

NAME : UDDHAV BAPU GUND
TASK : JENKINS CI/CD PIPELINE

DESIGNATION: JR. AWS DEVELOPER

1.PROJECT OVERVIEW:

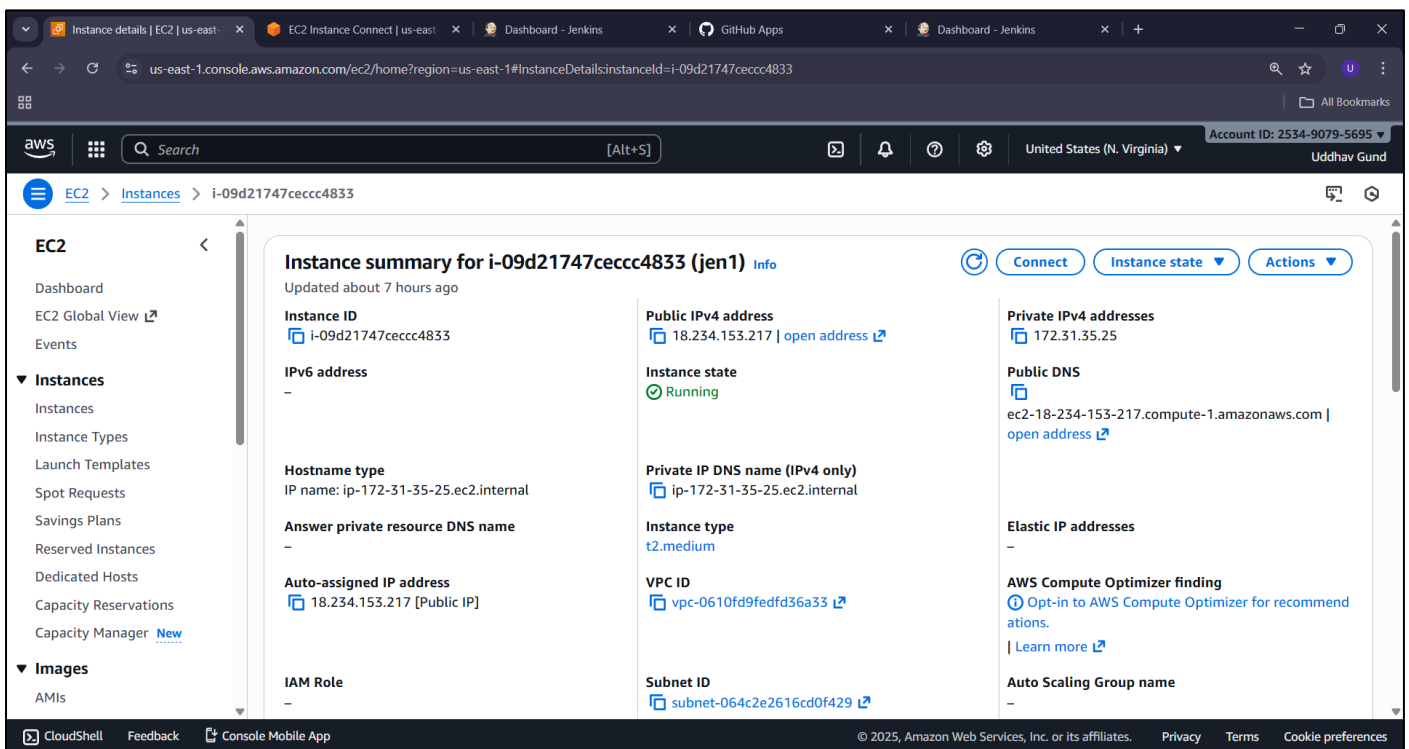
The project implements a complete **CI/CD pipeline** using **Github -> Jenkins -> Docker -> AWS EC2** for automated deployment of a Node.js Portfolio application.

