Greendeck

Take Home Assignment Report

For the Post of Devops Engineer

By: Arun AVK -18BCS012 , IIIT Dharwad

CONTENTS

1. INTRODUCTION

DevOps is a set of practices that combines software development and IT operations. It aims to shorten the systems development life cycle and provide continuous delivery with high software quality. DevOps is complementary with Agile software development; several DevOps aspects came from the Agile methodology.

2. STEPS FOLLOWED FOR EXECUTION

2.1 Setting up Instances :

For this project, we have created two instances in AWS named Web_Server , Jenkins_Server. Where,

Web_Server : The Webpage will be updated in this file; Tomcat will be installed in this server.

Jenkins_Server : The jenkins will be installed and managed.

2.2 Web_Server Configuration :

Firstly , we create an EC2 instance in Redhat Linux OS , We access it with the terminal in my Local Machine. We install Java With the following commands:

```
$ sudo yum install java-1.8.0-openjdk-devel
```

Secondly, we install Apache tomcat in the system with the following command sets;

```
wget http://www-eu.apache.org/dist/tomcat/tomcat-9/v9.0.0.M17/bin/apache-tomcat-9.0.0.M17.tar.gz

tar -zxpvf apache-tomcat-8.5.72

cd apache-tomcat-8.5.72

mv apache-tomcat-8.5.72

vi /opt/apache-tomcat-8.5.72/conf/tomcat-users.xml
```

Add the credentials :

```
<role rolename="manager-gui"/>
<role rolename="manager-script"/>
<role rolename="manager-jmx"/>
<role rolename="manager-status"/>
<user username="admin" password="admin" roles="manager-gui, manager-script, manager-jmx, manager-status"/>
<user username="deployer" password="deployer" roles="manager-scrpt"/>
<user username="tomcat" password="tomcat" roles="manager-gui"/>
```

type :wq to save the file and restart the instance.

cd /opt/apache-tomcat-8.5.72/bin/

./startup.sh

Now Your Apache tomcat page will be available for access in

http://3.144.32.41:8090/

Credentials:
For,GUI only, Username: tomcat | Password: tomcat
For GUI & Script, Username: deployer | Password: deployer
For Complete access, Username: admin | Password: admin

So, your initial setup for the web server is done.

2.3 Jenkins_Server configuration :

Firstly , we create an EC2 instance in the Redhat Linux OS. We access it with the terminal in my Local Machine. We install Java With the following commands:
$ sudo yum install java-1.8.0-openjdk-devel

Secondly, we install Jenkins in the system with the following command sets;

sudo wget -O /etc/yum.repos.d/jenkins.repo
https://pkg.jenkins.io/redhat-stable/jenkins.repo

sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key

yum install java-11-openjdk-devel

yum install jenkins

To check jenkins , type “service jenkins start”

Then,type “service jenkins status”
You can access the jenkins dashboard with the link below:

http://18.222.186.171:8080/

Credentials : Username: admin | Password: admin

As soon as you get into your Dashboard, Head to “**Manage Jenkins**” ,
Click on “**Manage Plugins**” and install the following plugins;

- Github
- Maven Invoker
- Deploy to container

After this Install and configure **Maven** in the **Jenkins_server**,

Maven Installation & Configuration:

Install Maven in the instance using the following commands,

```
sudo wget https://dlcdn.apache.org/maven/maven-3/3.8.3/binaries/apache-maven-3.8.3-bin.tar.gz
```

Setup M2_HOME & M2 PATH in .bash_profile of the user.

```
M2_HOME=/opt/maven/apache-maven-3.8.3
M2=$M2_HOME/bin
```

Head to **“Manage Jenkins”** in Jenkins Dashboard , click “**Global Tool Configuration**” and specify your maven path.

