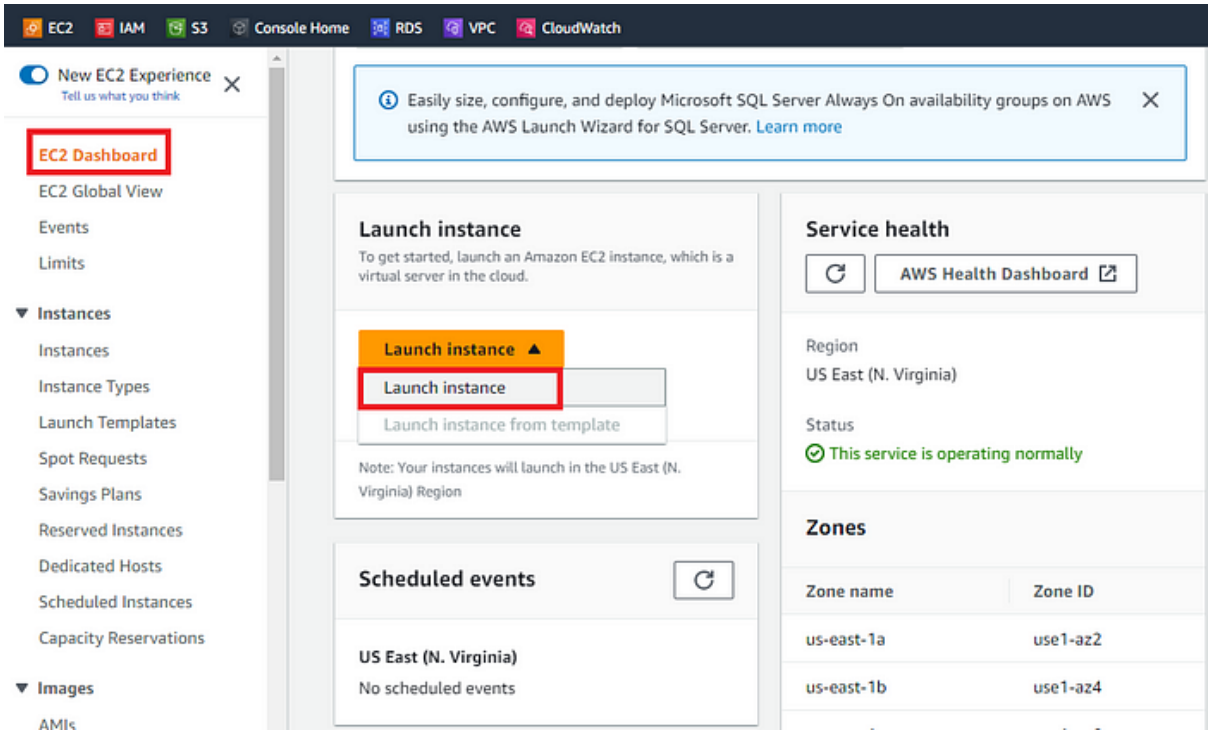
Task 10 -jenkins (projects and users)

Launch jenkins and explore creating projects and users:

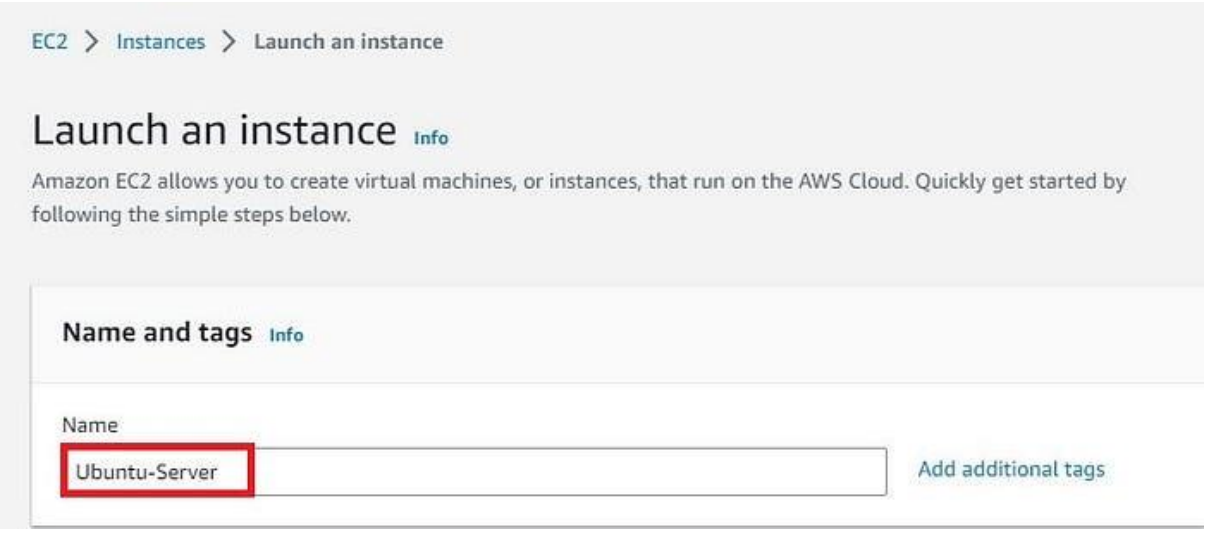
https://github.com/yasminjeelani/newcicd

Step 1: Create an EC2 instance

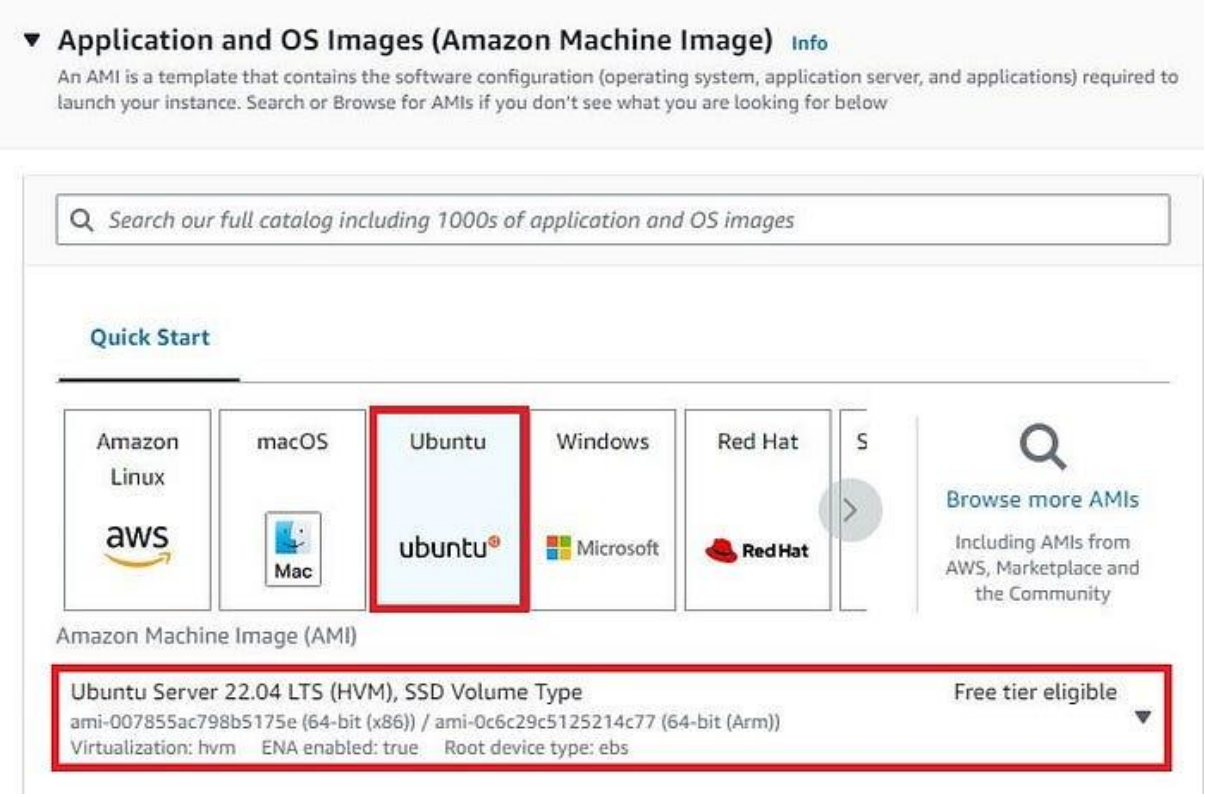
Log in to the Amazon management console, open EC2 Dashboard, click on the Launch Instance drop-down list, and click on Launch Instance as shown below:



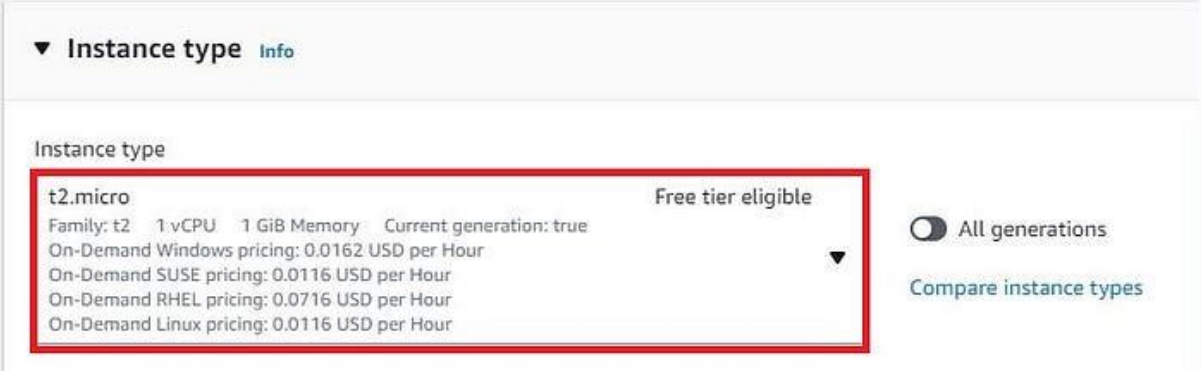
Once the **Launch an instance** window opens, provide the name of your EC2 Instance:



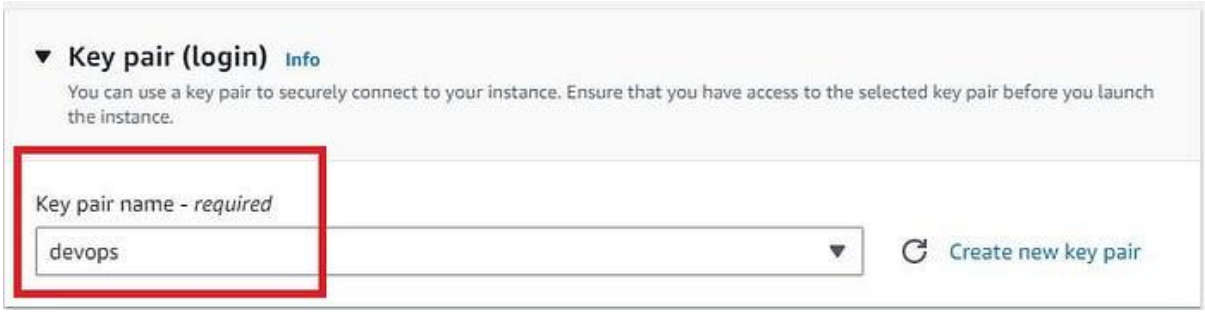
Let's choose Ubuntu Server 20.04 which is free tier eligible.



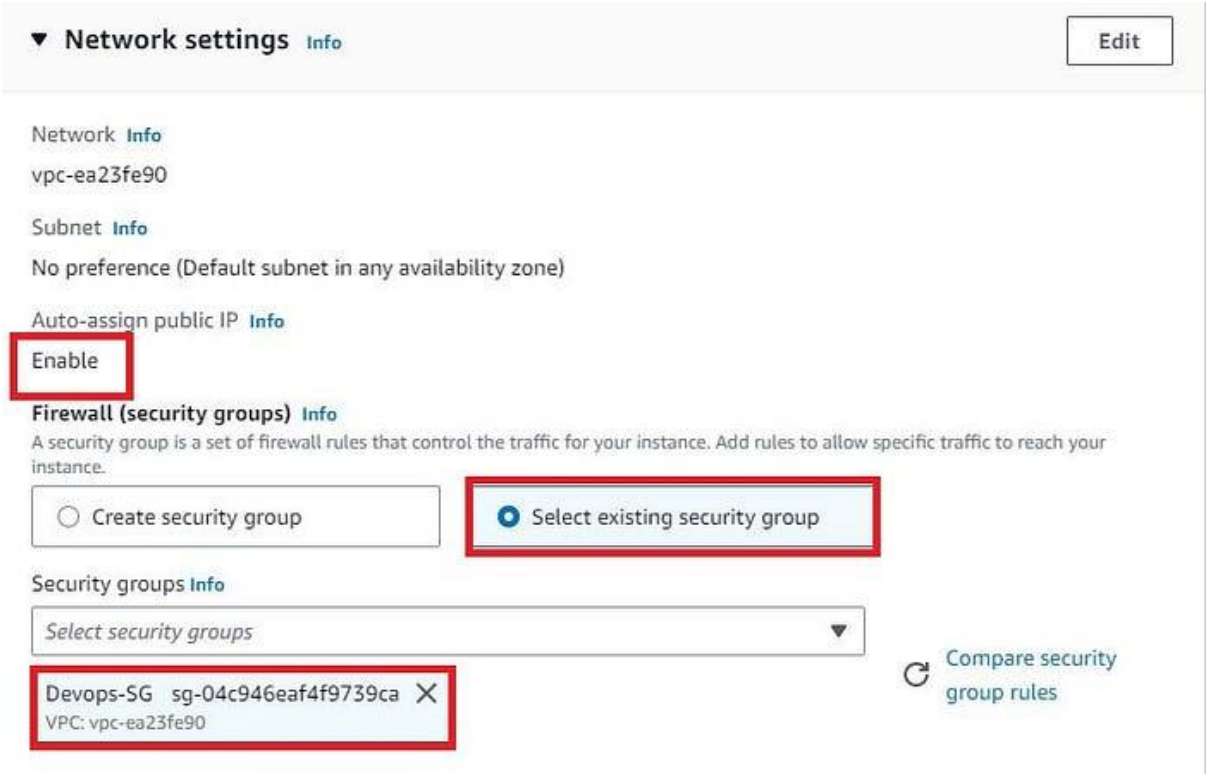
Now choose an **Instance Typ**e. Here you can select the type of machine, number of vCPUs, and memory that you want to have. Select t2.micro which is free-tier eligible.



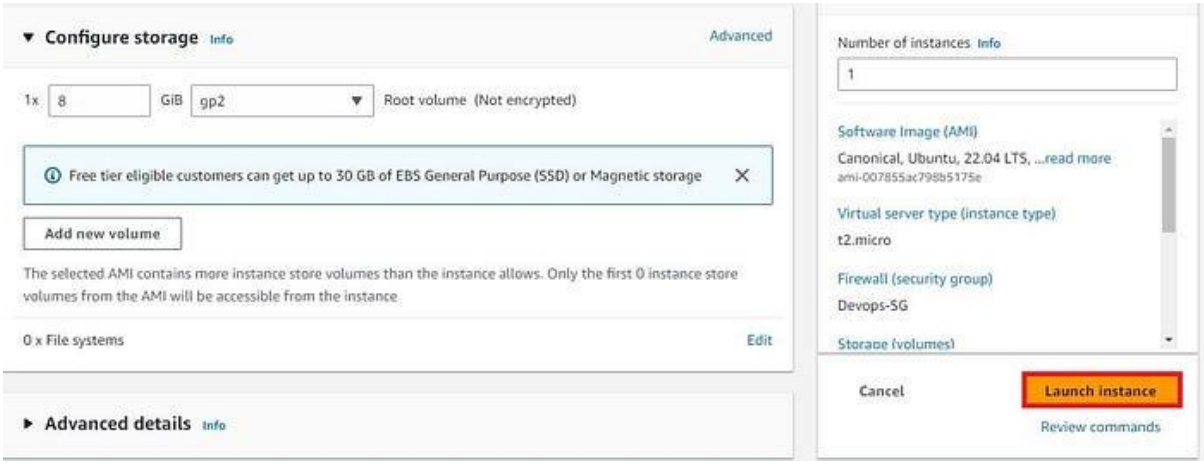
For this demo, I will select an already existing key pair. You can create new ones in case you don't have any existing ones.



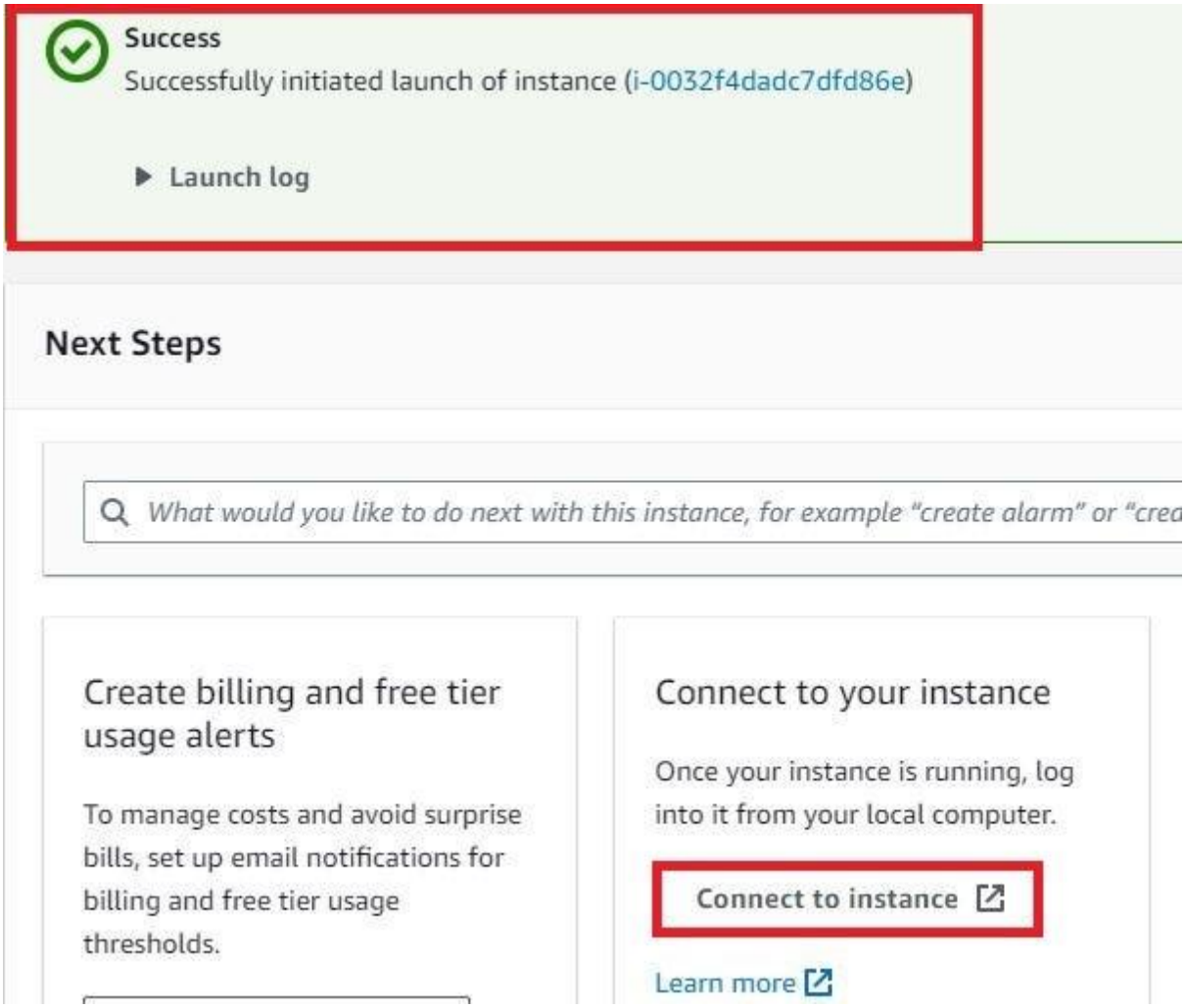
Now under **Network Settings**, Choose the default VPC with Auto-assign public IP in enable mode. Choose an existing **security group** that we have been using in all our previous projects with port 8080 open apart from SSH.



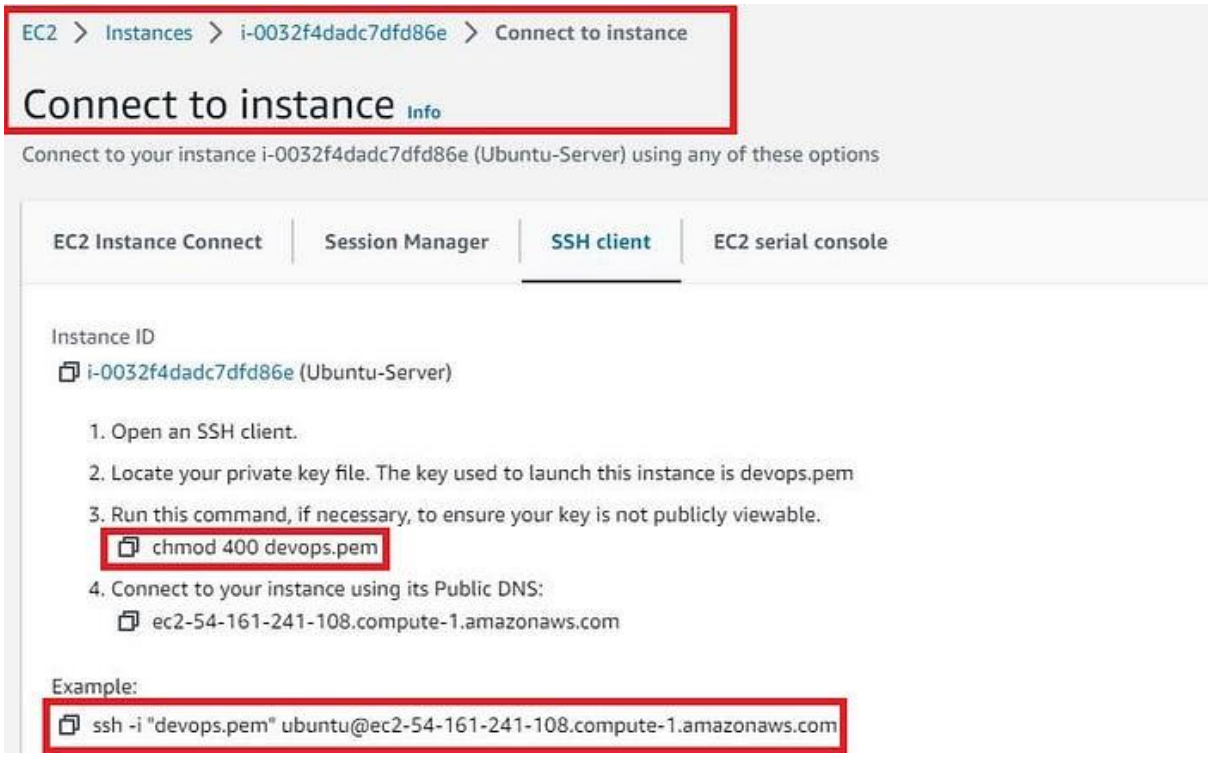
Rest of the settings we will keep them at default and go ahead and click on **Launch Instance**



On the next screen you can see a success message after the successful creation of the EC2 instance, click on Connect to instance button:



Now **connect to instance** wizard will open, go to SSH client tab and copy the provided chmod and SSH command:



Step 2: Install Jenkins on Ubuntu EC2 Machine

If you have been following my blog you should have seen that we have been installing Jenkins either from the official Jenkins website or using Jenkins on docker containers. Here we will install Jenkins using a script that basically has the same steps which we can find on the Jenkins official website.

```
# Update package lists

sudo apt-get update

# Install Java 11

sudo apt-get install -y openjdk-11-jdk

# Download Jenkins key and add it to system

sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \

  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key

echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \

  https://pkg.jenkins.io/debian-stable binary/ | sudo tee \

  /etc/apt/sources.list.d/jenkins.list > /dev/null

sudo apt-get update

sudo apt-get install Jenkins
```

(or)

```
sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \
  https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
  /etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt-get update
sudo apt-get install jenkins
```

Run above 4 commands one by one.

```
# Update package lists again to include Jenkins and install it
```

```
sudo apt-get update
```

```
sudo apt-get install -y jenkins
```

```
# Verify Jenkins is installed and working
```

```
sudo systemctl status Jenkins
```

Output: (this is yasmine output file)

```
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-32-49:~$ sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \
  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \
  https://pkg.jenkins.io/debian-stable binary/ | sudo tee \
  /etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt-get update
sudo apt-get install jenkins
--2024-06-20 11:06:21--  https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key
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: 3175 (3.1K) [application/pgp-keys]
Saving to: '/usr/share/keyrings/jenkins-keyring.asc'

/usr/share/keyrings/jenkins-keyring.as 100%[===================================================================>]   3.10K  --.-KB/s    in 0s

2024-06-20 11:06:21 (44.2 MB/s) - '/usr/share/keyrings/jenkins-keyring.asc' saved [3175/3175]

Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Ign:4 https://pkg.jenkins.io/debian-stable binary/ InRelease
Get:5 https://pkg.jenkins.io/debian-stable binary/ Release [2044 B]
Get:6 https://pkg.jenkins.io/debian-stable binary/ Release.gpg [833 B]
Hit:7 http://security.ubuntu.com/ubuntu jammy-security InRelease
Get:8 https://pkg.jenkins.io/debian-stable binary/ Packages [27.1 kB]
Fetched 30.0 kB in 1s (55.3 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... 50%
```

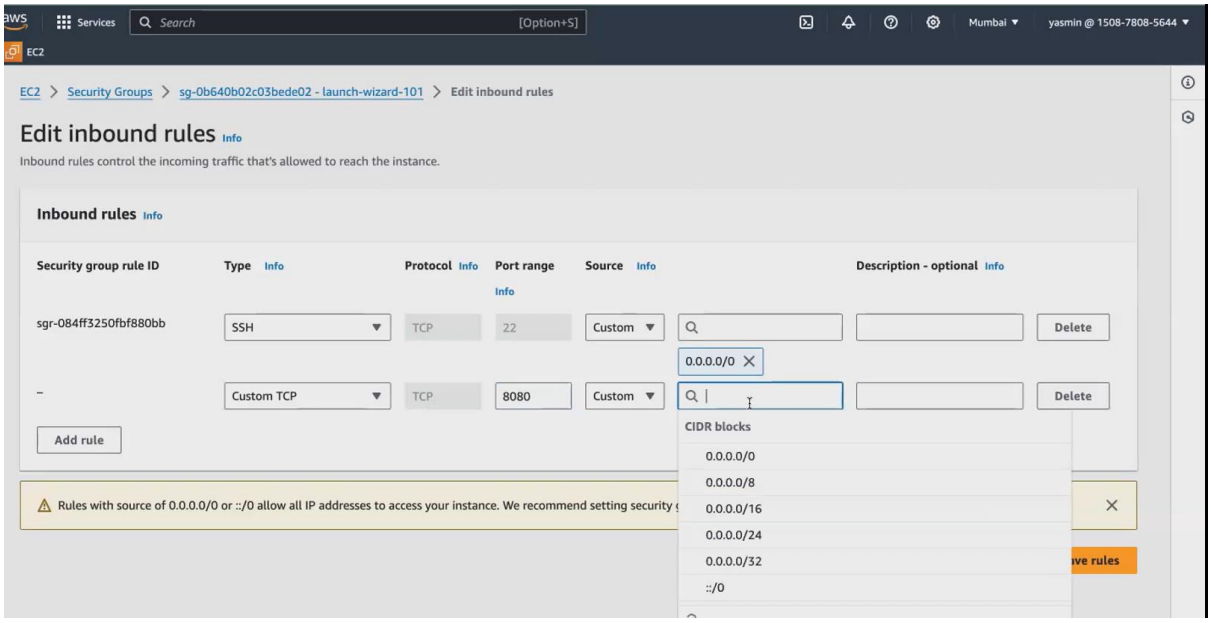
i-06bc228cbe511814b (jenkins-demo)

PublicIPs: 3.110.186.104 PrivateIPs: 172.31.32.49

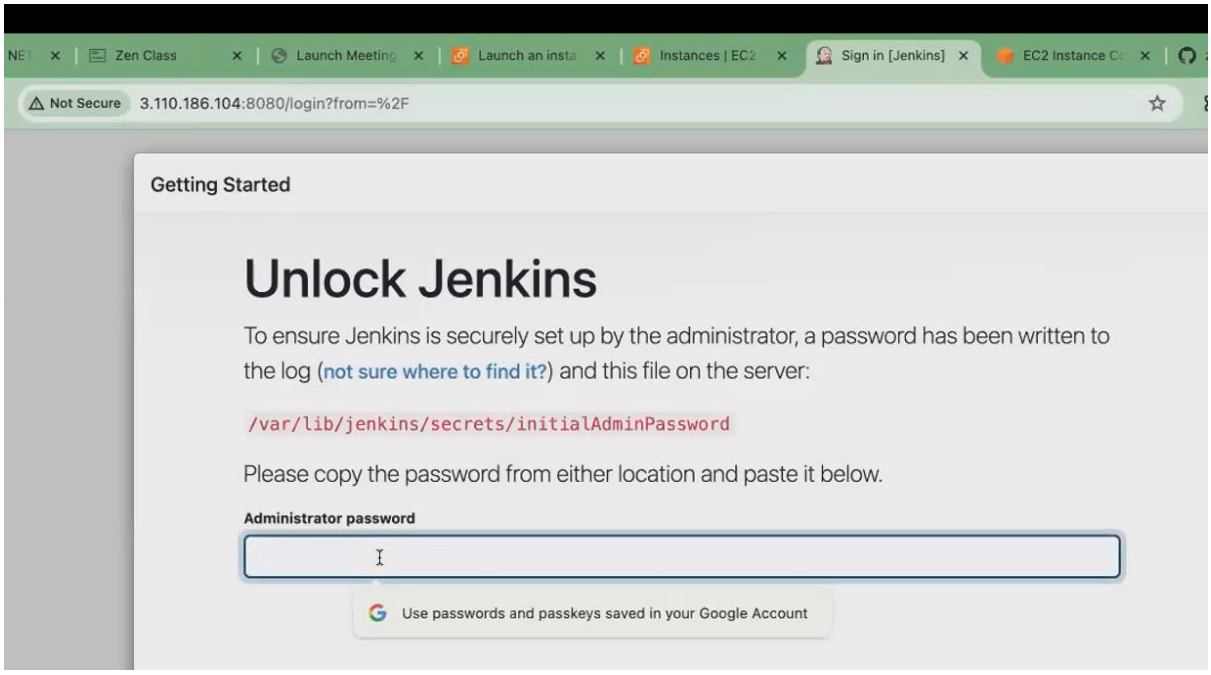
Let's now verify the status of Jenkins:

```
ubuntu@ubuntu-server:~$ systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
     Loaded: loaded (/lib/systemd/system/jenkins.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2023-05-19 14:14:00 UTC; 2min 43s ago
   Main PID: 5239 (java)
      Tasks: 36 (limit: 1141)
     Memory: 309.0M
        CPU: 44.737s
     CGroup: /system.slice/jenkins.service
             └─5239 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenk

May 19 14:13:25 ubuntu-server jenkins[5239]: 0cabca489c83489aa90284fed7270f9c
May 19 14:13:25 ubuntu-server jenkins[5239]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
May 19 14:13:25 ubuntu-server jenkins[5239]: *************************************************************
May 19 14:13:25 ubuntu-server jenkins[5239]: *************************************************************
May 19 14:13:25 ubuntu-server jenkins[5239]: *************************************************************
May 19 14:13:59 ubuntu-server jenkins[5239]: 2023-05-19 14:13:59.996+0000 [id=29]	INFO	jenkins.InitReac>
May 19 14:14:00 ubuntu-server jenkins[5239]: 2023-05-19 14:14:00.061+0000 [id=22]	INFO	hudson.lifecycle>
May 19 14:14:00 ubuntu-server systemd[1]: Started Jenkins Continuous Integration Server.
May 19 14:14:00 ubuntu-server jenkins[5239]: 2023-05-19 14:14:00.166+0000 [id=44]	INFO	h.m.DownloadServ>
May 19 14:14:00 ubuntu-server jenkins[5239]: 2023-05-19 14:14:00.169+0000 [id=44]	INFO	hudson.util.Retr>
ubuntu@ubuntu-server:~$
```

Lets now access the Jenkins Server using the public IP of our Ubuntu EC2 instance on port 8080 and you should see something like the below:



Open Jenkins in browser:



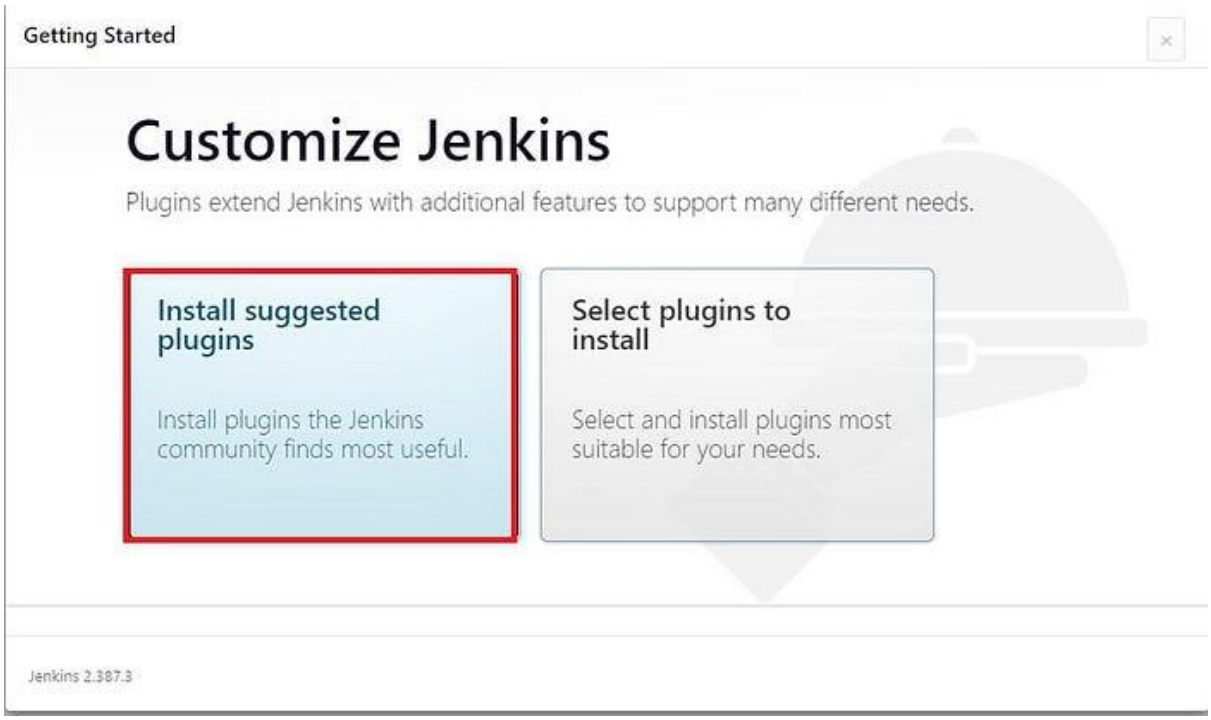
Get the password ,run the below command:

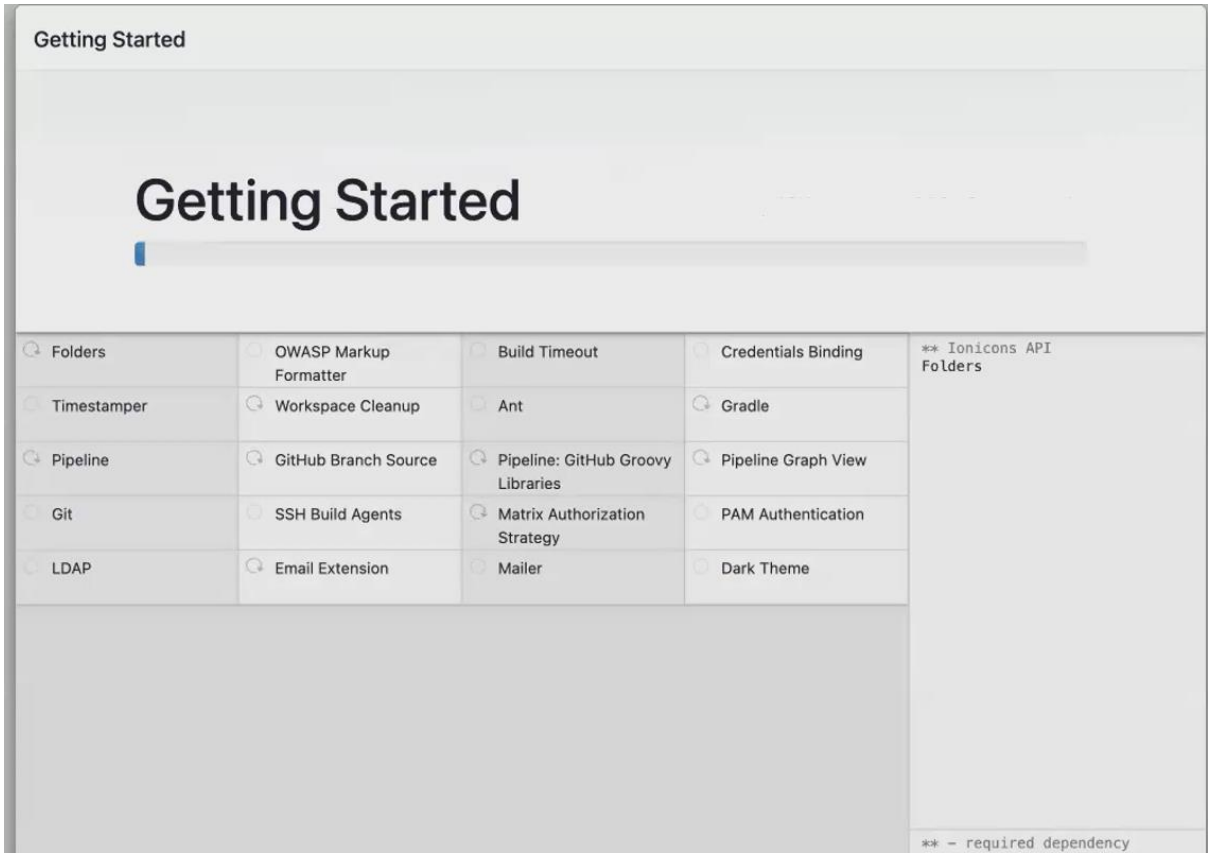
sudo cat /var/lib/Jenkins/initialAdminPassword

copy the password (ebfa701……) in administrator password

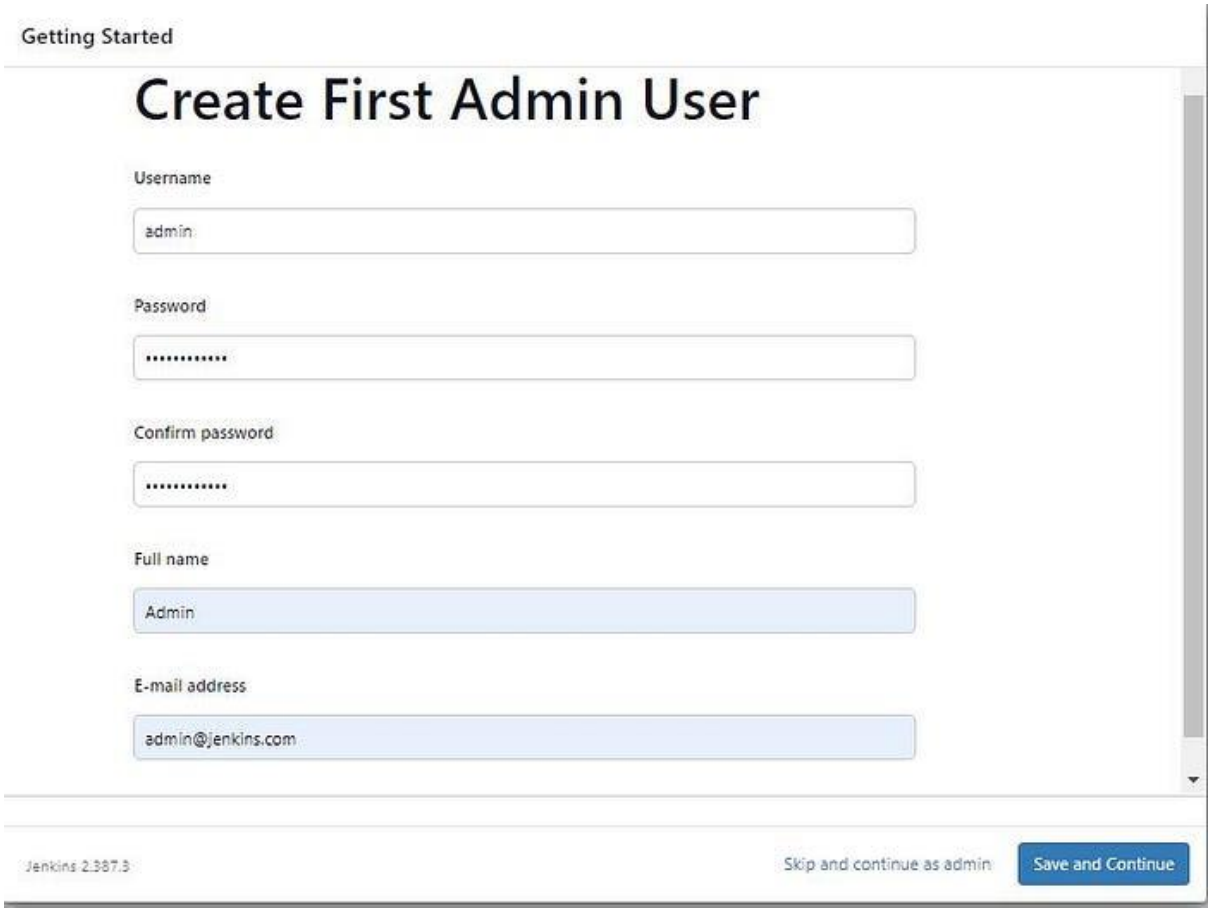
Navigate to /var/lib/jenkins/secrets/initialAdminPassword and copy the password and unlock Jenkins.

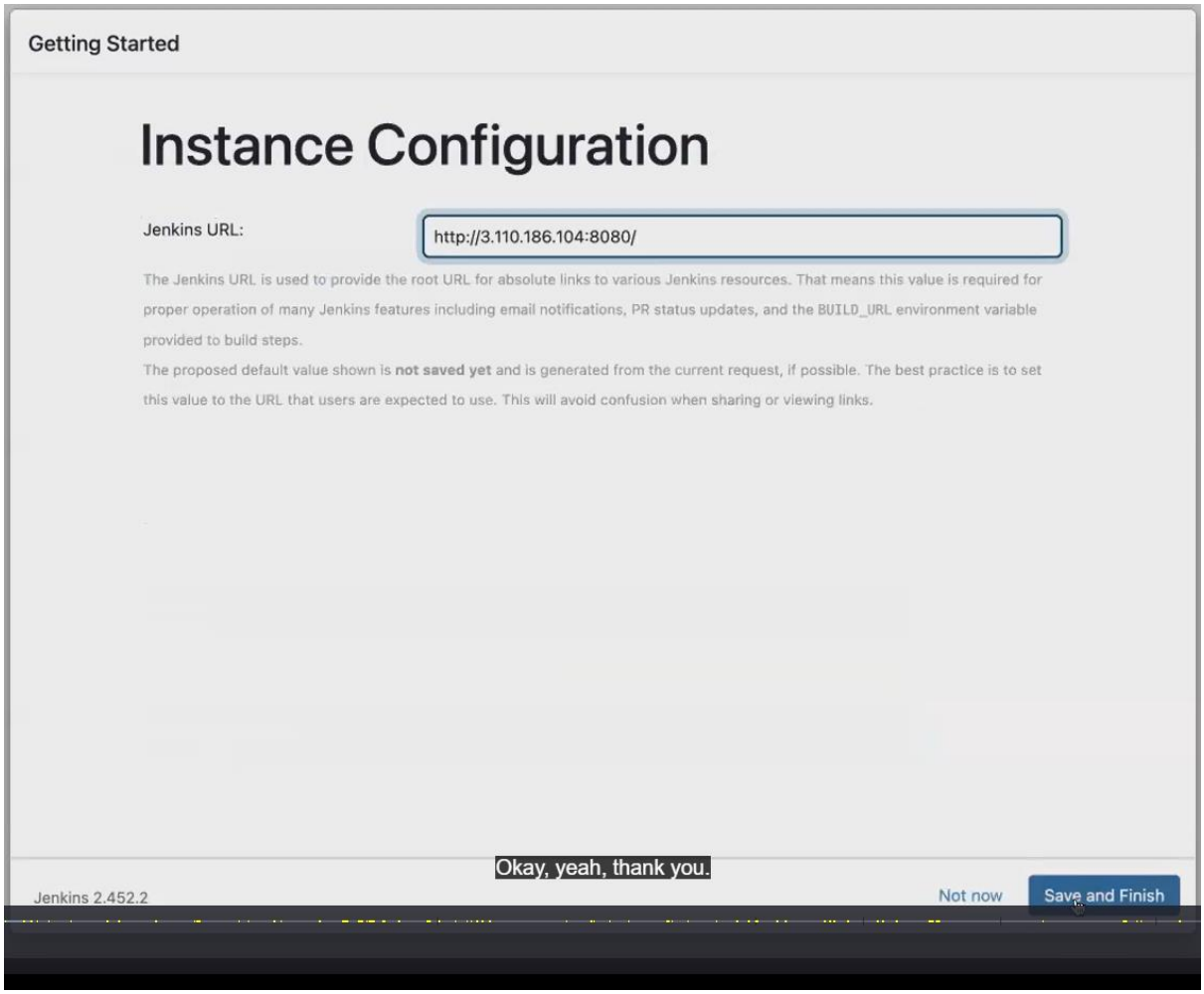
Then Install the suggested plugins:

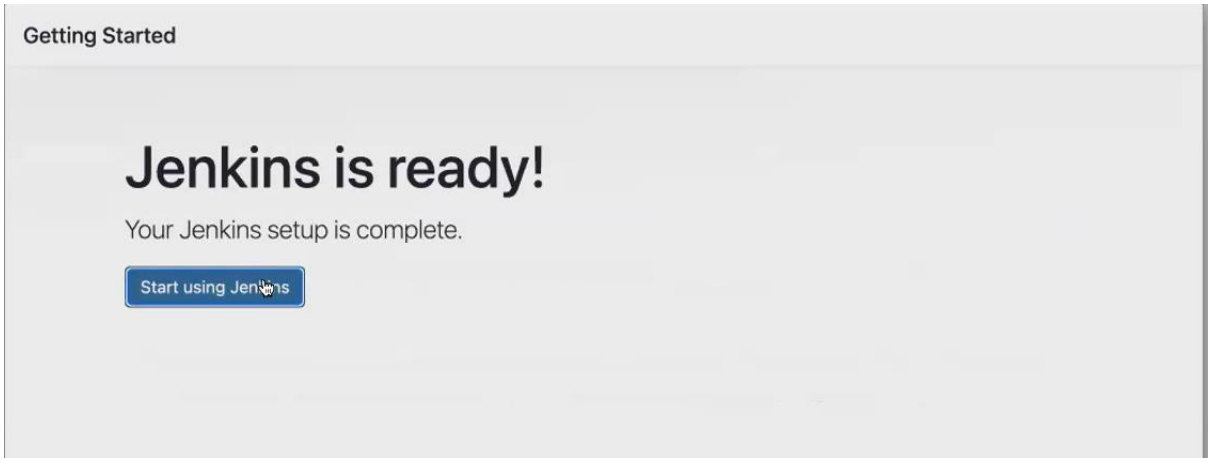


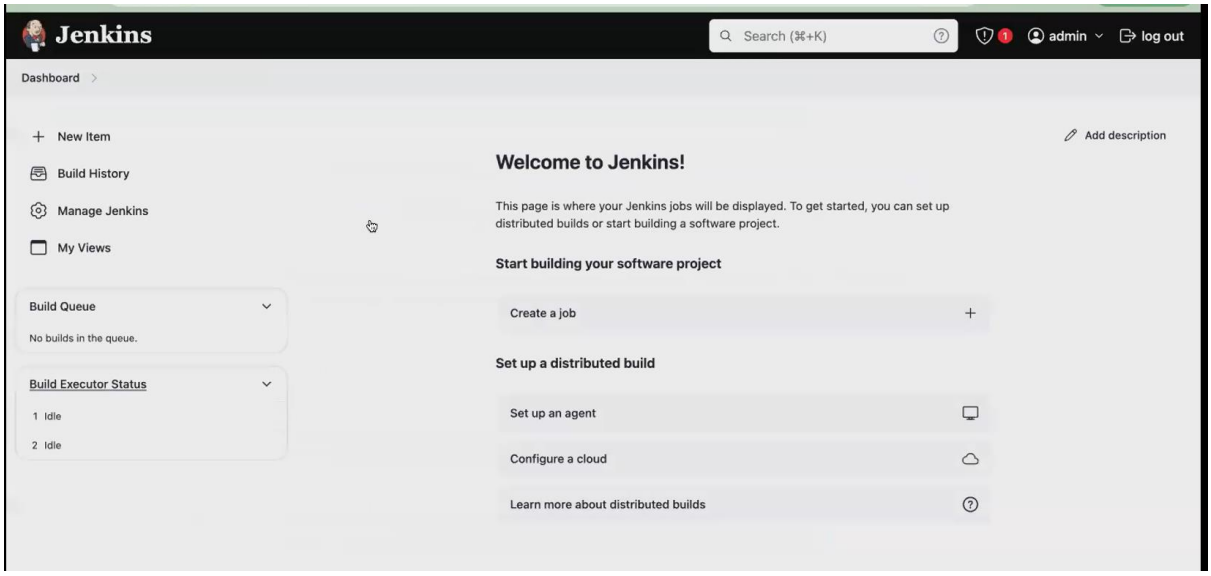

Getting Started
Getting Started
Folders
OWASP Markup Formatter
Build Timeout
Credentials Binding
Timestamper
Workspace Cleanup
Ant
Gradle
Pipeline
GitHub Branch Source
Pipeline: GitHub Groovy Libraries
Pipeline Graph View
Git
SSH Build Agents
Matrix Authorization Strategy
PAM Authentication
LDAP
Email Extension
Mailer
Dark Theme
** Ionicons API
Folders
** - required dependency

Now let’s create our first Admin user and provide the required info: Give username as admin, then only we can access all inside Jenkins. In username don’t give your name.In full name you give your name or any other name can be given.




Getting Started
Instance Configuration
Jenkins URL:
http://3.110.186.104:8080/
The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the BUILD_URL environment variable provided to build steps.
The proposed default value shown is not saved yet and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.
Okay, yeah, thank you.
Jenkins 2.452.2
Not now
Save and Finish

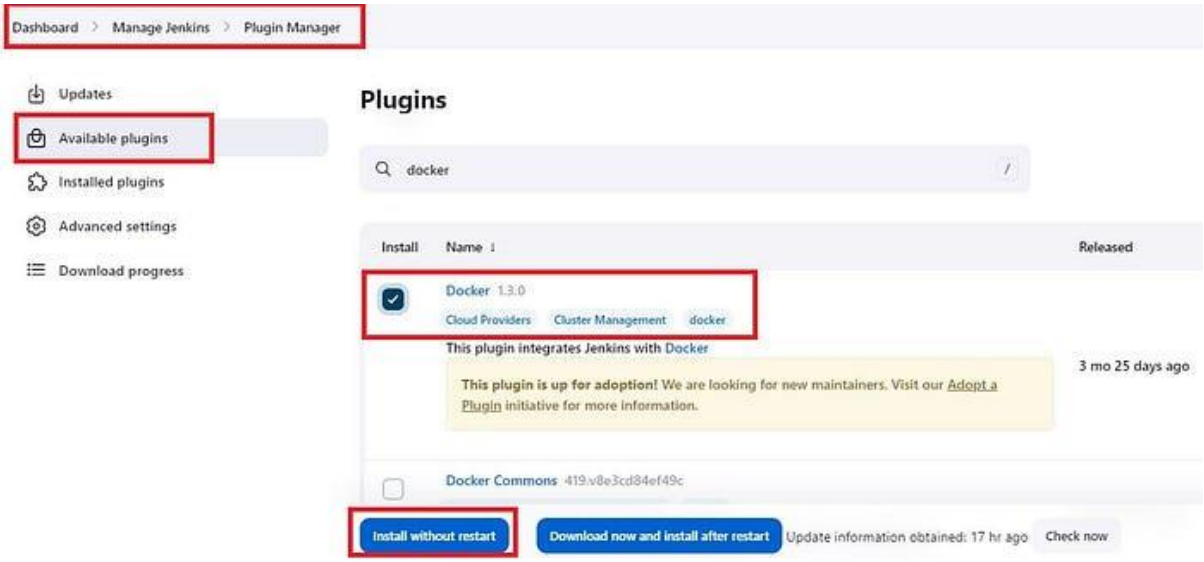

Getting Started
Jenkins is ready!
Your Jenkins setup is complete.
Start using Jenkins



After creating the user we should land into the Jenkins dashboard and let's install the Docker Plugin.

Click on Manage Jenkins -> Manage Plugins -> Available Plugins and **install without restart** the Docker plugin.

In github repo,we have docker file..so we installed



Step 3: Install Docker in ec2

After the successful installation of Docker in Jenkins ,install docker in ec2 also.

```
ubuntu@ip-172-31-32-49:~$ sudo apt install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

`sudo usermod -aG docker Jenkins`

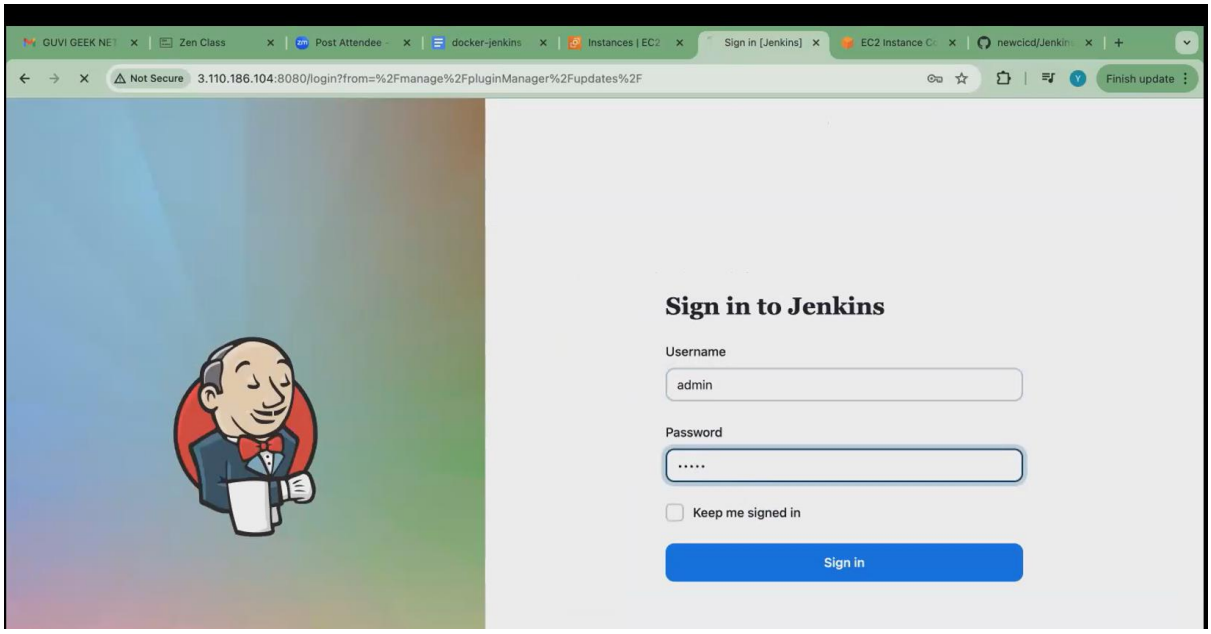
`sudo systemctl restart jenkins`

Give docker permission for Jenkins. Then only we use docker in jenkins

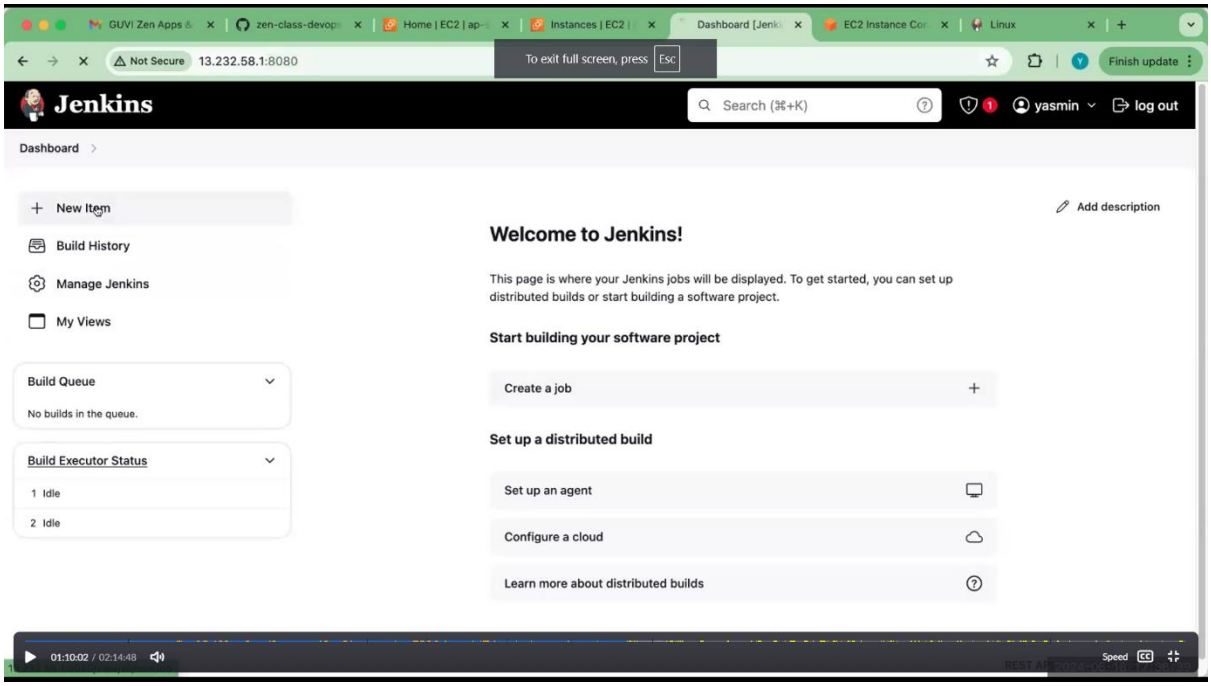
```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-32-49:~$ sudo usermod -aG docker jenkins
ubuntu@ip-172-31-32-49:~$ sudo systemctl restart jenkins
```

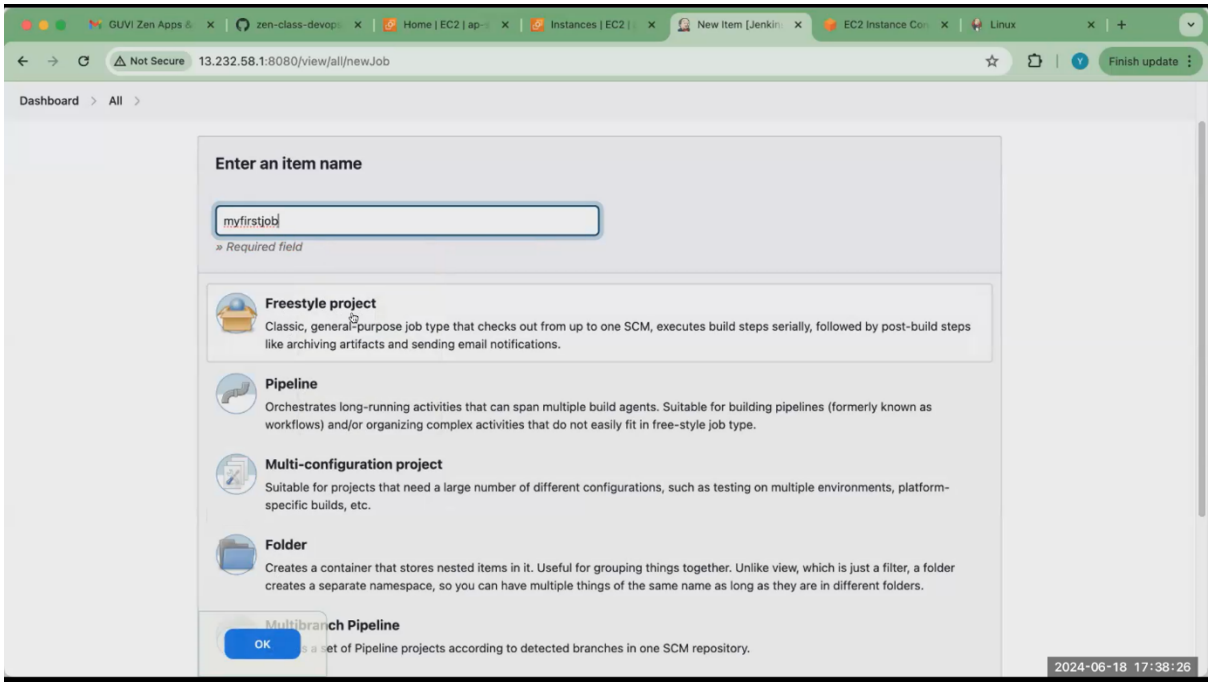
```
ubuntu@ubuntu-server:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
     Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2023-05-20 07:32:37 UTC; 1min 18s ago
TriggeredBy: ● docker.socket
       Docs: https://docs.docker.com
   Main PID: 17940 (dockerd)
      Tasks: 7
     Memory: 45.8M
        CPU: 347ms
     CGroup: /system.slice/docker.service
             └─17940 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

May 20 07:32:36 ubuntu-server systemd[1]: Starting Docker Application Container Engine...
May 20 07:32:36 ubuntu-server dockerd[17940]: time="2023-05-20T07:32:36.468040537Z" level=info msg="Starting up"
May 20 07:32:36 ubuntu-server dockerd[17940]: time="2023-05-20T07:32:36.474901235Z" level=info msg="detected 127.0.0.>
May 20 07:32:36 ubuntu-server dockerd[17940]: time="2023-05-20T07:32:36.652606877Z" level=info msg="Loading container>
May 20 07:32:37 ubuntu-server dockerd[17940]: time="2023-05-20T07:32:37.165371275Z" level=info msg="Loading container>
May 20 07:32:37 ubuntu-server dockerd[17940]: time="2023-05-20T07:32:37.358904484Z" level=info msg="Docker daemon" co>
May 20 07:32:37 ubuntu-server dockerd[17940]: time="2023-05-20T07:32:37.359412915Z" level=info msg="Daemon has comple>
May 20 07:32:37 ubuntu-server dockerd[17940]: time="2023-05-20T07:32:37.409161518Z" level=info msg="API listen on /ru>
May 20 07:32:37 ubuntu-server systemd[1]: Started Docker Application Container Engine.
```

Add the jenkins user to the docker group and refresh the Jenkins page/server. Again ask for sign in.



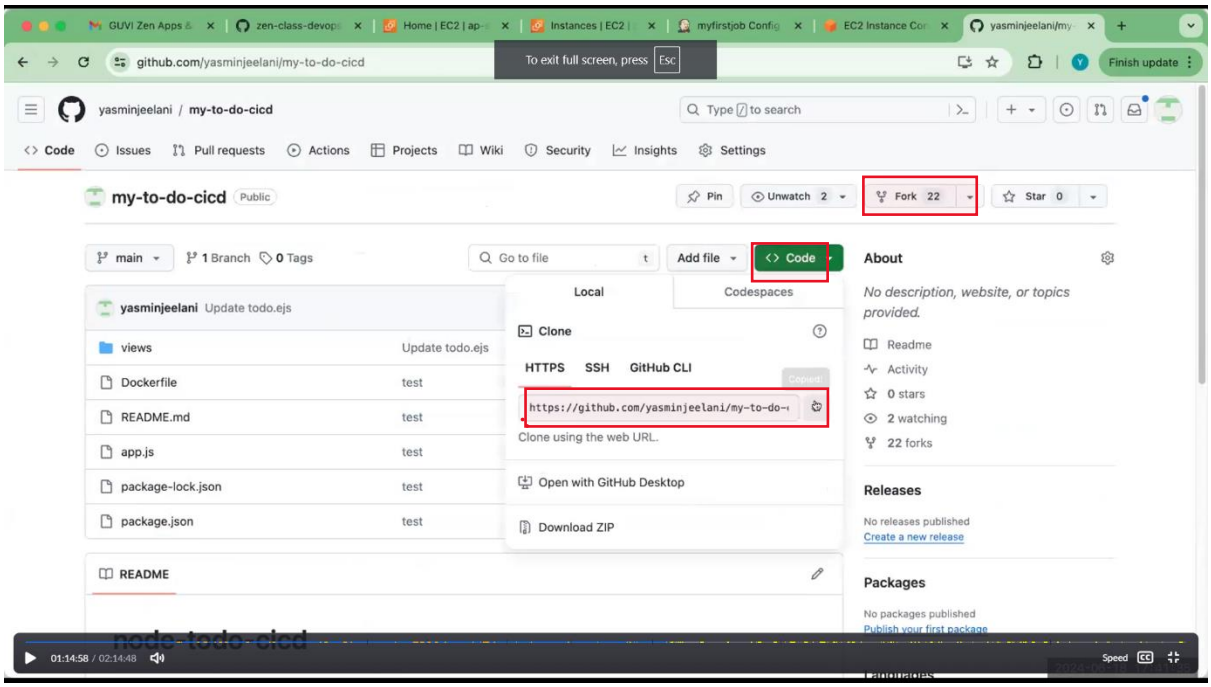
Create a project:

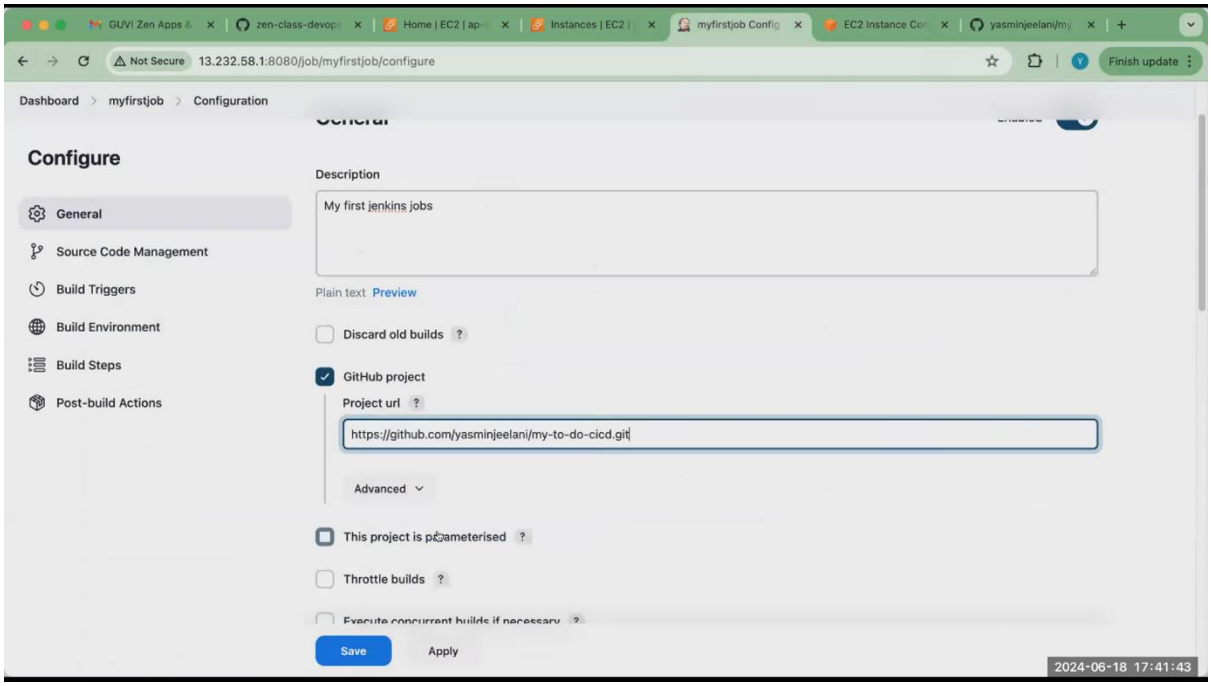


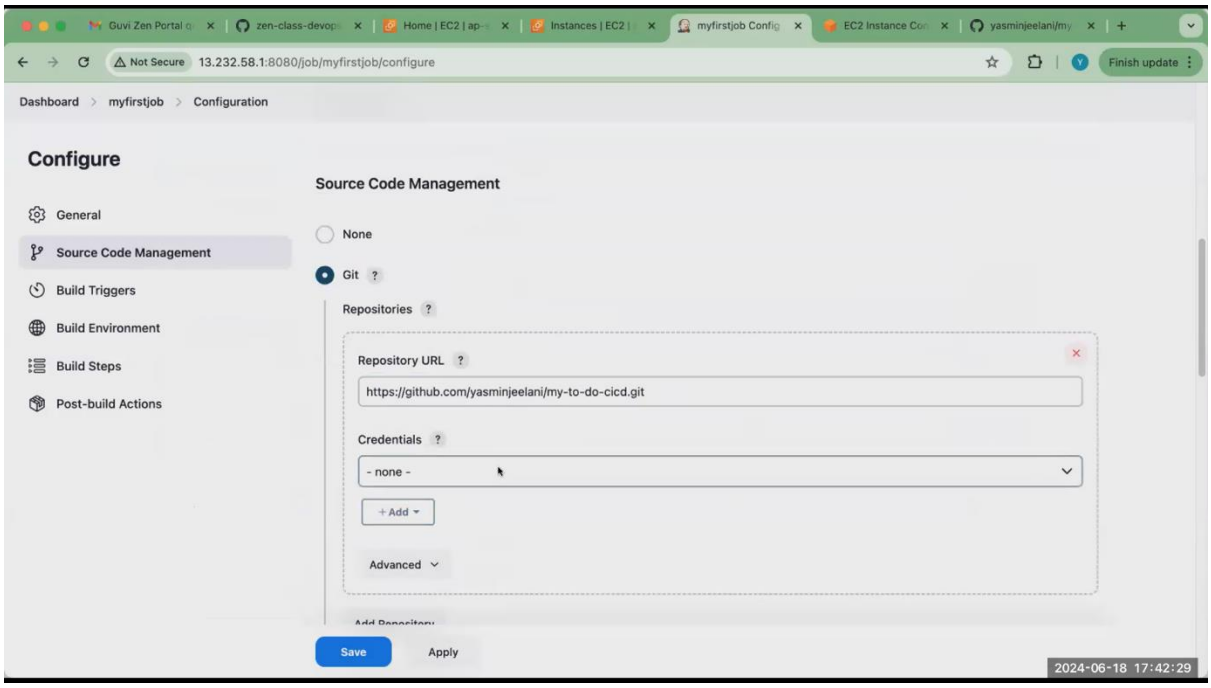


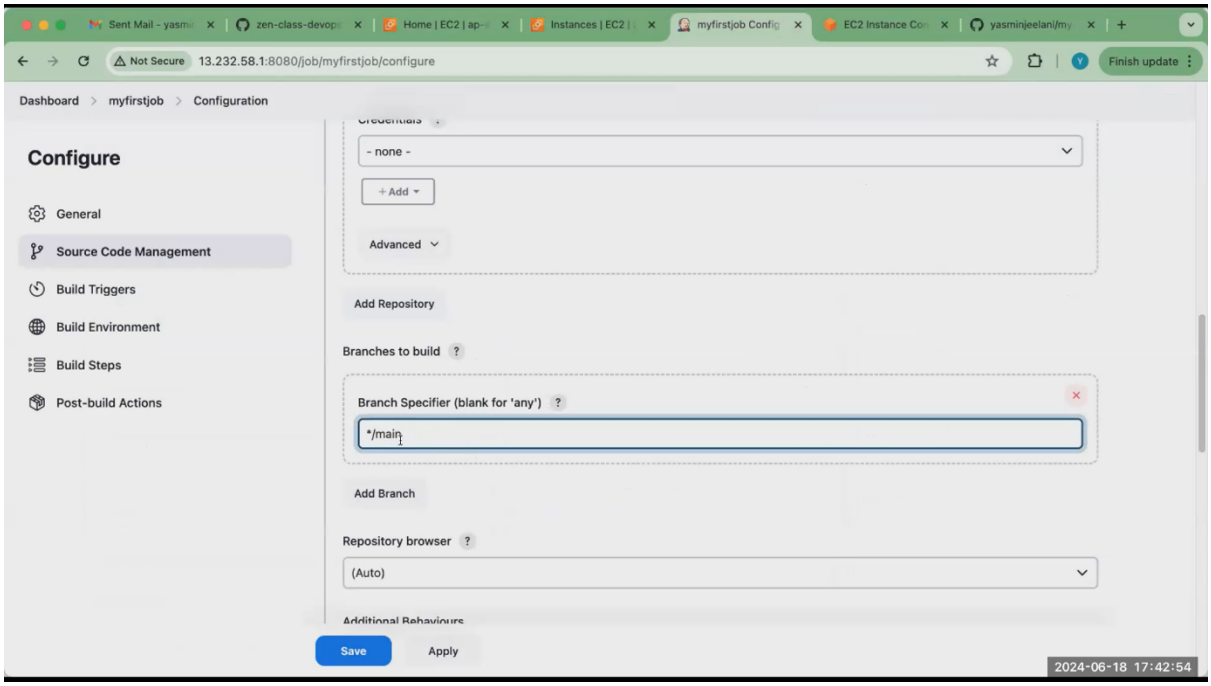
Jenkins get a job in github.

https://github.com/yasminjeelani/my-to-do-cicd.git (yasmine github link for creating this project)..by using fork option copy yasmine github to my GitHub. After fork ,our code generated. Copy this and paste it in Jenkins.

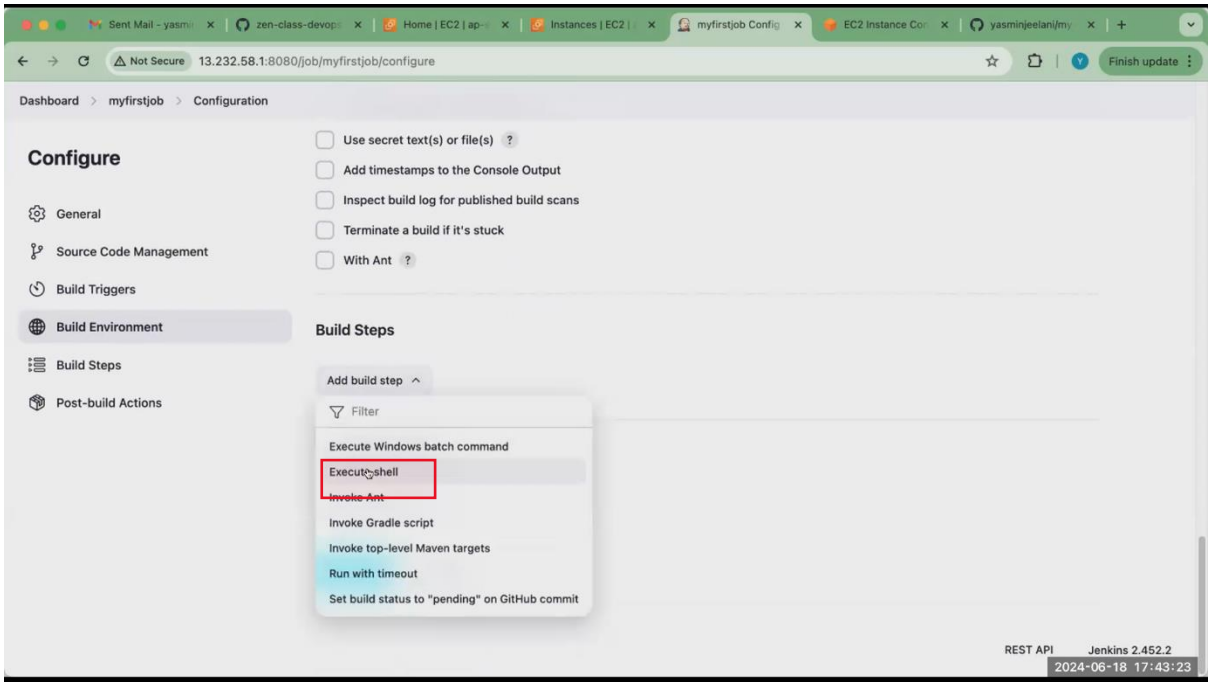



Not Secure 13.232.58.1:8080/job/myfirstjob/configure
Finish update
Dashboard > myfirstjob > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Description
My first jenkins jobs
Plain text Preview
Discard old builds
GitHub project
Project url
https://github.com/yasminjeelani/my-to-do-cicd.git
Advanced
This project is parameterised
Throttle builds
Save
Apply
2024-06-18 17:41:43

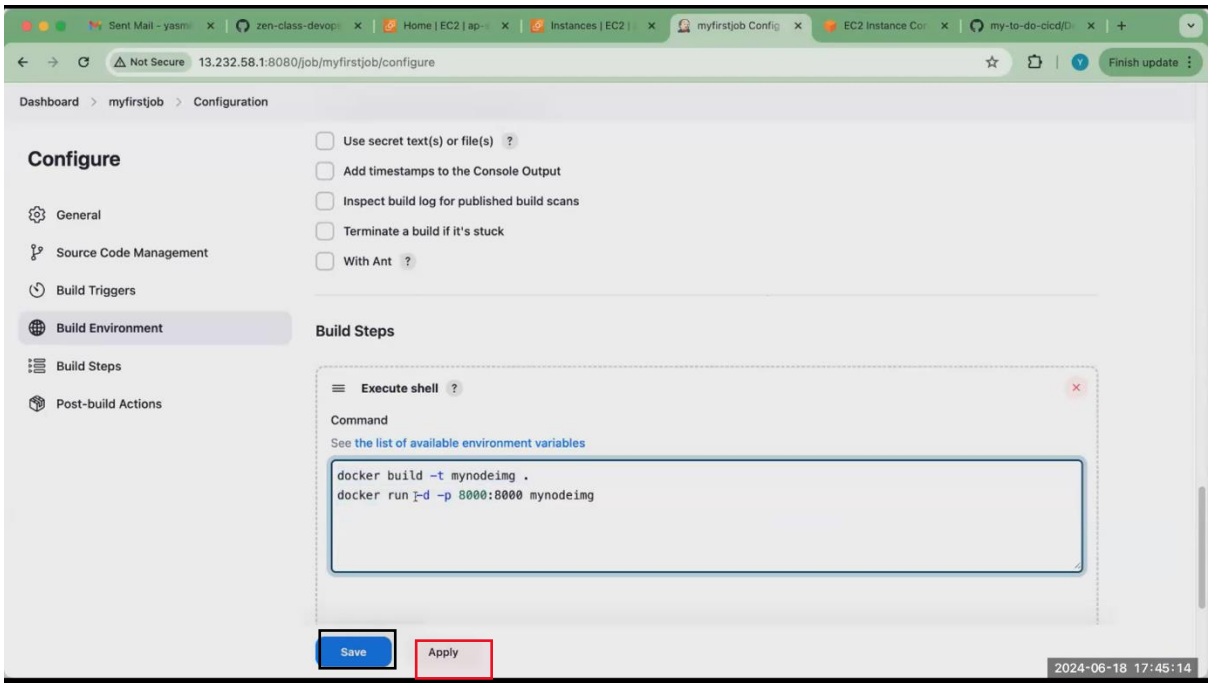

Not Secure 13.232.58.1:8080/job/myfirstjob/configure
Finish update
Dashboard > myfirstjob > Configuration
Configure
General
Source Code Management
Build Triggers
Build Environment
Build Steps
Post-build Actions
Source Code Management
None
Git
Repositories
Repository URL
https://github.com/yasminjeelani/my-to-do-cicd.git
Credentials
- none -
+ Add
Advanced
Save
Apply
2024-06-18 17:42:29



We have run this in shell.so select this



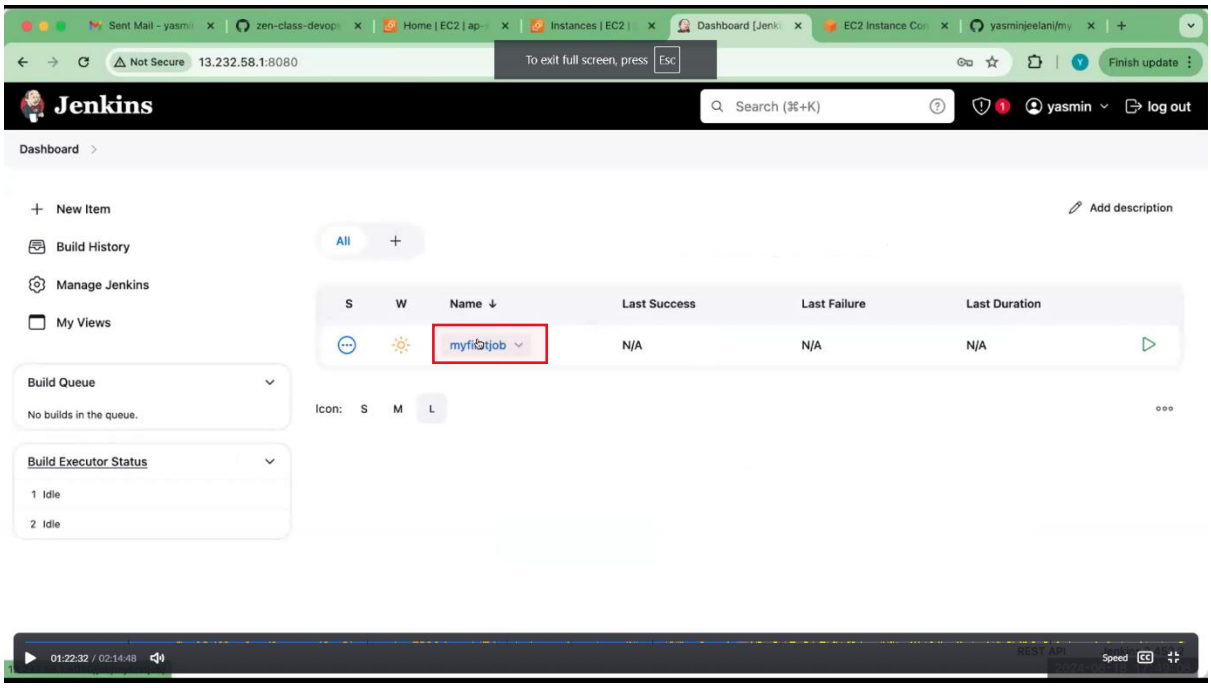
In github repo ,we have dockerfile .so build the image and run that image.give that command in execute shell.

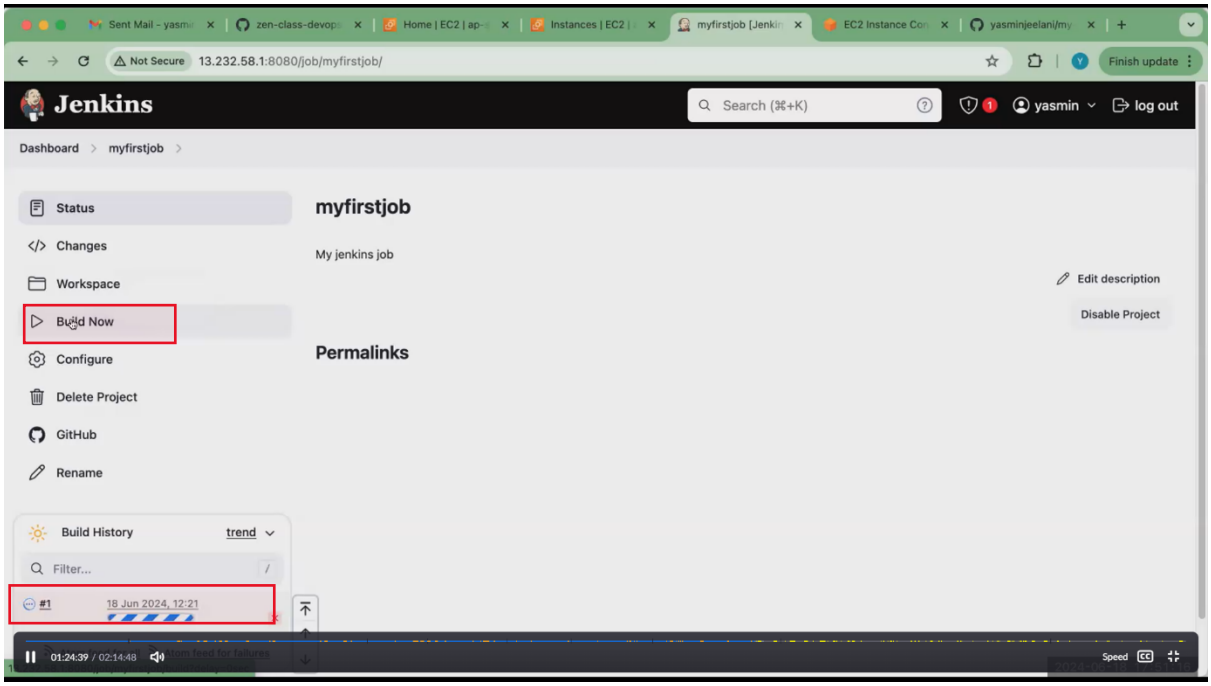


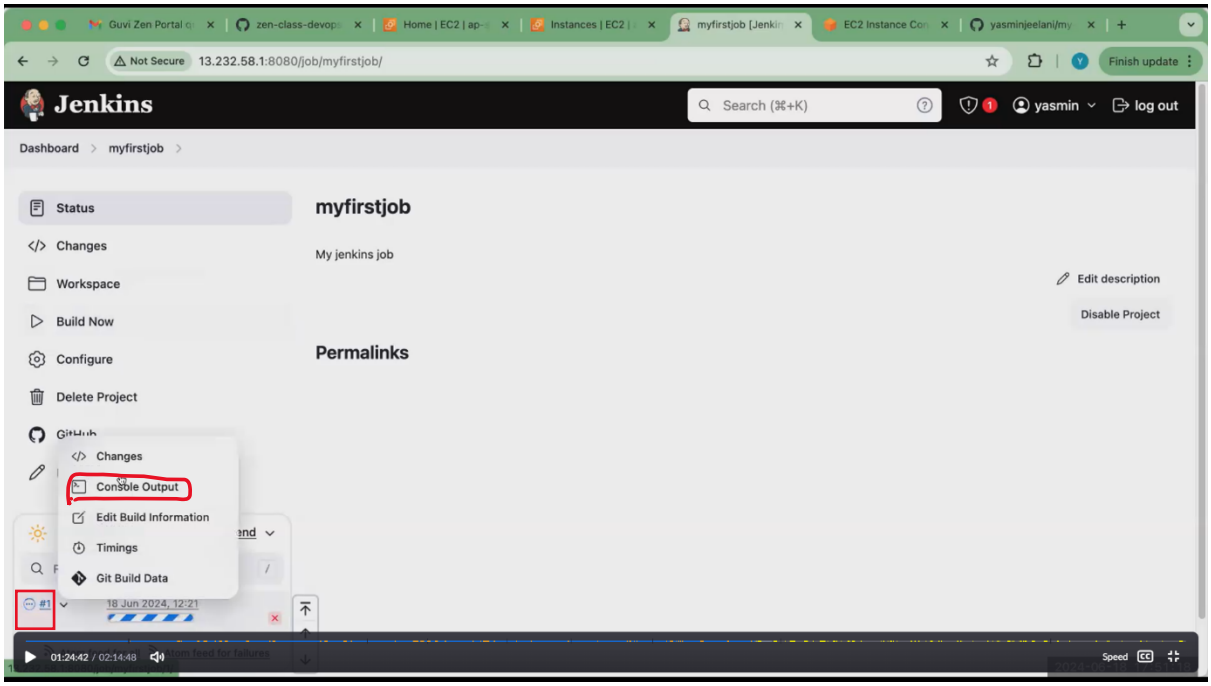
After this apply and save.

In step 3 install docker in ec2..before build this job check docker installed or not.

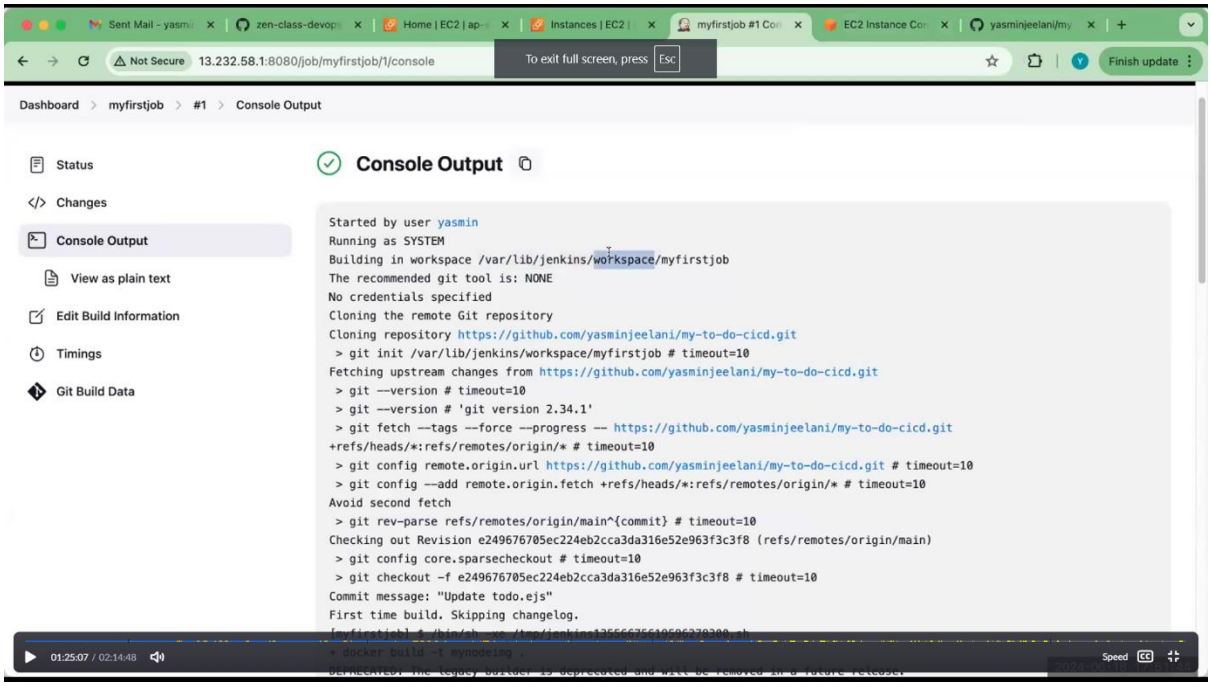
After apply and save our project created:

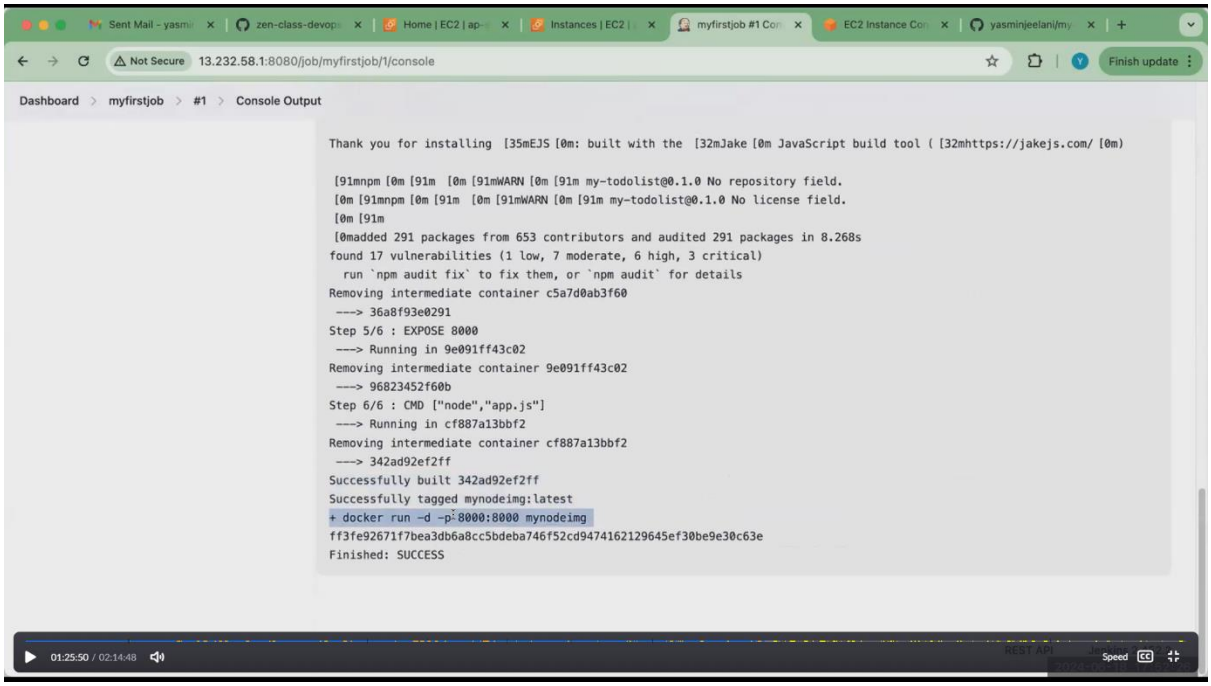




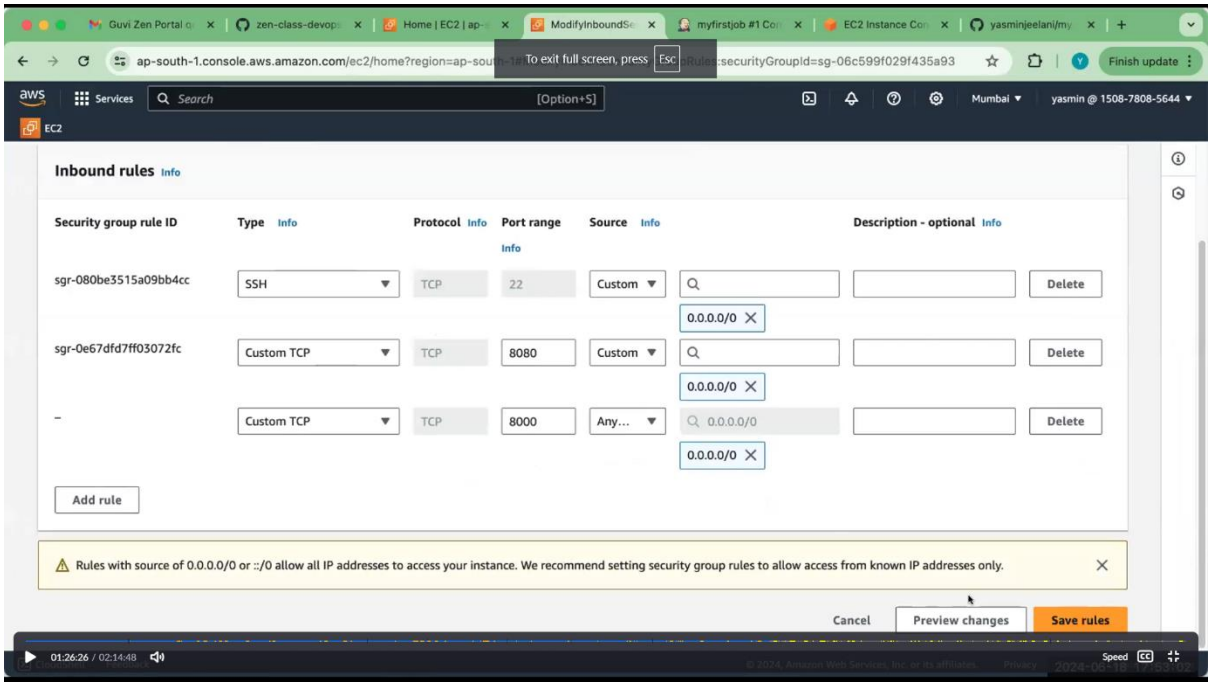


After select build now ,click console output. If you have no error console output was success.

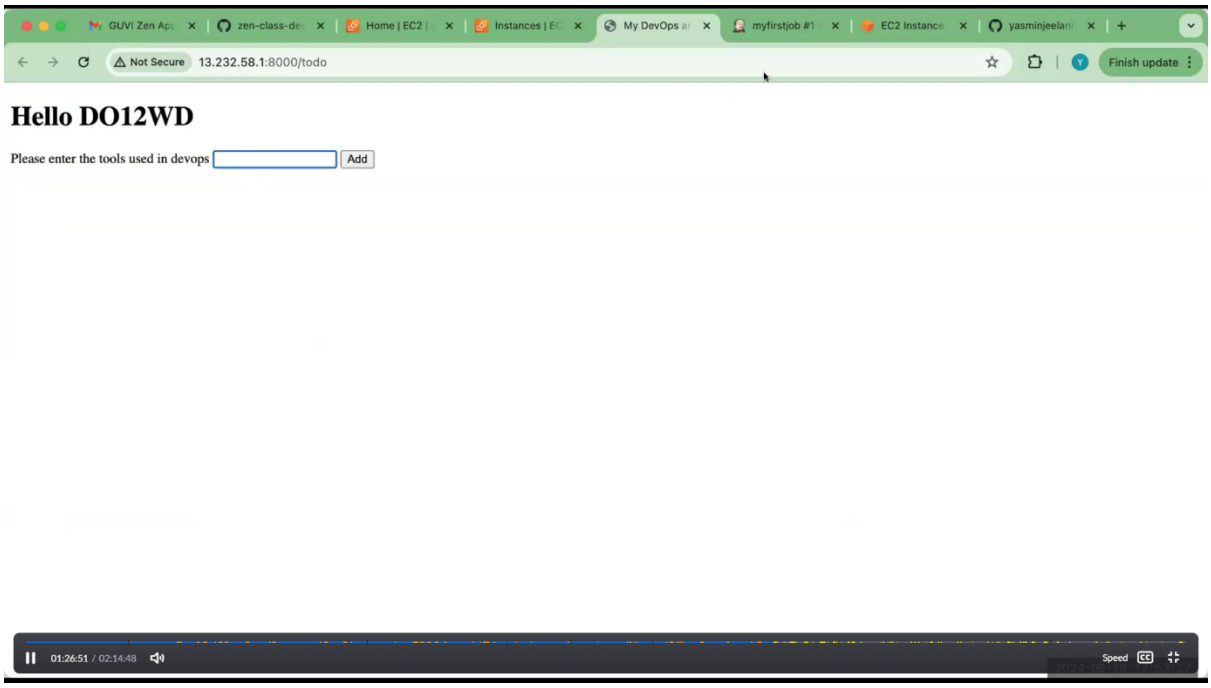




Application run in port no: 8000.open the port no in ec2

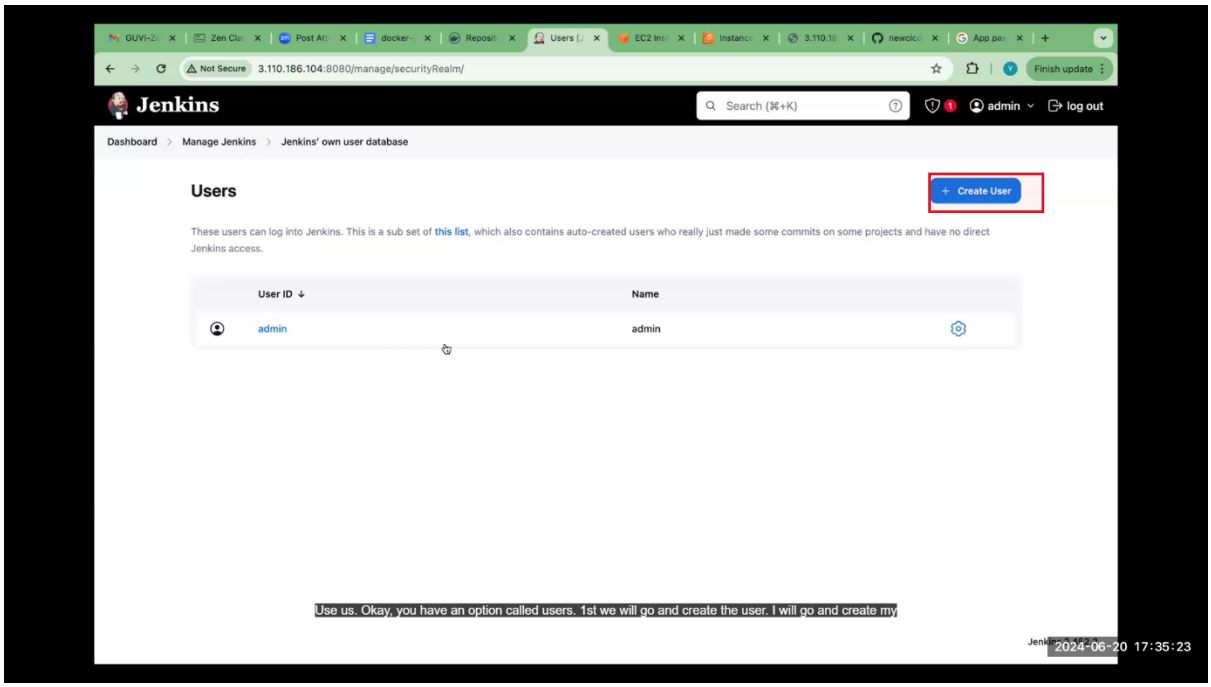


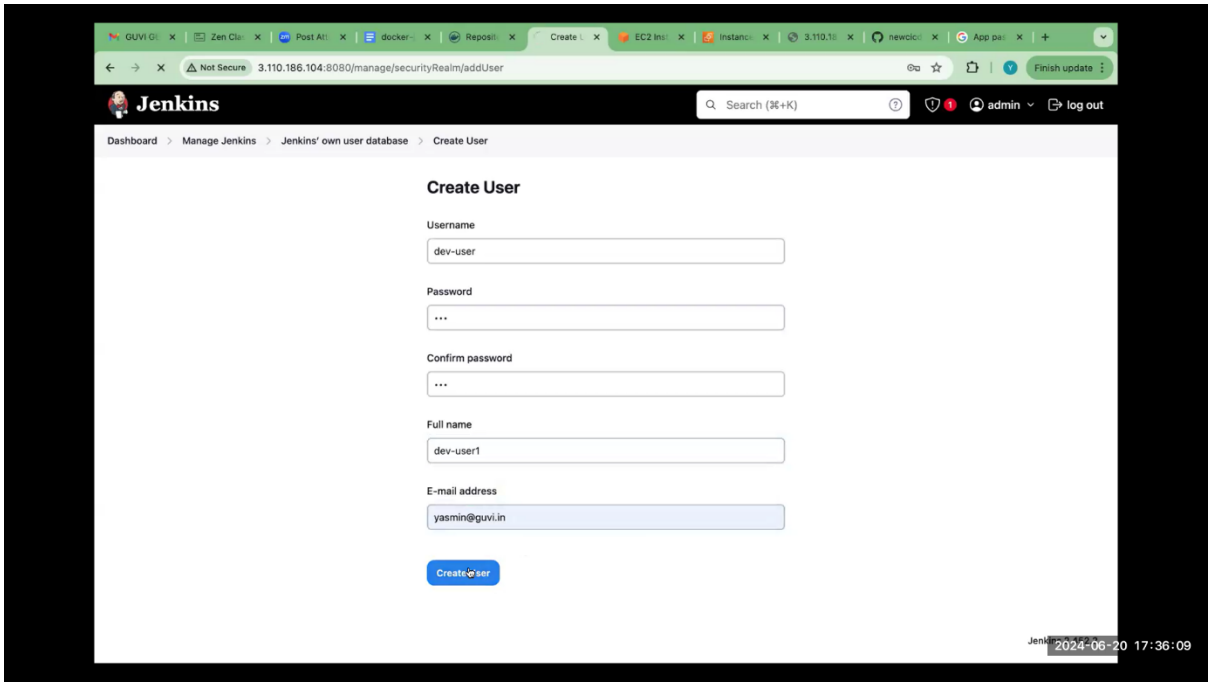
Jenkins get the application from github and runs

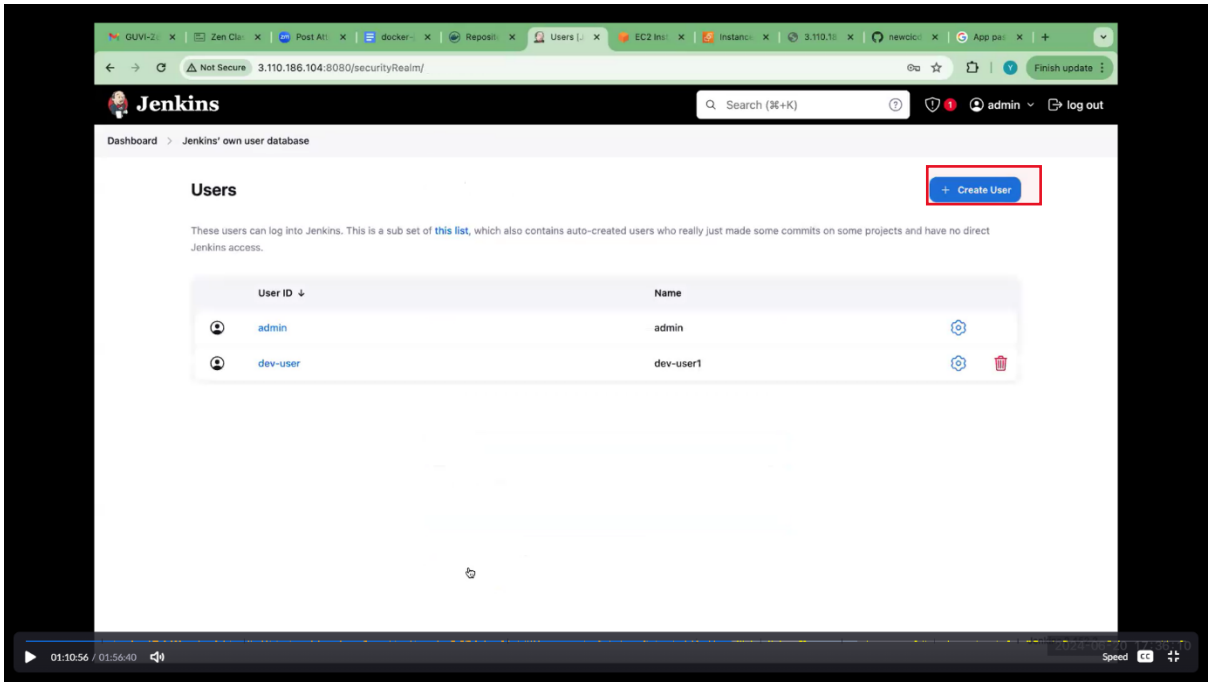


Now simple jenkin project can created successfully.

2. Create users (developer):

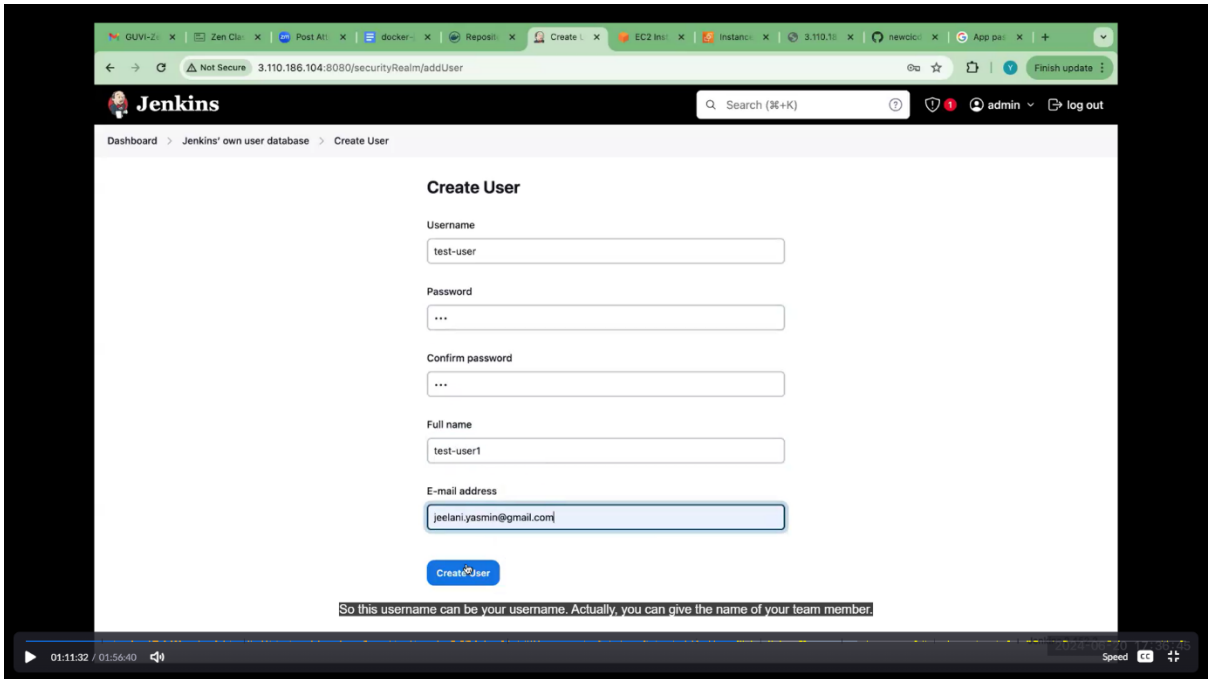


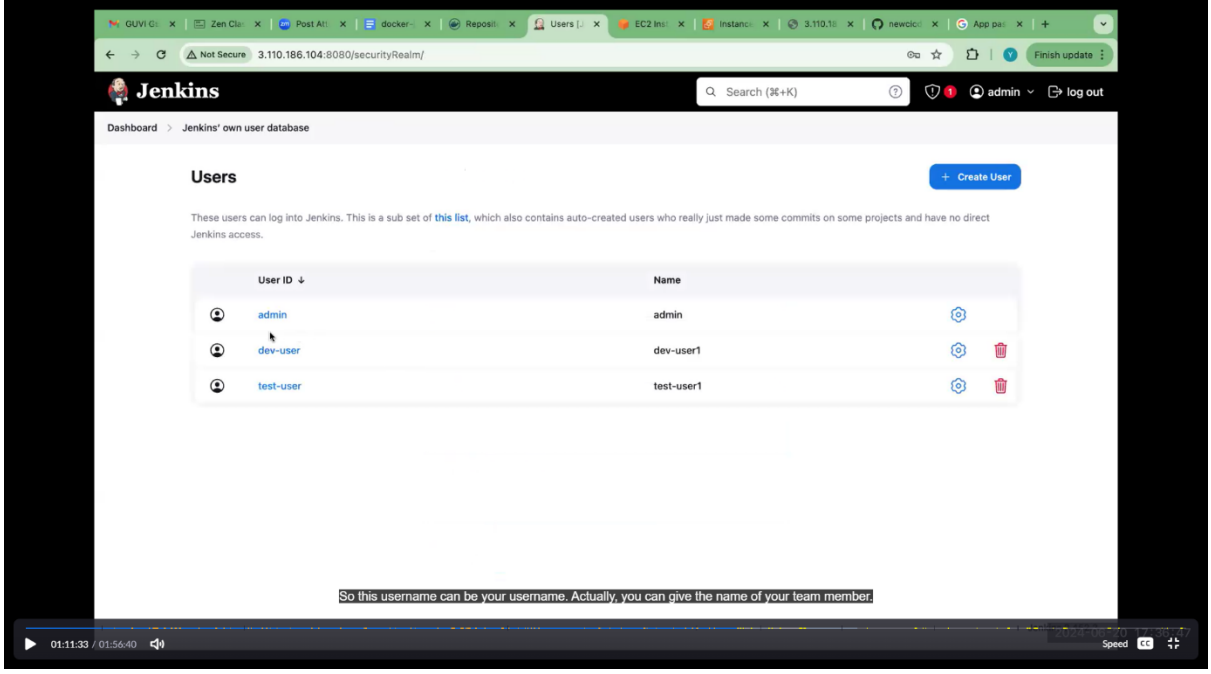




Dev user created

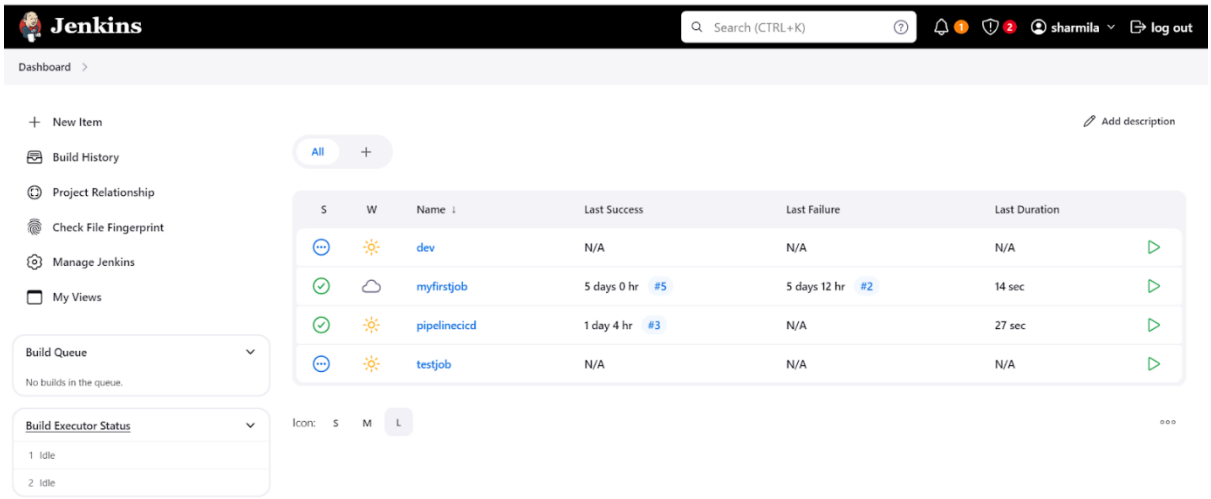
2. Create users (test user):

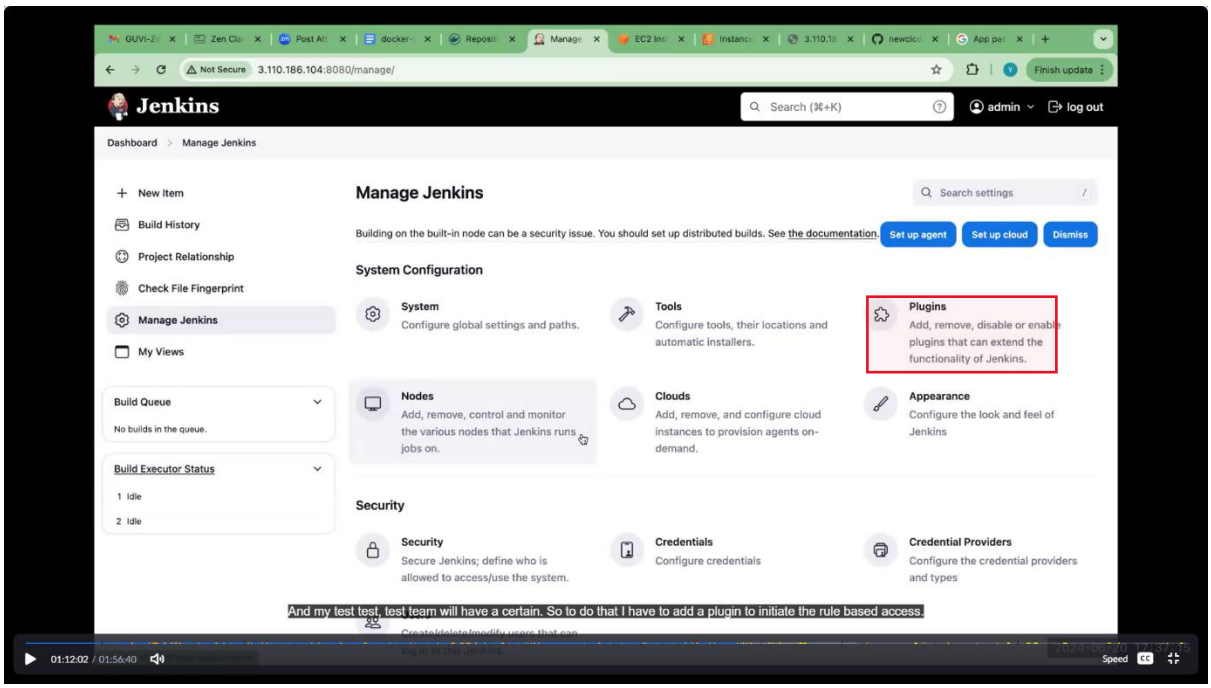




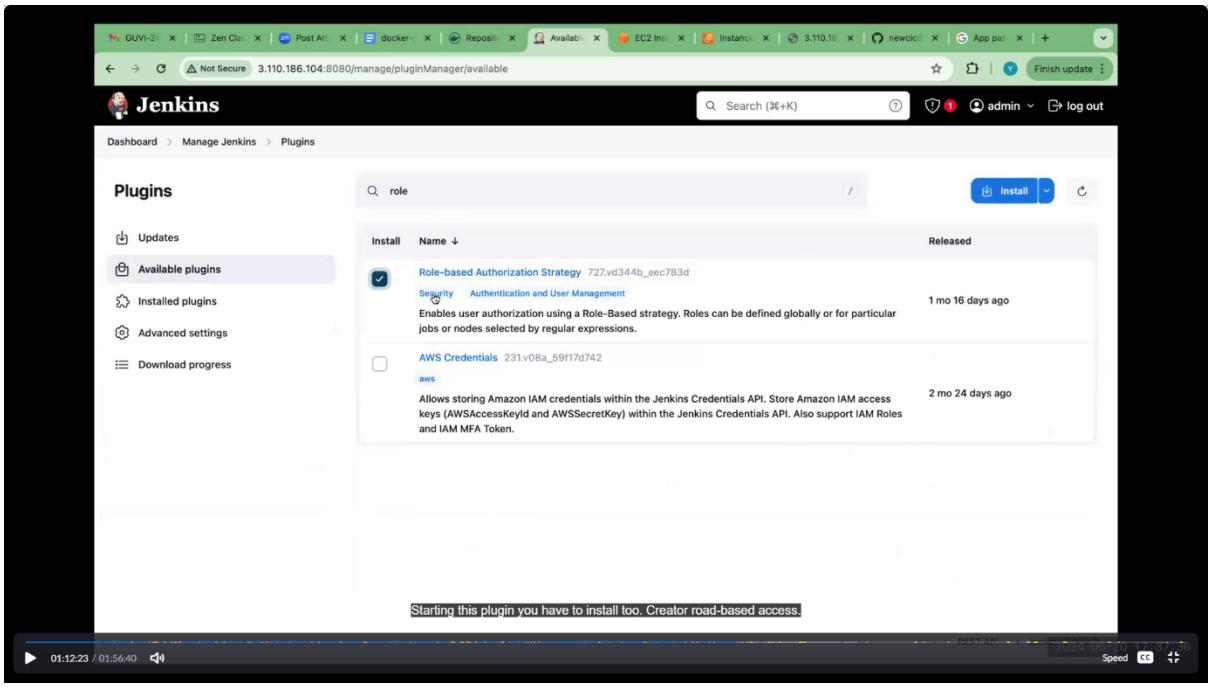
Test user created.

In dashboard you have to see the users

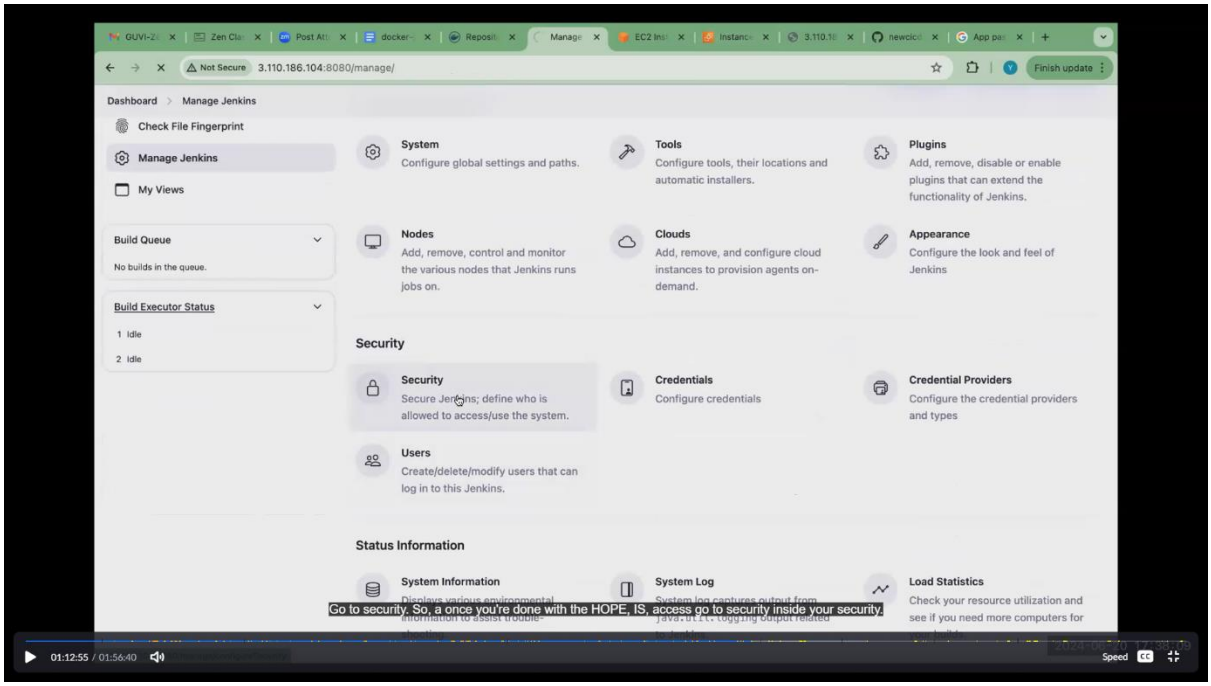




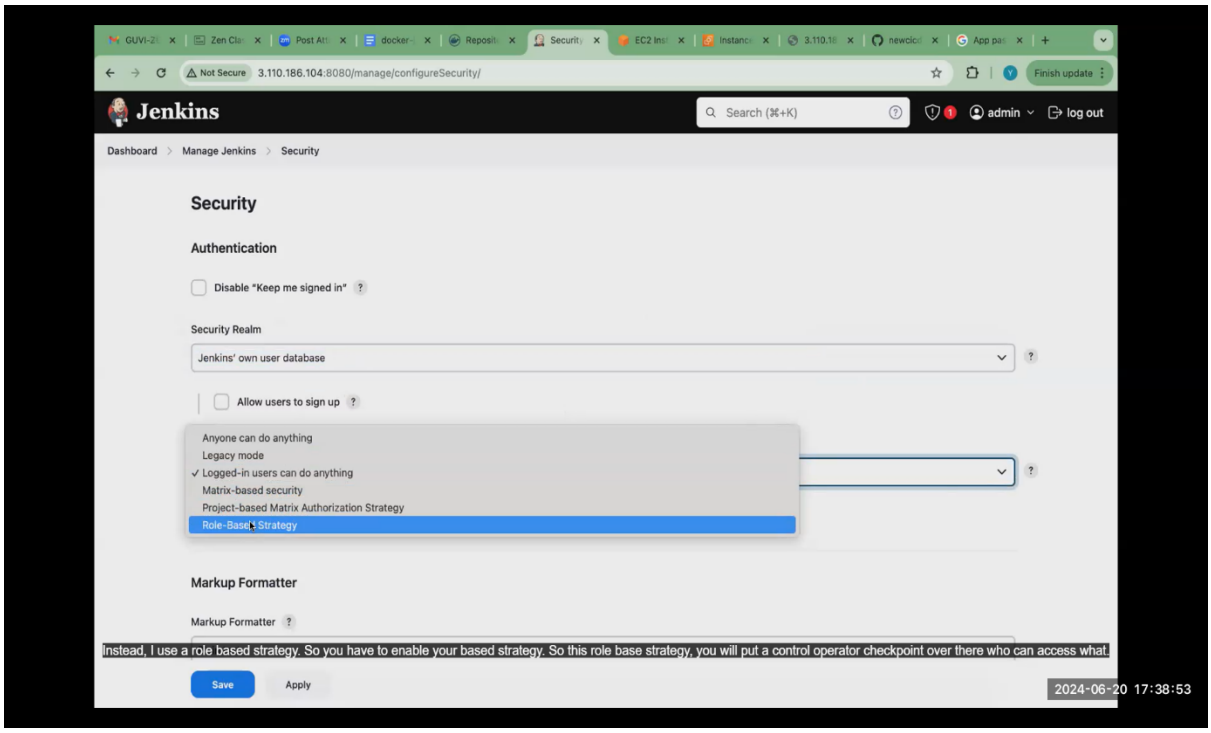
2 users can be created, after this ,we have to give permission for each users.for this we have to add plugin. Manage Jenkins-→pulgins→available pulgins→search role based authorization →install



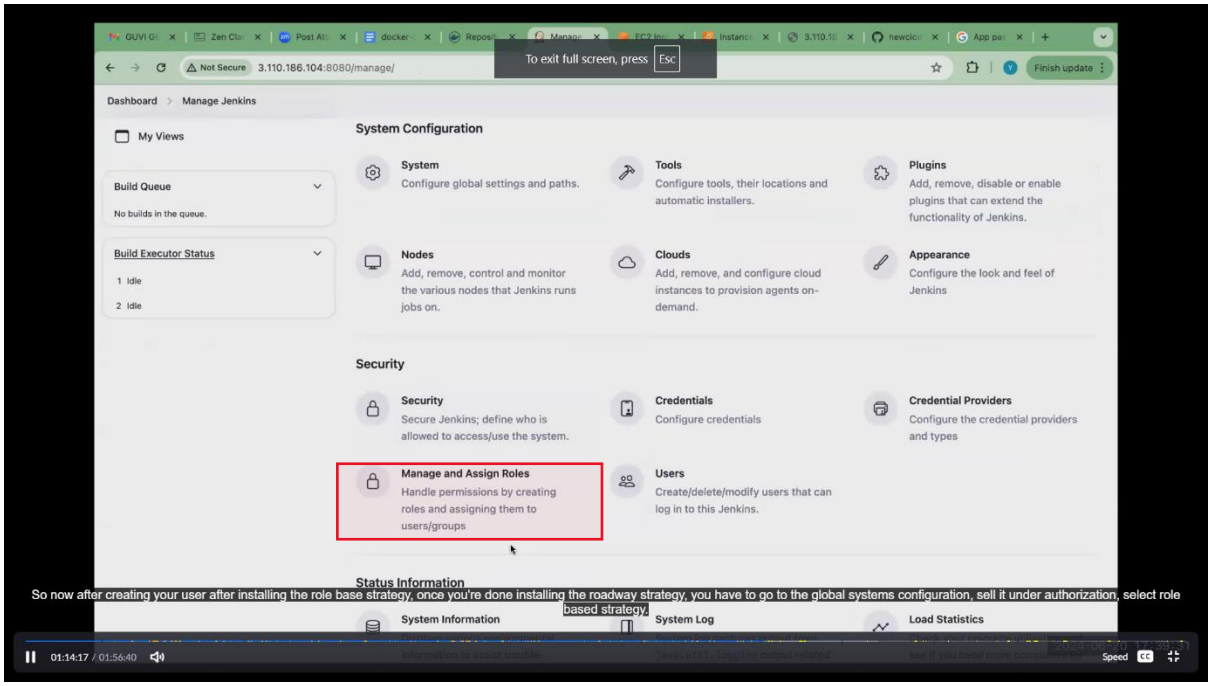
After installed role based authorization, go to manage Jenkins→security
Change allow users signin to role based strategy.



Change allow users signin to role based stratagy.then only we have to give permissions for each users. Then click apply →save.

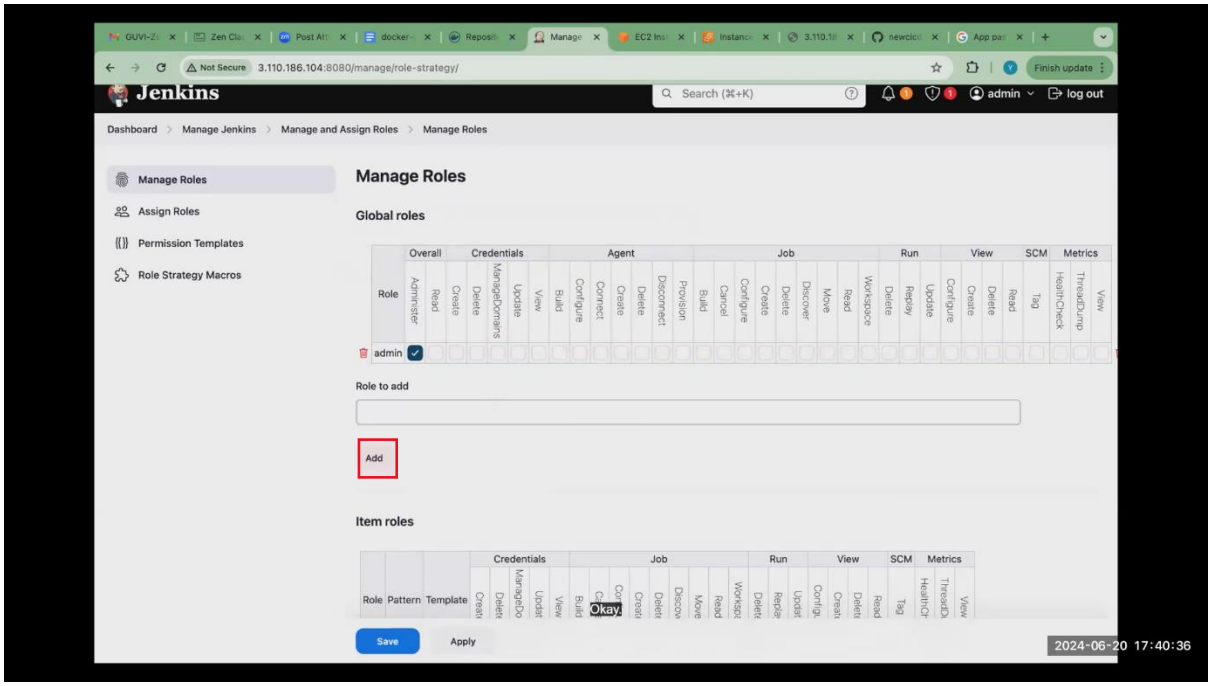


After that refresh the Jenkins page ,manage and assign role option available..

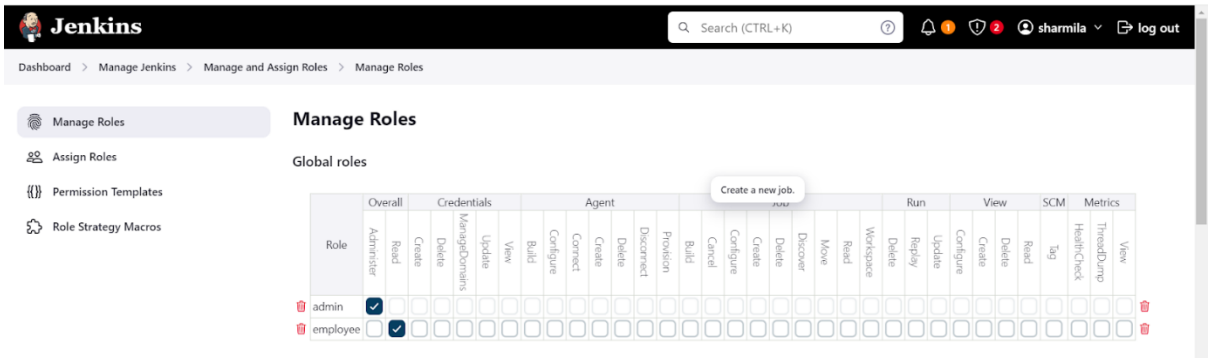


i) Manage roles:

a. Global roles:



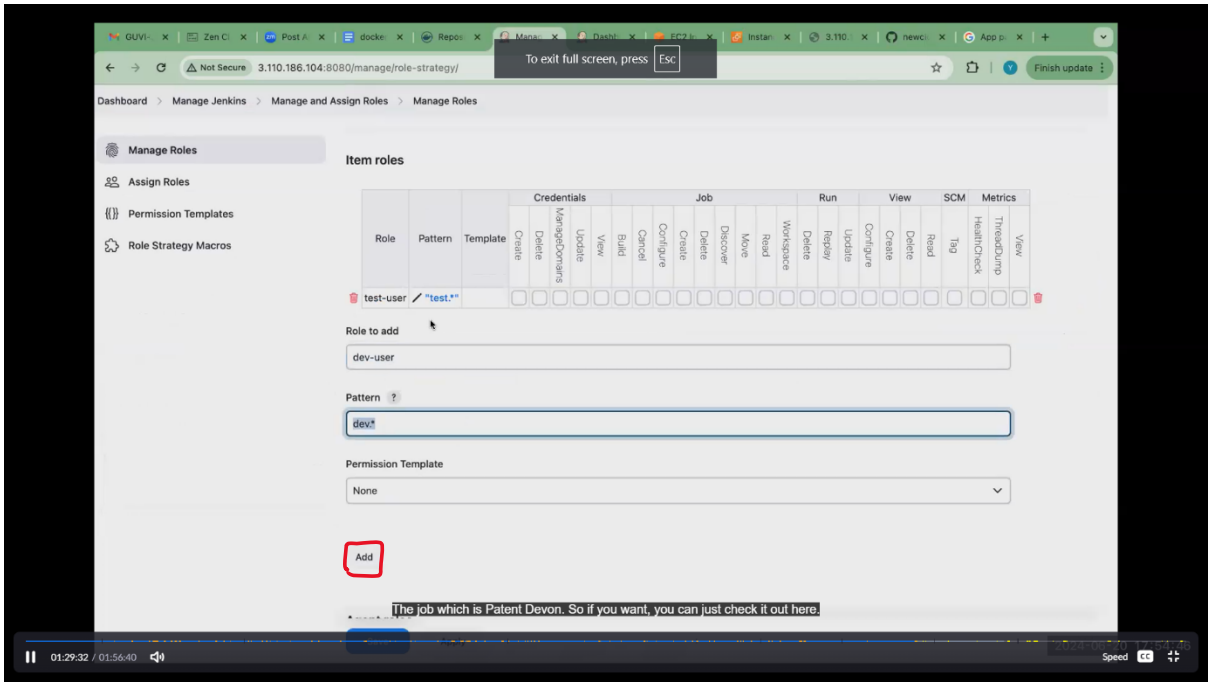
Click add you have to add employee. Give permission as I gave



b. **Item role:**

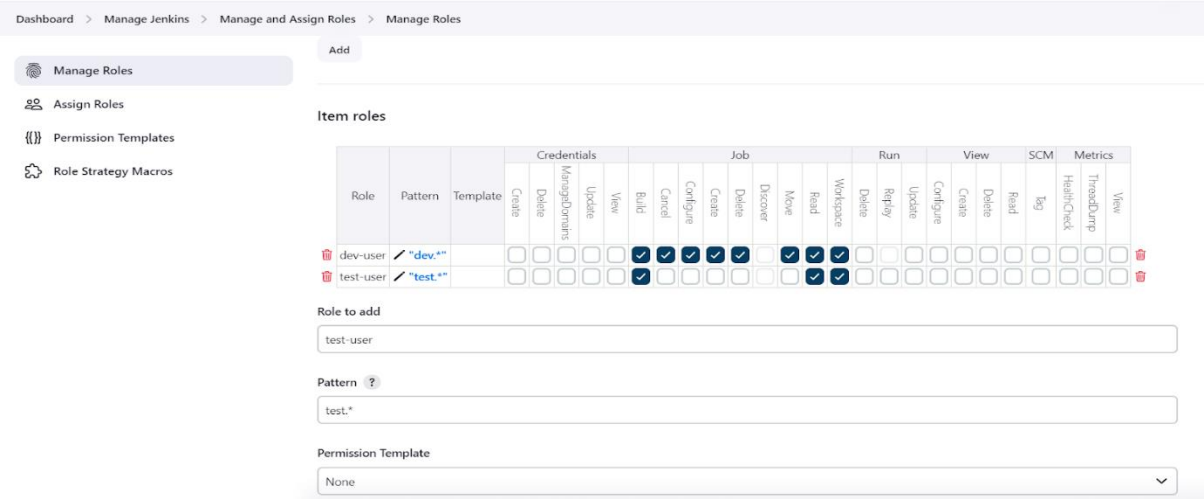
Pattern matched with job we create..so give the same pattern becz its case sensitive.

Dev user:



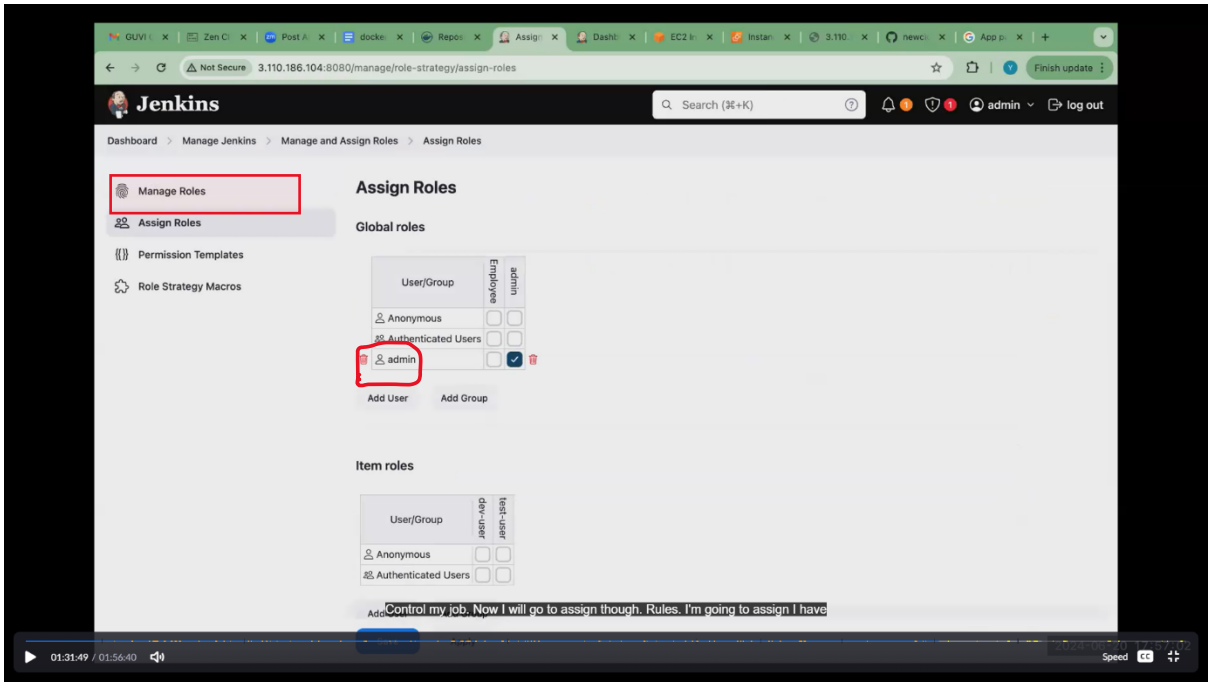
Test user:

Similar to dev user create test user ..permission gave as per in image

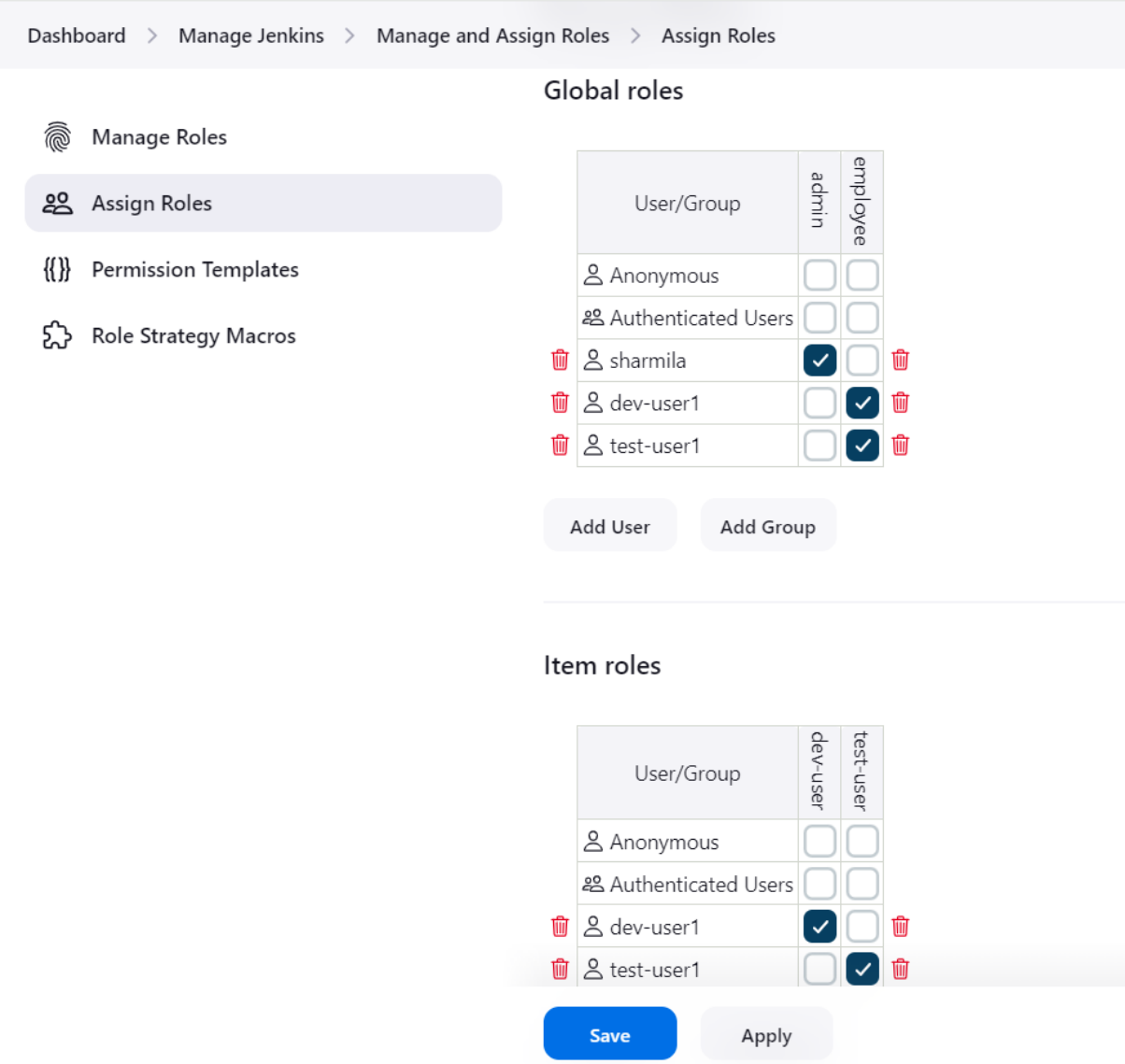


ii) Assign roles (global and item roles):

Click add user add dev user-1 and test user-1

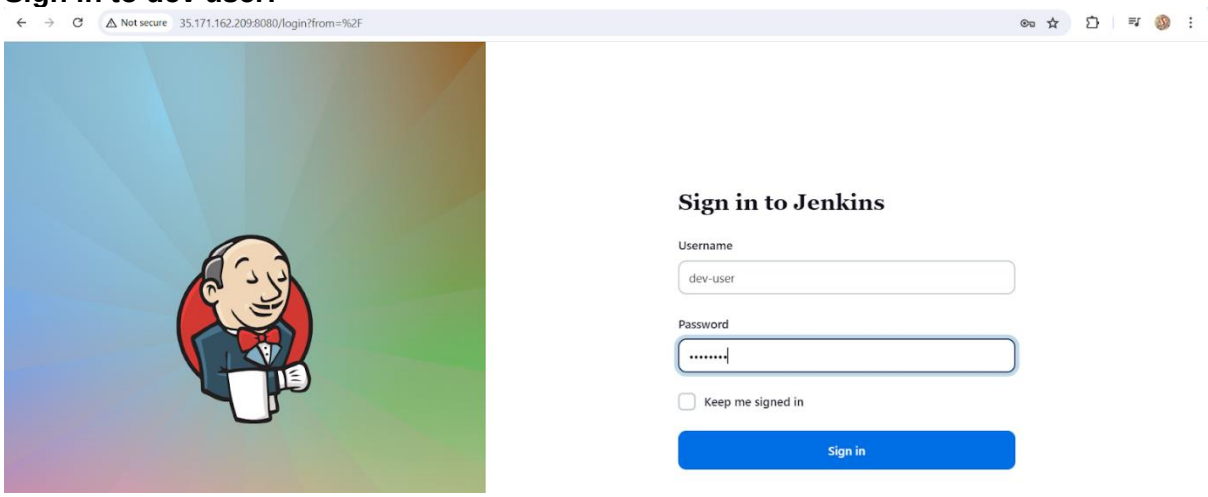


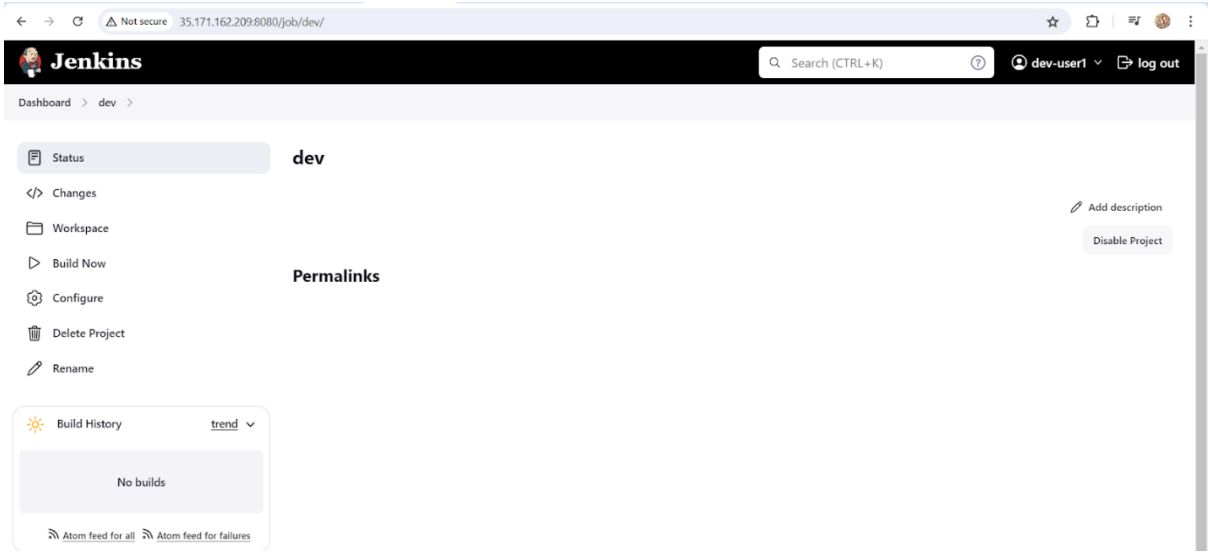
Click add user (global and item roles) add user id as dev-user1 and test-user1. Then apply and save.

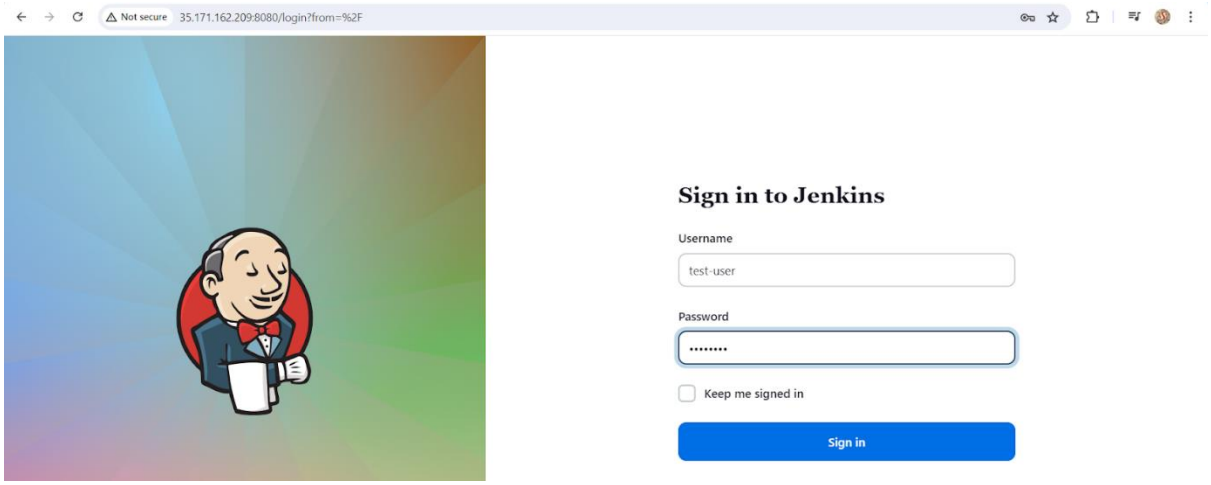


Logout admin ,sign in as dev user. At the time of creating job we give username and password.just for checking purpose we have to login

Sign in to dev user:





Sign in to test user:

Logout admin ,sign in as tset user. At the time of creating job we give username and password.