**DevOps Capstone Project 1**

**Problem Statement:**

You have been hired as a Sr. DevOps Engineer in Abode Software. They want to implement the DevOps Lifecycle in their company. You have been asked to implement this lifecycle as fast as possible. Abode Software is a product-based company and their product is available on this GitHub link.

<https://github.com/sutarakash2411/DevOps_Capstone_Proj1.git>

The following are the specifications of the lifecycle:

1. Install the necessary software on the machines using a configuration management tool

2. Git workflow has to be implemented

3. CodeBuild should automatically be triggered once a commit is made to the master branch or develop branch.

a. If a commit is made to the master branch, test and push to prod

b. If a commit is made to the develop branch, just test the product, do not push to prod.

4. The code should be containerized with the help of a Dockerfile. The Dockerfile should be built every time there is a push to GitHub. Use the following pre-built container for your application: hshar/webapp. The code should reside in '/var/www/html'

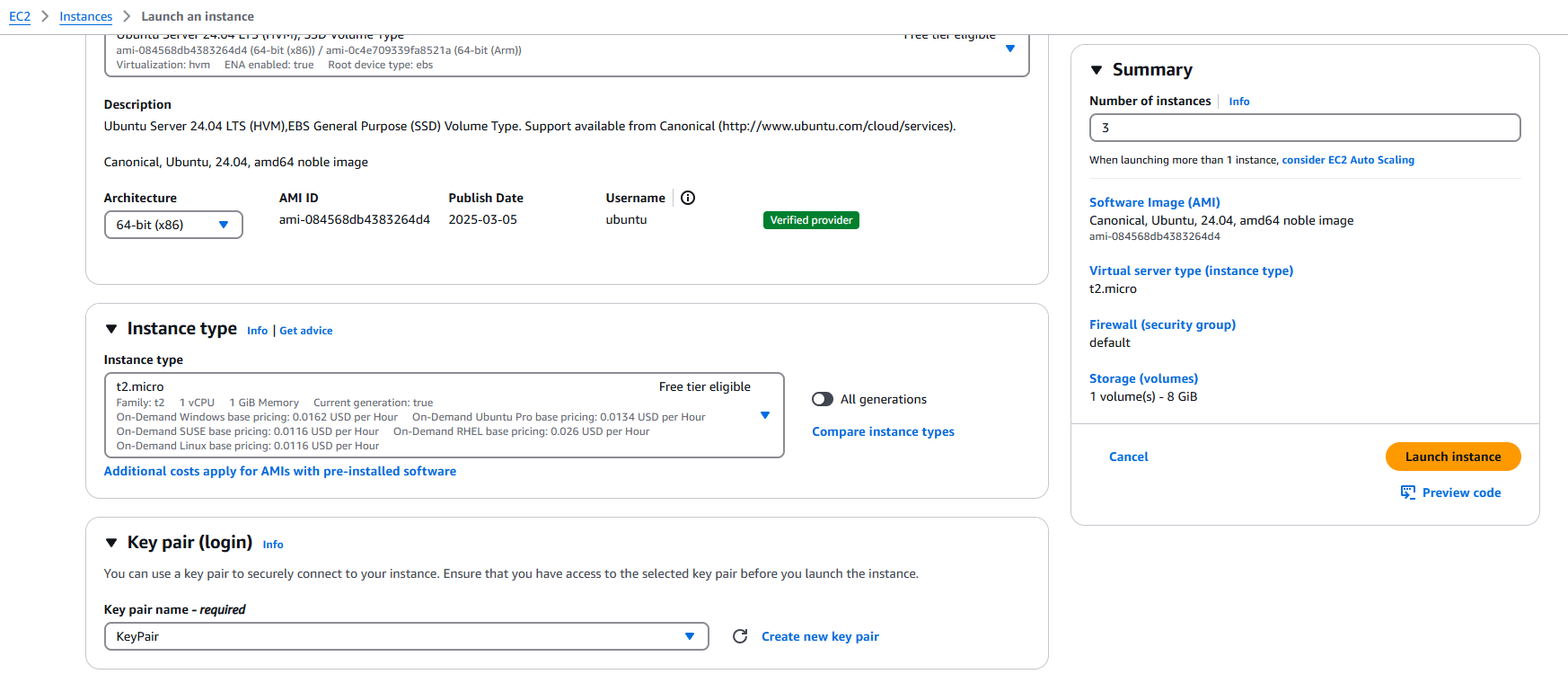
5. The above tasks should be defined in a Jenkins Pipeline with the following jobs:

a. Job1: build

b. Job2: test

c. Job3: Prod

Creating the 3 instances on aws :



Connecting to the project1\_master instance and installing ansible on it

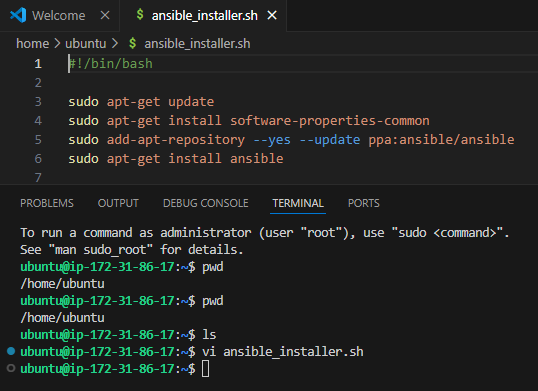
#!/bin/bash

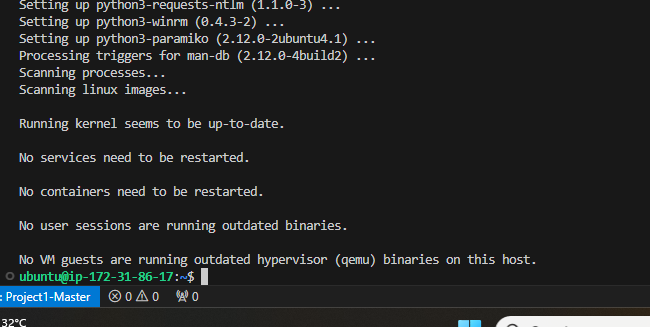
sudo apt-get update

sudo apt-get install software-properties-common

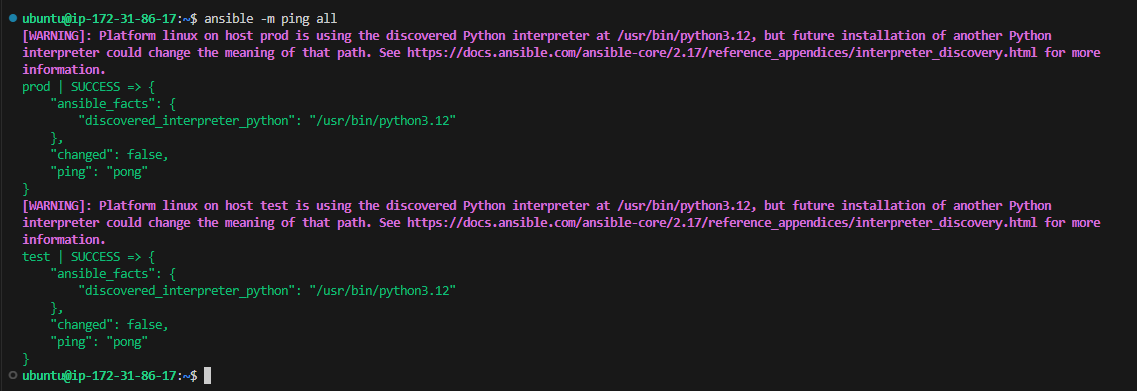
sudo add-apt-repository --yes --update ppa:ansible/ansible

sudo apt-get install ansible

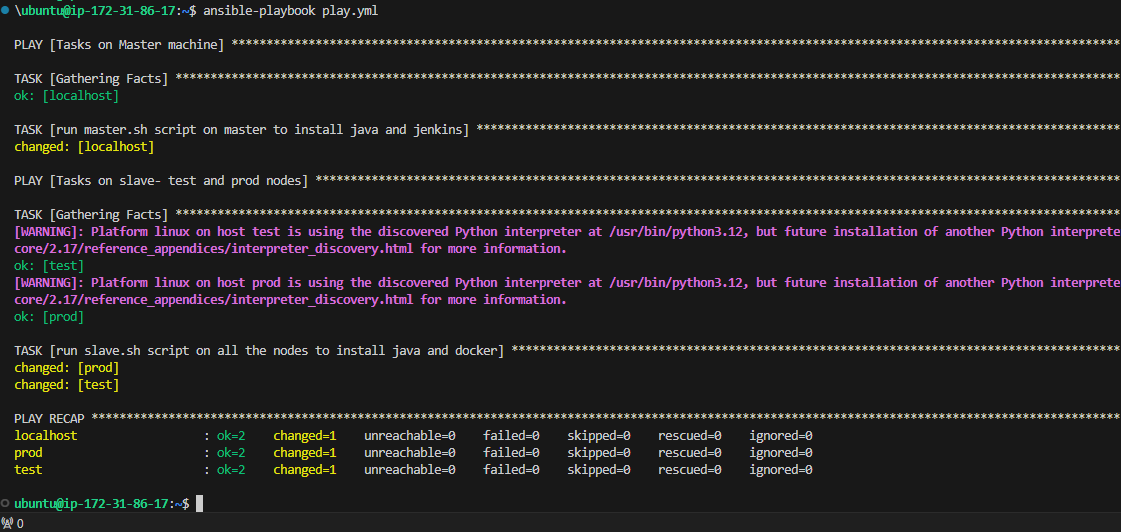




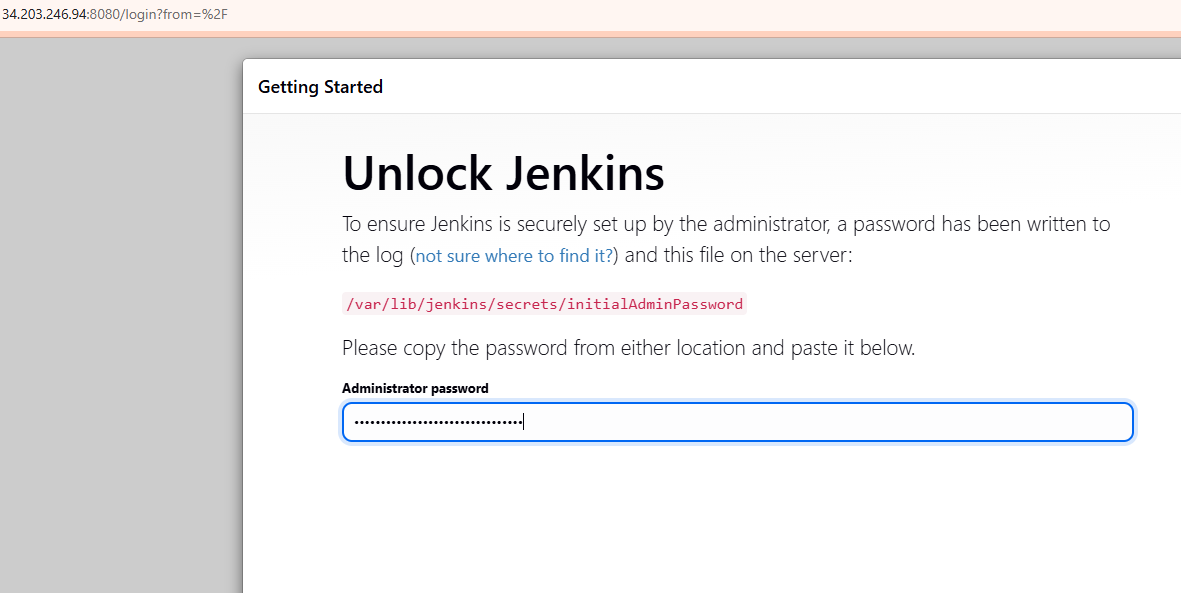
Successfully able to ping test and prod nodes from the master node:

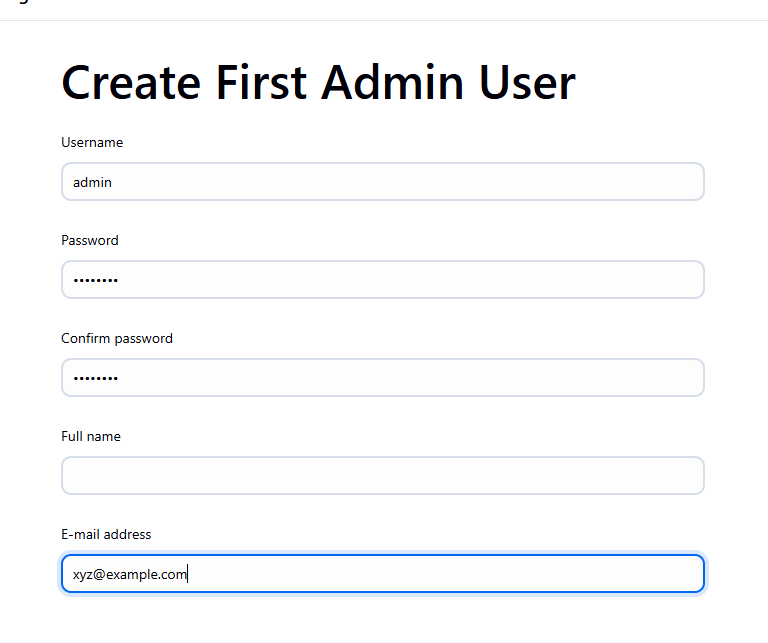


Running the play.yml playbook to install java and Jenkins on master and java and docker on test and prod nodes:



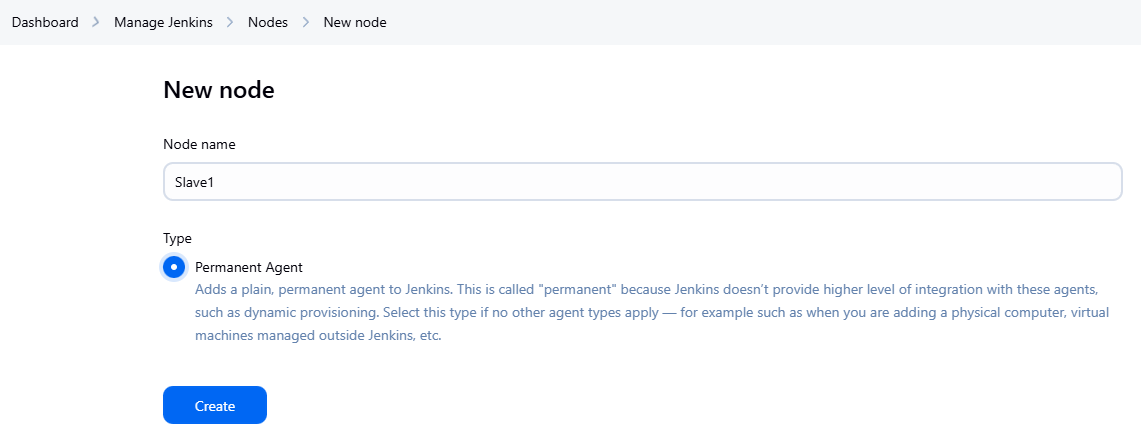
Jenkins installed accessing on ui using public IP:8080

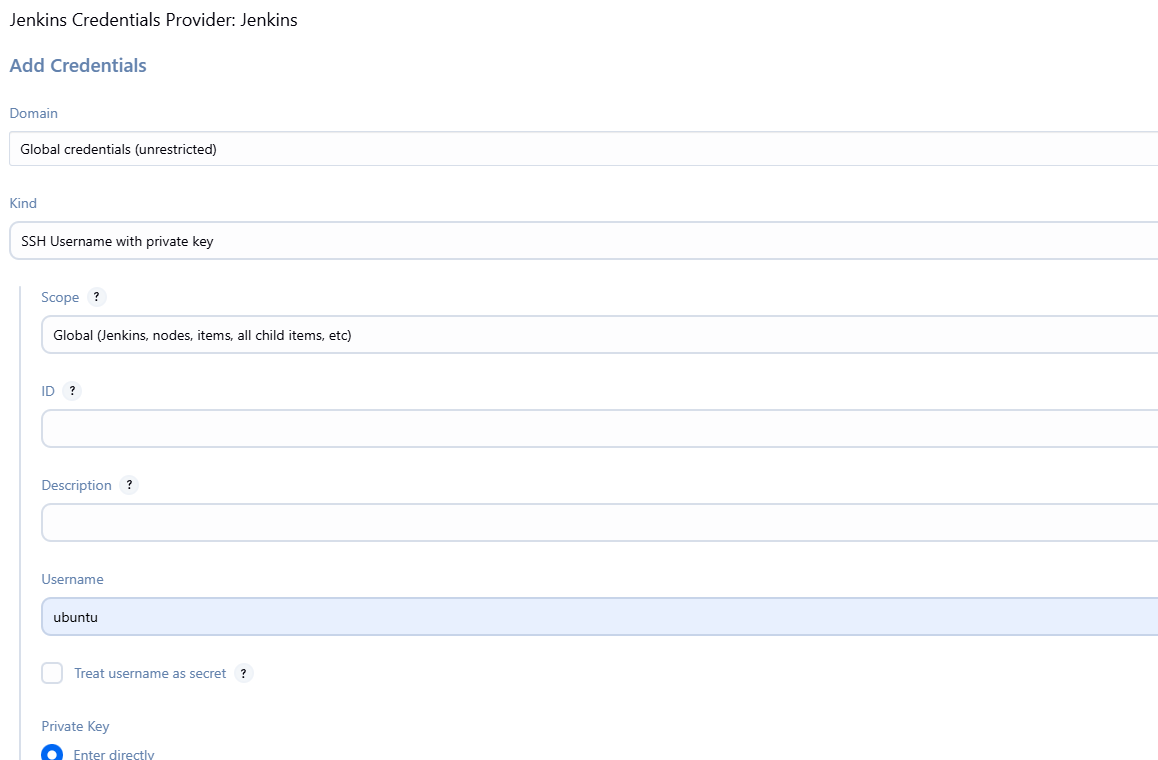
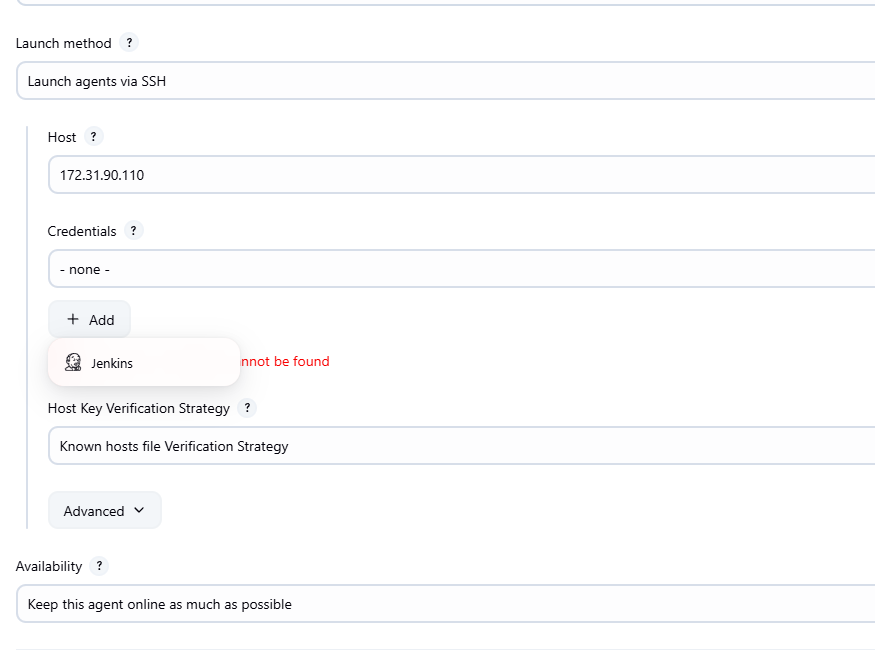
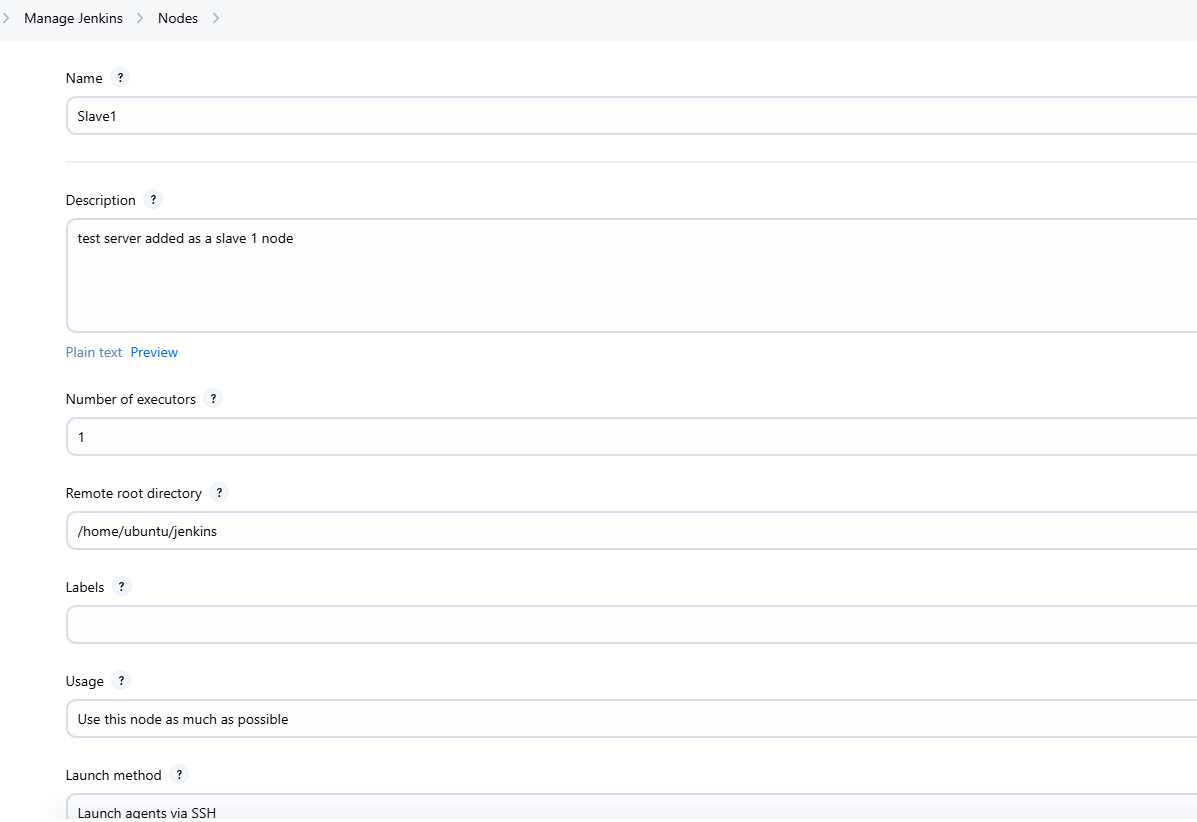




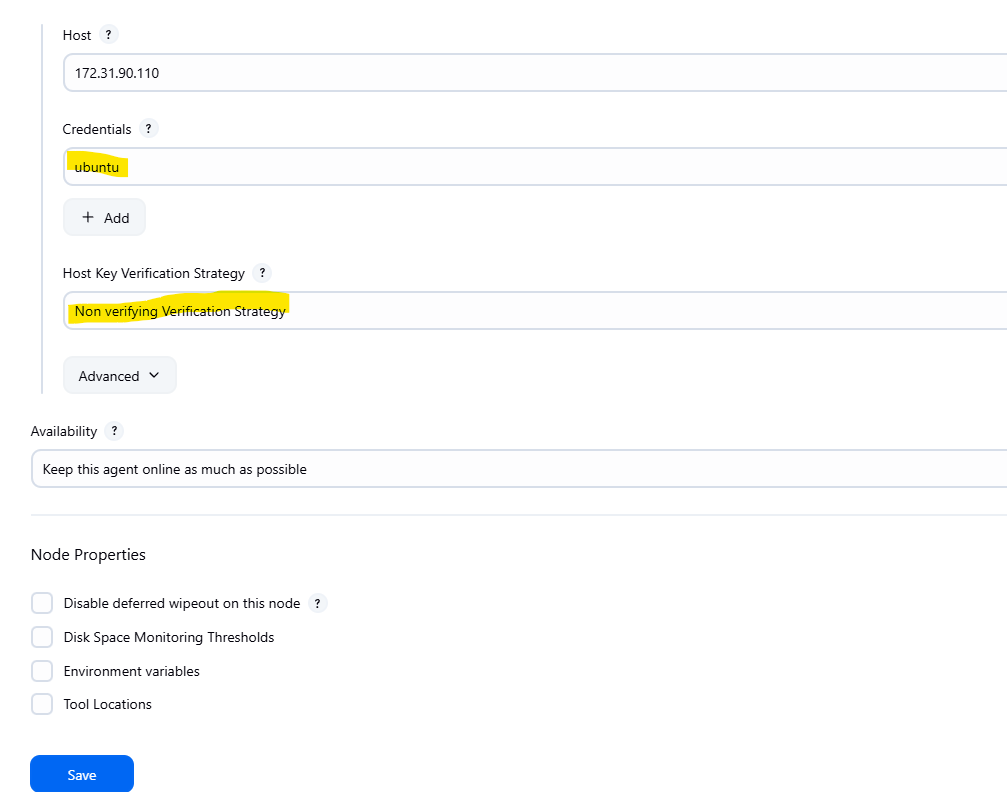
Once connected to Jenkins, adding the 2 nodes

Path: Dashboard > Manage Jenkins > Nodes > New Node

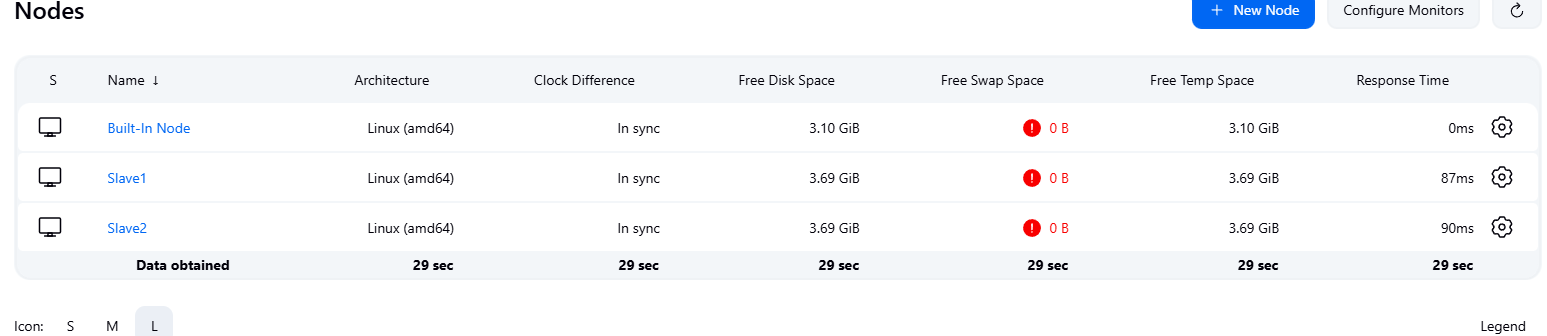




Private key value added directly



We can similarly create another node for prod

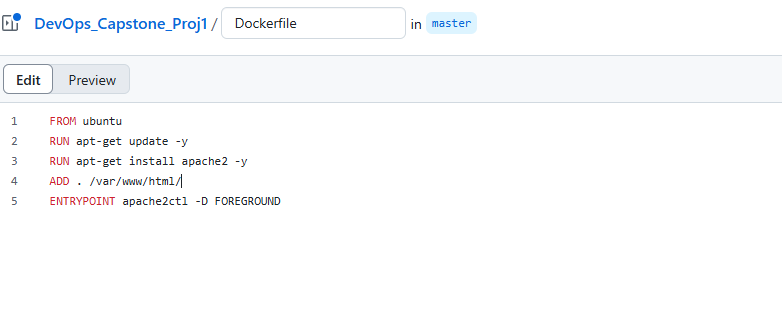


Website creation :

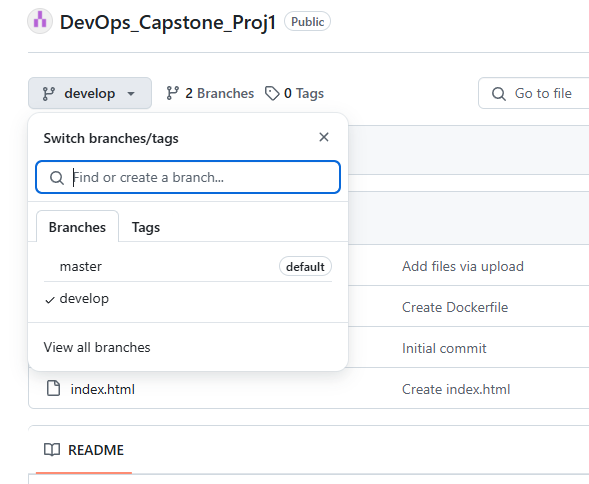
Index.html



Dockerfile to dockerize the website



Also, creating a new branch develop as per this project requirement:



Now, creating jobs on Jenkins server as per the following condition:

3. CodeBuild should automatically be triggered once a commit is made to the master branch or develop branch.

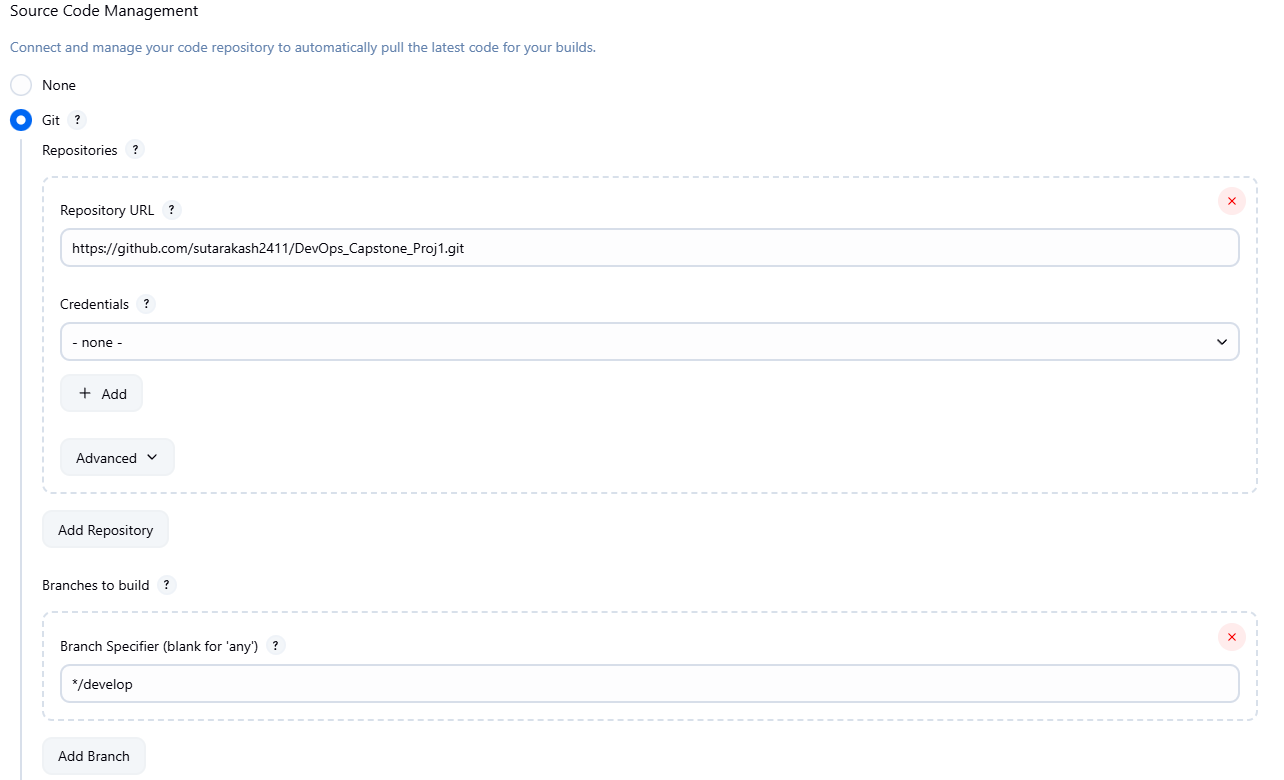
a. If a commit is made to the master branch, test and push to prod

**Job2 > Slave1 (Test)> master**

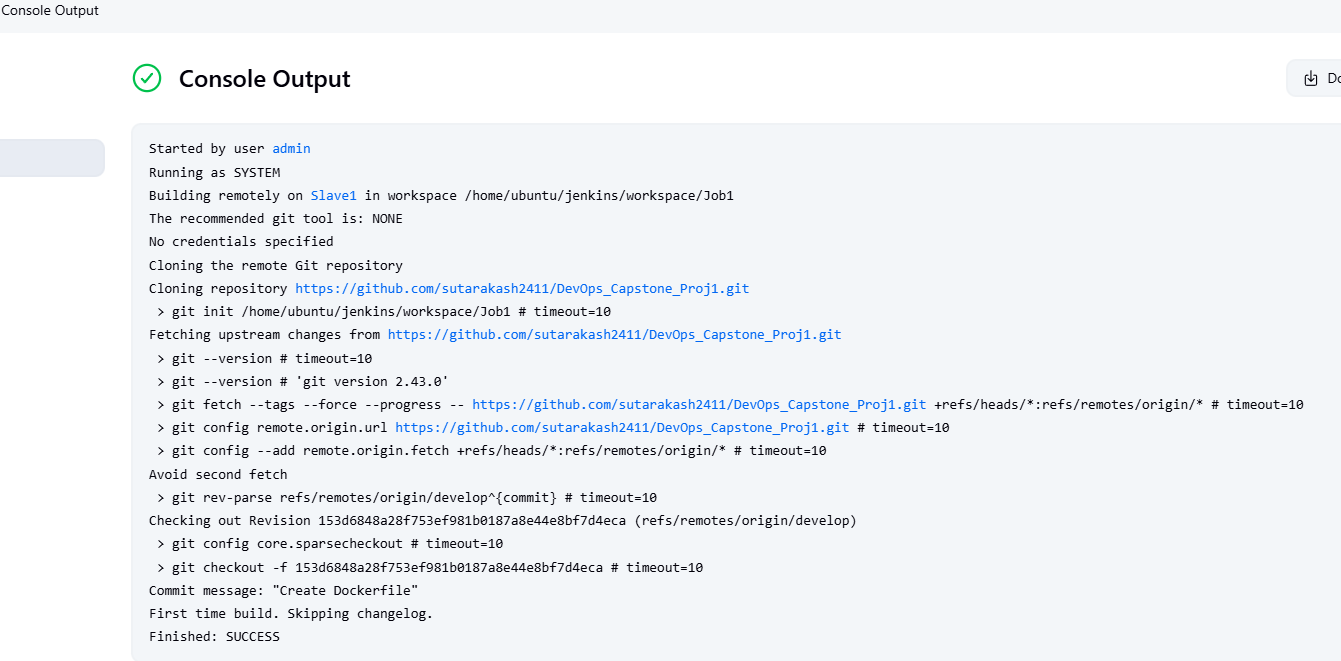
**Job3 > Slave2 (Prod)> master**

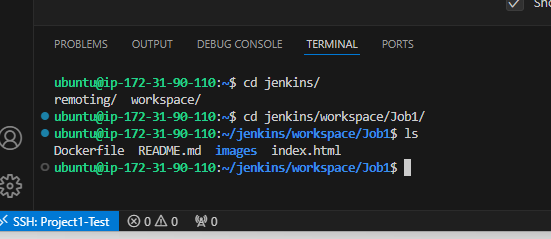
b. If a commit is made to the develop branch, just test the product, do not push to prod.

**Job1 > Slave1 (Test)> develop**

**** ****

On running this Job1, the git code is copied to slave1 (test)

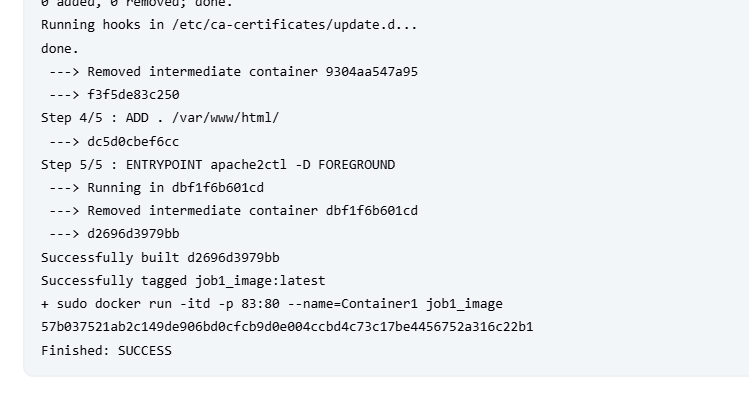
****

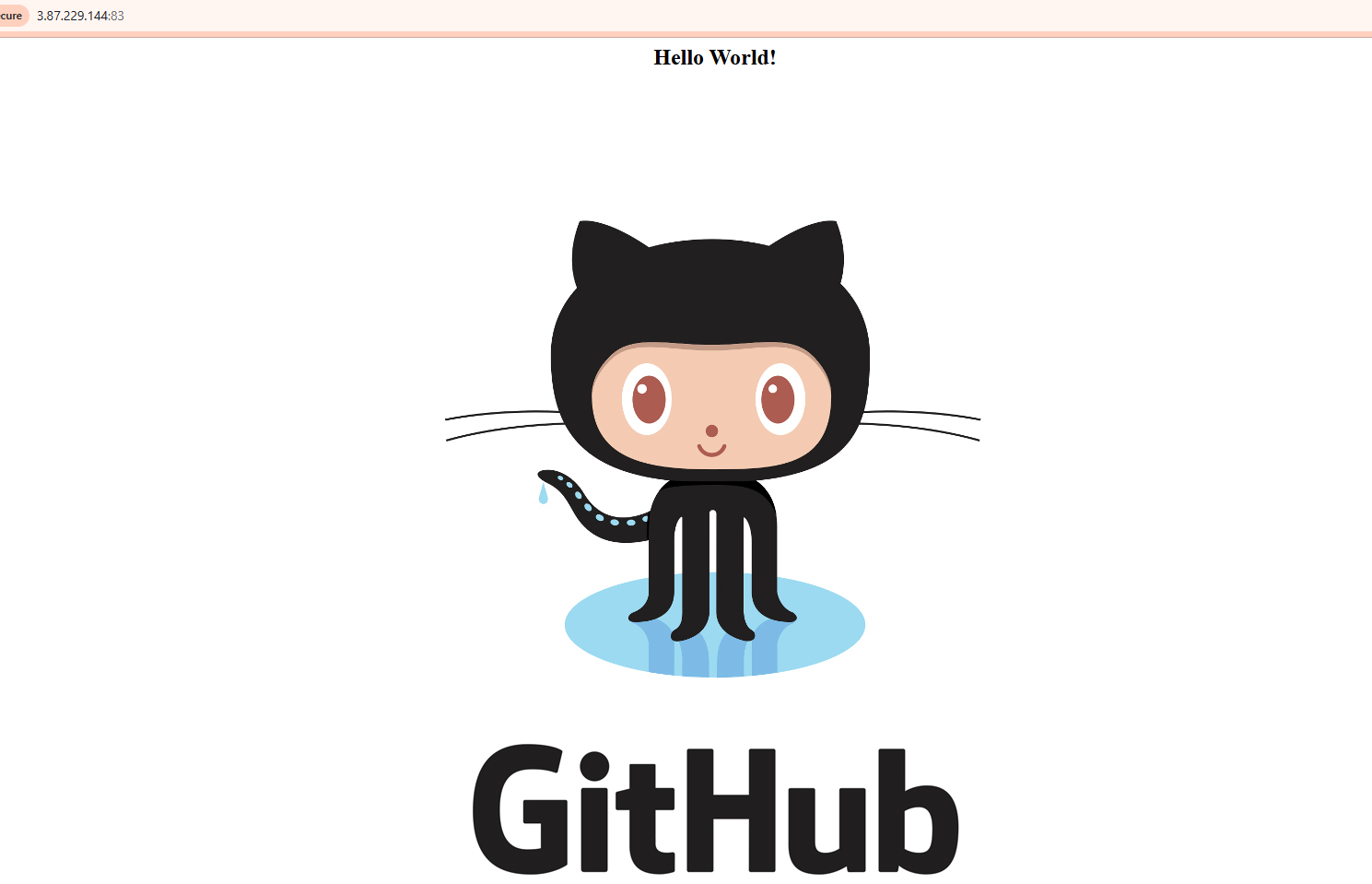
****

Further, to create an image from this Dockerfile, we need to update the Job1 configuration “Build Steps” and provide the command there itself, This will create an image from the Dockerfile on running the Job itself

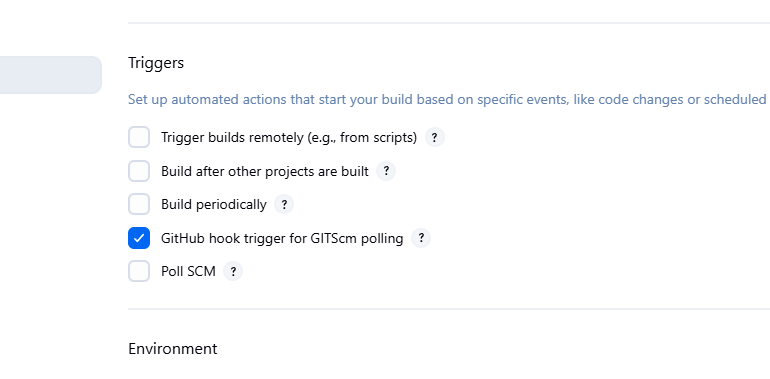


Once the above changes are done, and if we run the job again, then an image is created first and then the container is created from that image. We can see this container running on port 83

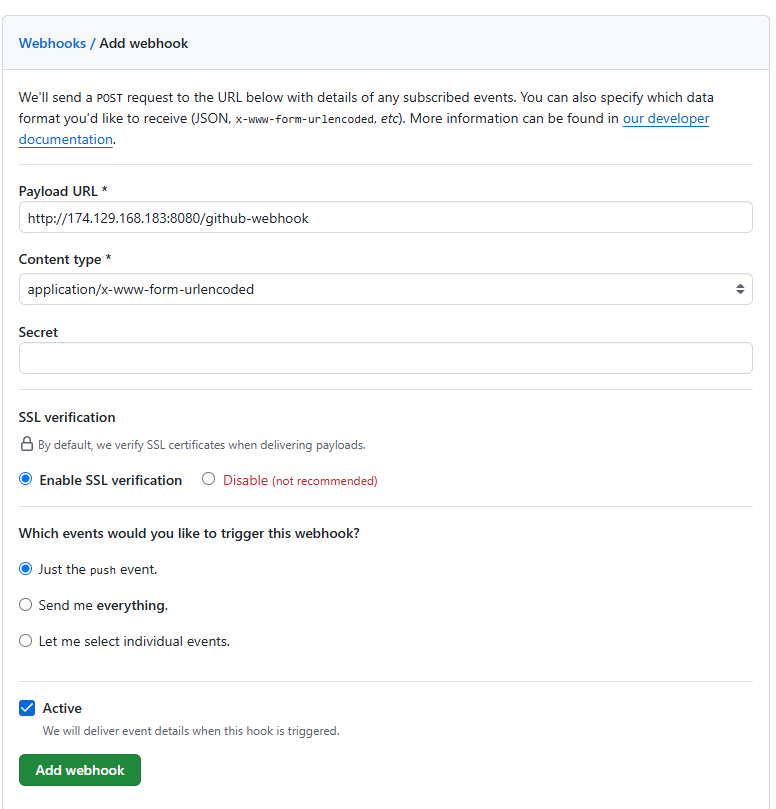




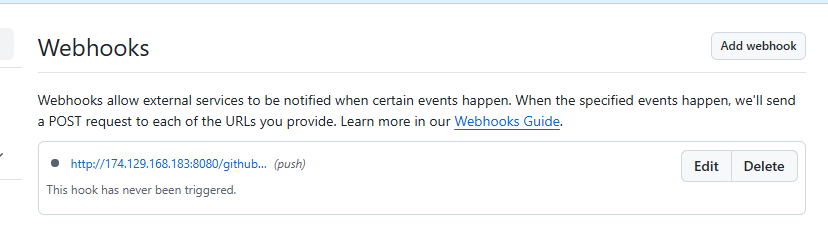
Now to automate the pipeline, we need to configure the GitHub WebHook



Copy the Jenkins Dashboard url and go to github > settings > webhook >add



Webhook created:



Whenever there is a code update and a commit, this webhook will trigger a job

Note: the Above webhook might not trigger the job if the public IP used does not match the public IP of the Jenkins master instance. To validate the same, follow the below check:

1. Verify if the webhook has the current public IP of the Jenkins Master node.
2. Navigate to Dashboard > Manage Jenkins > System > Jenkins URL and update the URL with the current public IP.

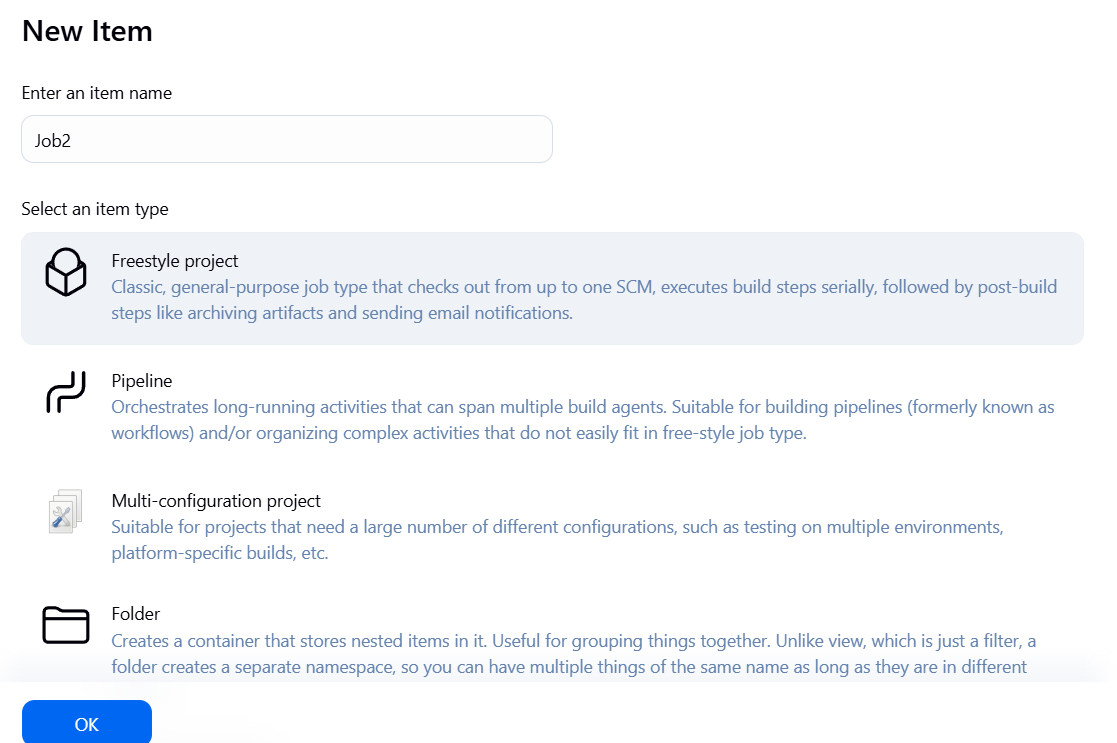
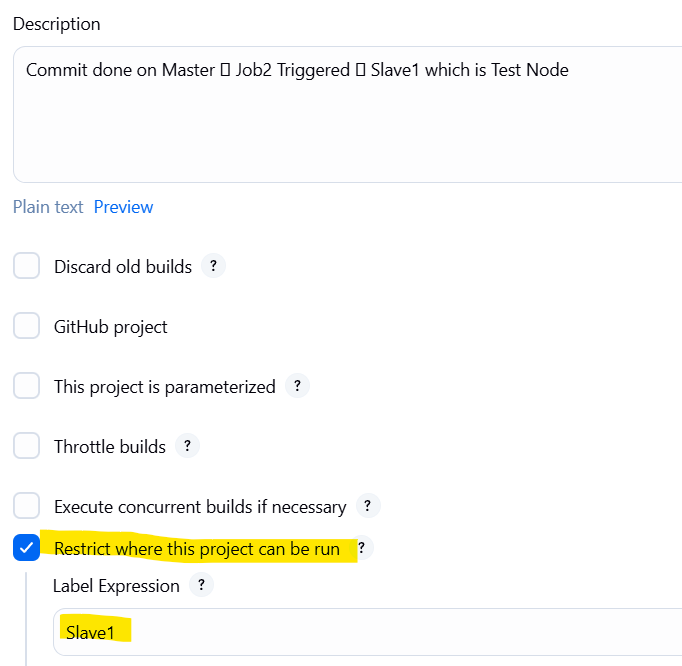
We need to change it every time a Jenkins server restarts, as the public IP changes. If the DNS URL is used, this issue and rectification is not required.

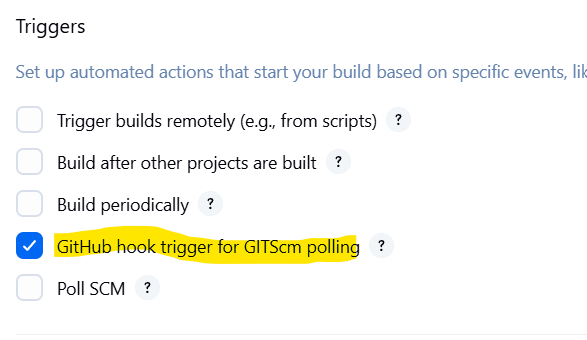
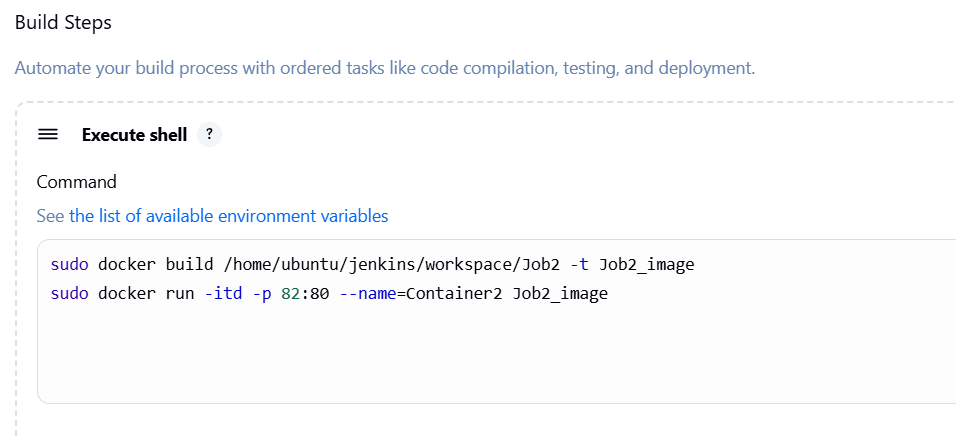
Now, creating another job “Job2” satisfying below condition:

If a commit is made to the master branch, test and push to prod

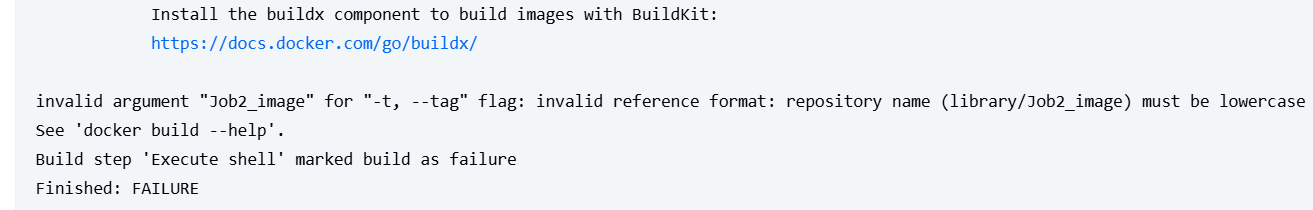
**Job2 > Slave1 (Test)> master**

**Commit** done on **Master** 🡪 **Job2** Triggered 🡪 **Slave1** which is **Test Node**

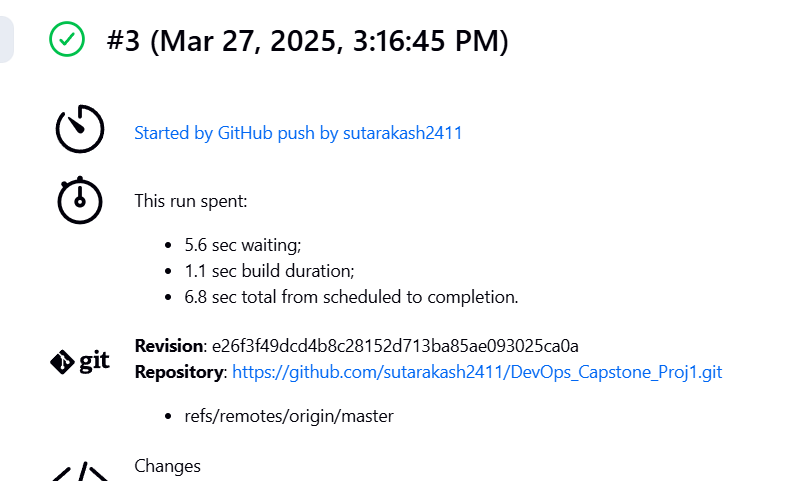
Testing job2 : failed due to following reason:



Rectification:



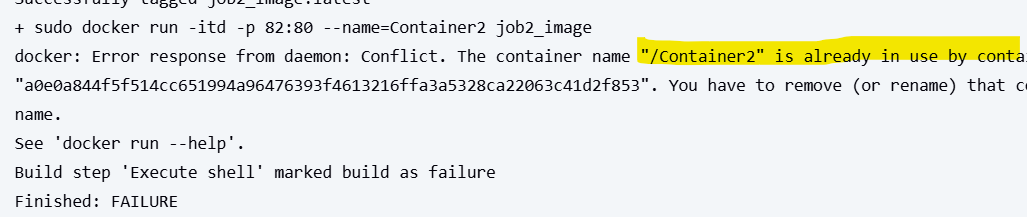
Post commiting the changes to github, job ran successfully





If another commit is made, webhook would trigger another job

The job would fail since the container with same name already exists, hence the already existing container should be deleted.

  
Updating the shell script as follow:



Doing another commit to validate:



