status jenkins
- sudo systemctl enable Jenkins

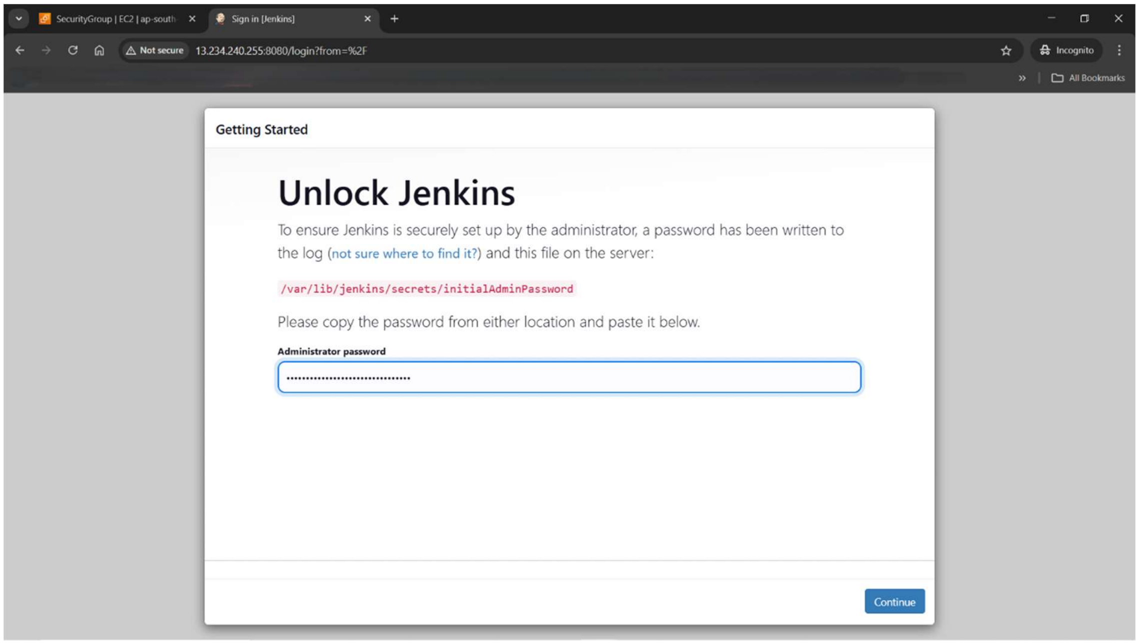

13.234.240.255 (ubuntu)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
2. 3.111.52.18 (ubuntu)
4. 13.234.240.255 (ubuntu)
/home/ubuntu/
Name
..
.cache
.ssh
.bash_logout
.bashrc
.profile
.Xauthority
Remote monitoring
Follow terminal folder
ip-172-31-42-82 0% 1.18 GB / 3.82 GB 0.01 Mb/s 0.00 Mb/s 16 min ubuntu /: 47% /boot: 10% /boot/efi: 6%
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

```
Running kernel seems to be up-to-date.

Restarting services...

Service restarts being deferred:
 /etc/needrestart/restart.d/dbus.service
 systemctl restart getty@tty1.service
 systemctl restart networkd-dispatcher.service
 systemctl restart serial-getty@ttyS0.service
 systemctl restart systemd-logind.service
 systemctl restart unattended-upgrades.service

No containers need to be restarted.

User sessions running outdated binaries:
 ubuntu @ session #2: sshd[1071]
 ubuntu @ session #4: sshd[1074]
 ubuntu @ user manager service: systemd[1079]

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-42-82:~$ systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
     Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
     Active: active (running) since Mon 2025-02-03 15:05:57 UTC; 1min 2s ago
   Main PID: 15126 (java)
      Tasks: 47 (limit: 4676)
     Memory: 770.1M (peak: 797.1M)
        CPU: 16.503s
     CGroup: /system.slice/jenkins.service
             └─15126 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

Feb 03 15:05:52 ip-172-31-42-82 jenkins[15126]: 0b4856187b544cc4b1fc4ab857a150e9
Feb 03 15:05:52 ip-172-31-42-82 jenkins[15126]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Feb 03 15:05:52 ip-172-31-42-82 jenkins[15126]: *************************************************************
Feb 03 15:05:52 ip-172-31-42-82 jenkins[15126]: *************************************************************
Feb 03 15:05:52 ip-172-31-42-82 jenkins[15126]: *************************************************************
Feb 03 15:05:57 ip-172-31-42-82 jenkins[15126]: 2025-02-03 15:05:57.407+0000 [id=30]        INFO        jenkins.InitReactorRunner$1#onAttained: Completed initial>
Feb 03 15:05:57 ip-172-31-42-82 jenkins[15126]: 2025-02-03 15:05:57.460+0000 [id=23]        INFO        hudson.lifecycle.Lifecycle#onReady: Jenkins is fully up a>
Feb 03 15:05:57 ip-172-31-42-82 systemd[1]: Started jenkins.service - Jenkins Continuous Integration Server.
Feb 03 15:05:59 ip-172-31-42-82 jenkins[15126]: 2025-02-03 15:05:59.233+0000 [id=48]        INFO        h.m.DownloadService$Downloadable#load: Obtained the updat>
Feb 03 15:05:59 ip-172-31-42-82 jenkins[15126]: 2025-02-03 15:05:59.233+0000 [id=48]        INFO        hudson.util.Retrier#start: Performed the action check upd>
lines 1-20/20 (END)
```

Get Jenkins Admin Password

- sudo cat /var/lib/jenkins/secrets/initialAdminPassword

➢ Login to Jenkins Web UI (http://<EC2-Public-IP>:8080) and enter the admin password.