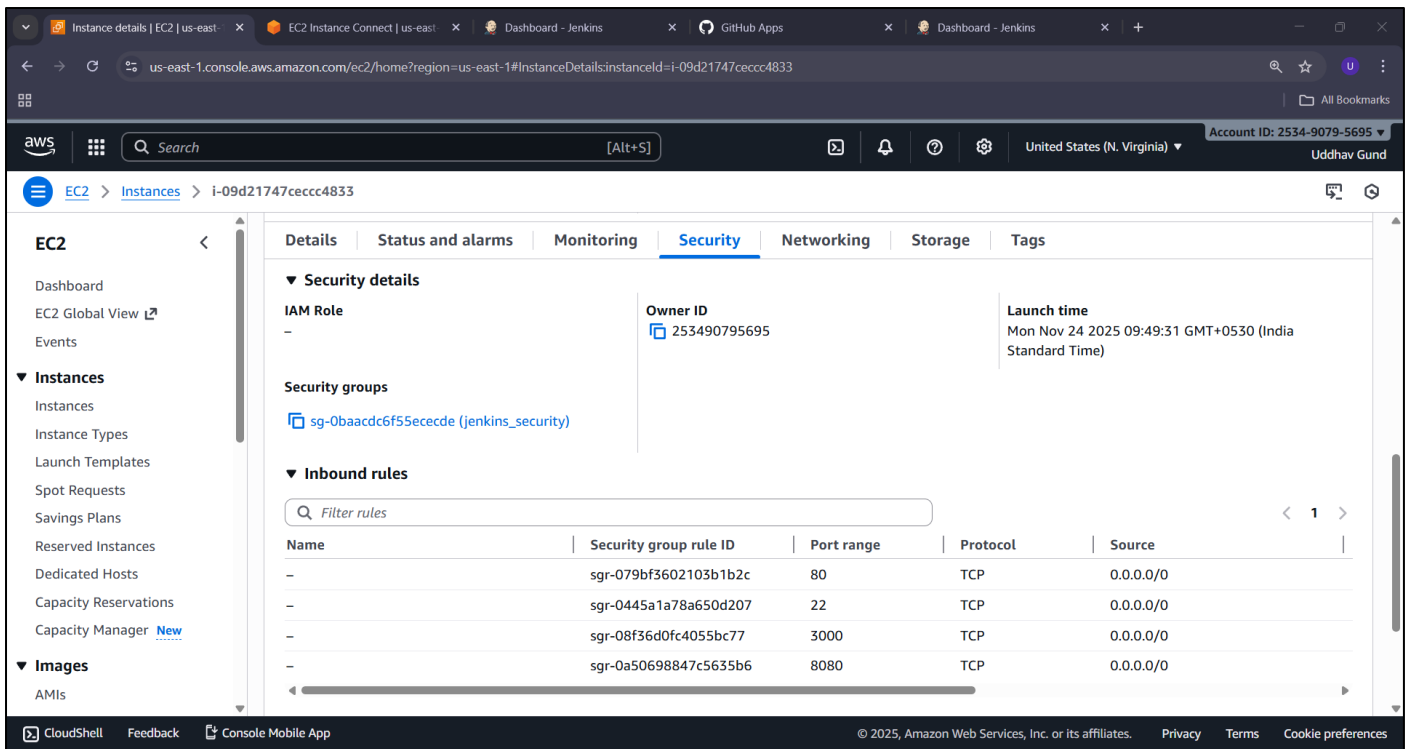
2. Tools & Technologies Used

- AWS EC2 (Amazon Linux 2023)
- Jenkins for CI/CD automation
- GitHub for source code management
- Docker for containerization
- Node.js & npm for runtime
- GitHub Webhook for auto-triggering builds

3. EC2 Instance Setup

1. Launched Amazon Linux 2023 EC2 instance.
2. Opened security group ports:
22 (SSH), 8080 (Jenkins), 3000 (App), 80 (optional).





3. Updated system packages:

```
sudo dnf update -y
sudo yum update -y
```

4. Installed Jenkins

1. Added Jenkins repository & GPG key.
2. Installed Jenkins(also java for Jenkins):

```
sudo dnf install jenkins -y
```

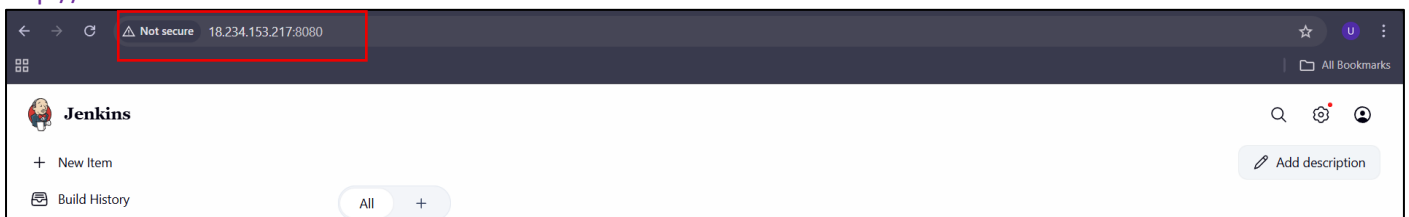
3. Started service:

```
sudo systemctl start jenkins
sudo systemctl enable jenkins
```

```
sudo dnf install java-17-amazon-corretto -y
sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
sudo dnf install jenkins -y
sudo systemctl enable jenkins
```

4. Access Jenkins at:

<http://<EC2-IP>:8080>



5. Installed Docker

1. Installed Docker on EC2:

```
sudo dnf install docker -y
```

2. Enabled and started Docker:

```
sudo systemctl start docker
```

```
sudo systemctl enable docker
```

3. Added Jenkins to Docker group:

```
sudo usermod -aG docker jenkins
```

```
sudo dnf install docker -y
sudo systemctl enable docker
sudo systemctl start docker
sudo usermod -aG docker ec2-user
sudo usermod -aG docker jenkins
```

6. Installed Node.js

```
sudo dnf install nodejs -y
```

```
[ec2-user@ip-172-31-35-25 ~]$ sudo dnf install nodejs -y
Last metadata expiration check: 2:59:27 ago on Mon Nov 24 04:29:35 2025.
Dependencies resolved.
```

Package	Size	Architecture	Version	Repository
Installing:				
nodejs	13 M	x86_64	1:18.20.8-1.amzn2023.0.2	amazonlinux

```
[ec2-user@ip-172-31-35-25 ~]$ node -v
v18.20.8
[ec2-user@ip-172-31-35-25 ~]$ npm -v
10.8.2
```

7. Jenkins Plugin Installation

Installed required plugins:

Git

GitHub

Pipeline

Docker Pipeline

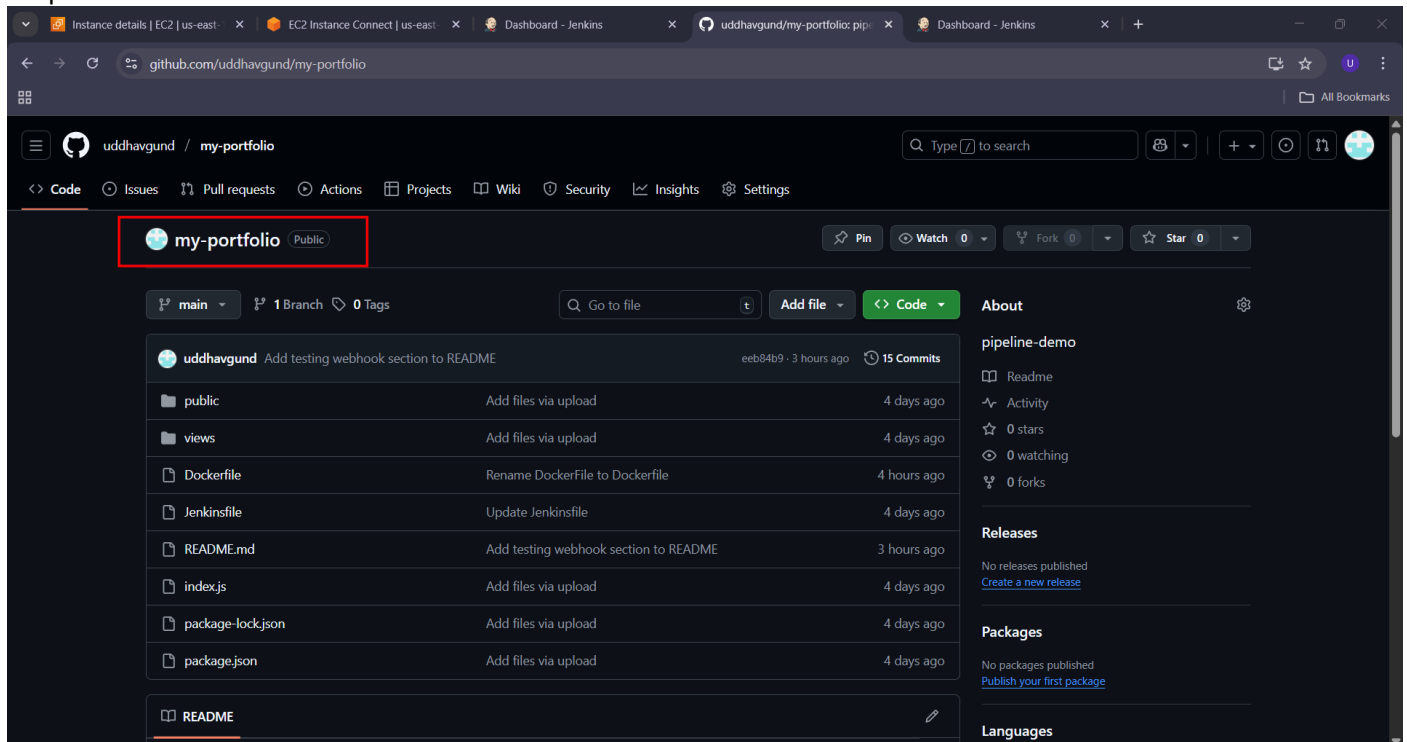
Credentials Binding

The screenshot shows the Jenkins web interface at the 'Manage Jenkins' page, specifically the 'Plugins' section. The page lists several installed plugins with their names, versions, health status, and whether they are enabled. The plugins listed are:

Name	Version	Health	Enabled
Ant Plugin	520.vd082ecfb_16a_9	96	Yes
Apache HttpComponents Client 4.x API Plugin	4.5.14-269.vfa_2321039a_83	96	Yes
Apache HttpComponents Client 5.x API Plugin	5.5-170.v023de017ccd7	100	Yes
ASM API Plugin	9.9-185.va_6c6b_3348b_c3	100	Yes
Authentication Tokens API Plugin	1.144.v5ff4a_5ec5c33	100	Yes
Bootstrap 5 API Plugin	5.3.8-895.v4d0d8e47fea_d	96	Yes

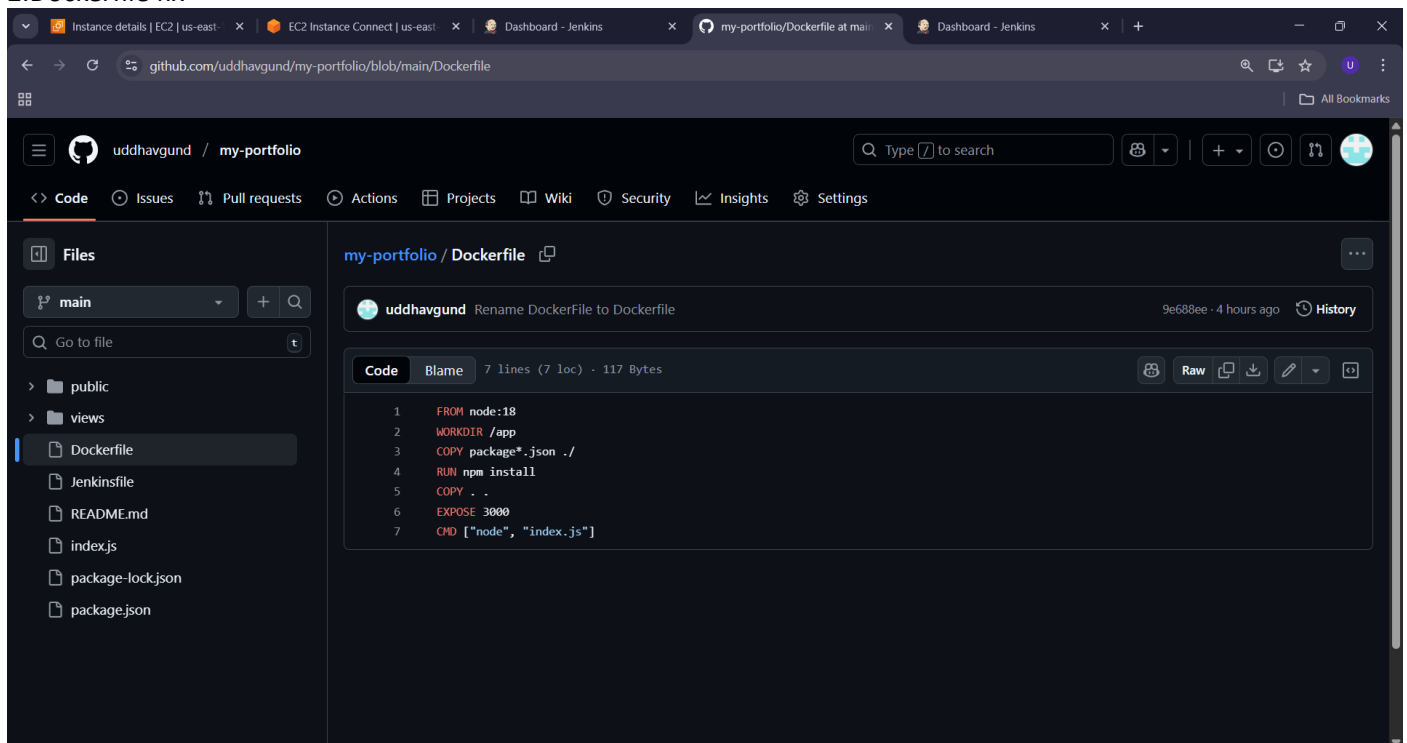
8. Uploading my-portfolio project on GitHub

1. Uploaded files



The screenshot shows the GitHub repository page for 'my-portfolio' by user 'uddhavgund'. The repository is public and has 15 commits. The file list includes: public, views, Dockerfile, Jenkinsfile, README.md, index.js, package-lock.json, and package.json. The README.md file is highlighted with a red box. The right sidebar shows the repository's activity, including a pipeline-demo, README, Activity, 0 stars, 0 watching, 0 forks, and a link to create a new release.

