

# DeVops Jenkins Assignment

Assignment-02: Jenkins Master Slave pipeline intellipaat.

Roll No and Name – Ankitha S Madanbhavi (18BCS008 )

Anuj C Shiragave ( 18BCS009 )

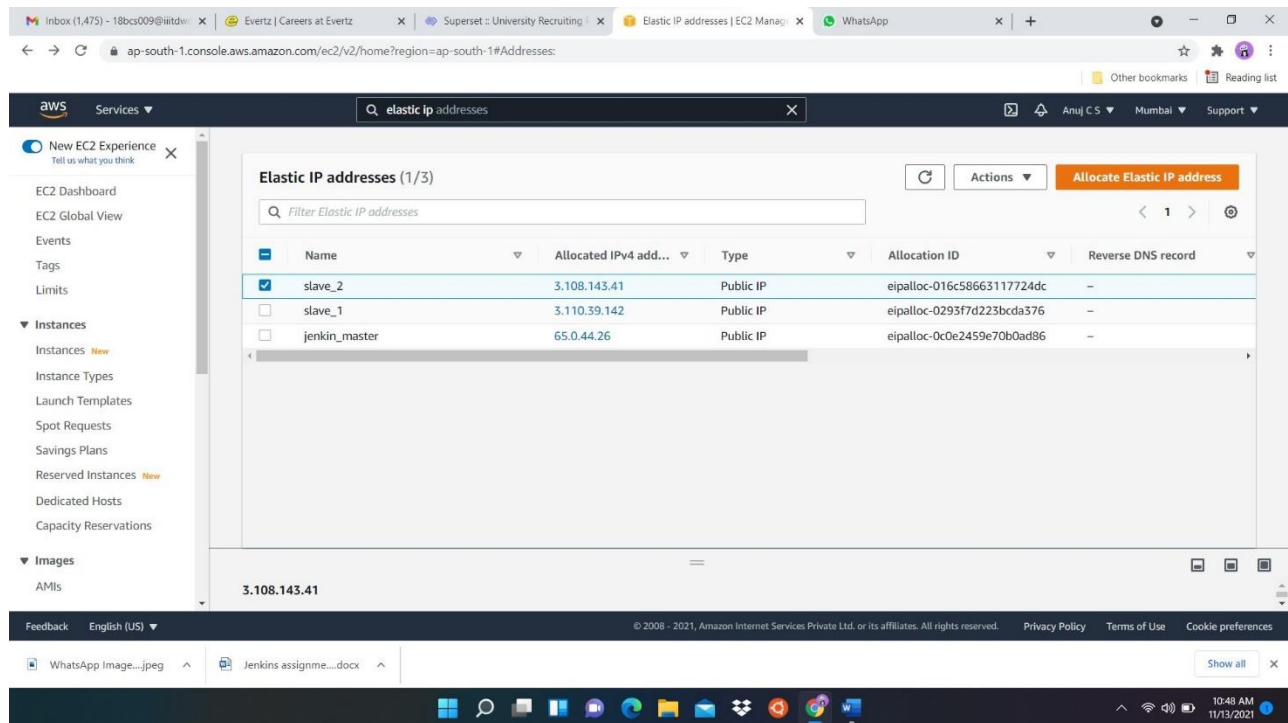
Step-1: First, we are creating three aws ec2 instances and naming them as jenkins\_master, slave-1, slave-2.

The screenshot displays the AWS Management Console interface. The top navigation bar shows the AWS logo, a search bar with 'elastic ip addresses', and user information 'Anuj C S' from 'Mumbai'. The left sidebar contains navigation links for 'New EC2 Experience', 'EC2 Dashboard', 'EC2 Global View', 'Events', 'Tags', 'Limits', 'Instances', 'Instance Types', 'Launch Templates', 'Spot Requests', 'Savings Plans', 'Reserved Instances', 'Dedicated Hosts', 'Capacity Reservations', 'Images', and 'AMIs'. The main content area is titled 'Instances (3)' and features a table with the following data:

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
<input type="checkbox"/>	jenkins_master	i-0878d993b8685eb5d	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1a	ec2-13-23...
<input type="checkbox"/>	slave_1	i-0eaca2988fa8222a9	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1a	ec2-15-20...
<input type="checkbox"/>	slave_2	i-043400efe2c96d2da	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1a	ec2-3-6-37...

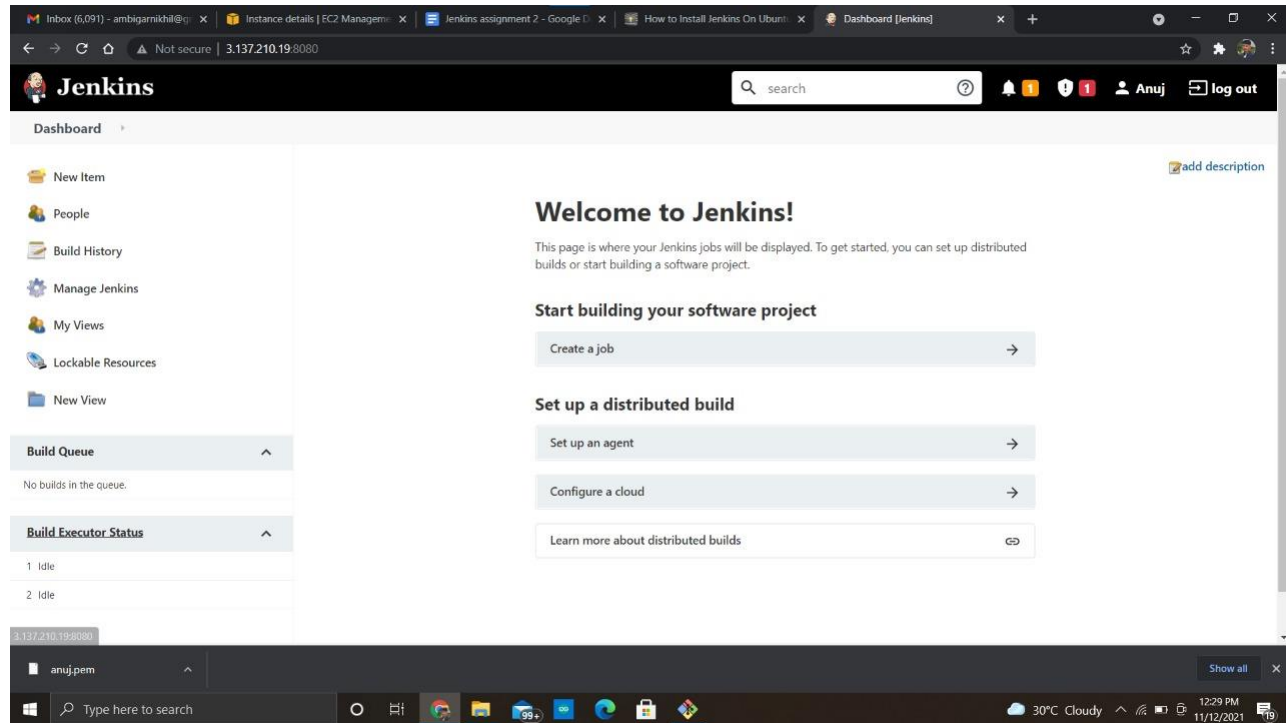
Below the table, there is a section titled 'Select an instance above' with a search bar and a list of instance types. The bottom of the screen shows a Windows taskbar with various application icons and a system clock indicating 10:49 AM on 11/13/2021.

## Step-2: Creating three elastic IP addresses and associating these elastic IP addresses to the ec2 instances.

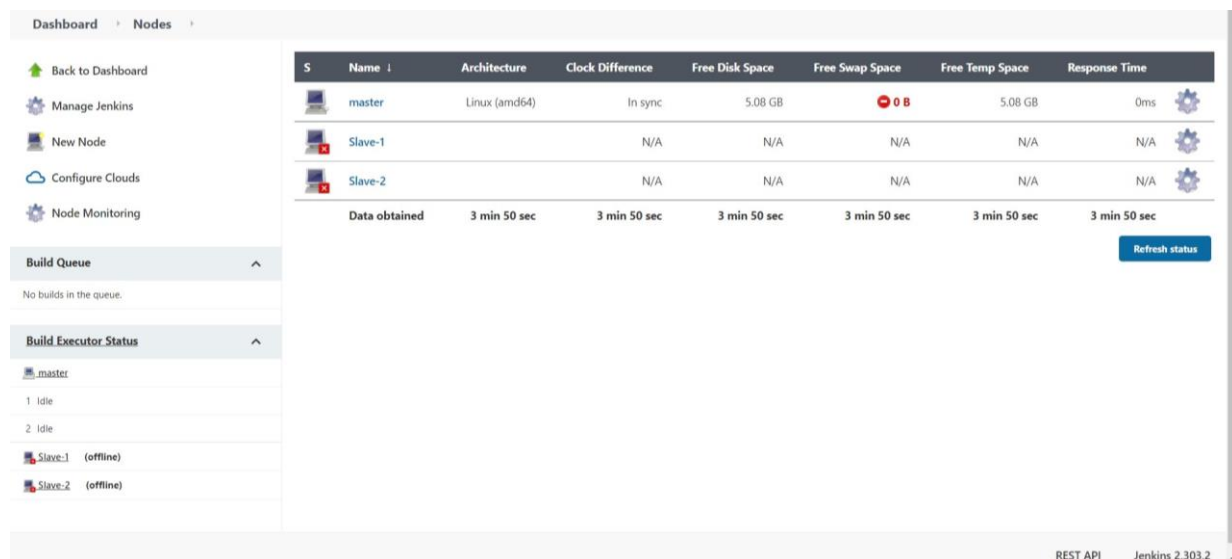


## Step-3: Install Jenkins on jenkins\_master ec2 instance. Commands to install Jenkins on ubuntu ec2 instance.

```
sudo apt-get update -y
sudo apt-get install openjdk-8-jdk
wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key
| sudo apt-key add -
sudo sh -c 'echo deb https://pkg.jenkins.io/debian-stable
binary/ > \
    /etc/apt/sources.list.d/jenkins.list'
sudo apt-get update
apt-get install jenkins
sudo service jenkins start
```

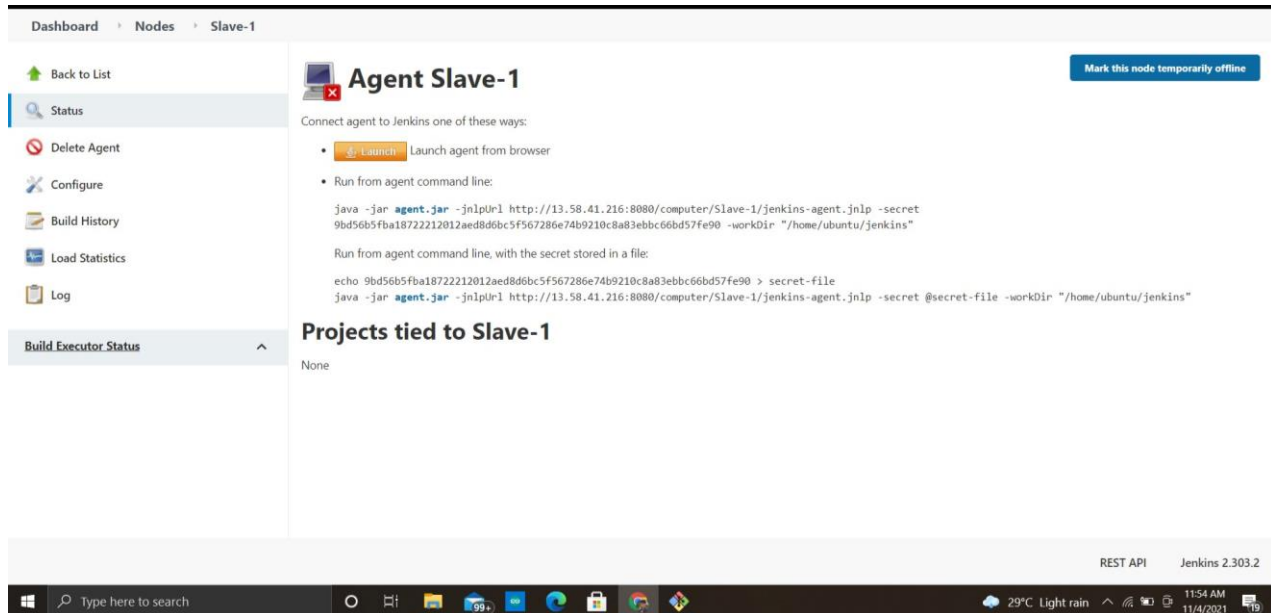


Step-4: Creating two nodes. (one for ec2 slave-1 instance and the other for ec2 slave-2 instance.)



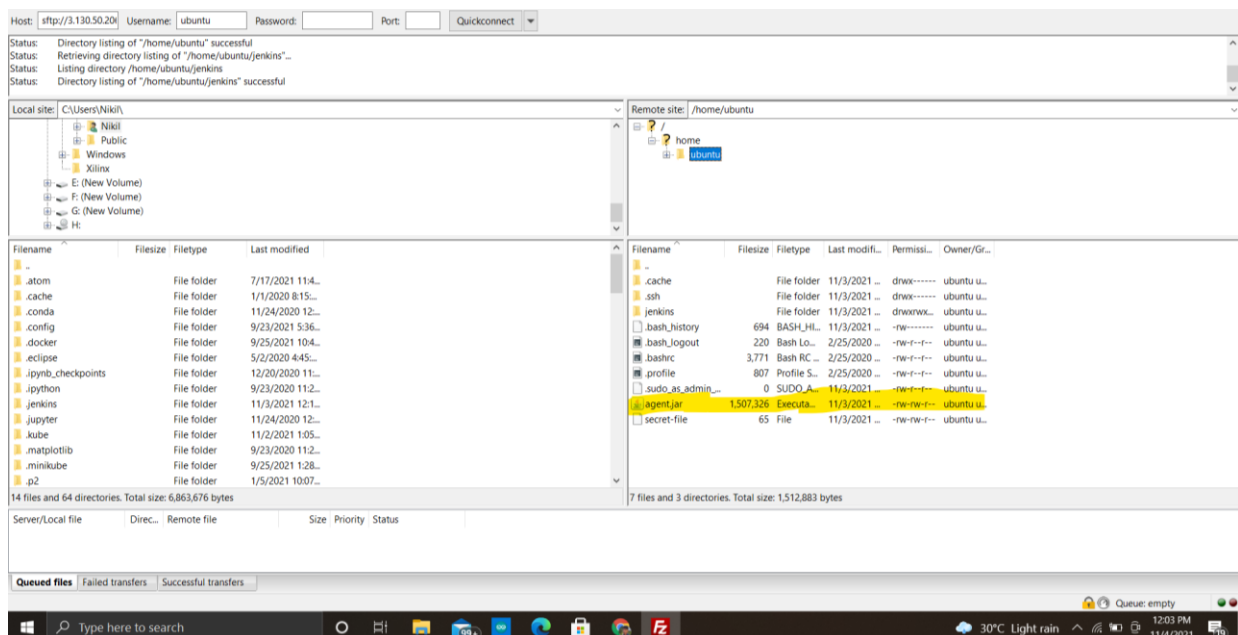
## Step-5:

Download agent.jar from Slave-1 node and using Filezilla, transfer that file to slave-1 ec2 instance.



The screenshot shows the Jenkins 'Agent Slave-1' configuration page. On the left is a sidebar with links: 'Back to List', 'Status', 'Delete Agent', 'Configure', 'Build History', 'Load Statistics', and 'Log'. The main area is titled 'Agent Slave-1' and includes a 'Mark this node temporarily offline' button. Below the title, it says 'Connect agent to Jenkins one of these ways:' and provides two options: 'Launch agent from browser' (with a 'Launch' button) and 'Run from agent command line'. The command line section shows two methods: one using a long secret string and another using a file named 'secret-file'. Below this, it says 'Projects tied to Slave-1' and lists 'None'. At the bottom of the page, it shows 'REST API' and 'Jenkins 2.303.2'.

## SLAVE-1



The screenshot shows the FileZilla interface. At the top, the 'Host' is 'sftp://3.130.50.20', 'Username' is 'ubuntu', and 'Port' is empty. The 'Local site' is 'C:\Users\Nikhil\...' and the 'Remote site' is '/home/ubuntu'. The interface is divided into two panes. The left pane shows the local file system with a tree view of directories like 'atom', '.cache', '.conda', '.config', '.docker', '.eclipse', '.ipynb\_checkpoints', '.python', '.jenkins', '.jupyter', '.kubernetes', '.matplotlib', '.minikube', and '.p2'. The right pane shows the remote file system with a tree view of directories like '.cache', '.ssh', '.jenkins', '.bash\_history', '.bash\_logout', '.bashrc', '.profile', '.sudo\_as\_admin...', 'agent.jar', and 'secret-file'. The 'agent.jar' file is highlighted in yellow. Below the panes, there is a table with columns 'Filename', 'Filesize', 'Filetype', 'Last modified', 'Permissions', and 'Owner/Gr...'. The table shows the details of the files in the remote site. At the bottom, there is a 'Queued files' section and a status bar showing 'Queue: empty'.

## Step-5:

Download agent.jar from Slave-2 node and using Filezilla, transfer that file to slave-2 ec2 instance.

The screenshot shows the Jenkins Dashboard with the 'Nodes' tab selected, displaying 'Slave-2'. The left sidebar contains links for 'Back to List', 'Status', 'Delete Agent', 'Configure', 'Build History', 'Load Statistics', and 'Log'. The main content area is titled 'Agent Slave-2' and includes a 'Mark this node temporarily offline' button. Below the title, it provides instructions on how to connect the agent to Jenkins, including a 'Launch agent from browser' button and a 'Run from agent command line' section. The command line section shows the following commands:

```
java -jar agent.jar -jnlpUrl http://13.58.41.216:8080/computer/Slave-2/jenkins-agent.jnlp -secret d6047dd30438303e888efaa8420025b16f3c16ed710a3d2d8ebefeb3b4e8f72 -workDir "/home/ubuntu/jenkins"
```

Below the commands, it states 'Run from agent command line, with the secret stored in a file:' and shows the following commands:

```
echo d6047dd30438303e888efaa8420025b16f3c16ed710a3d2d8ebefeb3b4e8f72 > secret-file
java -jar agent.jar -jnlpUrl http://13.58.41.216:8080/computer/Slave-2/jenkins-agent.jnlp -secret @secret-file -workDir "/home/ubuntu/jenkins"
```

The 'Projects tied to Slave-2' section shows 'None'. The bottom status bar indicates 'REST API' and 'Jenkins 2.303.2'.

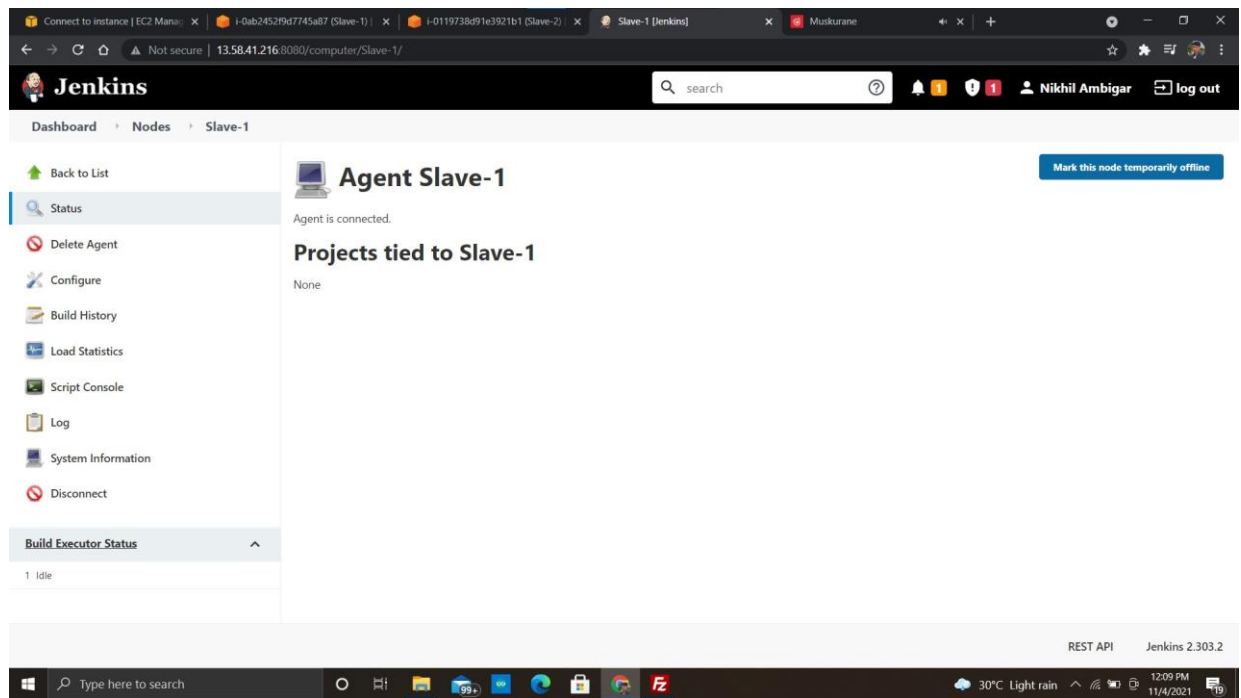
## SLAVE-2

The screenshot shows the FileZilla interface with the local site set to 'C:\Users\Nikhil\...' and the remote site set to '/home/ubuntu'. The local site pane shows a directory tree with folders like 'atom', '.cache', '.conda', '.config', '.docker', '.eclipse', '.ipynb\_checkpoints', '.python', '.jenkins', '.jupyter', '.kubernetes', '.matplotlib', '.minikube', and '.p2'. The remote site pane shows a directory tree with folders like '.cache', '.ssh', '.jenkins', '.bash\_history', '.bash\_logout', '.bashrc', '.profile', '.sudo\_as\_admin\_', and files like 'agent.jar' and 'secret-file'. The 'agent.jar' file is highlighted in the remote site pane. The bottom status bar shows 'Queue: empty'.

Step-6:

Run the command on the ec2 instances so that we can connect the nodes to the Jenkins.

## SLAVE-1 CONNECTED



# SLAVE-2 CONNECTED.

i-0119738d91e3921b1 (Slave-2)

Public IPs: 3.131.206.11 Private IPs: 172.31.35.96

Type here to search

30°C Light rain 12:12 PM 11/4/2021

Connect to instance | EC2 Manu... x | i-0ab2452f9d7745a87 (Slave-1) x | i-0119738d91e3921b1 (Slave-2) x | Slave-2 [Jenkins] x | Pal

Not secure | 13.58.41.216:8080/computer/Slave-2/

Jenkins search Nikhil Ambigar log out

Dashboard Nodes Slave-2

Back to List

Status

Delete Agent

Script Console

Disconnect

Build Executor Status

## Agent Slave-2

Agent is connected.

Projects tied to Slave-2

Mark this node temporarily offline

REST API Jenkins 2.303.2





## Step 8:

Create two jobs (test for Slave-1 and prod for Slave-2)

In configure, we are setting source code management as git and passing our Github repo link, in build we are selecting execute shell and writing some commands to run.

The screenshot shows the Jenkins Dashboard with a sidebar on the left containing links like 'New Item', 'People', 'Build History', 'Manage Jenkins', 'My Views', 'Lockable Resources', and 'New View'. The main area displays a table of build history for the 'CICD' view. The table has columns for 'S' (Status), 'W' (Workspace), 'Name', 'Last Success', 'Last Failure', and 'Last Duration'. Two jobs are listed: 'Prod' and 'Test'. Both jobs show a successful status (green checkmark) and a duration of 1.8 sec and 3.5 sec respectively. Below the table, there is a legend for 'Atom feed' links.

S	W	Name	Last Success	Last Failure	Last Duration
✓	Cloud	Prod	1 min 55 sec - #4	15 min - #2	1.8 sec
✓	Slave	Test	2 min 6 sec - #4	29 min - #2	3.5 sec

Icon: S M L

Legend: Atom feed for all, Atom feed for failures, Atom feed for just latest builds

## Step 9: Build Test job.

The screenshot shows the 'Project Test' dashboard. On the left is a sidebar with navigation links: 'Back to Dashboard', 'Status' (selected), 'Changes', 'Workspace', 'Build Now', 'Configure', 'Delete Project', 'GitHub', and 'Rename'. Below these is a 'Build History' section with a search bar containing 'find' and a table of build records. The main content area is titled 'Project Test' and includes links for 'Workspace' and 'Recent Changes'. It also features a 'Downstream Projects' section with a 'Prod' link and a 'Permalinks' section with a list of build status and timing information. A 'Disable Project' button is located in the top right corner.

Dashboard > Test >

Back to Dashboard

Status

Changes

Workspace

Build Now

Configure

Delete Project

GitHub

Rename

Build History trend ^

find

#4	Nov 4, 2021 7:42 AM
#3	Nov 4, 2021 7:17 AM
#2	Nov 4, 2021 7:15 AM
#1	Nov 4, 2021 7:01 AM

### Project Test

add description

Disable Project

Workspace

Recent Changes

### Downstream Projects

Prod

### Permalinks

- Last build (#4), 8 min 53 sec ago
- Last stable build (#4), 8 min 53 sec ago
- Last successful build (#4), 8 min 53 sec ago
- Last failed build (#2), 36 min ago
- Last unsuccessful build (#2), 36 min ago
- Last completed build (#4), 8 min 53 sec ago

31°C Partly sunny 1:22 PM 11/4/2021

Dashboard > Test > #1

Back to Project

Status

Changes

Console Output

View as plain text

Edit Build Information

Delete build '#1'

Git Build Data

Console Output

Started by user **Nikhil Ambigar**  
Running as SYSTEM  
Building remotely on **Slave-1** in workspace /home/ubuntu/workspace/Test  
The recommended git tool is: NONE  
No credentials specified  
Cloning the remote Git repository  
Cloning repository **https://github.com/hshar/devopsIQ.git**  
> git init /home/ubuntu/workspace/Test # timeout=10  
Fetching upstream changes from **https://github.com/hshar/devopsIQ.git**  
> git --version # timeout=10  
> git --version # 'git version 2.25.1'  
> git fetch --tags --force --progress -- **https://github.com/hshar/devopsIQ.git** +refs/heads/\*:refs/remotes/origin/\* # timeout=10  
> git config remote.origin.url **https://github.com/hshar/devopsIQ.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 **cc26380b7f45bfcf31831fc4f4f6e438fd6e2ab2** (refs/remotes/origin/master)  
> git config core.sparsecheckout # timeout=10  
> git checkout -f **cc26380b7f45bfcf31831fc4f4f6e438fd6e2ab2** # timeout=10  
Commit message: "Set up CI with Azure Pipelines"  
First time build. Skipping changelog.  
Finished: SUCCESS

REST API Jenkins 2.303.2

Type here to search 30°C Light rain 12:32 PM 11/4/2021

Step 10: Build prod job.

Dashboard > Prod

Back to Dashboard

Status

Changes

Workspace

Build Now

Configure

Delete Project

GitHub

Rename

Build History

trend

find

#1 Nov 4, 2021 7:32 AM

#2 Nov 4, 2021 7:29 AM

#1 Nov 4, 2021 7:28 AM

Atom feed for all Atom feed for failures

Project Prod

add description

Disable Project

Workspace

Recent Changes

Permalinks

- Last build (#2), 1 min 41 sec ago
- Last failed build (#2), 1 min 41 sec ago
- Last unsuccessful build (#2), 1 min 41 sec ago
- Last completed build (#2), 1 min 41 sec ago

Type here to search 30°C Partly sunny 1:03 PM 11/4/2021

Dashboard > Prod

### Project Prod

[add description](#)  
[Disable Project](#)

[Workspace](#)  
[Recent Changes](#)

### Upstream Projects

[Test](#)

### Permalinks

- Last build (#4), 9 min 17 sec ago
- Last stable build (#4), 9 min 17 sec ago
- Last successful build (#4), 9 min 17 sec ago
- Last failed build (#2), 22 min ago
- Last unsuccessful build (#2), 22 min ago
- Last completed build (#4), 9 min 17 sec ago

[Back to Dashboard](#)

[Status](#)  
[Changes](#)  
[Workspace](#)  
[Build Now](#)  
[Configure](#)  
[Delete Project](#)  
[GitHub](#)  
[Rename](#)

### Build History

trend ^

find

Build	Time
#4	Nov 4, 2021 7:42 AM
#3	Nov 4, 2021 7:32 AM
#2	Nov 4, 2021 7:29 AM
#1	Nov 4, 2021 7:28 AM

Windows taskbar: Type here to search, 31°C Partly sunny, 1:22 PM 11/4/2021

## Step-11: Creating the pipeline

Dashboard > CICD

### Build Pipeline: CICD

[Run](#) [History](#) [Configure](#) [Add Step](#) [Delete](#) [Manage](#)

**Pipeline #4**

**#4 Test**  
Nov 4, 2021 7:42:43 AM  
3.5 sec  
ambigamithi

**#4 Prod**  
Nov 4, 2021 7:42:54 AM  
1.5 sec

REST API Jenkins 2.303.2

Windows taskbar: Type here to search, 31°C Partly sunny, 1:14 PM 11/4/2021

Step-12: After successfully building our project we can see our website using slave-1 IP at port 82.

We can see our website using slave-2 IP at port 82.

THANK YOU

