

Insure Me

Introduction:

This is a project which demonstrates Devops, Continuous Integration and Continuous Deployment. The following tools and their purpose is mentioned in brief.

- AWS : For creating ec2 machines as servers and deploy the web application.
- 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.

Here is the flow of project:

Commit in SCM → Jenkins pipeline triggers → Maven Build → Docker images is created and pushed onto docker hub → Docker container is created and deployed using ansible playbook → Selenium test is performed on the deployed application.

The entire project uses AWS EC2 instance for infrastructure. It uses 2 EC2 instances namely Jenkins_server and Deploy_serever.

The git repository URL : <https://github.com/Srija1991/Insure-me.git>

Here is the *readme.md* file on repo where there is a brief description about the project Insure-me.

The screenshot shows a GitHub browser interface. The tab title is "AWS Management Console" and the active tab is "Insure-me/README.md at 5e76da...". The page content displays the README.md file. At the top, it shows a commit by "Srija1991 Updated Readme File" made 1 minute ago. Below that, it says "1 contributor". The main content area contains the following text:

Insure me-project

This project will help you to understand various concept related to Insurance domain. Please read the Insurance-domain.pdf to get more details.

This project front is based on simple HTML, CSS and Angular Js ad Backend is Java Spring Boot.

The source of java application for INSUREME is provided by staragile team.

This is a CI/CD pipeline using Git as a source code management tool, maven as build tool,docker as containerising tool,ansible as configuration management tool.

The pipeline script is used to create CI/CD pipeline.

System tray icons are visible at the bottom right, including signal strength, battery, and system status.

The screenshot shows a GitHub browser interface. The tab title is "AWS Management Console" and the active tab is "Insure-me/README.md at 5e76da...". The page content displays the README.md file. The content is identical to the one in the first screenshot, describing the project as based on simple HTML, CSS, and Angular JS with a Java Spring Boot backend, using a CI/CD pipeline with Git, Maven, Docker, and Ansible. The system tray icons at the bottom right show ENG IN, 15:13, and 22-04-2023.

Two instances are configured namely *Jenkins_server* and *Deploy_server*.

Jenkins_server: t2.medium

The screenshot shows the AWS EC2 'Launch an instance' wizard. In the 'Name and tags' section, the name 'Jenkins_Server' is entered. Under 'Application and OS Images (Amazon Machine Image)', the AMI 'ami-02eb7a4783e7e9317' is selected. In the 'Summary' section, the instance type is set to 't2.medium', the storage is '1 volume(s) - 8 GiB', and the 'Launch instance' button is highlighted.

This screenshot shows the same 'Launch an instance' wizard, but with a different AMI selected: 'ami-02eb7a4783e7e9317'. The instance type 't2.medium' is again chosen, and the 'Launch instance' button is highlighted.

Launch an instance | EC2 Management

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#LaunchInstances:

Simple-DevOps-Pro... Dec-3rd Batch - Go... Docker Hub Architecture Comp... Application 3 tier A... Jenkins Tomcat Dep... Devops GI (03 Dec-... Book Your Appoint... Weekly Feedback F...

Mumbai Sria

aws Services Search [Alt+S]

instance.

Create security group Select existing security group

Security groups Info Select security groups launch-wizard-14 sg-0c69f259b0ea97dc0 VPC: vpc-0243302b4b1acc768

Compare security group rules

Configure storage Info Advanced

1x 10 GiB gp2 Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

Add new volume

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

Number of instances Info 1

Software Image (AMI) Canonical, Ubuntu, 22.04 LTS, ...read more ami-02eb7a4783e7e9317

Virtual server type (instance type) t2.medium

Firewall (security group) launch-wizard-14

Storage (volumes) 1 volume(s) - 10 GiB

Cancel Launch instance Review commands

CloudShell Feedback Language © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences 19:25 20-04-2023

Instances | EC2 Management

ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#Instances:instanceState=running

Simple-DevOps-Pro... Dec-3rd Batch - Go... Docker Hub Architecture Comp... Application 3 tier A... Jenkins Tomcat Dep... Devops GI (03 Dec-... Book Your Appoint... Weekly Feedback F...

Mumbai Sria

aws Services Search [Alt+S]

New EC2 Experience Tell us what you think

EC2 Dashboard EC2 Global View Events Tags Limits

Instances Instances Instance Types Launch Templates Spot Requests Savings Plans Reserved Instances Dedicated Hosts Capacity Reservations

Images

Instances (2) Info Connect Instance state Actions Launch instances

Find instance by attribute or tag (case-sensitive)

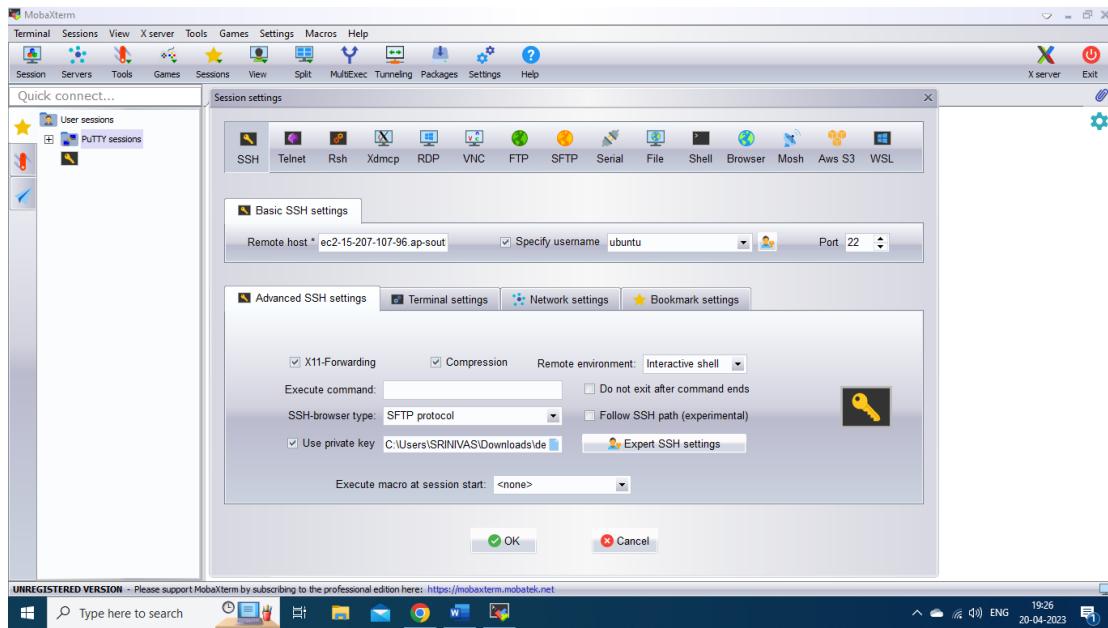
Instance state = running Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Available
Jenkins_Server	i-09f554fec7e640f0b	Running	t2.medium	2/2 checks passed...	No alarms	ap-sout
Deploy_Server	i-03ef7b668fb9a17eb	Running	t2.micro	2/2 checks passed...	No alarms	ap-sout

Select an instance

CloudShell Feedback Language © 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences 22:12 20-04-2023

This *Jenkins_server* is connected via *mobaXterm*.



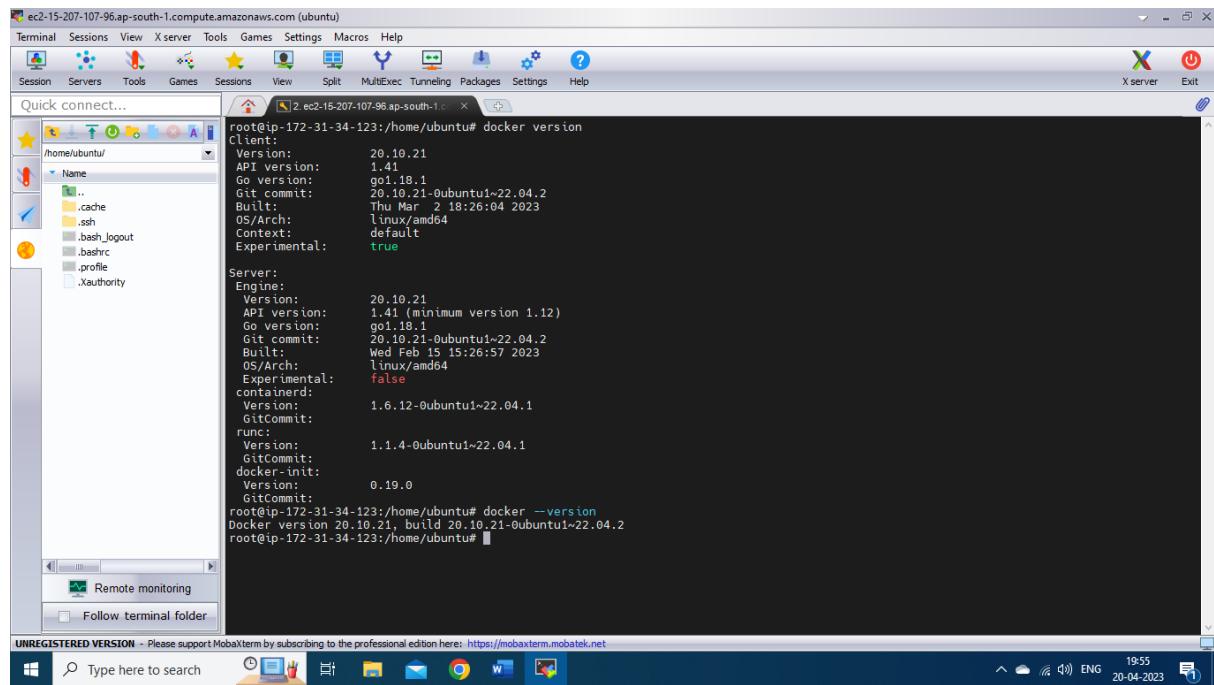
Here, this server is used to execute Jenkins pipeline and so the java version 17 is installed, and maven version 3.6.3 is installed and environment variables are set accordingly.

Jenkins is installed and configured to access on port number :8080.

