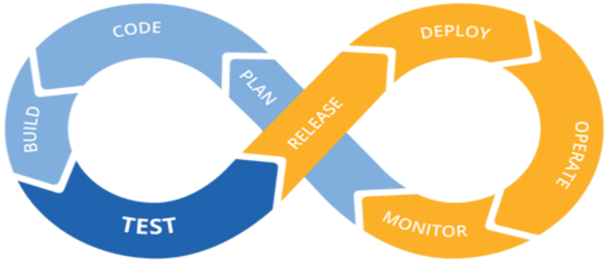
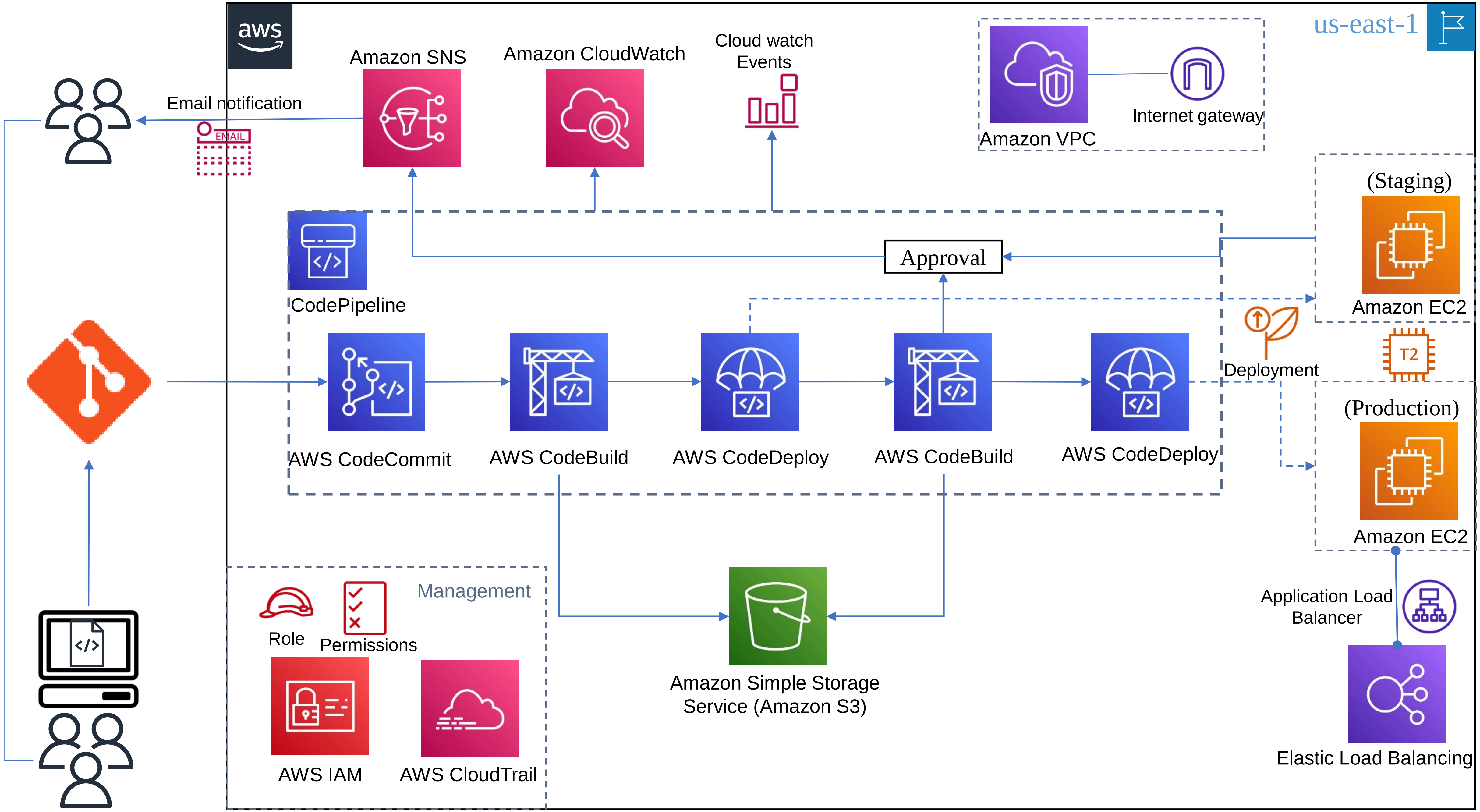
Automated Cloud Development Pipeline using AWS DevOps Cluster



Developed and Designed by
Vinod Kumar Mylapilli


aws
us-east-1
Amazon SNS
Amazon CloudWatch
Cloud watch Events
Amazon VPC
Internet gateway
Email notification
EMAIL
CodePipeline
Approval
(Staging)
Amazon EC2
Deployment
T2
(Production)
Amazon EC2
AWS CodeCommit
AWS CodeBuild
AWS CodeDeploy
AWS CodeBuild
AWS CodeDeploy
Management
Role
Permissions
AWS IAM
AWS CloudTrail
Amazon Simple Storage Service (Amazon S3)
Application Load Balancer
Elastic Load Balancing

AWS Services Used: AWS IAM, Amazon EC2, AWS CodeCommit, AWS CodeBuild, AWS CodeDeploy, AWS CodePipeline, AWS S3, Elastic Load Balancing (Application Load balancer), Amazon SNS, Amazon CloudWatch, Amazon Cloud Trail, Amazon VPC.

Application Tools Used: Python, Django, SQlite3, HTML, CSS, JS, Git hub, Git Bash, Nginx.
Libraries Used: AWS CLI, Whitenoise, gunicorn, Pillow, sqlparse, asgiref, tzdata.

Step-1:

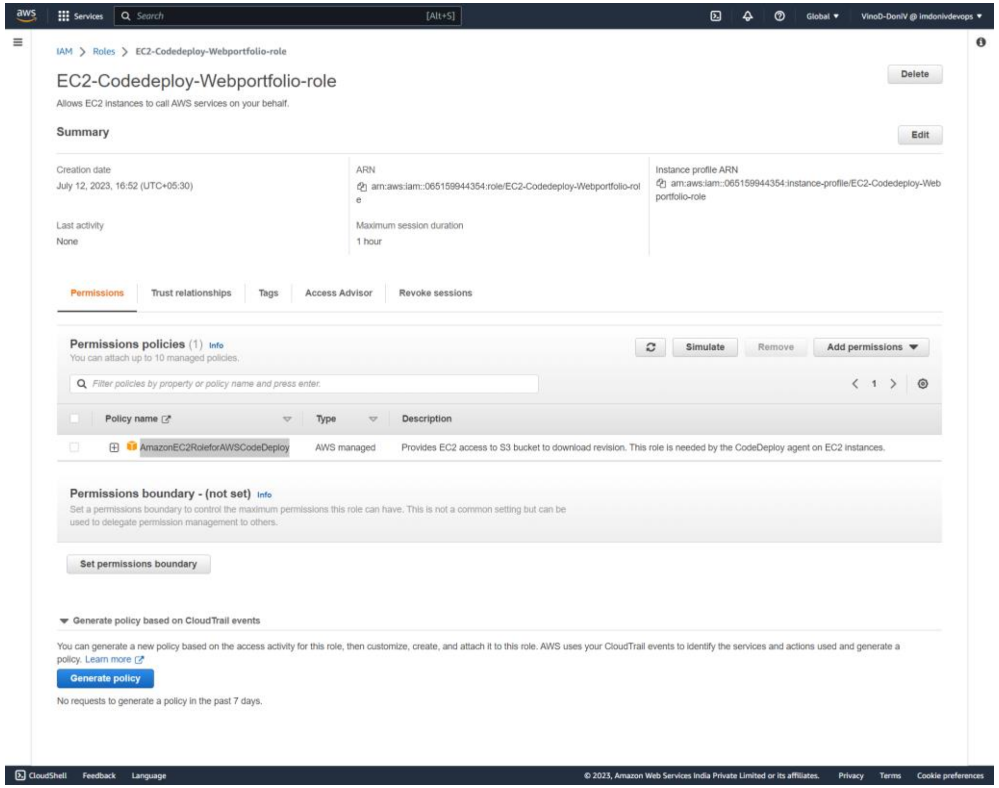
Create an Amazon EC2 and AWS CodeDeploy IAM Roles

First Create an Amazon EC2 IAM Role

1. Open the IAM Console at https://console.aws.amazon.com/iam/.
2. In the navigation pane of the IAM console, Choose **Roles**, and then choose **Create role**.
3. Under Trusted entity type, select **AWS service**.
4. From the Use cases, Choose **EC2** and then click Next.
5. On the Add Permissions tab, do the following:
    i. In the Filter policies box, enter “**AmazonEC2RoleforAWSCodeDeploy**”
    ii. Choose Next.
6. On the Name, review and create tab, do the following:
    i. For Role name, enter a unique name for role, such as **EC2-Codedeploy-Webportfolio-role.**
    ii. For Description, enter descriptive text “Provide Ec2 access to S3 bucket to download revision. This role is needed by the CodeDeploy agent on EC2 instance”
    iii. Click Create role.

Second Create an Amazon CodeDeploy IAM Role

1. Open the IAM Console at https://console.aws.amazon.com/iam/.
2. In the navigation pane of the IAM console, Choose Roles, and then choose Create role.
3. Under Trusted entity type, select AWS service.
4. From the Use cases, other AWS services dropdown list, Choose CodeDeploy and then click Next.
5. On the Add Permissions tab, Leave default and then click Next.
6. On the Name, review and create tab, do the following:
   i. For Role name, enter a unique name for role, such as Codedeploy-allow-allServices-role.
   ii. For Description, enter descriptive text "Provide CodeDeploy service access to expand tags and interact with Auto Scaling on your behalf"
   iii. Click Create role.



EC2-Codedeploy-Webportfolio-role

Allows EC2 instances to call AWS services on your behalf.

Summary

Creation date
July 12, 2023, 16:52 (UTC+05:30)

Last activity
None

ARN
arn:aws:iam::065159944354:role/EC2-Codedeploy-Webportfolio-role

Maximum session duration
1 hour

Instance profile ARN
arn:aws:iam::065159944354:instance-profile/EC2-Codedeploy-Webportfolio-role

Permissions policies (1) Info
You can attach up to 10 managed policies.

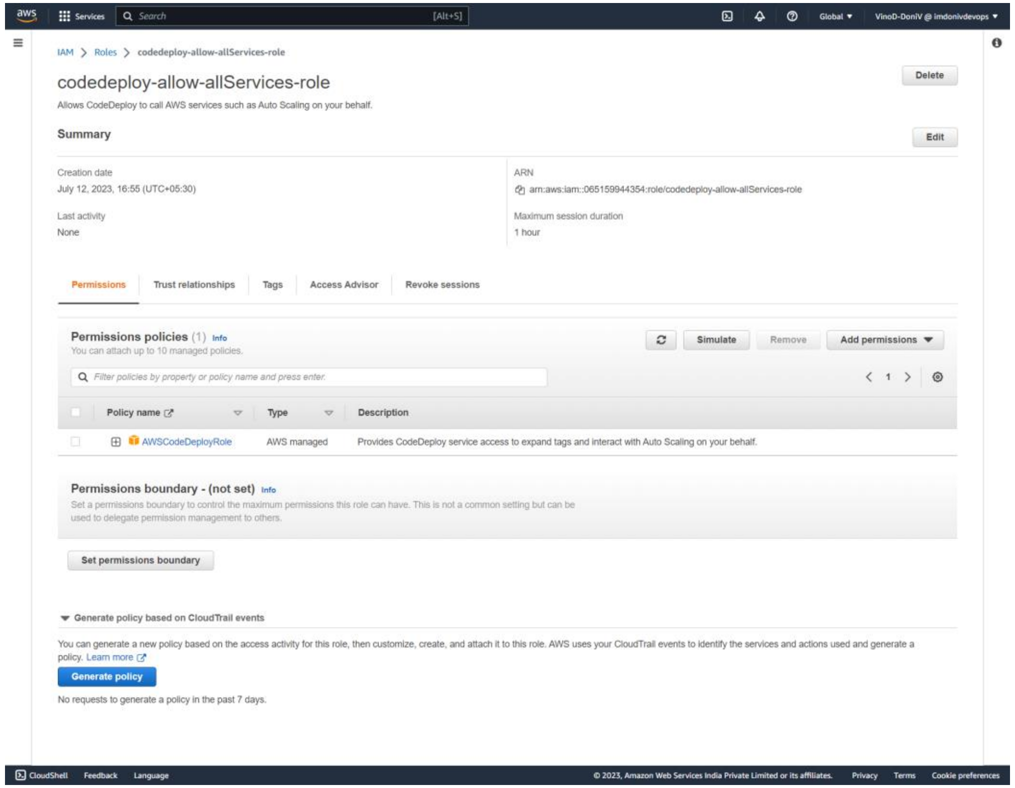
Policy name	Type	Description
AmazonEC2RoleforAWSCodeDeploy	AWS managed	Provides EC2 access to S3 bucket to download revision. This role is needed by the CodeDeploy agent on EC2 instances.

Permissions boundary - (not set) Info
Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting but can be used to delegate permission management to others.

Generate policy based on CloudTrail events
You can generate a new policy based on the access activity for this role, then customize, create, and attach it to this role. AWS uses your CloudTrail events to identify the services and actions used and generate a policy. Learn more

No requests to generate a policy in the past 7 days.

© 2023, Amazon Web Services India Private Limited or its affiliates.



codedeploy-allow-allServices-role

Allows CodeDeploy to call AWS services such as Auto Scaling on your behalf.

Summary

Creation date
July 12, 2023, 16:55 (UTC+05:30)

Last activity
None

ARN
arn:aws:iam::065159944354:role/codedeploy-allow-allServices-role

Maximum session duration
1 hour

Permissions policies (1) Info
You can attach up to 10 managed policies.

Policy name	Type	Description
AWSCodeDeployRole	AWS managed	Provides CodeDeploy service access to expand tags and interact with Auto Scaling on your behalf.

Permissions boundary - (not set) Info
Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting but can be used to delegate permission management to others.

Generate policy based on CloudTrail events
You can generate a new policy based on the access activity for this role, then customize, create, and attach it to this role. AWS uses your CloudTrail events to identify the services and actions used and generate a policy. Learn more

No requests to generate a policy in the past 7 days.

© 2023, Amazon Web Services India Private Limited or its affiliates.

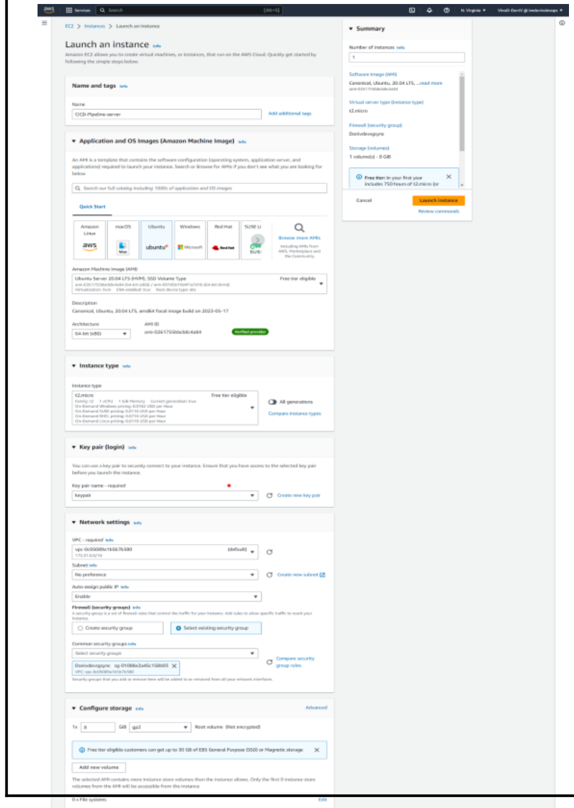
Step-2 :

Create EC2 resources and launch EC2 instance.

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.
2. Choose Launch Instance.
3. For Name and tags, text the name (CICD-Pipeline-server).
4. For application and OS Images(Amazon Machine Image),Choose an Amazon Machine Image (AMI), find the Ubuntu 20.04 LTS AMI at the top of the list and choose Select, rest all default.
5. For Instance type, Choose instance type in the dropdown list t2.micro.
6. For Key pair (login), give the existing key pair or else create a new key pair (by using putty create .ppk file or using ssh client create .pem file), Here I am using keypair.pem file.
7. On the Network settings, do following:
   - i. VPC : Default
   - ii. Subnet : Default
   - iii. Auto-assign public IP : Choose Enable
   - iv. For Firewall (security groups), Choose Select existing security group (Donivdevsgsync)
8. Configure storage leave as default.
9. On Advanced details, do the following:
   - i. For IAM instance profile, in dropdown list choose existing role (EC2-Codedeploy-Webportfolio-role).
   - ii. Rest all leave as it default
   - If didn’t give in IAM role in Advance settings, do the following

     In home page click our instance and click on actions on header > security > modify IAM role > chose IAM role which we created and update IAM role

     now we need to reboot select> instance state >reboot instance
10. Click on Launch instance.



EC2 > Instances > Launch an instance

Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Name: CICD-Pipeline-server

Application and OS Images (Amazon Machine Image)

Amazon Machine Image (AMI): Ubuntu Server 20.04 LTS (HVM), SSD Volume Type (Free tier eligible)

Description: Canonical, Ubuntu, 20.04 LTS, amd64 focal image build on 2023-05-17

Architecture: 64-bit (x86)

AMI ID: ami-0261755bbcb8c4a84

Instance type

Instance type: t2.micro (Free tier eligible)

Family: t2 1 vCPU 1 GiB Memory Current generation: true

Key pair (login)

Key pair name - required: keypair

Network settings

VPC - required: vpc-0c05089a1b5b7b380 (default)

Subnet: No preference

Auto-assign public IP: Enable

Firewall (security groups): ○ Create security group ● Select existing security group

Common security groups: Donivdevopsync sg-01088e2a45c158b93

Configure storage

1x 8 GiB gp2 Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

Summary

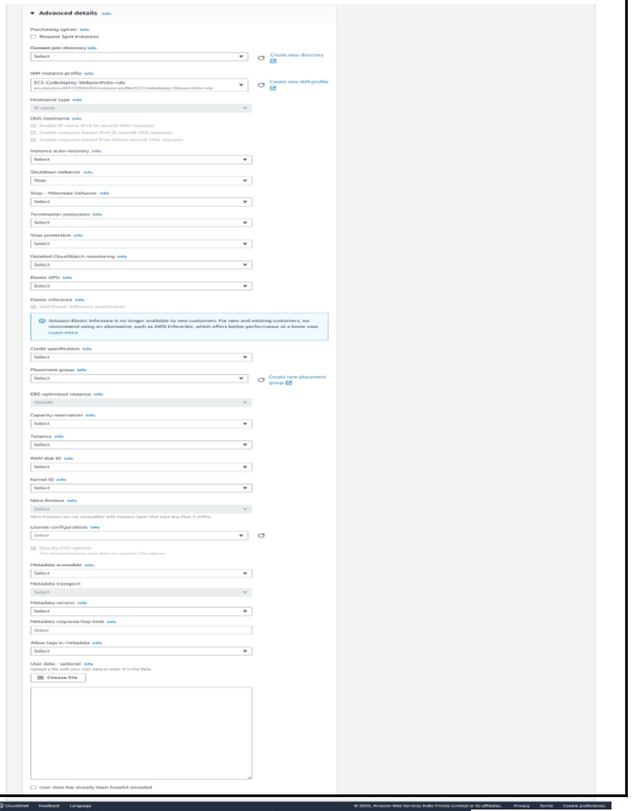
Number of instances: 1

Software Image (AMI): Canonical, Ubuntu, 20.04 LTS, ...read more

Virtual server type (instance type): t2.micro

Firewall (security group): Donivdevopsync

Storage (volumes): 1 volume(s) - 8 GiB



Advanced details

Purchasing option: □ Request Spot Instances

Domain join directory: Select

IAM instance profile: EC2-Codedeploy-Webportfolio-role

Hostname type: IP name

Instance auto-recovery: Select

Shutdown behavior: Stop

Stop - Hibernate behavior: Select

Termination protection: Select

Stop protection: Select

Detailed CloudWatch monitoring: Select

Elastic GPU: Select

Amazon Elastic Inference is no longer available to new customers. For new and existing customers, we recommend using an alternative, such as AWS Inferentia, which offers better performance at a lower cost.

