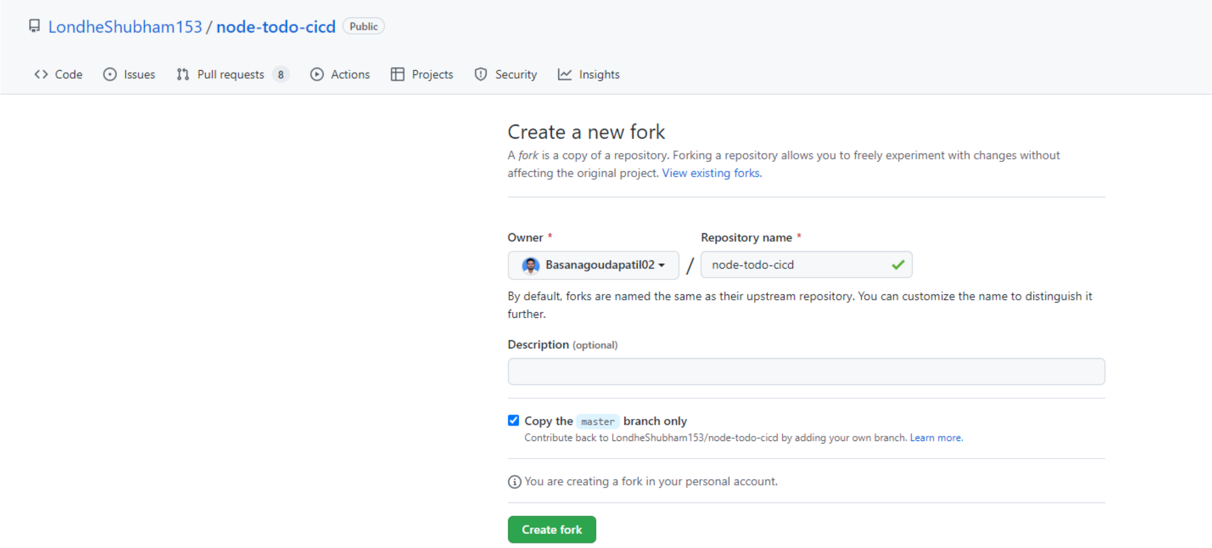
Day 24 Task: Complete Jenkins CI/CD Project

Let's make a CI/CD Pipeline for your Node JS Application

Task-01

- Fork (https://github.com/LondheShubham153/node-todo-cicd.git) repository:



Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. View existing forks.

Owner * Basanagoudapatil02 / Repository name * node-todo-cicd ✓

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

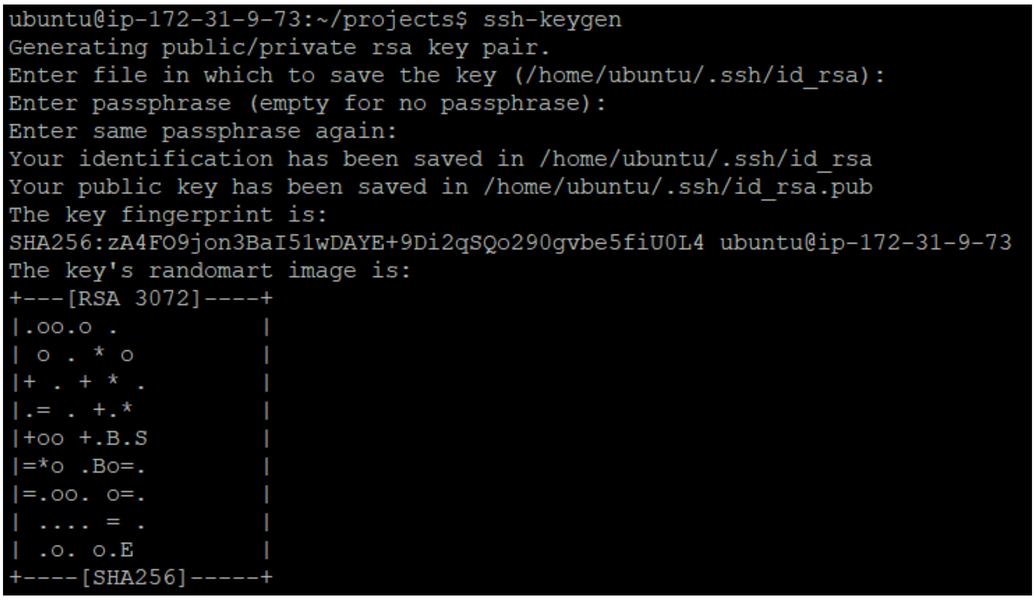
Description (optional)

☑ Copy the `master` branch only
Contribute back to LondheShubham153/node-todo-cicd by adding your own branch. Learn more.

ⓘ You are creating a fork in your personal account.

- Generate the SSH keys for integrating your Jenkins project with your git repository. Use ssh-keygen command to create public and private key.

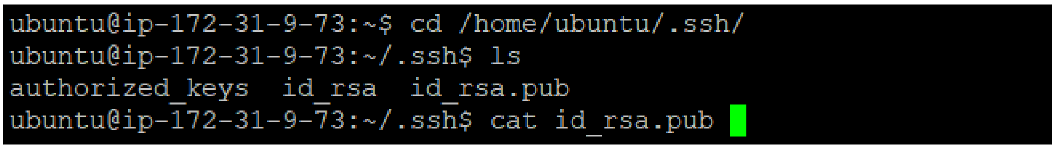
○ To Generate the Public and Private key For EC2
ssh-keygen
cd .ssh



```
ubuntu@ip-172-31-9-73:~/projects$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ubuntu/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ubuntu/.ssh/id_rsa
Your public key has been saved in /home/ubuntu/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:zA4FO9jon3BaI51wDAYE+9Di2qSQo290gvbe5fiU0L4 ubuntu@ip-172-31-9-73
The key's randomart image is:
+---[RSA 3072]----+
|.oo.o .          |
| o . * o         |
|+ . + * .        |
|.= . +.*         |
|+oo +.B.S        |
|=*o .Bo=.        |
|=.oo. o=.        |
| .... = .        |
| .o. o.E         |
+----[SHA256]-----+
```

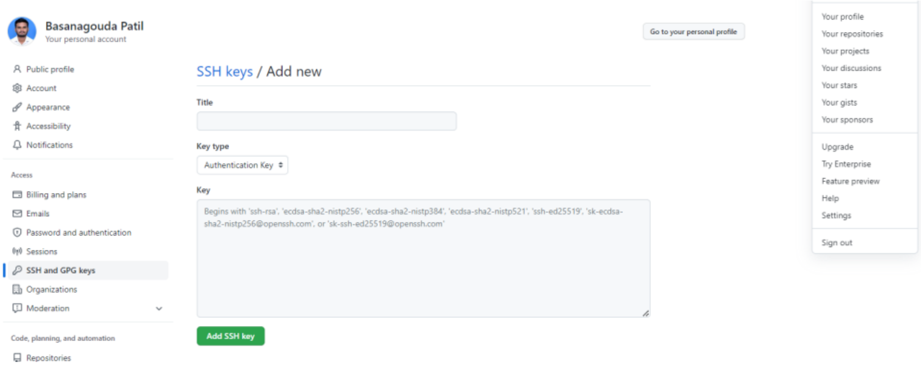
- As now we have generated the keys as shown below , so public key is needed to be given to SSH Key Generation in Github , So Github Can connect to your Server

  id_rsa.pub — Public key
  id_rsa — Private key

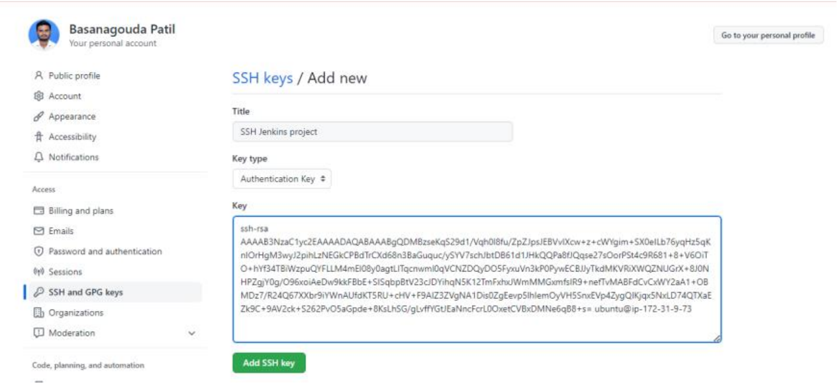


```
ubuntu@ip-172-31-9-73:~$ cd /home/ubuntu/.ssh/
ubuntu@ip-172-31-9-73:~/.ssh$ ls
authorized_keys  id_rsa  id_rsa.pub
ubuntu@ip-172-31-9-73:~/.ssh$ cat id_rsa.pub
```

- **Configuring GitHub:**
  - Go to your GitHub account settings.



  - Go to SSH and GPG keys, Add public key that we created using ssh-keygen and select key-type Authentication key.
    cat id_rsa.pub



Title: SSH Jenkins project

Key type: Authentication Key

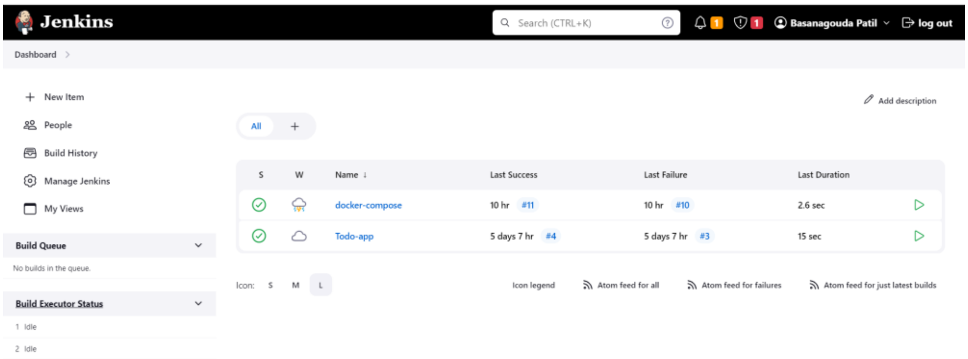
Key:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQDMBzseKqS29d1/Vqh0l8fu/ZpZJpsJEBVvlXcw+z+cWYgim+SX0eILb76yqHz5qK
nlOrHgM3wyJ2pihLzNEGkCPBdTrCXd68n3BaGuquc/ySYV7schJbtDB61d1JHkQQPa8fJQqse27sOorPSt4c9R681+8+V6OiT
O+hYf34TBiWzpuQYFLLM4mEI08y0agtLITqcnwml0qVCNZDQyDO5FyxuVn3kP0PywECBJJyTkdMKVRiXWQZNUGrX+8J0N
HPZgjY0g/O96xoiAeDw9kkFBbE+SISqbpBtV23cJDYihqN5K12TmFxhxJWmMMGxmfsIR9+nefTvMABFdCvCxWY2aA1+OB
MDz7/R24Q67XXbr9iYWnAUfdKT5RU+cHV+F9AIZ3ZVgNA1Dis0ZgEevp5lhlemOyVH5SnxEVp4ZygQIKjqx5NxLD74QTXaE
Zk9C+9AV2ck+S262PvO5aGpde+8KsLhSG/gLvffYGtJEaNncFcrL0OxetCVBxDMNe6qB8+s= ubuntu@ip-172-31-9-73
```

  - Thus with above command you will get Public key and add this public key in “Key” Section of “SSH & GPG Keys”

- Create a connection to your Jenkins job and your GitHub Repository via GitHub Integration.

1. Open Jenkins and Click on create a job.



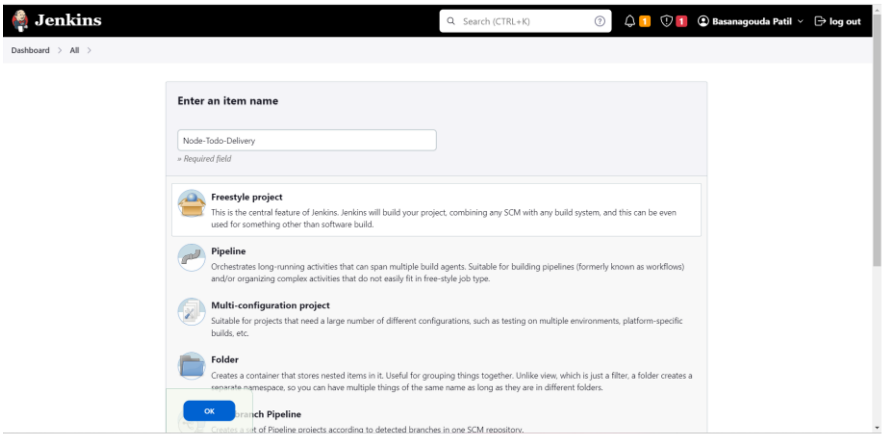
No builds in the queue.

1 Idle

2 Idle

S	W	Name ↓	Last Success	Last Failure	Last Duration
✓		docker-compose	10 hr #11	10 hr #10	2.6 sec
✓		Todo-app	5 days 7 hr #4	5 days 7 hr #3	15 sec

2. Enter name and click on freestyle project and finally click on ok.



Enter an item name

Node-Todo-Delivery

» Required field

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

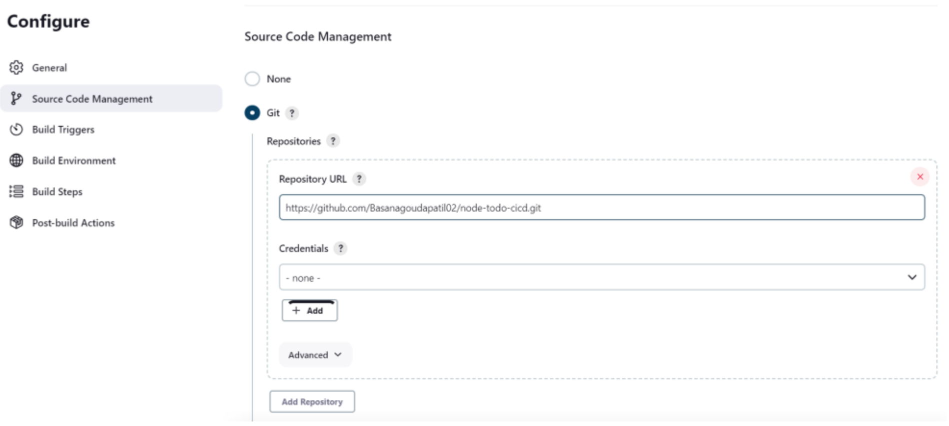
Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

ranch Pipeline
Creates a set of Pipeline projects according to detected branches in one SCM repository.

3. Provide github url and click on Add icon to add jenkins credentials.



Source Code Management

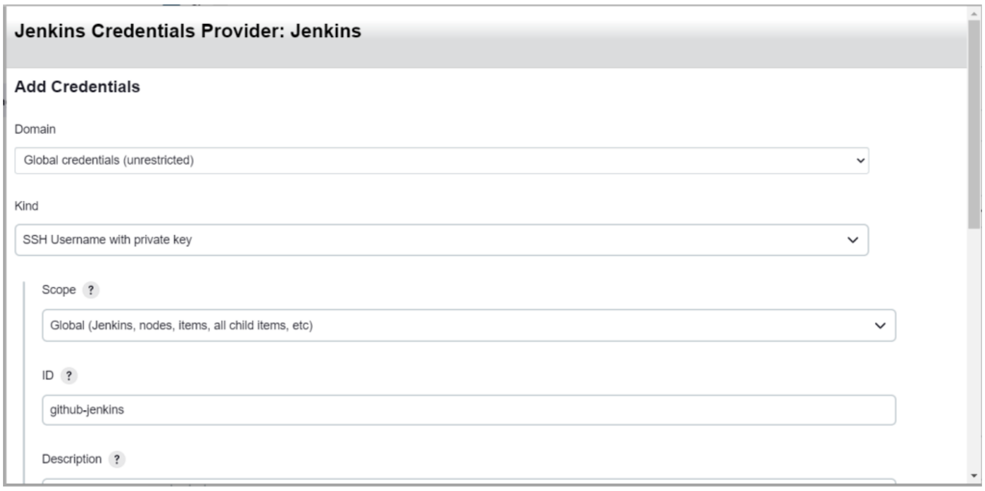
○ None

● Git

Repositories

Repository URL: https://github.com/Basanagoudapatil02/node-todo-cicd.git

Credentials: - none -



Jenkins Credentials Provider: Jenkins

Add Credentials

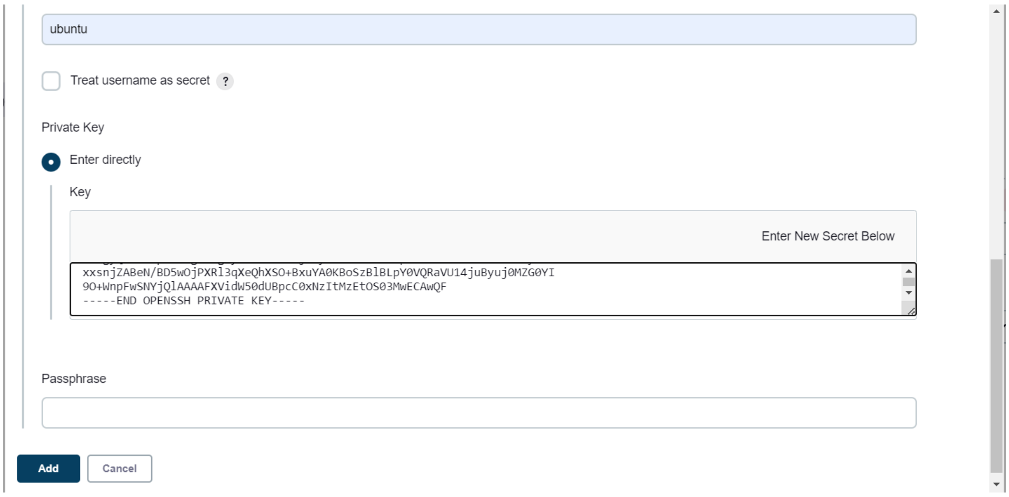
Domain: Global credentials (unrestricted)

Kind: SSH Username with private key

Scope: Global (Jenkins, nodes, items, all child items, etc)

ID: github-jenkins

Description:



ubuntu

□ Treat username as secret

Private Key

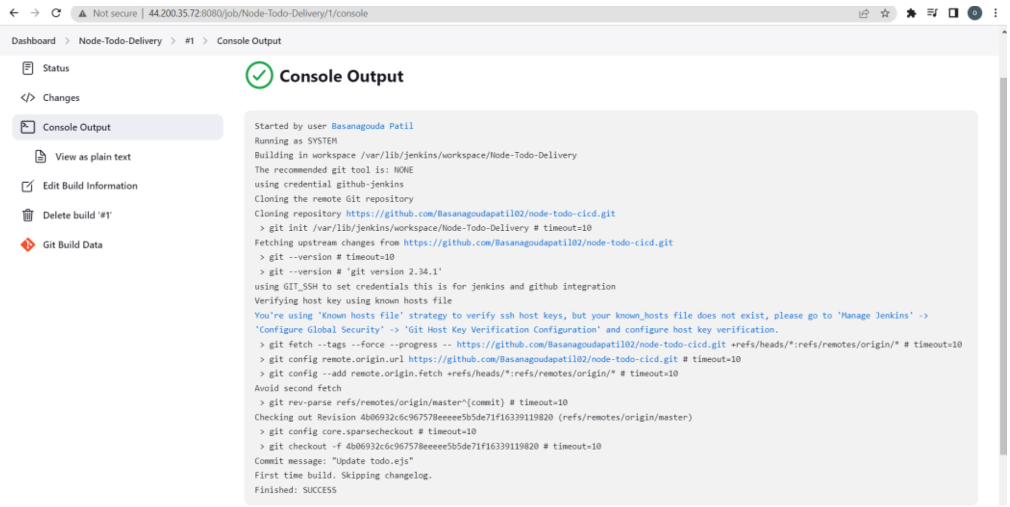
● Enter directly

Key

```
xxsnjZABeN/BD5wOjPXRl3qXeQhXSO+BxuYA0KBoSzBlBLpY0VQRaVU14juByuj0MZG0YI
9O+WnpFwSNYjQlAAAAFXVidW50dUBpcC0xNzItMzEtOS03MwECAwQF
-----END OPENSSH PRIVATE KEY-----
```

Passphrase

4. Click on Save button and the job will appear on dashboard screen.

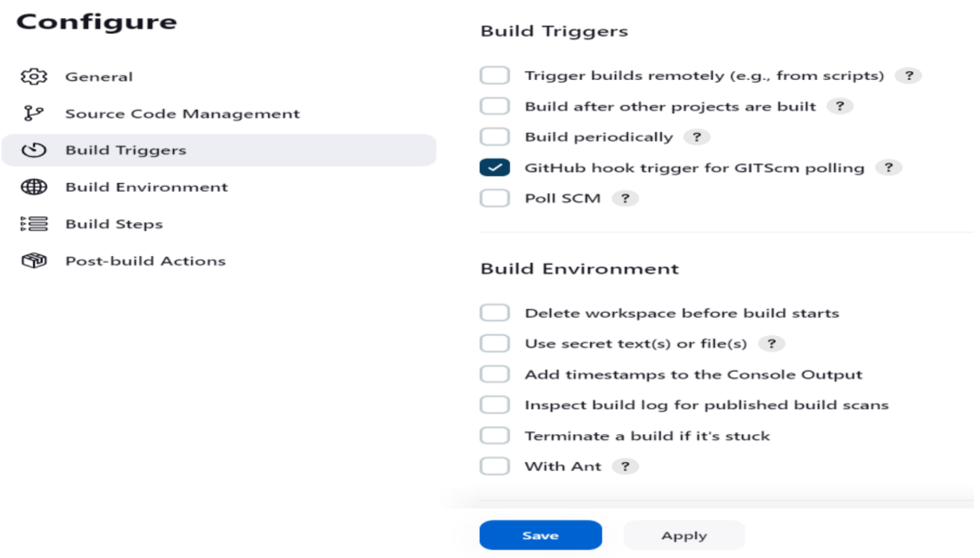

Not secure | 44.200.35.72:8080/job/Node-Todo-Delivery/1/console
Dashboard > Node-Todo-Delivery > #1 > Console Output
Status
Changes
Console Output
View as plain text
Edit Build Information
Delete build '#1'
Git Build Data

Console Output

```
Started by user Basanagouda Patil
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/Node-Todo-Delivery
The recommended git tool is: NONE
using credential github-jenkins
Cloning the remote Git repository
Cloning repository https://github.com/Basanagoudapatil02/node-todo-cicd.git
 > git init /var/lib/jenkins/workspace/Node-Todo-Delivery # timeout=10
Fetching upstream changes from https://github.com/Basanagoudapatil02/node-todo-cicd.git
 > git --version # timeout=10
 > git --version # 'git version 2.34.1'
using GIT_SSH to set credentials this is for jenkins and github integration
Verifying host key using known hosts file
You're using 'Known hosts file' strategy to verify ssh host keys, but your known_hosts file does not exist, please go to 'Manage Jenkins' ->
'Configure Global Security' -> 'Git Host Key Verification Configuration' and configure host key verification.
 > git fetch --tags --force --progress -- https://github.com/Basanagoudapatil02/node-todo-cicd.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git config remote.origin.url https://github.com/Basanagoudapatil02/node-todo-cicd.git # timeout=10
 > git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
Checking out Revision 4b06932c6c967578eeeee5b5de71f16339119820 (refs/remotes/origin/master)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 4b06932c6c967578eeeee5b5de71f16339119820 # timeout=10
Commit message: "Update todo.ejs"
First time build. Skipping changelog.
Finished: SUCCESS
```

5. Read about webhook and make sure you have CICD setup

- Click on the ‘Build Triggers’ tab and then on the ‘GitHub hook trigger for GITScm polling’.



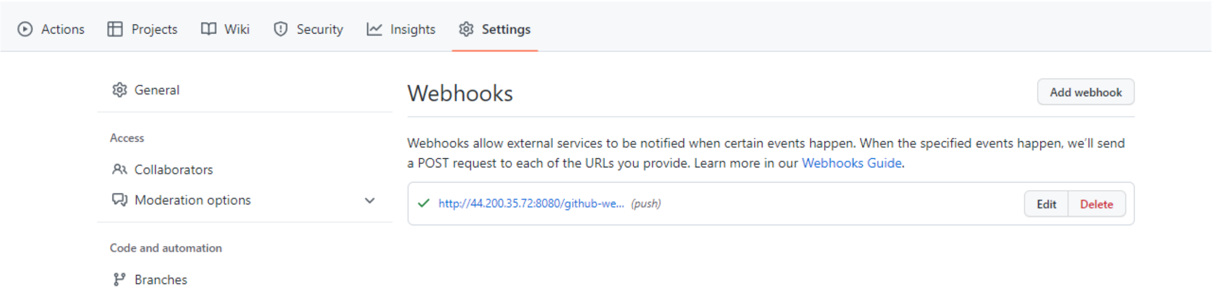
Build Triggers

- ☐ 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) ?
- ☐ Add timestamps to the Console Output
- ☐ Inspect build log for published build scans
- ☐ Terminate a build if it's stuck
- ☐ With Ant ?

- Once the connection between the Jenkins server and Github is successful a green tick as highlighted below will be shown.



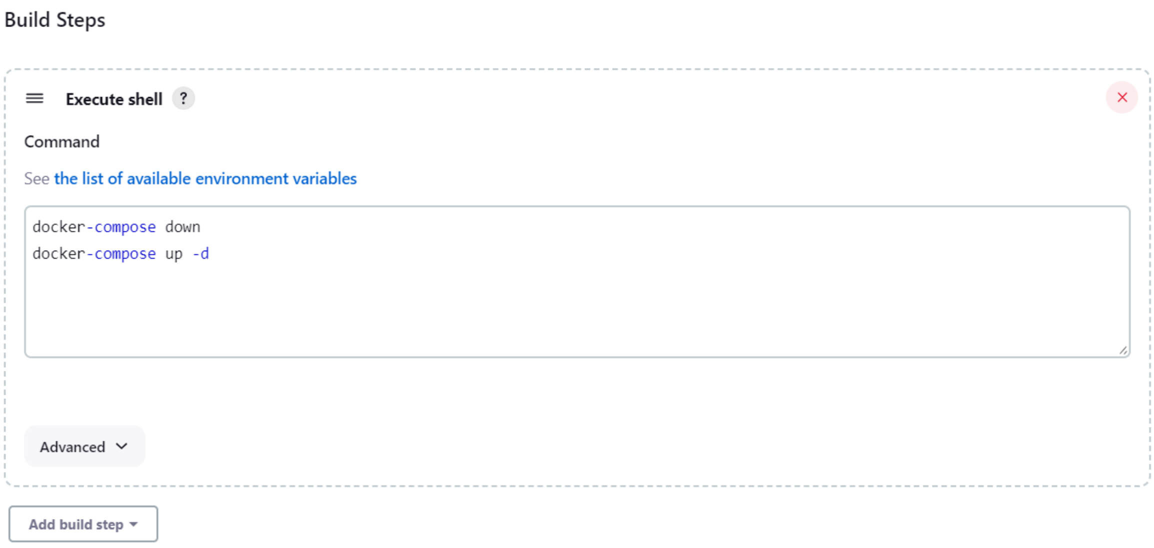
Webhooks

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.

✓ http://44.200.35.72:8080/github-we... (push)

Task-02

- In the Execute shell run the application using Docker compose



Build Steps

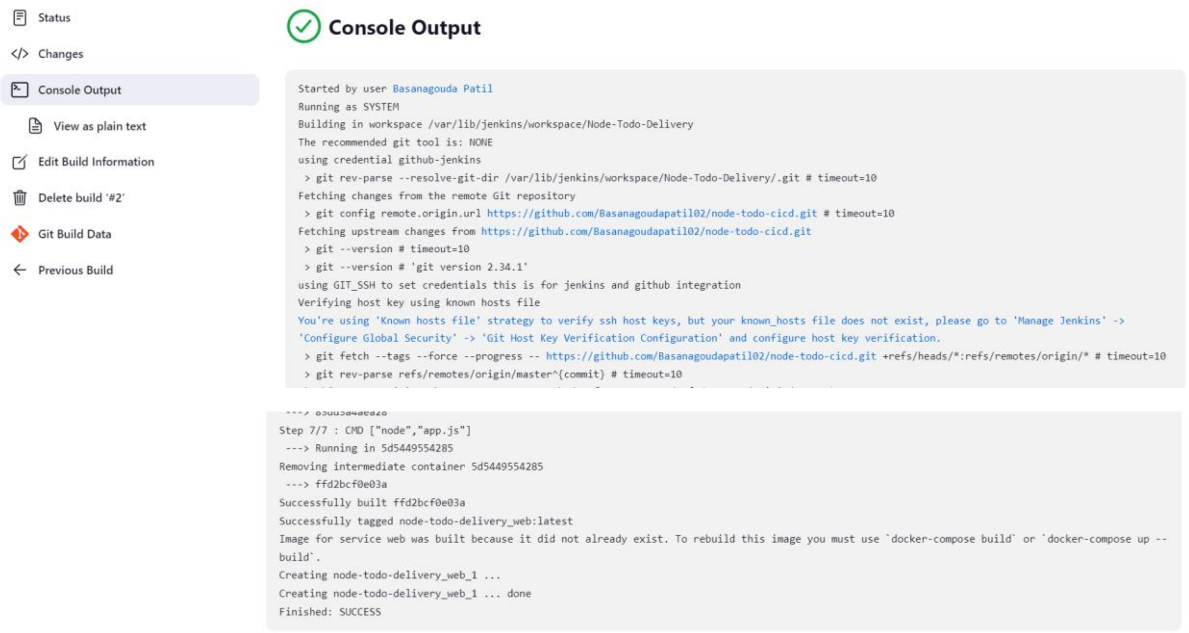
Execute shell

Command

See the list of available environment variables

```
docker-compose down
docker-compose up -d
```

- You will have to make a Docker Compose file for this Project and run the project and give yourself a treat:)

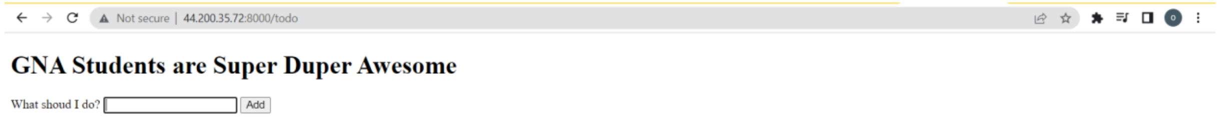


Console Output

```
Started by user Basanagouda Patil
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/Node-Todo-Delivery
The recommended git tool is: NONE
using credential github-jenkins
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Node-Todo-Delivery/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/Basanagoudapatil02/node-todo-cicd.git # timeout=10
Fetching upstream changes from https://github.com/Basanagoudapatil02/node-todo-cicd.git
 > git --version # timeout=10
 > git --version # 'git version 2.34.1'
using GIT_SSH to set credentials this is for jenkins and github integration
Verifying host key using known hosts file
You're using 'Known hosts file' strategy to verify ssh host keys, but your known_hosts file does not exist, please go to 'Manage Jenkins' ->
'Configure Global Security' -> 'Git Host Key Verification Configuration' and configure host key verification.
 > git fetch --tags --force --progress -- https://github.com/Basanagoudapatil02/node-todo-cicd.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
```

```
Step 7/7 : CMD ["node","app.js"]
 ---> Running in 5d5449554285
Removing intermediate container 5d5449554285
 ---> ffd2bcf0e03a
Successfully built ffd2bcf0e03a
Successfully tagged node-todo-delivery_web:latest
Image for service web was built because it did not already exist. To rebuild this image you must use `docker-compose build` or `docker-compose up --
build`.
Creating node-todo-delivery_web_1 ...
Creating node-todo-delivery_web_1 ... done
Finished: SUCCESS
```

Now, copy public IP of your Ec2 instance and past it in web-browser with port number.



GNA Students are Super Duper Awesome

What shoud I do? [ ] Add

Happy Learning:)