

# Online Shopping

Clone the below mentioned repo and deploy the application. (Run the application in port 80 [HTTP])

Repo URL : <https://github.com/sriram-R-krishnan/devops-build.git>

This project demonstrates React application deployment using Docker compose and Jenkins for CI/CD.

## Architecture Overview

Application: React

Containerization: Docker

CI/CD Pipeline: Jenkins

Cloud Provider: AWS

Monitoring & Observability: Prometheus and Grafana dashboards

## Setup Instructions

### Prerequisites

- Docker
- Docker Compose
- AWS CLI
- Jenkins

## CI/CD Pipeline Explanation

### Pipeline Stages

1. Code Checkout
2. Build Docker Image
3. Push Docker Image to Docker Hub
4. Deploy to Kubernetes

## Pipeline Flow

- On code push to the Git repository, Jenkins automatically triggers the build via webhooks.
- The Docker image is built and pushed to Docker Hub.
- Jenkins Multibranch Pipeline deploy the application on the server.

## Project Submission:

Github repo URL :

[https://github.com/udhayakumarethiraj-git/Project\\_3\\_Online\\_shopping.git](https://github.com/udhayakumarethiraj-git/Project_3_Online_shopping.git)

Deployed site URL

Prod : <http://13.235.54.222/>

Dev : <http://13.235.54.222:8080/>

Docker images name

Dev : udhayakumarethiraj/dev:latest

Prod : udhayakumarethiraj/prod:latest

## Project Screenshots and Proof of Implementation

This section includes screenshots that provide visual proof of the successful implementation and execution of the project. The screenshots demonstrate key stages and components of the deployment, including:

- ❖ Source code repositories and project structure
- ❖ Successful Docker image build and push to Docker Hub
- ❖ Jenkins CI/CD pipeline execution (checkout, build, push, and deploy stages)
- ❖ Application accessibility via the exposed application URL
- ❖ Prometheus metrics collection and Grafana dashboards displaying real-time cluster and application health

These screenshots collectively validate the end-to-end workflow, from source code to a production-ready, containerized application deployed on Kubernetes.

## Project Configuration and Deployment Files

The repository includes all necessary configuration files for this project, including the Dockerfile, Jenkinsfile, Prometheus datasource files, as well as build.sh and deploy.sh scripts.

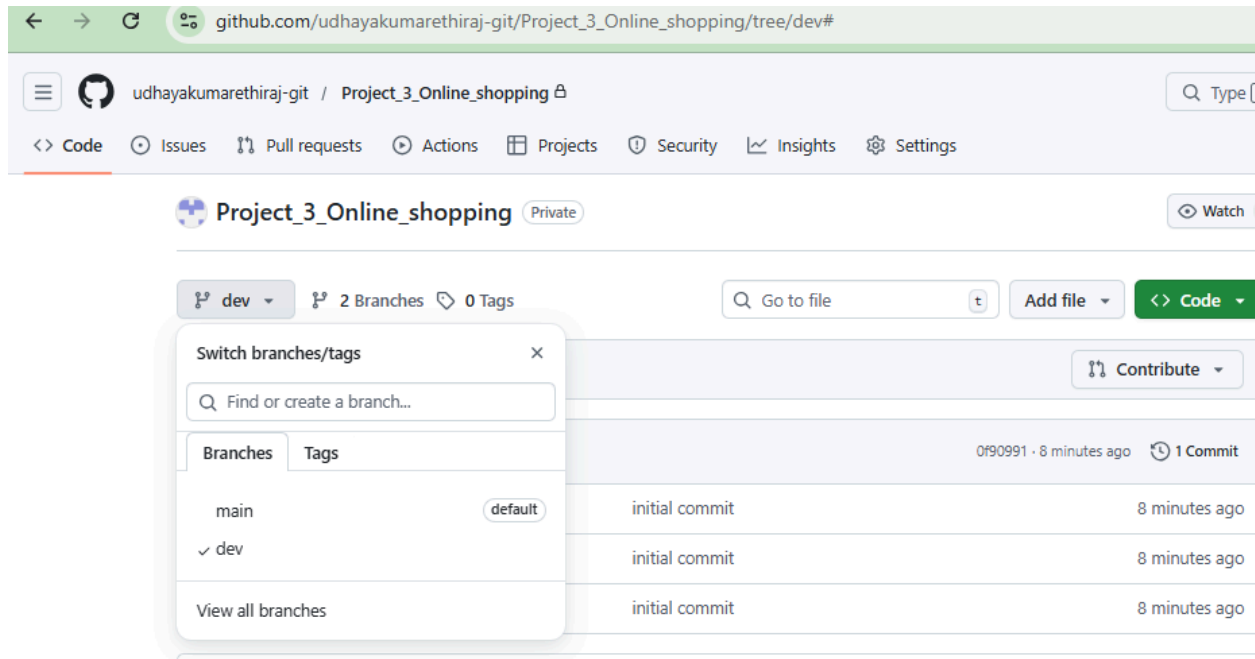
URL : [Link](#)

## Stage : Application

Git repository have been created and cloned with 2 branches named main and dev  
Project\_3\_Online\_shopping (Private):

URL : [https://github.com/udhayakumarethiraj-git/Project\\_3\\_Online\\_shopping.git](https://github.com/udhayakumarethiraj-git/Project_3_Online_shopping.git)

```
ubuntu@ip-172-31-28-193:~/projects/project_3$ git pull origin main
From github.com:udhayakumarethiraj-git/Project_3_Online_shopping
 * branch          main          -> FETCH_HEAD
Already up to date.
ubuntu@ip-172-31-28-193:~/projects/project_3$ ll
total 24
drwxrwxr-x 4 ubuntu ubuntu 4096 Jan 29 08:12 ./
drwxrwxr-x 6 ubuntu ubuntu 4096 Jan 28 06:15 ../
drwxrwxr-x 8 ubuntu ubuntu 4096 Jan 29 08:28 .git/
-rw-rw-r-- 1 ubuntu ubuntu 237 Jan 28 06:21 Dockerfile
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 29 08:21 devops-build/
-rw-rw-r-- 1 ubuntu ubuntu 146 Jan 28 06:44 docker-compose.yml
ubuntu@ip-172-31-28-193:~/projects/project_3$ git checkout dev
Switched to branch 'dev'
ubuntu@ip-172-31-28-193:~/projects/project_3$ git pull origin dev
From github.com:udhayakumarethiraj-git/Project_3_Online_shopping
 * branch          dev          -> FETCH_HEAD
Already up to date.
ubuntu@ip-172-31-28-193:~/projects/project_3$ git remote -v
origin  git@github.com:udhayakumarethiraj-git/Project_3_Online_shopping.git (fetch)
origin  git@github.com:udhayakumarethiraj-git/Project_3_Online_shopping.git (push)
ubuntu@ip-172-31-28-193:~/projects/project_3$ vi Dockerfile
ubuntu@ip-172-31-28-193:~/projects/project_3$ ll
total 24
drwxrwxr-x 4 ubuntu ubuntu 4096 Jan 29 08:30 ./
drwxrwxr-x 6 ubuntu ubuntu 4096 Jan 28 06:15 ../
drwxrwxr-x 8 ubuntu ubuntu 4096 Jan 29 08:29 .git/
-rw-rw-r-- 1 ubuntu ubuntu 237 Jan 28 06:21 Dockerfile
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 29 08:21 devops-build/
-rw-rw-r-- 1 ubuntu ubuntu 146 Jan 28 06:44 docker-compose.yml
ubuntu@ip-172-31-28-193:~/projects/project_3$
```



## Stage : Docker

Containerize the application by creating a Dockerfile, building the image, and testing the output.