Docker is installed with version 20.10

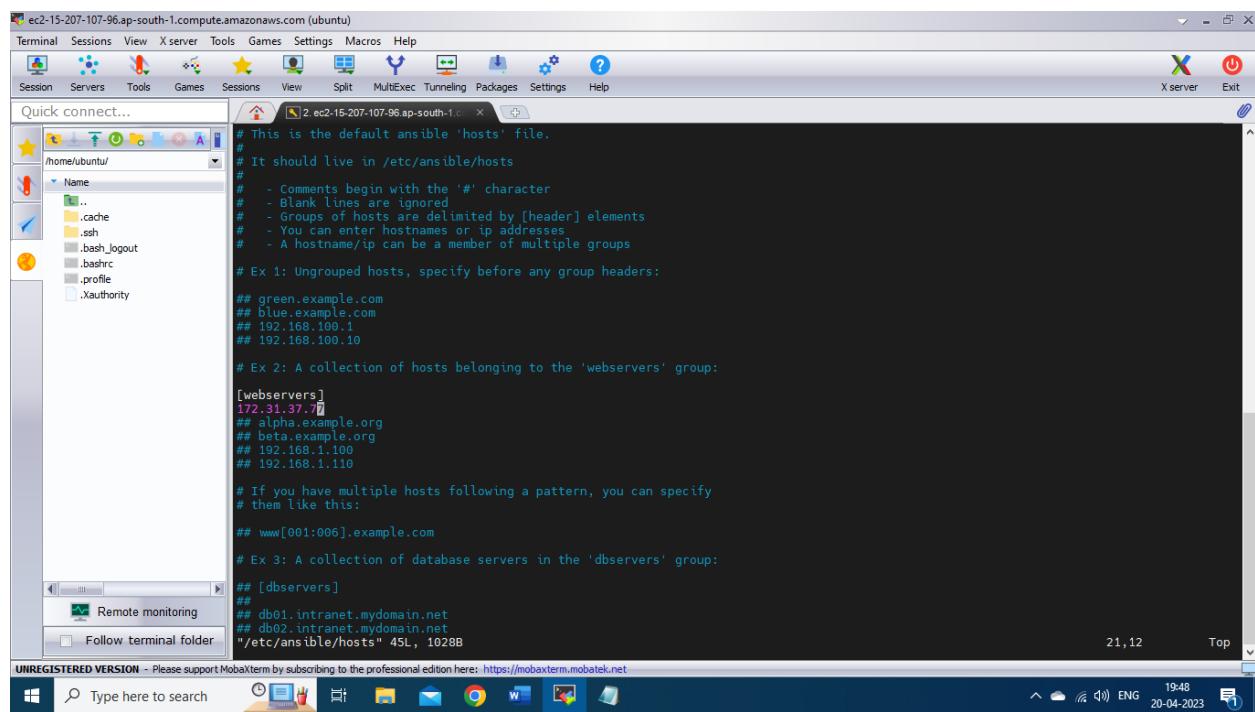
Ansible is installed and deploy_server is added to /etc/ansible/hosts.

A screenshot of the mobaXterm terminal window. The terminal shows the output of several commands: 'java --version' (OpenJDK Runtime Environment, Java 17), 'mvn --version' (Apache Maven 3.6.3), 'systemctl status jenkins' (jenkins.service is active and running), and 'java -jar jenkins.war' (starting Jenkins). The terminal also shows the output of 'ansible --version' (Ansible 2.14.4) and 'ansible -m module_list' (listing available modules and collections).



```
root@ip-172-31-34-123:/home/ubuntu# docker version
Client:
  Version:          20.10.21
  API version:      1.41
  Go version:       go1.18.1
  Git commit:       2010.21-0ubuntu1~22.04.2
  Built:            Thu Mar  2 18:26:04 2023
  OS/Arch:          linux/amd64
  Context:          default
  Experimental:    true

Server:
  Engine:
    Version:          20.10.21
    API version:      1.41 (minimum version 1.12)
    Go version:       go1.18.1
    Git commit:       2010.21-0ubuntu1~22.04.2
    Built:            Wed Feb 15 15:26:57 2023
    OS/Arch:          linux/amd64
    Experimental:    false
  containerd:
    Version:          1.6.12-0ubuntu1~22.04.1
    GitCommit:        04c49b3
  runc:
    Version:          1.1.4-0ubuntu1~22.04.1
    GitCommit:        901bf6e
  docker-init:
    Version:          0.19.0
    GitCommit:        958a5b3
root@ip-172-31-34-123:/home/ubuntu# docker --version
Docker version 20.10.21, build 2010.21-0ubuntu1~22.04.2
root@ip-172-31-34-123:/home/ubuntu#
```



```
# This is the default ansible 'hosts' file.
# It should live in /etc/ansible/hosts
# - Comments begin with the '#' character
# - Blank lines are ignored
# - Groups of hosts are delimited by [header] elements
# - You can enter hostnames or ip addresses
# - A hostname/ip can be a member of multiple groups

# Ex 1: Ungrouped hosts, specify before any group headers:
## green.example.com
## blue.example.com
## 192.168.100.1
## 192.168.100.10

# Ex 2: A collection of hosts belonging to the 'webservers' group:
[webservers]
172.31.37.7
## alpha.example.org
## beta.example.org
## 192.168.1.100
## 192.168.1.110

# If you have multiple hosts following a pattern, you can specify
# them like this:
## www[001:006].example.com

# Ex 3: A collection of database servers in the 'dbservers' group:
[dbservers]
##
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
"/etc/ansible/hosts" 45L, 1028B
```

This the code is used to build docker image using the jar file obtained after maven build.

The screenshot shows a GitHub repository page for 'Srija1991 / Insure-me'. The Dockerfile content is displayed:

```

FROM openjdk:17
ARG JAR_FILE=target/*.jar
COPY ${JAR_FILE} app.jar
ENTRYPOINT ["java","-jar","/app.jar"]

```

Here is the pipeline script in Jenkins file executed to fetch code using ‘checkout’ and the next stage is ‘package the application’ where the maven output is packed as jar file.

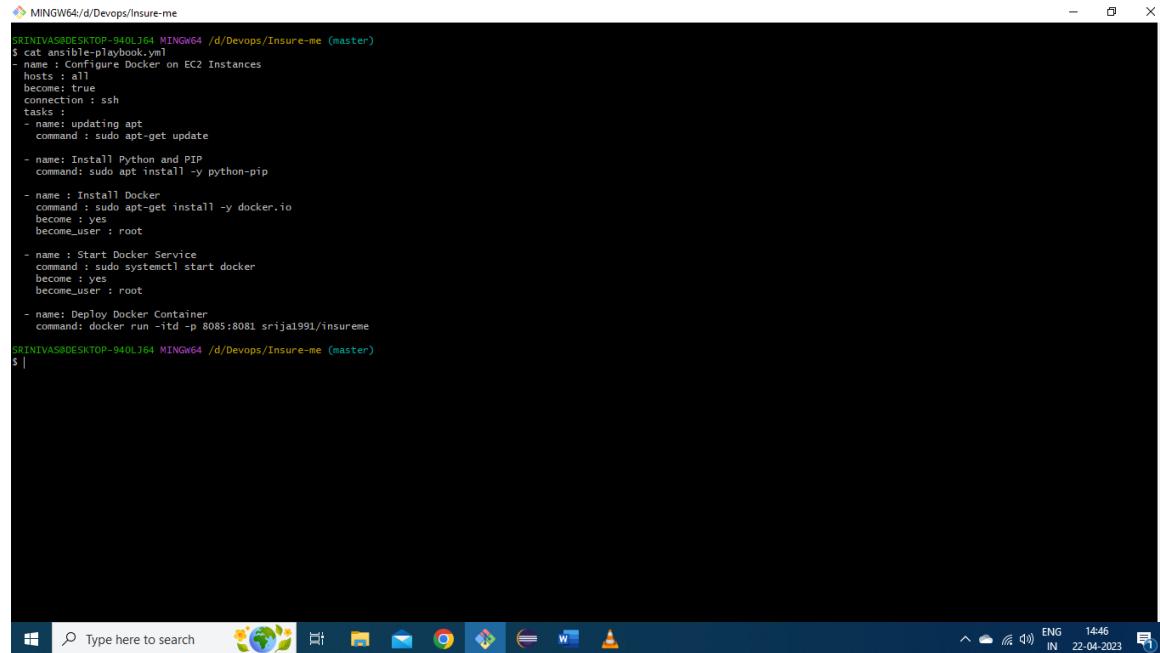
```

pipeline {
    agent any
    tools {
        maven 'maven'
    }
    stages{
        stage('Checkout'){
            steps{
                echo '*****Checkout Git URL*****'
                git branch: 'master', url: 'https://github.com/Srija1991/Insure-me.git'
            }
        }
        stage('Package the application'){
            steps{
                echo "*****Packaging the application*****"
                sh 'mvn clean package'
            }
        }
        stage('Publish the Reports using HTML'){
            steps{
                echo "*****Publishing The HTML Reports*****"
                publishHTML(allowMissing: false, alwaysLinkToLastBuild: false, keepAll: false, reportDir: '/var/lib/jenkins/workspace/Insure_me/target/surefire-reports', reportFiles: 'index.html', reportName: 'HTML Report', reportTitles: 'Reports', useWrapperFileDirectly: true)
            }
        }
        stage('Docker Image creation'){
            steps{
                echo "*****Docker image is created*****"
                sh 'docker build -t srija1991/insureme .'
            }
        }
        stage('docker login'){
            steps{
                withCredentials([usernamePassword(credentialsId: 'logindocker', passwordVariable: 'docker_pswd', usernameVariable: 'docker_usr')]) {
                    sh "docker login -u $env.docker_usr -p ${env.docker_pswd}"
                    echo "*****Docker LOGIN Successful*****"
                }
            }
        }
        stage('Push Image to docker Hub'){
            steps{
                sh 'docker push srija1991/insureme'
                echo "*****Image pushed succesfully onto DockerHUB*****"
            }
        }
        stage('Deploy using ansible'){
            steps{
                ansiblePlaybook_credentialsId: 'ansible', disableHostKeyChecking: true, installation: 'ansible', inventory: '/etc/ansible/hosts', playbook: 'ansible-playbook.yml'
                Jenkins_file [dos] (21:59 20/04/2023)
                Jenkins_file [dos] 55L, 1988B
            }
        }
    }
}

```

Then next stage is to publish the html reports. Docker image is created in the next stage and image is pushed onto dockerhub. Then stage is deploying using ansible.

