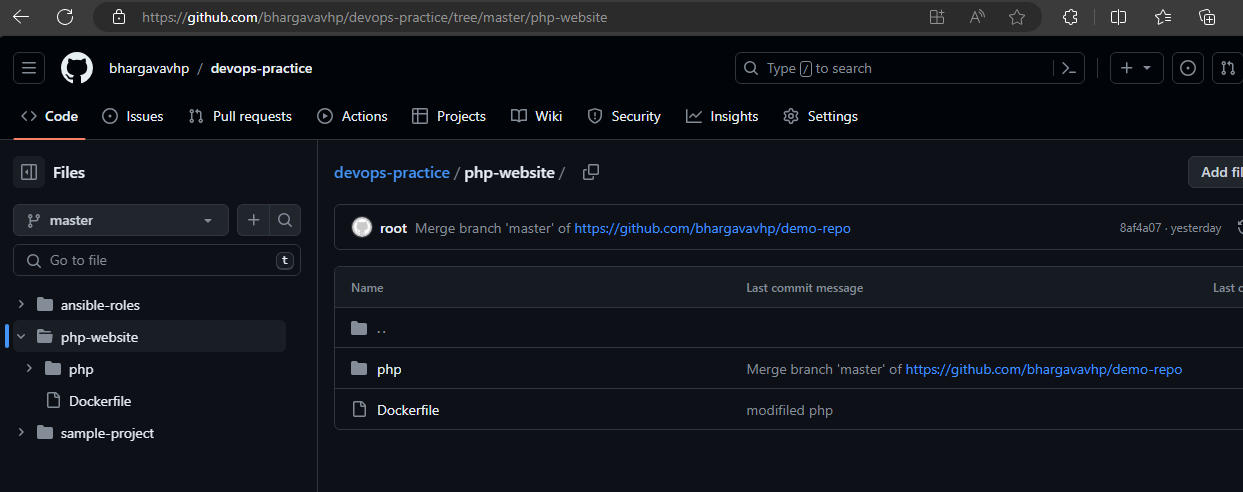
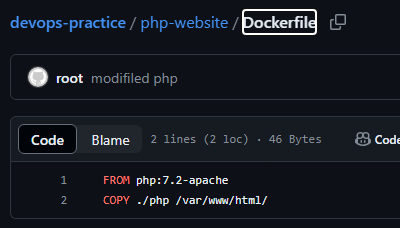
**Name: Bhargava Varanasi**

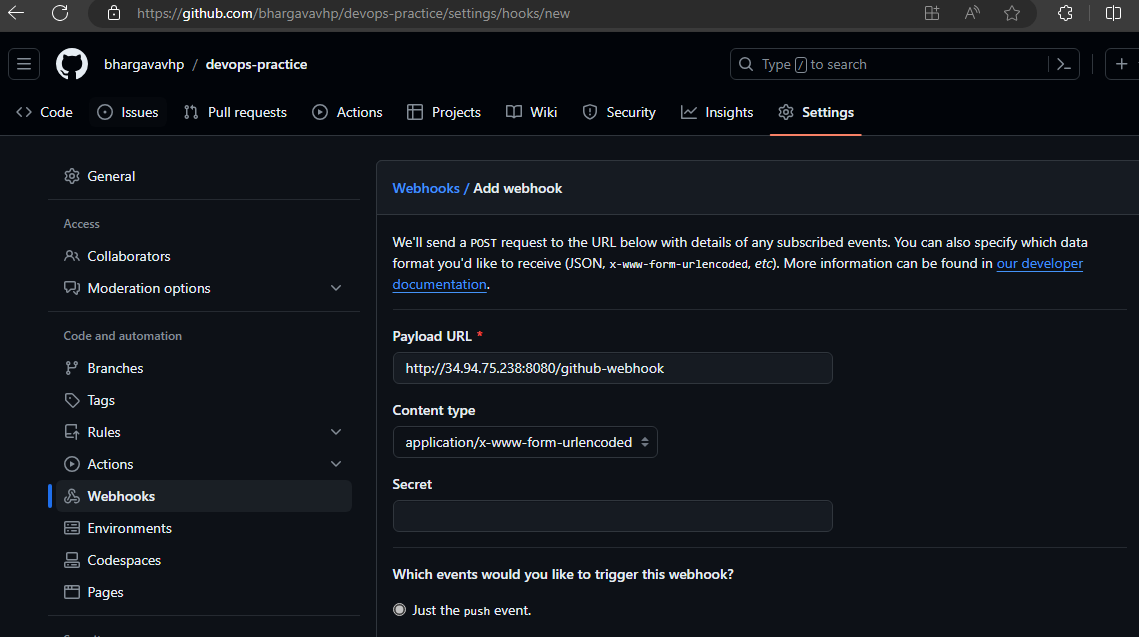
**Certification Project – 1:**

* Use the Master VM for Jenkins, Ansible, GIT etc.
* Use the fresh instance for Jenkins Slave Node (Test Server)
* Change the IP address of the VMs accordingly
* Add Build Pipeline Plugin and Post-build task plugin to Jenkins on the master VM
* Install python, openssh-server and git on the slave node manually
* Use the image devopsedu/webapp and add your PHP website to it using a Dockerfile
* Push the PHP website, and the Dockerfile to a git repository