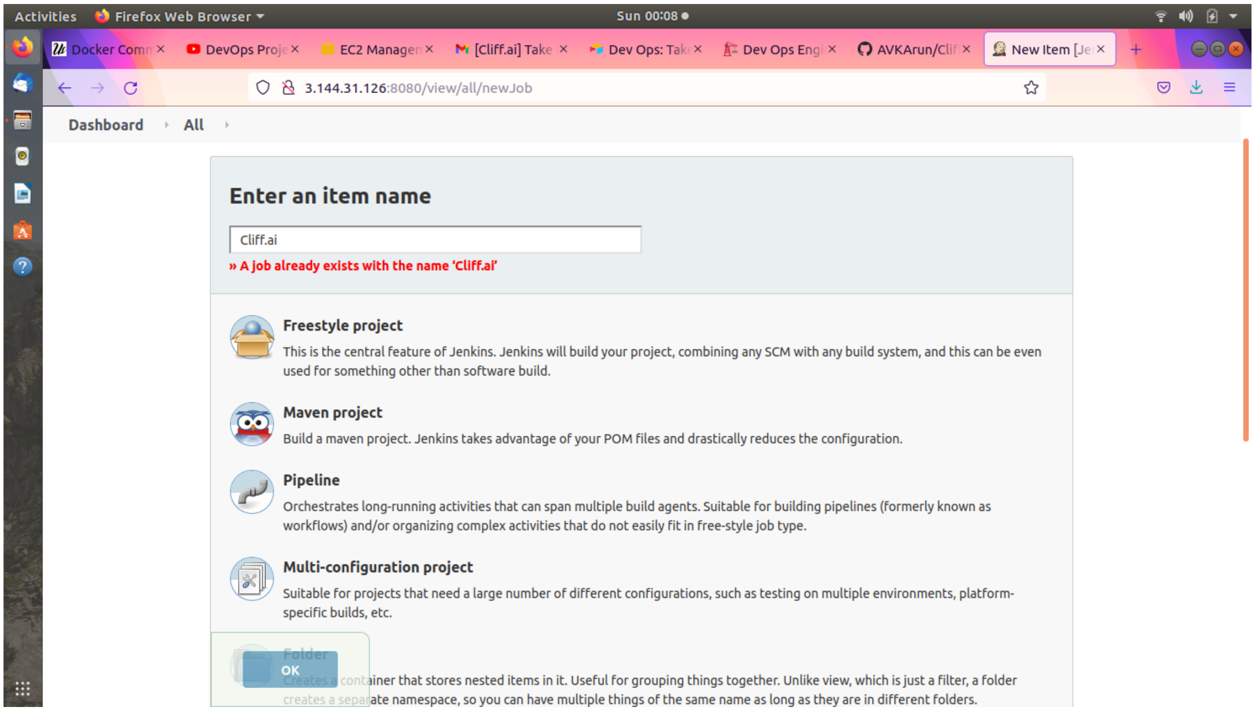
Now, Create your first job in Jenkins Dashboard,

Go to “Manage Jenkins” , Add credentials of the tomcat server to access tomcat With Jenkins .

In our Case, We have added the Credentials of the user called “deployer” .

Click on “Create Job”.

Create a “Maven Project” and name it as “Cliff.ai_Assignment” & Click on “OK”.



Enter an item name

Cliff.ai

» A job already exists with the name ‘Cliff.ai’

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

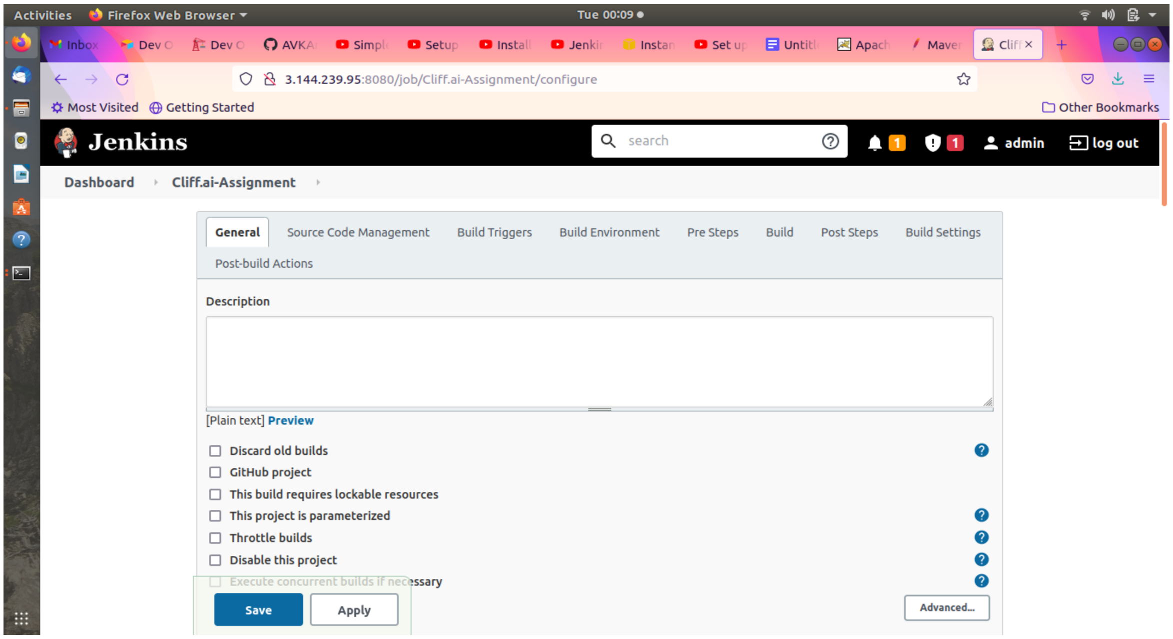
Maven project
Build a maven project. Jenkins takes advantage of your POM files and drastically reduces the configuration.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

Add Description if you want , it’s not mandatory. Don’t select anything in this General column.

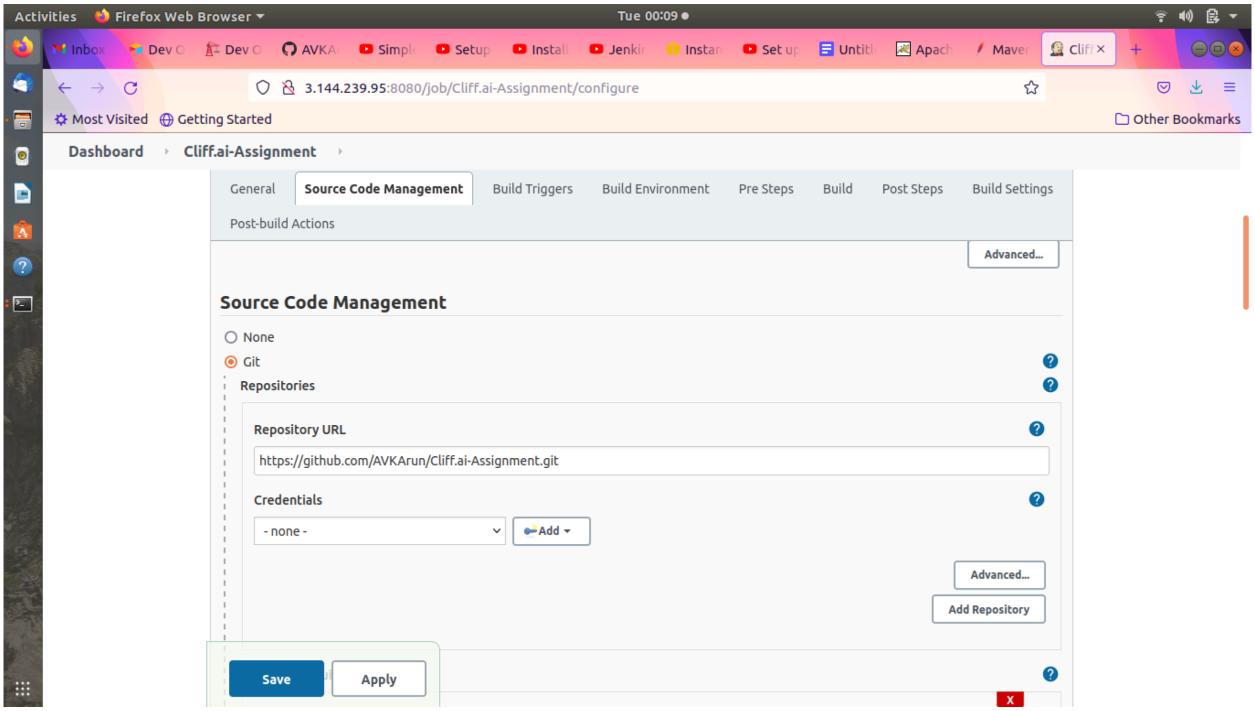


Description

[Plain text] Preview

☐ Discard old builds
☐ GitHub project
☐ This build requires lockable resources
☐ This project is parameterized
☐ Throttle builds
☐ Disable this project
☐ Execute concurrent builds if necessary

Add your Git Repository and provide credentials if your repository is private & mention the branch in the below box.



3.144.239.95:8080/job/Cliff.ai-Assignment/configure

Source Code Management

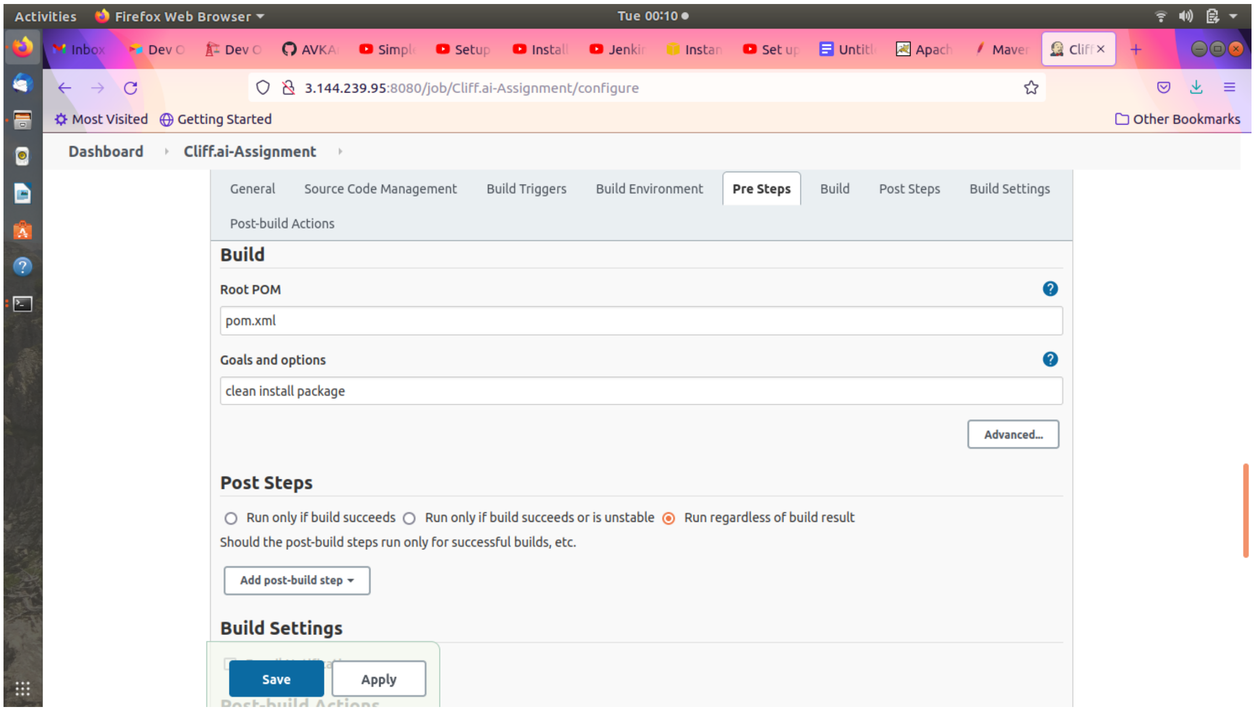
○ None
● Git

Repositories

Repository URL: https://github.com/AVKArun/Cliff.ai-Assignment.git

Credentials: - none -

After adding your git repository, you should be getting a pom.xml file automatically inthe Root POM ; In Goals ,Write “clean install package”.



