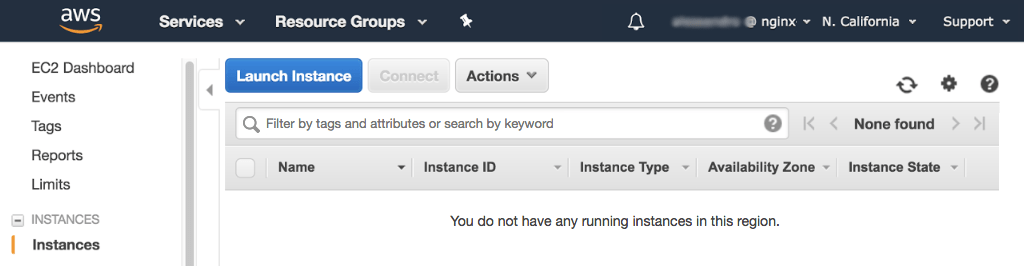
# **Installing Nginx Web Server on AWS EC2 with CI/CD pipeline on GitHub**

## **Prerequisites:**

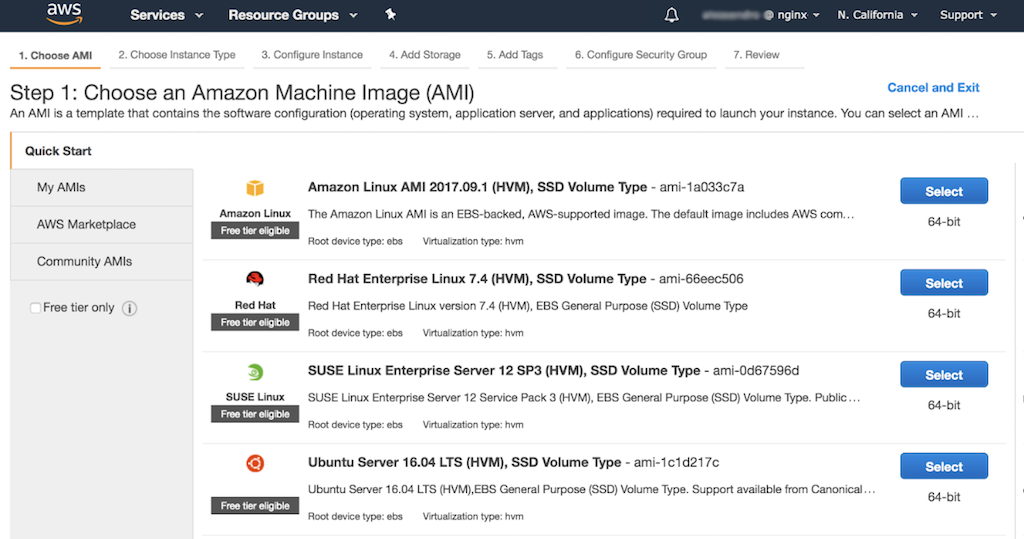
1. AWS free tier account
2. GitHub Account
3. Domain Account

**Step 1 : Creating Amazon EC2 Instance**

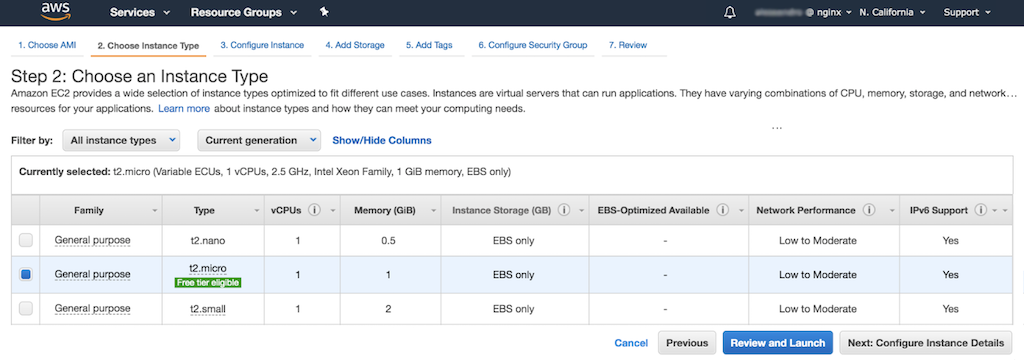
1. Creating an EC2 instance with amazon Linux image using appropriate configurations required to install Nginx Server, Below are the steps to create an Ec2 Instance.



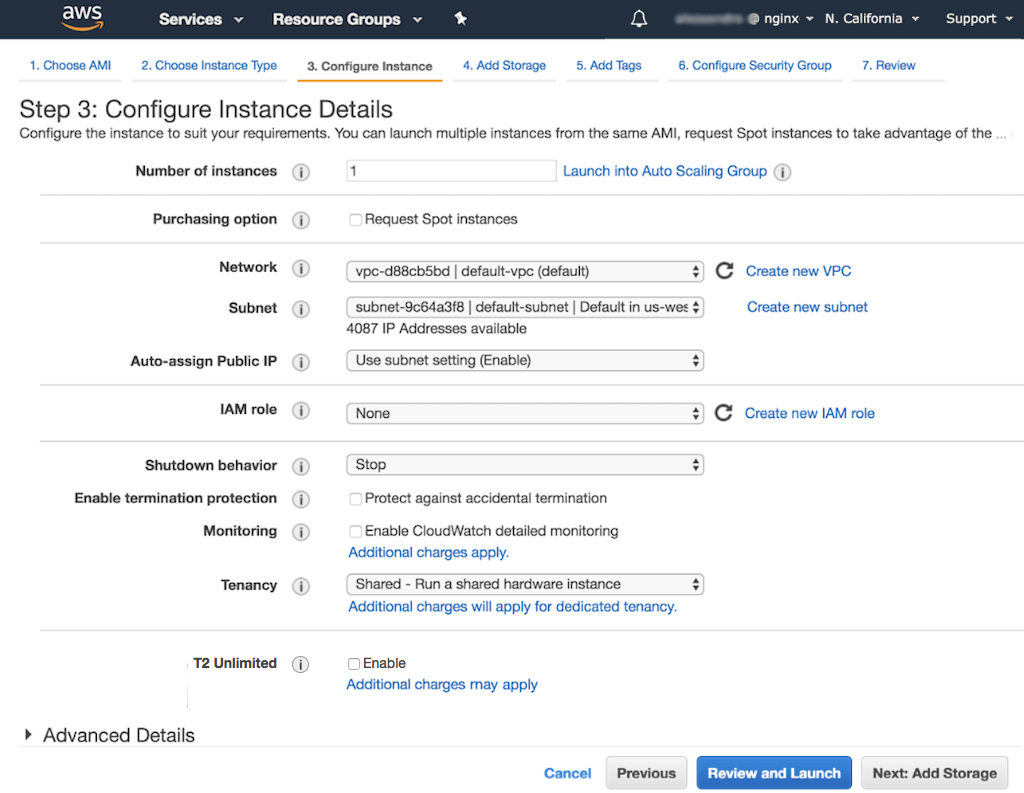
1. Choose an **Amazon Machine Image (AMI)** window, click the  Select  button for the Linux distribution of your choice.



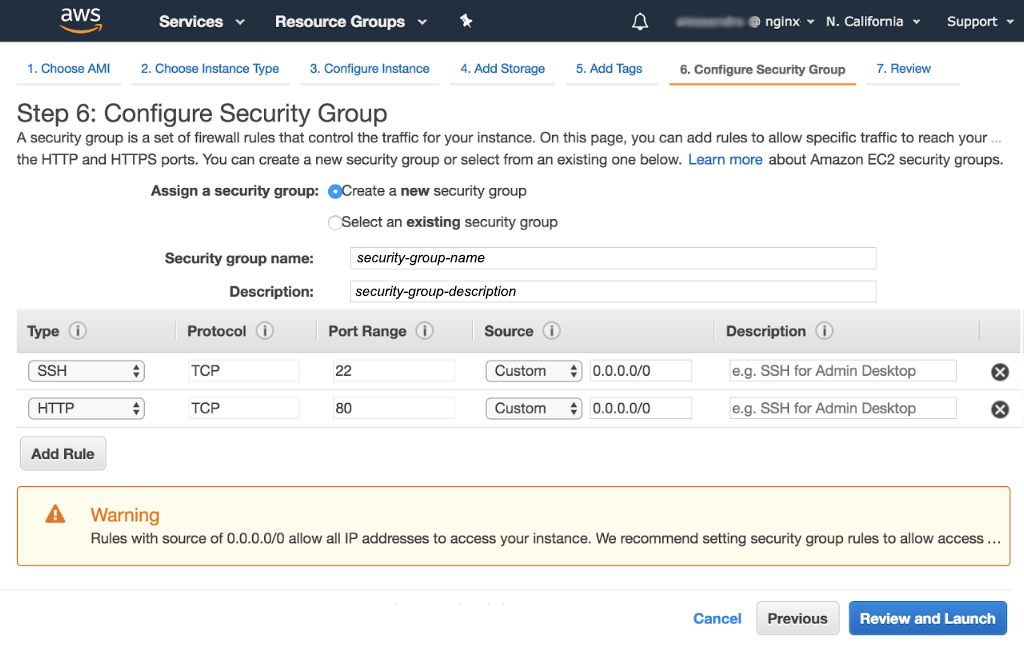
1. Choose an Instance Type.



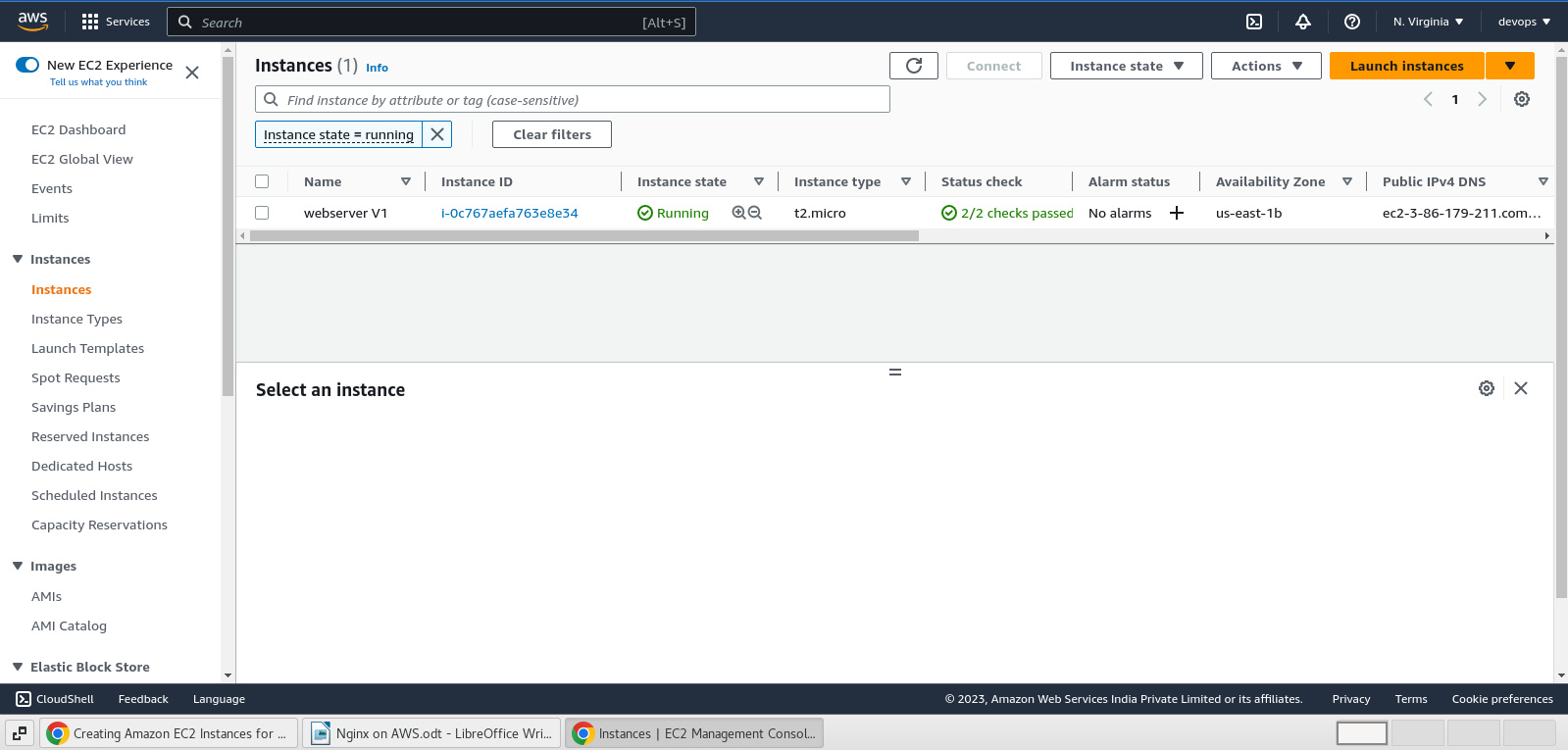
1. Configure Instance Details .



1. Configure Security Group



1. After clicking on Review and Launch navigate to instances window to see the running instance.



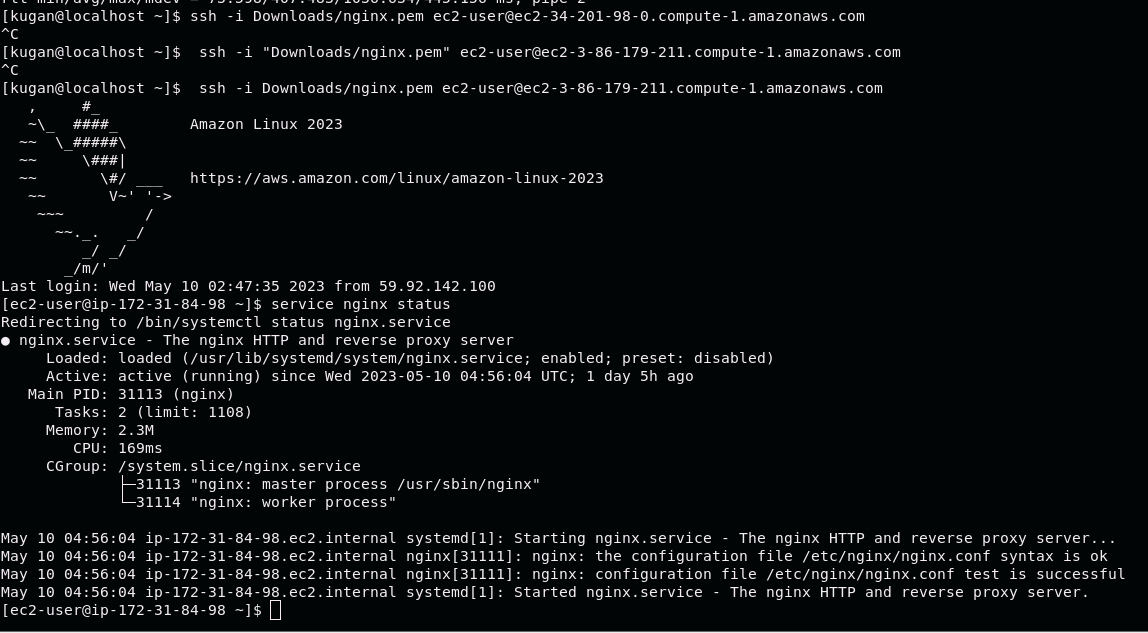
**STEP 2 : Installing Nginx Server on EC2 instance**

1. Log in into the ec2 instance launched using ssh and perform the below mentioned commands to install the nginx server.

SSH to AWS Instance Command : **ssh -i "nginx.pem" ec2-user@ec2-3-86-179-211.compute-1.amazonaws.com**

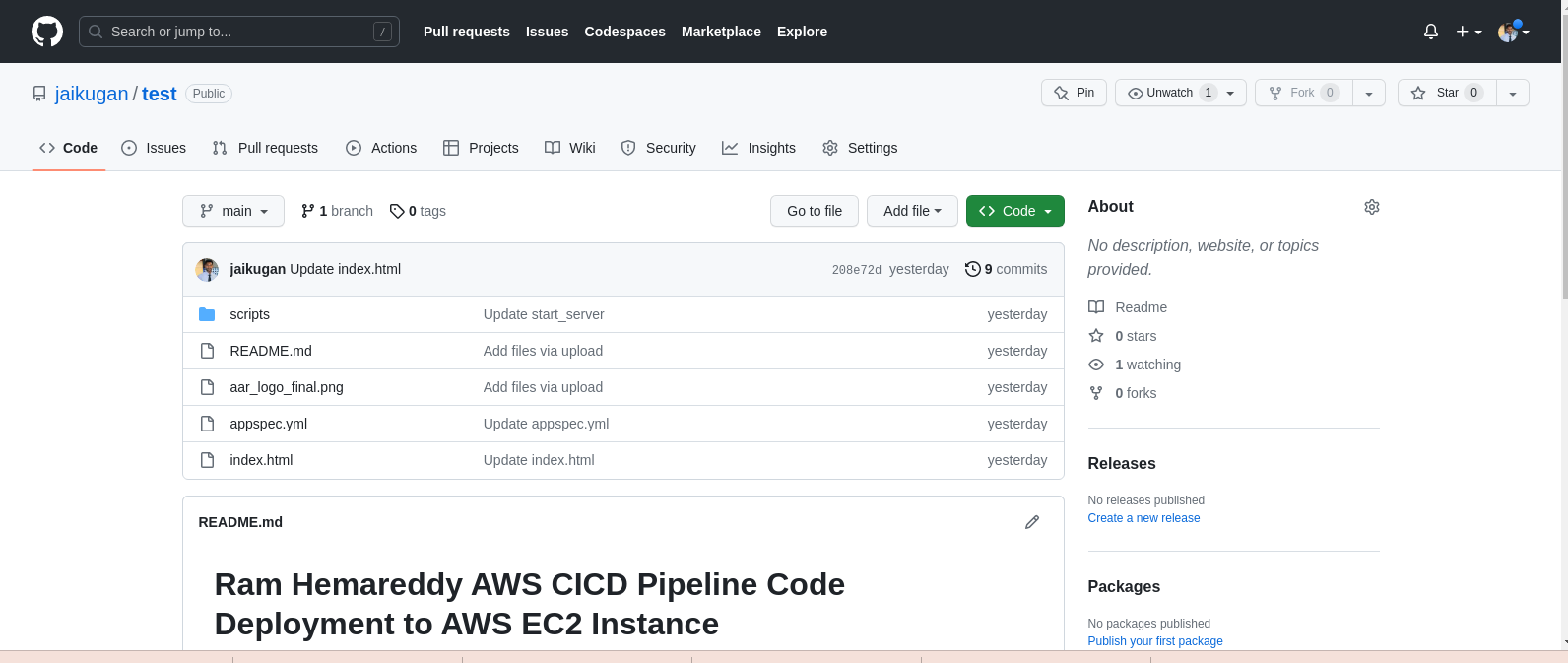
Commands:

* $ yum update -y
* $ yum install nginx -y
* $ service nginx start
* $ systemctl enable nginx
* $ service nginx status

****

**STEP 3 :** **Automating** **the** **code** **commits** **in** **GitHub** **to Nginx** **server using CI/CD Pipeline**

1. Create a GitHub account and then create a repository to store all your codes into the repo.

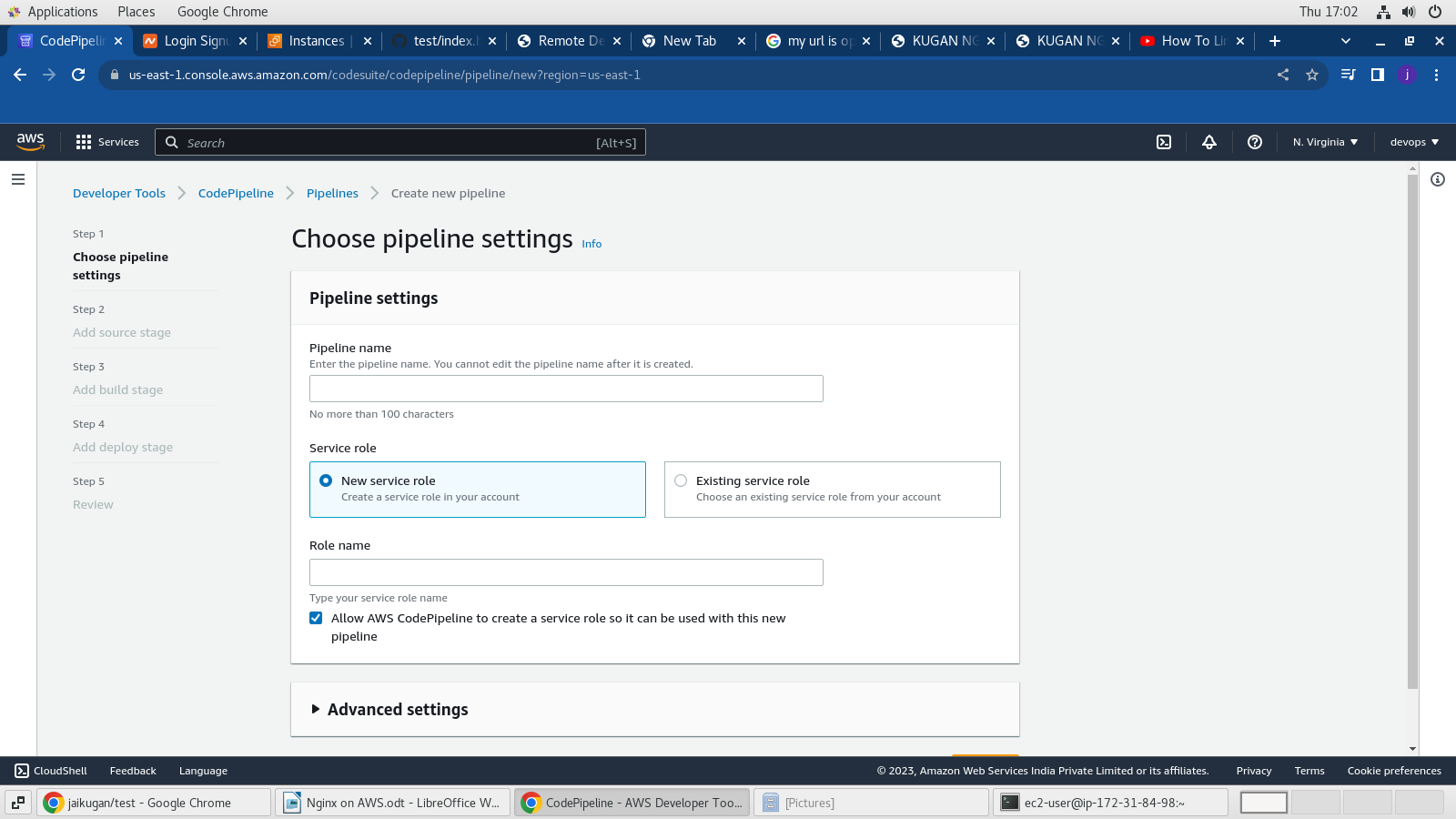


1. We will be using AWS Code Pipeline to pull the code from GitHub and to deploy it in Nginx server hosted in EC2 Instance.

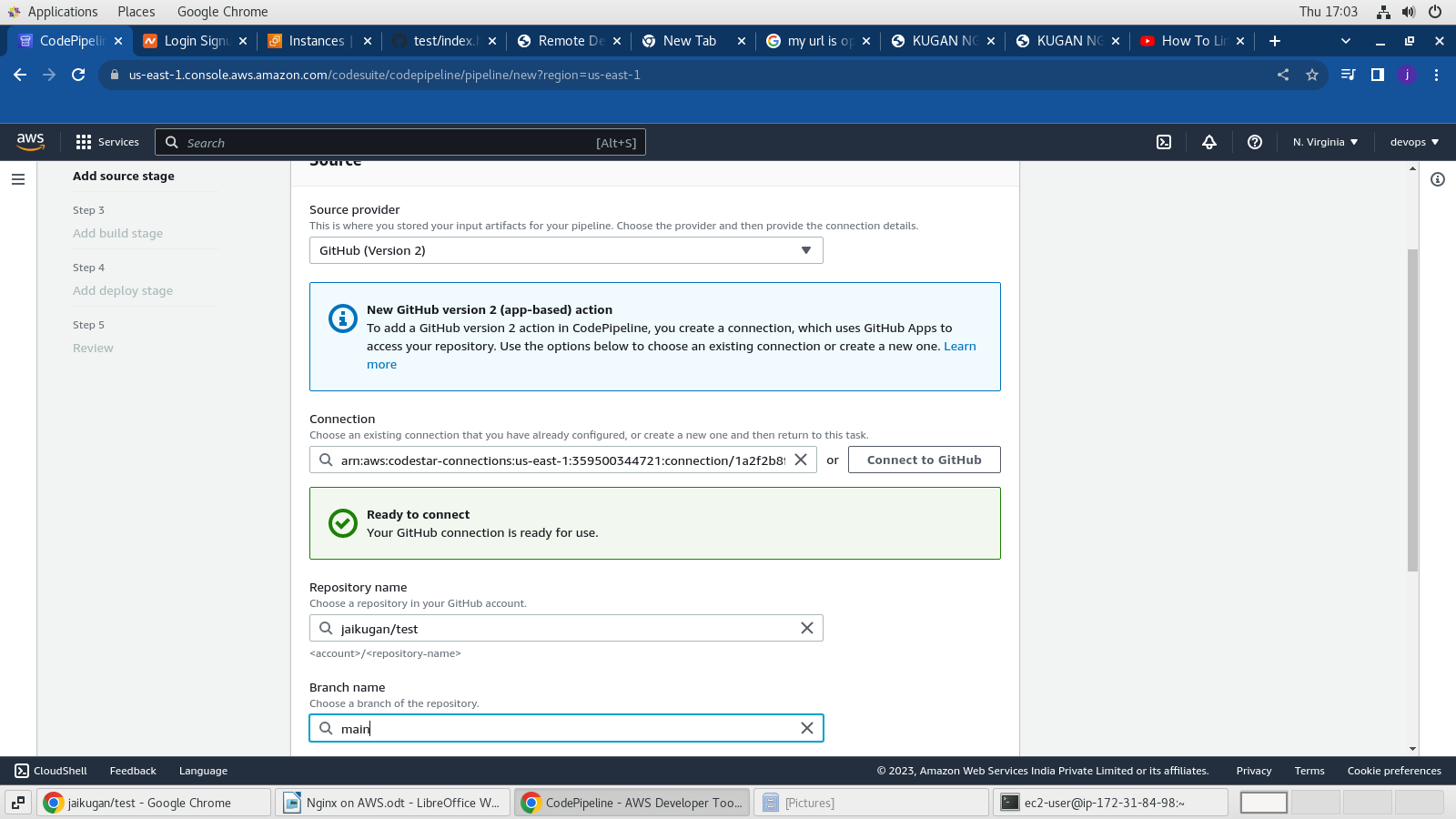
There will be four stages of process while creating the Pipeline

* **Pipeline settings**
* **Source action provider**
* **Build action provider**
* **Deploy action provider**

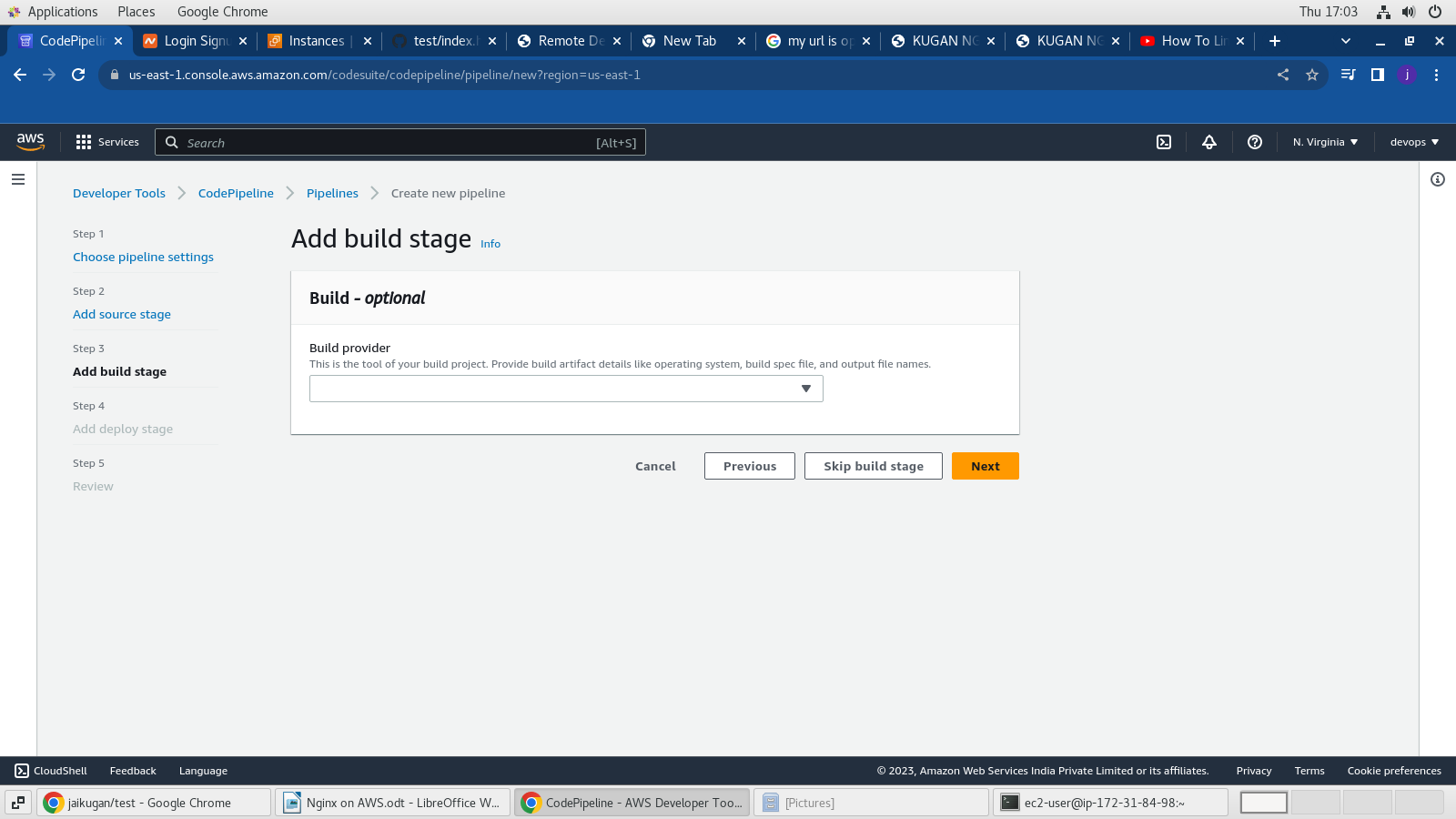
1. Pipeline Settings is the place where we add the Name and service role created for an EC2 instance



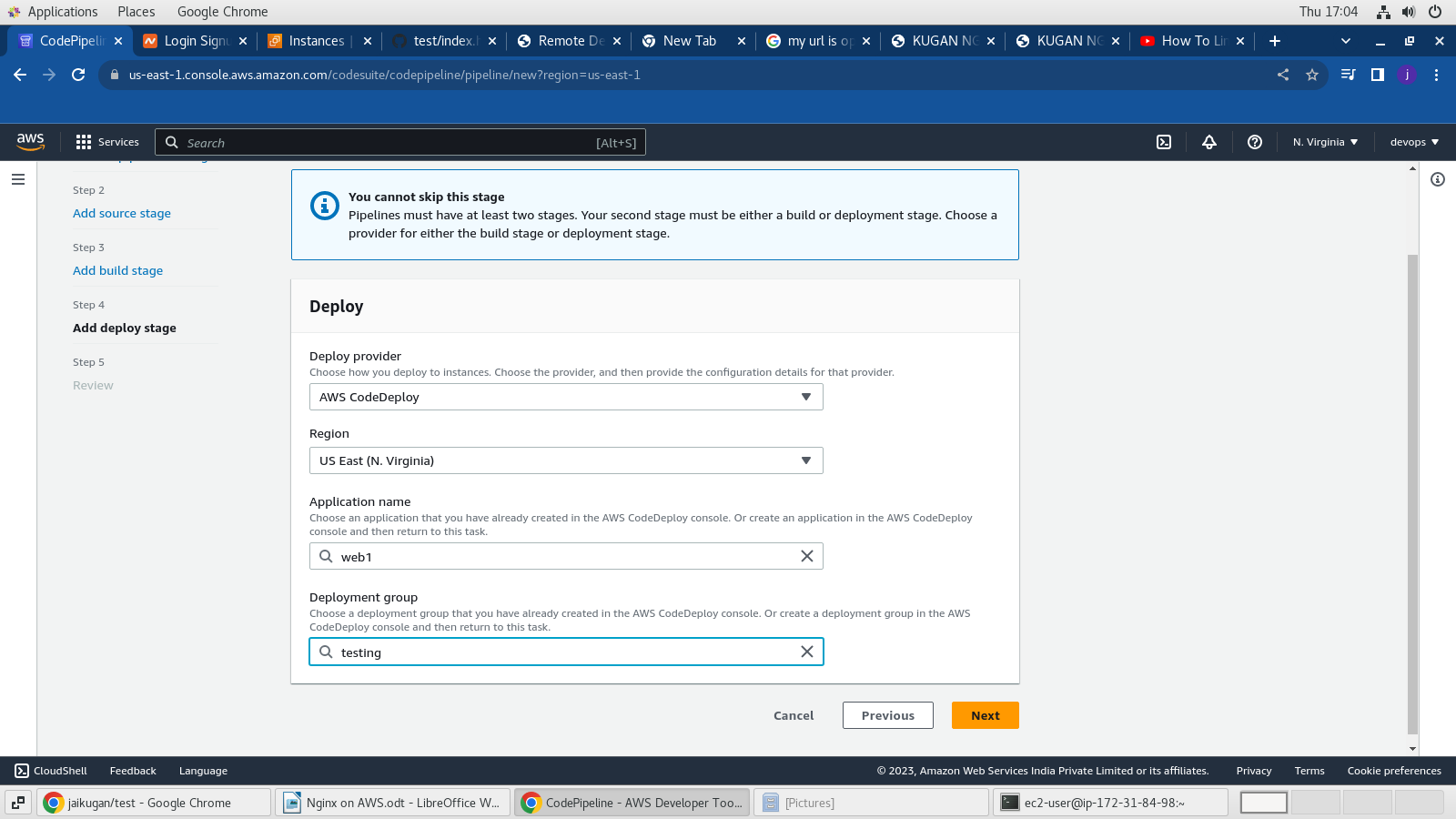
1. Choose the source provider GitHub and get connected to the GitHub account by providing the Credentials post that choose the repository along with branch.



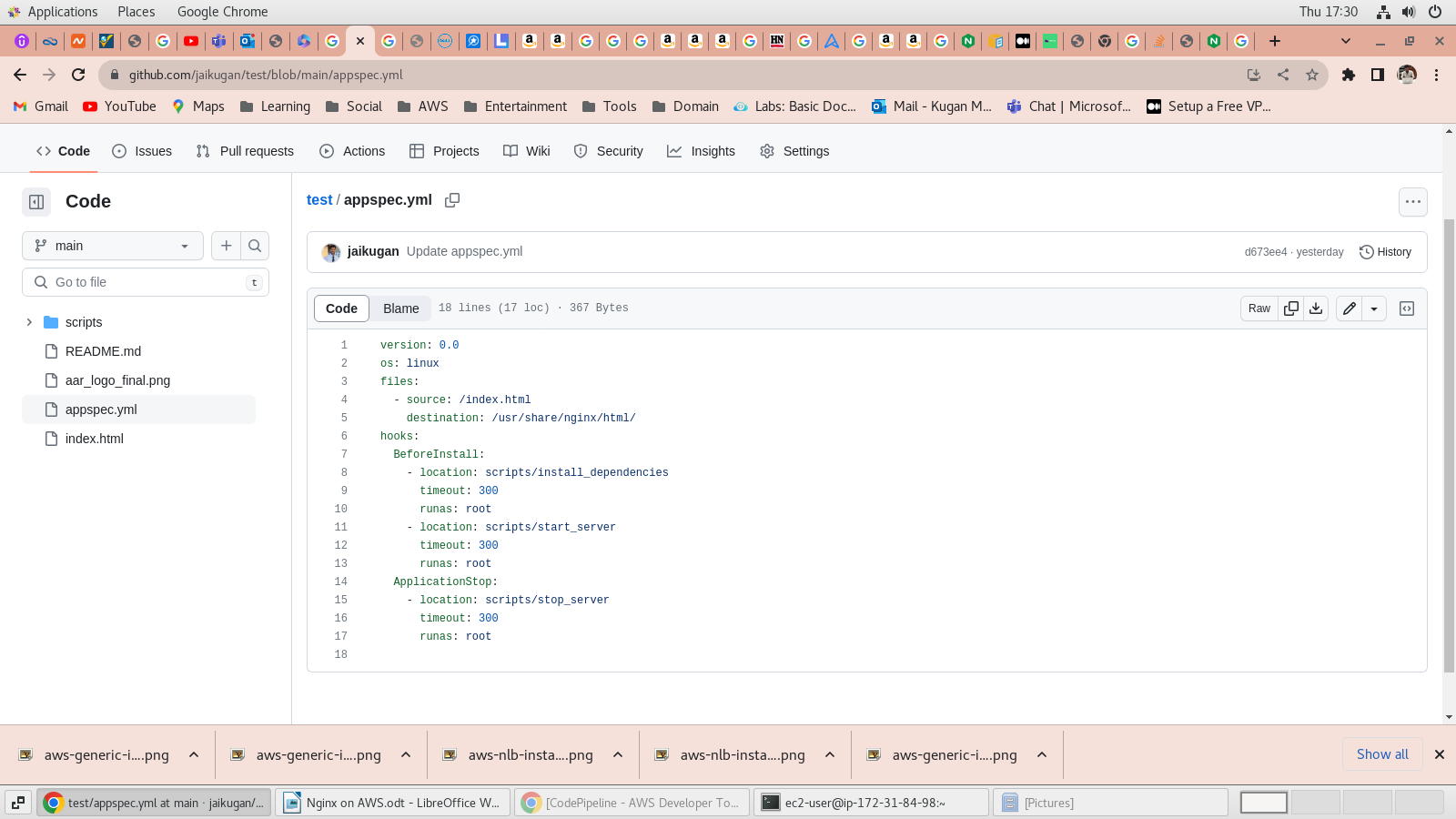
1. We can skip the build part since we are not performing any build action.



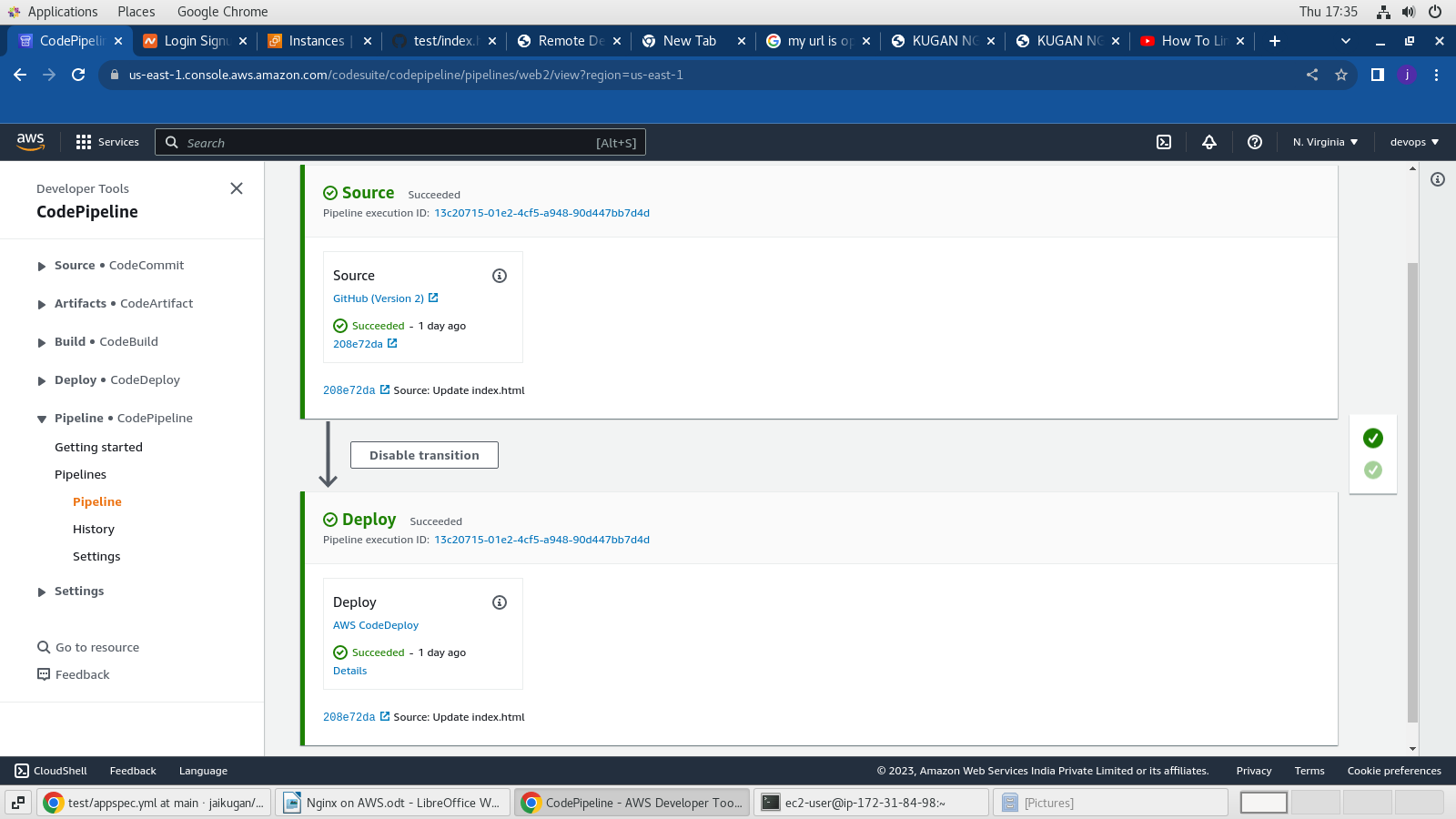
1. Final step is to provide the details of App and deployment group name along with service role for AWS CodeDeploy.



1. On the next step review the pipeline configurations and click on create.
2. Once the pipeline is created we need to write a script and place it in GitHub source repository which will do the actions specified on that script file.
3. The script file should me named with standard format as “appspec.yml” which contains the actions to be done when a new code is committed in GitHub.

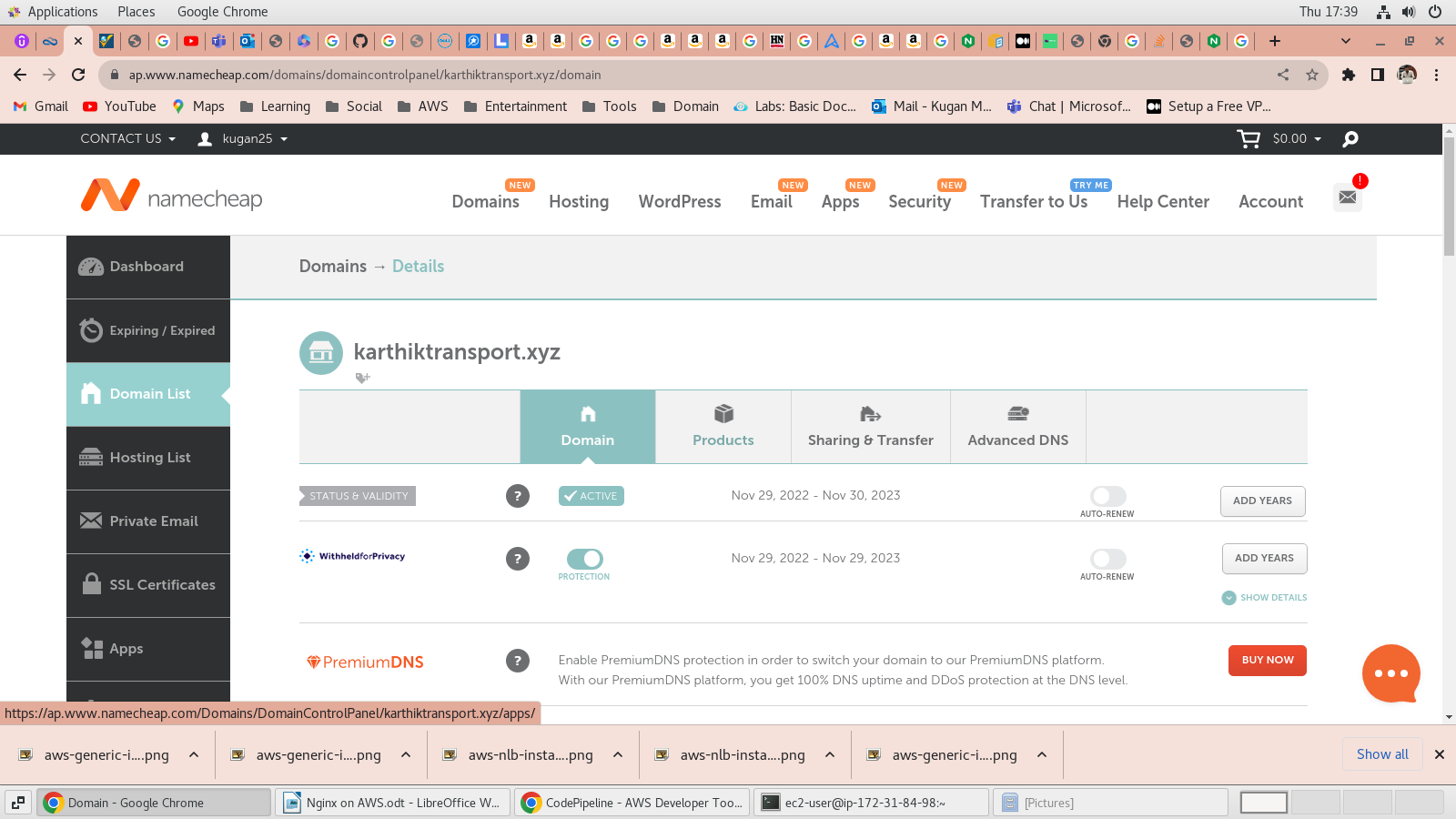


1. Once all the setup is done open the pipeline to see the latest changes deployed on the nginx server.

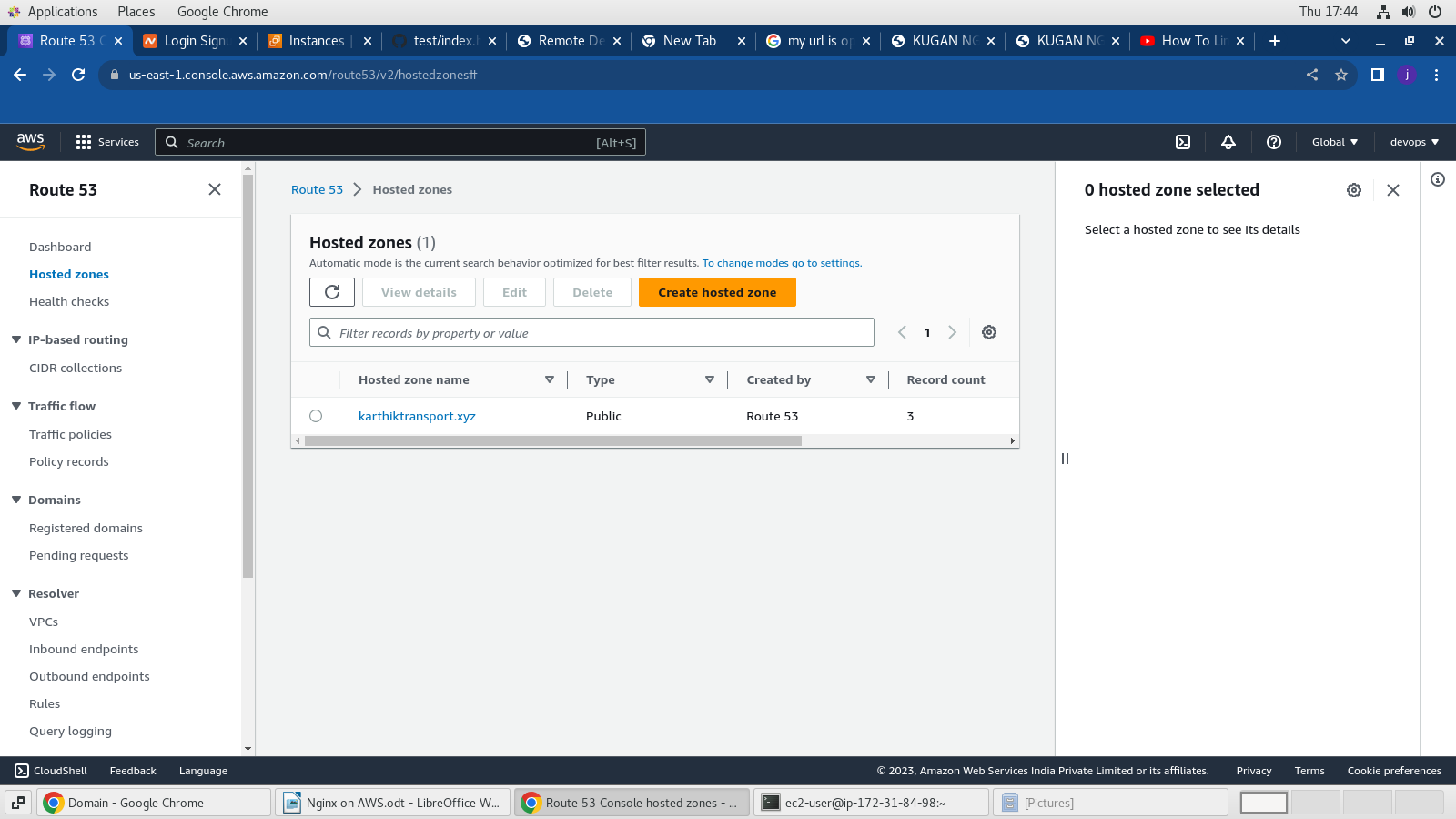


**STEP 4 : Mapping our domain name to the server IP Address using Route 53**

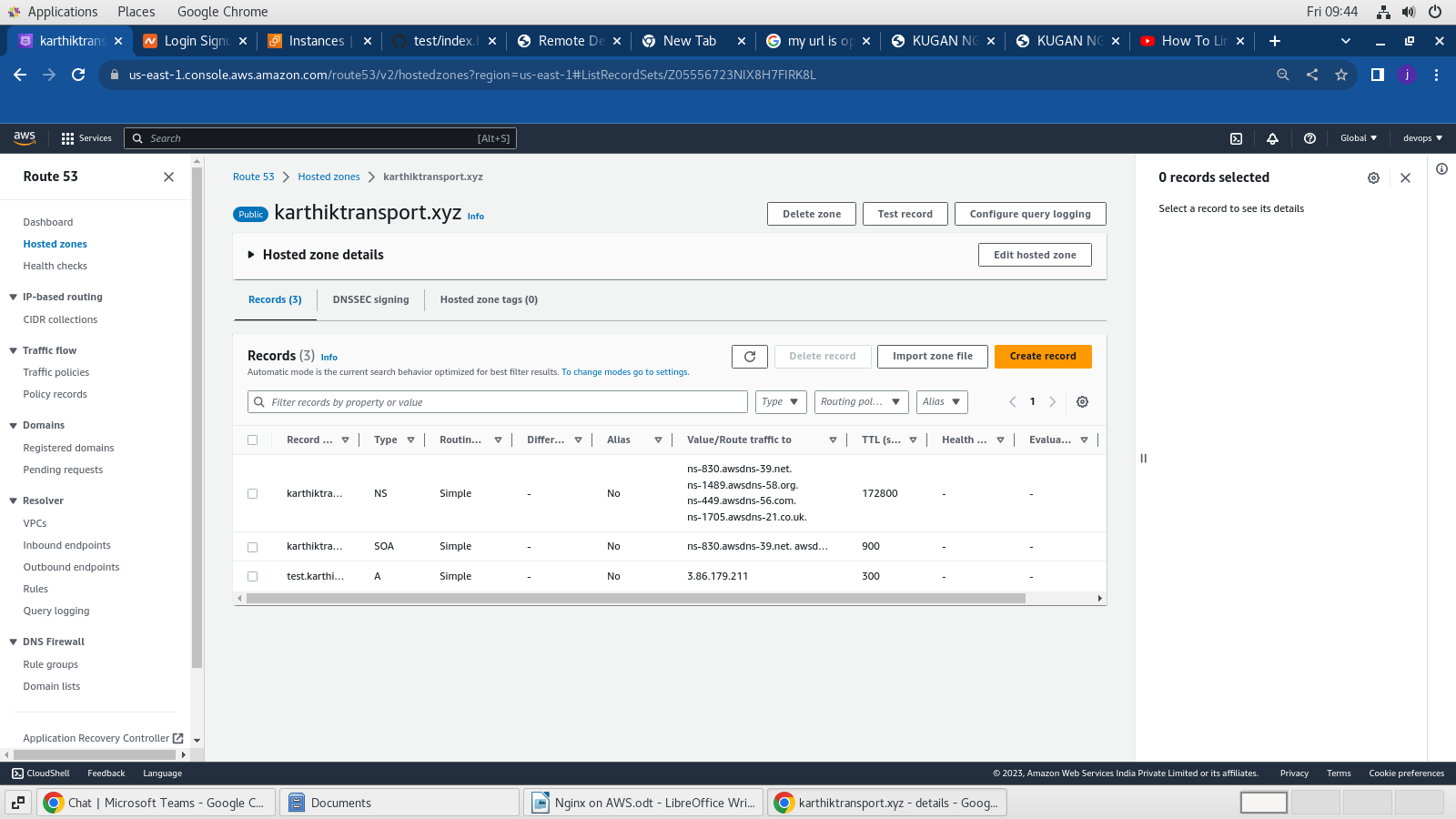
1. Purchase a domain name from any domain register for eg: godaddy,namecheap etc.

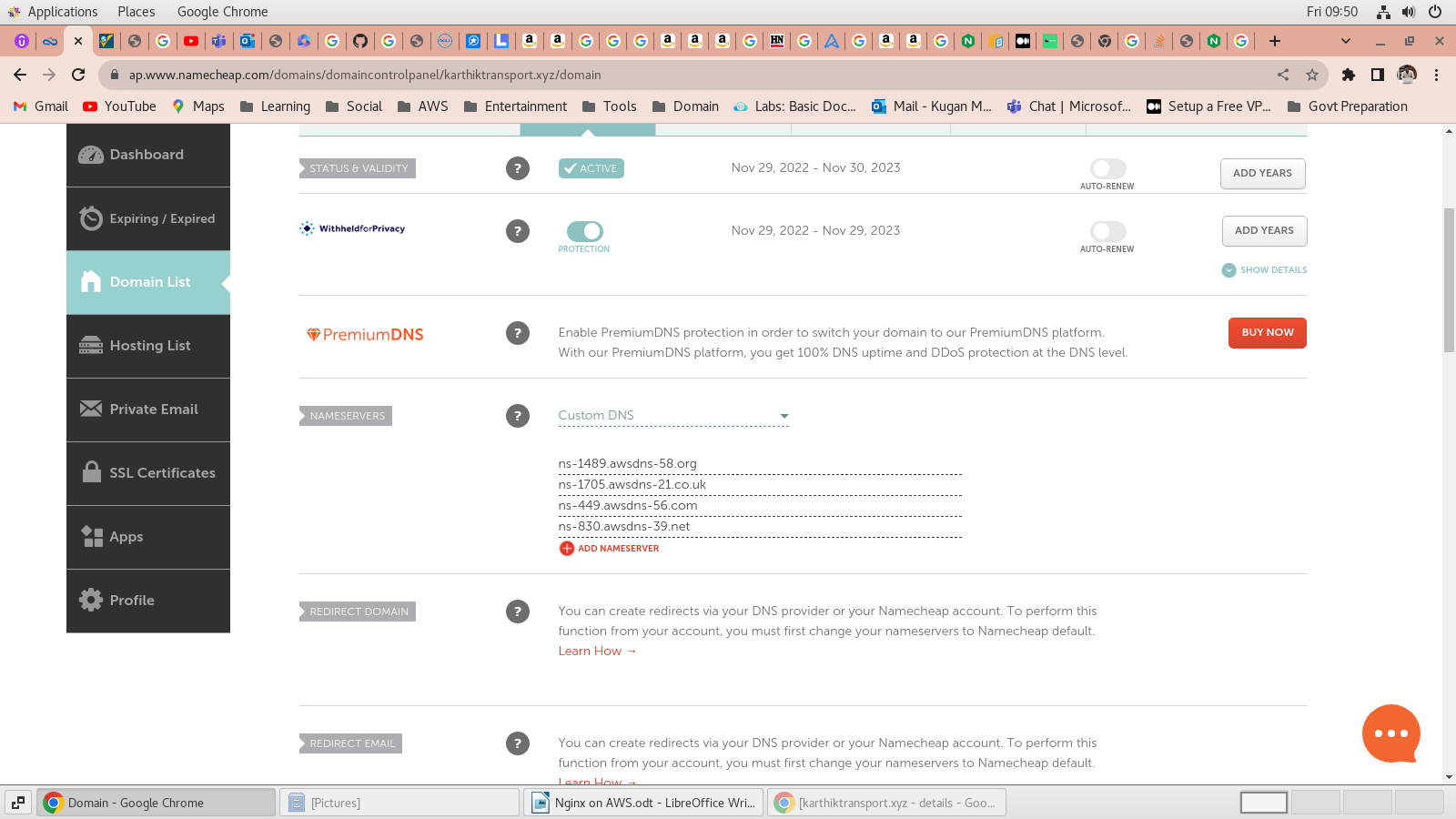


1. Now navigate to the route53 AWS service to create a hosted zone for the purchased domain name.



1. Create a new A type record to map IP address to our Domain name “karthiktransport.xyz”. We have mapped with following name “test.karthiktransport.xyz”



1. Now Copy the nameservers from the Hosted zones on route53 and paste it in the Domain registered Custom DNS.
2. Now we can access the web page with the following URL : [**http://test.karthiktransport.xyz**](http://test.karthiktransport.xyz/)