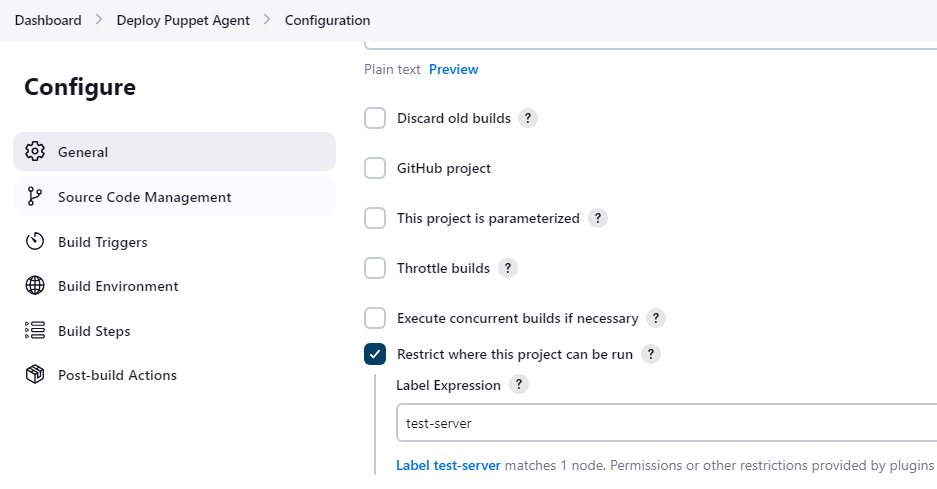
Configure github webhook for the project



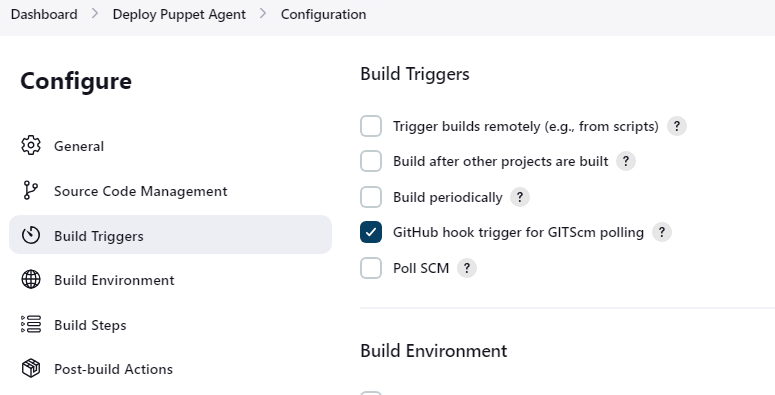
Below tasks should be automated through Jenkins by creating a pipeline:

**1.Install and configure puppet agent on the slave node (Job 1)**

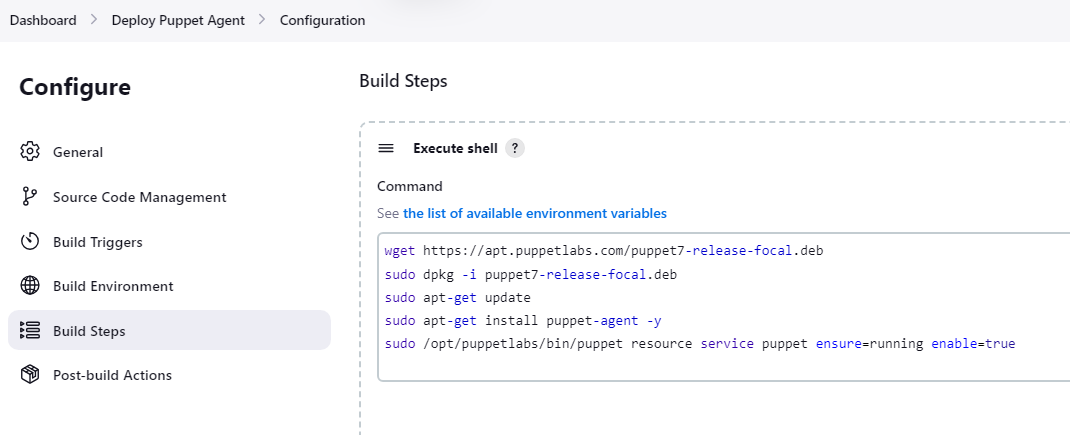
Test-server Jenkins slave is added to Jenkins master, “Deploy Puppet Agent” job is created.