A Dockerfile was created to containerize the application, the Docker image was successfully built, and the output was verified by running the Docker container.

```
ubuntu@ip-172-31-28-193:~/projects/project_3$ ll
total 24
drwxrwxr-x 4 ubuntu ubuntu 4096 Jan 29 08:30 ./
drwxrwxr-x 6 ubuntu ubuntu 4096 Jan 28 06:15 ../
drwxrwxr-x 8 ubuntu ubuntu 4096 Jan 29 08:29 .git/
-rw-rw-r-- 1 ubuntu ubuntu 237 Jan 28 06:21 Dockerfile
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 29 08:21 devops-build/
-rw-rw-r-- 1 ubuntu ubuntu 146 Jan 28 06:44 docker-compose.yml
ubuntu@ip-172-31-28-193:~/projects/project_3$ more Dockerfile
FROM nginx:alpine

# Remove default nginx static files
RUN rm -rf /usr/share/nginx/html/*

# Copy Vite build output to nginx
COPY devops-build/build /usr/share/nginx/html

# Expose port 80
EXPOSE 80

CMD ["nginx", "-g", "daemon off;"]

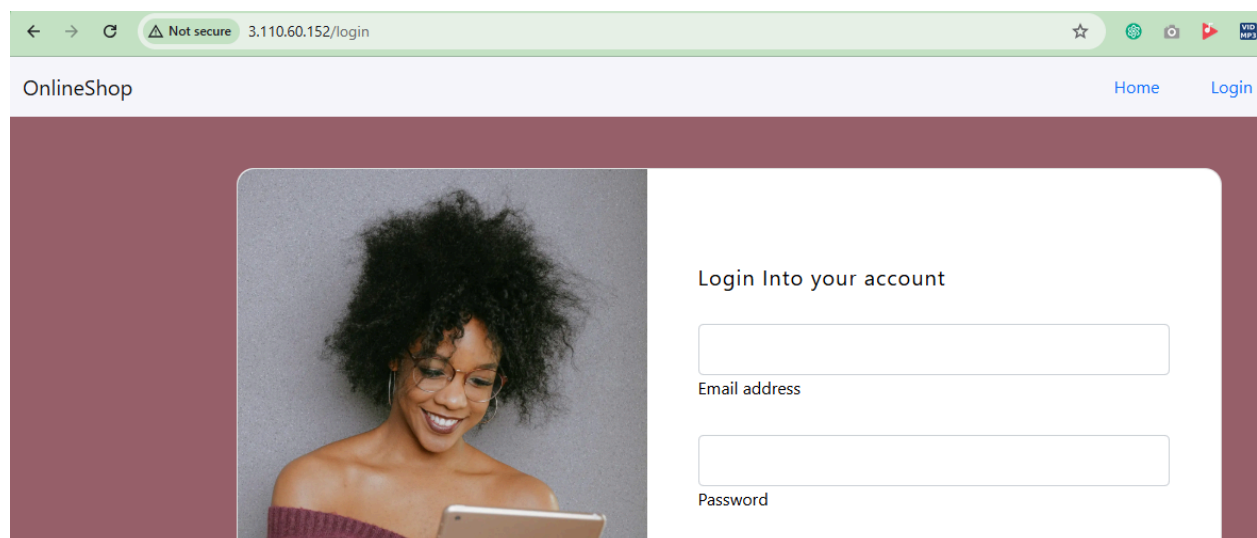
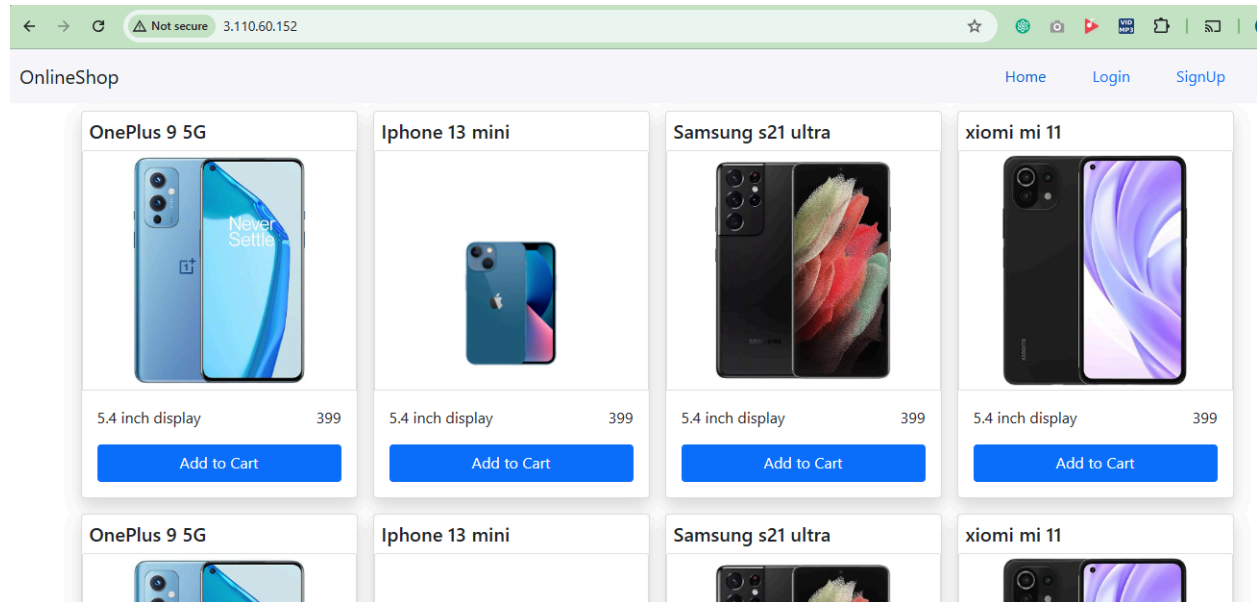
ubuntu@ip-172-31-28-193:~/projects/project_3$
```

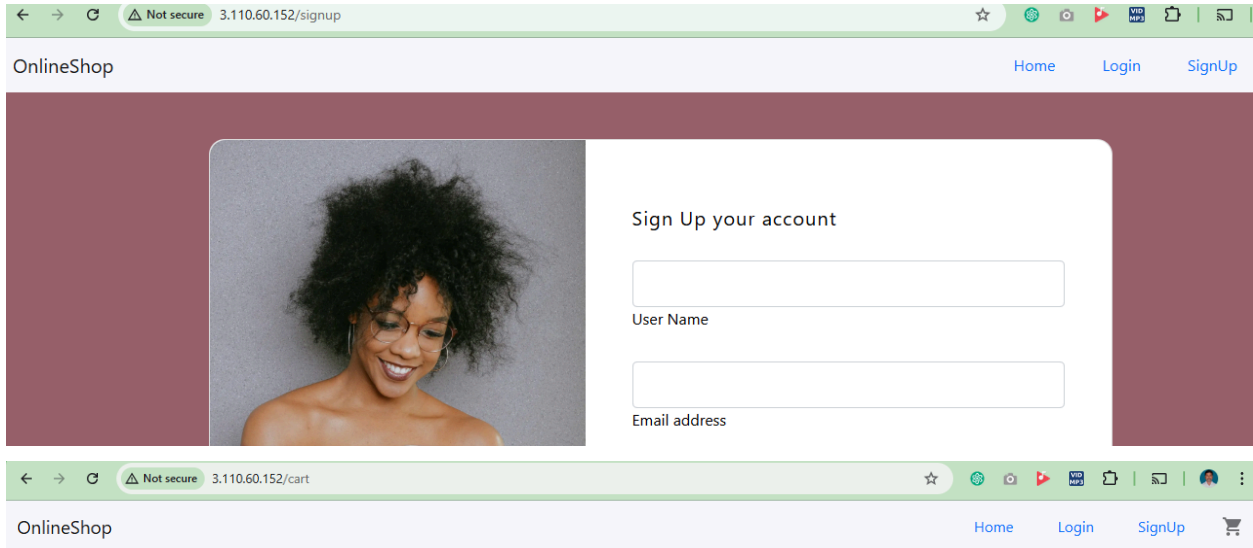
```

ubuntu@ip-172-31-28-193:~/projects/project_3$ docker images
IMAGE ID DISK USAGE CONTENT SIZE EXTRA
devops:latest f4b1d7765243 95.6MB 26.7MB U
ubuntu@ip-172-31-28-193:~/projects/project_3$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
e861c0fe234a devops:latest "/docker-entrypoint..." 26 hours ago Up 22 minutes 0.0.0.0:80->80/tcp, [::]:80->80/tcp devops-bu
ild
ubuntu@ip-172-31-28-193:~/projects/project_3$