SecurityGroup | EC2 | ap-south
Sign in [Jenkins]
Not secure
13.234.240.255:8080/login?from=%2F
Incognito
All Bookmarks

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

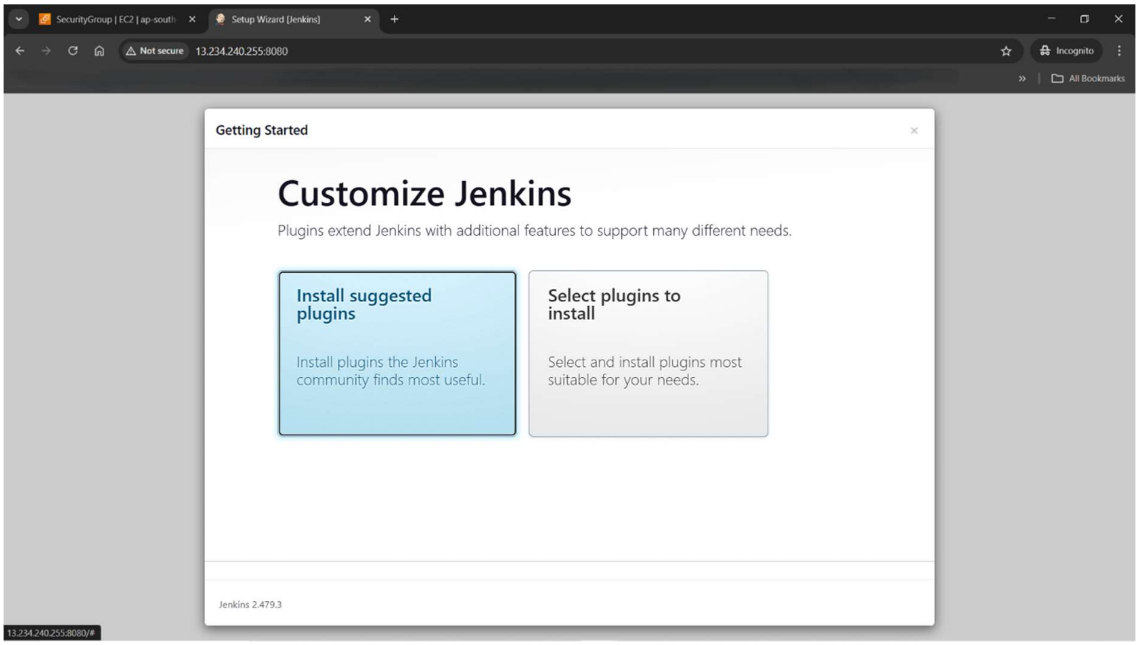
`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

································

Continue



Getting Started

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

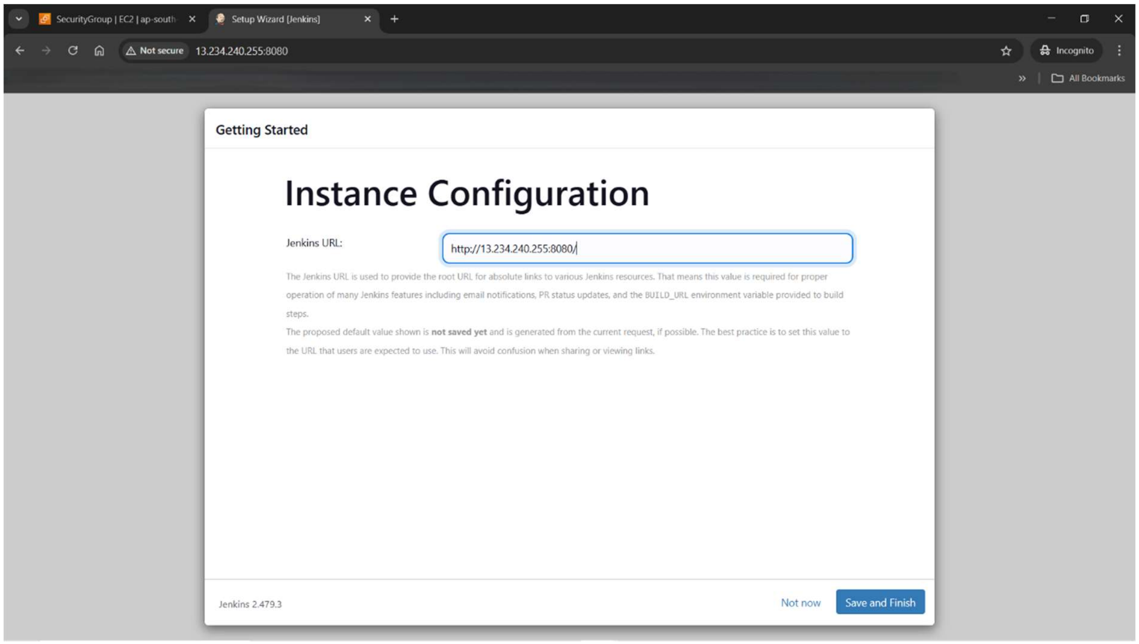
Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Jenkins 2.479.3



Getting Started

Instance Configuration

Jenkins URL: http://13.234.240.255:8080/

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the `BUILD_URL` environment variable provided to build steps.

The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

Jenkins 2.479.3

Add Jenkins User to Docker Group

- sudo usermod -aG docker jenkins
- sudo systemctl restart jenkins

Add Jenkins User to Docker Group

- sudo usermod -aG docker jenkins
- sudo systemctl restart jenkins

Grant Jenkins User Passwordless sudo Access

To allow the jenkins user to execute sudo commands without a password

Step 1: Edit the sudoers File

- sudo visudo

Step 2: Add This Line at the End

- jenkins ALL=(ALL) NOPASSWD: ALL

Step 3: Save and Exit

- Press **CTRL + X**
- Press **Y** to confirm
- Press **Enter**

Create a Simple Webpage

- mkdir webpage
- cd webpage
- echo "<h1>Welcome to My Webpage</h1>" > index.html

Create a Dockerfile

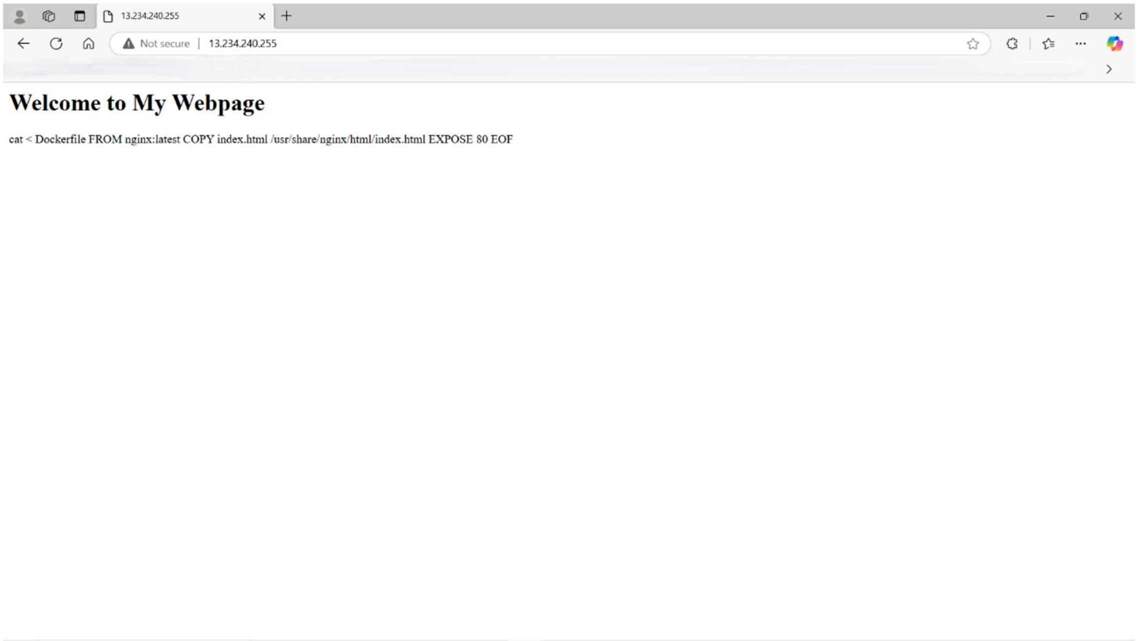
- Inside the `webpage/` directory, create a `Dockerfile`:

```
cat <<EOF > Dockerfile

FROM nginx:latest

COPY index.html /usr/share/nginx/html/index.html

EXPOSE 80

EOF
```

