Project Statement I

1. Set up Jenkins Environment

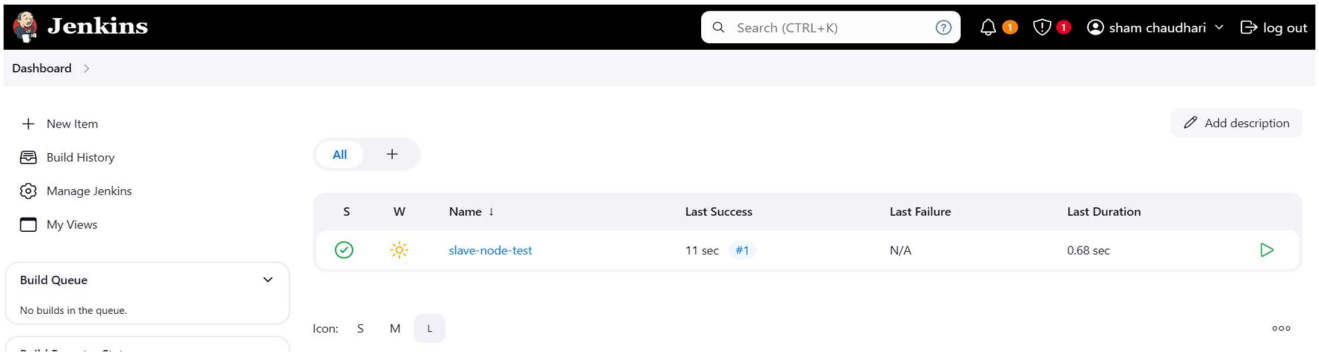
Master Node Setup:

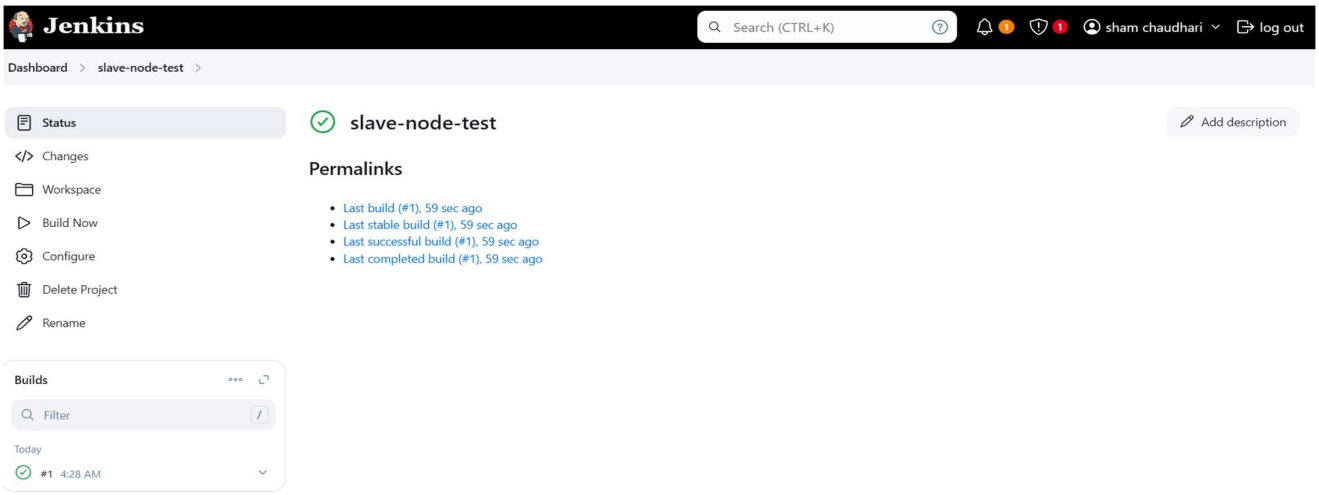
```
root@jenkins-m:~# jenkins --version
2.479.2
root@jenkins-m:~# java --version
openjdk 17.0.13 2024-10-15
OpenJDK Runtime Environment (build 17.0.13+11-Ubuntu-2ubuntu120.04)
OpenJDK 64-Bit Server VM (build 17.0.13+11-Ubuntu-2ubuntu120.04, mixed mode, sharing)
root@jenkins-m:~# ansible --version
ansible [core 2.12.10]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/root/.ansible/plugins/modules', '/usr/share/ansible/
plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /root/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.8.10 (default, Nov  7 2024, 13:10:47) [GCC 9.4.0]
  jinja version = 2.10.1
  libyaml = True
root@jenkins-m:~# git --version
git version 2.25.1
root@jenkins-m:~#
```

