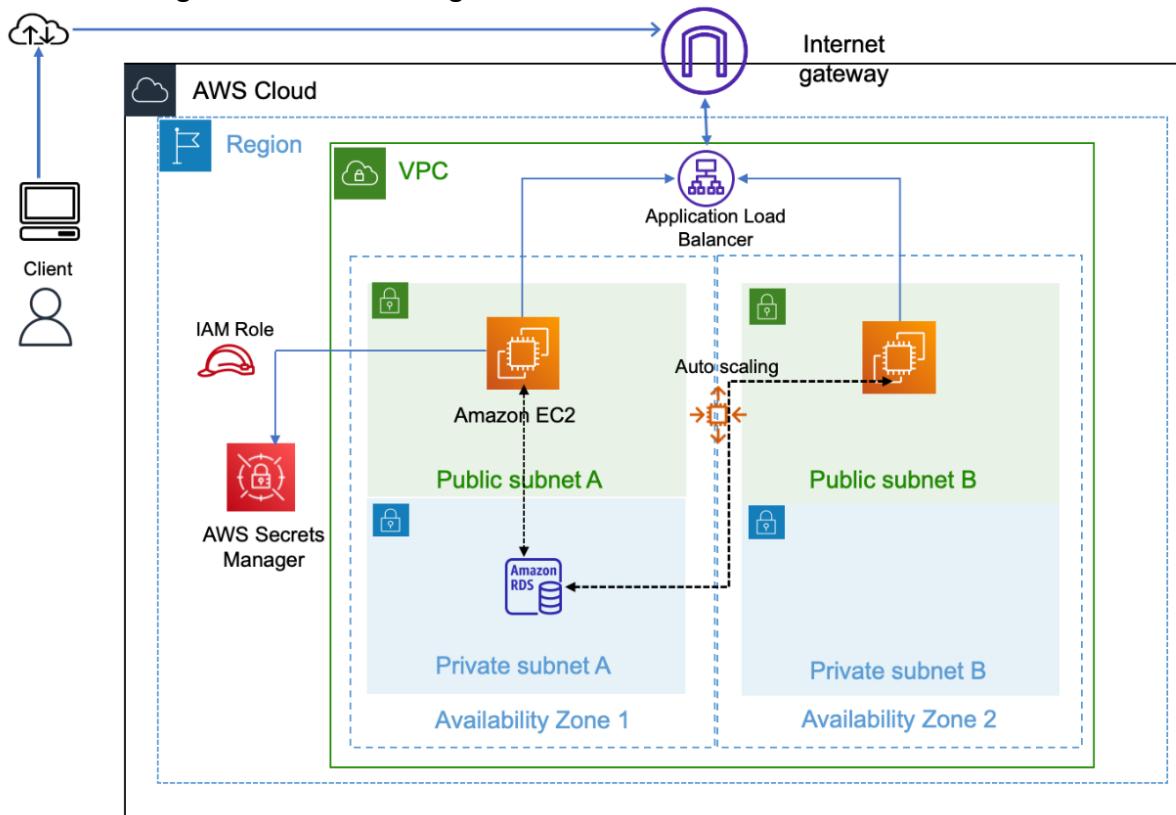


Phase 1 – Planning the design and estimating cost

Task 1: Creating an architectural diagram



Task 2: Developing a cost estimate

The screenshot shows the AWS Pricing Calculator interface with the title "My Estimate". The "Estimate summary" section displays the following costs:

Upfront cost	Monthly cost	Total 12 months cost
0.00 USD	71.35 USD	856.20 USD

This total includes the upfront cost of \$0.00 and a monthly cost of \$71.35 USD for 12 months. The "My Estimate" table lists the services used:

Service Name	Status	Upfront cost	Monthly cost	Description	Region	Config Summary
AWS Secrets Manager	-	0.00 USD	5.80 USD	Number of secrets (2), Av...	US East (N. Virginia)	Number of secrets (2), Av...
Amazon RDS for MySQL	-	0.00 USD	31.72 USD	Storage for each RDS inst...	US East (N. Virginia)	Storage for each RDS inst...
Amazon EC2	-	0.00 USD	17.16 USD	Tenancy (Shared Instance...)	US East (N. Virginia)	Tenancy (Shared Instance...)
Elastic Load Balancing	-	0.00 USD	16.67 USD	Number of Application Lo...	US East (N. Virginia)	Number of Application Lo...

The "Getting Started with AWS" sidebar offers links to "Get started for free" and "Contact Sales". The bottom of the page includes an "Acknowledgement" note about the estimate's limitations and a footer with standard links like Privacy, Site terms, and Cookie preferences.