```

## Output Test result





## Stage : Bash Scripting

[build.sh](#) file created for build the docker image.

```
ubuntu@ip-172-31-28-193:~/projects/project_3$ ll
total 28
drwxrwxr-x 4 ubuntu ubuntu 4096 Jan 29 08:41 ./
drwxrwxr-x 6 ubuntu ubuntu 4096 Jan 28 06:15 ../
drwxrwxr-x 8 ubuntu ubuntu 4096 Jan 29 08:29 .git/
-rw-rw-r-- 1 ubuntu ubuntu 237 Jan 28 06:21 Dockerfile
-rwxrwxr-x 1 ubuntu ubuntu 383 Jan 29 08:41 build.sh*
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 29 08:21 devops-build/
-rw-rw-r-- 1 ubuntu ubuntu 146 Jan 28 06:44 docker-compose.yml
ubuntu@ip-172-31-28-193:~/projects/project_3$ more build.sh
#!/bin/bash

# Exit immediately if a command fails
set -e

# Variables
IMAGE_NAME="devops-build"
IMAGE_TAG="latest"
DOCKERFILE_PATH="Dockerfile"
BUILD_CONTEXT="."

echo "🔧 Building Docker image..."
echo "Image: ${IMAGE_NAME}:${IMAGE_TAG}"

docker build \
  -t ${IMAGE_NAME}:${IMAGE_TAG} \
  -f ${DOCKERFILE_PATH} \
  ${BUILD_CONTEXT}

echo "✅ Docker image built successfully!"

ubuntu@ip-172-31-28-193:~/projects/project_3$
```

```

ubuntu@ip-172-31-28-193:~/projects/project_3$ ./build.sh
Building Docker image...
Image: devops-build:latest
[+] Building 0.9s (8/8) FINISHED
-> [internal] load build definition from Dockerfile
-> => transferring dockerfile: 276B
-> [internal] load metadata for docker.io/library/nginx:alpine
-> [internal] load .dockerignore
-> => transferring context: 2B
-> [1/3] FROM docker.io/library/nginx:alpine@sha256:7d7a15b8a280c661051955f14c2b91fed3e23724ddb18d2f53e8b44e74ab37a
-> => resolve docker.io/library/nginx:alpine@sha256:7d7a15b8a280c661051955f14c2b91fed3e23724ddb18d2f53e8b44e74ab37a
-> [internal] load build context
-> => transferring context: 1.12kB
-> CACHED [2/3] RUN rm -rf /usr/share/nginx/html/*
-> CACHED [3/3] COPY devops-build/build /usr/share/nginx/html
-> exporting to image
-> => exporting layers
-> => exporting manifest sha256:05475ca27954a40fd97591cd6b02df3510c78e4b31fa36648a2727357626e7c1
-> => exporting config sha256:bc2850f174cf69bbf46584d907585ab5b22bb2685731f1541ca5d237ec26bb9e
-> => exporting attestation manifest sha256:8cf70c40c9dca536be3913b9d34ed50c67f0e142b7a58fa5d2024c5b574ae2b7
-> => exporting manifest list sha256:61b41fca6dcae5ba0ff84347ea88af7ddee8c3e3e33553fe09d5b68ed936c26
-> => naming to docker.io/library/devops-build:latest
-> => unpacking to docker.io/library/devops-build:latest
[+] Docker image built successfully!
ubuntu@ip-172-31-28-193:~/projects/project_3$ docker images

```

IMAGE	ID	DISK USAGE	CONTENT SIZE	EXTRA
devops-build:latest	61b41fca6dca	95.6MB	26.7MB	

```

ubuntu@ip-172-31-28-193:~/projects/project_3$

```

[deploy.sh](#) file created to deploy the docker image on remote server

```

ubuntu@ip-172-31-28-193:~/projects/project_3$ ll
total 36
drwxrwxr-x 4 ubuntu ubuntu 4096 Jan 29 09:36 ./
drwxrwxr-x 6 ubuntu ubuntu 4096 Jan 28 06:15 ../
drwxrwxr-x 8 ubuntu ubuntu 4096 Jan 29 08:29 .git/
-rw-rw-r-- 1 ubuntu ubuntu 237 Jan 28 06:21 Dockerfile
-rwxrwxr-x 1 ubuntu ubuntu 383 Jan 29 08:41 build.sh*
-rwxrwxr-x 1 ubuntu ubuntu 1396 Jan 29 09:17 deploy.old*
-rwxrwxr-x 1 ubuntu ubuntu 1239 Jan 29 09:36 deploy.sh*
drwxrwxr-x 3 ubuntu ubuntu 4096 Jan 29 08:21 devops-build/
-rw-rw-r-- 1 ubuntu ubuntu 152 Jan 29 09:35 docker-compose.yml
ubuntu@ip-172-31-28-193:~/projects/project_3$

```

```

ubuntu@ip-172-31-28-193:~/projects/project_3$ chmod +x deploy.sh
ubuntu@ip-172-31-28-193:~/projects/project_3$ ./deploy.sh
Checking local Docker image...
Saving Docker image...
Copying files to remote server...
devops-build.tar                                100% 25MB 152.5MB/s 00:00
docker-compose.yml                             100% 152 292.4KB/s 00:00
Deploying on remote server...
Pseudo-terminal will not be allocated because stdin is not a terminal.
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-1044-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Thu Jan 29 09:36:17 UTC 2026

System load: 0.0          Processes:              153
Usage of /: 37.0% of 11.45GB Users logged in:          1
Memory usage: 12%        IPv4 address for enp3s0: 172.31.27.2
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

7 updates can be applied immediately.
3 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

2 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

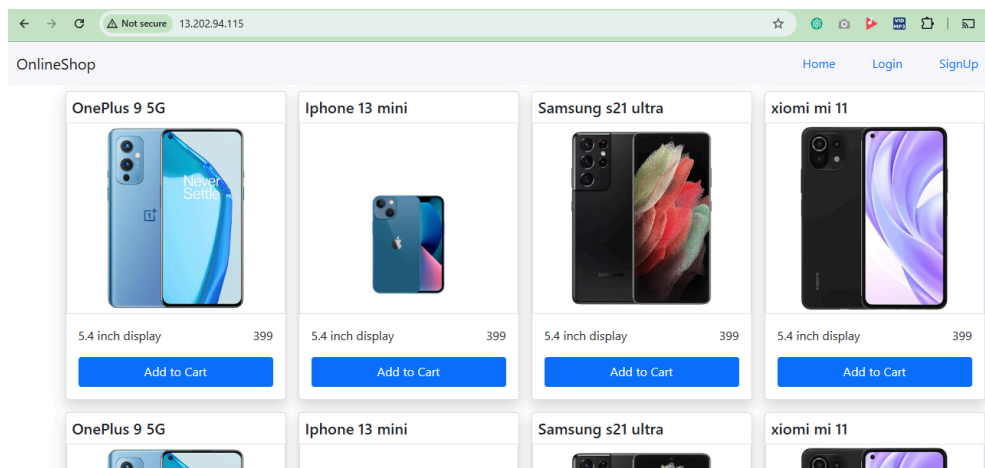
New release '24.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

```

```

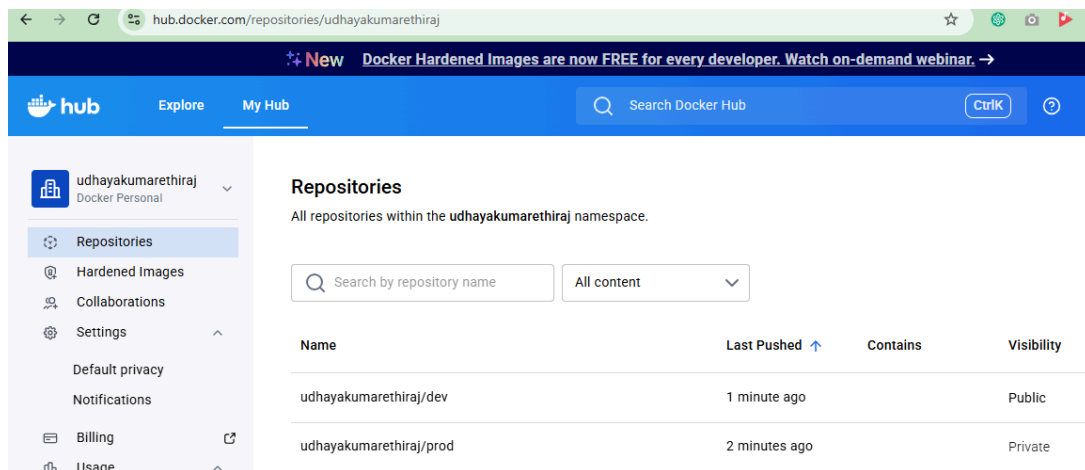
📦 Loading Docker image...
Loaded image: devops-build:latest
● Stopping existing containers...
time="2026-01-29T09:36:19Z" level=warning msg="/home/ubuntu/project_3/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
▶ Starting containers...
time="2026-01-29T09:36:19Z" level=warning msg="/home/ubuntu/project_3/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
Network project_3_default Creating
Network project_3_default Created
Container devops-build Creating
Container devops-build Created
Container devops-build Starting
Container devops-build Started
📦 Running containers:
time="2026-01-29T09:36:20Z" level=warning msg="/home/ubuntu/project_3/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
NAME          IMAGE          COMMAND          SERVICE    CREATED      STATUS      PORTS
devops-build   devops-build:latest  "/docker-entrypoint..."  app        1 second ago Up Less than a second  0.0.0.0:80->80/tcp, [::]:80->80/tcp
🎉 Deployment successful!

```



## Stage : Docker HUB

Docker hub repo created





## EC2 Instance

EC2 Instance launched to deploy the application

The screenshot shows the AWS Management Console interface for an EC2 instance. The left sidebar contains navigation links for EC2, Instances, Dashboard, AWS Global View, Events, and a list of instance types and plans. The main content area displays the 'Instance summary for i-0c9914620391fe54f (Server)'. The instance is in a 'Running' state. Key details include: Instance ID (i-0c9914620391fe54f), Public IPv4 address (13.235.54.222), Private IPv4 addresses (172.31.27.2), Public DNS (ec2-13-235-54-222.ap-south-1.compute.amazonaws.com), Private IP DNS name (ip-172-31-27-2.ap-south-1.compute.internal), and Hostname type (IP name: ip-172-31-27-2.ap-south-1.compute.internal).

**Instance summary for i-0c9914620391fe54f (Server)** Info

Updated less than a minute ago

**Instance ID**  
i-0c9914620391fe54f

**Public IPv4 address**  
13.235.54.222 | [open address](#)

**Private IPv4 addresses**  
172.31.27.2

**Instance state**  
Running

**Public DNS**  
ec2-13-235-54-222.ap-south-1.compute.amazonaws.com | [open address](#)

**Private IP DNS name (IPv4 only)**  
ip-172-31-27-2.ap-south-1.compute.internal

**IP name:** ip-172-31-27-2.ap-south-1.compute.internal

**Answer private resource DNS name**

**Instance type**

**Elastic IP addresses**

## Security Group configuration

The screenshot shows the AWS Management Console interface for a Security Group. The left sidebar contains navigation links for EC2, Security Groups, sg-080ac39de520a1fbc - SG\_Monitoring, Dashboard, AWS Global View, Events, and a list of instance types and plans. The main content area displays the 'Inbound rules' tab for the Security Group. The table lists 6 inbound rules, all of which are Custom TCP rules with a source of 0.0.0.0/0. The rules are: sgr-0cdc7afc9fcd5114e (Port 9090), sgr-001d00caeabf7078a (Port 22), sgr-010992b48756eb722 (Port 80), sgr-0821e1f2d445833aa (Port 3000), sgr-0059e816bdbfe191d (Port 9115), and sgr-0ff59664d94a580d7 (Port 8080).

**Owner**  
758234806674

**Inbound rules count**  
6 Permission entries

**Outbound rules count**  
1 Permission entry

**Inbound rules (6)**

Security group rule ID	IP version	Type	Protocol	Port range	Source
sgr-0cdc7afc9fcd5114e	IPv4	Custom TCP	TCP	9090	0.0.0.0/0
sgr-001d00caeabf7078a	IPv4	SSH	TCP	22	0.0.0.0/0
sgr-010992b48756eb722	IPv4	HTTP	TCP	80	0.0.0.0/0
sgr-0821e1f2d445833aa	IPv4	Custom TCP	TCP	3000	0.0.0.0/0
sgr-0059e816bdbfe191d	IPv4	Custom TCP	TCP	9115	0.0.0.0/0
sgr-0ff59664d94a580d7	IPv4	Custom TCP	TCP	8080	0.0.0.0/0

## SSH to remote Server



# Jenkins installed

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

Password

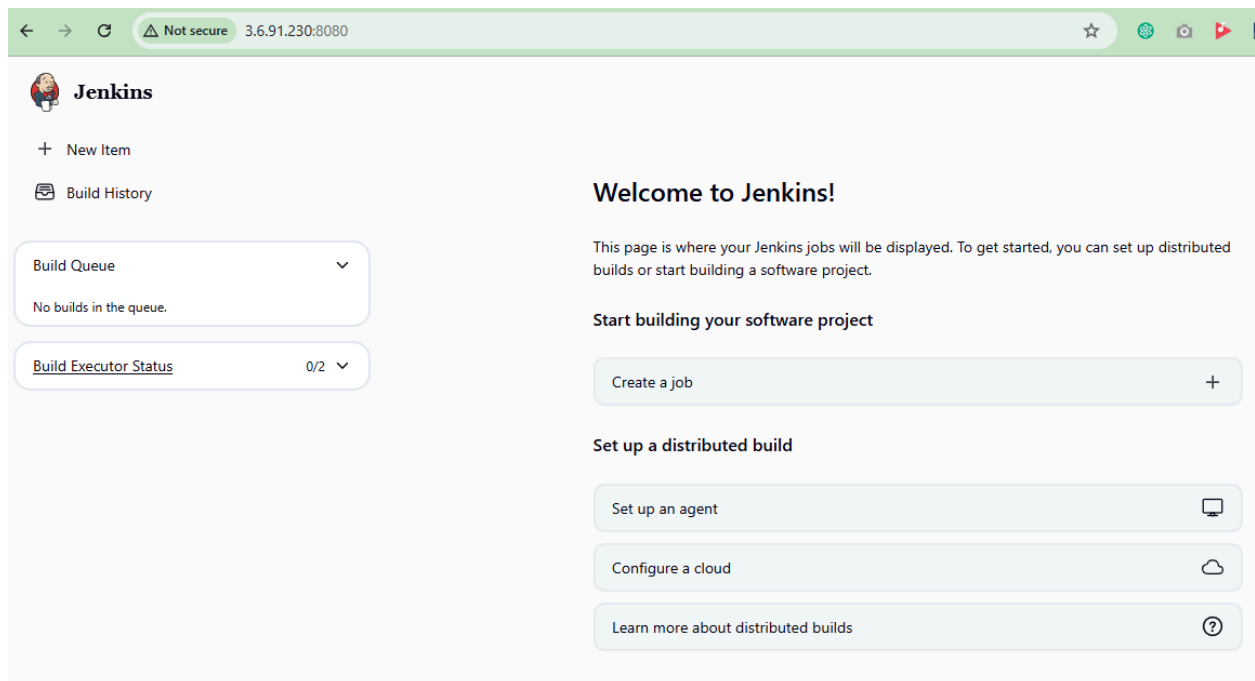
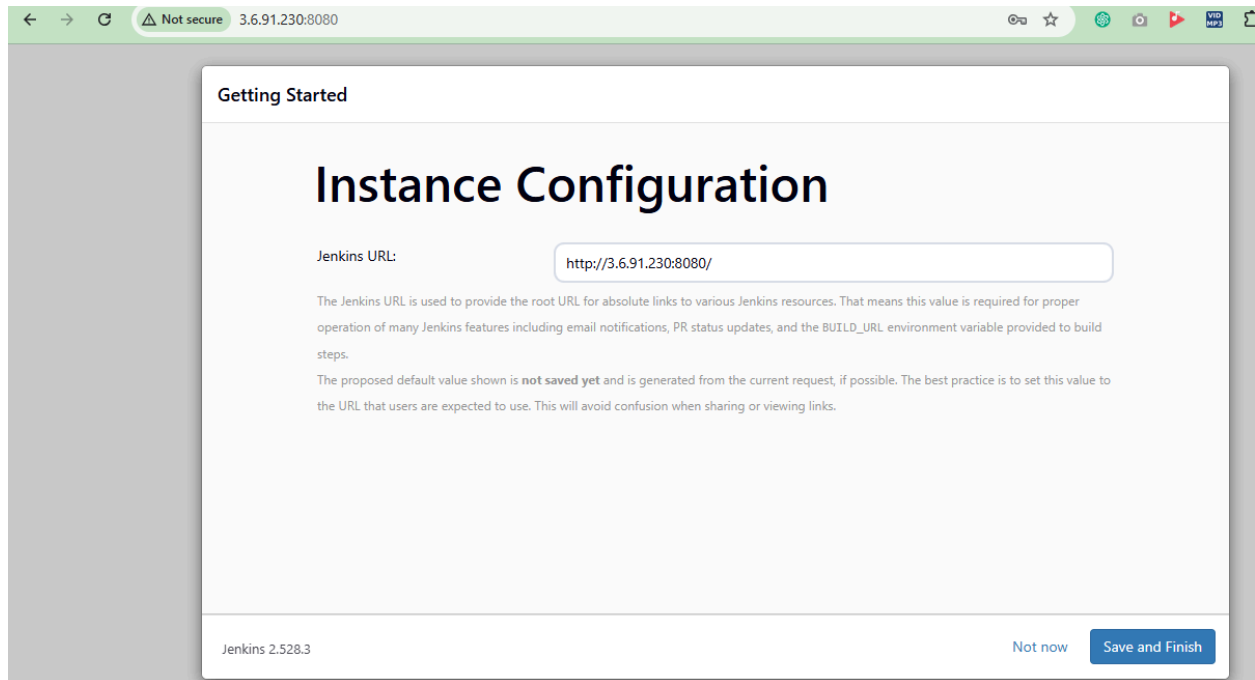
Confirm password

