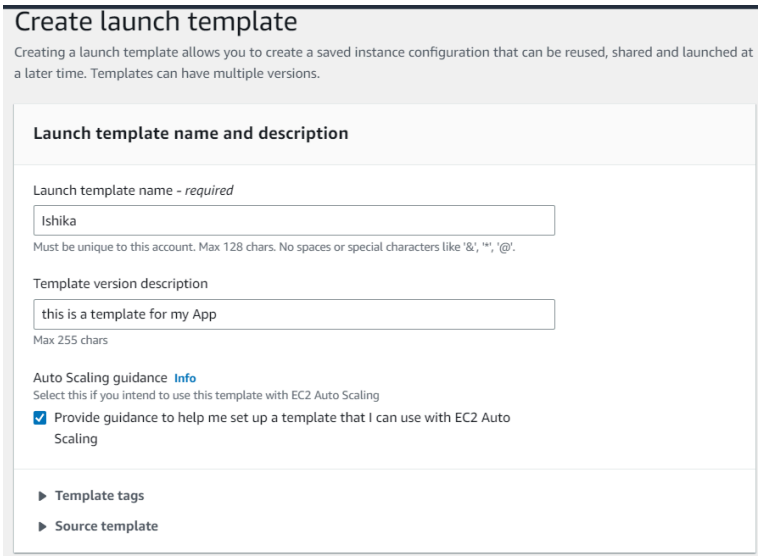
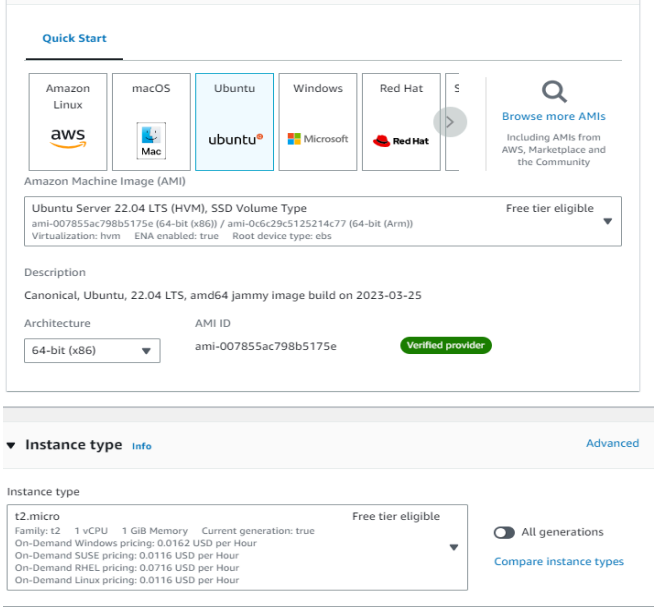
Assignment 11: Build Scaling plans in AWS that balance the load on different EC2 instances.

1. Creating a launch template

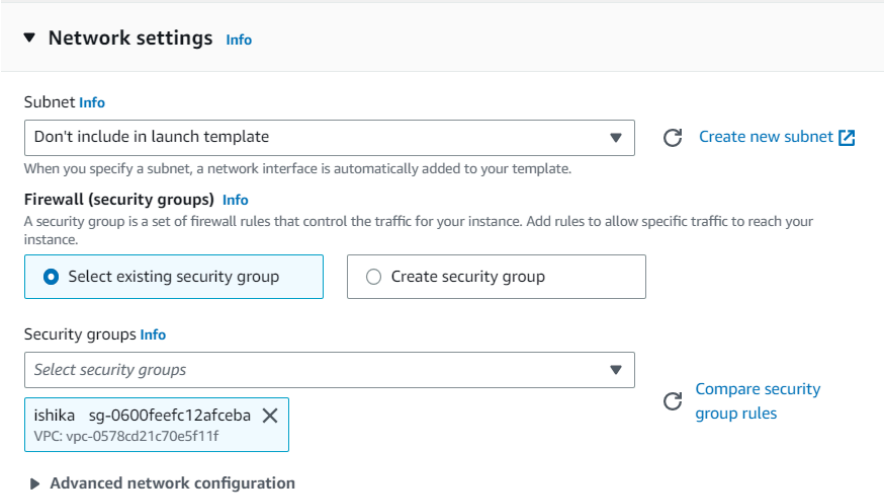
a. In the left panel, under instances, go to **Launch Templates** option and click on **Create launch template**.

b. Give the name and description of the template and select the checkbox. Under **Application and OS Images** under **Quick Start** select **UBUNTU**, under **Instance type** select **t2.micro**, and create a key pair.





c. Under **Network Settings**, select the **security group** created before.



d. Go to **Advanced details** and in user data write the required data/commands.

- #!/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_18.x | sudo -E bash -
- apt-get install -y nodejs
- git clone https://github.com/ishika-ghosh/awsproject.git
- cd repo awsproject
- npm install
- node index.js

(Before creating the Template make the required GitHub repo public)

User data - *optional* **Info**

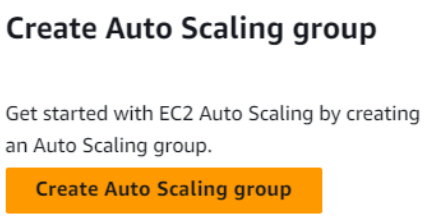
Enter user data in the field.

```
#!/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_18.x | sudo -E bash -
apt-get install -y nodejs
git clone https://github.com/ishika-ghosh/awsproject.git
cd awsproject
npm install
node index.js
```

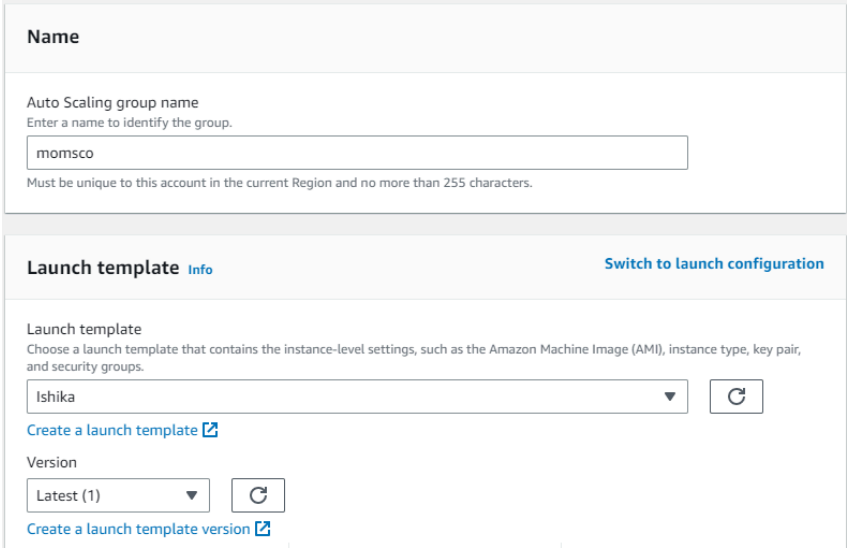
e. Create the template.

2. Creating Auto Scaling Group

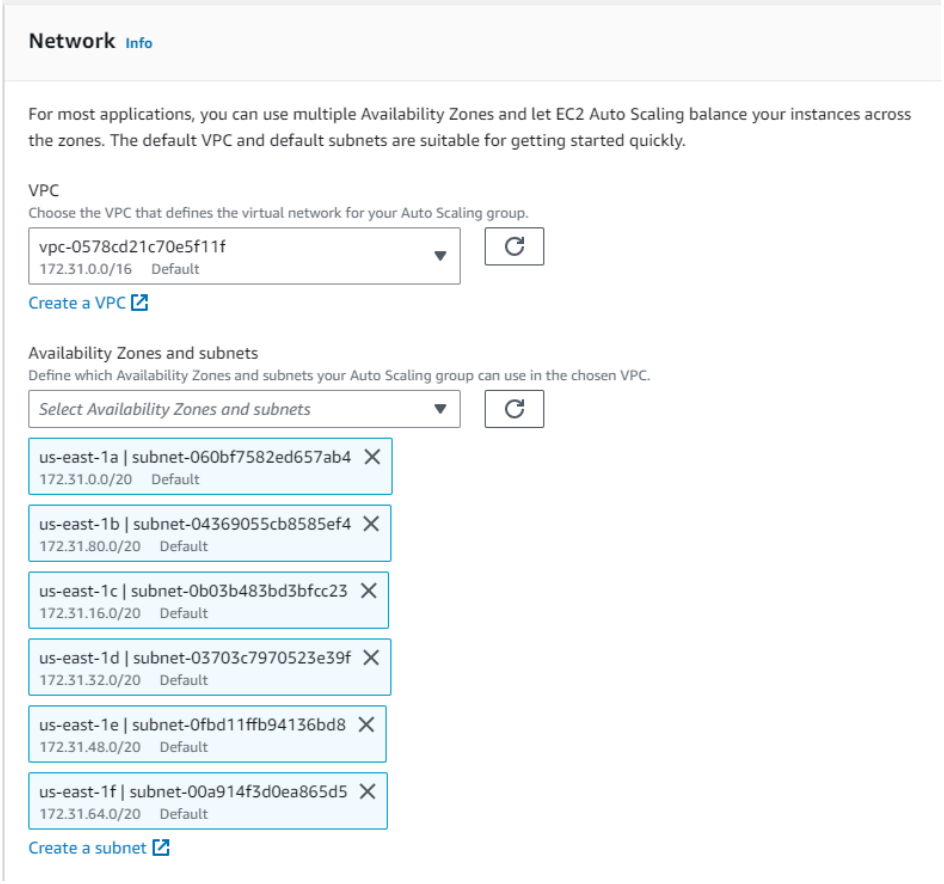
a. In the left panel, under **Auto Scaling,** go to **Auto Scaling Groups.** Click on **Create Auto Scaling Group**.



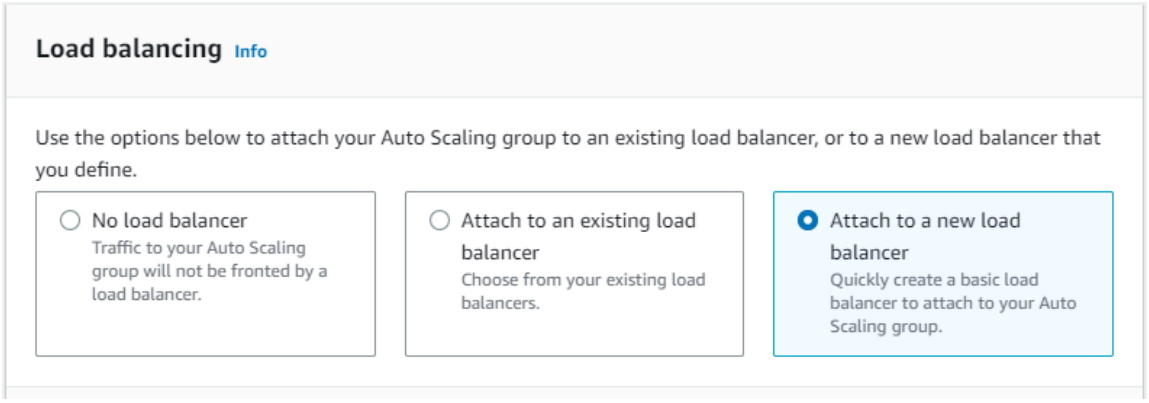
b. Give Name of the auto scaling group,select the launch template created and select version as **Latest(1).** Click on **Next**.

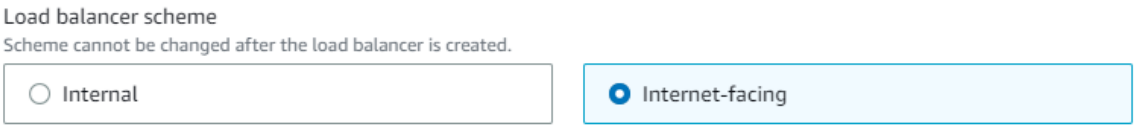


c. Choose all the **Availability zones** and **subnets** and go to next page.

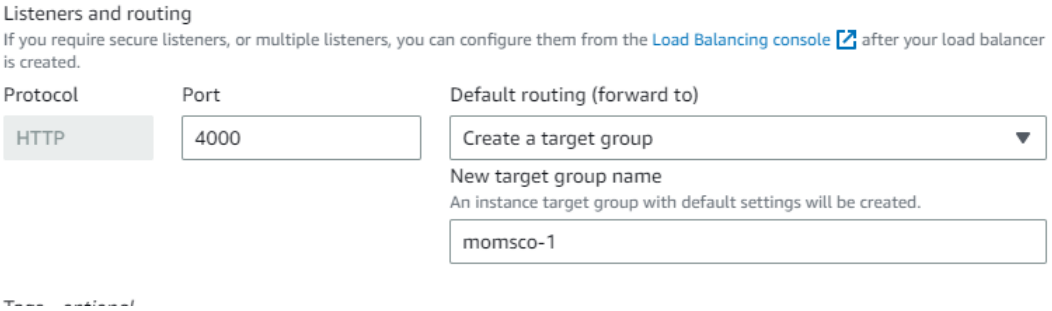


d. Select **Attach to a new load balancer**. Choose Load balancer schema as **Internet-facing**.



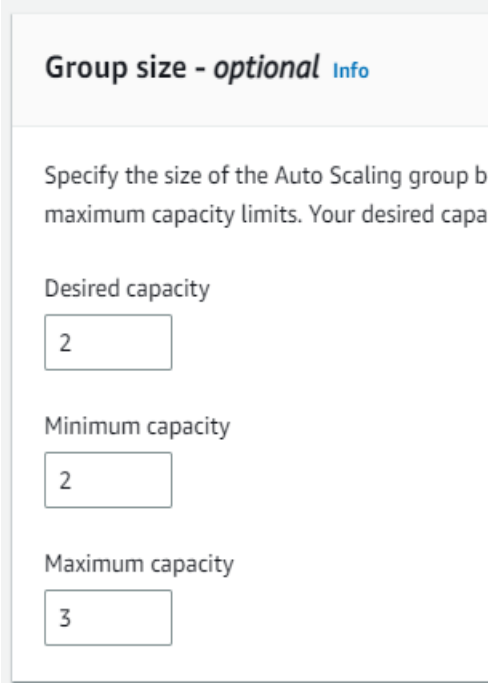


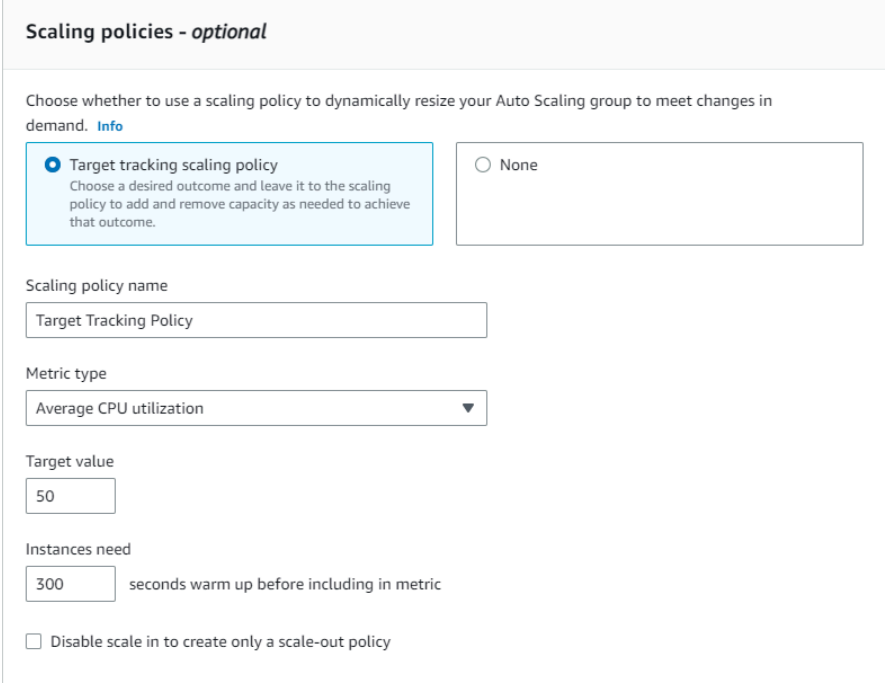
e. In **Listeners and routing** give **Port** as **4000** and for **default routing** select **Create a target group** and click the target group showing. Click on **next**.



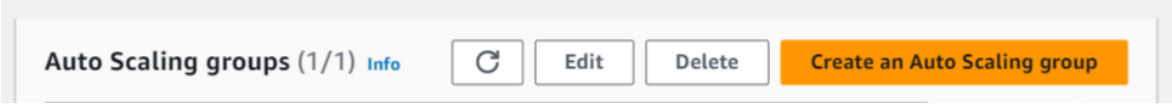
f. In Group Size give the **Desired**, **Minimum** and **Maximum** capacity as **2,2 and 3** respectively.

g. In **Scaling policies** select **Target tracking scaling policy**. Select **Target value** as **50** and set Instances need **300 seconds warmup**.

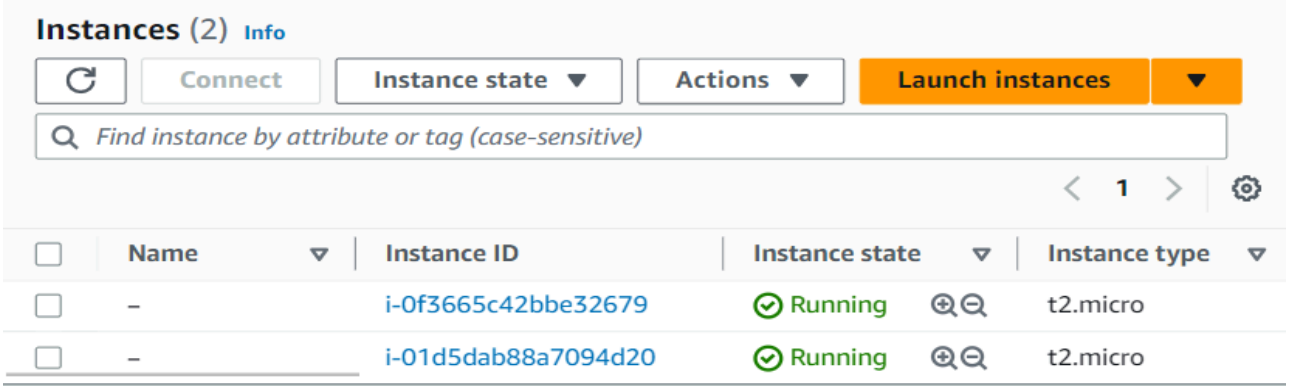




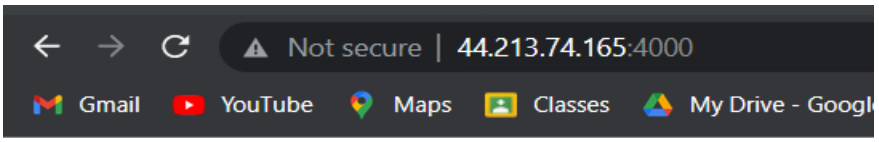
h. Go clicking on Next and create auto scaling group. **Auto scaling group created**.



3. Open the Instance and we will see two instances created.

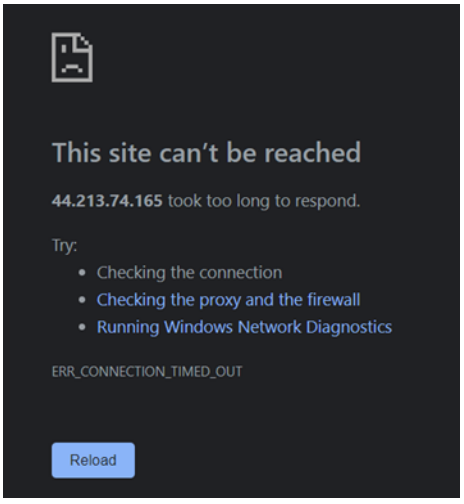


4. Copy the public IPv4 of an instance and open in web browser to see it is working properly. Give port as 4000 to see the proper project running.

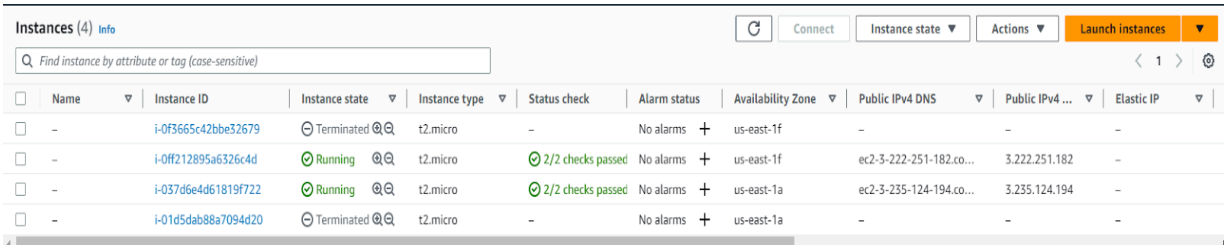


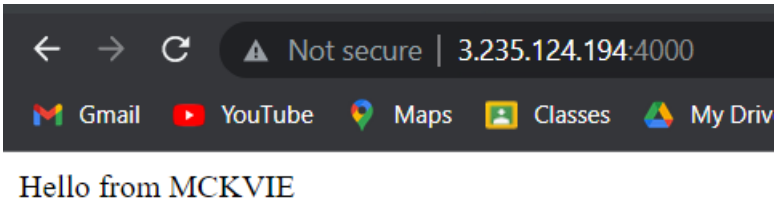
Hello from MCKVIE

5. Now select both instances and stop them (crashing the server) and refresh the website. We will no longer see the project running.

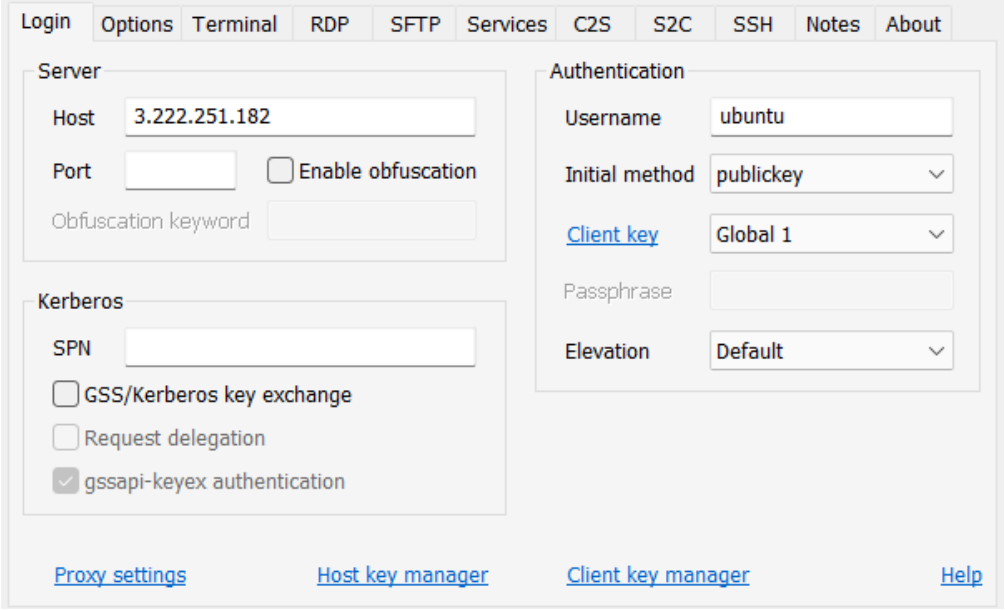


6. New Instance will be created automatically. Open the project with the help of this instances' public IP address and port 4000. The project will be running.

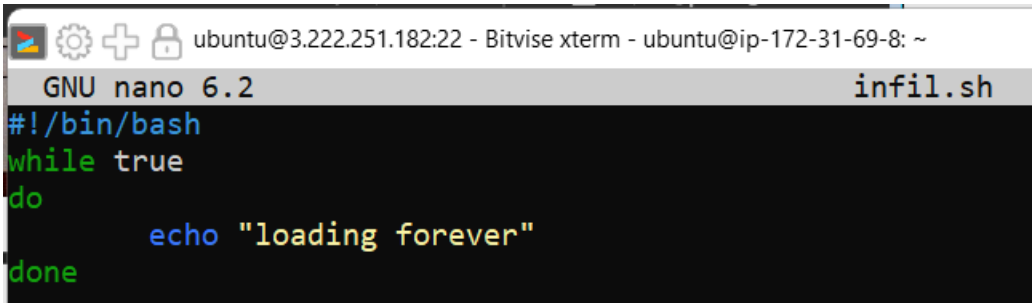


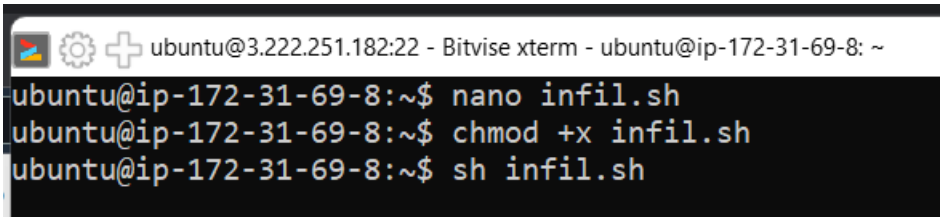


7. Open **Bitvise SSH Client and connect a running instance.**



8. Login and open a new terminal console. Write an infinite loop code save and necessary permission (**chmod +x infil.sh**) and execute it.



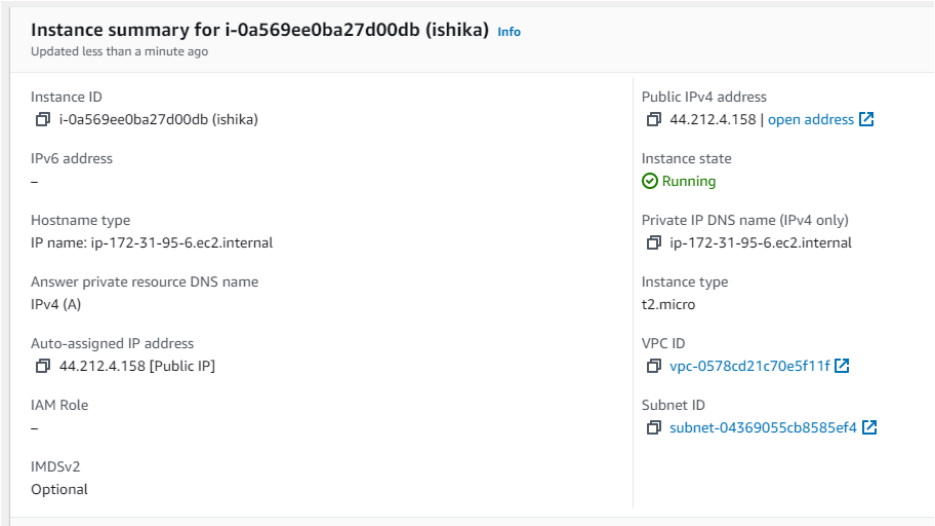


9. The server running the infinite loop will be overloaded and new instance will be created.
10. Selecting the 3 running instances, go to **Monitoring** section and enlarge the **CPU Utilization** section. Go to **custom** and select the **local time zone**. We can see new instance is created when one of the servers crossed 50% and the performance of these instances.

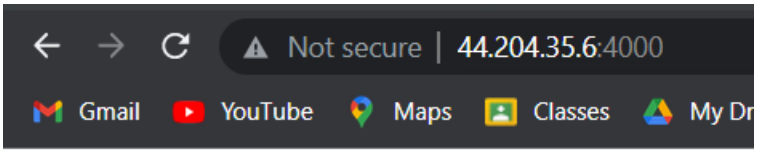
Assignment 12: Deploy and run the project in AWS without using Port.

1. Create an **EC2 instance** with **Ubuntu** OS and the **security Group** created before. In Advanced details type the following in user data field.
    - #!/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_18.x | sudo -E bash -
    - apt-get install -y nodejs
    - git clone repo link
    - cd repo name
    - npm install
    - node index.js

    create the instance.

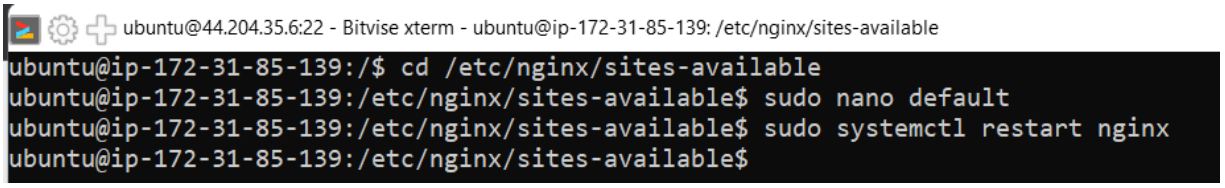


2. Run the project using public IPv4 and port 4000.



Hello from MCKVIE

3. Connect to the instance using Bitvise SSH Client and open new terminal console.
4. Go to the directory **cd /etc/nginx/sites-available** and open the default file using command :**sudo nano default**.



5. Comment the existing location part of the code and write the given code in that place.

```
location / {
proxy_pass http://localhost:4000;
proxy_http_version 1.1;
proxy_set_header Upgrade $http_upgrade;
```

proxy_set_header Connection 'Upgrade';

proxy_set_header Host $host;

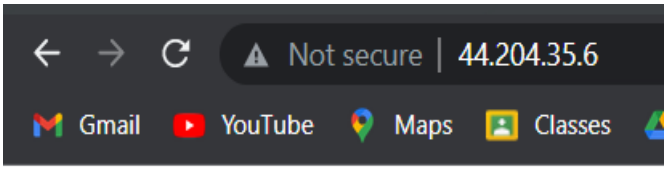
proxy_cache_bypass $http_upgrade;

}

6. Save and close the file.

```
#       location / {
                # First attempt to serve request as file, then
                # as directory, then fall back to displaying a 404.
#               try_files $uri $uri/ =404;
#       }
location / {
proxy_pass http://localhost:4000;
proxy_http_version 1.1;
proxy_set_header Upgrade $http_upgrade;
proxy_set_header Connection 'Upgrade';
proxy_set_header Host $host;
proxy_cache_bypass $http_upgrade;
}
```

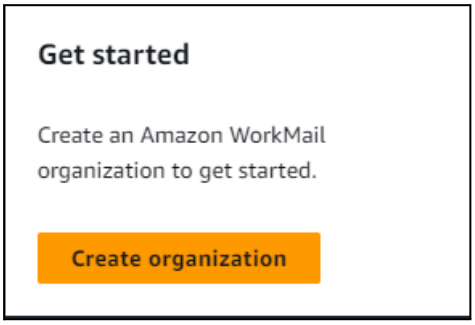
7. Type the command to restart nginx : **sudo systemctl restart nginx**
8. Run the project using IPv4 address only.



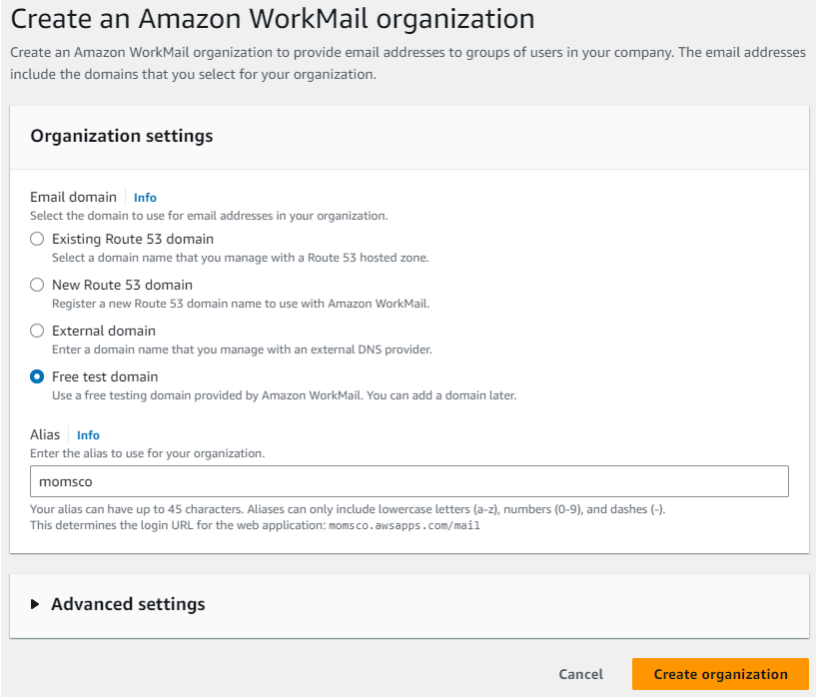
Hello from MCKVIE

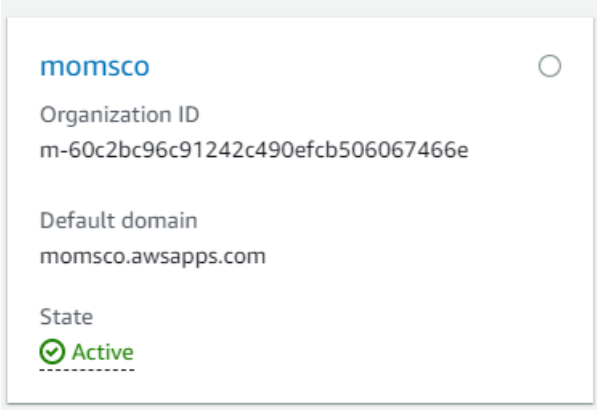
Assignment 13: Create a work-mail for your organization

1. Go to **Amazon Workmail** service and click on **Create Organization**.

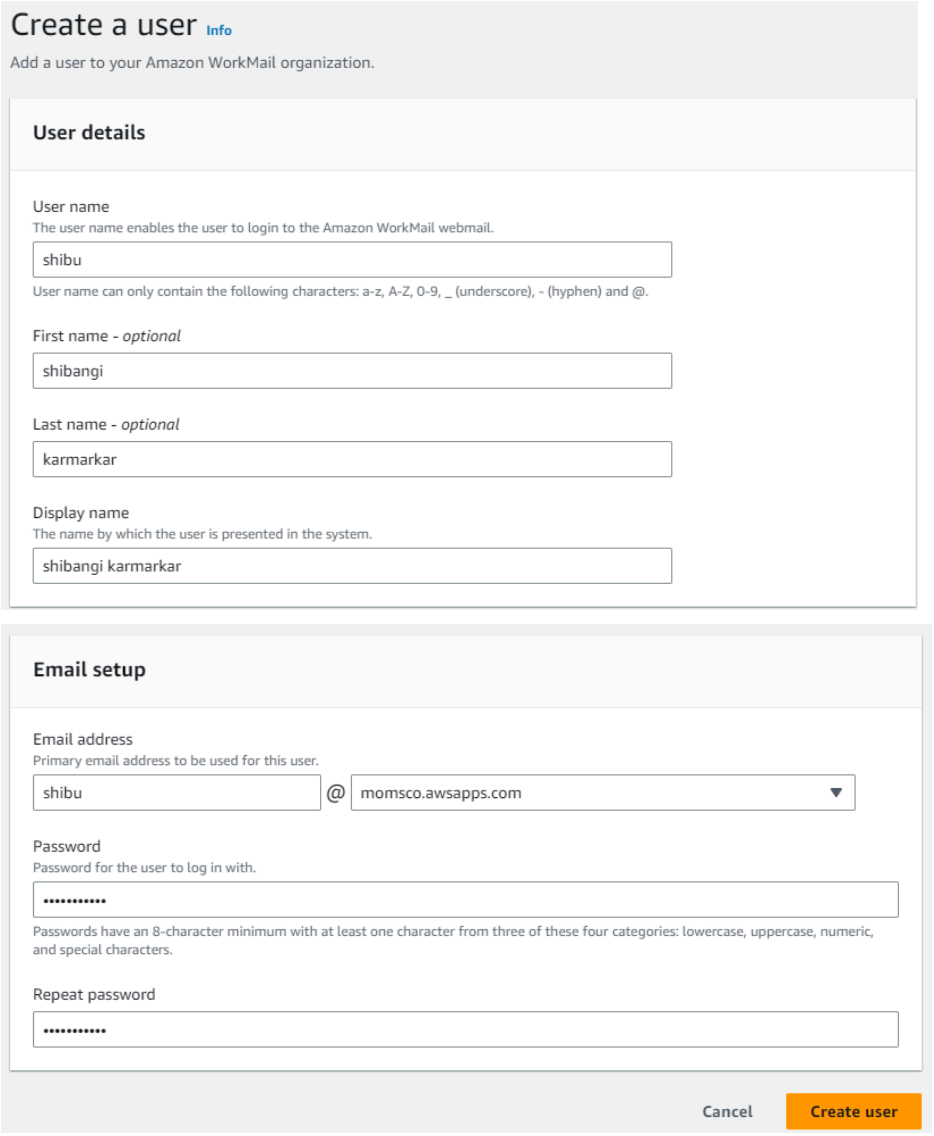


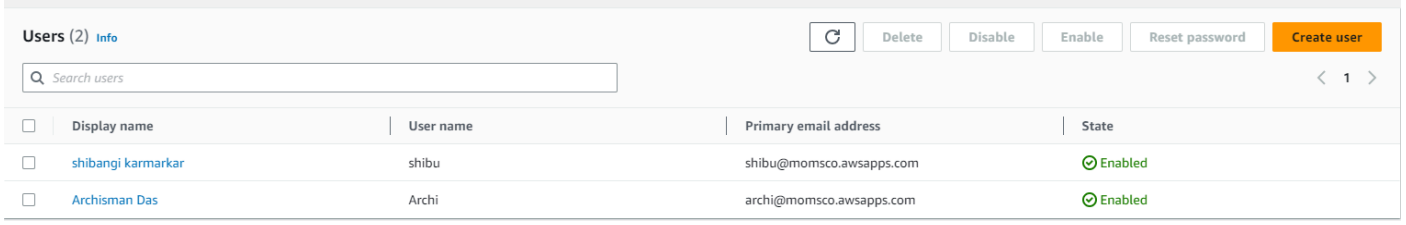
2. Select **Free test domain** and give an **alias** for your organization. Then click **Create Organization.**



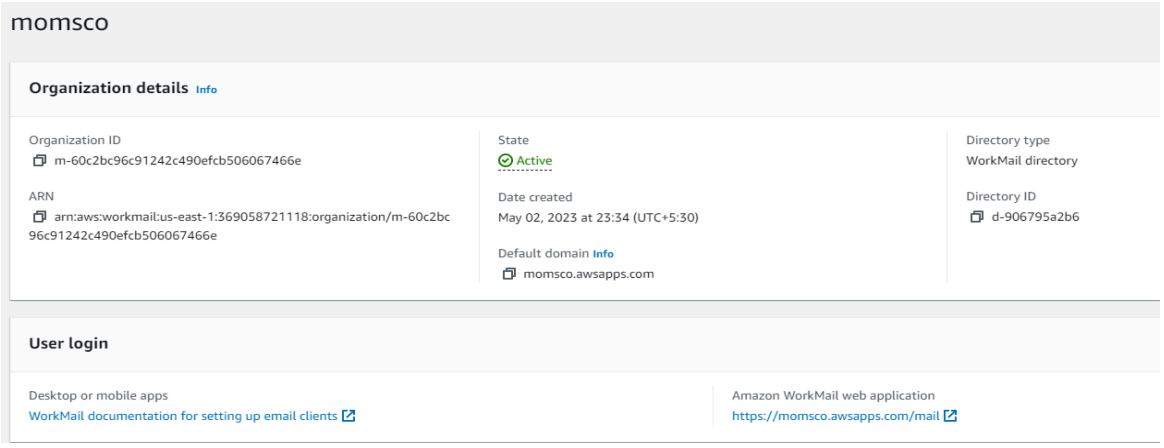


3. Go inside the organization and in the left panel go to Users and click on create user (create at least two users).

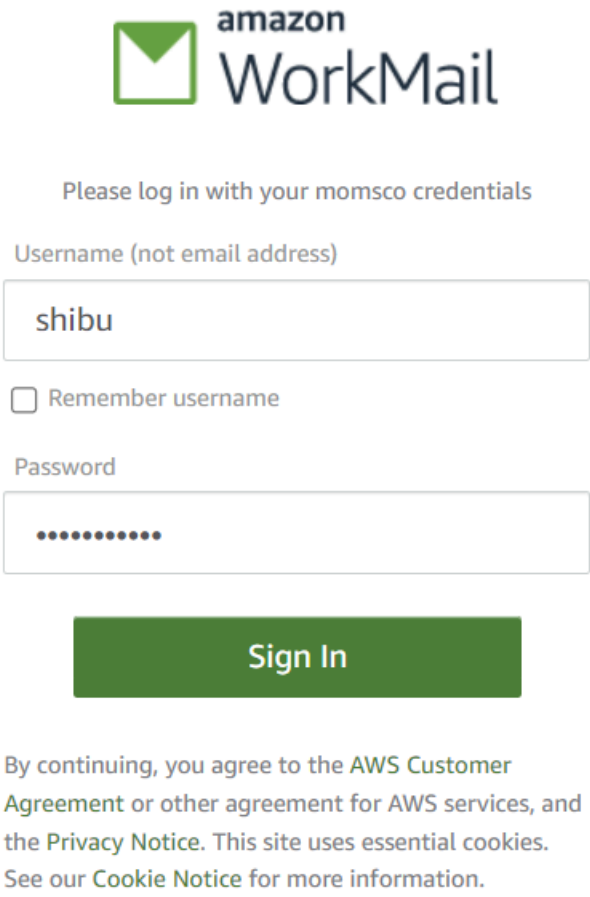




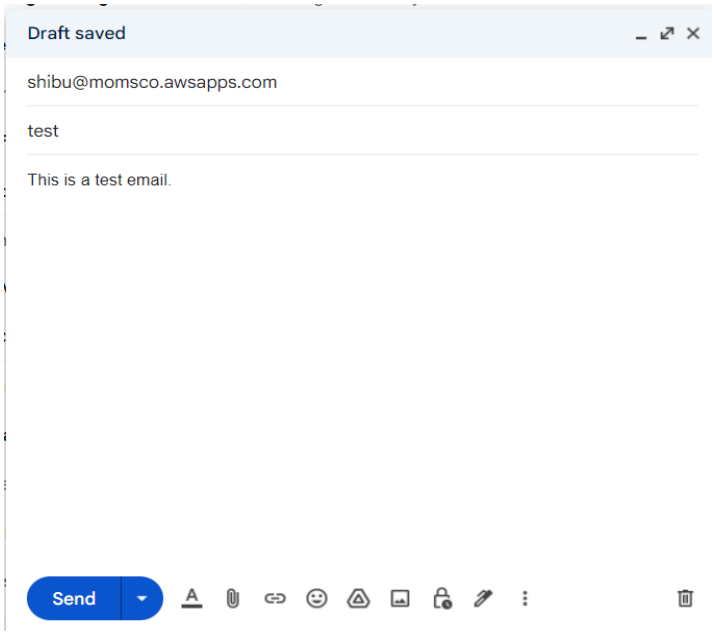
4. Then go back to the organization and open the Amazon WorkMail web application.

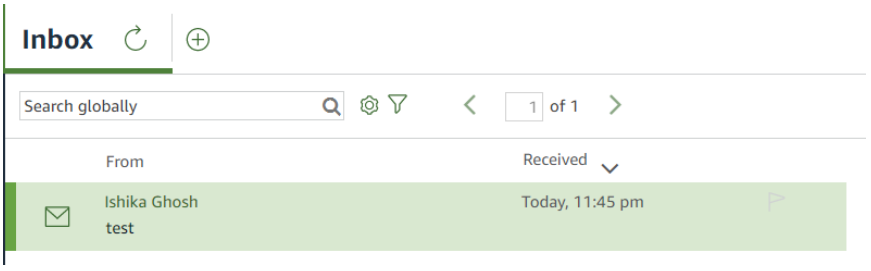


5. Authenticate with the username and password of the created user. And sign in.



6. Now from your gmail account send a mail to the created user's email of your organization. You will receive in the workmail


Draft saved
shibu@momsco.awsapps.com
test
This is a test email.
Send


Inbox
Search globally
1 of 1
From
Received
Ishika Ghosh
test
Today, 11:45 pm