Here is the ansible-playbook.yml to configure docker, and run the container on it.



```
MINGW64/d/Devops/Insure-me
$ cat ansible-playbook.yml
- name : Configure Docker on EC2 Instances
  hosts: all
  become: true
  connection : ssh
  tasks :
    - name: updating apt
      command : sudo apt-get update

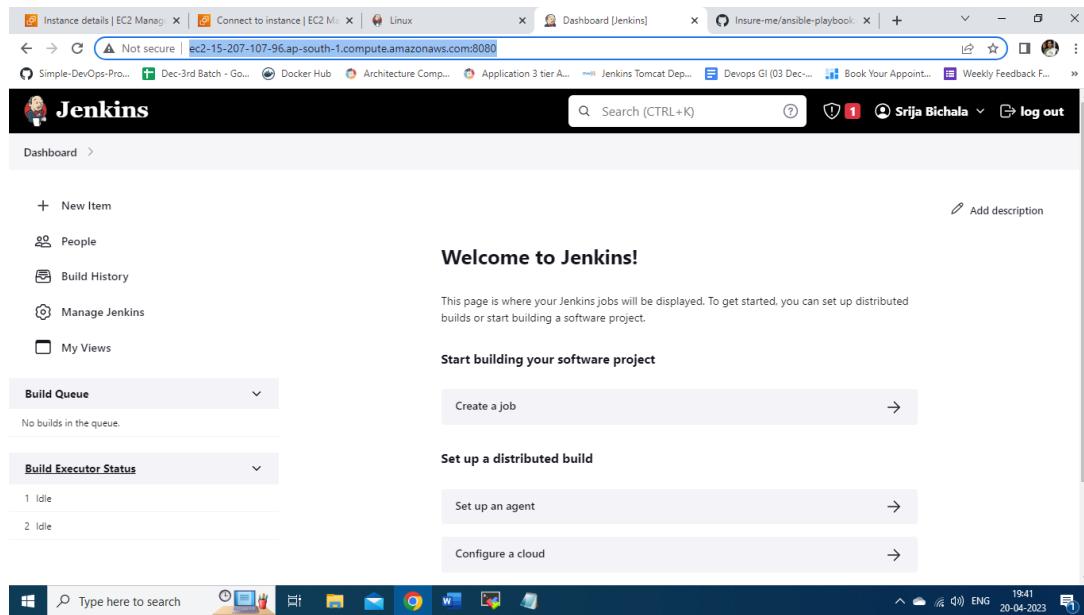
    - name: Install Python and PIP
      command: sudo apt install -y python-pip

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

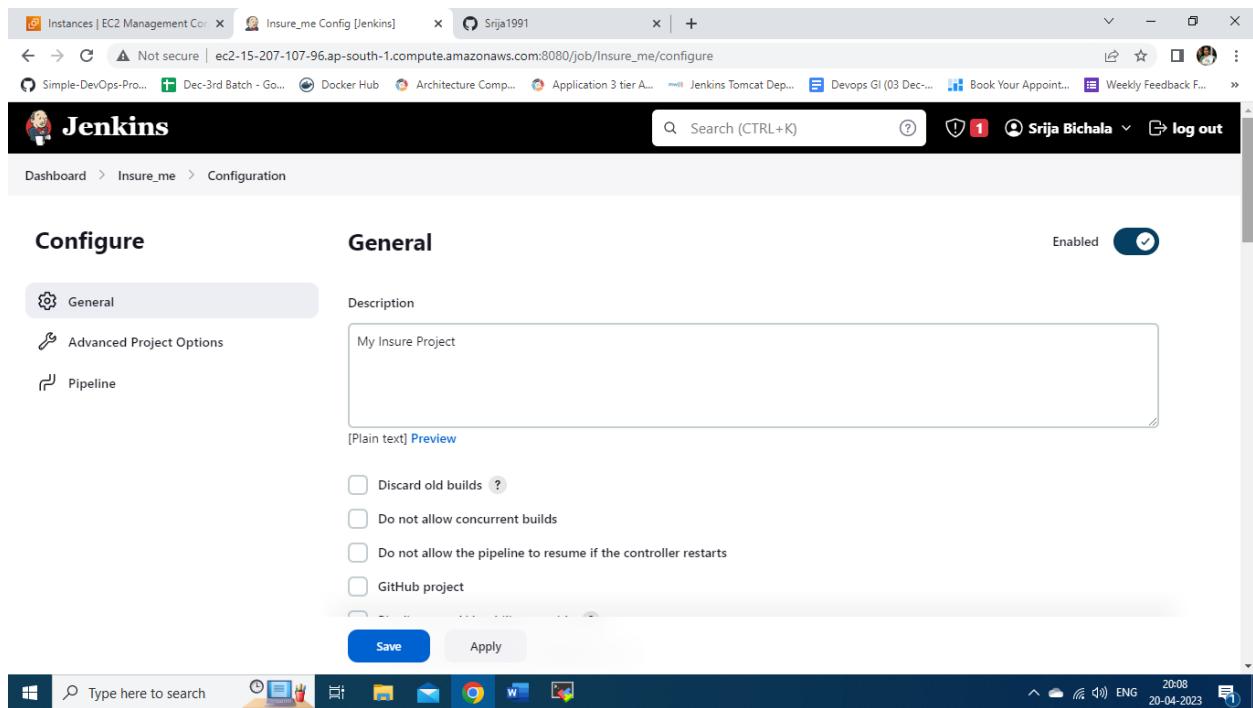
    - name : Start Docker Service
      command : sudo systemctl start docker
      become : yes
      become_user : root

    - name: Deploy Docker Container
      command: docker run -itd -p 8085:8081 srija1991/insureme
$ |
```

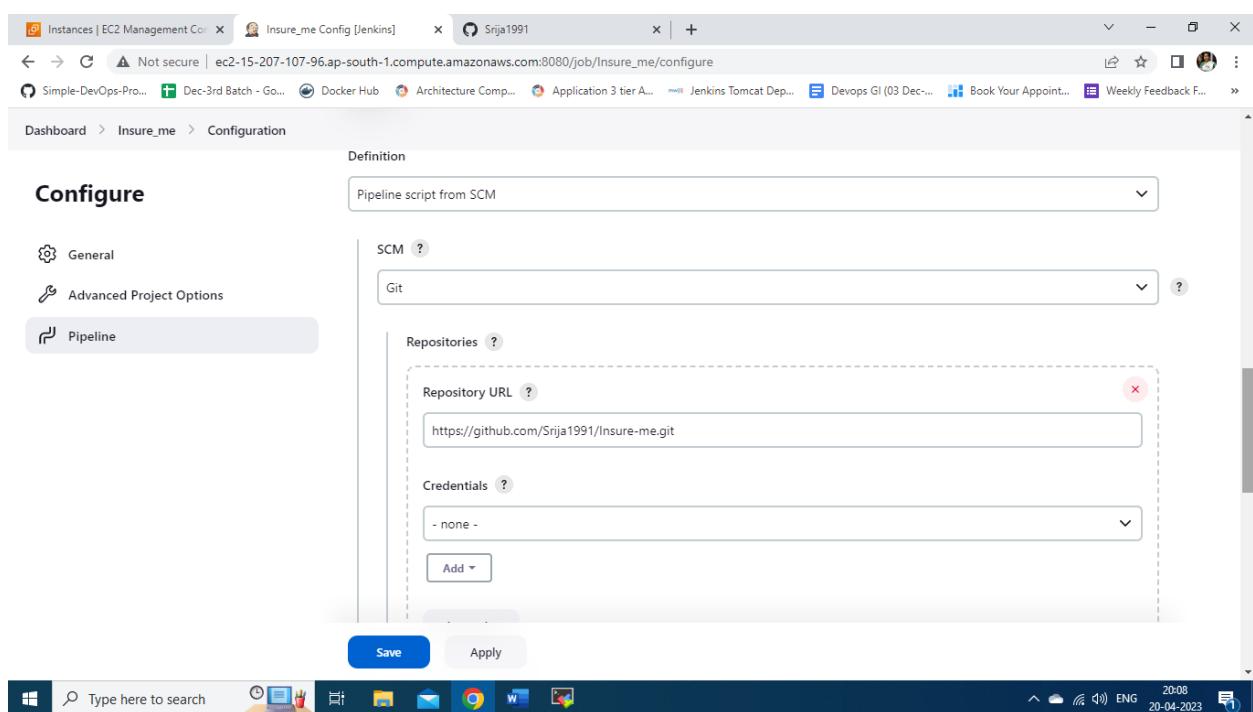
The Jenkins server is operated on browser at port 8080. The pipeline is created with name Insure-me.



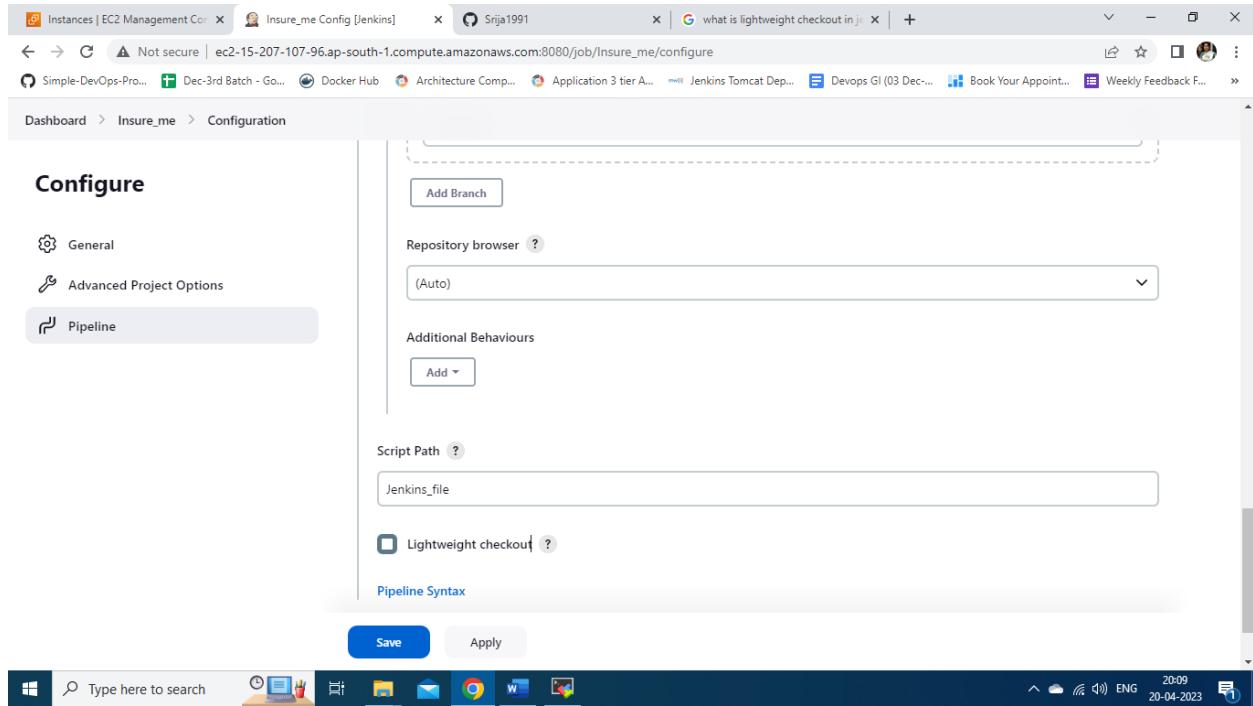
The screenshot shows a browser window with the Jenkins dashboard loaded. The address bar shows the URL `ec2-15-207-107-96.ap-south-1.compute.amazonaws.com:8080`. The dashboard features a "Welcome to Jenkins!" message and sections for "Start building your software project" and "Set up a distributed build". On the left, there are links for "New Item", "People", "Build History", "Manage Jenkins", and "My Views". A "Build Queue" section indicates "No builds in the queue". A "Build Executor Status" section shows two idle executors. The top navigation bar includes links for "Instance details | EC2 Manager", "Connect to instance | EC2 Manager", "Linux", "Dashboard [Jenkins]", and "Insure-me/ansible-playbook". The bottom status bar shows the date and time as "20-04-2023 19:41".



