

TASK 2 - Automating CI/CD Pipeline with Jenkins

NAME: SOWBARANIGA K

ROLL NO: 22CSR202

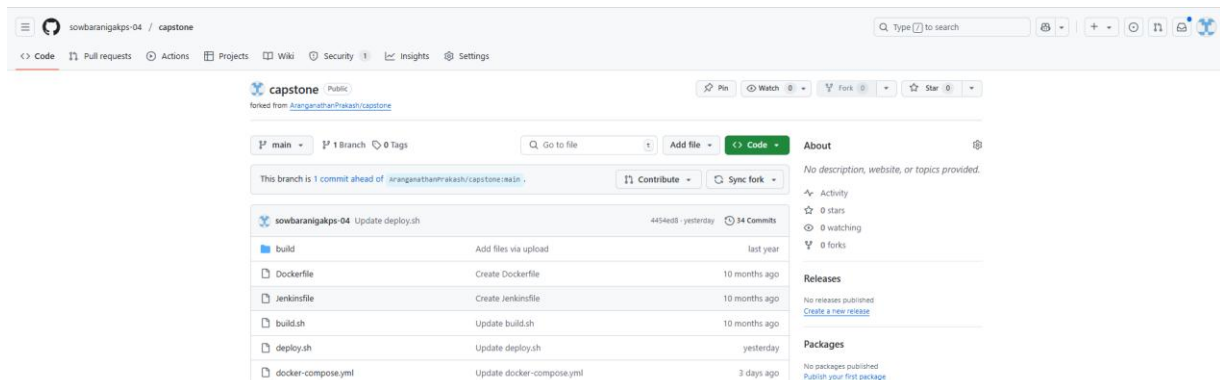
STEP 1: Installation of Docker

Install the Docker using the following commands:

- `sudo apt install docker.io`
- `docker --version`
- `sudo systemctl start docker`
- `sudo systemctl enable docker`
- `sudo systemctl status docker`

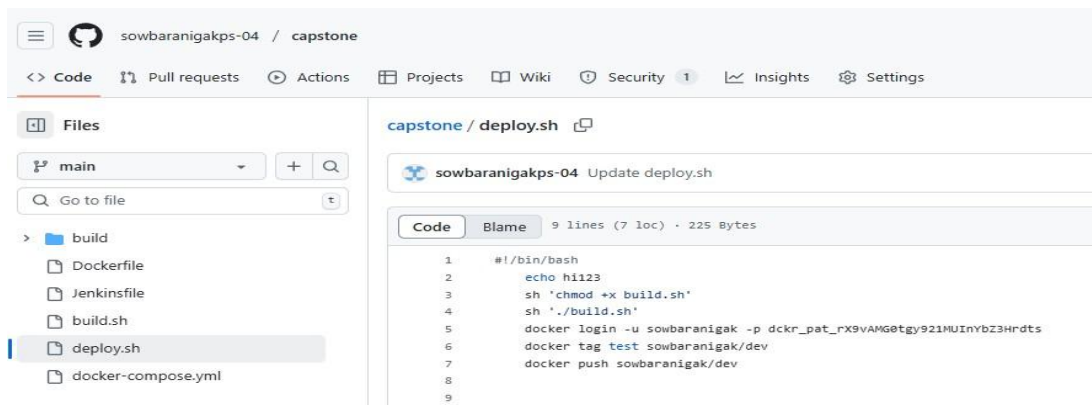
STEP 2: Fork a Github Repository

Fork a copy of a GitHub repository, which will create a clone of that repository in your own account.



STEP 3: Change the username and image name

In the `deploy.sh` file, update your Docker Hub username, Personal Access Token (PAT), and change the image name to the one you created in Docker.



STEP 3: Create a Pipeline Job in the Jenkins

- In Jenkins, create a job using a pipeline.
- In the 'Definition' section, choose 'Pipeline script from SCM'.
- Under SCM, select 'Git', then paste the GitHub repository link in the 'Repository URL' field, which contains the script files for your job.
- Change the branch to match the one you're using in Git, and verify the filename.
- Finally, click 'OK' to proceed.

New Item

Enter an item name

trail

Select an item type



Freestyle project

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



Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.



Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments,

OK

Dashboard > DevopsPipeline > Configuration

Configure

- General
- Triggers
- Pipeline
- Advanced

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

https://github.com/sowbaranigakps-04/capstone.git

Credentials ?

- none -

+ Add

Advanced

Add Repository

Branches to build ?

