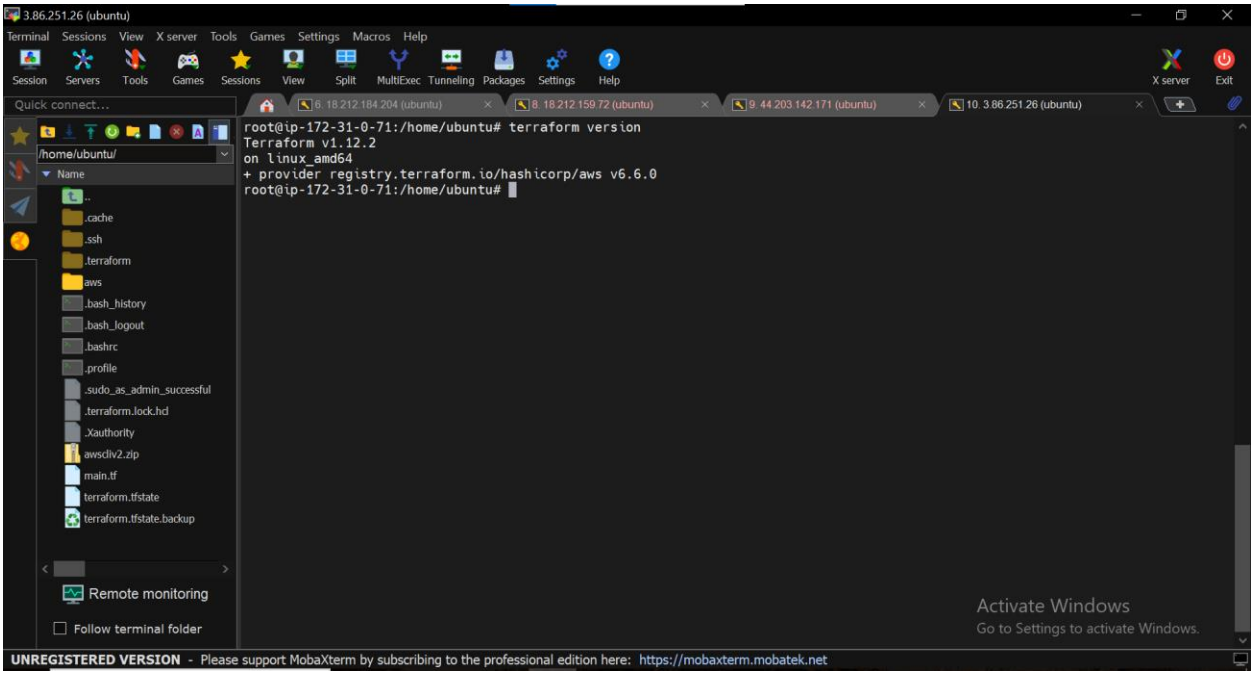
Banking & Finance Domain

Submitted by Pratheeban palani

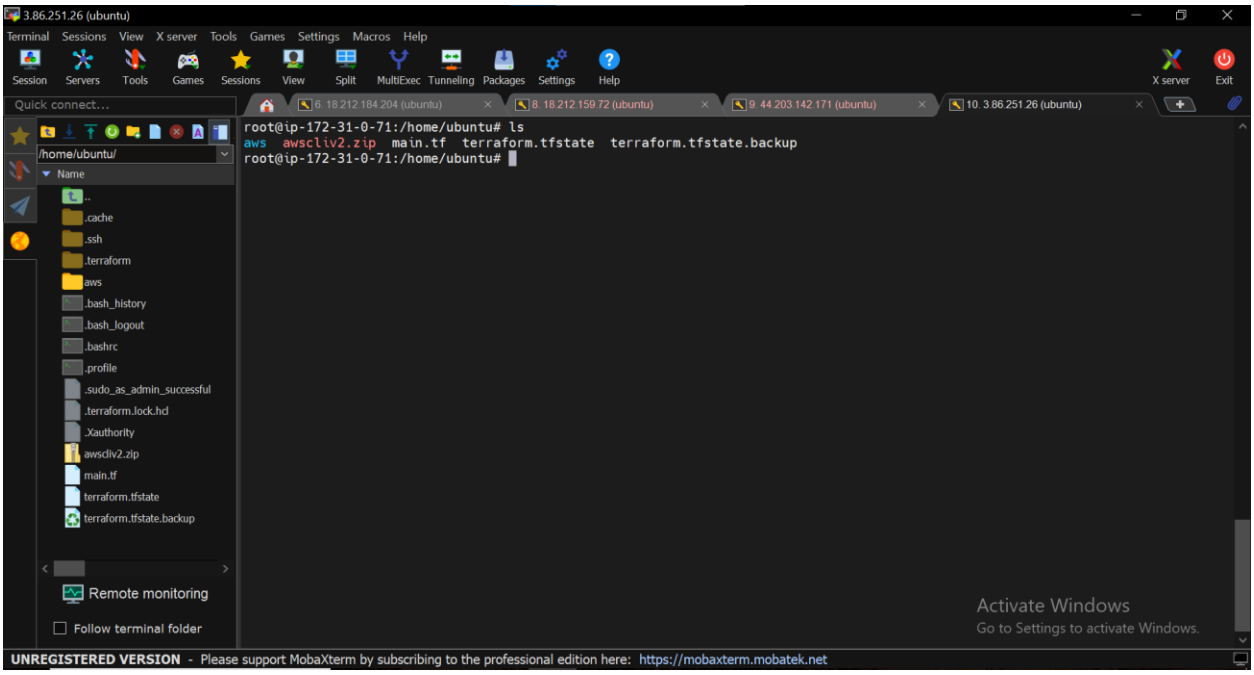
Submitted to vikul

Date : 08/03/2025

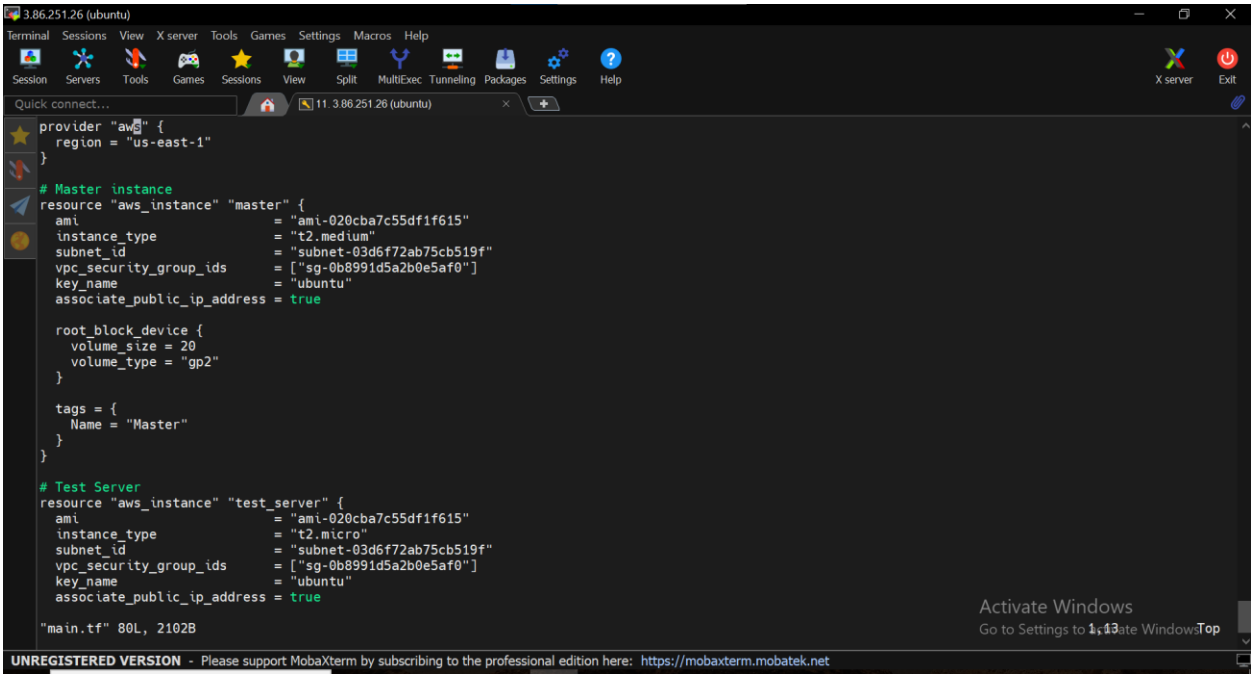
Step 1 : Create and launch an instance also install terraform in it.

