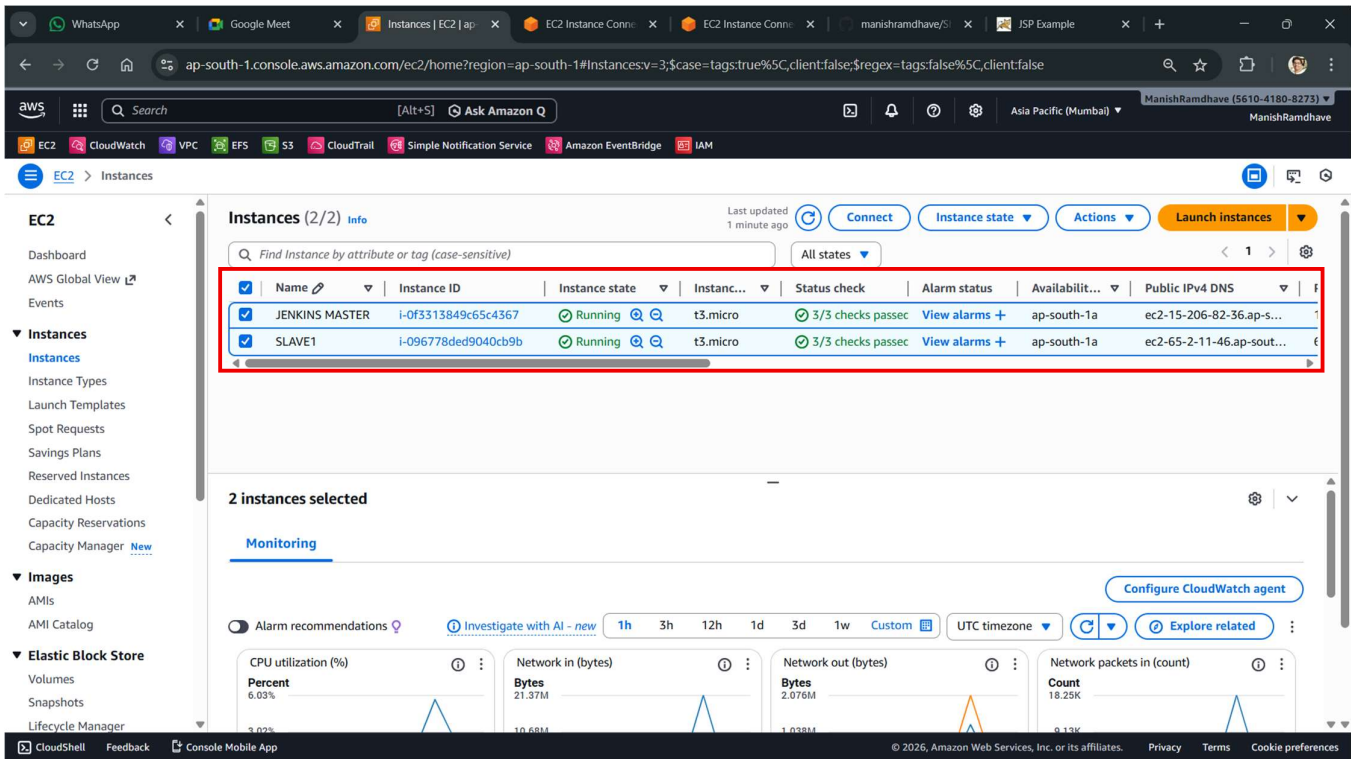