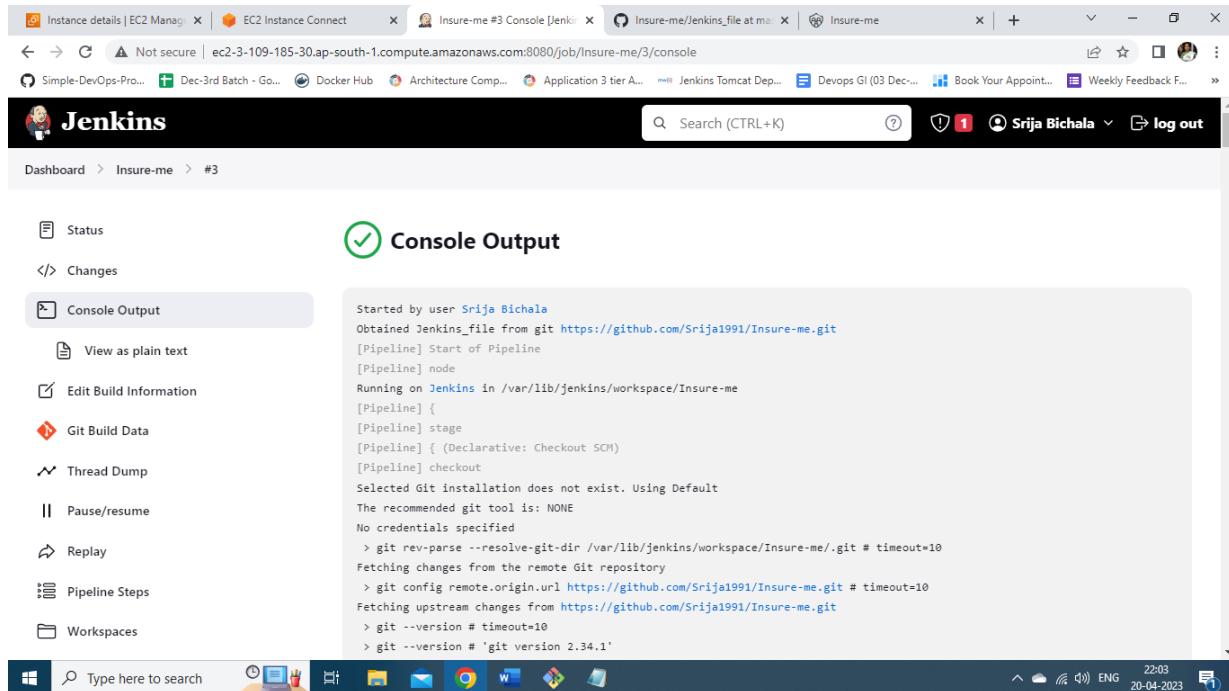
The screenshot shows the Jenkins 'General' configuration page for a project named 'Insure_me'. The 'Enabled' toggle switch is turned on. The 'Description' field contains the text 'My Insure Project'. Under the 'Advanced' section, there are four checkboxes: 'Discard old builds', 'Do not allow concurrent builds', 'Do not allow the pipeline to resume if the controller restarts', and 'GitHub project'. At the bottom are 'Save' and 'Apply' buttons.



The screenshot shows the Jenkins 'Pipeline' configuration page for the same 'Insure_me' project. The 'Definition' dropdown is set to 'Pipeline script from SCM'. The 'SCM' section is configured for 'Git' with the repository URL 'https://github.com/Srija1991/Insure-me.git' and no credentials. At the bottom are 'Save' and 'Apply' buttons.



The pipeline is build and is success, it menas it created image and container is run on port 8085 and application is accessible.



```
[Pipeline] echo
*****Checkout Git URL*****
[Pipeline] git
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Insure-me/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/Srija1991/Insure-me.git # timeout=10
Fetching upstream changes from https://github.com/Srija1991/Insure-me.git
> git --version # timeout=10
> git --version # 'git version 2.34.1'
> git fetch --tags --force --progress -- https://github.com/Srija1991/Insure-me.git +refs/heads/*:refs/remotes/origin/*
timeout=10
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision d42a067fab3b468efcaa33227bfcb1d547f125b (refs/remotes/origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f d42a067fab3b468efcaa33227bfcb1d547f125b # timeout=10
> git branch -a -v --no-abbrev # timeout=10
> git branch -D master # timeout=10
> git checkout -b master d42a067fab3b468efcaa33227bfcb1d547f125b # timeout=10
Commit message: "Updated Docker"
[Pipeline]
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // stage
```

```
[Pipeline] envVarsForTool
[Pipeline] withEnv
[Pipeline] {
[Pipeline] echo
*****Packaging the application*****
[Pipeline] sh
+ mvn clean package
[0;34mINFO[m] Scanning for projects...
[0;34mINFO[m] [1m-----< [0;36mcom.project.staragile:insure-me[0;1m >-----[m
[0;34mINFO[m] [1mBuilding Insure-me 1.0@[m
[0;34mINFO[m] [1m-----[ jar ]-----[m
[0;34mINFO[m]
[0;34mINFO[m] [1m--- [0;32mmaven-clean-plugin:3.2.0:clean@[1m(default-clean)@[m @ [36minsure-me@0;1m ---@[m
[0;34mINFO[m] Deleting /var/lib/jenkins/workspace/Insure-me/target
[0;34mINFO[m]
[0;34mINFO[m] [1m--- [0;32mmaven-resources-plugin:3.2.0:resources@[m @[1m(default-resources)@[m @ [36minsure-me@0;1m ---@[m
[0;34mINFO[m] Using 'UTF-8' encoding to copy filtered resources.
[0;34mINFO[m] Using 'UTF-8' encoding to copy filtered properties files.
[0;34mINFO[m] Copying 1 resource
[0;34mINFO[m] Copying 31 resources
[0;34mINFO[m]
[0;34mINFO[m] [1m--- [0;32mmaven-compiler-plugin:3.10.1:compile@[m @[1m(default-compile)@[m @ [36minsure-me@0;1m ---@[m
[0;34mINFO[m] Changes detected - recompiling the module!
```

Instance details | EC2 Manager | EC2 Instance Connect | Insure-me #3 Console [Jenkins] | Insure-me/jenkins_file at master | Insure-me

Not secure | ec2-3-109-185-30.ap-south-1.compute.amazonaws.com:8080/job/Insure-me/3/console

Simple-DevOps-Pro... Dec-3rd Batch - Go... Docker Hub Architecture Comp... Application 3 tier A... Jenkins Tomcat Dep... Devops GI (03 Dec... Book Your Appoint... Weekly Feedback F...

Dashboard > Insure-me > #3

```
[Pipeline] echo
*****Publishing The HTML Reports*****
[Pipeline] publishHTML
[htmlpublisher] Archiving HTML reports...
[htmlpublisher] Archiving at PROJECT level /var/lib/jenkins/workspace/Insure_me/target/surefire-reports to
/var/lib/jenkins/jobs/Insure-me/htmlreports/HTML_20Report
[Pipeline]
[Pipeline] // withEnv
[Pipeline]
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Docker Image creation)
[Pipeline] tool
[Pipeline] envVarsForTool
[Pipeline] withEnv
[Pipeline] {
[Pipeline] echo
*****Docker image is created*****
[Pipeline] sh
+ docker build -t srija1991/insureme .
Sending build context to Docker daemon 47.62MB

Step 1/4 : FROM openjdk:17
--> 5e28ba2b4cd8
Step 2/4 : ARG JARFILE=target/*.jar
--> Using cache
--> 6c8f7f9cc71
```

22:04 20-04-2023

Instance details | EC2 Manager | EC2 Instance Connect | Insure-me #3 Console [Jenkins] | Insure-me/jenkins_file at master | Insure-me

Not secure | ec2-3-109-185-30.ap-south-1.compute.amazonaws.com:8080/job/Insure-me/3/console

Simple-DevOps-Pro... Dec-3rd Batch - Go... Docker Hub Architecture Comp... Application 3 tier A... Jenkins Tomcat Dep... Devops GI (03 Dec... Book Your Appoint... Weekly Feedback F...

Dashboard > Insure-me > #3

```
WARNING! Your password will be stored unencrypted in /var/lib/jenkins/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[Pipeline] echo
*****Docker LOGIN Successful*****
[Pipeline]
[Pipeline] // withCredentials
[Pipeline]
[Pipeline] // withEnv
[Pipeline]
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Push Image to docker Hub)
[Pipeline] tool
[Pipeline] envVarsForTool
[Pipeline] withEnv
[Pipeline] {
[Pipeline] {
[Pipeline] sh
+ docker push srija1991/insureme
Using default tag: latest
The push refers to repository [docker.io/srija1991/insureme]
aecd7f6fa10: Preparing
dc9fa3d8b576: Preparing
27ee19dc88f2: Preparing
```

22:05 20-04-2023

The screenshot shows a Windows desktop environment with a taskbar at the bottom. The taskbar includes icons for File Explorer, Mail, Google Chrome, and other system tools. The system tray shows the date as 20-04-2023 and the time as 22:05. The desktop background is white.

The main window is a Jenkins pipeline console for job #3. The output shows the following Ansible playbook execution:

```
[Pipeline] echo
*****Image pushed succesfully onto DockerHUB*****
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy using ansible)
[Pipeline] tool
[Pipeline] envVarsForTool
[Pipeline] withEnv
[Pipeline] {
[Pipeline] ansiblePlaybook
[Insure-me] $ /usr/bin/ansible-playbook ansible-playbook.yml -i /etc/ansible/hosts --private-key /var/lib/jenkins/workspace/Insure-me/ssh648008610903458801.key -u ubuntu

PLAY [Configure Docker on EC2 Instances] *****

TASK [Gathering Facts] *****
ok: [172.31.37.77]

TASK [updating apt] *****
changed: [172.31.37.77]

TASK [Install Python and PIP] *****
changed: [172.31.37.77]
```

This screenshot shows the same Jenkins pipeline console window, but with more detailed Ansible playbook steps visible in the output. The steps include installing Docker and starting the Docker service.

```
[Insure-me] $ /usr/bin/ansible-playbook ansible-playbook.yml -i /etc/ansible/hosts --private-key /var/lib/jenkins/workspace/Insure-me/ssh648008610903458801.key -u ubuntu

PLAY [Configure Docker on EC2 Instances] *****

TASK [Gathering Facts] *****
ok: [172.31.37.77]

TASK [updating apt] *****
changed: [172.31.37.77]

TASK [Install Python and PIP] *****
changed: [172.31.37.77]

TASK [Install Docker] *****
changed: [172.31.37.77]

TASK [Start Docker Service] *****
changed: [172.31.37.77]

TASK [Deploy Docker Container] *****
changed: [172.31.37.77]

PLAY RECAP *****
172.31.37.77 : ok=6    changed=5    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Instance details | EC2 Management | EC2 Instance Connect | Insure-me #3 Console [Jenkins] | Insure-me/jenkins_file_at_ma... | Insure-me | + | - | X

Not secure | ec2-3-109-185-30.ap-south-1.compute.amazonaws.com:8080/job/Insure-me/3/console

Simple-DevOps-Pro... Dec-3rd Batch - Go... Docker Hub Architecture Comp... Application 3 tier A... Jenkins Tomcat Dep... Devops GI (03 Dec-... Book Your Appoint... Weekly Feedback F...

Dashboard > Insure-me > #3