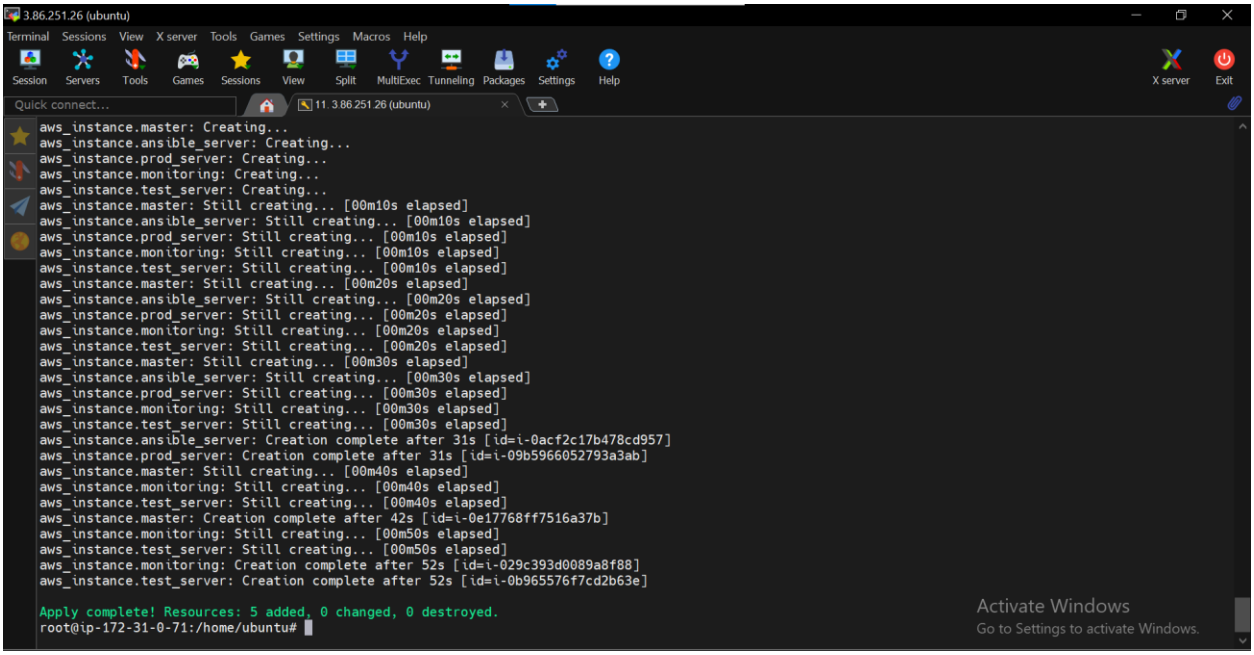


Step 2: Install aws cli and configure the instance with aws account.

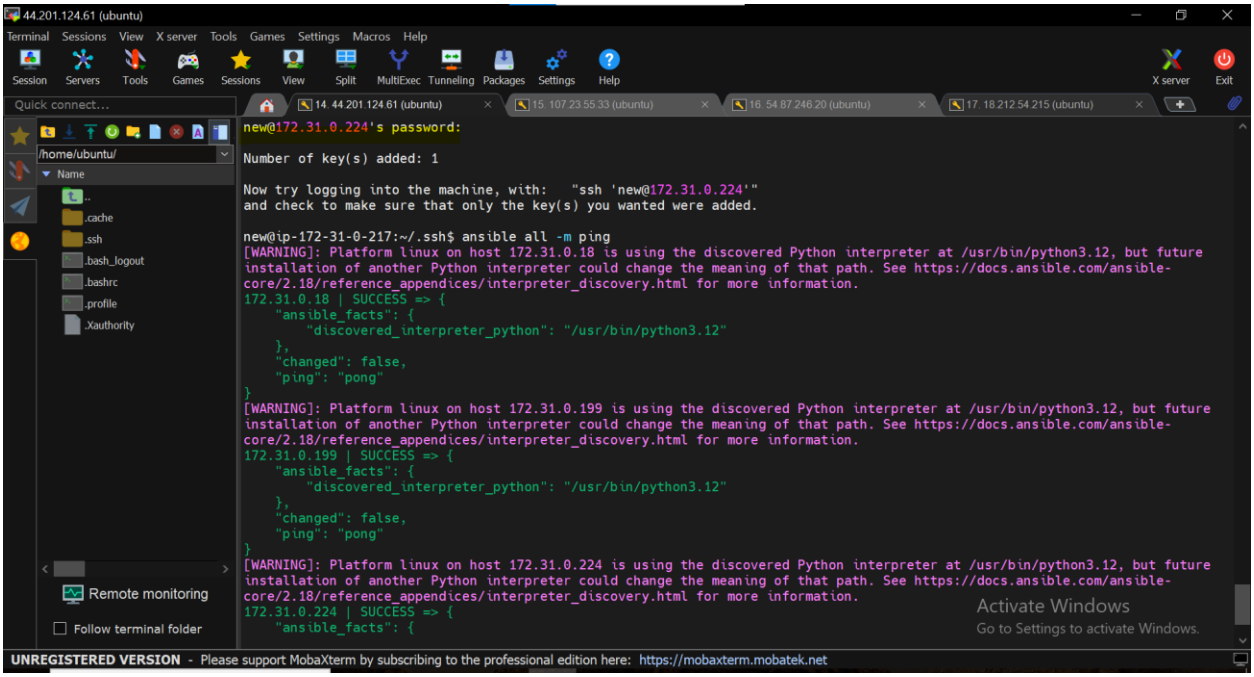


Step 3 : Write a script to create 5 EC2 instance.

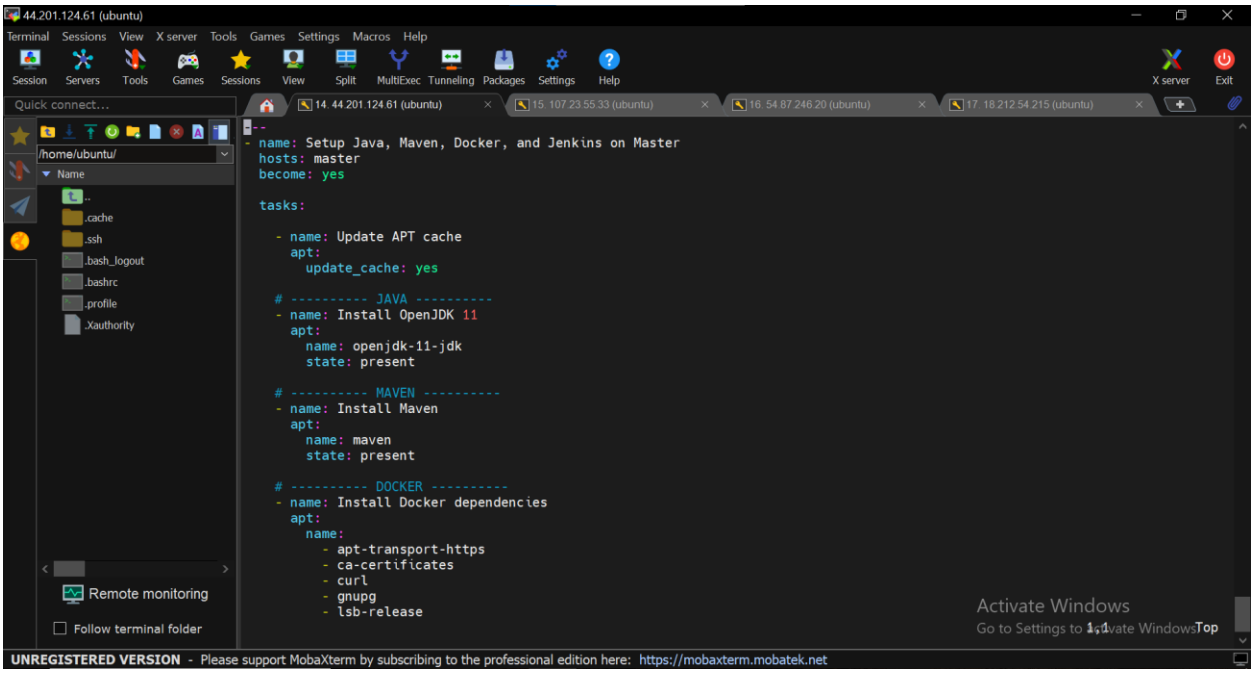




Step 4: Launch the ansible machine, install ansible in it and configure it to the node machines. I created a user name “new” and connected it with all the node machine with ssh connection.

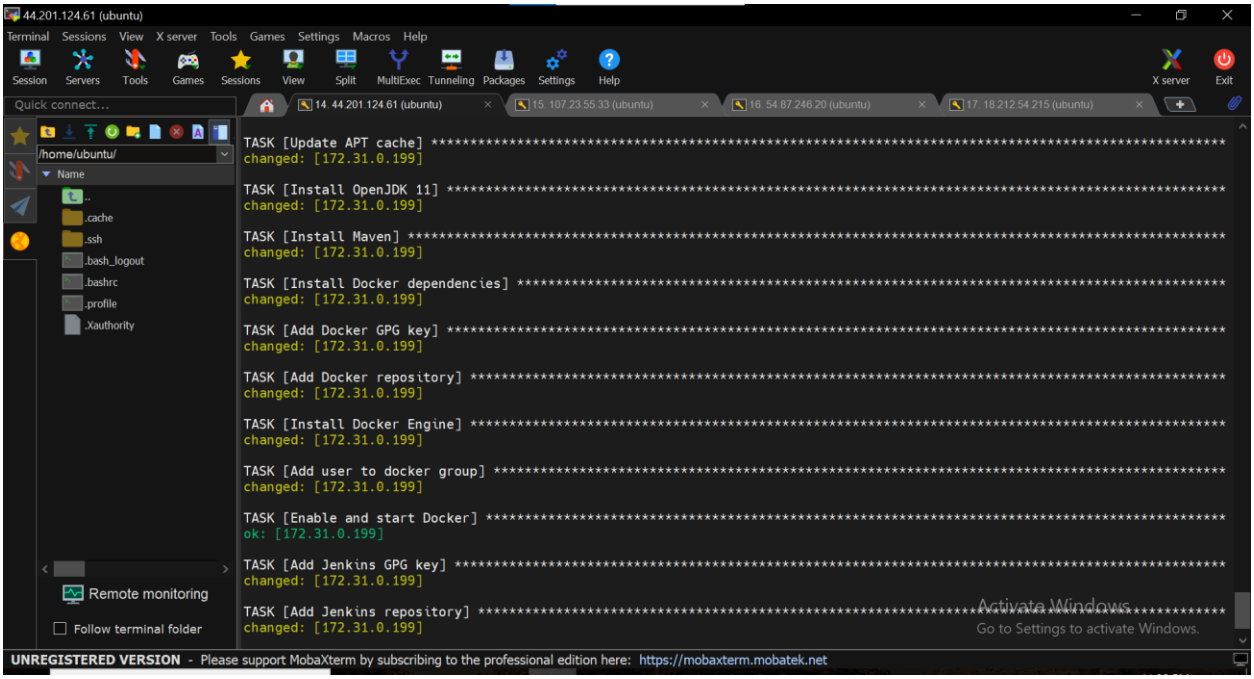


Step 7 :create an ansible playbook to install java, maven, Jenkins, docker in master machine.



