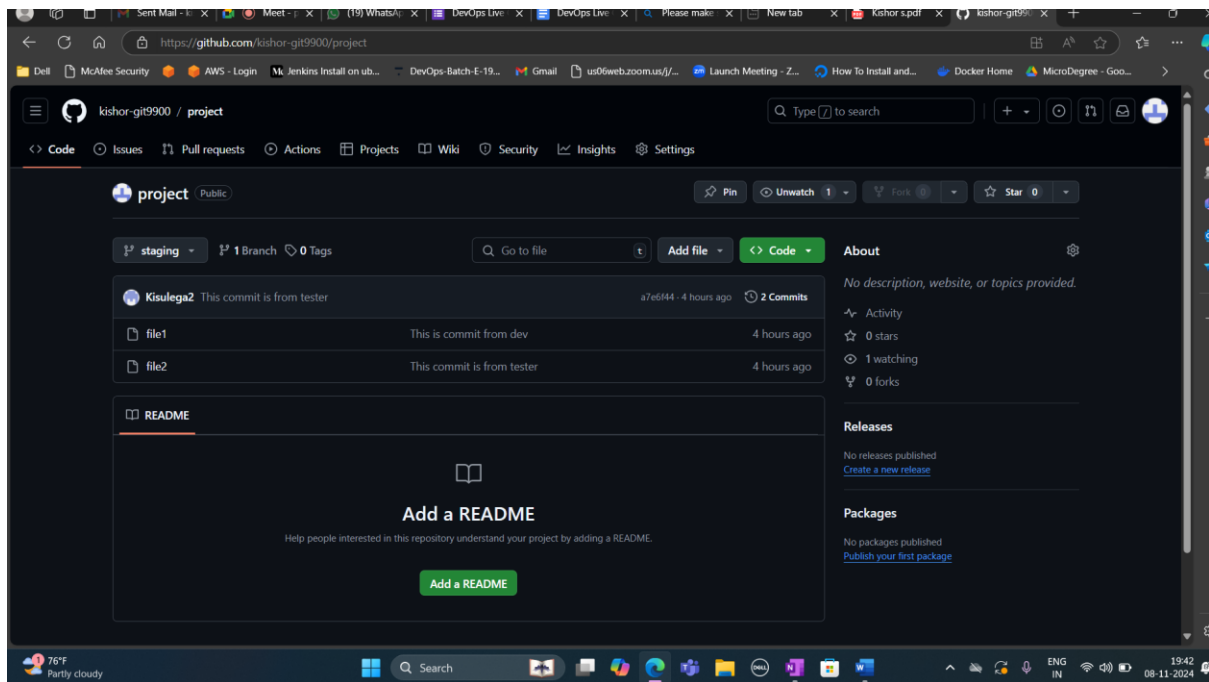


Task 1:

Created a Staging branch in GitHub, pushed the code from local repository to remote

```
EC2 S3 VPC EFS
37 git push origin staging
38 history
root@ip-172-31-25-241:~/project1# history>file3.pdf
root@ip-172-31-25-241:~/project1# ls
file1 file2 file3.pdf
root@ip-172-31-25-241:~/project1# cat file3
cat: file3: No such file or directory
root@ip-172-31-25-241:~/project1# cat file3.pdf
 1 git config --global user.name "kishor"
 2 git config --global user.email "kishor@gmail.com"
 3 git config --list
 4 git init
 5 git branch
 6 ls
 7 git branch Staging
 8 cat file1
 9 cat>file1
10 git status
11 mkdir project1
12 cd project1/
13 git init
14 ls
15 cat>file1
16 git status
17 git banch
18 git branch
19 git checkout -b staging
16 git status
17 git banch
18 git branch
19 git checkout -b staging
20 git branch
21 ls
22 git log
23 git add .
24 git commit -m "This is commit from dev"
25 git status
26 git log
27 git branch
28 git remote add origin https://github.com/kishor-git9900/project.git
29 git push origin staging
30 git checkout master
31 ls
32 cat>file2
33 git add .
34 git commit -m "This commit is from tester"
35 git log
36 git status
37 git push origin staging
38 history
39 history>file3.pdf
```



Task 2:

Create a sample Jenkins CICD pipeline and share output logs and pipeline script

-Pipeline script

```
pipeline {
  agent any
  stages {
    stage ("Code") {
      steps {
        echo "Cloning the Code"
        git url:"https://github.com/adhya2020/django-notes-app.git" , branch: "main"
      }
    }
    stage ("Build"){
      steps {
        echo "Build the image"
        sh "docker build -t my-note-app ."
      }
    }
    stage ("Pushing the image to Dockerhub"){
      steps {
        echo "pushing image to docker hub"
        withCredentials([usernamePassword(credentialsId:"dockerHub",passwordVariable:"dockerHubPass",usernameVariable:"dockerHubUser")]) {
          sh "docker tag my-note-app ${env.dockerHubUser}/my-note-app:latest"
          sh "docker login -u ${env.dockerHubUser} -p ${env.dockerHubPass}"
          sh "docker push ${env.dockerHubUser}/my-note-app:latest"
        }
      }
    }
    stage ("Deploy"){
      steps {
```

```

echo "Deploying the container"
sh "docker run -d -p 8000:8000 kishorss/my-note-app:latest"
}
}
}
}

```

Console output logs

```

Started by user kishor ss
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/project69
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Code)
[Pipeline] echo
Cloning the Code
[Pipeline] git
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/adhya2020/django-notes-app.git
> git init /var/lib/jenkins/workspace/project69 # timeout=10
Fetching upstream changes from https://github.com/adhya2020/django-notes-app.git
> git --version # timeout=10
> git --version # 'git version 2.43.0'
> git fetch --tags --force --progress -- https://github.com/adhya2020/django-notes-app.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/adhya2020/django-notes-app.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 79a271958caa4243f4df714153eed5323393bace
(refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 79a271958caa4243f4df714153eed5323393bace # timeout=10
> git branch -a -v --no-abbrev # timeout=10
> git checkout -b main 79a271958caa4243f4df714153eed5323393bace # timeout=10
Commit message: "Update index.html"
First time build. Skipping changelog.
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] echo
Build the image
[Pipeline] sh
+ docker build -t my-note-app .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/
Sending build context to Docker daemon 8.396MB
Step 1/7 : FROM python:3.9
3.9: Pulling from library/python
7d98d813d54f: Pulling fs layer
da802df85c96: Pulling fs layer

```

7aad5092c3b: Pulling fs layer
ad1c7cfc347f: Pulling fs layer
4eb48115a042: Pulling fs layer
ccecc6c1c4bf: Pulling fs layer
63dc518f902b: Pulling fs layer
ad1c7cfc347f: Waiting
4eb48115a042: Waiting
ccecc6c1c4bf: Waiting
63dc518f902b: Waiting
da802df85c96: Verifying Checksum
da802df85c96: Download complete
7d98d813d54f: Verifying Checksum
7d98d813d54f: Download complete
7aad5092c3b: Verifying Checksum
7aad5092c3b: Download complete
4eb48115a042: Verifying Checksum
4eb48115a042: Download complete
ccecc6c1c4bf: Verifying Checksum
ccecc6c1c4bf: Download complete
63dc518f902b: Verifying Checksum
63dc518f902b: Download complete
ad1c7cfc347f: Verifying Checksum
ad1c7cfc347f: Download complete
7d98d813d54f: Pull complete
da802df85c96: Pull complete
7aad5092c3b: Pull complete
ad1c7cfc347f: Pull complete
4eb48115a042: Pull complete
ccecc6c1c4bf: Pull complete
63dc518f902b: Pull complete
Digest:
sha256:ed8b9dd4e9f89c111f4bdb85a55f8c9f0e22796a298449380b15f627d9914095
Status: Downloaded newer image for python:3.9 ---> 24e94023d809
Step 2/7 : WORKDIR /app/backend ---> Running in e00638206e39
Removing intermediate container e00638206e39 ---> ad0da5688bb3
Step 3/7 : COPY requirements.txt /app/backend
---> 15c81df5b1f7
Step 4/7 : RUN pip install -r requirements.txt ---> Running in e56bb9ab53b5
Collecting asgiref==3.6.0
Downloading asgiref-3.6.0-py3-none-any.whl (23 kB)
Collecting Django==4.1.5
Downloading Django-4.1.5-py3-none-any.whl (8.1 MB)
----- 8.1/8.1 MB
54.0 MB/s eta
0:00:00
Collecting django-cors-headers==3.13.0
Downloading django_cors_headers-3.13.0-py3-none-any.whl (13 kB)
Collecting django-rest-framework==3.14.0
Downloading django-rest-framework-3.14.0-py3-none-any.whl (1.1 MB)
----- 1.1/1.1 MB
82.6 MB/s eta
0:00:00
Collecting gunicorn==20.1.0
Downloading gunicorn-20.1.0-py3-none-any.whl (79 kB)
----- 79.5/79.5
kB 17.3 MB/s eta

```

0:00:00
Collecting pytz==2022.7.1
Downloading pytz-2022.7.1-py2.py3-none-any.whl (499 kB)
----- 499.4/499.4 kB

70.4 MB/s eta
0:00:00
Collecting sqlparse==0.4.3
Downloading sqlparse-0.4.3-py3-none-any.whl (42 kB)
----- 42.8/42.8

kB 10.2 MB/s eta
0:00:00
Collecting tzdata==2022.7
Downloading tzdata-2022.7-py2.py3-none-any.whl (340 kB)
----- 340.1/340.1 kB

46.1 MB/s eta
0:00:00
Collecting whitenoise==6.3.0
Downloading whitenoise-6.3.0-py3-none-any.whl (19 kB)
Requirement already satisfied: setuptools>=3.0 in /usr/local/lib/python3.9/site-packages
(from gunicorn==20.1.0->-r requirements.txt (line 5)) (58.1.0)
Installing collected packages: pytz, whitenoise, tzdata, sqlparse, gunicorn, asgiref, Django,
djangorestframework, django-cors-headers
Successfully installed Django-4.1.5 asgiref-3.6.0 django-cors-headers-3.13.0
djangorestframework-3.14.0 gunicorn-20.1.0 pytz-2022.7.1 sqlparse-0.4.3 tzdata-2022.7
whitenoise-6.3.0
[91mWARNING: Running pip as the 'root' user can result in broken permissions and
conflicting behaviour with the system package manager. It is recommended to use a virtual
environment instead: https://pip.pypa.io/warnings/venv
[0m [91m
[notice] A new release of pip is available: 23.0.1 -> 24.2
[notice] To update, run: pip install --upgrade pip
[0mRemoving intermediate container e56bb9ab53b5 ---> 974787a936f4
Step 5/7 : COPY . /app/backend ---> 0e260cfe00b8
Step 6/7 : EXPOSE 8000 ---> Running in e8950e6c7108
Removing intermediate container e8950e6c7108
---> 0eb1c801de16
Step 7/7 : CMD python /app/backend/manage.py runserver 0.0.0.0:8000 ---> Running in 167bfdbc3a48
Removing intermediate container 167bfdbc3a48 ---> d011eef81c6d
Successfully built d011eef81c6d
Successfully tagged my-note-app:latest
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Pushing the image to Dockerhub)
[Pipeline] echo
pushing image to docker hub
[Pipeline] withCredentials
Masking supported pattern matches of $dockerHubPass
[Pipeline] {
[Pipeline] sh
+ docker tag my-note-app kishorss/my-note-app:latest
[Pipeline] sh
Warning: A secret was passed to "sh" using Groovy String interpolation, which is insecure.
Affected argument(s) used the following variable(s): [dockerHubPass]
See https://jenkins.io/redirect/groovy-string-interpolation for details.
+ docker login -u kishorss -p ****

