Part 1: VPC Architecture Design

- **Task 1.1**

Create Custom VPC

Design and create a VPC with the following specifications:

VPC Name: student-vpc

CIDR Block: 10.0.0.0/16

Enable DNS hostnames: Yes

Enable DNS resolution: Yes

- **Task 1.2**

Create Subnets

Create the following subnets in different Availability Zones:

1) Public Subnet 1: 10.0.1.0/24 (AZ: us-east-1a)
2) Public Subnet 2: 10.0.2.0/24 (AZ: us-east-1b)
3) Private Subnet 1: 10.0.3.0/24 (AZ: us-east-1a)
4) Private Subnet 2: 10.0.4.0/24 (AZ: us-east-1b)
5)

- **Task 1.3**

Configure Internet Gateway

Create an Internet Gateway named student-igw

Attach it to your VPC

Update route table for public subnets to route 0.0.0.0/0 to IGW

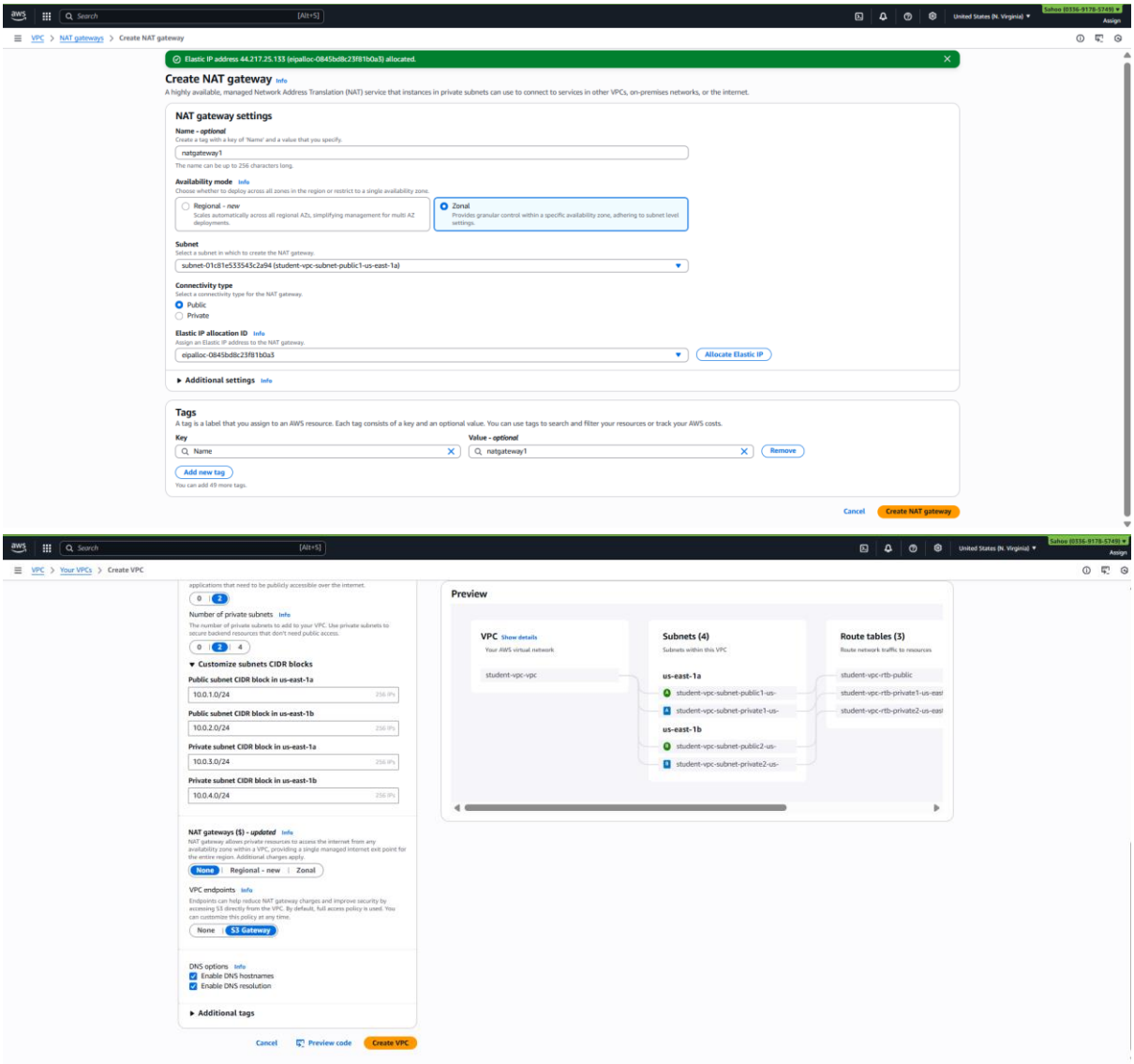
- **Task 1.4**

Configure NAT Gateway

Create NAT Gateway in Public Subnet 1

Allocate Elastic IP for NAT Gateway

Update private subnet route table to route 0.0.0.0/0 to NAT Gateway



Part 2: Security Configuration

- **Task 2.1**

Create Security Groups

ALB Security Group (alb-sg):

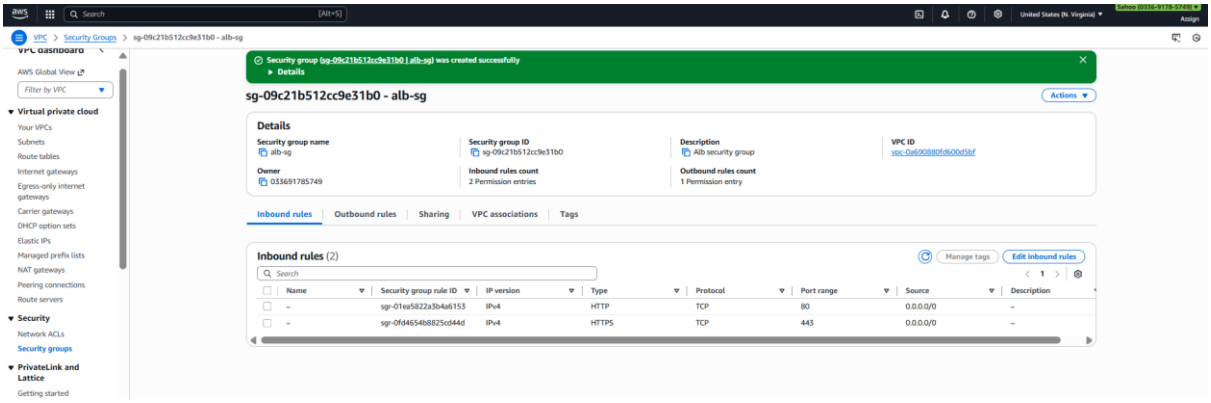
Inbound Rules:

- HTTP (80) from 0.0.0.0/0

- HTTPS (443) from 0.0.0.0/0

Outbound Rules:

- All traffic to 0.0.0.0/0



Web Server Security Group (web-sg):

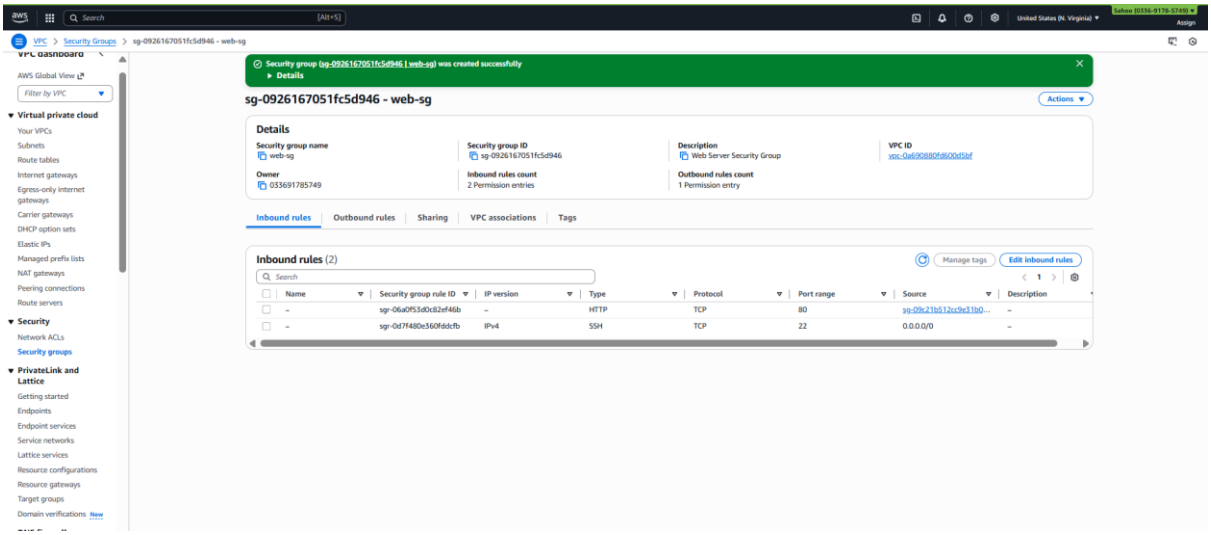
Inbound Rules:

- HTTP (80) from alb-sg

- SSH (22) from your IP

Outbound Rules:

- All traffic to 0.0.0.0/0



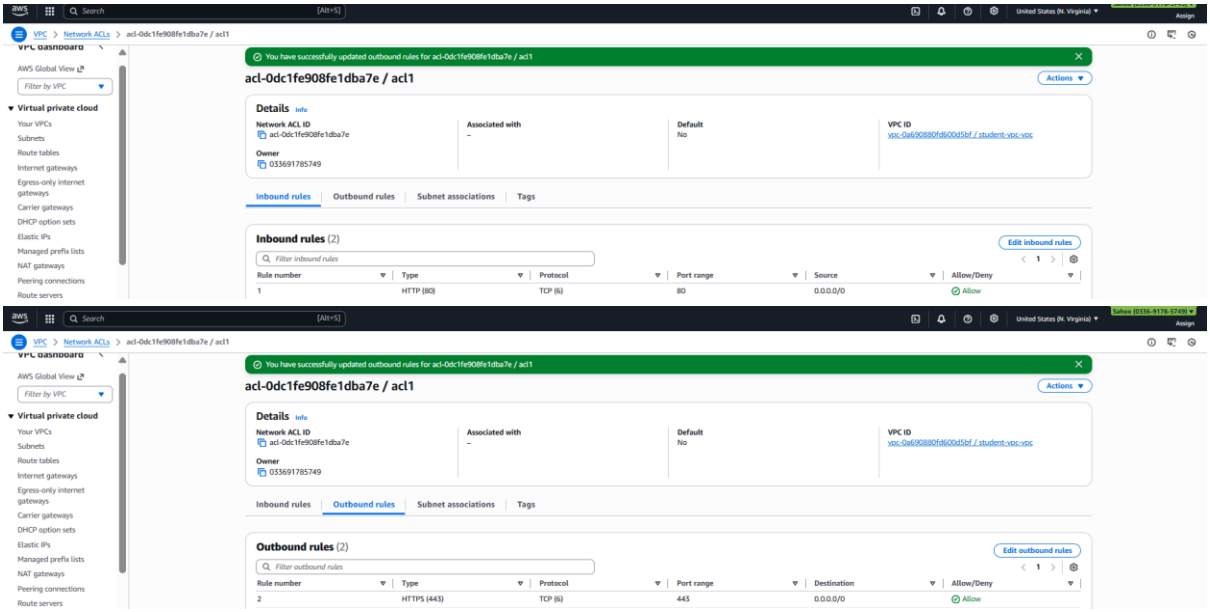
- **Task 2.2**

Configure Network ACLs

Create custom NACL for public subnets

Allow HTTP/HTTPS inbound and outbound

Allow ephemeral ports (1024-65535)



Part 3: Deploy Web Servers

- **Task 3.1**

  ## Launch EC2 Instances

  Launch 2 EC2 instances with the following configuration:

  AMI: Amazon Linux 2

  Instance Type: t2.micro

  Network: student-vpc

  Subnet: Private Subnet 1 and Private Subnet 2

  Security Group: web-sg

  Auto-assign Public IP: Disable

- **Task 3.2**

  ## Configure Web Server

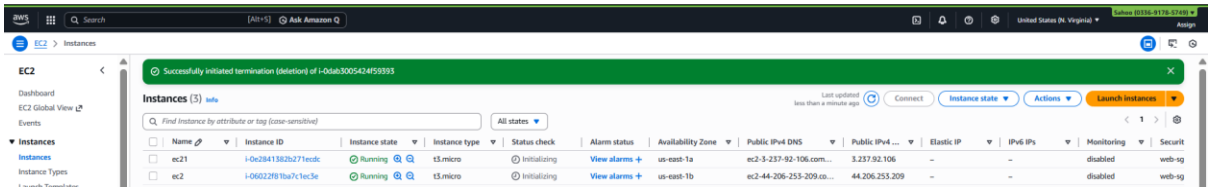
  Use the following User Data script to install Apache web server:

```
#!/bin/bash
yum update -y
yum install -y httpd
systemctl start httpd
systemctl enable httpd

# Create custom page
INSTANCE_ID=$(curl -s http://169.254.169.254/latest/meta-data/instance-id)
AZ=$(curl -s http://169.254.169.254/latest/meta-data/placement/availability-zone)

cat <<EOF > /var/www/html/index.html
<!DOCTYPE html>
<html>
```

<head><title>Student Project</title></head>
<body style="font-family: Arial; text-align: center; padding: 50px;">
<h1>VPC & ALB Project</h1>
<h2>Instance ID: $INSTANCE_ID</h2>
<h3>Availability Zone: $AZ</h3>
Student Name: [Your Name]
</body>
</html>
EOF



Part 4: Application Load Balancer Setup

- Task 4.1

Create Target Group

Name: student-tg
Target type: Instances
Protocol: HTTP
Port: 80
VPC: student-vpc
Health check path: /
Health check interval: 30 seconds
Healthy threshold: 2
Unhealthy threshold: 2

Register both EC2 instances to this target group.

- Task 4.2

Create Application Load Balancer

Name: student-alb
Scheme: Internet-facing
IP address type: IPv4
VPC: student-vpc
Subnets: Select both public subnets
Security Group: alb-sg

