Auxilo CI/CD setup

1. Install jenkins on linux:-

<https://www.virtualizationhowto.com/2021/09/install-jenkins-on-ubuntu-21-04-with-self-signed-ssl-certificate/>

1. Install prometheus on linux:-

<https://www.vultr.com/docs/install-prometheus-server-on-ubuntu-20-04>

1. Install Sonarqube on linux:-

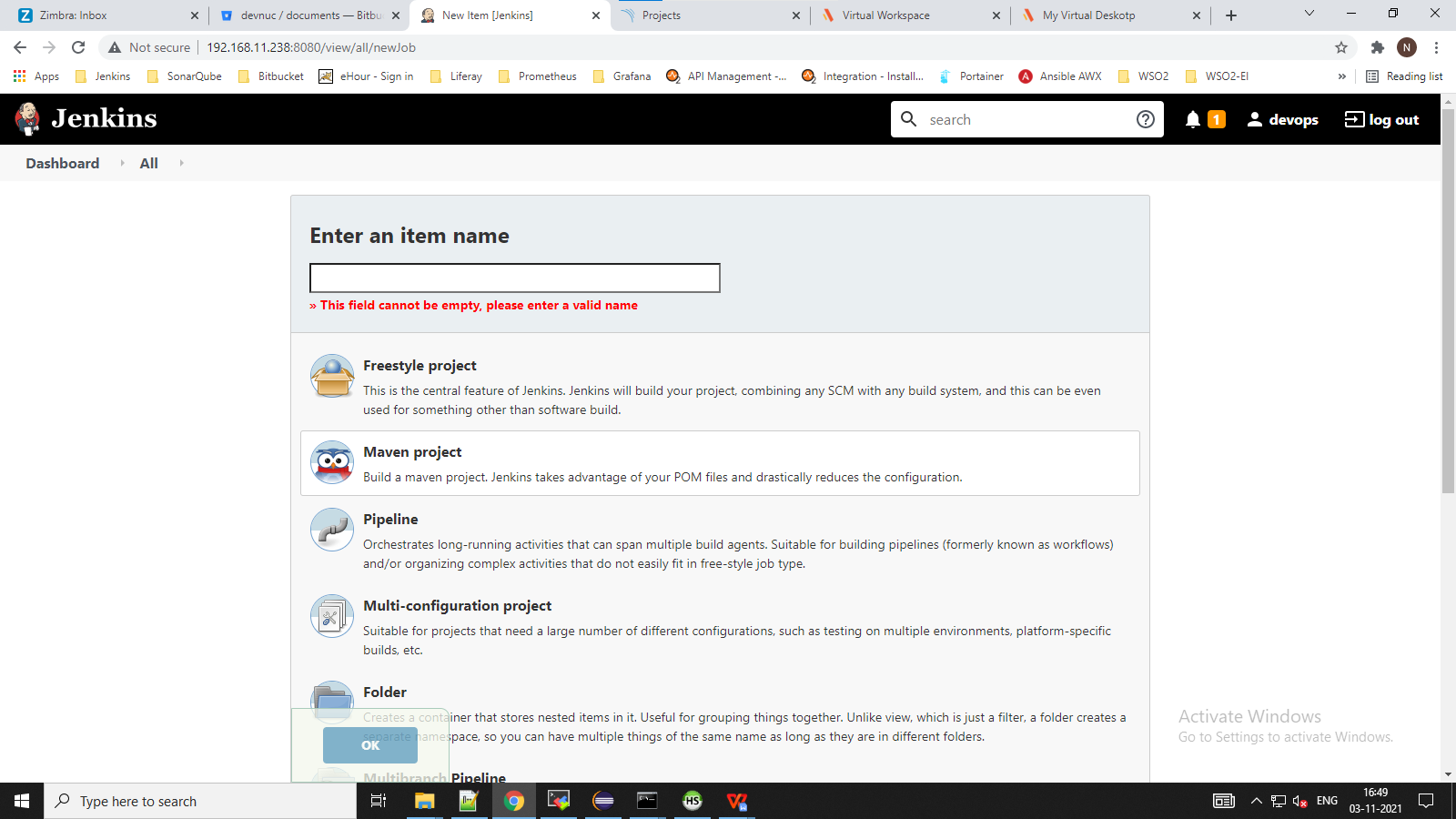
[Install SonarQube on Ubuntu 20.04 LTS - Vultr.com](https://www.vultr.com/docs/install-sonarqube-on-ubuntu-20-04-lts)

1. Install Sonar-scanner on linux:-

[Tutorial - Sonarqube Scanner Installation on Ubuntu Linux [ Step by Step ]](https://techexpert.tips/sonarqube/sonarqube-scanner-installation-ubuntu-linux/)

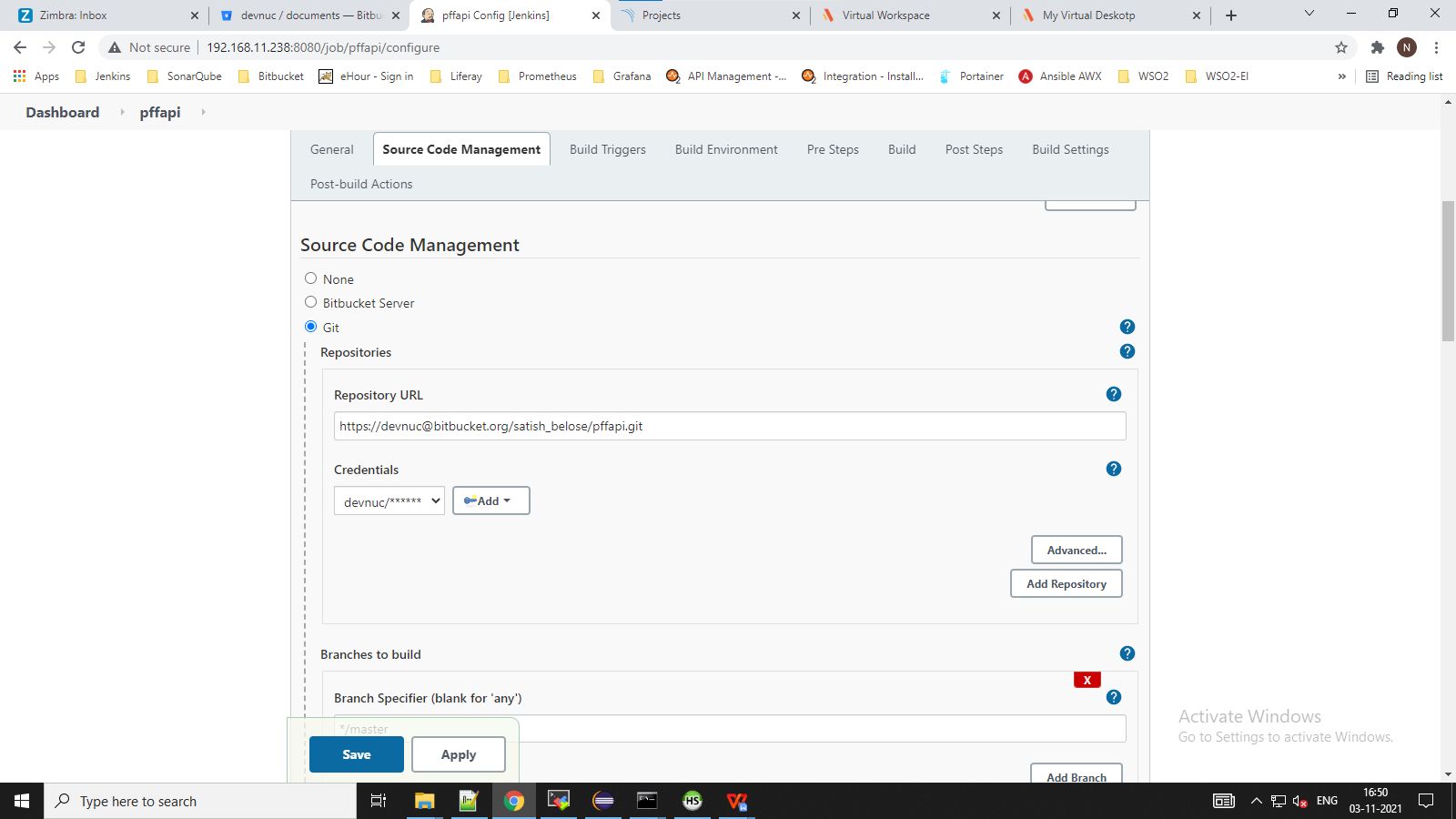
Step1:- Create Maven Project with specific API Name

Example:-**pffapi**



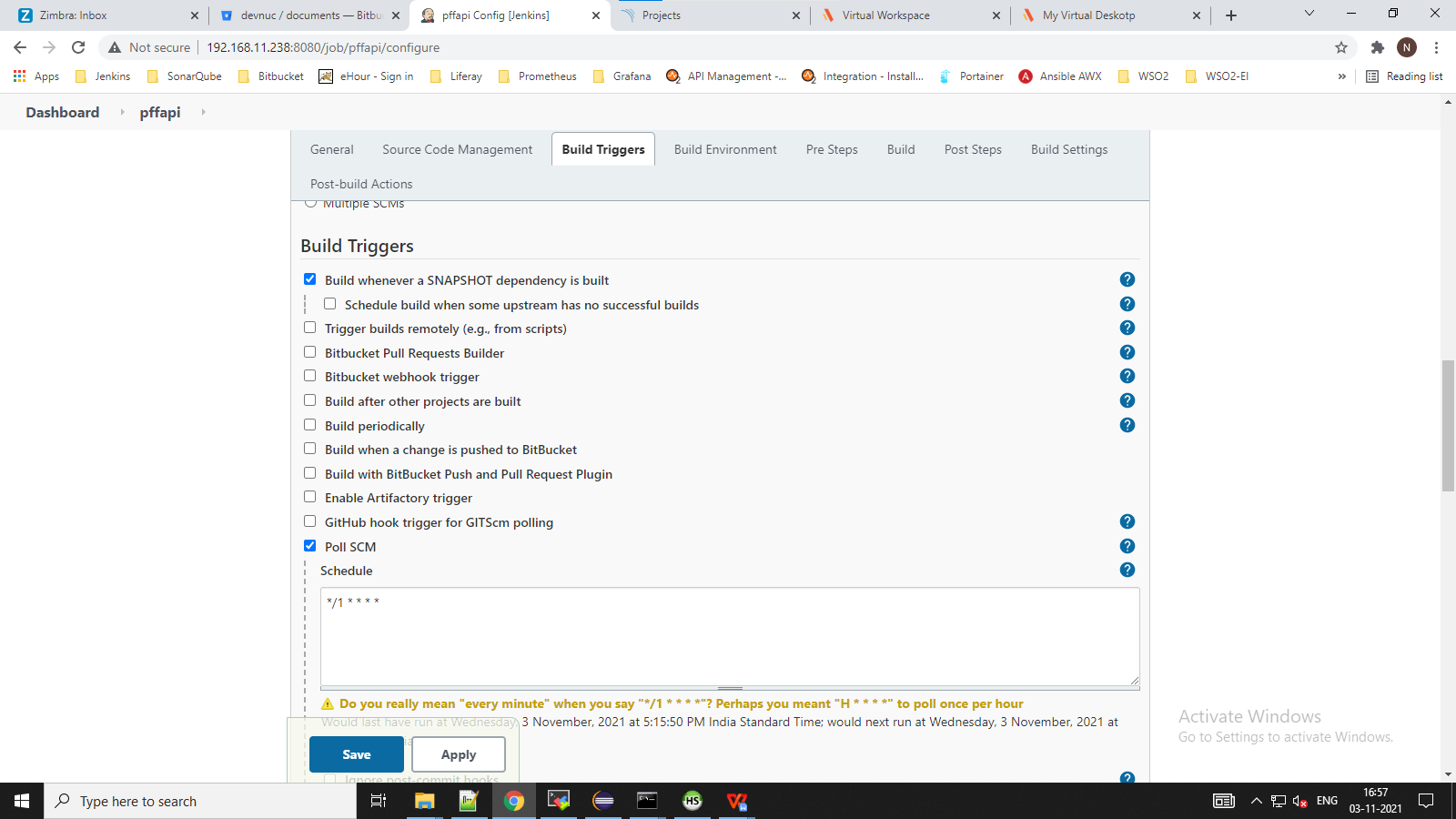
Step 2:- for bitbucket repository we select git as Source Code Management

**Provide repository URL and Credentials**



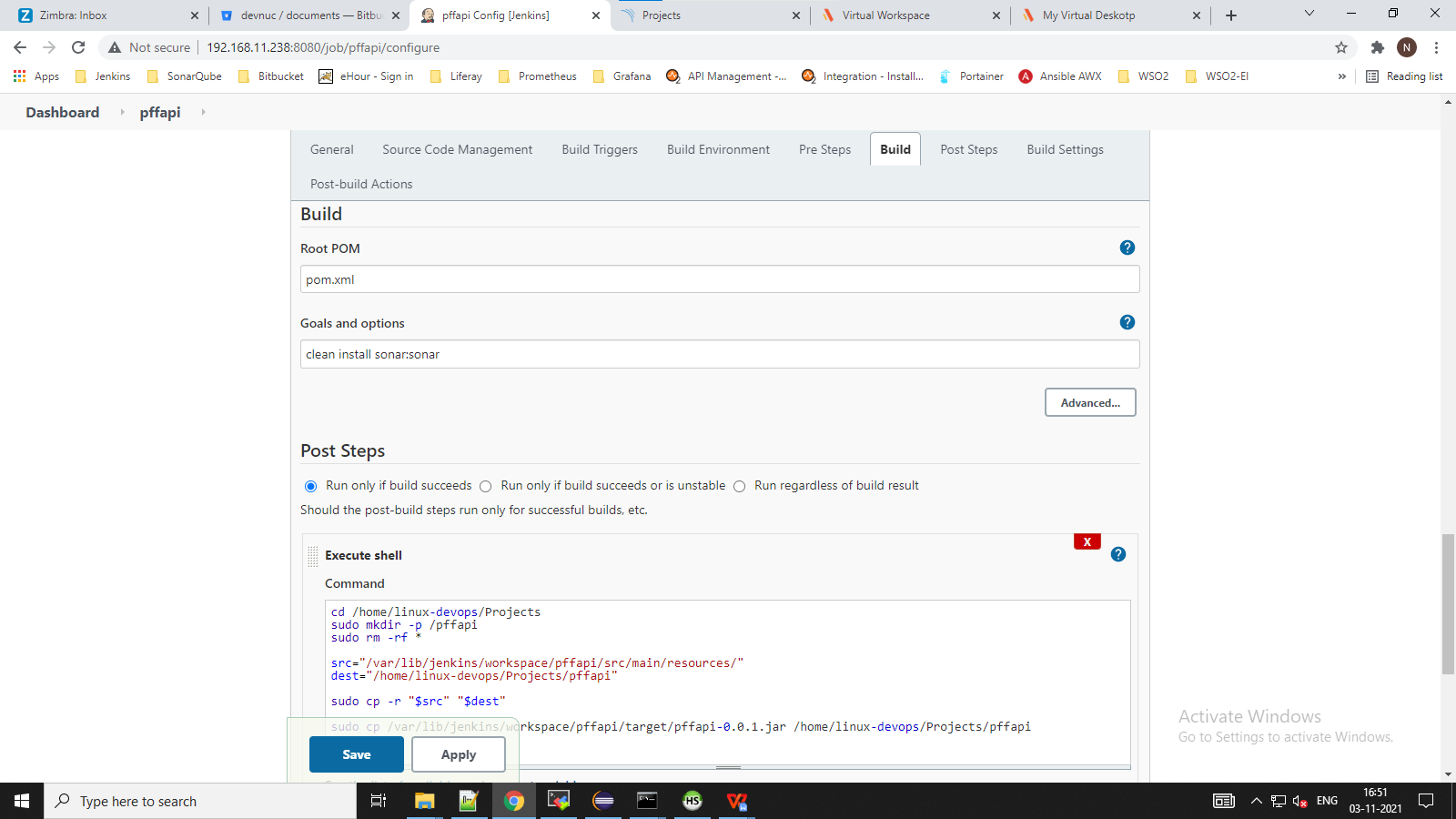
Step 3:- select Build Triggers as **Build whenever a SNAPSHOT dependency is built**

And Poll SCM schedule as \*/1 \* \* \* \* (whenever anything push to master repository it will trigger the build)



Step 4:- In build section

Root POM as **pom.xml** and Goals and options as **clean install sonar:sonar** (Build jar file and execute sonar scan analysis)



Step 5:- In Post Steps select **Run only if build succeeds** and select execute shell script as below:-

cd /home/linux-devops/Projects //change directory to desired location

sudo mkdir -p /pffapi //(create folder as API name for example:- pffapi )

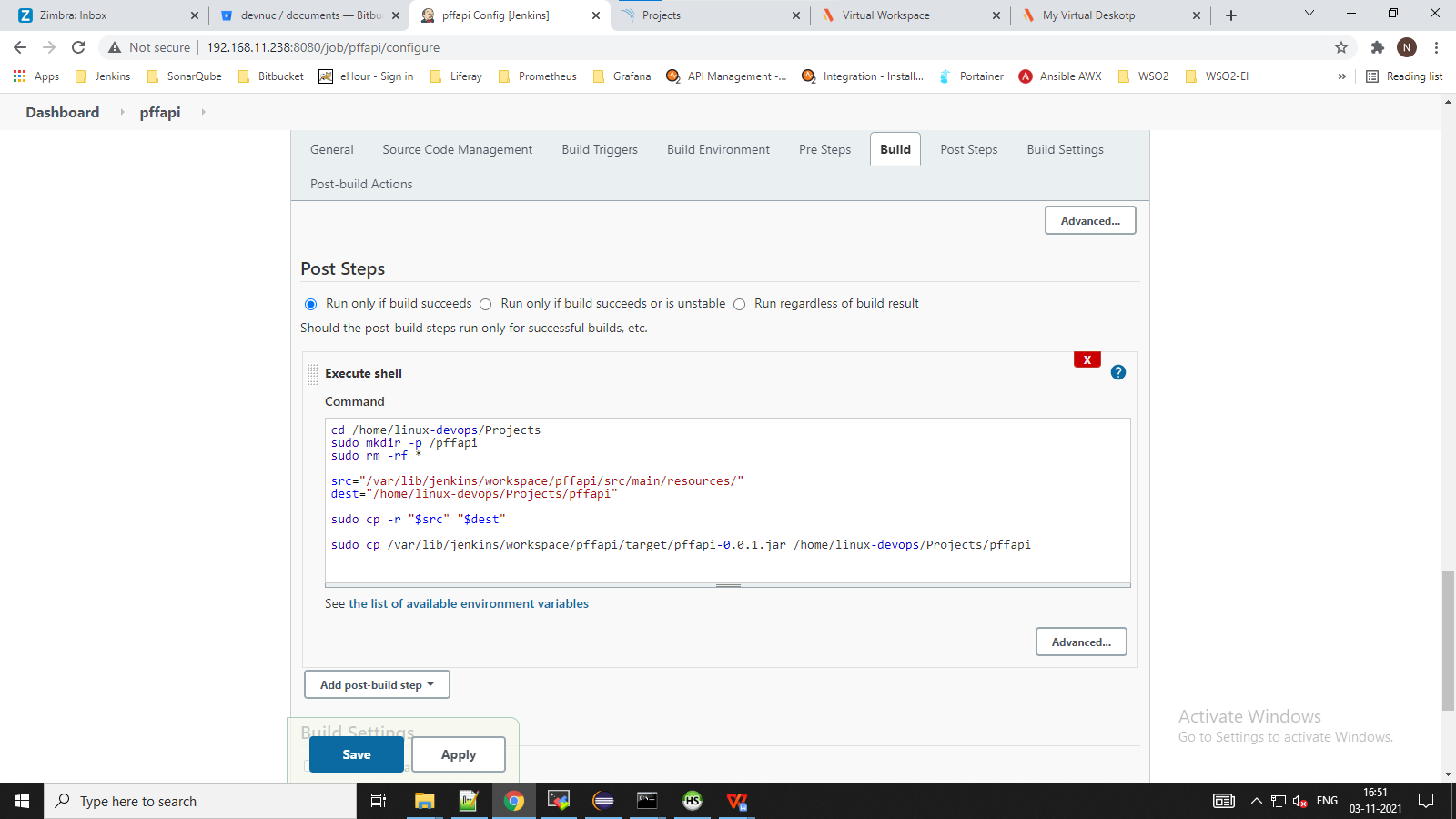
sudo rm -rf \* //(clean everything from pffapi folder before copying from jenkins workspace)

src="/var/lib/jenkins/workspace/pffapi/src/main/resources/" //(create source path variable as src)

dest="/home/linux-devops/Projects/pffapi" //(create destination path variable as dest)

sudo cp -r "$src" "$dest" //(copy folders/file from source to destination)

sudo cp /var/lib/jenkins/workspace/pffapi/target/pffapi-0.0.1.jar /home/linux-devops/Projects/pffapi (Copy jar file from source to destination)



Declarative pipeline scripts:-

pipeline {

agent any

environment {

JAVA = "/usr/lib/jvm/java-11-openjdk-amd64"

MAVEN = "/usr/share/maven"

}

stages {

stage ('Clone Repo') {

steps {

git credentialsId: 'dfde11ad-70f1-44d3-a6bd-d5884100e400', url: 'https://devnuc@bitbucket.org/satish\_belose/pffapi.git'

}

}

stage ('Build') {

steps {

script {

try {

sh "mvn clean install"

currentBuild.result = 'SUCCESS'

emailext (

attachLog: true, compressLog: true,

subject: "BUILD SUCCESSFUL: Job '${env.JOB\_NAME} [${env.BUILD\_NUMBER}]'",

body: """<p>BUILD SUCCESSFUL: Job '${env.JOB\_NAME} : [${env.BUILD\_NUMBER}]'</p>

<p>Check console output at :-<a href='${env.BUILD\_URL}consoleText for more details'>${env.JOB\_NAME} [${env.BUILD\_NUMBER}]</a></p>""",

to: 'mahadev\_rane@nucsoft.in'

)

}

catch (err) {

currentBuild.result = 'FAILURE'

emailext (

attachLog: true, compressLog: true,

subject: "BUILD FAILED: Job '${env.JOB\_NAME} [${env.BUILD\_NUMBER}]'",

body: """<p>FAILED: Job '${env.JOB\_NAME} [${env.BUILD\_NUMBER}]':</p>

<p>project build error is here: <a href='${env.BUILD\_URL}consoleText for more details'>${env.JOB\_NAME} [${env.BUILD\_NUMBER}]</a></p>""",

recipientProviders: [[$class: 'CulpritsRecipientProvider'],[$class: 'UpstreamDevelopersRecipientProvider']],

to: 'mahadev\_rane@nucsoft.in'

)

throw err

}}

}

}

stage ('Code Quality Check via SonarQube') {

steps {

sh "mvn sonar:sonar"

}

}

stage ('deployment') {

steps {

sh ''' cd /home/linux-devops/Projects

sudo mkdir -p /pffapi

sudo rm -rf \*

src="/var/lib/jenkins/workspace/pffapi/src/main/resources/"

dest="/home/linux-devops/Projects/pffapi"

sudo cp -r "$src" "$dest"

sudo cp /var/lib/jenkins/workspace/pffapi/target/pffapi-0.0.1.jar /home/linux-devops/Projects/pffapi'''

script {

def data = "java -jar pffapi-0.0.1.jar"

writeFile(file: 'start.sh', text: data)

sh "sudo cp /var/lib/jenkins/workspace/Auxilo-pffapi/start.sh /home/linux-devops/Projects/pffapi"

}

emailext (

attachLog: true, compressLog: true,

subject: "DEPLOYMENT SUCCESSFUL: Job '${env.JOB\_NAME} [${env.BUILD\_NUMBER}]'",

body: """<p>DEPLOYMENT SUCCESSFUL: Job '${env.JOB\_NAME} [${env.BUILD\_NUMBER}]':</p>

<p>Check console output at :- <a href='${env.BUILD\_URL}consoleText for more details'>${env.JOB\_NAME} [${env.BUILD\_NUMBER}]</a></p>""",

to: 'mahadev\_rane@nucsoft.in'

)

}

}

}

}