

# 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$
```

The screenshot shows a GitHub repository page for 'Project\_3\_Online\_shopping'. The 'Code' tab is selected. A modal window titled 'Switch branches/tags' is open, showing the 'Branches' tab with 'main' (default) and 'dev' selected. The main area displays the commit history for the 'dev' branch, which has three commits from 0f90991, each made 8 minutes ago. The commits are labeled 'initial commit'.

## 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$ docker images
 Info -  In Use
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
e861c0fe234a  devops:latest  "/docker-entrypoint..."  26 hours ago  Up 22 minutes  0.0.0.0:80->80/tcp, [::]:80->80/tcp  NAMES
           devops-bu
ubuntu@ip-172-31-28-193:~/projects/project_3$ 

```

## Output Test result

OnlineShop

<b>OnePlus 9 5G</b>	<b>Iphone 13 mini</b>	<b>Samsung s21 ultra</b>	<b>xiomi mi 11</b>
			
5.4 inch display	5.4 inch display	5.4 inch display	5.4 inch display
399	399	399	399
<a href="#">Add to Cart</a>	<a href="#">Add to Cart</a>	<a href="#">Add to Cart</a>	<a href="#">Add to Cart</a>
<b>OnePlus 9 5G</b>	<b>Iphone 13 mini</b>	<b>Samsung s21 ultra</b>	<b>xiomi mi 11</b>
			

OnlineShop

	<p>Login Into your account</p> <input type="text"/> <p>Email address</p> <input type="password"/> <p>Password</p>
---	--

The image contains two screenshots of a web application. The top screenshot shows a sign-up page with a placeholder image of a woman with curly hair and glasses, and fields for User Name and Email address. The bottom screenshot shows a shopping cart page indicating the cart is currently empty.

## 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 .dockerrcignore
-> => transferring context: 2B
-> [1/3] FROM docker.io/library/nginx:alpine@sha256:7d7a15b8a280c661051955f14c2b91fed3e23724ddba18d2f53e8b44e74ab37a
-> => resolve docker.io/library/nginx:alpine@sha256:7d7a15b8a280c661051955f14c2b91fed3e23724ddba18d2f53e8b44e74ab37a
-> [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 sha256:054750a27954a40fd97591cd6b02dF3510c78e4b31fe36648a2727357626e701
-> => exporting config sha256:bc2850f174cf69bbf46514d907585db5b22bb2685731f1541ca5d237cc26bb9e
-> => exporting attestation manifest sha256:8cf70c40c9dc53be3913b9d34ed50c67f0e142b7a58fa5d2024c5b574ae2b7
-> => exporting manifest list sha256:61b41fcfa6dcae5bacf84347ea88af7ddeec6c3e33553fe09d5b68ed936c26
-> => naming to docker.io/library/devops-build:latest
-> => unpacking to docker.io/library/devops-build:latest
ubuntu@ip-172-31-28-193:~/projects/project_3$ 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 | 61b41fcfa6dca | 95.6MB     | 26.7MB       | 0.0s  |