Save Apply

Dashboard > trail > Configuration

Configure

⚙️ General

🕒 Triggers

🔗 Pipeline

🔧 **Advanced**

Branch Specifier (blank for 'any') ?

*/main

Add Branch

Repository browser ?

(Auto)

Additional Behaviours

Add ▾

Script Path ?

Jenkinsfile

Save

Apply

STEP 4: Built Now

After creating the job, build it. The console output will be displayed, and the Docker image will be built and pushed to your Docker Hub repository.

Dashboard > DevOpsPipeline > #5

Status

Changes

Console Output

Edit Build Information

Delete build '#5'

Git Build Data

Restart from Stage

Replay

Pipeline Steps

Workspaces

Previous Build

Next Build

✓ Console Output

```

Started by user admin
Obtained Jenkinsfile from git https://github.com/soubaranigakps-04/capstone.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/DevOpsPipeline
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
Selected Git installation does not exist. Using Default
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/soubaranigakps-04/capstone.git
> git init /var/lib/jenkins/workspace/DevOpsPipeline # timeout=10
Fetching upstream changes from https://github.com/soubaranigakps-04/capstone.git
> git --version # timeout=10
> git --version # 'git version 2.43.0'
> git fetch --tags --force --progress -- https://github.com/soubaranigakps-04/capstone.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/soubaranigakps-04/capstone.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 4454ed86eeef9f558c5957eeac06cc1feea065 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 4454ed86eeef9f558c5957eeac06cc1feea065 # timeout=10
Commit message: "Update deploy.sh"
First time build. Skipping changelog.
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Build and Push Docker Image)
[Pipeline] sh
1287Fbecdfcc: Waiting
03d9365bc5dc: Layer already exists
d26dc06ef910: Layer already exists
d98dcc720ae0: Layer already exists
aa82c57cd9fe: Layer already exists
7920683f3d87: Layer already exists
135f786ad046: Layer already exists
ad2f08e39a9d: Layer already exists
1287Fbecdfcc: Layer already exists
latest: digest: sha256:471b701f4c71384c1990238823b0db750177477b87826034f57da3747c1d962f size: 1988
Creating network "task2_default" with the default driver
Creating task2_react-capstone_1 ...
Creating task2_react-capstone_1 ... done
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS

```

Dashboard > task2 > #7

1287Fbecdfcc: Waiting

03d9365bc5dc: Layer already exists

d26dc06ef910: Layer already exists

d98dcc720ae0: Layer already exists

aa82c57cd9fe: Layer already exists

7920683f3d87: Layer already exists

135f786ad046: Layer already exists

ad2f08e39a9d: Layer already exists

1287Fbecdfcc: Layer already exists

latest: digest: sha256:471b701f4c71384c1990238823b0db750177477b87826034f57da3747c1d962f size: 1988

Creating network "task2_default" with the default driver

Creating task2_react-capstone_1 ...

Creating task2_react-capstone_1 ... done

[Pipeline] }

[Pipeline] // stage

[Pipeline] }

[Pipeline] // withEnv

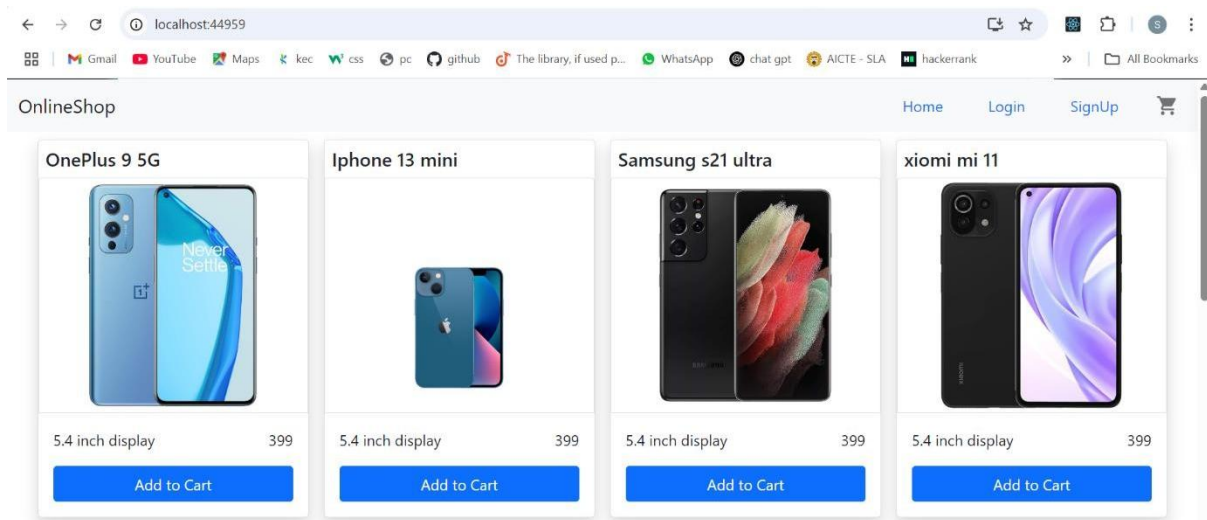
[Pipeline] }

[Pipeline] // node

[Pipeline] End of Pipeline

Finished: SUCCESS

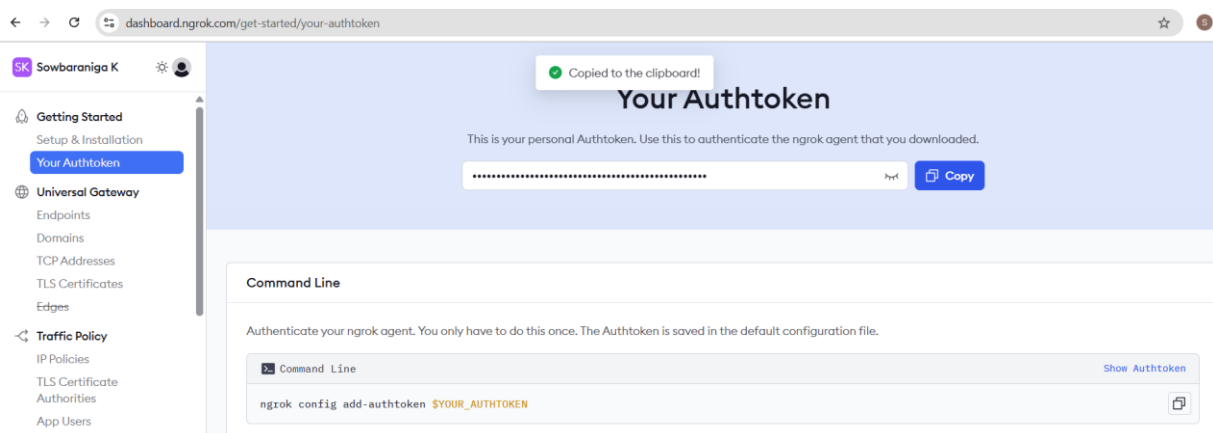
STEP 5: View the output in the browser



Setting Up a GitHub Webhook for Automatic Project Builds in Jenkins:

STEPS:

- To create a webhook in GitHub, install ngrok using the following command :
sudo snap install ngrok
- Create an account on the official website, and obtain the authentication or configuration command.
- After running the command in the terminal, you will receive the webhook URL for the GitHub repository.
- Add this webhook URL to your GitHub repository for automatic builds of the project.
- Additionally, in the Jenkins configuration page, check the box labelled GitHub hook trigger for GITScm polling.



```
Version      3.21.0
Region      India (in)
Latency     118ms
Web Interface http://127.0.0.1:4040
Forwarding   https://7d0e-103-196-28-176.ngrok-free.app -> http://localhost:8080

Connections  ttl    opn    rt1    rt5    p50    p90
              2      0      0.00   0.00   30.31   30.56

HTTP Requests
-----
12:15:33.571 UTC POST /github-webhook/      200 OK
12:09:49.879 UTC POST /github-webhook/      200 OK
```

[Code](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [3](#) [Insights](#) [Settings](#)

General

Access

Collaborators

Moderation options

Code and automation

Branches

Webhooks

Add webhook

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

✓ https://7d0e-103-196-28-176.ngrok... (push)

EditDelete

Last delivery was successful.

Dashboard > trail > Configuration

Configure

- General
- Triggers
- Pipeline
- Advanced

Triggers

Set up automated actions that start your build based on specific events, like code changes or scheduled times.

- ☐ Build after other projects are built ?
- ☐ Build periodically ?
- ☐ GitHub Branches
- ☐ GitHub Pull Requests ?
- ☒ GitHub hook trigger for GITScm polling ?
- ☐ Poll SCM ?
- ☐ Trigger builds remotely (e.g., from scripts) ?

Pipeline

Define your Pipeline using Groovy directly or pull it from source control.

Save

Apply