

A PROJECT REPORT ON  
**DEPLOYING DOCKER IMAGE ON EC2  
INSTANCE**  
COURSE  
**DEVOPS WITH AWS**

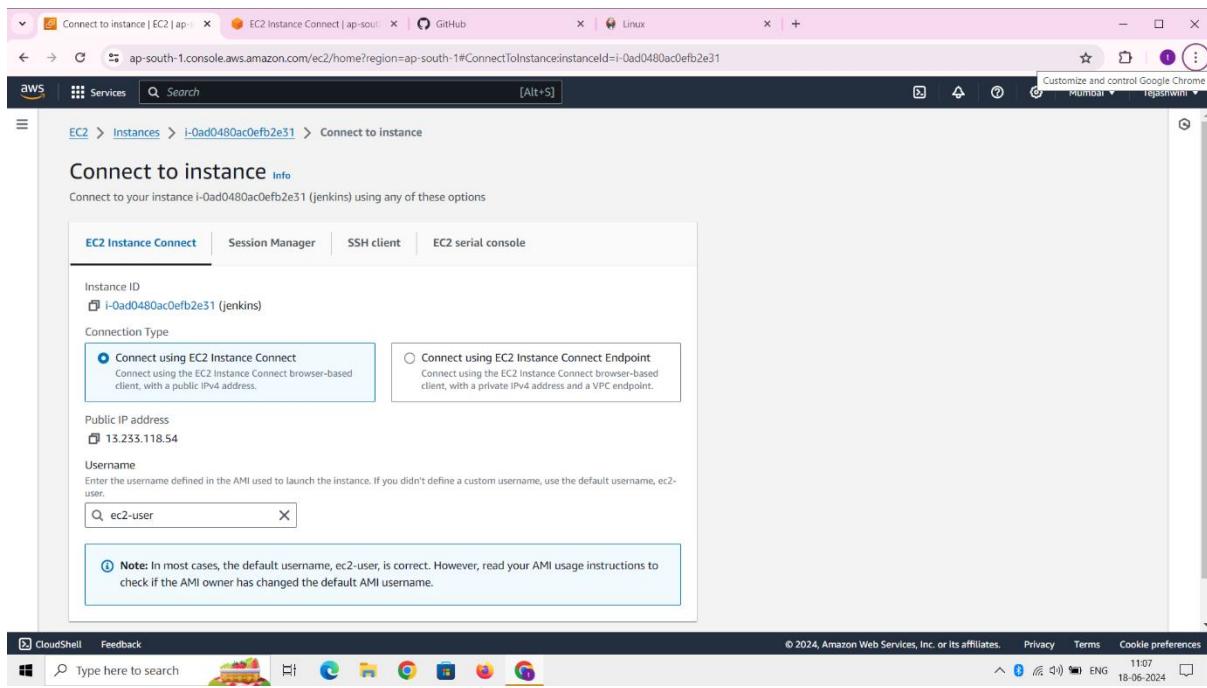
Submitted by  
**Tejashwini Gannamaneni**  
Under The Guidance Of  
**VAMSI BYRAMALA(TRAINER)**  
**ASHOK REKHA(MENTOR)**



Greatcoder Training Institute in Madhapur Hyderabad  
#201, 2<sup>nd</sup> floor, Above Ageless Building, Beside Indian Bank  
Madhapur, Hyderabad-500081  
Landmark: Beside Karachi Bakery, lane, Hyderabad, Telangana 50001

## To launch an instance

- Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2>
- From the EC2 console dashboard
- We want to **Launch instance**
- Under **Name and tags** we want to give **instance name**: Docker.... (give any name)
- Under **Application and OS Images** we want to select **Amazon Linux (AWS) OS**
- Under **Amazon Machine Image (AMI)** we want to select version **Amazon Linux 2(AMI)**
- Under Instance Type want to select **t2. micro (Free tier eligible)**
- Under **Key pair (login)** we want to select Key pair
- Click on **Launch Instance**



- Connect Instance

# Jenkins installation and Configure:

- Jenkins Installation (Search in google)
- Select Installing Jenkins
- Select Linux
- Click on [Red Hat/Alma/Rocky](#)
- Under **Long Term support release**

## Run these commands:

- Long term support release:
- 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
- Yum list | grep "java" (select java-17)
- Yum install java-17-amazon-corretto.x86\_64
- Sudo yum install Jenkins
- Systemctl start Jenkins
- Systemctl enable --now Jenkins

The screenshot shows a terminal window on an AWS EC2 instance. The terminal session is as follows:

```
[root@ip-172-31-5-107 ~]# sudo wget -O /etc/yum.repos.d/jenkins.repo \
> https://pkg.jenkins.io/redhat-stable/jenkins.repo
--2024-06-18 05:42:01-- https://pkg.jenkins.io/redhat-stable/jenkins.repo
Resolving pkg.jenkins.io (pkg.jenkins.io)... 151.101.154.133, 2a04:4e42:24::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)|151.101.154.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 85
Saving to: '/etc/yum.repos.d/jenkins.repo'

100%[=====] 85          ---.K/s   in 0s

2024-06-18 05:42:01 (2.90 MB/s) - '/etc/yum.repos.d/jenkins.repo' saved [85/85]

[root@ip-172-31-5-107 ~]# sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key
[root@ip-172-31-5-107 ~]# sudo yum upgrade
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
No packages marked for update
[root@ip-172-31-5-107 ~]#
```

Below the terminal window, a status bar displays:

i-0ad0480ac0efb2e31 (jenkins)  
PublicIPs: 13.233.118.54 PrivateIPs: 172.31.5.107

At the bottom, a browser interface shows the AWS CloudShell dashboard.

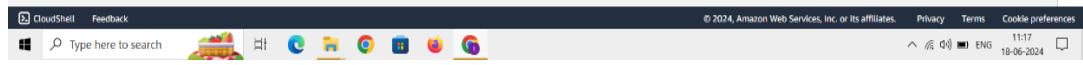
```

root@ip-172-31-5-107:~# yum list | grep "java"
g tether java doc.noarch           1.13.1-13.amzn2      amzn2 core
ant antunit java doc.noarch       1.2-10.amzn2        amzn2 core
ant-contrib java doc.noarch       1.0-0.23.b3.amzn2    amzn2 core
ant java doc.noarch              1.9.16-1.amzn2.0.1   amzn2 core
ant-mail java doc.noarch         1.9.16-1.amzn2.0.1   amzn2 core
antl java doc.noarch             2.7.7-30.amzn2.0.2   amzn2 core
apalache-jev-doc.noarch          1.0-0.1.amzn2        amzn2 core
apache-commons-beanutils java doc.noarch 1.0.3-15.amzn2    amzn2 core
apache-commons-cli [-] java doc.noarch 1.2-13.amzn2      amzn2 core
apache-commons-codec java doc.noarch 1.0-7.amzn2        amzn2 core
apache-commons-collections java doc.noarch 1.5-4.amzn2.0.1  amzn2 core
apache-commons-configuration java doc.noarch          amzn2 core
apache-commons-daemon java doc.noarch 1.0.13-7.amzn2     amzn2 core
apache-commons-dhcp java doc.noarch 1.4-17.amzn2        amzn2 core
apache-commons-digester java doc.noarch 1.8.1-19.amzn2    amzn2 core
apache-commons-exec java doc.noarch 1.1-11.amzn2        amzn2 core
apache-commons-io java doc.noarch 1.12.4-12.amzn2.0.1   amzn2 core
apache-commons-jexl java doc.noarch 2.1.1-9.amzn2        amzn2 core
apache-commons-jxpath java doc.noarch 1.3-20.amzn2       amzn2 core
apache-commons-lang java doc.noarch 2.6-10.amzn2        amzn2 core
apache-commons-lang3 java doc.noarch 3.1-9.amzn2        amzn2 core
apache-commons-logging java doc.noarch 1.1.2-7.amzn2     amzn2 core
apache-commons-mail java doc.noarch 3.2-8.amzn2        amzn2 core
apache-commons-pool java doc.noarch 1.6-9.amzn2        amzn2 core
apache-commons-validator java doc.noarch 1.4.0-8.amzn2    amzn2 core
apache-commons-vfs java doc.noarch 2.0-11.amzn2        amzn2 core

```

i-Oad0480ac0efb2e31 (jenkins)

PublicIPs: 13.233.118.54 PrivateIPs: 172.31.5.107



```

root@ip-172-31-5-107:~# yum install -y java-17-amazon-corretto.x86_64
Verifying : fontconfig-2.13.0-4.3.amzn2.x86_64                                         17/28
Verifying : libXt-1.1.5-3.amzn2.0.2.x86_64                                           18/28
Verifying : glib-2.69-9.amzn2.0.2.x86_64                                            19/28
Verifying : libXinerama-1.1.3-2.1.amzn2.0.2.x86_64                                     20/28
Verifying : libXt-1.7.9-1.amzn2.0.2.x86_64                                           21/28
Verifying : libjava-17-amazon-corretto-17.0.11+9-1.amzn2.1.x86_64                     22/28
Verifying : libXslt-1.1.28-6.amzn2.x86_64                                           23/28
Verifying : python-javapackages-3.4.1-11.amzn2.noarch                                24/28
Verifying : libXst-1.2.3-1.amzn2.0.2.x86_64                                         25/28
Verifying : alsalib-1.1.4.1-2.amzn2.x86_64                                         26/28
Verifying : libatk-1.0.9-9.amzn2.0.2.x86_64                                         27/28
Verifying : javapackages-tools-3.4.1-11.amzn2.noarch                                28/28

Installed:
  java-17-amazon-corretto.x86_64 1:17.0.11+9-1.amzn2.1

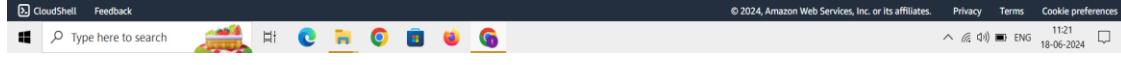
Dependency Installed:
  alsalib.x86_64 0:1.1.4.1-2.amzn2
  dejavu-sans-mono-fonts.noarch 0:2.33-6.amzn2
  fontpackages-filesystem.noarch 0:1.44-8.amzn2
  javapackages-tools.noarch 0:3.4.1-11.amzn2
  libX11.x86_64 0:1.6.7-3.amzn2.0.5
  libXext.x86_64 0:1.0.3-3.amzn2.0.2
  libXrandr.x86_64 0:1.5.1-2.amzn2.0.3
  libXst.x86_64 0:1.2.3-1.amzn2.0.2
  log4j-cve-2021-44228-hotpatch.noarch 0:1.3-7.amzn2
  python-javapackages.noarch 0:3.4.1-11.amzn2

Complete!
root@ip-172-31-5-107:~# 

```

i-Oad0480ac0efb2e31 (jenkins)

PublicIPs: 13.233.118.54 PrivateIPs: 172.31.5.107



```

root@ip-172-31-5-107 ~# yum install jenkins -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Resolving Dependencies
--> Running transaction check
-->> Package jenkins.noarch 0:2.452.2-1.1 will be installed
-->> Finished Dependency Resolution

Dependencies Resolved

-----  


| Package     | Arch    | Version     | Repository | Size |
|-------------|---------|-------------|------------|------|
| installing: | jenkins | 2.452.2-1.1 | jenkins    | 89 M |


Transaction Summary
Install 1 Package

Total download size: 89 M
Installed size: 89 M
Downloading packages:
jenkins-2.452.2-1.1.noarch.rpm
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : jenkins-2.452.2-1.1.noarch
1/1

i-Oad0480ac0efb2e31 (jenkins)
PublicIPs: 13.233.118.54 PrivateIPs: 172.31.5.107

```

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 11:22 18-06-2024

```

Transaction Summary
Install 1 Package

Total download size: 89 M
Installed size: 89 M
Downloading packages:
jenkins-2.452.2-1.1.noarch.rpm
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : jenkins-2.452.2-1.1.noarch
  Verifying : jenkins-2.452.2-1.1.noarch
1/1

Installed:
  jenkins.noarch 0:2.452.2-1.1

Complete!
[root@ip-172-31-5-107 ~]# systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
  Loaded: loaded (/usr/lib/systemd/system/jenkins.service; disabled; vendor preset: disabled)
    Active: inactive (dead)
[root@ip-172-31-5-107 ~]# systemctl start jenkins
[root@ip-172-31-5-107 ~]# systemctl enable jenkins
bash: systemctl: command not found
[root@ip-172-31-5-107 ~]# systemctl enable jenkins
Created symlink from /etc/systemd/system/multi-user.target.wants/jenkins.service to /usr/lib/systemd/system/jenkins.service.
[root@ip-172-31-5-107 ~]# 

i-Oad0480ac0efb2e31 (jenkins)
PublicIPs: 13.233.118.54 PrivateIPs: 172.31.5.107

```

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 11:25 18-06-2024

- We should edit inbound rules in security groups
- We have to give Port number (8080-8083) and Anywhere-IPv4
- Anywhere-IPv4 to make port available for public
- Save rules

The screenshot shows the AWS EC2 Security Groups inbound rules configuration. It lists two rules:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-04258a3bda114661d	SSH	TCP	22	Custom	0.0.0.0/0
-	Custom TCP	TCP	8080 - 8083	Anywh...	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." There are buttons for "Cancel", "Preview changes", and "Save rules".

- Copy the public Ips from Instance
- Jenkins runs on port number 8080
- Paste in google PublicIPs:8080

The screenshot shows the Jenkins 'Unlock Jenkins' setup page. It displays the following text and fields:

**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
```

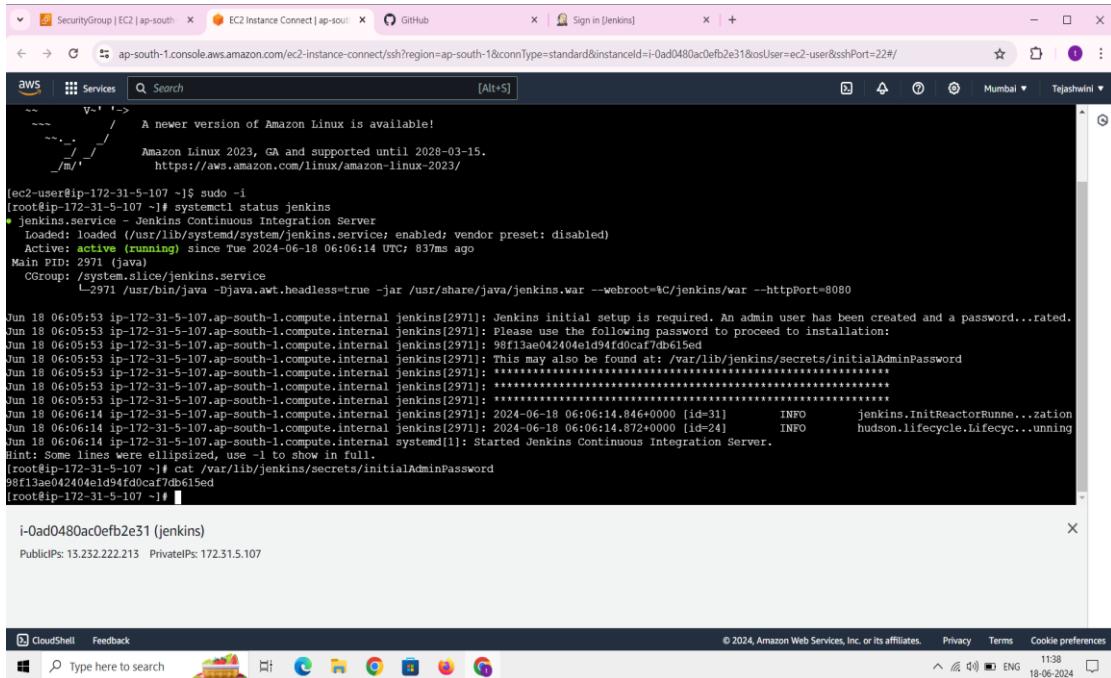
Please copy the password from either location and paste it below.

**Administrator password**

**Continue**

The browser address bar shows the URL: `Not secure 13.232.222.213:8080/login?from=%2F`.

- Copy this path **/var/lib/Jenkins/secrets/initialAdminPassword**
- Cat **/var/lib/Jenkins/secrets/initialAdminPassword**
- Using “cat” command to show the initial admin password for Jenkins



```

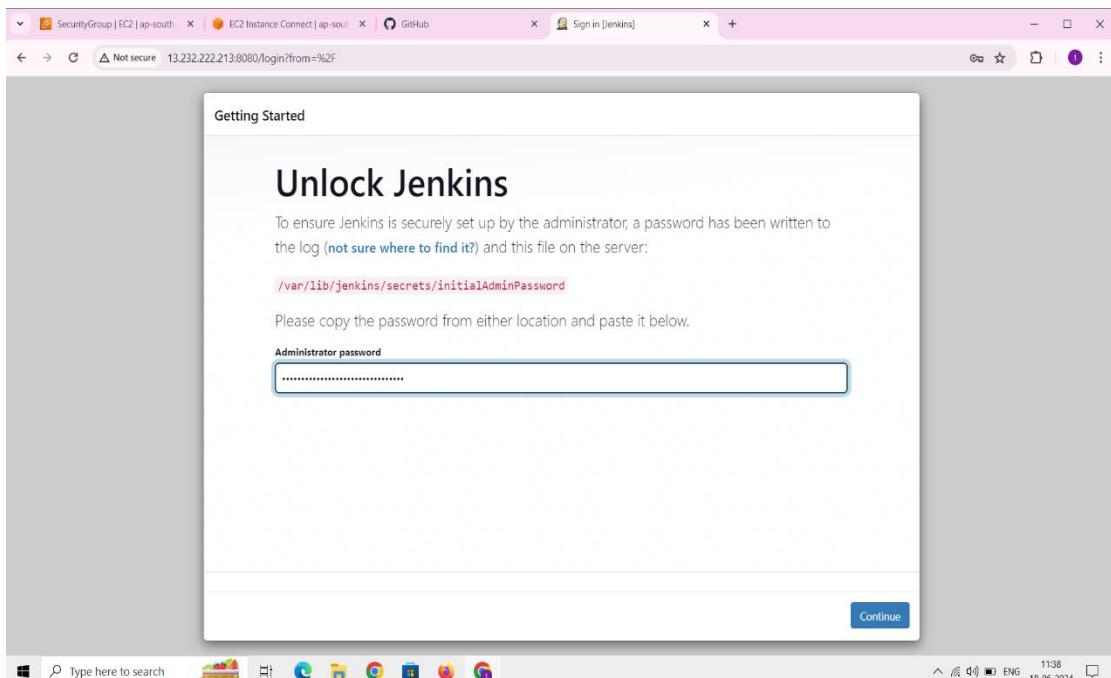
SecurityGroup | EC2 | ap-south-1 | EC2 Instance Connect | ap-south-1 | GitHub | Sign in [Jenkins] | + | - | □ | ×
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-south-1&connType=standard&instanceId=i-0ad0480ac0efb2e31&iosUser=ec2-user&sshPort=22#/ 
Mumbai | Tejaswini | 

aws Services Search [Alt+S]
A newer version of Amazon Linux is available!
Amazon Linux 2023, GA and supported until 2028-03-15.
https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-172-31-5-107 ~]$ sudo -i
[root@ip-172-31-5-107 ~]# systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
  Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; vendor preset: disabled)
  Active: active (running) since Tue 2024-06-18 06:06:14 UTC; 837ms ago
    Main PID: 2971 (java)
   CGroup: /system.slice/jenkins.service
          └─2971 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --httpPort=8080

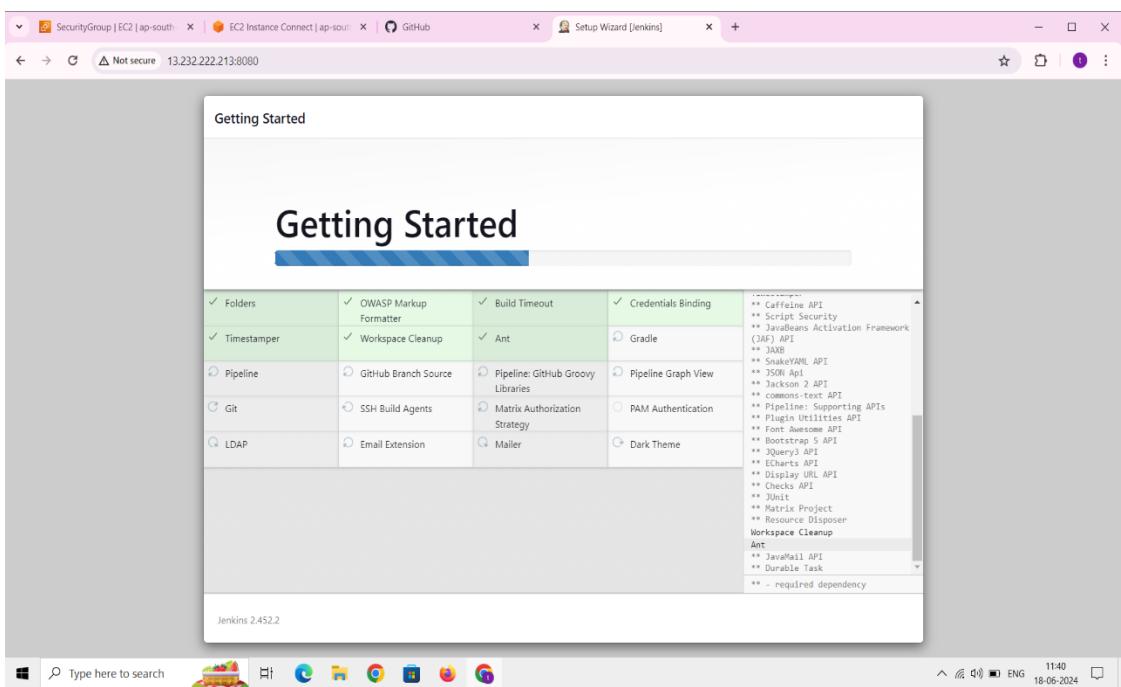
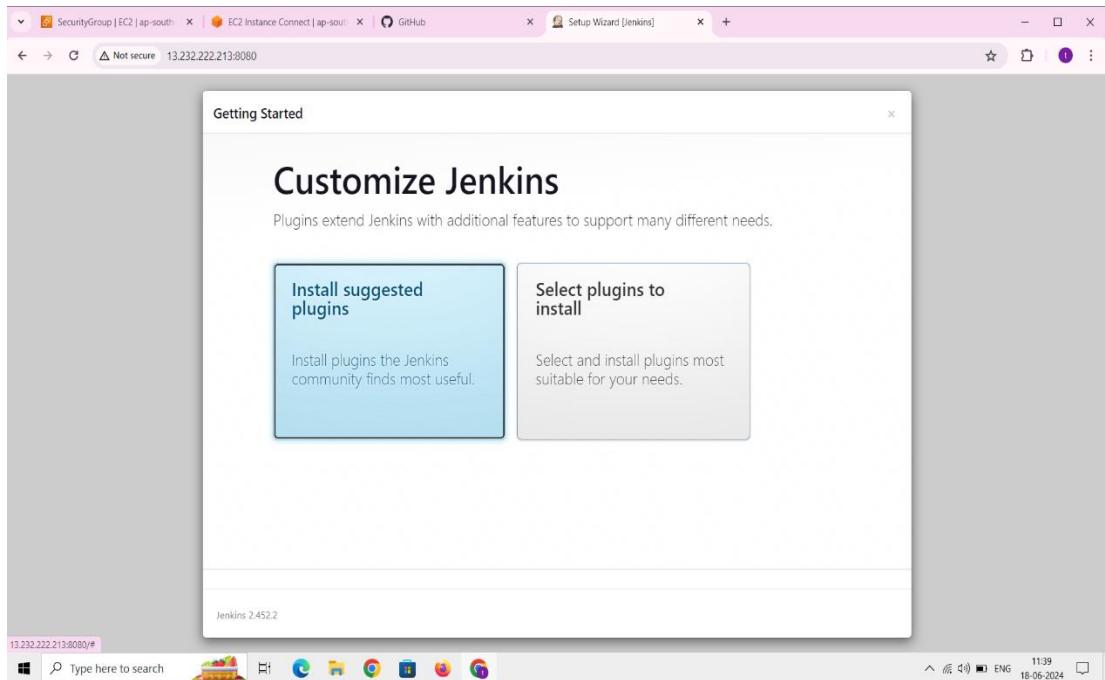
Jun 18 06:05:53 ip-172-31-5-107.ap-south-1.compute.internal jenkins[2971]: Jenkins initial setup is required. An admin user has been created and a password...rated.
Jun 18 06:05:53 ip-172-31-5-107.ap-south-1.compute.internal jenkins[2971]: Please use the following password to proceed to installation:
Jun 18 06:05:53 ip-172-31-5-107.ap-south-1.compute.internal jenkins[2971]: 98f13ae042404e1d94fd0caf7db615ed
Jun 18 06:05:53 ip-172-31-5-107.ap-south-1.compute.internal jenkins[2971]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Jun 18 06:05:53 ip-172-31-5-107.ap-south-1.compute.internal jenkins[2971]: ****
Jun 18 06:05:53 ip-172-31-5-107.ap-south-1.compute.internal jenkins[2971]: ****
Jun 18 06:05:53 ip-172-31-5-107.ap-south-1.compute.internal jenkins[2971]: ****
Jun 18 06:06:14 ip-172-31-5-107.ap-south-1.compute.internal jenkins[2971]: 2024-06-18 06:06:14.846+0000 [id=31]      INFO  jenkins.InitReactorRunn...
Jun 18 06:06:14 ip-172-31-5-107.ap-south-1.compute.internal jenkins[2971]: 2024-06-18 06:06:14.872+0000 [id=24]      INFO  hudson.lifecycle.Lifecyc...
Jun 18 06:06:14 ip-172-31-5-107.ap-south-1.compute.internal systemd[1]: Started Jenkins Continuous Integration Server.
Hint: Some lines were ellipsized, use -l to show in full.
[root@ip-172-31-5-107 ~]# cat /var/lib/jenkins/secrets/initialAdminPassword
98f13ae042404e1d94fd0caf7db615ed
[root@ip-172-31-5-107 ~]#

```

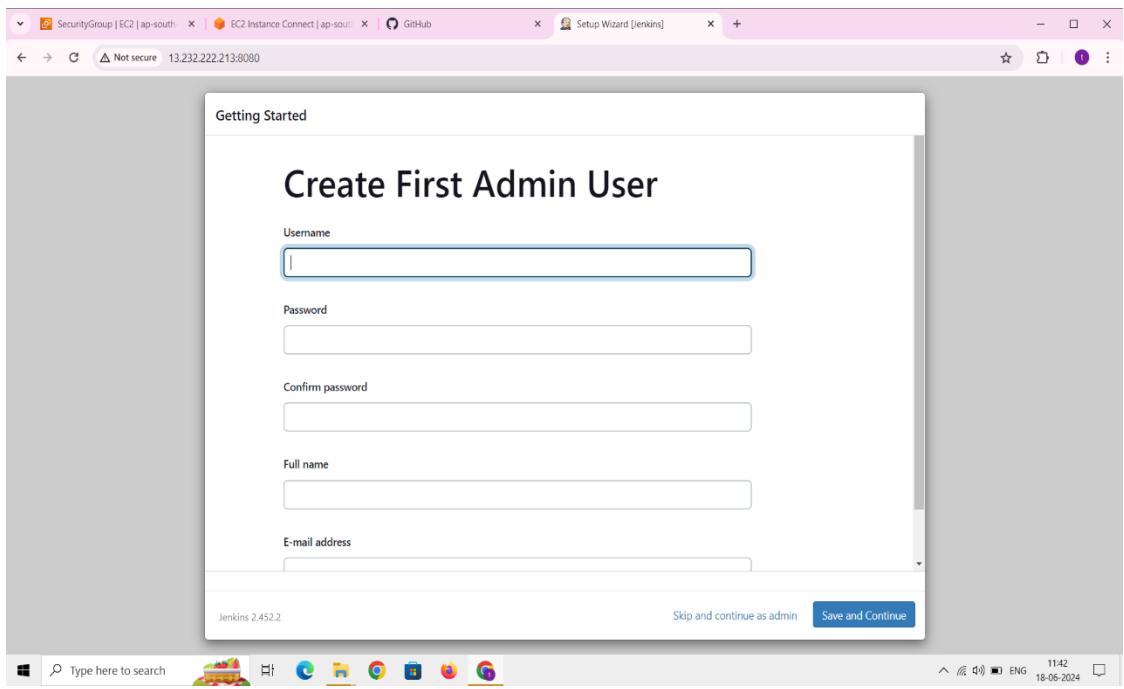
i-0ad0480ac0efb2e31 (jenkins)  
PublicIPs: 15.232.222.213 PrivateIPs: 172.31.5.107



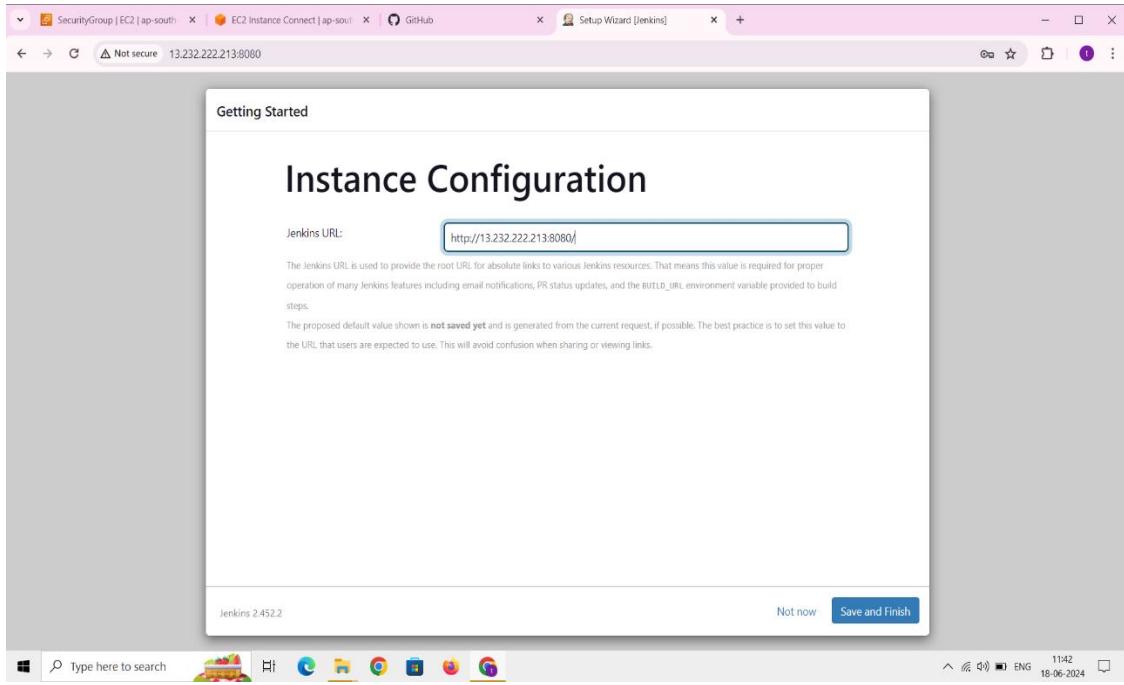
- Click on continue



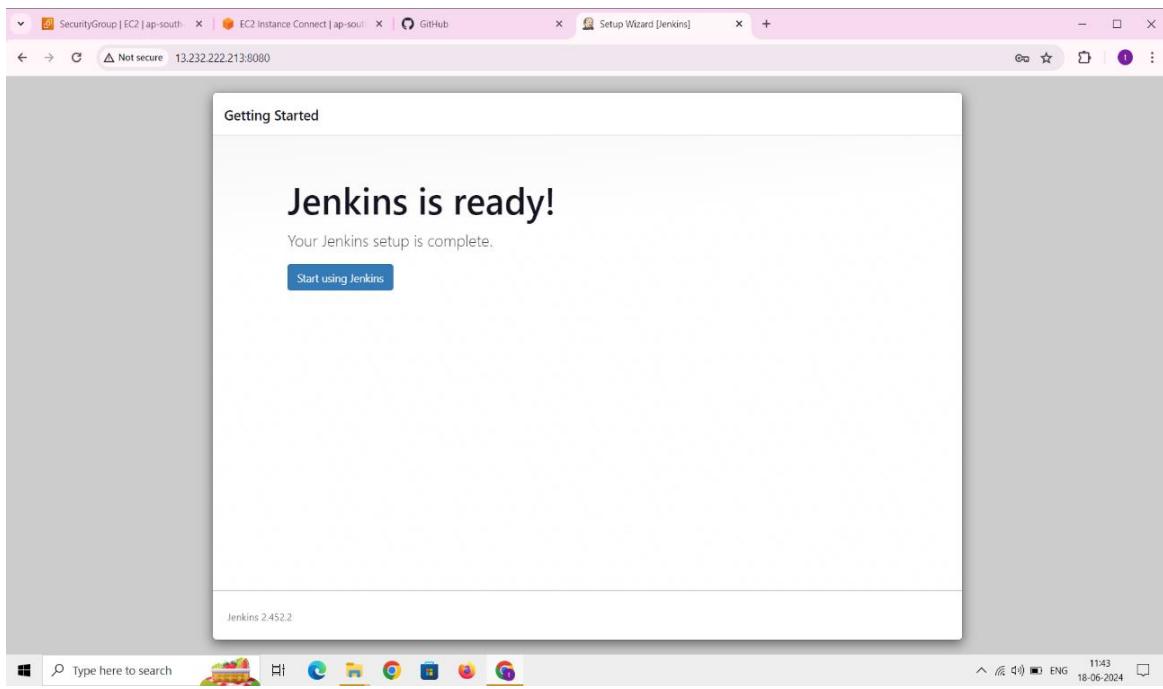
- Jenkins is Getting Started



- Create First Admin User Details
- Click Save and Continue



- Click on Save and Finish



Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Set up a distributed build

REST API Jenkins 2.452.2



- It will open Jenkins Dashboard

## Create a Job:

- ❖ Click on New Item
- ❖ Enter an item name (job 1)
- ❖ Select Free Style Project
- ❖ Click on OK

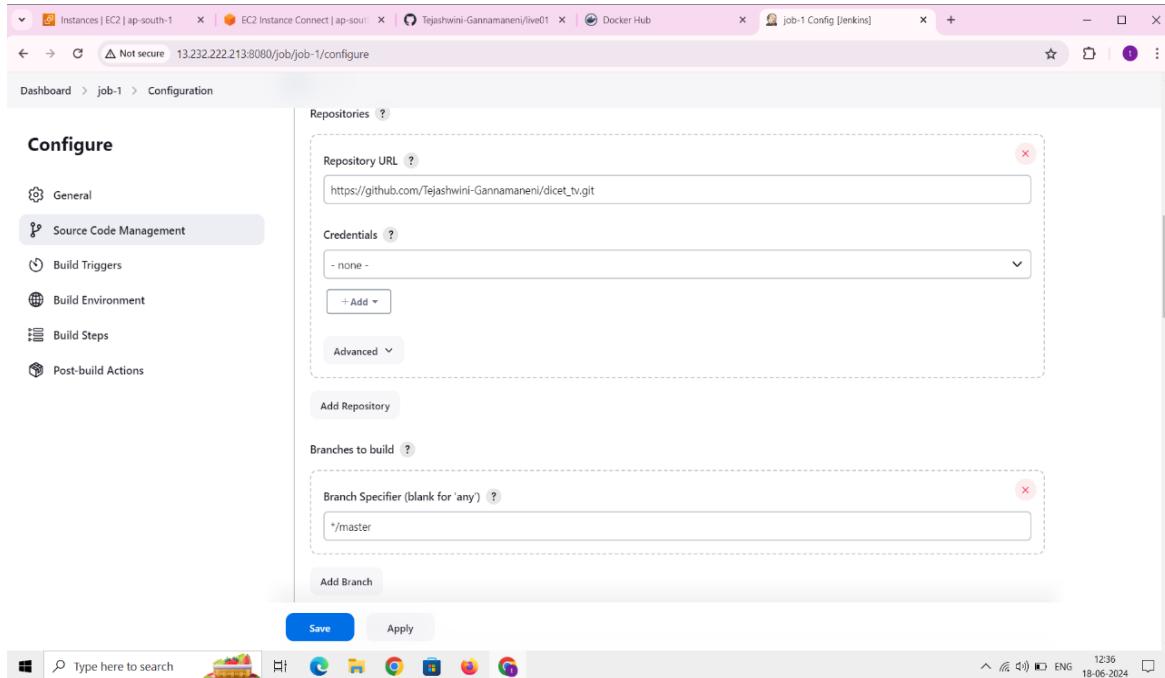
The screenshot shows the Jenkins dashboard with a search bar at the top containing 'Search (CTRL+K)'. A user profile 'Tejaswini Gannamaneni' is logged in. Below the search bar, there's a 'Dashboard' link. The main area is titled 'Enter an item name' with a required field 'job-1'. It lists several job types: 'Freestyle project' (classic general-purpose), 'Pipeline' (orchestrates long-running activities), 'Multi-configuration project' (suitable for multiple configurations), 'Folder' (creates a container for nested items), and 'Multi-branch Pipeline' (creates a pipeline for multiple branches). A note at the bottom says 'Creates a set of Pipeline projects according to detected branches in one SCM repository'. The system status bar at the bottom shows 'Type here to search' and various icons.

The screenshot shows the Jenkins configuration page for 'job-1'. The title bar indicates 'Instances | EC2 | ap-south-1' and 'Not secure'. The URL is '13.232.222.213:8080/job/job-1/configure'. The Jenkins logo is at the top. The navigation bar shows 'Dashboard > job-1 > Configuration'. On the left, a sidebar lists 'Configure' sections: General, Source Code Management, Build Triggers, Build Environment, Build Steps, and Post-build Actions. The 'General' section is active, showing a 'Description' input field and several checkboxes for build triggers: 'Discard old builds', 'GitHub project', 'This project is parameterized', 'Throttle builds', and 'Execute concurrent builds if necessary'. At the bottom, there are 'Save' and 'Apply' buttons. The system status bar at the bottom shows 'Type here to search' and various icons.

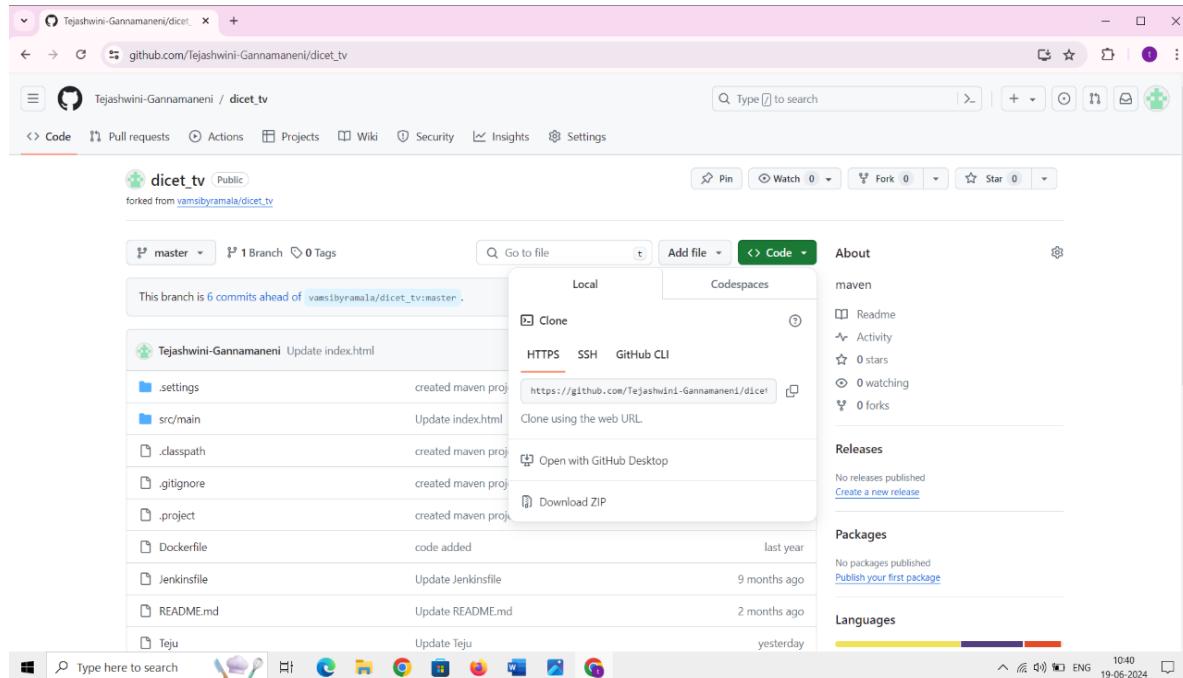
❖ Now we have to Configure the job

# Configure the Job:

- ❖ We have to select GIT



- ❖ We have to give Repository URL
- ❖ Copy the URL from GitHub
- ❖ Iam taking the dicet\_tv repository from my GitHub Account



- ❖ Copy the link from <https://github.com/Tejashwini-Gannamaneni/dicet-tv>
- ❖ Paste the link on Repository URL (Job1-configure)

The screenshot shows the Jenkins job configuration interface for 'job-1'. Under the 'Source Code Management' section, the 'Repository URL' is set to [https://github.com/Tejashwini-Gannamaneni/dicet\\_tv.git](https://github.com/Tejashwini-Gannamaneni/dicet_tv.git). The 'Branch Specifier' is set to `*/master`. The Jenkins bar at the bottom shows various icons and the date/time as 12:36 18-06-2024.

- ❖ Change the Branch Specifier (master) in my GitHub dicet-tv is in master branch
- ❖ We don't want to give credentials for Git
- ❖ Click on Save
- ❖ In server we want to install Git (yum install git -y)

```
[root@ip-172-31-5-107 ~]# yum install git -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Resolving Dependencies
--> Running transaction check
--> Package git.x86_64 0:2.40.1-1.amzn2.0.3 will be installed
--> Processing Dependency: git-core = 2.40.1-1.amzn2.0.3 for package: git-2.40.1-1.amzn2.0.3.x86_64
--> Processing Dependency: git-core-doc = 2.40.1-1.amzn2.0.3 for package: git-2.40.1-1.amzn2.0.3.x86_64
--> Processing Dependency: perl-git = 2.40.1-1.amzn2.0.3 for package: git-2.40.1-1.amzn2.0.3.x86_64
--> Processing Dependency: perl-git for package: git-2.40.1-1.amzn2.0.3.x86_64
--> Processing Dependency: perl(Term::ReadKey) for package: git-2.40.1-1.amzn2.0.3.x86_64
--> Running transaction check
--> Package git-core.x86_64 0:2.40.1-1.amzn2.0.3 will be installed
--> Package git-core-doc.noarch 0:2.40.1-1.amzn2.0.3 will be installed
--> Package perl-git.noarch 0:2.40.1-1.amzn2.0.3 will be installed
--> Processing Dependency: perl(Error) for package: perl-git-2.40.1-1.amzn2.0.3.noarch
--> Package perl-TermReadKey.x86_64 0:2.30-20.amzn2.0.2 will be installed
--> Running transaction check
--> Package perl-Error.noarch 1:0.17020-2.amzn2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

Package          Arch      Version       Repository      Size
Installing:    git        x86_64     2.40.1-1.amzn2.0.3   amzn2-core      54 k
Installing for dependencies:

i-Oad0480ac0efb2e31 (jenkins)
PublicIPs: 13.232.222.213 PrivateIPs: 172.31.5.107
```

- ❖ Go to Jenkins Dashboard
- ❖ Click on Manage Jenkins

The screenshot shows the Jenkins Manage Jenkins interface. The 'Manage Jenkins' tab is active in the sidebar. The main content area displays the 'Operating system end of life monitor' message, which informs the user that Jenkins is running on Amazon Linux 2, which stopped supporting Amazon Linux 2 as of 2023-11-16. It also includes sections for System Configuration (System, Tools, Plugins, Nodes, Clouds, Appearance) and Security (Security, Credentials, Credential Providers). A 'Build Queue' section indicates 'No builds in the queue.'

- ❖ Click on plugins

The screenshot shows the Jenkins Plugin Manager interface. The 'Plugins' tab is active in the sidebar. The main content area displays the 'Updates' section, which shows a message indicating 'No updates available'. Below this, a note states 'Disabled rows are already upgraded, awaiting restart. Shaded but selectable rows are in progress or failed.' The status bar at the bottom right shows 'Jenkins 2.452.2'.

- ❖ Click on Available plugins
- ❖ Search for Docker plugins
- ❖ Select Docker and CloudBees Docker Build and Publish plugins
- ❖ Click on Install

The screenshot shows the Jenkins plugin manager interface. The left sidebar has tabs for 'Updates', 'Available plugins' (which is selected), 'Installed plugins', 'Advanced settings', and 'Download progress'. A search bar at the top right contains the query 'docker'. Below the search bar is a table listing several Docker-related plugins:

Install	Name	Released
<input checked="" type="checkbox"/>	Docker 1.6.2	14 days ago
<input type="checkbox"/>	Docker Commons 439.va_3cb_0a_6a_fb_29	11 mo ago
<input type="checkbox"/>	Docker Pipeline 580.vc0c340686b_54	27 days ago
<input type="checkbox"/>	Docker API 3.3.6-90.ve7c5c7535ddd	15 days ago
<input type="checkbox"/>	docker-build-step 2.12	

At the bottom right of the table, there is an 'Install' button. The status bar at the bottom right shows '12:00 18-06-2024'.

- It allows to build docker images on a docker server and then publish them to DockerHub

This screenshot shows the same Jenkins plugin manager interface as the previous one, but with a different set of results. The 'Available plugins' tab is still selected, and the search term 'docker' is in the search bar. The table now includes:

Install	Name	Released
<input type="checkbox"/>	docker-build-step 2.12	21 days ago
<input checked="" type="checkbox"/>	CloudBees Docker Build and Publish 1.4.0	1 yr 9 mo ago
<input type="checkbox"/>	Amazon ECR 1.114.vfd22430621ff	1 yr 4 mo ago
<input type="checkbox"/>	Docker Compose Build Step 1.0	5 yr 11 mo ago
<input type="checkbox"/>	Docker Slaves 1.0.7	6 yr 11 mo ago

The 'Install' button is present at the top right of the table area. The status bar at the bottom right shows '12:00 18-06-2024'.

The screenshot shows the Jenkins 'Manage Jenkins > Plugins' page. The left sidebar has sections for 'Updates', 'Available plugins', 'Installed plugins', 'Advanced settings', and 'Download progress'. The 'Download progress' section is currently selected. A table lists various Jenkins components and their status: PAM Authentication, LDAP, Email Extension, Mailer, Theme Manager, Dark Theme, Loading plugin extensions, Cloud Statistics, Authentication Tokens API, Docker Commons, Apache HttpComponents Client 5x API, Docker API, Docker, SSH server, CloudBees Docker Build and Publish, and Loading plugin extensions, all marked as 'Success' with green checkmarks.

→ 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

REST API Jenkins 2.452.2

## ❖ Go to Manage Jenkins

The screenshot shows the Jenkins 'Manage Jenkins' page. The left sidebar includes 'New Item', 'Build History', 'Manage Jenkins' (which is selected), and 'My Views'. The main area features several configuration sections: 'Operating system end of life monitor' (warning about support ending on 2023-11-16), 'System Configuration' (with links to 'System', 'Tools', 'Nodes', 'Clouds', and 'Plugins'), 'Security' (with links to 'Security', 'Credentials', and 'Credential Providers'), and 'Appearance'.

More Info Ignore

Search settings

Operating system end of life monitor

You are running Jenkins on Amazon Linux 2. Jenkins stopped supporting Amazon Linux 2 as of **2023-11-16**.  
Please upgrade to a supported operating system.  
Refer to [the documentation](#) for details.

System Configuration

System Tools Plugins

Nodes Clouds Appearance

Security Credentials Credential Providers

REST API Jenkins 2.452.2

## ❖ Click on Tools

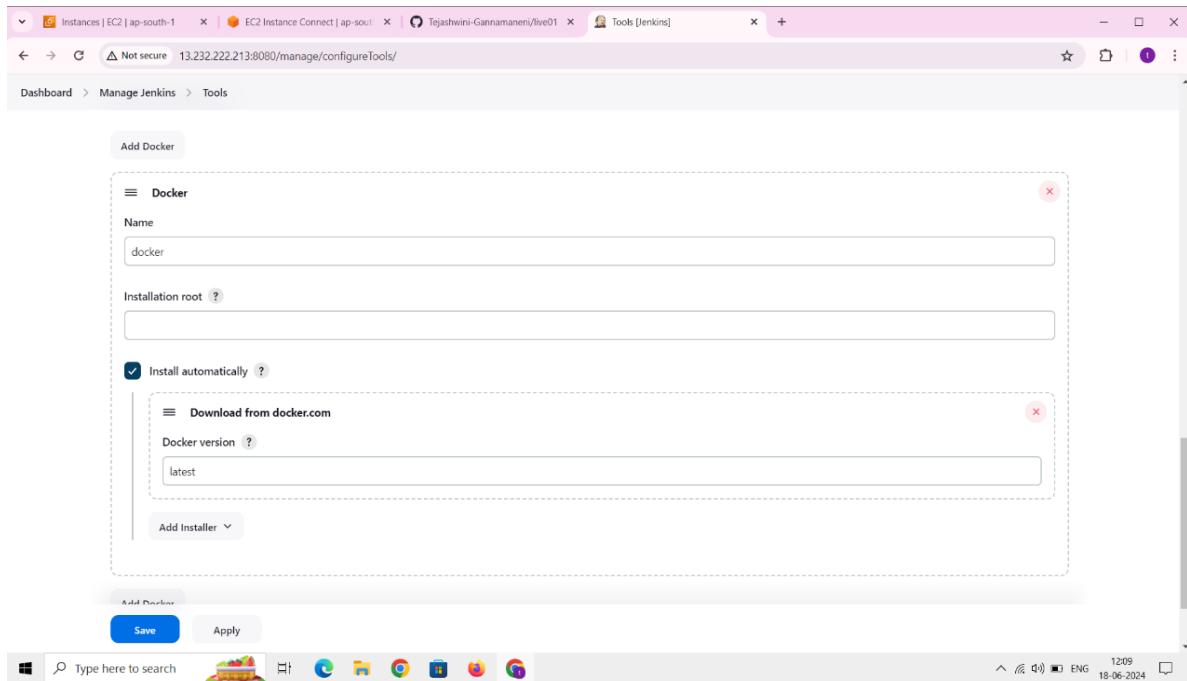
The screenshot shows the Jenkins 'Tools' configuration page. At the top, there are dropdown menus for 'Default settings provider' (set to 'Use default maven settings') and 'Default global settings provider' (set to 'Use default maven global settings'). Below these are sections for 'JDK installations' (with a 'Add JDK' button) and 'Git installations'. At the bottom right of the page, there are 'Save' and 'Apply' buttons.

❖ Click on Add Maven

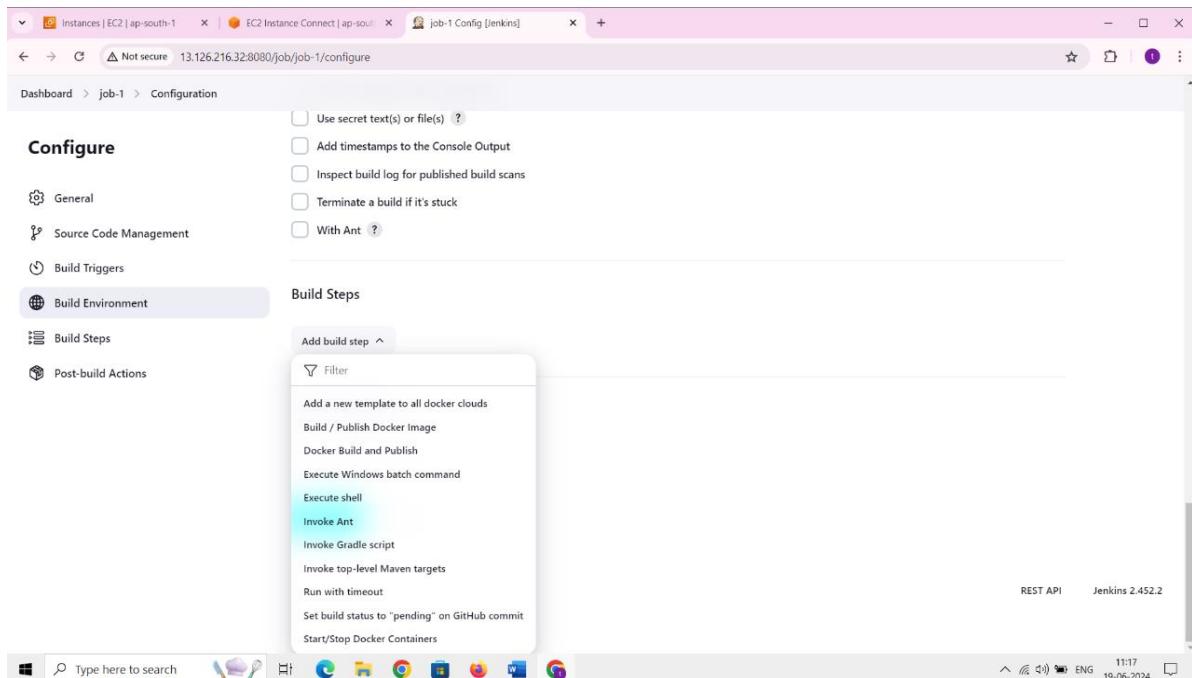
The screenshot shows the 'Add Maven' configuration dialog. It has a 'Name' field containing 'maven' and a checked 'Install automatically' checkbox. Under the 'Install from Apache' section, the 'Version' dropdown is set to '3.9.8'. There is also an 'Add Installer' button. At the bottom of the dialog are 'Save' and 'Apply' buttons.

- ❖ Click on Add Docker  
❖ Select checkbox Install Automatically

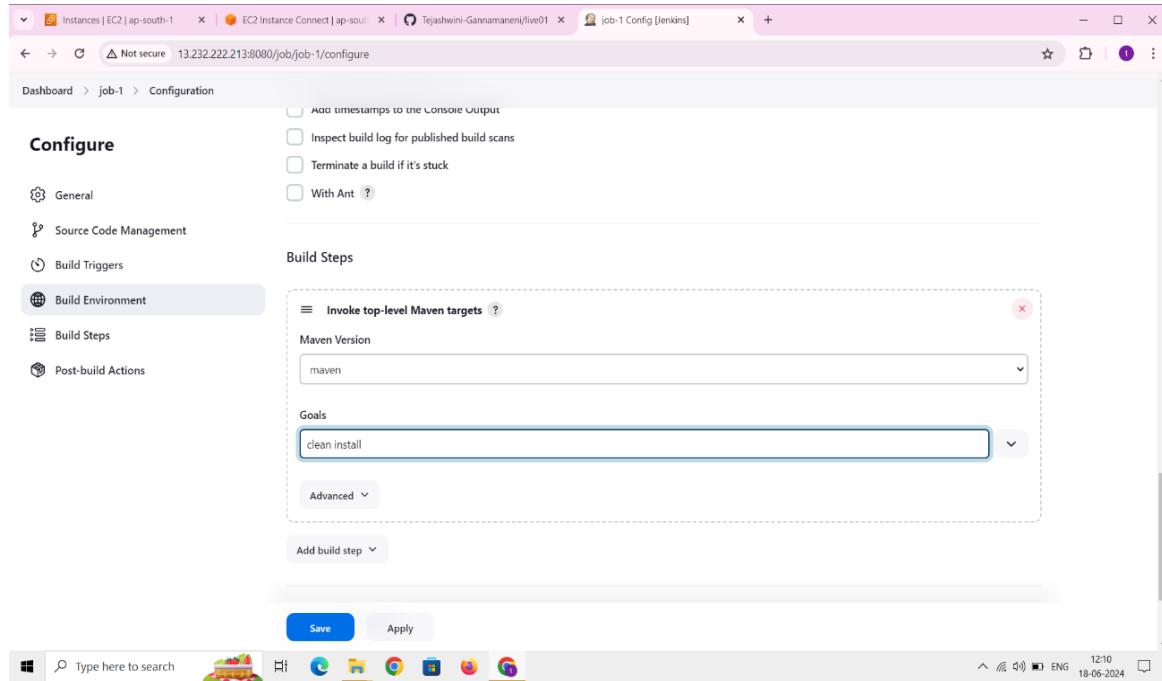
- ❖ Click on Add Installer
- ❖ Select Download from docker.com



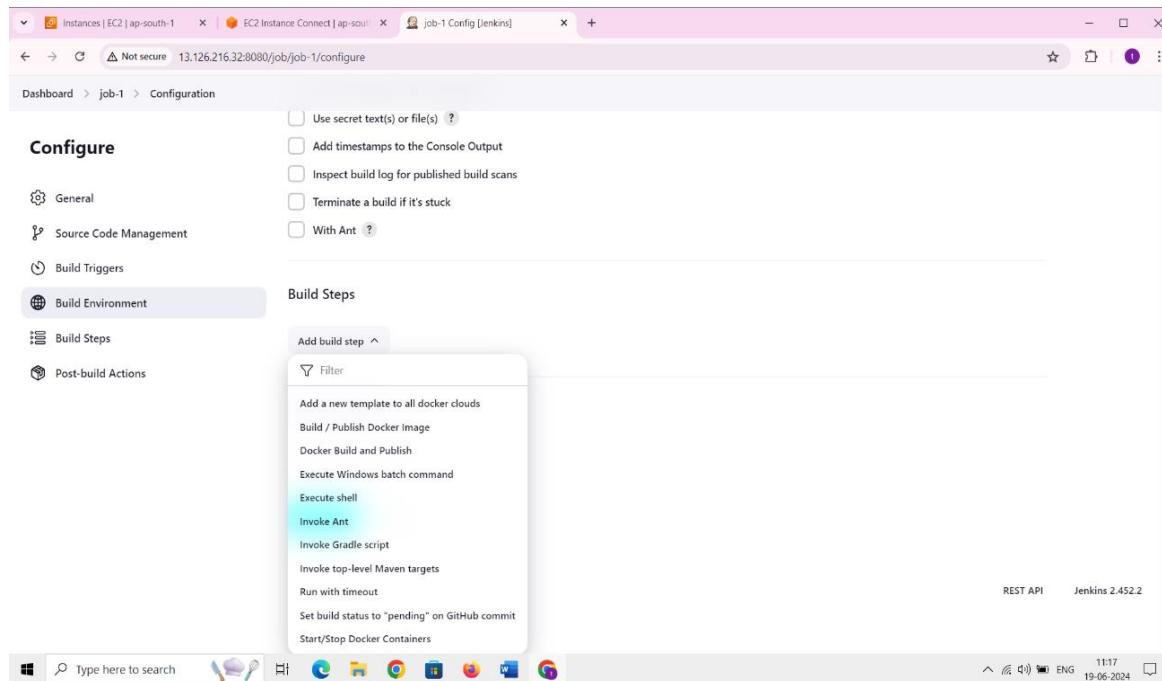
- ❖ Click on Save
- ❖ Go to Jenkins Dashboard
- ❖ Configure the Job1
- ❖ Click on Add build step



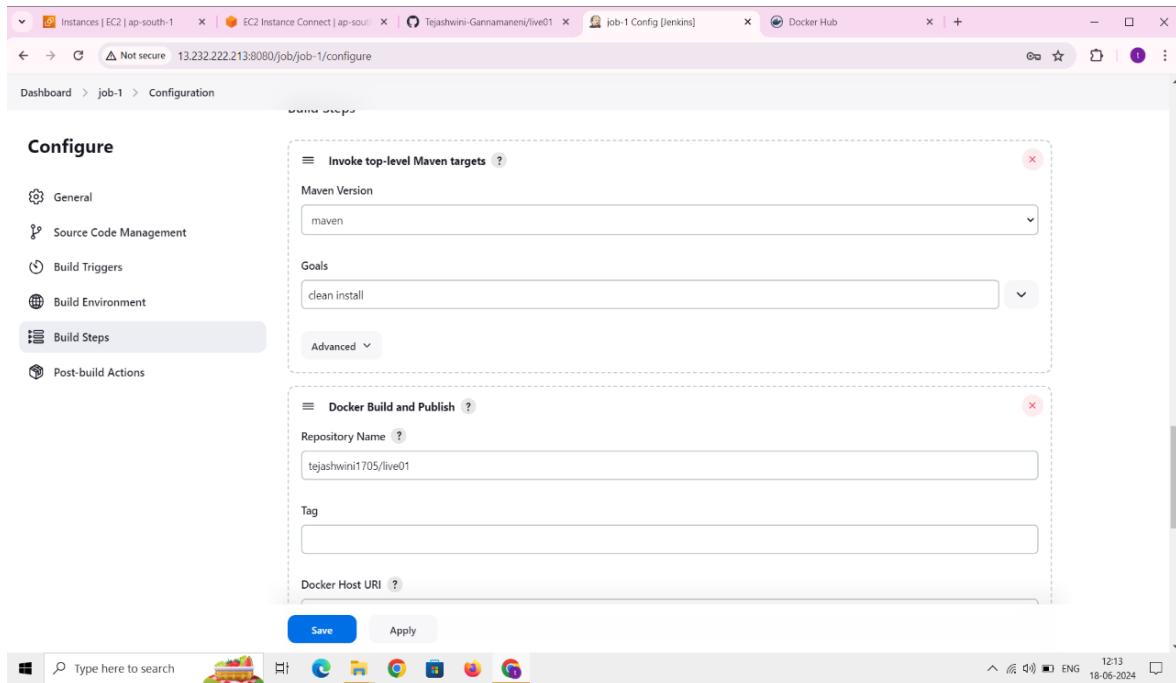
- ❖ Select Invoke top-level Maven targets
- ❖ Click on Maven version
- ❖ Select Maven
- ❖ Click on Goals
- ❖ We have to give clean install



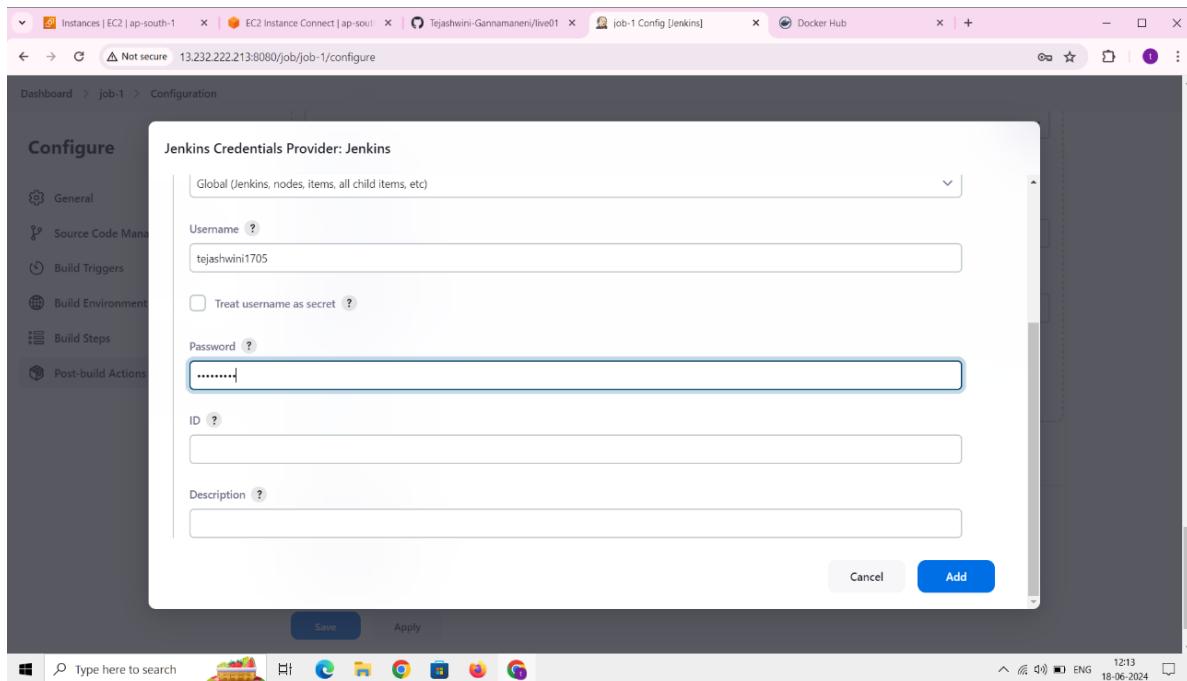
- ❖ Click on Add Build step



## ❖ Select Docker and Publish



- ❖ We have to give DockerHub username/image name at Repository name
- ❖ After we have to give Registry credentials to Docker
- ❖ Click on Add
- ❖ Select Jenkins



- ❖ We have to give username as DockerHub username

- ❖ We have to give password as DockerHub password
- ❖ Click on Add

The screenshot shows the Jenkins job configuration interface for a job named 'job-1'. In the 'Post-build Actions' section, there is a 'Docker Hub' action. The 'Registry credentials' dropdown is set to 'tejashwini1705/\*\*\*\*\*'. Below the dropdown, there is an 'Advanced' section and a 'Save' button.

- ❖ Click on Save
- ❖ In server we have to install Docker (yum install Docker -y)

The screenshot shows the AWS CloudShell terminal window. The user is running the command `yum install docker`. The output shows the package being installed along with its dependencies. Once completed, it displays the installed packages and their details.

```
[root@ip-172-31-5-107 ~]# yum install docker
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Resolving Dependencies
--> Running transaction check
--> Package docker.x86_64 0:20.10.25-1.amzn2.0.4 will be installed
--> Processing Dependency: containerd >= 1.3.2 for package: docker-20.10.25-1.amzn2.0.4.x86_64
--> Processing Dependency: libcgroup >= 0.40.rcl-5.15 for package: docker-20.10.25-1.amzn2.0.4.x86_64
--> Processing Dependency: runc >= 1.0.0 for package: docker-20.10.25-1.amzn2.0.4.x86_64
--> Processing Dependency: pigz for package: docker-20.10.25-1.amzn2.0.4.x86_64
--> Running transaction check
--> Package containerd.x86_64 0:1.7.11-1.amzn2.0.1 will be installed
--> Package libcgroup.x86_64 0:0.41-21.amzn2 will be installed
--> Package pigz.x86_64 0:2.3.4-1.amzn2.0.1 will be installed
--> Package runc.x86_64 0:1.1.1-1.amzn2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

Transaction Summary
  Install  4 Packages

Total download size: 3.6 kB
  00:00:00

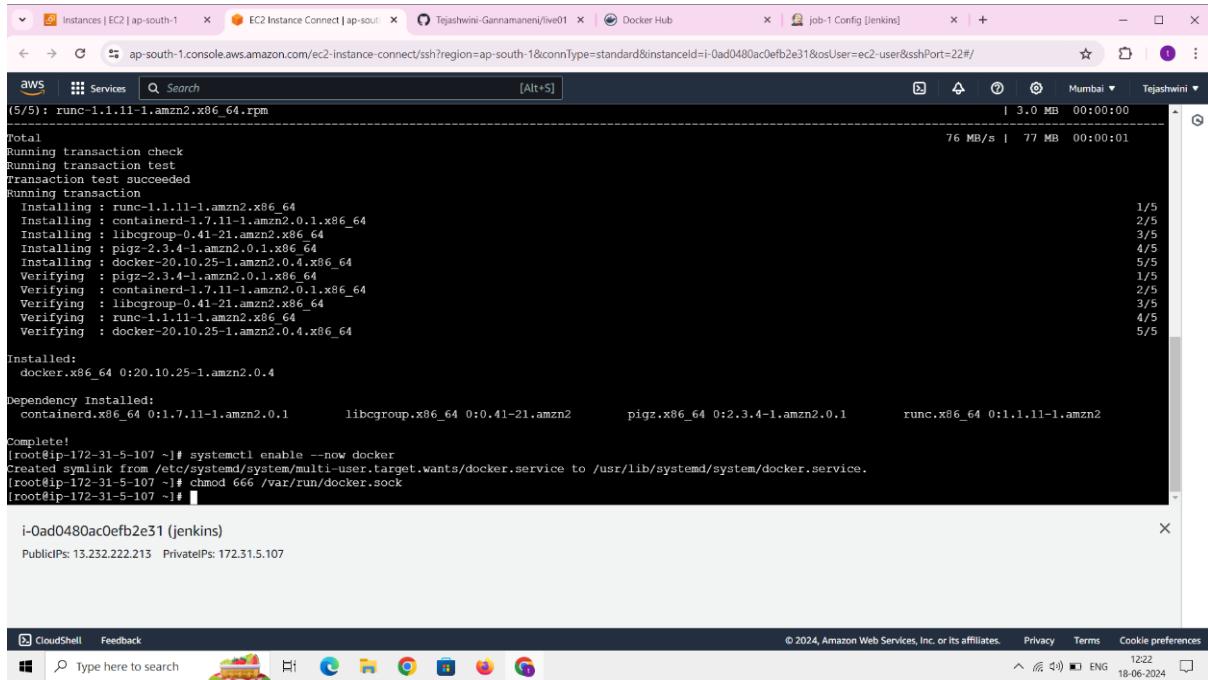
```

Package	Arch	Version	Repository	Size
docker	x86_64	20.10.25-1.amzn2.0.4	amzn2extra-docker	43 M
containerd	x86_64	1.7.11-1.amzn2.0.1	amzn2extra-docker	30 M
libcgroup	x86_64	0.41-21.amzn2	amzn2-core	66 k
pigz	x86_64	2.3.4-1.amzn2.0.1	amzn2-core	81 k
runc	x86_64	1.1.1-1.amzn2	amzn2extra-docker	3.0 M

i-0ad0480ac0efb2e31 (jenkins)  
PublicIPs: 13.232.222.213 PrivateIPs: 172.31.5.107

- ❖ First, we have to start Docker by using this command Systemctl start docker

- ❖ Give permissions for docker.sock by using command chmod, because Jenkins user hasn't enough permissions to access the docker socket to communicate with the engine.



```

Instances | EC2 | ap-south-1 x EC2 Instance Connect | ap-south-1 x Tejashwini-Gannamaneni/live01 x job-1 Config [Jenkins] x + 
← → ⌂ ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-south-1&connType=standard&instanceId=i-0ad0480ac0efb2e31&osUser=ec2-user&sshPort=22#
Docker Hub x Docker Hub [Alt+S]
Mumbai Tejashwini 
aws Services Search [Alt+S]
(5/5) : runc-1.1.11-1.amzn2.x86_64.rpm
Total 76 MB/s | 77 MB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : runc-1.1.11-1.amzn2.x86_64 1/5
  Installing : containerd-1.7.11-1.amzn2.0.1.x86_64 2/5
  Installing : libcgroup-0.41-21.amzn2.x86_64 3/5
  Installing : pigz-2.3.4-1.amzn2.0.1.x86_64 4/5
  Installing : docker-20.10.25-1.amzn2.0.4.x86_64 5/5
  Verifying : runc-1.1.11-1.amzn2.x86_64 1/5
  Verifying : containerd-1.7.11-1.amzn2.0.1.x86_64 2/5
  Verifying : libcgroup-0.41-21.amzn2.x86_64 3/5
  Verifying : pigz-2.3.4-1.amzn2.0.1.x86_64 4/5
  Verifying : docker-20.10.25-1.amzn2.0.4.x86_64 5/5
  Verifying : docker-20.10.25-1.amzn2.0.4.x86_64 1/5
  Verifying : docker-20.10.25-1.amzn2.0.4.x86_64 2/5
  Verifying : docker-20.10.25-1.amzn2.0.4.x86_64 3/5
  Verifying : docker-20.10.25-1.amzn2.0.4.x86_64 4/5
  Verifying : docker-20.10.25-1.amzn2.0.4.x86_64 5/5
Installed:
  docker.x86_64 0:20.10.25-1.amzn2.0.4

Dependency Installed:
  containerd.x86_64 0:1.7.11-1.amzn2.0.1           libcgroup.x86_64 0:0.41-21.amzn2.0.1           pigz.x86_64 0:2.3.4-1.amzn2.0.1           runc.x86_64 0:1.1.11-1.amzn2.0.1

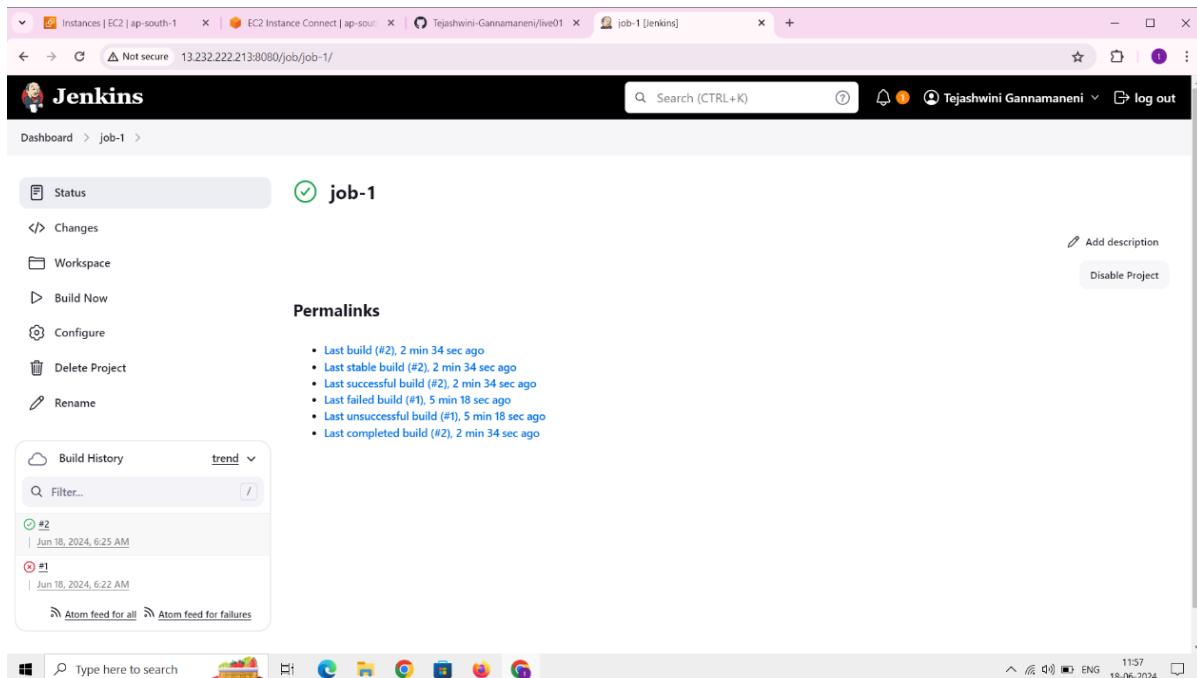
Complete!
[root@ip-172-31-5-107 ~]# systemctl enable --now docker
Created symlink from /etc/systemd/system/multi-user.target.wants/docker.service to /usr/lib/systemd/system/docker.service.
[root@ip-172-31-5-107 ~]# chmod 666 /var/run/docker.sock
[root@ip-172-31-5-107 ~]# 

i-0ad0480ac0efb2e31 (jenkins)
PublicIP: 13.232.222.213 PrivateIP: 172.31.5.107

```

The terminal window shows the installation of Docker components (runc, containerd, libcgroup, pigz, docker) and the creation of a symbolic link for docker.service. It also shows the chmod command being run on /var/run/docker.sock to change its permissions to 666. The Jenkins instance is identified as i-0ad0480ac0efb2e31.

- ❖ Go to Jenkins Dashboard
- ❖ Select the job1
- ❖ Click on Build now



**Jenkins**

Dashboard > job-1 >

Status: **job-1** (green circle with checkmark)

Changes, Workspace, Build Now, Configure, Delete Project, Rename

Add description, Disable Project

**Permalinks**

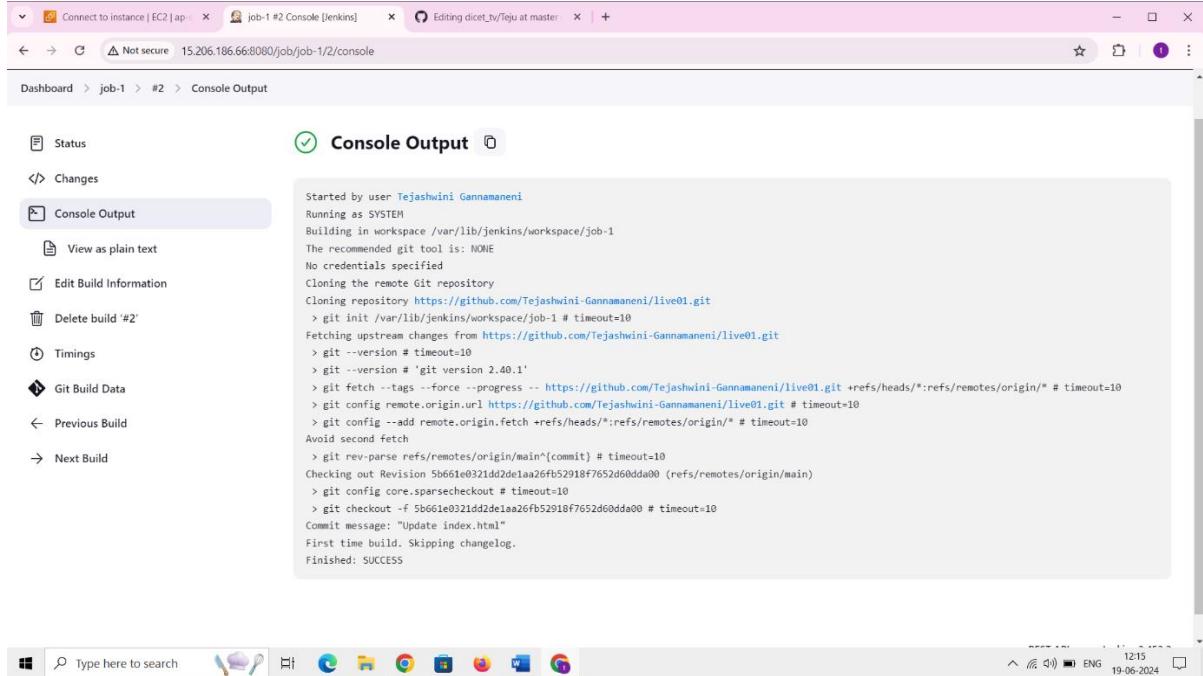
- Last build (#2), 2 min 34 sec ago
- Last stable build (#2), 2 min 34 sec ago
- Last successful build (#2), 2 min 34 sec ago
- Last failed build (#1), 5 min 18 sec ago
- Last unsuccessful build (#1), 5 min 18 sec ago
- Last completed build (#2), 2 min 34 sec ago

**Build History** (trend dropdown, Filter...)

Build #	Date
#2	Jun 18, 2024, 6:25 AM
#1	Jun 18, 2024, 6:22 AM

Atom feed for all Atom feed for failures

Type here to search

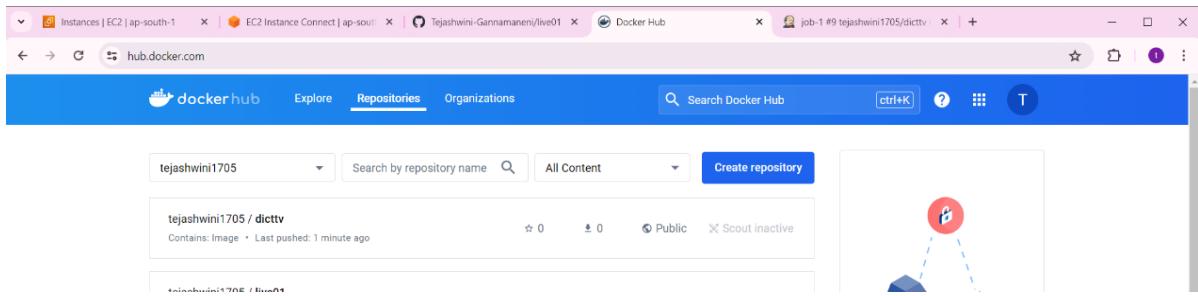


```

Started by user Tejashwini Gannamaneni
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/job-1
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository https://github.com/Tejashwini-Gannamaneni/live01.git
> git init /var/lib/jenkins/workspace/job-1 # timeout=10
Fetching upstream changes from https://github.com/Tejashwini-Gannamaneni/live01.git
> git --version # timeout=10
> git -version # 'git version 2.40.1'
> git fetch --tags --force --progress -- https://github.com/Tejashwini-Gannamaneni/live01.git +refs/heads/*:refs/remotes/origin/*
> git config remote.origin.url https://github.com/Tejashwini-Gannamaneni/live01.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/*
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/*
Avoid second fetch
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 5b661e0321dd2de1aa26fb52918f7652d60dda00 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 5b661e0321dd2de1aa26fb52918f7652d60dda00 # timeout=10
Commit message: "Update index.html"
First time build. Skipping changelog.
Finished: SUCCESS

```

- ❖ Build is Success
- ❖ Open DockerHub Account

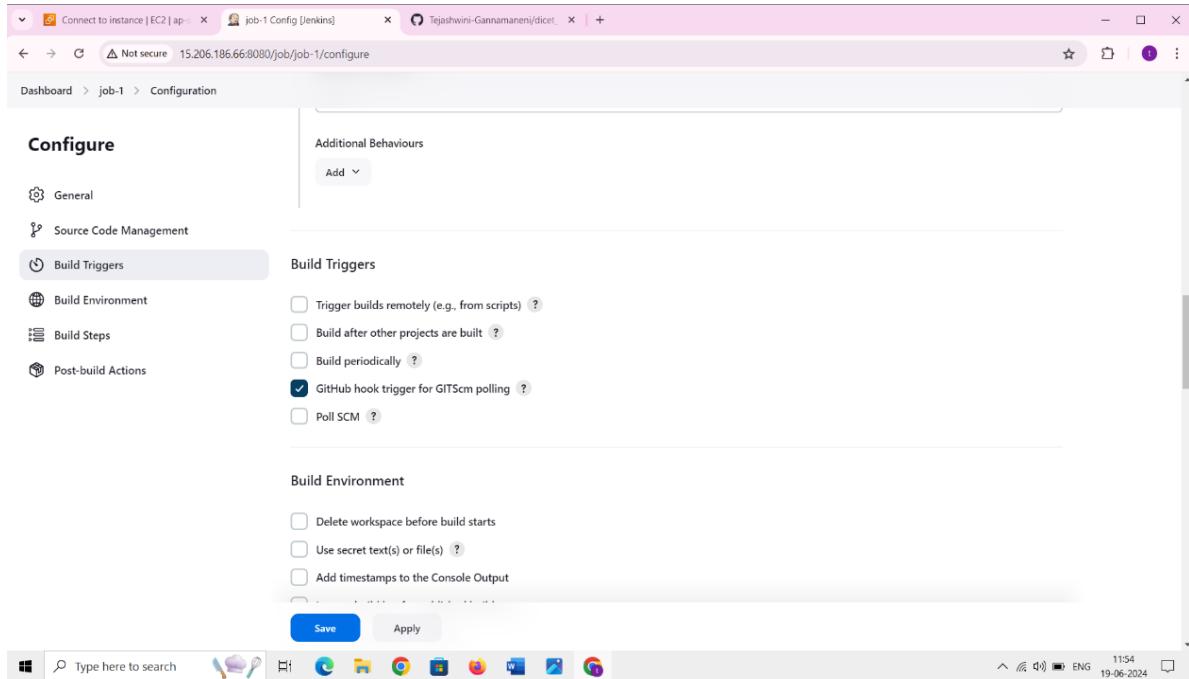


- ❖ After getting Build Success automatically image push into my DockerHub account

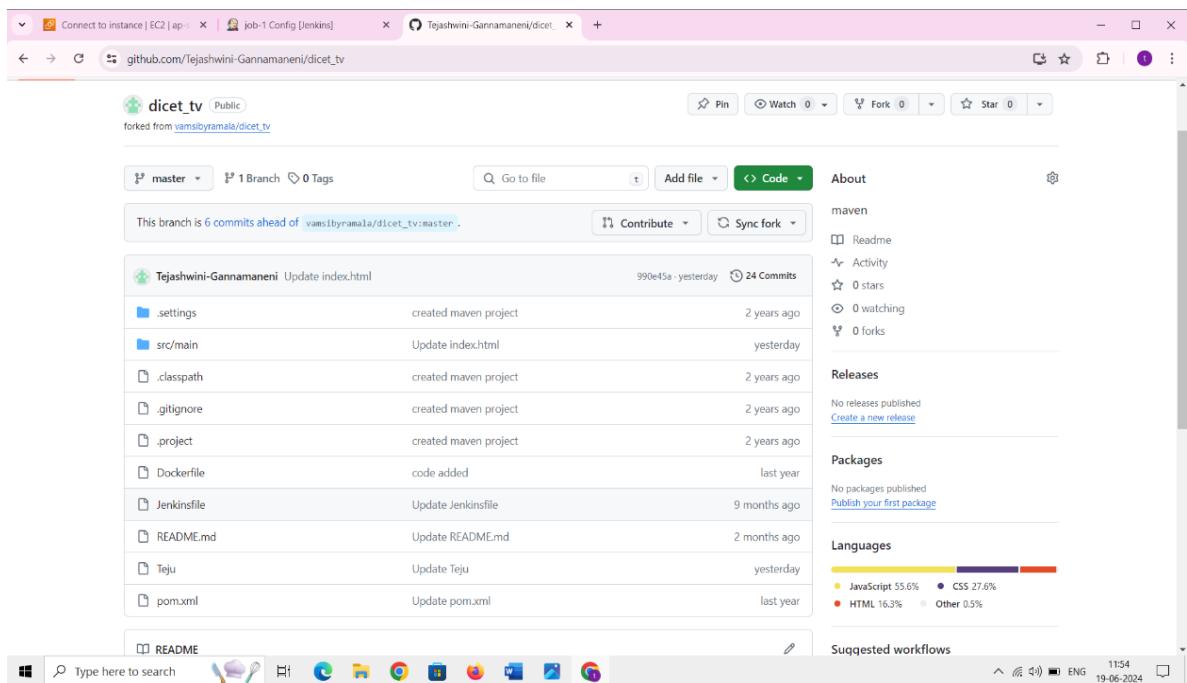
## Configuration of Build Triggers

- Open Jenkins Dashboard
- Select Job1
- Click on Configure

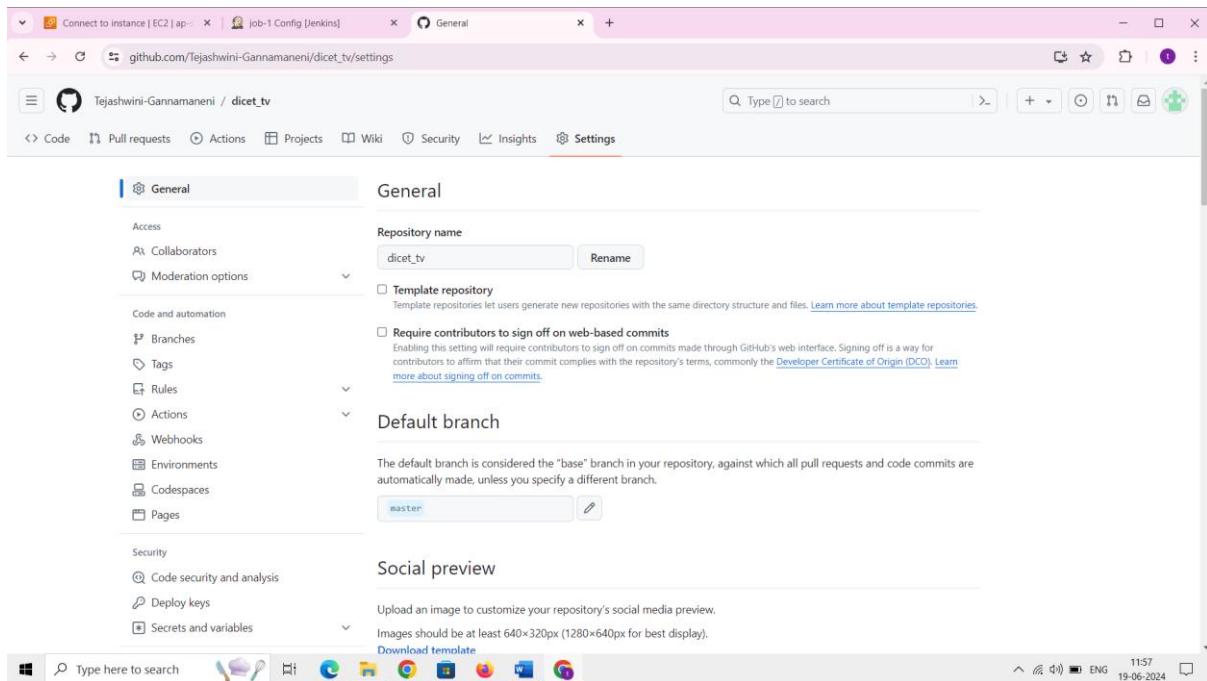
- Now we have to Select Build Triggers
- Click on GitHub hook trigger for GITscm polling



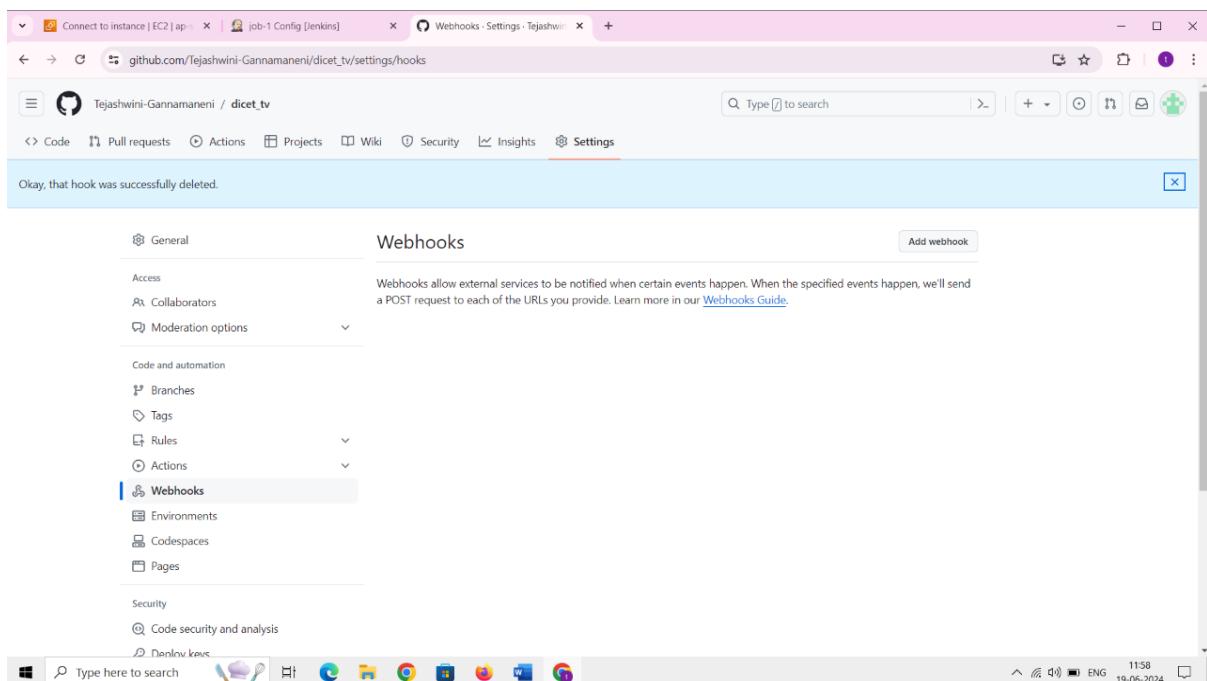
- Click on save
- Open GitHub Account
- Select Repository(dicet\_tv)



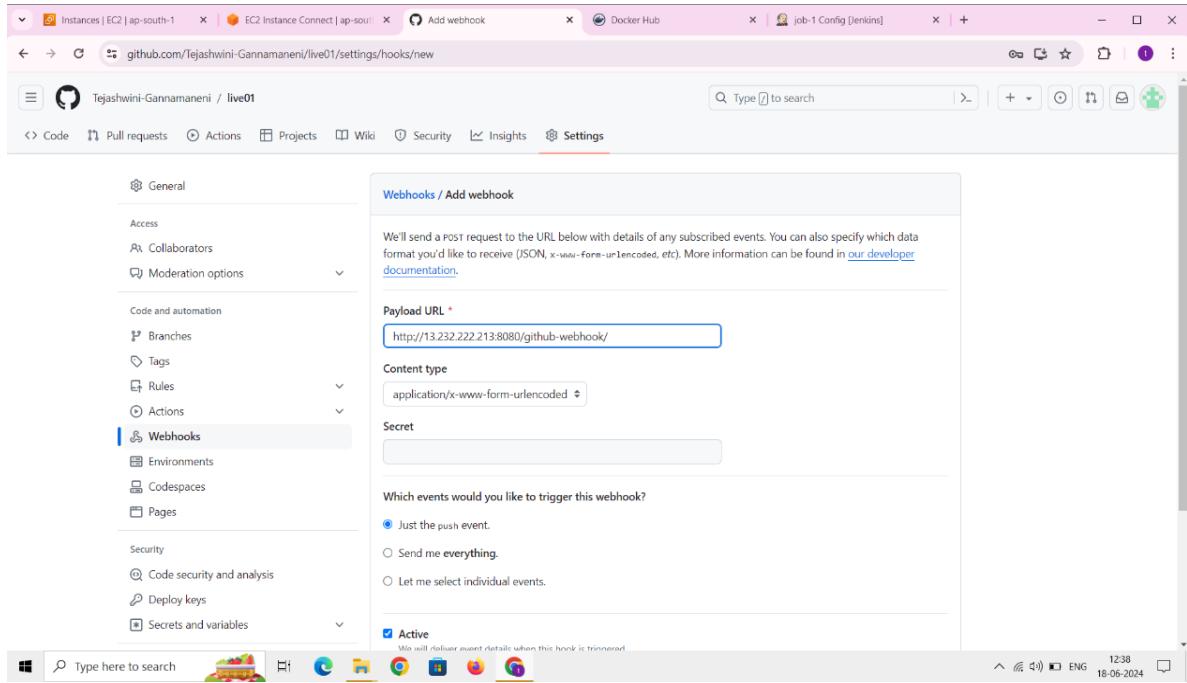
- Click on settings



- Select Webhooks



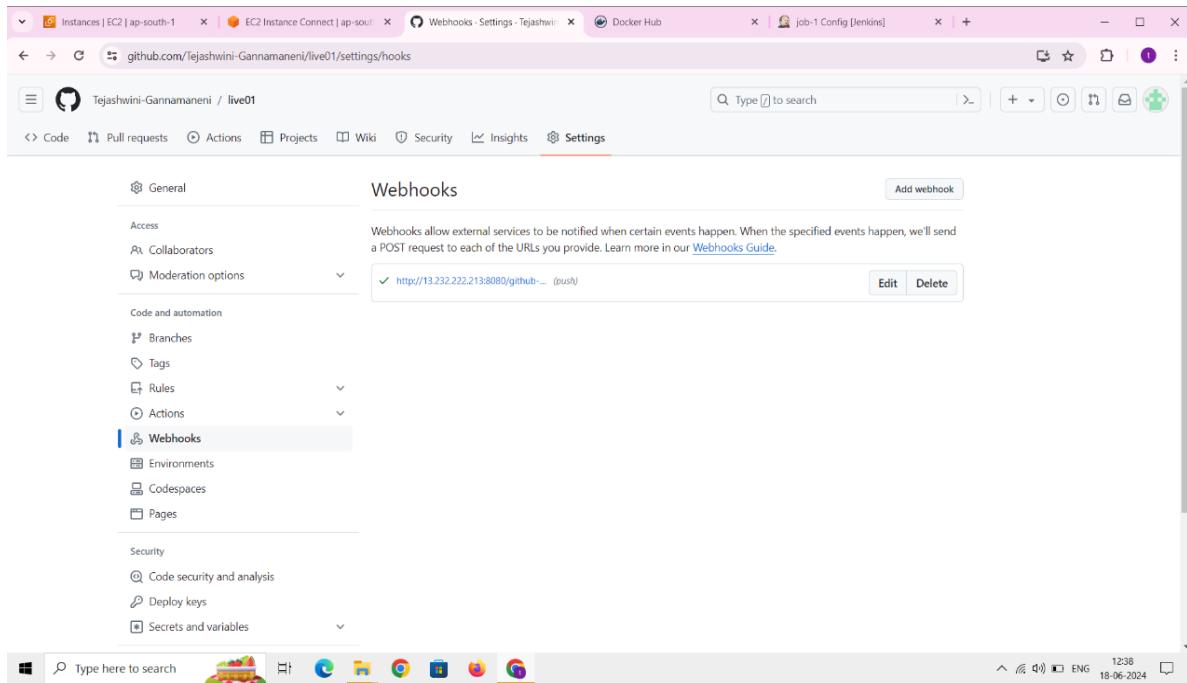
- Click on Add Webhook
- We have to give Playload URL



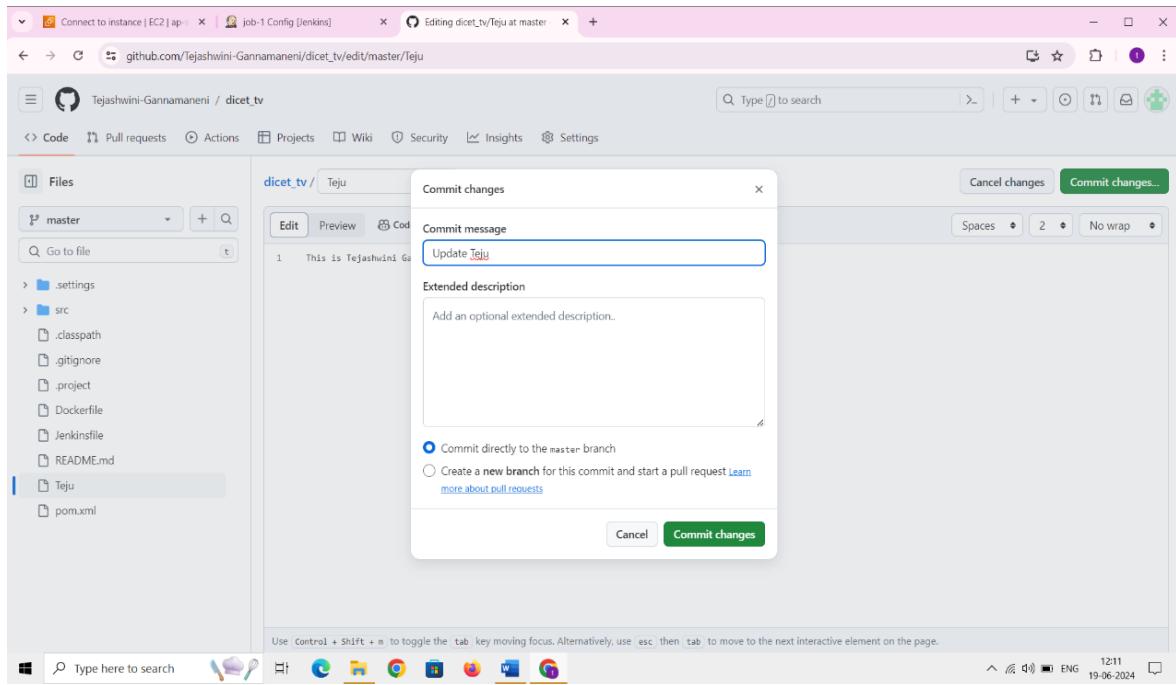
- Payload URL:

[http:// Jenkins server PublicIPs:8080/github-webhook/](http://13.232.222.213:8080/github-webhook/)  
[\(http://13.232.222.213:8080/github-webhook/\)](http://13.232.222.213:8080/github-webhook/)

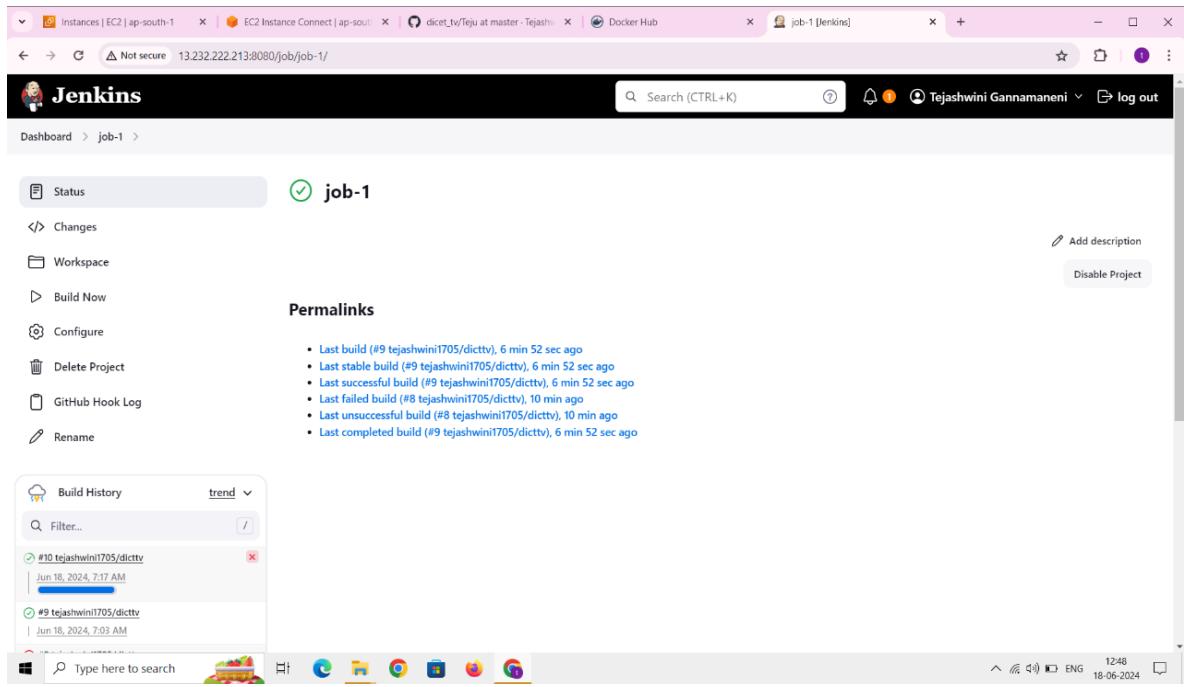
- Click on Add Webhook



- Now we can do some changes in code (GitHub)
- After Doing changes



- Click on commit changes
- It will Build the job1 automatically in Jenkins



## Deployment of An Application

- Click on Configure (job1)
- Click on Add build step
- Select Execute Shell

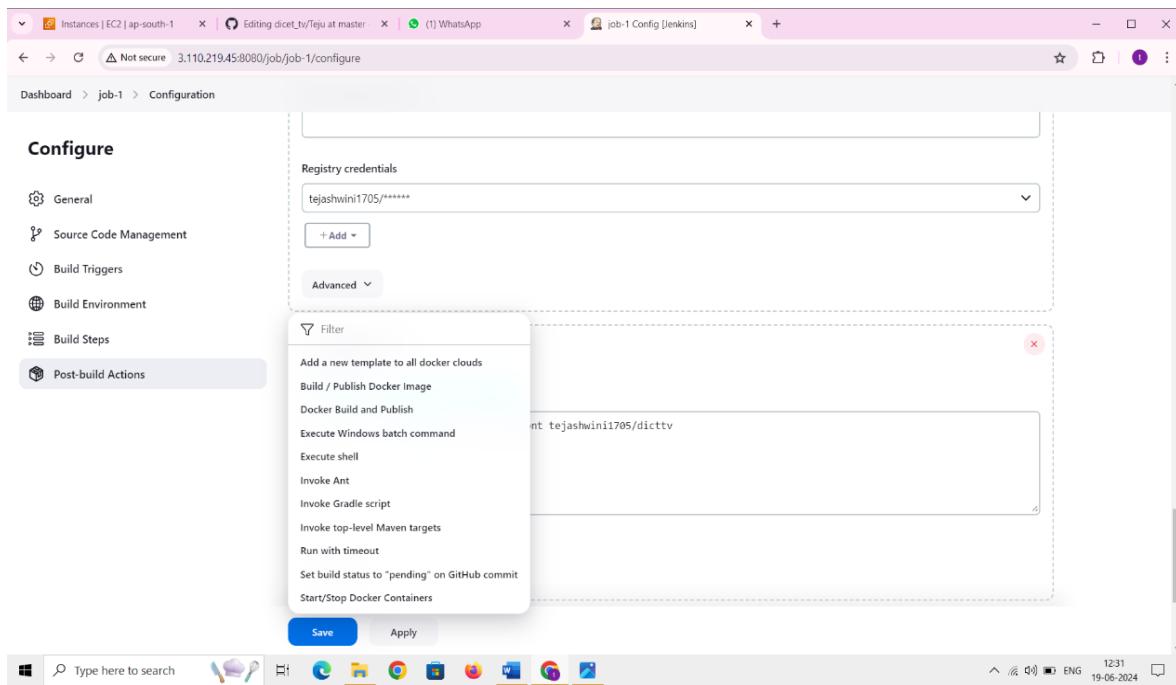
We have to select this After maven

- docker rm -f my-cont

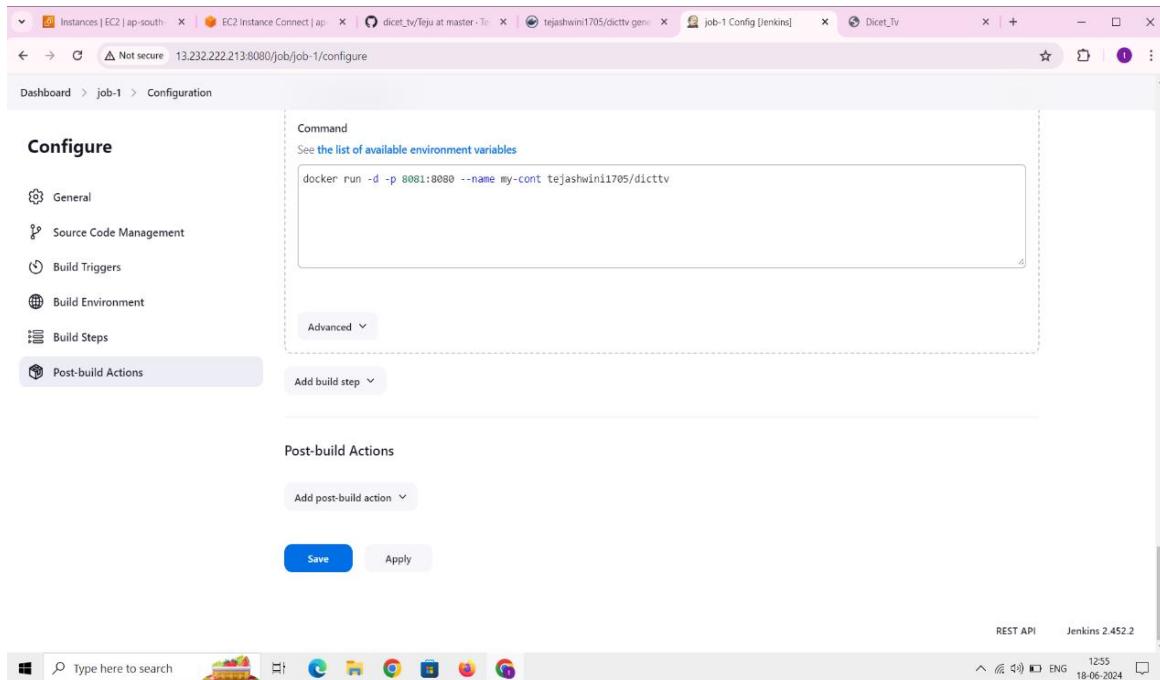
When we relaunch the application, it will remove previous container.

- docker rmi tejashwini1705/dicet\_tv

When we relaunch the application, it will remove previous image.



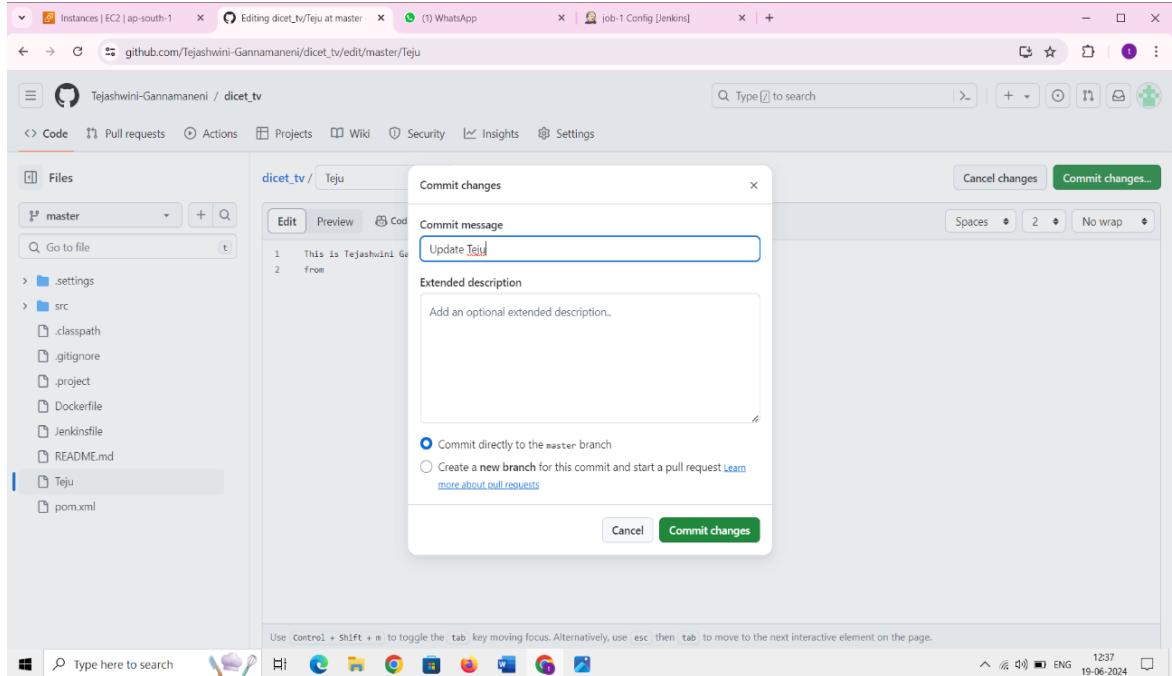
- Select Execute shell
- We have to Select this after Docker



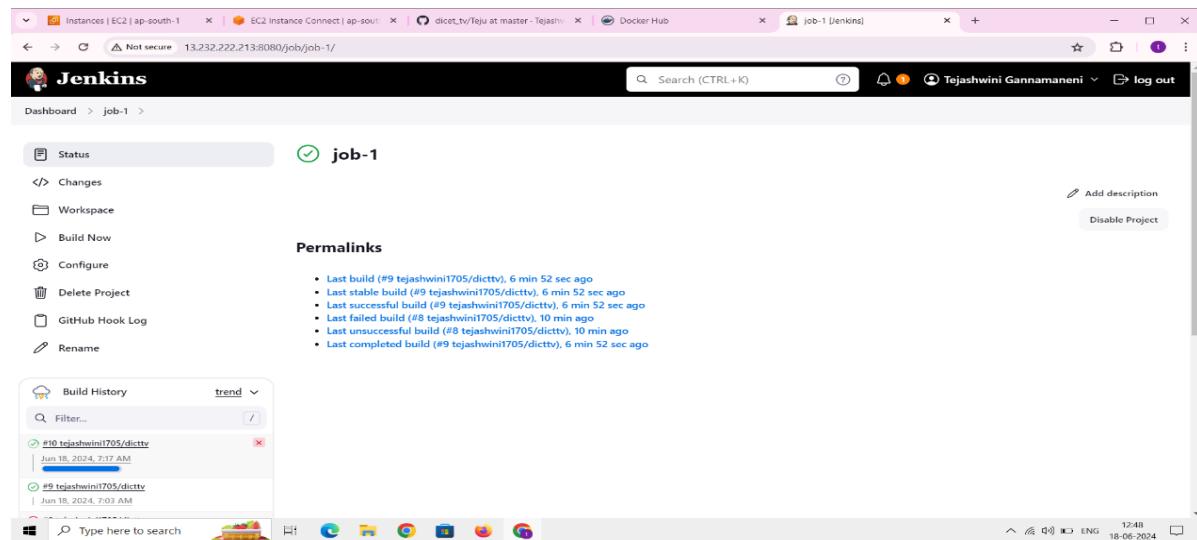
- We have to write a command

```
docker run -d -p 8081:8080 -name my-cont tejashwini1705/dicctv
-name=we have to give any name (container name)
Tejashwini1705/dicctv=docker image
```

- Click on save
- Now we do some changes in code (github)



- Click on commit changes
- It will build application automatically



- Copy the PublicIPs from Jenkins server

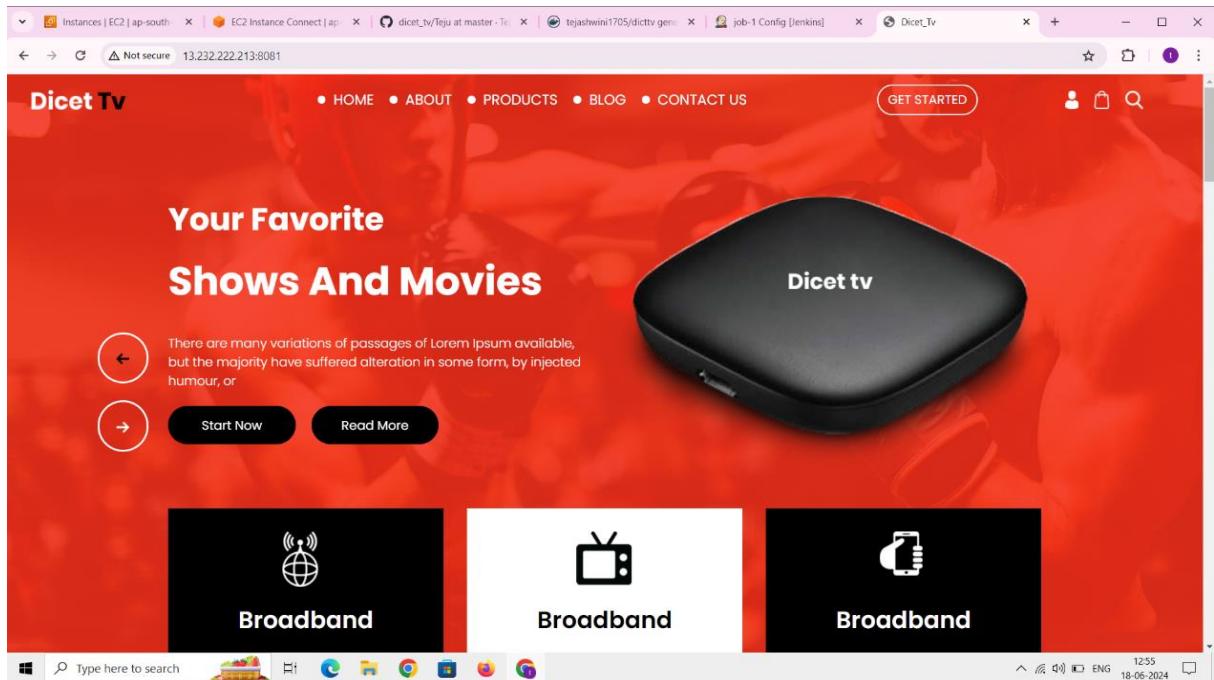
```

Last login: Wed Jun 19 05:41:50 2024 from ec2-13-233-177-5.ap-south-1.compute.amazonaws.com
[ec2-user@ip-172-31-5-107 ~]$
```

Copy      Ctrl+C  
Copy link to highlight  
Go to 3.110.219.45  
Print...      Ctrl+P  
Translate selection to English  
Open in reading mode

i-0ad0480ac0...  
PublicIPs: 3.110.219.45

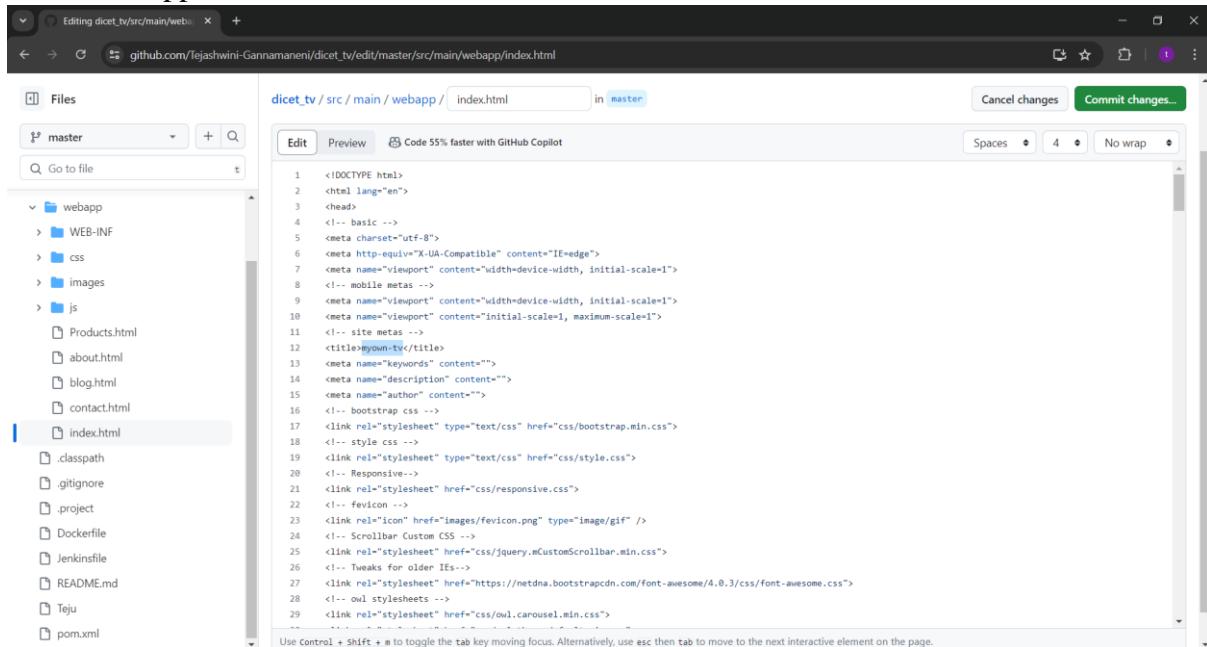
- Paste in google PublicIPs:8081
- Our application will be live



- This is our application
- Now I have to edit title
- In my present application
- Title=dicet-tv
- I want to update my application as

➤ Title=myown-tv

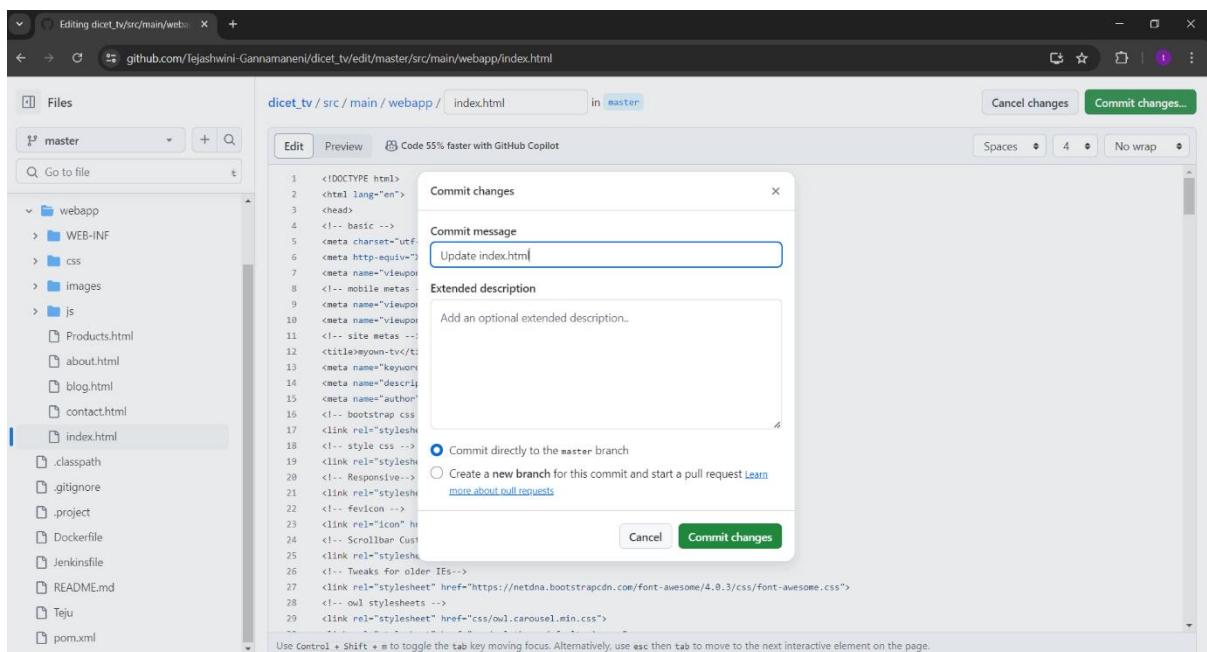
- Open GitHub Account
- Click on your repository (dicet-tv)
- Edit your code at
- src
- main
- webapps



```

<!DOCTYPE html>
<html lang="en">
<head>
<!-- basic -->
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<!-- mobile metas -->
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1">
<meta name="viewport" content="initial-scale=1, maximum-scale=1">
<!-- site metas -->
<title>myown-tv</title>
<meta name="keywords" content="">
<meta name="description" content="">
<meta name="author" content="">
<!-- bootstrap css -->
<link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
<!-- style.css -->
<link rel="stylesheet" type="text/css" href="css/style.css">
<!-- Responsive-->
<link rel="stylesheet" href="css/responsive.css">
<!-- favicon -->
<link rel="icon" href="images/favicon.png" type="image/gif" />
<!-- Scrollbar Custom CSS -->
<link rel="stylesheet" href="css/jquery.mCustomScrollbar.min.css">
<!-- Tweaks for older IEs-->
<link rel="stylesheet" href="https://netdna.bootstrapcdn.com/font-awesome/4.0.3/css/font-awesome.css">
<!-- owl stylesheets -->
<link rel="stylesheet" href="css/owl.carousel.min.css">

```



- click on commit changes
- Automatically it will build the job 1

- Refresh the page
- Automatically changes has done
- Updated Application