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
docker-compose.yml
⠄ 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 enp39s0: 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!

```

OnlineShop

Home Login SignUp

OnePlus 9 5G 	Iphone 13 mini 	Samsung s21 ultra 	xiom mi 11 
5.4 inch display 399 <a href="#">Add to Cart</a>			
OnePlus 9 5G 	Iphone 13 mini 	Samsung s21 ultra 	xiom mi 11 

## Stage : Docker HUB

Docker hub repo created

hub.docker.com/repositories/udhayakumarethiraj

\* New Docker Hardened Images are now FREE for every developer. Watch on-demand webinar. →

hub Explore My Hub

Repositories

All repositories within the **udhayakumarethiraj** namespace.

Name	Last Pushed	Contains	Visibility
udhayakumarethiraj/dev	1 minute ago		Public
udhayakumarethiraj/prod	2 minutes ago		Private

# EC2 Instance

EC2 Instance launched to deploy the application

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like Dashboard, AWS Global View, Events, Instances (with sub-links for Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, and Reserved Instances), and Images. The main content area is titled "Instance summary for i-0c9914620391fe54f (Server)". It displays the following information:

- Instance ID:** i-0c9914620391fe54f
- IPv6 address:** -
- Hostname type:** IP name: ip-172-31-27-2.ap-south-1.compute.internal
- Answer private resource DNS name:** -
- Public IPv4 address:** 13.235.54.222 | [open address](#)
- Instance state:** Running
- Private IP DNS name (IPv4 only):** ip-172-31-27-2.ap-south-1.compute.internal
- Instance type:** -
- Private IPv4 addresses:** 172.31.27.2
- Public DNS:** ec2-13-235-54-222.ap-south-1.compute.amazonaws.com | [open address](#)
- Elastic IP addresses:** -

## Security Group configuration

The screenshot shows the AWS Security Groups page. The sidebar includes links for Dashboard, AWS Global View, Events, Instances, and Images. The main content shows a security group named "sg-080ac39de520a1fbe - SG\_Monitoring". It displays the following information:

Owner	Inbound rules count	Outbound rules count
758234806674	6 Permission entries	1 Permission entry

The "Inbound rules" tab is selected, showing six entries:

Security group rule ID	IP version	Type	Protocol	Port range	Source
sgr-0cdc7afc9fc5114e	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-0821ef2d445833aa	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

```

ubuntu@ip-172-31-28-193:~$ ssh ubuntu@172.31.27.2
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 Fri Jan 30 22:48:51 UTC 2026

System load:  0.0          Processes:           170
Usage of /:   40.1% of 11.45GB  Users logged in:      0
Memory usage: 12%          IPv4 address for enp39s0: 172.31.27.2
Swap usage:   0%         

Expanded Security Maintenance for Applications is not enabled.

5 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

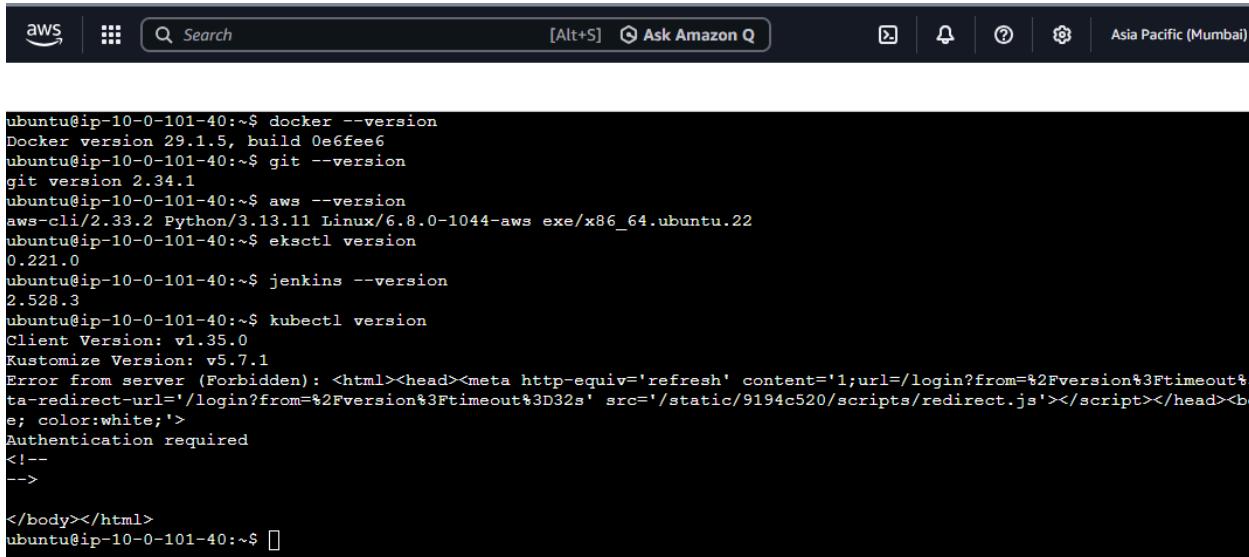
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