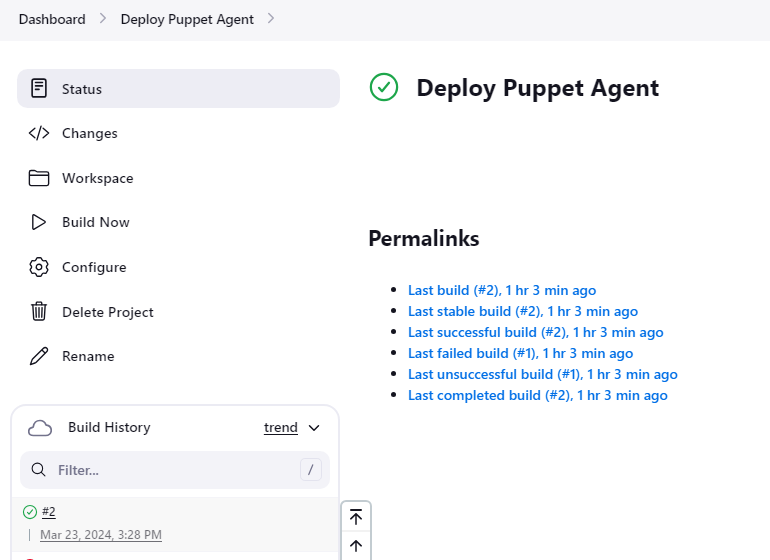


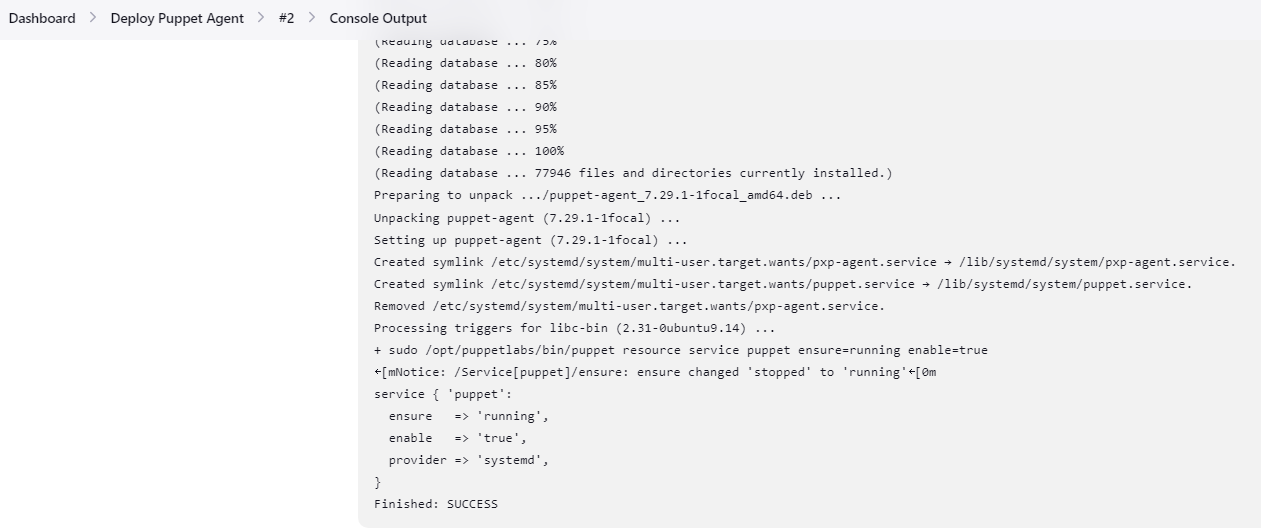
Trigger this job using github webhook

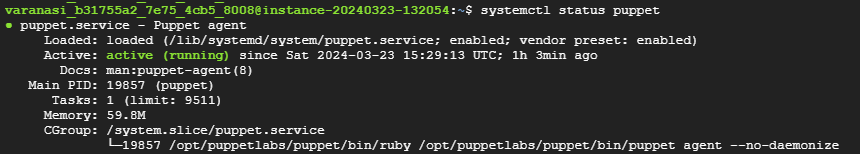


Execute commands to install puppet-agent on test-server node



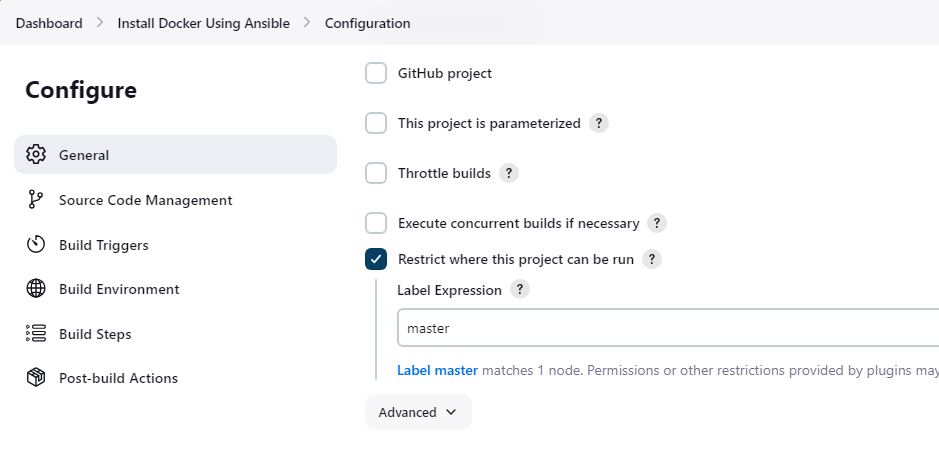