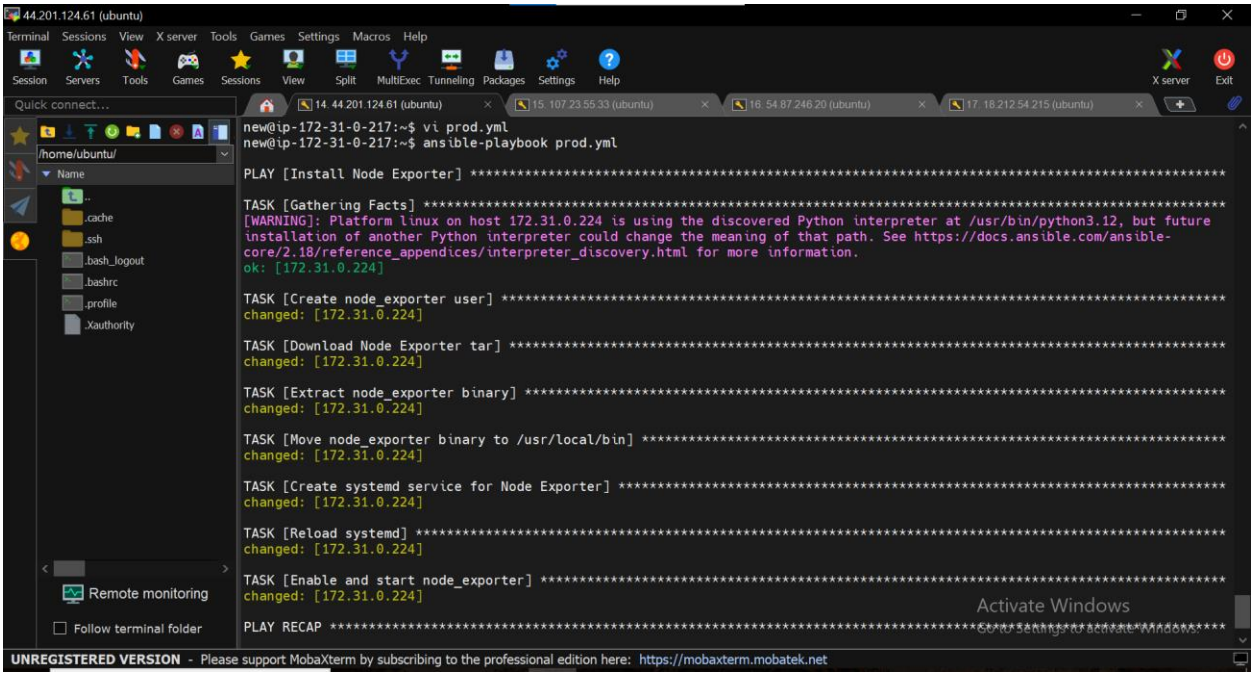

```
---
- name: Setup Java, Maven, Docker, and Jenkins on Master
  hosts: master
  become: yes

  tasks:

    - name: Update APT cache
      apt:
        update_cache: yes

    # ---------- JAVA ----------
    - name: Install OpenJDK 11
      apt:
        name: openjdk-11-jdk
        state: present

    # ---------- MAVEN ----------
    - name: Install Maven
      apt:
        name: maven
        state: present

    # ---------- DOCKER ----------
    - name: Install Docker dependencies
      apt:
        name:
          - apt-transport-https
          - ca-certificates
          - curl
          - gnupg
          - lsb-release
```

Step 8 : Run the playbook, all packages installed.

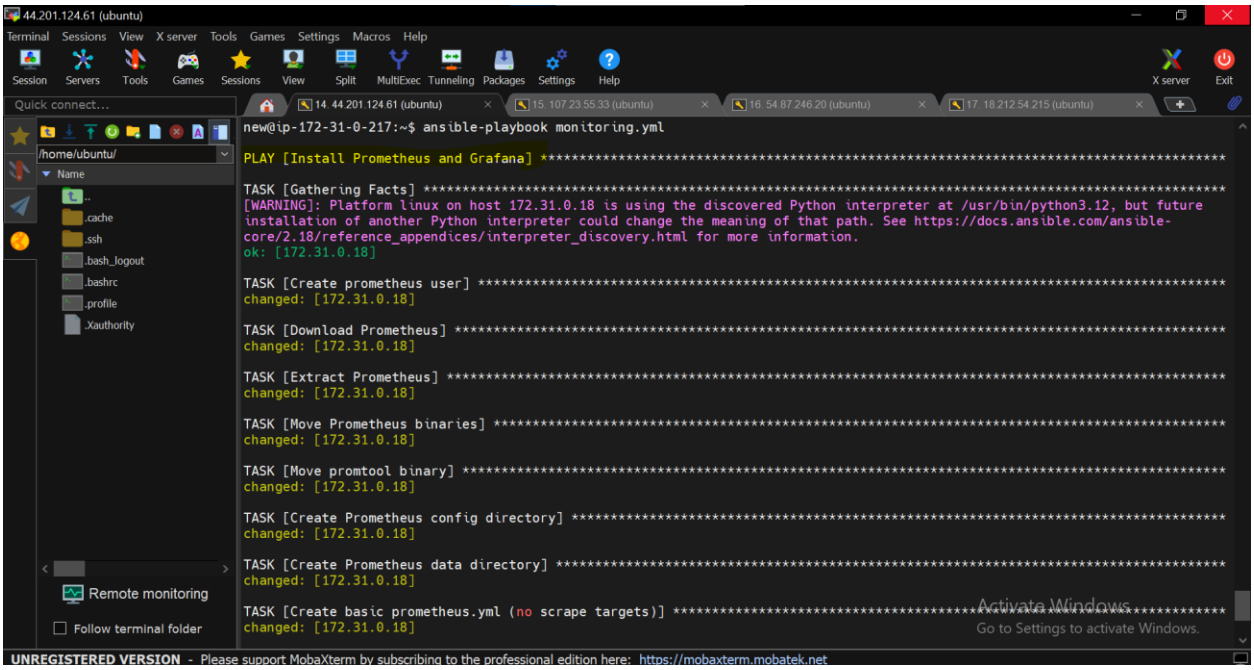


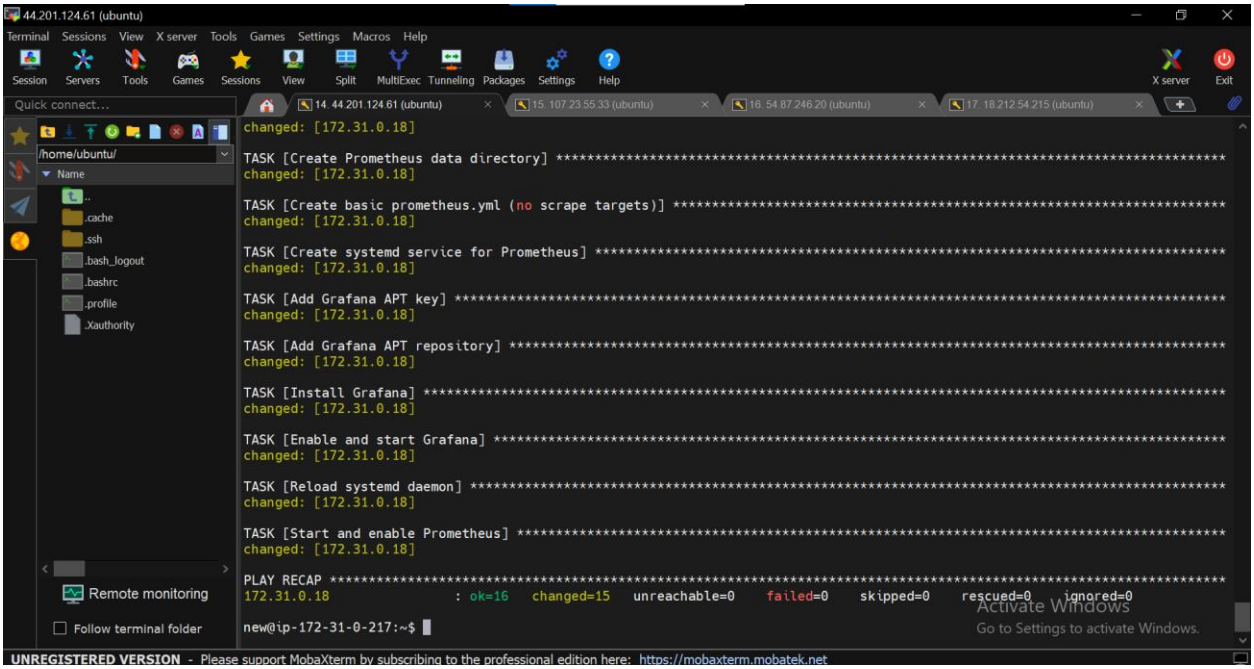

```
TASK [Update APT cache] ************************************************************************
changed: [172.31.0.199]

TASK [Install OpenJDK 11] ************************************************************************
changed: [172.31.0.199]

TASK [Install Maven] ************************************************************************
changed: [172.31.0.199]

TASK [Install Docker dependencies] ************************************************************************
changed: [172.31.0.199]

TASK [Add Docker GPG key] ************************************************************************
changed: [172.31.0.199]

TASK [Add Docker repository] ************************************************************************
changed: [172.31.0.199]

TASK [Install Docker Engine] ************************************************************************
changed: [172.31.0.199]

TASK [Add user to docker group] ************************************************************************
changed: [172.31.0.199]

TASK [Enable and start Docker] ************************************************************************
ok: [172.31.0.199]

TASK [Add Jenkins GPG key] ************************************************************************
changed: [172.31.0.199]

TASK [Add Jenkins repository] ************************************************************************
changed: [172.31.0.199]
```