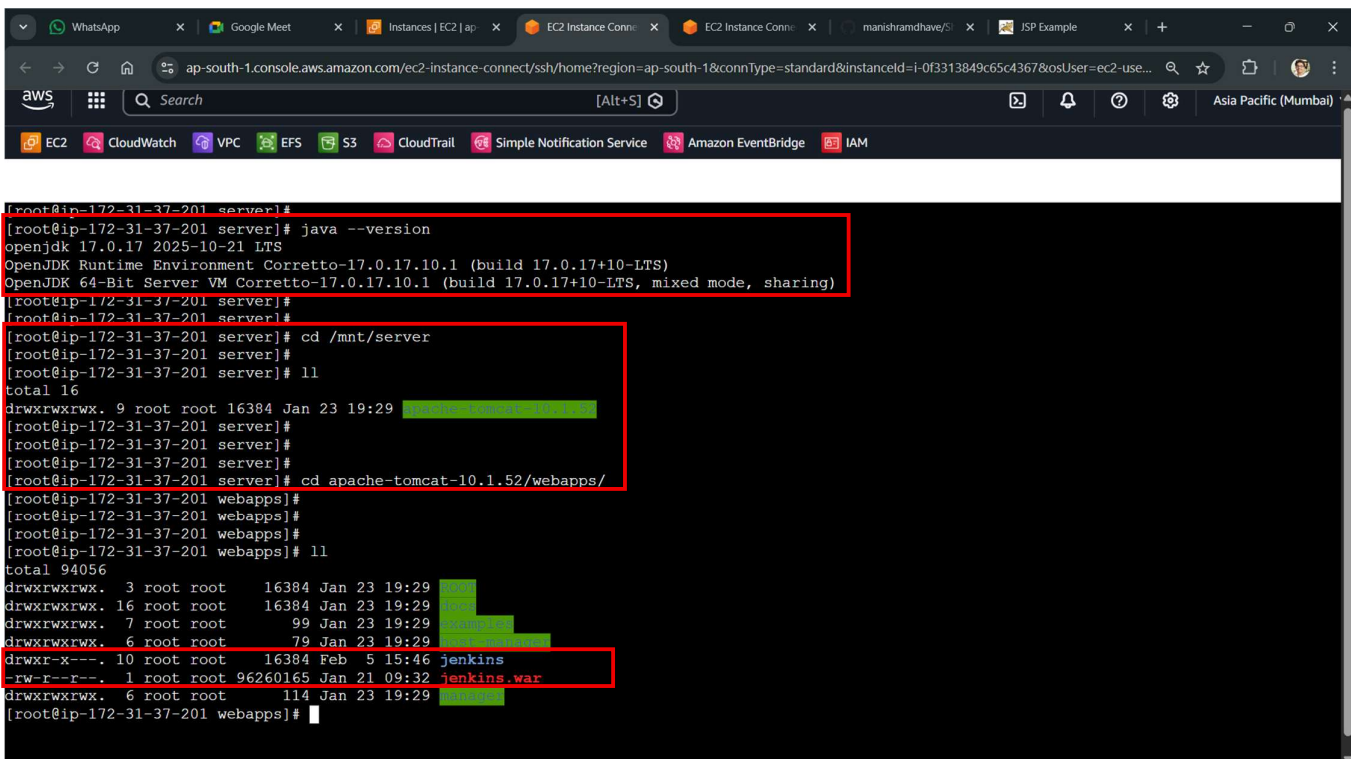