```
TASK [Deploy Docker Container] ****
changed: [172.31.37.77]

PLAY RECAP ****
172.31.37.77 : ok=6    changed=5    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[Pipeline] echo
*****Deployment onto Server is Sucessful*****
[Pipeline]
[Pipeline] // withEnv
[Pipeline]
[Pipeline] // stage
[Pipeline]
[Pipeline] // withEnv
[Pipeline]
[Pipeline] // withEnv
[Pipeline]
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```



Instances | EC2 Management | EC2 Instance Connect | Insure-me [Jenkins] | Insure-me/jenkins_file_at_ma... | Insure-me | + | - | X

Not secure | ec2-3-109-185-30.ap-south-1.compute.amazonaws.com:8080/job/Insure-me/

Simple-DevOps-Pro... Dec-3rd Batch - Go... Docker Hub Architecture Comp... Application 3 tier A... Jenkins Tomcat Dep... Devops GI (03 Dec-... Book Your Appoint... Weekly Feedback F...

d > Insure-me >

Pipeline Insure-me

Status

Ranges

Project

Build Now

Edit description

Configure

Delete Pipeline

View Stage View

Timeline Report

Timeline

Pipeline Syntax

Stage View

	Declarative: Checkout SCM	Declarative: Tool Install	Checkout	Package the application	Publish the Reports using HTML	Docker Image creation	docker login	Push Image to docker Hub	Deploy using Success
Average stage times: (Average full run time: ~2min 3s)	1s	69ms	771ms	11s	185ms	3s	1s	5s	1min 26s
#3 History trend	916ms	69ms	768ms	12s	154ms	3s	2s	15s	Logs
builds... /	Apr 20 22:01	1 commit							

Type here to search

22:17 20-04-2023

The screenshot shows the Jenkins dashboard with a successful build for the project 'Insure-me'. The build status is 'Last Success' at 11 min, 'Last Failure' at 21 min, and 'Last Duration' at 2 min 3 sec. The Jenkins interface includes sections for People, Build History, Project Relationship, Check File Fingerprint, Manage Jenkins, and My Views. A build queue section indicates 'No builds in the queue.' Below the main dashboard, a taskbar shows various application icons.

These are the details of deploy server. Public IP: 65.2.170.196 and the ip is used to write the selenium script for automation and to verify the deployment.

The screenshot shows the AWS EC2 Instances details page for the instance i-03ef7b668fb9a17eb. The instance summary table provides the following information:

Instance ID	Public IPv4 address	Private IPv4 addresses
i-03ef7b668fb9a17eb (Deploy_Server)	65.2.170.196	ec2-65-2-170-196.ap-south-1.compute.amazonaws.com
IPv6 address	Instance state	Elastic IP addresses
-	Running	-
Hostname type	Private IP DNS name (IPv4 only)	AWS Compute Optimizer finding
IP name: ip-172-31-37-77.ap-south-1.compute.internal	ip-172-31-37-77.ap-south-1.compute.internal	Opt-in to AWS Compute Optimizer for recommendations.
Answer private resource DNS name	Instance type	Auto Scaling Group name
IPv4 (A)	t2.micro	-
Auto-assigned IP address	VPC ID	Learn more
65.2.170.196 [Public IP]	vpc-0243302b4b1acc768	
IAM Role	Subnet ID	
	subnet-029b370252239a107	



Here is the selenium script in eclipse and verified the automation testing on the server.

```

eclipse-workspace - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X selclass.java X Insure-me/pom.xml
1 package seltestfile;
2
3 import org.openqa.selenium.By;
4
5 public class selclass {
6
7     public static void main(String[] args) throws InterruptedException{
8         System.out.println("WELCOME");
9         // System.setProperty("webdriver.chrome.driver","C:\\Users\\SRINIVAS\\Downloads\\chromedriver.exe");
10        System.setProperty("webdriver.chrome.driver","/usr/bin/chromedriver");
11        ChromeOptions chromeOptions = new ChromeOptions();
12        chromeOptions.addArguments("--headless");
13        chromeOptions.addArguments("--no-sandbox");
14        chromeOptions.addArguments("--disable-dev-shm-usage");
15        WebDriver driver = new ChromeDriver(chromeOptions);
16        System.out.println("Driver initialized");
17        driver.get("http://65.2.170.196:8085/contact.html");
18        driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
19        driver.manage().window().maximize();
20        driver.findElement(By.id("inputname")).sendKeys("Srija");
21        driver.findElement(By.id("inputnumber")).sendKeys("123456789");
22        driver.findElement(By.id("inputmail1")).sendKeys("reachsrija@gmail.com");
23        driver.findElement(By.id("inputmessage")).sendKeys("Hello");
24        driver.findElement(By.id("my-button")).click();
25        Thread.sleep(2000);
26        String url=driver.getCurrentUrl();
27        if (url.equals("http://65.2.170.196:8085/contact.html")) {
28            System.out.println("Script Executed Successfully");
29        }
30    }
31
32 }
33
34 }
```

eclipse-workspace - insure-me[src/main/java/seltest/selclass.java] - Eclipse IDE

File Edit Source Refactor Source Navigate Search Project Run Window Help

Package Explorer X selclass.java X insure-me/pom.xml

1 package seltest;

Javadoc Declaration Console X Progress Servers

<terminated> selclass [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (21-Apr-2023, 8:05:20 pm - 8:05:41 pm) [pid: 12716]

WELCOME

20:05:23.798 [main] DEBUG io.netty.util.internal.logging.InternalLoggerFactory - Using SLF4J as the default logging framework

20:05:23.826 [main] DEBUG io.netty.util.ResourceLeakDetector - -Dio.netty.leakDetection.level: simple

20:05:23.827 [main] DEBUG io.netty.util.ResourceLeakDetector - -Dio.netty.leakDetection.targetRecords: 4

20:05:23.861 [main] DEBUG io.netty.util.ResourceLeakDetectorFactory - Loaded default ResourceLeakDetector: io.netty.util.ResourceLeakDetector@4

20:05:23.915 [main] DEBUG io.netty.util.internal.PlatformDependent0 - -Dio.netty.noUnsafe: false

20:05:23.916 [main] DEBUG io.netty.util.internal.PlatformDependent0 - Java version: 17

20:05:23.934 [main] DEBUG io.netty.util.internal.PlatformDependent0 - sun.misc.Unsafe.theUnsafe: available

20:05:23.935 [main] DEBUG io.netty.util.internal.PlatformDependent0 - sun.misc.Unsafe.copyMemory: available

20:05:23.937 [main] DEBUG io.netty.util.internal.PlatformDependent0 - sun.misc.Unsafe.storeFence: available

20:05:23.938 [main] DEBUG io.netty.util.internal.PlatformDependent0 - java.nio.Buffer.address: available

20:05:23.939 [main] DEBUG io.netty.util.internal.PlatformDependent0 - direct buffer constructor: unavailable: Reflective setAccessible(true) di

20:05:23.941 [main] DEBUG io.netty.util.internal.PlatformDependent0 - java.nio.Bits.unaligned: available, true

20:05:23.943 [main] DEBUG io.netty.util.internal.PlatformDependent0 - jdk.internal.misc.Unsafe.allocateUninitializedArray(int): unavailable: cl

20:05:23.946 [main] DEBUG io.netty.util.internal.PlatformDependent0 - sun.osx.A.dylib.DIRECTB2F.init(long, int): unavailable

20:05:23.946 [main] DEBUG io.netty.util.internal.PlatformDependent0 - sun.misc.Unsafe: available

20:05:23.971 [main] DEBUG io.netty.util.internal.PlatformDependent0 - sun.nio.ch.DirectBuffer: 2107637760 bytes (maybe)

20:05:23.972 [main] DEBUG io.netty.util.internal.PlatformDependent0 - Dio.netty.tmpdir: C:\Users\SRINIVAS\AppData\Local\Temp (java.io.tmpdir)

20:05:23.973 [main] DEBUG io.netty.util.internal.PlatformDependent0 - Dio.netty.bithode: 64 (sun.arch.data.model)

Platform: Windows

20:05:23.975 [main] DEBUG io.netty.util.internal.PlatformDependent0 - Dio.netty.maxDirectMemory: -1 bytes

20:05:23.975 [main] DEBUG io.netty.util.internal.PlatformDependent0 - Dio.netty.uninitializedNameAllocationThreshold: -1

20:05:23.986 [main] DEBUG io.netty.util.internal.CleanerJava9 - java.nio.ByteBuffer.cleaner(): available

20:05:23.990 [main] DEBUG io.netty.util.internal.PlatformDependent0 - -Dio.netty.noPreferDirect: false

20:05:24.093 [main] DEBUG io.netty.util.internal.PlatformDependent0 - org.jctools-core.MpscChunkedArrayQueue: available

20:05:24.594 [main] DEBUG io.netty.util.internal.InternalThreadLocalMap - -Dio.netty.threadLocalMap.stringBuilder.initialSize: 1024

20:05:24.595 [main] DEBUG io.netty.util.internal.InternalThreadLocalMap - -Dio.netty.threadLocalMap.stringBuilder.maxSize: 4096

20:05:25.300 [main] DEBUG io.netty.handler.ssl.JdkSslContext - Default protocols (JDK): [TLSv1.3, TLSv1.2]

20:05:25.300 [main] DEBUG io.netty.handler.ssl.JdkSslContext - Default cipher suites (JDK): [TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384, TLS_ECDHE]

20:05:25.468 [main] DEBUG io.netty.channel.MultithreadEventLoopGroup - -Dio.netty.eventLoopThreads: 8

20:05:25.509 [main] DEBUG io.netty.channel.nio.NioEventLoop - -Dio.netty.noKeySetOptimization: false

20:05:25.509 [main] DEBUG io.netty.channel.nio.NioEventLoop - -Dio.netty.selectorAutoRebuildThreshold: 512

20:05:25.853 [main] DEBUG io.netty.buffer.PooledByteBufAllocator - -Dio.netty.allocator.numHeapArenas: 8

20:05:26.024 [main] DEBUG io.netty.buffer.PooledByteBufAllocator - -Dio.netty.allocator.numDirectArenas: 0