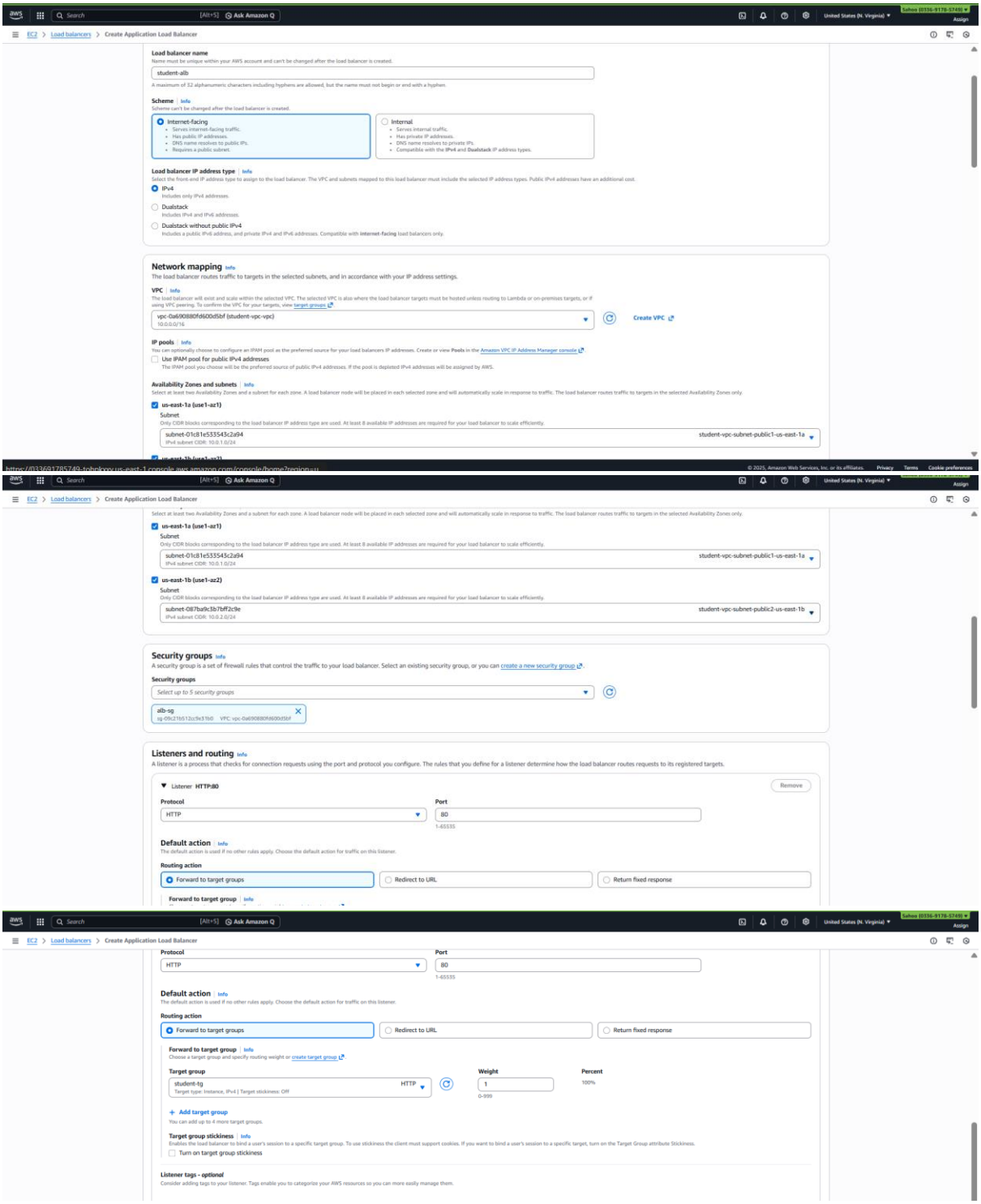
- Task 4.3

Configure Listener

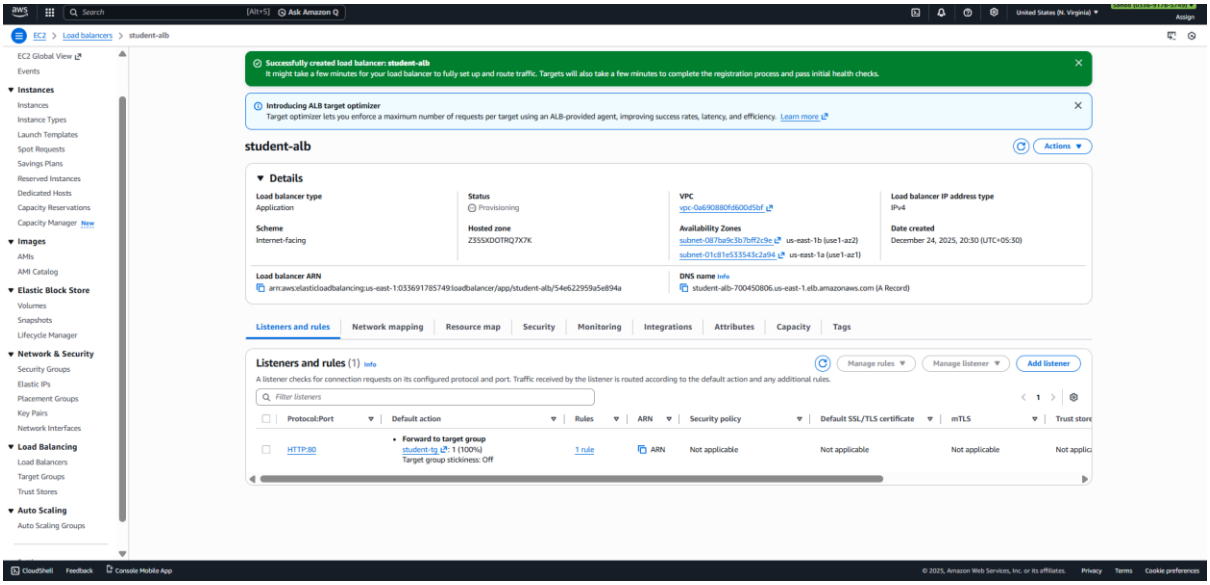
Protocol: HTTP

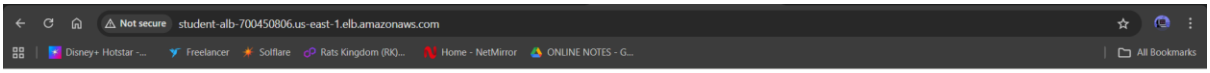
Port: 80

Default action: Forward to student-tg



Final output





VPC & ALB Project

Instance ID:

Availability Zone:

Student Name: [Your Name]