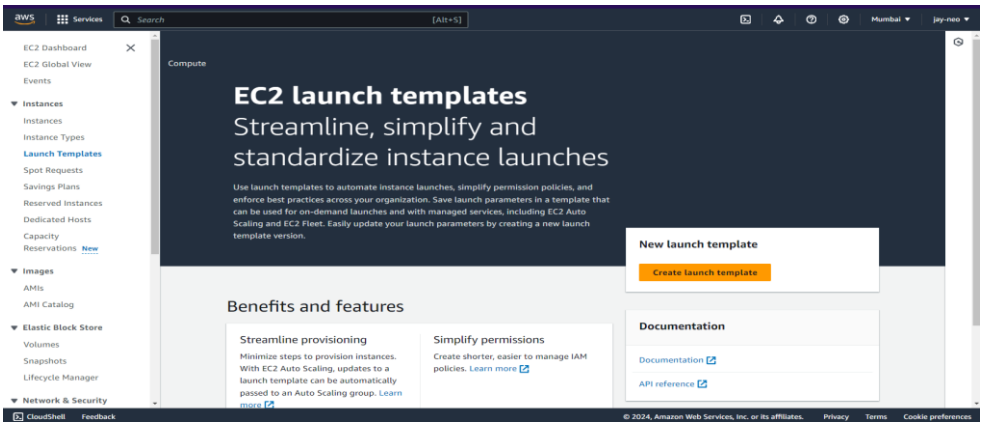
Assignment: 11

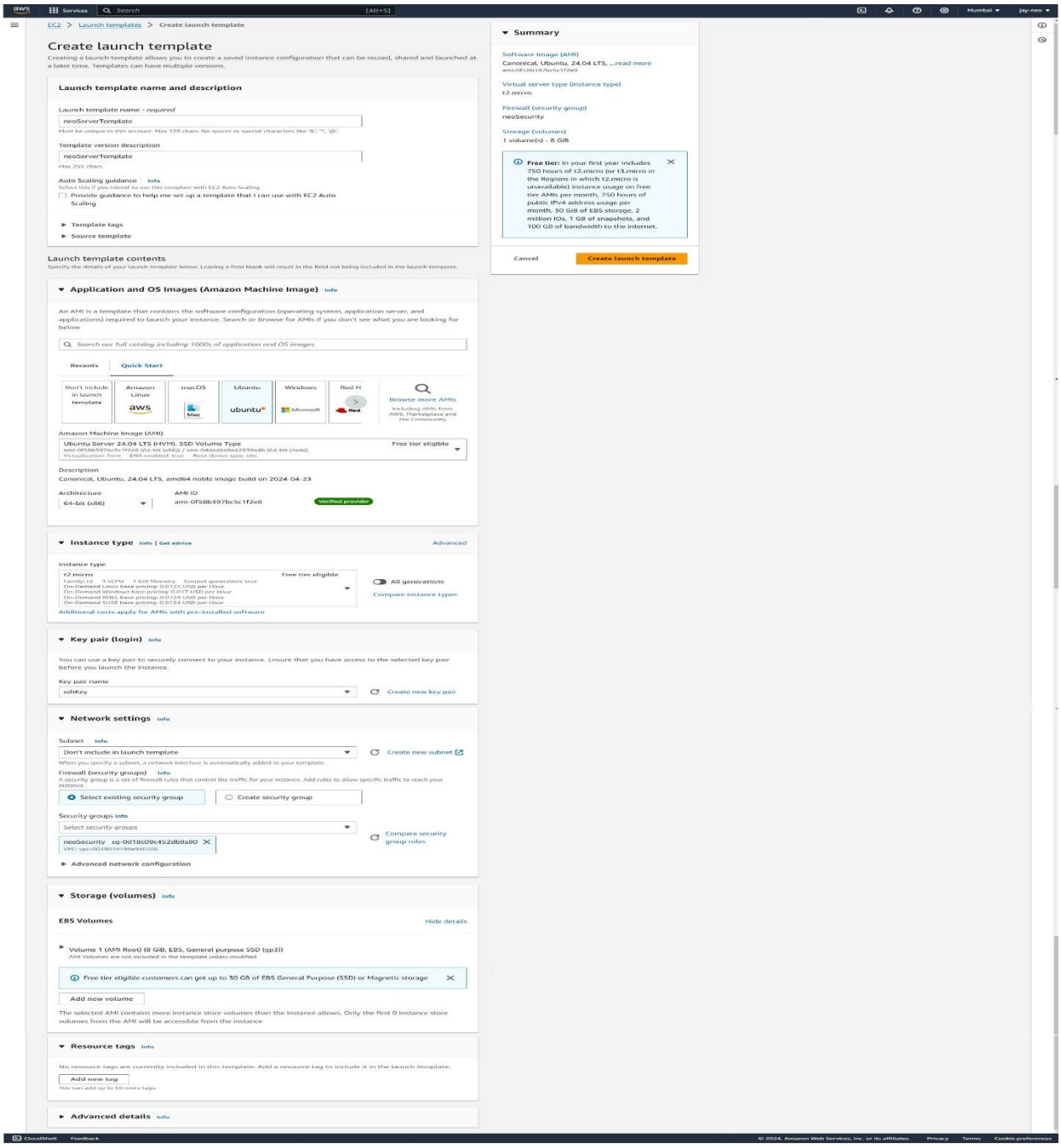
Problem Statement: Build scaling plans in AWS that balance the load on different EC2 instance.

Procedure:

Step 1: Go to the Launch Template under the section on EC2 Dashboard, then click **Create launch template**

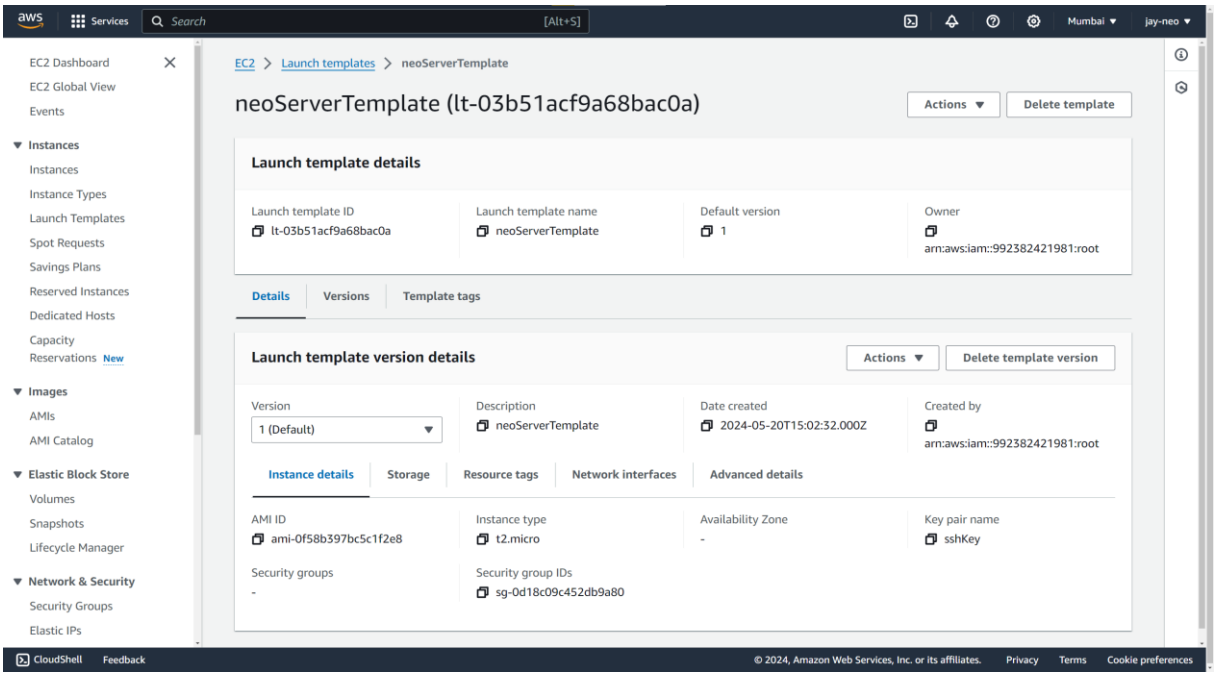


Step 2: Fill the required boxes following below,

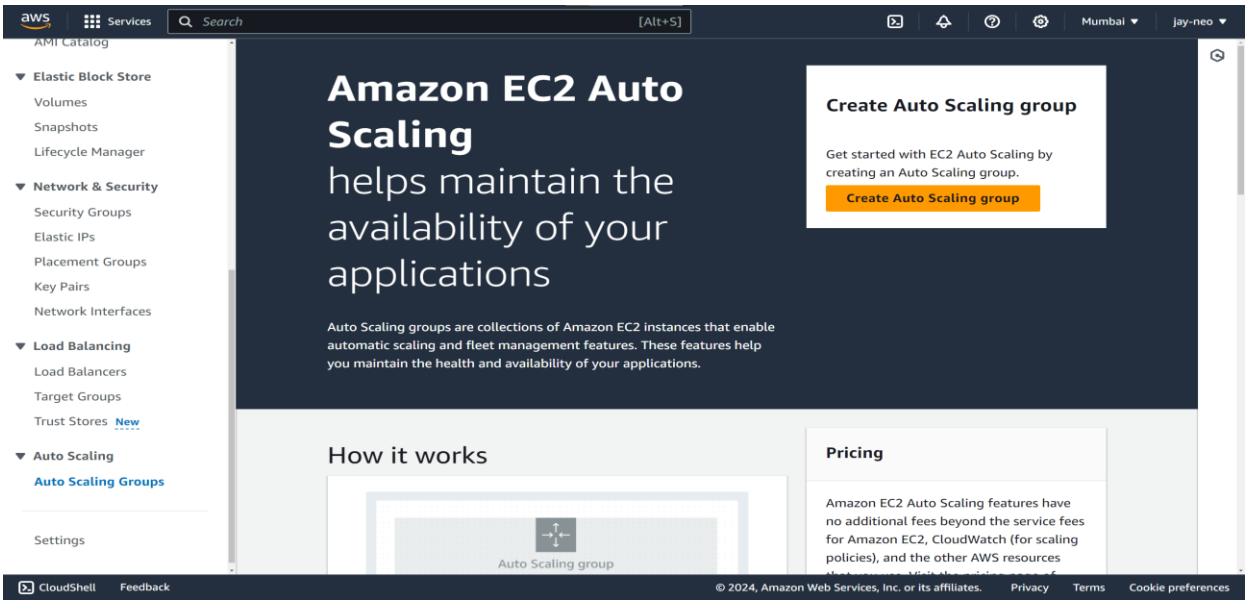


Then go to the Editor section exits in Advanced details and paste the following bash code, next click **Create launch instance**.

```
#!/bin/bash
apt-get update
apt-get install -y nginx
systemctl start nginx
systemctl enable nginx
apt-get install -y git
curl -SL https://deb.nodesource.com/setup_16.x|sudo -E bash -
apt-get install -y nodejs
git clone http://github.com/jay-neo/IT-Workshop-AWS-10.git
cd IT-Workshop-AWS-10
npm install
node index.js
```

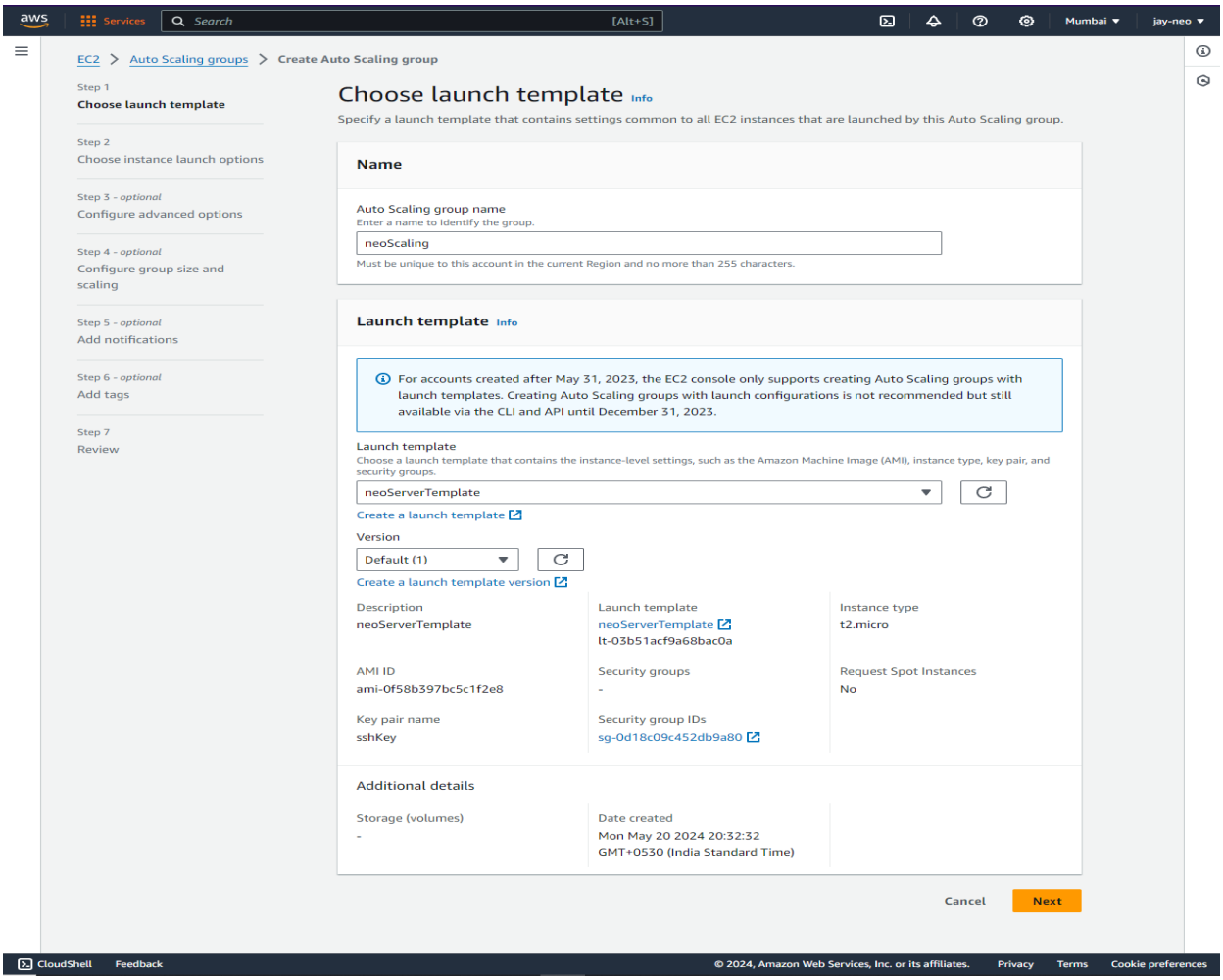


Step 3: Go to the **Auto Scaling Groups** under the section of EC2 Dashboard and click Create **Auto Scaling group** button.

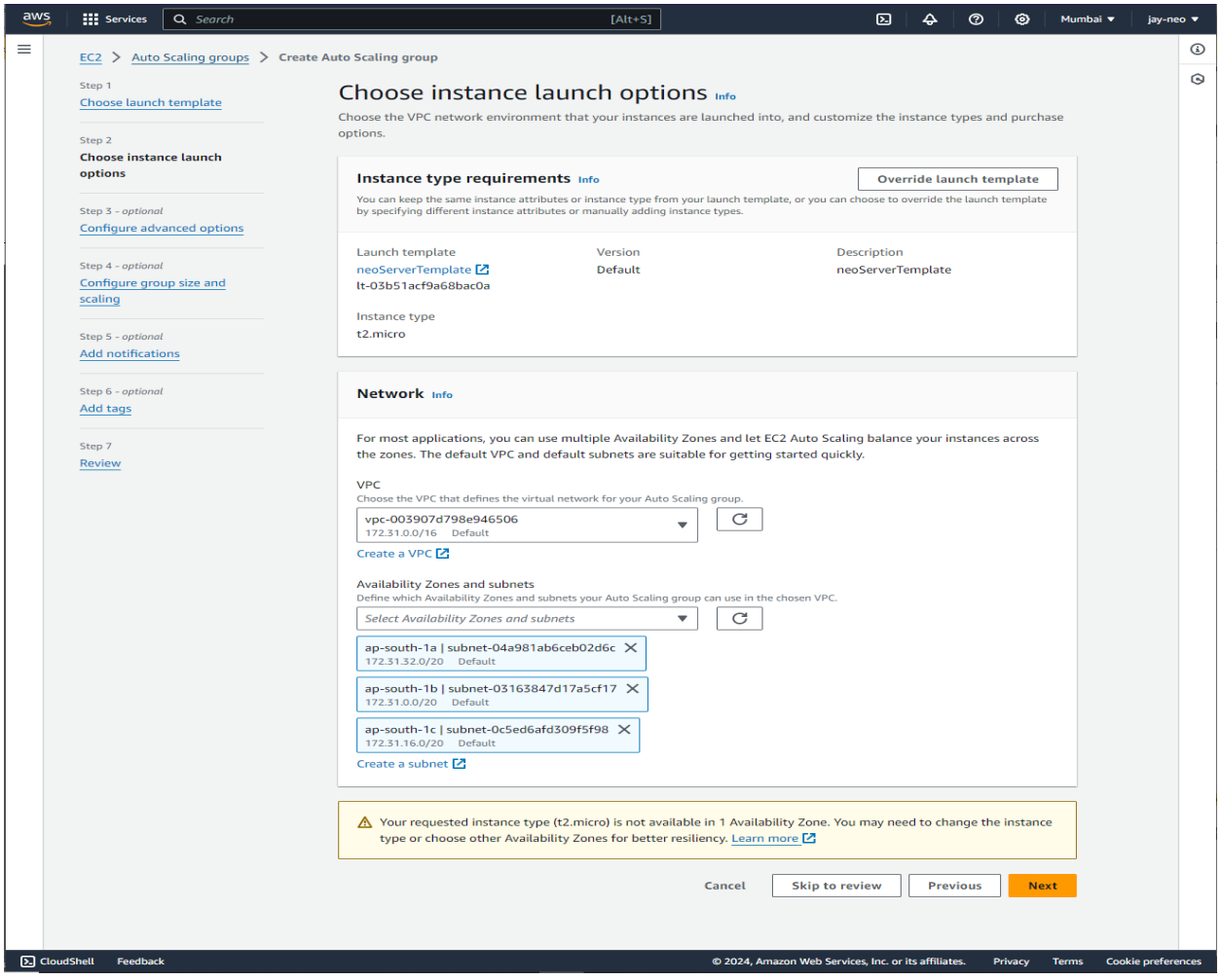


Step 4: Write After clicking the **Auto Scaling group** button fille all the necessary details following bellow,

Step 4.1: Give a name of Auto scaling group and select the Launch template that created at Step-2.

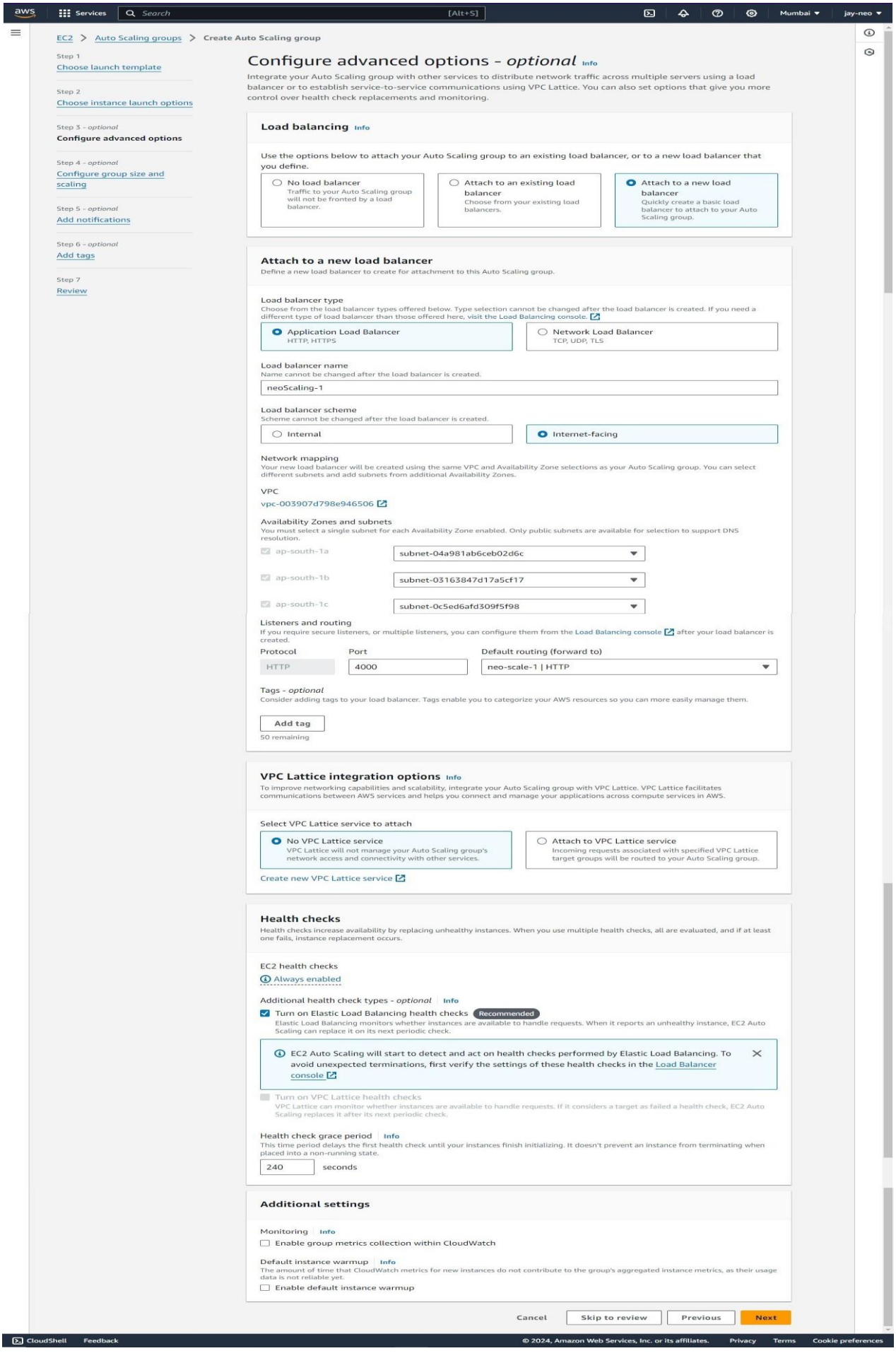


Step 4.2: Select the default VPC and choose all the Availability Zones and subnets.

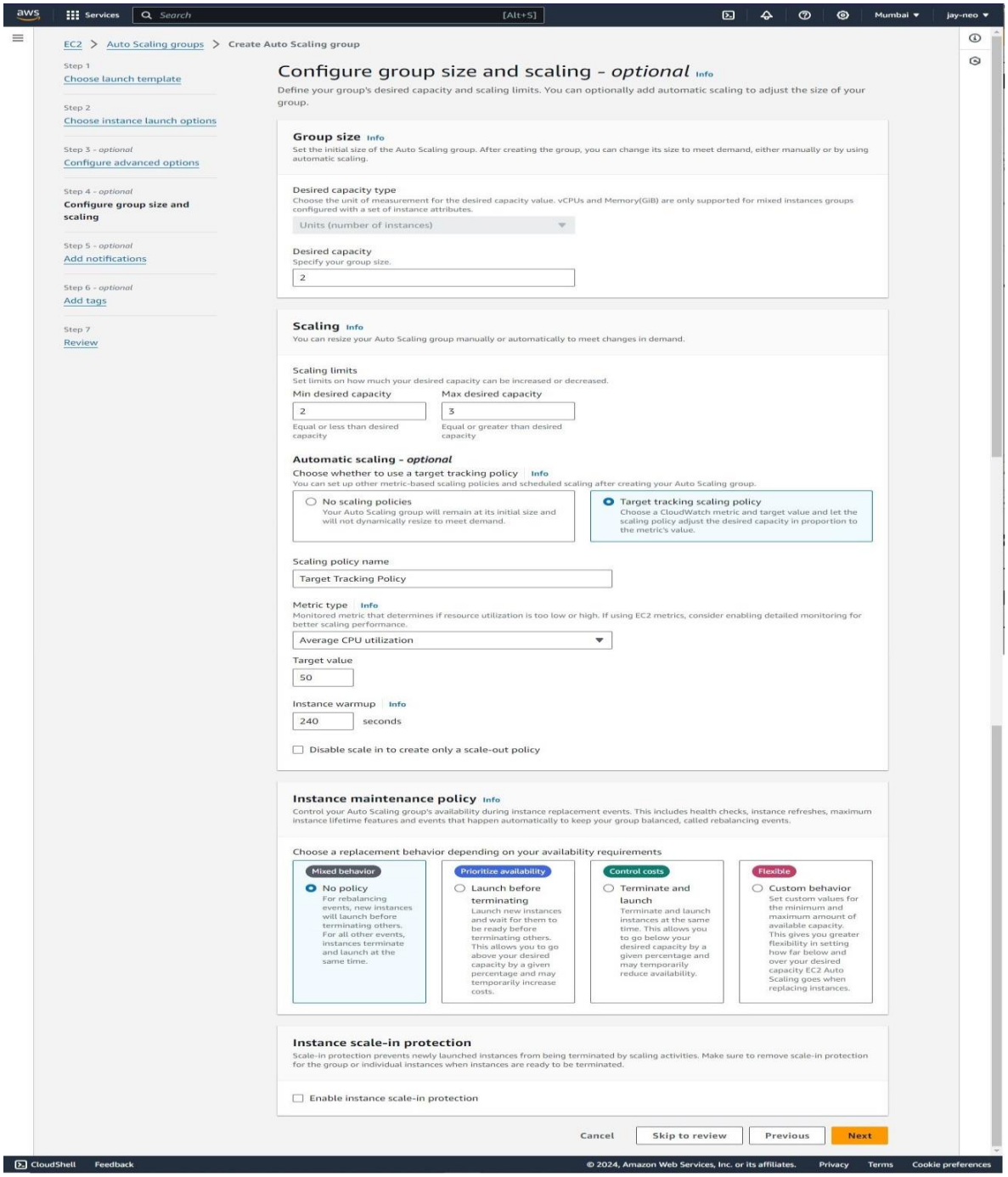


Step 4.3: Fill the configure advanced options following bellow,

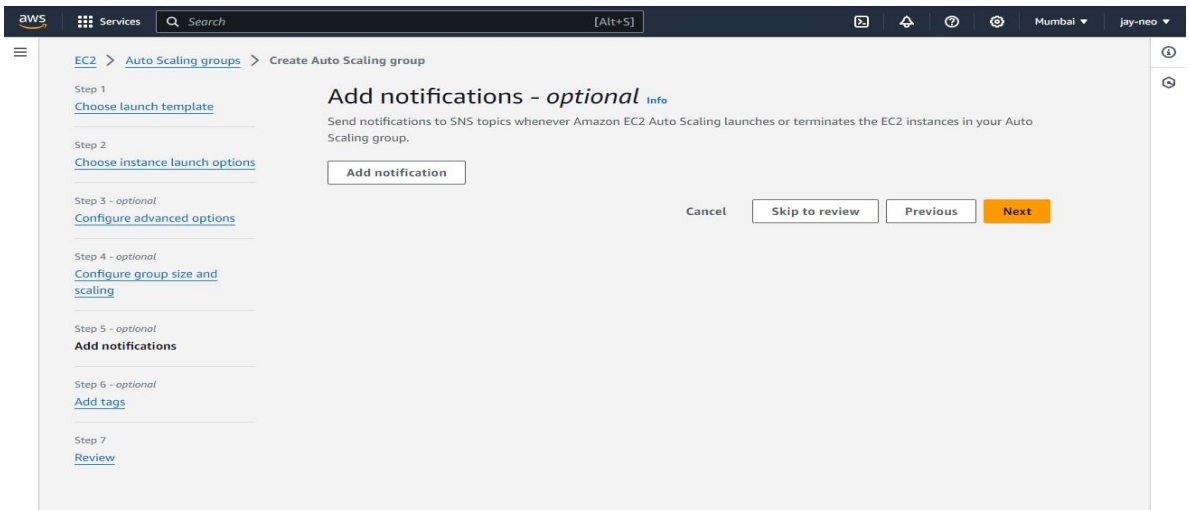
Load balancing: Attach to a new load, **Load balancer type**: Application Load Balancer, **Load balancer scheme**: Internet-facing, **Protocol**: HTTP 4000, **Target**: neo-scale-1 | HTTP
Check the **Turn on Elastic Load Balancing health checks**, **Health check grace period**: 240



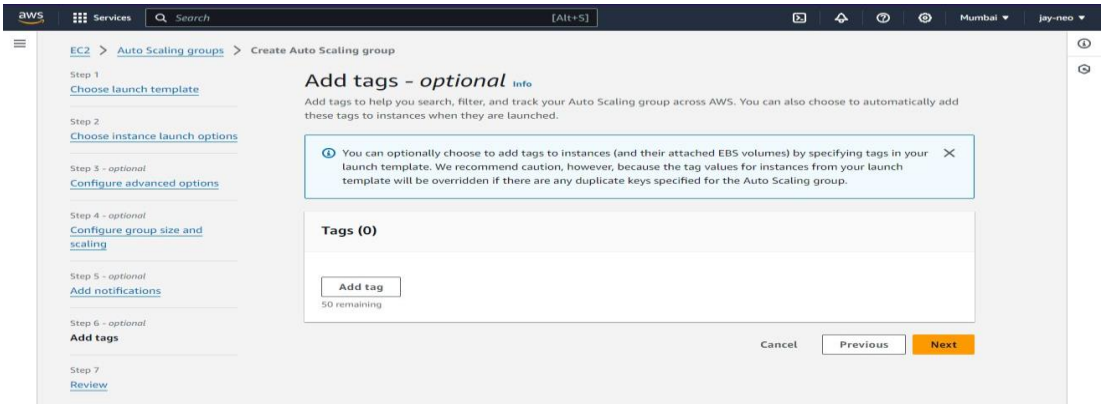
Step 4.3: In Confiure group size and scaling, **Desired capacity**: 2, **Min desired capacity**: 2, **Max desired capacity**: 3, select **Target tracking scaling policy**, **Instance warmup**: 240



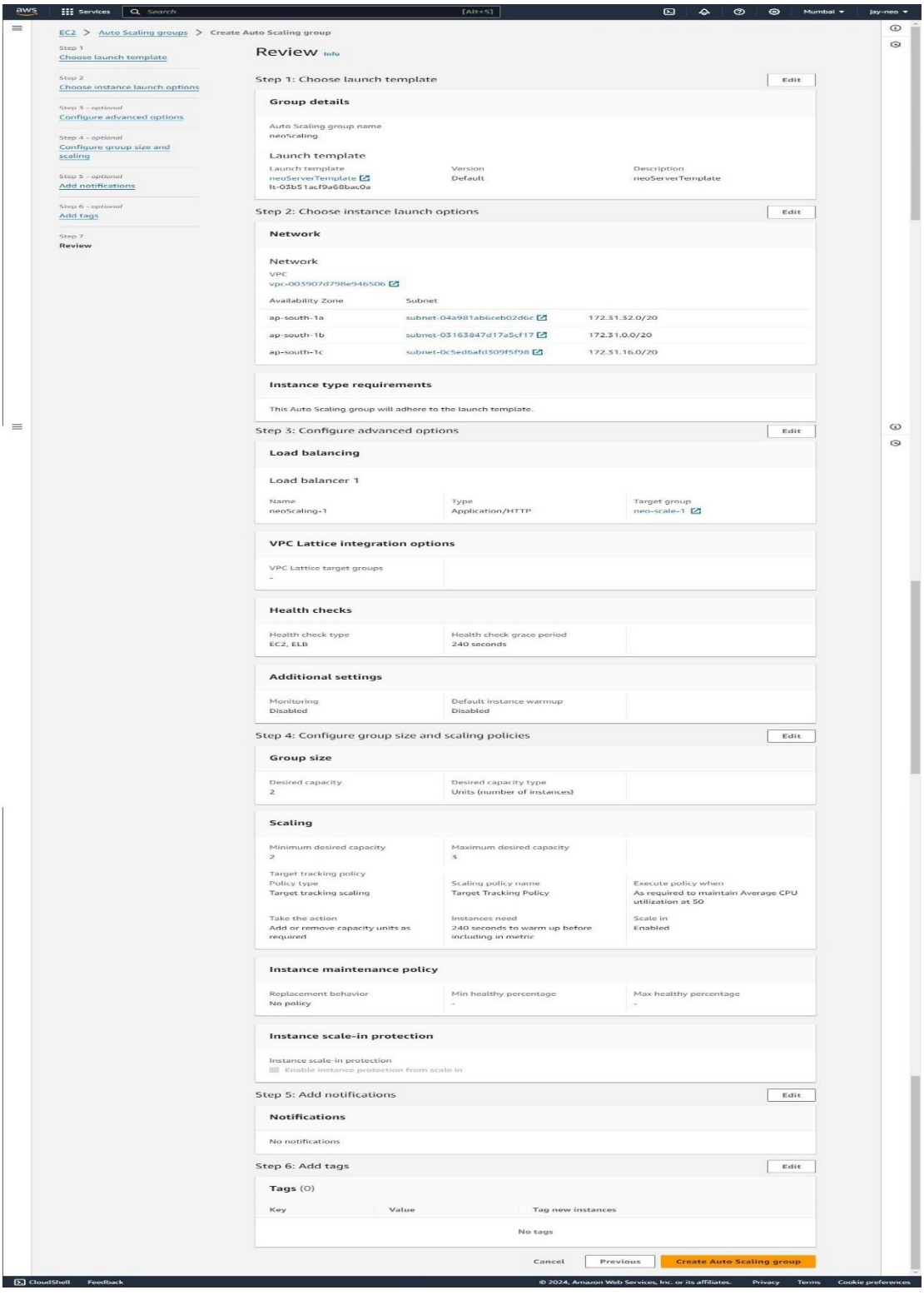
Step 4.4: In Add Notifications no change just click Next button,

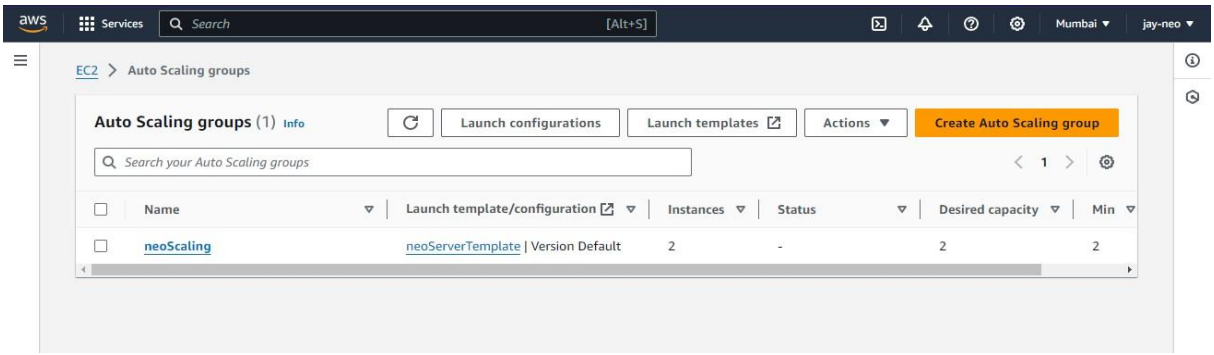


Step 4.5: In Add tags section no change just click Next button,

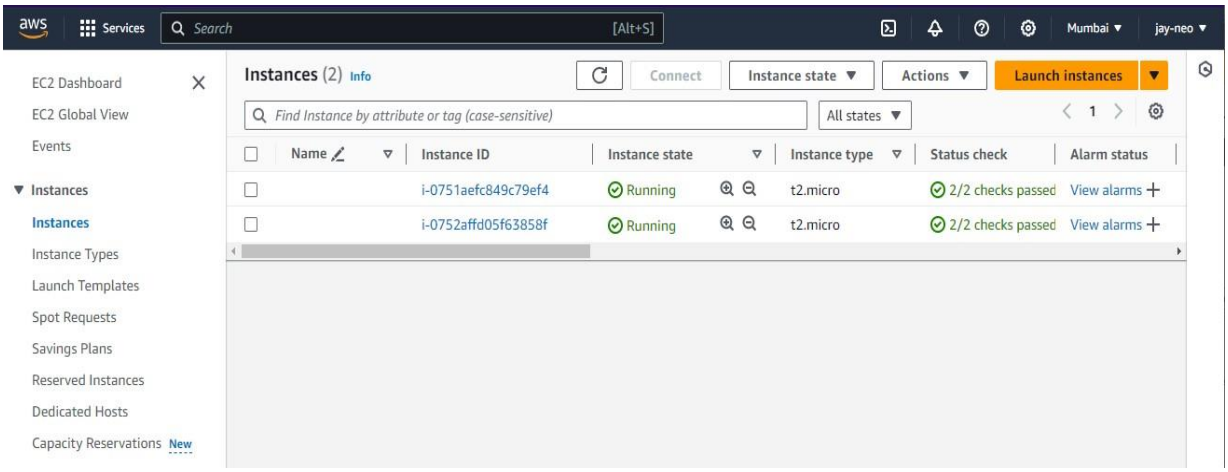


Step 4.5: Finally Review the Auto Scaling Group and Create Auto Scaling group

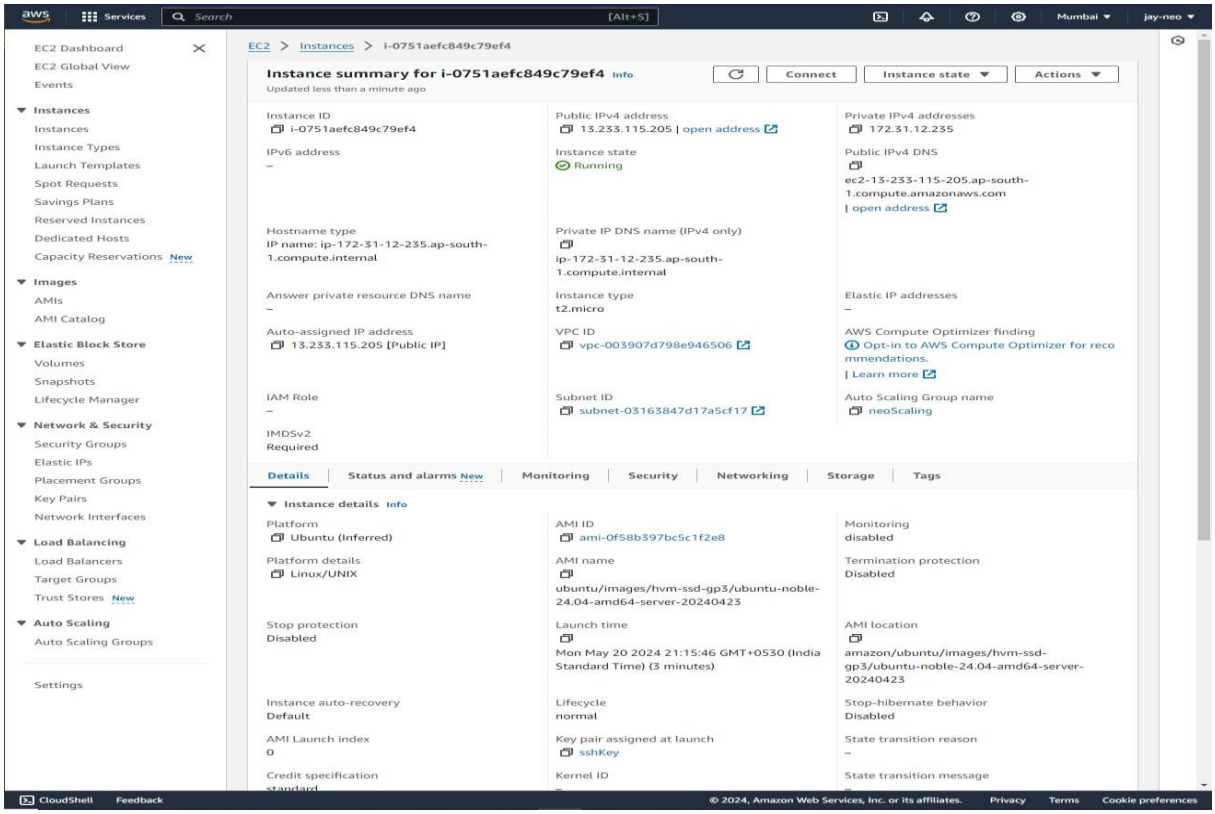


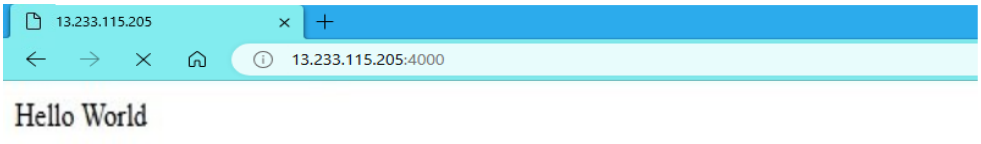


Step 5: After creating Auto Scaling group namely neoScaling, go to the Instance page under EC2 dashboard



Step 6: Copy public IPv4 of one instance from the any two and connect it using ssh command from terminal





Hello World

```
>> ssh ssh -i ~/Downloads/sshKey.pem ubuntu@13.233.115.205
```

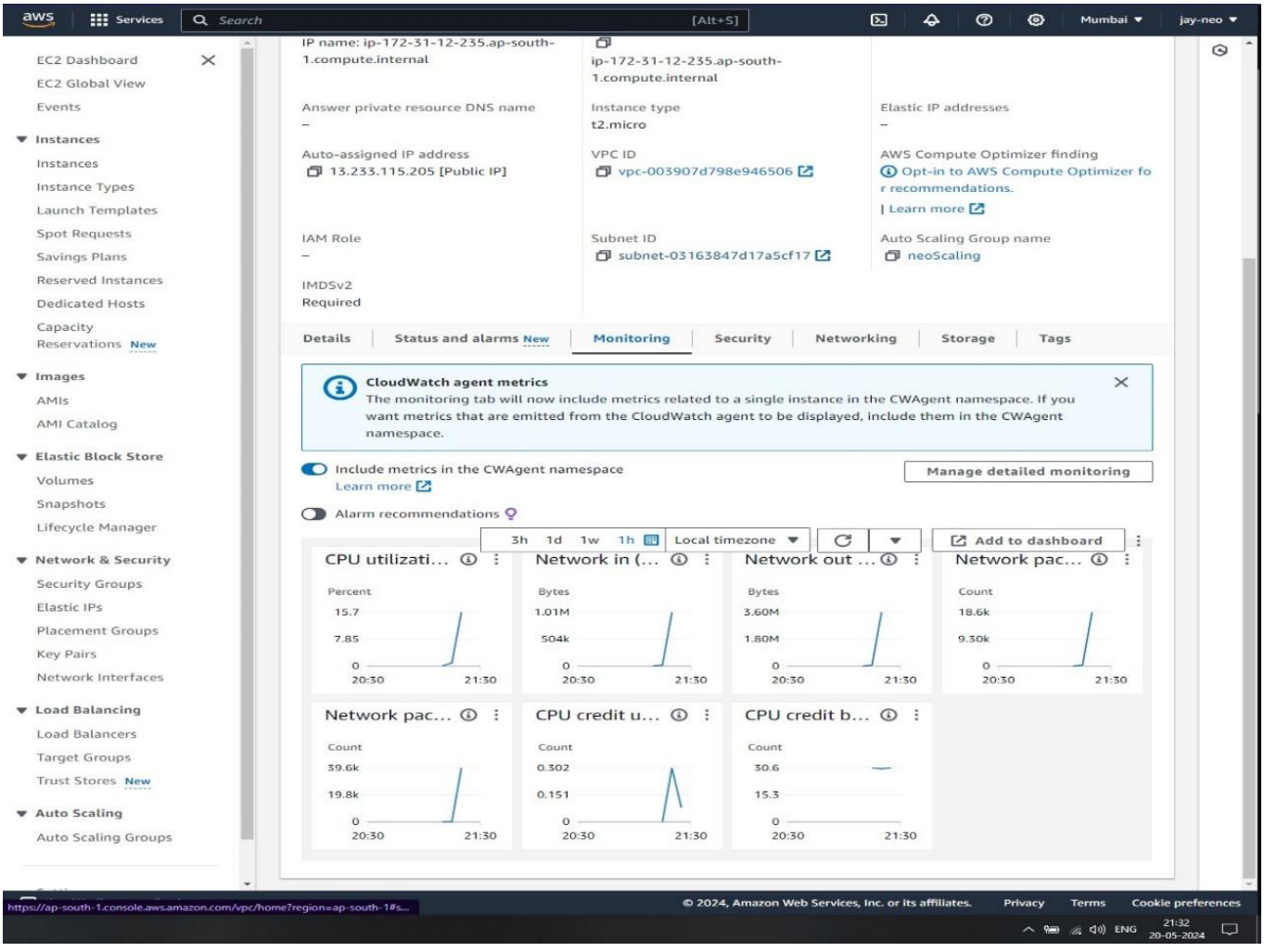
Step 7: Next, create a Bash script to implement an infinite loop program that monitors the server CPU utilization. When it reaches its limit, the auto scaling policy should create another instance to balance the load of the auto-scaling group.

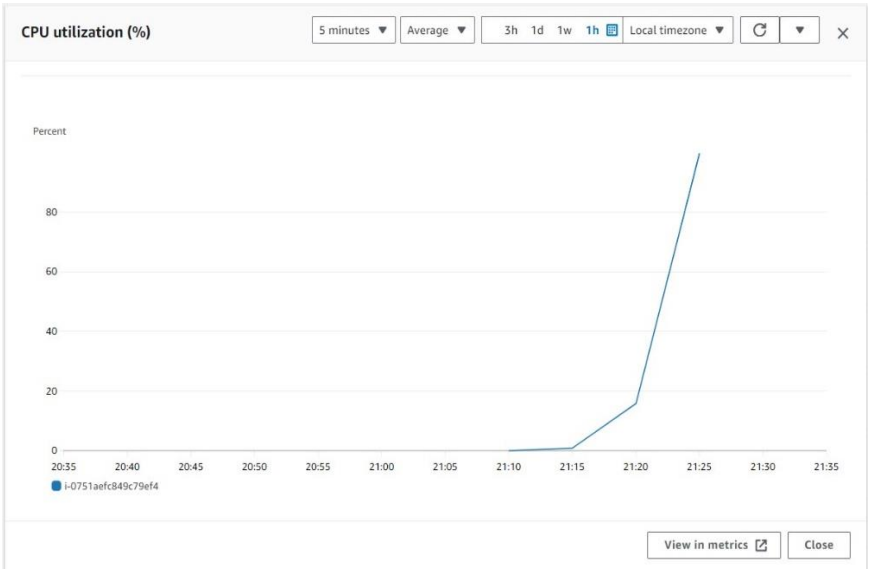
The bash script code in the **infy.sh** file is:

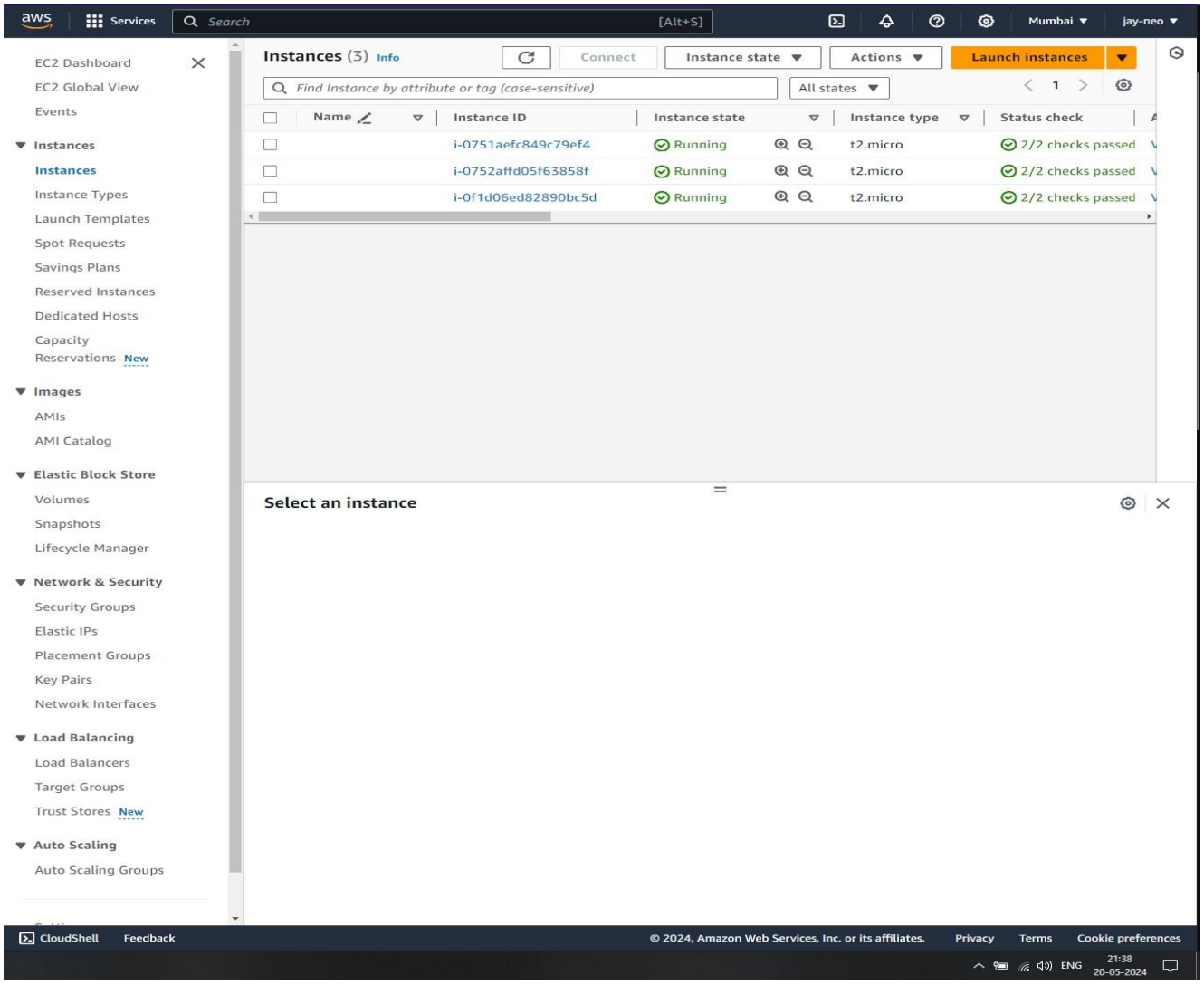
```
#!/bin/bash
while(true)
do
        echo "Inside loop …"
done
```

and the following commands run in the terminal is

Step 8: Then monitor the CPU utilization graph of the selected intance where the code is running







Step 9: To delete the instances permanently delete the Auto Scaling group as well as Load Balancers, Target Groups and Launch Template.