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows the project structure for "insure-me".
- Java Editor:** Displays the code for "seltest.java".
- Console:** Shows the output of the Java application, which includes logs from the Java runtime environment and the application itself.
- Bottom Status Bar:** Shows system information like battery level, signal strength, and date/time.

```
java -version
java version "1.8.0_312"
Java(TM) SE Runtime Environment (build 1.8.0_312-b07)
Java HotSpot(TM) 64-Bit Server VM (build 25.312-b07, mixed mode)
```

```
20:05:25.854 [main] DEBUG io.netty.buffer.PooledByteBufAllocator - Dio.netty.allocator.pageSize: 8192
20:05:25.854 [main] DEBUG io.netty.buffer.PooledByteBufAllocator - Dio.netty.allocator.maxOrder: 9
20:05:25.854 [main] DEBUG io.netty.buffer.PooledByteBufAllocator - Dio.netty.allocator.chunkSize: 4194304
20:05:25.854 [main] DEBUG io.netty.buffer.PooledByteBufAllocator - Dio.netty.allocator.smallCacheSize: 256
20:05:25.854 [main] DEBUG io.netty.buffer.PooledByteBufAllocator - Dio.netty.allocator.normalCacheSize: 64
20:05:25.855 [main] DEBUG io.netty.buffer.PooledByteBufAllocator - Dio.netty.allocator.maxCachedBufferCapacity: 32768
20:05:25.855 [main] DEBUG io.netty.buffer.PooledByteBufAllocator - Dio.netty.allocator.cacheTrimInterval: 8192
20:05:25.855 [main] DEBUG io.netty.buffer.PooledByteBufAllocator - Dio.netty.allocator.cacheTrimIntervalMillis: 0
20:05:25.855 [main] DEBUG io.netty.buffer.PooledByteBufAllocator - Dio.netty.allocator.useCacheForAllThreads: false
20:05:25.855 [main] DEBUG io.netty.buffer.PooledByteBufAllocator - Dio.netty.allocator.maxCachedByteBuffersPerChunk: 1023
20:05:25.879 [main] DEBUG io.netty.buffer.ByteBufUtil - Dio.netty.allocator.type: pooled
20:05:25.879 [main] DEBUG io.netty.buffer.ByteBufUtil - Dio.netty.threadLocalDirectBufferSize: 16384
Starting ChromeDriver 112.0.5615.49 (bd2a7cb881c11e8cf03e7878078032934e3916914-refs/branch-heads/5615@{#936}) on port 36469
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
20:05:28.677 [Forwarding newSession on session null to remote] DEBUG io.netty.channel.DefaultChannelId - Dio.netty.processId: 12716 (auto-detected)
20:05:28.703 [Forwarding newSession on session null to remote] DEBUG io.netty.util.NetUtil - -Djava.net.preferIPv4Stack: false
20:05:28.703 [Forwarding newSession on session null to remote] DEBUG io.netty.util.NetUtil - -Djava.net.preferIPv6Addresses: false
20:05:28.718 [Forwarding newSession on session null to remote] DEBUG io.netty.util.NetUtilInitializations - Loopback interface: lo (Software Loopback Interface 1)
20:05:28.733 [Forwarding newSession on session null to remote] DEBUG io.netty.util.NetUtil - Failed to get SOMAXCONN from sysctl and file /proc/sys/net/core/somaxconn
20:05:28.761 [Forwarding newSession on session null to remote] DEBUG io.netty.channel.DefaultChannelId - Dio.netty.machinedId: 74:e5:43:ff:fe:20
20:05:29.136 [AsyncHttpClient-1-2] DEBUG io.netty.buffer.AbstractByteBuf - Dio.netty.buffer.checkAccessible: true
20:05:29.136 [AsyncHttpClient-1-2] DEBUG io.netty.buffer.AbstractByteBuf - Dio.netty.buffer.checkMarked: true
20:05:29.184 [AsyncHttpClient-1-2] DEBUG io.netty.util.ResourceLeakDetectorFactory - Loaded default ResourceLeakDetector: io.netty.util.ResourceLeakDetector
20:05:29.258 [AsyncHttpClient-1-2] DEBUG io.netty.handler.Recycler - Using new Channel [id: 0xe95d0b25, L:/127.0.0.1:36469]
20:05:29.258 [AsyncHttpClient-1-2] DEBUG io.netty.util.Recycler - Dio.netty.recycler.maxCapacityPerThread: 4096
20:05:29.258 [AsyncHttpClient-1-2] DEBUG io.netty.util.Recycler - Dio.netty.recycler.ratio: 8
20:05:29.258 [AsyncHttpClient-1-2] DEBUG io.netty.util.Recycler - Dio.netty.recycler.chunkSize: 32
20:05:29.259 [AsyncHttpClient-1-2] DEBUG io.netty.util.Recycler - Dio.netty.recycler.blocking: false
20:05:32.436 [AsyncHttpClient-1-2] DEBUG io.netty.handler.codec.compression.Brotli - brotli4j not in the classpath; Brotli support will be unavailable
20:05:32.436 [AsyncHttpClient-1-2] DEBUG org.asynchttpclient.netty.handler.HttpHandler -
```

eclipse-workspace - insure-me - src/main/java/seltest/selclass.java - Eclipse IDE

File Edit Source Refactor Source Navigate Search Project Run Window Help

Package Explorer *selclass.java insure-me/pom.xml

```

1 package seltest;

```

Javadoc Declaration Console Progress Servers

@Javadoc Declaration Console Progress Servers

Response DefaultHttpResponse(decodeResult: success, version: HTTP/1.1)
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
cache-control: no-cache
content-length: 14

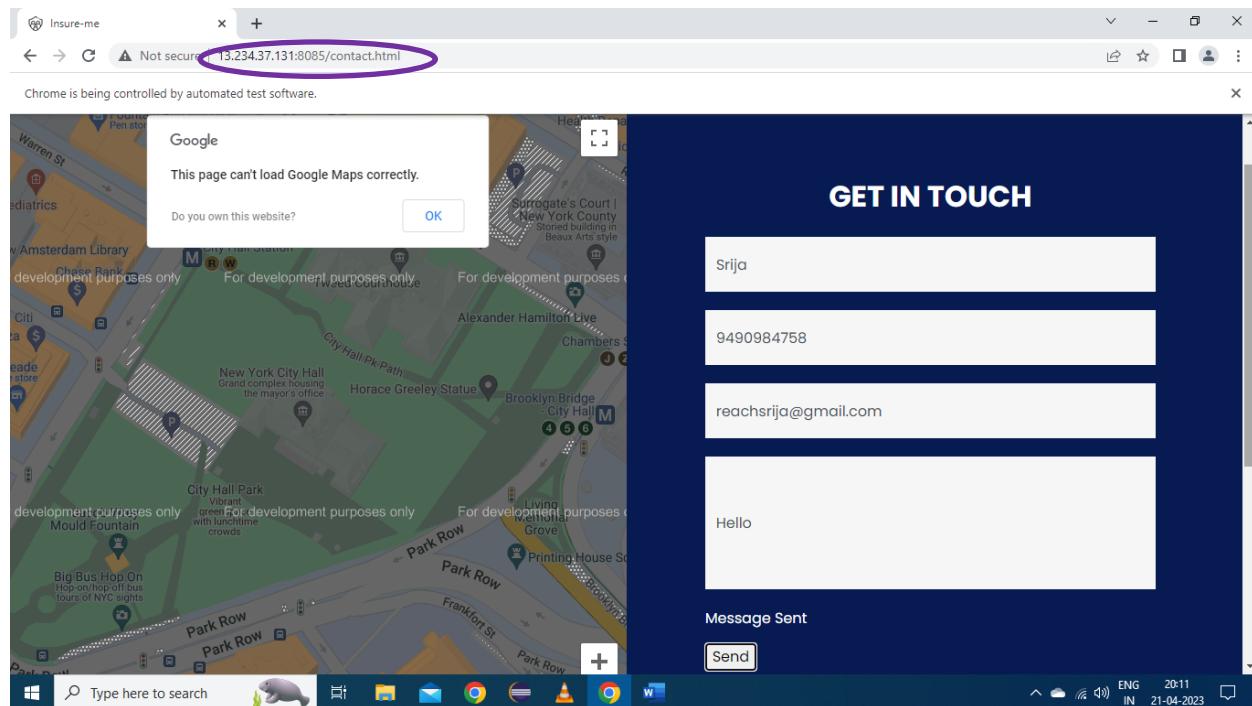
20:05:39.208 [AsyncHttpClient-1-2] DEBUG org.asynchttpclient.netty.channel.ChannelManager - Adding key: http://localhost:36469 for channel [id: 0x1000000000000000L]
20:05:39.612 [AsyncHttpClient-1-1] DEBUG org.asynchttpclient.netty.channel.DefaultChannelPool - Entry count for : http://localhost:36469 : 1
20:05:39.612 [AsyncHttpClient-1-1] DEBUG org.asynchttpclient.netty.channel.DefaultChannelPool - Entry count for : http://localhost:36469 : 1
20:05:40.208 [AsyncHttpClient-1-1] DEBUG org.asynchttpclient.netty.channel.DefaultChannelPool - Entry count for : http://localhost:36469 : 1
20:05:40.717 [AsyncHttpClient-1-1] DEBUG org.asynchttpclient.netty.channel.DefaultChannelPool - Entry count for : http://localhost:36469 : 1
20:05:41.212 [Forwarding getCurrentUrl on session 0deb16050c06228b7ab8de31adfe923e to remote] DEBUG org.asynchttpclient.netty.request.NettyRequest
20:05:41.212 [Forwarding getCurrentUrl on session 0deb16050c06228b7ab8de31adfe923e to remote] DEBUG org.asynchttpclient.netty.request.NettyRequest
20:05:41.258 [AsyncHttpClient-1-2] DEBUG org.asynchttpclient.netty.handler.HttpHandler - Script Executed Successfully

Request DefaultFullHttpRequest(decodeResult: success, version: HTTP/1.1, content: EmptyByteBuf@E)
GET /session/0deb16050c06228b7ab8de31adfe923e/uri HTTP/1.1
User-Agent: selenium/4.9.0 (java windows)
Cache-Control: no-cache
Content-Type: application/json; charset=utf-8
host: localhost:36469
accept: */*