3.144.239.95:8080/job/Cliff.ai-Assignment/configure

Build

Root POM: pom.xml

Goals and options: clean install package

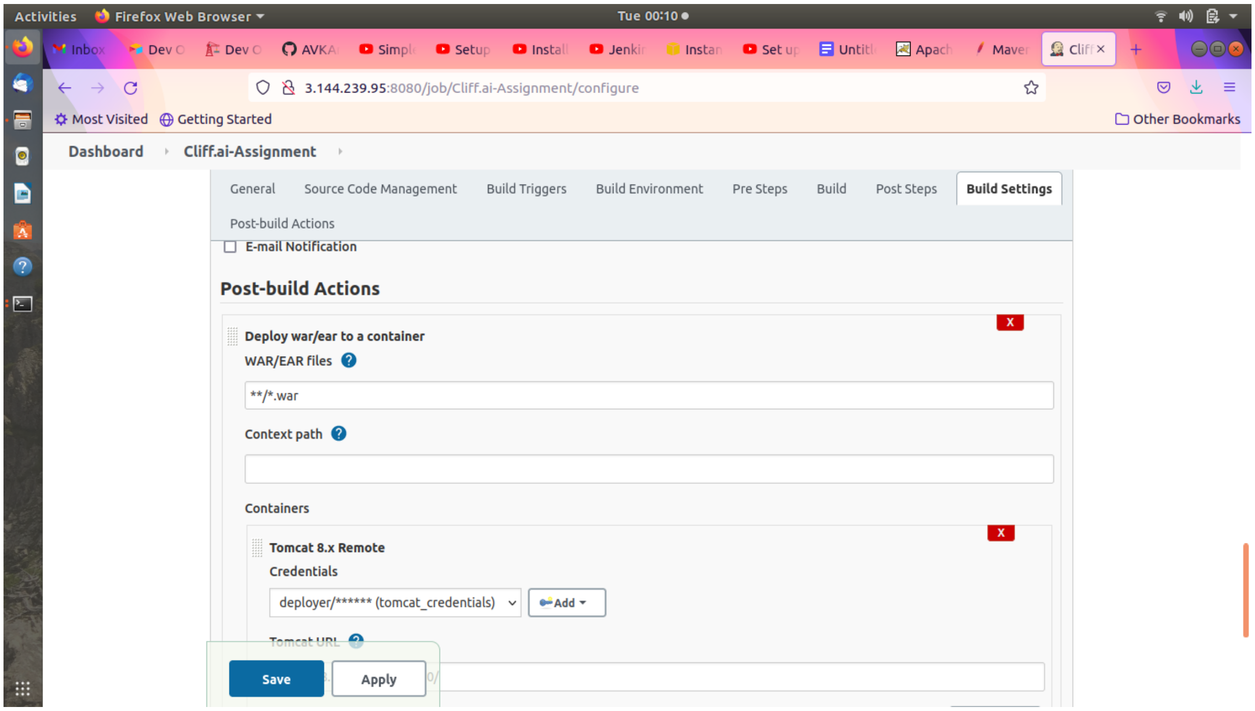
Post Steps

○ Run only if build succeeds ○ Run only if build succeeds or is unstable ● Run regardless of build result

Should the post-build steps run only for successful builds, etc.

Build Settings

In Post-build Actions, Select “Deploy war/ear to a container” ; Add “/*.war” in the first column. Add container “Tomcat 8.x Remote” and add the saved deployer credentials.**



□ E-mail Notification

Post-build Actions

Deploy war/ear to a container

WAR/EAR files: **/*.war

Context path:

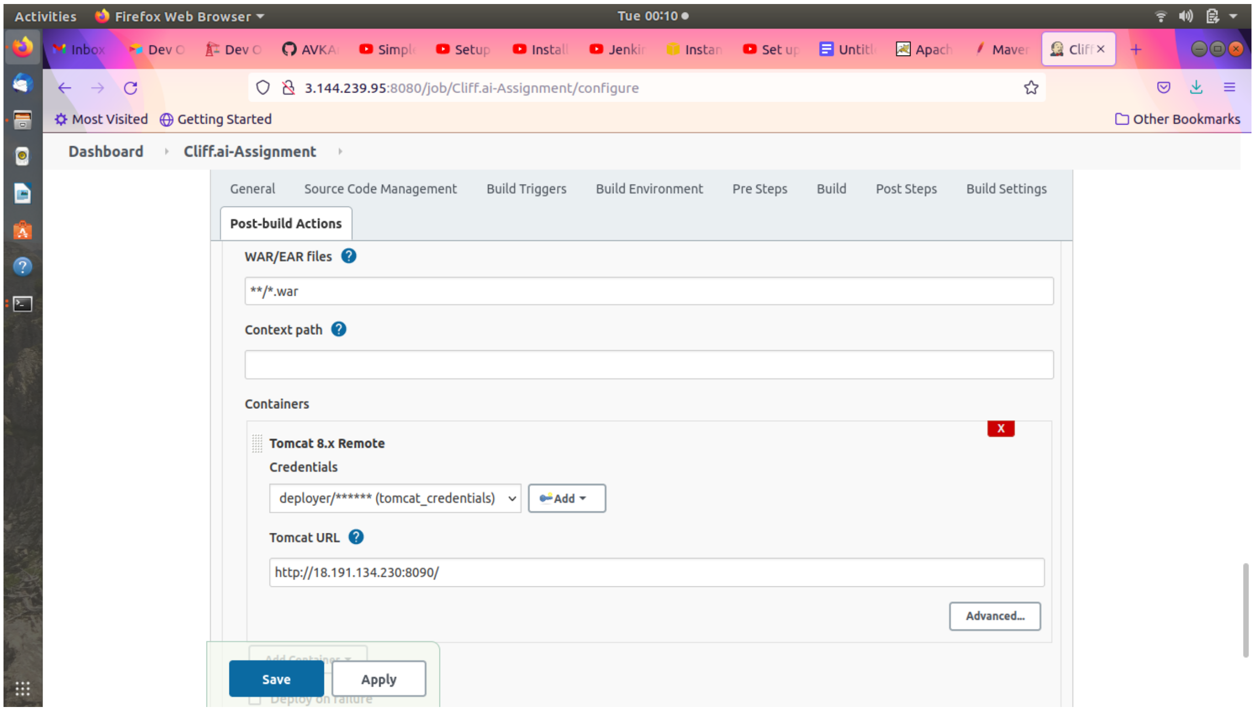
Containers

Tomcat 8.x Remote

Credentials: deployer/****** (tomcat_credentials)

Tomcat URL

Add the tomcat server url in “Tomcat URL”



WAR/EAR files: **/*.war

Context path:

Containers

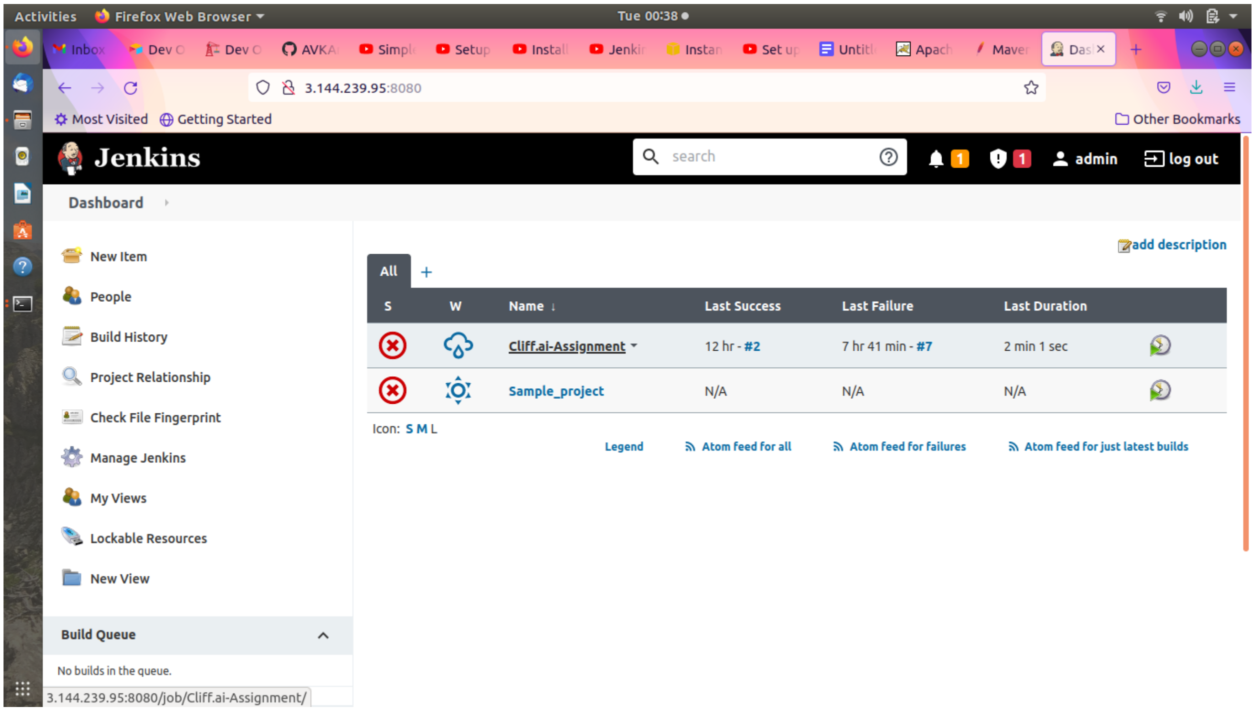
Tomcat 8.x Remote

Credentials: deployer/****** (tomcat_credentials)

Tomcat URL: http://18.191.134.230:8090/

Click on “APPLY” & “SAVE”

Now,click on the “Cliff.ai-Assignment pipeline”.