Step 9 : Node expoter installed in prod server.

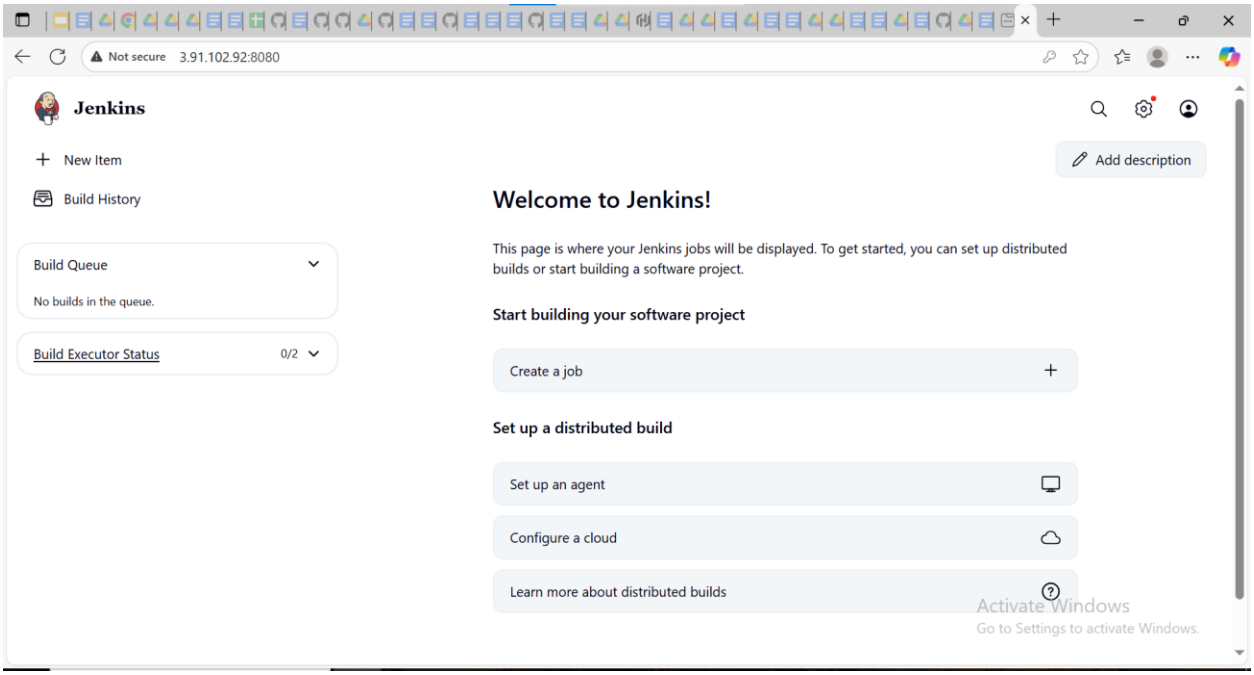


Step 10 : Installing Prometheus and grafana

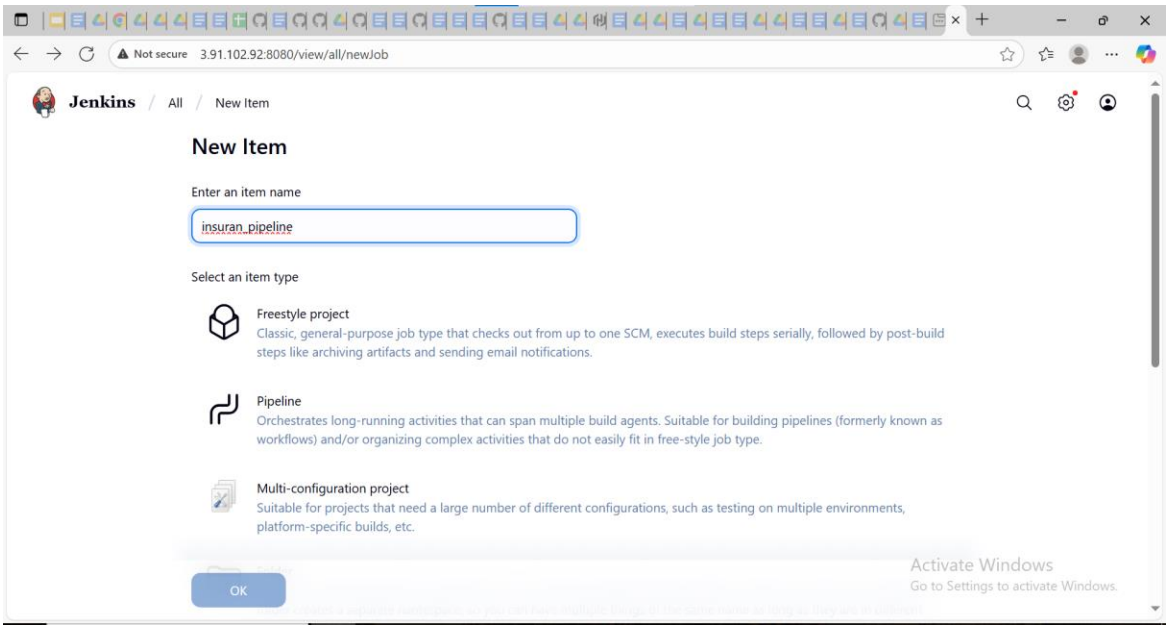




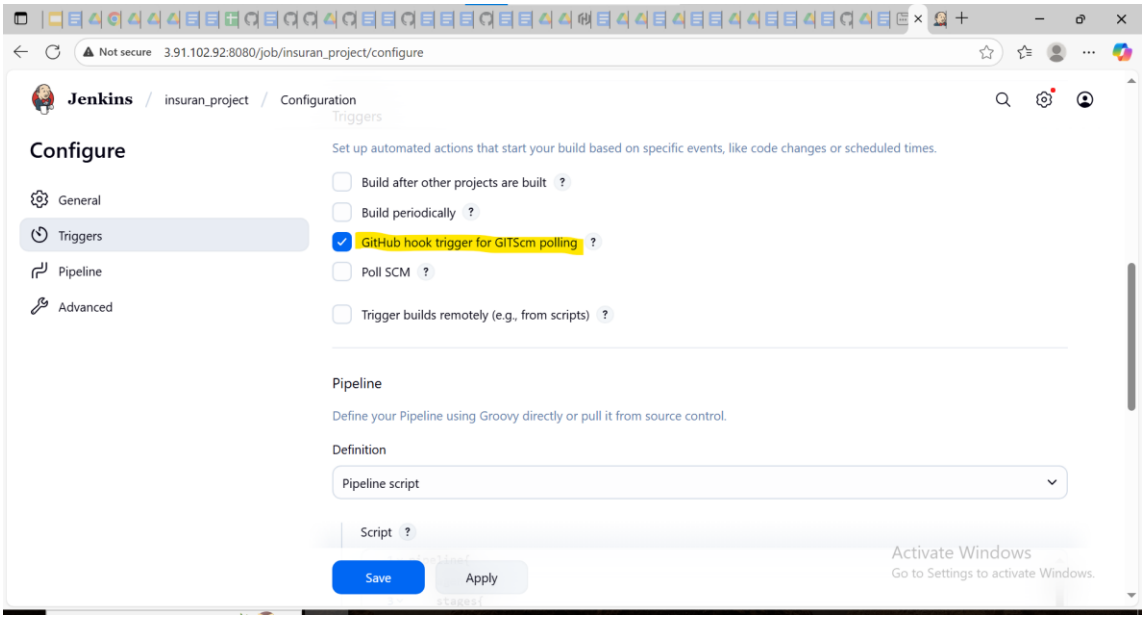
Step 11 : Launch Jenkins in the browser



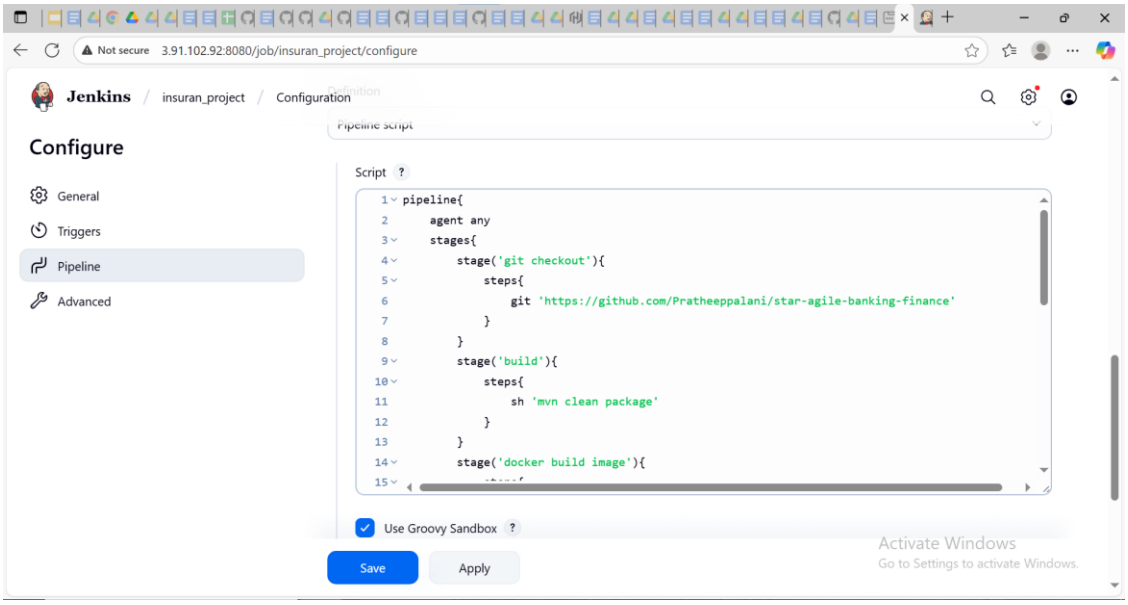
Step 12 : Now create a pipeline, click on new item And enter name, choose pipeline and click ok



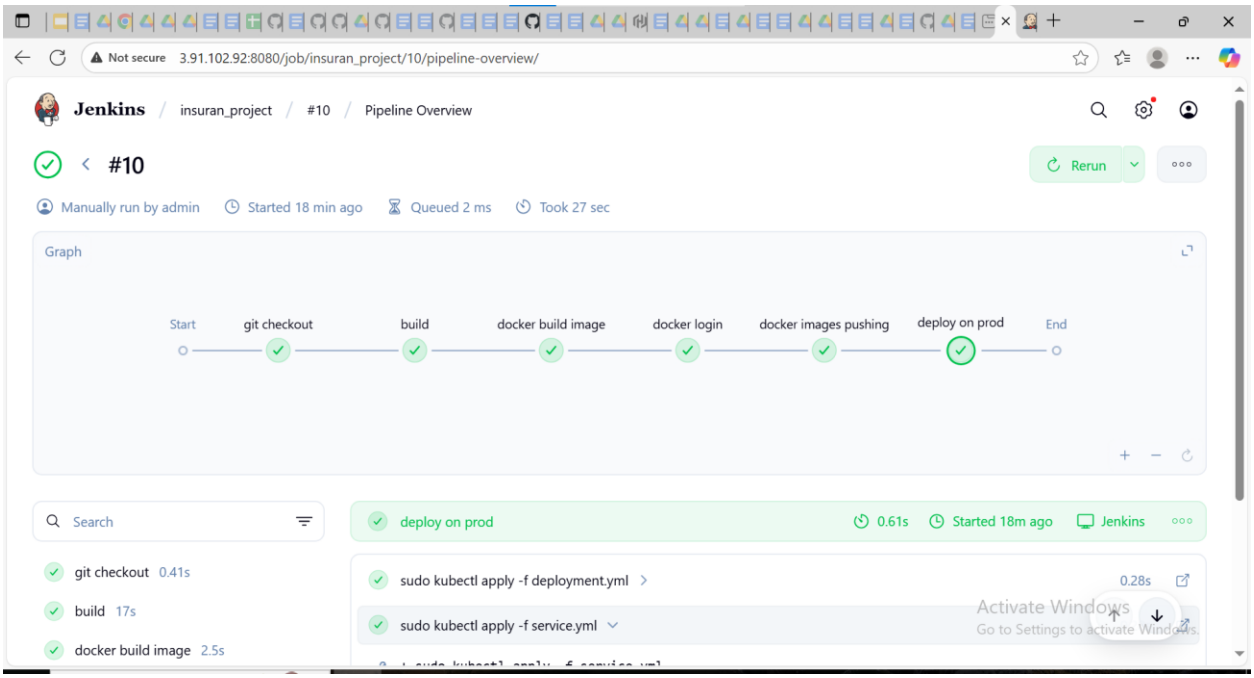
Step 13 : select github hook trigger to automate the build whenever there is change made in the repo.



Step 14 : Enter the script to checkout the repo, build it, create a docker image, login to dockerhub, push the image, deploy through kubernetes.



Step 15 : After that click on save and build the pipeline



Not secure 3.91.102.92:8080/job/insuran_project/10/pipeline-overview/?selected-node=6

Jenkins / insuran_project / #10 / Pipeline Overview

Search

- git checkout 0.41s
- build 17s
- docker build image 2.5s
- docker login 0.38s
- docker images pushing 5.7s
- deploy on prod 0.61s

Git https://github.com/Pratheeppalani/star-agile-banking-finance 0.37s

```
5  Fetching upstream changes from https://github.com/Pratheeppalani/star-agile-banking-finance
6   > git --version # timeout=10
7   > git --version # 'git version 2.43.0'
8   > git fetch --tags --force --progress -- https://github.com/Pratheeppalani/star-agile-banking-finance
    +refs/heads/*:refs/remotes/origin/* # timeout=10
9   > git rev-parse refs/remotes/origin/master^{commit} # timeout=10
10 Checking out Revision e0d0ad21b1853ec00594a3d551a6a095e5616b16 (refs/remotes/origin/master)
11  > git config core.sparsecheckout # timeout=10
12  > git checkout -f e0d0ad21b1853ec00594a3d551a6a095e5616b16 # timeout=10
13  > git branch -a -v --no-abbrev # timeout=10
14  > git branch -D master # timeout=10
15  > git checkout -b master e0d0ad21b1853ec00594a3d551a6a095e5616b16 # timeout=10
16 Commit message: "Create service.yml"
17  > git rev-list --no-walk e0d0ad21b1853ec00594a3d551a6a095e5616b16 # timeout=10
```

Activate Windows
Go to Settings to activate Windows.

Not secure 3.91.102.92:8080/job/insuran_project/10/pipeline-overview/?selected-node=11

Jenkins / insuran_project / #10 / Pipeline Overview

Search

- git checkout 0.41s
- build 17s
- docker build image 2.5s
- docker login 0.38s
- docker images pushing 5.7s
- deploy on prod 0.61s

mvn clean package 17s

```
90  [INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0
91  [INFO]
92  [INFO]
93  [INFO] --- maven-jar-plugin:3.2.2:jar (default-jar) @ banking ---
94  [INFO] Building jar: /var/lib/jenkins/workspace/insuran_project/target/banking-0.0.1-SNAPSHOT.jar
95  [INFO]
96  [INFO] --- spring-boot-maven-plugin:2.7.4:repackage (repackage) @ banking ---
97  [INFO] Replacing main artifact with repackaged archive
98  [INFO] ------------------------------------------------------------------------
99  [INFO] BUILD SUCCESS
100 [INFO] ------------------------------------------------------------------------
101 [INFO] Total time:  15.459 s
102 [INFO] Finished at: 2025-08-04T07:12:12Z
103 [INFO] ------------------------------------------------------------------------
```

Activate Windows
Go to Settings to activate Windows.

Not secure 3.91.102.92:8080/job/insuran_project/10/pipeline-overview/?selected-node=16

Jenkins / insuran_project / #10 / Pipeline Overview

Search

- git checkout 0.41s
- build 17s
- docker build image 2.5s
- docker login 0.38s
- docker images pushing 5.7s
- deploy on prod 0.61s

docker build -t pratheep04/banking:1.0 . 2.4s

```
#4 [1/2] FROM docker.io/library/openjdk:11@sha256:99bac5bf83633e3c7399aed725c8415e7b569b54e03e4599e580fc9cdb7c21ab
#4 CACHED

#6 [2/2] COPY target/*.jar app.jar
#6 DONE 1.2s

#7 exporting to image
#7 exporting layers
#7 exporting layers 0.2s done
#7 writing image sha256:d1cdf0093a45aedeb9f4c727decfd653c23b42ee679630fdb0309b303e033166 done
#7 naming to docker.io/pratheep04/banking:1.0 done
#7 DONE 0.3s
```

Activate Windows
Go to Settings to activate Windows.

Not secure 3.91.102.92:8080/job/insuran_project/10/pipeline-overview/?selected-node=21

Jenkins / insuran_project / #10 / Pipeline Overview

docker build image docker login docker images pushing deploy on prod

Search

- git checkout 0.41s
- build 17s
- docker build image 2.5s
- docker login 0.38s
- docker images pushing 5.7s
- deploy on prod 0.61s

docker login 0.38s Started 21m ago Jenkins

docker login -u pratheep04 -p ${dockercreds} 0.27s

```
+ docker login -u pratheep04 -p ****
WARNING! Using --password via the CLI is insecure. Use --password-stdin.
Login Succeeded
```

Activate Windows
Go to Settings to activate Windows.

Not secure 3.91.102.92:8080/job/insuran_project/10/pipeline-overview/?selected-node=30

Jenkins / insuran_project / #10 / Pipeline Overview

Search

- git checkout 0.41s
- build 17s
- docker build image 2.5s
- docker login 0.38s
- docker images pushing 5.7s
- deploy on prod 0.61s

docker push pratheep04/banking:1.0 5.7s

```
8  03127cdb479b: Preparing
9  9c742cd6c7a5: Preparing
10 293d5db30c9f: Waiting
11 03127cdb479b: Waiting
12 9c742cd6c7a5: Waiting
13 826c3ddbb29c: Layer already exists
14 7b7f3078e1db: Layer already exists
15 b626401ef603: Layer already exists
16 9b55156abf26: Layer already exists
17 03127cdb479b: Layer already exists
18 293d5db30c9f: Layer already exists
19 9c742cd6c7a5: Layer already exists
20 524c8a9e7299: Pushed
21 1.0: digest: sha256:6345bf075ad8b430315670ac8722d0e95d49e96b9b2c31fa9affcc90594f8499 size: 2007
```

Activate Windows
Go to Settings to activate Windows.

Not secure 3.91.102.92:8080/job/insuran_project/10/pipeline-overview/?selected-node=36

Jenkins / insuran_project / #10 / Pipeline Overview

Search

- git checkout 0.41s
- build 17s
- docker build image 2.5s
- docker login 0.38s
- docker images pushing 5.7s
- deploy on prod 0.61s

deploy on prod 0.61s Started 22m ago Jenkins

sudo kubectl apply -f deployment.yml 0.28s

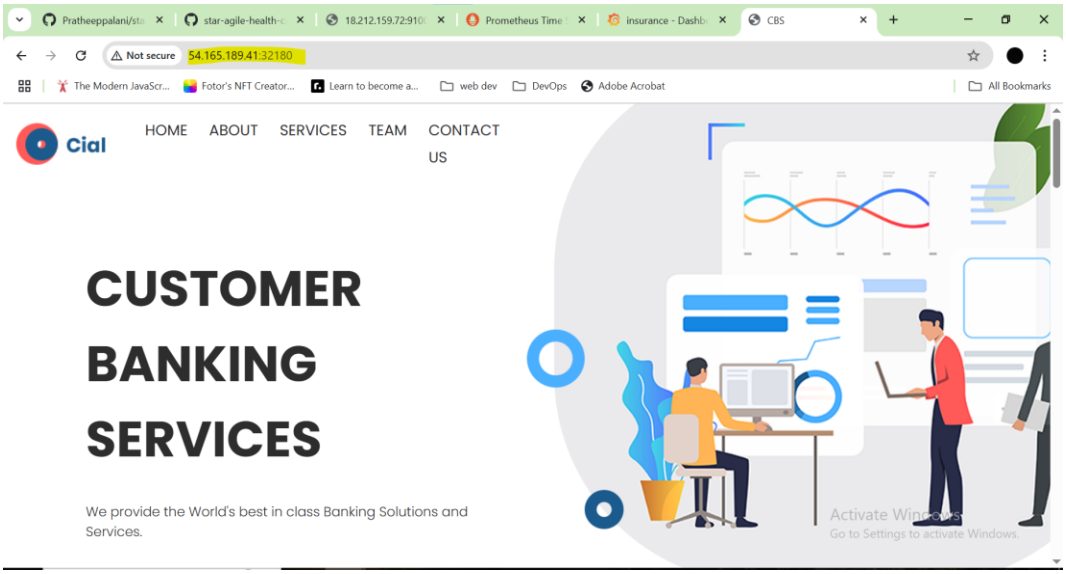
```
0 + sudo kubectl apply -f deployment.yml
1 deployment.apps/insureme created
```

sudo kubectl apply -f service.yml 0.28s

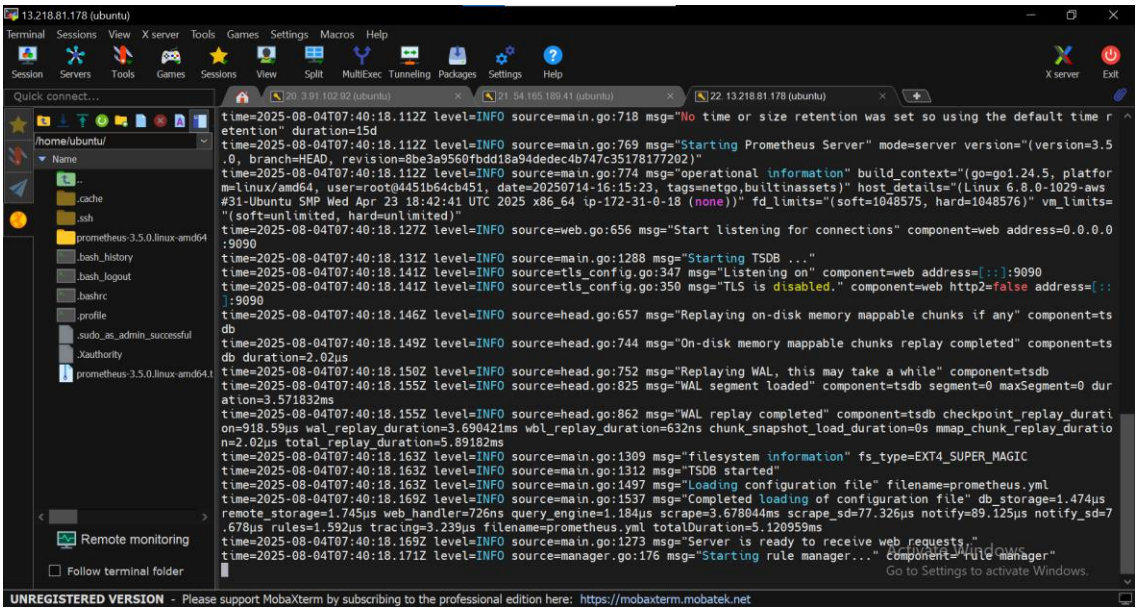
```
0 + sudo kubectl apply -f service.yml
1 service/insureme created
```

Activate Windows
Go to Settings to activate Windows.

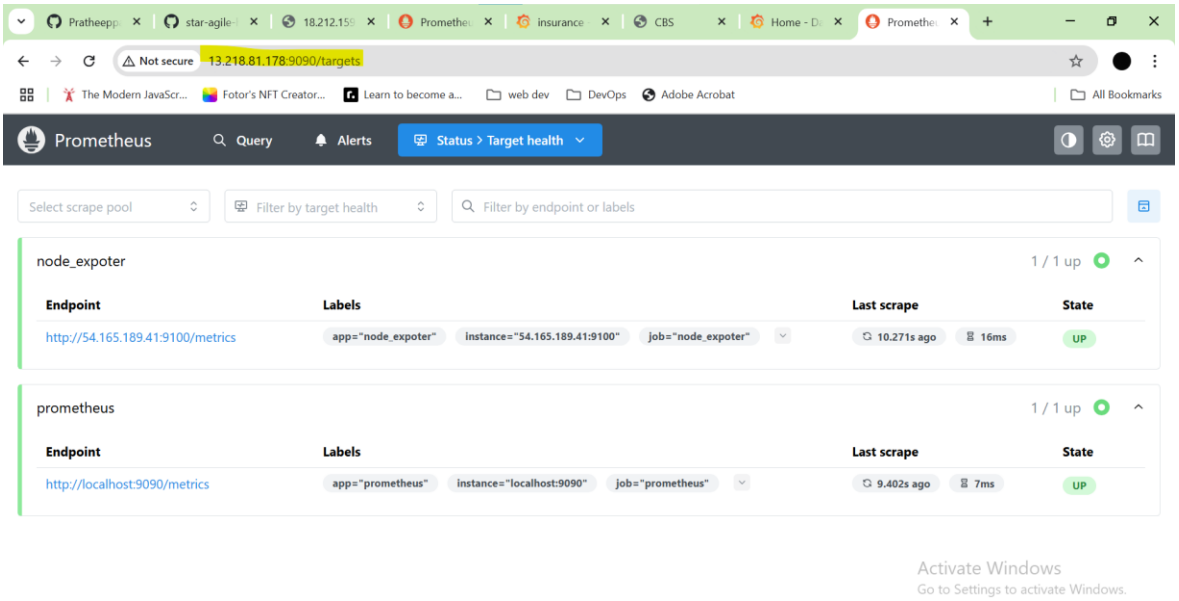
Step 16 : Now paste the prod machine public ip with node prot 32180. The application is successfully deployed.