Full name

E-mail address


Jenkins 2.528.3

[Skip and continue as admin](#) [Save and Continue](#)



# Stage : Jenkins

## Multibranch pipeline created

 Jenkins

Project\_Online\_Shopping

Configuration

Search

Help

Feedback

Configuration

General

Branch Sources

Build Configurations

Scan Multibranch Pipeline Triggers

Orphaned Item Strategy

Appearance

Health metrics

Properties

General

Enabled

Display Name ?

Project\_Online\_Shopping

Description

Plan text [Preview](#)

Branch Sources

Git

Project Repository ?

git@github.com:udhayakumararathinam:git-Project\_Online\_Shopping.git

Credentials ?

git (ghrepo)

Add

Behaviours

Discover branches ?

Add

Property strategy

All branches get the same properties

Suppress automatic SCM triggering ?

Branch names to build automatically ?

\*/\*devman/\*

Suppression strategy

For matching branches schedule all builds (nothing is suppressed)

Add property

Add source

Build Configuration

Mode

by Jenkinsfile

Script Path ?

Jenkinsfile

Scan Multibranch Pipeline Triggers

☐ Periodically if not otherwise run ?

Orphaned Item Strategy

Jobs for removed SCM heads (i.e. deleted branches) can be removed immediately or kept based on a desired retention strategy. By default, jobs will be removed as soon as Jenkins determines their associated SCM head no longer exists. As an example, it may be useful to configure a different retention strategy to be able to examine build results of a branch after it has been removed.

☐ Abort builds ?

☒ Discard old items

Days to keep old items

If not empty, old items are only kept up to this number of days

Max # of old items to keep

If not empty, only up to this number of old items are kept

2

Appearance

Icon ?

Metadata Folder icon

Health metrics

Health metrics

Properties

Pipeline Libraries

Shareable libraries available to any Pipeline jobs inside this folder. These libraries will be untrusted, meaning their code runs in the Groovy sandbox.

Add

Save

Apply

REST API Jenkins 2.528.3

Branches (2) [Project\_Online\_Shopping] Online\_Shopping Config - Jenkins Installed plugins - Plugins - main [Project\_Online\_Shopping] Project\_Online\_Shopping

65.2.54.8:8080/job/Project\_Online\_Shopping/

**Jenkins** / Project\_Online\_Shopping

- Status
- Configure
- Scan Multibranch Pipeline Now
- Scan Multibranch Pipeline Log
- Multibranch Pipeline Events
- Delete Multibranch Pipeline
- Build History
- Project Relationship
- Check File Fingerprint
- Rename
- Pipeline Syntax
- Credentials

**Project\_Online\_Shopping**

Branches (2)

S	W	Name	Last Success	Last Failure	Last Duration
✓	☀	dev	23 min #27	2 hr 51 min #18	31 sec
✓	☁	main	23 min #19	1 hr 24 min #15	33 sec

Icon: S M L

Build Queue

## Jenkins Pipeline Stages

65.2.54.8:8080/job/Project\_Online\_Shopping/job/main/multi-pipeline-graph/

**Jenkins** / Project\_Online\_Shopping / main / Stages

### Stages

30 January 2026

#19 14:42 - 33s

Start Checkout... Build Docker.. Push Docker.. Deploy Dock... Post Actions End

3.0s 1.9s 13s 3.1s 0.59s

#18 14:22 - 25s - 1 change

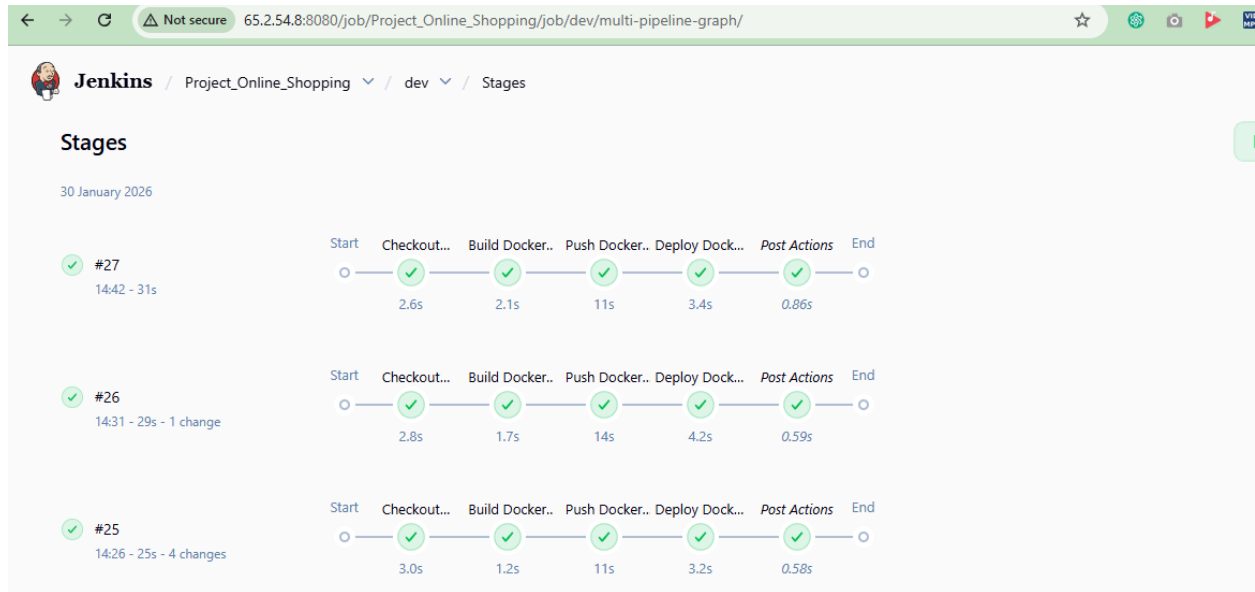
Start Checkout... Build Docker.. Push Docker.. Deploy Dock... Post Actions End

2.9s 1.9s 11s 2.9s 0.59s

#17 14:09 - 28s - 1 change

Start Checkout... Build Docker.. Push Docker.. Deploy Dock... Post Actions End

3.3s 3.1s 11s 3.1s 0.59s



GitHub Webhook configured for build auto trigger

The GitHub Webhooks configuration page for 'Project\_3\_Online\_shopping' shows the 'Settings' tab. The 'Payload URL' is set to 'http://65.254.8:8080/github-webhook/'. The 'Content type' is set to 'application/json'. The 'Secret' field is empty. The 'SSL verification' section is expanded, showing 'Enable SSL verification' selected.

**Webhooks / Manage webhook**

**Settings** | Recent Deliveries

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](#).

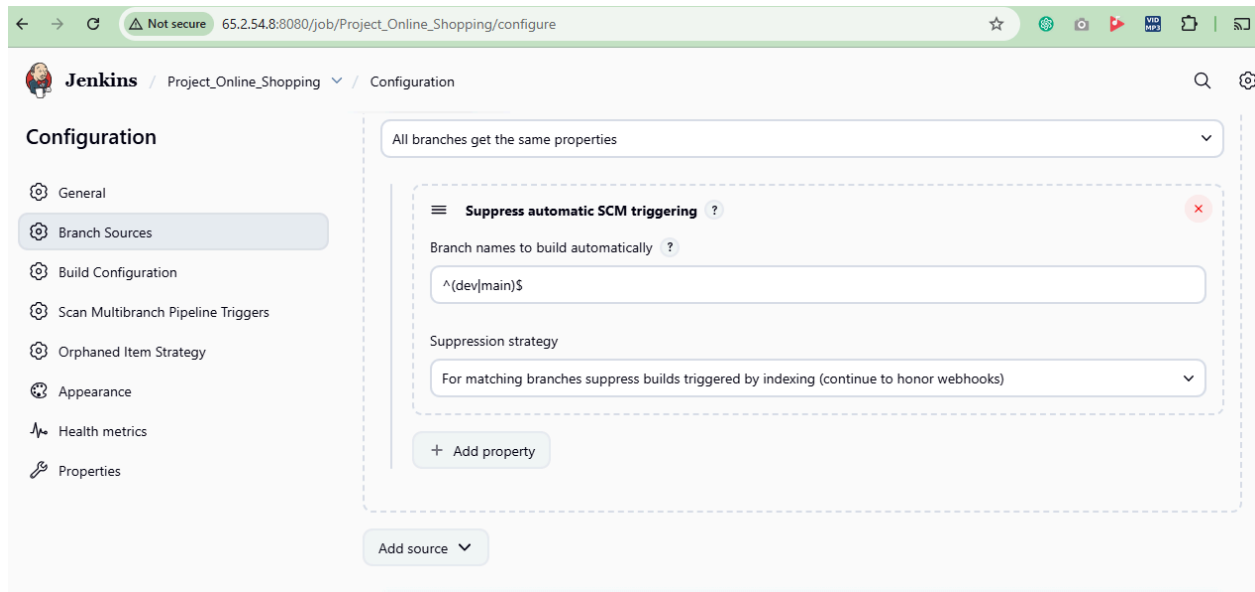
**Payload URL \***  
http://65.254.8:8080/github-webhook/

**Content type \***  
application/json

**Secret**

**SSL verification**  
By default, we verify SSL certificates when delivering payloads.  
☒ Enable SSL verification ☐ Disable (not recommended)

## Jenkins configure for Build auto Trigger



## Committed in Dev branch

```
ubuntu@ip-172-31-28-193:~/projects/project_3$ vi Jenkinsfile
ubuntu@ip-172-31-28-193:~/projects/project_3$ git branch
* dev
  main
ubuntu@ip-172-31-28-193:~/projects/project_3$ git add .
ubuntu@ip-172-31-28-193:~/projects/project_3$ git commit -m "Jenkinsfile updated"
[dev 83c35f8] Jenkinsfile updated
Committer: Ubuntu <ubuntu@ip-172-31-28-193.ap-south-1.compute.internal>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

    git config --global user.name "Your Name"
    git config --global user.email you@example.com

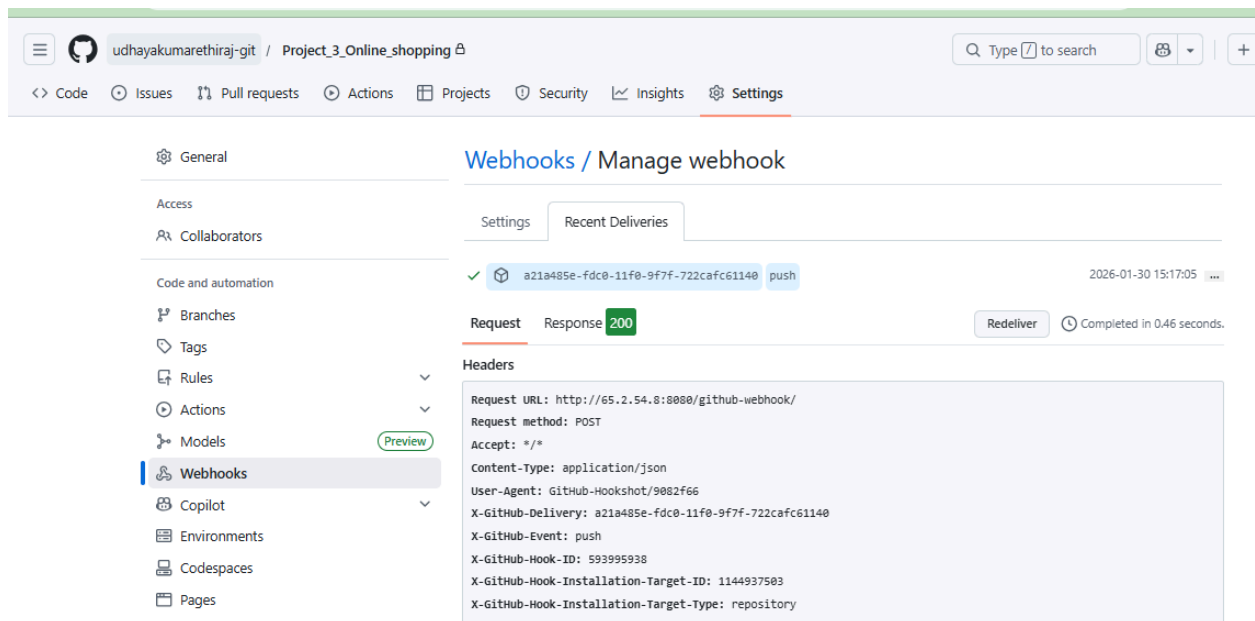
After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 1 insertion(+)
ubuntu@ip-172-31-28-193:~/projects/project_3$ git push origin dev
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 2 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 310 bytes | 310.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To github.com:udhayakumarethiraj-git/Project_3_Online_shopping.git
   27b3cc8..83c35f8  dev -> dev
ubuntu@ip-172-31-28-193:~/projects/project_3$
```

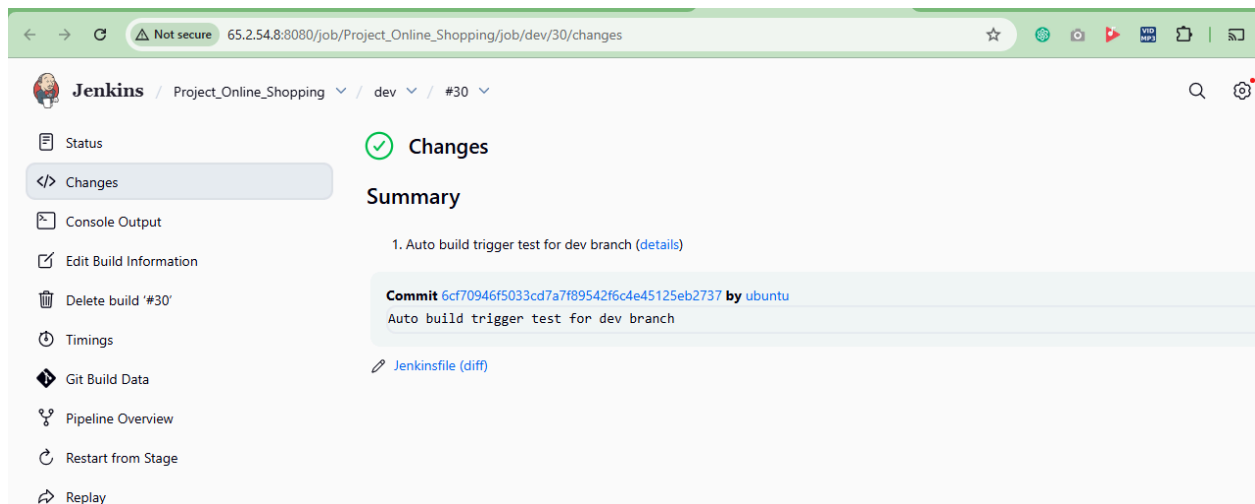


## GitHub Webhook triggered



The screenshot shows the GitHub repository settings for 'Project\_3\_Online\_shopping'. The 'Webhooks' tab is selected in the left sidebar. The 'Manage webhook' section displays a recent delivery from 'a21a485e-fdc0-11f0-9f7f-722cafc61140' with a status of 'push' and a response code of '200'. The 'Request' tab is active, showing the following headers:

```
Request URL: http://65.2.54.8:8080/github-webhook/
Request method: POST
Accept: */*
Content-Type: application/json
User-Agent: GitHub-Hookshot/9082f66
X-GitHub-Delivery: a21a485e-fdc0-11f0-9f7f-722cafc61140
X-GitHub-Event: push
X-GitHub-Hook-ID: 593995938
X-GitHub-Hook-Installation-Target-ID: 1144937503
X-GitHub-Hook-Installation-Target-Type: repository
```