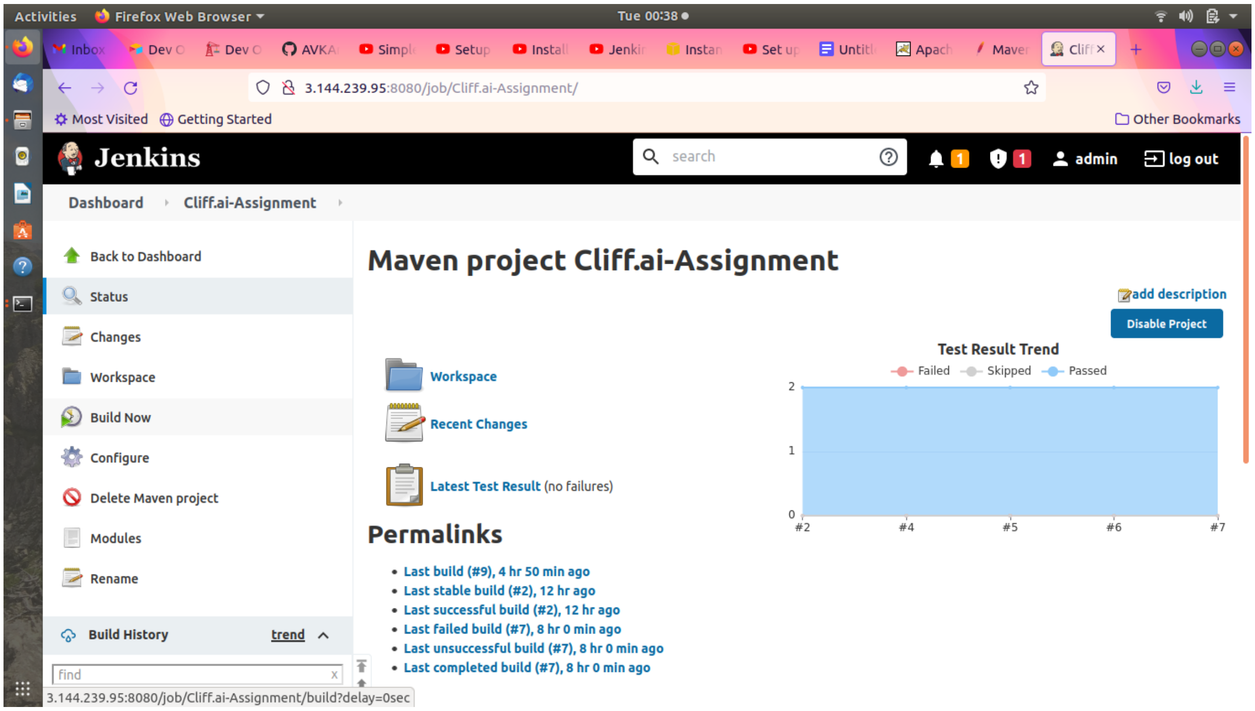
S	W	Name ↓	Last Success	Last Failure	Last Duration
⊗		Cliff.ai-Assignment	12 hr - #2	7 hr 41 min - #7	2 min 1 sec
⊗		Sample_project	N/A	N/A	N/A

Icon: S M L

No builds in the queue.

3.144.239.95:8080/job/Cliff.ai-Assignment/

Click on “Build Now” in the left side column.



Maven project Cliff.ai-Assignment

Latest Test Result (no failures)

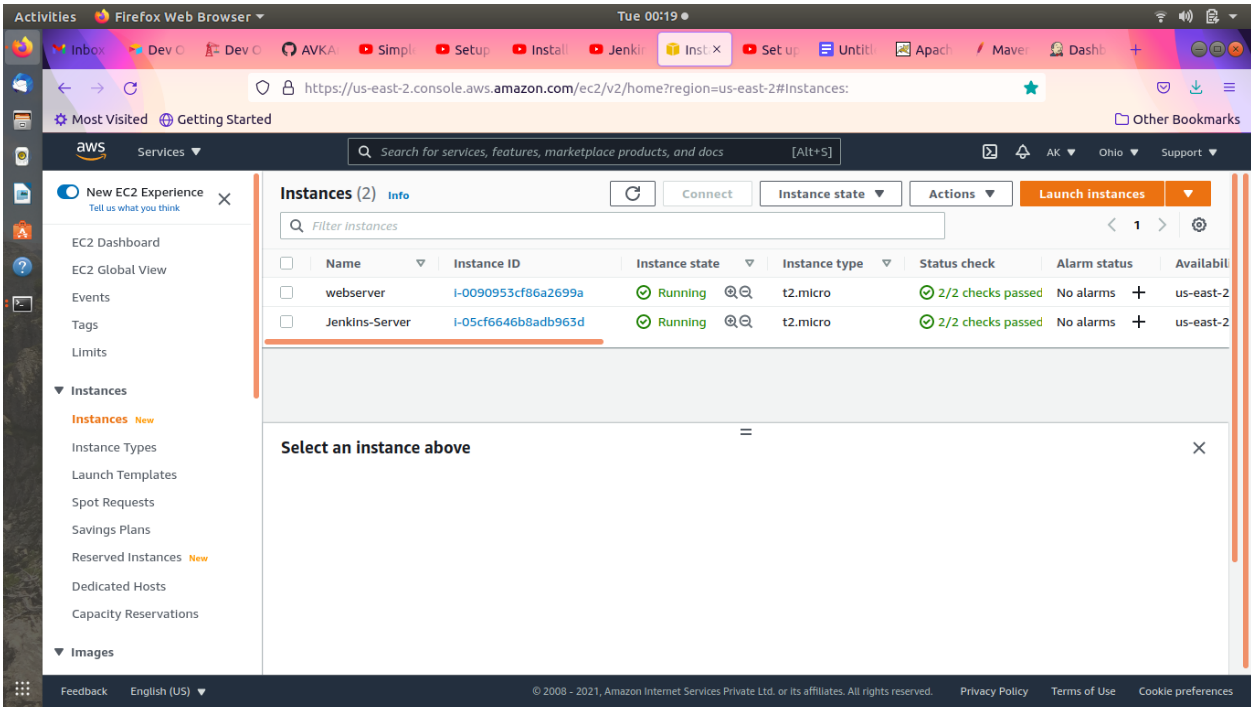
Permalinks

- Last build (#9), 4 hr 50 min ago
- Last stable build (#2), 12 hr ago
- Last successful build (#2), 12 hr ago
- Last failed build (#7), 8 hr 0 min ago
- Last unsuccessful build (#7), 8 hr 0 min ago
- Last completed build (#7), 8 hr 0 min ago

3.144.239.95:8080/job/Cliff.ai-Assignment/build?delay=0sec

OUTPUT

AWS EC2 Instances:

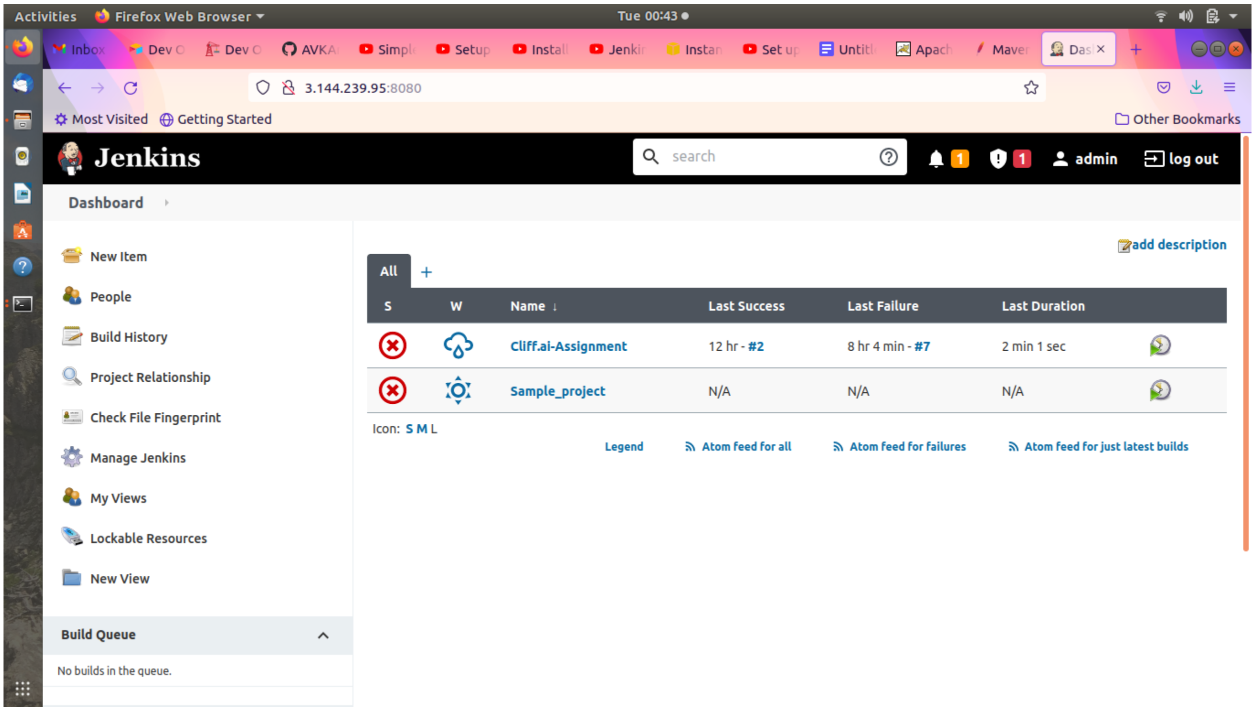


Instances (2) Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availabil
webserver	i-0090953cf86a2699a	Running	t2.micro	2/2 checks passed	No alarms	us-east-2
Jenkins-Server	i-05cf6646b8adb963d	Running	t2.micro	2/2 checks passed	No alarms	us-east-2

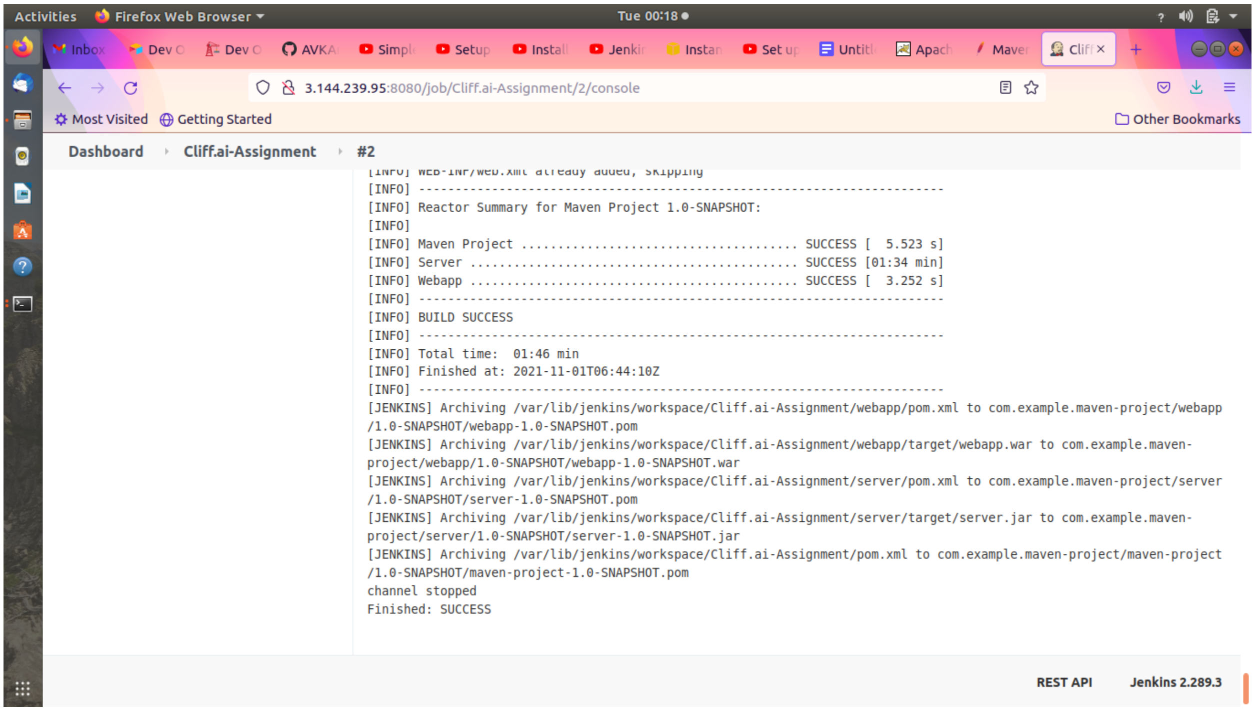
Select an instance above

Jenkins Dashboard:



S	W	Name ↓	Last Success	Last Failure	Last Duration
⊗		Cliff.ai-Assignment	12 hr - #2	8 hr 4 min - #7	2 min 1 sec
⊗		Sample_project	N/A	N/A	N/A

Build Report:



```
[INFO] WEB-INF/web.xml already added, skipping
[INFO] ------------------------------------------------------------------------
[INFO] Reactor Summary for Maven Project 1.0-SNAPSHOT:
[INFO]
[INFO] Maven Project ...................................... SUCCESS [  5.523 s]
[INFO] Server ............................................. SUCCESS [01:34 min]
[INFO] Webapp ............................................. SUCCESS [  3.252 s]
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  01:46 min
[INFO] Finished at: 2021-11-01T06:44:10Z
[INFO] ------------------------------------------------------------------------
[JENKINS] Archiving /var/lib/jenkins/workspace/Cliff.ai-Assignment/webapp/pom.xml to com.example.maven-project/webapp
/1.0-SNAPSHOT/webapp-1.0-SNAPSHOT.pom
[JENKINS] Archiving /var/lib/jenkins/workspace/Cliff.ai-Assignment/webapp/target/webapp.war to com.example.maven-
project/webapp/1.0-SNAPSHOT/webapp-1.0-SNAPSHOT.war
[JENKINS] Archiving /var/lib/jenkins/workspace/Cliff.ai-Assignment/server/pom.xml to com.example.maven-project/server
/1.0-SNAPSHOT/server-1.0-SNAPSHOT.pom
[JENKINS] Archiving /var/lib/jenkins/workspace/Cliff.ai-Assignment/server/target/server.jar to com.example.maven-
project/server/1.0-SNAPSHOT/server-1.0-SNAPSHOT.jar
[JENKINS] Archiving /var/lib/jenkins/workspace/Cliff.ai-Assignment/pom.xml to com.example.maven-project/maven-project
/1.0-SNAPSHOT/maven-project-1.0-SNAPSHOT.pom
channel stopped
Finished: SUCCESS
```

4. PROJECT URLS

GitHub Link : https://github.com/AVKArun/Cliff.ai-Assignment.git

Tomcat Server Page : http://3.144.32.41:8090/

Jenkins DashBoard : http://3.144.239.95:8080/

For more information : Mail @ 18bcs012@iiitdwd.ac.in.
A Project by Arun AVK,B.Tech CSE, IIIT,Dharwad.