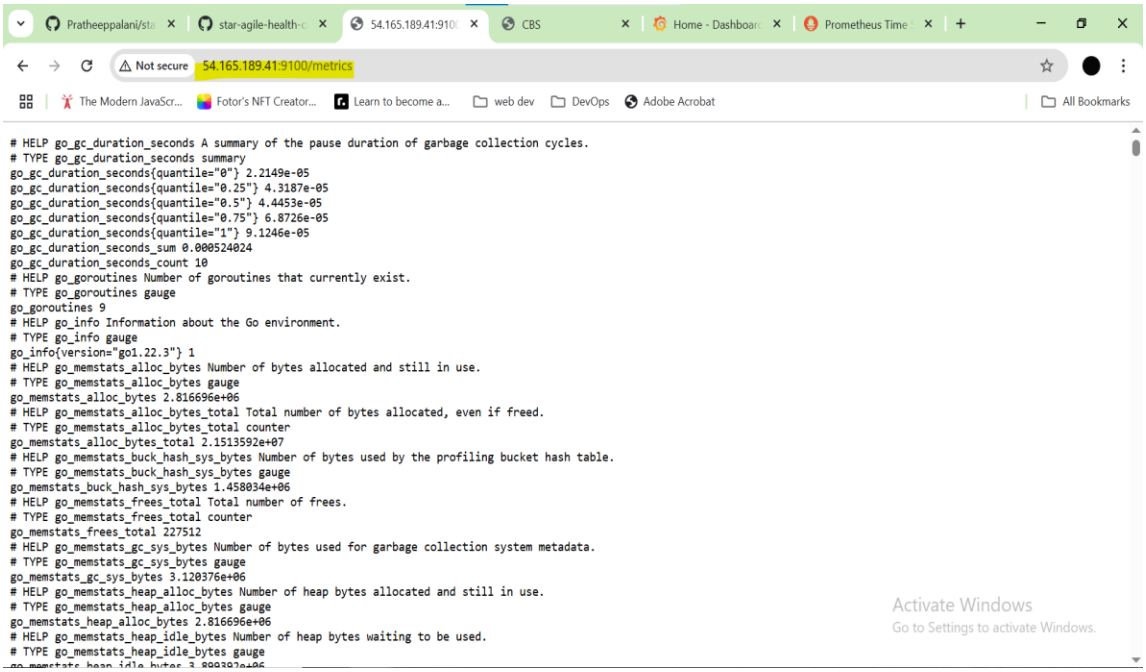
Step 17 : Now we have to monitor the system, for that start grafana and Prometheus in monitoring machine.



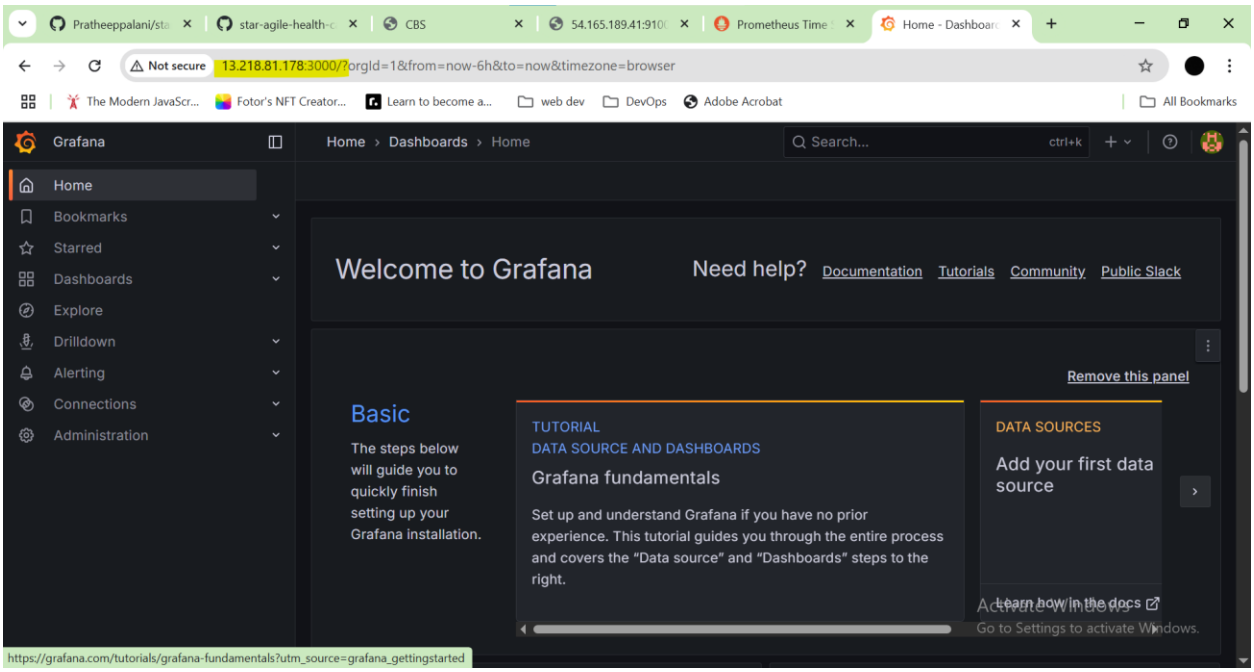
Step 18 : After you started it, paste the public ip of monitoring machine with prot number 9090. Now you can node expoter is up and running.



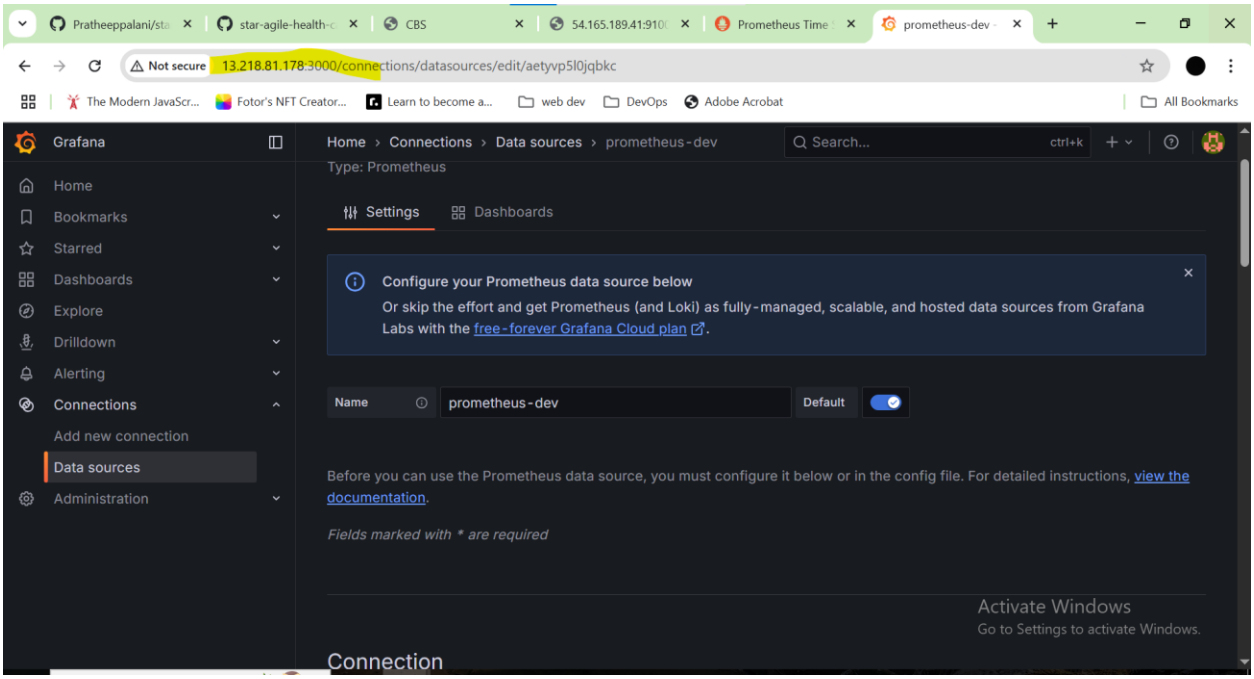
Step 19 : we got the metrics from prod server and its loaded into Prometheus.



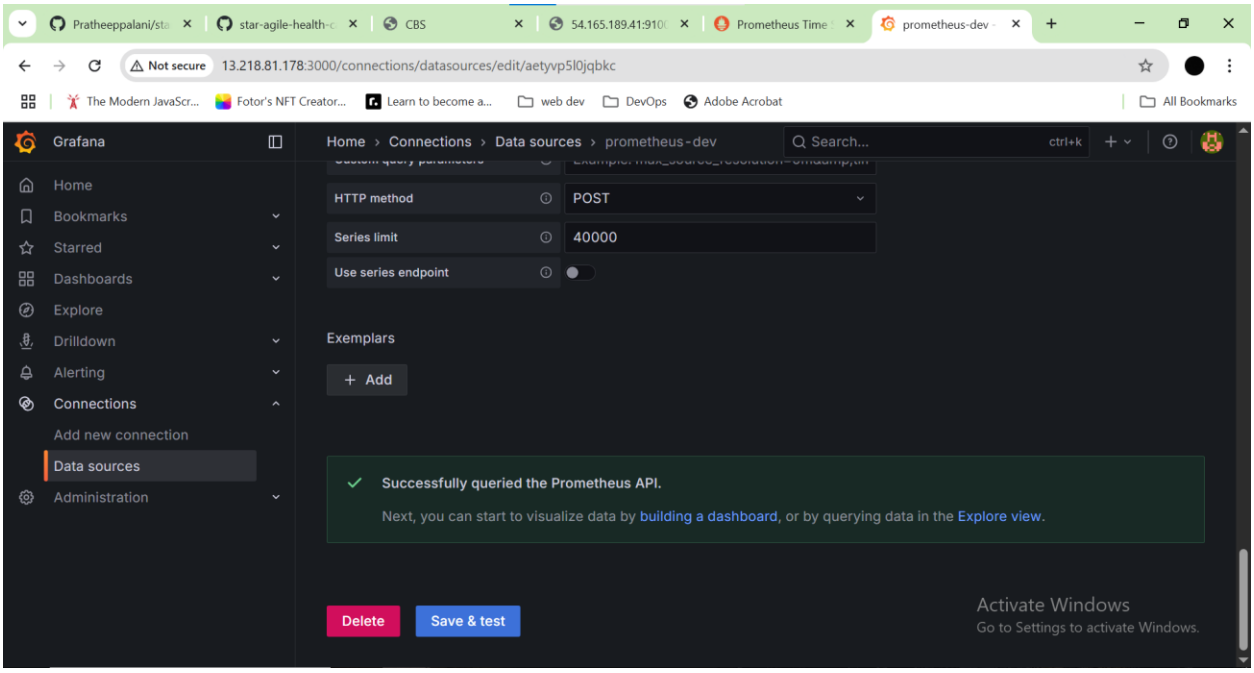
Step 20 : Now we have to display the metrics through grafana. Launch grafana in the browser.



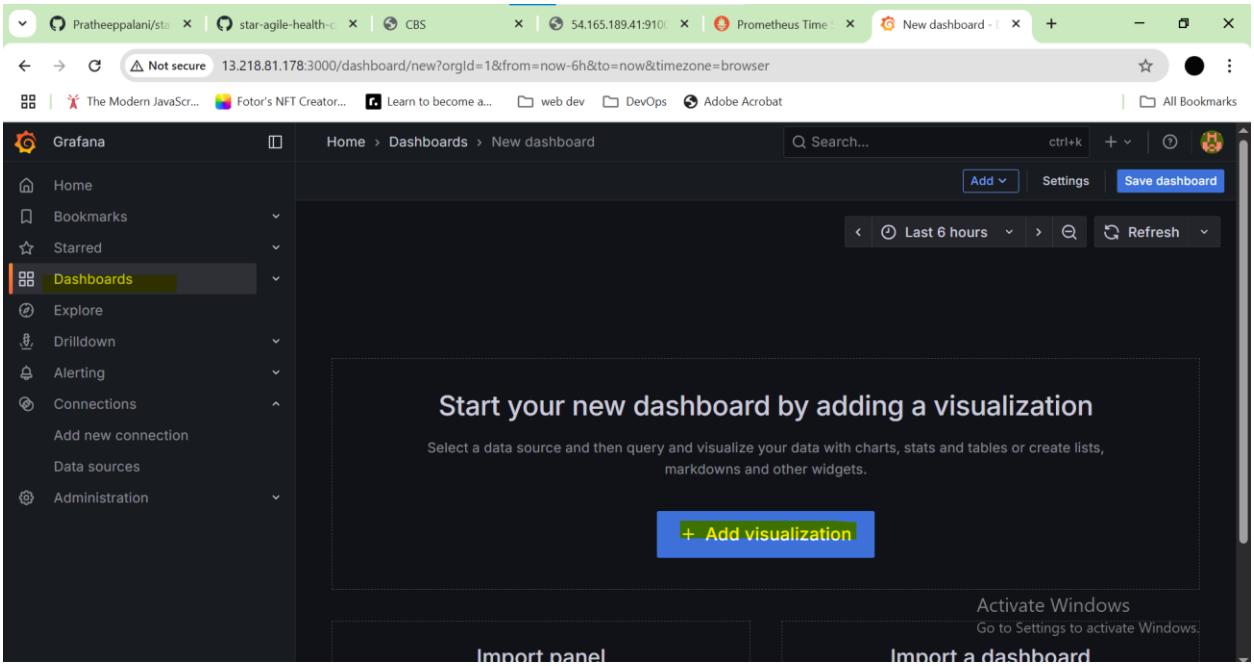
Step 21 : click on data source and Prometheus in it. Enter the name and paste the Prometheus url.



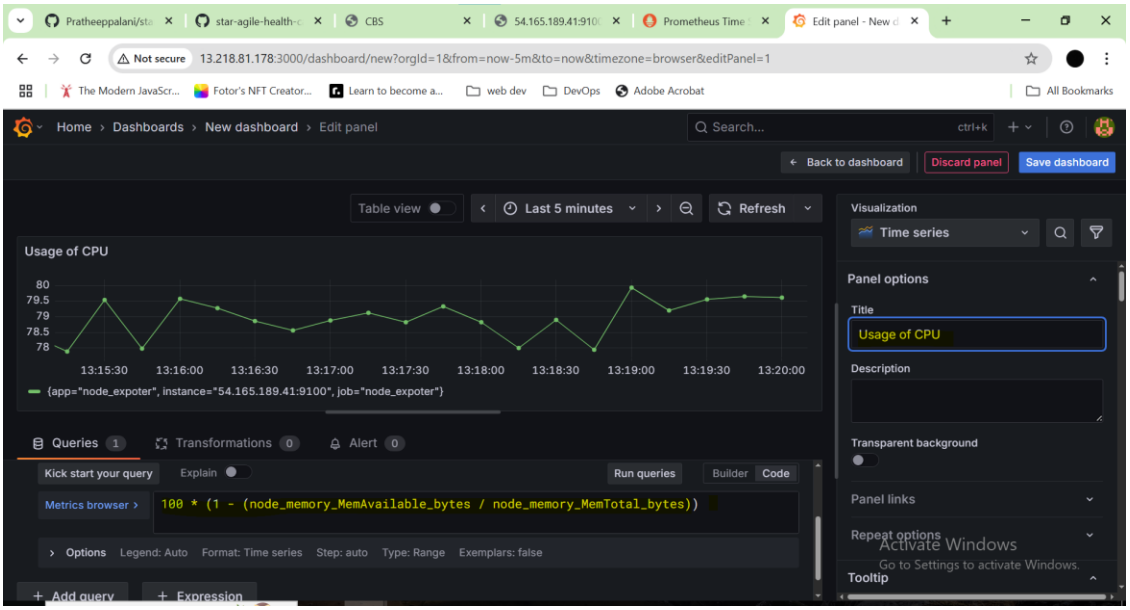
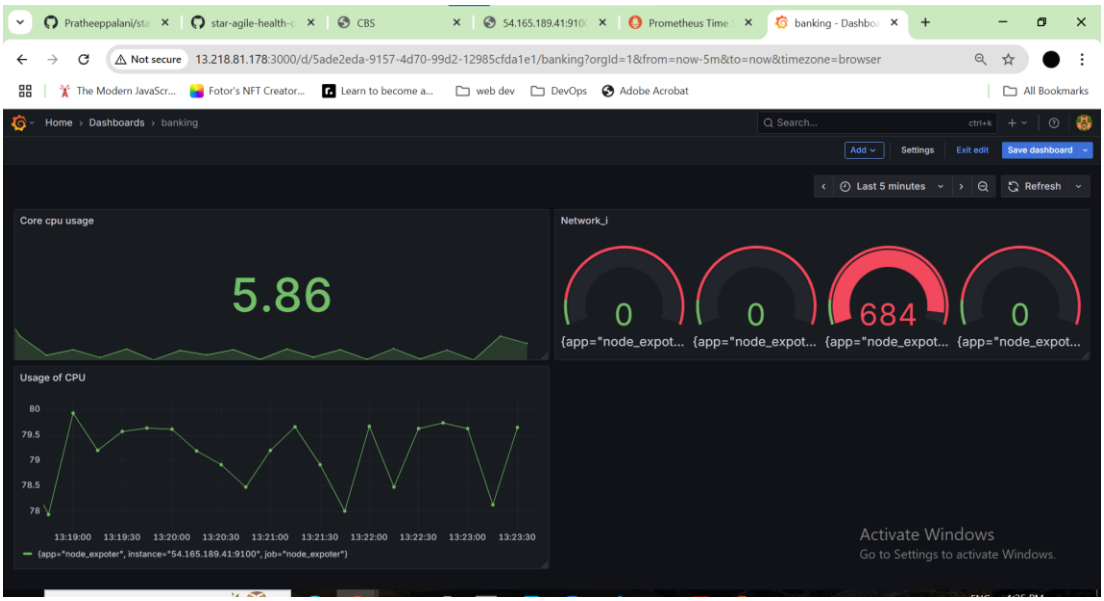
Step 22 : Click on save & test



Step 23 : Goto dashboard click on add visualization.



Step 24 : Enter the title and metrics in the metric browser and run the code, changes the timeline and save dashboard.



Step 25 : Repeat the same process with some other metrics, cpu usage, cpu core usage, network incoming traffic.