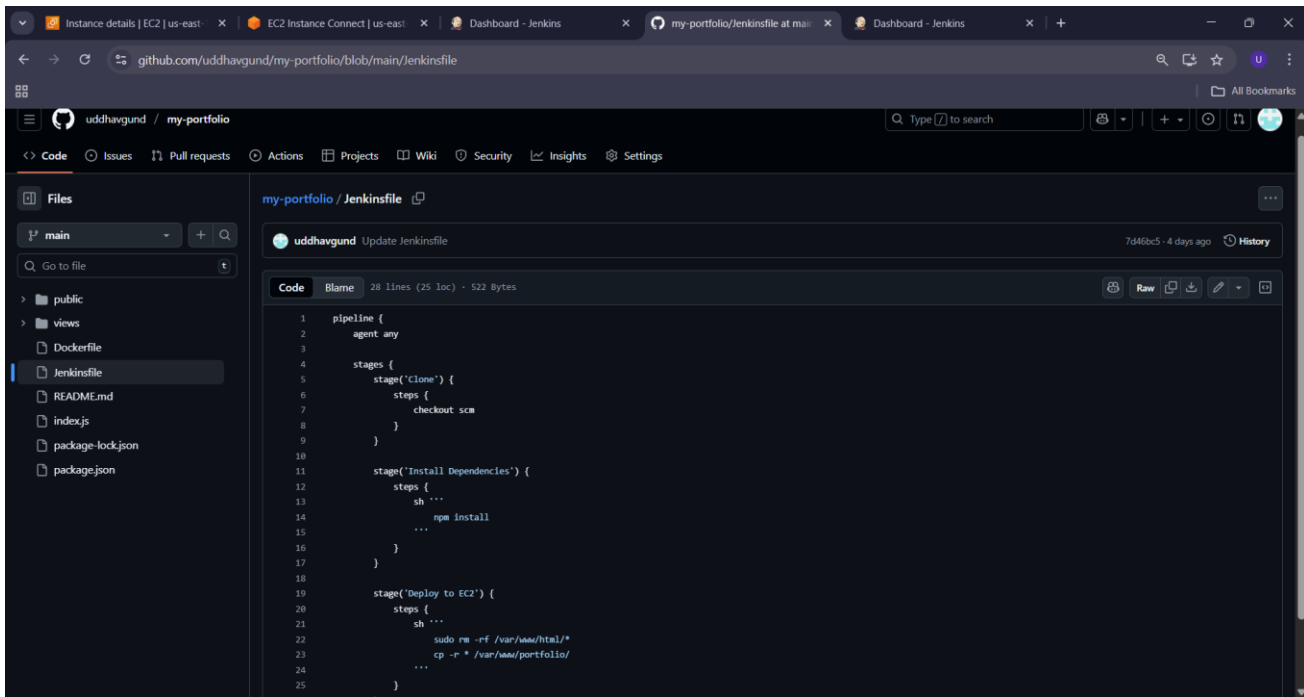
2. Dockerfile fix



The screenshot shows the GitHub repository page for 'my-portfolio' by user 'uddhavgund', specifically the Dockerfile file. The file is named 'Dockerfile' and is located in the 'my-portfolio' directory. The file content is as follows:

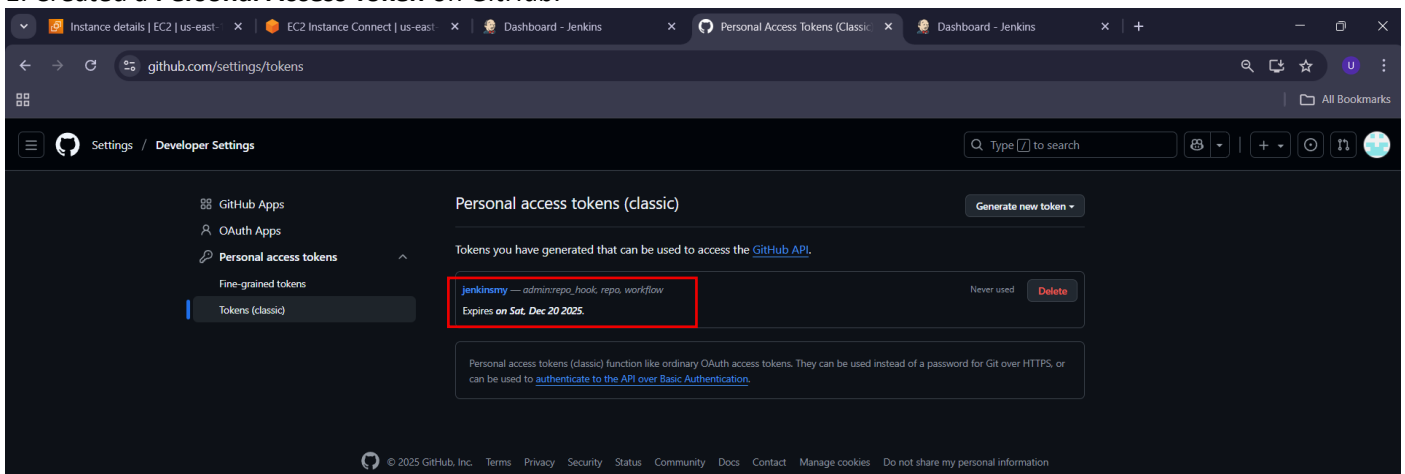
```
1 FROM node:18
2 WORKDIR /app
3 COPY package*.json ./
4 RUN npm install
5 COPY . .
6 EXPOSE 3000
7 CMD ["node", "index.js"]
```