Build and run the Docker Image

- docker build -t my-webpage .
- docker run -d -p 80:80 my-webpage

- Access the webpage via http://your-ec2-public-ip



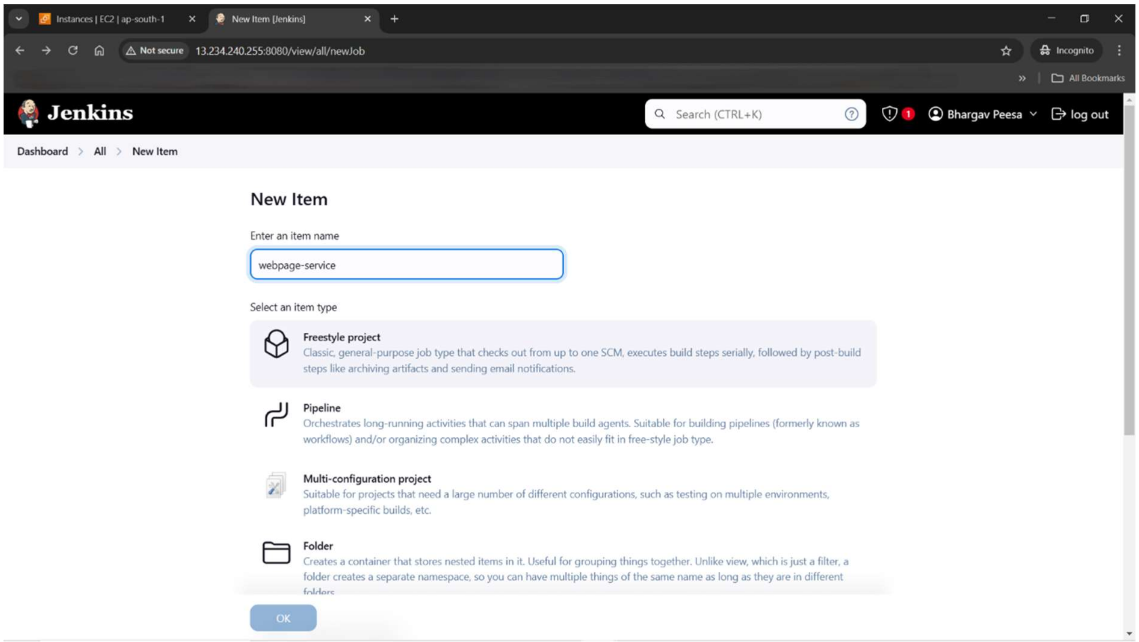
Welcome to My Webpage

cat < Dockerfile FROM nginx:latest COPY index.html /usr/share/nginx/html/index.html EXPOSE 80 EOF

Setup Jenkins for CI/CD - Automate Deployment with Jenkins

Configure Jenkins Job

- Go to Jenkins Dashboard → New Item
- Select Freestyle Project → Give it a name → Click OK



13.234.240.255:8080/view/all/newJob

New Item

Enter an item name: webpage-service

Select an item type

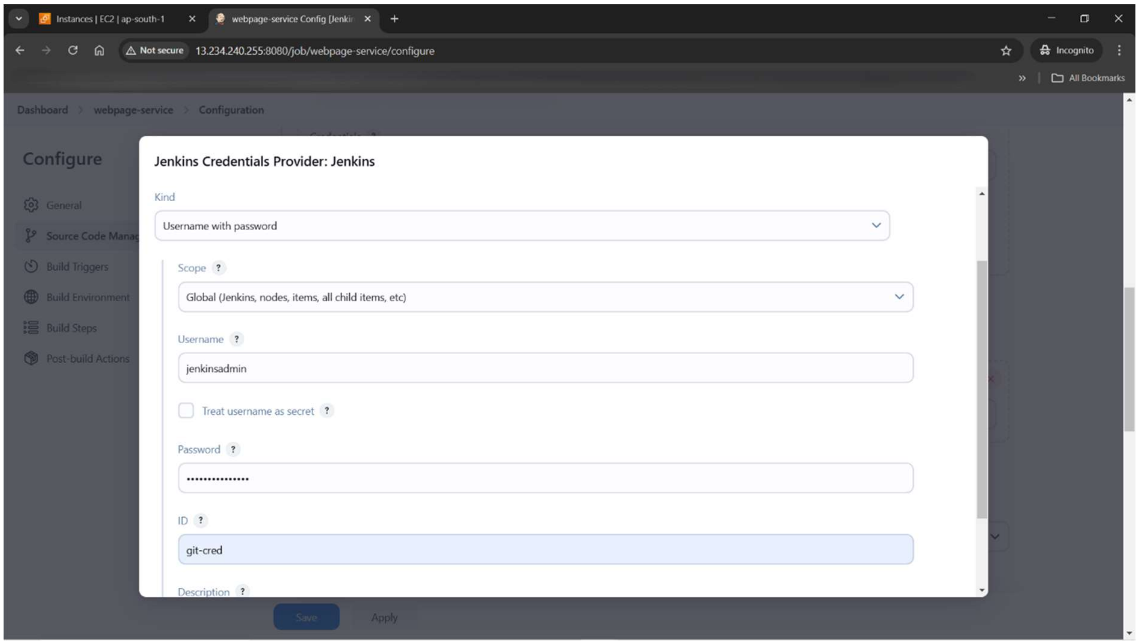
Freestyle project
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

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 separate namespace, so you can have multiple things of the same name as long as they are in different folders.