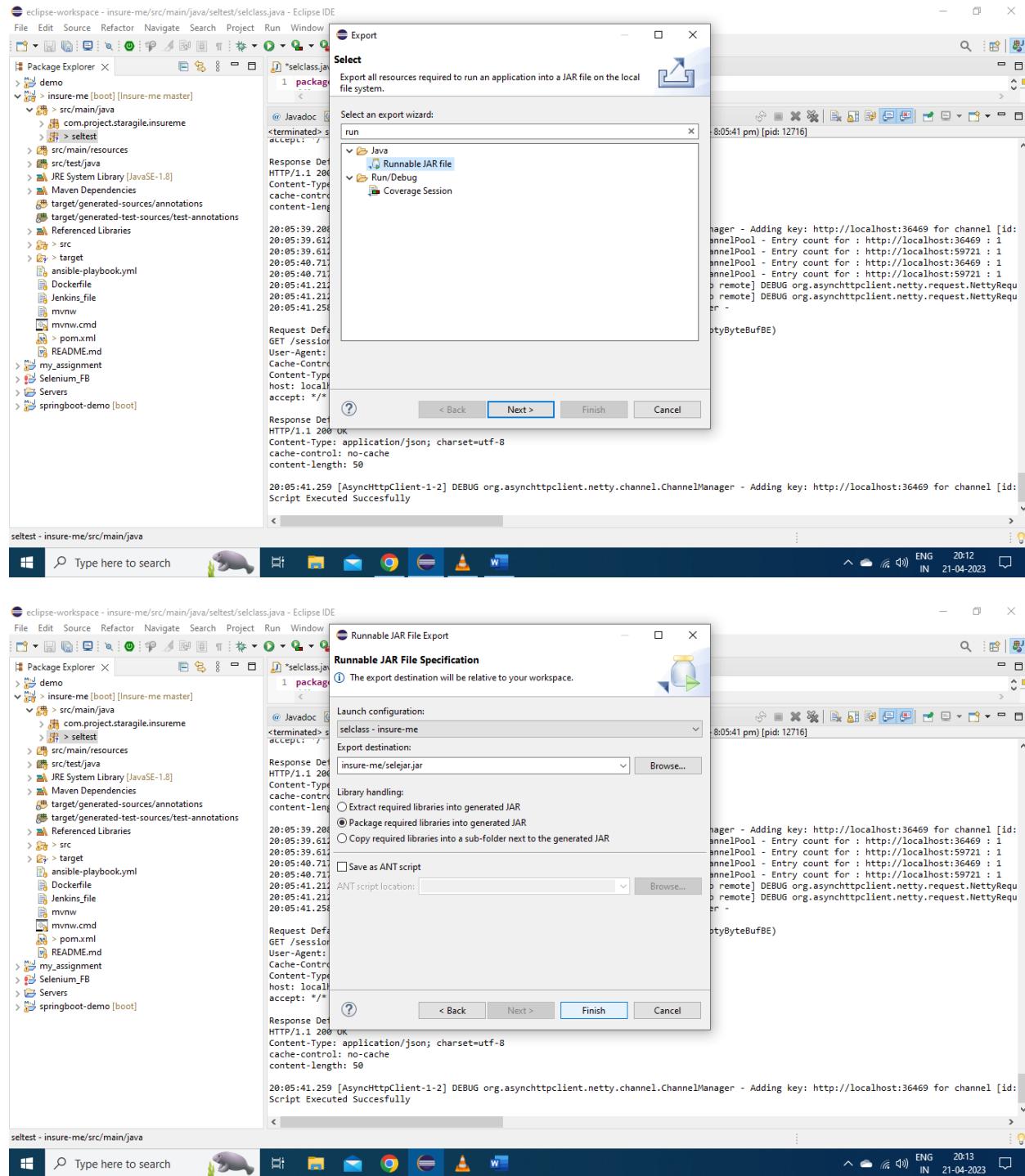
Response DefaultHttpResponse(decodeResult: success, version: HTTP/1.1)
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
cache-control: no-cache
content-length: 50

20:05:41.259 [AsyncHttpClient-1-2] DEBUG org.asynchttpclient.netty.channel.ChannelManager - Adding key: http://localhost:36469 for channel [id: 0x1000000000000000L]
Script Executed Successfully

Type here to search



The selenium script is exported as runnable jar and pushed on to GitHub.



The screenshot shows a GitHub repository page for 'Srija1991/Jsure-me'. The 'Code' tab is selected, showing the 'master' branch with 1 branch and 0 tags. The commit history lists several commits, with one commit at the bottom circled in green: 'JAR FILE ADDED2' by Srija1991, 1 hour ago.

The chromedriver and google-chrome application of same version is installed on the jenkins server.

The screenshot shows an AWS CloudShell terminal window. The user runs the command 'google-chrome --version' and 'chromedriver --version'. Both commands return the same output: 'Google Chrome 112.0.5615.165' and 'Chromedriver 112.0.5615.49'. This indicates that both applications are installed and are of the same version.

```
root@ip-172-31-34-123:/home/ubuntu# google-chrome --version
Google Chrome 112.0.5615.165
root@ip-172-31-34-123:/home/ubuntu# whereis google-chrome
google-chrome: /usr/bin/google-chrome /usr/share/man/man1/google-chrome.1.gz
root@ip-172-31-34-123:/home/ubuntu# chromedriver --version
Chromedriver 112.0.5615.49 (bd2a7bcd81c11e8cf3078709362934e3916914-refs/branch-heads/5615@{#936})
root@ip-172-31-34-123:/home/ubuntu# whereis chromedriver
chromedriver: /usr/bin/chromedriver
root@ip-172-31-34-123:/home/ubuntu#
```

A new job named Selenium -test is created and coded to run the selejar.jar file. This job is triggered after the Insure-me build is success.

The screenshot shows the Jenkins configuration page for the 'Selenium Test' job. The 'General' tab is selected. The 'Description' field contains the text: 'Test the contact page on Insure-me'. Under the 'Build Triggers' section, the 'Trigger only if build is stable' option is highlighted with a green oval. Other trigger options shown include 'Trigger even if the build is unstable', 'Trigger even if the build fails', 'Always trigger, even if the build is aborted', 'Build periodically', 'GitHub hook trigger for GITScm polling', 'Poll SCM', and 'Quiet period'. The Jenkins interface includes a top navigation bar with various links like 'Instance details', 'EC2 Instance Connect', 'Selenium Test Config', 'Srija1991/Insure-me', and 'Insure-me'. The status bar at the bottom shows 'ENG IN 14:33 22-04-2023'.

The screenshot shows the Jenkins Pipeline configuration interface. The pipeline script is defined as follows:

```
1 ~ node{  
2 ~   stage('Git Checkout'){  
3 ~     git 'https://github.com/Srija1991/Insure-me.git'  
4 ~     echo 'Git Checkout Successful'  
5 ~   }  
6 ~   script{  
7 ~     run 'java -jar selejar.jar'  
8 ~   }  
9 ~ }
```

A green oval highlights the command `sh 'java -jar selejar.jar'` in the script editor.

The screenshot shows the Jenkins Console Output page for build #8. The build log displays the following output:

```
Started by user Srija Bichala  
Obtained Jenkins_file from git https://github.com/Srija1991/Insure-me.git  
[Pipeline] Start of Pipeline  
[Pipeline] node  
Running on Jenkins in /var/lib/jenkins/workspace/Insure-me  
[Pipeline] {  
[Pipeline] stage  
[Pipeline] { (Declarative: Checkout SCM)  
[Pipeline] checkout  
The recommended git tool is: git  
No credentials specified  
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Insure-me/.git # timeout=10  
Fetching changes from the remote Git repository  
> git config remote.origin.url https://github.com/Srija1991/Insure-me.git # timeout=10  
Fetching upstream changes from https://github.com/Srija1991/Insure-me.git  
> git --version # timeout=10  
> git --version # 'git version 2.34.1'  
> git fetch --tags --force --progress -- https://github.com/Srija1991/Insure-me.git +refs/heads/*:refs/remotes/origin/* #
```

A green oval highlights the URL `https://github.com/Srija1991/Insure-me.git` in the log output.

Instance details | EC2 x EC2 Instance Connect x EC2 Instance Connect x Insure-me #8 Console x Srijaa1991/Insure-me: x Insure-me x +

Not secure | ec2-13-126-50-152.ap-south-1.compute.amazonaws.com:8080/job/Insure-me/8/console

Simple-DevOps-Pro... Dec-3rd Batch - Go... Docker Hub Architecture Comp... Application 3 tier A... Jenkins Tomcat Dep... Devops GI (03 Dec... Book Your Appoint... Weekly Feedback F...

Dashboard > Insure-me > #8

```
TASK [Start Docker Service] ****
changed: [172.31.37.77]

TASK [Deploy Docker Container] ****
changed: [172.31.37.77]

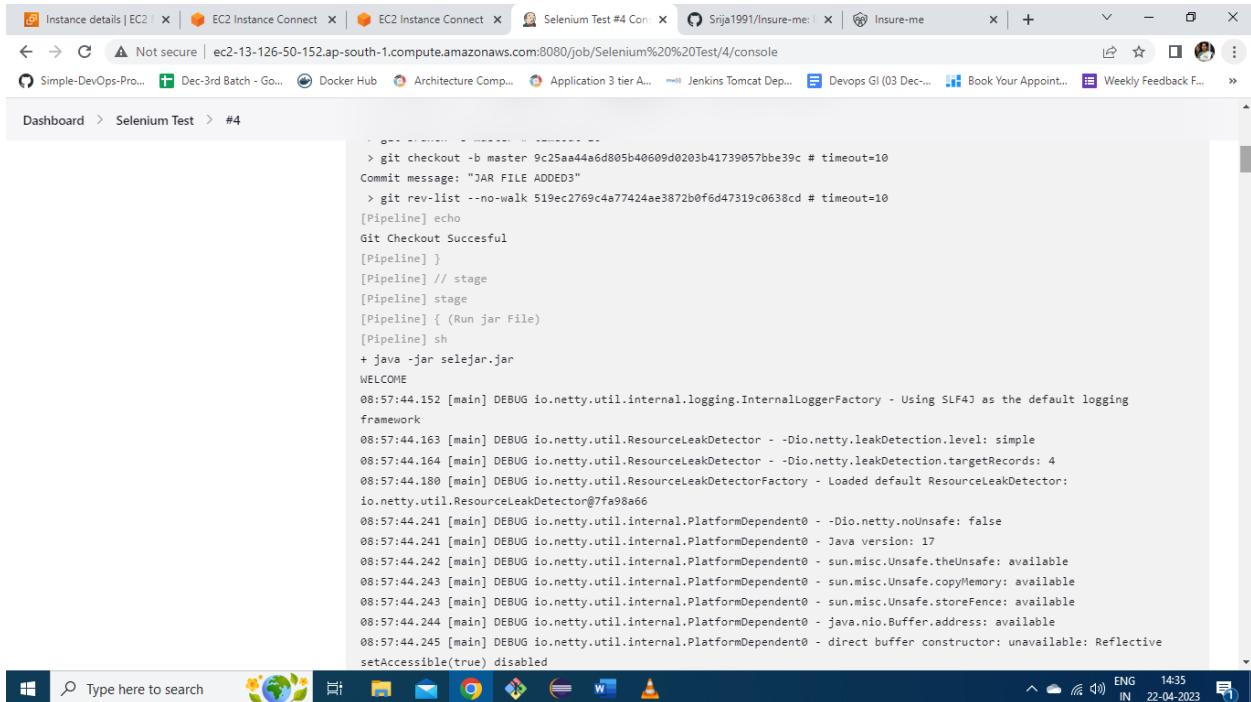
PLAY RECAP ****
172.31.37.77 : ok=6    changed=5    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

[Pipeline] echo
*****Deployment onto Server is Sucessful*****
[Pipeline]
[Pipeline] // withEnv
[Pipeline]
[Pipeline] // stage
[Pipeline]
[Pipeline] // withEnv
[Pipeline]
[Pipeline] // withEnv
[Pipeline]
[Pipeline] // withEnv
[Pipeline]
[Pipeline] // node
[Pipeline] // pipeline
Triggering a new build of Selenium Test #4
Finished: SUCCESS
```

Here the Insure-me build is success and Selenium job is triggered and this job is success. And it is showing the “script is executed successfully”.