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

Last login: Fri Jan 30 14:13:39 2026 from 49.37.200.197
ubuntu@ip-172-31-27-2:~$ █

```

## Installed packages



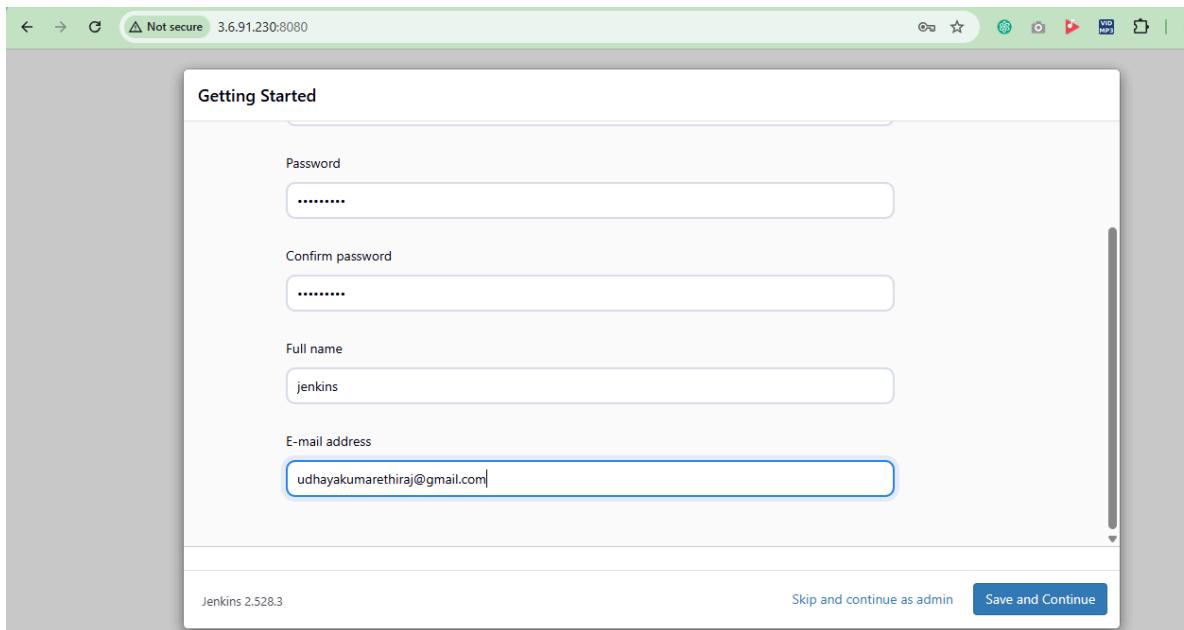
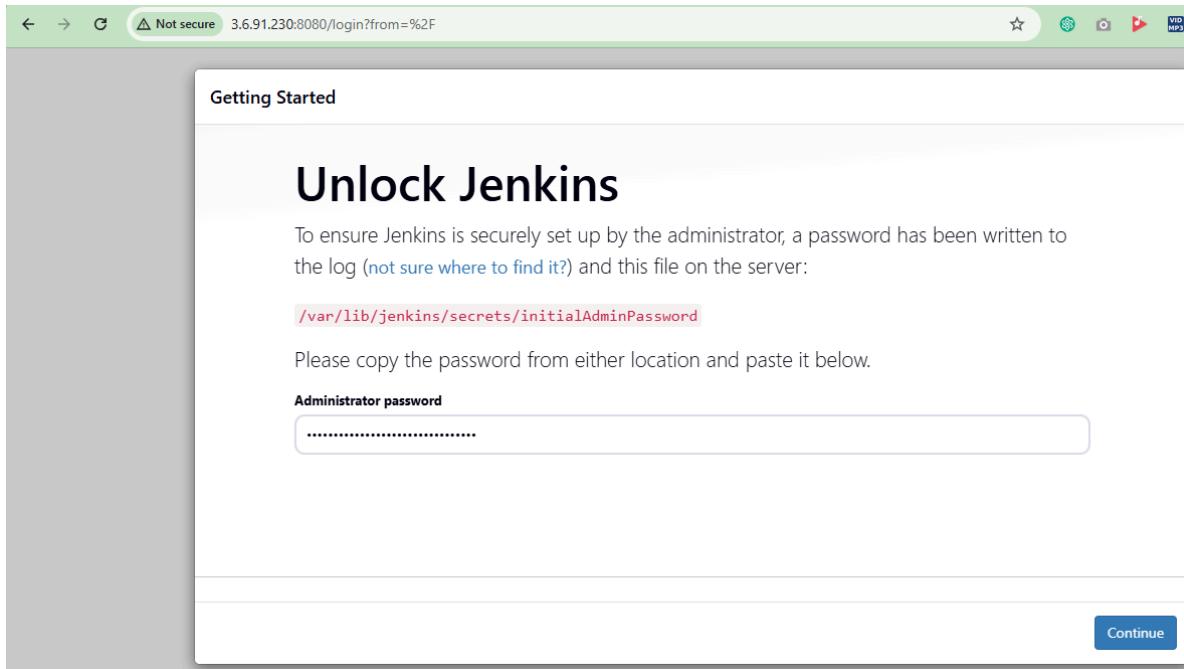
The screenshot shows the AWS Lambda console interface. At the top, there's a navigation bar with the AWS logo, a search bar containing 'Search', and links for '[Alt+S]', 'Ask Amazon Q', and account information ('Asia Pacific (Mumbai)'). Below the navigation bar, the main content area displays a terminal window showing the output of several command-line tools:

```

ubuntu@ip-10-0-101-40:~$ docker --version
Docker version 29.1.5, build 0e6feef6
ubuntu@ip-10-0-101-40:~$ git --version
git version 2.34.1
ubuntu@ip-10-0-101-40:~$ aws --version
aws-cli/2.33.2 Python/3.13.11 Linux/6.8.0-1044-aws exe/x86_64.ubuntu.22
ubuntu@ip-10-0-101-40:~$ eksctl version
0.221.0
ubuntu@ip-10-0-101-40:~$ jenkins --version
2.528.3
ubuntu@ip-10-0-101-40:~$ kubectl version
Client Version: v1.35.0
Kustomize Version: v5.7.1
Error from server (Forbidden): <html><head><meta http-equiv='refresh' content='1;url=/login?from=%2Fversion%3Ftimeout%3Fta-redirect-url=%2Flogin?from=%2Fversion%3Ftimeout%3D32s' src='/static/9194c520/scripts/redirect.js'></script></head><body>Authentication required<!--<br>--></body></html>
ubuntu@ip-10-0-101-40:~$ █

```

## Jenkins installed



Not secure 3.6.91.230:8080

## Getting Started

# Instance Configuration

Jenkins URL:

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 2.528.3 Not now Save and Finish

Not secure 3.6.91.230:8080

# Jenkins

+ New Item

Build History

Build Queue ▾  
No builds in the queue.

Build Executor Status 0/2 ▾

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job +

Set up a distributed build

Set up an agent 

Configure a cloud 

Learn more about distributed builds 

## Stage : Jenkins

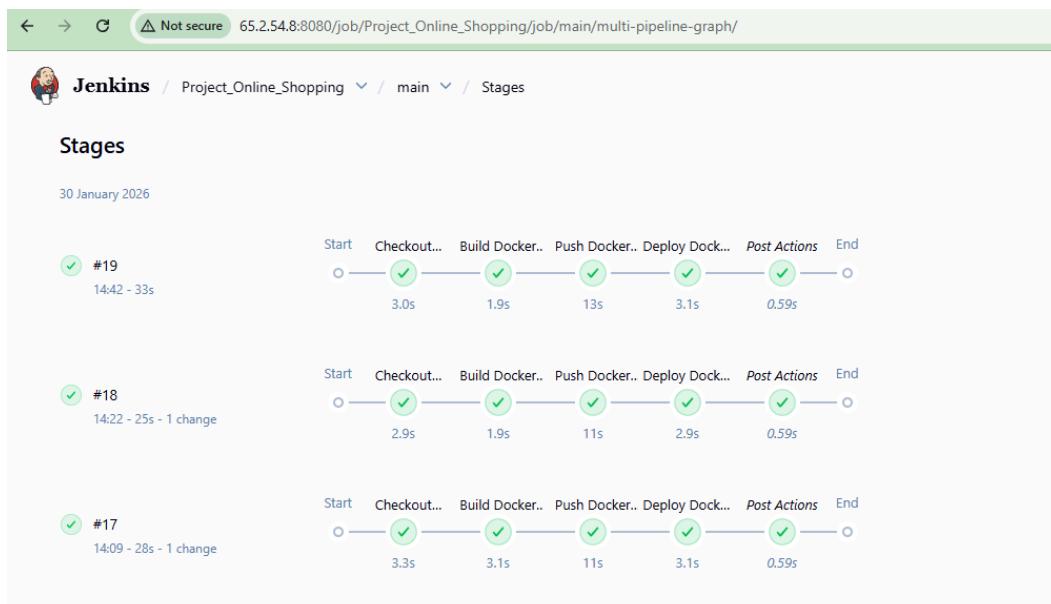
### Multibranch pipeline created

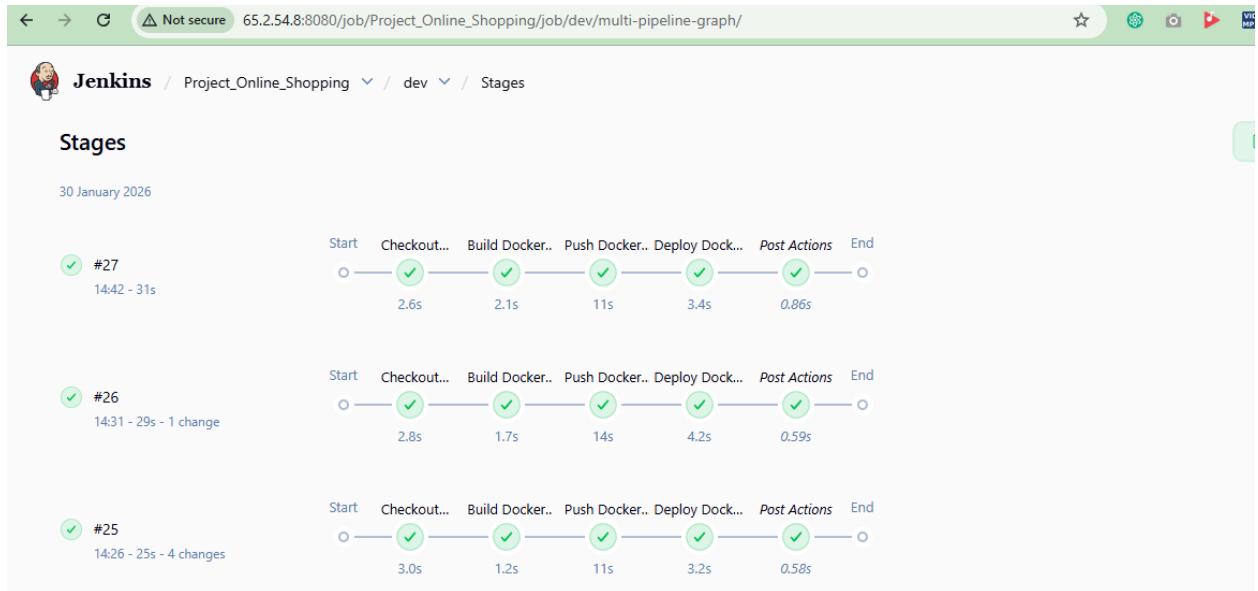
The screenshot shows the Jenkins configuration interface for a Multibranch Pipeline named 'Project\_Online\_Shopping'. The 'General' section is enabled. Under 'Branch Sources', there is a single 'Git' repository defined with the URL 'git@github.com:utheyakumarngit/Project\_Online\_Shopping.git' and credentials 'git (ghrepo)'. The 'Behaviours' section