Credit specification: Select

Placement group: Select

EBS-optimized instance: Disable

Capacity reservation: Select

Tenancy: Select

RAM disk ID: Select

Kernel ID: Select

Nitro Enclave: Select

License configurations: Select

Metadata accessible: Select

Metadata transport: Select

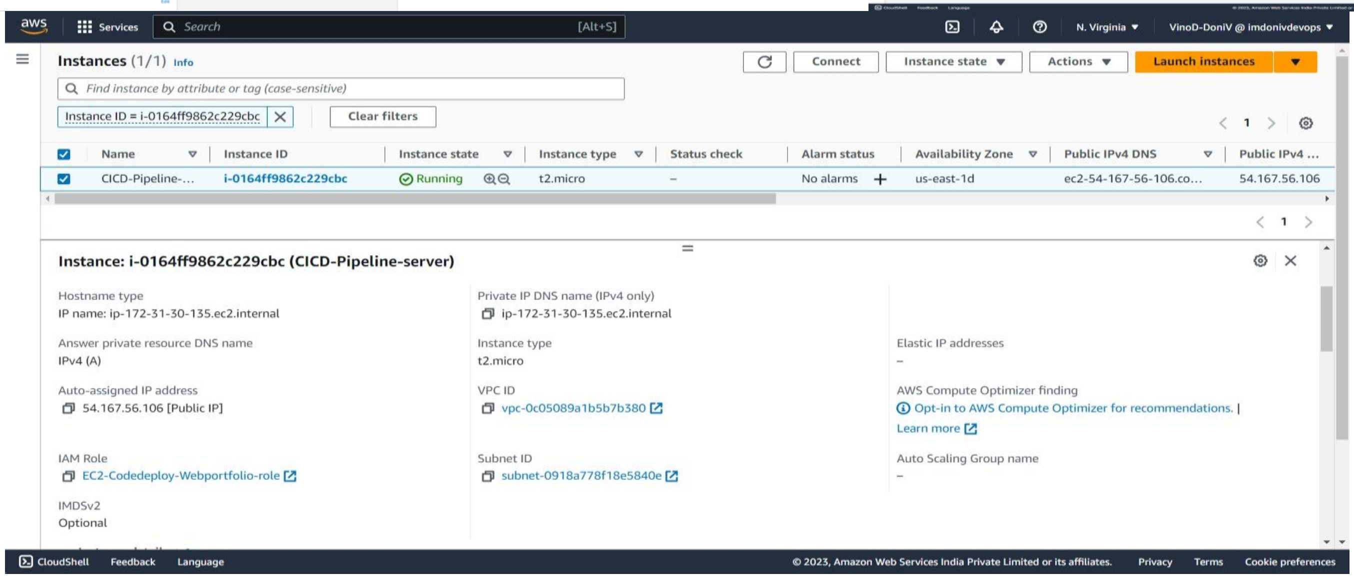
Metadata version: Select

Metadata response hop limit: Select

Allow tags in metadata: Select

User data - optional

□ User data has already been base64 encoded



Instances (1/1) Info

Find instance by attribute or tag (case-sensitive)

Instance ID = i-0164ff9862c229cbc ✕   Clear filters

☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...
☑	CICD-Pipeline-...	i-0164ff9862c229cbc	✓ Running	t2.micro	–	No alarms +	us-east-1d	ec2-54-167-56-106.co...	54.167.56.106

Instance: i-0164ff9862c229cbc (CICD-Pipeline-server)

Hostname type
IP name: ip-172-31-30-135.ec2.internal

Private IP DNS name (IPv4 only)
ip-172-31-30-135.ec2.internal

Answer private resource DNS name
IPv4 (A)

Instance type
t2.micro

Elastic IP addresses
–

Auto-assigned IP address
54.167.56.106 [Public IP]

VPC ID
vpc-0c05089a1b5b7b380

AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more

IAM Role
EC2-Codedeploy-Webportfolio-role

Subnet ID
subnet-0918a778f18e5840e

Auto Scaling Group name
–

IMDSv2
Optional

CloudShell
Feedback
Language
© 2023, Amazon Web Services India Private Limited or its affiliates.
Privacy
Terms
Cookie preferences

Step-3 :

Code Deploy Agent Installation

Interact with AWS EC2 instance via SSH Client , Here I used Git bash.

1. Check that Git is installed, Open the terminal and run the below command. If it is not installed, the download git from the official website: https://git-scm.com/downloads.

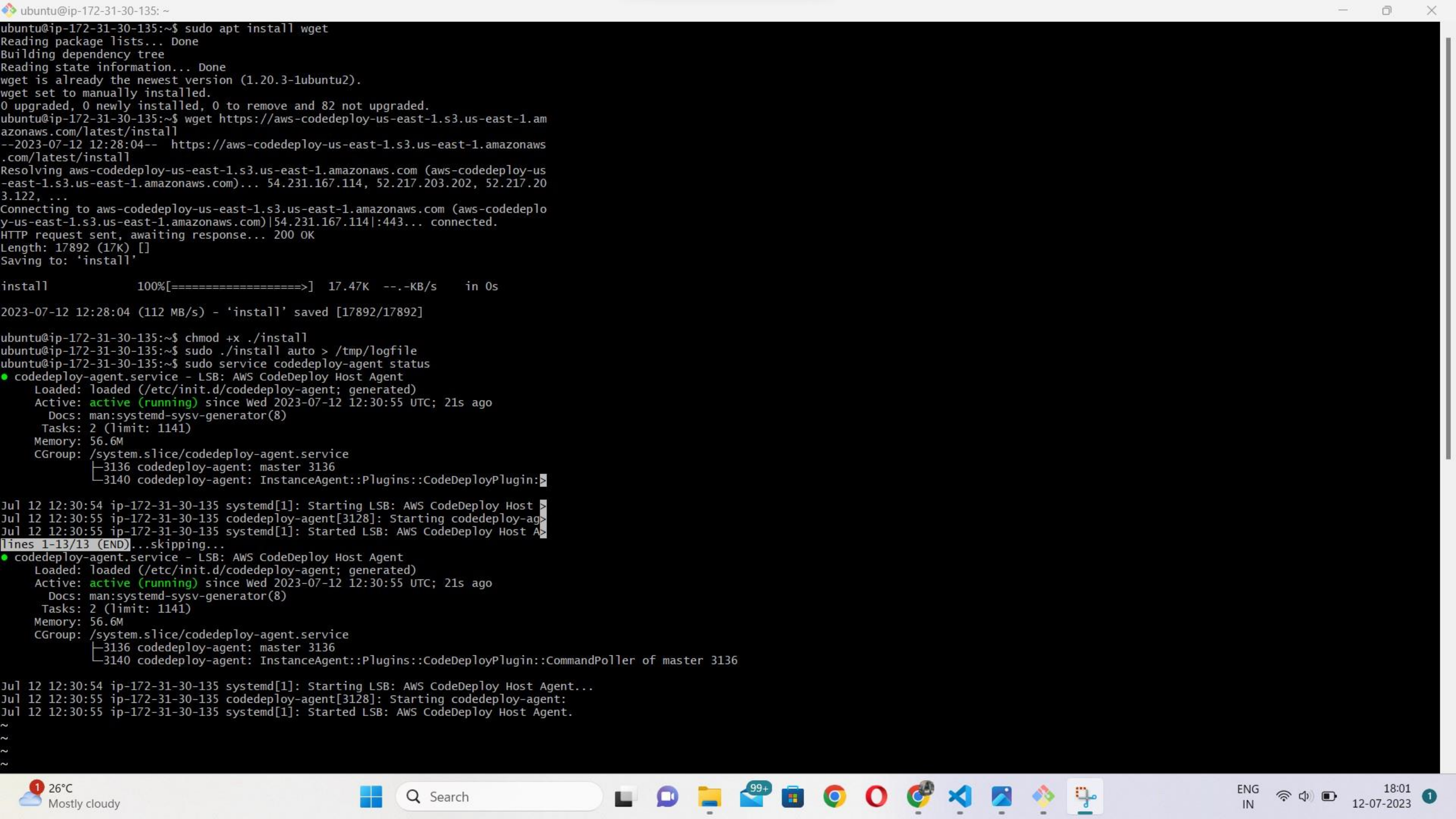
   ```
   $ git –version
   ```

2. Open the git bash terminal on where .pem key file stored in that folder. Or else locate that folder by using commands.
3. Set the permission .pem file, Here my pem file is keypair.pem.

   ```
   $ chmod 400 keypair.pem
   ```

4. Open EC2 console, Select your instance and click “Connect”.
5. For Connect to instance, Choose SSH client tab. Here shows everything.
6. Copy the example (ssh -i "keypair.pem" ubuntu@ec2-172-31-30-135.compute-1.amazonaws.com) and paste in your git bash terminal.
7. Once connect to the instance, do following run commands:

   ```
   $ sudo apt update  # Update our machine
   $ sudo apt install ruby-full   # Install ruby-full in our machine
   $ sudo apt install wget  # Install wget (web get)
   $ wget https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com/lastest/install   # Download the files and
                                                                                              interact with S3
   $ chmod +x ./install      #ext up, change the permission on the install file
   $ sudo ./install auto > /tmp/logfile            # Install the codedeploy-agent
   $ sudo service codedeploy-agent status            # Agent is running or not
   ```



```
ubuntu@ip-172-31-30-135:~$ sudo apt install wget
Reading package lists... Done
Building dependency tree
Reading state information... Done
wget is already the newest version (1.20.3-1ubuntu2).
wget set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 82 not upgraded.
ubuntu@ip-172-31-30-135:~$ wget https://aws-codedeploy-us-east-1.s3.us-east-1.am
azonaws.com/latest/install
--2023-07-12 12:28:04--  https://aws-codedeploy-us-east-1.s3.us-east-1.amazonaws
.com/latest/install
Resolving aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com (aws-codedeploy-us
-east-1.s3.us-east-1.amazonaws.com)... 54.231.167.114, 52.217.203.202, 52.217.20
3.122, ...
Connecting to aws-codedeploy-us-east-1.s3.us-east-1.amazonaws.com (aws-codedeplo
y-us-east-1.s3.us-east-1.amazonaws.com)|54.231.167.114|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 17892 (17K) []
Saving to: 'install'

install             100%[===================>]  17.47K  --.-KB/s    in 0s

2023-07-12 12:28:04 (112 MB/s) - 'install' saved [17892/17892]

ubuntu@ip-172-31-30-135:~$ chmod +x ./install
ubuntu@ip-172-31-30-135:~$ sudo ./install auto > /tmp/logfile
ubuntu@ip-172-31-30-135:~$ sudo service codedeploy-agent status
● codedeploy-agent.service - LSB: AWS CodeDeploy Host Agent
     Loaded: loaded (/etc/init.d/codedeploy-agent; generated)
     Active: active (running) since Wed 2023-07-12 12:30:55 UTC; 21s ago
       Docs: man:systemd-sysv-generator(8)
      Tasks: 2 (limit: 1141)
     Memory: 56.6M
     CGroup: /system.slice/codedeploy-agent.service
             ├─3136 codedeploy-agent: master 3136
             └─3140 codedeploy-agent: InstanceAgent::Plugins::CodeDeployPlugin:>

Jul 12 12:30:54 ip-172-31-30-135 systemd[1]: Starting LSB: AWS CodeDeploy Host >
Jul 12 12:30:55 ip-172-31-30-135 codedeploy-agent[3128]: Starting codedeploy-ag>
Jul 12 12:30:55 ip-172-31-30-135 systemd[1]: Started LSB: AWS CodeDeploy Host A>
lines 1-13/13 (END)...skipping...
● codedeploy-agent.service - LSB: AWS CodeDeploy Host Agent
     Loaded: loaded (/etc/init.d/codedeploy-agent; generated)
     Active: active (running) since Wed 2023-07-12 12:30:55 UTC; 21s ago
       Docs: man:systemd-sysv-generator(8)
      Tasks: 2 (limit: 1141)
     Memory: 56.6M
     CGroup: /system.slice/codedeploy-agent.service
             ├─3136 codedeploy-agent: master 3136
             └─3140 codedeploy-agent: InstanceAgent::Plugins::CodeDeployPlugin::CommandPoller of master 3136

Jul 12 12:30:54 ip-172-31-30-135 systemd[1]: Starting LSB: AWS CodeDeploy Host Agent...
Jul 12 12:30:55 ip-172-31-30-135 codedeploy-agent[3128]: Starting codedeploy-agent:
Jul 12 12:30:55 ip-172-31-30-135 systemd[1]: Started LSB: AWS CodeDeploy Host Agent.
~
~
~
~
```

Step-4 :

** Project Structure Configuration**

Create a **gunicorn** folder in the root directory of the Django project (WebPortfolio) with the following:

i. Create a *gunicorn.service* file in the **gunicorn folder** with the following contents:

```
[Unit]
Description=gunicorn daemon
Requires=gunicorn.socket
After=network.target
[Service]
User=ubuntu
Group=www-data
WorkingDirectory=/home/ubuntu/WebPortfolio
ExecStart=/home/ubuntu/env/bin/gunicorn --access-logfile - --workers 3 --bind
unix:/home/ubuntu/WebPortfolio/portfolio/portfolio.sock portfolio.wsgi:application
[Install]
WantedBy=multi-user.target
```

ii. Create a *gunicorn.socket* file in the gunicorn folder with the following contents:

```
[Unit]
Description=gunicorn socket
[Socket]
ListenStream=/run/gunicorn.sock
[Install]
WantedBy=sockets.target
```

Create a ****nginx**** folder in the root directory of the Django project (WebPortfolio) with the following:

Create a *nginx.conf* file in the **nginx folder** with the following contents:

```
server {
    listen 80 default_server;
    server_name 54.167.56.106;   # Add Public Ip address from ec2 instance.
    location = /favicon.ico { access_log off; log_not_found off; }
    location /staticfiles/ {
        root /home/ubuntu/WebPortfolio;
    }
    location / {
        include proxy_params;
        proxy_pass http://unix:/run/gunicorn.sock;
    }
}
```

Create a ****scripts**** folder in the root directory of the Django project (WebPortfolio) with the following:

Create a ***gunicorn.sh, python_dependencies.sh, instance_os_dependencies.sh, nginx.sh, after_install.sh, clean_instance.sh, start_app.sh,*** files in the **scripts folder** with the following contents:

gunicorn.sh

```
#!/usr/bin/bash
sudo cp
/home/ubuntu/WebPortfolio/gunicorn/gunicorn.socket  /etc/s
ystemd/system/gunicorn.socket
sudo cp
/home/ubuntu/WebPortfolio/gunicorn/gunicorn.service  /etc/
systemd/system/gunicorn.service

sudo systemctl start gunicorn.service
sudo systemctl enable gunicorn.service
```

Python_dependencies.sh

```
#!/usr/bin/env bash

virtualenv /home/ubuntu/env
source /home/ubuntu/env/bin/activate
pip install -r /home/ubuntu/WebPortfolio/requirements.txt
```

Instance_os_dependencies.sh

```
#!/usr/bin/env bash

sudo apt install -y python3-pip
sudo apt install -y nginx
sudo apt install -y virtualenv
```

nginx.sh

```
#!/usr/bin/bash

sudo systemctl daemon-reload
sudo rm -f /etc/nginx/sites-enabled/default

sudo cp /home/ubuntu/WebPortfolio/nginx/nginx.conf /etc/nginx/sites-available/portfolio
sudo ln -s /etc/nginx/sites-available/portfolio /etc/nginx/sites-enabled/
#sudo ln -s /etc/nginx/sites-available/blog /etc/nginx/sites-enabled
#sudo nginx -t
sudo gpasswd -a www-data ubuntu
sudo systemctl restart nginx
```

after_install.sh

```
#!/usr/bin/bash

echo "Pull Finished"
sudo systemctl daemon-reload
sudo systemctl restart nginx
```

clean_instance.sh

```
#!/usr/bin/env bash
#
sudo rm -rf /home/ubuntu/WebPortfolio/*
```

start_app.sh

```
#!/usr/bin/bash

sed -i 's/\[]/\["54.144.250.113"]/' /home/ubuntu/WebPortfolio/portfolio/settings.py

python manage.py migrate
python manage.py makemigrations
python manage.py collectstatic
sudo service gunicorn restart
sudo service nginx restart
#sudo tail -f /var/log/nginx/error.log
#sudo systemctl reload nginx
#sudo tail -f /var/log/nginx/error.log
#sudo nginx -t
#sudo systemctl restart gunicorn
#sudo systemctl status gunicorn
#sudo systemctl status nginx
# Check the status
#systemctl status gunicorn
# Restart:
#systemctl restart gunicorn
#sudo systemctl status nginx
```

Create a ****appspec.yml**** file in the root directory of the Django project (WebPortfolio) with the following contents:

```
version: 0.0
os: linux
files:
  - source: /
    destination: /home/ubuntu/WebPortfolio
permissions:
  - object: /home/ubuntu/WebPortfolio
    owner: ubuntu
    group: ubuntu
hooks:
  BeforeInstall:
      - location: scripts/clean_instance.sh
        timeout: 300
        runas: ubuntu
  AfterInstall:
      - location: scripts/instance_os_dependencies.sh
        timeout: 300
        runas: ubuntu
      - location: scripts/python_dependencies.sh
        timeout: 300
        runas: ubuntu
      - location: scripts/gunicorn.sh
        timeout: 300
        runas: ubuntu
      - location: scripts/nginx.sh
        timeout: 300
        runas: ubuntu
  ApplicationStop:
      - location: scripts/stop_app.sh
        timeout: 300
        runas: ubuntu
  ApplicationStart:
      - location: scripts/start_app.sh
        timeout: 300
        runas: ubuntu
```

Create a **build**spec.yml**** file in the root directory of the Django project (WebPortfolio) with the following contents:

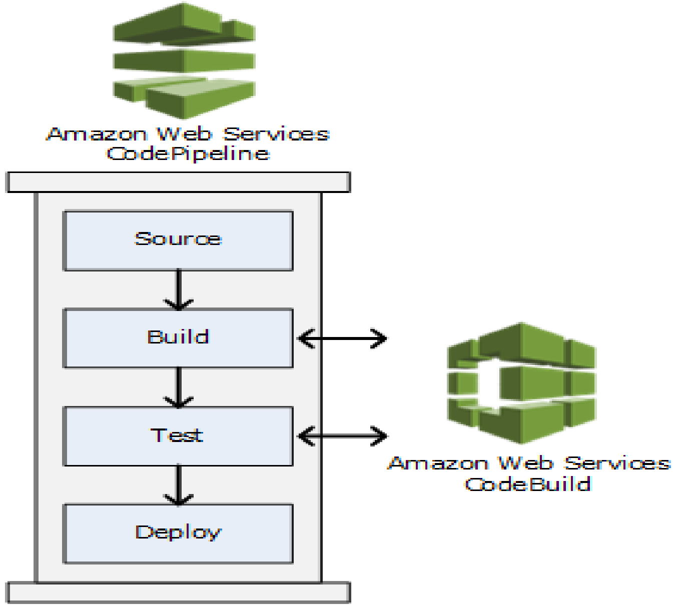
```
version: 0.1

# environment_variables:
  # plaintext:
  #   DJANGO_SETTINGS_MODULE: config.settings.test
  #   SECRET_KEY: nosecret
  #   DATABASE_DEFAULT_URL: sqlite:///db1.sqlite3
  #   DATABASE_STREAMDATA_URL: sqlite:///db2.sqlite3
  #   STREAM_INCOMING_PRIVATE_KEY: changeme
  #   STREAM_INCOMING_PUBLIC_KEY: changeme
  #   GOOGLE_API_KEY: changeme
  #   OPBEAT_ENABLED: False

phases:
  pre_build:
    commands:
      - echo Prebuild ops
      - pip3 install -r requirements.txt
  build:
    commands:
      - echo Building the application
  post_build:
    commands:
      - echo Build completed on `date`
artifacts:
  files:
    - '**/*'
```

Step-5 :

** CodePipeline Deploy **

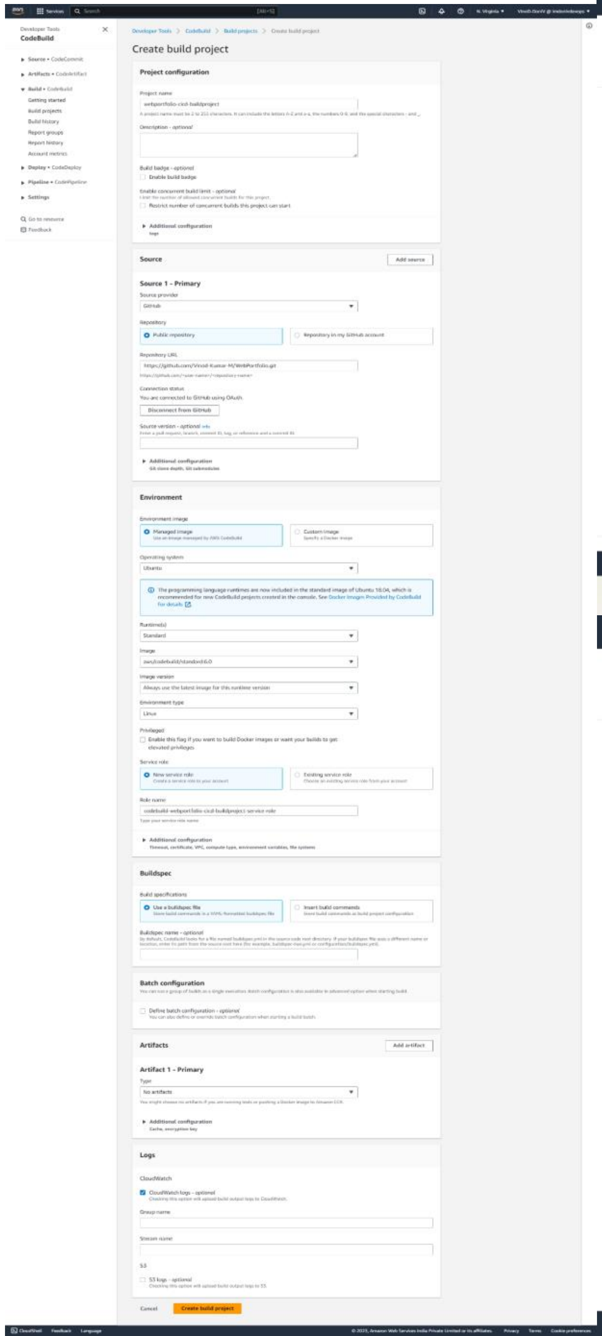

Amazon Web Services CodePipeline
Source
Build
Test
Deploy
Amazon Web Services CodeBuild

1. Open the AWS **CodePipeline** console at https://console.aws.amazon.com/codesuite/codepipeline/home.
2. In left side pane, Choose **CodeBuild** and then click **Getting started and Create project**.
3. On **Create build project**, do the following:
   - ❖ For **Project configuration**, Enter the Project name (**webportfolio-cicd-buildproject**). Rest all leave it as default.
   - ❖ For **Source**, In source provide dropdown list choose **GitHub**
     - In **Repository**, choose **Connect using Oauth** and then click **Connect to GitHub**
     - After click **Connect to GitHub**, Automatically open one tab in that tab **Processing Oauth request** shows and then click **Confirm**
     - After GitHub Connect, In Repository choose **Public repository**.
     - In **Repository URL**, Enter the Git repository url (https://github.com/Vinod-Kumar-M/WebPortfolio.git)
     - Rest all leave it as default.

- For **Environment**, do the following:
  - In Environment image, Choose **Managed image**.
  - In operating system, Choose **Ubuntu** in the dropdown list.
  - In Runtime(s), Choose **Standard** in the dropdown list.
  - In Image, Choose **aws/codebuild/standard:6.0** in the dropdown list # here always choose latest one
  - In Image version, Choose **Always use the latest image for this runtime version** in the dropdown list
  - In Environment type, Choose **Linux** in the dropdown list.
  - In Privileged, leave it as default.
  - In Service role, Choose **New service role**.
  - In Role name, give the **codebuild-webportfolio-cicd-buildproject-service-role.**
  - In Additional configuration, leave it as default.
- Rest all leave it as default, and then click **Create build project**.

- Choose the Build history in left pane you can see the Build history and Batch history
  - Click the Build history you can see the following details:
    - Build run, Status, Project, Build number, Source version, Submitter, Duration, Completed.

Create Build project



Developer Tools > CodeBuild > Build projects > Create build project

Create build project

Project configuration

Project name: webportfolio-cicd-buildproject

Description - optional

Build badge - optional
□ Enable build badge

Enable concurrent build limit - optional
□ Restrict number of concurrent builds this project can start

▶ Additional configuration
tags

Source [Add source]

Source 1 - Primary

Source provider: GitHub

Repository: ● Public repository ○ Repository in my GitHub account

Repository URL: https://github.com/Vinod-Kumar-M/WebPortfolio.git

Connection status
You are connected to GitHub using OAuth.
[Disconnect from GitHub]

Source version - optional Info

▶ Additional configuration
Git clone depth, Git submodules

Environment

Environment image: ● Managed image (Use an image managed by AWS CodeBuild) ○ Custom image (Specify a Docker image)

Operating system: Ubuntu

The programming language runtimes are now included in the standard image of Ubuntu 18.04, which is recommended for new CodeBuild projects created in the console. See Docker Images Provided by CodeBuild for details.

Runtime(s): Standard

Image: aws/codebuild/standard:6.0

Image version: Always use the latest image for this runtime version

Environment type: Linux

Privileged
□ Enable this flag if you want to build Docker images or want your builds to get elevated privileges

Service role: ● New service role (Create a service role in your account) ○ Existing service role (Choose an existing service role from your account)

Role name: codebuild-webportfolio-cicd-buildproject-service-role

▶ Additional configuration
Timeout, certificate, VPC, compute type, environment variables, file systems

Buildspec

Build specifications: ● Use a buildspec file (Store build commands in a YAML-formatted buildspec file) ○ Insert build commands (Store build commands as build project configuration)

Buildspec name - optional

Batch configuration

□ Define batch configuration - optional

Artifacts [Add artifact]

Artifact 1 - Primary

Type: No artifacts

▶ Additional configuration
Cache, encryption key

Logs

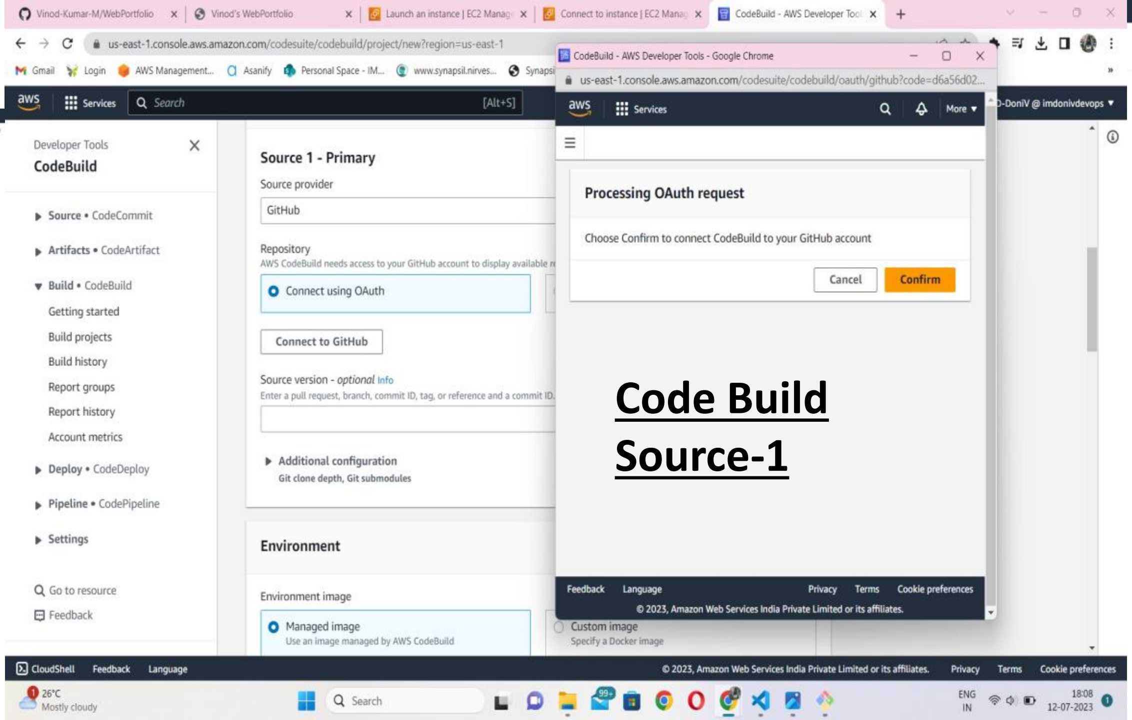
CloudWatch
☑ CloudWatch logs - optional
Checking this option will upload build output logs to CloudWatch.

Group name

Stream name

S3
□ S3 logs - optional
Checking this option will upload build output logs to S3.

Cancel [Create build project]



us-east-1.console.aws.amazon.com/codesuite/codebuild/project/new?region=us-east-1

Source 1 - Primary

Source provider: GitHub

Repository
AWS CodeBuild needs access to your GitHub account to display available r

● Connect using OAuth

[Connect to GitHub]

Source version - optional Info
Enter a pull request, branch, commit ID, tag, or reference and a commit ID.

▶ Additional configuration
Git clone depth, Git submodules

Environment

Environment image: ● Managed image (Use an image managed by AWS CodeBuild) ○ Custom image (Specify a Docker image)

CodeBuild - AWS Developer Tools - Google Chrome
us-east-1.console.aws.amazon.com/codesuite/codebuild/oauth/github?code=d6a56d02...

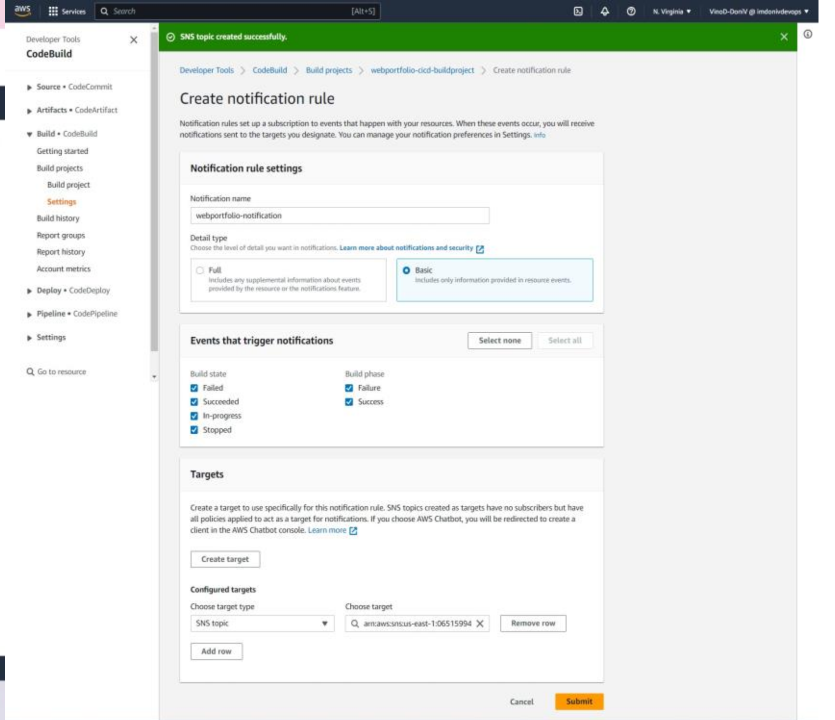
Processing OAuth request

Choose Confirm to connect CodeBuild to your GitHub account

[Cancel] [Confirm]

© 2023, Amazon Web Services India Private Limited or its affiliates.

Code Build Source-1



SNS topic created successfully.

Developer Tools > CodeBuild > Build projects > webportfolio-cicd-buildproject > Create notification rule

Create notification rule

Notification rules set up a subscription to events that happen with your resources. When these events occur, you will receive notifications sent to the targets you designate. You can manage your notification preferences in Settings. Info

Notification rule settings

Notification name: webportfolio-notification

Detail type
Choose the level of detail you want in notifications. Learn more about notifications and security

○ Full — Includes any supplemental information about events provided by the resource or the notifications feature.
● Basic — Includes only information provided in resource events.

Events that trigger notifications [Select none] [Select all]

Build state
☑ Failed
☑ Succeeded
☑ In-progress
☑ Stopped

Build phase
☑ Failure
☑ Success

Targets

Create a target to use specifically for this notification rule. SNS topics created as targets have no subscribers but have all policies applied to act as a target for notifications. If you choose AWS Chatbot, you will be redirected to create a client in the AWS Chatbot console. Learn more

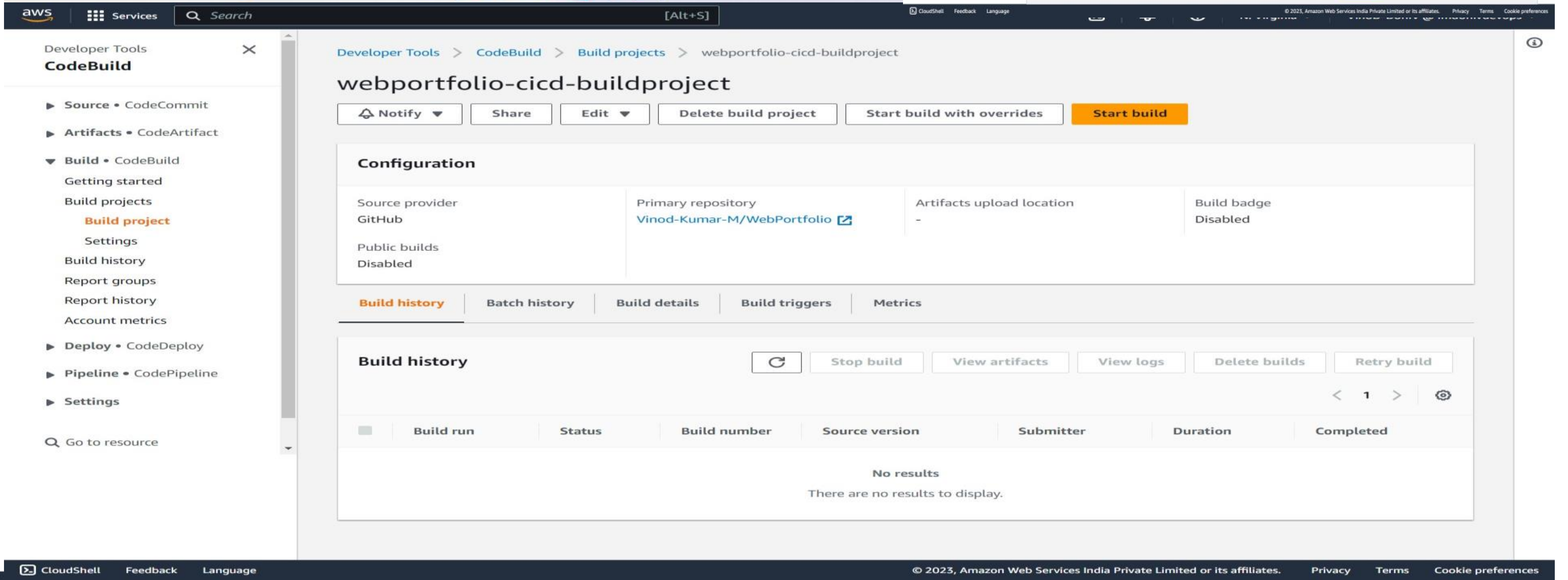
[Create target]

Configured targets

Choose target type: SNS topic
Choose target: arn:aws:sns:us-east-1:06515994
[Remove row]

[Add row]

Cancel [Submit]



Developer Tools > CodeBuild > Build projects > webportfolio-cicd-buildproject

webportfolio-cicd-buildproject

Configuration

Source provider	Primary repository	Artifacts upload location	Build badge
GitHub	Vinod-Kumar-M/WebPortfolio	-	Disabled

Public builds: Disabled

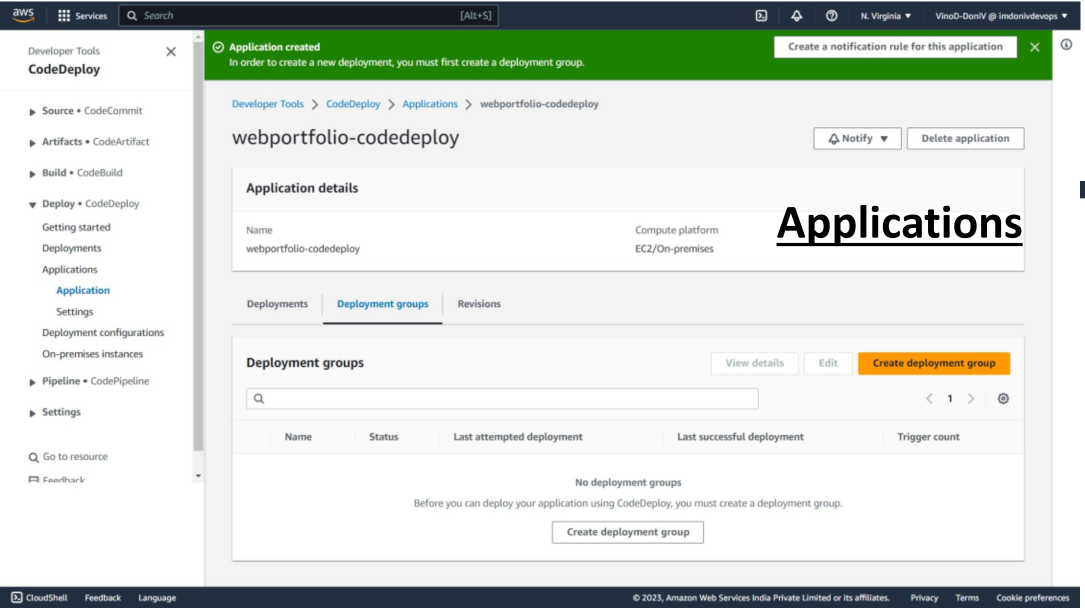
Build history

Build run	Status	Build number	Source version	Submitter	Duration	Completed

No results
There are no results to display.