Jenkins Assignment 6

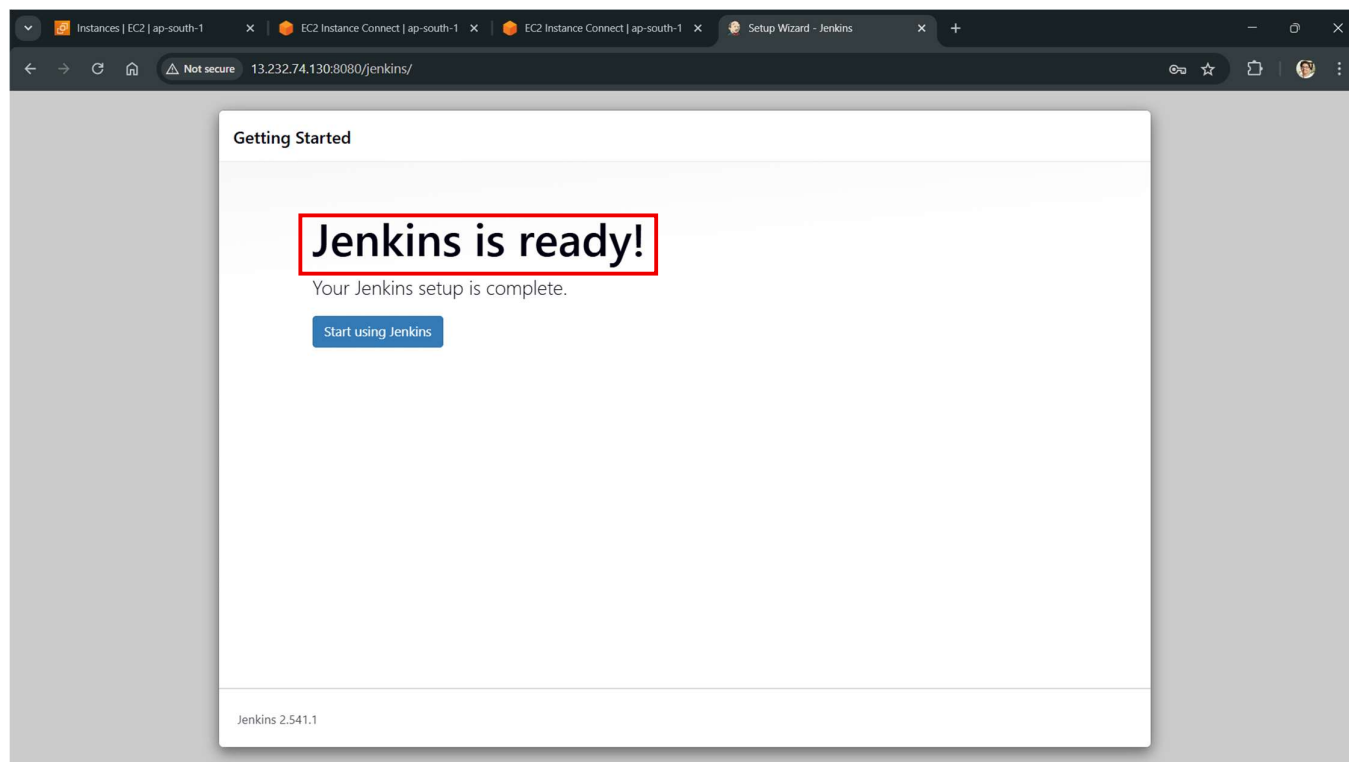
Step 1: Launched two instances, Jenkins Master and Slave1



Step 2: On Jenkins Master, we have installed Java-17, Apache-Tomcat-10.1.52 and deployed the Jenkins on it:



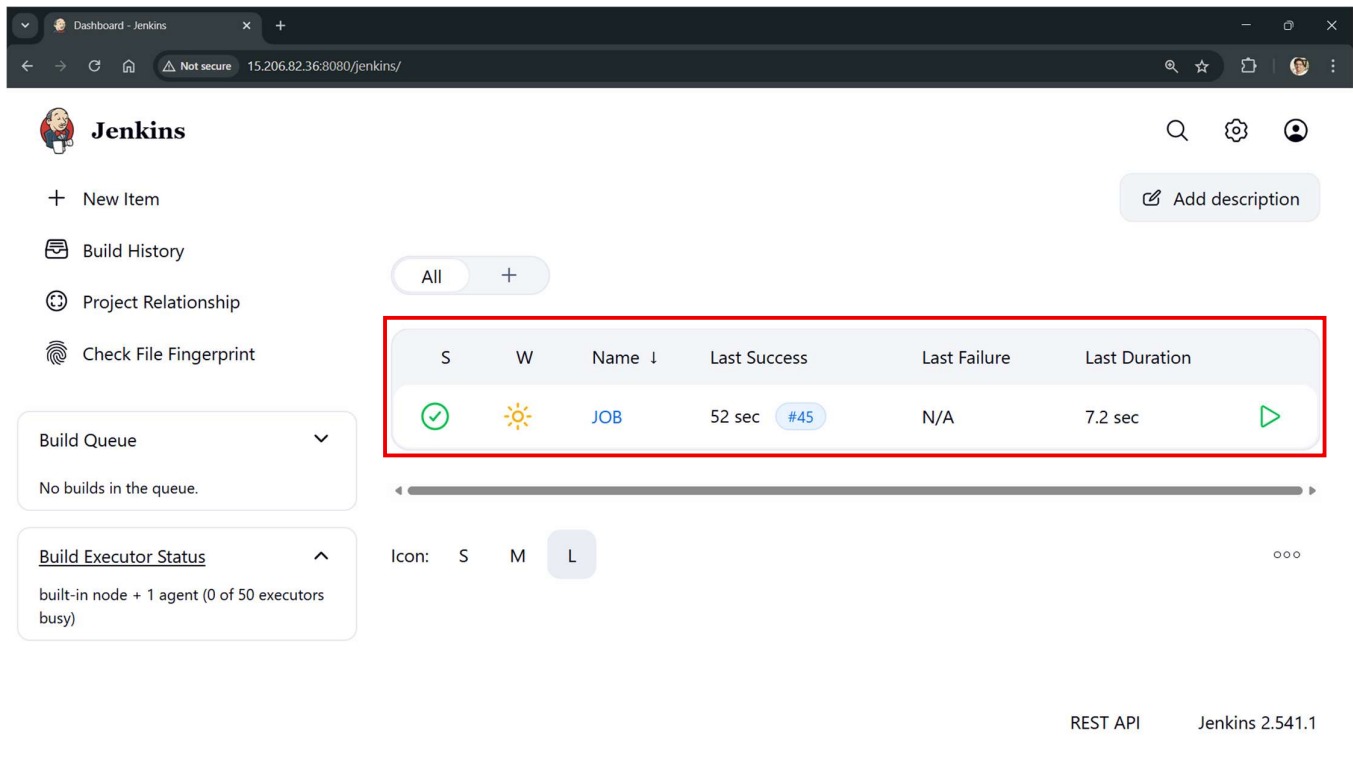
Step 3: Installed the **Jenkins** on **Jenkins Master Instance**:



Step 4: Created a **Node** for to establish the connection between Jenkins Master and Slave Instance using the ‘**Credential**’ and ‘**Manually trusted key Verification Strategy**’:

A screenshot of the Jenkins 'Nodes' page. The page has a white background with a blue header bar. The header bar contains the Jenkins logo, the text 'Jenkins / Nodes', and a search icon. Below the header bar, there is a blue button labeled '+ New Node' and a grey button labeled 'Configure Monitors'. The main content area has a title 'Nodes' and a subtitle 'Add, remove, control and monitor the various nodes that Jenkins runs jobs on.' Below this is a table with the following columns: S, Name, Architecture, Clock Difference, Free Disk Space, Free Swap Space, Free Temp Space, and Response Time. The table has two rows: 'Built-In Node' and 'Slave1'. The 'Slave1' row is highlighted with a red box. Below the table, there is a legend with the text 'Icon: S M L' and 'Legend'. At the bottom right, there is a footer with the text 'REST API' and 'Jenkins 2.541.1'.

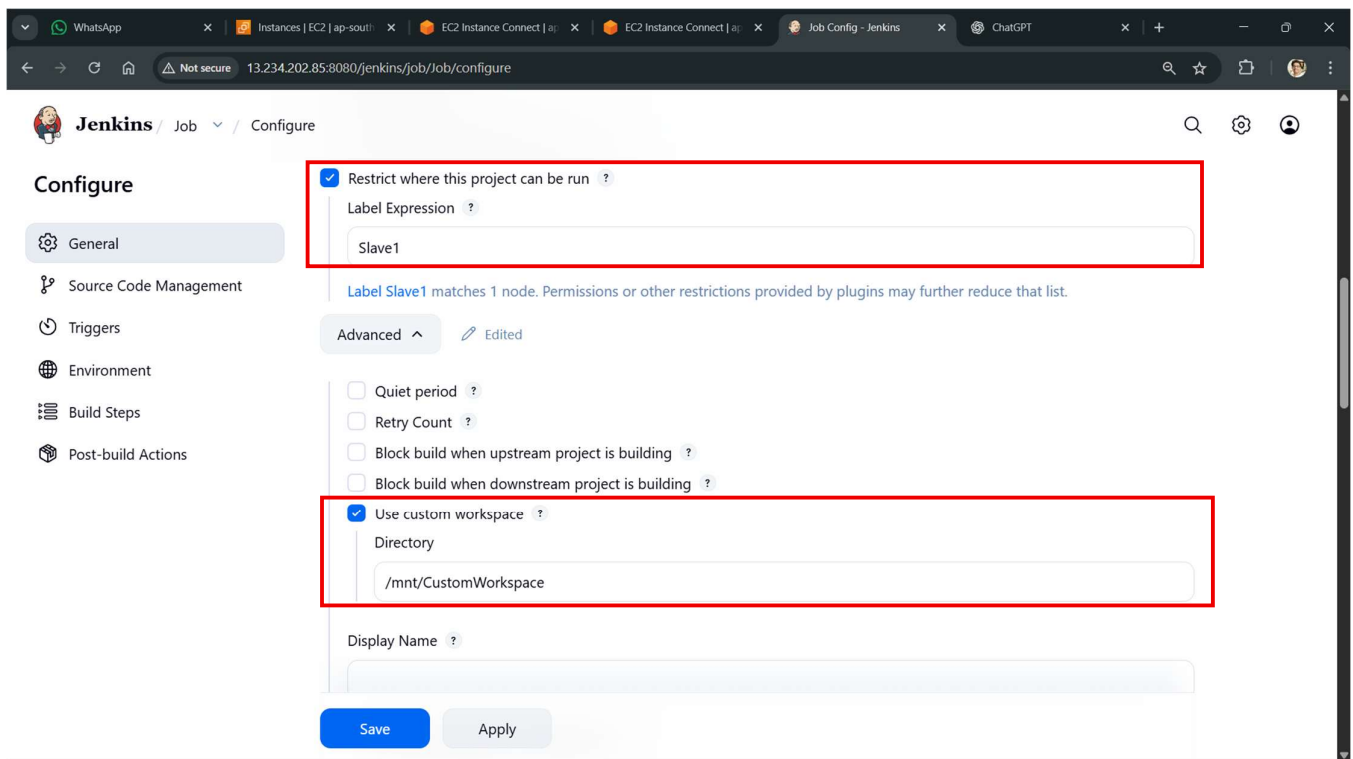
Step 5: Launched the Jenkins and created a **Freestyle Job**:



The screenshot shows the Jenkins Dashboard at the URL `15.206.82.36:8080/jenkins/`. The dashboard includes a sidebar with navigation links: **New Item**, **Build History**, **Project Relationship**, and **Check File Fingerprint**. Below these are two panels: **Build Queue** (showing 'No builds in the queue.') and **Build Executor Status** (showing 'built-in node + 1 agent (0 of 50 executors busy)'). The main area displays a table of jobs, with one job named **JOB** highlighted. The table has columns: **S** (Status), **W** (Icon), **Name**, **Last Success**, **Last Failure**, and **Last Duration**. The **JOB** entry shows a green checkmark, a sun icon, a duration of 52 sec, and a last success time of 7.2 sec. A red box highlights the job entry. At the bottom right, there are links for **REST API** and **Jenkins 2.541.1**.