Phase 2 – Creating a basic functional web application

Task 1: Creating a virtual network

The screenshot shows the AWS VPC Details page for the VPC `vpc-070adf3d119beef22 / CapstoneVPC`. The main pane displays the following details:

VPC ID	State	DNS hostnames	DNS resolution
<code>vpc-070adf3d119beef22</code>	Available	Enabled	Enabled
Tenancy	DHCP option set	Main route table	Main network ACL
Default	<code>dopt-076bbeec2f7f58057</code>	<code>rtb-071736ca77f35d40b</code>	<code>acl-0597610e6c6a22d60</code>
Default VPC	IPv4 CIDR	IPv6 pool	IPv6 CIDR (Network border group)
No	<code>10.0.0.0/16</code>	–	–
Network Address Usage metrics	Route 53 Resolver DNS Firewall rule groups	Owner ID	–
Disabled	–	<code>744647157586</code>	–

Below the details, there are tabs for Resource map, CIDRs, Flow logs, Tags, and Integrations. The Resource map section shows:

- VPC**: `Show details`, Your AWS virtual network, CapstoneVPC.
- Subnets**: (0) Subnets within this VPC.
- Route tables**: (1) Route network traffic to resources, `rtb-071736ca77f35d40b`.
- Network connections**: (0) Connections to other networks.

At the bottom right, there are links for © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

The screenshot shows the AWS Internet Gateway Details page for the Internet gateway `igw-0404efd42b97f622d`. The main pane displays the following details:

Internet gateway ID	State	VPC ID	Owner
<code>igw-0404efd42b97f622d</code>	Attached	<code>vpc-070adf3d119beef22 CapstoneVPC</code>	<code>744647157586</code>

Below the details, there is a **Tags** section with a search bar and a table:

Key	Value
Name	CapstoneIGW

At the bottom right, there are links for © 2024, Amazon Web Services, Inc. or its affiliates., Privacy, Terms, and Cookie preferences.

The screenshot shows the AWS VPC Management Subnets page. A green banner at the top indicates "You have successfully created 1 subnet: subnet-0b7ad1153cf2dfe60". The main table displays one subnet entry:

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
Public Subnet 1	subnet-0b7ad1153cf2dfe60	Available	vpc-070adf3d119beef22 CapstoneVPC	10.0.1.0/24	-

The left sidebar navigation includes sections for VPC dashboard, EC2 Global View, Filter by VPC, Virtual private cloud (Your VPCs, Subnets, Route tables, Internet gateways, Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, Peering connections), Security (Network ACLs, Security groups), DNS firewall (Rule groups, Domain lists), and CloudShell/Feedback.

The screenshot shows the AWS VPC Console RouteTables page. The main table displays two route tables:

Name	Route table ID	Explicit subnet assoc...	Edge associations	Main	VPC
-	rtb-071736ca77f35d40b	-	-	Yes	vpc-070adf3d119beef22 CapstoneVPC
-	rtb-092ffee23eb4d5002	-	-	Yes	vpc-0ec58bc1482871b30

The details for the first route table (rtb-071736ca77f35d40b) are shown in the bottom panel:

Details			
Route table ID rtb-071736ca77f35d40b	Main Yes	Explicit subnet associations -	Edge associations -
VPC vpc-070adf3d119beef22 CapstoneVPC	Owner ID 744647157586		

The left sidebar navigation is identical to the previous screenshot, including sections for VPC dashboard, EC2 Global View, Filter by VPC, Virtual private cloud, Security, and DNS firewall.

Task 2: Creating a virtual machine

The screenshot shows the AWS EC2 Instances page. On the left, a sidebar lists various services: EC2 Dashboard, EC2 Global View, Events, Console-to-Code, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volume, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, CloudShell, and Feedback. The main content area displays a table titled "Instances (1) Info". The table has columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, and Public IPv4 IP. One row is shown: CapstonePOC, i-094fd663d0bf3ffa9, Running, t2.micro, 2/2 checks passed, us-east-1a, ec2-54-225-60-0.comp..., 54.225.60.0. Below the table, a modal window titled "Select an instance" is open, showing the same information as the table.

Task 3: Testing the deployment

The screenshot shows a web browser window with the URL "54.225.60.0". The page title is "Students". The header features a logo of graduates under umbrellas and the text "XYZ University". It includes links for "Home" and "Students list". The main content area is titled "Welcome" and contains the text "Use this app to keep track of your student inquiries" and a link "List of students".

Name	Address	City	State	Email	Phone
Hannah	1 College View	Starkville	MS	hannahemail@msstate.edu	123456789101
Anne	54 Dorm Room	Here	MS	hahaha@gmail.com	25678876543
Megan	42 Not here drive	New Orleans	LA	meganisnthere@yahoo.com	86754654234

[Add a new student](#)

Phase 3 – Decoupling the application components

Task 1: Changing the VPC configuration

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
Private Subnet 1	subnet-0f80be38d5d41a3aa	Available	vpc-070adf3d119beef22 Caps...	10.0.2.0/24	-
-	subnet-093283ebad39d5d	Available	vpc-0ec58bc1482871b30	172.31.32.0/20	-
-	subnet-099ec51723e229c00	Available	vpc-0ec58bc1482871b30	172.31.64.0/20	-
-	subnet-03b500e50d8af150a	Available	vpc-0ec58bc1482871b30	172.31.0.0/20	-
-	subnet-03f165476c8062c50	Available	vpc-0ec58bc1482871b30	172.31.80.0/20	-
-	subnet-018a12cac069a5ab1	Available	vpc-0ec58bc1482871b30	172.31.48.0/20	-
Public Subnet 1	subnet-0b7ad1153cf2df6e0	Available	vpc-070adf3d119beef22 Caps...	10.0.1.0/24	-
-	subnet-0f45df669ff55b2a5	Available	vpc-0ec58bc1482871b30	172.31.16.0/20	-

[Create subnet](#)

Route table rtb-027a1f6795a9ba131 | CapstonePrivateRT was created successfully.