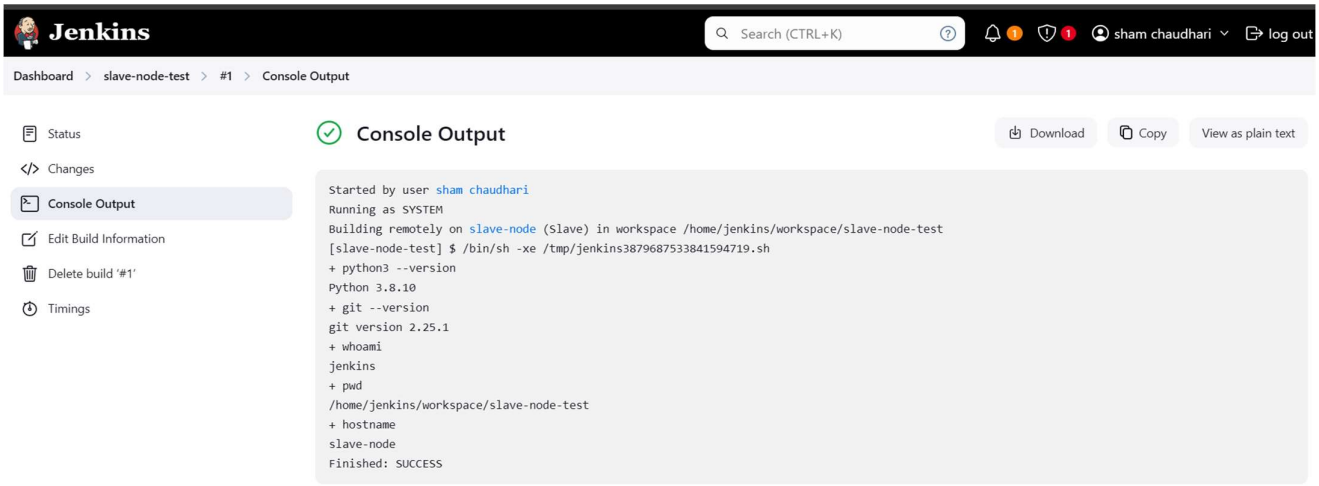
Slave Node Setup:

```
jenkins@slave-node:~$ python3 --version
Python 3.8.10
jenkins@slave-node:~$ git --version
git version 2.25.1
jenkins@slave-node:~$ ssh -V
OpenSSH_8.2p1 Ubuntu-4ubuntu0.11, OpenSSL 1.1.1f  31 Mar 2020
jenkins@slave-node:~$
```

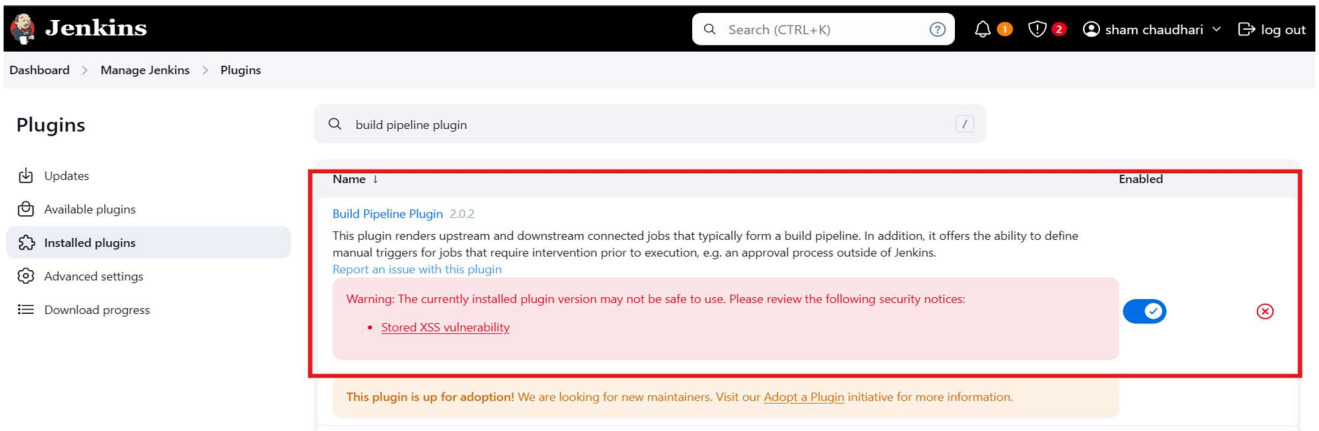
Verify master node and slave node connection using agent.