3. Jenkinsfile fix



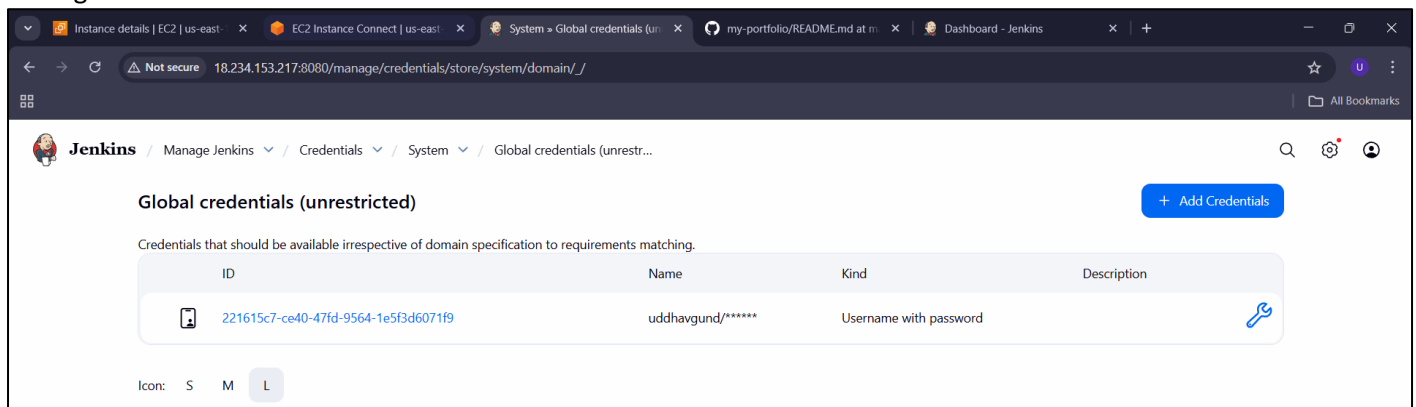
9. GitHub Credentials Setup

1. Created a Personal Access Token on GitHub.



2. Added it into Jenkins:

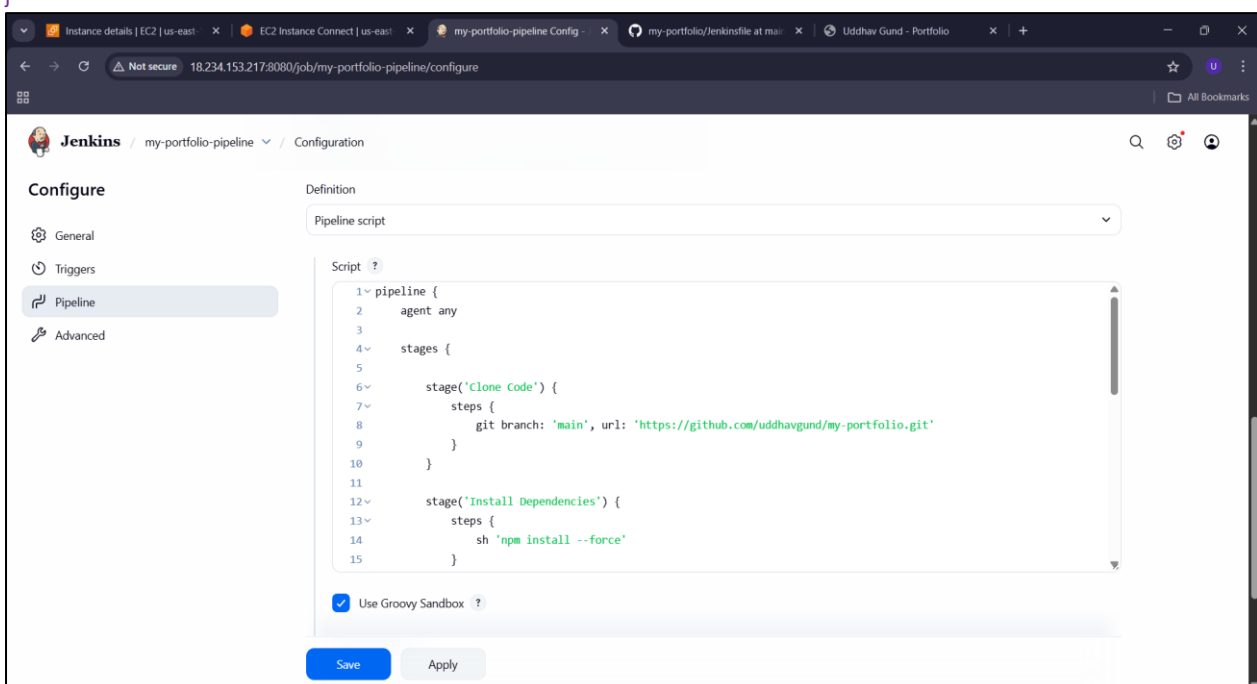
Manage Jenkins → Credentials → Global → Add Credentials

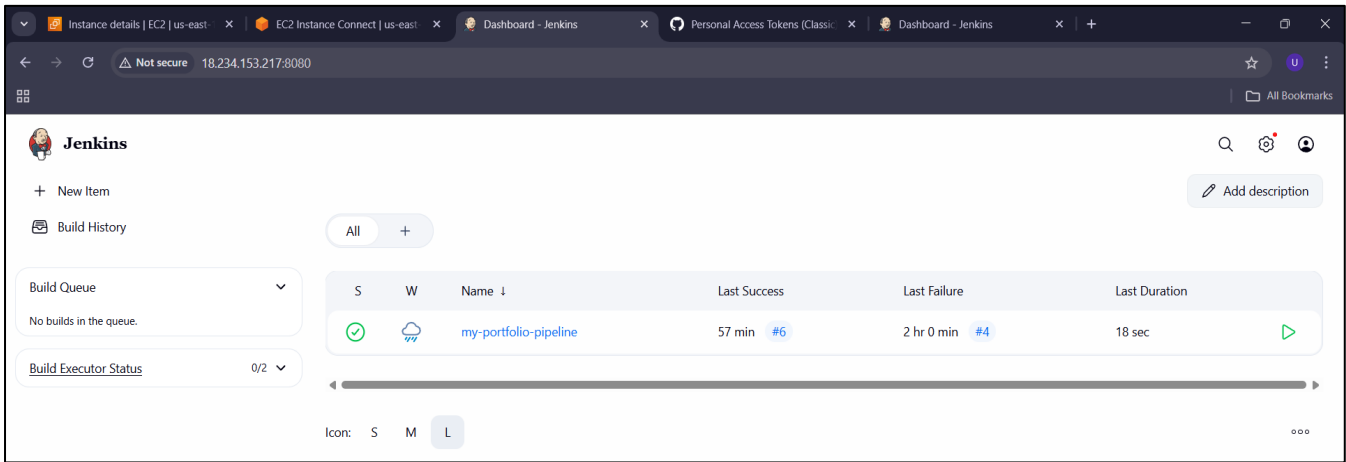


10. Jenkins Pipeline Job Setup

1. Created new Jenkins job: **my-portfolio-pipeline**
2. Selected Pipeline project type
3. Added pipeline script:

```
pipeline {
  agent any
  stages {
    stage('Clone Code') {
      steps {
        git branch: 'main', url: 'https://github.com/uddhavgund/my-portfolio.git'
      }
    }
    stage('Install Dependencies') {
      steps {
        sh 'npm install --force'
      }
    }
    stage('Docker Build') {
      steps {
        sh 'docker build -t my-portfolio .'
      }
    }
    stage('Stop Old Container') {
      steps {
        sh '''
        docker ps -q --filter "name=my-portfolio" | grep -q . && docker stop my-portfolio || true
        docker rm my-portfolio || true
        '''
      }
    }
    stage('Run New Container') {
      steps {
        sh 'docker run -d -p 3000:3000 --name my-portfolio my-portfolio'
      }
    }
  }
}
```