contains a 'Discover branches' step. In the 'Property strategy' section, 'All branches get the same properties' is selected, with a 'Suppress automatic SCM triggering' rule for branches matching '(domain)\$'. The 'Build Configuration' section uses a 'Jenkinsfile' script path. 'Scan Multibranch Pipeline Triggers' is set to run 'Periodically if not otherwise run'. The 'Orphaned Item Strategy' section includes rules for 'Abort builds', 'Discard old items' (with a limit of 2 days), and 'Keep items up to this number of days' (also 2 days). The 'Appearance' section allows setting icons and metadata folder icons. The 'Properties' section is currently empty. At the bottom, there are 'Save' and 'Apply' buttons.

The screenshot shows the Jenkins Project\_Online\_Shopping dashboard. On the left, there's a sidebar with options like 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, and Credentials. The main area displays two branches: 'dev' and 'main'. The 'dev' branch has a green checkmark icon, a sun icon, and last success at 23 min #27. The 'main' branch also has a green checkmark icon, a sun icon, and last success at 23 min #19. Below the branches is a table with columns: S, W, Name, Last Success, Last Failure, and Last Duration.

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

## Jenkins Pipeline Stages





## GitHub Webhook configured for build auto trigger

The screenshot shows the GitHub repository settings for Project\_3\_Online\_shopping. The left sidebar shows navigation options like General, Access, Collaborators, Code and automation, Branches, Tags, Rules, Actions, Models, Webhooks (which is selected), Copilot, Environments, Codespaces, and Pages. The main area is titled "Webhooks / Manage webhook". It includes sections for "Settings" and "Recent Deliveries". A note explains that a POST request will be sent to the Payload URL with event details. The "Payload URL" field contains "http://65.254.8:8080/github-webhook/". The "Content type" is set to "application/json". There is a "Secret" field and an "SSL verification" section where "Enable SSL verification" is selected. A note states that SSL certificates are verified by default.

## Jenkins configure for Build auto Trigger

The screenshot shows the Jenkins configuration page for a project named "Project\_Online\_Shopping". The left sidebar lists various configuration options: General, Branch Sources (selected), Build Configuration, Scan Multibranch Pipeline Triggers, Orphaned Item Strategy, Appearance, Health metrics, and Properties. The main panel displays a "Branch Sources" section titled "All branches get the same properties". It contains a "Suppress automatic SCM triggering" section with a "Branch names to build automatically" input field containing "^dev\$main\$". Below it is a "Suppression strategy" dropdown set to "For matching branches suppress builds triggered by indexing (continue to honor webhooks)". A "+ Add property" button is also present.

## 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 settings interface for managing webhooks. On the left, a sidebar lists various GitHub features like General, Access, Collaborators, Code and automation, Branches, Tags, Rules, Actions, Models, and Webhooks (which is currently selected). The main area is titled "Webhooks / Manage webhook" and shows a "Recent Deliveries" tab. A single delivery is listed for a push event on 2026-01-30 at 15:17:05. The "Request" tab is active, displaying the following details:

```
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 a Jenkins job page for "Project\_Online\_Shopping". The left sidebar includes links for Status, Changes (which is currently selected), Console Output, Edit Build Information, Delete build '#30', Timings, Git Build Data, Pipeline Overview, Restart from Stage, and Replay. The main content area is titled "Changes" and shows a summary of the build trigger:

Summary

- 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$ 

```

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

**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

Repositories / prod / General

Using 1 of 1 private repositories.

**Docker commands**

To push a new tag to this repository:

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

**Tags**

This repository contains 1 tag(s).

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

**buildcloud**

Build with Docker Build Cloud

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

Docker Build Cloud executes builds on optimally.

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

Repositories / dev / General

udhayakumarethiraj/dev

Last pushed about 2 hours ago • Repository size: 25.7 MB • 0 stars • 74 forks

Add a description Add a category

General Tags Image Management Collaborators Webhooks Settings

Tags

This repository contains 1 tag(s).

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

Using 1 of 1 private repository

Docker commands

To push a new tag to this repository:

```
docker push udhayakumarethiraj/dev:tagname
```

Public view

buildcloud

Build with Docker Build Cloud

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

Docker Build Cloud executes builds on optimally-dimensioned cloud infrastructure with dedicated per-

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

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

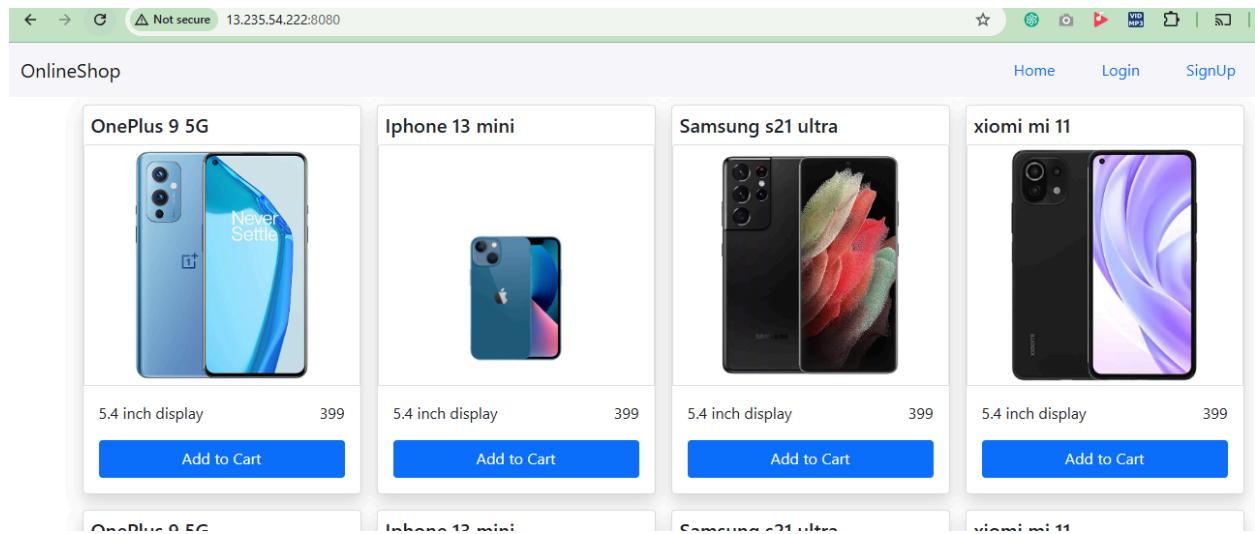
## Application Output

Prod

OnePlus 9 5G	Iphone 13 mini	Samsung s21 ultra	xiomi mi 11
5.4 inch display Add to Cart			

OnePlus 9 5G      Iphone 13 mini      Samsung s21 ultra      xiomi mi 11

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

The screenshot shows the Prometheus Targets page. At the top, there's a navigation bar with links for Alerts, Graph, Status, and Help. Below the navigation, there's a search bar and a filter section with checkboxes for Unknown, Unhealthy, and a third state. The main content area is divided into two sections: 'Targets' and 'prometheus'.

**Targets** section:

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

**prometheus** section:

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

## Added Data source to Grafana

The screenshot shows the Grafana interface at the URL [13.235.54.222:3000/connections/datasources](http://13.235.54.222:3000/connections/datasources). The left sidebar is dark-themed and includes links for Home, Bookmarks, Starred, Dashboards, Explore, Drilldown, Alerting, Connections, Data sources (which is selected and highlighted in orange), and Administration. The main content area is titled "Data sources" and displays a single entry: "prometheus-1" (Prometheus | http://localhost:9090). There are buttons for "Build a dashboard" and "Explore". A search bar at the top right says "Search..." and there is a "Sort by A-Z" dropdown.

## Grafana Dashboards :

### Application URL