- Under Source Code Management, choose Git and enter the repository URL where your webpage code is stored (GitHub, Bitbucket, etc.).



13.234.240.255:8080/job/webpage-service/configure

Jenkins Credentials Provider: Jenkins

Kind: Username with password

Scope: Global (Jenkins, nodes, items, all child items, etc)

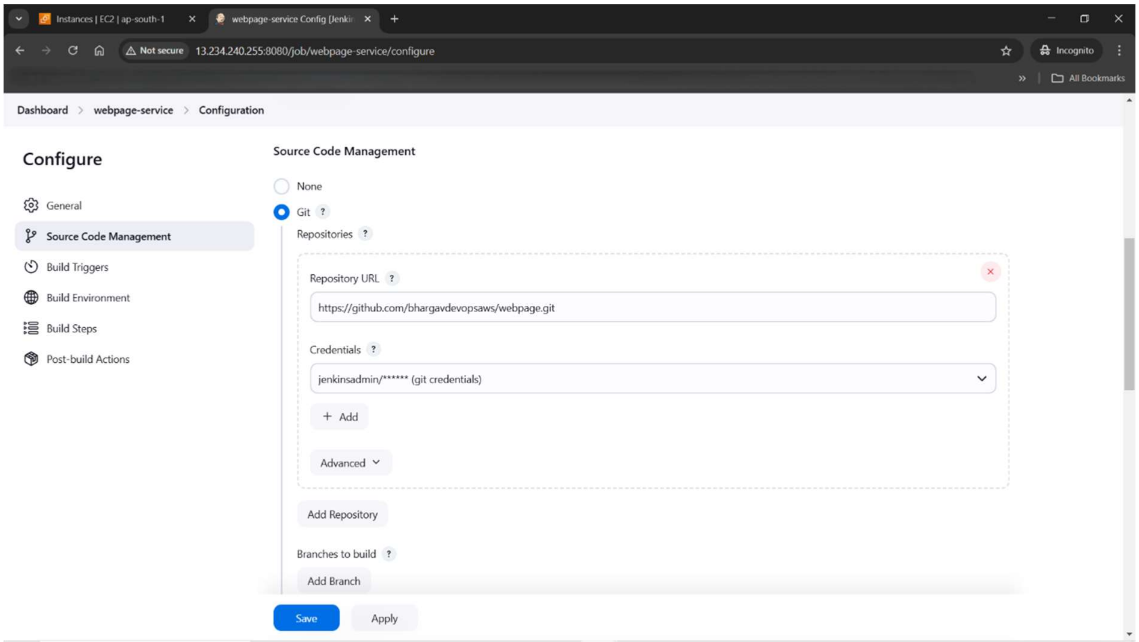
Username: jenkinsadmin

□ Treat username as secret

Password: •••••••••••••••

ID: git-cred

Description



Source Code Management

○ None

● Git

Repositories

Repository URL: https://github.com/bhargavdevopsaws/webpage.git

Credentials: jenkinsadmin/****** (git credentials)

Branches to build

- Under Build Steps, select Execute Shell and enter:

```
docker stop my-webpage || true

docker rm my-webpage || true

docker rmi my-webpage || true

docker build -t my-webpage .

docker run -d -p 80:80 my-webpage
```

- Save the job and build now to trigger deployment.

Push Webpage Code to GitHub

1. Make changes in index.html
2. Push the changes to GitHub

- Git init
- Git add .
- git clone https://github.com/bhargavdevopsaws/webpage.git
- touch webpage

- nano webpage

#!/bin/bash

# Stop and remove the existing container (ignore errors if it doesn't exist)

docker stop my-webpage || true

docker rm my-webpage || true

# Remove old Docker image

docker rmi my-webpage || true

# Build a new Docker image

docker build -t my-webpage .

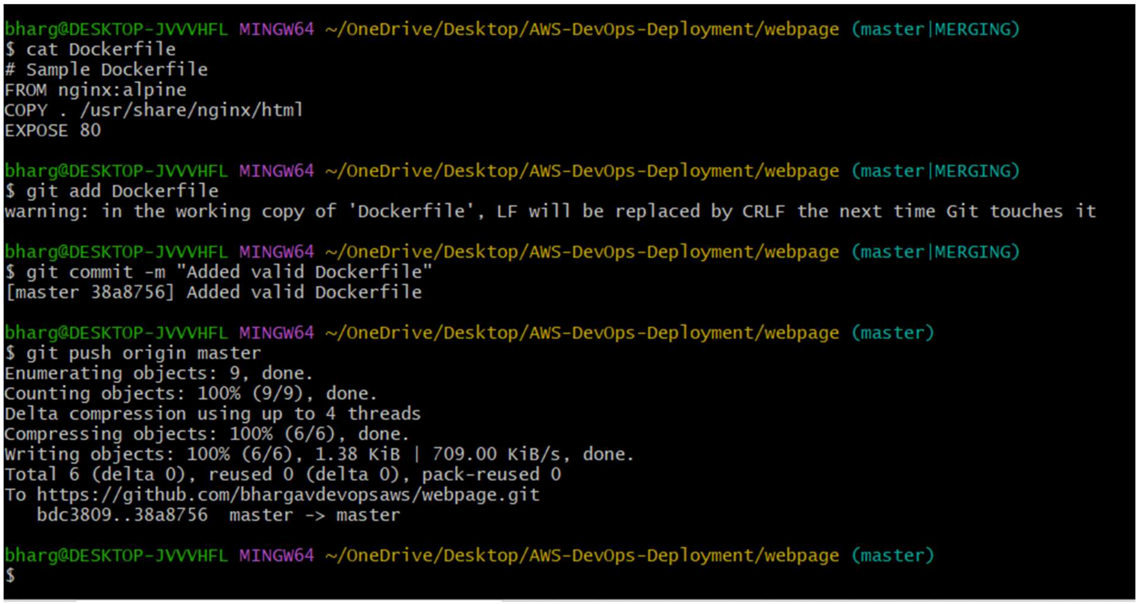
# Run the container in detached mode (-d) and expose port 80

docker run -d -p 80:80 --name my-webpage my-webpage

- git commit -m "Initial commit"
- git branch -M main
- git remote add origin git remote add origin https://github.com/your-username/webpage-ci-cd.git
- touch dockerfile
- nano dockerfile

```
# Sample Dockerfile
FROM nginx:alpine
COPY . /usr/share/nginx/html
EXPOSE 80
```

- git add dockerfile
- git commit -m "Added valid Dockerfile"
- git push -u origin main