```

WARNING! Using --password via the CLI is insecure. Use --password-stdin.
WARNING! Your password will be stored unencrypted in
/var/lib/jenkins/.docker/config.json.
Configure a credential helper to remove this warning. See
<https://docs.docker.com/engine/reference/commandline/login/#credentials-store>
Login Succeeded
[Pipeline] sh
+ docker push kishorss/my-note-app:latest
The push refers to repository [docker.io/kishorss/my-note-app]
8e7837aa202d: Preparing
d2b7d6a9a826: Preparing
bfa256344eb1: Preparing
bd0d59bebbdb: Preparing
18a1e5930724: Preparing
324e6299c847: Preparing
ef4ab4b82327: Preparing
d23b5e6144a7: Preparing
e5ee1bd83fe3: Preparing
43da071b5e0c: Preparing
ef5f5ddeb0a6: Preparing
324e6299c847: Waiting
ef4ab4b82327: Waiting
d23b5e6144a7: Waiting
e5ee1bd83fe3: Waiting
43da071b5e0c: Waiting
ef5f5ddeb0a6: Waiting
18a1e5930724: Mounted from library/python
bfa256344eb1: Pushed
bd0d59bebbdb: Pushed
324e6299c847: Mounted from library/python
8e7837aa202d: Pushed
ef4ab4b82327: Mounted from library/python
d23b5e6144a7: Mounted from library/python
43da071b5e0c: Mounted from library/python
e5ee1bd83fe3: Mounted from library/python
ef5f5ddeb0a6: Mounted from library/python
d2b7d6a9a826: Pushed
latest: digest:
sha256:1ea2589aac4536a1a002d854f58858767b340324aaed279938cddad791e4a4f4
size: 2633
[Pipeline] }
[Pipeline] // withCredentials
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy)
[Pipeline] echo
Deploying the conatiner
[Pipeline] sh
+ docker run -d -p 8000:8000 kishorss/my-note-app:latest
fee5ba35bbf1ef2e8df256f76b2322fb8205ff604b1fa21caba9cd839b6334ab
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS