

Name: AbdelRahman Moustafa Mohamed Abdallah

Group name: AWS Cloud Specialist (ALX1_SW D8_M1e)

Student ID: 1110242719

AWS Academy Lab Project - Cloud Web Application Builder
Building a Highly Available, Scalable Web Application

Supervise by: Eng. Merhan Adel

Badge Link



About Project

Deploy a web app on EC2, migrate the database to RDS, and ensure high availability and scalability using a load balancer and an Auto Scaling group.

Steps for the Project:

Phase 1: Planning the Design and Estimating Cost

1. **Creating an Architectural Diagram:**
 - Design an architecture diagram illustrating the AWS services and their interactions.
2. **Developing a Cost Estimate:**
 - Use the AWS Pricing Calculator to estimate the cost of running the solution in the us-east-1 region for 12 months.

Phase 2: Creating a Basic Functional Web Application

1. **Creating a Virtual Network:**
 - Set up a virtual private cloud (VPC) and necessary subnets for hosting the application.
2. **Creating a Virtual Machine:**
 - Deploy a virtual machine (EC2) with the latest Ubuntu AMI to host the web application.
3. **Testing the Deployment:**
 - Ensure the web application is accessible from the internet and functions correctly (view, add, delete, modify records).

Phase 3: Decoupling the Application Components

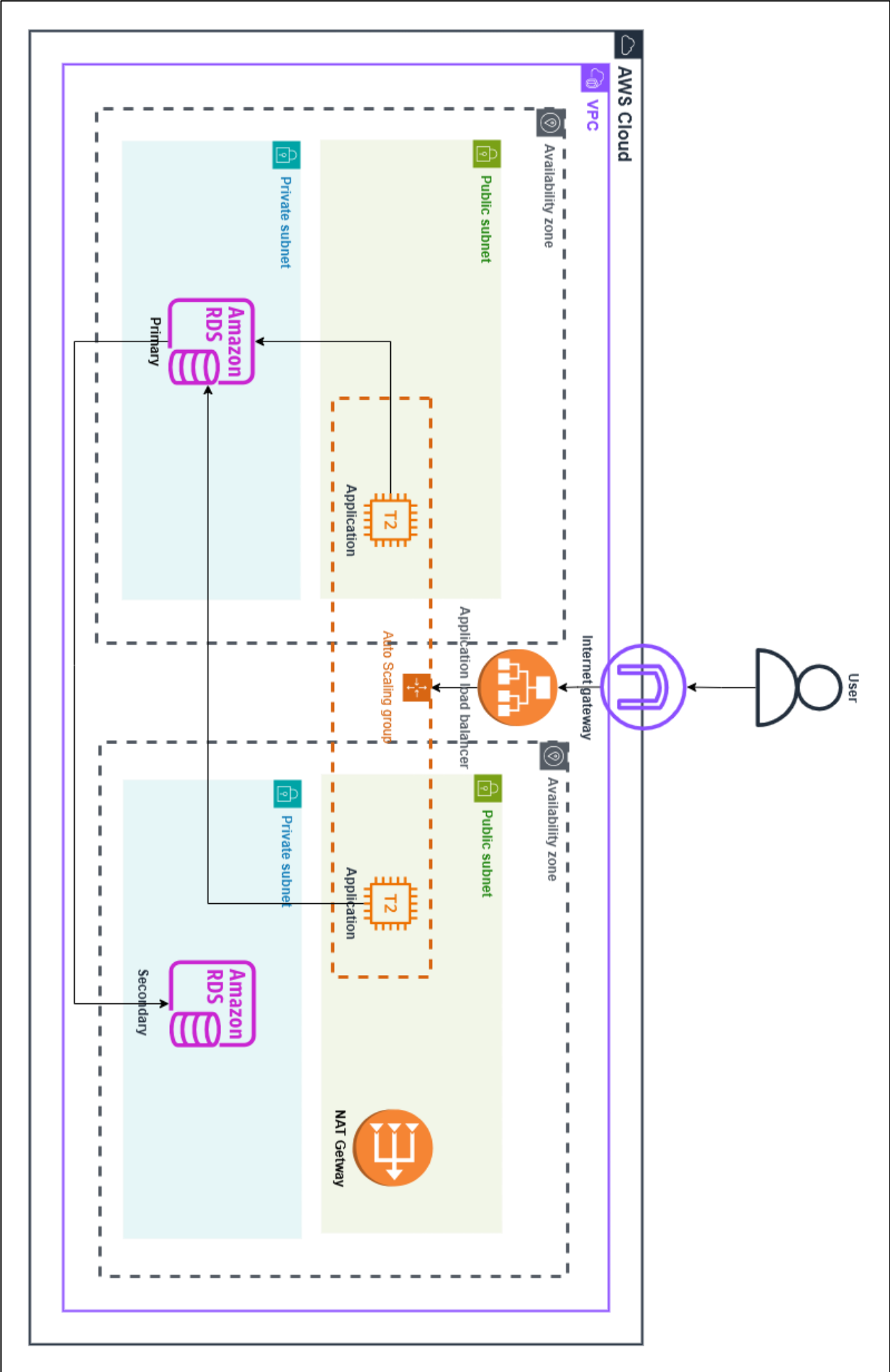
1. **Changing the VPC Configuration:**
 - Update or recreate the VPC to support separate hosting for the database and web server.
2. **Creating and Configuring the Amazon RDS Database:**
 - Set up an Amazon RDS instance running a MySQL engine, ensuring only the web application can access it.
3. **Configuring the Development Environment:**
 - Provision an AWS Cloud9 environment for executing AWS CLI commands.
4. **Provisioning Secrets Manager:**
 - Use AWS Secrets Manager to store database credentials securely and configure the web application to use these credentials.
5. **Provisioning a New Instance for the Web Server:**
 - Deploy a new EC2 instance for the web application or reconfigure the existing instance to connect to Amazon RDS.
6. **Migrating the Database:**
 - Migrate data from the original database on the EC2 instance to the new Amazon RDS database.
7. **Testing the Application:**
 - Verify the application functionality by performing CRUD operations on student records.

