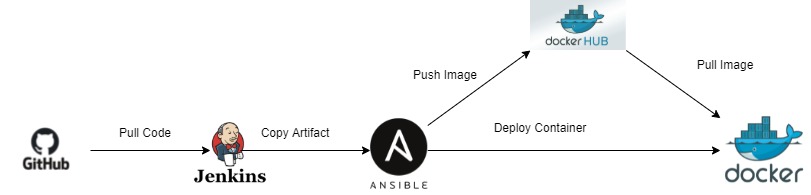
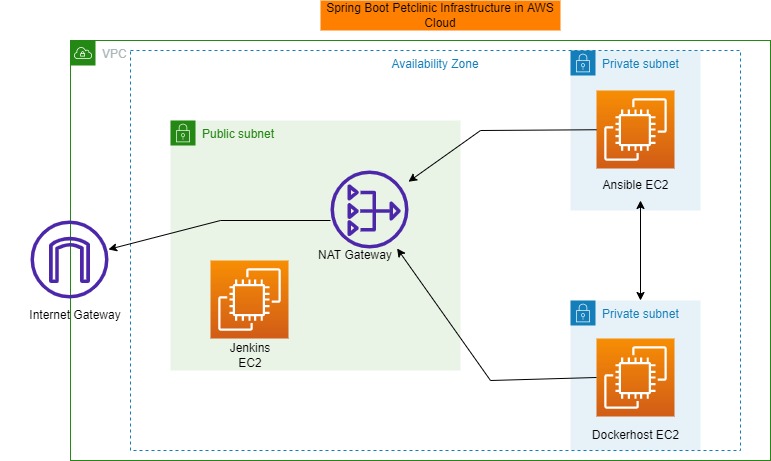
**Pet-Clinic Spring Boot Assignment Documentation**

**1# Assignment Technical Workflow**

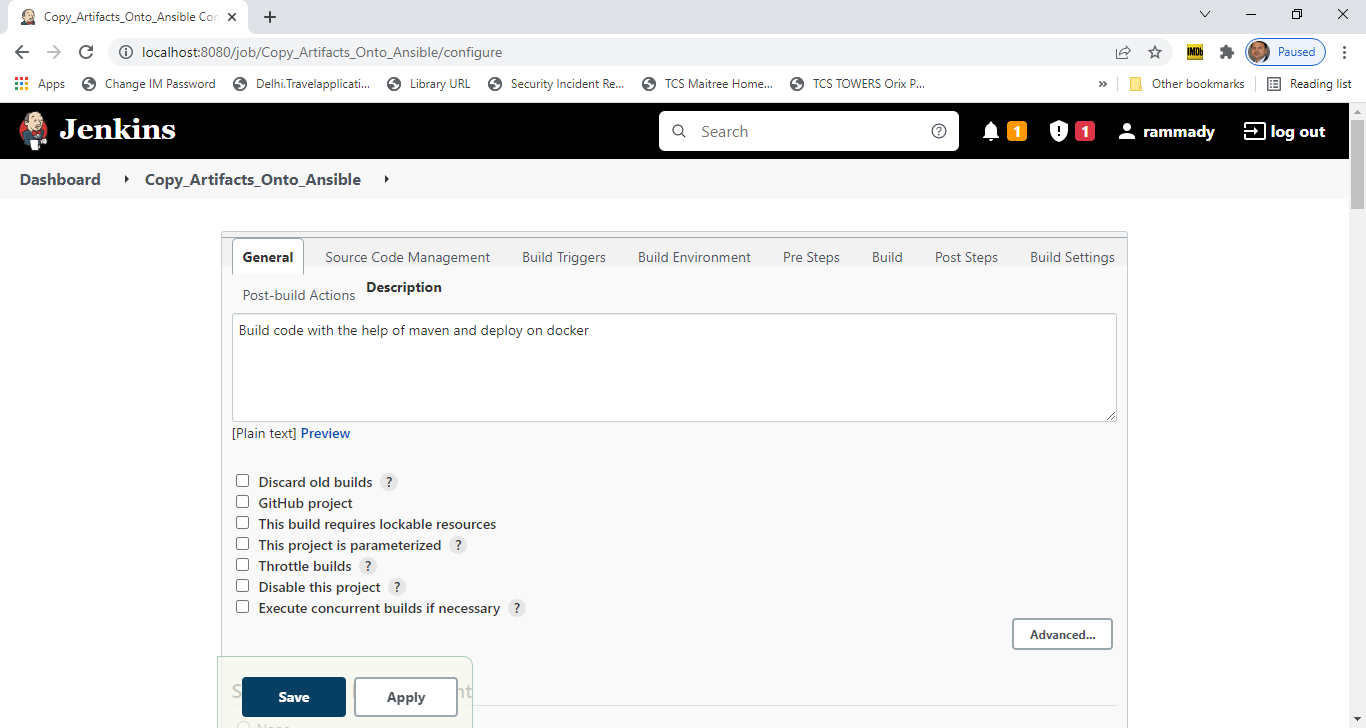


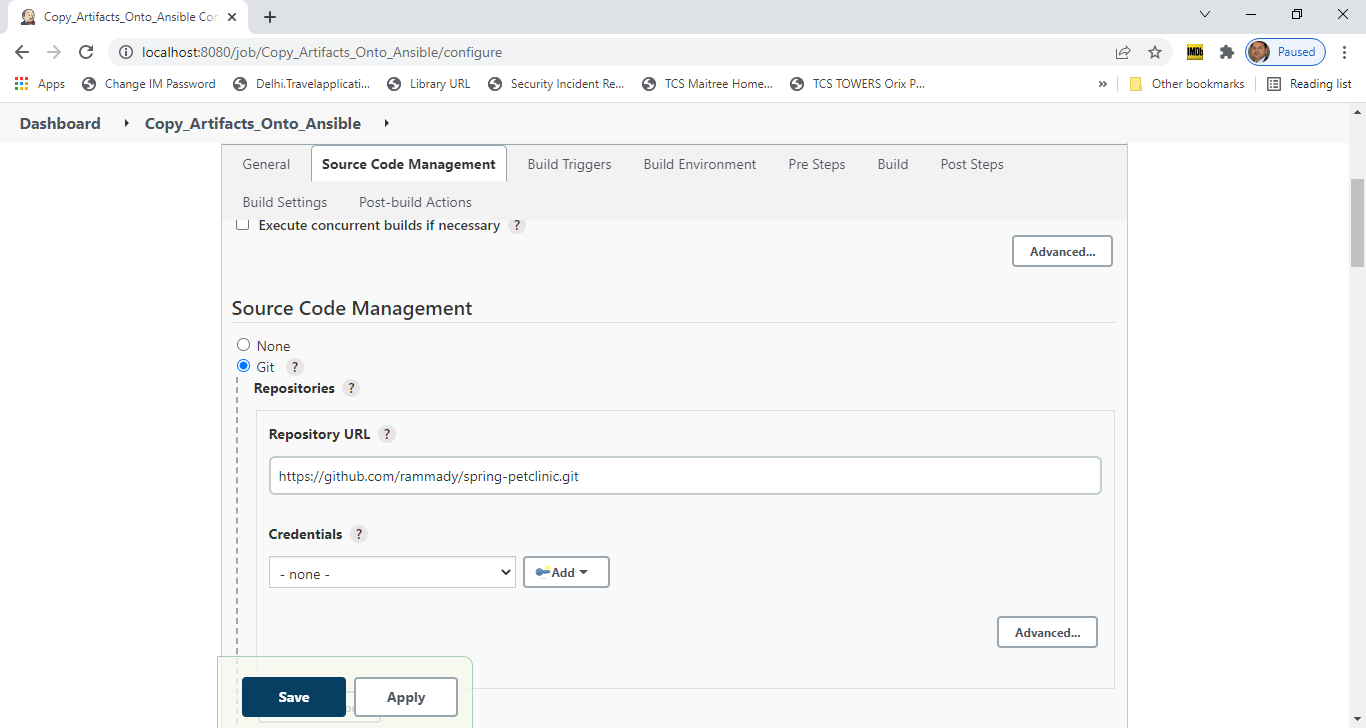
**2# Infrastructure creation using Terraform**

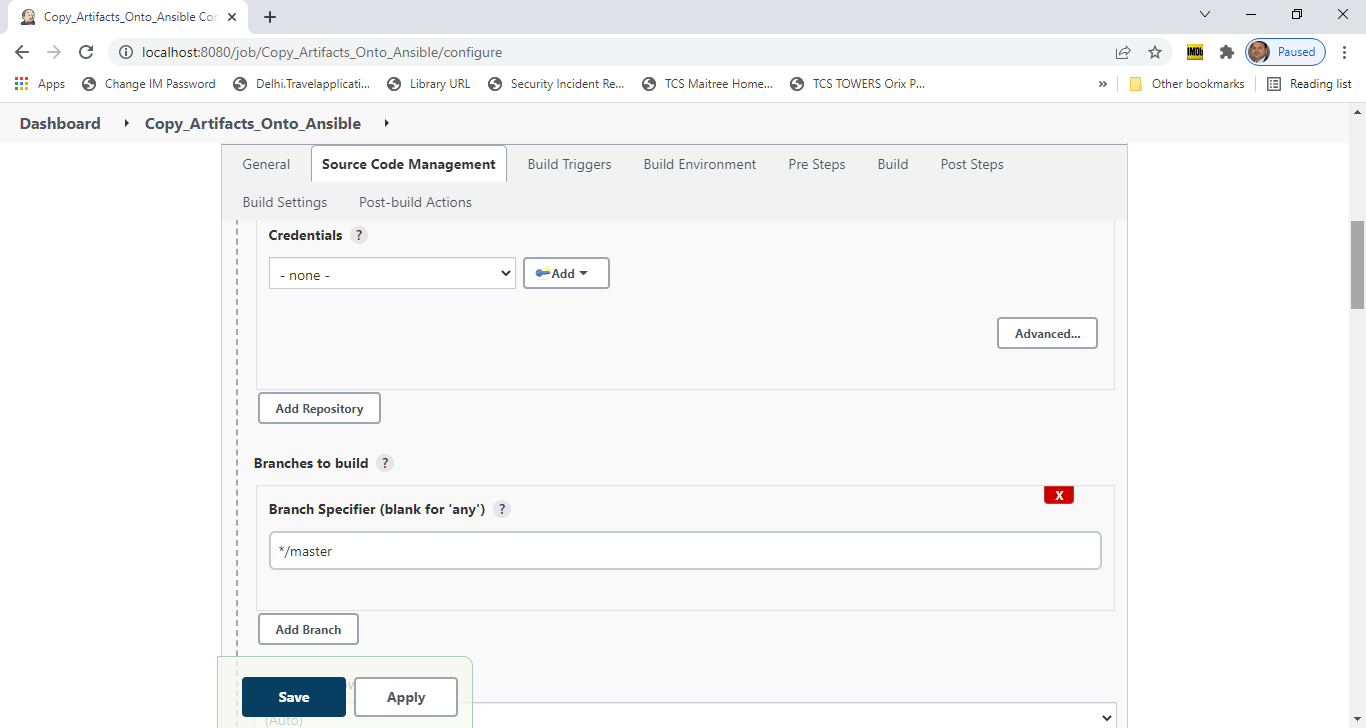
****

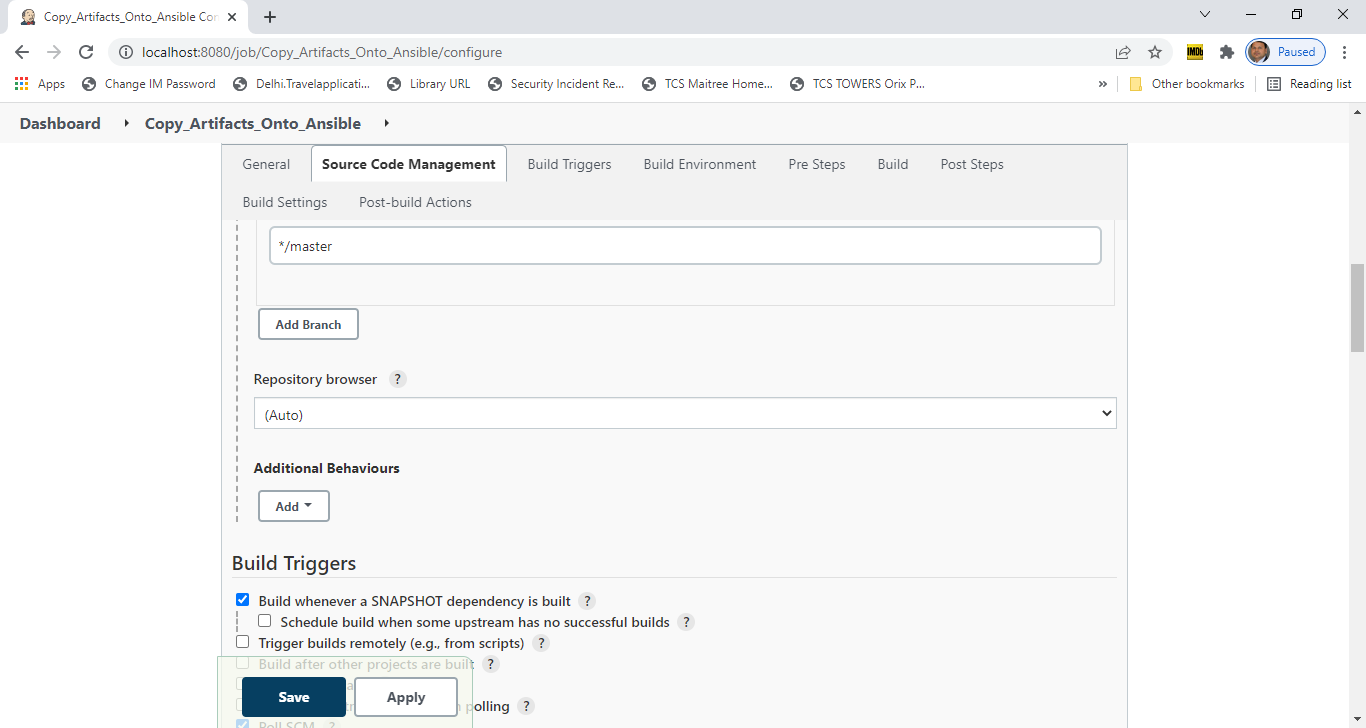
Terraform will create 1 VPC and 3 EC2 instances, 1 Public subnet and 2 private subnets, NAT gateway, IG gateway, security groups etc

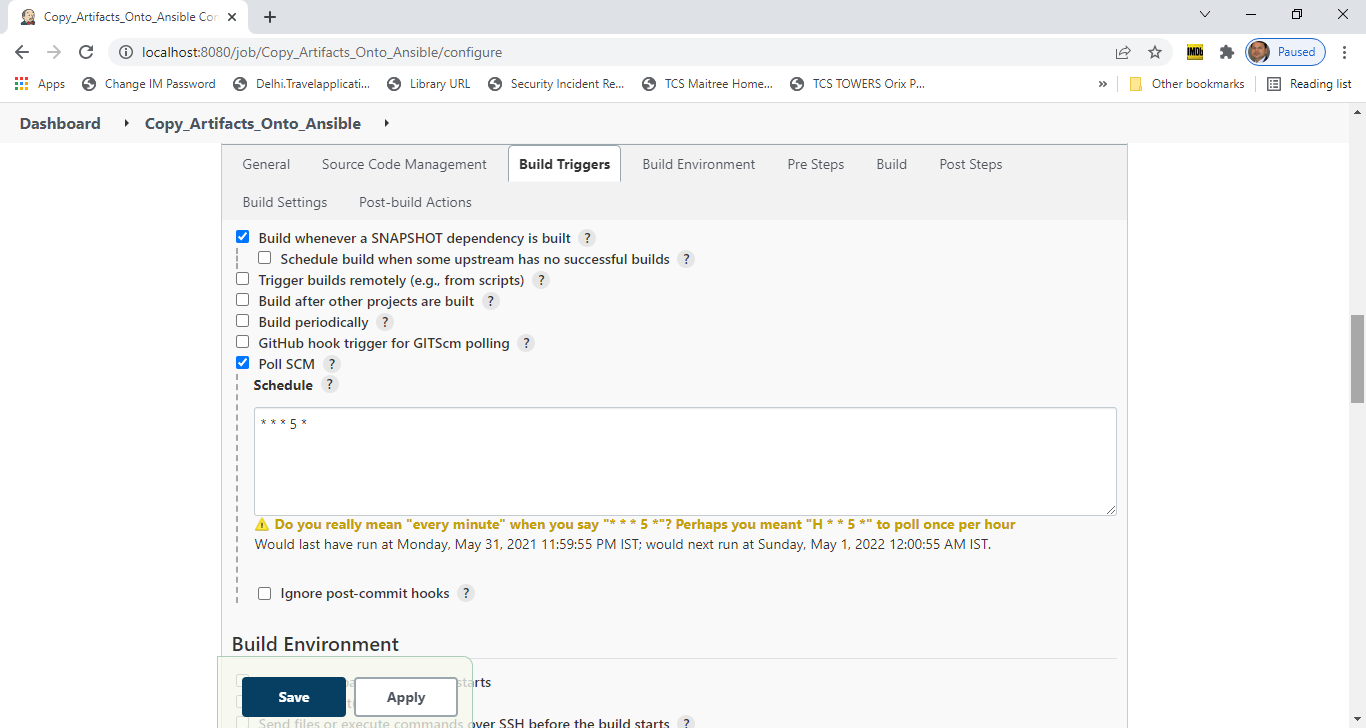
**2# Jenkins Job:**  
Jenkins job will create the war file using maven build and Ansible playbook will copy the war file from using Ansible EC2 machine to the dockerhost EC2 machine and will run the application in a docker which will having tomcat and petclinic spring boot application.

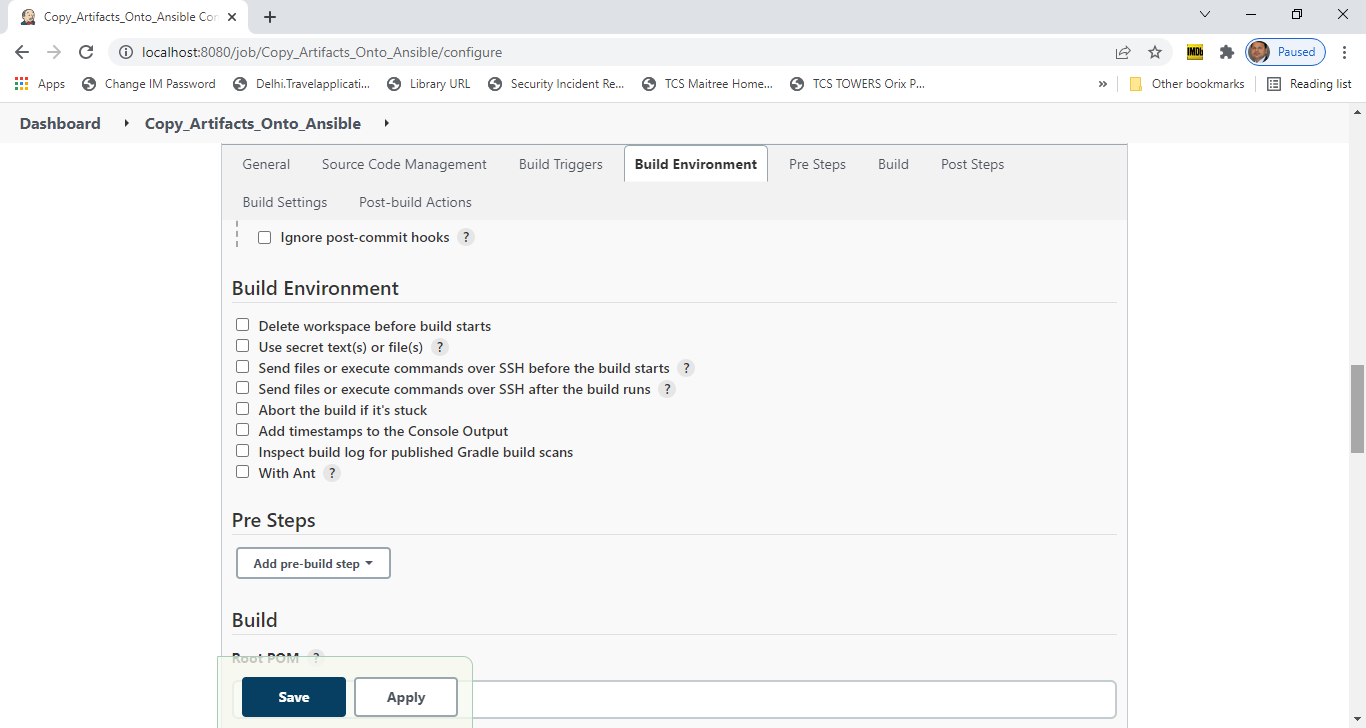


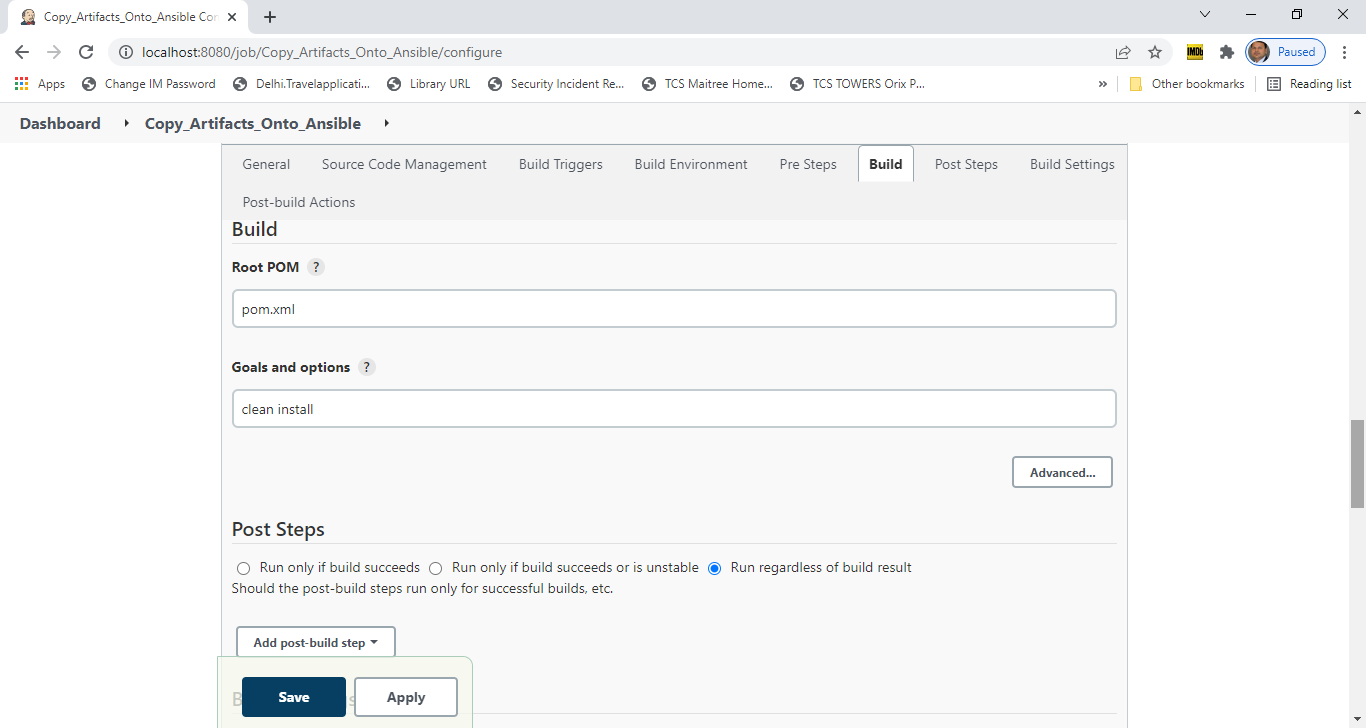


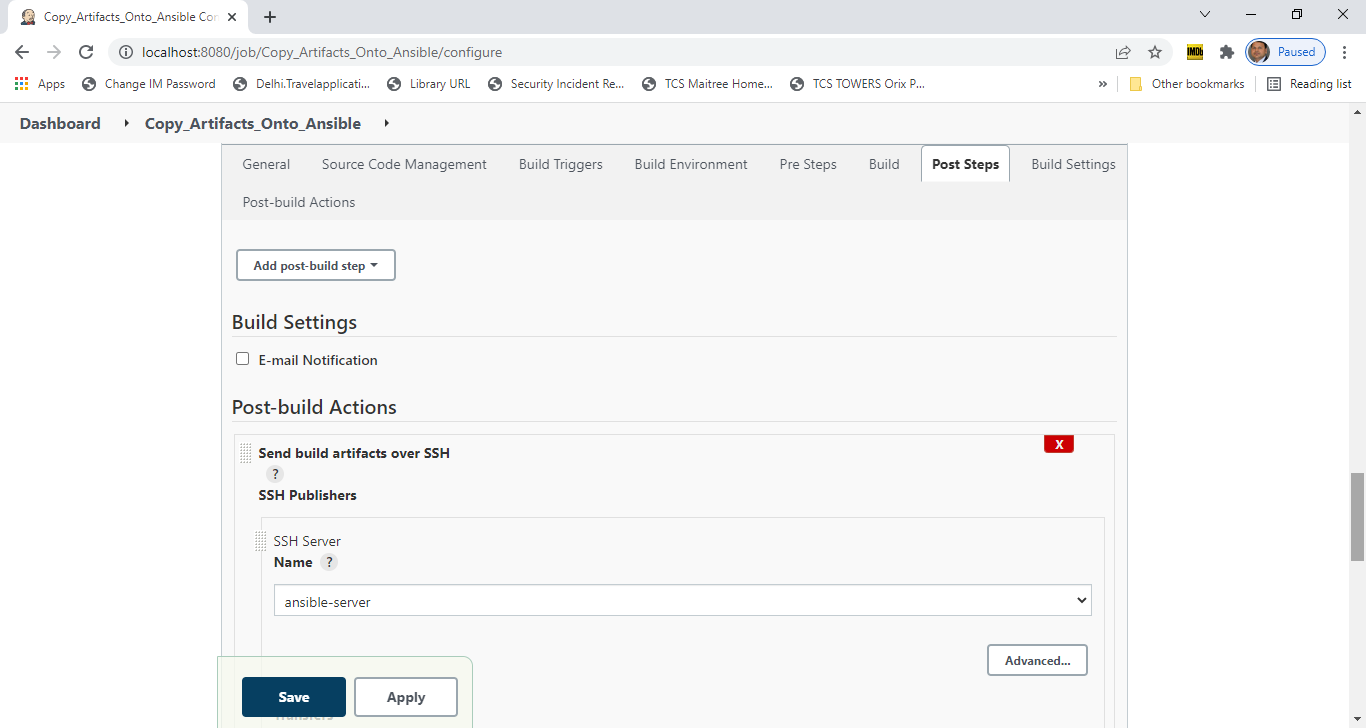


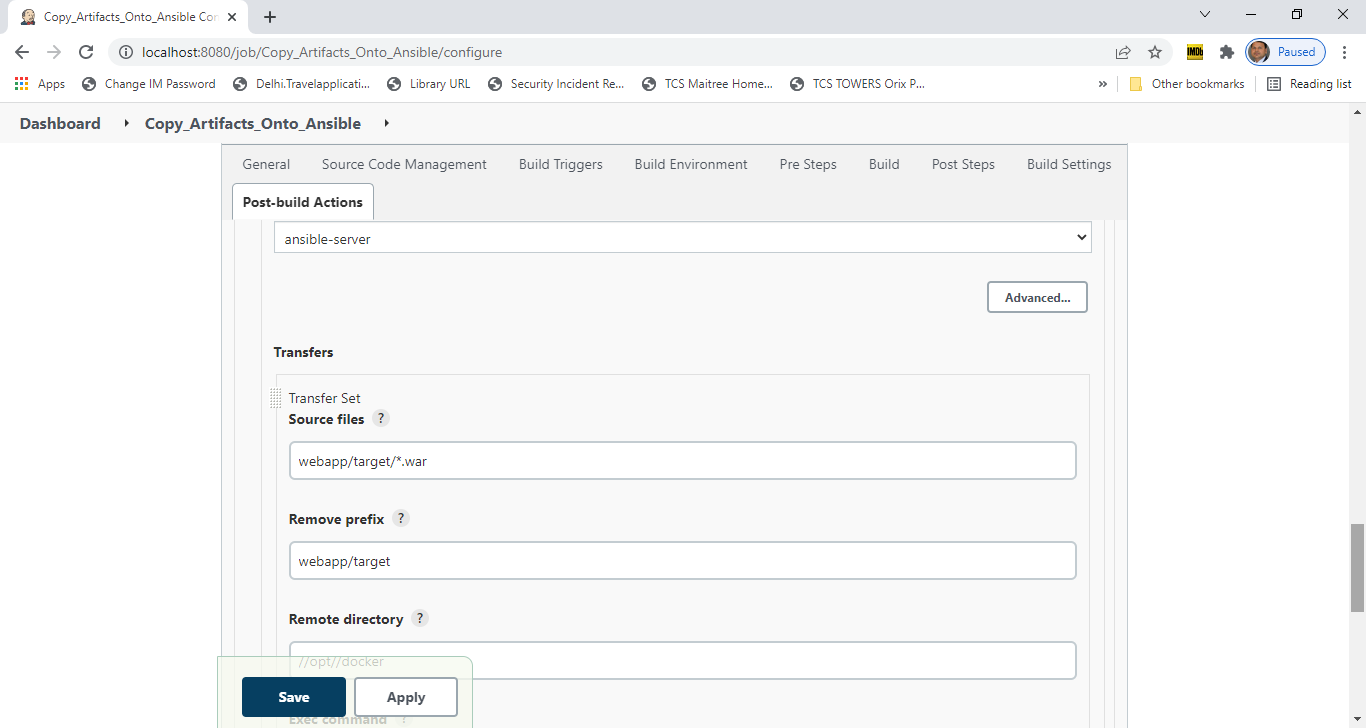


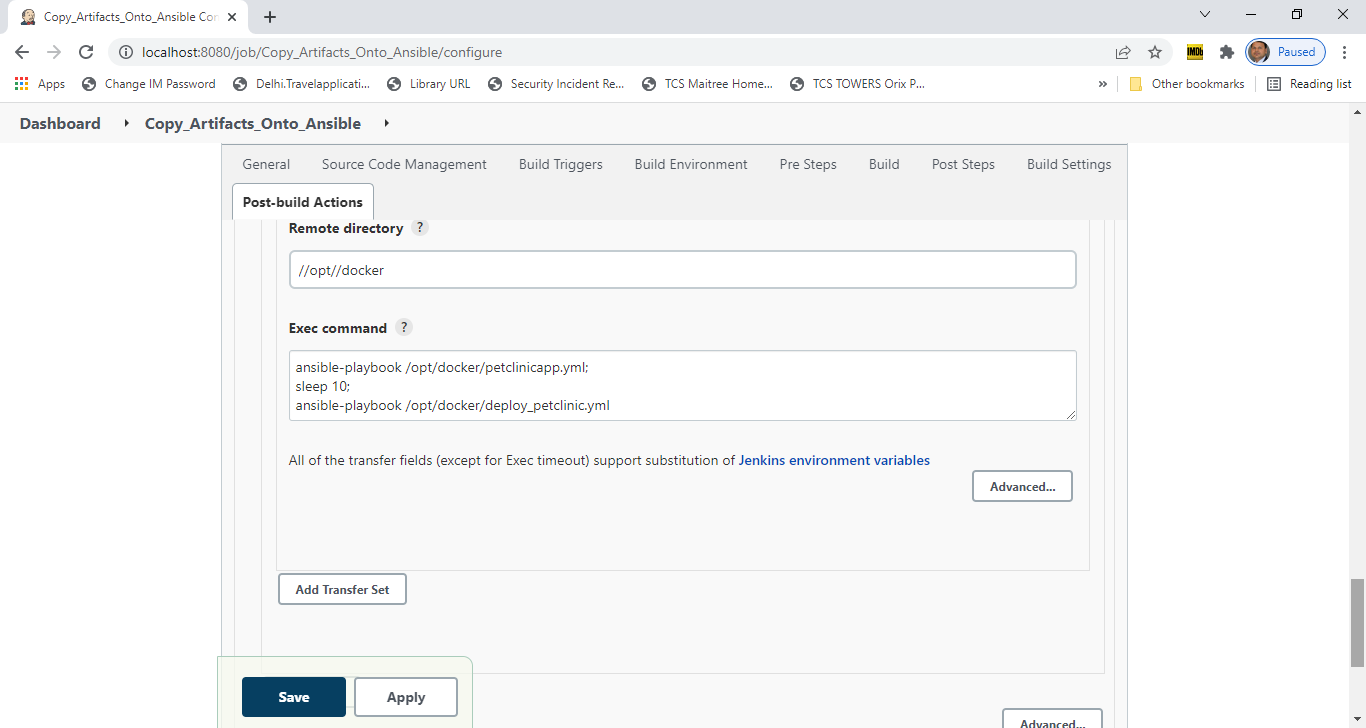


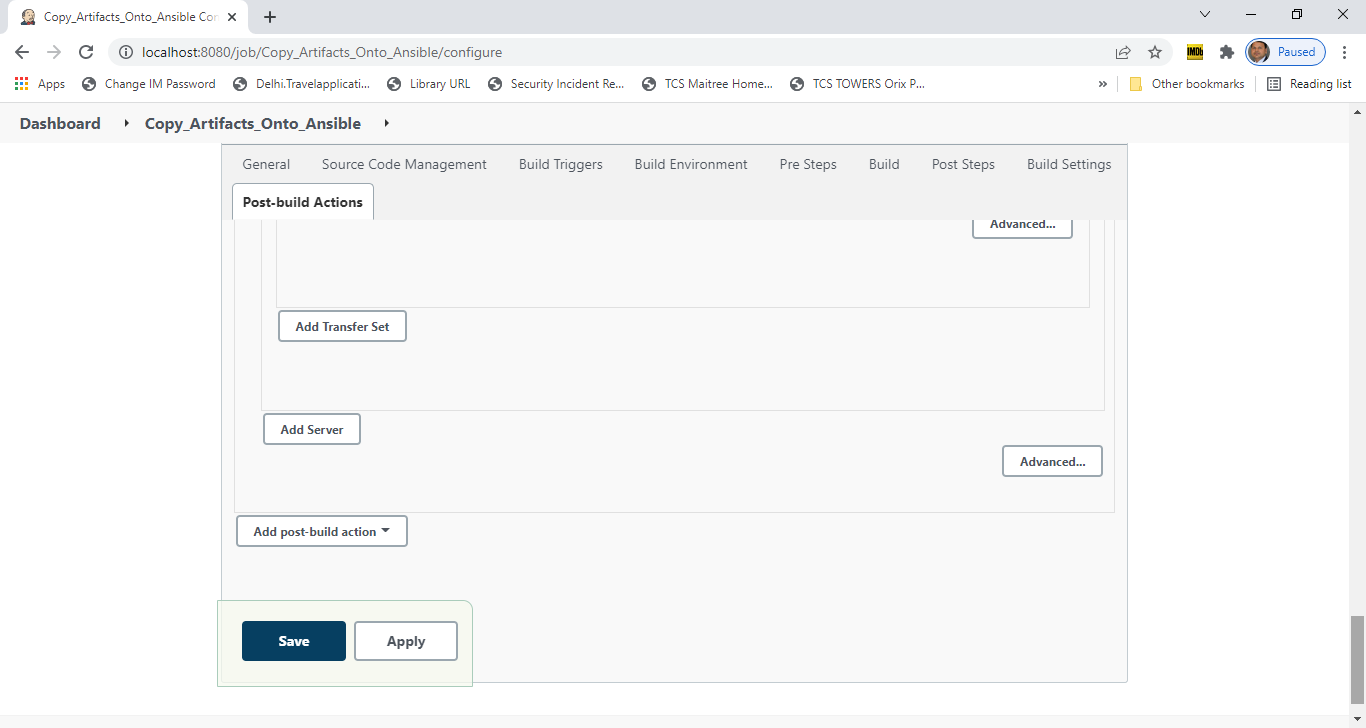












3# Runbook of this assignment

1# create infra using terraform

It contains 1VPC, 2 EC2 instance , one for Jenkins , one for Ansible and one for dockerhost machine where application is running in docker container.

2# Jenkins EC2 instance will host the Jenkins

3# ANSIBLE Ec2 instance

(i)- Ansible is going to take the war file and with the help of Dockerfile create the docker

image and this docker image we can commit into the dockerhub .

(ii)- Now here rather than deploying the war on the plain vm ,deploying it on the container

using pushed image.

(iii)- To connect to the target server [dockethost EC2 instance] create public and private key

using ssh-keygen and copy public key(id\_rsa.pub key) to the target machine

location where I should copy the public key - In user's home directory we will have .ssh

and under that we should copy the public key in the athorized\_key

user\_nam/.ssh/authorized\_key

i - Create ansadmin user

$useradd ansadmin

$passwd ansadmin

ii - Add user to sudoers file

$visudo

ansadmin ALL=(ALL) NOPASSWD: ALL

iii - Generate ssh keys for ansadmin user

$ sudo su - ansadmin

ansadmin@ansible-server$ ssh-keygen

iv - Enable password based authentication/login

$vi /etc/ssh/sshd\_config

uncomment - PasswordAuthentication yes

commented - PasswordAuthentication no

$ service sshd reload

4#Integrate dockcerhost with jenkins

i - create dockeradmin user on the dockethost machine and add this user to docker group

ii - Copy artifact from Jenkins to dockerhost -- for this Install "Publish over ssh" plugin

iii - Add dockerhost to Jenkins "configure system" so that we can communicate with the

dockerhost from the Jenkins

On the dockerhost machine

$useradd dockeradmin

$passwd dokeradmin

$usermod -aG docker dockeradmin

enable password based authentication [By default EC2 istance doesn't support password

based authentication]

$ vi /etc/ssh/sshd\_config

uncomment - PasswordAuthentication yes

comment - PasswordAuthentication no

$ service sshd reload

5# Jenkins job to build and copy artifcats on to dockerhost

6# Integrate dockerhost with Ansible

(i)- Add docker host to Ansible as a managed node so that our ansible control node can ableto

manage our Docker host .

(ii)- To image get build , we need to initialize the docker host to create a container, that we

are going to do with the Ansible control node.

I mean we are going to write a playbook , that playbook is going to tell our Docker host

how to create a container. For that of course, we need to add our Docker host [second

vm] as a client/slave to our Ansible system.

1# On the dockerhost -Activities to be performed

(i) - cerate ansadmin [with ansadmin user ansible is going to manage our docker host]

$useradd ansadmin

$passwd ansadmin

(ii) - Add ansadmin to sudoers files

$visudo

ansadmin ALL=(ALL) NOPASSWD: ALL

(Iii)- Enabled password based authentication

$vi /etc/ssh/sshd\_config

uncomment - PasswordAuthentication yes

commented - PasswordAuthentication no

$ service sshd reload

2# On Ansible Node -Activities to be performed

(i) - Add the docker host IP address in the inventory file

$vi /etc/ansible/hosts

(ii) - Add Ansible servicer's public key to the docker host's ansadmin user [ansadmin –

/.ssh/authorized\_key] so that password based authentication get enabled

$sudo su - ansadmin

$ssh-copy-id <target-system-ip-address> [it will create a autorized\_keys file in .ssh

directory on dockerhost system, it will contain the public key of the ansible VM]

7# Machine EC2 where Ansible is installed we need to install docker [docker get installed using

Terraform user\_data]

(i)- Add ansadmin to the docker group then only we can execute docker commands as

ansadmin otherwise it will throow errors

$sudo usermod -aG docker ansadin

$id andadmin --- it should belongs to docker group

$systemctl start docker.service

(ii)-Create docker image and for this we need Dockerfile

FROM tomcat:latest

RUN cp -R /usr/local/tomcat/webapp.dist/\* /usr/local/tomcat/webapps

COPY ./\*.war /usr/local/tomcat/webapps

8# Ansible playbook to create image and container

On the Ansible machine

$cd /opt/docker

Dockerfile webapp.war

made entry n the inventry file - /etc/ansible/hosts

$vi /etc/ansible/hosts

[dockerhost]

172.31.21

[ansible]

172.31.30.109

create ansible file

$pwd

/opt/docker

$vi petclinicapp.yml

---

- host: ansible

tasks:

- name: create docker image

command: docker built -t petclinicapp:latest .

args:

chdir: /opt/docker

9# Copy image onto dockerhub

On the Ansible server

$pwd

/opt/docker

$docker login

Tag the image

$docker tag <image-id> rammady/peclinicapp:latest

$docker push <image-name>:latest

$docker push valaxy/petclinicapp:latest

From dockerhub dockerhost going to pull this image and create container out of IT.

10# Ansibe playbook to build, package and deploy

On Ansible server

$pwd

/opt/docker

$ vi petclinicapp.yml

---

- host: ansible

tasks:

- name: create docker image

command: docker built -t petclinicapp:latest .

args:

chdir: /opt/docker

- name: create tag to push image onto dockcerhub

command: docker tag petclinicapp:latest rammady/petclinicapp:latest

- name: push docker image

command: docker push rammady/petclinicapp:latest

11# How to create container on dockerhost using ansible playbook

Login into ansible system using anadmin user

$pwd

/opt/docker

Craete a new ansible playbook

$vi deploy\_petclinicapp.yml

---

- host: dockerhost

tasks:

- name: stop existing container

command: docker stop petclinicapp-server

ignore\_errors: yes

- name: remove the container

command: docker rm petclinicapp-server

ignore\_errors: yes

- name: remove image

command: docker rmi rammady/petclinicapp:latest

ignore\_errors: yes

- name: create container

command: docker run -d --name petclinicapp-server -p 8082:8080 rammady/petclinicapp:latest

12# Jenkins CICD to deploy on container using Ansible

Add this playbook into the Copy\_Ansible\_onto\_Ansible Jenkins job's post\_build actions --exec command

ansible-playbook /opt/docker/petclinicapp.yml;

sleep 10;

ansible-playbook /opt/docker/deploy\_petclinicapp.yml

Improvement:

1# Proper System design should be done.  
2# Application Security aspects should be evaluated – Infra security, Network security, authorization and identifications, data security.

3# Federation and SSO should be evaluated.

4# Load balancer selection, DB selection, caching etc  
5# High availability of the application can be improved

6# If we think to create a distributed system design then this solution needs improvements.

7# If our container is terminated then how we will come to know that it’s not working or how we can create a new container automatically. We don’t have such kind of mechanism here that’s where container management comes into the picture.  
We can make use of K8s as container management system for high availability