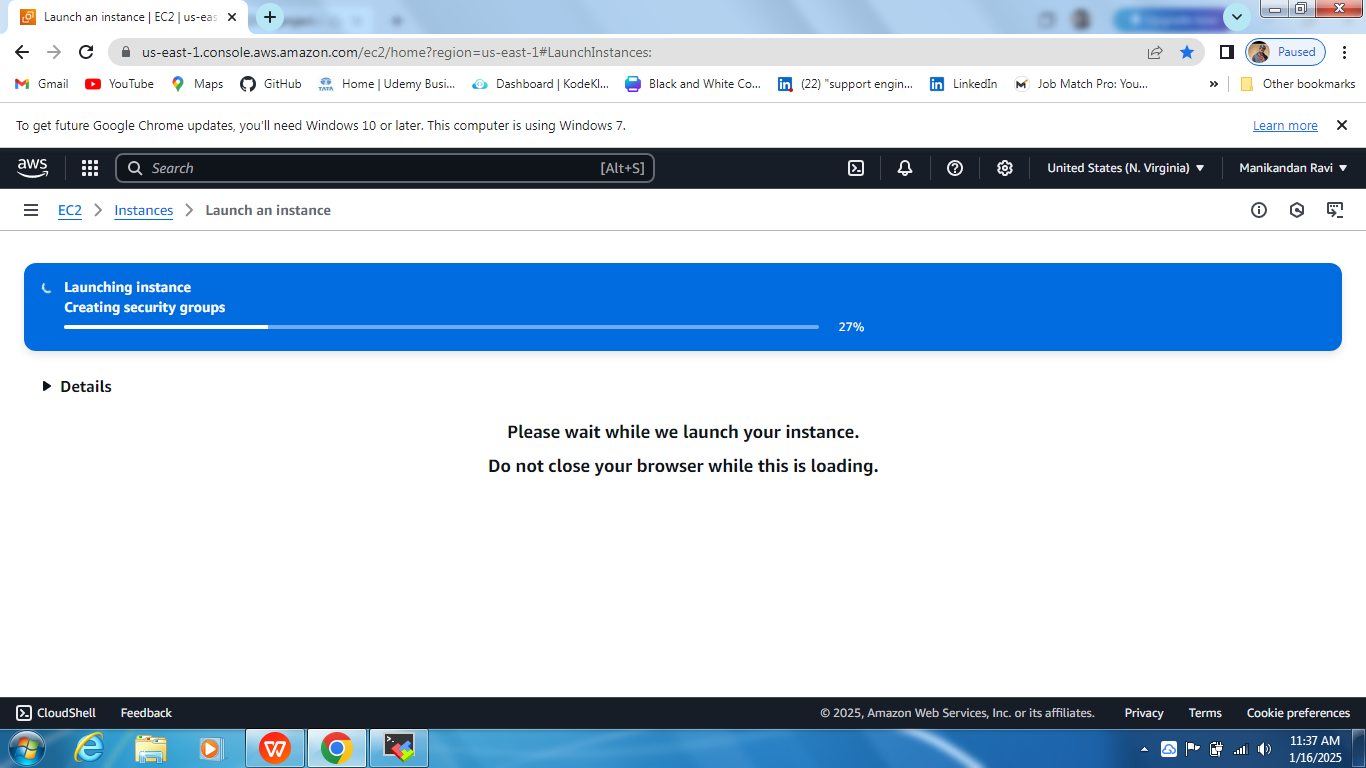
## **DevOps CI/CD Pipeline Project**

### 1. **Introduction**

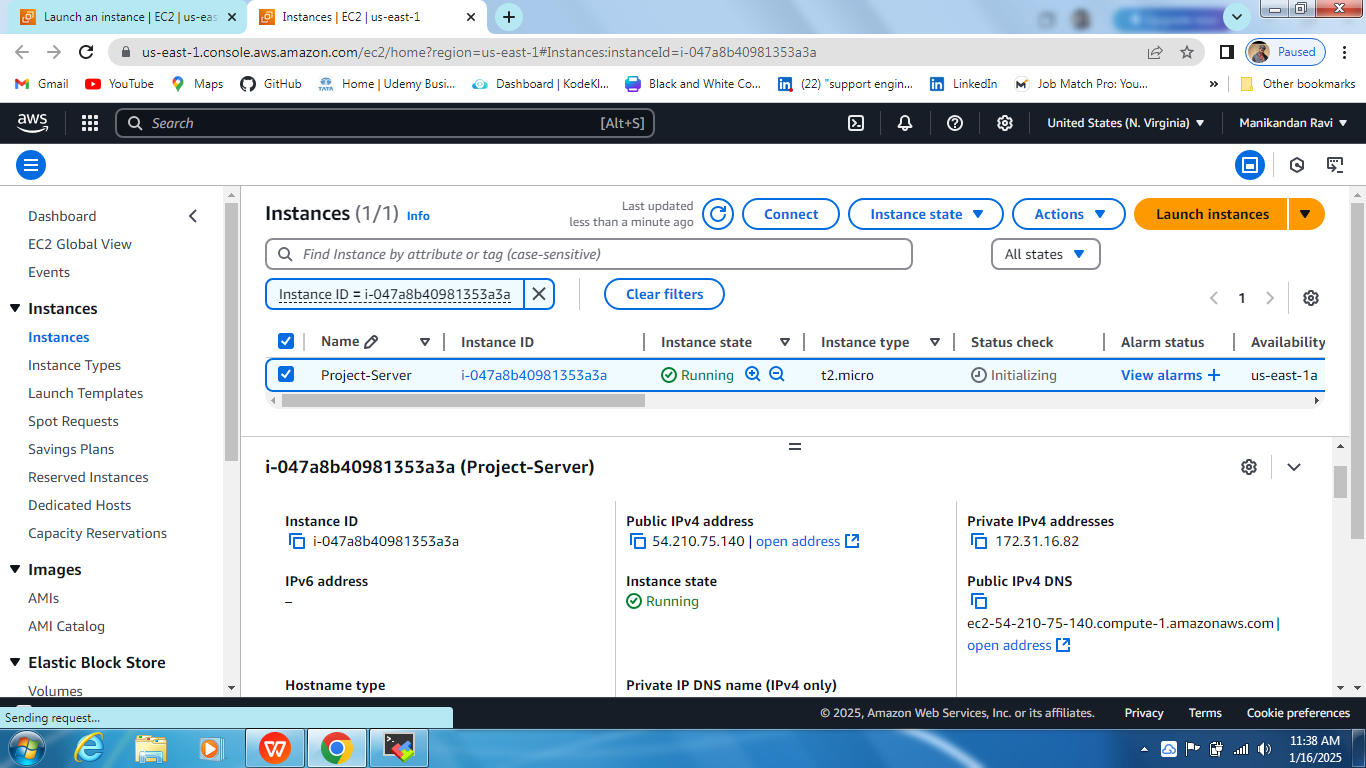
This project demonstrates setting up a CI/CD pipeline using Jenkins on AWS. The pipeline automates code integration, building, testing, and deployment processes, ensuring efficient and reliable delivery of software.

### 2. Setting Up the Environment

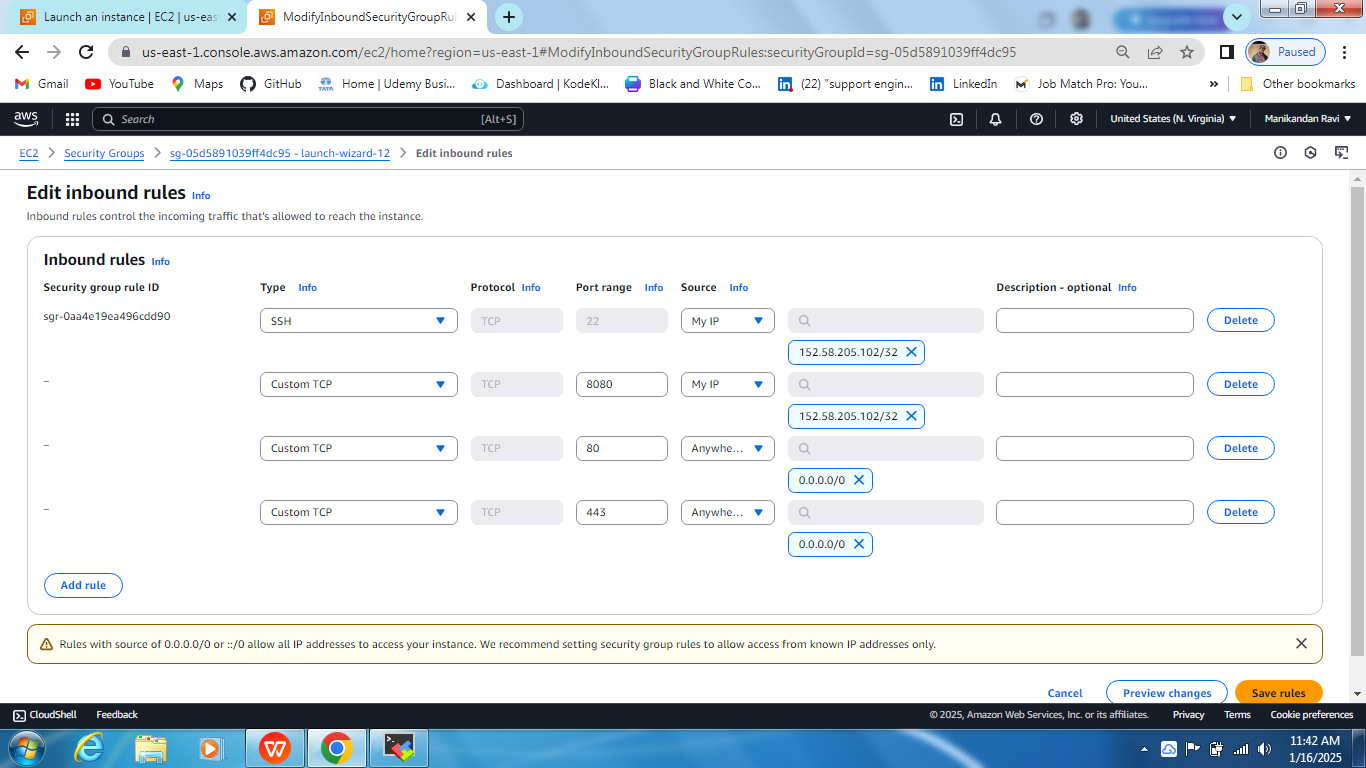


#### ****2.1 Creating an EC2 Instance****

* Launched an EC2 instance on AWS.

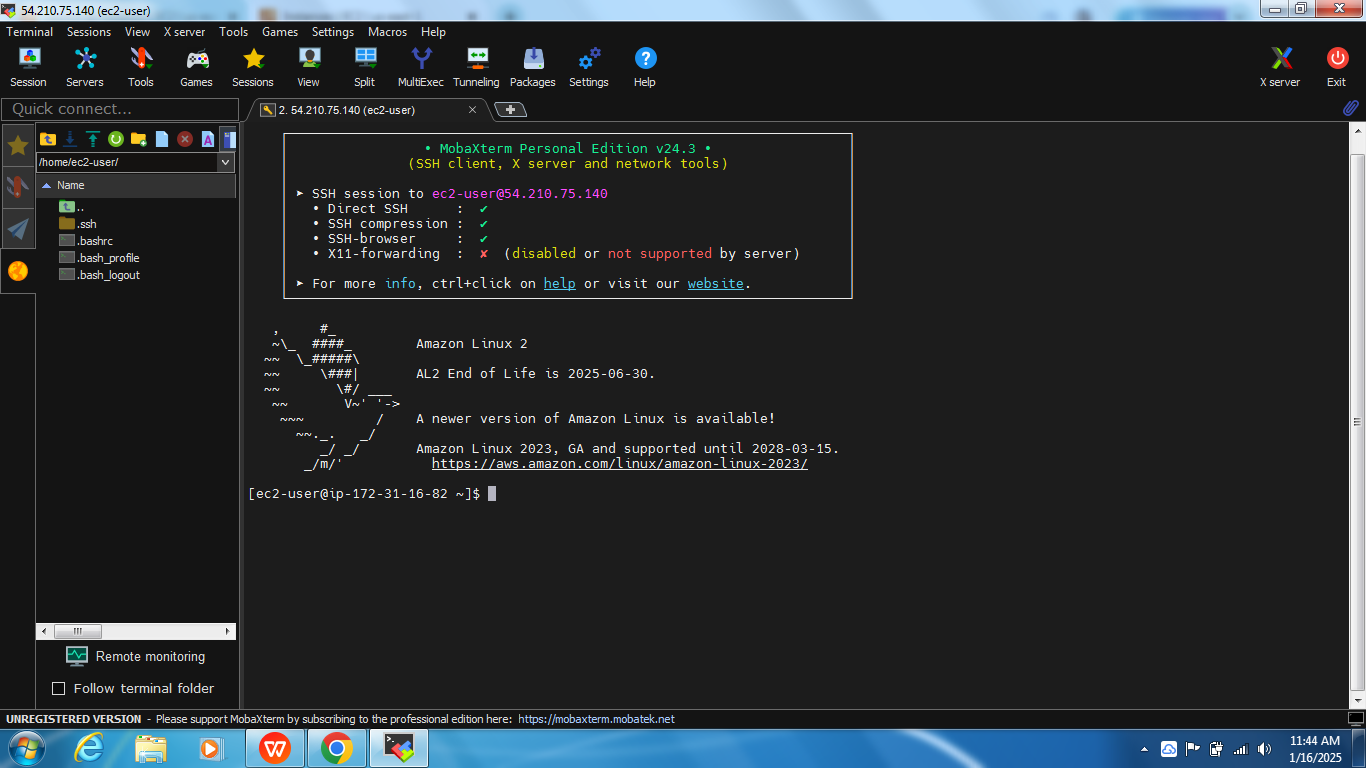


* Configured security group rules to allow required ports (e.g., SSH, HTTP, and Jenkins).



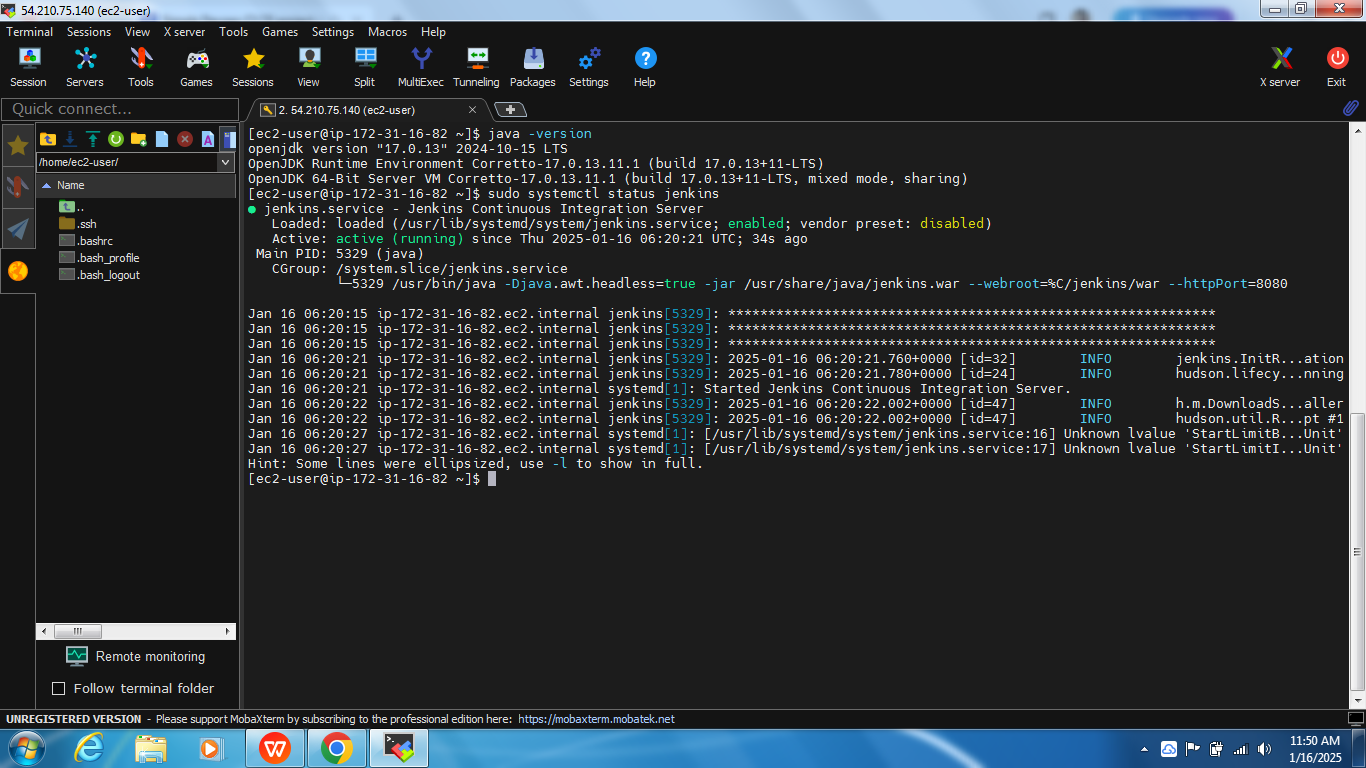
#### ****2.2 Connecting to the Instance****

* Connected to the server via SSH.



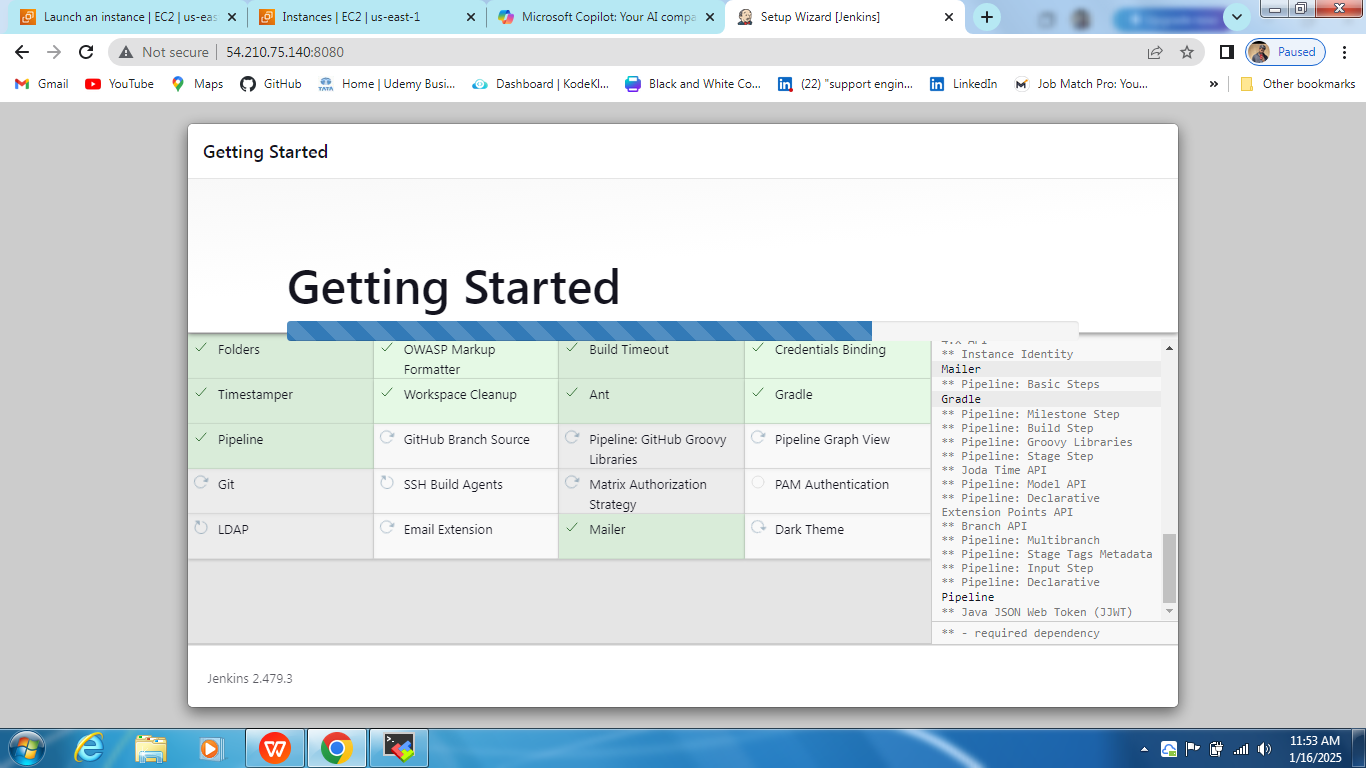
#### 2.3 ****Installing Jenkins and Java****

* sudo yum update -y
* sudo yum install java-17-amazon-corretto-devel -y
* sudo yum install java-1.8.0-openjdk -y
* sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
* sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
* sudo yum install jenkins -y
* sudo systemctl start jenkins
* sudo systemctl enable jenkins



#### 2.4 ****Accessing Jenkins****

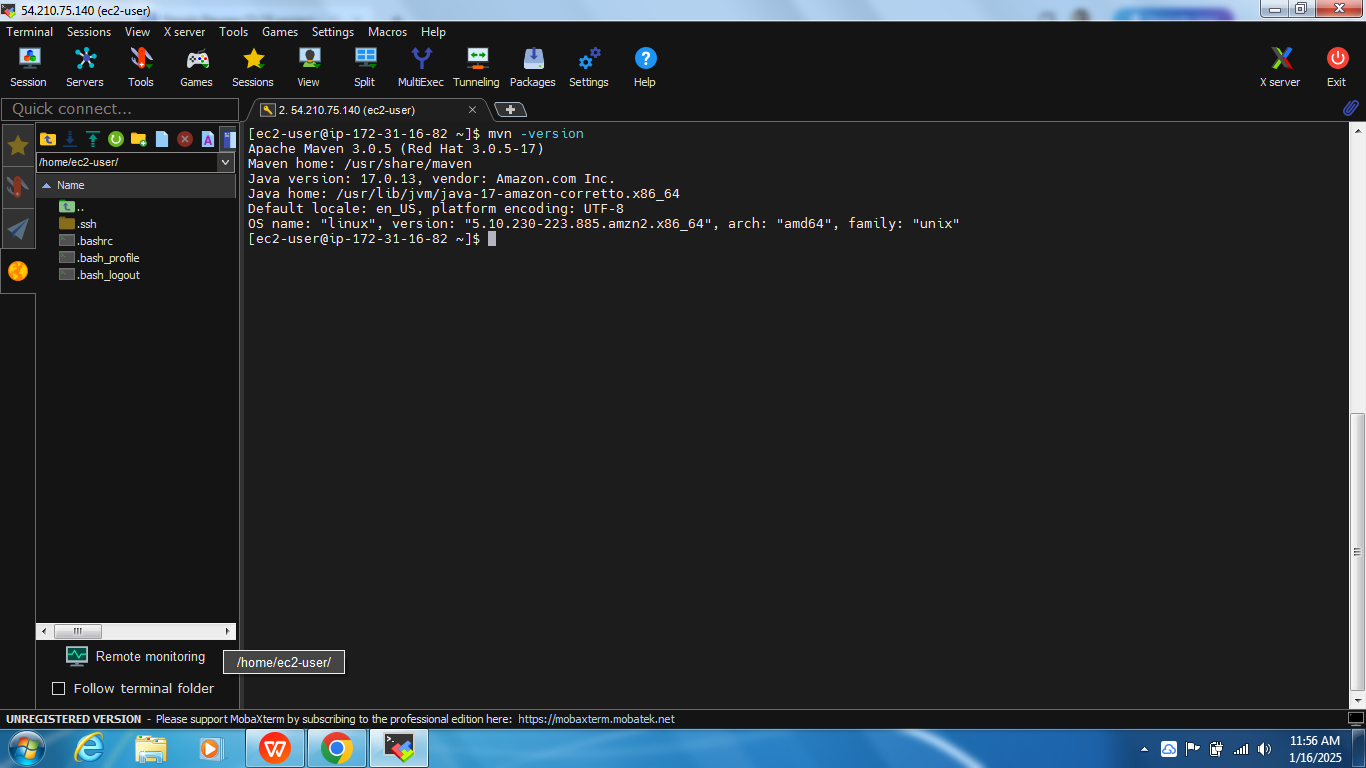
* Opened the Jenkins server on port 8080



### 3.Installing Required Tools

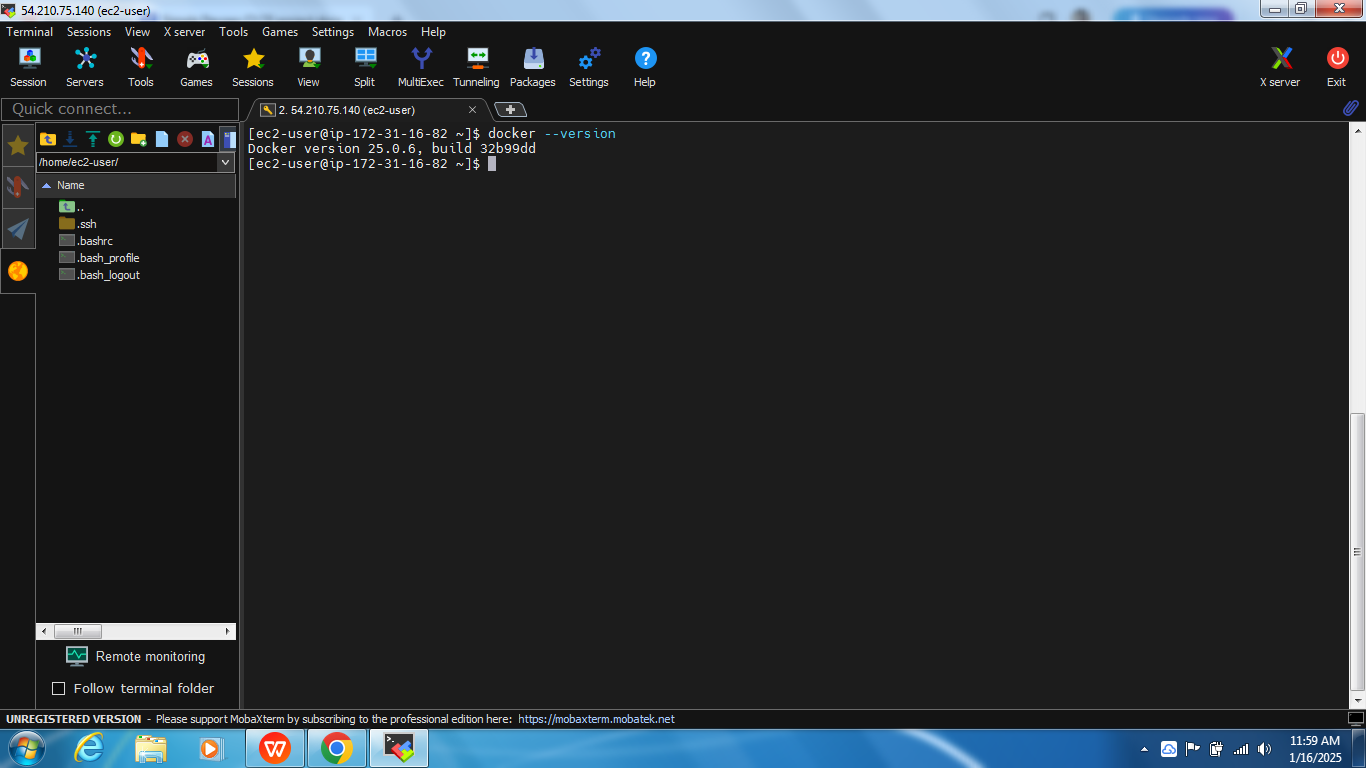
#### 3.1 ****Maven Installation****

* sudo yum install maven -y
* mvn -version



#### 3.2 ****Docker Installation****

* sudo yum install docker -y
* sudo systemctl start docker && sudo systemctl enable docker



### 4. Configuring Jenkins

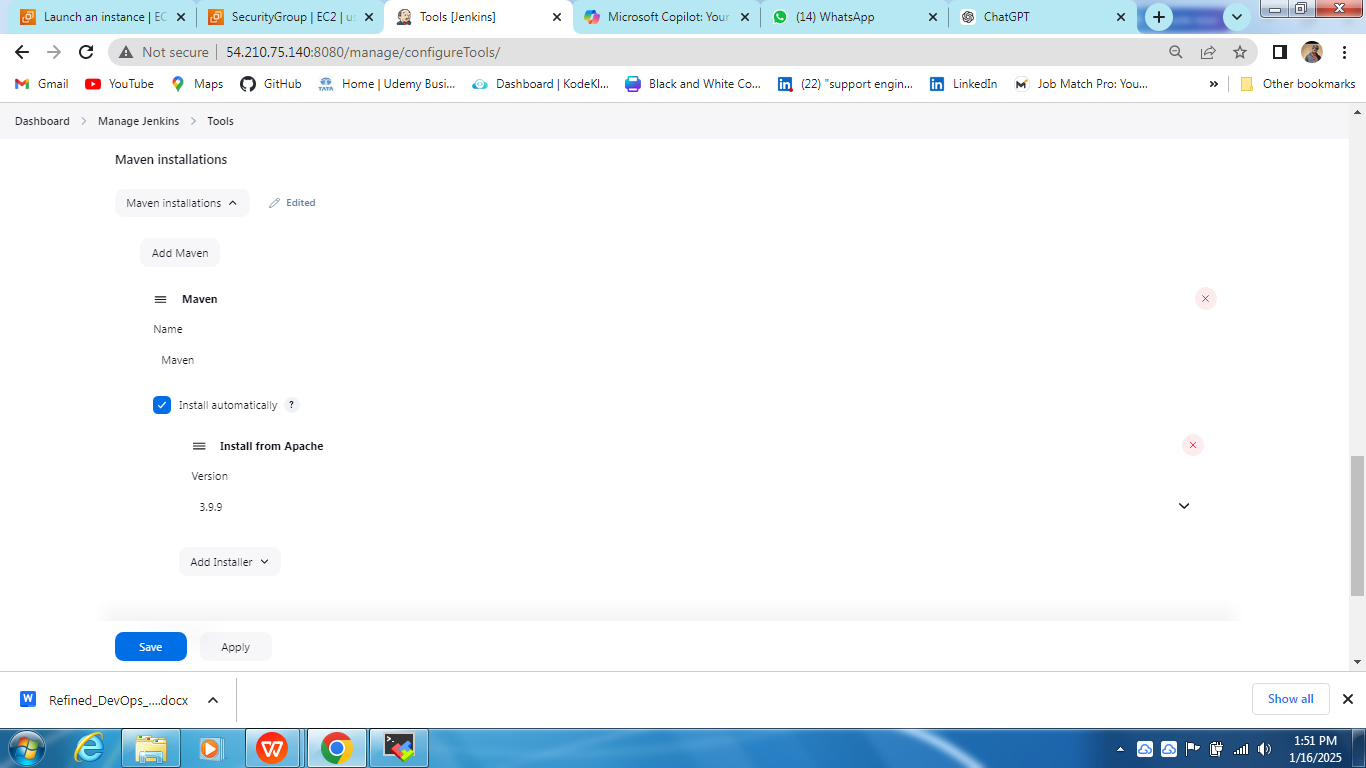
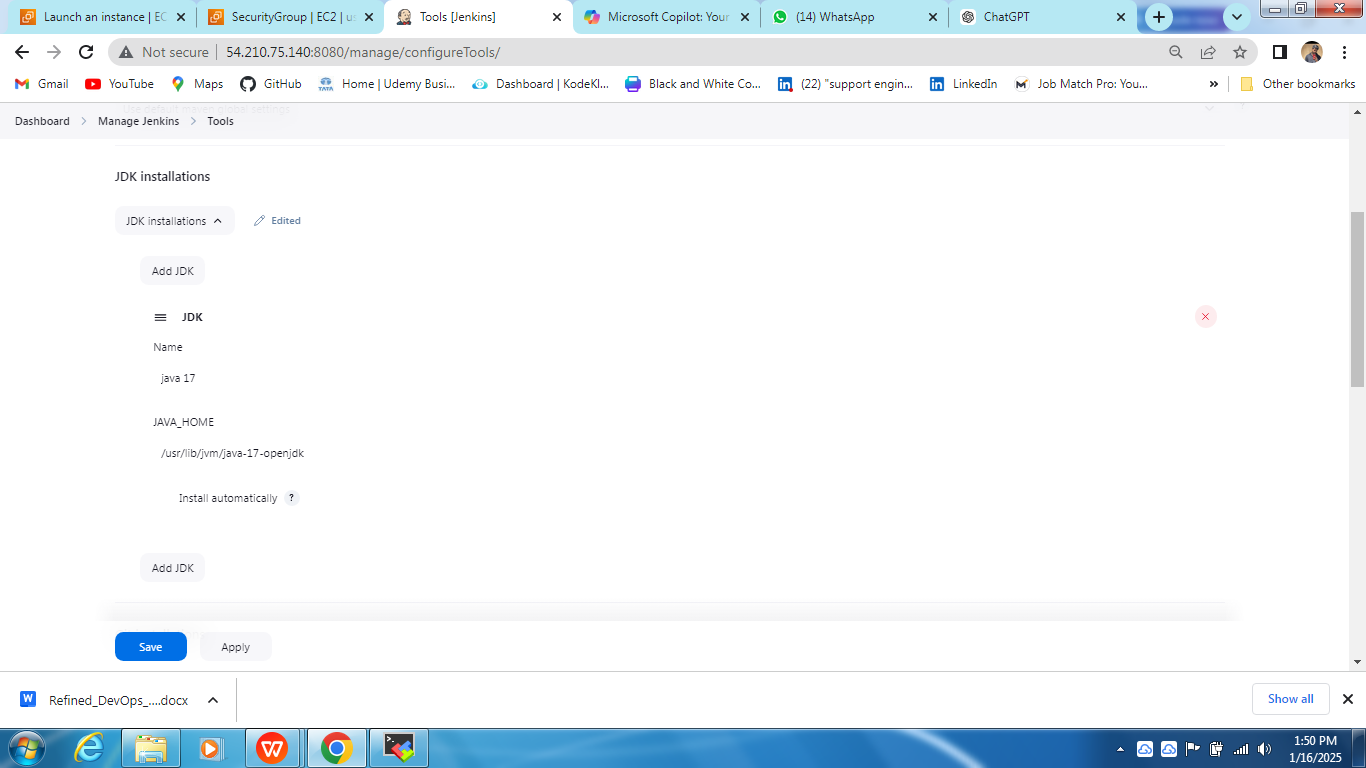
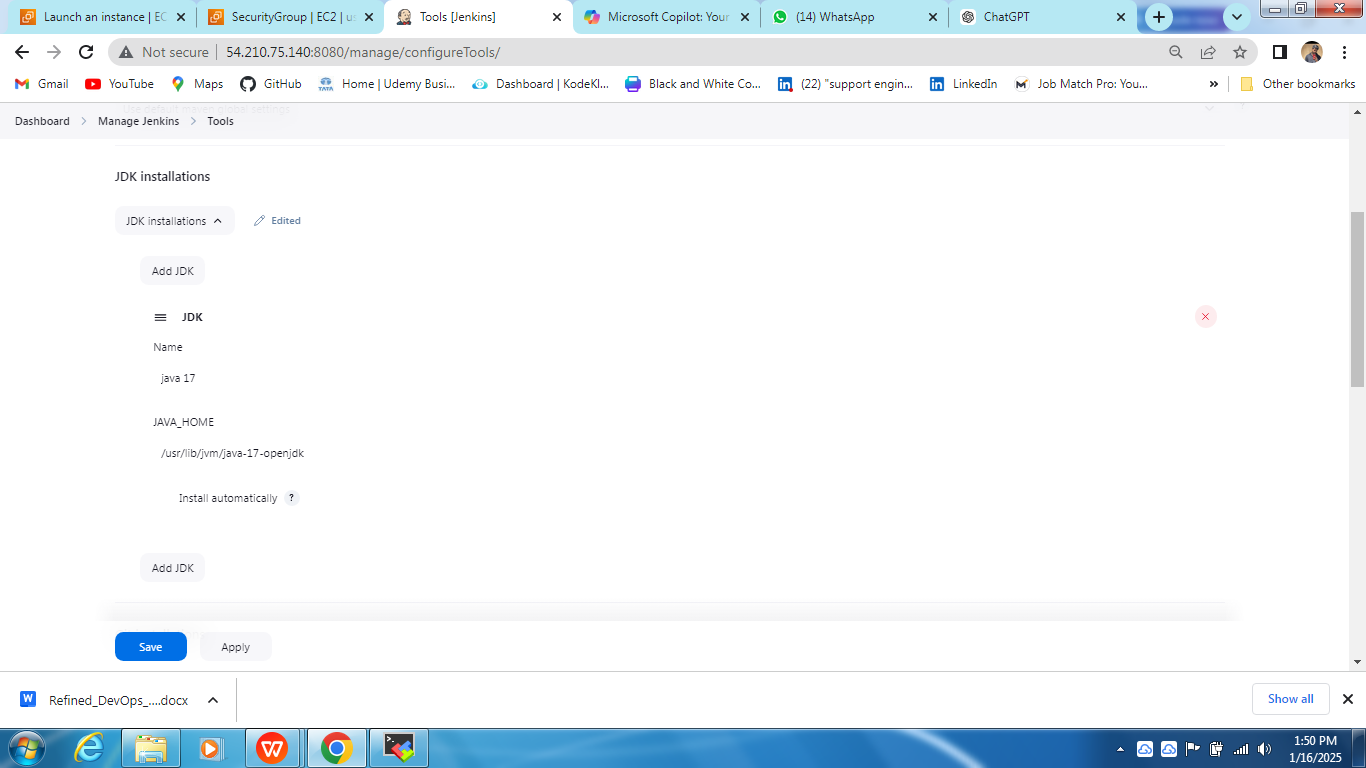
#### 4.1 ****Installing Plugins****

Through Jenkins Plugin Manager, installed:

* Maven Integration Plugin
* Git Plugin
* Docker Pipeline Plugin

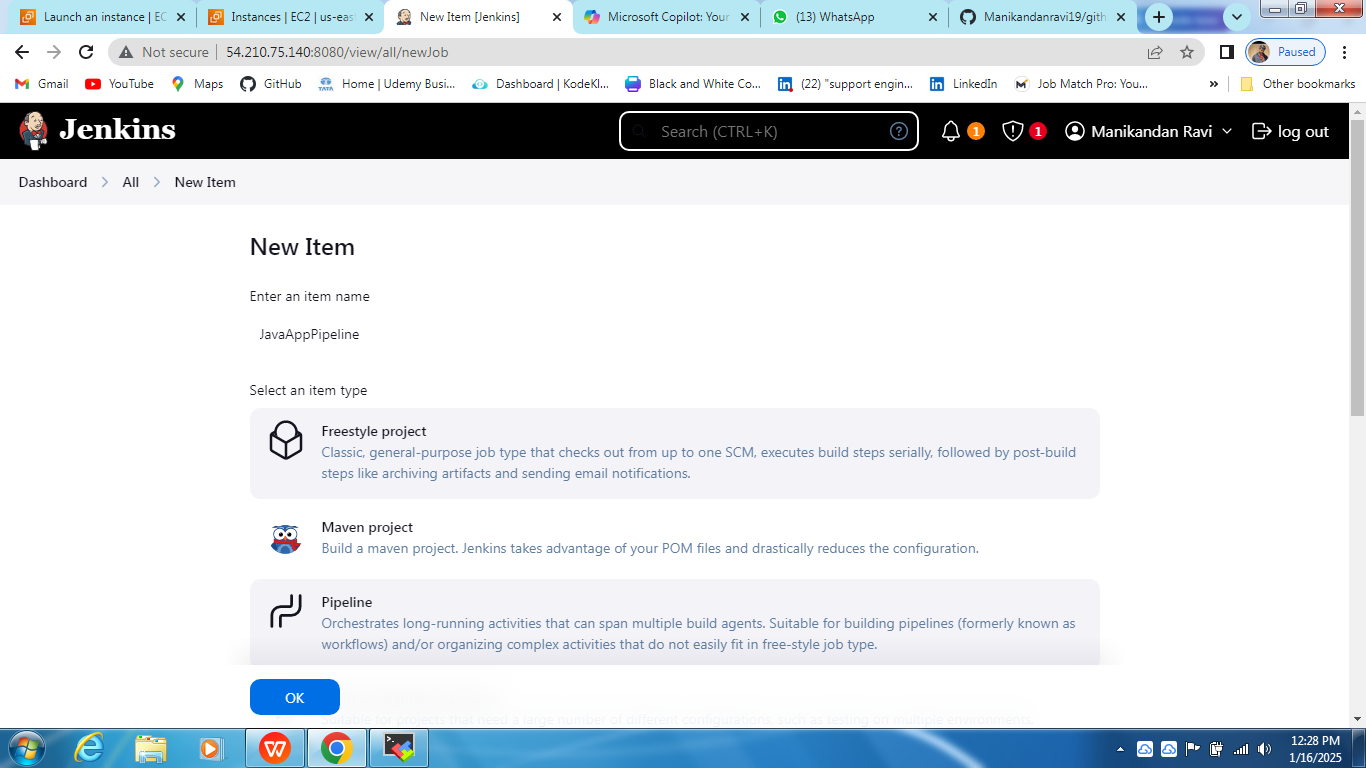
#### 4.2 ****Configuring Paths****

* Setting up a Maven, Git, and Java paths in Jenkins settings

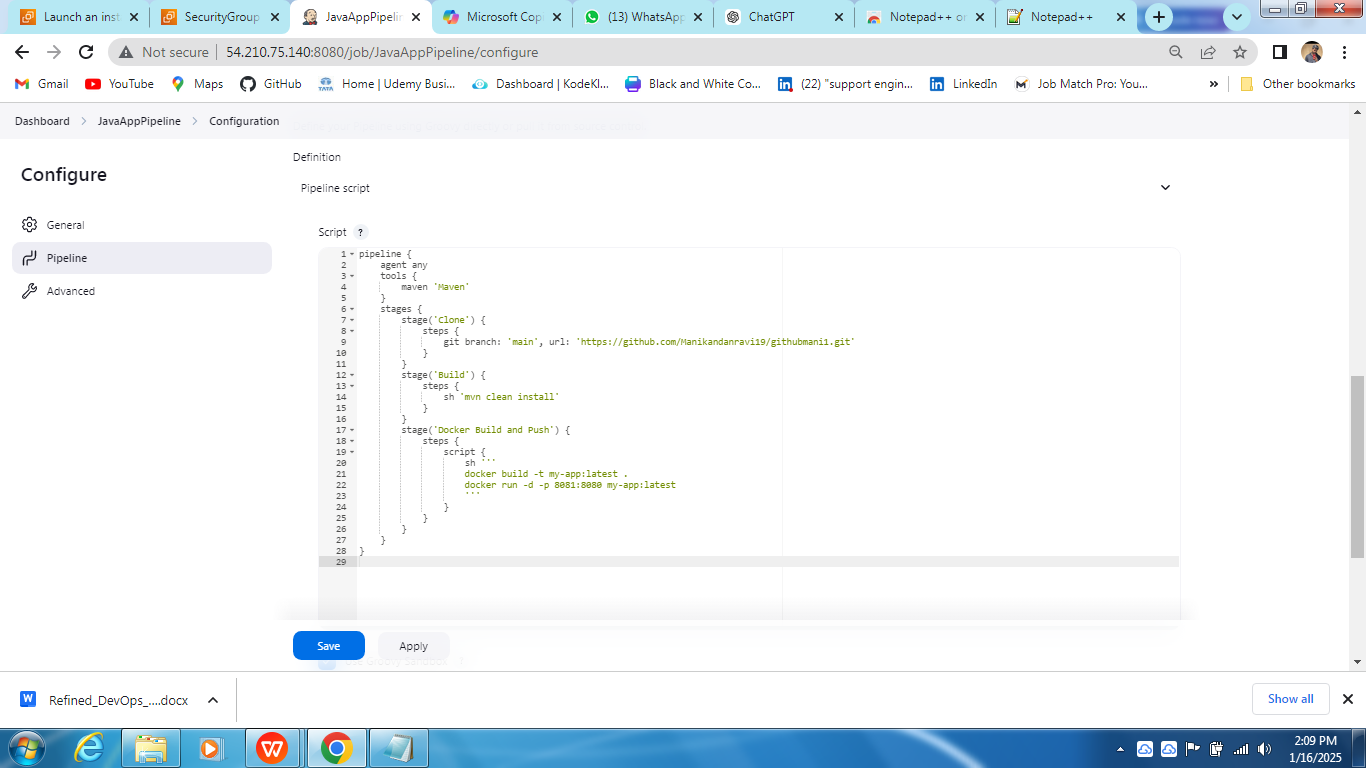


### 5.Creating the Pipeline

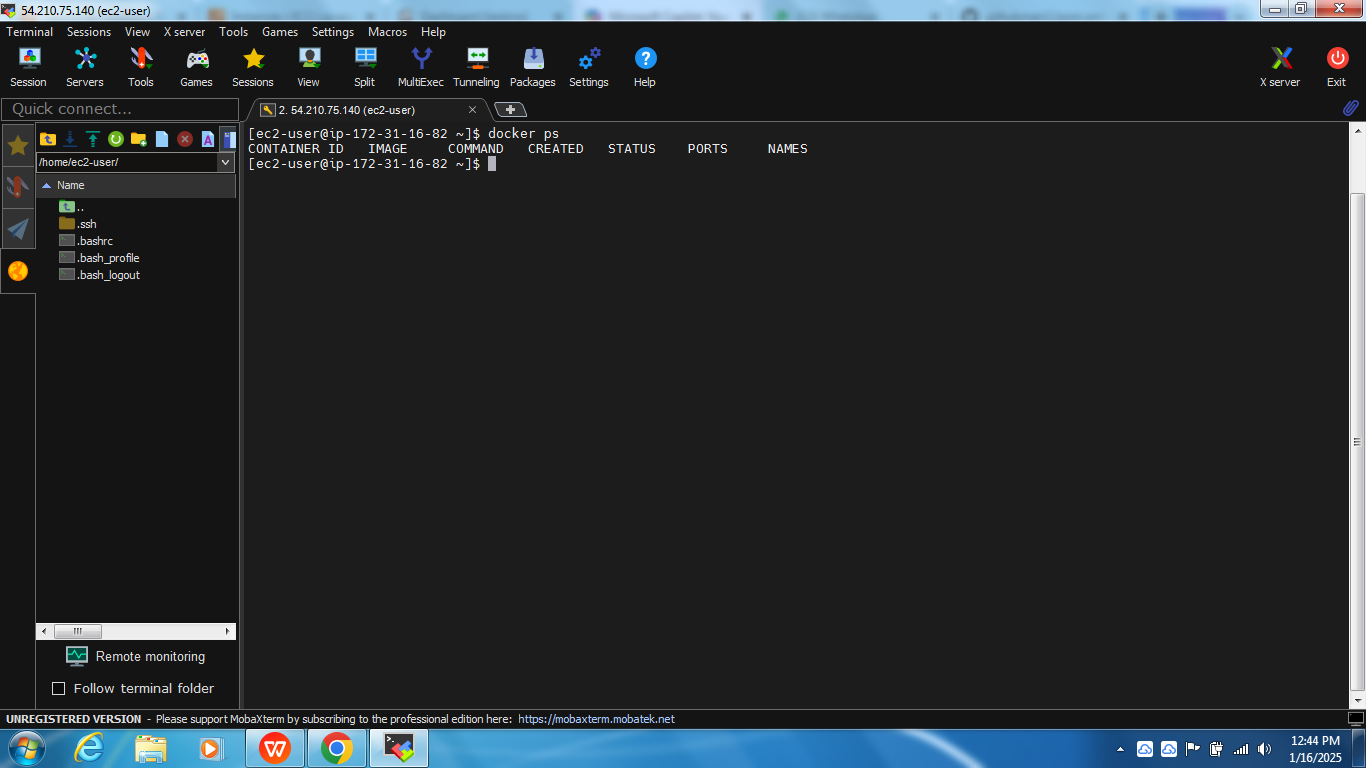
#### 5.1 ****Pipeline Configuration****



* Creating a new Jenkins item and selected "Pipeline Project."
* Below is the script used



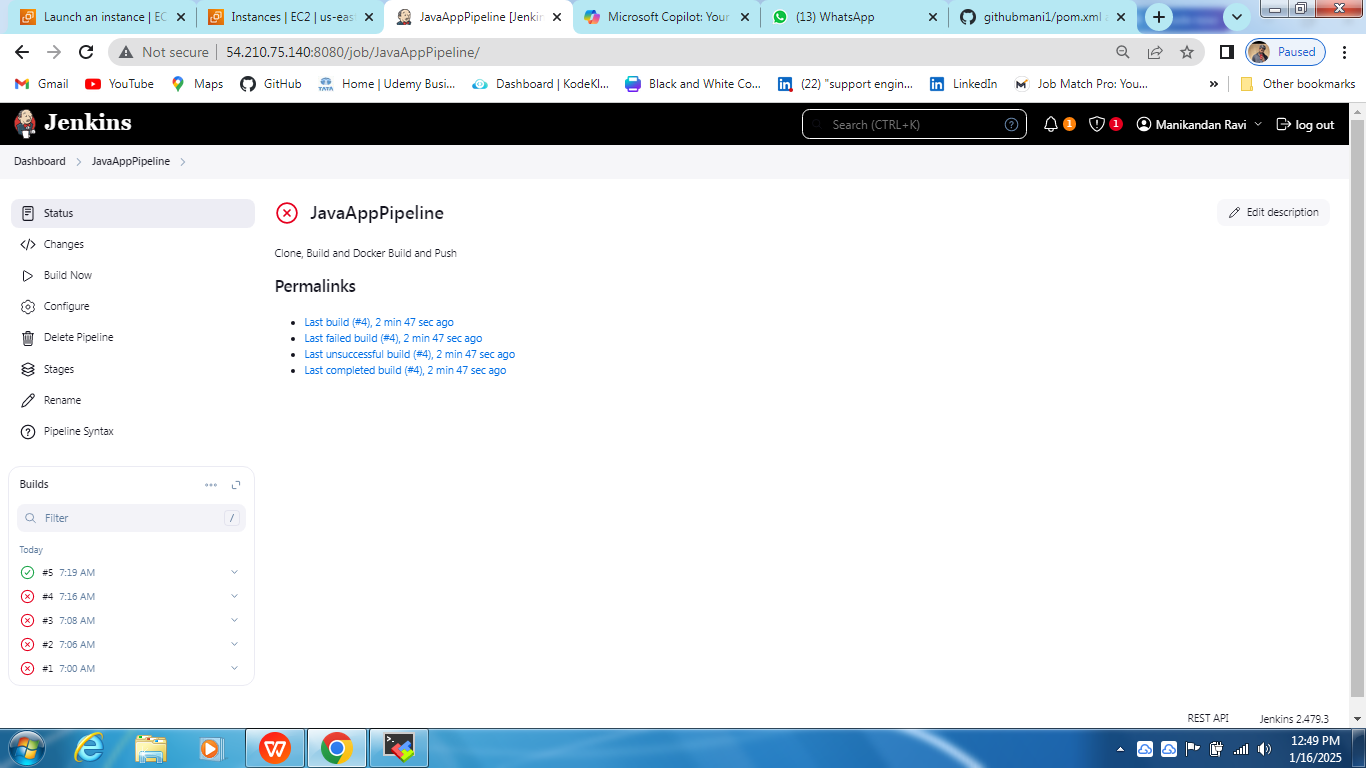
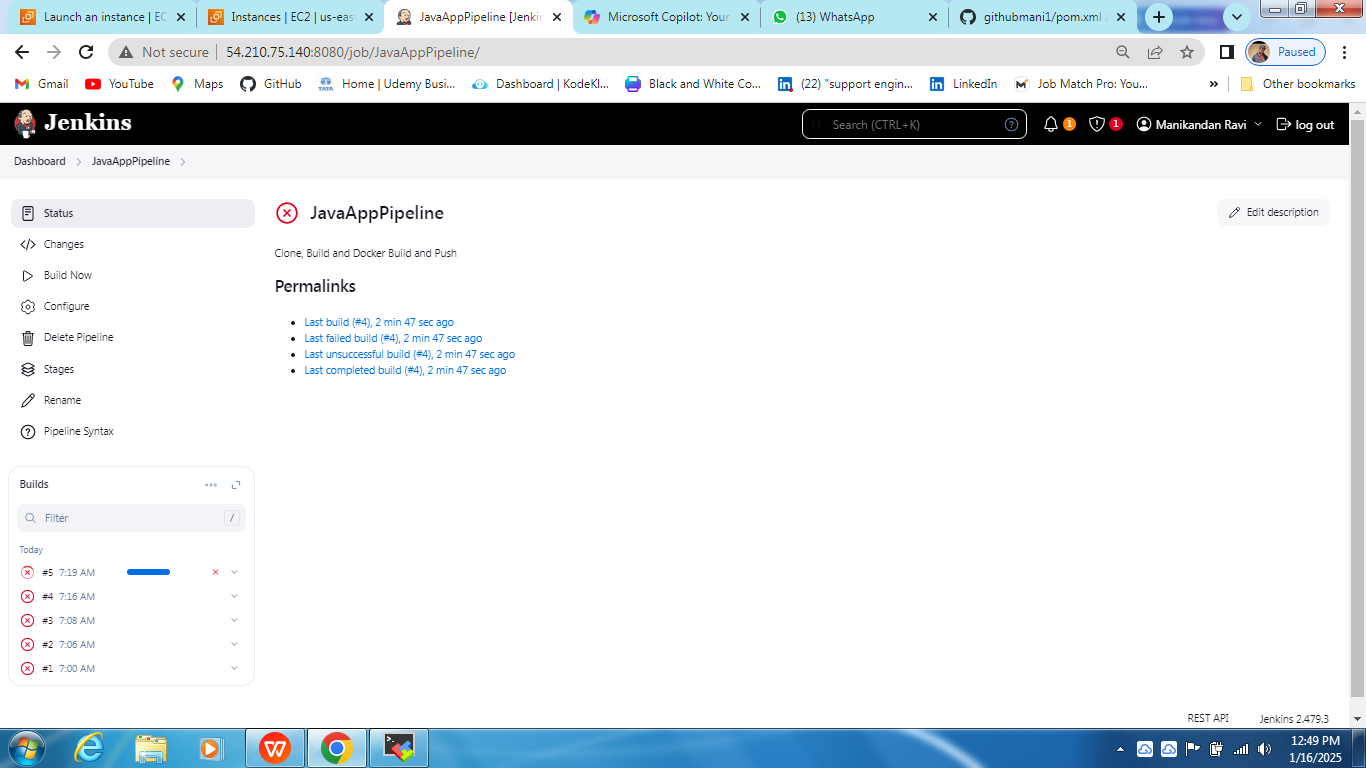
* Before running the pipeline there is no docker image running



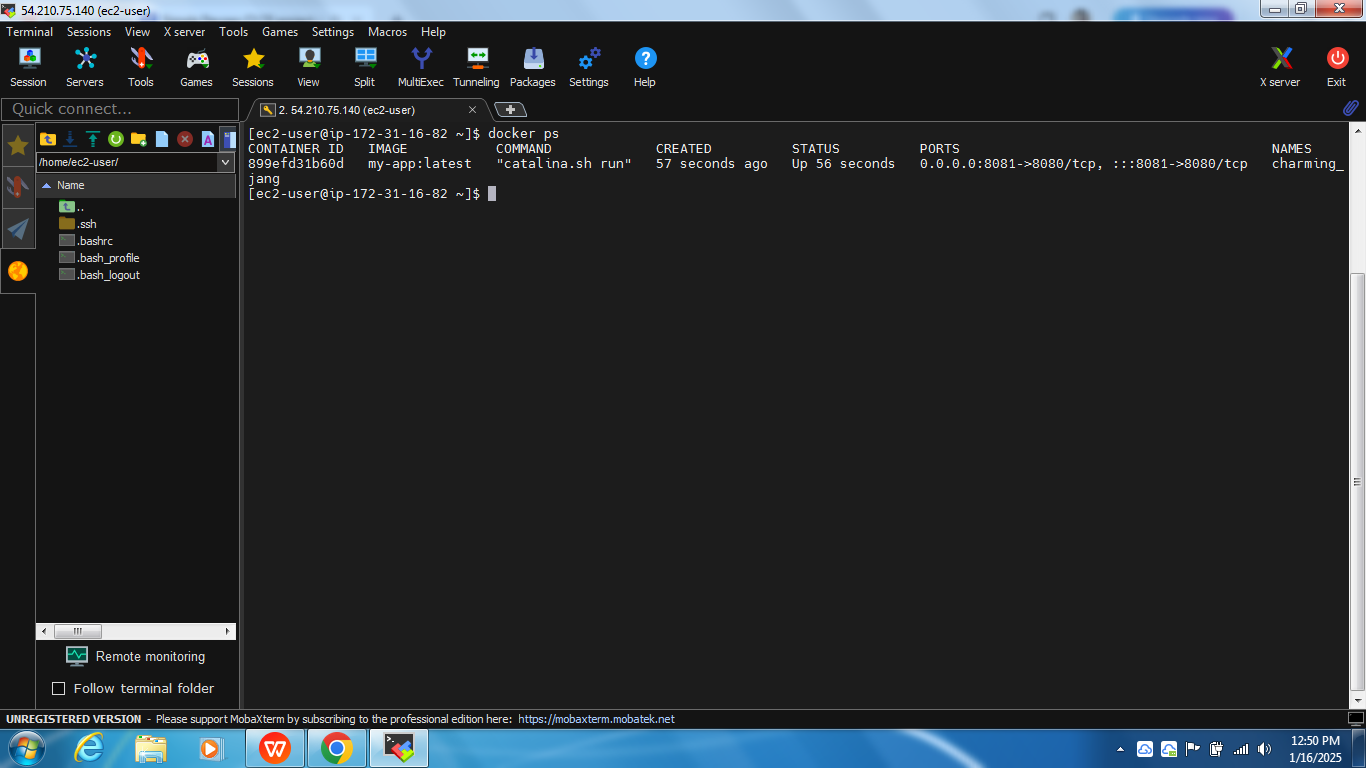
### Running the Pipeline

#### ****6.1 Execution****

* Built the Jenkins job successfully.

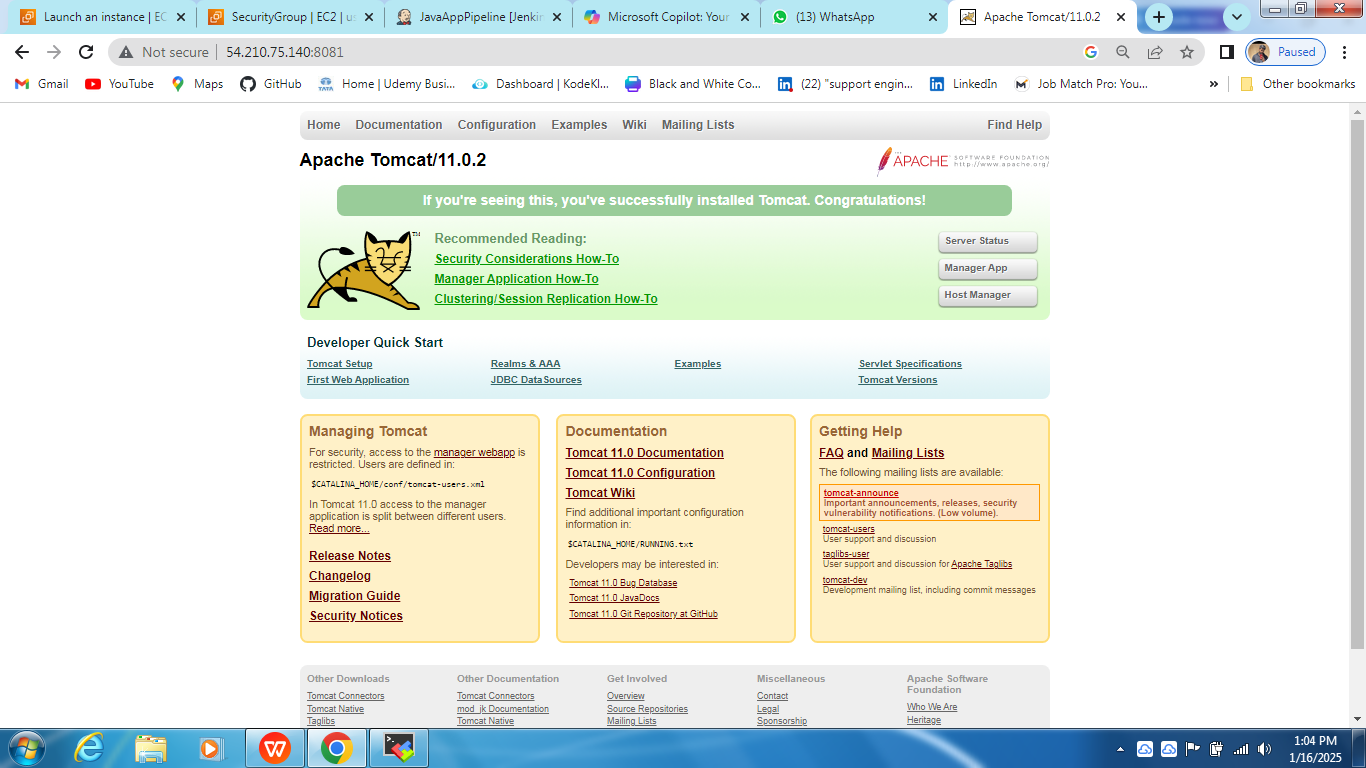


* Verified that the Docker image was created and running.



#### 6.2 ****Validation****

* Accessed the Tomcat application successfully through the specified port.



### Conclusion

This project showcased the seamless integration of Jenkins, Maven, and Docker in creating a robust CI/CD pipeline. The deployment of a sample application demonstrated the power of automation in modern DevOps practices.