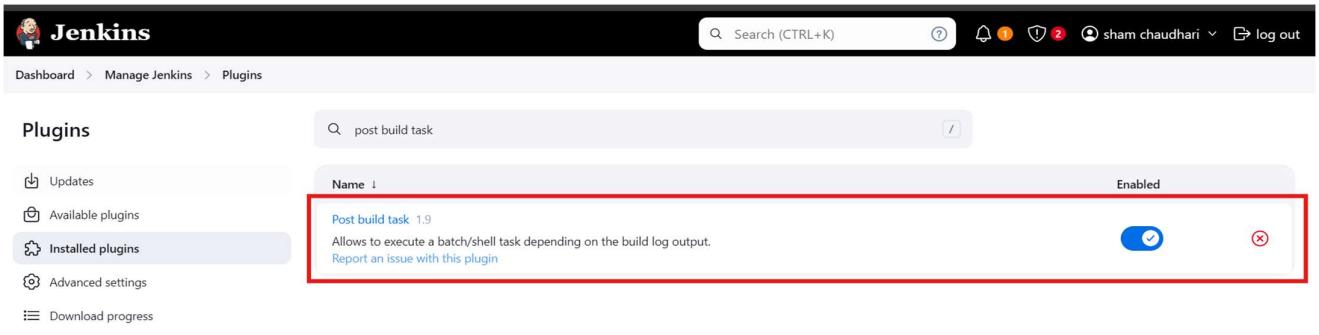






2. To add the Build Pipeline Plugin and Post-Build Task Plugin to Jenkins on your master VM.



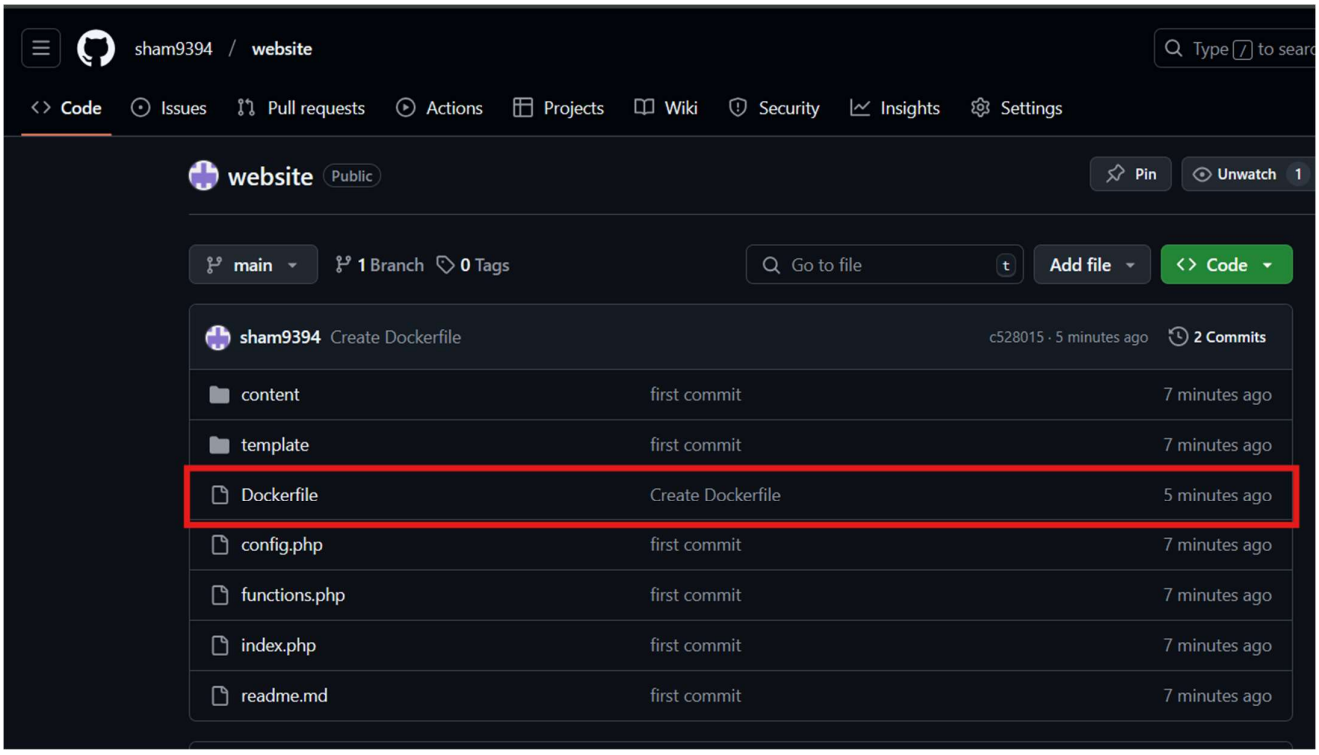


3. **Use the image devopsedu/webapp and add your PHP website to it using a Dockerfile.**

Create the Dockerfile using base image devopsedu/webapp

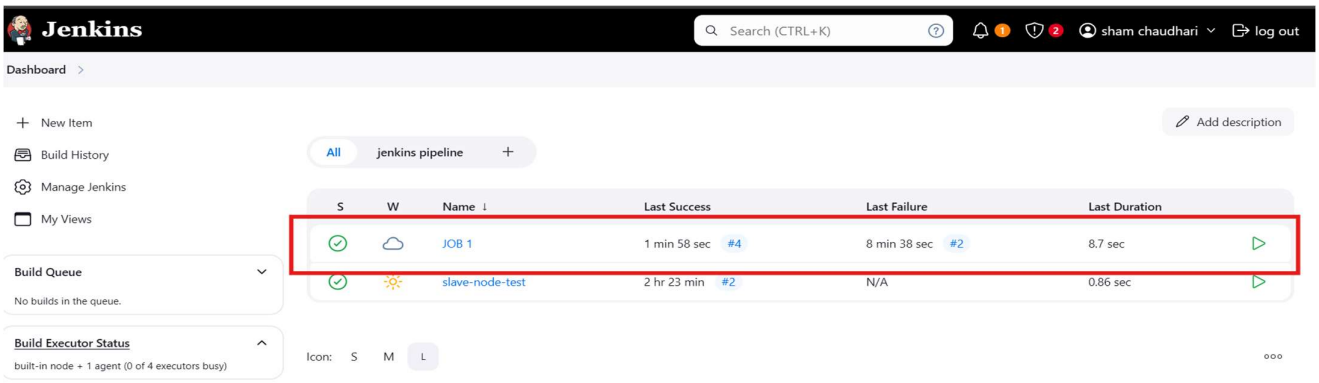
```
# Use the devopsedu/webapp as the base image
FROM devopsedu/webapp

# Set the working directory in the container
WORKDIR /var/www/html

# Copy your PHP website files into the container
COPY website/ /var/www/html/

# Expose the default Apache port
EXPOSE 80

# Start the Apache server
CMD ["apachectl", "-D", "FOREGROUND"]
```

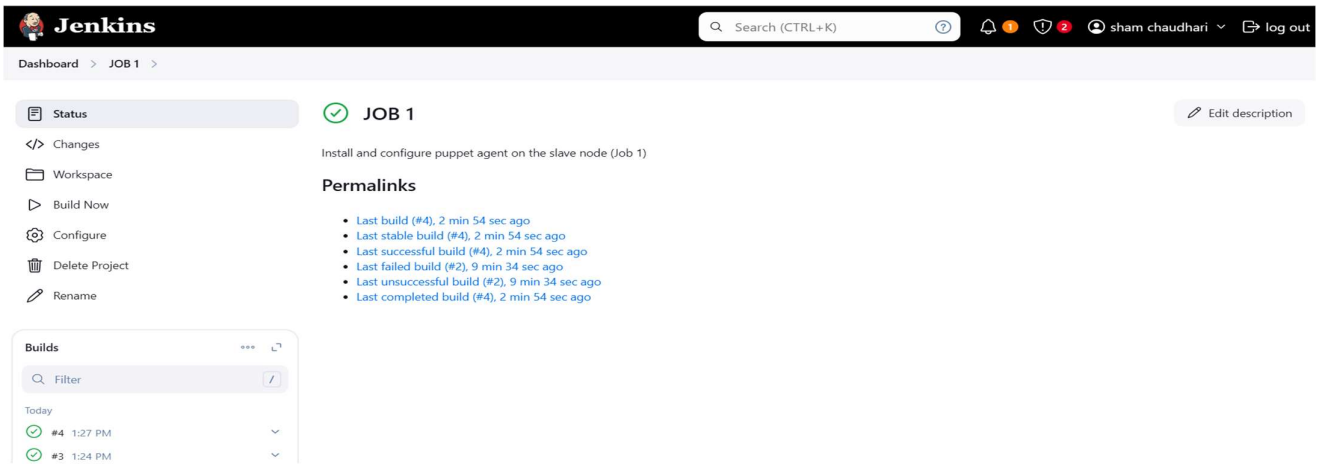
4. **Push the PHP website and the Dockerfile to a git repository**

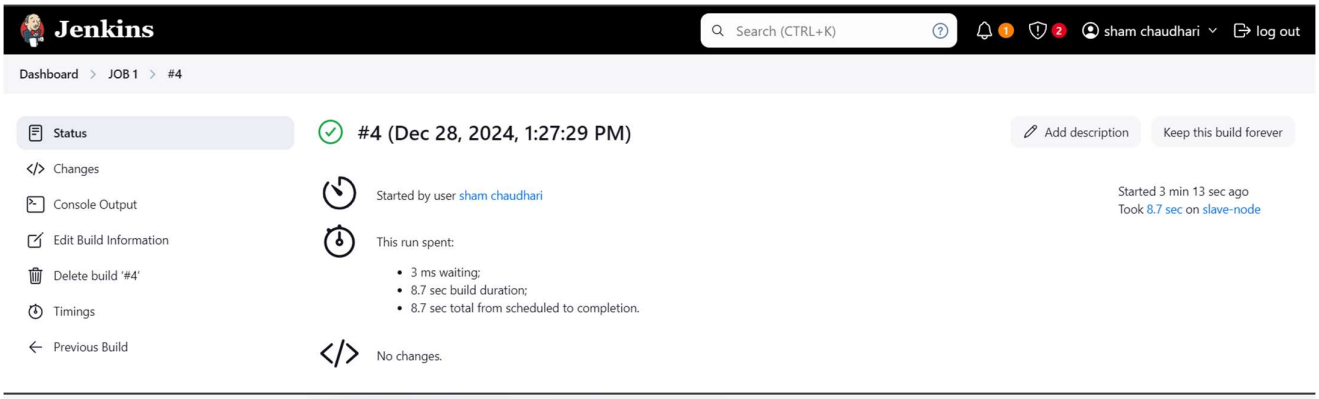


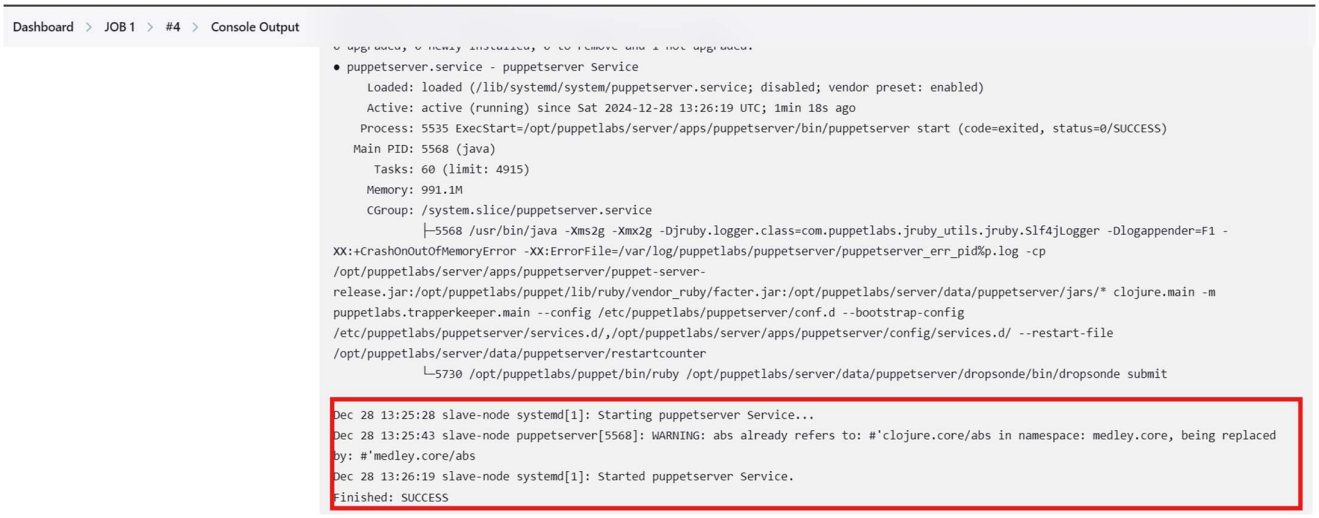
JOB 1

Install and configure puppet agent on the slave node



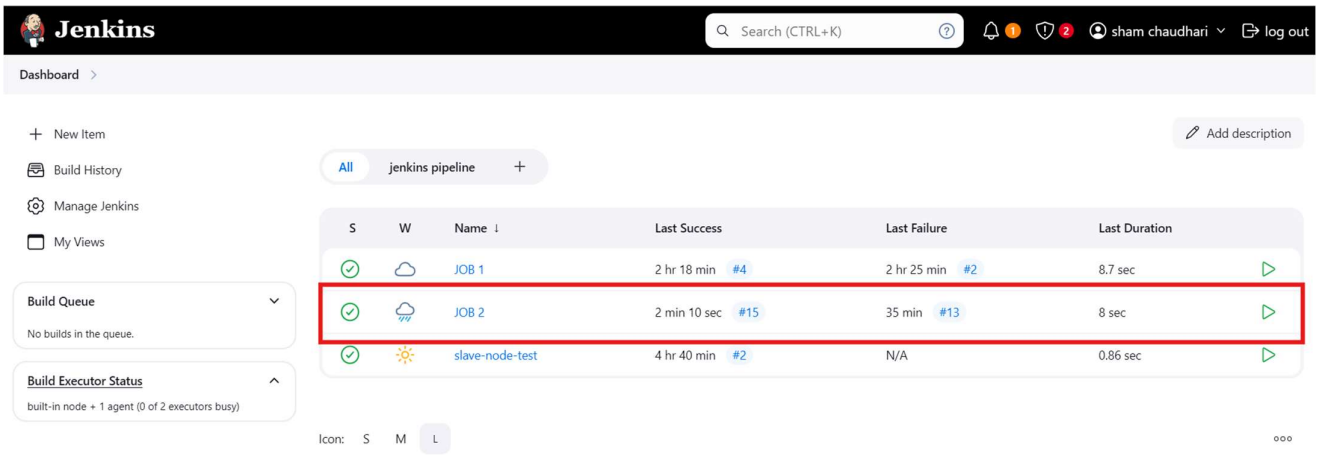


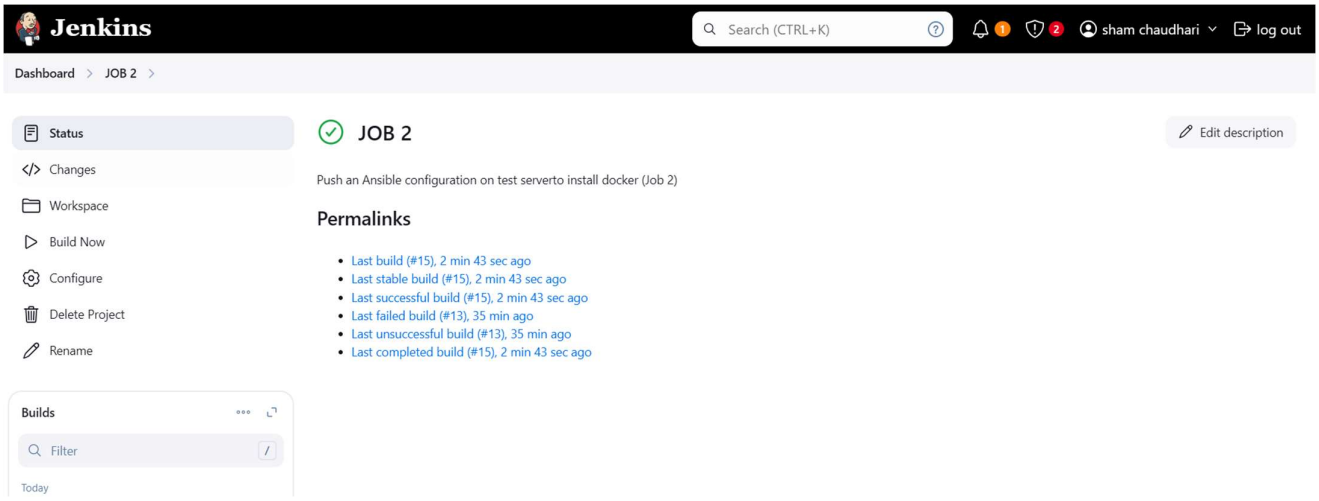


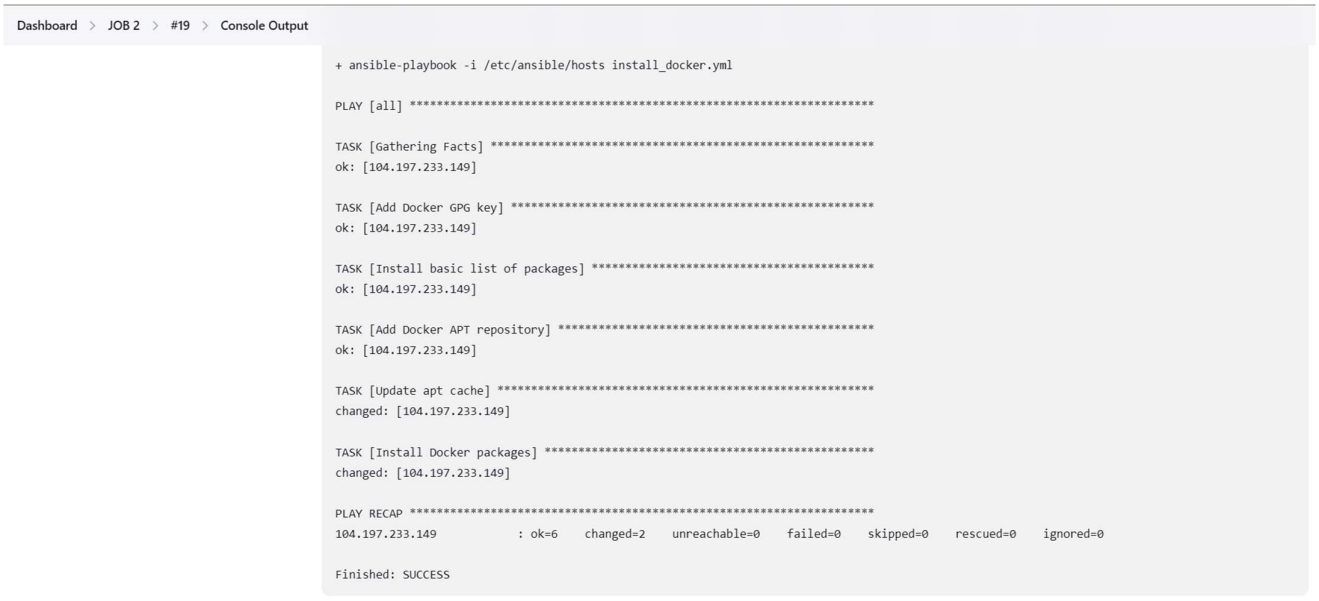


JOB 2

Push an Ansible configuration on test server to install docker





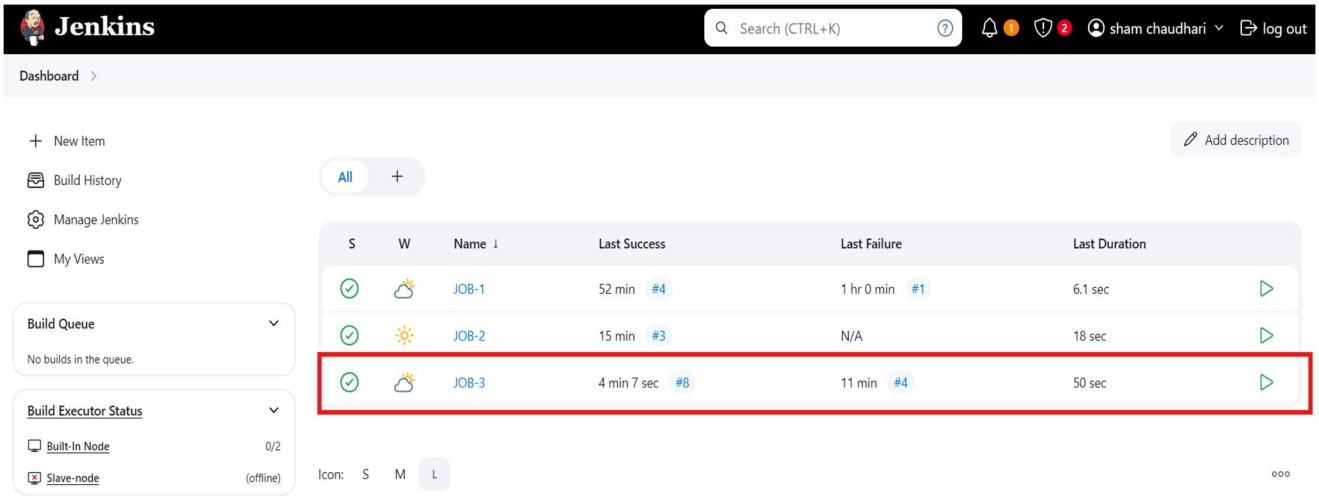


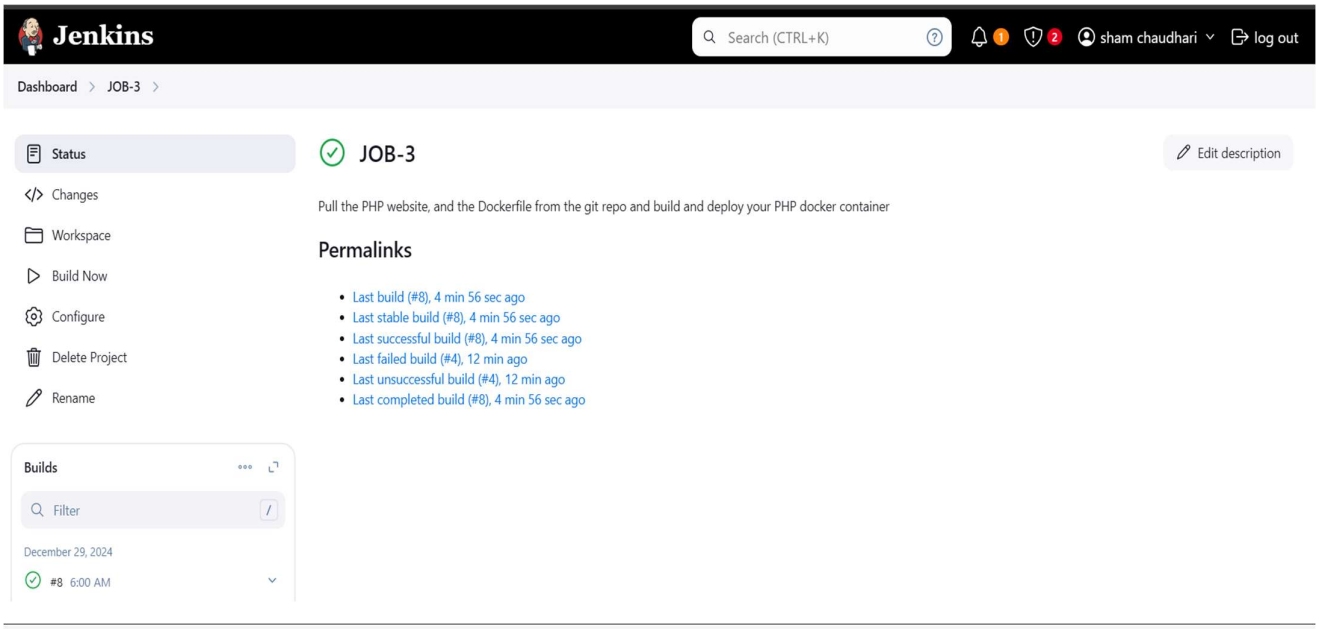
Verify:

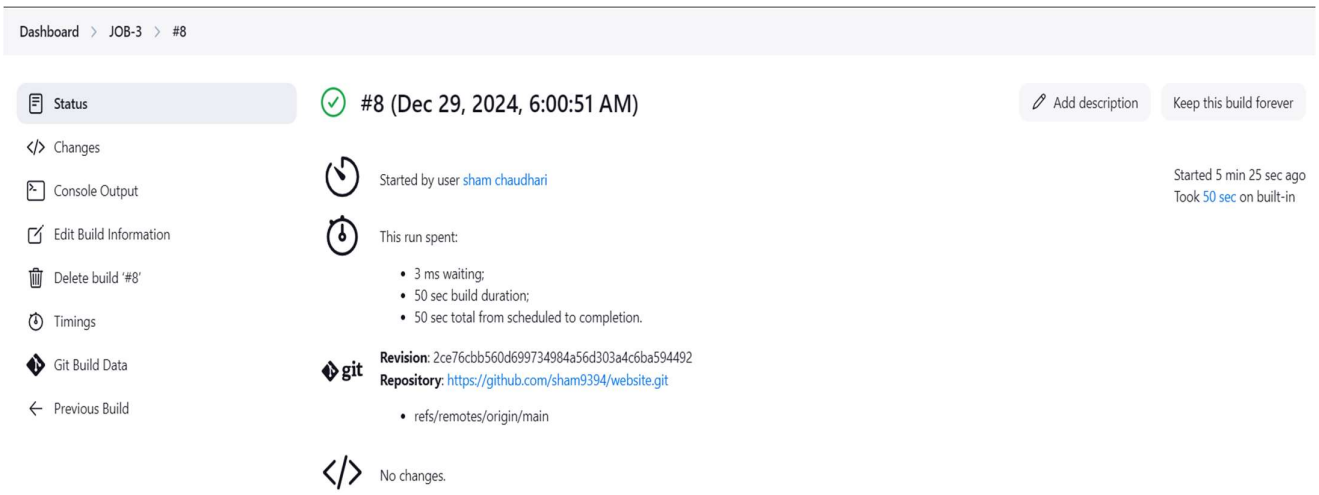
```
root@slave-node:/home# docker --version
Docker version 27.4.1, build b9d17ea
root@slave-node:/home#
```

JOB 3

Pull the PHP website, and the Dockerfile from the git repo and build and deploy your PHP docker container.







Dashboard > JOB-3 > #8 > Console Output

Status
Changes
Console Output
Edit Build Information
Delete build '#8'
Timings
Git Build Data
Previous Build

Console Output

Download Copy View as plain text

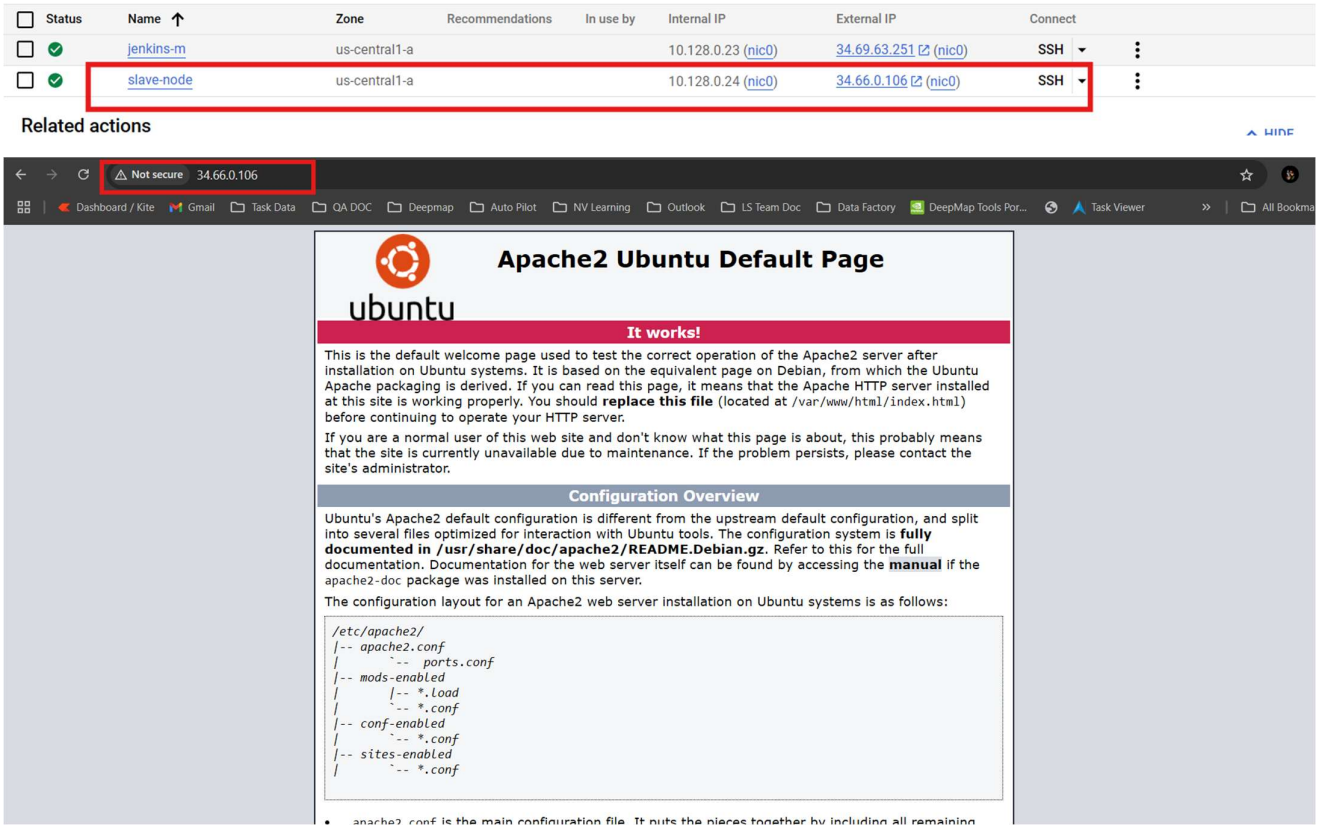
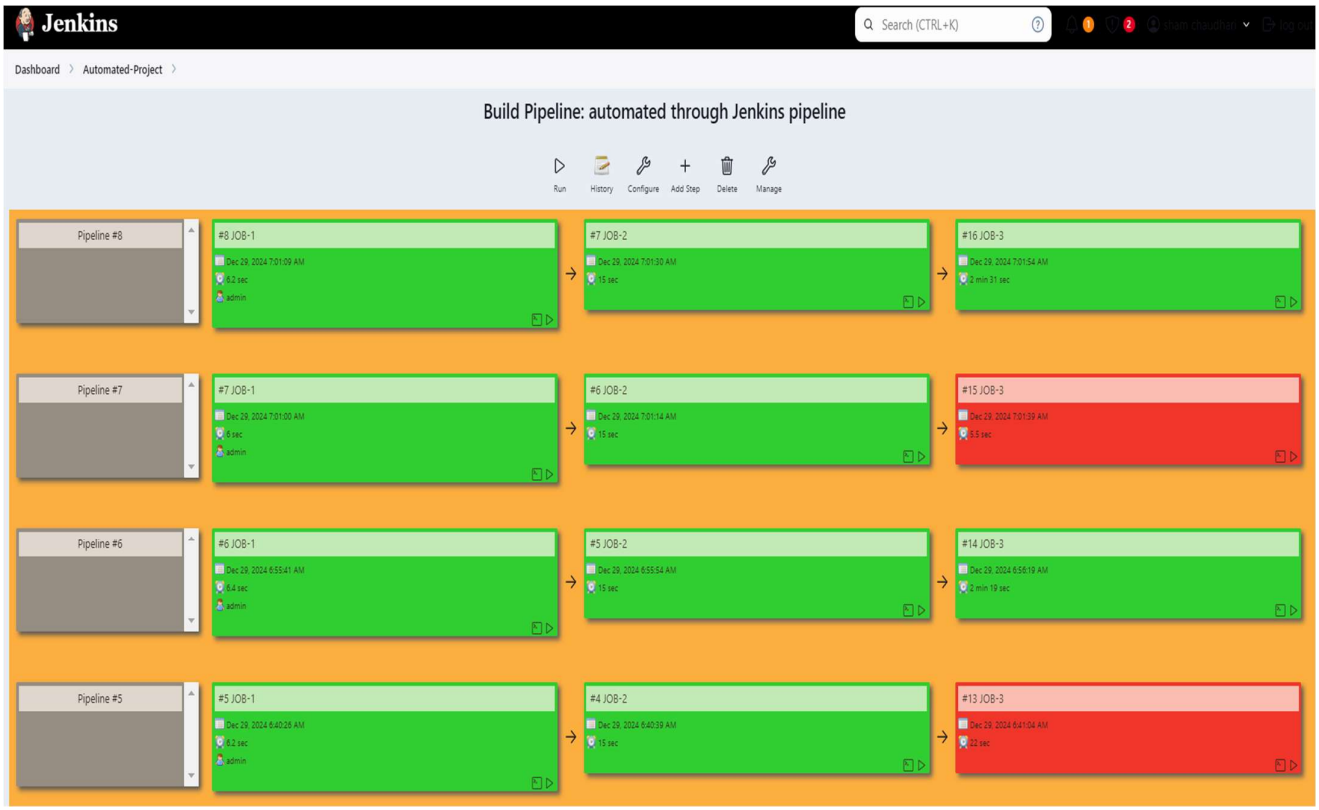
```
Started by user sham chaudhari
Running as SYSTEM
Building on the built-in node in workspace /var/lib/jenkins/workspace/JOB-3
The recommended git tool is: NONE
No credentials specified
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/JOB-3/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/sham9394/website.git # timeout=10
Fetching upstream changes from https://github.com/sham9394/website.git
 > git --version # timeout=10
 > git --version # 'git version 2.25.1'
 > git fetch --tags --force --progress -- https://github.com/sham9394/website.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 2ce76cbb560d699734984a56d303a4c6ba594492 (refs/remotes/origin/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 2ce76cbb560d699734984a56d303a4c6ba594492 # timeout=10
Commit message: "Update Dockerfile"
 > git rev-list --no-walk 2ce76cbb560d699734984a56d303a4c6ba594492 # timeout=10
[JOB-3] $ /bin/sh -xe /tmp/jenkins15211684761236012731.sh
+ cd /var/lib/jenkins/workspace/JOB-3
+ docker build -t my-php-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/
```

Dashboard > JOB-3 > #8 > Console Output

```
Sending build context to Docker daemon  238.7MB

Step 1/5 : FROM devopsedu/webapp
 ---> 0ef91240e173
Step 2/5 : WORKDIR /var/www/html
 ---> Using cache
 ---> a45b6da9ea0f
Step 3/5 : COPY ./ /var/www/html/
 ---> 4da40235805c
Step 4/5 : EXPOSE 80
 ---> Running in 73c9f38befde
Removing intermediate container 73c9f38befde
 ---> 66c201a87ee2
Step 5/5 : CMD ["apachectl", "-D", "FOREGROUND"]
 ---> Running in 2903e56f1549
Removing intermediate container 2903e56f1549
 ---> 74657ef9c52a
Successfully built 74657ef9c52a
Successfully tagged my-php-app:latest
+ docker save my-php-app
+ gzip
+ scp my-php-app.tar.gz root@34.66.0.106:/tmp
+ ssh root@34.66.0.106
Pseudo-terminal will not be allocated because stdin is not a terminal.
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-1073-gcp x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
```

Dashboard > JOB-3 > #8 > Console Output

```
 System information as of Sun Dec 29 06:01:23 UTC 2024

  System load:  0.02              Processes:             166
  Usage of /:   43.0% of 9.51GB   Users logged in:       1
  Memory usage: 6%                IPv4 address for ens4: 10.128.0.24
  Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

7 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 '22.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


Loaded image: my-php-app:latest
0187be8070947e4ff2294abc0589f224f4d417b5737560ad46e6e47820cfb6c9
Finished: SUCCESS
```



5. **automated through Jenkins by creating a pipeline: using Build Pipeline Plugin.**