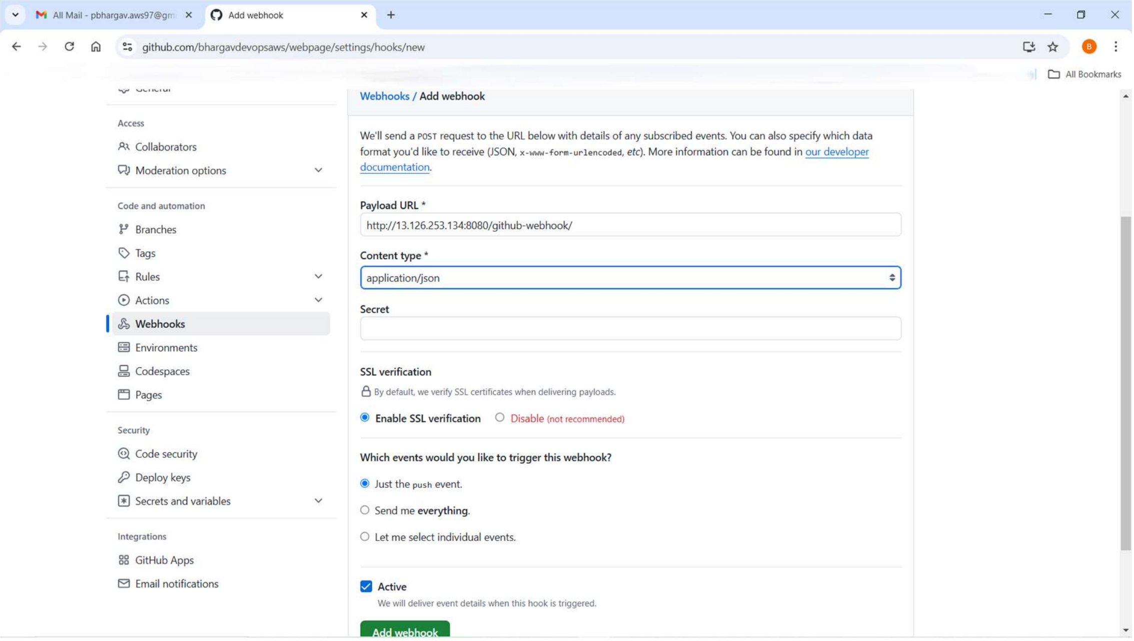


```
bharg@DESKTOP-JVVVHFL MINGW64 ~/OneDrive/Desktop/AWS-DevOps-Deployment/webpage (master|MERGING)
$ cat Dockerfile
# Sample Dockerfile
FROM nginx:alpine
COPY . /usr/share/nginx/html
EXPOSE 80

bharg@DESKTOP-JVVVHFL MINGW64 ~/OneDrive/Desktop/AWS-DevOps-Deployment/webpage (master|MERGING)
$ git add Dockerfile
warning: in the working copy of 'Dockerfile', LF will be replaced by CRLF the next time Git touches it

bharg@DESKTOP-JVVVHFL MINGW64 ~/OneDrive/Desktop/AWS-DevOps-Deployment/webpage (master|MERGING)
$ git commit -m "Added valid Dockerfile"
[master 38a8756] Added valid Dockerfile

bharg@DESKTOP-JVVVHFL MINGW64 ~/OneDrive/Desktop/AWS-DevOps-Deployment/webpage (master)
$ git push origin master
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 1.38 KiB | 709.00 KiB/s, done.
Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/bhargavdevopsaws/webpage.git
   bdc3809..38a8756  master -> master

bharg@DESKTOP-JVVVHFL MINGW64 ~/OneDrive/Desktop/AWS-DevOps-Deployment/webpage (master)
$
```

Set Up Webhook for CI/CD

To automate deployment whenever you push changes to Git:

- **GitHub Webhook**:
  - Go to **GitHub Repo** → **Settings** → **Webhooks**
  - Click **Add Webhook** and set:
    - Payload URL: http://your-ec2-public-ip:8080/github-webhook/
    - Content type: application/json
    - Trigger: Push
  - In Jenkins, install **GitHub Integration Plugin**
  - Enable **Build when a change is pushed to GitHub** in the Jenkins job.



github.com/bhargavdevopsaws/webpage/settings/hooks/new

Webhooks / Add webhook

We'll send a `POST` request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, `x-www-form-urlencoded`, *etc*). More information can be found in our developer documentation.

Payload URL *
http://13.126.253.134:8080/github-webhook/

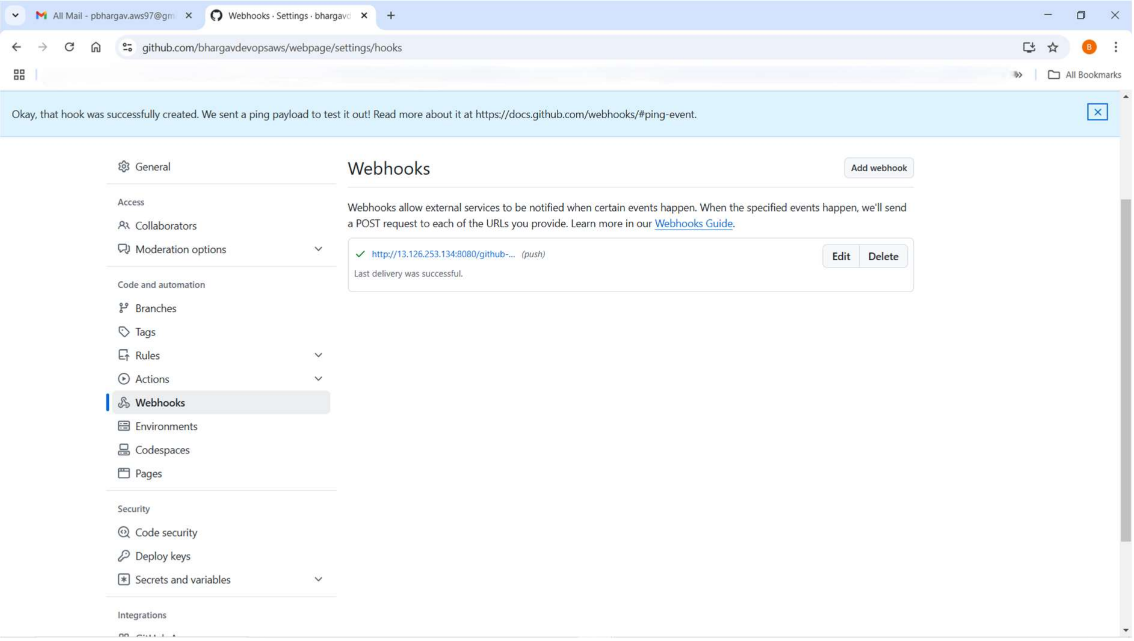
Content type *
application/json

Secret

SSL verification
By default, we verify SSL certificates when delivering payloads.
● Enable SSL verification ○ Disable (not recommended)

Which events would you like to trigger this webhook?
● Just the `push` event.
○ Send me **everything**.
○ Let me select individual events.

☑ Active
We will deliver event details when this hook is triggered.



Okay, that hook was successfully created. We sent a ping payload to test it out! Read more about it at https://docs.github.com/webhooks/#ping-event.

Webhooks

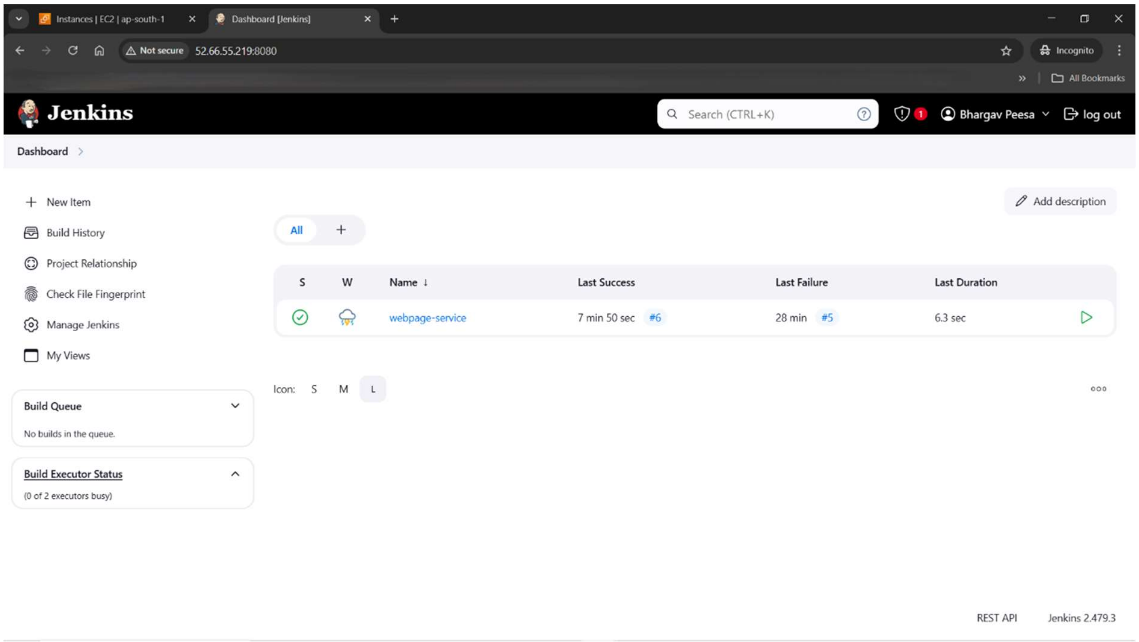
Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our Webhooks Guide.

✓ http://13.126.253.134:8080/github-... (push)

Last delivery was successful.

Run the Pipeline

- **Click "Build Now"** in Jenkins.
- It will fetch the latest code, build the Docker image, and deploy the container.



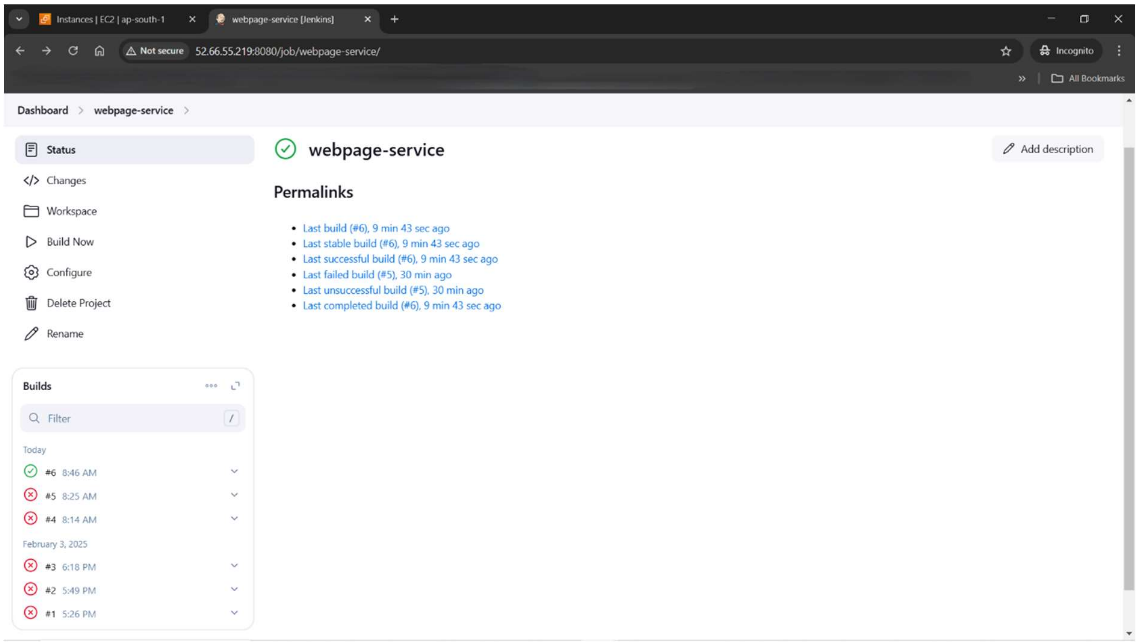
S	W	Name ↓	Last Success	Last Failure	Last Duration
✓		webpage-service	7 min 50 sec #6	28 min #5	6.3 sec

Build Queue

No builds in the queue.

Build Executor Status

(0 of 2 executors busy)


Instances | EC2 | ap-south-1
webpage-service [Jenkins]
Not secure
52.66.55.219:8080/job/webpage-service/
Incognito
All Bookmarks
Dashboard > webpage-service >
Status
Changes
Workspace
Build Now
Configure
Delete Project
Rename
Add description
Builds
Filter

webpage-service

Permalinks

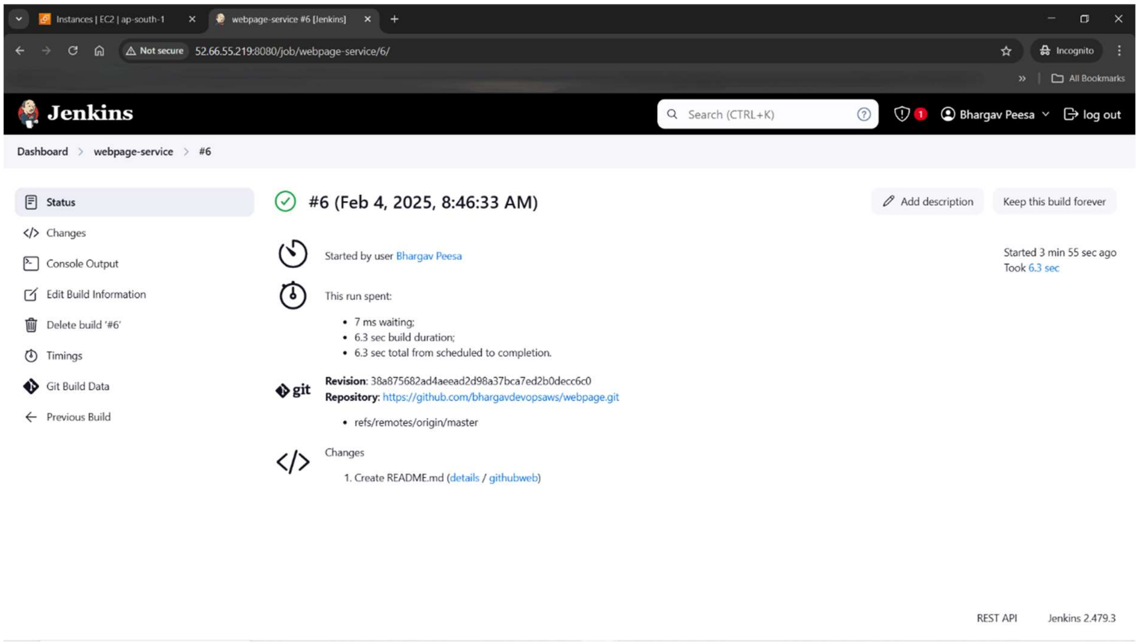
- Last build (#6), 9 min 43 sec ago
- Last stable build (#6), 9 min 43 sec ago
- Last successful build (#6), 9 min 43 sec ago
- Last failed build (#5), 30 min ago
- Last unsuccessful build (#5), 30 min ago
- Last completed build (#6), 9 min 43 sec ago

Today

✓ #6 8:46 AM
✗ #5 8:25 AM
✗ #4 8:14 AM

February 3, 2025

✗ #3 6:18 PM
✗ #2 5:49 PM
✗ #1 5:26 PM



#6 (Feb 4, 2025, 8:46:33 AM)

Started by user Bhargav Peesa

Started 3 min 55 sec ago
Took 6.3 sec

This run spent:

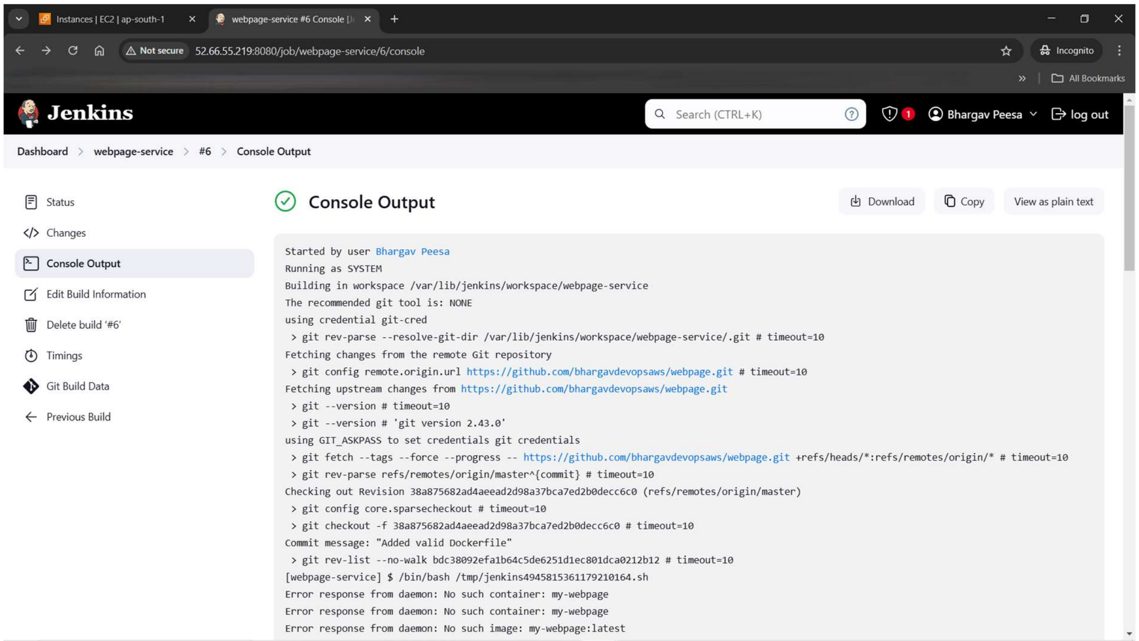
- 7 ms waiting;
- 6.3 sec build duration;
- 6.3 sec total from scheduled to completion.

Revision: 38a875682ad4aeead2d98a37bca7ed2b0decc6c0
Repository: https://github.com/bhargavdevopsaws/webpage.git

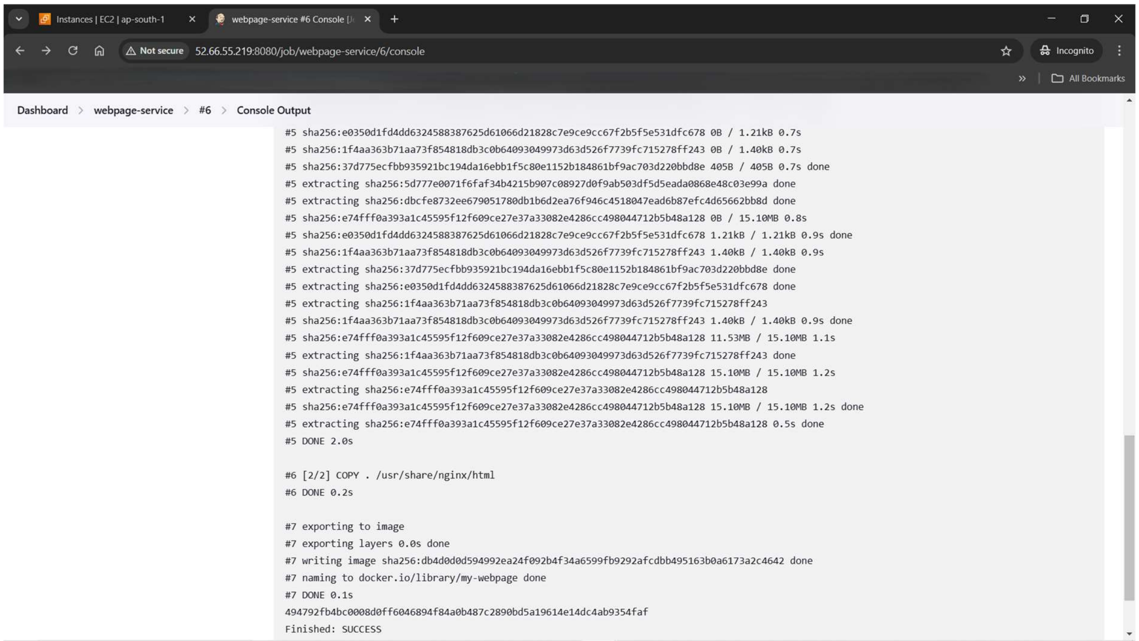
- refs/remotes/origin/master

Changes

1. Create README.md (details / githubweb)



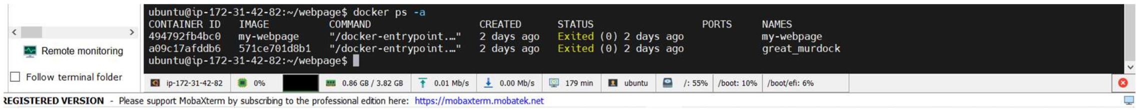
```
Started by user Bhargav Peesa
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/webpage-service
The recommended git tool is: NONE
using credential git-cred
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/webpage-service/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/bhargavdevopsaws/webpage.git # timeout=10
Fetching upstream changes from https://github.com/bhargavdevopsaws/webpage.git
 > git --version # timeout=10
 > git --version # 'git version 2.43.0'
using GIT_ASKPASS to set credentials git credentials
 > git fetch --tags --force --progress -- https://github.com/bhargavdevopsaws/webpage.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision 38a875682ad4aeead2d98a37bca7ed2b0decc6c0 (refs/remotes/origin/master)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 38a875682ad4aeead2d98a37bca7ed2b0decc6c0 # timeout=10
Commit message: "Added valid Dockerfile"
 > git rev-list --no-walk bdc38092efa1b64c5de6251d1ec801dca0212b12 # timeout=10
[webpage-service] $ /bin/bash /tmp/jenkins4945815361179210164.sh
Error response from daemon: No such container: my-webpage
Error response from daemon: No such container: my-webpage
Error response from daemon: No such image: my-webpage:latest
```


Instances | EC2 | ap-south-1
webpage-service #6 Console [J
Not secure
52.66.55.219:8080/job/webpage-service/6/console
Incognito
All Bookmarks
Dashboard > webpage-service > #6 > Console Output

```
#5 sha256:e0350d1fd4dd6324588387625d61066d21828c7e9ce9cc67f2b5f5e531dfc678 0B / 1.21kB 0.7s
#5 sha256:1f4aa363b71aa73f854818db3c0b64093049973d63d526f7739fc715278ff243 0B / 1.40kB 0.7s
#5 sha256:37d775ecfbb935921bc194da16ebb1f5c80e1152b184861bf9ac703d220bbd8e 405B / 405B 0.7s done
#5 extracting sha256:5d777e0071f6faf34b4215b907c08927d0f9ab503df5d5eada0868e48c03e99a done
#5 extracting sha256:dbcfe8732ee679051780db1b6d2ea76f946c4518047ead6b87efc4d65662bb8d done
#5 sha256:e74fff0a393a1c45595f12f609ce27e37a33082e4286cc498044712b5b48a128 0B / 15.10MB 0.8s
#5 sha256:e0350d1fd4dd6324588387625d61066d21828c7e9ce9cc67f2b5f5e531dfc678 1.21kB / 1.21kB 0.9s done
#5 sha256:1f4aa363b71aa73f854818db3c0b64093049973d63d526f7739fc715278ff243 1.40kB / 1.40kB 0.9s
#5 extracting sha256:37d775ecfbb935921bc194da16ebb1f5c80e1152b184861bf9ac703d220bbd8e done
#5 extracting sha256:e0350d1fd4dd6324588387625d61066d21828c7e9ce9cc67f2b5f5e531dfc678 done
#5 extracting sha256:1f4aa363b71aa73f854818db3c0b64093049973d63d526f7739fc715278ff243
#5 sha256:1f4aa363b71aa73f854818db3c0b64093049973d63d526f7739fc715278ff243 1.40kB / 1.40kB 0.9s done
#5 sha256:e74fff0a393a1c45595f12f609ce27e37a33082e4286cc498044712b5b48a128 11.53MB / 15.10MB 1.1s
#5 extracting sha256:1f4aa363b71aa73f854818db3c0b64093049973d63d526f7739fc715278ff243 done
#5 sha256:e74fff0a393a1c45595f12f609ce27e37a33082e4286cc498044712b5b48a128 15.10MB / 15.10MB 1.2s
#5 extracting sha256:e74fff0a393a1c45595f12f609ce27e37a33082e4286cc498044712b5b48a128
#5 sha256:e74fff0a393a1c45595f12f609ce27e37a33082e4286cc498044712b5b48a128 15.10MB / 15.10MB 1.2s done
#5 extracting sha256:e74fff0a393a1c45595f12f609ce27e37a33082e4286cc498044712b5b48a128 0.5s done
#5 DONE 2.0s

#6 [2/2] COPY . /usr/share/nginx/html
#6 DONE 0.2s

#7 exporting to image
#7 exporting layers 0.0s done
#7 writing image sha256:db4d0d0d594992ea24f092b4f34a6599fb9292afcdbb495163b0a6173a2c4642 done
#7 naming to docker.io/library/my-webpage done
#7 DONE 0.1s
494792fb4bc0008d0ff6046894f84a0b487c2890bd5a19614e14dc4ab9354faf
Finished: SUCCESS
```

Check Running Containers

- docker ps -a



```
ubuntu@ip-172-31-42-82:~/webpage$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED      STATUS                    PORTS     NAMES
494792fb4bc0   my-webpage     "/docker-entrypoint.…"   2 days ago   Exited (0) 2 days ago               my-webpage
a09c17afddb6   571ce701d8b1   "/docker-entrypoint.…"   2 days ago   Exited (0) 2 days ago               great_murdock
ubuntu@ip-172-31-42-82:~/webpage$ 
```

REGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net