Phase 4: Implementing High Availability and Scalability

1. **Creating an Application Load Balancer:**
 - Set up an Application Load Balancer across at least two Availability Zones.
2. **Implementing Amazon EC2 Auto Scaling:**
 - Create a launch template and an Auto Scaling group to manage EC2 instances hosting the web application.
3. **Accessing the Application:**
 - Test application functionality by performing CRUD operations again.
4. **Load Testing the Application:**
 - Conduct a load test using scripts to simulate 1,000 requests to evaluate performance under load.

Phase 1: Planning the design and estimating cost

Task 1: Creating an architectural diagram [Link Diagram](#)



Task 2: Developing a cost estimate [Link CSV File](#)

1. Application Load Balancer (ALB) :

- Monthly Cost: \$16.46
- Annual Cost: \$197.52
- Quantity: 1

2. Amazon EC2 :

- Monthly Cost: \$40.16
- Annual Cost: \$481.92
- Configuration: 4 t2.micro instances running on Linux, utilizing 700 hours per month, with 20 GB of EBS storage.

3. Amazon RDS for MySQL :

- Monthly Cost: \$14.20
- Annual Cost: \$170.40
- Configuration: 1 db.t3.micro instance with 20 GB of gp3 SSD storage, deployed in a single availability zone.

The Total Cost:

Monthly Payments:

- Total Monthly Cost: \$70.82 USD
You will pay \$70.82 each month for the services provided (Application Load Balancer, EC2 instances, and RDS for MySQL).

Yearly Payments:

- Total Yearly Cost: \$849.84 USD
If you choose to pay annually, the total amount due for the 12 months will be \$849.84 USD.

This setup assumes no upfront payments and is based on on-demand pricing.

Phase 2: Creating a basic functional web application

Task 1: Creating a virtual network

The screenshot shows the AWS VPC dashboard. On the left, the 'VPC dashboard' sidebar is visible with a search bar and a list of resources. The main content area displays 'Your VPCs (1/2)' with a table listing VPCs. Below the table, the details for 'vpc-08f782e6ac7591678 / WebApp-vpc' are shown, including tabs for Details, Resource map, CIDRs, Flow logs, Tags, and Integrations. The 'Details' tab is active, showing various attributes of the VPC.

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP option set	Main route table
WebApp-vpc	vpc-08f782e6ac7591678	Available	10.0.0.0/16	-	dopt-0ce84d1bc2a3b92f4	rtb-022014f59b59054f1
-	vpc-0c1c02e6bcdfa44b9	Available	172.31.0.0/16	-	dopt-0ce84d1bc2a3b92f4	rtb-0375bc5d312ef8dde

vpc-08f782e6ac7591678 / WebApp-vpc

Details

VPC ID	vpc-08f782e6ac7591678	State	Available	DNS hostnames	Enabled	DNS resolution	Enabled
Tenancy	Default	DHCP option set	dopt-0ce84d1bc2a3b92f4	Main route table	rtb-022014f59b59054f1	Main network ACL	acl-04d13a25618b7c82c
Default VPC	No	IPv4 CIDR	10.0.0.0/16	IPv6 pool	-	IPv6 CIDR (Network border group)	-
Network Address Usage metrics	Disabled	Route 53 Resolver DNS Firewall rule groups	-	Owner ID	207526410461		

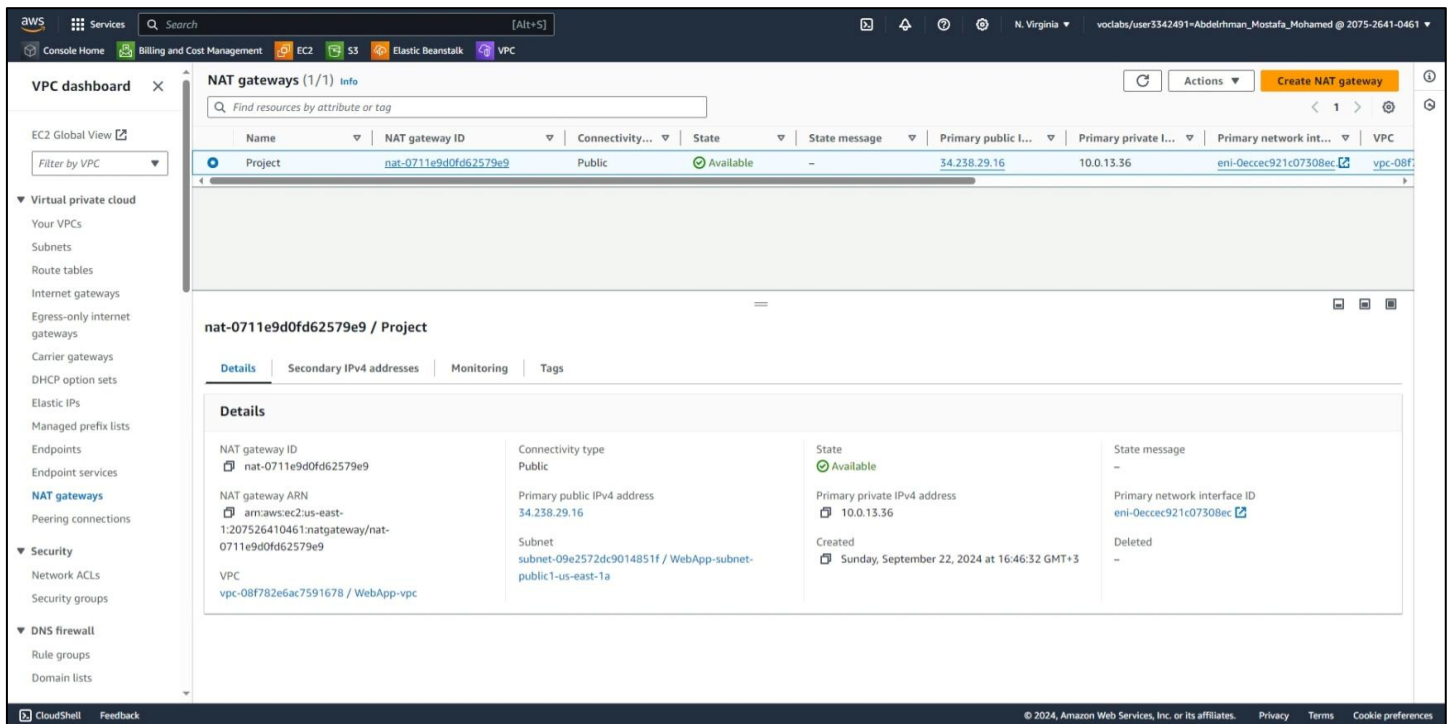
The screenshot shows the AWS VPC dashboard with the 'Internet gateways' section selected. The main content area displays 'Internet gateways (1/2)' with a table listing internet gateways. Below the table, the details for 'igw-04b85407542376b6d / WebApp-igw' are shown, including tabs for Details and Tags. The 'Details' tab is active, showing various attributes of the internet gateway.

Name	Internet gateway ID	State	VPC ID	Owner
WebApp-igw	igw-04b85407542376b6d	Attached	vpc-08f782e6ac7591678 WebApp-vpc	207526410461
-	igw-0b74313da4cfbcc15	Attached	vpc-0c1c02e6bcdfa44b9	207526410461

igw-04b85407542376b6d / WebApp-igw

Details

Internet gateway ID	igw-04b85407542376b6d	State	Attached	VPC ID	vpc-08f782e6ac7591678 WebApp-vpc	Owner	207526410461
---------------------	-----------------------	-------	----------	--------	------------------------------------	-------	--------------



Security Groups [File Link](#)

1) EC2-SG (Security Group for EC2 Instances):

- **Description:** This security group is used for EC2 instances, allowing inbound SSH (port 22) and HTTP (port 80) access. It is ideal for controlling remote access and web traffic to the EC2 instances.
- **Inbound Rules:** 3 rules, likely including SSH, HTTP, and another service-specific rule.
- **Outbound Rules:** 1 rule, usually allowing all outbound traffic.

2) DB-SG (Database Security Group):

- **Description:** This security group is designed for database instances, allowing inbound MySQL traffic on port 3306, ensuring secure database access.
- **Inbound Rules:** 1 rule for MySQL (port 3306) access.
- **Outbound Rules:** 1 rule, likely allowing all outbound traffic.

3) ELB-SG (Security Group for Elastic Load Balancer):

- **Description:** This security group is for the Elastic Load Balancer (ELB), allowing inbound traffic for load-balanced services such as HTTP and HTTPS.
- **Inbound Rules:** 2 rules, probably for HTTP (port 80) and HTTPS (port 443) traffic.
- **Outbound Rules:** 1 rule, typically allowing all outbound traffic.

Task 2: Creating a virtual machine

EC2 Dashboard

EC2 Global View

Events

Console-to-Code [Preview](#)

▼ Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity

Reservations [New](#)

▼ Images

AMIs

AMI Catalog

▼ Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

▼ Network & Security

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Services

Search

[Alt+S]

EC2

S3

Elastic Beanstalk

VPC

Instances (1/4) info

Find Instance by attribute or tag (case-sensitive)

Running

Connect

Instance state

Actions

Launch instances

webapp

WebApp

i-01b3b2ac22e908017

i-0dd9ccfc4bafcf99e

i-02ce152545b5cae60

i-0a1570bc340f1deec

Running

Running

Running

t2.micro

t2.micro

t2.micro

2/2 checks passed

2/2 checks passed

2/2 checks passed

View alarms

View alarms

View alarms

us-east-1a

us-east-1a

us-east-1b

us-east-1a

ec2-3-84-187-185.com...

ec2-3-82-44-43.comput...

ec2-54-85-108-152.co...

ec2-34-224-80-209.co...

3.84.187.185

3.82.44.43

54.85.108.152

34.224.80.209

–

–

–

i-0a1570bc340f1deec

Instance ID

i-0a1570bc340f1deec

IPv6 address

–

Hostname type

IP name: ip-10-0-15-235.ec2.internal

Answer private resource DNS name

–

Auto-assigned IP address

34.224.80.209 [Public IP]

IAM Role

–

IMDSv2

Optional

EC2 recommends setting IMDSv2 to required | [Learn more](#)

▼ Instance details

Info

Public IPv4 address

34.224.80.209 | [open address](#)

Instance state

Running

Private IP DNS name (IPv4 only)

ip-10-0-15-235.ec2.internal

Instance type

t2.micro

VPC ID

vpc-08f782e6ac7591678 (WebApp-vpc) | [open address](#)

Subnet ID

subnet-09e2572dc9014851f (WebApp-subnet-public1-us-east-1a) | [open address](#)

Instance ARN

arn:aws:ec2:us-east-1:207526410461:instance/i-0a1570bc340f1deec

Private IPv4 addresses

10.0.15.235

Public IPv4 DNS

ec2-34-224-80-209.compute-1.amazonaws.com | [open address](#)

Elastic IP addresses

–

AWS Compute Optimizer finding

Opt-in to AWS Compute Optimizer for recommendations. | [Learn more](#)

Auto Scaling Group name

WebApp-ASG

Task 3: Testing the deployment

Students

Not secure | 3.84.187.185

XYZ University

Home

Students list

Welcome

Use this app to keep track of your student inquiries

[List of students](#)

Phase 3: Decoupling the application components

Task 1: Changing the VPC configuration

Note: You need private subnets in a minimum of two Availability Zones.

The screenshot shows the AWS VPC console. On the left, the 'VPC dashboard' sidebar is visible with categories like 'Virtual private cloud', 'Security', 'DNS firewall', and 'Network Firewall'. The main content area displays a list of subnets under the heading 'Subnets (1/4) Info'. The list includes columns for Name, Subnet ID, State, VPC, IPv4 CIDR, IPv6 CIDR, and IPv6 CIDR association. Below the list, the details for 'subnet-05b5421eef52d0de3 / WebApp-subnet-private1-us-east-1a' are shown, including tabs for Details, Flow logs, Route table, Network ACL, CIDR reservations, Sharing, and Tags. The 'Details' tab is active, showing various attributes like Subnet ID, Subnet ARN, State, IPv4 CIDR, Availability Zone, Route table, and more.

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR	IPv6 CIDR association
WebApp-subnet-private1-us-east-1a	subnet-05b5421eef52d0de3	Available	vpc-08f782e6ac7591678 Web...	10.0.128.0/20	-	-
WebApp-subnet-public1-us-east-1a	subnet-09e2572dc9014851f	Available	vpc-08f782e6ac7591678 Web...	10.0.0.0/20	-	-
WebApp-subnet-private2-us-east-1b	subnet-0f3c791425e2afc1c	Available	vpc-08f782e6ac7591678 Web...	10.0.144.0/20	-	-
WebApp-subnet-public2-us-east-1b	subnet-0b13efdfdb607f84	Available	vpc-08f782e6ac7591678 Web...	10.0.16.0/20	-	-

subnet-05b5421eef52d0de3 / WebApp-subnet-private1-us-east-1a

Details

- Subnet ID: subnet-05b5421eef52d0de3
- Subnet ARN: arn:aws:ec2:us-east-1:207526410461:subnet/subnet-05b5421eef52d0de3
- State: Available
- IPv4 CIDR: 10.0.128.0/20
- Availability Zone: us-east-1a
- Route table: rtb-01abee6036ac519d | WebApp-rtb-private1-us-east-1a
- Auto-assign IPv4 address: No
- Outpost ID: -
- IP name: -

The screenshot shows the AWS VPC console. On the left, the 'VPC dashboard' sidebar is visible. The main content area displays a list of VPCs under the heading 'Your VPCs (1/1) Info'. The list includes columns for Name, VPC ID, State, IPv4 CIDR, IPv6 CIDR, DHCP option set, and Main route table. Below the list, the details for 'vpc-08f782e6ac7591678 / WebApp-vpc' are shown, including tabs for Details, Resource map, CIDRs, Flow logs, Tags, and Integrations. The 'Resource map' tab is active, showing a diagram of the VPC resources and their connections.

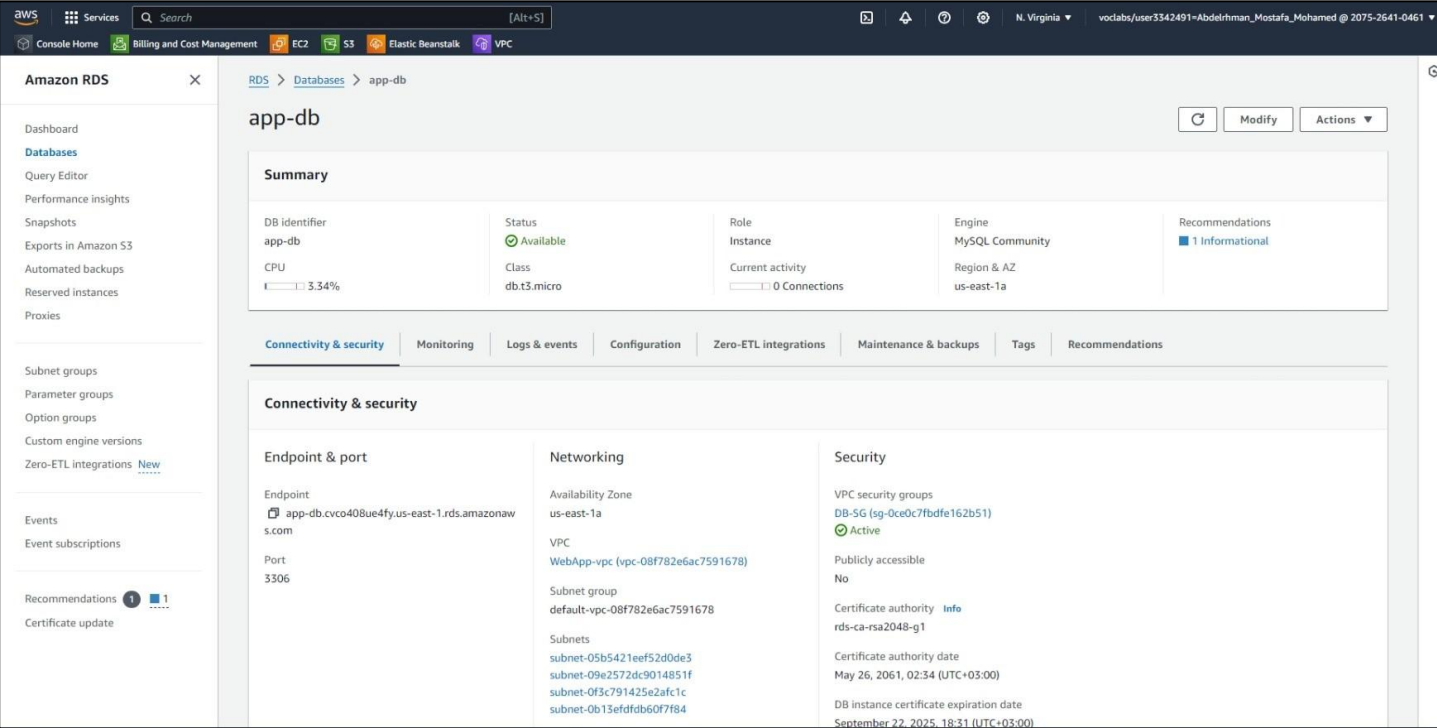
Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP option set	Main route table
WebApp-vpc	vpc-08f782e6ac7591678	Available	10.0.0.0/16	-	dopt-0ce84d1bc2a3b92f4	rtb-022014f59b59054f1

vpc-08f782e6ac7591678 / WebApp-vpc

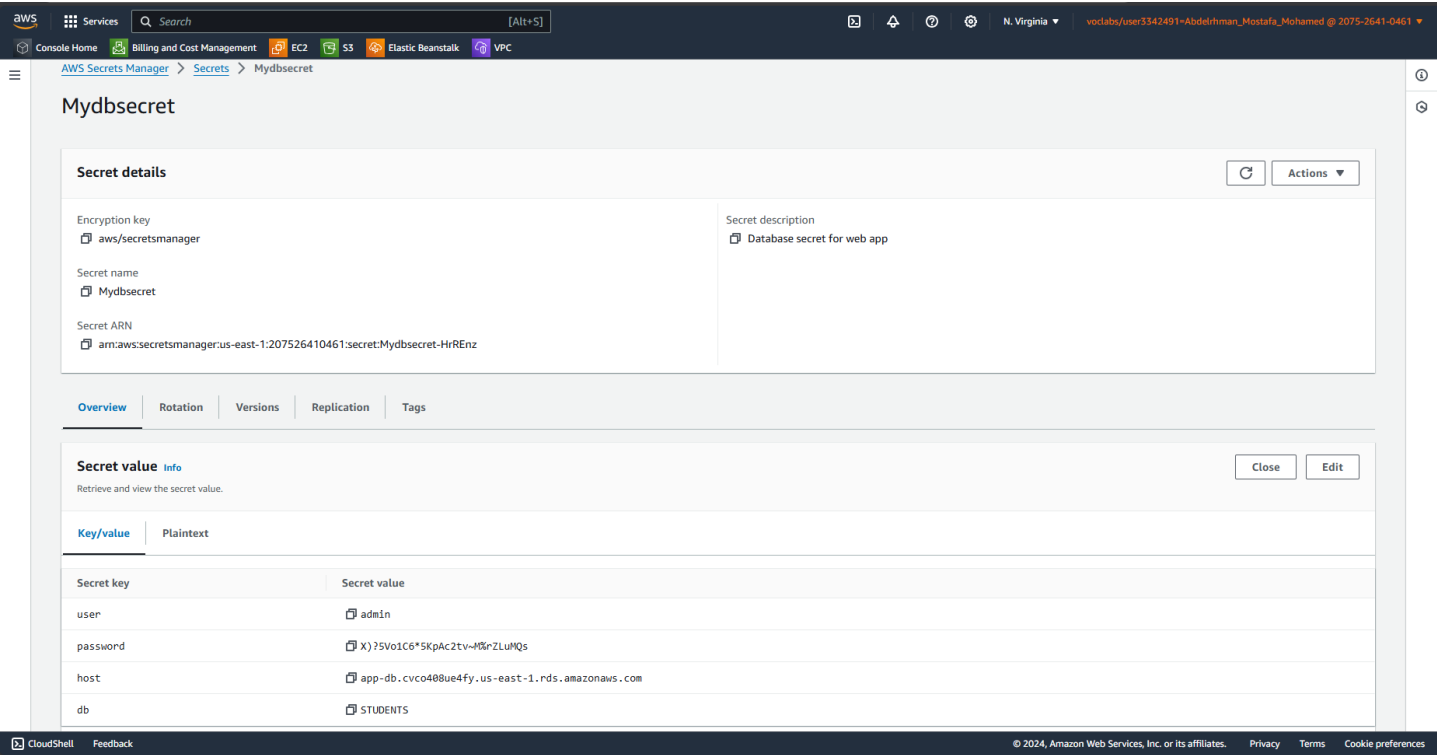
Resource map

- VPC: WebApp-vpc
- Subnets (4):
 - us-east-1a: WebApp-subnet-public1-us-east-1a, WebApp-subnet-private1-us-east-1a
 - us-east-1b: WebApp-subnet-public2-us-east-1b, WebApp-subnet-private2-us-east-1b
- Route tables (4): WebApp-rtb-public, WebApp-rtb-private1-us-east-1a, rtb-022014f59b59054f1, WebApp-rtb-private2-us-east-1b
- Network connections (2): WebApp-igw, Project

Task 2: Creating and configuring the Amazon RDS database



Task 4: Provisioning Secrets Manager



Task 5: Provisioning a new instance for the web server

EC2 Dashboard

EC2 Global View

Events

Console-to-Code [Preview](#)

▼ Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity

Reservations [New](#)

▼ Images

AMIs

AMI Catalog

▼ Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

▼ Network & Security

Security Groups

Elastic IPs

Placement Groups

Key Pairs

Network Interfaces

Instances (1/4) Info

Find Instance by attribute or tag (case-sensitive)

All states

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic I
webapp	i-01b3b2ac22e908017	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2-54-225-38-160.co...	54.225.38.160	-
WebApp	i-0dd9ccfc4bafc99e	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2-54-196-138-174.co...	54.196.138.174	-
	i-073beef500327f761	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1b	ec2-107-22-1-137.com...	107.22.1.137	-

i-0dd9ccfc4bafc99e (WebApp)

Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

▼ Instance summary Info

Instance ID

i-0dd9ccfc4bafc99e (WebApp)

IPv6 address

-

Hostname type

IP name: ip-10-0-7-226.ec2.internal

Answer private resource DNS name

-

Auto-assigned IP address

54.196.138.174 [Public IP]

IAM Role

LabRole

IMDSv2

Required

▼ Instance details Info

Public IPv4 address

54.196.138.174 | [open address](#)

Instance state

Running

Private IP DNS name (IPv4 only)

ip-10-0-7-226.ec2.internal

Instance type

t2.micro

VPC ID

vpc-08f782e6ac7591678 (WebApp-vpc) | [open address](#)

Subnet ID

subnet-09e2572dc9014851f (WebApp-subnet-public-1-us-east-1a) | [open address](#)

Instance ARN

arn:aws:ec2:us-east-1:207526410461:instance/i-0dd9ccfc4bafc99e

Private IPv4 addresses

10.0.7.226

Public IPv4 DNS

ec2-54-196-138-174.compute-1.amazonaws.com | [open address](#)

Elastic IP addresses

-

AWS Compute Optimizer finding

[Opt-in to AWS Compute Optimizer for recommendations.](#) | [Learn more](#)

Auto Scaling Group name

-

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie preferences

Task 6: Migrating the database

```
54.221.137.124 (ubuntu) (1)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help

Quick connect...
Name Size (KB)
.cache
.ssh
aws
resources
.bash_history 1
.bash_logout 1
.bashrc 3
.profile 1
sudo_get_admin_successful 0
.Yaauthority 1
awscli2.zip 64.445
code.zip 397
code.zip.1 1.369

ubuntu@ip-10-0-7-14:~$ mysqldump -h 10.0.7.14 -u nodeapp -p --databases STUDENTS > data.sql
Enter password:
ubuntu@ip-10-0-7-14:~$ ls
aws awscli2.zip code.zip code.zip.1 data.sql resources
ubuntu@ip-10-0-7-14:~$ mysql -h app-db.cvco408ue4fy.us-east-1.rds.amazonaws.com -u admin -p < data.sql
Enter password:
mysql -h app-db.cvco408ue4fy.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 1680
Server version: 8.0.35 Source distribution

Copyright (c) 2000, 2024, Oracle and/or its affiliates.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.


mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| STUDENTS |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql> USE STUDENTS;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> SHOW TABLES;
+-----+
| Tables_in_STUDENTS |
+-----+
| students |
+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM students;
+-----+
| id | name | address | city | state | email | phone |
+-----+
| 1 | AbdelRahman | test | test | sss | abdulrahamanmoustafa2002@outlook.com | 1234567890 |
+-----+
1 row in set (0.00 sec)

mysql>
```



XYZ University

Home

Students list


All students

Name	Address	City	State	Email	Phone
AbdelRahman	test	test	sss	abdulrahamanmoustafa2002@outlook.com	1234567890

edit

Add a new student

Task 7: Testing the application



XYZ University

[Home](#)
[Students list](#)

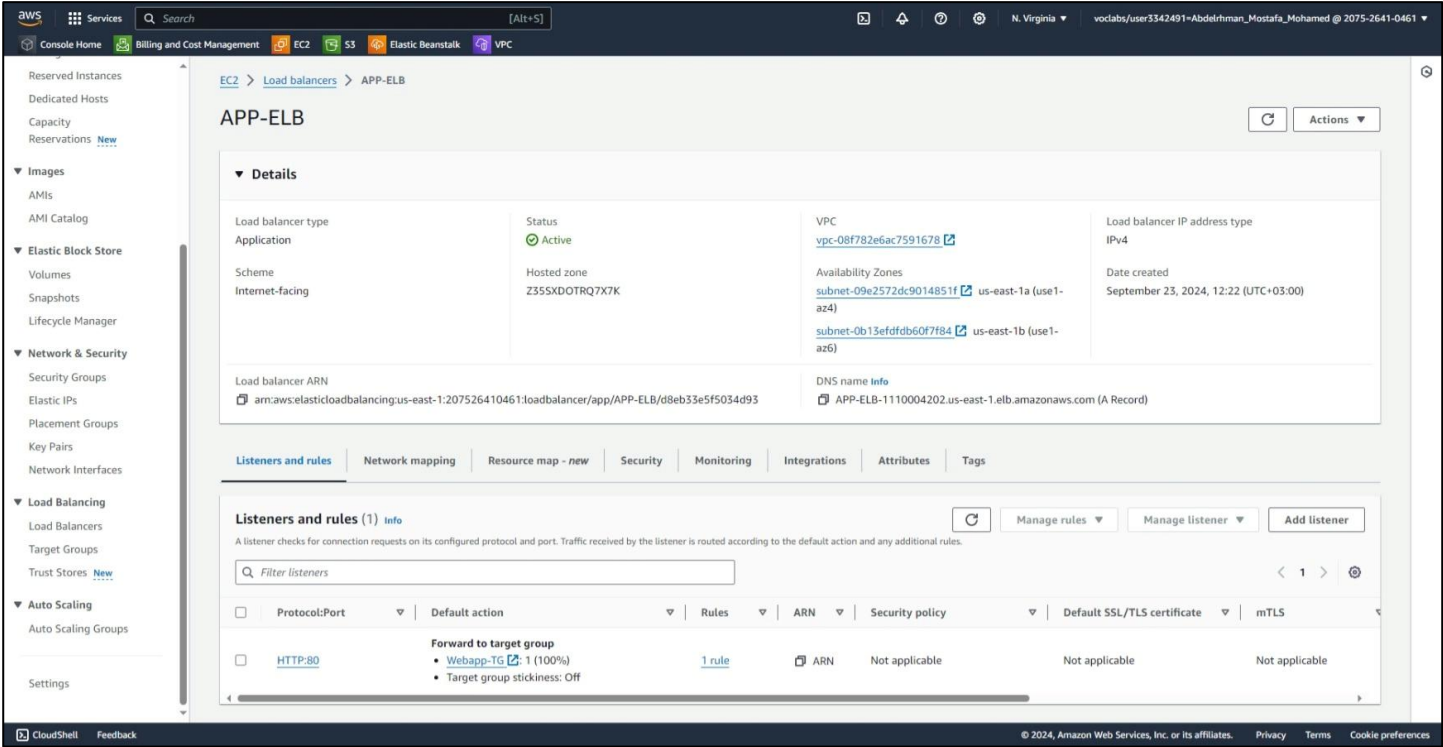
All students

Name	Address	City	State	Email	Phone	
AbdelRahman	test	test	sss	abdulrahanmoustafa2002@outlook.com	1234567890	edit
عبدالرحمن مصطفى محمد	test	Alexandria	sss	abdulrahanmoustafa2002@gmail.com	01121435744	edit

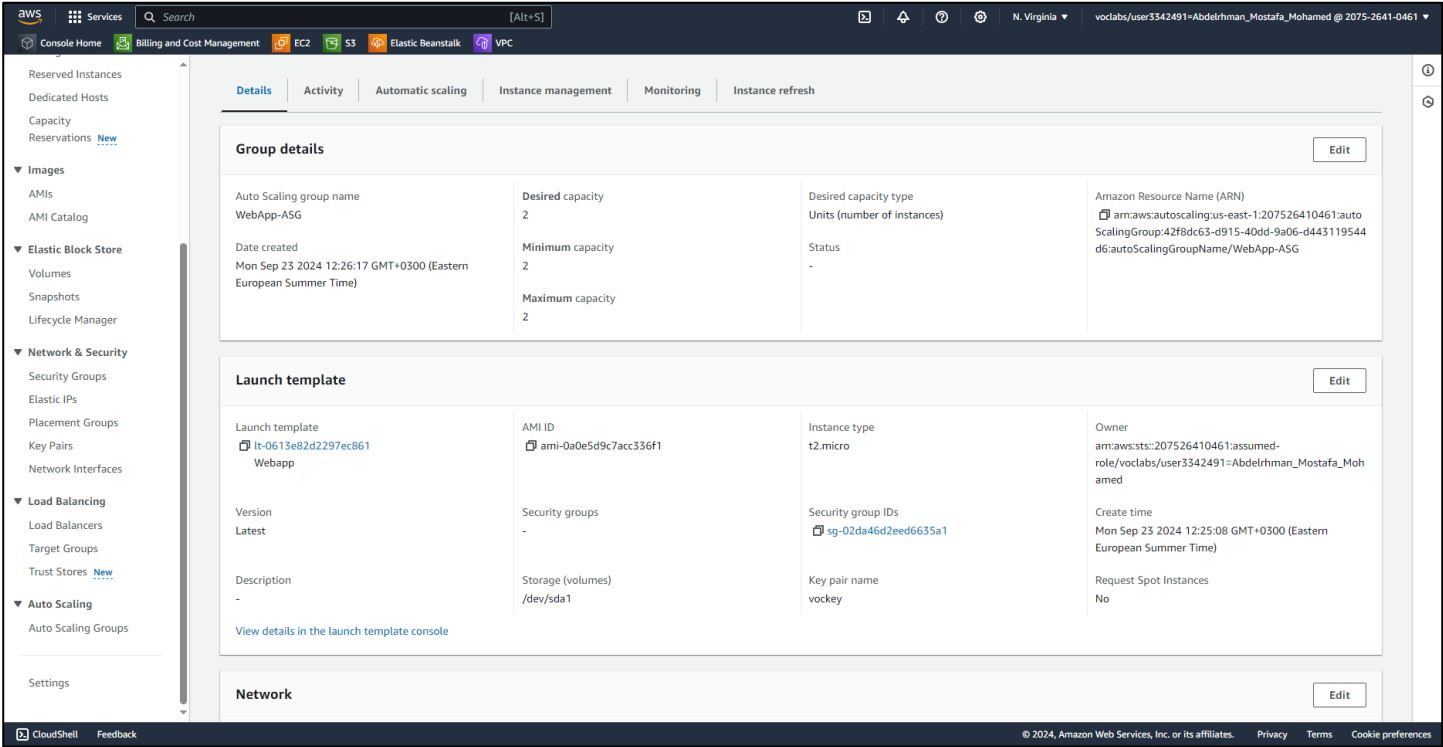
Add a new student

Phase 4: Implementing high availability and scalability

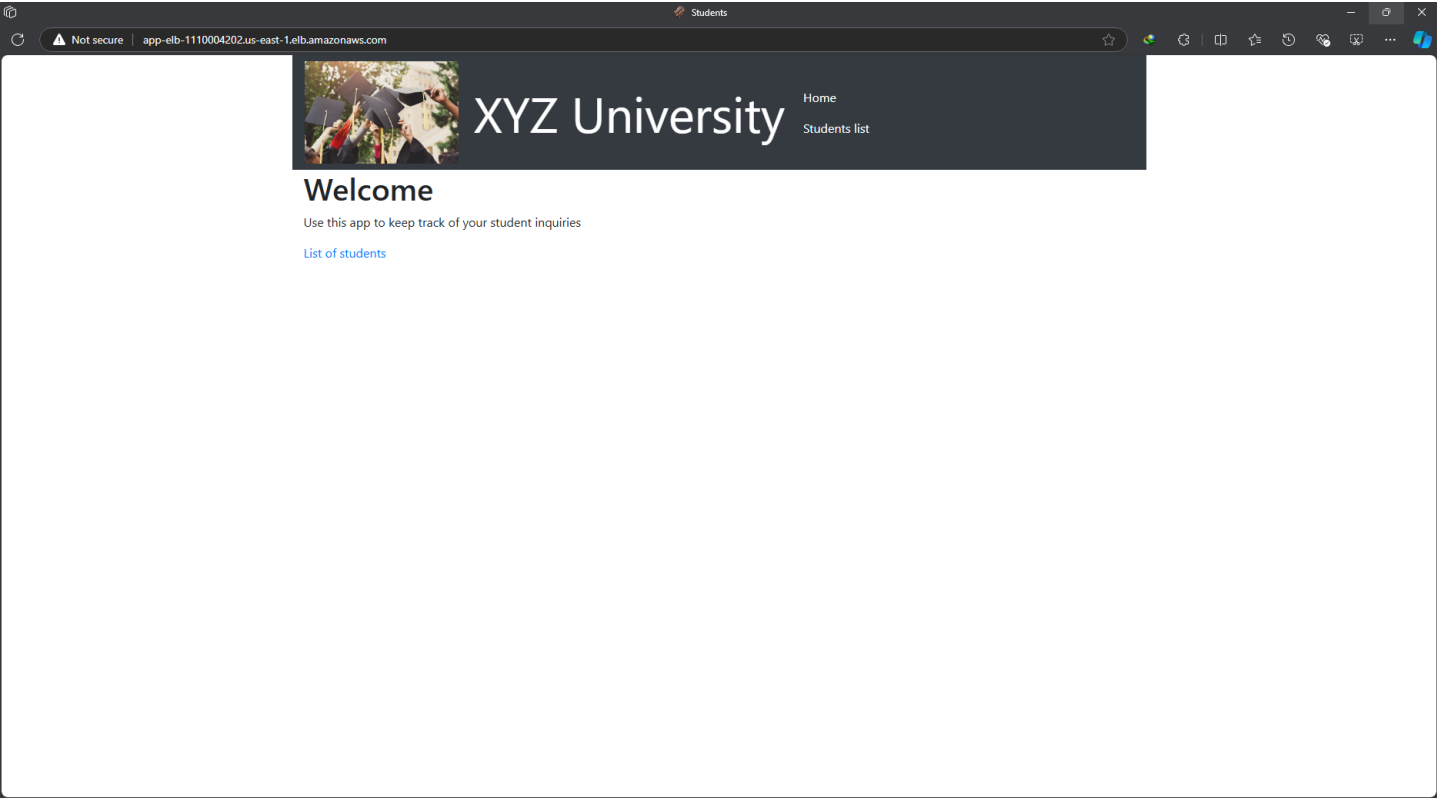
Task 1: Creating an Application Load Balancer



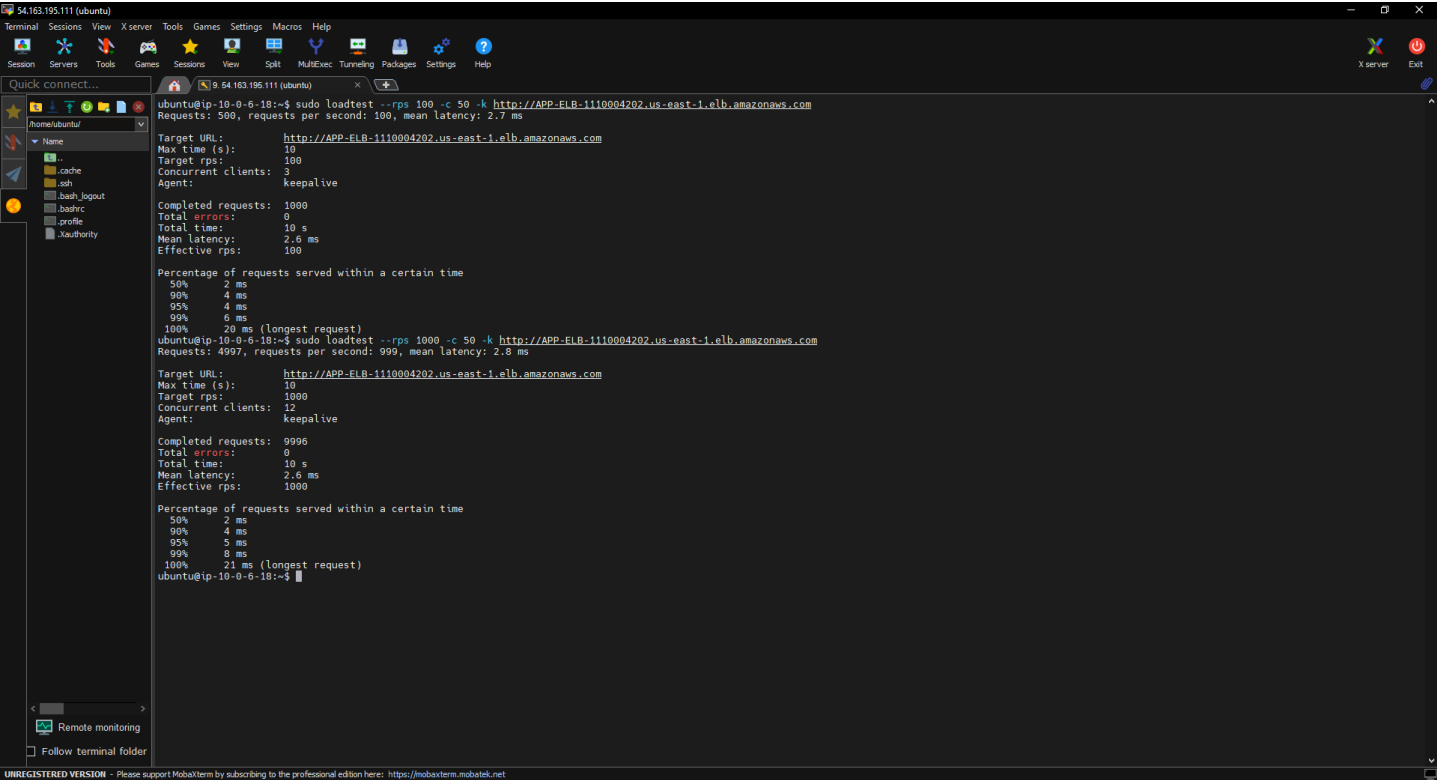
Task 2: Implementing Amazon EC2 Auto Scaling



Task 3: Accessing the application



Task 4: Load testing the application



Badge Link