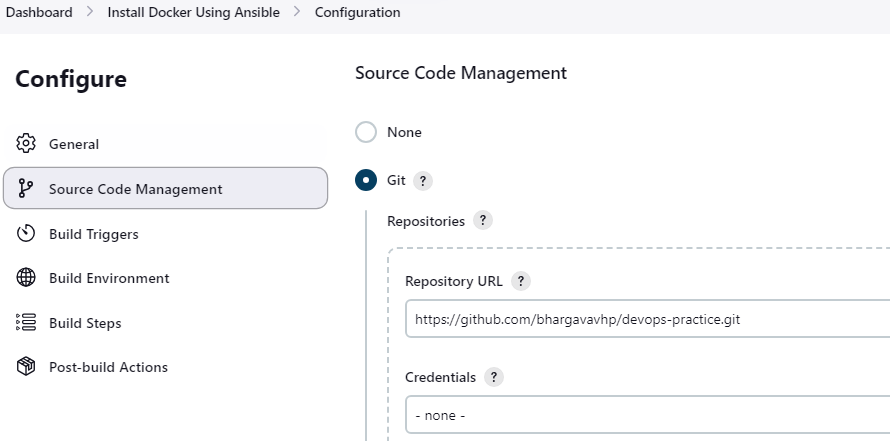




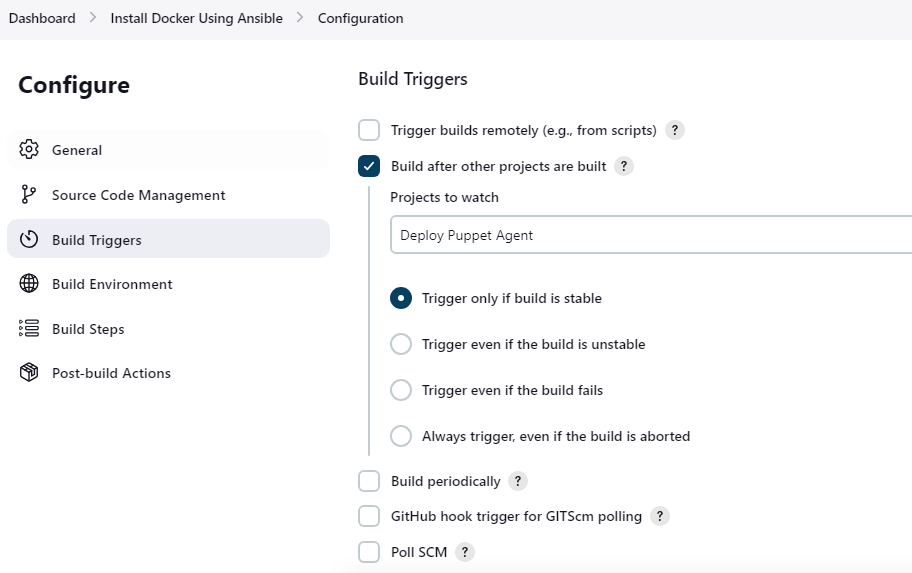
**2.Push an Ansible configuration on test server to install docker (Job 2)**

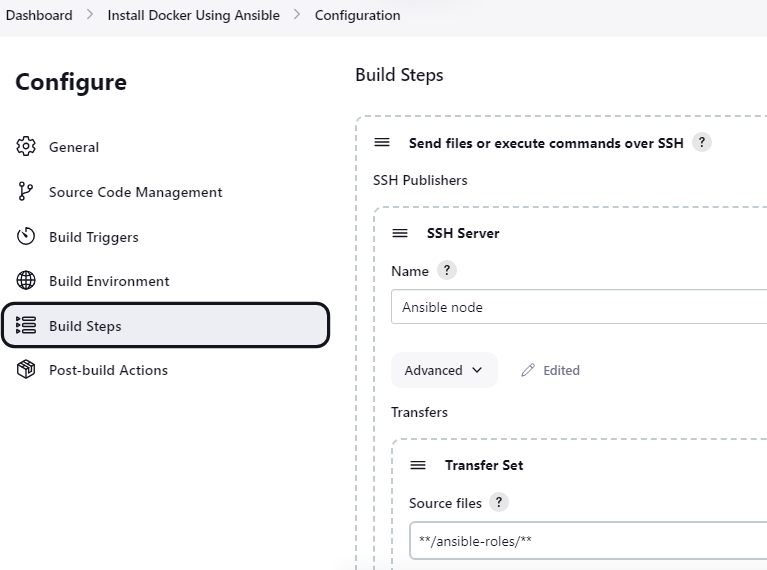
“Install Docker Using Ansible” job is created





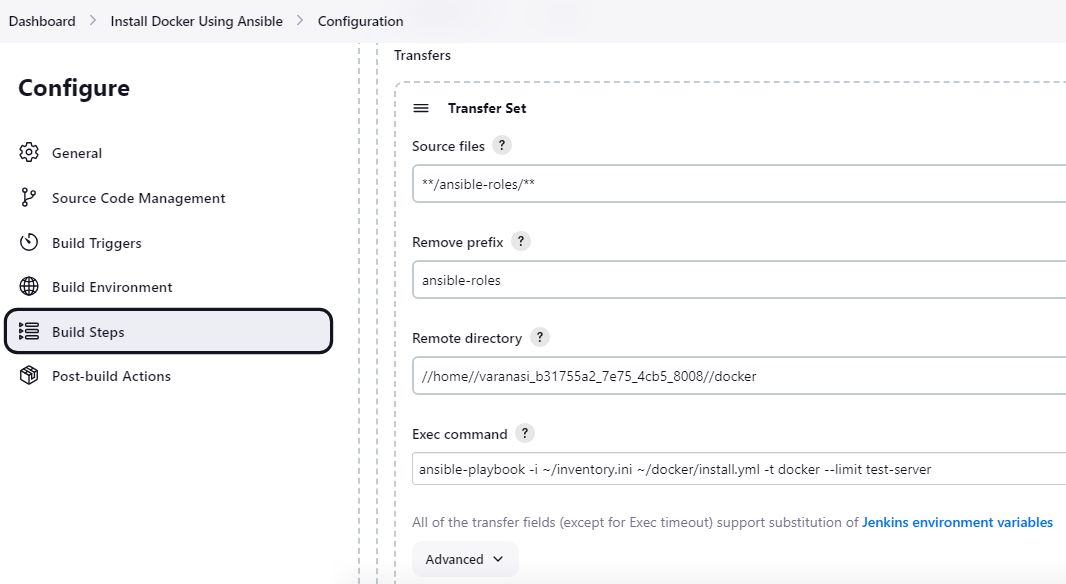
Build this job after “Deploy Puppet Agent” is successful

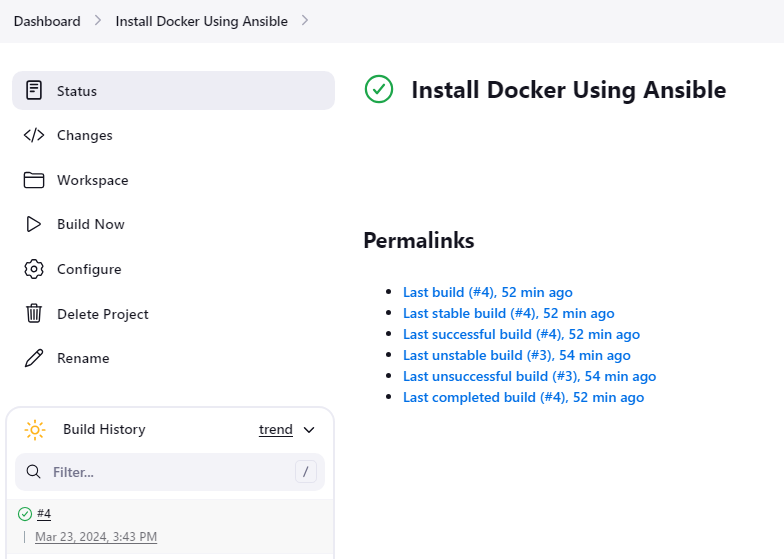


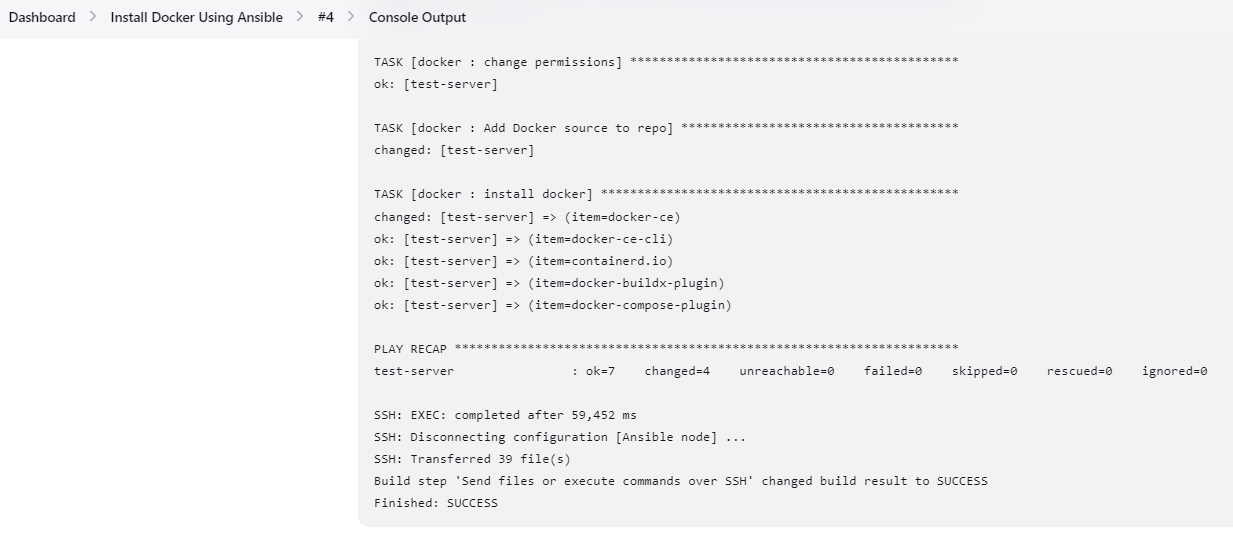


Trigger ansible playbook to deploy docker tool on test-server

Code at - [devops-practice/ansible-roles/docker/tasks/main.yml at master · bhargavavhp/devops-practice (github.com)](https://github.com/bhargavavhp/devops-practice/blob/master/ansible-roles/docker/tasks/main.yml)



