DevOps Certification Training

Certification Project – Insure Me

Insurance Domain

By: Ganesh Mudavath

Insure Me is a Global leading Insurance provider based out of USA. The company offers products and services like Home Insurance, Health Insurance, Car Insurance and Life Insurances. Initially the company was using a Monolithic application architecture, As the company grown, It started facing difficulties in managing the application infrastructure and application deployments.

Insure-Me has decided to transform its monolithic application architecture to microservice application architecture and opted to go DevOps by implementing CICD pipeline and necessary automations. Insure me has decided to use AWS as primary cloud services provider to create servers, databases, and application deployments.

The company's goal is to deliver the product updates frequently to production with High quality & Reliability. They also want to accelerate software delivery speed, quality and reduce feedback time between developers and testers.

Following are the problems the company is facing at the moment

Building Complex builds is difficult

Manual efforts to test various components/modules of the project

Incremental builds are difficult to manage, test and deploy

Creation of infrastructure and configure it manually is very time consuming

Continuous manual monitoring the application is quite challenging.

In order to implement a POC, you are requested to develop a mavenized microservice using spring boot and in memory h2 database.

1. a microservice which exposes below mentioned endpoints as APIs and uses in memory h2 database to store the data. a. /createPolicy (HTTP Method : POST) (Request Body : JSON)

b. /updatePolicy/{policy id} (HTTP Method : PUT) (Request Body : JSON)

c. /viewPolicy/{policy id} (HTTP Method : GET) (No Request Body)

d. /deletePolicy/{policy id} (HTTP Method : DELETE) (No Request Body)

2. Write necessary Junit testcase.
3. Generate HTML report using TestNG.
4. Push your code into your GitHub Repository.

Note : Preload some data into the database. www.staragile.com

Later, you need to implement Continuous Integration & Continuous Deployment using following tools:
Git - For version control for tracking changes in the code files
Jenkins - For continuous integration and continuous deployment
Docker - For deploying containerized applications
Ansible - Configuration management tools
Selenium - For automating tests on the deployed web application
AWS : For creating ec2 machines as servers and deploy the web application.
This project will be about how to test the services and deploy code to dev/stage/prod etc, just on a click of button.

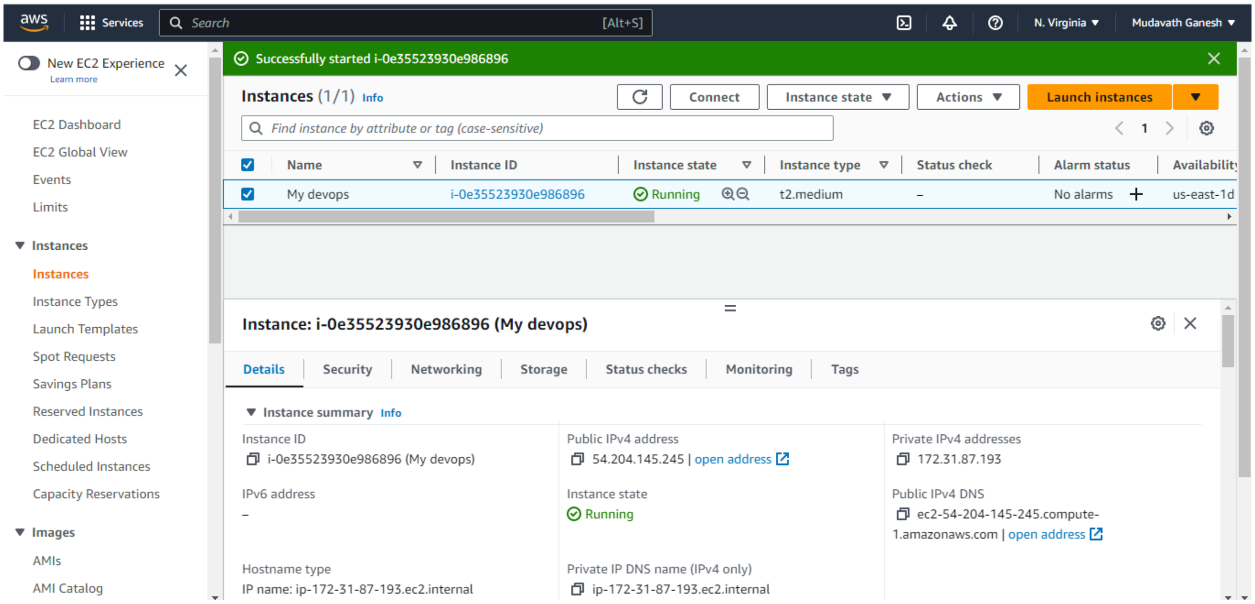
Business challenge/requirement

As soon as the developer pushes the updated code on the GIT master branch, the Jenkins job should be triggered using a GitHub Webhook and Jenkins job should be triggered, The code should be checked out, compiled, tested, packaged and containerized and deployed to the preconfigured test-server automatically.

The deployment should then be tested using a test automation tool (Selenium), and if the build is successful, it should be deployed to the prod server. All this should happen automatically and should be triggered from a push to the GitHub master branch.

Note : To have a detailed information about running the application and exposed APIs, Input/Output format, Refer to the README.md in the GitHub repository. - name: Copy file with owner and permission, using symbolic representation

```
  ansible.builtin.copy:

    src: /home/ubuntu/insuresele.jar

    dest: /home/ubuntu/insuresele.jar

    owner: ubuntu

    group: ubuntu

    mode: u=rw,g=r,o=r



 - name : run selenium jar

  command : java -jar /home/ubuntu/insuresele.jar

  become : yes

  become_user : ubuntu
```



Successfully started i-0e35523930e986896

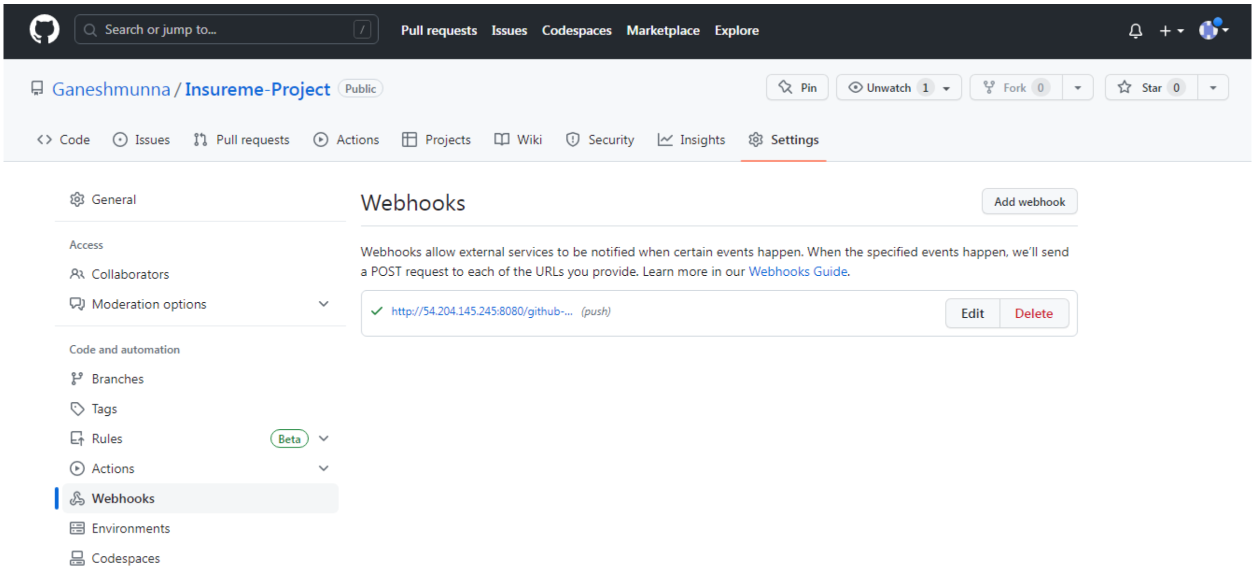
Instances (1/1) Info

☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
☑	My devops	i-0e35523930e986896	✓ Running	t2.medium	–	No alarms	us-east-1d

Instance: i-0e35523930e986896 (My devops)

Instance summary Info

Instance ID i-0e35523930e986896 (My devops)	Public IPv4 address 54.204.145.245 | open address	Private IPv4 addresses 172.31.87.193
IPv6 address –	Instance state ✓ Running	Public IPv4 DNS ec2-54-204-145-245.compute-1.amazonaws.com | open address
Hostname type IP name: ip-172-31-87-193.ec2.internal	Private IP DNS name (IPv4 only) ip-172-31-87-193.ec2.internal	

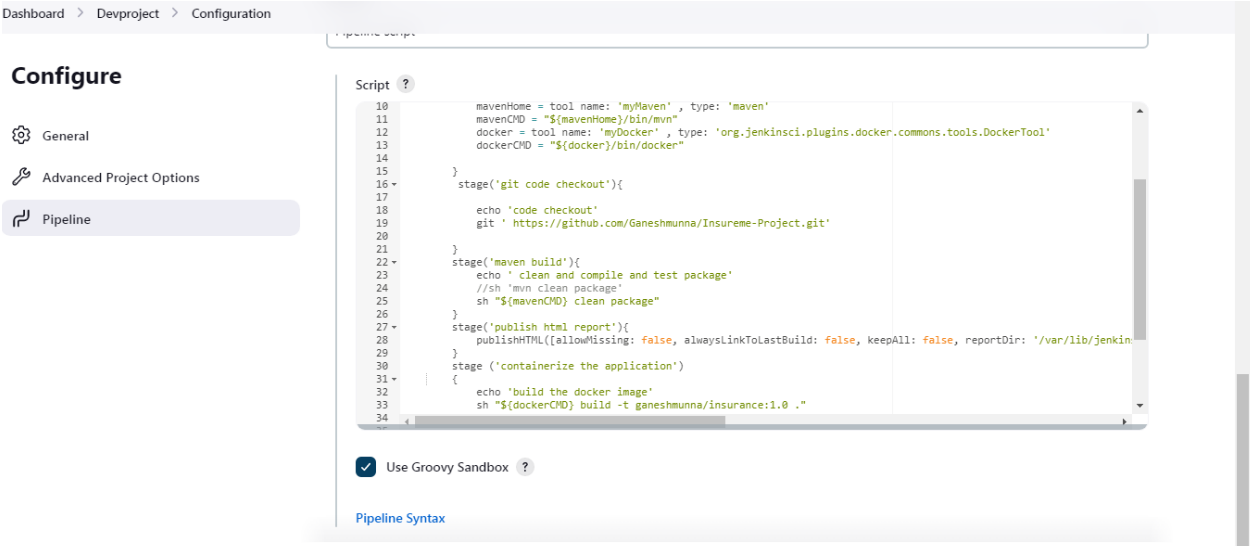


Ganeshmunna / Insureme-Project Public

Webhooks

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our Webhooks Guide.

✓ http://54.204.145.245:8080/github-... (push)

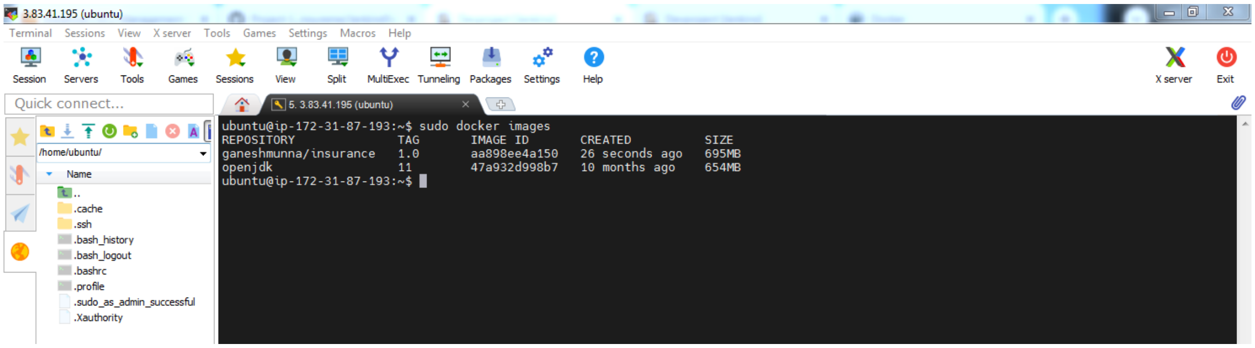


Script

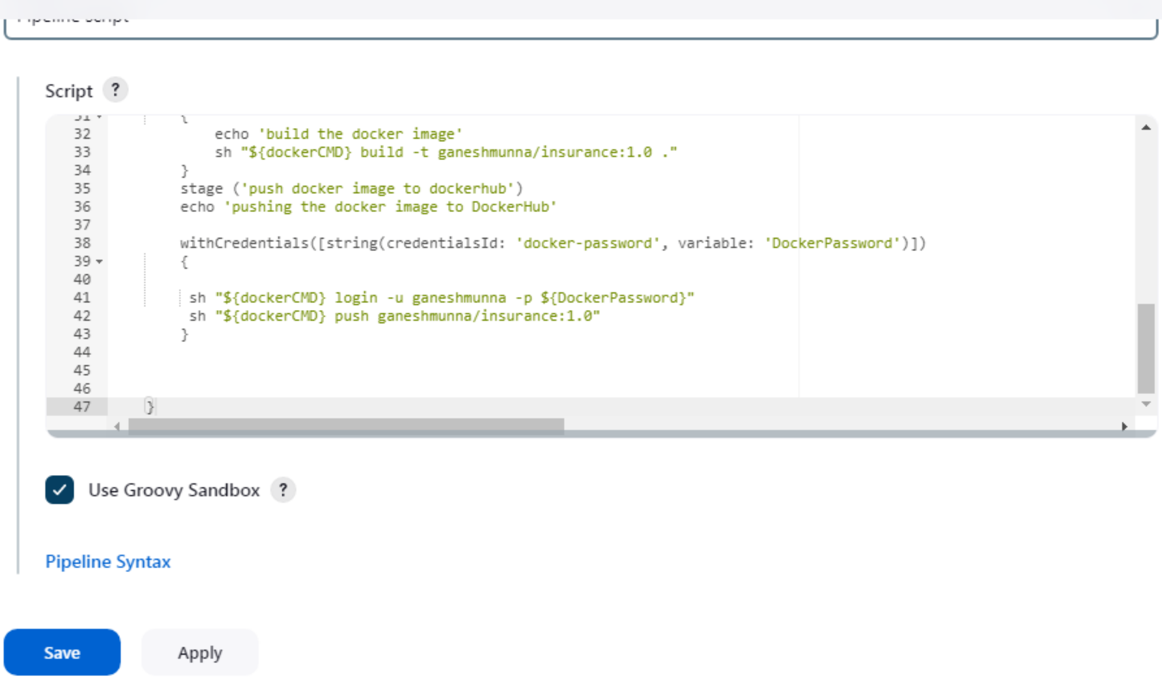
```
            mavenHome = tool name: 'myMaven' , type: 'maven'
            mavenCMD = "${mavenHome}/bin/mvn"
            docker = tool name: 'myDocker' , type: 'org.jenkinsci.plugins.docker.commons.tools.DockerTool'
            dockerCMD = "${docker}/bin/docker"

        }
         stage('git code checkout'){

            echo 'code checkout'
            git ' https://github.com/Ganeshmunna/Insureme-Project.git'

        }
        stage('maven build'){
            echo ' clean and compile and test package'
            //sh 'mvn clean package'
            sh "${mavenCMD} clean package"
        }
        stage('publish html report'){
            publishHTML([allowMissing: false, alwaysLinkToLastBuild: false, keepAll: false, reportDir: '/var/lib/jenkin
        }
        stage ('containerize the application')
        {
            echo 'build the docker image'
            sh "${dockerCMD} build -t ganeshmunna/insurance:1.0 ."
```

☑ Use Groovy Sandbox

Pipeline Syntax



```
ubuntu@ip-172-31-87-193:~$ sudo docker images
REPOSITORY              TAG       IMAGE ID       CREATED          SIZE
ganeshmunna/insurance   1.0       aa898ee4a150   26 seconds ago   695MB
openjdk                 11        47a932d998b7   10 months ago    654MB
ubuntu@ip-172-31-87-193:~$
```

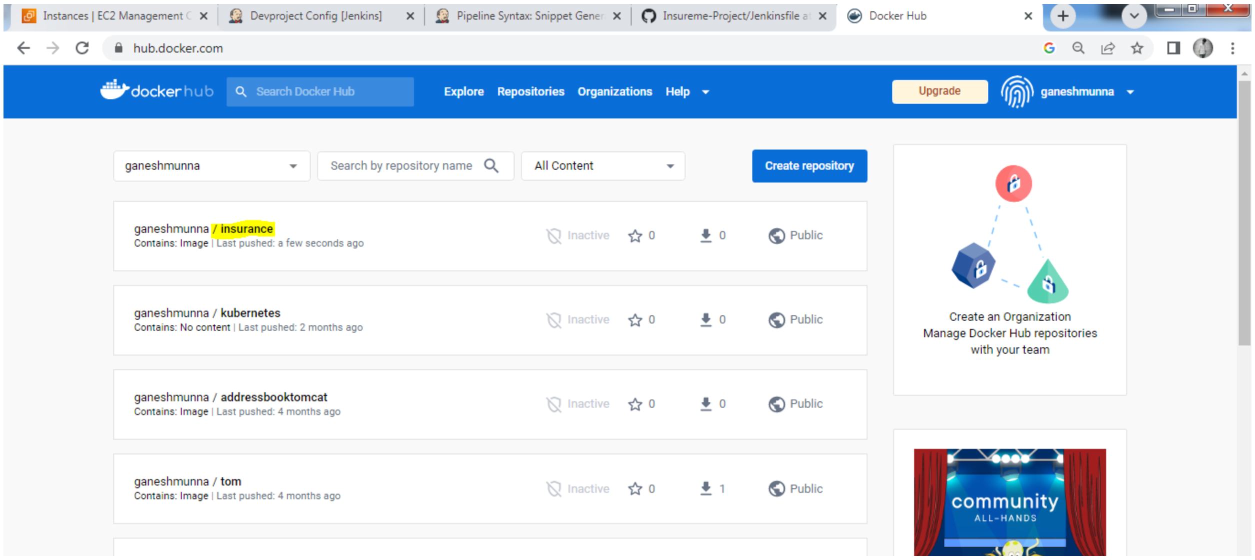


Script

```
        {
            echo 'build the docker image'
            sh "${dockerCMD} build -t ganeshmunna/insurance:1.0 ."
        }
        stage ('push docker image to dockerhub')
        echo 'pushing the docker image to DockerHub'

        withCredentials([string(credentialsId: 'docker-password', variable: 'DockerPassword')])
        {

         sh "${dockerCMD} login -u ganeshmunna -p ${DockerPassword}"
         sh "${dockerCMD} push ganeshmunna/insurance:1.0"
        }



    }
```

☑ Use Groovy Sandbox

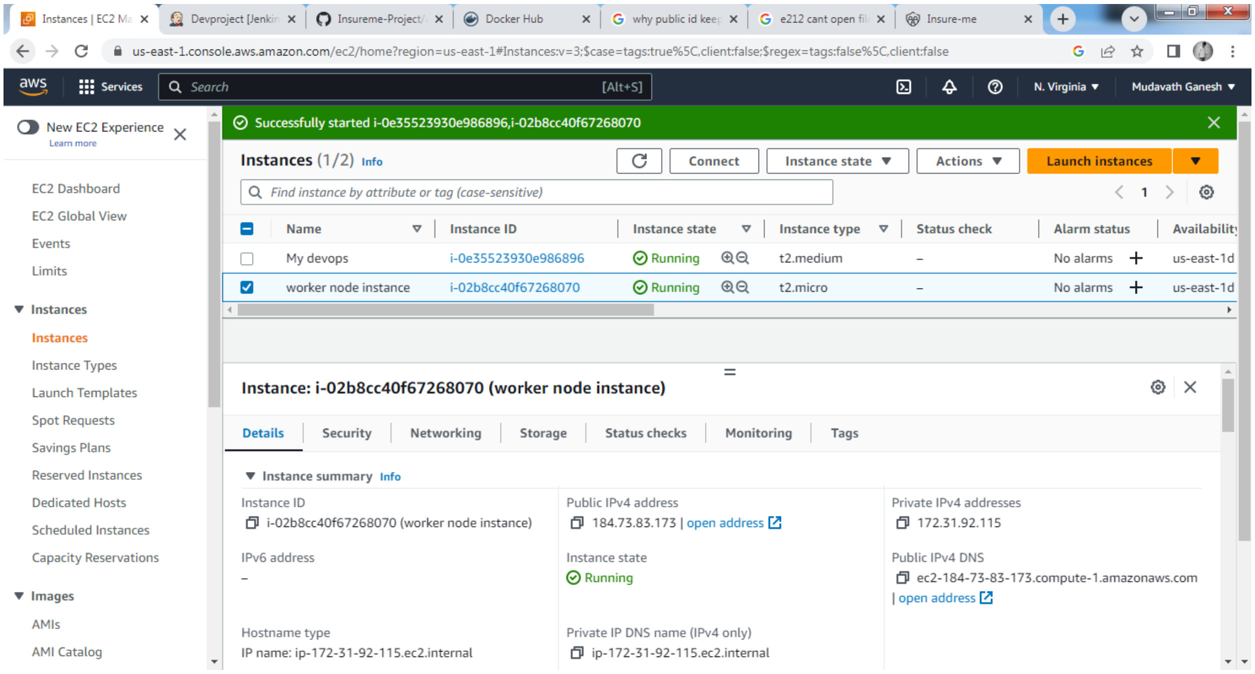
Pipeline Syntax



ganeshmunna | Search by repository name

Repository	Status	Stars	Downloads	Visibility
ganeshmunna / insurance Contains: Image | Last pushed: a few seconds ago	Inactive	0	0	Public
ganeshmunna / kubernetes Contains: No content | Last pushed: 2 months ago	Inactive	0	0	Public
ganeshmunna / addressbooktomcat Contains: Image | Last pushed: 4 months ago	Inactive	0	0	Public
ganeshmunna / tom Contains: Image | Last pushed: 4 months ago	Inactive	0	1	Public

Create an Organization
Manage Docker Hub repositories
with your team



Successfully started i-0e35523930e986896,i-02b8cc40f67268070

Instances (1/2) Info

Find instance by attribute or tag (case-sensitive)

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
☐	My devops	i-0e35523930e986896	Running	t2.medium	–	No alarms	us-east-1d
☑	worker node instance	i-02b8cc40f67268070	Running	t2.micro	–	No alarms	us-east-1d

Instance: i-02b8cc40f67268070 (worker node instance)

Instance summary Info

Instance ID
i-02b8cc40f67268070 (worker node instance)

Public IPv4 address
184.73.83.173 | open address

Private IPv4 addresses
172.31.92.115

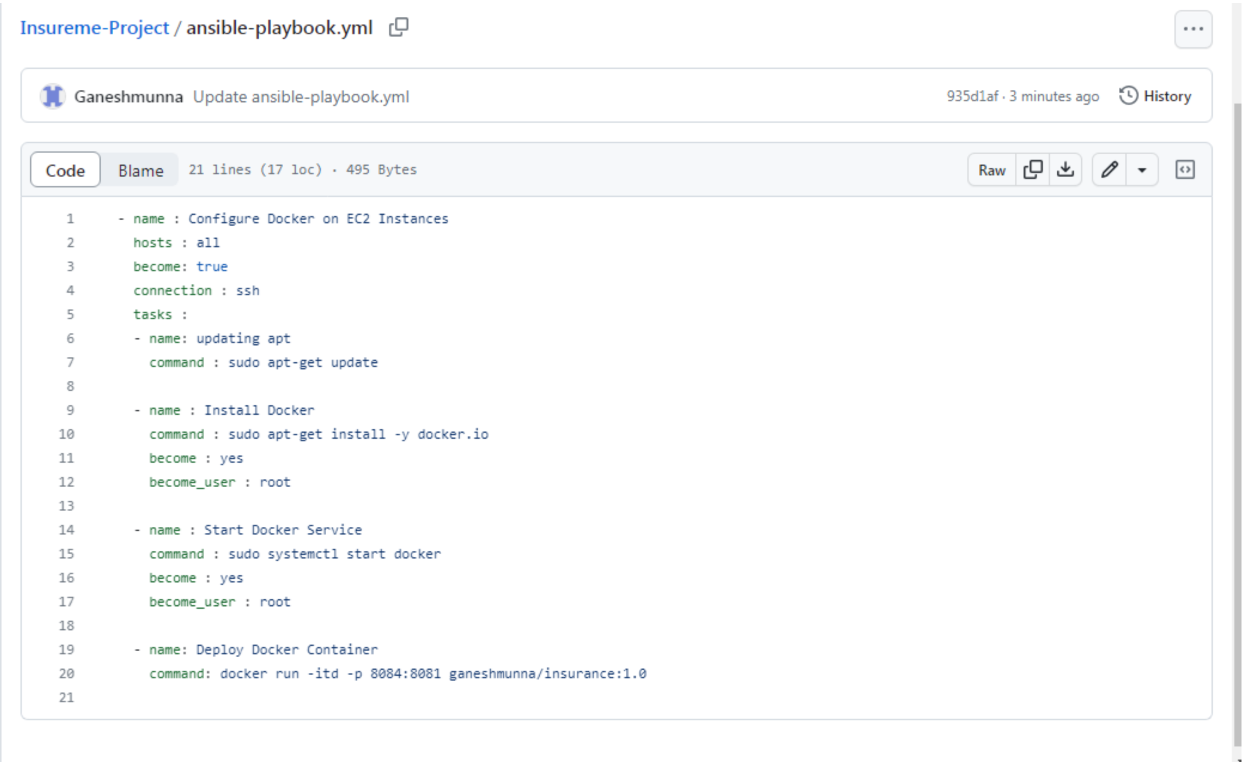
IPv6 address
–

Instance state
Running

Public IPv4 DNS
ec2-184-73-83-173.compute-1.amazonaws.com | open address

Hostname type
IP name: ip-172-31-92-115.ec2.internal

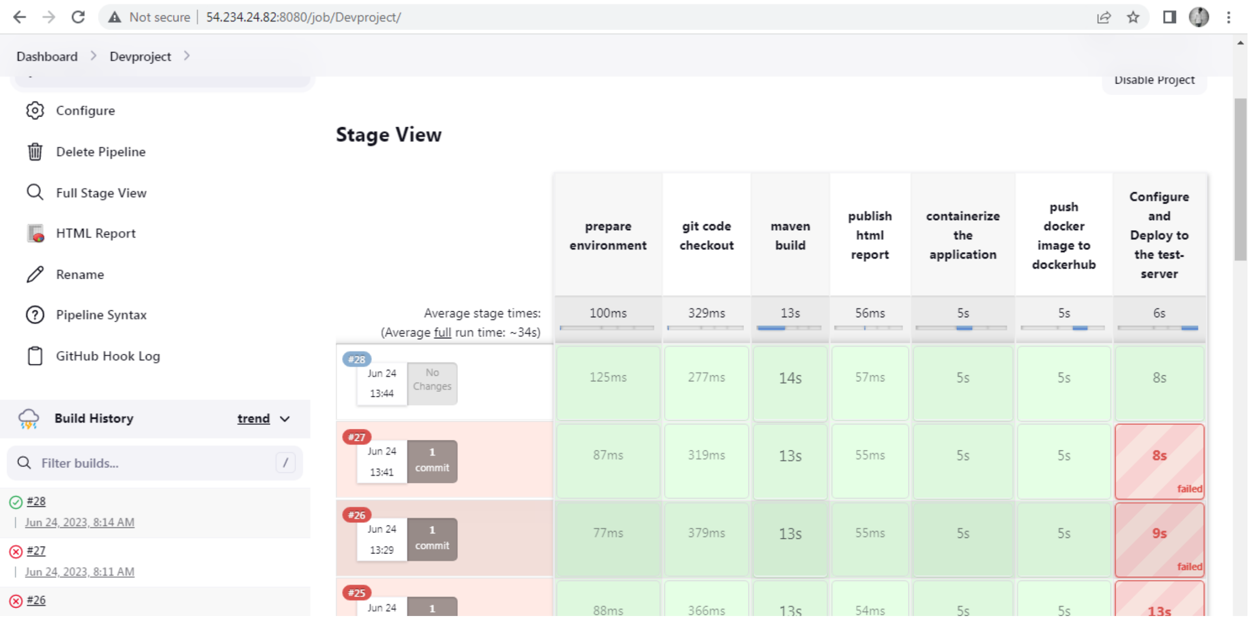
Private IP DNS name (IPv4 only)
ip-172-31-92-115.ec2.internal


Insureme-Project / ansible-playbook.yml

Ganeshmunna Update ansible-playbook.yml 935d1af · 3 minutes ago

21 lines (17 loc) · 495 Bytes

```yaml
- name : Configure Docker on EC2 Instances
  hosts : all
  become: true
  connection : ssh
  tasks :
  - name: updating apt
    command : sudo apt-get update

  - name : Install Docker
    command : sudo apt-get install -y docker.io
    become : yes
    become_user : root

  - name : Start Docker Service
    command : sudo systemctl start docker
    become : yes
    become_user : root

  - name: Deploy Docker Container
    command: docker run -itd -p 8084:8081 ganeshmunna/insurance:1.0

```


Not secure | 54.234.24.82:8080/job/Devproject/

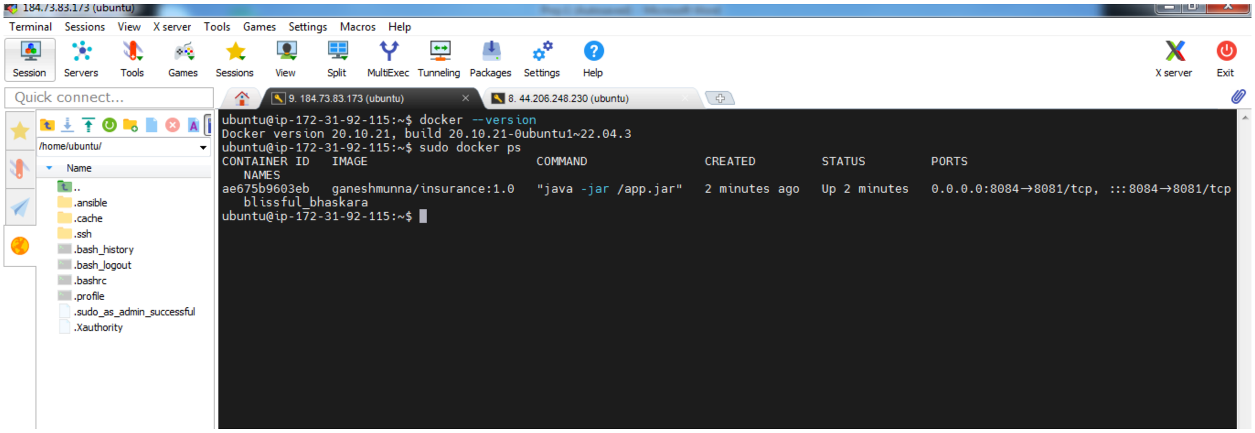
Stage View

	prepare environment	git code checkout	maven build	publish html report	containerize the application	push docker image to dockerhub	Configure and Deploy to the test-server
Average stage times: (Average full run time: ~34s)	100ms	329ms	13s	56ms	5s	5s	6s
#28 Jun 24 13:44 No Changes	125ms	277ms	14s	57ms	5s	5s	8s
#27 Jun 24 13:41 1 commit	87ms	319ms	13s	55ms	5s	5s	8s failed
#26 Jun 24 13:29 1 commit	77ms	379ms	13s	55ms	5s	5s	9s failed
#25 Jun 24 1	88ms	366ms	13s	54ms	5s	5s	13s

#28
Jun 24, 2023, 8:14 AM

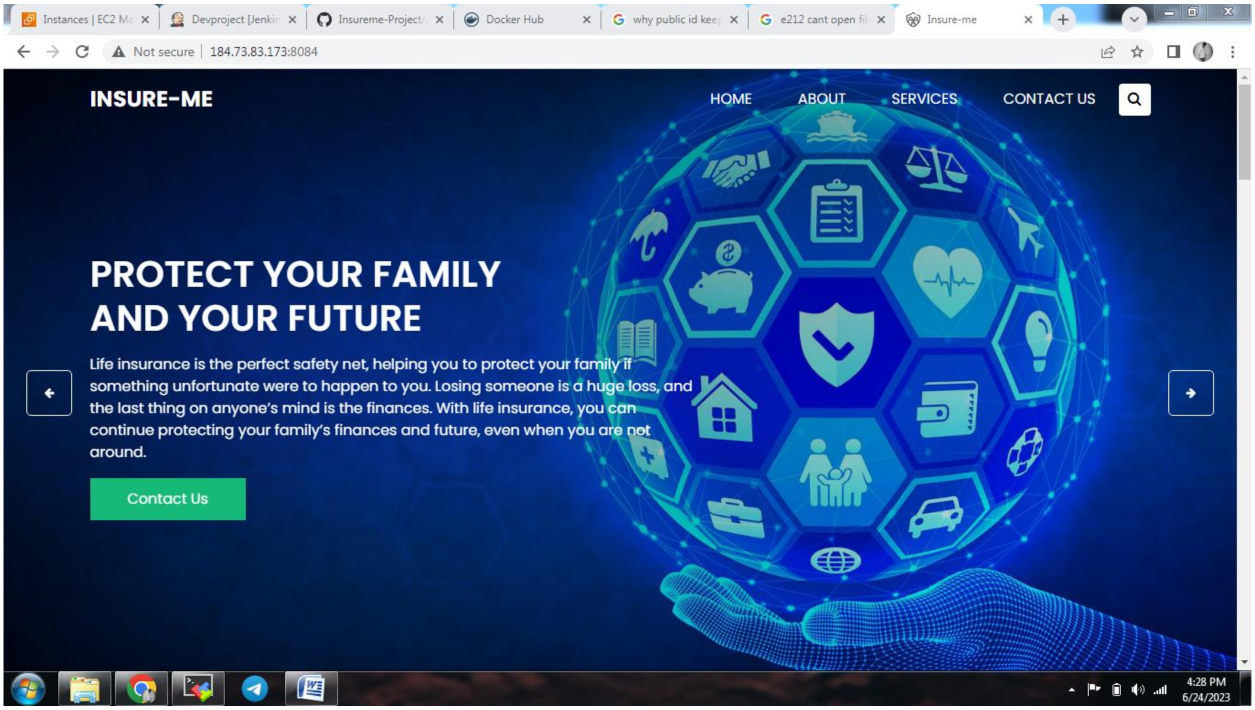
#27
Jun 24, 2023, 8:11 AM

#26



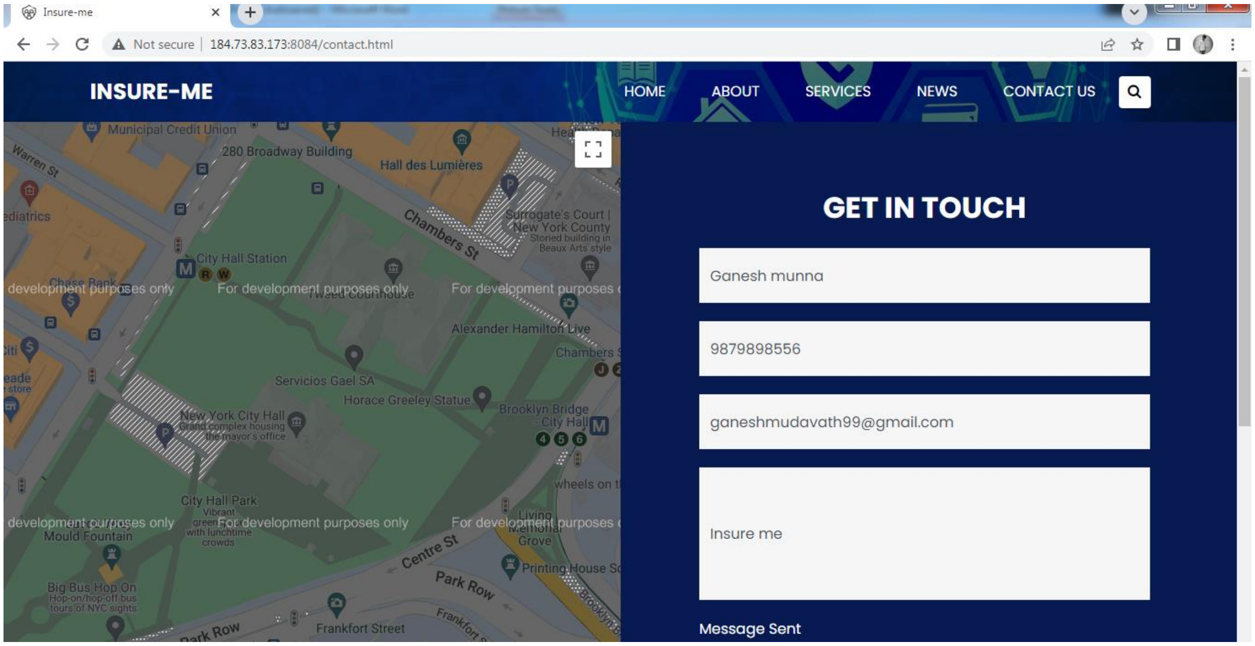
184.73.83.173 (ubuntu)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
9. 184.73.83.173 (ubuntu)
8. 44.206.248.230 (ubuntu)
/home/ubuntu/
Name
..
.ansible
.cache
.ssh
.bash_history
.bash_logout
.bashrc
.profile
.sudo_as_admin_successful
.Xauthority

```
ubuntu@ip-172-31-92-115:~$ docker --version
Docker version 20.10.21, build 20.10.21-0ubuntu1~22.04.3
ubuntu@ip-172-31-92-115:~$ sudo docker ps
CONTAINER ID   IMAGE                      COMMAND                CREATED         STATUS         PORTS
   NAMES
ae675b9603eb   ganeshmunna/insurance:1.0   "java -jar /app.jar"   2 minutes ago   Up 2 minutes   0.0.0.0:8084→8081/tcp, :::8084→8081/tcp
   blissful_bhaskara
ubuntu@ip-172-31-92-115:~$
```



PROTECT YOUR FAMILY AND YOUR FUTURE

Life insurance is the perfect safety net, helping you to protect your family if something unfortunate were to happen to you. Losing someone is a huge loss, and the last thing on anyone's mind is the finances. With life insurance, you can continue protecting your family's finances and future, even when you are not around.



GET IN TOUCH

Ganesh munna

9879898556

ganeshmudavath99@gmail.com

Insure me

Message Sent