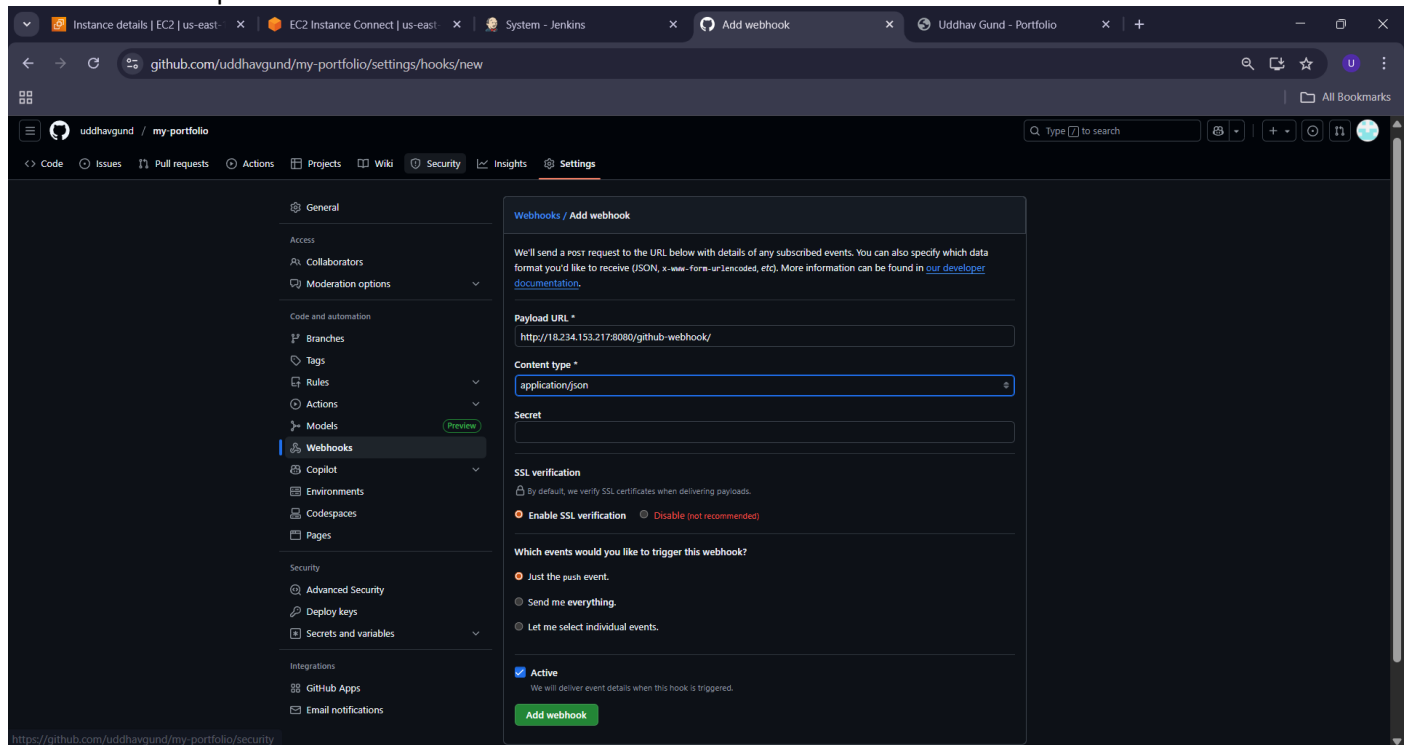
11. GitHub Webhook Setup (Auto Deployment)

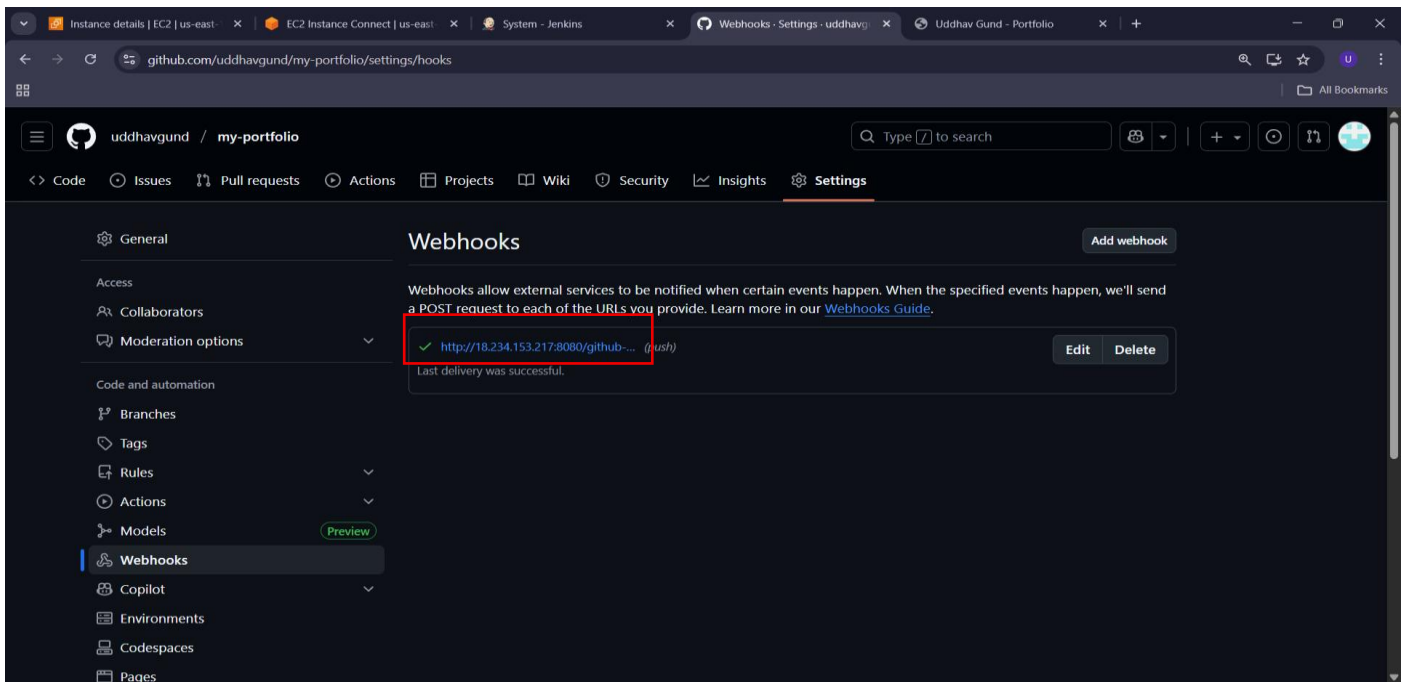
1. GitHub → Repo → Settings → Webhooks → Add Webhook

2. Payload URL:

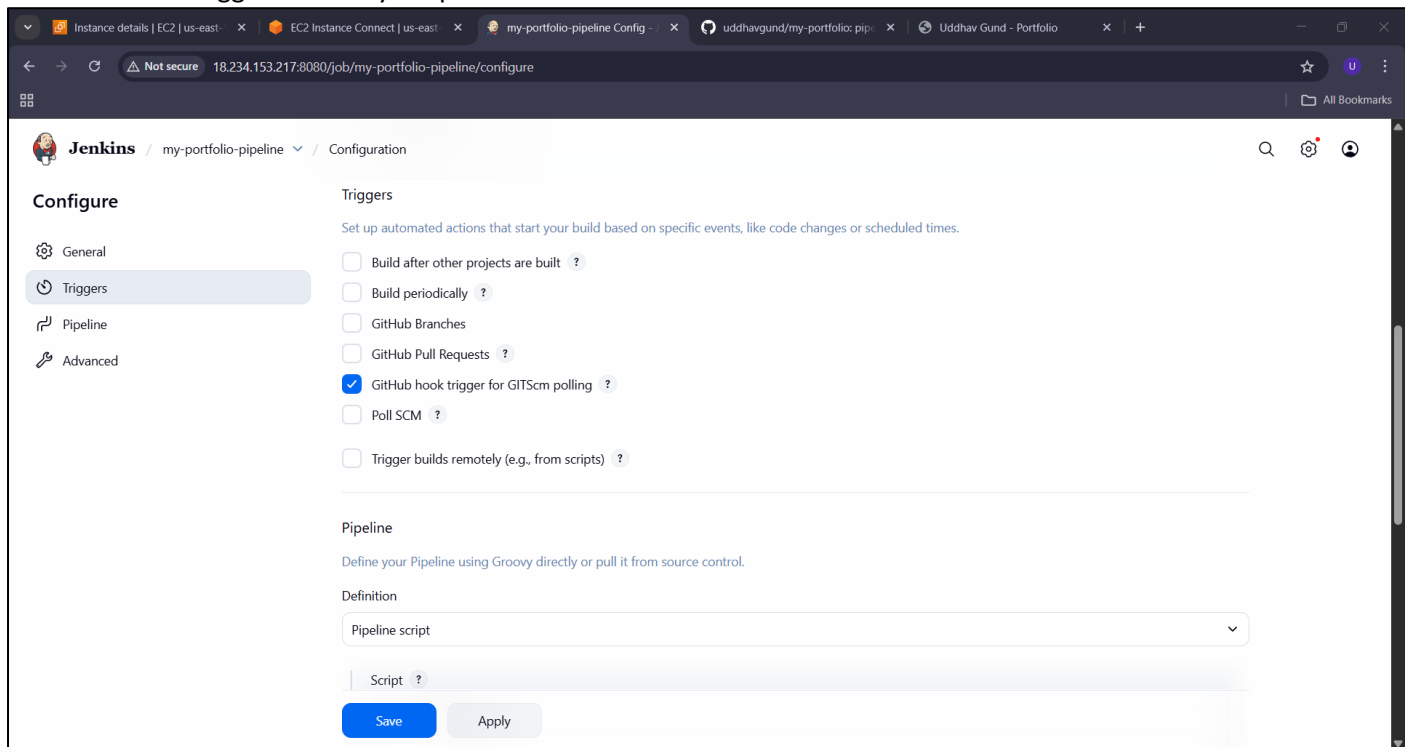
<http://<EC2-IP>:8080/github-webhook/>

3. Event: Just the push event





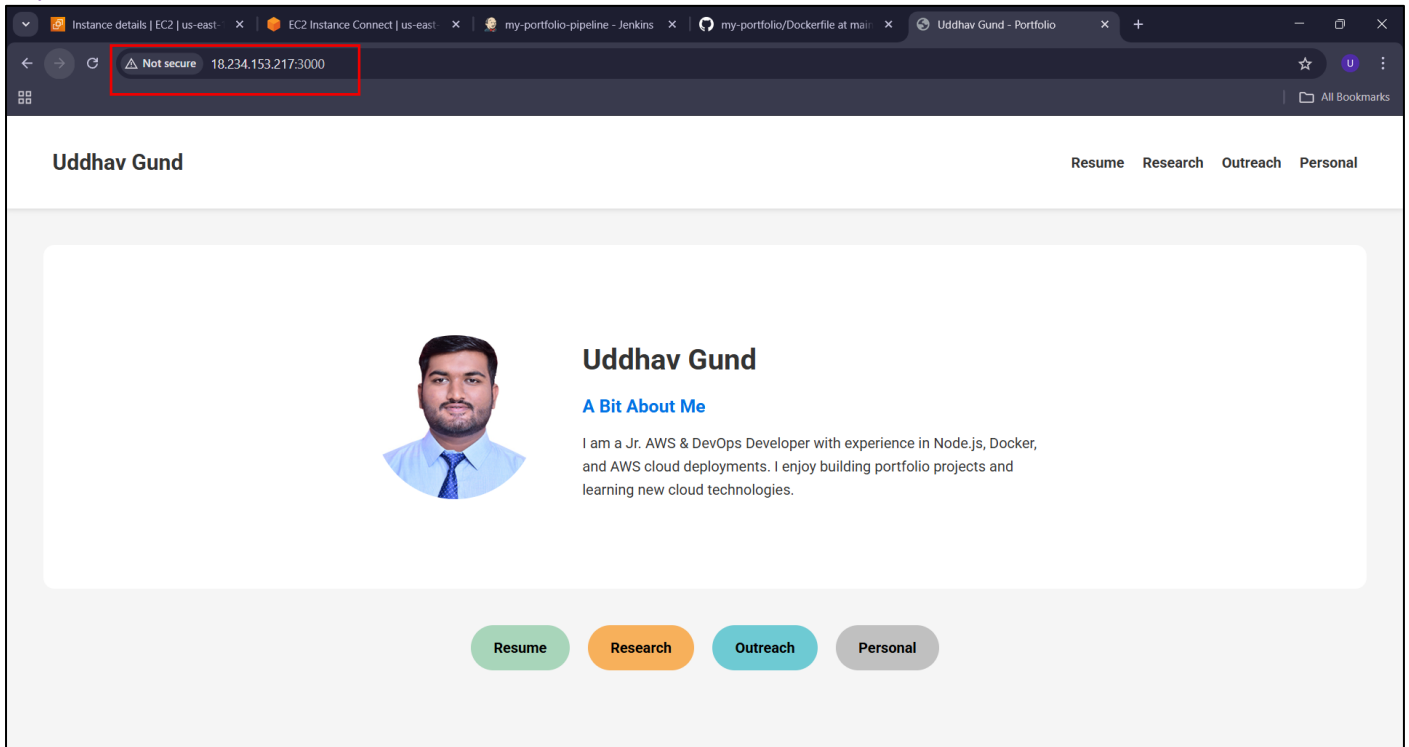
4. Jenkins auto-triggers on every Git push.



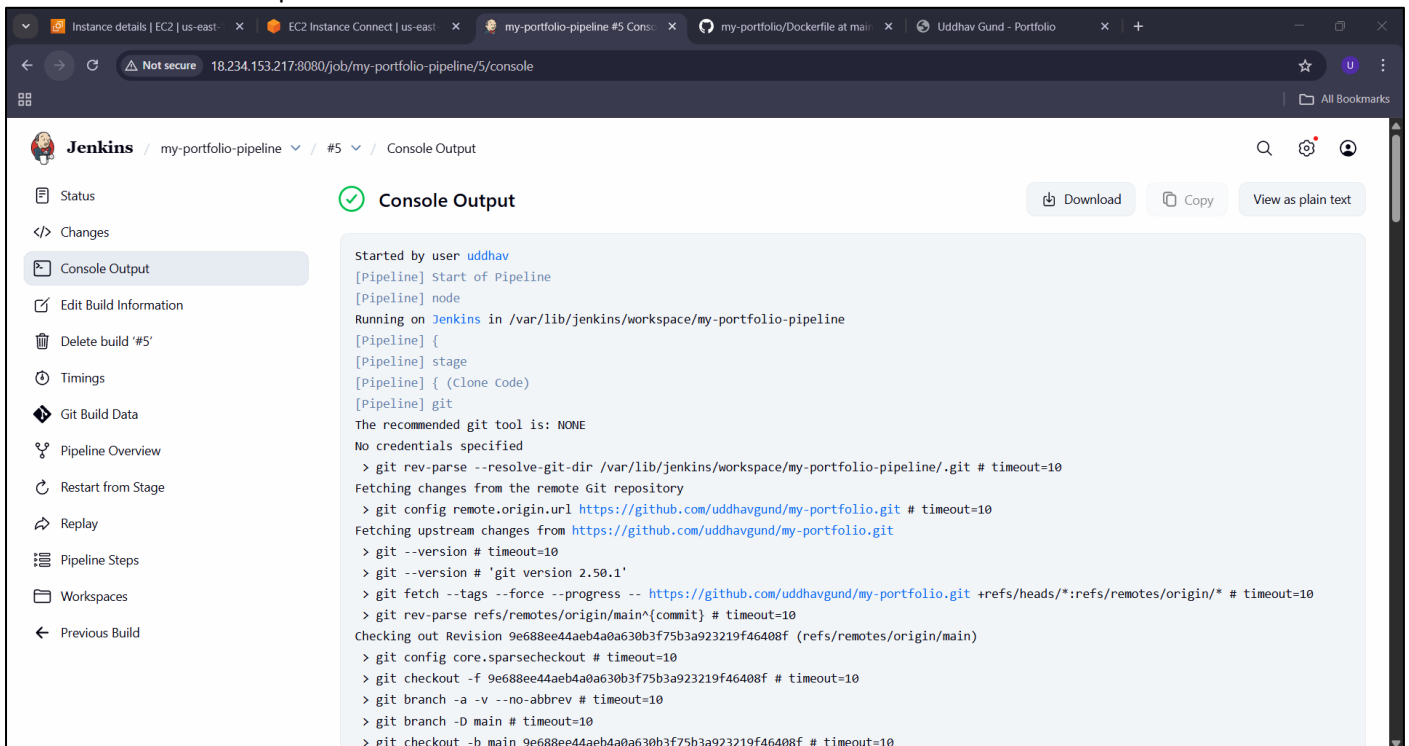
12. Final Deployment

After successful build, application runs on:

http://<EC2-IP>:3000



Successful Console Output



The screenshot shows the Jenkins dashboard with the 'my-portfolio-pipeline' build highlighted. The build is in a successful state, indicated by a green checkmark icon. The build details show it was completed 57 minutes ago, with a duration of 18 seconds. The build is triggered by a GitHub push event.

S	W	Name	Last Success	Last Failure	Last Duration
✓	☁	my-portfolio-pipeline	57 min #6	2 hr 0 min #4	18 sec

Auto-deployment of GitHub Push

The screenshot shows the 'Last GitHub Push' log for the 'my-portfolio-pipeline'. The log details the build process, including the event trigger, the strategy used, and the build steps. The build was successful, and the latest remote head revision was found.

```

Started on Nov 24, 2025, 8:35:53 AM
Started by event from 140.82.115.242 => http://18.234.153.217:8080/github-webhook/ on Mon Nov 24 08:35:53 UTC 2025
Using strategy: Default
[poll] Last Built Revision: Revision 9e688ee44aeb4a0a630b3f75b3a923219f46408f (refs/remotes/origin/main)
The recommended git tool is: NONE
No credentials specified
> git --version # timeout=10
> git --version # 'git version 2.50.1'
> git ls-remote -h -- https://github.com/uddhavgund/my-portfolio.git # timeout=10
Found 1 remote heads on https://github.com/uddhavgund/my-portfolio.git
[poll] Latest remote head revision on refs/heads/main is: eeb84b94ad9decc814a6c5b977d5c8534c532563
Done. Took 81 ms
Changes found
  
```

Docker images generated !

```

[ec2-user@ip-172-31-35-25 ~]$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
my-portfolio        latest             5ad711b3d40c       4 minutes ago      1.1GB
<none>              <none>            8d195c587983       About an hour ago  1.1GB
  
```

Docker Container Generated!

```

[ec2-user@ip-172-31-35-25 ~]$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED             STATUS             PORTS                               NAMES
830d9390bdc9  my-portfolio  "docker-entrypoint.s..." 5 minutes ago       Up 5 minutes      0.0.0.0:3000->3000/tcp, :::3000->3000/tcp  my-portfolio
  
```

Docker container automatically updates on every GitHub push.

!!!!!! Pipeline Completed Successfully !!!!!!!