The screenshot shows the Jenkins interface for a Selenium Test job. The left sidebar lists various options like Status, Changes, Console Output (which is selected and highlighted with a green oval), View as plain text, Edit Build Information, Delete build '#4', Git Build Data, Replay, Pipeline Steps, Workspaces, and Previous Build. The main content area is titled 'Console Output' with a green checkmark icon. It displays the following log output:

```
Started by upstream project "Insure-me" build number 8
originally caused by:
Started by user Srijा Bichala
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/Selenium Test
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Git Checkout)
[Pipeline] git
The recommended git tool is: git
No credentials specified
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Selenium Test/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/SrijाBichala/Insure-me.git # timeout=10
Fetching upstream changes from https://github.com/SrijाBichala/Insure-me.git
> git --version # timeout=10
> git --version # 'git version 2.34.1'
```



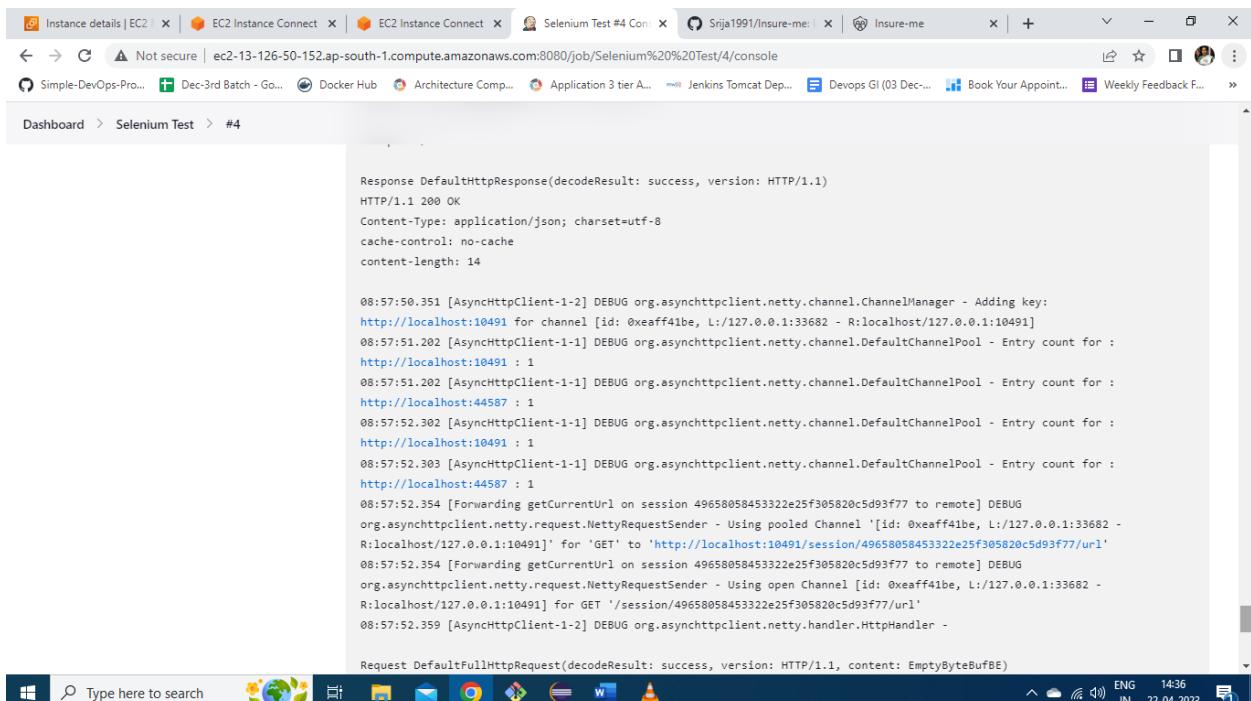
Instance details | EC2 | EC2 Instance Connect | EC2 Instance Connect | Selenium Test #4 Con... | Srijan1991/Insure-me | Insure-me | + | - | X

← → C Not secure | ec2-13-126-50-152.ap-south-1.compute.amazonaws.com:8080/job/Selenium%20%20Test/4/console

Simple-DevOps-Pro... Dec-3rd Batch - Go... Docker Hub Architecture Comp... Application 3 tier A... Jenkins Tomcat Dep... Devops GI (03 Dec-... Book Your Appoint... Weekly Feedback F...

Dashboard > Selenium Test > #4

```
> git checkout -b master 9c25aa44a6d805b40609d0203b41739057bbe39c # timeout=10
Commit message: "JAR FILE ADDED3"
> git rev-list --no-walk 519ec2769c4a77424ae3872b0f6d47319c0638cd # timeout=10
[Pipeline] echo
Git Checkout Successful
[Pipeline]
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Run jar File)
[Pipeline] sh
+ java -jar selejar.jar
WELCOME
08:57:44.152 [main] DEBUG io.netty.util.internal.logging.InternalLoggerFactory - Using SLF4J as the default logging framework
08:57:44.163 [main] DEBUG io.netty.util.ResourceLeakDetector - -Dio.netty.leakDetection.level: simple
08:57:44.164 [main] DEBUG io.netty.util.ResourceLeakDetector - -Dio.netty.leakDetection.targetRecords: 4
08:57:44.180 [main] DEBUG io.netty.util.ResourceLeakDetectorFactory - Loaded default ResourceLeakDetector: io.netty.util.ResourceLeakDetector@7fa98a66
08:57:44.241 [main] DEBUG io.netty.util.internal.PlatformDependent0 - -Dio.netty.noUnsafe: false
08:57:44.241 [main] DEBUG io.netty.util.internal.PlatformDependent0 - Java version: 17
08:57:44.242 [main] DEBUG io.netty.util.internal.PlatformDependent0 - sun.misc.Unsafe.theUnsafe: available
08:57:44.243 [main] DEBUG io.netty.util.internal.PlatformDependent0 - sun.misc.Unsafe.copyMemory: available
08:57:44.243 [main] DEBUG io.netty.util.internal.PlatformDependent0 - sun.misc.Unsafe.storeFence: available
08:57:44.244 [main] DEBUG io.netty.util.internal.PlatformDependent0 - java.nio.Buffer.address: available
08:57:44.245 [main] DEBUG io.netty.util.internal.PlatformDependent0 - direct buffer constructor: unavailable: Reflective setAccessible(true) disabled
```



Instance details | EC2 | EC2 Instance Connect | EC2 Instance Connect | Selenium Test #4 Con... | Srijan1991/Insure-me | Insure-me | + | - | X

← → C Not secure | ec2-13-126-50-152.ap-south-1.compute.amazonaws.com:8080/job/Selenium%20%20Test/4/console

Simple-DevOps-Pro... Dec-3rd Batch - Go... Docker Hub Architecture Comp... Application 3 tier A... Jenkins Tomcat Dep... Devops GI (03 Dec-... Book Your Appoint... Weekly Feedback F...

Dashboard > Selenium Test > #4

```
Response DefaultHttpResponse(decodeResult: success, version: HTTP/1.1)
HTTP/1.1 200 OK
Content-Type: application/json; charset=utf-8
cache-control: no-cache
content-length: 14

08:57:50.351 [AsyncHttpClient-1-2] DEBUG org.asynchttpclient.netty.channel.ChannelManager - Adding key: http://localhost:10491 for channel [id: 0xeaff41be, L:/127.0.0.1:33682 - R:localhost/127.0.0.1:10491]
08:57:51.202 [AsyncHttpClient-1-1] DEBUG org.asynchttpclient.netty.channel.DefaultChannelPool - Entry count for: http://localhost:10491 : 1
08:57:51.202 [AsyncHttpClient-1-1] DEBUG org.asynchttpclient.netty.channel.DefaultChannelPool - Entry count for: http://localhost:44587 : 1
08:57:52.302 [AsyncHttpClient-1-1] DEBUG org.asynchttpclient.netty.channel.DefaultChannelPool - Entry count for: http://localhost:10491 : 1
08:57:52.303 [AsyncHttpClient-1-1] DEBUG org.asynchttpclient.netty.channel.DefaultChannelPool - Entry count for: http://localhost:44587 : 1
08:57:52.354 [Forwarding getCurrentUrl on session 49658058453322e25f305820c5d93f77 to remote] DEBUG org.asynchttpclient.netty.request.NettyRequestSender - Using pooled Channel [id: 0xeaff41be, L:/127.0.0.1:33682 - R:localhost/127.0.0.1:10491] for 'GET' to 'http://localhost:10491/session/49658058453322e25f305820c5d93f77?url'
08:57:52.354 [Forwarding getCurrentUrl on session 49658058453322e25f305820c5d93f77 to remote] DEBUG org.asynchttpclient.netty.request.NettyRequestSender - Using open Channel [id: 0xeaff41be, L:/127.0.0.1:33682 - R:localhost/127.0.0.1:10491] for GET '/session/49658058453322e25f305820c5d93f77?url'
08:57:52.359 [AsyncHttpClient-1-2] DEBUG org.asynchttpclient.netty.handler.HttpHandler -
```

Request DefaultFullHttpRequest(decodeResult: success, version: HTTP/1.1, content: EmptyByteBufBE)

```
08:57:52.359 [AsyncHttpClient-1-2] DEBUG org.asynchttpclient.netty.handler.HttpHandler -  
Request DefaultFullHttpRequest(decodeResult: success, version: HTTP/1.1, content: EmptyByteBuf@E)  
GET /session/49658058453322e25f305820c5d93f77?url HTTP/1.1  
User-Agent: selenium/4.9.0 (java unix)  
Cache-Control: no-cache  
Content-Type: application/json; charset=utf-8  
host: localhost:10491  
accept: */*  
  
Response DefaultHttpResponse(decodeResult: success, version: HTTP/1.1)  
HTTP/1.1 200 OK  
Content-Type: application/json; charset=utf-8  
cache-control: no-cache  
content-length: 49  
  
08:57:52.359 [AsyncHttpClient-1-2] DEBUG org.asynchttpclient.netty.channel.ChannelManager - Adding key:  
http://localhost:10491 for channel [id: 0x6fa1be, L:/127.0.0.1:33682 - R:localhost/127.0.0.1:10491]  
Script Executed Successfully  
[Pipeline]  
[Pipeline] // stage  
[Pipeline]  
[Pipeline] // node  
[Pipeline] End of Pipeline  
Finished: SUCCESS
```

THANK YOU