**End-to-end Project (using GitHub to the webserver)**

|  |  |
| --- | --- |
| **Author:** | **Signature/ Date:** |
| Rabindra Nath Moharana |  |
| **Approver:** | **Signature/ Date:** |
|  |  |

List of reviewers:

|  |  |  |
| --- | --- | --- |
| **Name of Reviewer** | **Role / Function** | **Date of Review** |
|  |  |  |

Table of Contents:

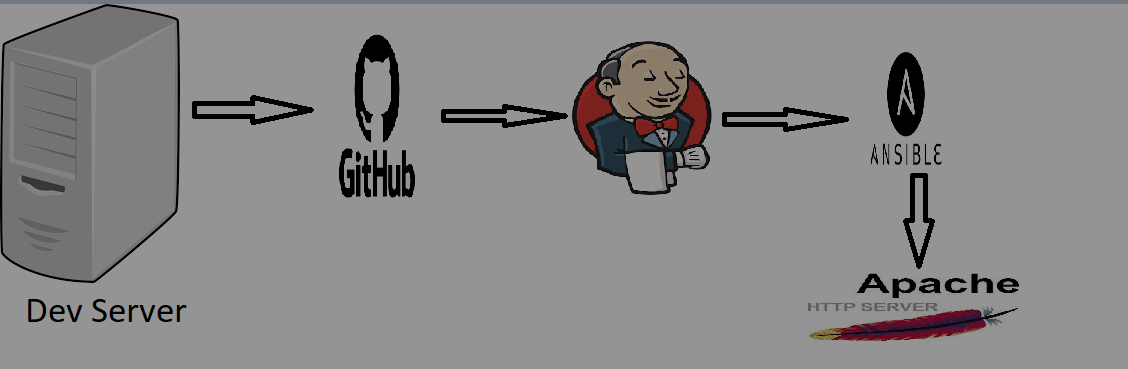
Prerequisites ………………………………………………………………………………….03

Integration of Tools in Jenkins ………………………………………………………..03

Process details……………………………………………………………………………04-20

Conclusion……………………………………………………………………………………….20

**Prerequisites:-**

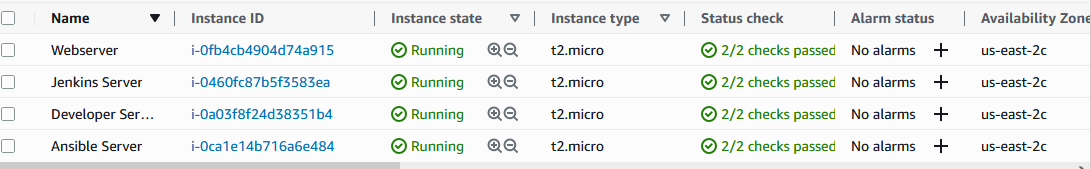
****

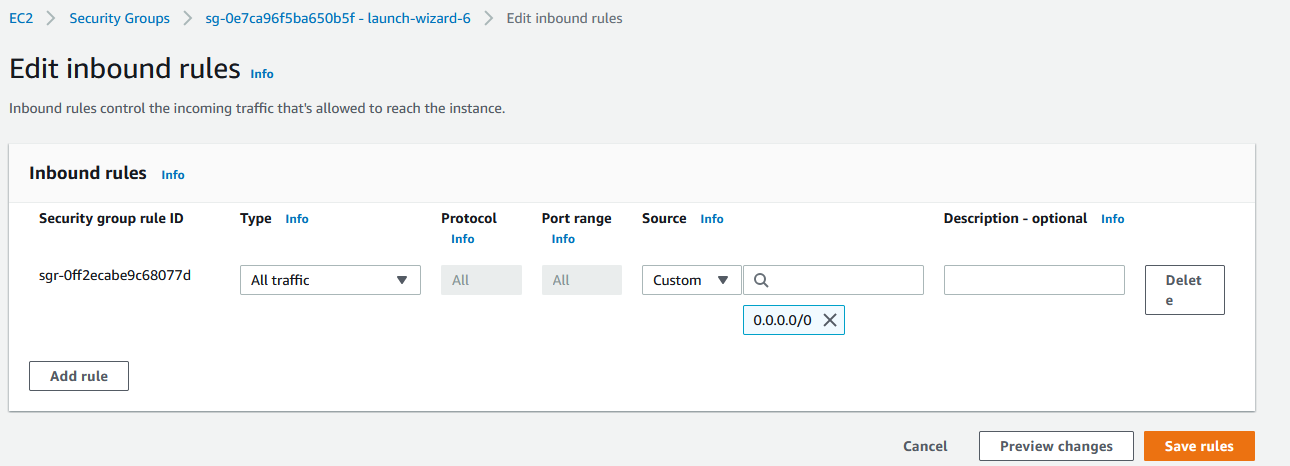
1. We need to have a Cloud Platform account Like ( GCP, AWS, or Azure )
2. GitHub account used as a version control
3. At least Four VM to Execute the Project.

* Developer Server ( 172.31.43.127 )
* Ansible Server (172.31.38.101 )
* Jenkins Server (172.31.46.73 )
* Webserver (172.31.39.171 )

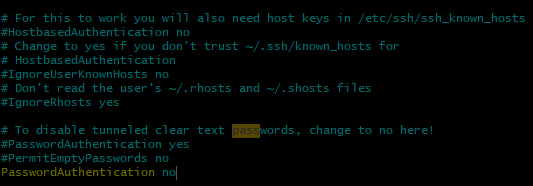
**Integration of Tools Jenkins:-**

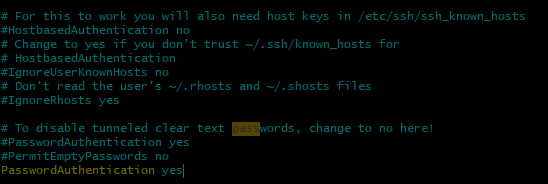
1. A Developer system that helps to use the codes
2. Integrating GitHub
3. Integrating Ansible
4. Integrating Webserver ( HTTP/ Apache2 )

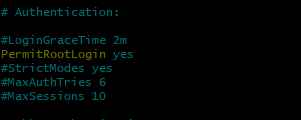




* Before we work on instances we need to uncommented PasswordAuthentication in Jenkins server, ansible server & webserver.
* Now login to that instances
* sudo su
* cd
* sudo vi /etc/ssh/sshd\_config ( it’s the configuration file where we have to allow the PasswordAuthentication. )
* And change the root password of all servers because we’ll use the root user of all servers.
* sudo passwd root
* And set the password of root ( LoveIndia )

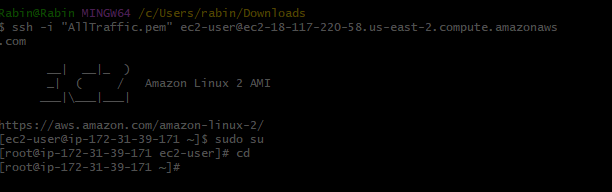






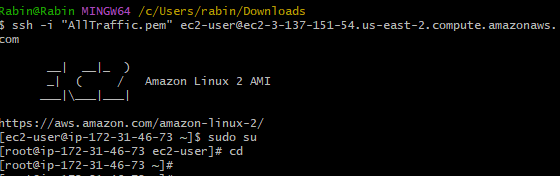
* After changes type ( esc !wq )
* sudo systemctl restart sshd ( To restart the services because we change the configuration file )

Now we’ll set up the **Developer VM (Amazon Linux)**:-

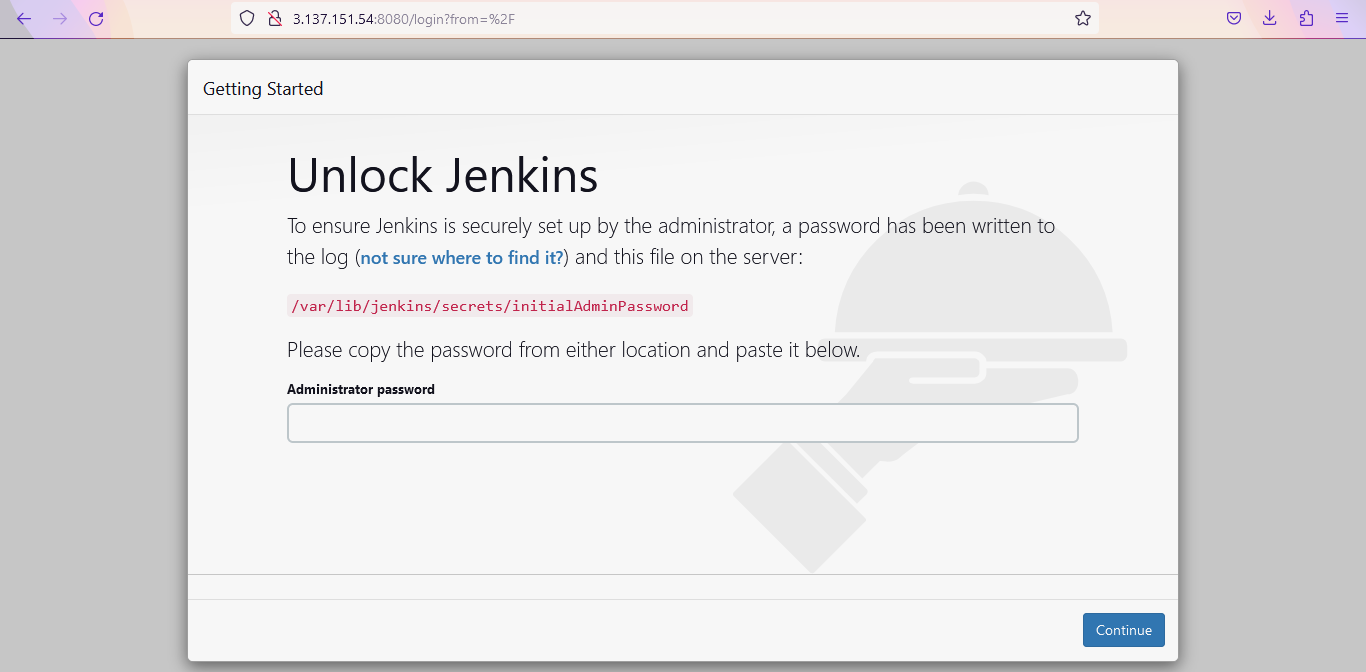


* sudo su
* cd
* yum install git –y
* yum update –y

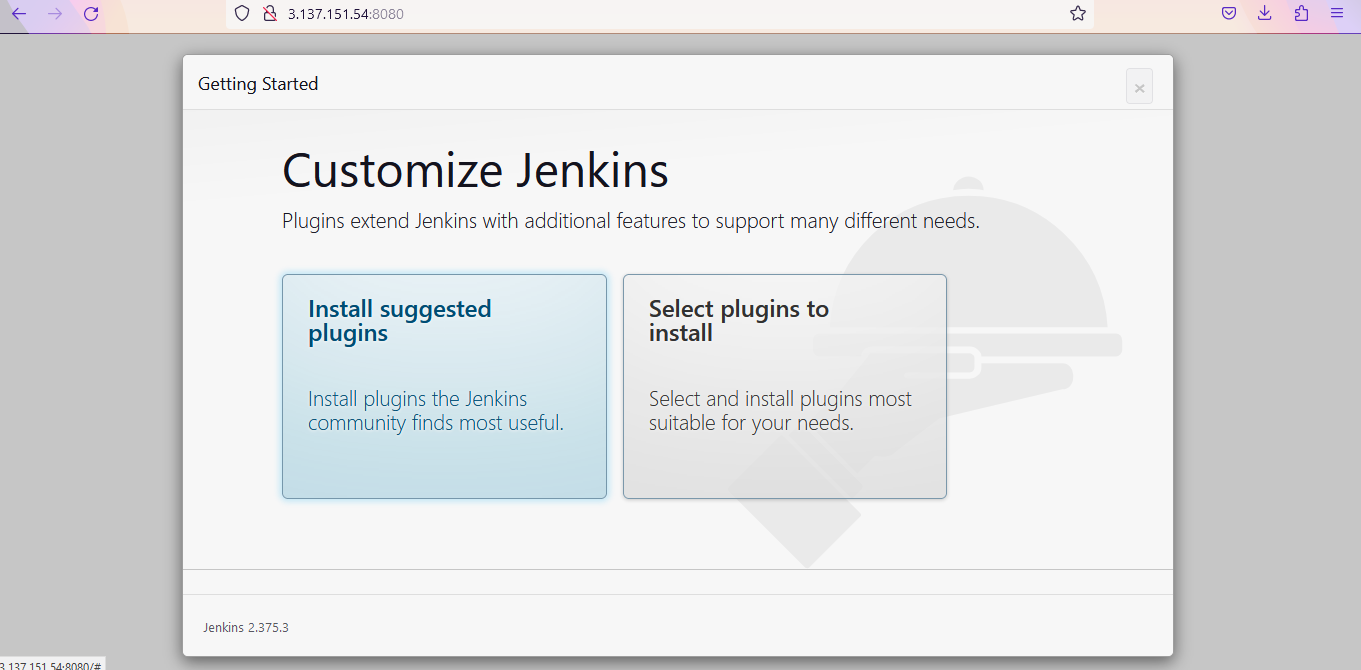
Now we’ll set up the **Jenkins Server (Amazon Linux)**:-



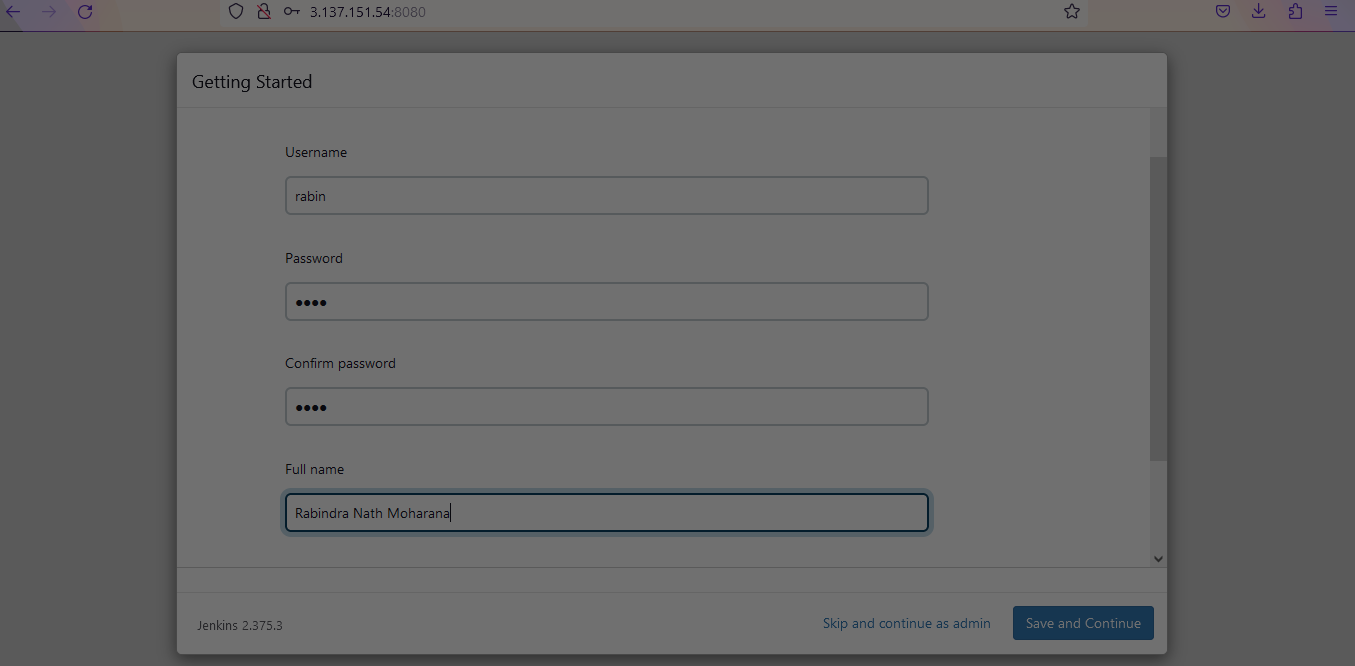
* sudo su
* cd
* yum install java\* -y
* yum install git –y
* wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
* rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
* yum install jenkins -y
* systemctl start jenkins
* systemctl enable Jenkins
* Now login to the Jenkins Dashboard
* Copy the public IP of the Instance
* Go to the browser and type in the search bar
* http:// 3.137.151.54:8080

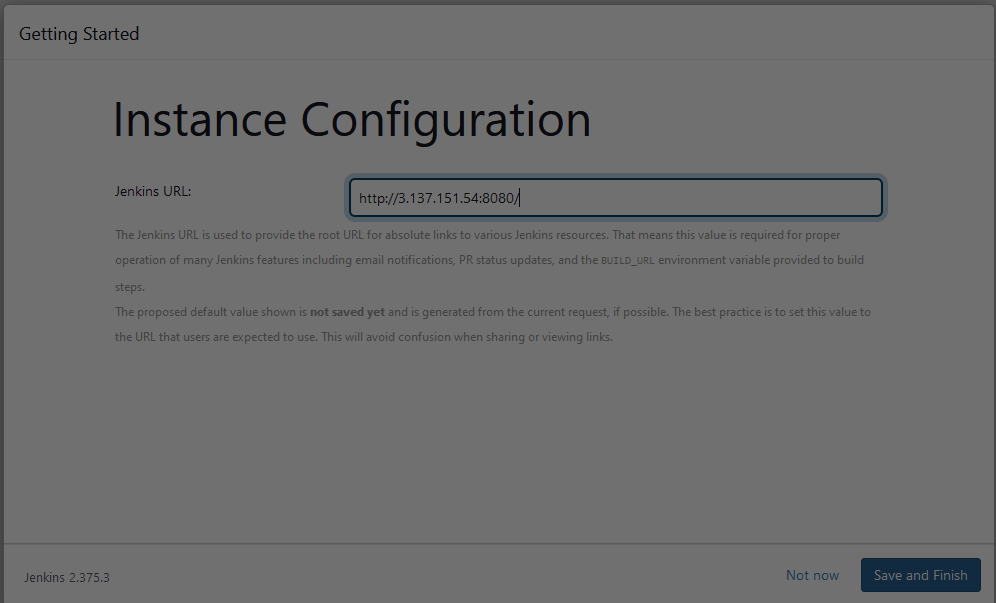


* cat /var/lib/jenkins/secrets/initialAdminPassword
* copy the password ( b4b44de38bd5455aa61c555f62409806 )

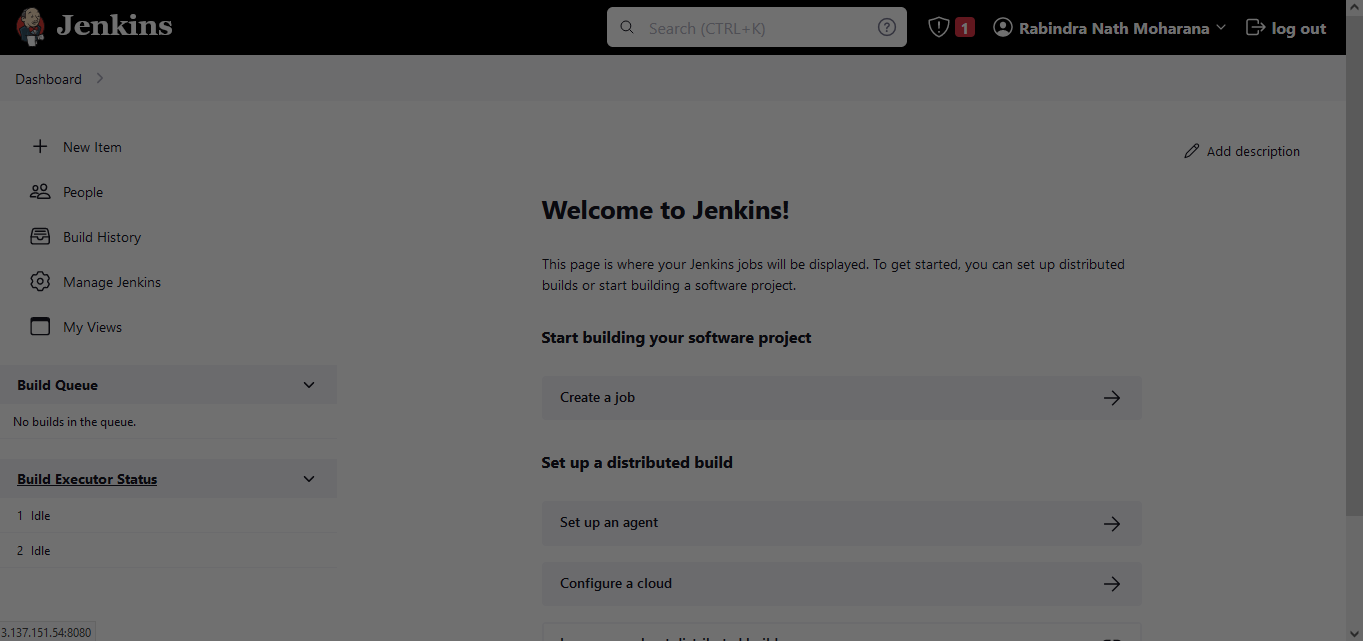


* Now select ( Install Suggested Plugins )
* Set the user name / Password/ Full Name

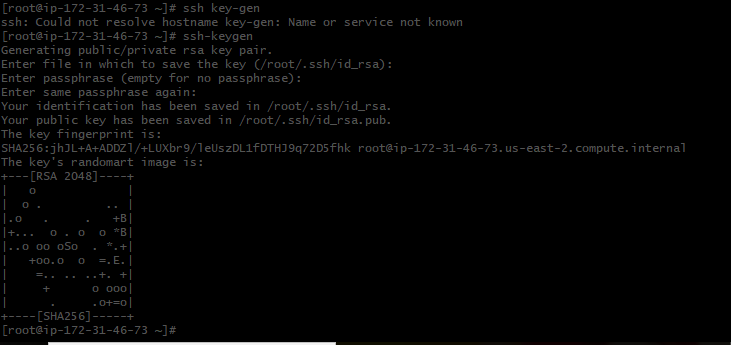




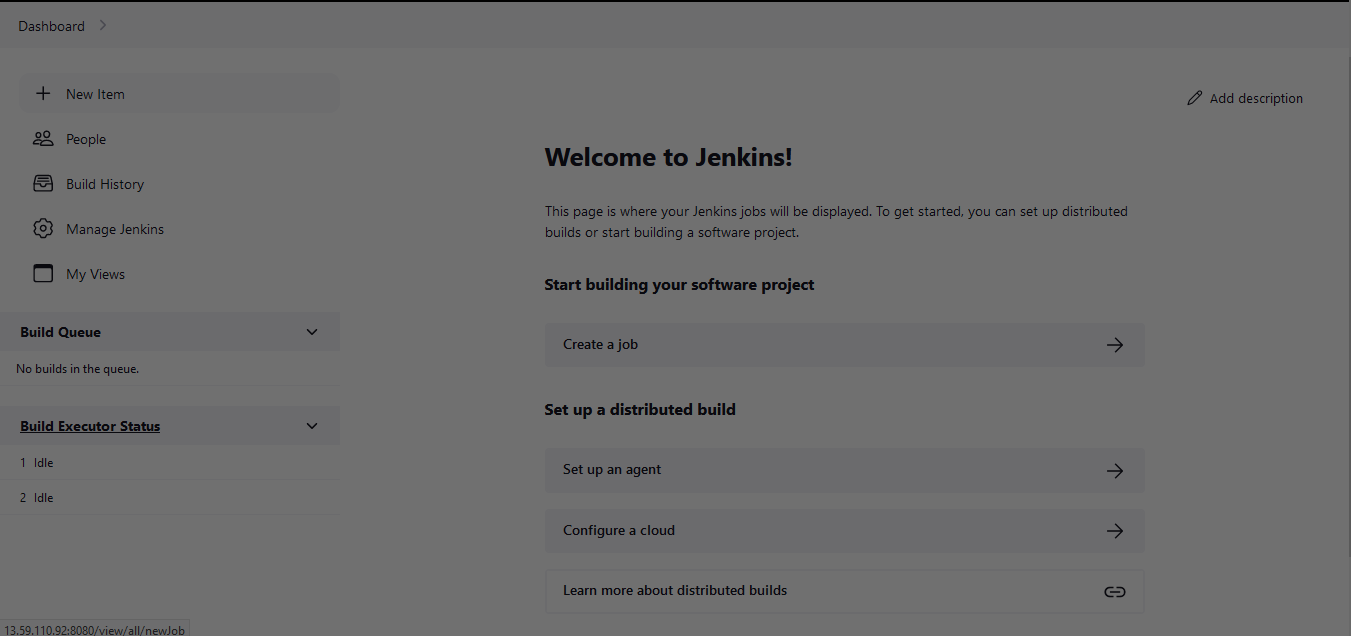
* Now the Jenkins Dashboard will open.

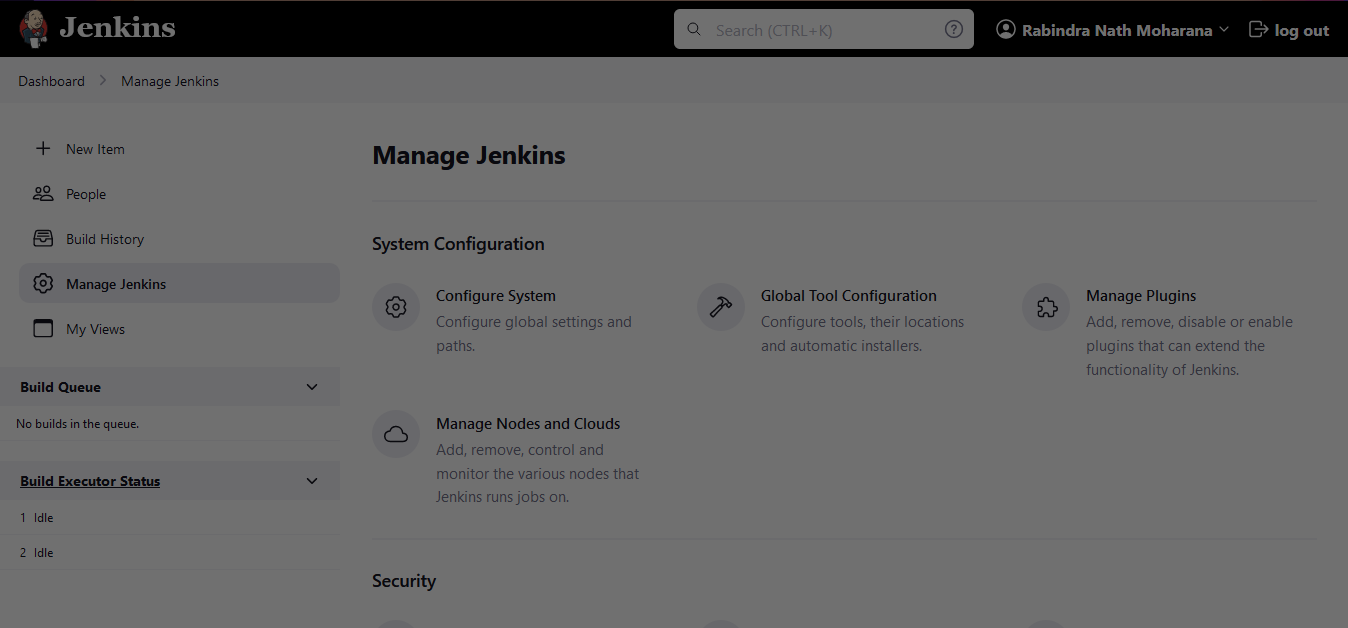


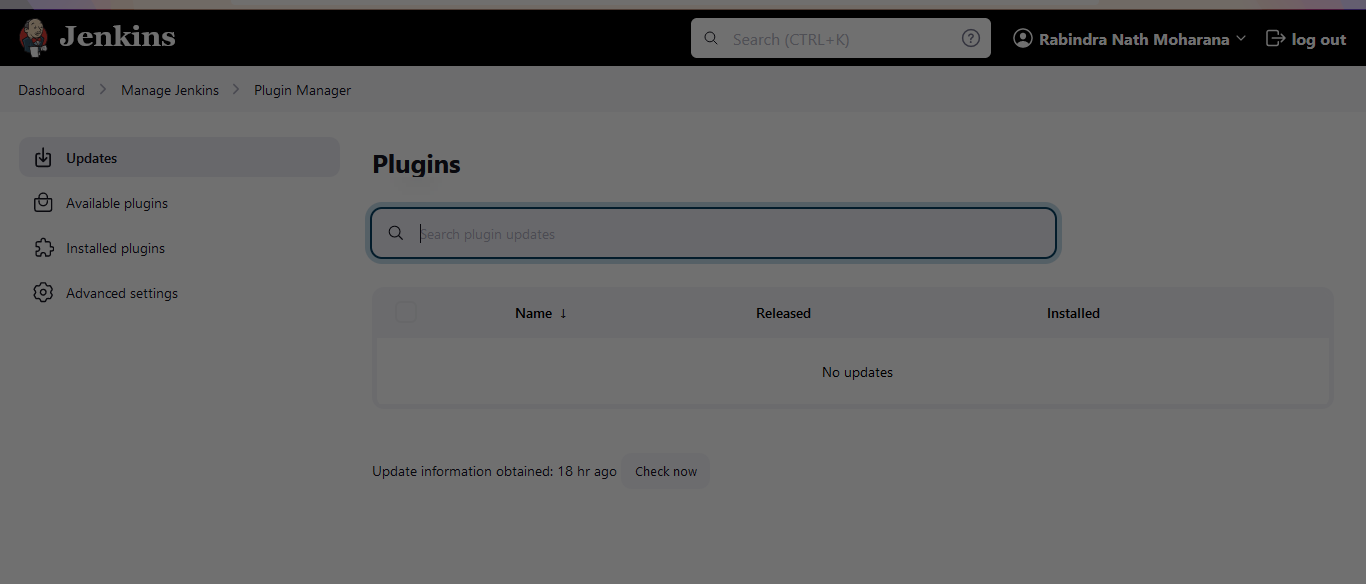
* Now we’ll set the ssh-key because we’ll connect the ansible and Jenkins.
* ssh-keygen

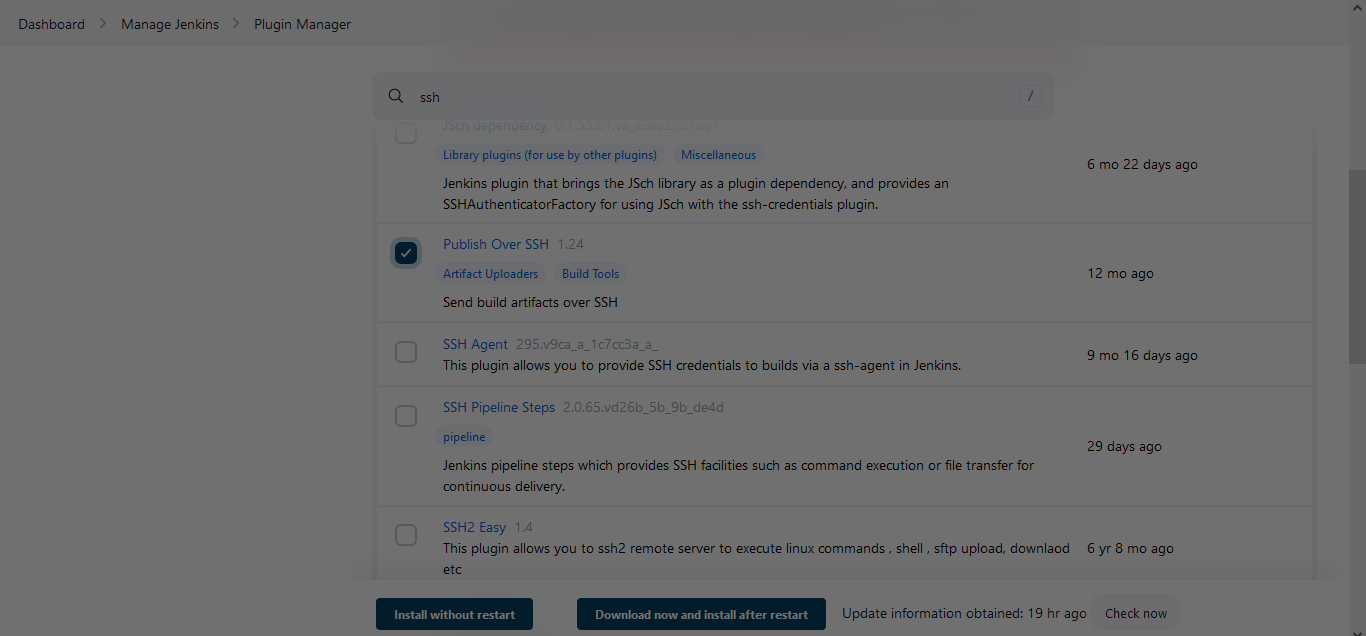


* Now we’ll pass the ssh key to our ansible server for password less authentication.
* Now login to the ansible server from Jenkins server to check whether the PasswordAuthentication is perfectly working.
* ssh-copy-id root@ 172.31.38.101 ( ansible server Pvt. IP )
* **Now we install publishovershh plugin for that we need to go Jenkins Dashboard**.
* **This plugin includes a builder which enables the use of the publisher during the build process**. This means that you can send newly created packages to a remote server and install them, start and stop services that the build may depend on and many other use cases.
* Manage Jenkins>plugin manager > available plugins> publishOverSSH > install with restart

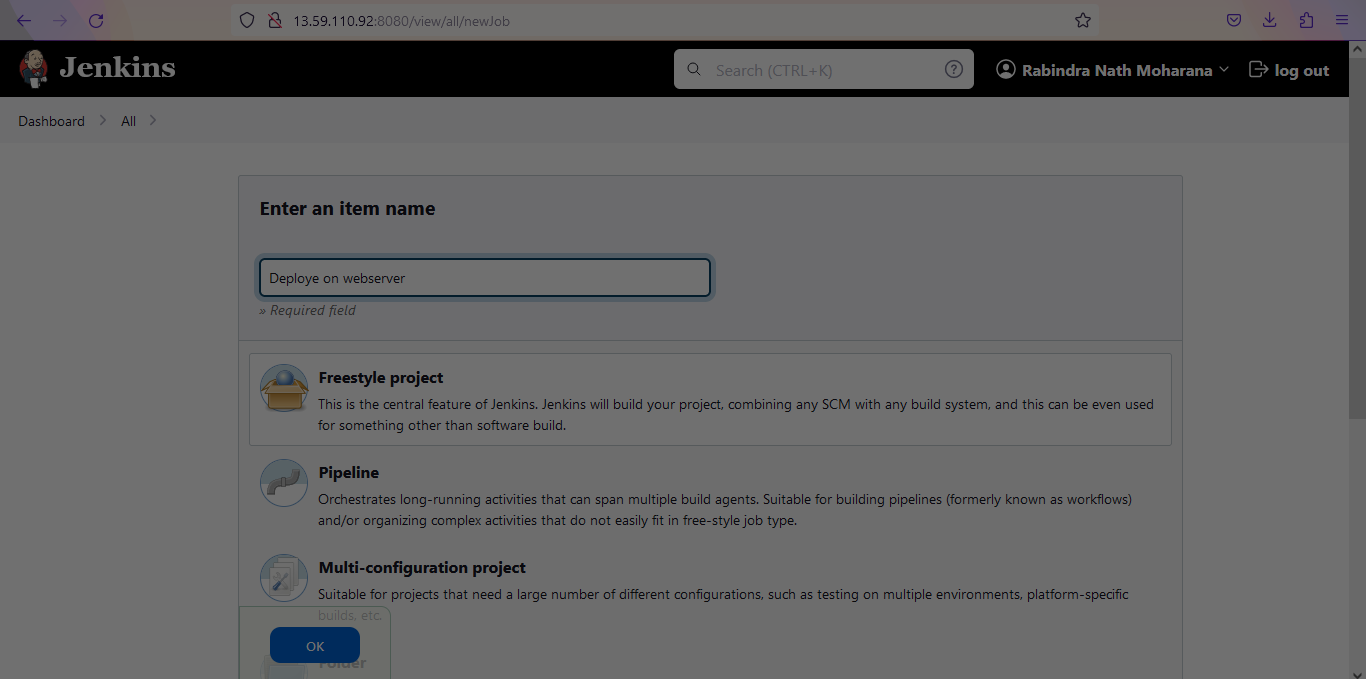


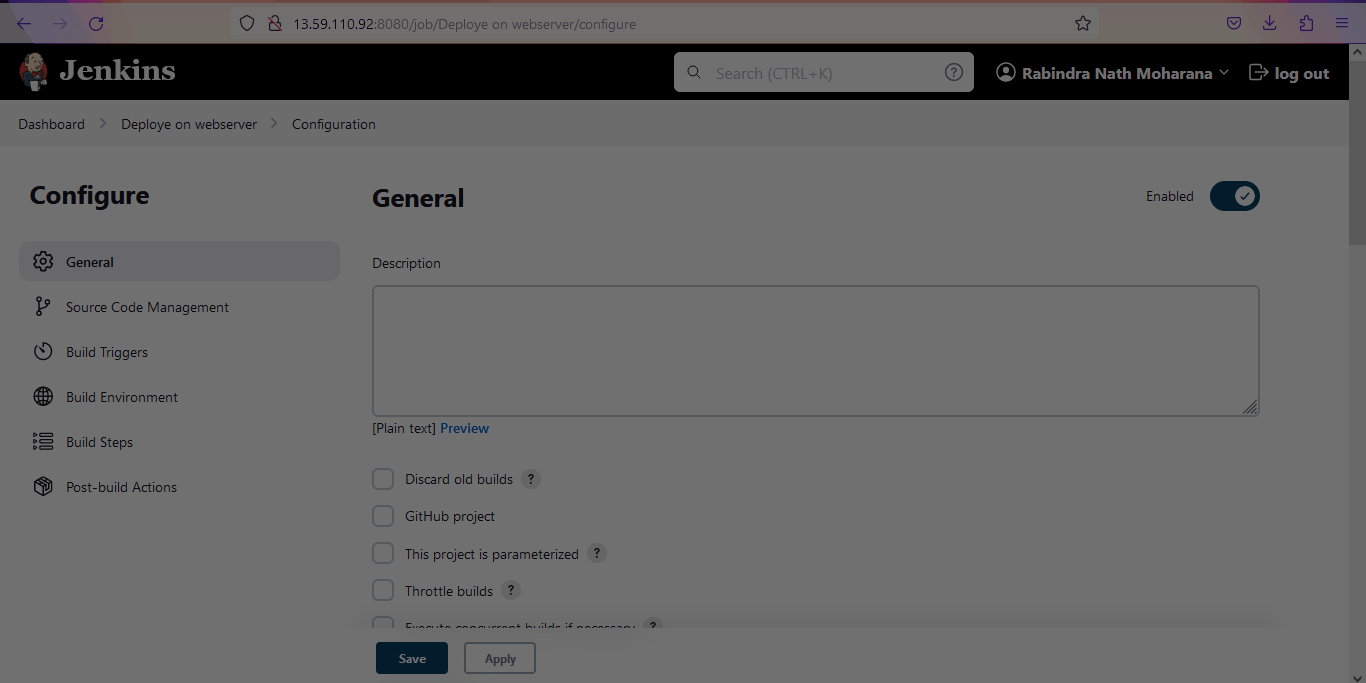






* Now we’ll create a Jenkins project for that new need to go New Item
* We’ll create a free style project.
* Jenkins freestyle projects **allow users to automate simple jobs, such as running tests, creating and packaging applications, producing reports, or executing commands**. Freestyle projects are repeatable and contain both build steps and post-build actions.



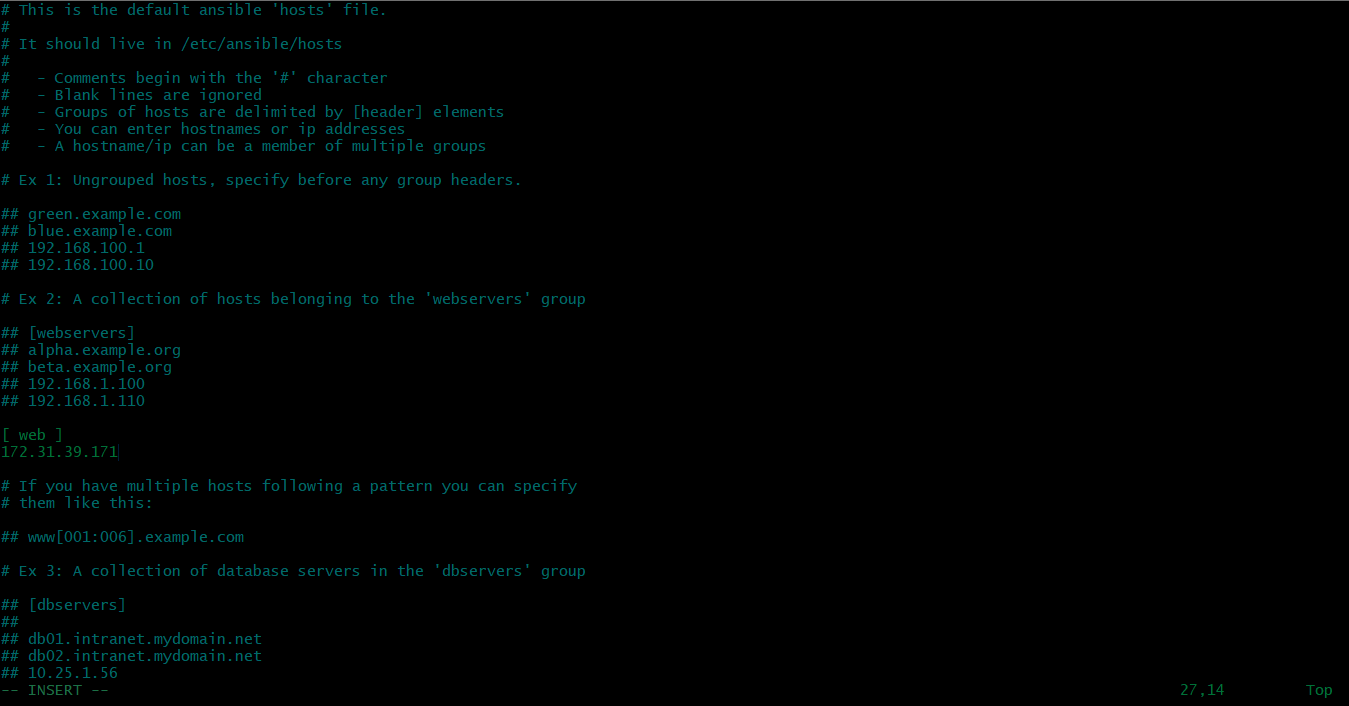


Now we’ll set up the **Ansible Server (Amazon Linux)**:-

* sudo su
* cd
* For ansible python is need so let’s check, whether python is available.
* python --version (Python 2.7.18 )
* sudo amazon-linux-extras install ansible2 –y
* sudo yum install ansible –y
* Now we’ll create a host file to manage our server.
* vi /etc/ansible/hosts

[webserver]

172.31.39.171 ( Pvt. Ip of webserver )



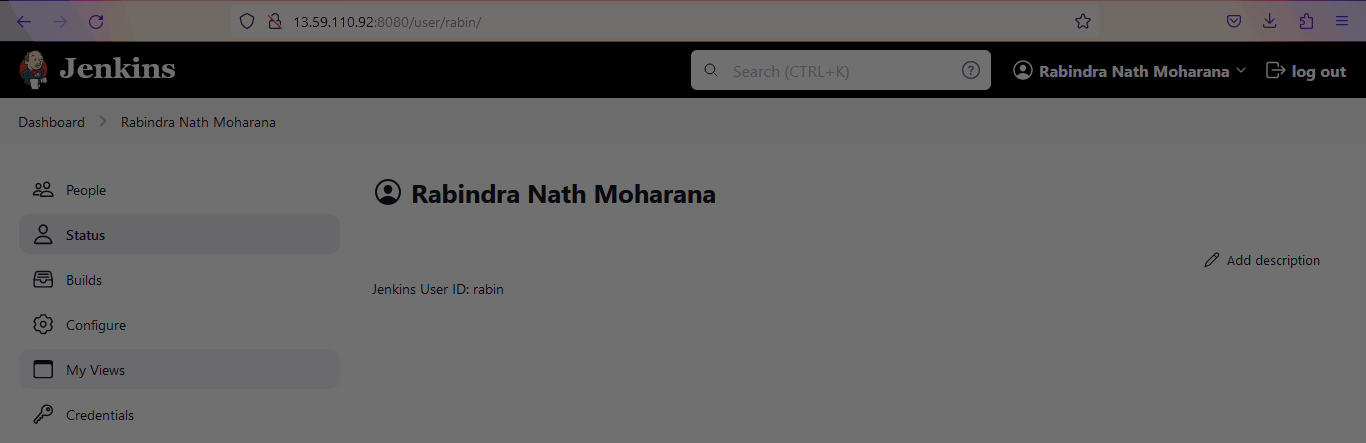
* :wq
* Now we’ll install the httpd web server, where we can deploy our code.
* ssh-copy-id [root@172.31.39.171](mailto:root@172.31.39.171) ( for password less authentication )
* exit

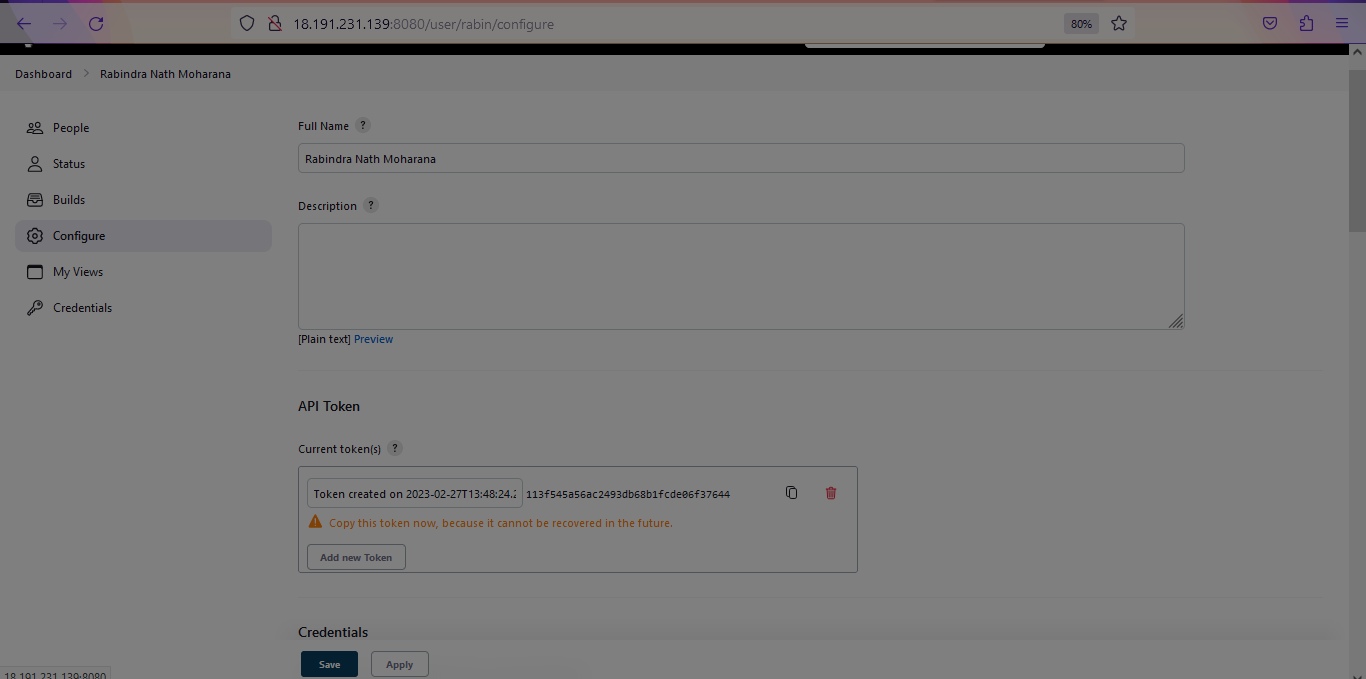
Now we’ll set up the **WebServer (Amazon Linux)**:-

* Login to the webserver
* sudo su
* cd
* sudo yum install httpd –y
* systemctl start httpd
* systemctl enable httpd

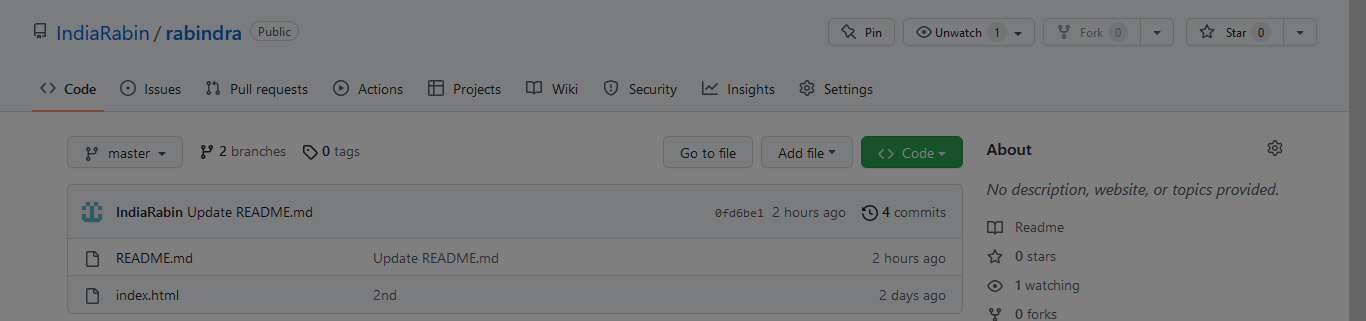
**Now login to the Jenkins Dashboard**

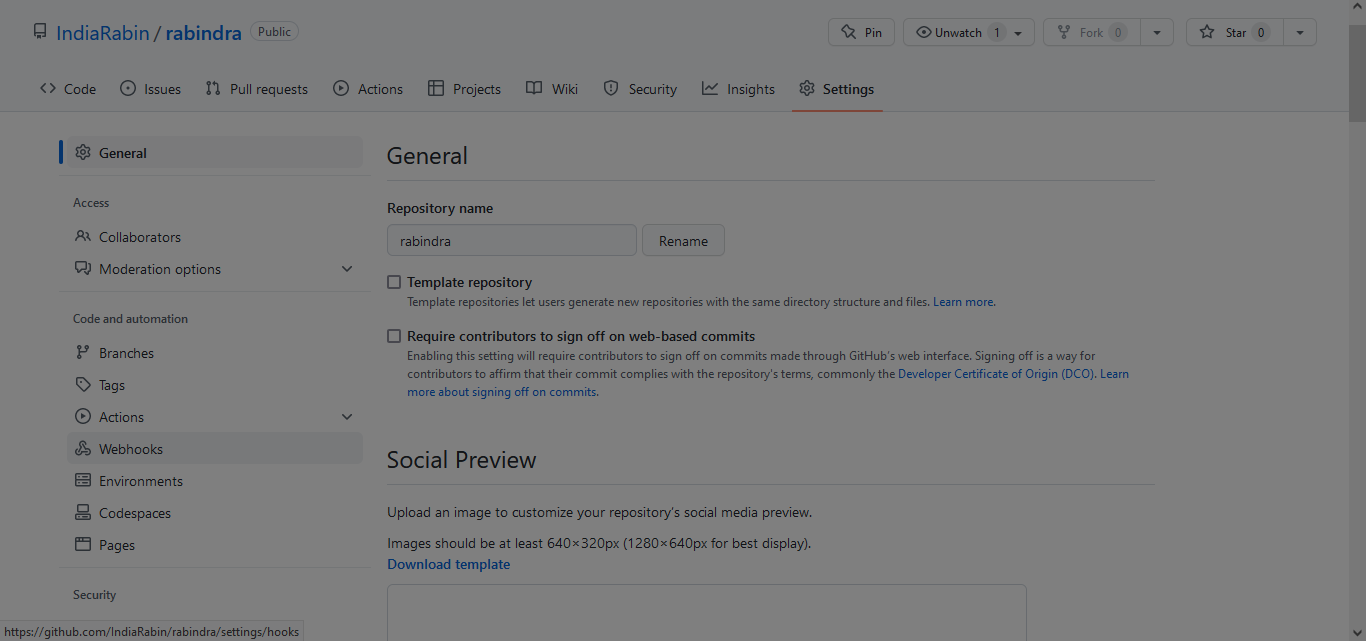
* Now we’ll generate the token in Jenkins to establish the communication between GitHub and Jenkins.
* Profile > configure > API Token > add new token> generate.

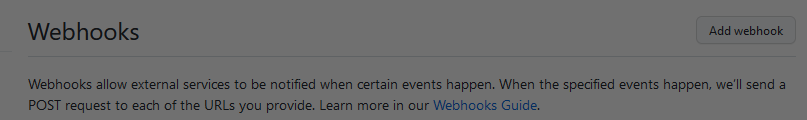


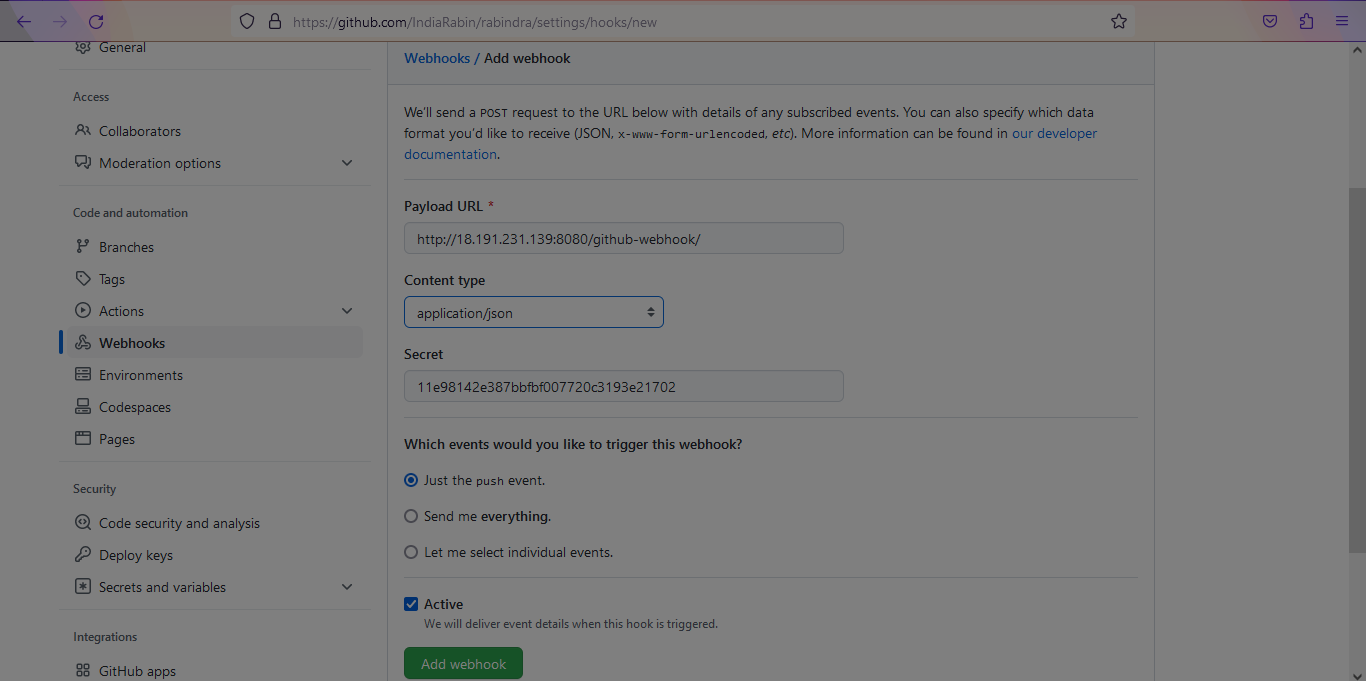


* Now copy the token and paste it into GitHub to automate the process.
* For that we need to go the GitHub account.
* Go to the repository and click on the setting.
* Click on add webhooks then add webhook.

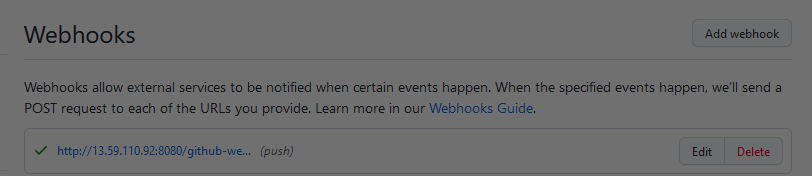




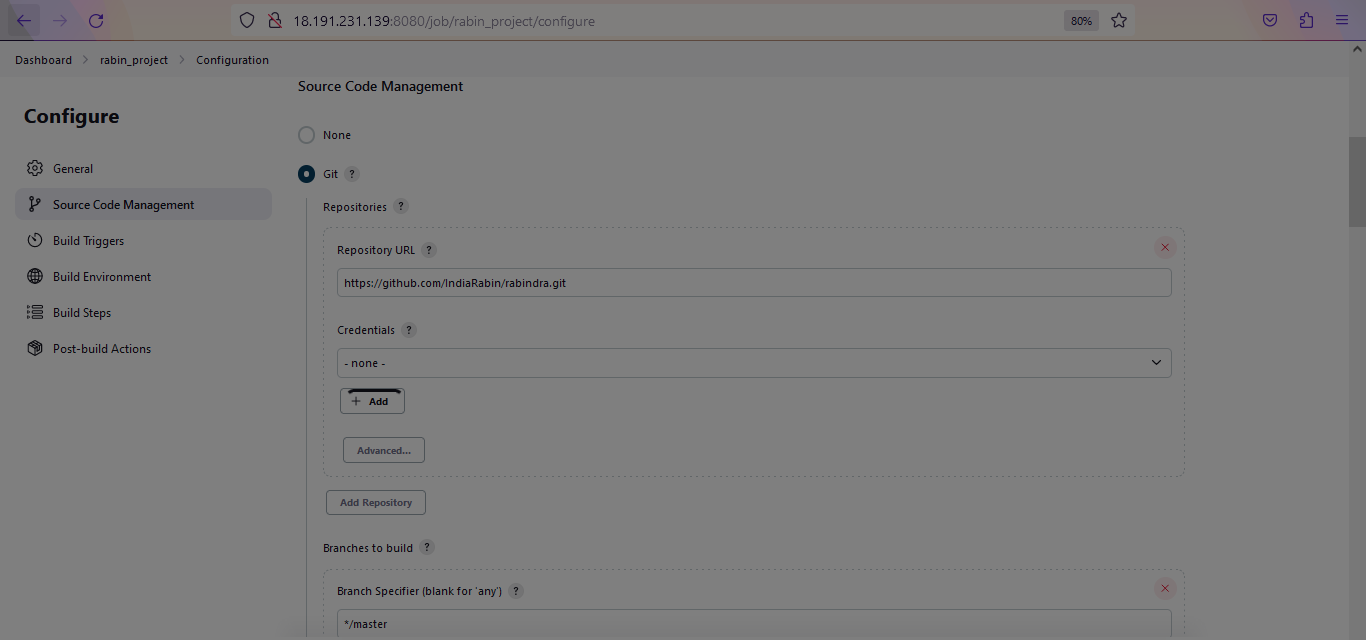




* Then refresh the page it’ll shows the green mark like below picture.



* Before we automate the project we need to do the configuration.



* For that we need to go to the project and configure that like ( source code management, Build trigger, Build Environment, Build steps, Post-build actions )
* In source code management we need to give the URL of GitHub.
* In Build Triggers we need to select Send files or execute commands over SSH

What is build steps?

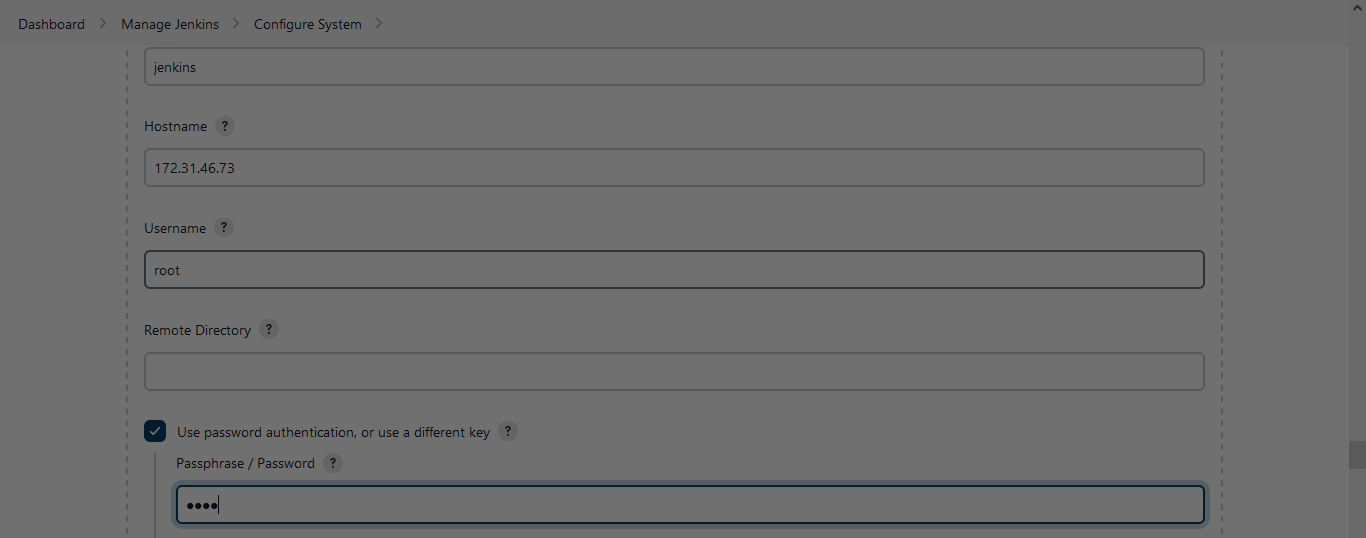
**Contains the configuration for automating a specific task or step in the application building process**.

* In post-build action we need to select send build artifacts over SSH

What is post build actions?

It **allows user to associate shell or a batch scripts that perform some tasks on Jenkins depending on the build log output**.

* Before that we need to do the connection SSH server in Jenkins and ansible.

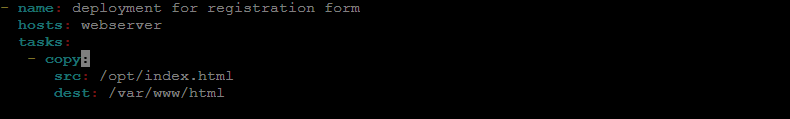




* Then apply and save

Now we will login into the ansible server:-

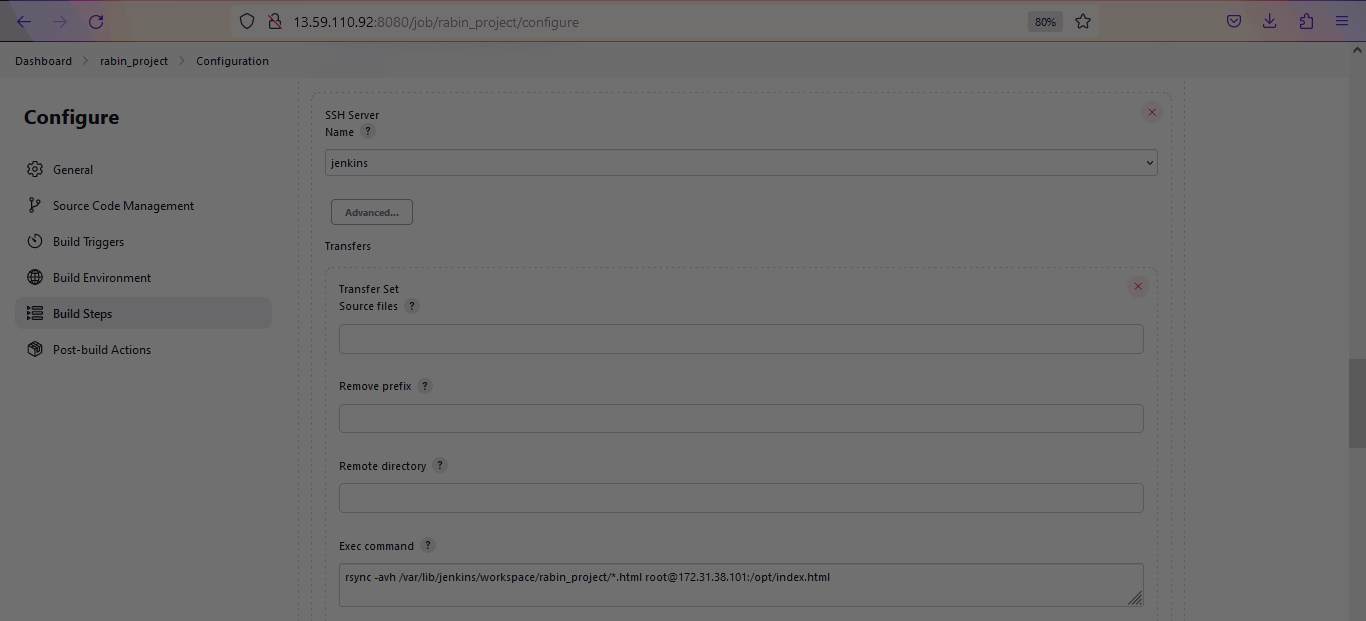
* create a directory in the of project
* mkdir deployment
* cd deployment
* now we’ll create a playbook inside this directory
* vi service.yml



* :wq

Now go to the Jenkins Dashboard:-

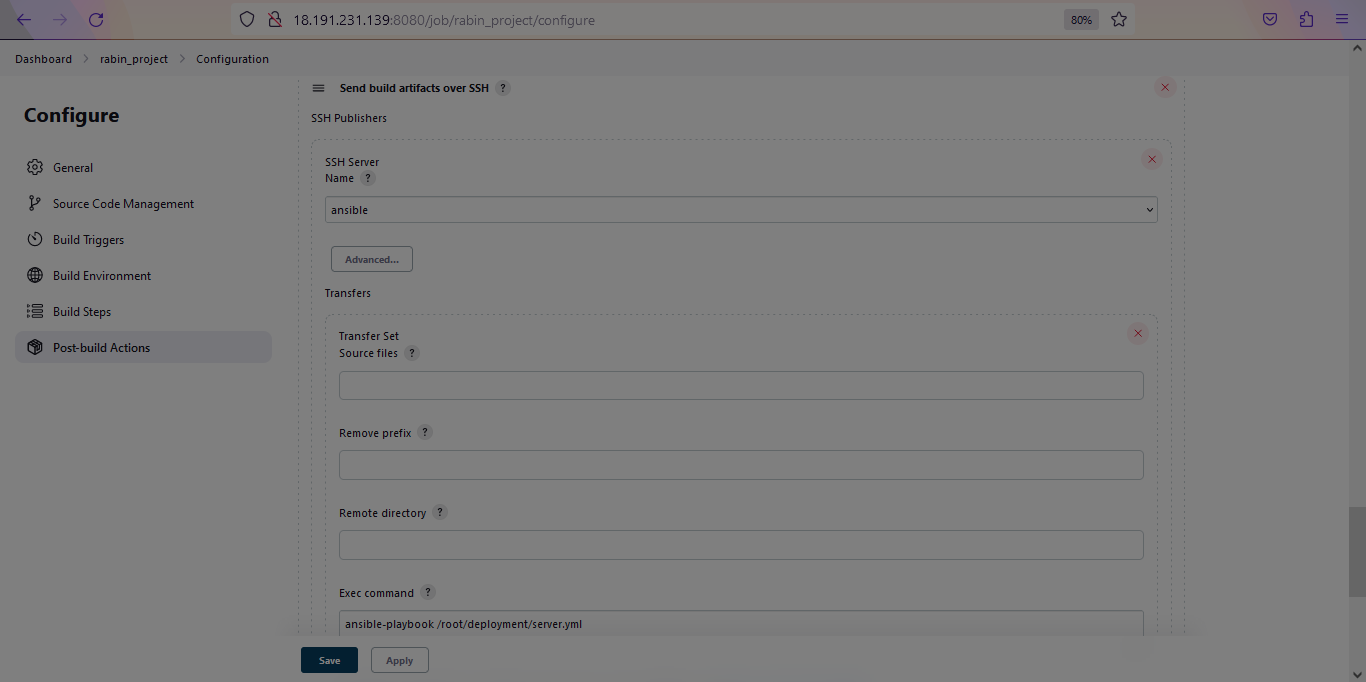
Build steps:

****

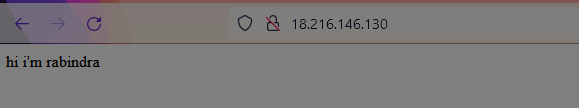
What is the purpose of rsync?

Rsync is a free software utility for Unix- and Linux-like systems that copies files and directories from one host to another.

Post-Build actions:



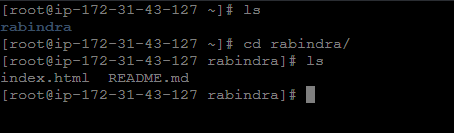
* Apply and Save
* Now we can build the Project.
* And check whether the web server is properly working. We have to copy the public IP of webserver and search in the web Brower server.



* If we want to change the code in the web browser. We have to go the Developer server and change it according to our need.

Now Login to the Developer server:-

* sudo su
* cd
* git config --global user.name “rabin”
* git config --global user.email [rabin@gmail.com](mailto:rabin@gmail.com)
* git branch –M master
* git clone <https://github.com/IndiaRabin/rabindra.git>



* vi index.html

<Html>

<head>

<title>

Registration Page

</title>

</head>

<body bgcolor="Lightskyblue">

<br>

<br>

<form>

<label> Firstname </label>

<input type="text" name="firstname" size="15"/> <br> <br>

<label> Middlename: </label>

<input type="text" name="middlename" size="15"/> <br> <br>

<label> Lastname: </label>

<input type="text" name="lastname" size="15"/> <br> <br>

<label>

Course :

</label>

<select>

<option value="Course">Course</option>

<option value="BCA">BCA</option>

<option value="BBA">BBA</option>

<option value="B.Tech">B.Tech</option>

<option value="MBA">MBA</option>

<option value="MCA">MCA</option>

<option value="M.Tech">M.Tech</option>

</select>

<br>

<br>

<label>

Gender :

</label><br>

<input type="radio" name="male"/> Male <br>

<input type="radio" name="female"/> Female <br>

<input type="radio" name="other"/> Other

<br>

<br>

<label>

Phone :

</label>

<input type="text" name="country code" value="+91" size="2"/>

<input type="text" name="phone" size="10"/> <br> <br>

Address

<br>

<textarea cols="80" rows="5" value="address">

</textarea>

<br> <br>

Email:

<input type="email" id="email" name="email"/> <br>

<br> <br>

Password:

<input type="Password" id="pass" name="pass"> <br>

<br> <br>

Re-type password:

<input type="Password" id="repass" name="repass"> <br> <br>

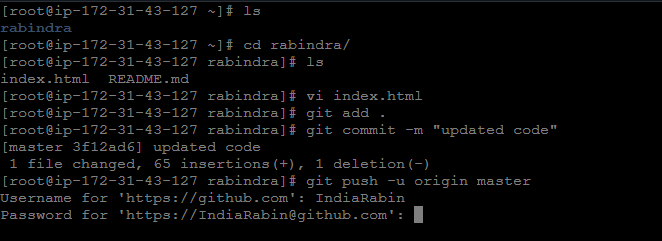
<input type="button" value="Submit"/>

</form>

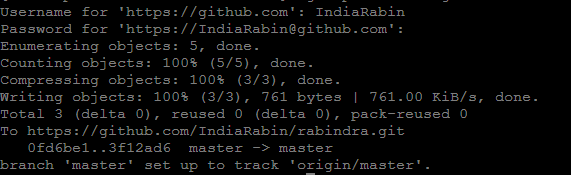
</body>

</html>

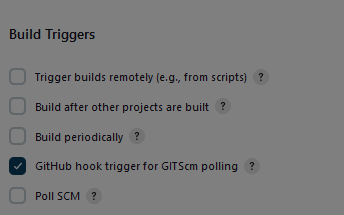
:wq



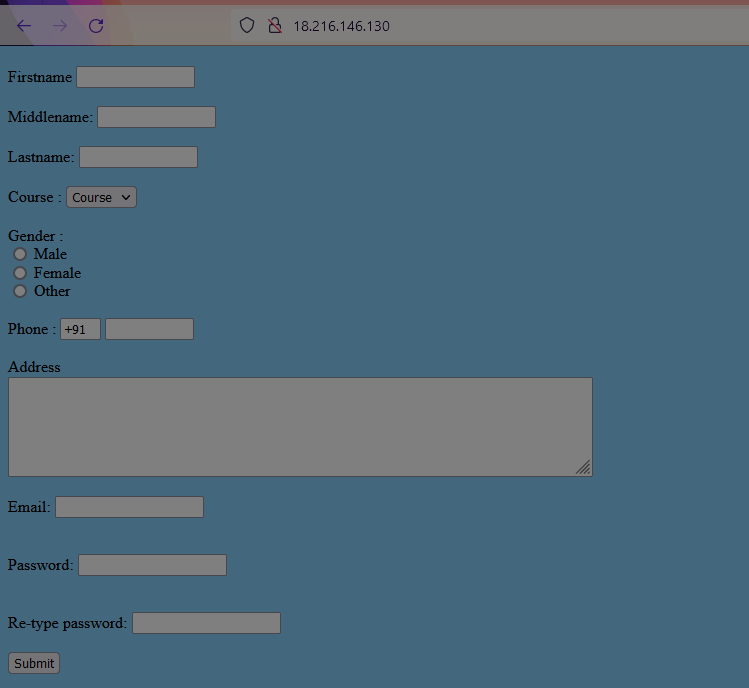
* In password we need to create the git token from GitHub.
* Then copy the token from the GitHub, use the token instead of password.



* After we pushed the code to GitHub.
* Jenkins pipeline will trigger automatically because we did Webhooks trigger



* Now we’ll see the updates in the web-server
* Copy the public IP and search in the browser.



* As per our code it’ll show like this.

Conclusion:-

In this project, we learned the concept of automating the Jenkins pipeline. Integrate the tools through ssh connection in Jenkins.