VPC > Route tables > rtb-027a1f6795a9ba131

rtb-027a1f6795a9ba131 / CapstonePrivateRT

Details **Info**

Route table ID	Main	Explicit subnet associations	Edge associations
rtb-027a1f6795a9ba131	No	-	-
VPC	Owner ID	Edge associations	
vpc-070adf3d119bee22 CapstoneVPC	744647157586	-	

Routes **Subnet associations** **Edge associations** **Route propagation** **Tags**

Routes (1)

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

Both Edit routes < 1 >

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

You have successfully created 1 subnet: subnet-0554892310c30d08b

Subnets (9) **Info**

Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
Private Subnet 1	subnet-0f80be38d5d41a3aa	Available	vpc-070adf3d119bee22 Caps...	10.0.2.0/24	-
-	subnet-09c35283ehad39d3d	Available	vpc-0ec58bc1482871b30	172.31.32.0/20	-
-	subnet-099ec31723e229c00	Available	vpc-0ec58bc1482871b30	172.31.64.0/20	-
-	subnet-03b500e50d8af150a	Available	vpc-0ec58bc1482871b30	172.31.0.0/20	-
-	subnet-03f165476c8062c50	Available	vpc-0ec58bc1482871b30	172.31.80.0/20	-
-	subnet-018a12ca069a5ab1	Available	vpc-0ec58bc1482871b30	172.31.48.0/20	-
Private Subnet 2	subnet-0554892310c30d08b	Available	vpc-070adf3d119bee22 Caps...	10.0.4.0/24	-
Public Subnet 1	subnet-0b7ad1153c72dfe66	Available	vpc-070adf3d119bee22 Caps...	10.0.1.0/24	-
-	subnet-0f45df669ff55b2a5	Available	vpc-0ec58bc1482871b30	172.31.16.0/20	-

Select a subnet

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS VPC Console with the URL <https://us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#subnets>. A green success message at the top states: "Subnet (subnet-0f90be38d5d41a5aa) has been successfully associated with route table (rtb-027a1f6795a9ba151)." The main table lists nine subnets under "Subnets (9) Info". The columns include Name, Subnet ID, State, VPC, IPv4 C..., Available..., and Route table. One row is highlighted in blue, showing "subnet-0f90be38d5d41a5aa" with its details.

Name	Subnet ID	State	VPC	IPv4 C...	Available...	Route table
Public Subnet 1	subnet-0f90be38d5d41a5aa	Available	vpc-070adfd3d119beef22 Capsto...	10.0.1.0/24	us-east-1a	-
Private Subnet 2	subnet-0554892310c30d08b	Available	vpc-070adfd3d119beef22 Capsto...	10.0.4.0/24	us-east-1b	rtb-027a1f6795a9ba131 Caps...
Private Subnet 1	subnet-0f90be38d5d41a5aa	Available	vpc-070adfd3d119beef22 Capsto...	10.0.2.0/24	us-east-1a	rtb-027a1f6795a9ba131 Caps...
-	subnet-093285ebad39d5d	Available	vpc-0ec5bbc1482871b30	172.31.32...	us-east-1a	-
-	subnet-09ec51723e229c00	Available	vpc-0ec5bbc1482871b30	172.31.64...	us-east-1f	-
-	subnet-03b500e50d8af150a	Available	vpc-0ec5bbc1482871b30	172.31.0...	us-east-1b	-
-	subnet-03f165476c8062c50	Available	vpc-0ec5bbc1482871b30	172.31.80...	us-east-1c	-
-	subnet-018a12cac069a5ab1	Available	vpc-0ec5bbc1482871b30	172.31.48...	us-east-1e	-
-	subnet-0f45df669ff55b2a5	Available	vpc-0ec5bbc1482871b30	172.31.16...	us-east-1d	-

Task 2: Creating and configuring the Amazon RDS database

The screenshot shows the AWS EC2 Security Groups console with the URL <https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#securityGroups>. A green success message at the top states: "Security group (sg-08fdbbf3e0047cd03 | CapstoneDBSG) was created successfully". The main page displays the "sg-08fdbbf3e0047cd03 - CapstoneDBSG" security group details. It includes fields for Security group name (CapstoneDBSG), Security group ID (sg-08fdbbf3e0047cd03), Description (Security group for database), VPC ID (vpc-070adfd3d119beef22), Owner (744647157586), Inbound rules count (1 Permission entry), and Outbound rules count (1 Permission entry). Below this, the "Inbound rules (1)" section shows one rule: "sgr-0d0f59ede5692822a" with IP version IPv4, Type MySQL/Aurora, Protocol TCP, Port range 3306, and Source 10.0.0.0/16.

Name	Security group rule...	IP version	Type	Protocol	Port range	Source
sgr-0d0f59ede5692822a	IPv4	MySQL/Aurora	TCP	3306	10.0.0.0/16	

The screenshot shows the AWS RDS console with the URL us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#databases. The left sidebar includes links for Dashboard, Databases (selected), Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Zero-ETL Integrations, Events, Event subscriptions, Recommendations (0), and Certificate update. The main content area displays a table titled 'Databases (1)'. The table has columns: DB identifier, Status, Role, Engine, Region & AZ, Size, Recommendations, CPU, and Current activity. One row is shown for 'capstonebd', which is 'Available', running on 'MySQL Community' in 'us-east-1a' on a 'db.t3.micro' instance.

Task 3: Configuring the development environment

The screenshot shows the AWS Cloud9 console with the URL us-east-1.console.aws.amazon.com/cloud9control/home?region=us-east-1#/environments. The left sidebar includes links for My environments, Shared with me, All account environments, and Documentation. The main content area displays a table titled 'Environments (1)'. The table has columns: Name, Cloud9 IDE, Environment type, Connection, Permissions, and Owner ARN. One row is shown for 'CapstoneIDE', which is an 'EC2 instance' connected via 'Secure Shell (SSH)' and owned by the user with the ARN 'arn:awssts:744647157586:assumed-role/vocabs/user5197919=Heidebrecht_Anne_Marie_Ling'.

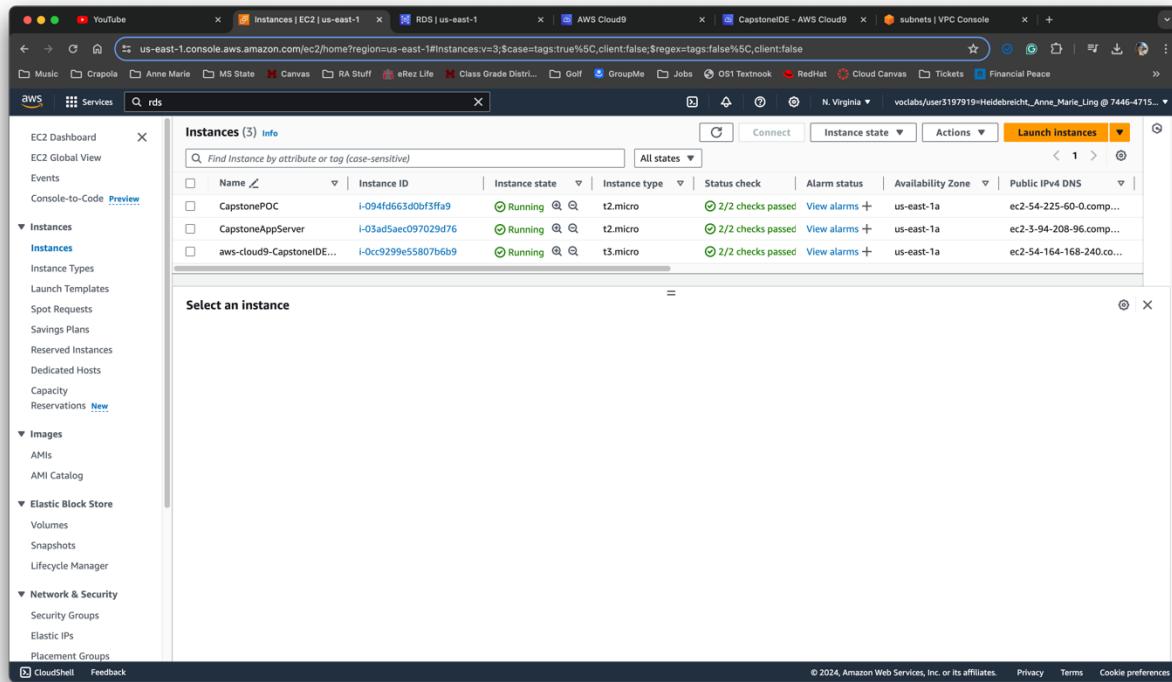
Task 4: Provisioning Secret Manager

```

aws - "ip-10-0-1-83.ec2.ir x Immediate x + 
{
    "ARN": "arn:aws:secretsmanager:us-east-1:744647157586:secret:Mydbsecret-542y99",
    "Name": "Mydbsecret",
    "VersionId": "5d5dfbd9-ac84-4554-b4e4-d431b0983b52"
}
(END)

```

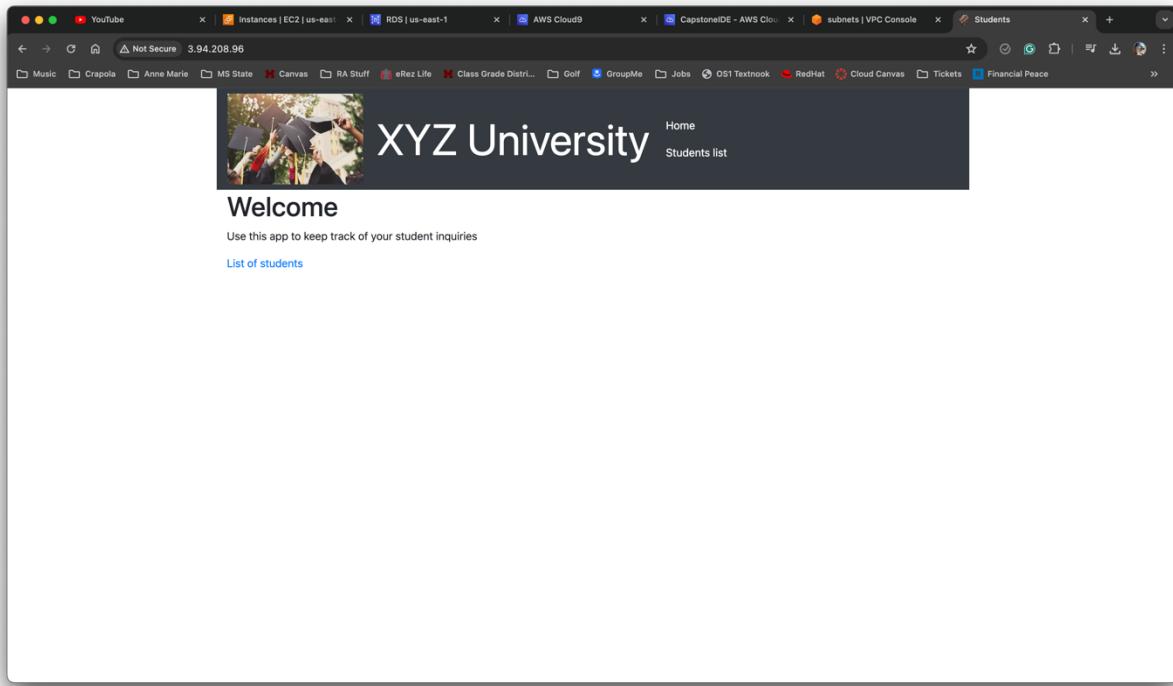
Task 5: Provisioning a new instance for the web server



The screenshot shows the AWS Cloud9 Instances page. The left sidebar is collapsed, and the main area displays a table of instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
CapstonePOC	i-094fd63d0bf3ffa9	Running	t2.micro	2/2 checks passed	View alarms	us-east-1a	ec2-54-225-60-0.comp...
CapstoneAppServer	i-03ad5aec097029d76	Running	t2.micro	2/2 checks passed	View alarms	us-east-1a	ec2-54-208-96.comp...
aws-cloud9-CapstoneIDE...	i-0cc299e55807b6b9	Running	t3.micro	2/2 checks passed	View alarms	us-east-1a	ec2-54-164-168-240.co...

A modal window titled "Select an instance" is open at the bottom, listing the same three instances.



Task 6: Migrating the database

```
1 -- MySQL dump 10.19 Distrib 10.5.23-MariaDB, for Linux (x86_64)
2 --
3 -- Host: 10.0.1.184 Database: STUDENTS
4 --
5 -- Server version 5.0.36-ubuntu0.22.04.1
6
7 /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@CHARACTER_SET_CLIENT */;
8 /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@CHARACTER_SET_RESULTS */;
9 /*!40101 SET @OLD_COLLATION_CONNECTION=@COLLATION_CONNECTION */;
10 /*!40101 SET NAMES utf8mb4 */;
11 /*!40103 SET @OLD_TIME_ZONE=@TIME_ZONE */;
12 /*!40103 SET TIME_ZONE='+00:00' */;
13 /*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
14 /*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
15 /*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
16 /*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
17
18 --
19 -- Current Database: STUDENTS
20 --
21 CREATE DATABASE /*!32312 IF NOT EXISTS*/ `STUDENTS` /*!40100 DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci */ /*!40101 DEFAULT ENCRYPTION='N' */;
22 USE `STUDENTS`;
23 --
24 --
25 -- Table structure for table `students`
26 --
27 -- Table structure for table `students`
28 --
29
30 DROP TABLE IF EXISTS `students`;
31 /*!40101 SET @OLD_CHARACTER_SET_CLIENT = @@character_set_client */;
32 /*!40101 SET @OLD_CHARACTER_SET_RESULTS = @@character_set_results */;
33 /*!40101 SET @OLD_COLLATION_CONNECTION = @@collation_connection */;
34 CREATE TABLE `students` (
35   `id` int NOT NULL AUTO_INCREMENT,
36   `name` varchar(255) NOT NULL,
37   `address` varchar(255) NOT NULL,
38   `city` varchar(255) NOT NULL,
39   `state` varchar(255) NOT NULL,
40   `email` varchar(255) NOT NULL,
41   `phone` varchar(255) NOT NULL
42 ) ENGINE=InnoDB AUTO_INCREMENT=1000000000000000000 DEFAULT CHARSET=utf8mb4;
```

```
mysql -p10-0-1-83.ec2.x
vclabs:/environment $ mysqldump -h 10.0.1.184 -u nodeapp -p --databases STUDENTS > data.sql
Enter password:
vclabs:/environment $ mysql -h capstoneb.cwhf4nvuflp5.us-east-1.rds.amazonaws.com -u nodeapp -p STUDENTS < data.sql
Enter password:
vclabs:/environment $
```

Task 7: Testing the application

Name	Address	City	State	Email	Phone
Anne	54 Dorm Room	Here	MS	hahaha@gmail.com	25678876543
Megan	42 Not here drive	New Orleans	LA	meganisnthere@yahoo.com	86754654234
NewEC2 Things	21 Haha Lane	North Pole	AL	notarealemail@hotmail.com	76543456788

[Add a new student](#)

Phase 4 – Implementing high availability and scalability

Task 1: Creating an Application Load Balancer

Name	Subnet ID	State	VPC	IPv4 CIDR Block	Availability Zone	Route table
Public Subnet 2	subnet-0d541b8e89da779bb	Available	vpc-070adf3d119beef22 Capstone	10.0.3.0/24	us-east-1b	-
Public Subnet 1	subnet-0b7ad1153cf2fde60	Available	vpc-070adf3d119beef22 Capstone	10.0.1.0/24	us-east-1a	-
Private Subnet 2	subnet-0554892310c30d0bb	Available	vpc-070adf3d119beef22 Capstone	10.0.4.0/24	us-east-1b	rtb-027a1f6795a9ba131 Caps...
Private Subnet 1	subnet-0f00be38d5d41a3aa	Available	vpc-070adf3d119beef22 Capstone	10.0.2.0/24	us-east-1a	rtb-027a1f6795a9ba131 Caps...
-	subnet-093328ebad39d5d	Available	vpc-0ec58bc1482871b30	172.31.32...	us-east-1a	-
-	subnet-099ec31723e229c00	Available	vpc-0ec58bc1482871b30	172.31.64...	us-east-1f	-
-	subnet-03500e50d8af150a	Available	vpc-0ec58bc1482871b30	172.31.0...	us-east-1b	-
-	subnet-03f165476c8062c50	Available	vpc-0ec58bc1482871b30	172.31.80...	us-east-1c	-

[Select a subnet](#)

The screenshot shows the AWS EC2 Load Balancers console with the 'CapstoneALB' application load balancer selected. The 'Details' tab is active, displaying the following information:

Load balancer type	Status	IP address type
Application	Active	IPv4
Scheme	Internet-facing	Availability Zones
	Hosted zone: Z355XD0TRQ7X7K	subnet-0d541b8e89da779bb us-east-1b (use1-az1)
		subnet-0b7ad1153cf2fe60 us-east-1a (use1-az6)
Load balancer ARN	DNS name info	Date created
arn:aws:elasticloadbalancing:us-east-1:744647157586:loadbalancer/app/CapstoneALB/57833f57e2291018	CapstoneALB-40081551.us-east-1.elb.amazonaws.com (A Record)	April 20, 2024, 17:36 (UTC-05:00)

Below the details, there are tabs for 'Listeners and rules', 'Network mapping', 'Resource map - new', 'Security', 'Monitoring', 'Integrations', 'Attributes', and 'Tags'. The 'Listeners and rules' tab is selected, showing one listener rule: 'Forward to target group'.

Task 2: Implementing Amazon EC2 Auto Scaling

The screenshot shows the AWS EC2 Machine Images (AMIs) console with the 'CapstoneAMI' AMI listed. The table shows the following information:

Name	AMI ID	Source	Owner	Visibility	Status	Creation date
CapstoneAMI	ami-0e8eaef7dec4858734	744647157586/CapstoneAMI	744647157586	Private	Available	2024/04/20 17:40 GMT-5

A modal window titled 'Select an AMI' is open, showing the same list of AMIs.

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchTemplates:

Launch Templates (1) Info

Search

Launch Template ID | Launch Template Name | Default Version | Latest Version | Create Time | Created By

Launch Template ID	Launch Template Name	Default Version	Latest Version	Create Time	Created By
lt-0cfe1ed65620576f	CapstoneTemplate	1	1	2024-04-20T22:45:01.000Z	arn:aws:sts::744647157586:assumed-role/v...

Select a launch template

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#AutoScalingGroups:

Auto Scaling groups (1) Info

Search your Auto Scaling groups

Launch configurations | Launch templates | Actions | Create Auto Scaling group

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zones
CapstoneAutoScalingGroup	CapstoneTemplate Version Default	1	-	1	1	4	us-east-1a, us-east-1b

0 Auto Scaling groups selected

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with various navigation options like EC2 Dashboard, EC2 Global View, Events, Console-to-Code, Instances, Instances Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, CloudShell, and Feedback.

The main area displays a table titled "Instances (1/4) Info". The table has columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4 DNS. There are four entries:

- CapstonePOC (t2.micro)
- CapstoneAppServer (t2.micro)
- aws-cloud9-CapstoneIDE... (t3.micro)
- Auto Scaling Instance** (t3.micro)

The "Auto Scaling Instance" row is highlighted with a blue border. Below the table, there's a detailed view for the selected instance:

Instance: i-05676ad8c406b53ef (Auto Scaling Instance)

Details tab is active. The instance summary includes:

- Instance ID: i-05676ad8c406b53ef (Auto Scaling Instance)
- Public IPv4 address: 18.207.98.241 [open address]
- Private IPv4 addresses: 10.0.3.72
- Public IPv4 DNS: ec2-18-207-98-241.compute-1.amazonaws.com [open address]
- IPv6 address: -
- Instance state: Running
- Hostname type: ip-10-0-3-72.ec2.internal
- Private IP DNS name (IPv4 only): ip-10-0-3-72.ec2.internal
- Instance type: t3.micro
- Auto-assigned IP address: 18.207.98.241 [Public IP]
- VPC ID: vpc-070adf5d119beef22 (CapstoneVPC)
- IAM Role: Subnet ID: -

Tags tab is also present. At the bottom right, there are links for AWS Compute Optimizer finding, Opt-in to AWS Compute Optimizer for recommendations, Learn more, and Auto Scaling Group name.

Task 3: Accessing the application

The screenshot shows a web browser displaying a student management application. The URL is [Not Secure capstonealb-40081551.us-east-1.elb.amazonaws.com/students](http://capstonealb-40081551.us-east-1.elb.amazonaws.com/students).

The page header features the XYZ University logo and navigation links for Home and Students list. The main content is titled "All students" and displays a table with the following data:

Name	Address	City	State	Email	Phone	Action
Anne	54 Dorm Room	Here	MS	hahaha@gmail.com	25678876543	edit
NewEC2 Things	21 Haha Lane	North Pole	AL	notarealemail@hotmail.com	76543456788	edit
CapstoneALB	123 DNS URL	I Did It	MS	yayme@gmail.com	36784364253	edit

A green button at the bottom left says "Add a new student".

Task 4: Load testing the application

CodeWhisperer AWS: profile:default

```
us-east-1.console.aws.amazon.com/cloud9/de/275ab103471d4e5b9516299257948cce?region=us-east-1
```

```
added 31 packages in 5s

1 package is looking for funding
  run 'npm fund' for details
  voclabs:~/environment $ loadtest --rps 2000 -c 1000 -k http://CapstoneALB-40081551.us-east-1.elb.amazonaws.com
  Requests: 6864, requests per second: 1373, mean latency: 512.9 ms

  Target URL: http://CapstoneALB-40081551.us-east-1.elb.amazonaws.com
  Max time (s): 10
  Target rps: 2000
  Concurrent clients: 3959
  Agent: keepalive

  Completed requests: 16216
  Total errors: 0
  Total time: 10.003 s
  Mean latency: 786.6 ms
  Effective rps: 1621

  Percentage of requests served within a certain time
  50% 594 ms
  90% 781 ms
  95% 861 ms
  99% 955 ms
  100% 7245 ms (longest request)
  voclabs:~/environment $ loadtest --rps 1000 -c 500 -k http://CapstoneALB-40081551.us-east-1.elb.amazonaws.com
  Requests: 4995, requests per second: 999, mean latency: 7 ms

  Target URL: http://CapstoneALB-40081551.us-east-1.elb.amazonaws.com
  Max time (s): 10
  Target rps: 1000
  Concurrent clients: 49
  Agent: keepalive

  Completed requests: 10000
  Total errors: 0
  Total time: 10.002 s
  Mean latency: 5.2 ms
  Effective rps: 1000

  Percentage of requests served within a certain time
  50% 2 ms
  90% 16 ms
  95% 24 ms
  99% 37 ms
  100% 63 ms (longest request)
  voclabs:~/environment $ ]
```

CodeWhisperer AWS: profile:default

```
us-east-1.console.aws.amazon.com/cloud9/de/275ab103471d4e5b9516299257948cce?region=us-east-1
```

```
added 31 packages in 5s

99% 5425 ms
100% 5909 ms (longest request)
voclabs:~/environment $ loadtest --rps 2000 -c 1000 -k http://CapstoneALB-40081551.us-east-1.elb.amazonaws.com
Requests: 6871, requests per second: 1363, mean latency: 628.4 ms

  Target URL: http://CapstoneALB-40081551.us-east-1.elb.amazonaws.com
  Max time (s): 10
  Target rps: 2000
  Concurrent clients: 4283
  Agent: keepalive

  Completed requests: 15823
  Total errors: 0
  Total time: 10.001 s
  Mean latency: 675.3 ms
  Effective rps: 1582

  Percentage of requests served within a certain time
  50% 683 ms
  90% 866 ms
  95% 1357 ms
  99% 1956 ms
  100% 7399 ms (longest request)
  voclabs:~/environment $ loadtest --rps 4000 -c 2000 -k http://CapstoneALB-40081551.us-east-1.elb.amazonaws.com
  Requests: 7593, requests per second: 3798, mean latency: 568.5 ms

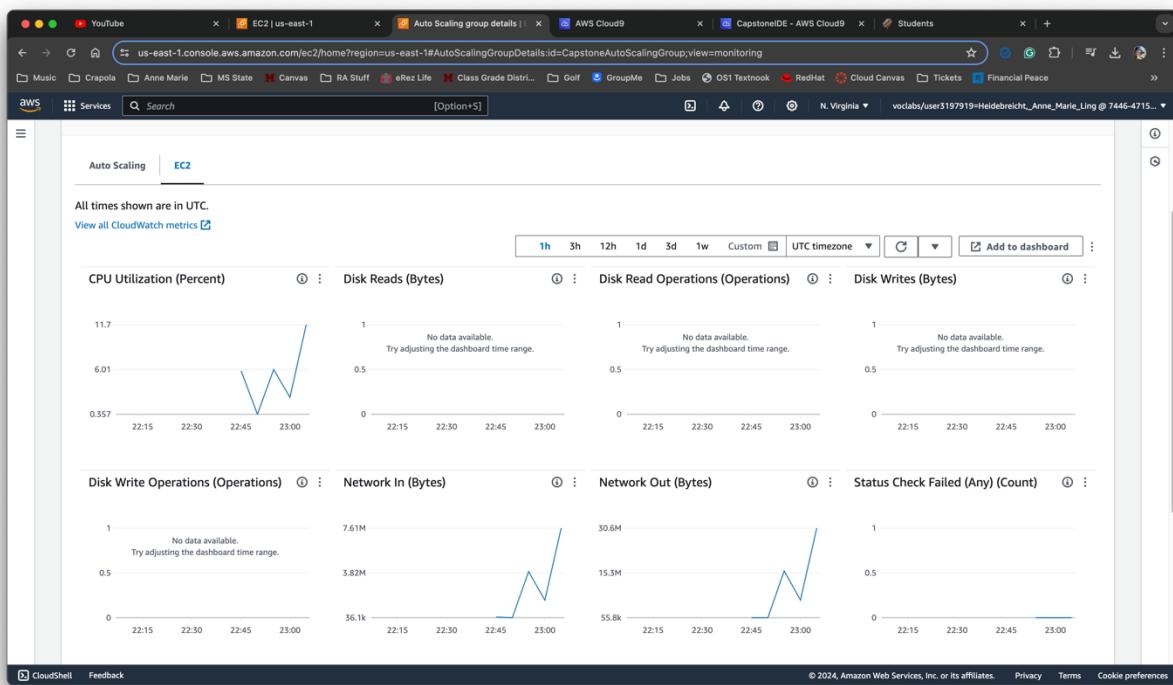
  Target URL: http://CapstoneALB-40081551.us-east-1.elb.amazonaws.com
  Max time (s): 10
  Target rps: 4000
  Concurrent clients: 21834
  Agent: keepalive

  Completed requests: 17515
  Total errors: 3
  Total time: 10.003 s
  Mean latency: 826 ms
  Effective rps: 1751

  Percentage of requests served within a certain time
  50% 627 ms
  90% 945 ms
  95% 1255 ms
  99% 5879 ms
  100% 7377 ms (longest request)

  -1: 3 errors
  voclabs:~/environment $ ]
```

The screenshot shows the AWS Cloud9 EC2 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, EC2 Global View, Events, Console-to-Code, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, CloudShell, and Feedback. The main area displays a table titled "Instances (1/5) Info" with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4 DNS. Five instances are listed: CapstonePOC, CapstoneAppServer, aws-cloud9-CapstoneIDE..., Auto Scaling Instance, and another Auto Scaling Instance. The second instance is selected. Below the table, a detailed view for the selected instance (Auto Scaling Instance i-05676ad8c406b53ef) is shown, including tabs for Details, Status and alarms, Monitoring, Security, Networking, Storage, and Tags. The Status checks section shows a green checkmark for System reachability check passed. The Metrics and Alarms sections are also visible.



Reflection

What are your reflections on the assignment? How did it help you, and what did you learn?
In this assignment, I learned how to use AWS resources to design and implement a highly available, scalable web application. It helped me understand how each component worked, and it also taught me how to troubleshoot my own issues that I ran into. I did have to look up some

resources to help me with this assignment. For example, Phase 3: Task 6, and all of Phase 4 referred to AWS Cloud Architecting. We do not have access to this course, so finding resources to help supplement this was difficult. Additionally, learning on the spot is not usually how I learn new things, so I think it took much longer than expected for me to complete this assignment. There were several times I became frustrated due to the lack of explanations given to achieve the tasks. Phase 3: Task 6 is a great example of not having enough guidance to understand how to properly get the database information. I wasn't exactly sure what I should screenshot, so I think I went a little overboard in making sure every step was accounted for. Once I had access to the guided solution, I redid my original project to ensure I had the correct version. I believe for most of my first attempts, many of my instances would not connect properly. I couldn't access any of the student's information, much less add and delete students. The second time through, I was able to complete the project in less than 3 hours. Overall, it was a good learning experience for me to develop my own understanding of cloud computing using AWS.