The screenshot shows the Jenkins job 'Project\_Online\_Shopping' in the 'dev' branch, build #30. The 'Changes' tab is selected, showing a summary of the build. The build was triggered by a commit '6cf70946f5033cd7a7f89542f6c4e45125eb2737' by 'ubuntu' with the message 'Auto build trigger test for dev branch'. The build status is 'Success' (indicated by a green checkmark).

Summary

1. Auto build trigger test for dev branch ([details](#))

**Commit** [6cf70946f5033cd7a7f89542f6c4e45125eb2737](#) by [ubuntu](#)  
Auto build trigger test for dev branch

[Jenkinsfile \(diff\)](#)

## dev branch merged to main branch

```

ubuntu@ip-172-31-28-193:~/projects/project_3$ git checkout main
Switched to branch 'main'
ubuntu@ip-172-31-28-193:~/projects/project_3$ git merge dev
Updating 5653e39..6cf7094
Fast-forward
 Jenkinsfile | 1 -
 1 file changed, 1 deletion(-)
ubuntu@ip-172-31-28-193:~/projects/project_3$ git push origin main
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:udhayakumarethiraj-git/Project_3_Online_shopping.git
 5653e39..6cf7094  main -> main
ubuntu@ip-172-31-28-193:~/projects/project_3$

```

The screenshot shows the Jenkins web interface for a job named 'Project\_Online\_Shopping'. The 'Changes' tab is selected, showing a summary of the build. The summary indicates that the build was triggered by an auto build trigger test for the dev branch. The commit hash is 6cf70946f5033cd7a7f89542f6c4e45125eb2737, committed by ubuntu. A link to the Jenkinsfile diff is provided.

**Jenkins** / Project\_Online\_Shopping / main / #21

**Changes**

**Summary**

1. Auto build trigger test for dev branch ([details](#))

**Commit** 6cf70946f5033cd7a7f89542f6c4e45125eb2737 **by** ubuntu  
Auto build trigger test for dev branch

[Jenkinsfile \(diff\)](#)

DockerHub prod repo (Private) image pushed using Jenkins pipeline

The screenshot shows the Docker Hub repository page for 'udhayakumarethiraj/prod'. The repository is private and contains 1 tag (latest). The page includes a sidebar with navigation options, a main content area with repository details, and a right sidebar with Docker commands and a Build Cloud advertisement.

**Repositories** / prod / General

**udhayakumarethiraj/prod**

Last pushed 2 minutes ago · Repository size: 25.5 MB · 0 · 0

[Add a description](#)

[Add a category](#)

**Tags**

This repository contains 1 tag(s).

Tag	OS	Type	Pulled	Pushed
latest		Image	less than 1 day	3 minutes

**Docker commands**

To push a new tag to this repository:

```
docker push udhayakumarethiraj/prod:tagname
```

**buildcloud**

Build with **Docker Build Cloud**

Accelerate image build times with access to cloud-based builders and shared cache.

Docker Build Cloud executes builds optimally.

## DockerHub dev repo (Public) image pushed using Jenkins pipeline

The screenshot shows the Docker Hub interface for a public repository named 'udhayakumarethiraj/dev'. The repository was last pushed about 2 hours ago and has a size of 25.7 MB. It has 0 stars and 74 downloads. The 'General' tab is selected, showing a table of tags. The 'latest' tag is listed with an OS of 'Image', pushed less than 1 day ago, and pulled about 2 hours ago. A sidebar on the left contains navigation links for Repositories, Hardened Images, Collaborations, Settings, Default privacy, Notifications, Billing, Usage, Pulls, and Storage. A 'Docker commands' section on the right provides a command to push a new tag: 'docker push udayakumarethiraj/dev:tagname'.

Tag	OS	Type	Pulled	Pushed
latest	Image	Image	less than 1 day	about 2 hours

Jenkins dev branch Build Console Log : [URL](#)

Jenkins main branch Build Console Log : [URL](#)

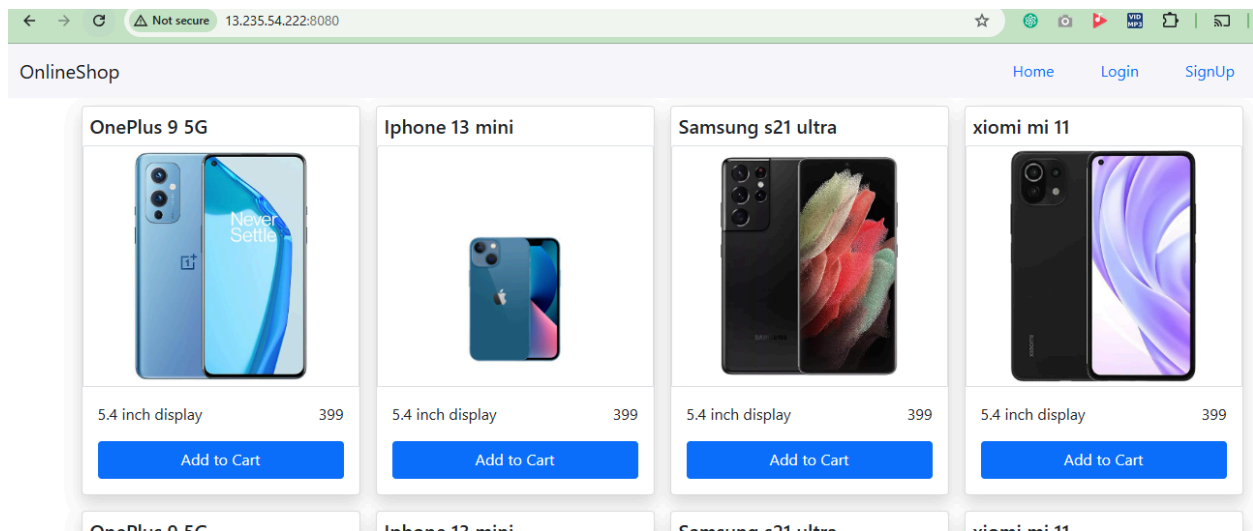
## Application Output

Prod

The screenshot displays the 'OnlineShop' application interface. The header includes navigation links for Home, Login, and SignUp. The main content area features four product cards, each showing a smartphone image, its name, specifications, price, and an 'Add to Cart' button. The products are: OnePlus 9 5G (5.4 inch display, price 399), Iphone 13 mini (5.4 inch display, price 399), Samsung s21 ultra (5.4 inch display, price 399), and xiami mi 11 (5.4 inch display, price 399). The browser address bar shows the URL '13.235.54.222'.

Product Name	Display	Price
OnePlus 9 5G	5.4 inch display	399
Iphone 13 mini	5.4 inch display	399
Samsung s21 ultra	5.4 inch display	399
xiami mi 11	5.4 inch display	399

Dev



## Stages : Monitoring

An open-source monitoring system was implemented using Prometheus to collect application metrics, and Grafana was configured with dashboards to monitor application health, application URL status, and real-time metrics data.