© 2023, Amazon Web Services India Private Limited or its affiliates.

4. In left side pane, Choose **CodeDeploy** and then click **Getting started and Create application**.
    - For Application configuration, do the following:
        - Application : **webportfolio-codedeploy**
        - Compute platform : Choose **Ec2/On-premises** in the dropdown list.
        - Click **Create application**.
    - In Applications console, Click our application (webportfolio-codedeploy) > Deployment groups and then click **Create deployment group**.
    - For Create deployment group, do the following:
        - In Deployment group name, Enter a deployment group name (**webportfolio-codedeploy-group**).
        - In Service role, Enter a service role (**arn:aws:iam::065159944354:role/codedeploy-allow-allServices-role**)
        - In Deployment type, Choose how to deploy your application (**In-place**).
        - In Environment configuration, Check **Amazon Ec2 instances**.
        - In Tags, key: **Name** and Value: **CICD-Pipeline-server**, rest default
        - In Agent configuration with AWS Systems Manager, Install AWS CodeDeploy Agent : Choose **Now and schedule updates** > **Basic Scheduler** > **14 Days**.
        - In Deployment settings, Deployment configuration : Choose **CodeDeploy.AllAtOnce** in the dropdown list.
        - In Load balancer, Check **Enable load balancing** : **Application Load Balancer or Network Load Balancer** and the Choose a target group : **cicd-tg** (Our target group name) in the dropdown list.
        - Rest all leave it as default.
        - Click **Create deployment group**.



Applications

Application created
In order to create a new deployment, you must first create a deployment group.

Create a notification rule for this application

Developer Tools > CodeDeploy > Applications > webportfolio-codedeploy

webportfolio-codedeploy

Notify ▾ | Delete application

Application details

Name	Compute platform
webportfolio-codedeploy	EC2/On-premises

Deployments | **Deployment groups** | Revisions

Deployment groups

View details | Edit | Create deployment group

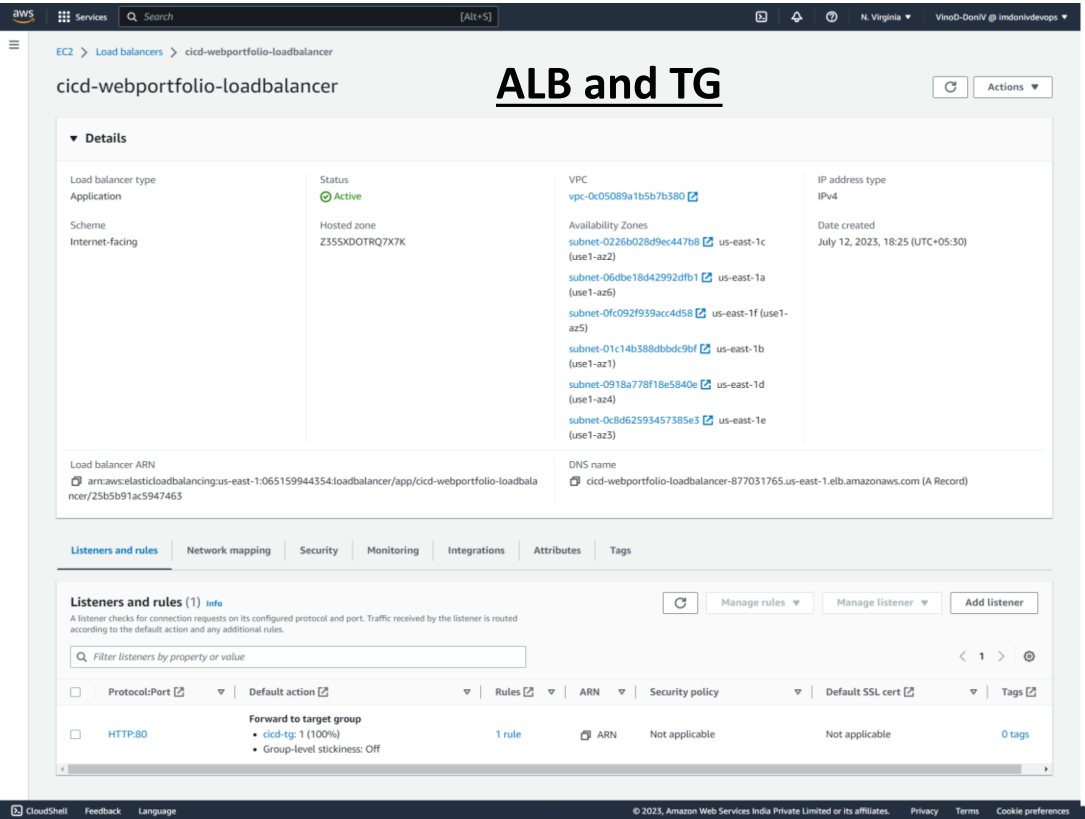
Name	Status	Last attempted deployment	Last successful deployment	Trigger count

No deployment groups

Before you can deploy your application using CodeDeploy, you must create a deployment group.

Create deployment group

© 2023, Amazon Web Services India Private Limited or its affiliates.



ALB and TG

EC2 > Load balancers > cicd-webportfolio-loadbalancer

cicd-webportfolio-loadbalancer

▼ **Details**

Load balancer type	Status	VPC	IP address type
Application	Active	vpc-0c05089a1b5b7b380	IPv4
Scheme	Hosted zone	Availability Zones	Date created
Internet-facing	Z35SXDOTRQ7X7K	subnet-0226b028d9ec447b8 us-east-1c (use1-az2) subnet-06dbe18d42992dfb1 us-east-1a (use1-az6) subnet-0fc092f939acc4d58 us-east-1f (use1-az5) subnet-01c14b388dbbdc9bf us-east-1b (use1-az1) subnet-0918a778f18e5840e us-east-1d (use1-az4) subnet-0c8d62593457385e3 us-east-1e (use1-az3)	July 12, 2023, 18:25 (UTC+05:30)

Load balancer ARN
arn:aws:elasticloadbalancing:us-east-1:065159944354:loadbalancer/app/cicd-webportfolio-loadbalancer/25b5b91ac5947463

DNS name
cicd-webportfolio-loadbalancer-877031765.us-east-1.elb.amazonaws.com (A Record)

Listeners and rules | Network mapping | Security | Monitoring | Integrations | Attributes | Tags

Listeners and rules (1) Info

A listener checks for connection requests on its configured protocol and port. Traffic received by the listener is routed according to the default action and any additional rules.

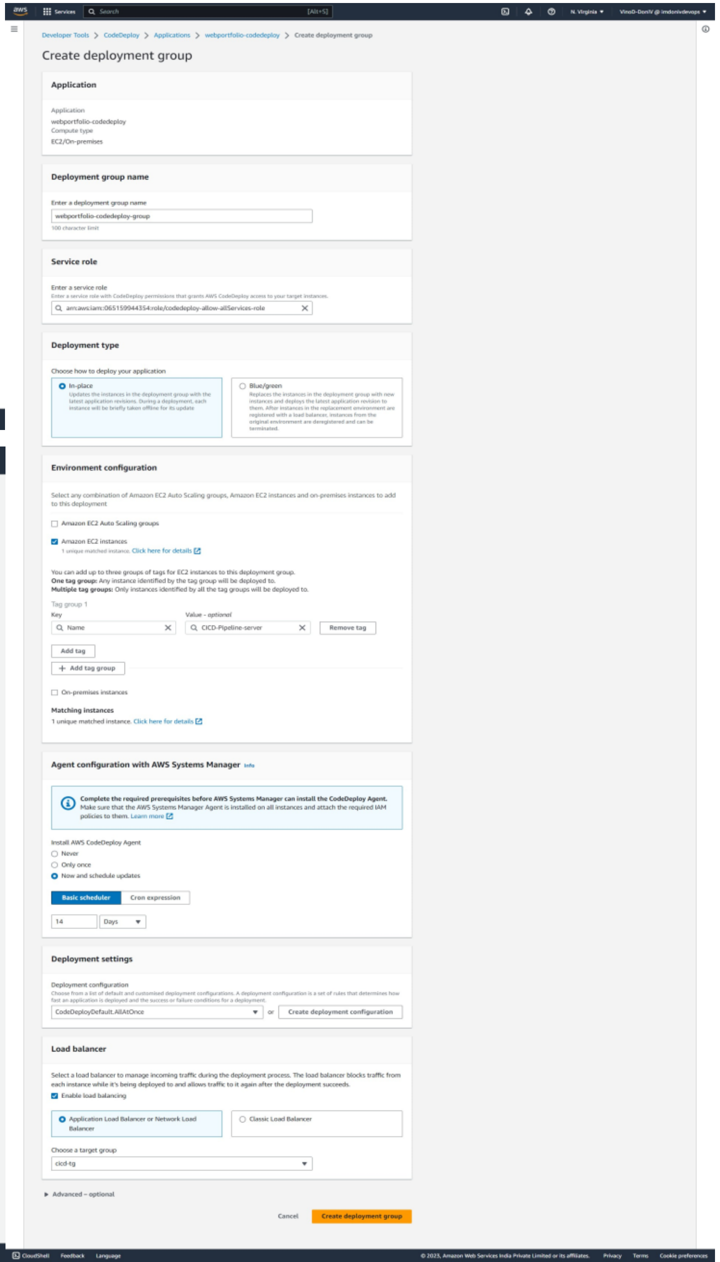
Manage rules ▾ | Manage listener ▾ | Add listener

Filter listeners by property or value

□	Protocol:Port	Default action	Rules	ARN	Security policy	Default SSL cert	Tags
□	HTTP:80	Forward to target group • cicd-tg: 1 (100%) • Group-level stickiness: Off	1 rule	ARN	Not applicable	Not applicable	0 tags

© 2023, Amazon Web Services India Private Limited or its affiliates.

Create Deployment Group



Developer Tools > CodeDeploy > Applications > webportfolio-codedeploy > Create deployment group

Create deployment group

Application

Application
webportfolio-codedeploy
Compute type
EC2/On-premises

Deployment group name

Enter a deployment group name
webportfolio-codedeploy-group
100 character limit

Service role

Enter a service role
Enter a service role with CodeDeploy permissions that grants AWS CodeDeploy access to your target instances.
arn:aws:iam::065159944354:role/codedeploy-allow-allServices-role

Deployment type

Choose how to deploy your application

● In-place
Updates the instances in the deployment group with the latest application revisions. During a deployment, each instance will be briefly taken offline for its update

○ Blue/green
Replaces the instances in the deployment group with new instances and deploys the latest application revision to them. After instances in the replacement environment are registered with a load balancer, instances from the original environment are deregistered and can be terminated.

Environment configuration

Select any combination of Amazon EC2 Auto Scaling groups, Amazon EC2 instances and on-premises instances to add to this deployment

□ Amazon EC2 Auto Scaling groups

☑ Amazon EC2 instances
1 unique matched instance. Click here for details

You can add up to three groups of tags for EC2 instances to this deployment group.
One tag group: Any instance identified by the tag group will be deployed to.
Multiple tag groups: Only instances identified by all the tag groups will be deployed to.

Tag group 1

Key	Value - *optional*	
Name	CICD-Pipeline-server	Remove tag

Add tag
+ Add tag group

□ On-premises instances

Matching instances
1 unique matched instance. Click here for details

Agent configuration with AWS Systems Manager Info

Complete the required prerequisites before AWS Systems Manager can install the CodeDeploy Agent. Make sure that the AWS Systems Manager Agent is installed on all instances and attach the required IAM policies to them. Learn more

Install AWS CodeDeploy Agent
○ Never
○ Only once
● Now and schedule updates

Basic scheduler | Cron expression

14 Days ▾

Deployment settings

Deployment configuration
Choose from a list of default and customised deployment configurations. A deployment configuration is a set of rules that determines how fast an application is deployed and the success or failure conditions for a deployment.
CodeDeployDefault.AllAtOnce ▾ or Create deployment configuration

Load balancer

Select a load balancer to manage incoming traffic during the deployment process. The load balancer blocks traffic from each instance while it's being deployed to and allows traffic to it again after the deployment succeeds.

☑ Enable load balancing

● Application Load Balancer or Network Load Balancer
○ Classic Load Balancer

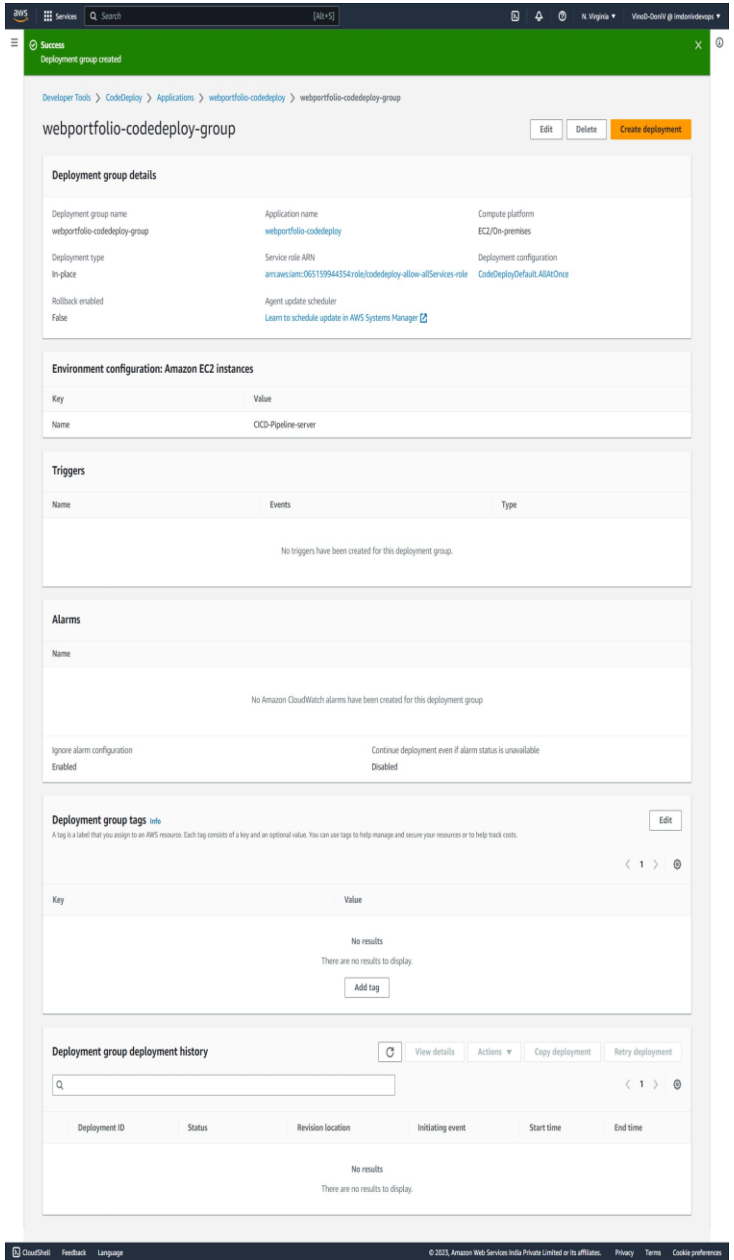
Choose a target group
cicd-tg ▾

▶ Advanced – optional

Cancel | Create deployment group

© 2023, Amazon Web Services India Private Limited or its affiliates.

Deployment Group Over view



Success
Deployment group created

Developer Tools > CodeDeploy > Applications > webportfolio-codedeploy > webportfolio-codedeploy-group

webportfolio-codedeploy-group

Edit | Delete | Create deployment

Deployment group details

Deployment group name	Application name	Compute platform
webportfolio-codedeploy-group	webportfolio-codedeploy	EC2/On-premises
Deployment type	Service role ARN	Deployment configuration
In-place	arn:aws:iam::065159944354:role/codedeploy-allow-allServices-role	CodeDeployDefault.AllAtOnce
Rollback enabled	Agent update scheduler	
False	Learn to schedule update in AWS Systems Manager	

Environment configuration: Amazon EC2 instances

Key	Value
Name	CICD-Pipeline-server

Triggers

Name	Events	Type

No triggers have been created for this deployment group.

Alarms

Name

No Amazon CloudWatch alarms have been created for this deployment group

Ignore alarm configuration	Continue deployment even if alarm status is unavailable
Enabled	Disabled

Deployment group tags Info

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to help manage and secure your resources or to help track costs.

Edit

Key	Value

No results
There are no results to display.
Add tag

Deployment group deployment history

View details | Actions ▾ | Copy deployment | Retry deployment

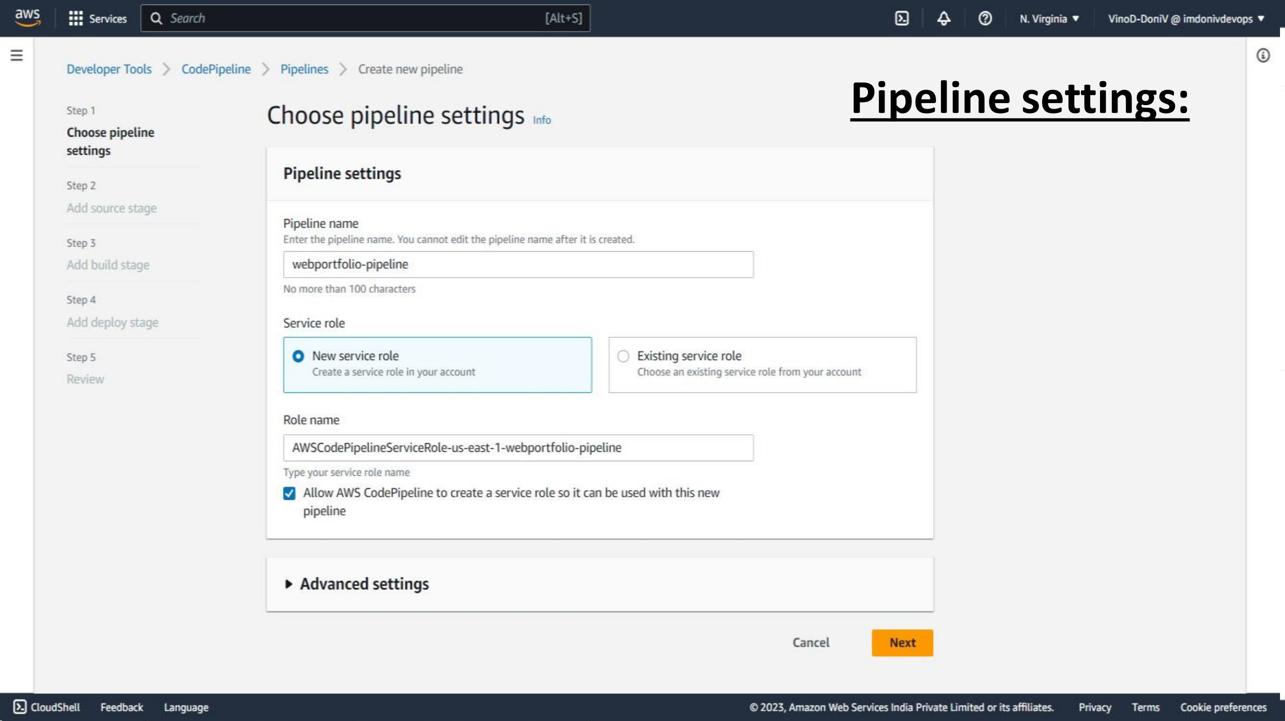
Deployment ID	Status	Revision location	Initiating event	Start time	End time

No results
There are no results to display.

© 2023, Amazon Web Services India Private Limited or its affiliates.

5. In left side pane, Choose **CodePipeline** and then click **Getting started and Create new pipeline.**
   - **Choose pipeline settings,** do the following:
     - In Pipeline settings, Enter the Pipeline name : **webportfolio-pipeline**.
     - In Service role, Choose **New service role**.
     - In Role name, Enter the **AWSCodePipelineServiceRole-us-east-1-webportfolio-pipeline**.
     - Check **Allow AWS CodePipeline** to create a service role so it can be used with this new pipeline.
     - Click **Next**.
   - Add source stage, do the following:
     - In Source provider, Choose **GitHub (version-1)** in the dropdown list and then click **Connected**.
     - In Repository, Give **Vinod-Kumar-M/WebPortfolio**
     - In Branch, Give **main**
     - In Change detection options, Choose **GitHub webhooks**
     - Click **Next**
   - Add build stage, do the following:
     - Build provider : choose **AWS Codebuild**, Region : **US East (N. Virginia)**, Project name : **webportfolio-cicd-buildproject**.
     - In Build type, Choose **Single build**
     - Click **Next**.
   - Add deploy stage, do the following:
     - Deploy provider : **AWS Codedeploy**, Region : **US East (N. Virginia),** Application name : **webportfolio-codedeploy**, Deployment group : **webportfolio-codedeploy-group .**
     - Click Next.
   - Review the steps and click **create pipeline**.

Pipeline settings:



Choose pipeline settings Info

Pipeline settings

Pipeline name
Enter the pipeline name. You cannot edit the pipeline name after it is created.
webportfolio-pipeline
No more than 100 characters

Service role
● New service role — Create a service role in your account
○ Existing service role — Choose an existing service role from your account

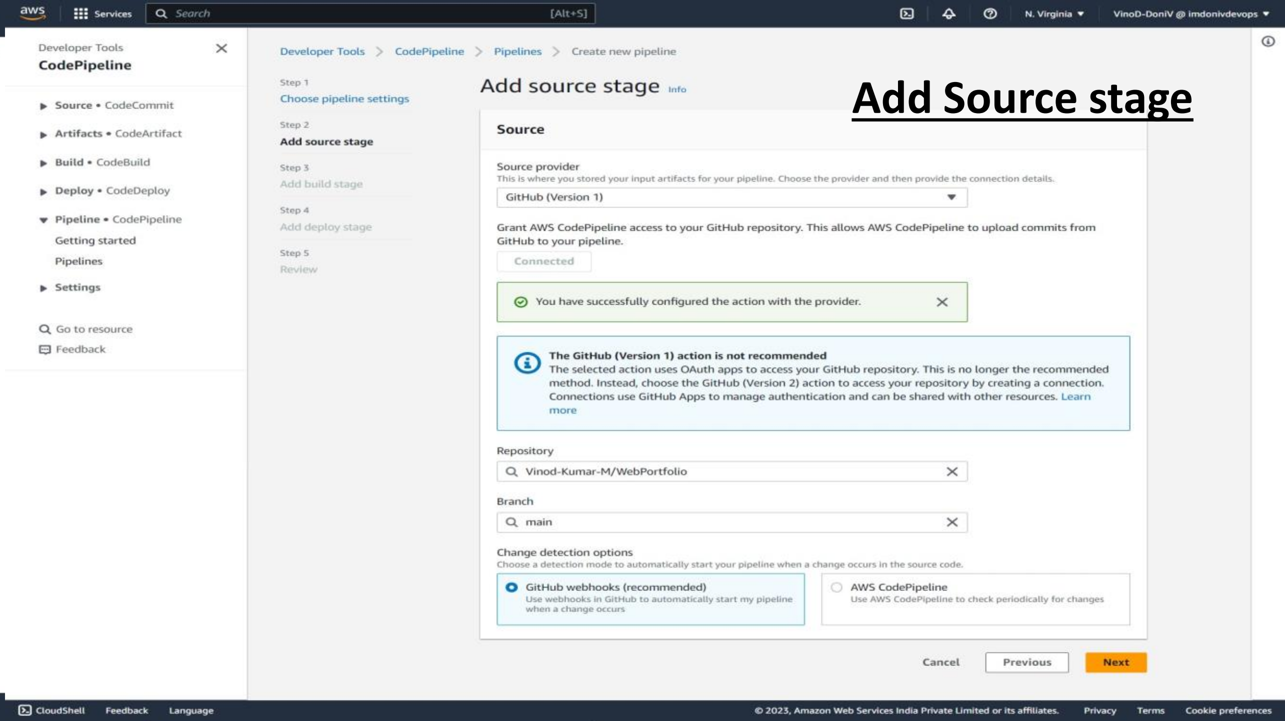
Role name
AWSCodePipelineServiceRole-us-east-1-webportfolio-pipeline
Type your service role name

☑ Allow AWS CodePipeline to create a service role so it can be used with this new pipeline

▸ Advanced settings

Cancel | Next

Add Source stage



Add source stage Info

Source

Source provider
This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.
GitHub (Version 1)

Grant AWS CodePipeline access to your GitHub repository. This allows AWS CodePipeline to upload commits from GitHub to your pipeline.
Connected

You have successfully configured the action with the provider.

The GitHub (Version 1) action is not recommended
The selected action uses OAuth apps to access your GitHub repository. This is no longer the recommended method. Instead, choose the GitHub (Version 2) action to access your repository by creating a connection. Connections use GitHub Apps to manage authentication and can be shared with other resources. Learn more

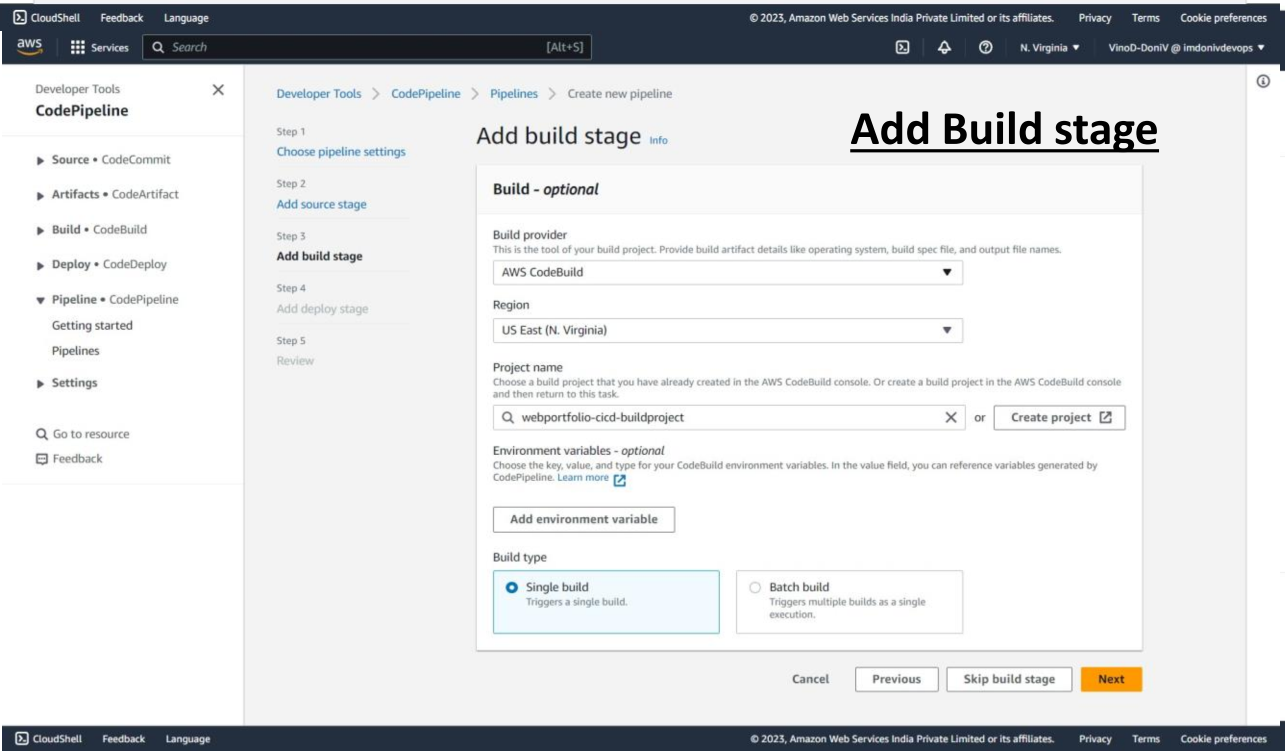
Repository
Vinod-Kumar-M/WebPortfolio

Branch
main

Change detection options
Choose a detection mode to automatically start your pipeline when a change occurs in the source code.
● GitHub webhooks (recommended) — Use webhooks in GitHub to automatically start my pipeline when a change occurs
○ AWS CodePipeline — Use AWS CodePipeline to check periodically for changes

Cancel | Previous | Next

Add Build stage



Add build stage Info

Build - *optional*

Build provider
This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.
AWS CodeBuild

Region
US East (N. Virginia)

Project name
Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.
webportfolio-cicd-buildproject or Create project

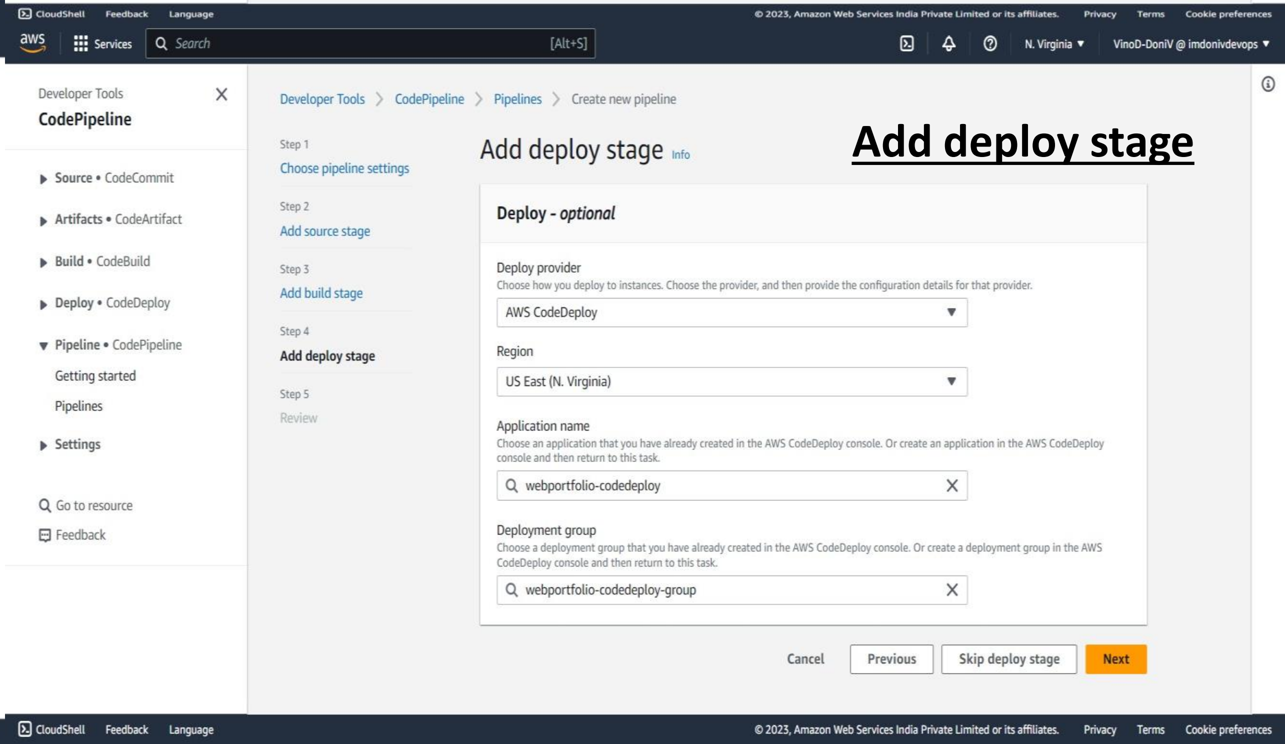
Environment variables - *optional*
Choose the key, value, and type for your CodeBuild environment variables. In the value field, you can reference variables generated by CodePipeline. Learn more

Add environment variable

Build type
● Single build — Triggers a single build.
○ Batch build — Triggers multiple builds as a single execution.

Cancel | Previous | Skip build stage | Next

Add deploy stage



Add deploy stage Info

Deploy - *optional*

Deploy provider
Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.
AWS CodeDeploy

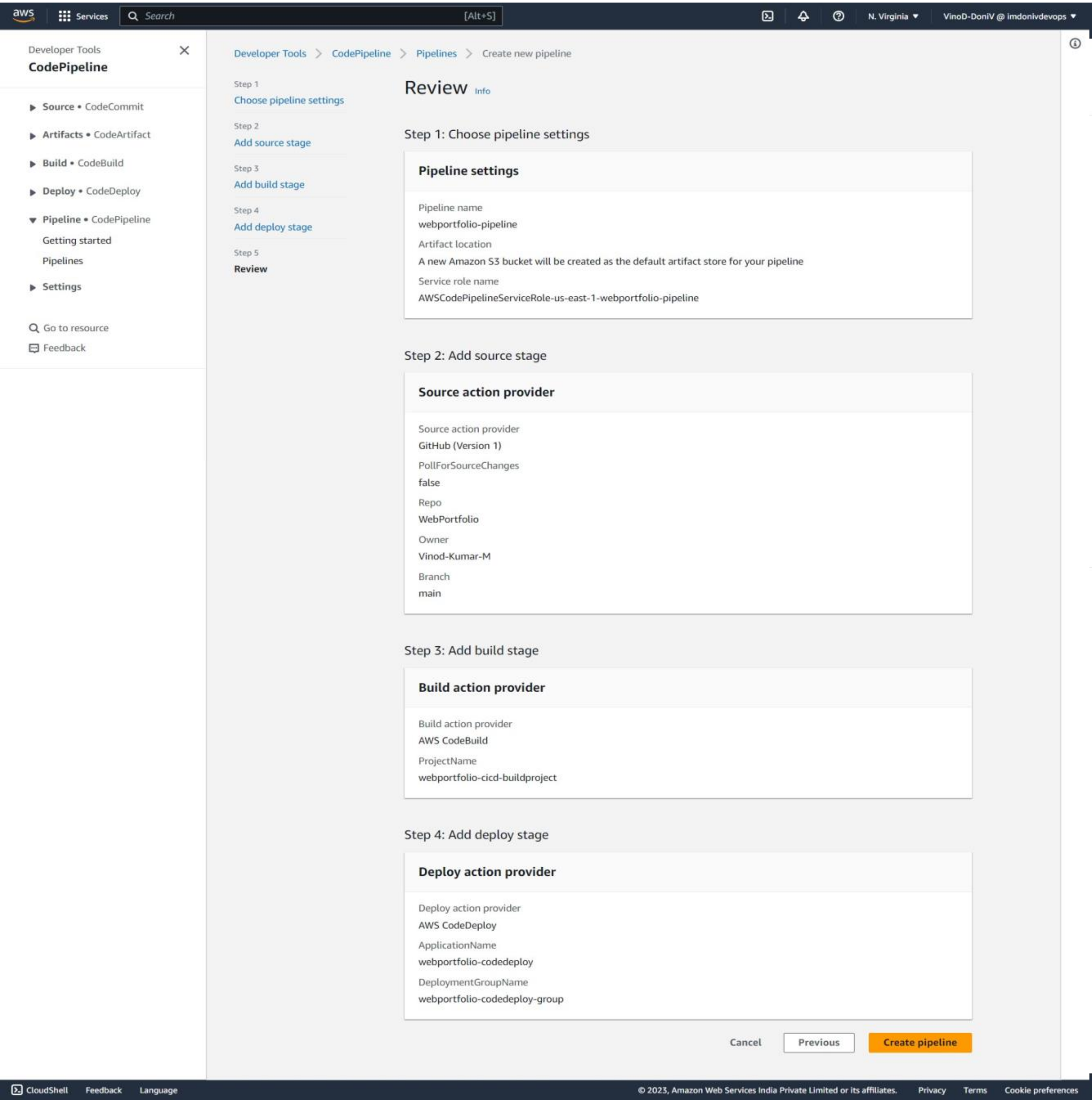
Region
US East (N. Virginia)

Application name
Choose an application that you have already created in the AWS CodeDeploy console. Or create an application in the AWS CodeDeploy console and then return to this task.
webportfolio-codedeploy

Deployment group
Choose a deployment group that you have already created in the AWS CodeDeploy console. Or create a deployment group in the AWS CodeDeploy console and then return to this task.
webportfolio-codedeploy-group

Cancel | Previous | Skip deploy stage | Next

Review of the CodePipeline:



Review

Step 1: Choose pipeline settings

Pipeline settings

Pipeline name
webportfolio-pipeline

Artifact location
A new Amazon S3 bucket will be created as the default artifact store for your pipeline

Service role name
AWSCodePipelineServiceRole-us-east-1-webportfolio-pipeline

Step 2: Add source stage

Source action provider

Source action provider
GitHub (Version 1)

PollForSourceChanges
false

Repo
WebPortfolio

Owner
Vinod-Kumar-M

Branch
main

Step 3: Add build stage

Build action provider

Build action provider
AWS CodeBuild

ProjectName
webportfolio-cicd-buildproject

Step 4: Add deploy stage

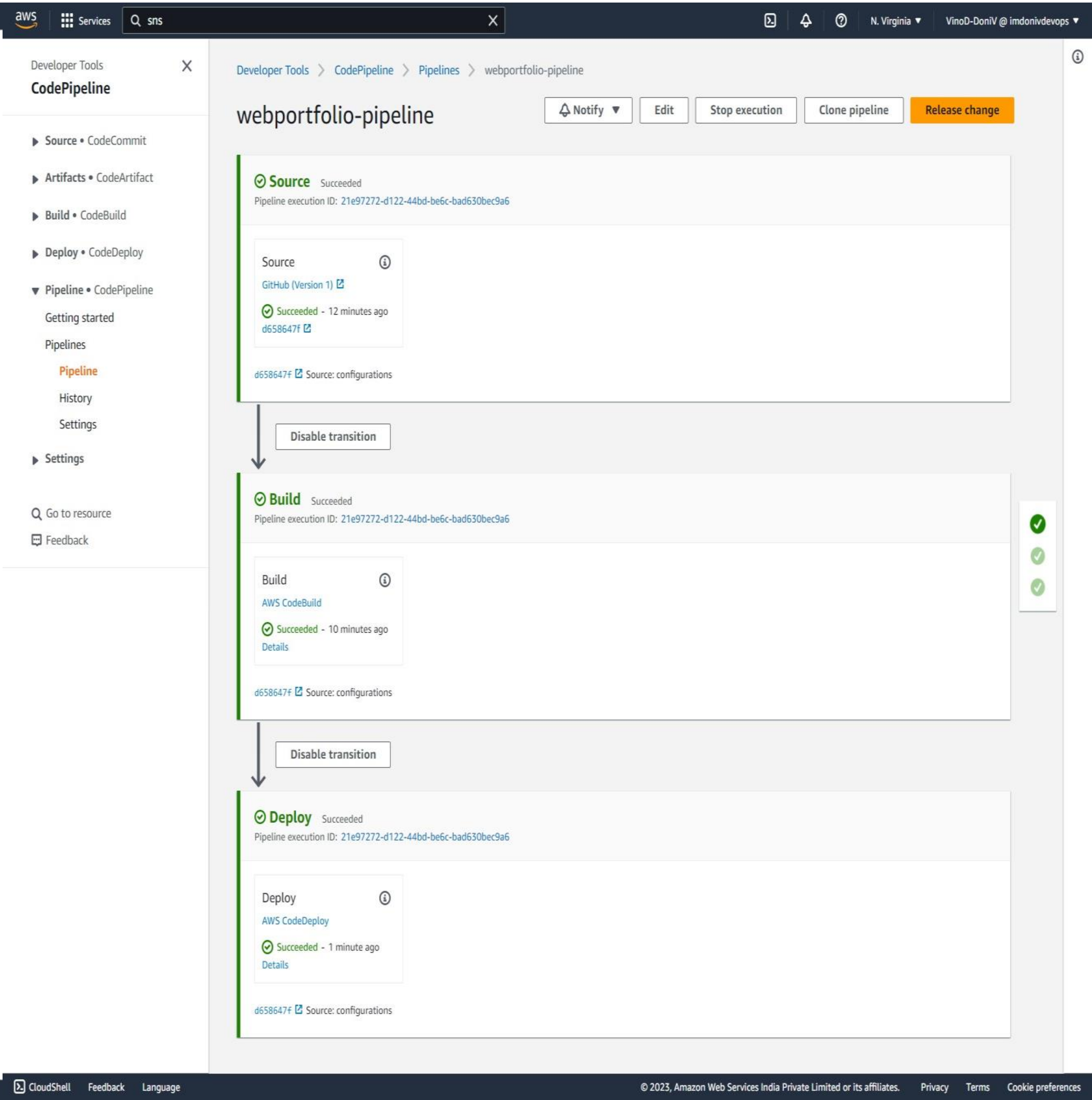
Deploy action provider

Deploy action provider
AWS CodeDeploy

ApplicationName
webportfolio-codedeploy

DeploymentGroupName
webportfolio-codedeploy-group

Pipeline Process :



webportfolio-pipeline

✓ **Source** Succeeded
Pipeline execution ID: 21e97272-d122-44bd-be6c-bad630bec9a6

Source
GitHub (Version 1)
✓ Succeeded - 12 minutes ago
d658647f

d658647f Source: configurations

✓ **Build** Succeeded
Pipeline execution ID: 21e97272-d122-44bd-be6c-bad630bec9a6

Build
AWS CodeBuild
✓ Succeeded - 10 minutes ago
Details

d658647f Source: configurations

✓ **Deploy** Succeeded
Pipeline execution ID: 21e97272-d122-44bd-be6c-bad630bec9a6

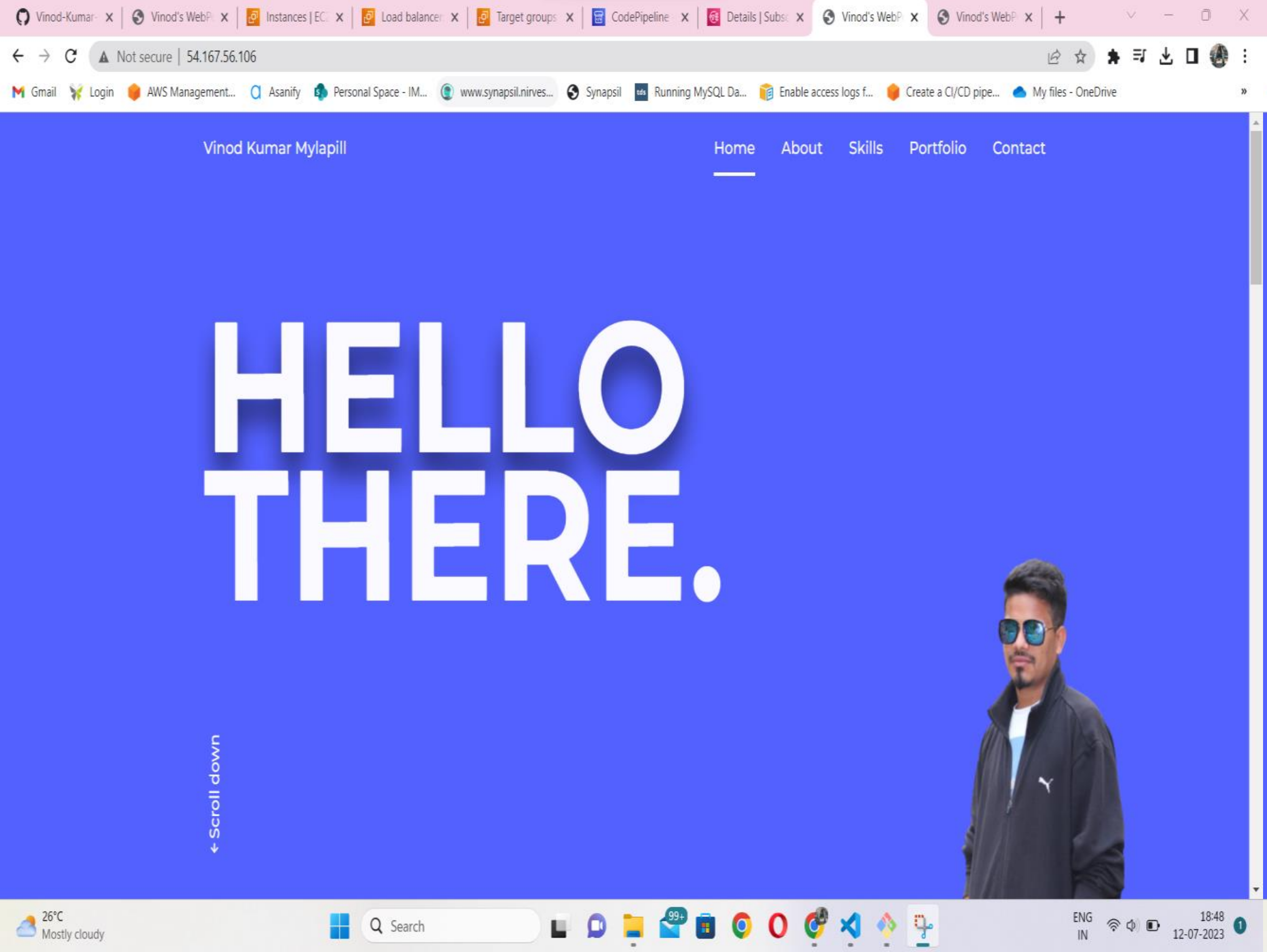
Deploy
AWS CodeDeploy
✓ Succeeded - 1 minute ago
Details

d658647f Source: configurations

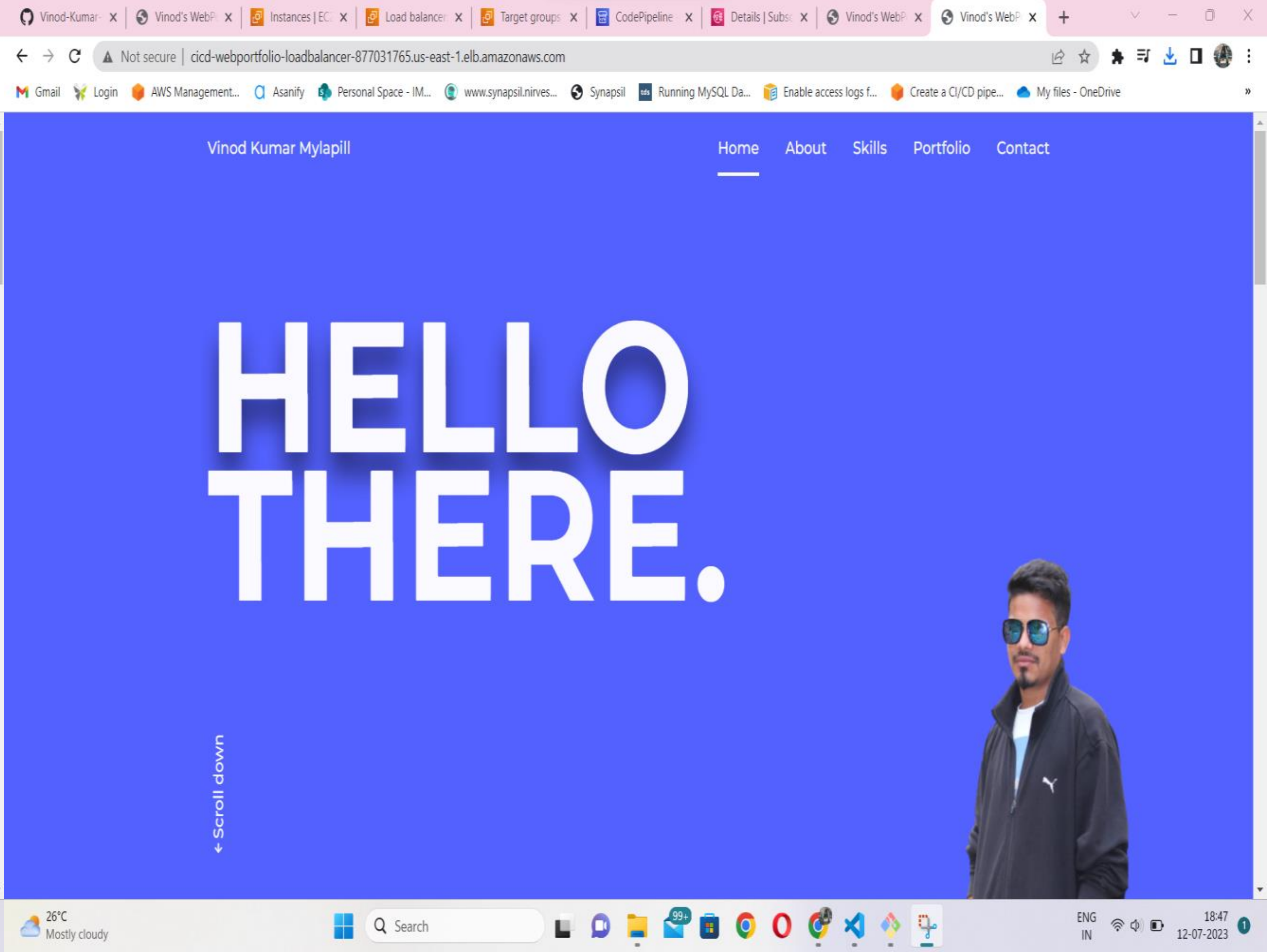
Pipeline Process shows **Succeeded** in all **Source, Build and Deploy** stages then after copy the **Public Ip** in EC2 instance and **DNS** in Load Balancer.
Paste in the browser you can see the result like below.

This result shows staging version.

<u>By using EC2 public IP (54.167.56.106)</u>



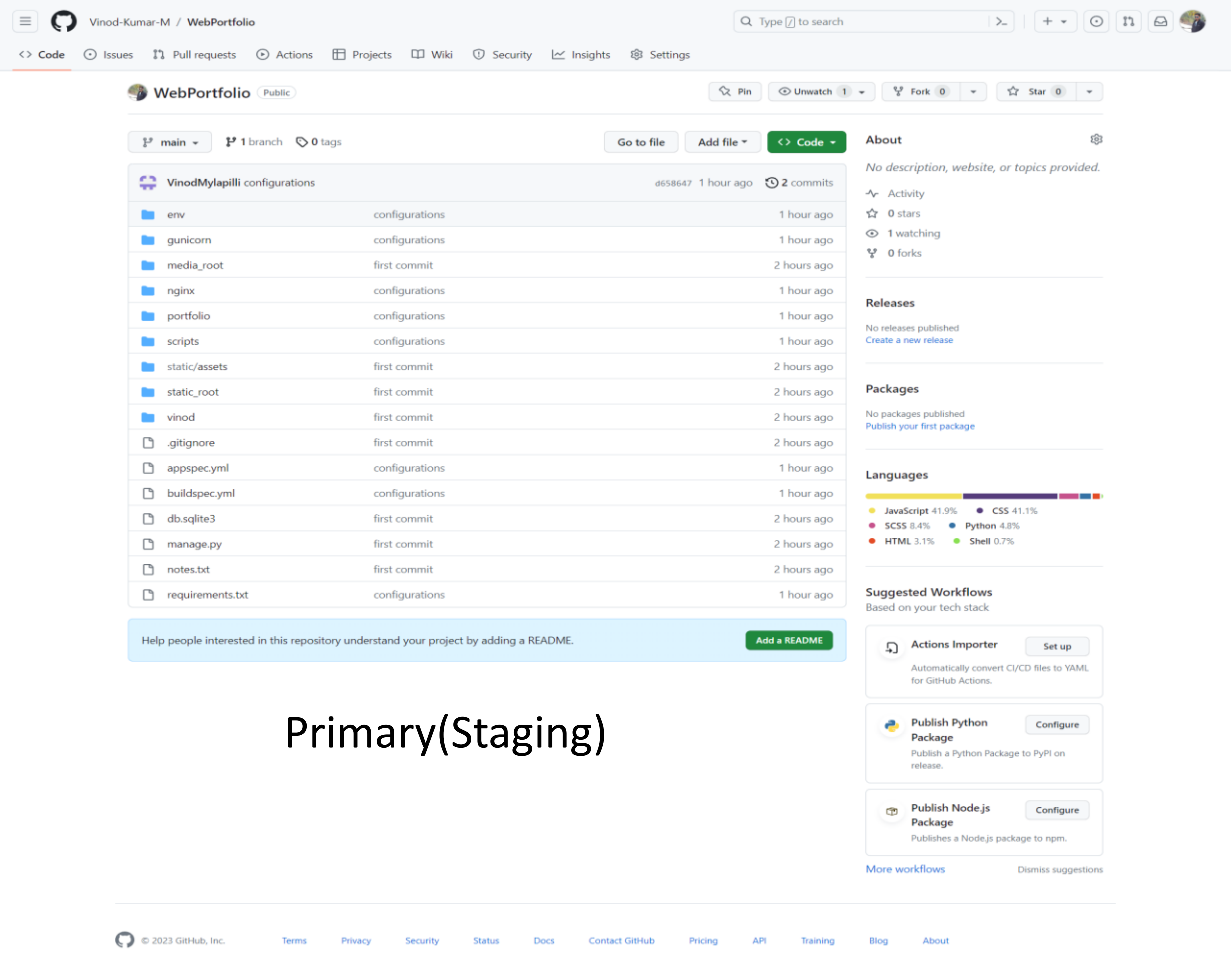
<u>By Using DNS Load Balancer</u>


Not secure | cicd-webportfolio-loadbalancer-877031765.us-east-1.elb.amazonaws.com
Vinod Kumar Mylapill
Home
About
Skills
Portfolio
Contact
HELLO THERE.
Scroll down

Now, Check the Automation Pipeline process working or not:

- First change the code in our project, what implementation want to shows.
- Then after push the code in GitHub, do the followings commands:

```
$ git remote add origin https://github.com/Vinod-Kumar-M/WebPortfolio.git
$ git branch –M main
$ git commit –m “Change the Home Greeting”
$ git push –u origin main
```



VinodMylapilli configurations d658647 1 hour ago 2 commits

env	configurations	1 hour ago
gunicorn	configurations	1 hour ago
media_root	first commit	2 hours ago
nginx	configurations	1 hour ago
portfolio	configurations	1 hour ago
scripts	configurations	1 hour ago
static/assets	first commit	2 hours ago
static_root	first commit	2 hours ago
vinod	first commit	2 hours ago
.gitignore	first commit	2 hours ago
appspec.yml	configurations	1 hour ago
buildspec.yml	configurations	1 hour ago
db.sqlite3	first commit	2 hours ago
manage.py	first commit	2 hours ago
notes.txt	first commit	2 hours ago
requirements.txt	configurations	1 hour ago

Help people interested in this repository understand your project by adding a README. Add a README

About

No description, website, or topics provided.

Activity
0 stars
1 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

JavaScript 41.9% CSS 41.1% SCSS 8.4% Python 4.8% HTML 3.1% Shell 0.7%

Suggested Workflows
Based on your tech stack

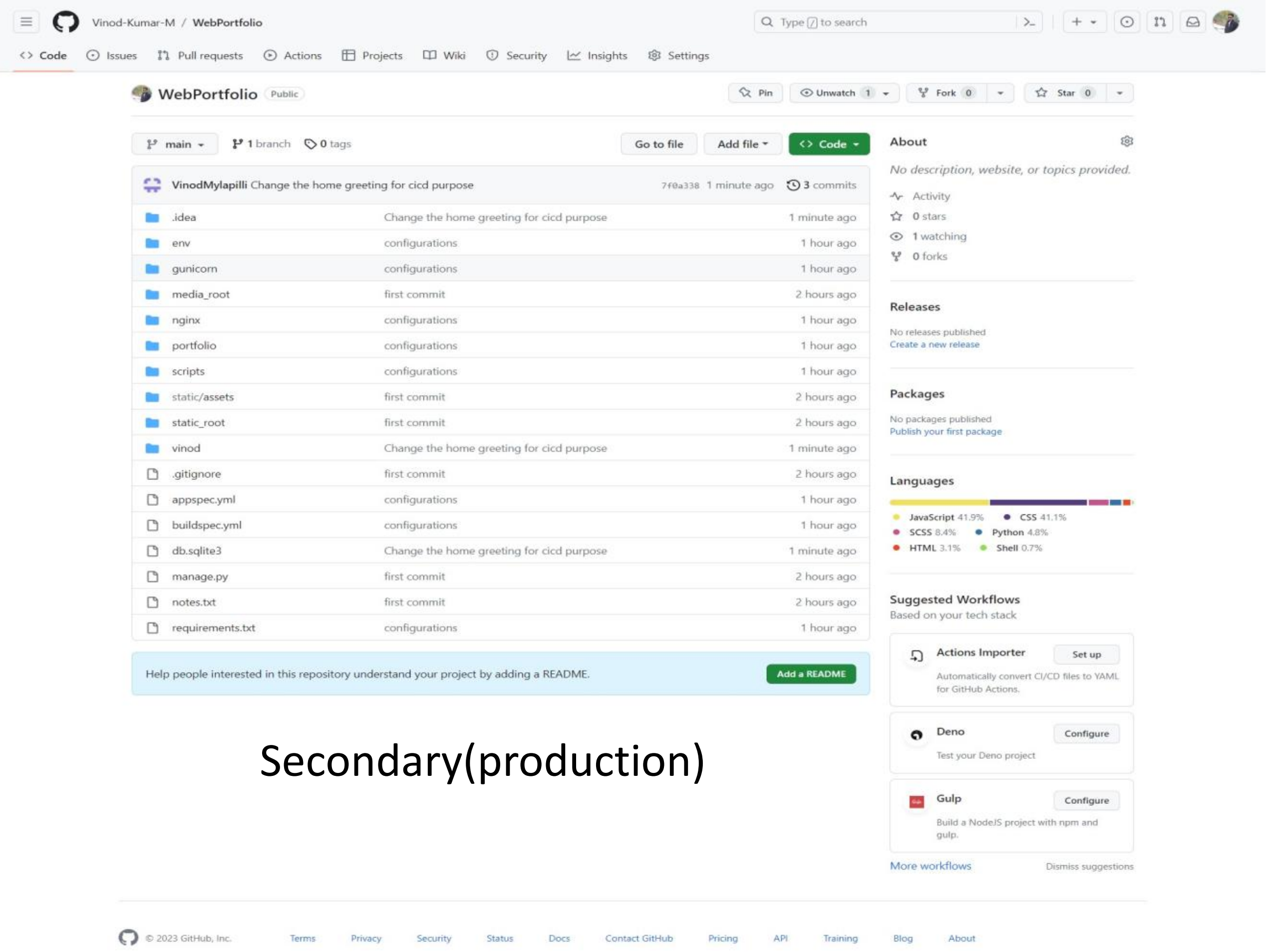
Actions Importer — Set up
Automatically convert CI/CD files to YAML for GitHub Actions.

Publish Python Package — Configure
Publish a Python Package to PyPI on release.

Publish Node.js Package — Configure
Publishes a Node.js package to npm.

More workflows Dismiss suggestions

Primary(Staging)



VinodMylapilli Change the home greeting for cicd purpose 7f0a338 1 minute ago 3 commits

.idea	Change the home greeting for cicd purpose	1 minute ago
env	configurations	1 hour ago
gunicorn	configurations	1 hour ago
media_root	first commit	2 hours ago
nginx	configurations	1 hour ago
portfolio	configurations	1 hour ago
scripts	configurations	1 hour ago
static/assets	first commit	2 hours ago
static_root	first commit	2 hours ago
vinod	Change the home greeting for cicd purpose	1 minute ago
.gitignore	first commit	2 hours ago
appspec.yml	configurations	1 hour ago
buildspec.yml	configurations	1 hour ago
db.sqlite3	Change the home greeting for cicd purpose	1 minute ago
manage.py	first commit	2 hours ago
notes.txt	first commit	2 hours ago
requirements.txt	configurations	1 hour ago

Help people interested in this repository understand your project by adding a README. Add a README

About

No description, website, or topics provided.

Activity
0 stars
1 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

JavaScript 41.9% CSS 41.1% SCSS 8.4% Python 4.8% HTML 3.1% Shell 0.7%

Suggested Workflows
Based on your tech stack

Actions Importer — Set up
Automatically convert CI/CD files to YAML for GitHub Actions.

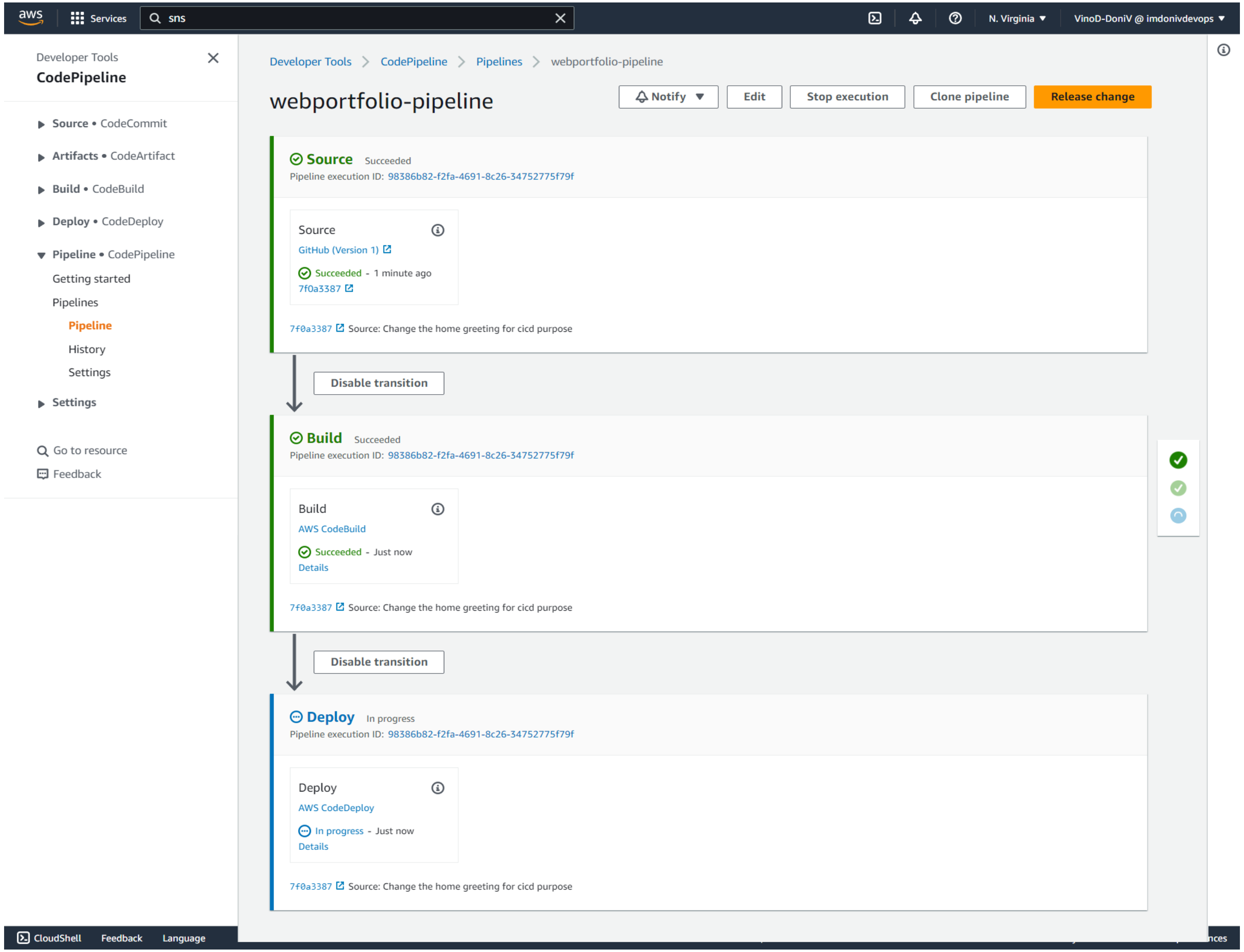
Deno — Configure
Test your Deno project

Gulp — Configure
Build a NodeJS project with npm and gulp.

More workflows Dismiss suggestions

Secondary(production)

- After Push the Code in GitHub, Automatically Pipeline process Starts. Shows below.
- See the Source: configurations ID changed here.
    - Primary (Staging) ID : d658647f
    - Secondary (production) ID : 7f0a3387



webportfolio-pipeline

Source Succeeded
Pipeline execution ID: 98386b82-f2fa-4691-8c26-34752775f79f

Source
GitHub (Version 1)
Succeeded - 1 minute ago
7f0a3387

7f0a3387 Source: Change the home greeting for cicd purpose

Build Succeeded
Pipeline execution ID: 98386b82-f2fa-4691-8c26-34752775f79f

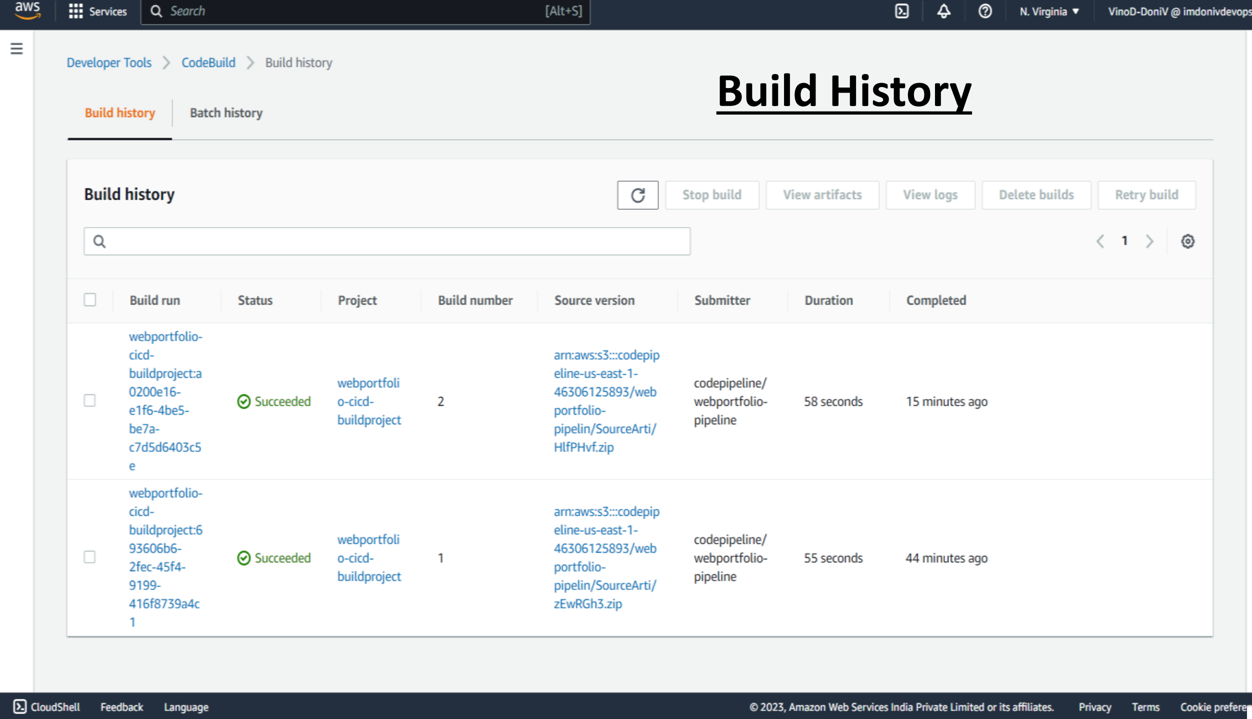
Build
AWS CodeBuild
Succeeded - Just now
Details

7f0a3387 Source: Change the home greeting for cicd purpose

Deploy In progress
Pipeline execution ID: 98386b82-f2fa-4691-8c26-34752775f79f

Deploy
AWS CodeDeploy
In progress - Just now
Details

7f0a3387 Source: Change the home greeting for cicd purpose

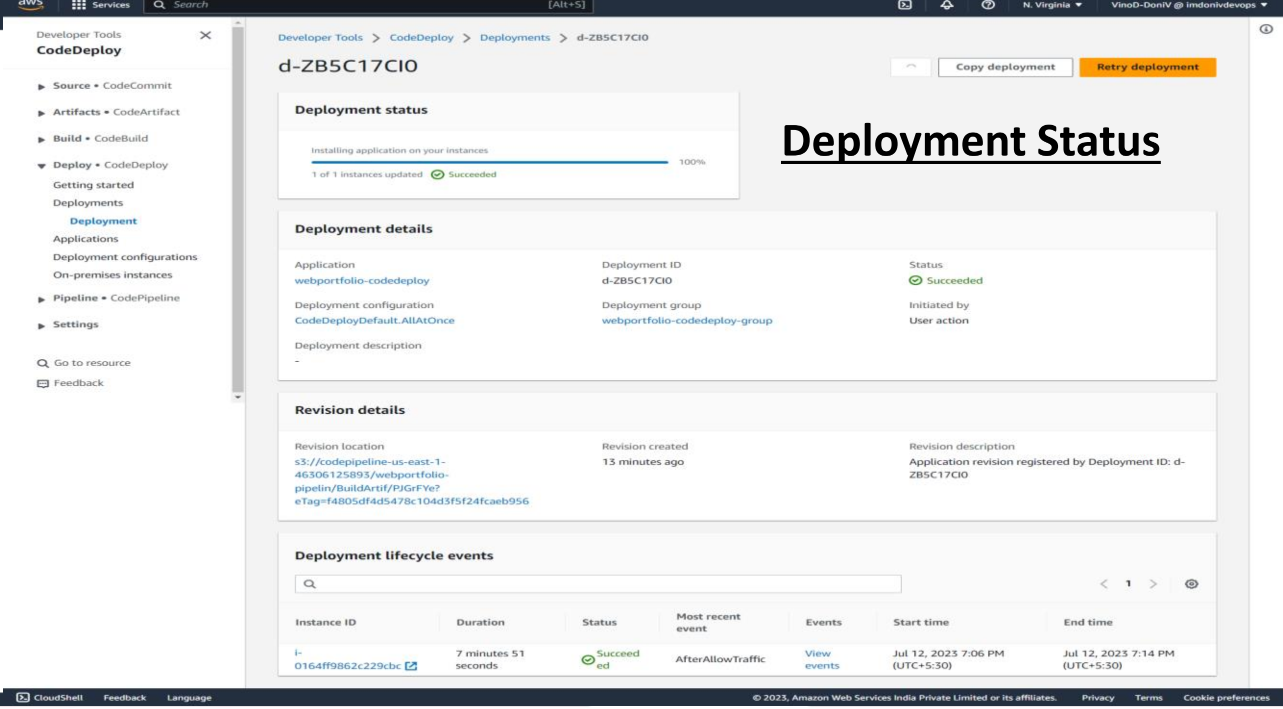


Build History

Developer Tools > CodeBuild > Build history

Build history

Build run	Status	Project	Build number	Source version	Submitter	Duration	Completed
webportfolio-cicd-buildproject:a0200e16-e1f6-4be5-be7a-c7d5d6403c5e	Succeeded	webportfolio-cicd-buildproject	2	arn:aws:s3:::codepipeline-us-east-1-46306125893/webportfolio-pipelin/SourceArti/HlfPHvf.zip	codepipeline/webportfolio-pipeline	58 seconds	15 minutes ago
webportfolio-cicd-buildproject:693606b6-2fec-45f4-9199-416f8739a4c1	Succeeded	webportfolio-cicd-buildproject	1	arn:aws:s3:::codepipeline-us-east-1-46306125893/webportfolio-pipelin/SourceArti/zEwRGh3.zip	codepipeline/webportfolio-pipeline	55 seconds	44 minutes ago



Deployment Status

Developer Tools > CodeDeploy > Deployments > d-ZB5C17CI0

d-ZB5C17CI0

Deployment status

Installing application on your instances 100%

1 of 1 instances updated Succeeded

Deployment details

Application: webportfolio-codedeploy

Deployment ID: d-ZB5C17CI0

Status: Succeeded

Deployment configuration: CodeDeployDefault.AllAtOnce

Deployment group: webportfolio-codedeploy-group

Initiated by: User action

Deployment description: -

Revision details

Revision location: s3://codepipeline-us-east-1-46306125893/webportfolio-pipelin/BuildArtif/PJGrFYe?eTag=f4805df4d5478c104d3f5f24fcaeb956

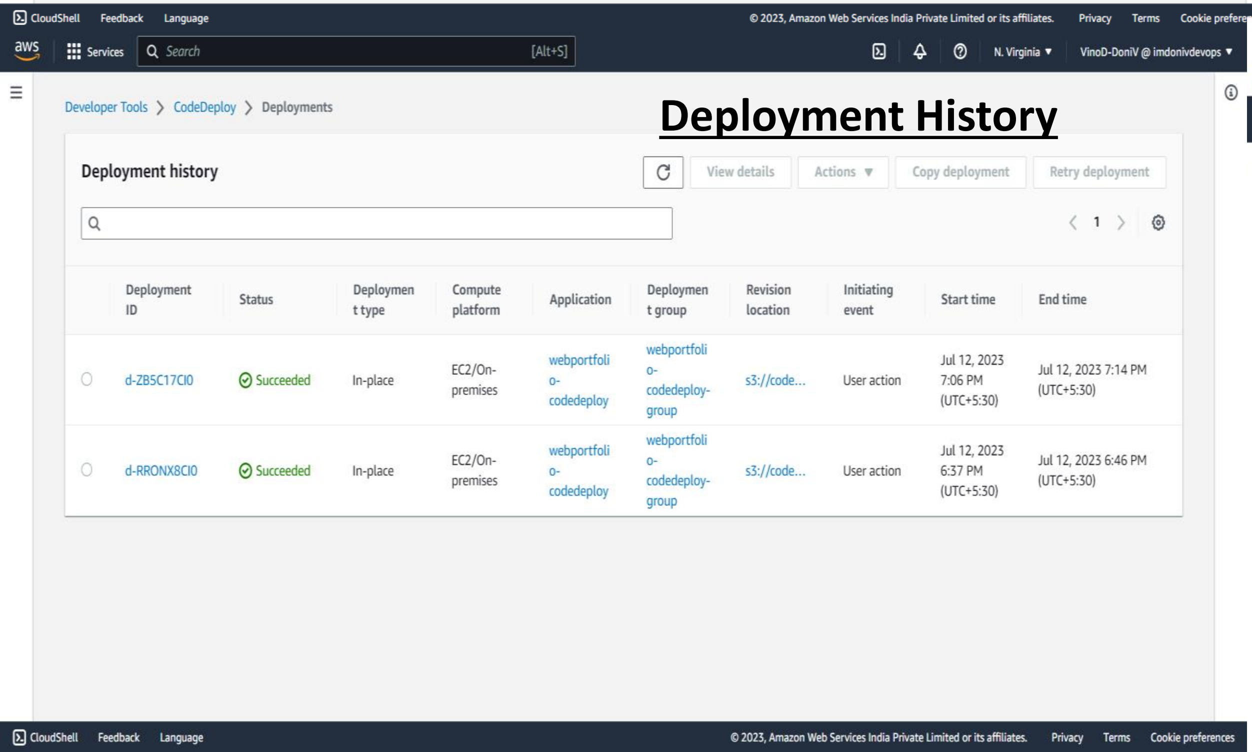
Revision created: 13 minutes ago

Revision description: Application revision registered by Deployment ID: d-ZB5C17CI0

Deployment lifecycle events

Instance ID	Duration	Status	Most recent event	Events	Start time	End time
i-0164ff9862c229cbc	7 minutes 51 seconds	Succeeded	AfterAllowTraffic	View events	Jul 12, 2023 7:06 PM (UTC+5:30)	Jul 12, 2023 7:14 PM (UTC+5:30)

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences



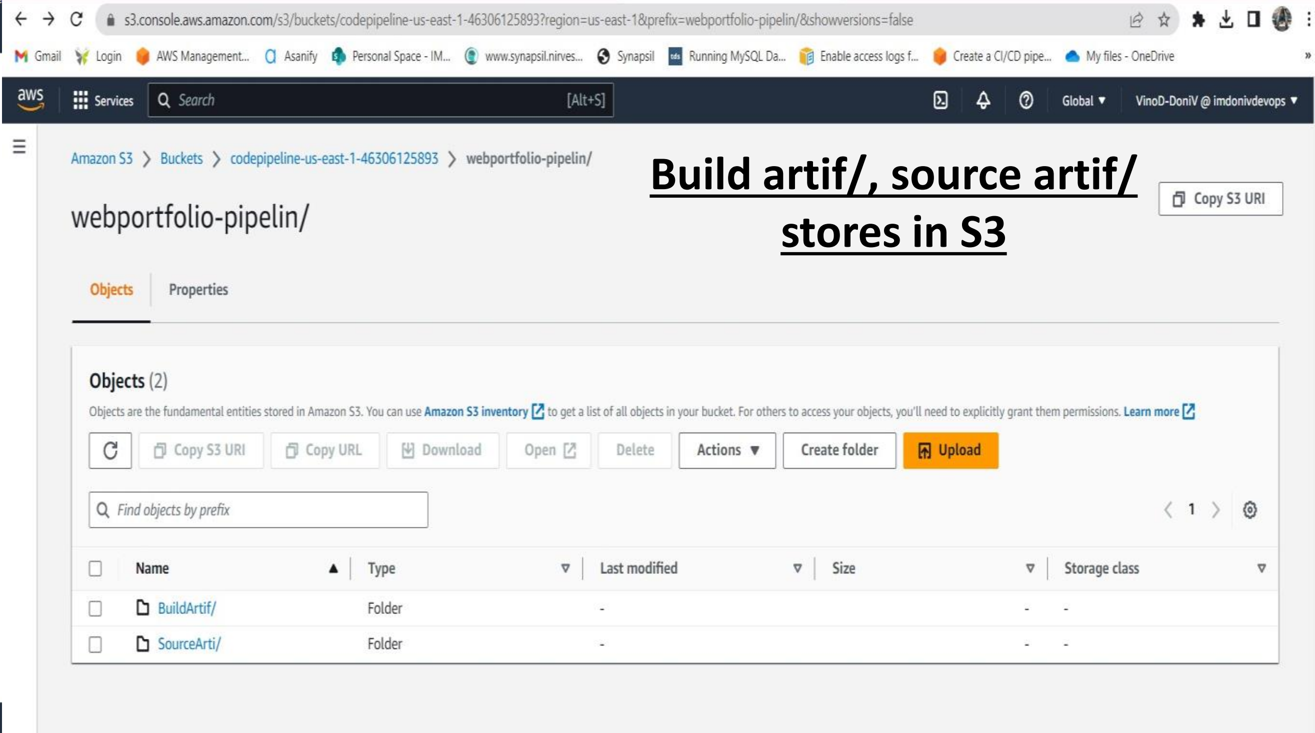
Deployment History

Developer Tools > CodeDeploy > Deployments

Deployment history

Deployment ID	Status	Deployment type	Compute platform	Application	Deployment group	Revision location	Initiating event	Start time	End time
d-ZB5C17CI0	Succeeded	In-place	EC2/On-premises	webportfolio-codedeploy	webportfolio-codedeploy-group	s3://code...	User action	Jul 12, 2023 7:06 PM (UTC+5:30)	Jul 12, 2023 7:14 PM (UTC+5:30)
d-RRONX8CI0	Succeeded	In-place	EC2/On-premises	webportfolio-codedeploy	webportfolio-codedeploy-group	s3://code...	User action	Jul 12, 2023 6:37 PM (UTC+5:30)	Jul 12, 2023 6:46 PM (UTC+5:30)

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences



Build artif/, source artif/ stores in S3

s3.console.aws.amazon.com/s3/buckets/codepipeline-us-east-1-46306125893?region=us-east-1&prefix=webportfolio-pipelin/&showversions=false

Amazon S3 > Buckets > codepipeline-us-east-1-46306125893 > webportfolio-pipelin/

webportfolio-pipelin/

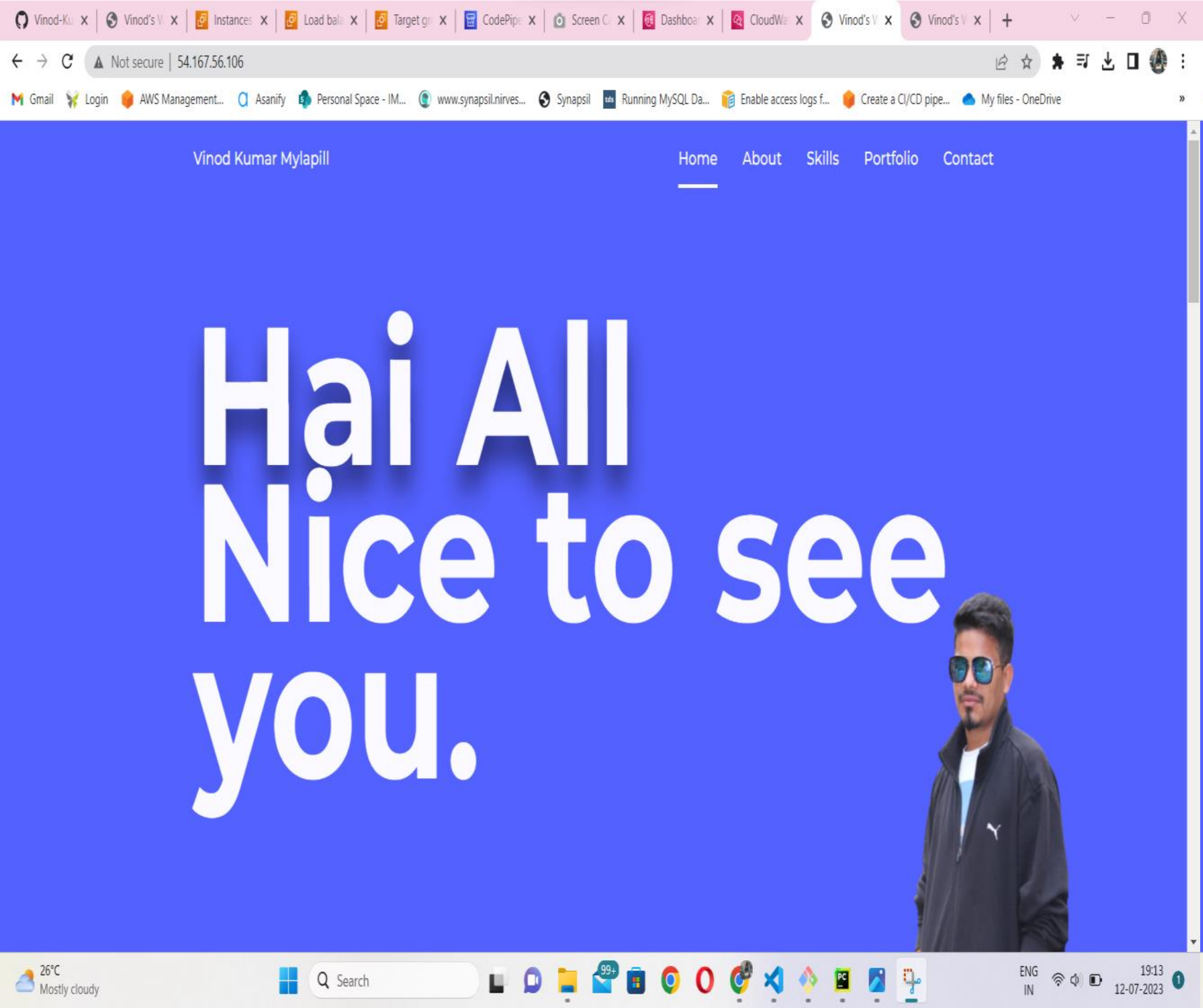
Objects (2)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Name	Type	Last modified	Size	Storage class
BuildArtif/	Folder	-	-	-
SourceArti/	Folder	-	-	-

The Result shows Here:

By using EC2 public IP (54.167.56.106)



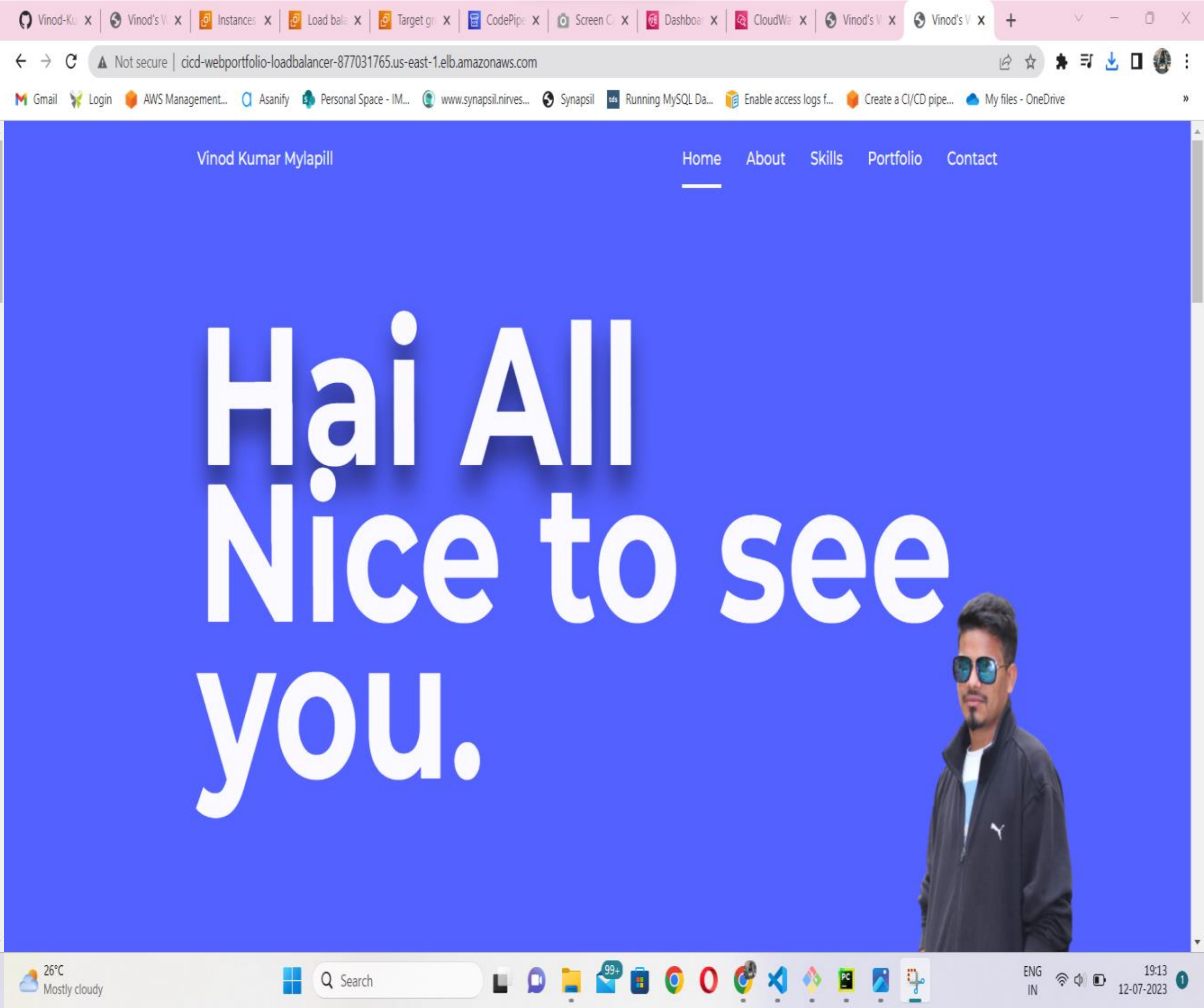
Vinod-Ku | Vinod's V | Instances | Load bala | Target gr | CodePipe | Screen C | Dashboar | CloudWa | Vinod's V | Vinod's V
Not secure | 54.167.56.106
Gmail | Login | AWS Management... | Asanify | Personal Space - IM... | www.synapsil.nirves... | Synapsil | Running MySQL Da... | Enable access logs f... | Create a CI/CD pipe... | My files - OneDrive
26°C Mostly cloudy
Search
ENG IN
19:13 12-07-2023

Vinod Kumar Mylapill

Home About Skills Portfolio Contact

Hai All
Nice to see
you.

By Using DNS Load Balancer



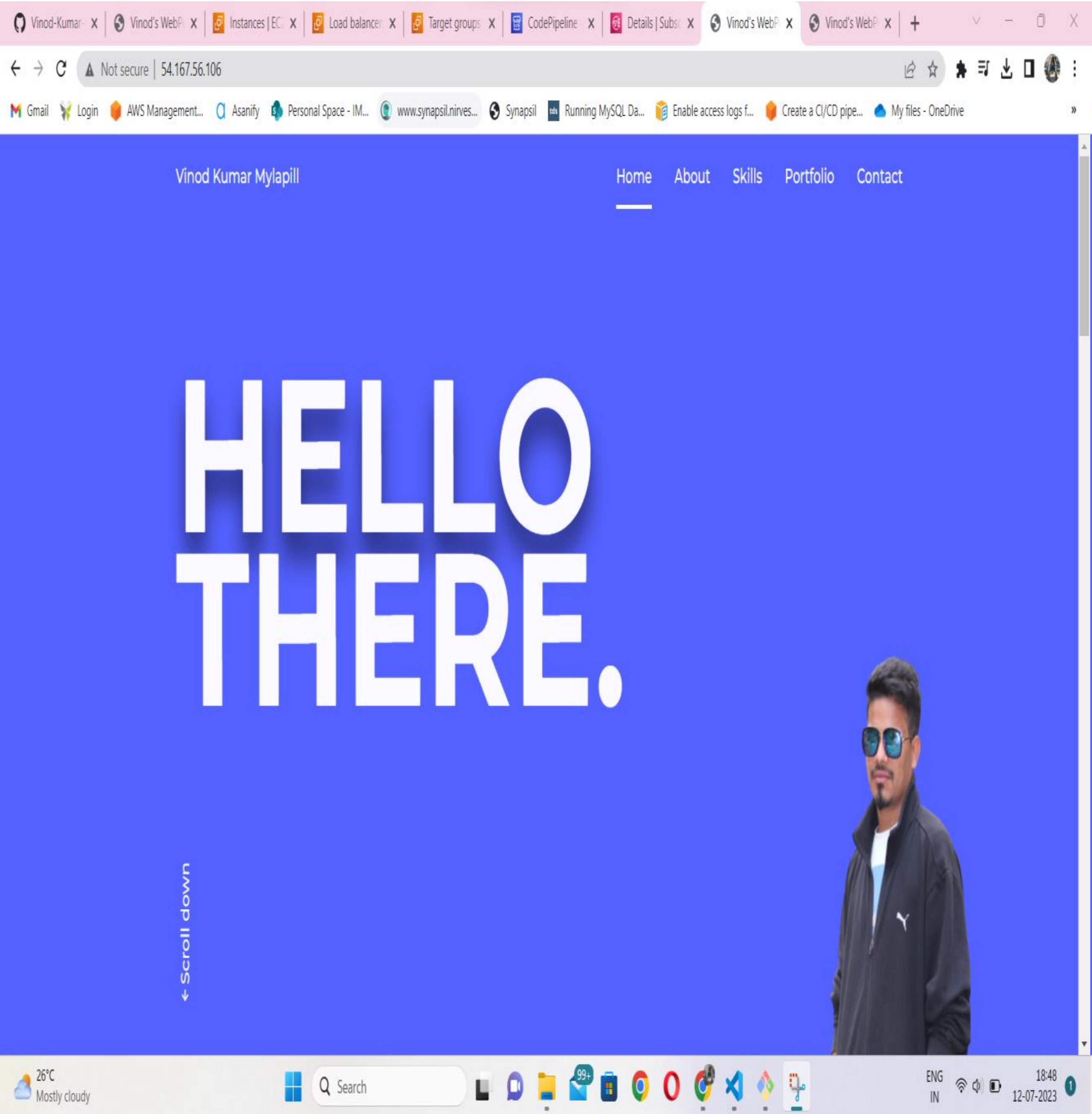
Vinod-Ku | Vinod's V | Instances | Load bala | Target gr | CodePipe | Screen C | Dashboar | CloudWa | Vinod's V | Vinod's V
Not secure | cicd-webportfolio-loadbalancer-877031765.us-east-1.elb.amazonaws.com
Gmail | Login | AWS Management... | Asanify | Personal Space - IM... | www.synapsil.nirves... | Synapsil | Running MySQL Da... | Enable access logs f... | Create a CI/CD pipe... | My files - OneDrive
26°C Mostly cloudy
Search
ENG IN
19:13 12-07-2023

Vinod Kumar Mylapill

Home About Skills Portfolio Contact

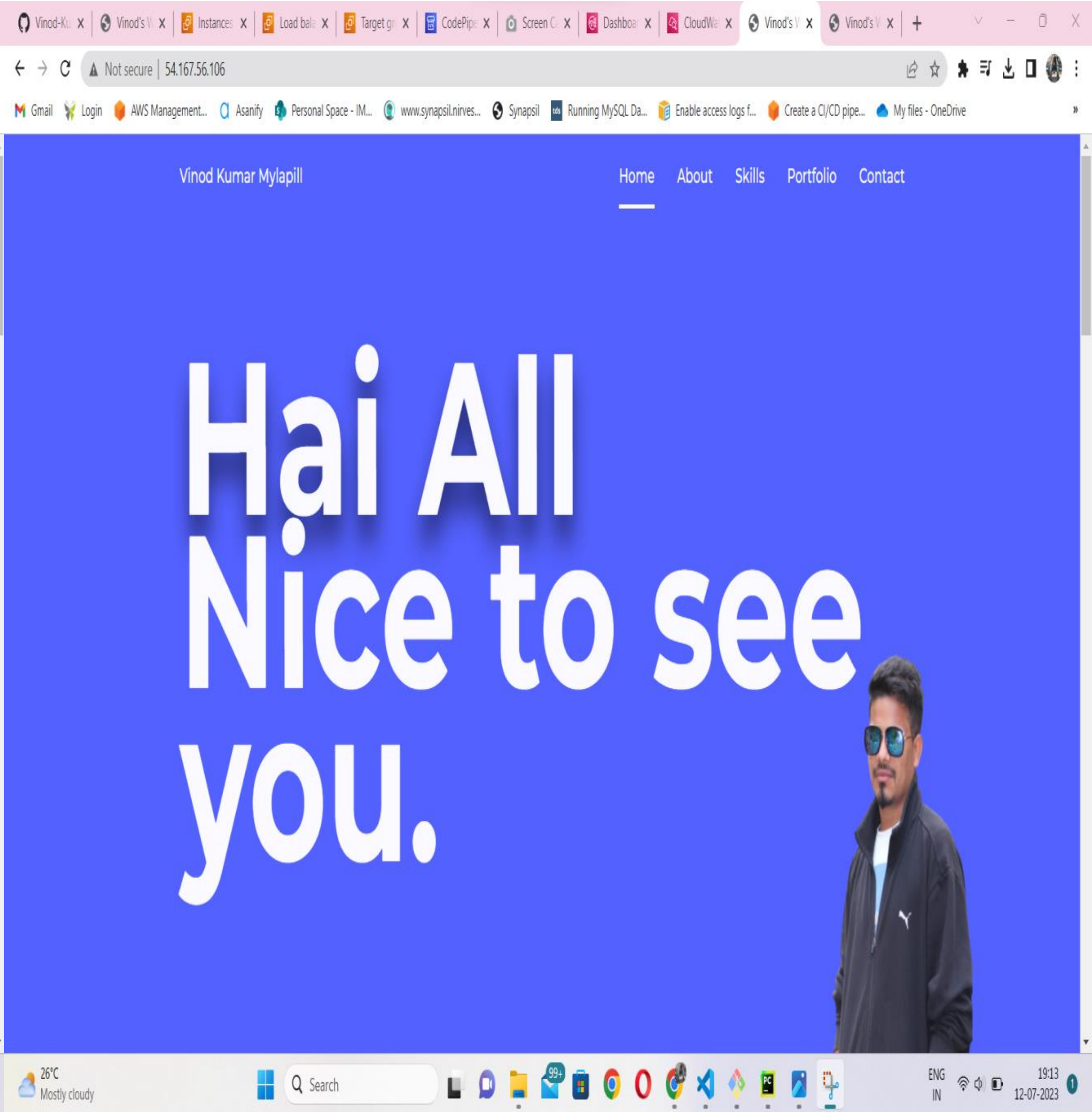
Hai All
Nice to see
you.

Primary (Staging)



HELLO THERE.

Secondary (Production)



Hai All Nice to see you.

Thank You !

See you soon.....