**DEVOPSTHON**

**SUBMISSION**

**Application is successfully deployed to AWS EC2**

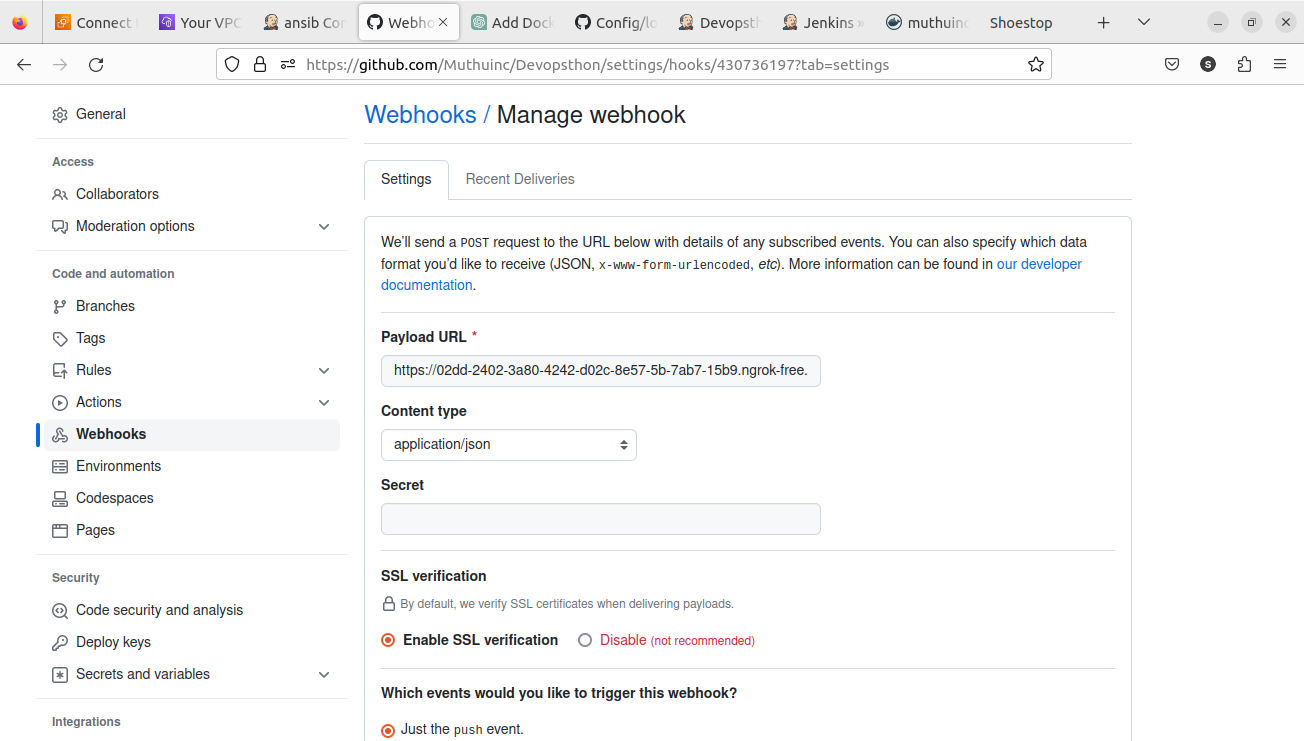
**no manual deployment is there except autoscaling and loadbalancer**

**Devops batch D09**

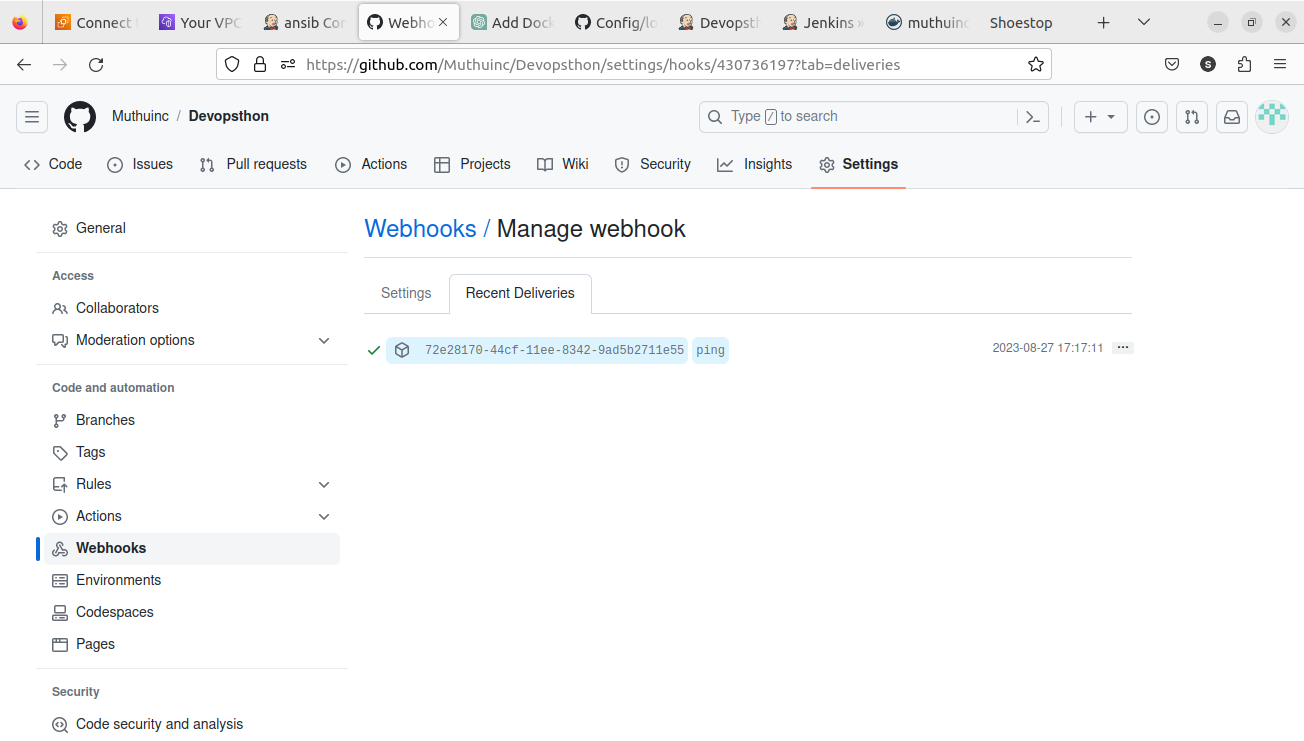
**Muthuramalingam**

**Github link : github.com/Muthuinc/Devopsthon**

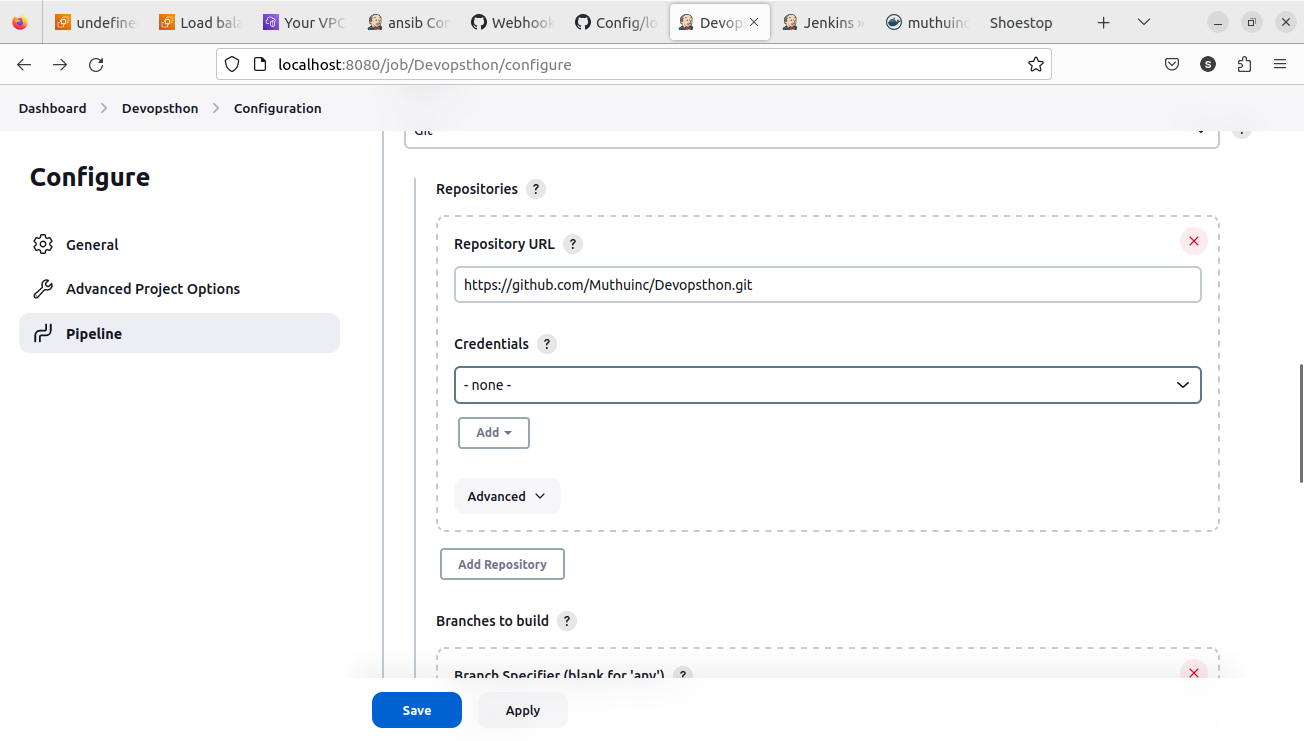
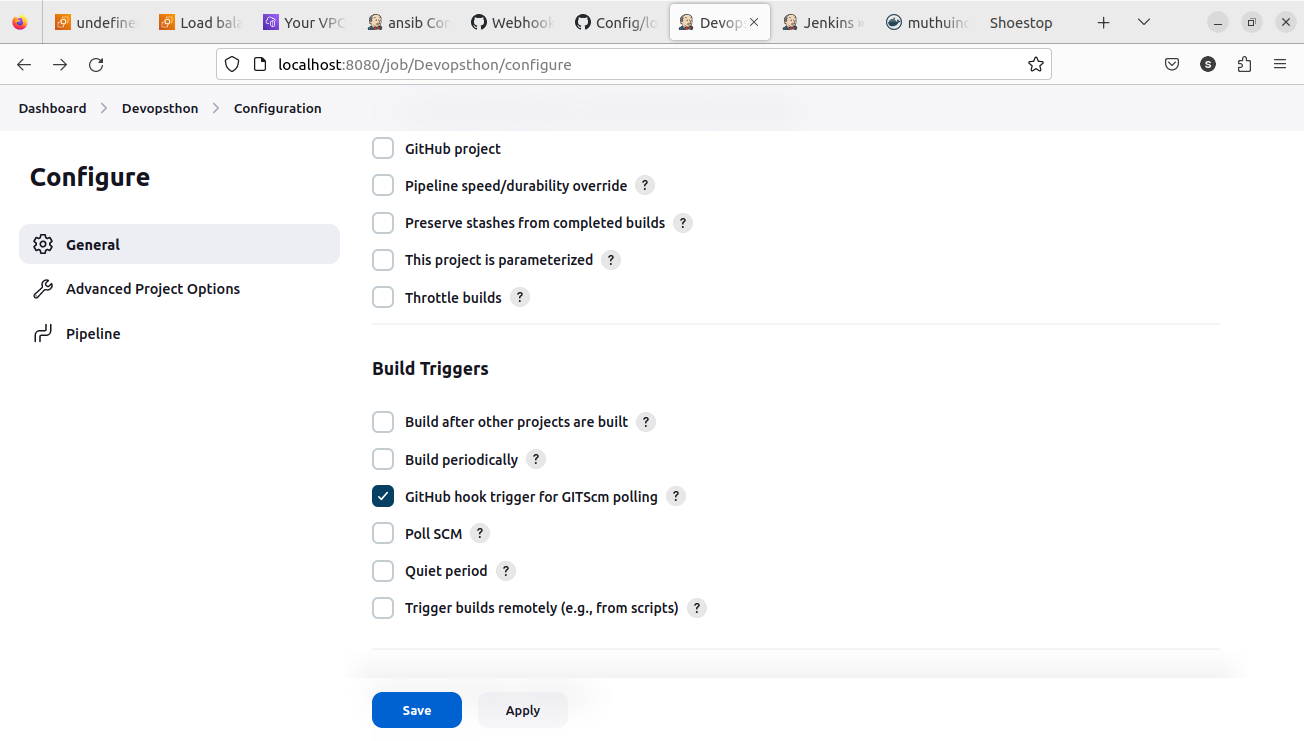
**Firstly application code has been downloaded. And copied the application files. In my folder.**

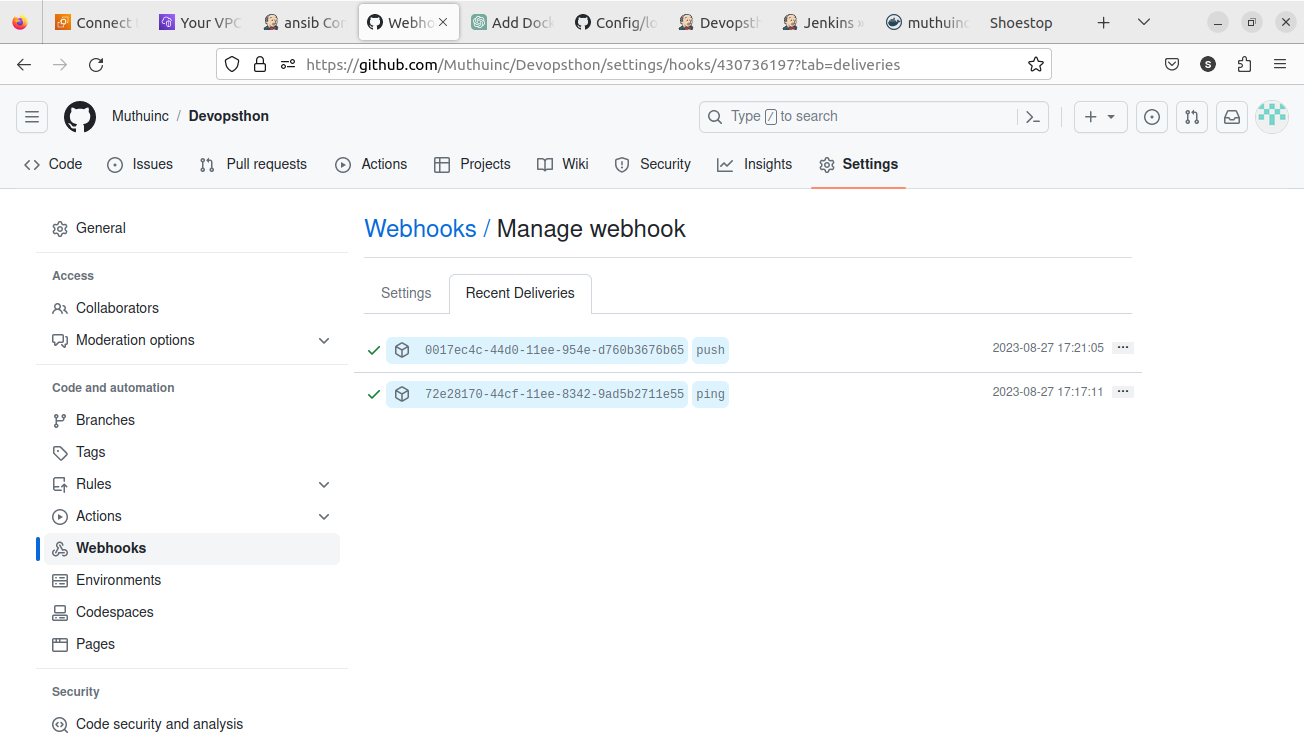


**Git hub webhook setup**



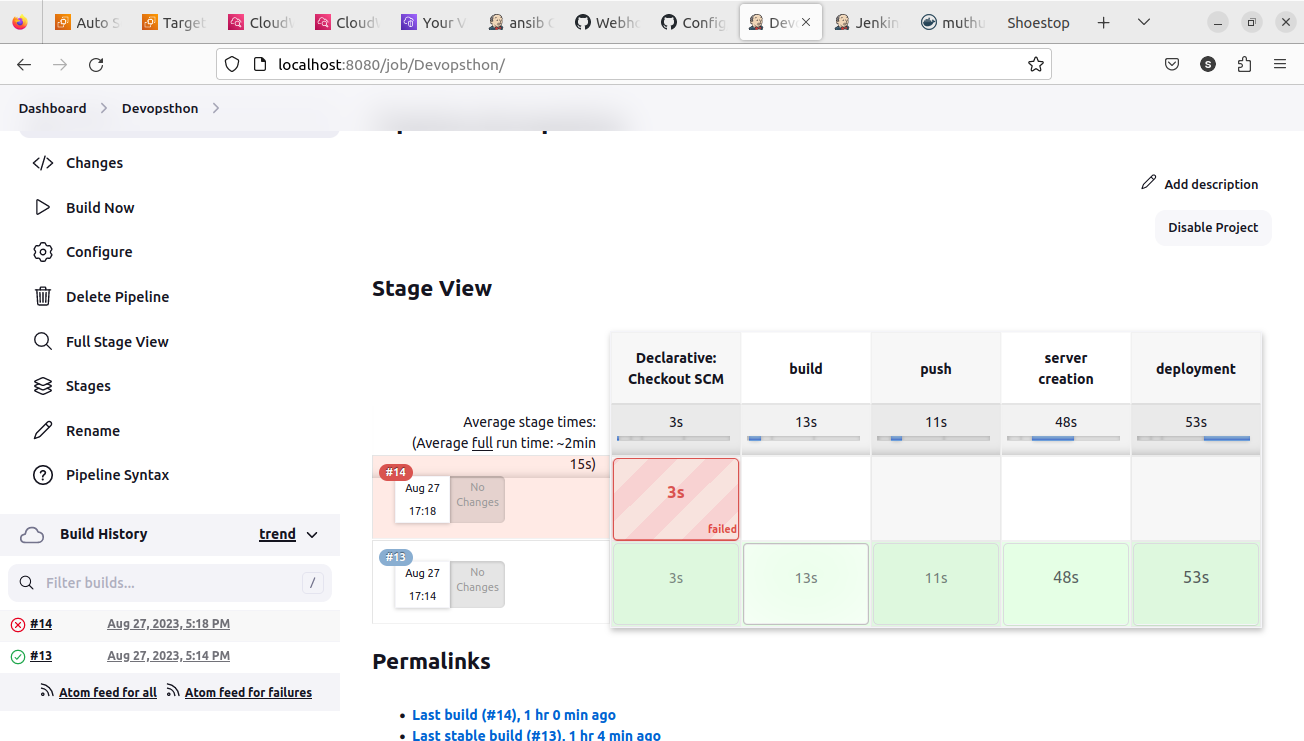
**Jenkins configuration – I used Jenkins on my local**





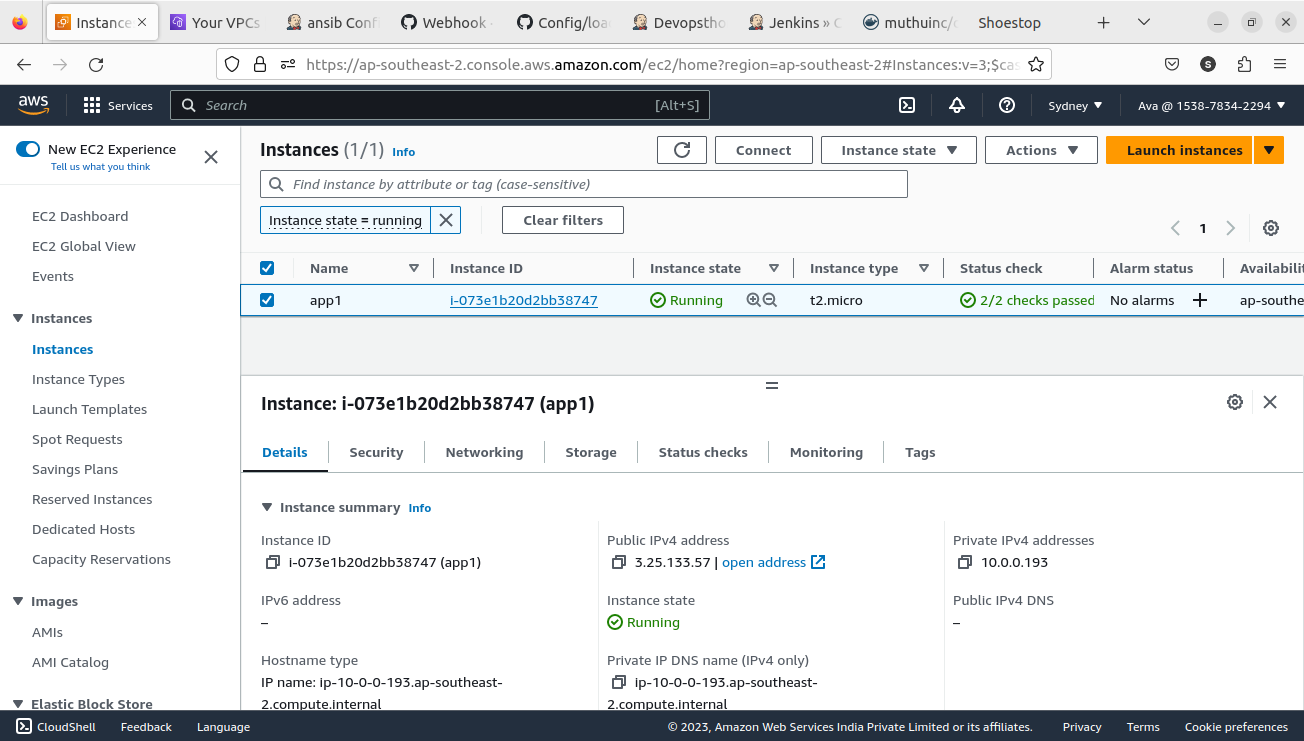
**Github webhook sent trigger based on the push event**

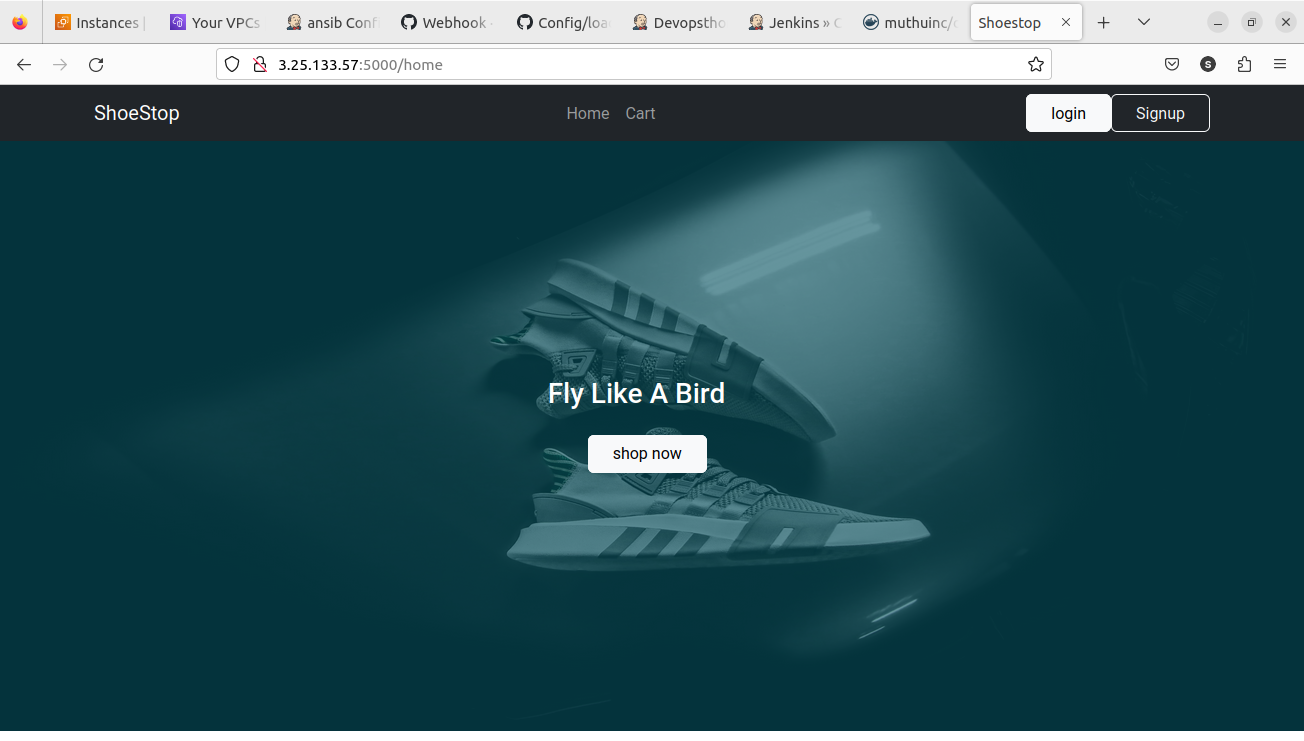
**Jenkins built the jobs based on the webhook.**

****

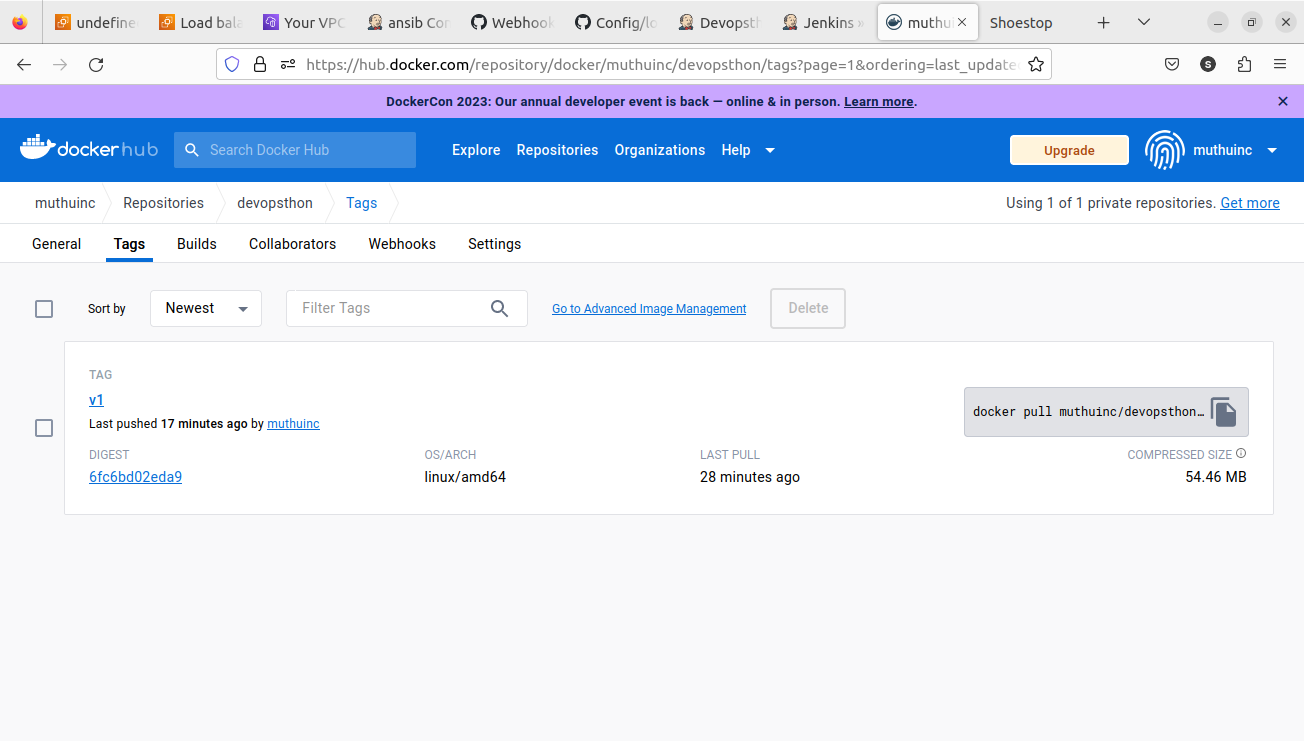
**I was trying to take everything even the autoscaling by terraform, but I couldn’t. So ignore the config files for the autoscaling in the github.**

**Ec2 instance created based on our config – refer main.tf file in terraform folder**



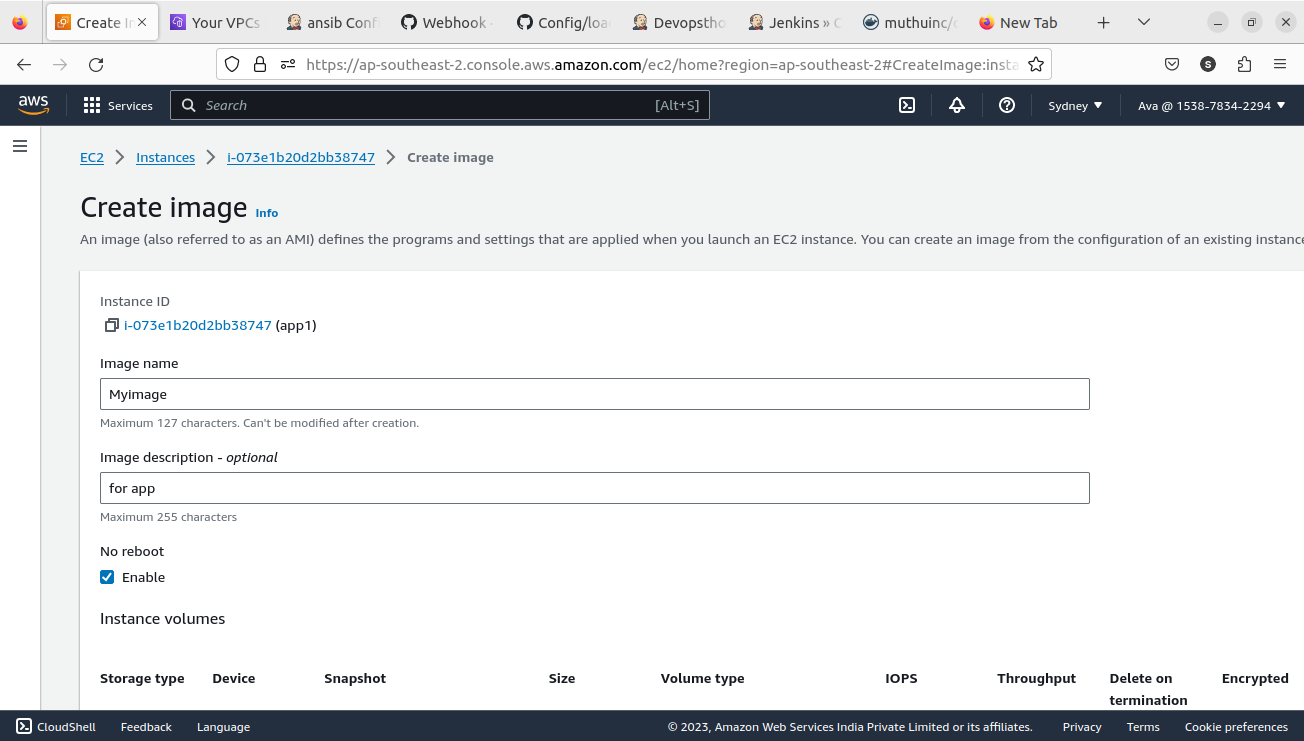


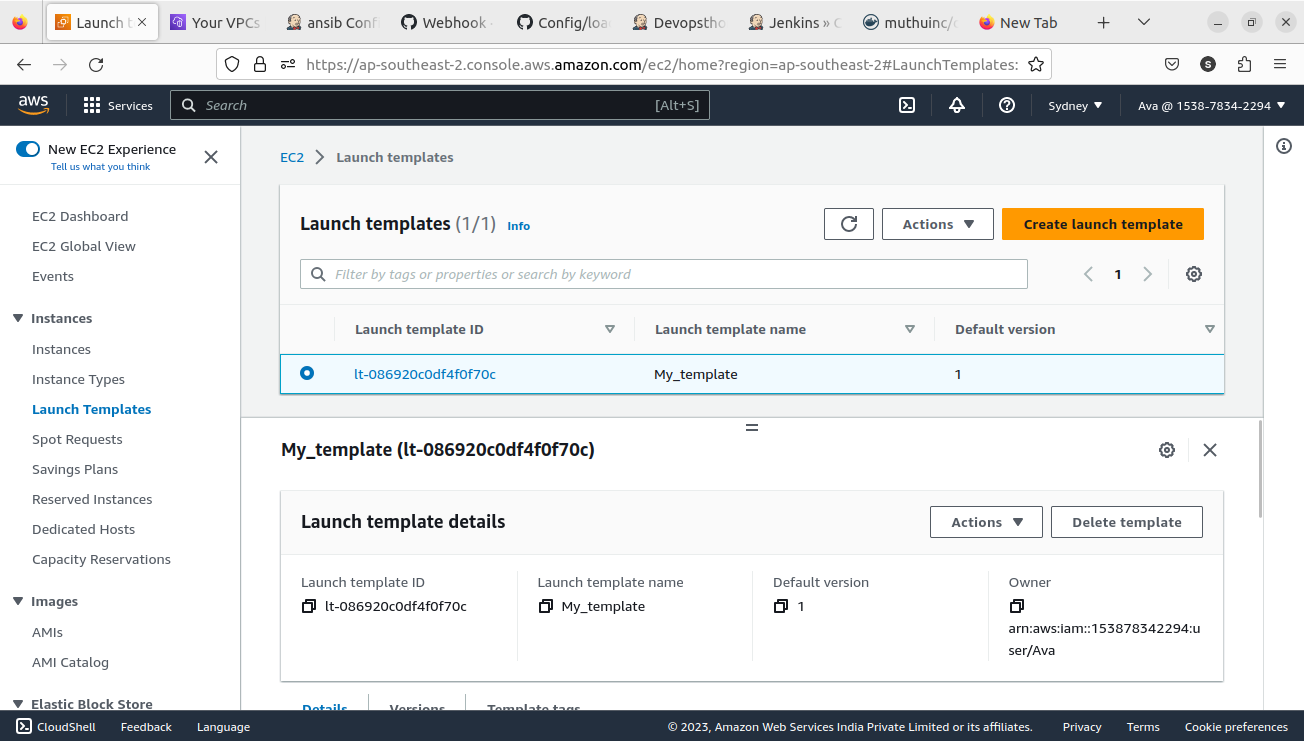
**Application deployed successfully**

**Docker hub repo image**

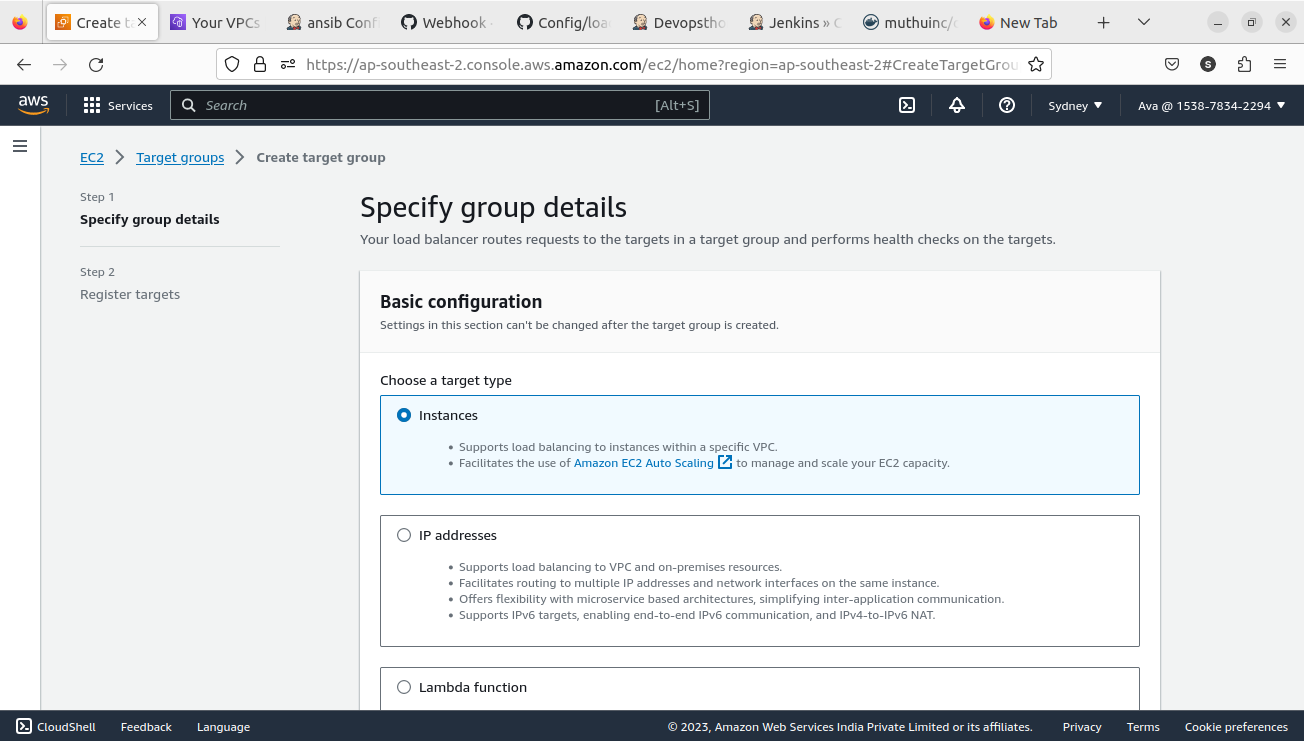
**For auto scaling the below steps done manually – before everything was done automatically by pipeline.**

**Creating AMI from our instance**

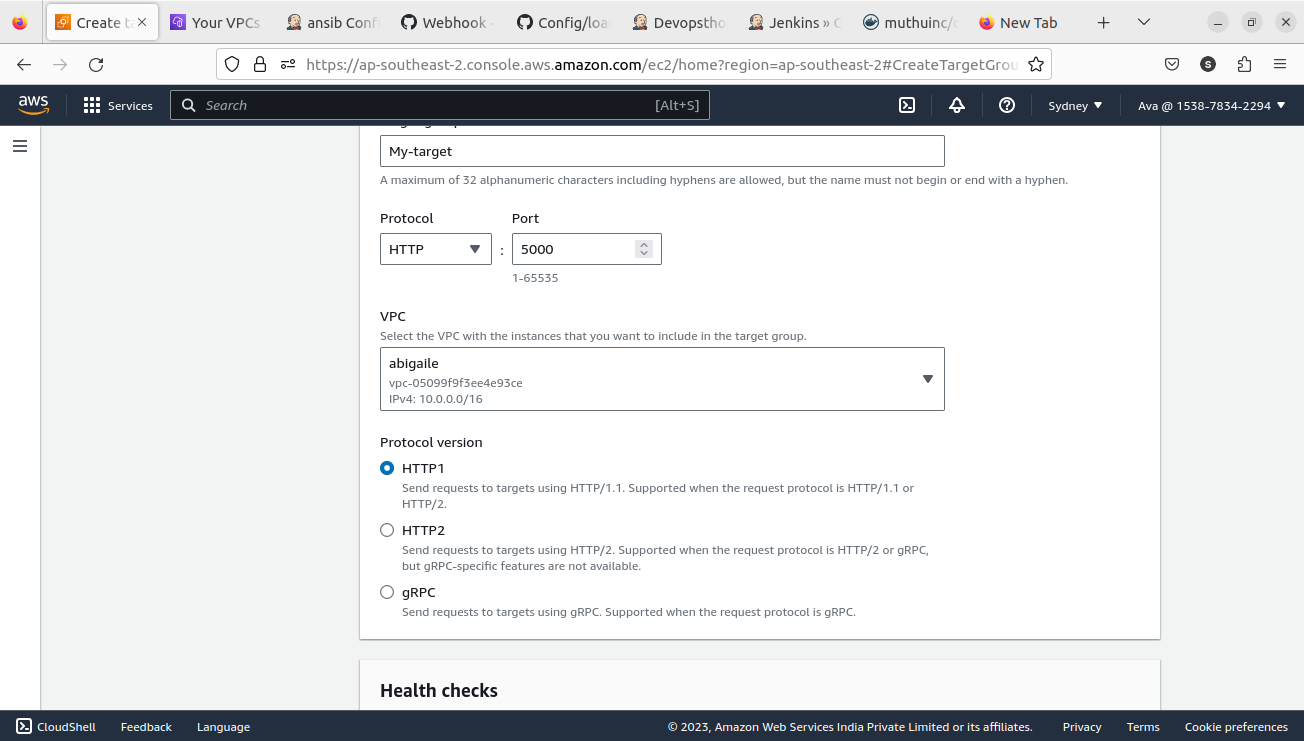


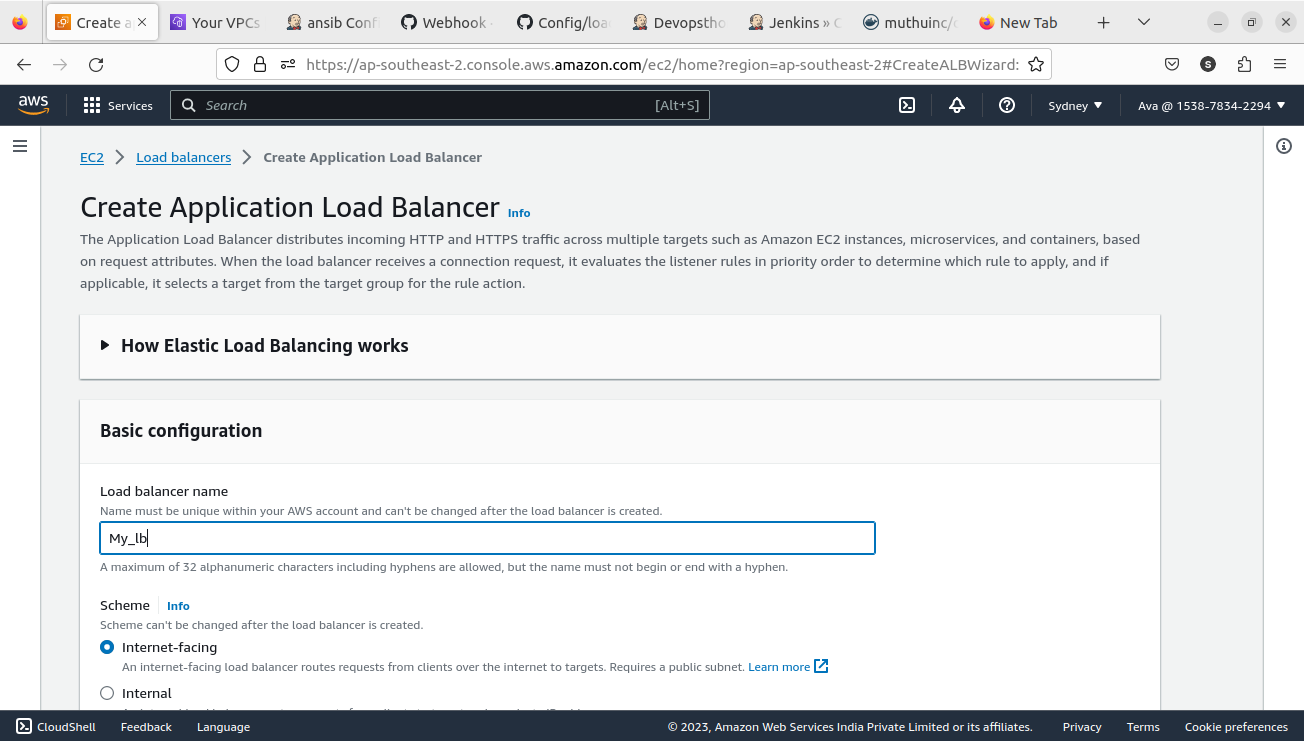
**Launch template made ready from the AMI for autoscaling** 

**Target group for loadbalancer**



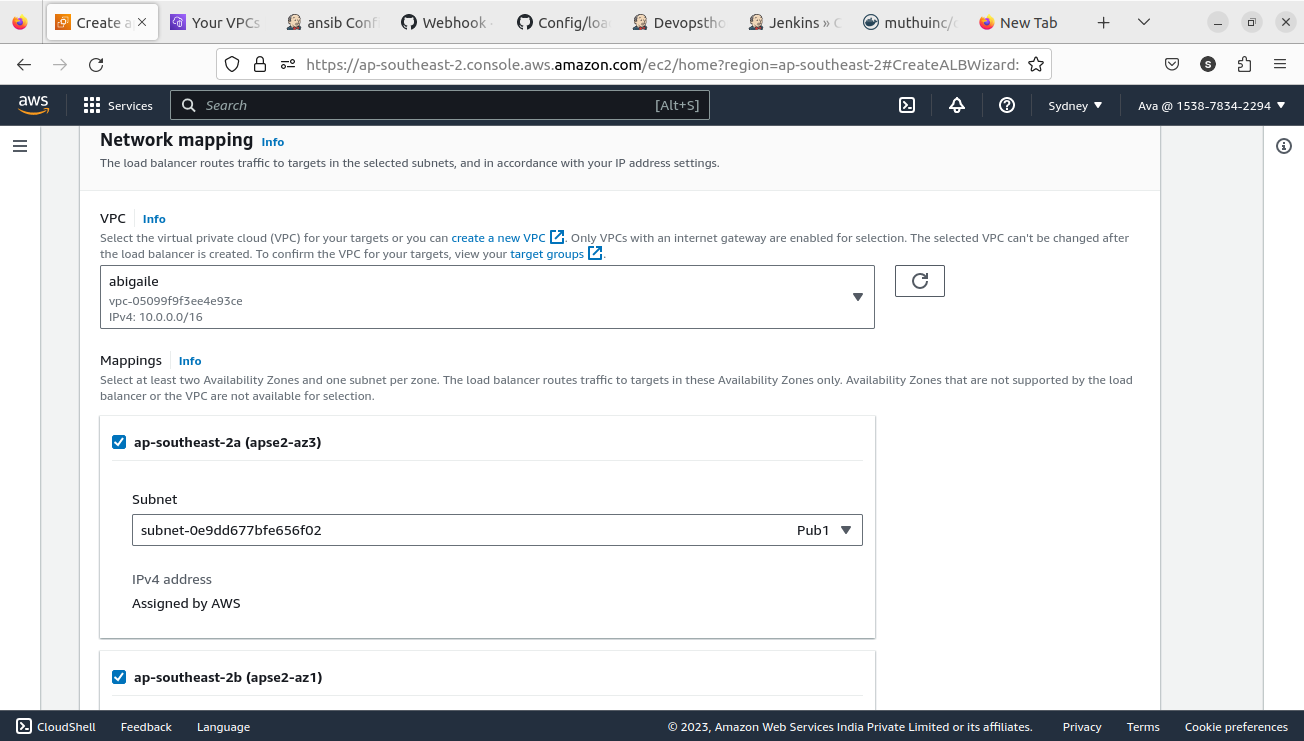
**It is configured for port 5000**



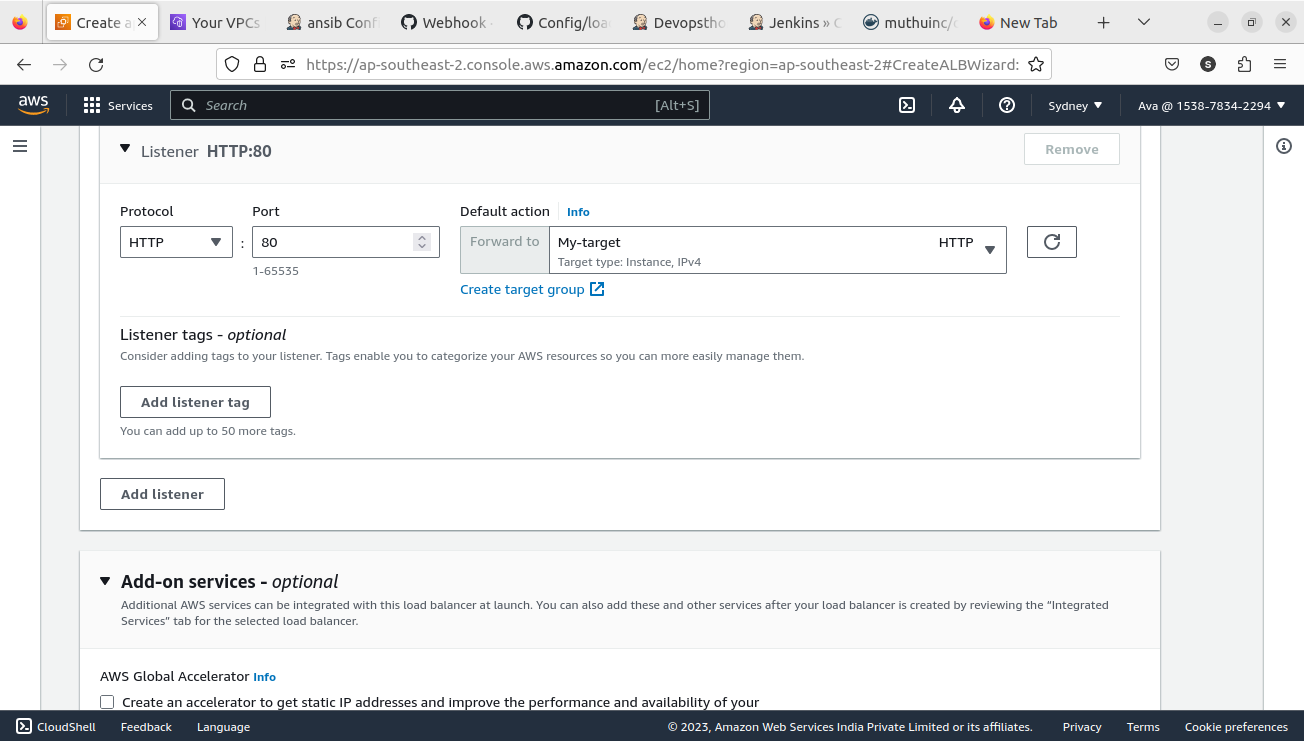
**Creating load balancer for the application** 

This is the vpc our application is running and it is created by the Jenkins pipeline

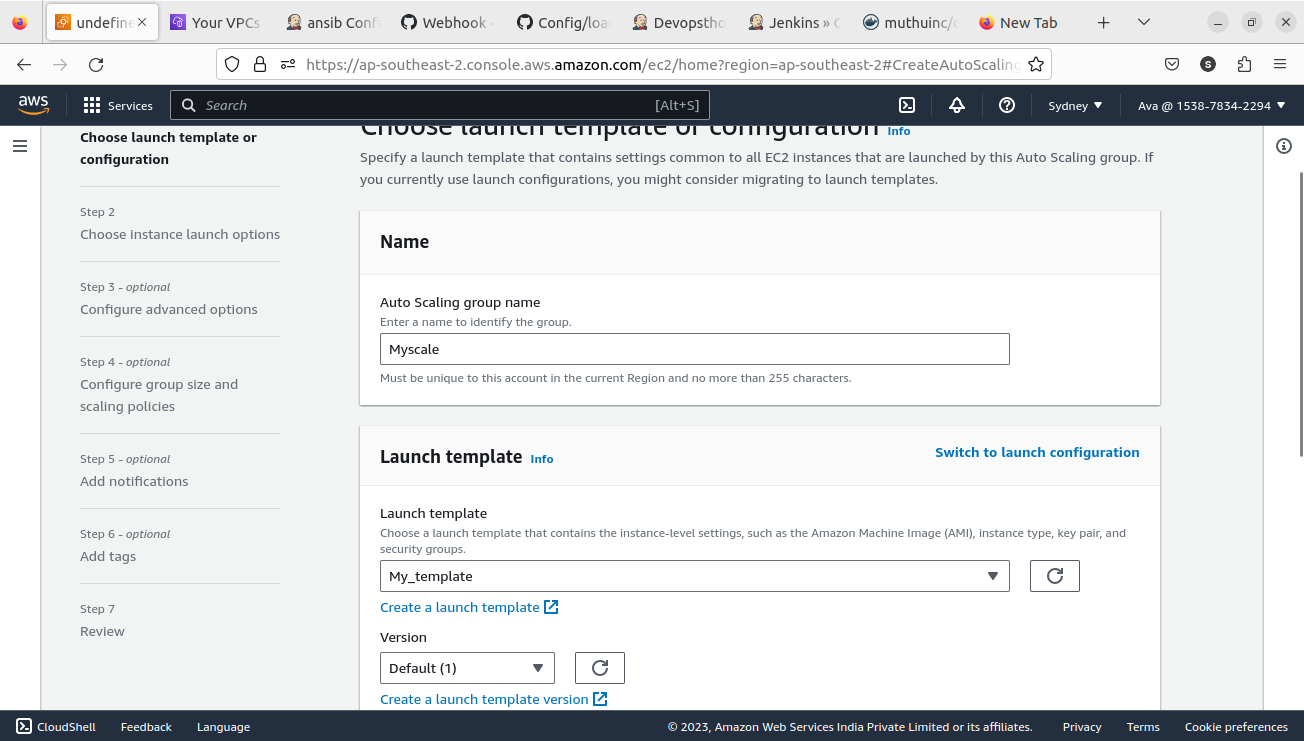
Our loadbalancer also runs here.



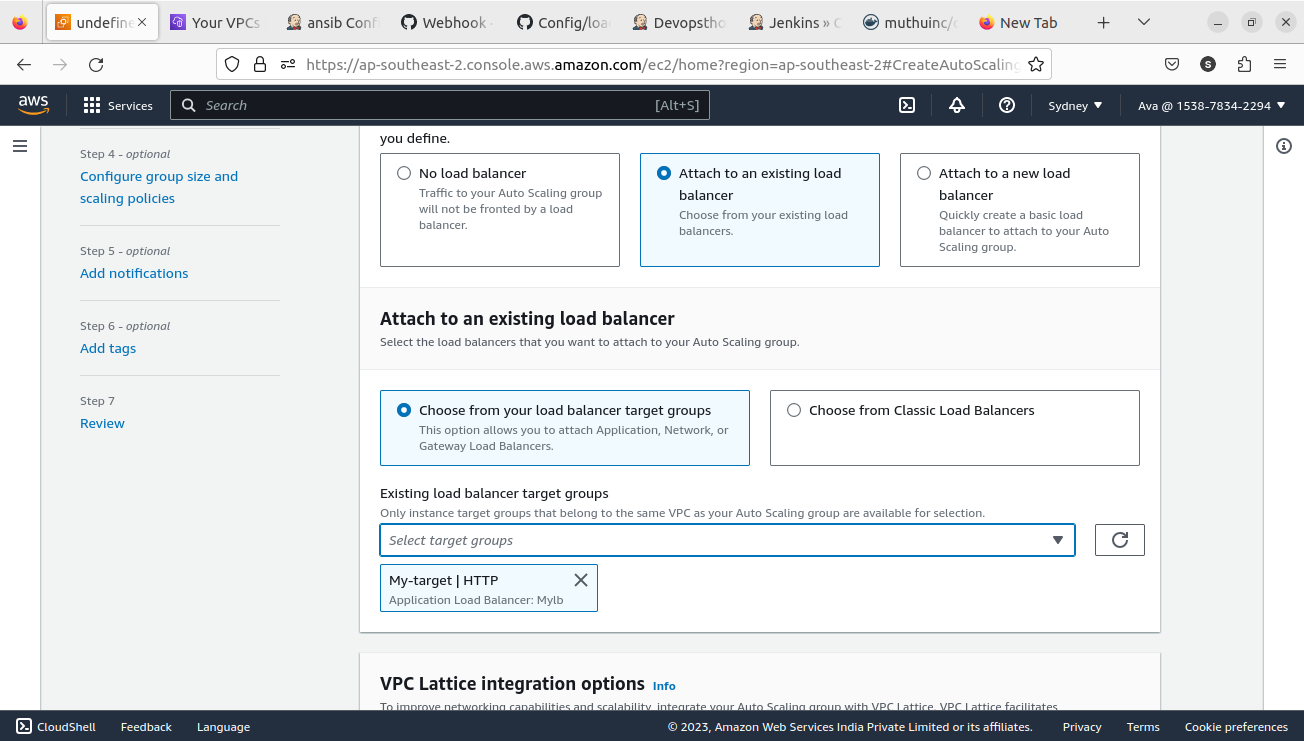
Listening to our target group which points our deployed server

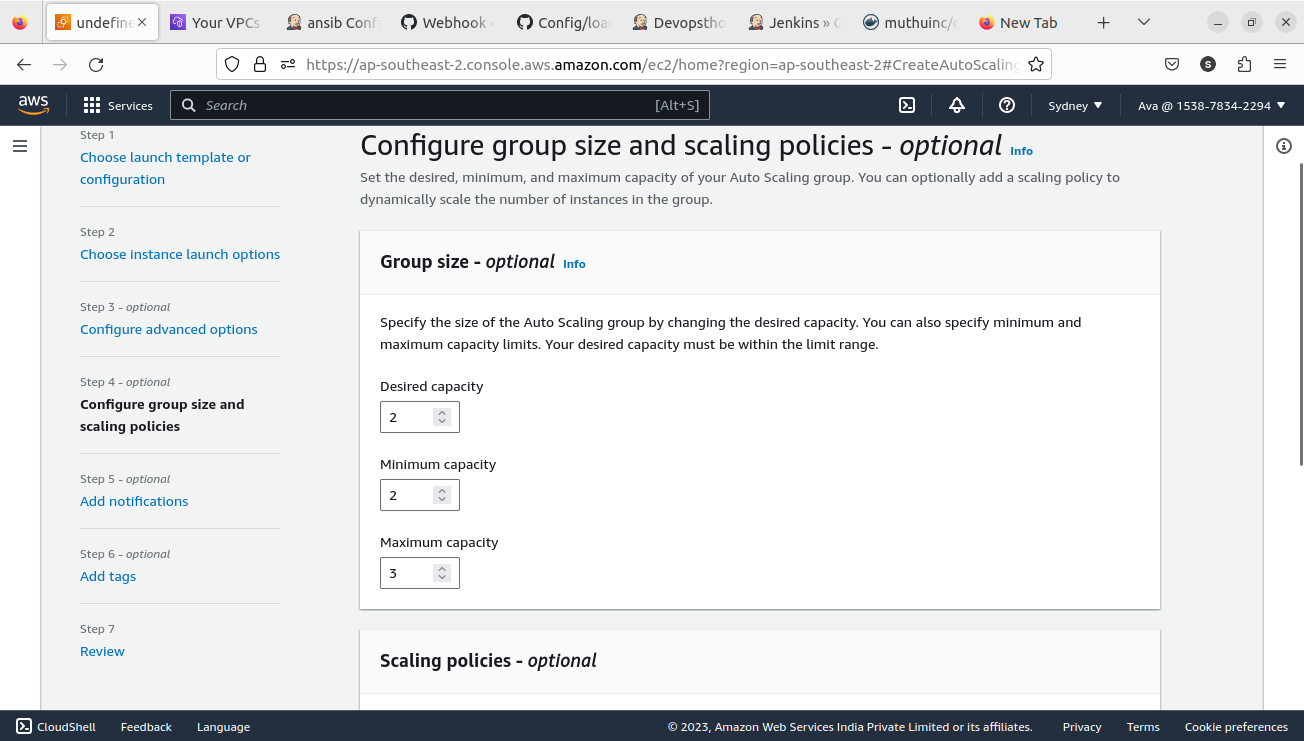


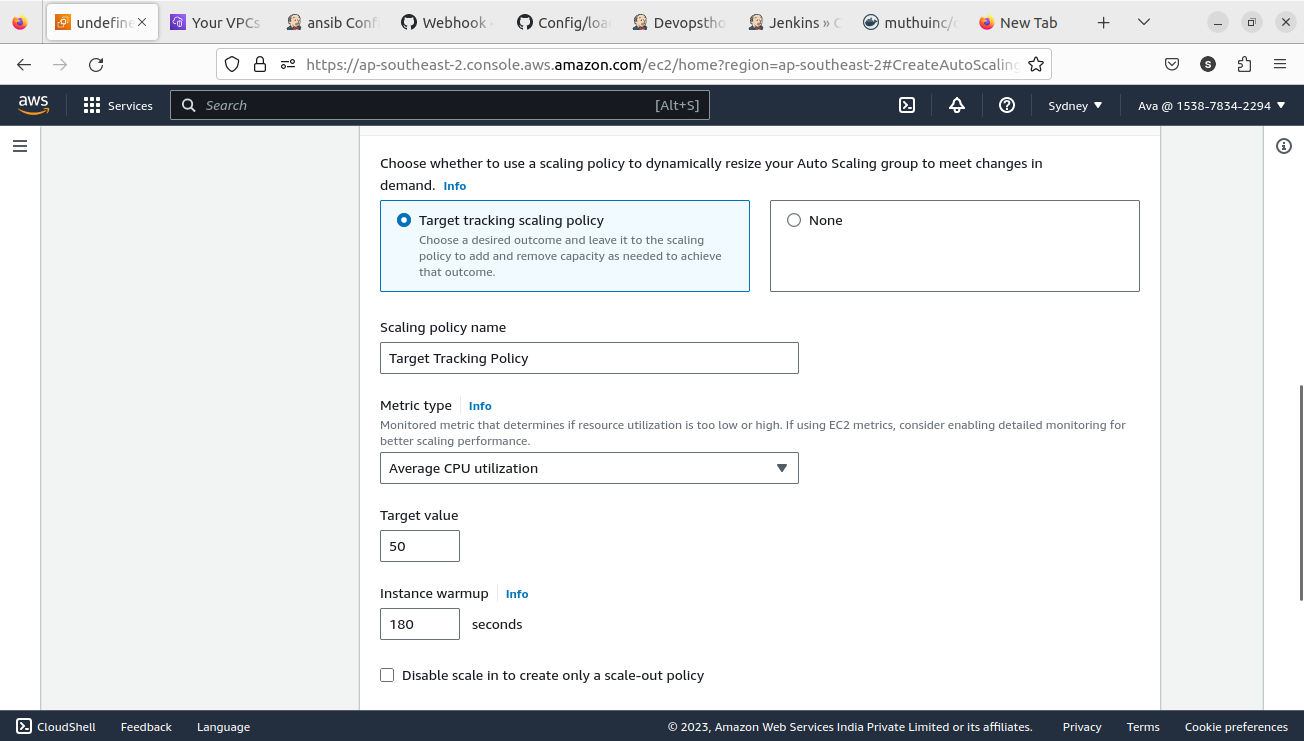
Autoscaling creation based on our template which I created manually in the previous steps, look at the names

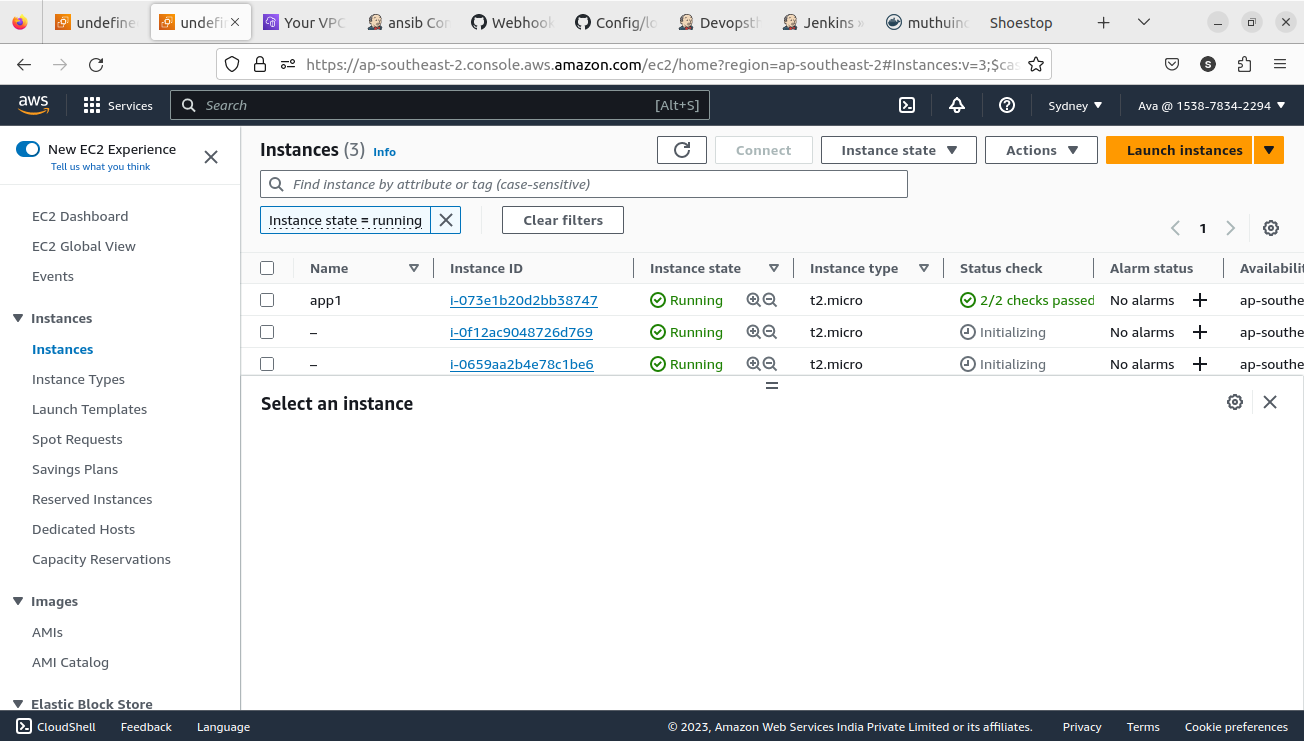


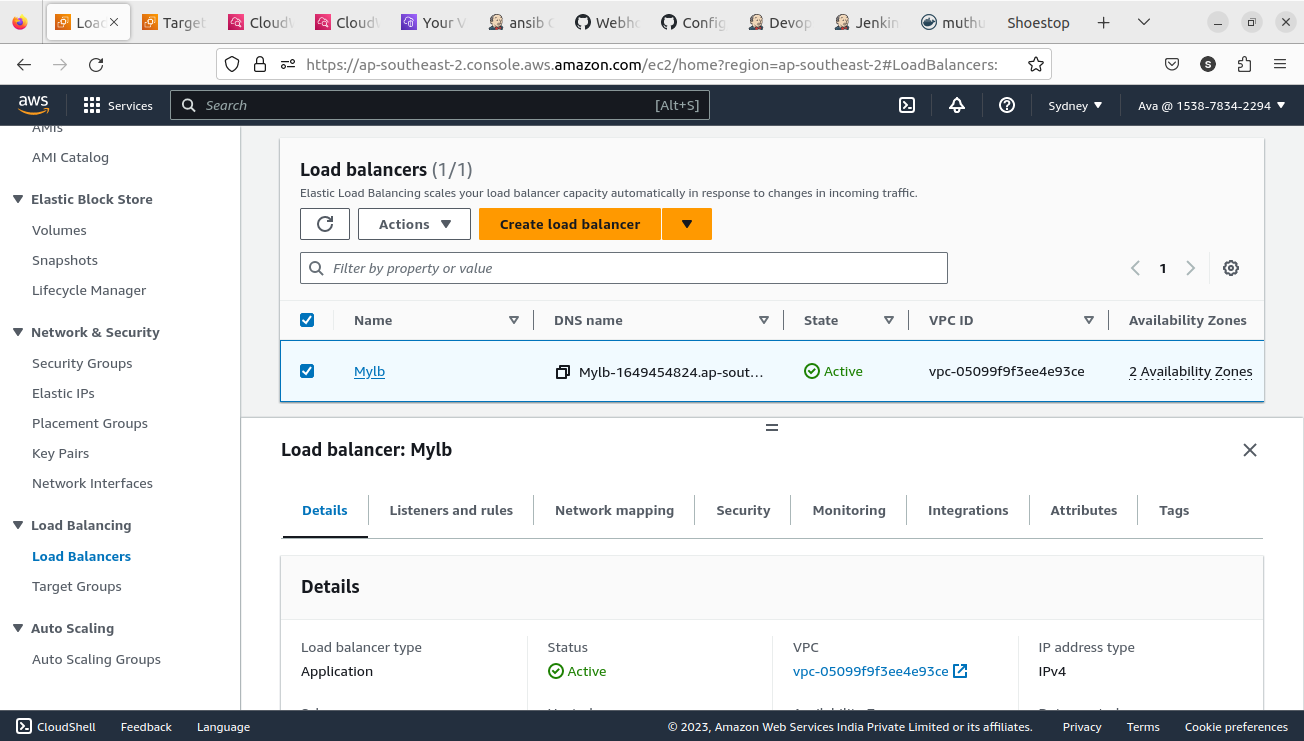
Attaching the autoscaling to our loadbalancer target group

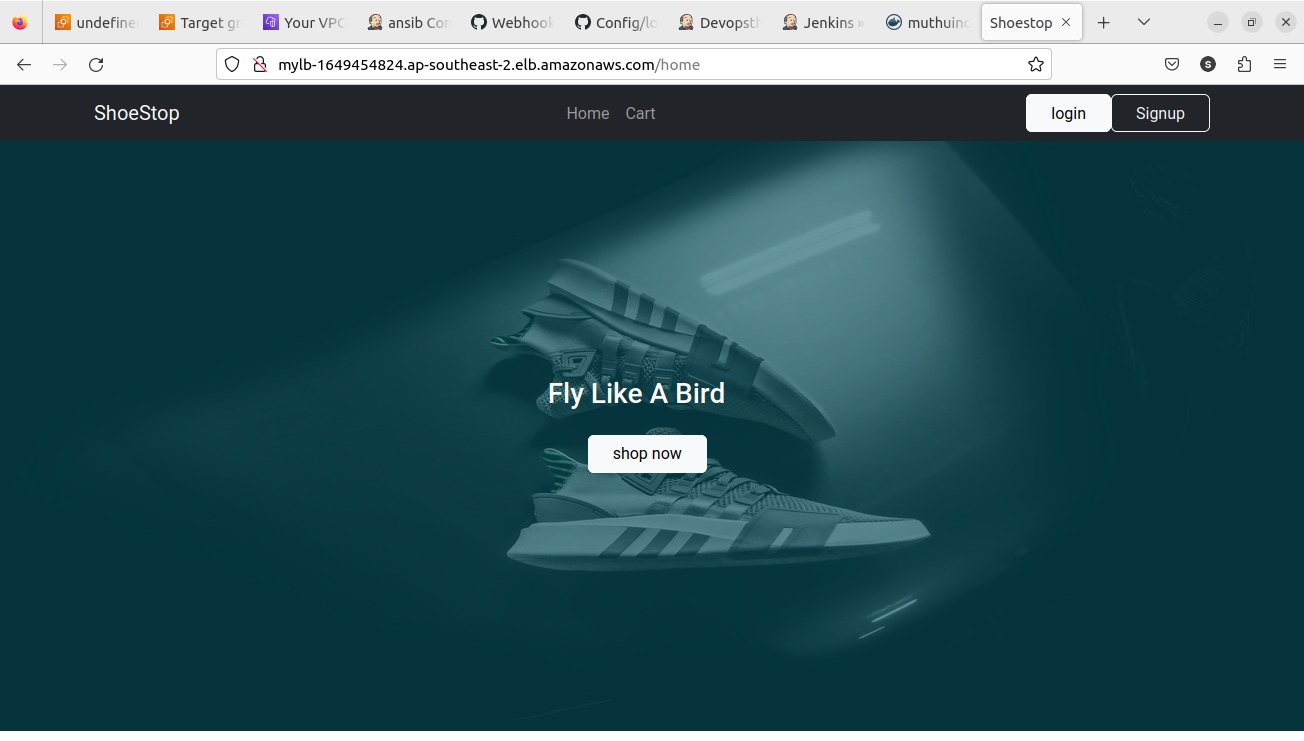






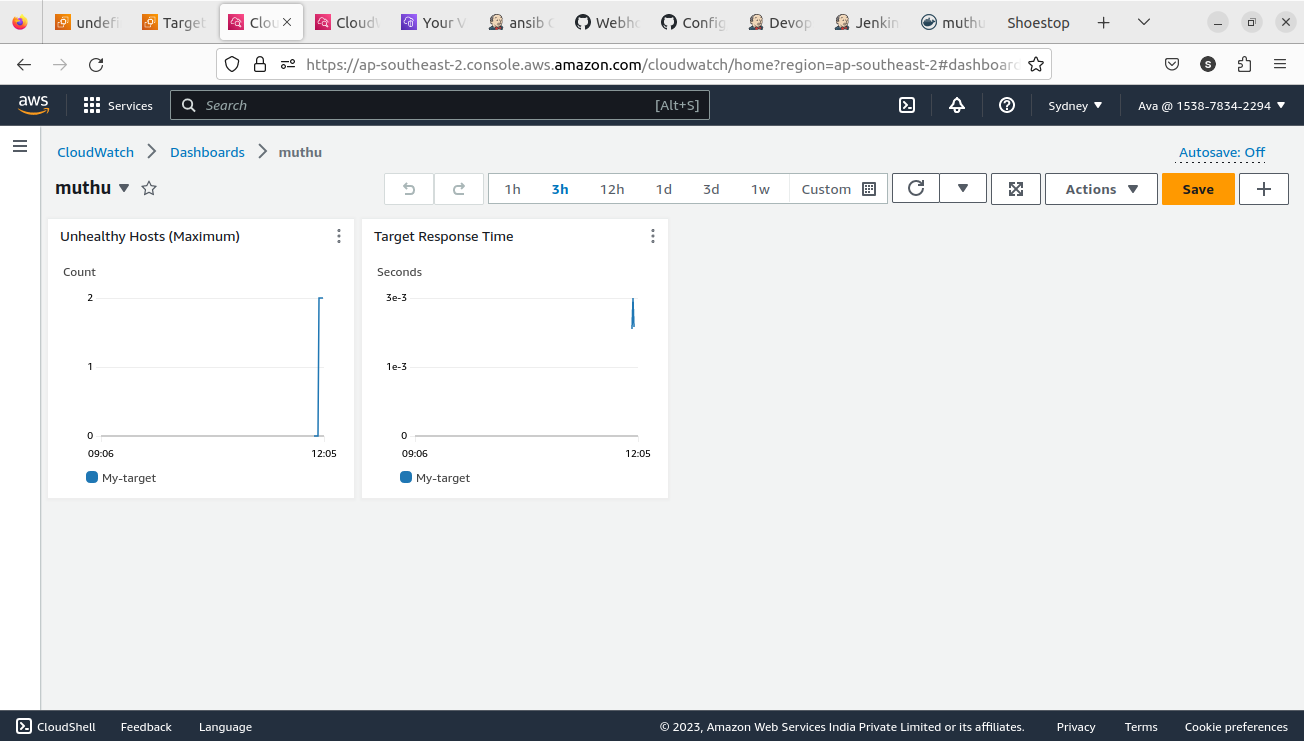
Auto scaling created the new applications based our condition 

the above picture is the loadbbalancer we created.

When we hit the loadbalancer DNS address our application is running 

We can use AWS route 53 to get a domain name for our application. The loadbalancer route traffic to all the instances one at a time link round-robin method.

This is the cloudwatch dashboard to monitor our application. Here I just put only 2 from the Loadbalancers metrics. We can configure more. We already configured in the autoscaling group to scale out when the cpu usage goes above 50%. It will take care of it.rgfvvcv



Here, we automated all the steps which is feasible for the normal process. From CI to CD. We achieved it. Autoscaling and loadbalancer creation done manually.

Thank you..