

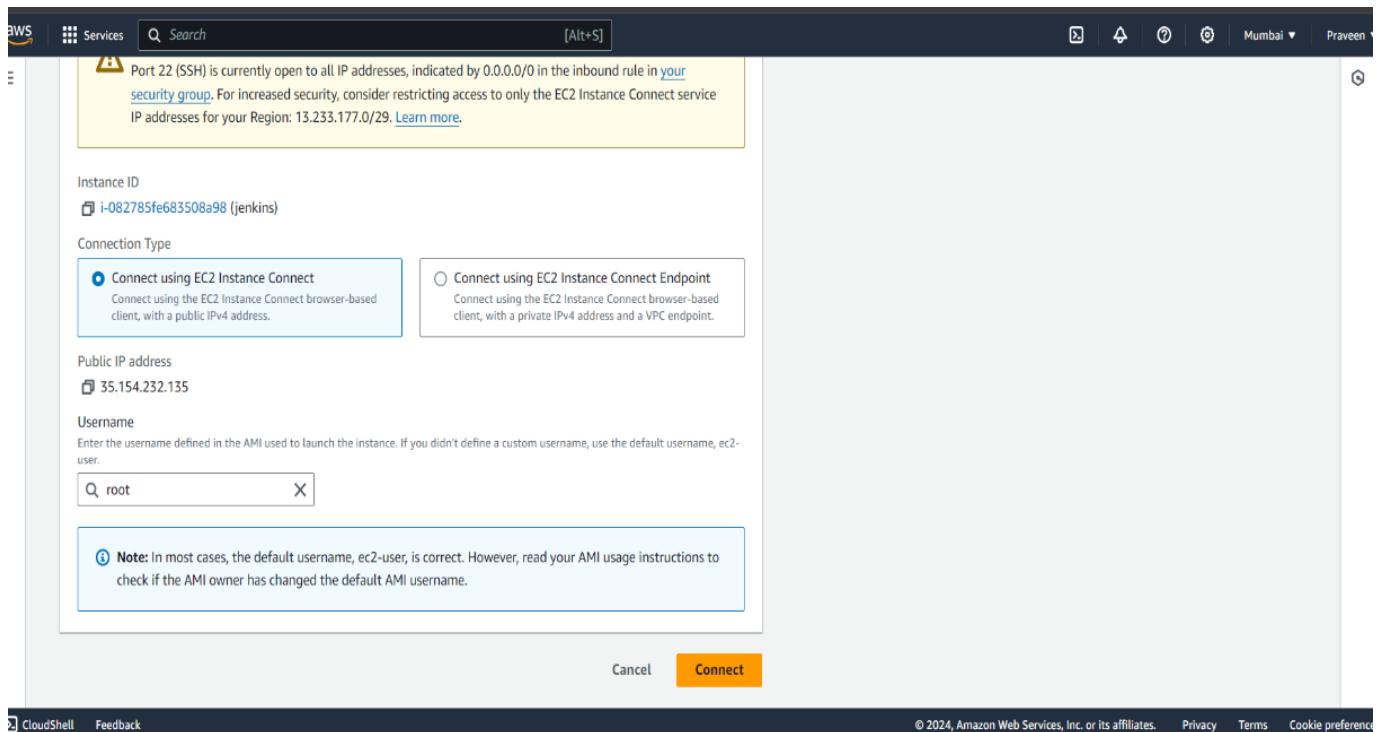
Done by: PRAVEEN GOGULA

A PROJECT REPORT ON

DEPLOYING OF DOCKER IMAGE ON EC2 INSTANCE

To Launch EC2 Instance :

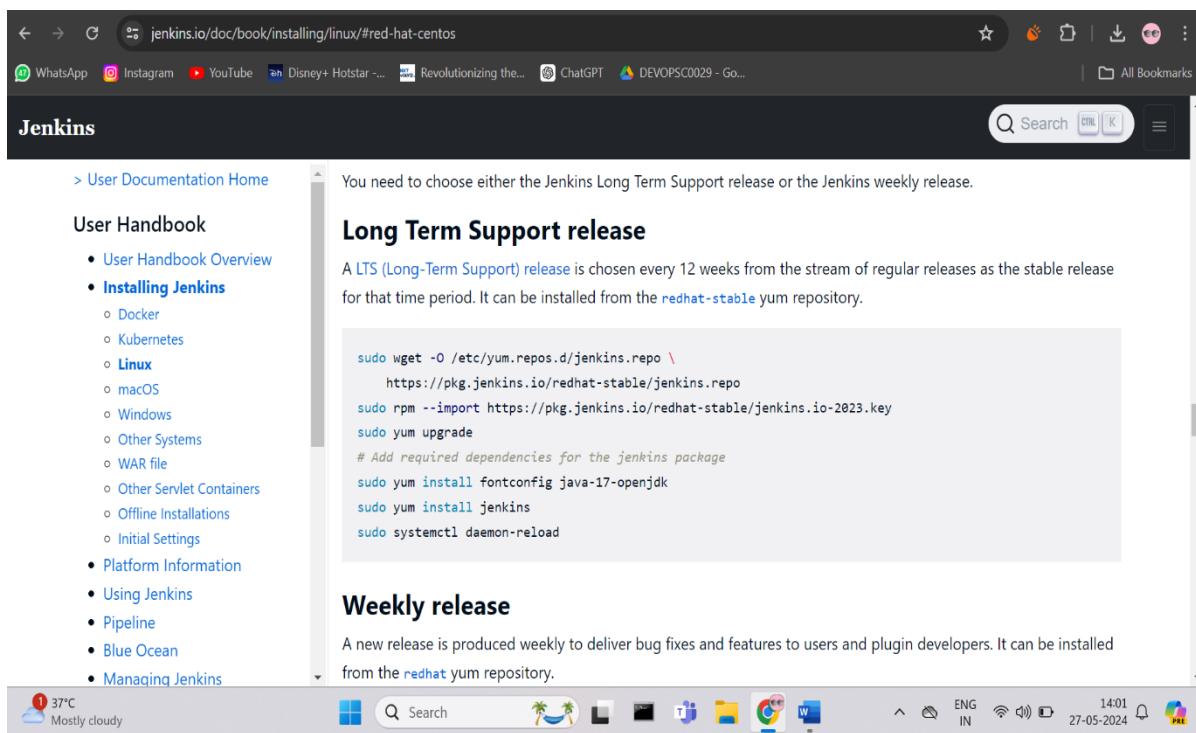
- Open the Amazon EC2 console at <https://ap-south-1.console.aws.amazon.com/ec2/>
- Open EC2 Dashboard and select launch instance and launch a new instance .
- Name the instance under the name and tags (ex : Jenkins)
- Under the Application and OS Images (Amazon Machine Image) select Amazon Linux at Quick Start this is the operation system for our instance. From Amazon Machine Image select the Amazon Linux 2 AMI.
- Under the instance type, list the instance type and select the t2.micro instance type. This instance type is hardware configuration to our instance
- Then go for key pair(login) create a new key pair by selecting creating new key pair
 - Name the key pair under Key Pair Name.
 - Under Key Pair type select RSA.
 - Under private Key file format select .pem.
 - Now go with the create Key Pair
 - Now select the created key
- Keep the default selections for the other configurations settings for our instance.
- Then go for the launch instance, New instance will be launched.
- Now go to the instance page. Here new instance will be launched in pending, wait for few minutes to come into running state the connect to the instance.



Jenkins installation and configure:

➤ Steps to install Jenkins:

- Search for Jenkins installation (google search)
- To install Jenkins in linux several distributions are available they are
 - Debain/Ubuntu
 - Fedora
 - Red Hat/Alma/Rocky
- From this distributions go with Red Hat/Alma/Rocky
<https://www.jenkins.io/doc/book/installing/linux/#red-hat-centos>
- Long Term support Releases :
 - 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-2023.key
 - sudo yum upgrade
 - # Add required dependencies for the jenkins package
 - sudo yum install fontconfig java-17-openjdk
 - sudo yum install jenkins
 - sudo systemctl daemon-reload



- Jenkins installation :

```

root@ip-172-31-44-94 ~]# sudo wget -O /etc/yum.repos.d/jenkins.repo \
  https://pkg.jenkins.io/redhat-stable/jenkins.repo
[sudo] password for root:
root@ip-172-31-44-94 ~]# sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key
root@ip-172-31-44-94 ~]# yum upgrade
[sudo] password for root:
root@ip-172-31-44-94 ~]# 

```

i-0556101bf52d307ca (jenkins)
PublicIPs: 65.0.103.71 PrivateIPs: 172.31.44.94

```

root@ip-172-31-44-94 ~]# sudo yum install fontconfig java-17-openjdk
[sudo] password for root:
root@ip-172-31-44-94 ~]# sudo yum install jenkins
[sudo] password for root:
root@ip-172-31-44-94 ~]# sudo systemctl daemon-reload
[sudo] password for root:
root@ip-172-31-44-94 ~]# Last metadata expiration check: 0:02:04 ago on Mon May 27 08:19:48 2024.
root@ip-172-31-44-94 ~]# No match for argument: java-17-openjdk
root@ip-172-31-44-94 ~]# Error: Unable to find a match: java-17-openjdk
root@ip-172-31-44-94 ~]# Last metadata expiration check: 0:02:05 ago on Mon May 27 08:19:48 2024.
root@ip-172-31-44-94 ~]# Dependencies resolved.
root@ip-172-31-44-94 ~]# 

```

Package	Architecture	Version	Repository	Size
jenkins	noarch	2.452.1-1.1	jenkins	89 M

```

Transaction Summary
Install 1 Package

Total download size: 89 M
Installed size: 89 M
Is this ok [y/N]: y
Downloading Packages:
jenkins-2.452.1-1.1.noarch.rpm
Total
Running transaction check

```

i-0556101bf52d307ca (jenkins)
PublicIPs: 65.0.103.71 PrivateIPs: 172.31.44.94

```
Verifying : pixman-0.40.0-3.amzn2023.0.3.x86_64
Verifying : xml-common-0.6.3-56.amzn2023.0.2.noarch

installed:
alsa-lib-1.2.7.2-1.amzn2023.0.2.x86_64
dejavu-sans-fonts-2.37-16.amzn2023.0.2.noarch
dejavu-serif-fonts-2.37-16.amzn2023.0.2.noarch
fonts-filesystem-1:2.0.5-12.amzn2023.0.2.noarch
glibflib-5.2.1-9.amzn2023.0.1.x86_64
google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch
harfbuzz-7.0-0.2.amzn2023.0.1.x86_64
java-17-amazon-corretto-headless-1:17.0.11+9.1.amzn2023.1.x86_64
java-22-amazon-corretto-1:22.0.1+8.1.amzn2023.1.x86_64
javapackages-filesystem-6.0-0.7.amzn2023.0.6.noarch
libICE-1.0.10-6.amzn2023.0.2.x86_64
libICE-1.0.10-6.amzn2023.0.2.x86_64
libXau-1.0.9-6.amzn2023.0.2.x86_64
libXi-1.7.10-6.amzn2023.0.2.x86_64
libXrandr-1.5.2-6.amzn2023.0.2.x86_64
libXt-1.2.0-4.amzn2023.0.2.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
libpng-2:1.6.37-10.amzn2023.0.6.x86_64
pixman-0.40.0-3.amzn2023.0.3.x86_64

cairo-1.17.6-2.amzn2023.0.1.x86_64
dejavu-sans-mono-fonts-2.37-16.amzn2023.0.2.noarch
fontconfig-2.13.94-2.amzn2023.0.2.x86_64
freetype-2.13.0-2.amzn2023.0.1.x86_64
google-noto-fonts-common-20201206-2.amzn2023.0.2.noarch
graphite2-1.3.14-7.amzn2023.0.2.x86_64
java-17-amazon-corretto-devel-1:17.0.11+9.1.amzn2023.1.x86_64
java-17-amazon-corretto-jmodsd-1:17.0.11+9.1.amzn2023.1.x86_64
java-22-amazon-corretto-headless-1:22.0.1+8.1.amzn2023.1.x86_64
langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch
libsM-1.2.3-8.amzn2023.0.2.x86_64
libx11-common-1.7.2-3.amzn2023.0.4.noarch
libxext-1.3.4-6.amzn2023.0.2.x86_64
libxinerama-1.1.4-8.amzn2023.0.2.x86_64
libxrender-0.9.10-14.amzn2023.0.2.x86_64
libxtst-1.2.3-14.amzn2023.0.2.x86_64
libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64
libxcb-1.13.1-7.amzn2023.0.2.x86_64
xml-common-0.6.3-56.amzn2023.0.2.noarch

Complete!
root@ip-172-31-44-94 ~]$
```

➤ Now Start Jenkins Using this commands :

- You can enable the Jenkins service to start at boot with the command
 - **`sudo systemctl enable Jenkins`**
 - You can start the Jenkins service with the command:
 - **`sudo systemctl start jenkins`**
 - You can check the status of the Jenkins service using the command:
 - **`sudo systemctl status Jenkins`**

```
[root@ip-172-31-46-220 ~]# sudo systemctl enable jenkins
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /usr/lib/systemd/system/jenkins.service.
[root@ip-172-31-46-220 ~]# sudo systemctl start jenkins
[root@ip-172-31-46-220 ~]# sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: disabled)
     Active: active (running) since Mon 2024-05-27 08:59:41 UTC; 8s ago
       Main PID: 26168 (java)
          Tasks: 46 (limit: 1114)
         Memory: 331.1M
            CPU: 39.742s
           CGroup: /system.slice/jenkins.service
                   └─26168 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

May 27 08:59:09 ip-172-31-46-220.ap-south-1.compute.internal jenkins[26168]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
May 27 08:59:09 ip-172-31-46-220.ap-south-1.compute.internal jenkins[26168]: ****
May 27 08:59:41 ip-172-31-46-220.ap-south-1.compute.internal jenkins[26168]: 2024-05-27 08:59:41.661+0000 [id=32]      INFO      jenkins.InitReactorRunner$1@onAttain>>
May 27 08:59:41 ip-172-31-46-220.ap-south-1.compute.internal jenkins[26168]: 2024-05-27 08:59:41.683+0000 [id=24]      INFO      hudson.lifecycle.Lifecycle$onReady: J>
May 27 08:59:41 ip-172-31-46-220.ap-south-1.compute.internal jenkins[26168]: 2024-05-27 08:59:42.544+0000 [id=47]      INFO      h.m.DownloadService$Downloadable$load>
May 27 08:59:42 ip-172-31-46-220.ap-south-1.compute.internal jenkins[26168]: 2024-05-27 08:59:42.545+0000 [id=47]      INFO      hudson.util.RetriggerStart: Performed >
May 27 08:59:42 ip-172-31-46-220.ap-south-1.compute.internal jenkins[26168]: 2024-05-27 08:59:46.798+0000 [id=61]      WARNING     h.n.DiskSpaceMonitorDescriptor#mar>
lines 1-20 [END]
[root@ip-172-31-46-220 ~]#
```

➤ Edit inbound rules in security groups :

- Add the port numbers in inbound rules
- Port range: 8080
- Source: Anywhere-ipv4
- Save rules

The screenshot shows the AWS EC2 console with the URL ap-south-1.console.aws.amazon.com/ec2/home?region=ap-south-1#ModifyInboundSecurityGroupRules:securityGroupId=sg-0e0b113.... The page title is "Edit inbound rules". It displays two security group rules:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-06dd2aecfee6dff9e	SSH	TCP	22	Custom	0.0.0.0/0
sgr-0190db17fd2fd3306	Custom TCP	TCP	8080	Custom	0.0.0.0/0

A warning message at the bottom states: "⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only." The browser status bar shows "CloudShell Feedback" and the date "27-05-2024".

➤ Steps to Launch the Jenkins dashboard :

- Copy public Ip address from the instance
- Browse to <http://localhost:8080> (publicIp:portnumber)

The screenshot shows a browser window with the URL [Not secure | 13.127.36.67:8080/login?from=%2F](http://13.127.36.67:8080/login?from=%2F). The page title is "Getting Started". It displays the "Unlock Jenkins" step:

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

```
/var/lib/jenkins/secrets/initialAdminPassword
```

Please copy the password from either location and paste it below.

Administrator password

Continue

The browser status bar shows "CloudShell Feedback" and the date "27-05-2024".

- To unlock Jenkins copy the from `/var/lib/jenkins/secrets/initialAdminPassword`
- Cat `/var/lib/jenkins/secrets/initialAdminPassword`
- Cat is a command is used to open the given path file to get Jenkins administrator password

```
lines 1-20/20 (END)
[root@ip-172-31-46-220 ~]# cat /var/lib/jenkins/secrets/initialAdminPassword
d0a91c238ecb4ce9ee498c23f196116
[root@ip-172-31-46-220 ~]# i-0fd550689ff74df8 (jenkins)
PublicIPs: 13.127.36.67 PrivateIPs: 172.31.46.220
```

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

.....

Continue

https://www.jenkins.io/redirect/find-jenkins-logs

Nifty smcap +1.13% 15:04 27-05-2024 ENG IN

- The go with CONTINUE

Instagram https://www.instagram.com Getting Started

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

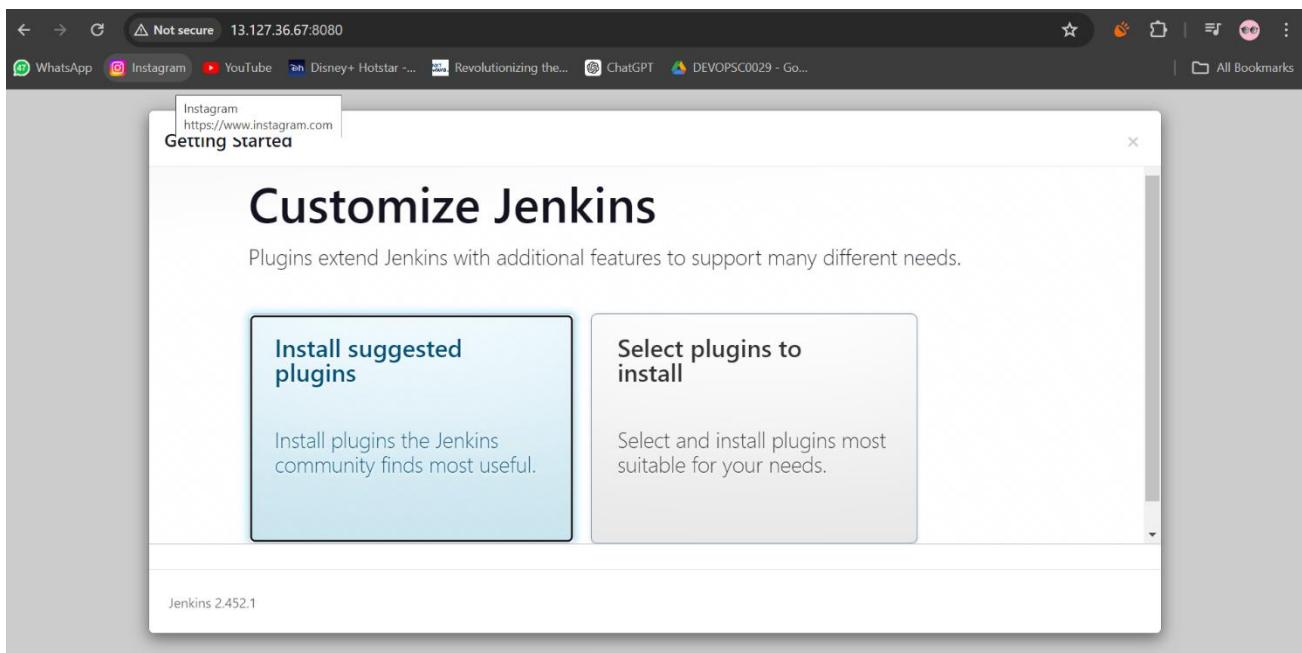
Install plugins the Jenkins community finds most useful.

Select plugins to install

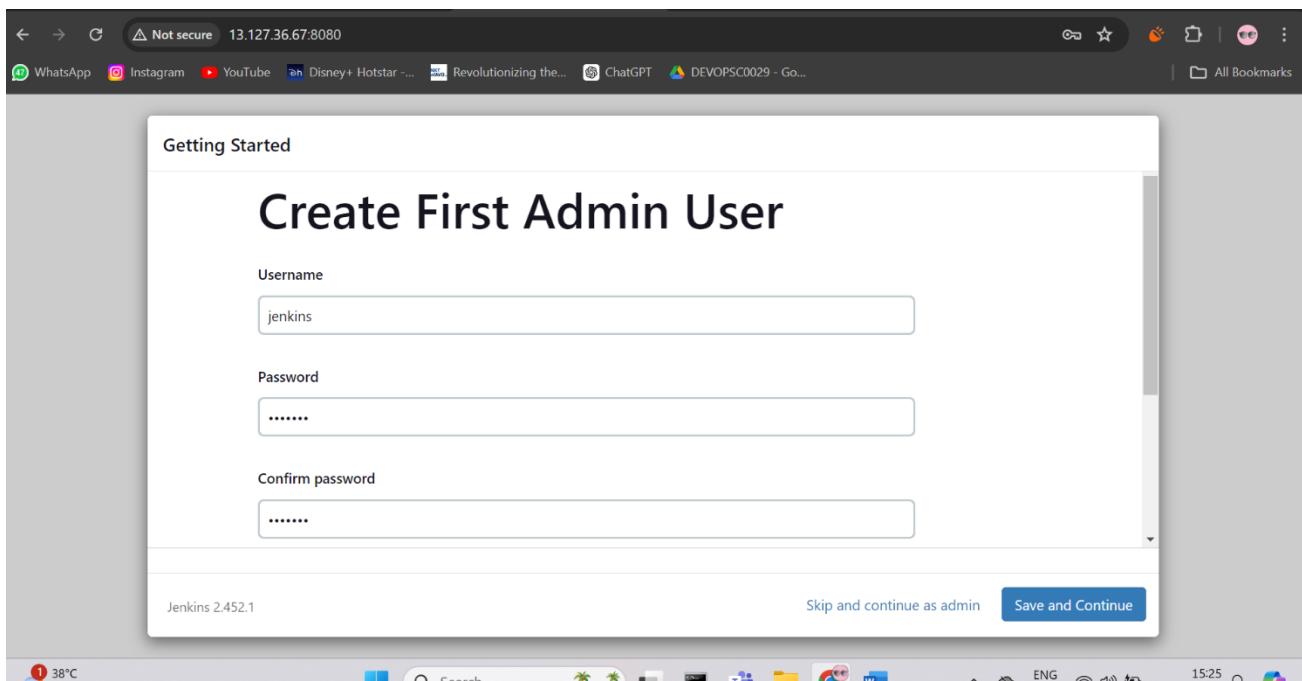
Select and install plugins most suitable for your needs.

Jenkins 2.452.1

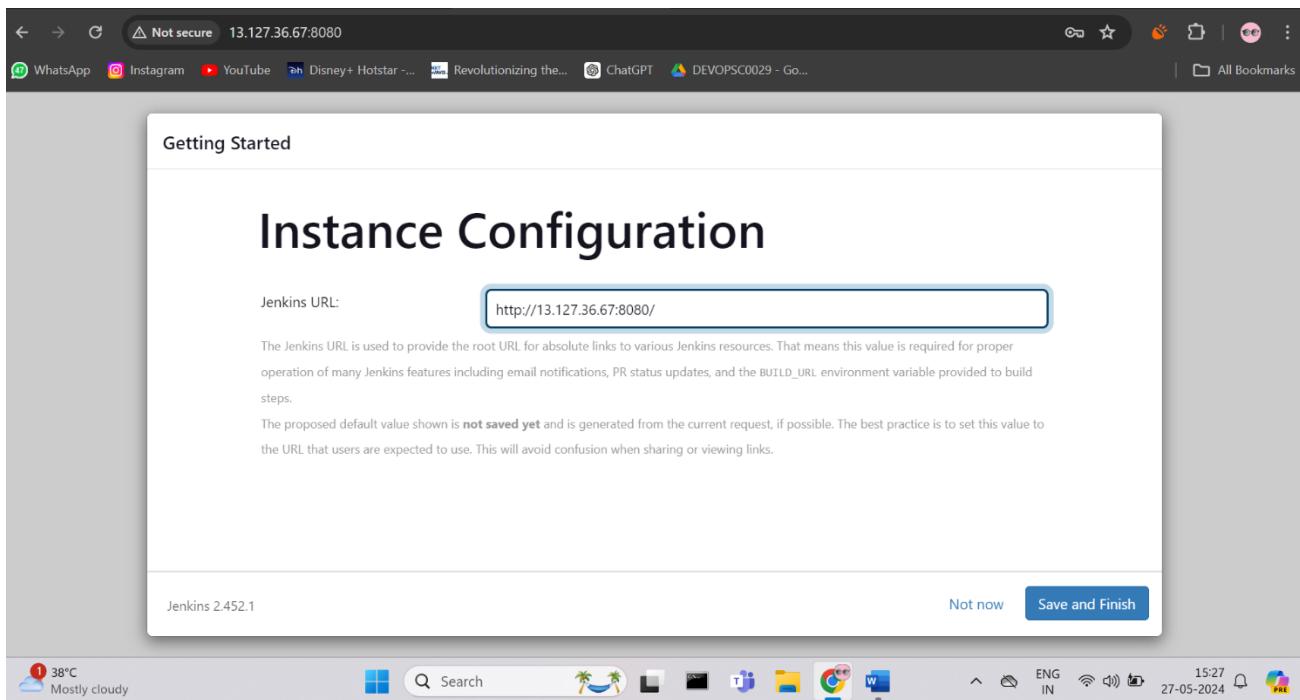
- Select the install suggested plugins in customize Jenkins page



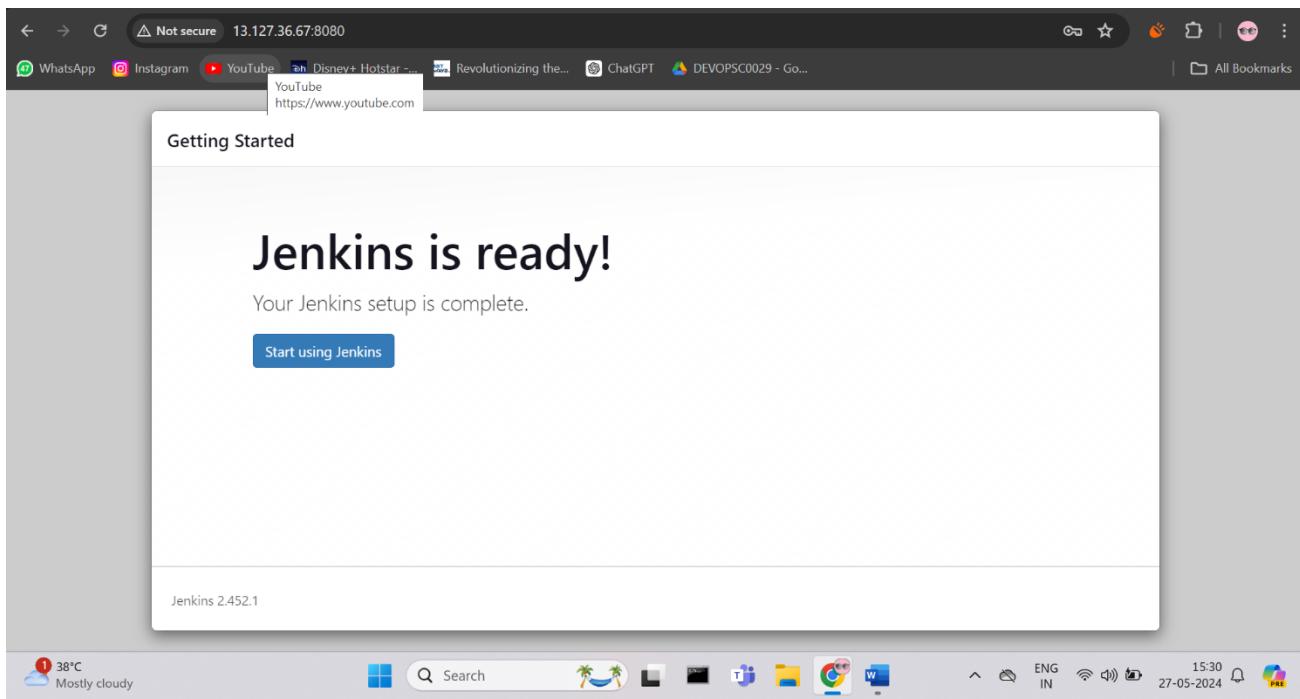
- After installation of suggested plugins go with creating first admin user's username and password



- Save and continue



- Save and finish then the Jenkins setup will be completed.
- Jenkins is ready to use



- Start using Jenkins

➤ Jenkins Dashboard :

The screenshot shows the Jenkins dashboard at 13.127.36.67:8080. The main content area features a "Welcome to Jenkins!" message and sections for "Start building your software project", "Set up a distributed build", and "Configure a cloud". On the left, there are links for "New Item", "Build History", "Manage Jenkins", and "My Views". Below these are "Build Queue" and "Build Executor Status" sections. The top navigation bar includes a search bar, user profile, and log out link.

➤ Job creation :

- Start building your software project, create a job
- Enter the project name (ex:project-1)
- Select Free Style Project and go with hitting OK
- Job is Created

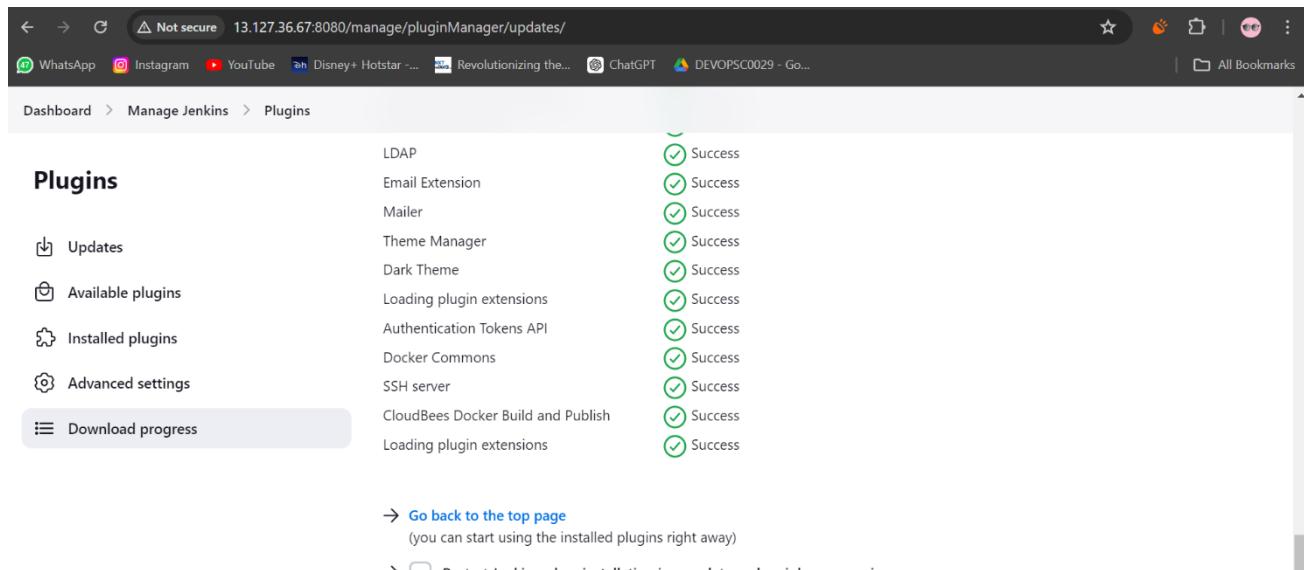
The screenshot shows the "Enter an item name" dialog in Jenkins. A single input field contains the text "project-1". Below it, three project types are listed: "Freestyle project", "Pipeline", and "Multi-configuration project". Each has a brief description and a "OK" button. The "Freestyle project" is currently selected. The bottom of the screen shows the Windows taskbar with various icons and system status.

Job configuration :

- **General settings** : This section contains general settings of the job.
- **Source code Management** : This section contains source code options like git.
- **Build Triggers** : The section contents trigger settings that trigger the build based on the specific condition match.
- **Build steps** : The section contains the build steps that can be performed by adding a Batch or shell command.
- **Post-build actions** : The section contains the build steps that can be performed after the build action.
- Save the configuration and click BUILD NOW.
- Check the CONSOLE OUTPUT.

Installation of required plugins:

- Go to manage Jenkins from Jenkins dashboard.
- Hit on Plugins and go for Available Plugins
- Install the required Plugins like “Docker, cloudbess docker build and publish”
- The plugin Cloubess Docker Build and Publish is used to building and deploying of docker image to the docker hub.



The screenshot shows the Jenkins plugin manager interface. The left sidebar has links for Dashboard, Manage Jenkins, Plugins, Updates, Available plugins, Installed plugins, Advanced settings, and Download progress. The main area lists installed plugins with green checkmarks and the word 'Success' next to each. A message at the bottom says "you can start using the installed plugins right away".

Plugin	Status
LDAP	Success
Email Extension	Success
Mailer	Success
Theme Manager	Success
Dark Theme	Success
Loading plugin extensions	Success
Authentication Tokens API	Success
Docker Commons	Success
SSH server	Success
CloudBees Docker Build and Publish	Success
Loading plugin extensions	Success

→ [Go back to the top page](#)
(you can start using the installed plugins right away)
→ Restart Jenkins when installation is complete and no jobs are running



Configure the Tools (Maven and Docker):

- In Manage Jenkins go for Tools.
- In tools configure add maven installation.
- Give the name as MAVEN.
- Select install automatically and give the latest version of maven.

The screenshot shows the Jenkins 'Manage Jenkins > Tools > Maven installations' page. A new Maven installation is being configured with the name 'maven'. The 'Install automatically' checkbox is checked, and the 'Version' dropdown is set to '3.9.7'. The 'Save' button is highlighted in blue, while 'Apply' is in grey.

- Now go for docker installation.
- Add docker installation.
- Give the name as docker .
- Select install automatically and give latest as the version column.
- Add installer (Download from docker.com)
- Then apply and save.

The screenshot shows the Jenkins 'Manage Jenkins > Tools > Docker installations' page. A new Docker installation is being configured with the name 'docker'. The 'Install automatically' checkbox is checked, and the 'Docker version' dropdown is set to 'latest'. The 'Save' button is highlighted in blue, while 'Apply' is in grey.

➤ Git, maven, docker installation in server :

- Git installation :

The screenshot shows a browser-based terminal session on an Amazon Linux 2 instance. The command `yum install git -y` is run, which installs the Git package and its dependencies. The output shows the packages being installed, their versions, repositories, and sizes.

Package	Architecture	Version	Repository	Size
git	x86_64	2.40.1-1.amzn2023.0.2	amazonlinux	54 k
git-core	x86_64	2.40.1-1.amzn2023.0.2	amazonlinux	4.3 M
git-core-doc	noarch	1:0.17029-5.amzn2023.0.2	amazonlinux	2.6 M
perl-Error	noarch	1.37-477.amzn2023.0.6	amazonlinux	41 k
perl-File-Find	noarch	2.40.1-1.amzn2023.0.2	amazonlinux	26 k
perl-Git	noarch	2.38-9.amzn2023.0.2	amazonlinux	42 k
perl-TermReadKey	x86_64	0.65-477.amzn2023.0.6	amazonlinux	36 k
perl-lib	x86_64		amazonlinux	15 k

Transaction Summary
Install 8 Packages
Total download size: 7.1 M
Installed size: 34 M
Downloading Packages:
/0: git-2.40.1-1.amzn2023.0.2.x86_64.rpm
/0: git-core-doc-2.40.1-1.amzn2023.0.2.noarch.rpm
851 kB/s | 54 kB 00:00
27 MB/s | 2.6 MB 00:00

i-Ofdd550689ff74df8 (jenkins2)
PublicIPs: 13.235.132.151 PrivateIPs: 172.31.46.220

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 32°C Sunny ENG IN 12:10 28-05-2024

- Maven installation :

The screenshot shows a browser-based terminal session on an Amazon Linux 2 instance. The command `yum install maven -y` is run, which installs the Maven package and its dependencies. The output shows the packages being installed, their versions, repositories, and sizes.

Package	Architecture	Version	Repository	Size
maven	noarch	1:3.8.4-3.amzn2023.0.5	amazonlinux	18 k
apache-commons-cli	noarch	1.5.0-3.amzn2023.0.3	amazonlinux	76 k
apache-commons-codec	noarch	1.15-6.amzn2023.0.3	amazonlinux	303 k
apache-commons-io	noarch	1:2.8.0-7.amzn2023.0.4	amazonlinux	284 k
apache-commons-lang3	noarch	3.12.0-7.amzn2023.0.3	amazonlinux	559 k
atinject	noarch	1.0.5-3.amzn2023.0.3	amazonlinux	23 k
cdi-api	noarch	2.0.2-6.amzn2023.0.3	amazonlinux	54 k
google-guice	noarch	4.2.3-8.amzn2023.0.6	amazonlinux	473 k
guava	noarch	31.1.1-3.amzn2023.0.6	amazonlinux	2.4 M
httpcomponents-client	noarch	4.5.13-3.amzn2023.0.4	amazonlinux	657 k
httpcomponents-core	noarch	4.4.13-6.amzn2023.0.3	amazonlinux	632 k
jakarta-annotations	noarch	1.3.5-13.amzn2023.0.3	amazonlinux	46 k
jansi	x86_64	2.4.0-3.amzn2023.0.3	amazonlinux	113 k
java-17-amazon-corretto-devel	x86_64	1:17.0.11+9-1.amzn2023.1	amazonlinux	142 k
jcl-over-slf4j	noarch	1.7.32-3.amzn2023.0.4	amazonlinux	25 k
jsoup	noarch	1.13.1-9.amzn2023.0.5	amazonlinux	377 k
jsr-305	noarch	3.0.2-5.amzn2023.0.4	amazonlinux	32 k
maven-amazon-corretto17	noarch	1:3.8.4-3.amzn2023.0.5	amazonlinux	9.4 k

i-Ofdd550689ff74df8 (jenkins2)
PublicIPs: 13.235.132.151 PrivateIPs: 172.31.46.220

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 32°C Sunny ENG IN 12:16 28-05-2024

- Docker installation :

```
WhatsApp Instagram YouTube Disney+ Hotstar ... Revolutionizing the... ChatGPT DEVOPSC0029 · Go... Search [Alt+S]
aws Services Mumbai Praween

root@ip-172-31-46-220 ~]# yum install docker -y
ast metadata expiration check: 21:50:09 ago on Mon May 27 08:54:57 2024.
dependencies resolved.

Package           Architecture      Version       Repository      Size
installing:
docker             x86_64          25.0.3-1.amzn2023.0.1   amazonlinux    44 M
installing dependencies:
containerd          x86_64          1.7.11-1.amzn2023.0.1   amazonlinux    35 M
iptables-libc      x86_64          1.8.8-3.amzn2023.0.2   amazonlinux    401 k
iptables-nft       x86_64          1.8.8-3.amzn2023.0.2   amazonlinux    183 k
libcgroup          x86_64          3.0-1.amzn2023.0.1     amazonlinux    75 k
libnetfilter_conntrack x86_64          1.0.8-2.amzn2023.0.2   amazonlinux    58 k
libnftnl            x86_64          1.0.1-19.amzn2023.0.2  amazonlinux    30 k
libnftnl           x86_64          1.2.2-2.amzn2023.0.2   amazonlinux    84 k
pigz               x86_64          2.5-1.amzn2023.0.3     amazonlinux    83 k
runc               x86_64          1.1.11-1.amzn2023.0.1  amazonlinux    3.0 M

transaction Summary
install 10 Packages

total download size: 83 M
installed size: 313 M
downloading Packages:

i-0fdd550689ff74df8 (jenkins2)
PublicIPs: 13.235.132.151 PrivateIPs: 172.31.46.220

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
32°C Sunny ENG IN 12:17 28-05-2024
```

- Use this commands to start docker :

- Systemctl start docker – to start the docker
 - Systemctl status docker – to check the docker status
 - Systemctl enable - –now docker – to start at reboot

```
Complete!
[root@ip-172-31-46-220 ~]# sudo systemctl start docker
[root@ip-172-31-46-220 ~]# sudo systemctl enable --now docker
created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
[root@ip-172-31-46-220 ~]# sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: disabled)
     Active: active (running) since Tue 2024-05-28 06:52:12 UTC, 35s ago
TriggeredBy: ● docker.socket
  Docs: https://docs.docker.com
 Main PID: 6065 (dockerd)
   Tasks: 8
  Memory: 41.5M
    CPU: 320ms
   CGroup: /system.slice/docker.service
           └─ 6065 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock --default-ulimit nofile=32768:65536

May 28 06:52:12 ip-172-31-46-220.ap-south-1.compute.internal systemd[1]: Starting docker.service - Docker Application Container Engine...
May 28 06:52:12 ip-172-31-46-220.ap-south-1.compute.internal dockerd[6065]: time="2024-05-28T06:52:12.247114611Z" level=info msg="Starting up"
May 28 06:52:12 ip-172-31-46-220.ap-south-1.compute.internal dockerd[6065]: time="2024-05-28T06:52:12.378288492Z" level=info msg="Loading containers: start."
May 28 06:52:12 ip-172-31-46-220.ap-south-1.compute.internal dockerd[6065]: time="2024-05-28T06:52:12.781619071Z" level=info msg="Loading containers: done."
May 28 06:52:12 ip-172-31-46-220.ap-south-1.compute.internal dockerd[6065]: time="2024-05-28T06:52:12.811068205Z" level=info msg="Docker daemon" commit=f417435 containerd=0.15.0
May 28 06:52:12 ip-172-31-46-220.ap-south-1.compute.internal dockerd[6065]: time="2024-05-28T06:52:12.811408567Z" level=info msg="Daemon has completed initialization"
May 28 06:52:12 ip-172-31-46-220.ap-south-1.compute.internal dockerd[6065]: time="2024-05-28T06:52:12.852576673Z" level=info msg="API listen on /run/docker.sock"
May 28 06:52:12 ip-172-31-46-220.ap-south-1.compute.internal systemd[1]: Started docker.service - Docker Application Container Engine.
lines 1-20/20 (END)
```



➤ Job configuration :

- Source code management : Add git and give the repository URL https://github.com/praveen-gogula/dicet_tv.git. Change the branch specifier

The screenshot shows the Jenkins job configuration interface for a project named 'project-1'. The 'Source Code Management' section is active, showing a 'Repository URL' input field containing 'https://github.com/praveen-gogula/dicet_tv.git'. Below it is a 'Credentials' dropdown set to '- none -' with a '+ Add' button. A dashed box highlights this section. At the bottom are 'Save' and 'Apply' buttons.

• **Add build steps :**

- Add Invoke top-level Maven targets then add maven version and goals.

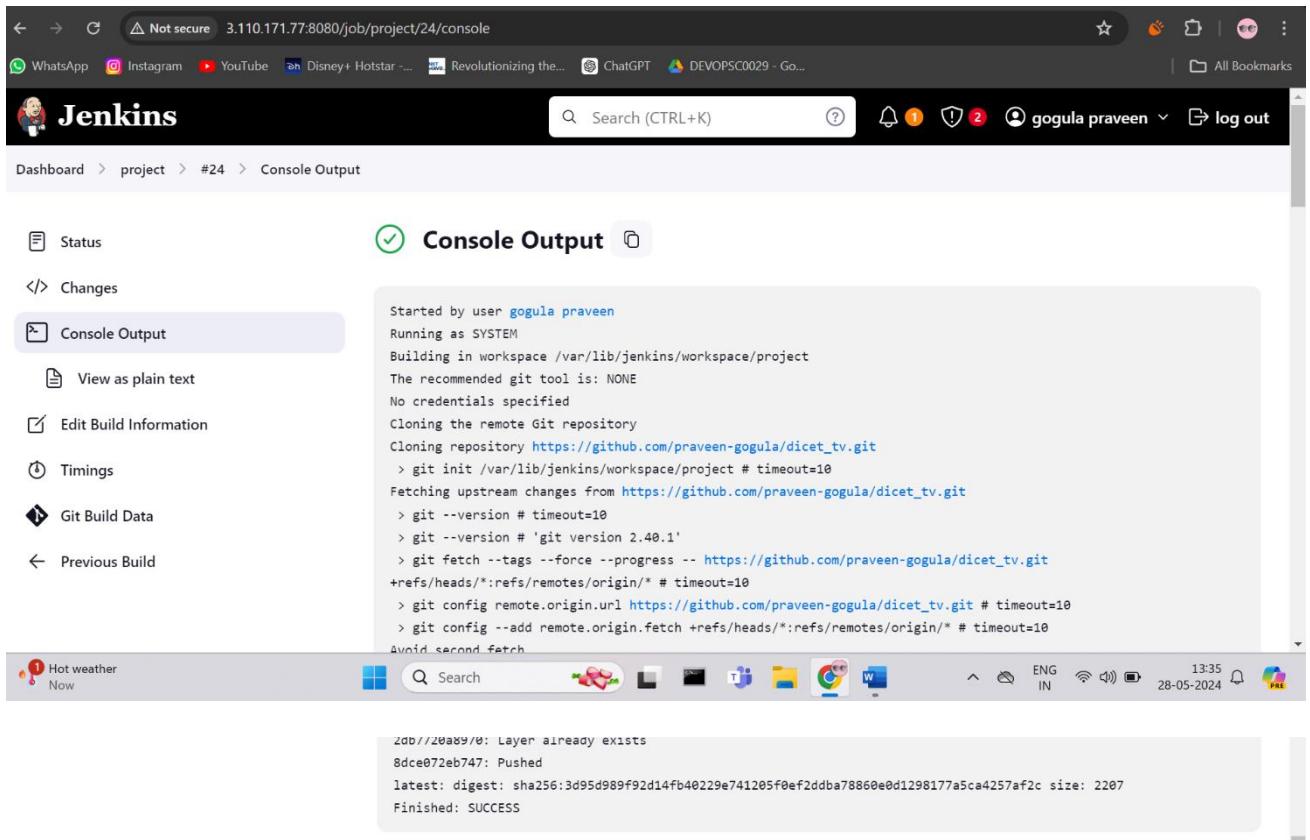
The screenshot shows the Jenkins job configuration interface for 'Build Steps'. The 'Build Steps' section is active, showing an 'Invoke top-level Maven targets' step. Under 'Maven Version', 'maven' is selected. Under 'Goals', 'clean install' is entered. A dashed box highlights this section. At the bottom are 'Save' and 'Apply' buttons.

- In build steps add docker build and publish give repository name as docker username and image name and the give the docker credentials with docker username and password

The screenshot shows the Jenkins configuration interface for a project named 'project-1'. The 'Build Steps' section is selected. A 'Docker Build and Publish' step is being configured. The 'Repository Name' field contains 'praveenreddygogula/dicet_tv'. The 'Tag' field is empty. The 'Docker Host URI' field is also empty. Under 'Server credentials', there is a dropdown menu set to '- none -' and a '+ Add' button. At the bottom are 'Save' and 'Apply' buttons.

The screenshot shows the Jenkins configuration interface for a Jenkins Credentials Provider. The 'Scope' dropdown is set to 'Global (Jenkins, nodes, items, all child items, etc.)'. The 'Username' field contains 'praveenreddygogula'. The 'Treat username as secret' checkbox is unchecked. The 'Password' field contains a masked password. The 'ID' field is empty. At the bottom are 'Save' and 'Apply' buttons.

- Save and build the job :



The screenshot shows the Jenkins interface with the 'Console Output' tab selected. The log output is as follows:

```

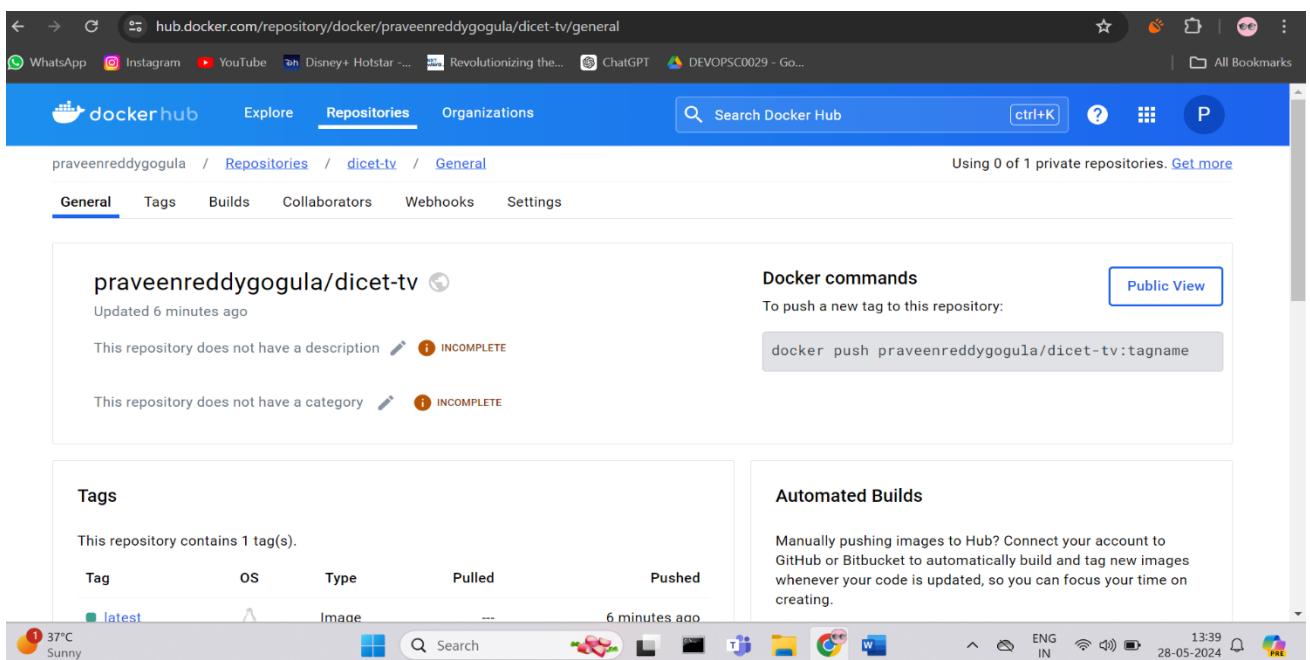
Started by user gogula praveen
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/project
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/praveen-gogula/dicet_tv.git
> git init /var/lib/jenkins/workspace/project # timeout=10
Fetching upstream changes from https://github.com/praveen-gogula/dicet_tv.git
> git --version # timeout=10
> git --version # 'git version 2.40.1'
> git fetch --tags --force --progress -- https://github.com/praveen-gogula/dicet_tv.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url https://github.com/praveen-gogula/dicet_tv.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
2dd//20a89/0: Layer already exists
8dce072eb747: Pushed
latest: digest: sha256:3d95d989f92d14fb40229e741205f0ef2ddba78860e0d1298177a5ca4257af2c size: 2207
Finished: SUCCESS

```

At the bottom of the Jenkins interface, there is a status bar showing 'Hot weather Now' and system icons like battery level, signal strength, and date/time.

REST API Jenkins 2.452.1

Build success and the image is pushed to docker hub.



- **Add build triggers :** select the GitHub hook trigger for GITScm polling build trigger
- This build trigger is used to build will be triggered automatically when changes made in the repository

Configure

Build Triggers

- Trigger builds remotely (e.g., from scripts) ?
- Build after other projects are built ?
- Build periodically ?
- GitHub hook trigger for GITScm polling ?
- Poll SCM ?

Build Environment

- Delete workspace before build starts
- Use secret text(s) or file(s) ?

Save Apply

- Add webhook in dicet-tv repository

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](#).

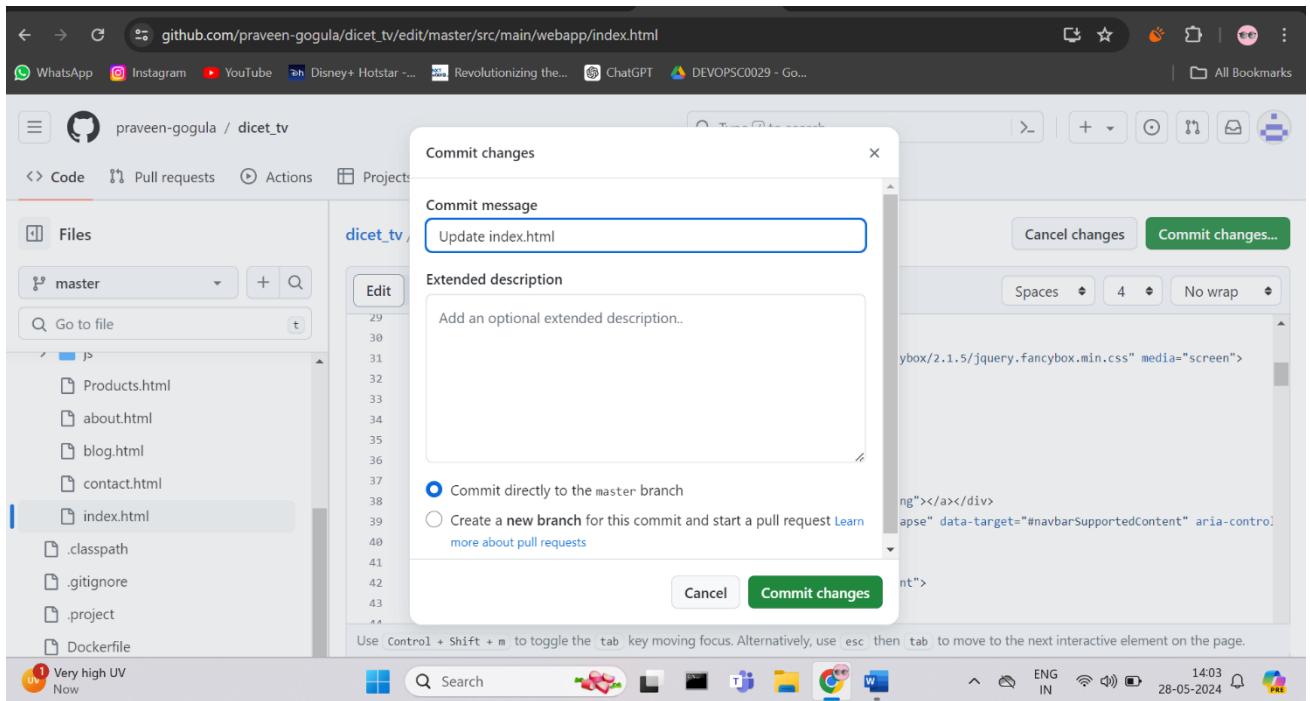
<input checked="" type="checkbox"/> https://3.110.171.77:8080/github-webhook/push (push)	Edit	Delete
--	-------------	---------------

Settings

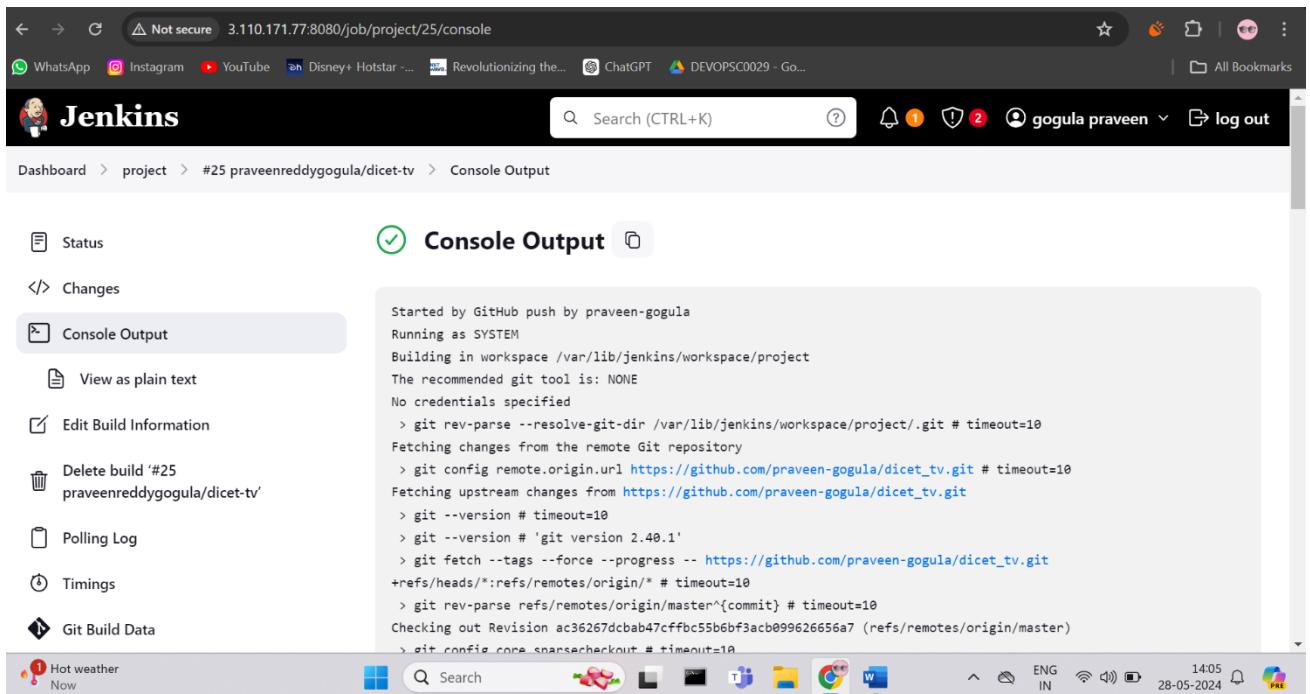
Webhooks

https://github.com/praveen-gogula/dicet_tv/actions

- Commit some changes in the repository :



- Autobuild is done :



- Add execute shell for maven and docker from build steps.

The screenshot shows the Jenkins project configuration page. On the left sidebar, under 'Build Steps', the 'Execute shell' step is selected and highlighted with a blue border. The main panel displays the command entered in the 'Execute shell' field:

```
docker rm -f cont - to delete the container to create new container
docker rmi praveenreddygogula/dicet-tv - to delete the images to relaunch
```

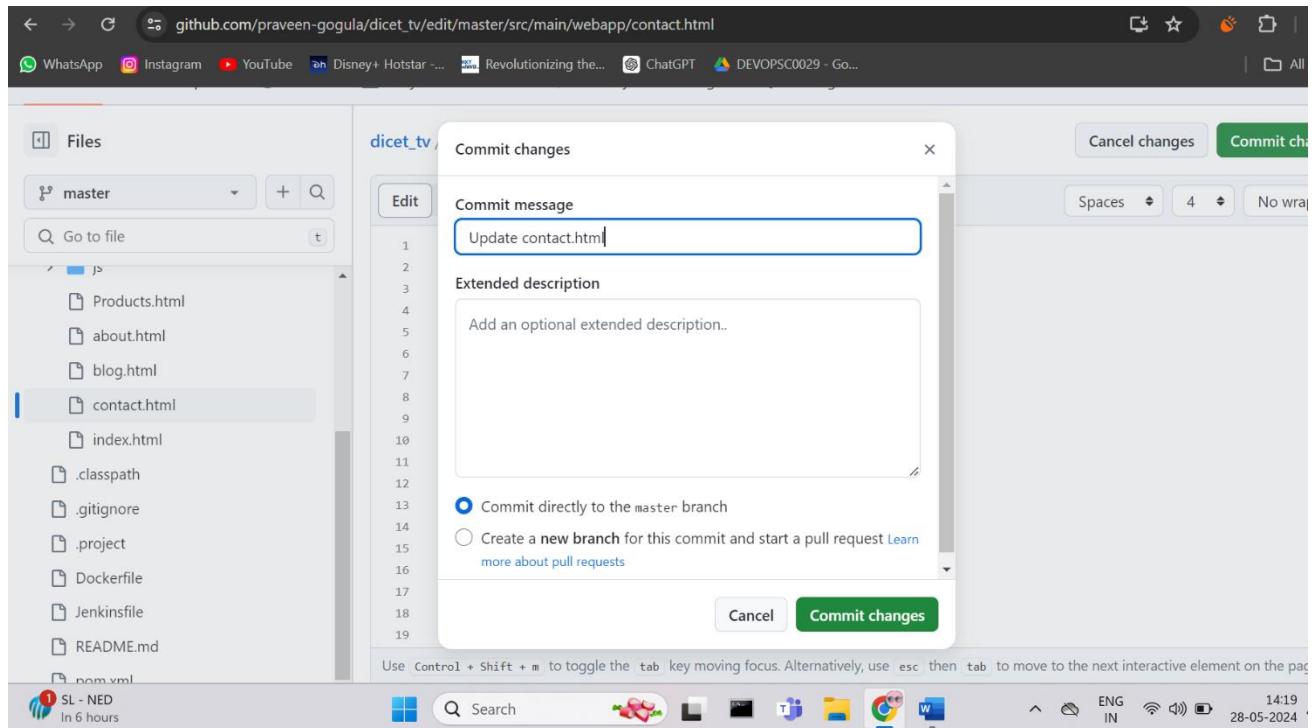
Below the command, there are 'Save' and 'Apply' buttons. The status bar at the bottom shows a weather icon for 37°C and a date/time of 28-05-2024 14:11.

This screenshot shows the same Jenkins configuration page as the previous one, but with a different command in the 'Execute shell' field:

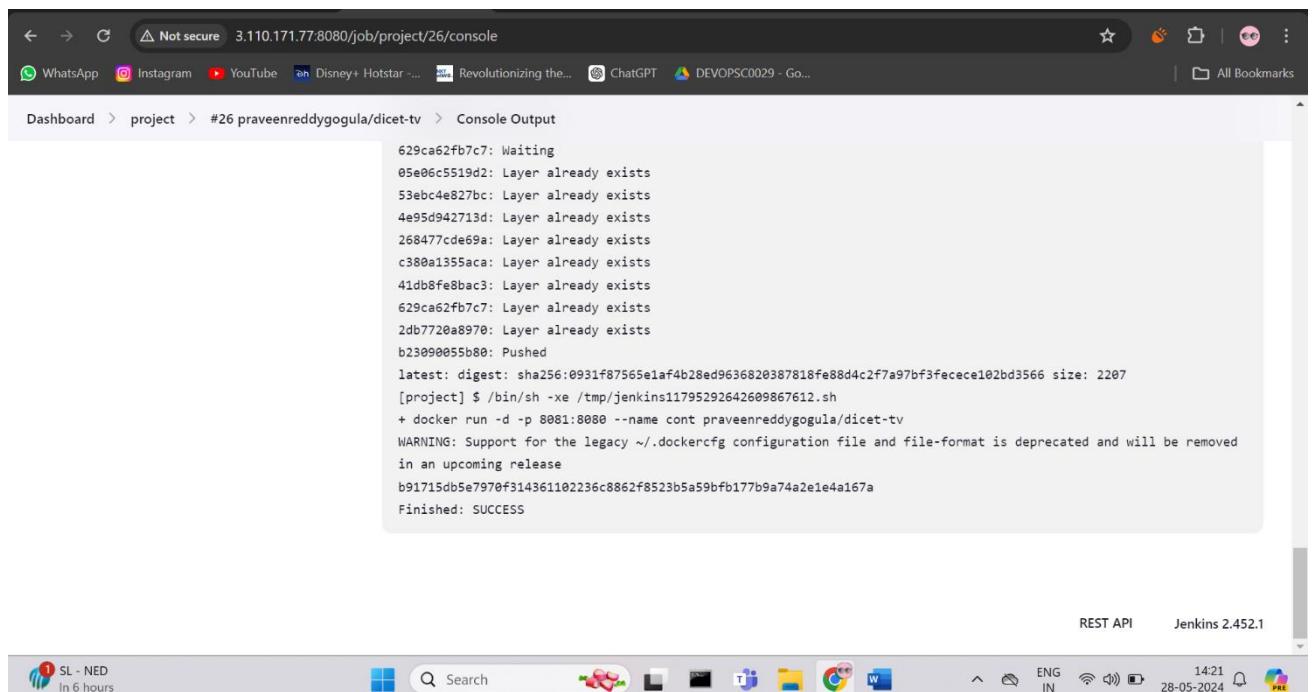
```
docker run -d -p 8081:8080 --name cont praveenreddygogula/dicet-tv - to run the docker image
```

The 'Advanced' dropdown menu is visible below the command field. The status bar at the bottom shows a weather icon for 37°C and a date/time of 28-05-2024 14:16.

- Save and commit some changes in code the build will trigger automatically

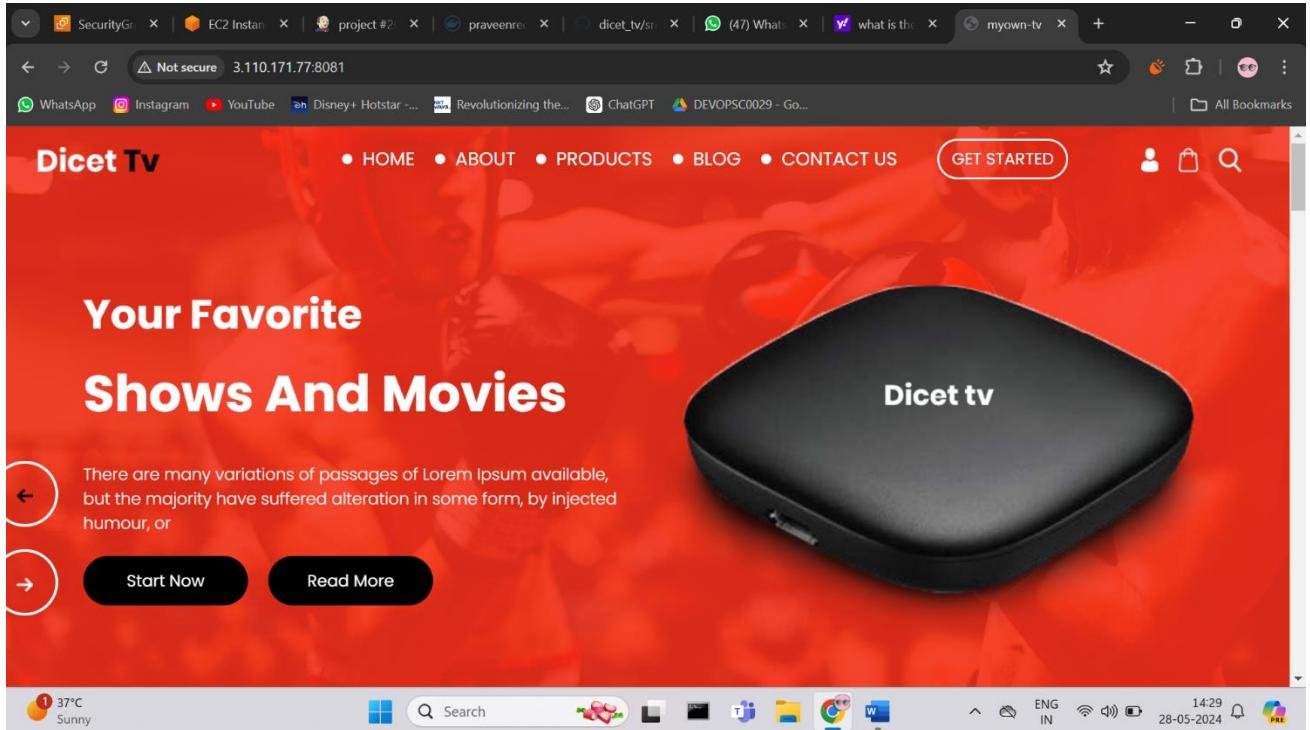


- Build is done automatically



➤ Now live the application:

- Copy the public ip from Jenkins server
- And Browes to publicip:portnumber <http://3.110.171.77:8081/>



- This is my application

I have made changes in title name from dicet_tv to myown_tv