Container data metrics collection to Prometheus

```
ubuntu@ip-172-31-27-2:~/project_3$ cat cadvisor
version: "3.8"

services:
  cadvisor:
    image: gcr.io/cadvisor/cadvisor:latest
    container_name: cadvisor
    privileged: true
    ports:
      - "8081:8080"
    volumes:
      - /:/rootfs:ro
      - /var/run:/var/run:ro
      - /sys:/sys:ro
      - /var/lib/docker:/var/lib/docker:ro
    restart: unless-stopped

ubuntu@ip-172-31-27-2:~/project_3$
```

## Application metrics config

```
ubuntu@ip-172-31-27-2:~/project_3$ sudo cat /etc/prometheus/prometheus.yml
global:
  scrape_interval: 15s

scrape_configs:
  # Prometheus self
  - job_name: "prometheus"
    static_configs:
      - targets: ["localhost:9090"]

  # cAdvisor metrics (filter only devops-build-main)
  - job_name: "devops-build-main"
    static_configs:
      - targets:
        - "localhost:8081"

ubuntu@ip-172-31-27-2:~/project_3$
```

## Targets added to Prometheus

← → ↻ Not secure 13.235.54.222:9090/targets?search= ☆ 🌐 📷 📄 📁 📂 📅 📆 📇 📈 📉 📊 📋 📌 📍 📎 📏 📐 📑 📒 📓 📔 📕 📖 📗 📘 📙 📚 📛 📜 📝 📞 📟 📠 📡 📢 📣 📤 📥 📦 📧 📨 📩 📪 📫 📬 📭 📮 📯 📰 📱 📲 📳 📴 📵 📶 📷 📸 📹 📺 📻 📼 📽 📾 📿 📠 📡 📢 📣 📤 📥 📦 📧 📨 📩 📪 📫 📬 📭 📮 📯 📰 📱 📲 📳 📴 📵 📶

Prometheus Alerts Graph Status ▾ Help ⚙

### Targets

All scrape pools ▾ All Unhealthy Collapse All 🔍 Filter by endpoint or labels ✓ Unknown ✓ Unhealthy ✓

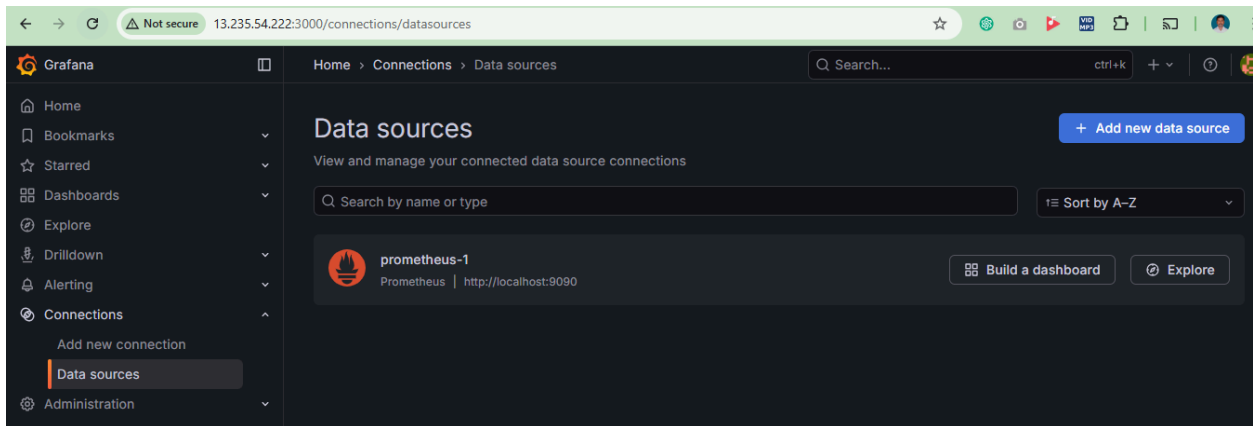
**devops-build-main (1/1 up)** [show less](#)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
<a href="http://localhost:8081/metrics">http://localhost:8081/metrics</a>	UP	<code>instance="localhost:8081"</code> <code>job="devops-build-main"</code> ▾	5.739s ago	65.052ms	

**prometheus (1/1 up)** [show less](#)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
<a href="http://localhost:9090/metrics">http://localhost:9090/metrics</a>	UP	<code>instance="localhost:9090"</code> <code>job="prometheus"</code> ▾	3.525s ago	4.474ms	

## Added Data source to Grafana



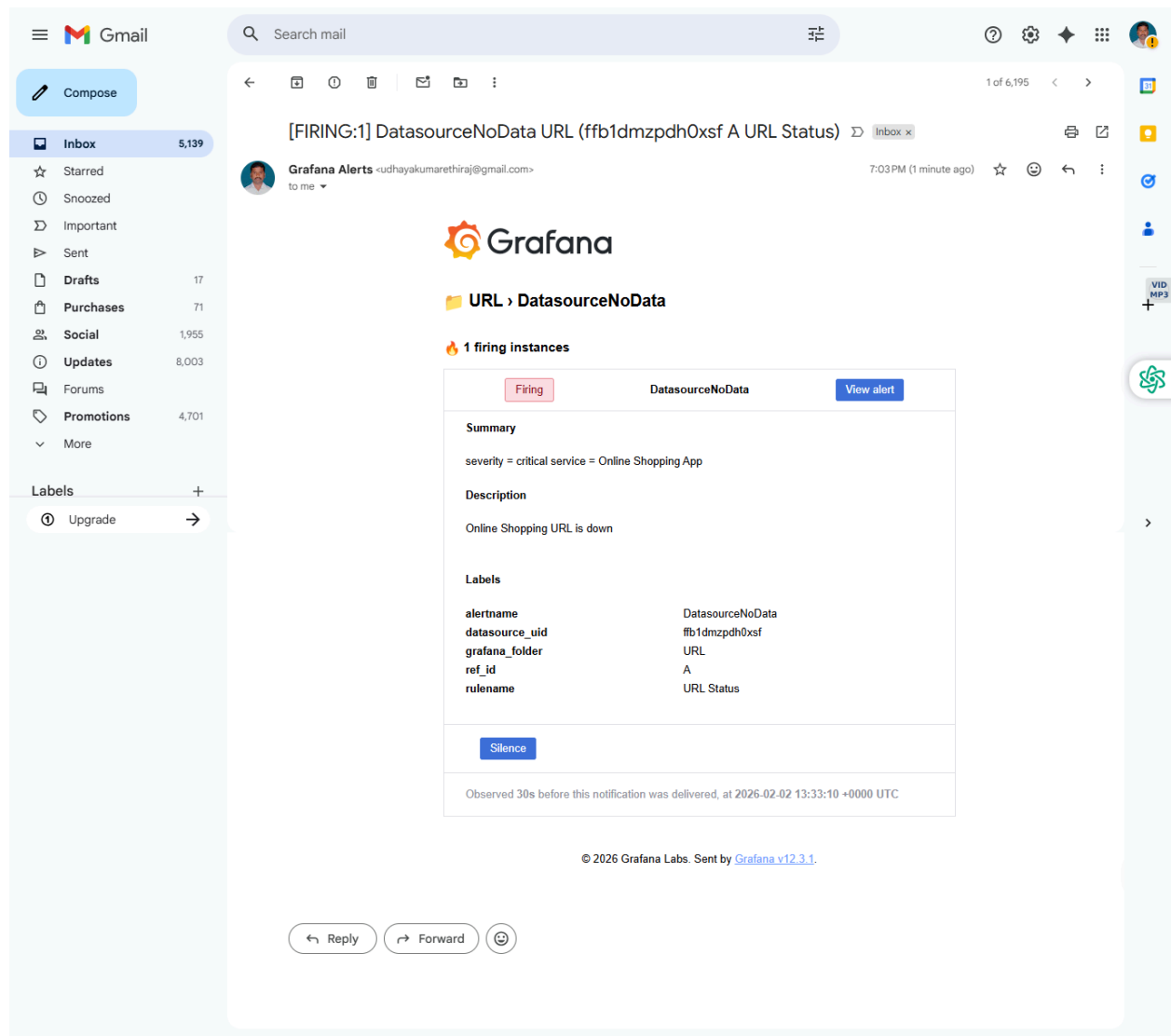
## Grafana Dashboards :

### Application URL



Email Alerting:

Configured Email alerting in Grafana and tested for URL Down



Test Email