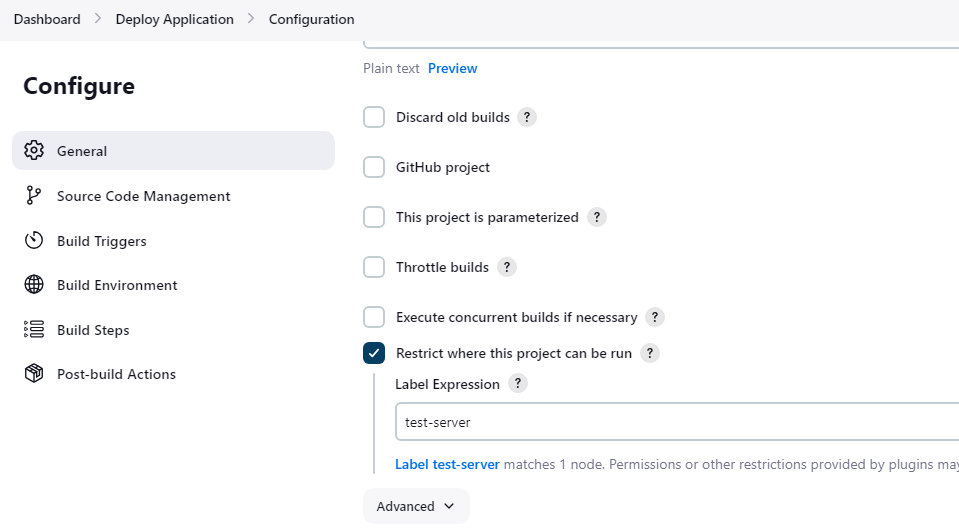


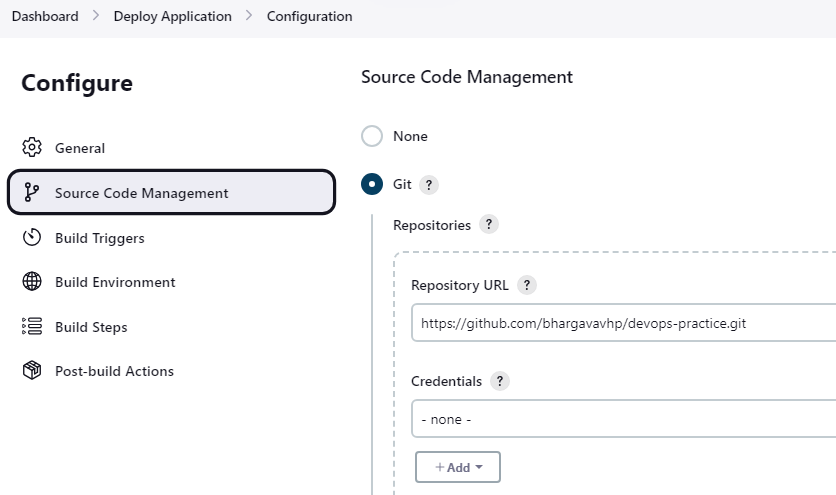




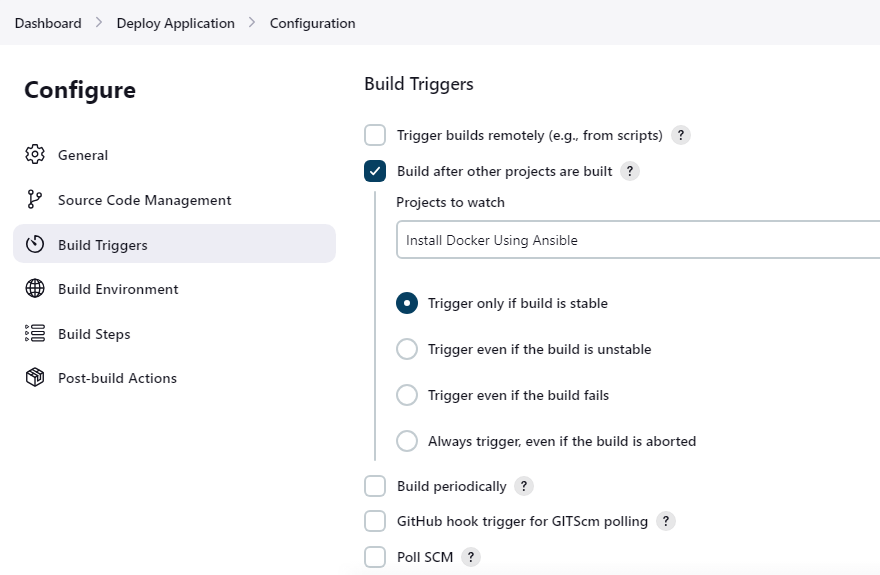
**3.Pull the PHP website, and the Dockerfile from the git repo and build and deploy your PHP docker container. After. (Job 3)**

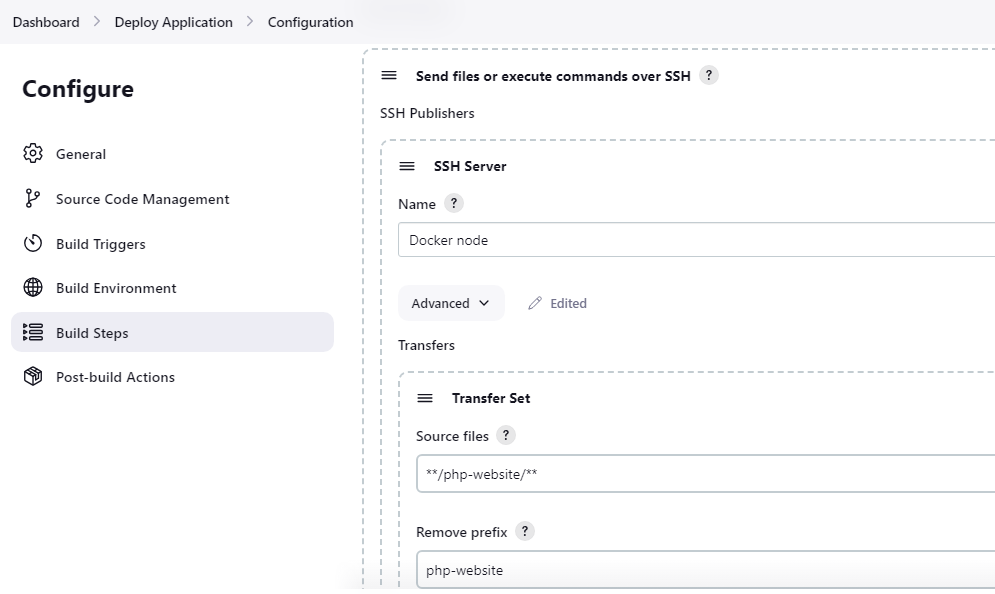
“Deploy Application” job is created



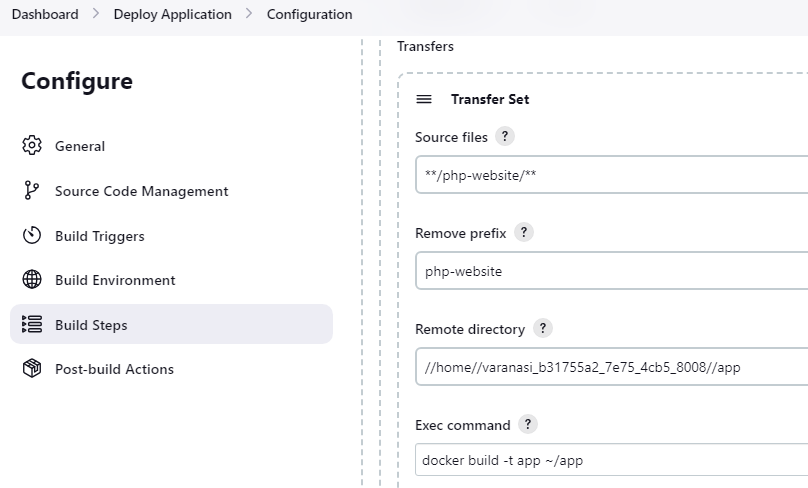


Build this after “Install Docker Using Ansible” job

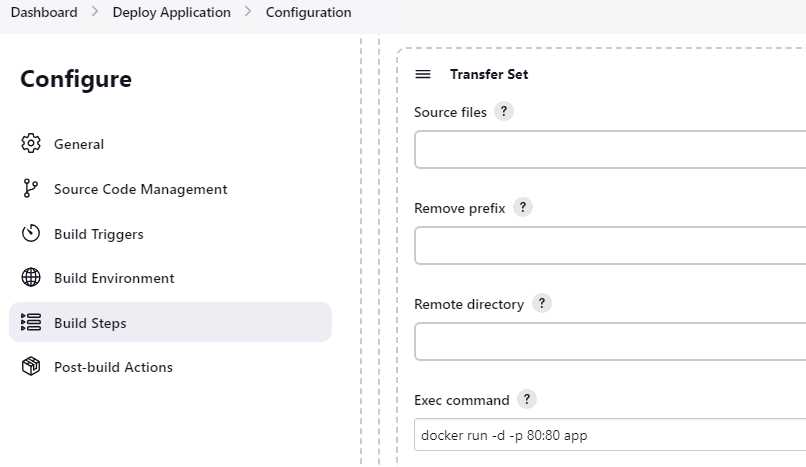


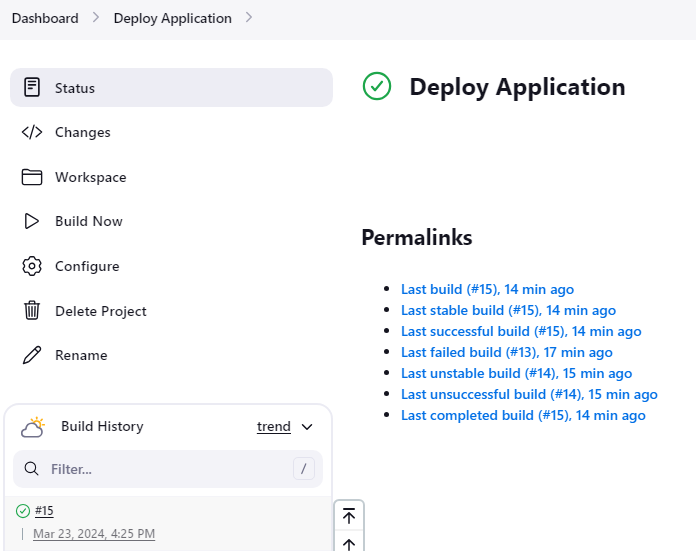


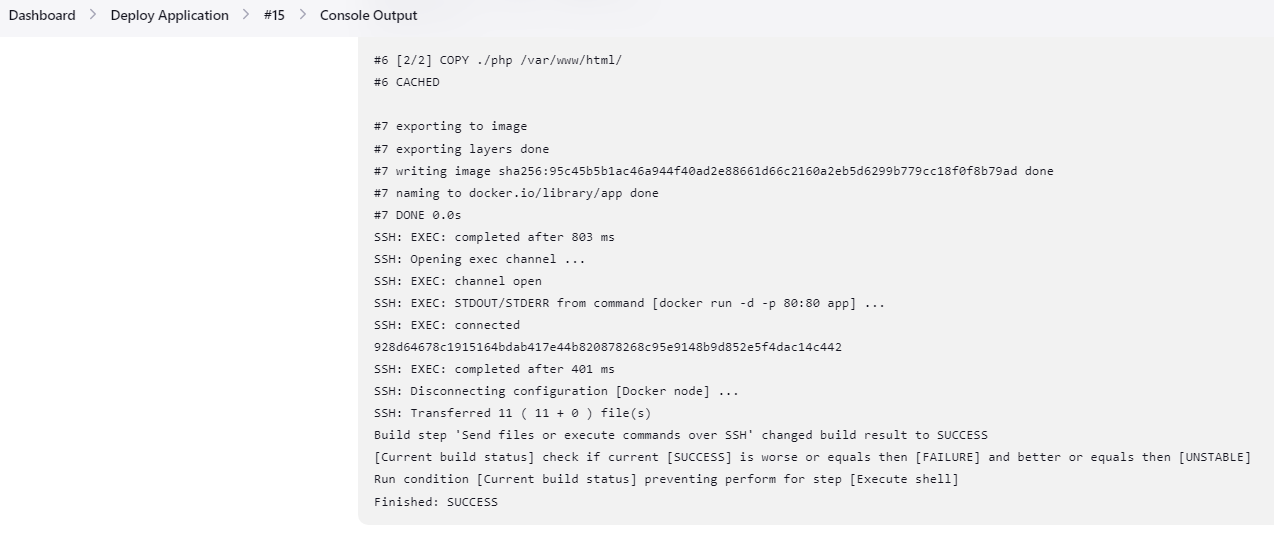
Build docker image “app”

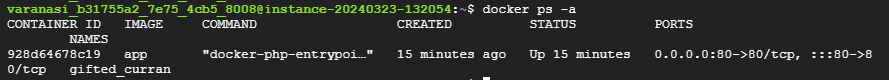


Run docker container on test-server and expose it to port 80





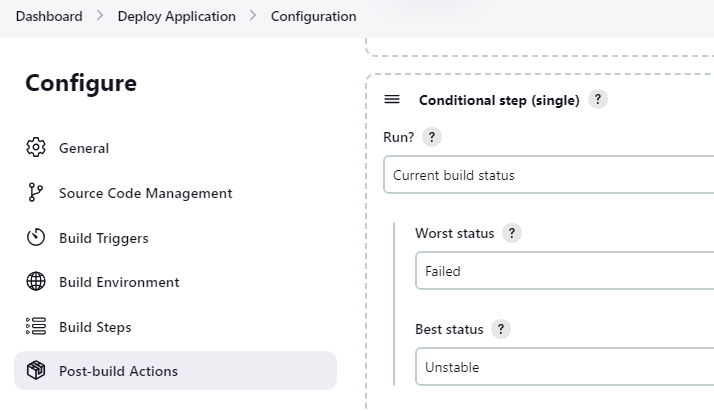


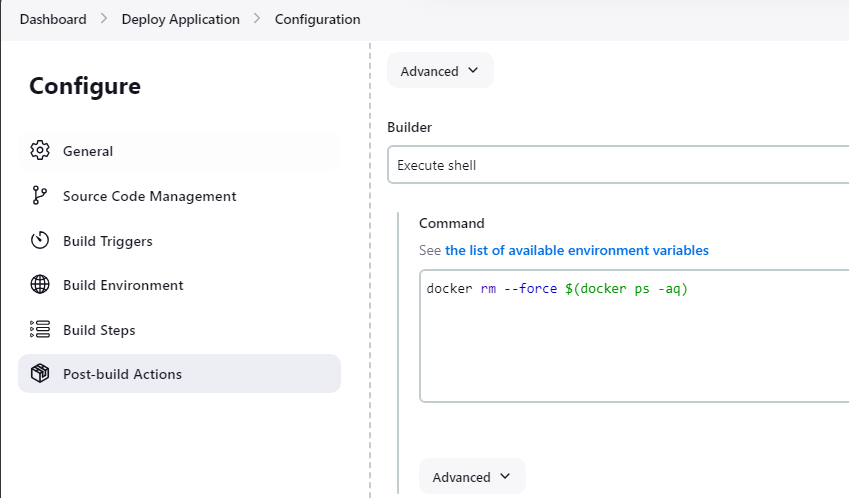


Application deployed and accessible on port 80 of test-server ip



**4.If Job 3 fails, delete the running container on Test Server**





**Built a pipeline to trigger jobs automatically**