S	W	Name	Last Success	Last Failure	Last Duration
✓	☀	JOB	52 sec #45	N/A	7.2 sec

Step 6: Enabled **‘Restrict where this project can be run’** on only Slave1 Instance and used a **custom workspace** to **‘/mnt/CustomWorkspace/’**:



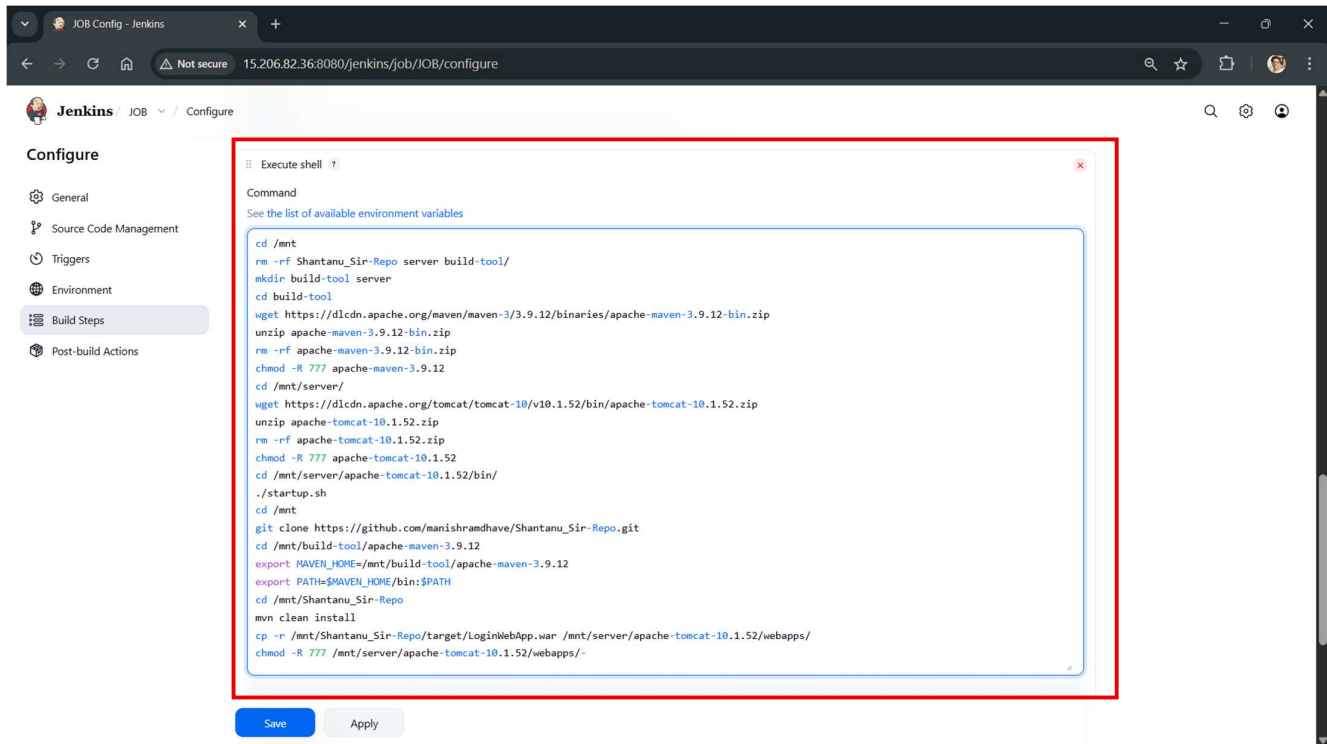
The screenshot shows the Jenkins Job Configuration page at the URL `13.234.202.85:8080/jenkins/job/Job/configure`. The page is titled **Configure** and has a sidebar with navigation links: **General**, **Source Code Management**, **Triggers**, **Environment**, **Build Steps**, and **Post-build Actions**. The **General** tab is selected. The configuration includes a section for **Restrict where this project can be run**, which is checked and has a label expression of **Slave1**. Below this is a section for **Use custom workspace**, which is also checked and has a directory of **/mnt/CustomWorkspace**. A red box highlights the **Restrict where this project can be run** section. At the bottom, there are buttons for **Save** and **Apply**.

☒ Restrict where this project can be run ?
Label Expression ?
Slave1
Label Slave1 matches 1 node. Permissions or other restrictions provided by plugins may further reduce that list.

☒ Use custom workspace ?
Directory
/mnt/CustomWorkspace

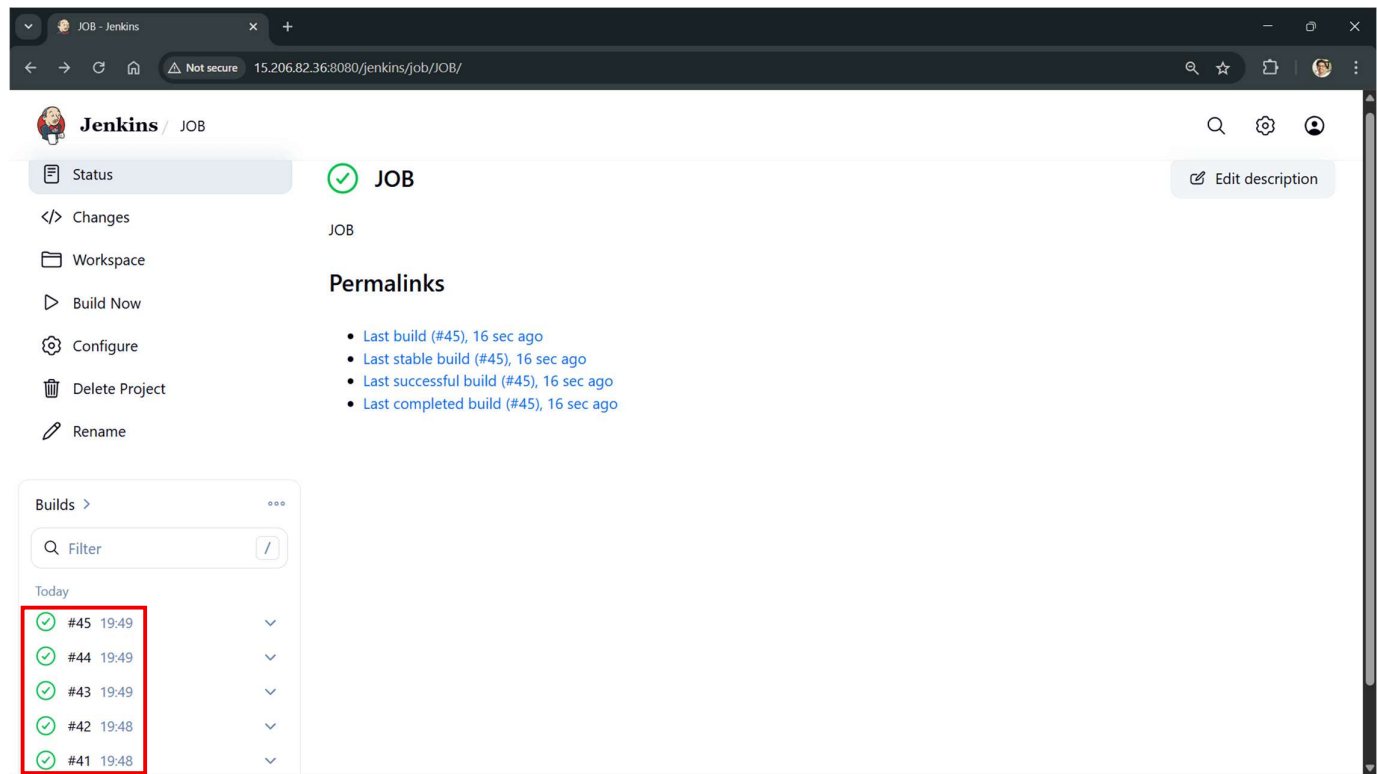
Step 7: Executed a shell script in the Build Steps on the Jenkins Job which will run on the Slave Instance. It will run all the commands below;

- A) Installation of Apache- Maven-3.9.12
- B) Installation of Apache-Tomcat-10.1.52 and running process
- C) Cloning the forked Repository to build LoginWebApp.war file
- D) Setting Environment Variable and its path
- E) Deployment of LoginWebApp Application on Tomcat Server



Result:

1. Our 5 consecutive Jobs' Builds are Successfully done:



2. Our LoginWebApp Application running successfully on Tomcat Server:

