Student Name: Ta Quang Tung

Student ID: 104222196

Tutorial time: Saturday 3:00 P.M.

Date of submission: 18/6/23

**COS20019 – Cloud Computing Architecture - Assignment 1B**

**Marking scheme**

|  |  |  |
| --- | --- | --- |
| **Infrastructure Requirements** | **Mark** | **Page** |
| VPC with 2 public and 2 private subnets | .5 | 7 |
| Correct Public and Private Routing tables with correct subnet associations | 1 | 8 |
| Security groups properly configured and attached. | 1 | 9-11 |
| Network ACL properly configured and attached | 1.5 | 30-32 |
| Correct Web server and Test instances running in correct subnets | .5 | 13, 19 |
| Database schema as specified | .5 | 29 |
| Database running in correct subnets | 1 | 26 |
| S3 objects publicly accessible, using proper access policy | .5 | 36 |
| **Functional Requirements** |  |  |
| album.php page displayed from EC2 Web server | 1 | 38 |
| Provided URL is persistent (Elastic IP Association) | .5 | 17 |
| Photos loaded from S3 with matching metadata from RDS | 1 | 38 |
| Web server instance reachable from Test instance via ICMP | 1 | 5 |
| **Deductions** |  |  |
| Documentation not as specified or poorly presented (up to minus 20) |  |  |
| Serious misconfigurations of AWS services being used (up to minus 20) |  |  |

Table of contents

[Table of contents 2](#_Toc137821972)

[Website URL 3](#_Toc137821973)

[Data records 4](#_Toc137821974)

[Successful ping from TestInstance to WebServer 5](#_Toc137821975)

[1 Infrastructure deployment 6](#_Toc137821976)

[1.1 Create a VPC 6](#_Toc137821977)

[1.2 Create security groups 9](#_Toc137821978)

[1.3 Create EC2 instances 12](#_Toc137821979)

[1.3.1 Bastion/Web server instance 12](#_Toc137821980)

[1.3.2 Test instance 18](#_Toc137821981)

[1.4 Create RDS instance 21](#_Toc137821982)

[1.5 Network ACL 30](#_Toc137821983)

[2 Functional requirements of Photo Album website 33](#_Toc137821984)

[2.1 Photo storage (S3) 33](#_Toc137821985)

[2.2 Photo meta-data in RDS Database 37](#_Toc137821986)

[2.3 Photo Album website functionality 38](#_Toc137821987)

Website URL

<http://ec2-44-216-98-59.compute-1.amazonaws.com/cos80001/photoalbum/album.php>

Data records

A screenshot of a computer

Description automatically generated with medium confidence

Two data records are present in the database.

Successful ping from TestInstance to WebServer

A screenshot of a computer

Description automatically generated

The Test Instance can ping the Web Server.

1. Infrastructure deployment
   1. Create a VPC

A screenshot of a computer

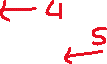
Description automatically generated



Create the VPC: (1) the name follows the correct pattern [FirstNameInitial][LastName]VPC, (2) set the CIDR block, (3) the subnet names indicate the region us-east-1

A screenshot of a computer

Description automatically generated



Create the VPC: (4) & (5) specify two public and two private subnets, (6) associate each subnet with their address blocks.

A screenshot of a computer

Description automatically generated

The VPC has been successfully created, and the subnets have been associated with the correct route tables.

* 1. Create security groups

A screenshot of a computer

Description automatically generated

Create the Test Instance security group that allows traffic from all sources.

A screenshot of a computer

Description automatically generated

Create the Web Server security group that allows HTTP and SSH traffic from all sources, and ICMP traffic from Test Instance.

A screenshot of a computer

Description automatically generated

Create the Database Server security group that allows MySQL traffic from the Web Server.

* 1. Create EC2 instances
     1. Bastion/Web server instance

A screenshot of a computer

Description automatically generated



Create the Bastion/Web Server instance from Amazon Linux 2 AMI (HVM), SSD Volume Type

A screenshot of a computer

Description automatically generated



Create the Bastion/Web Server instance: (3) set the type to t2.micro, (4) assign a key pair, (5) place the instance in public subnet 2.

A screenshot of a computer

Description automatically generated



Create the Bastion/Web Server instance: (6) assign the web server security group.

A screenshot of a computer

Description automatically generated



Create the Bastion/Web Server instance: (7) configure a user data script to install the Apache server.

A screenshot of a computer

Description automatically generated

Allocate an Elastic IP address in the us-east-1 region.

A screenshot of a computer

Description automatically generated



Assign the new Elastic IP address to the Bastion/Web Server instance.

* + 1. Test instance

A screenshot of a computer

Description automatically generated



Create the Test Instance.

A screenshot of a computer

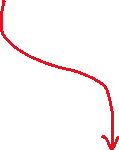
Description automatically generated



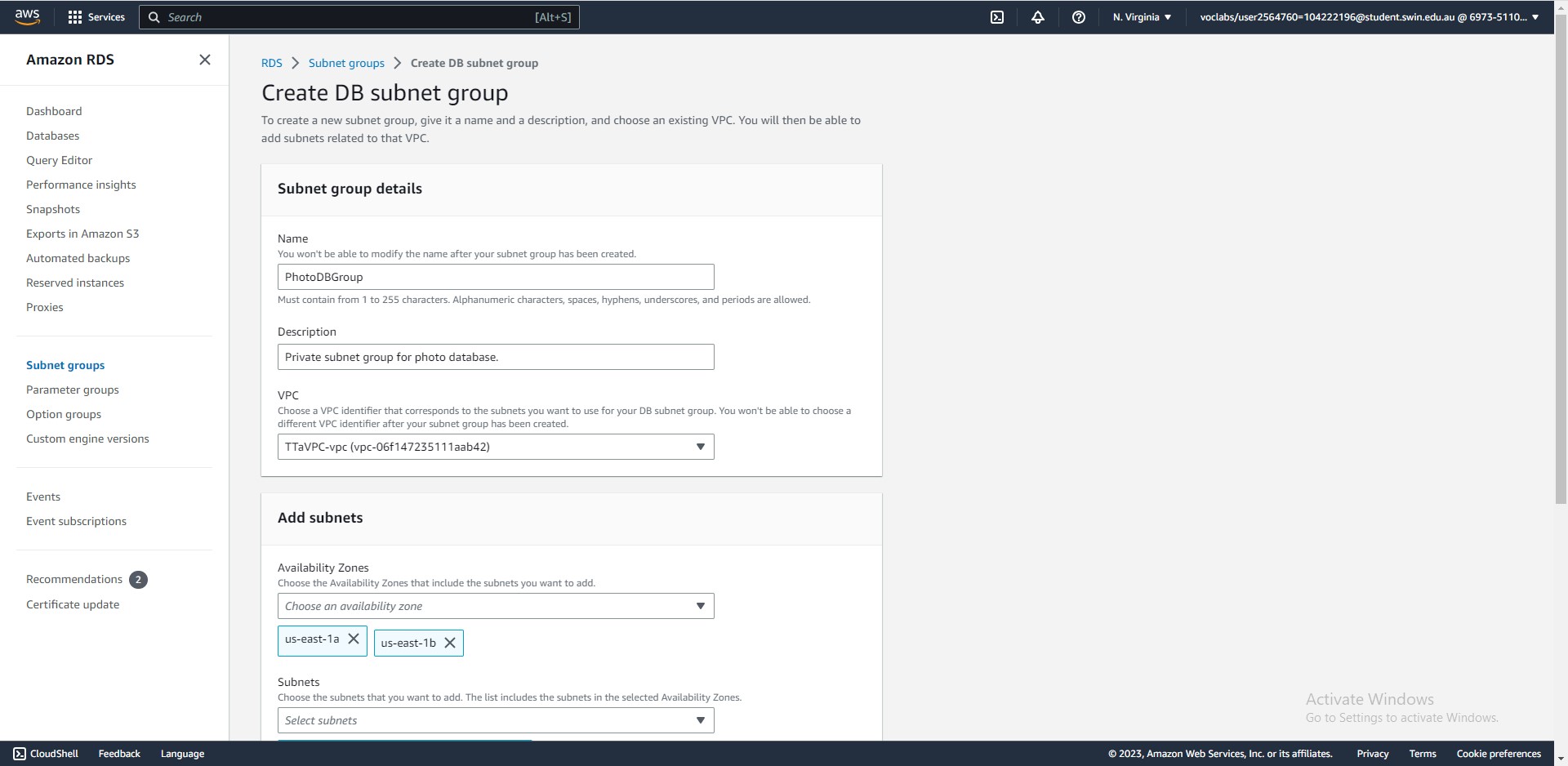
Create the Test Instance: assign a key pair, place the instance in private subnet 2, and assign the Test Instance security group.

A screenshot of a computer

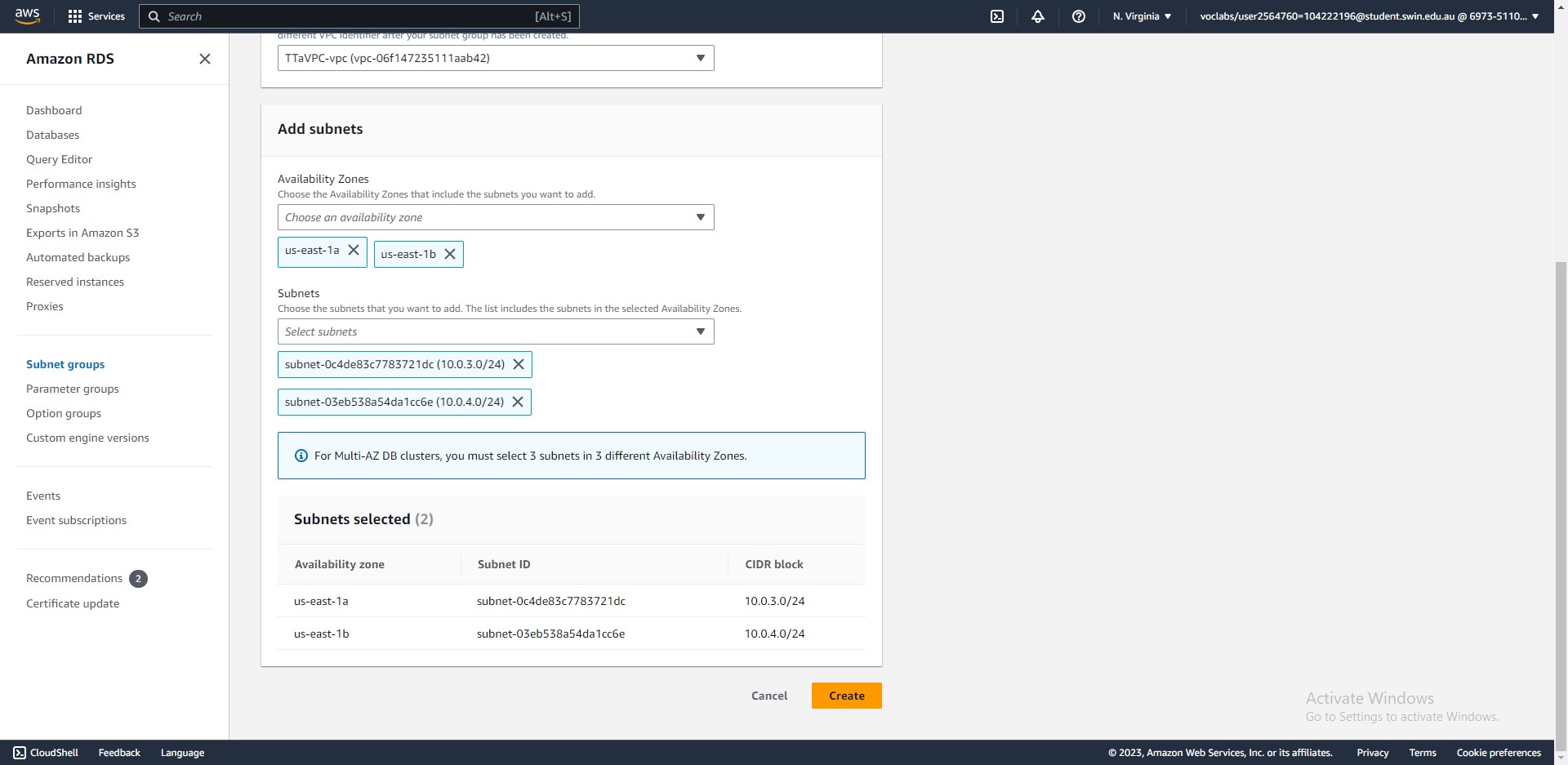
Description automatically generated



1. SSH into the Test instance from the Web Server instance, then (2) ping the Web Server’s private Ipv4 address.
   1. Create RDS instance

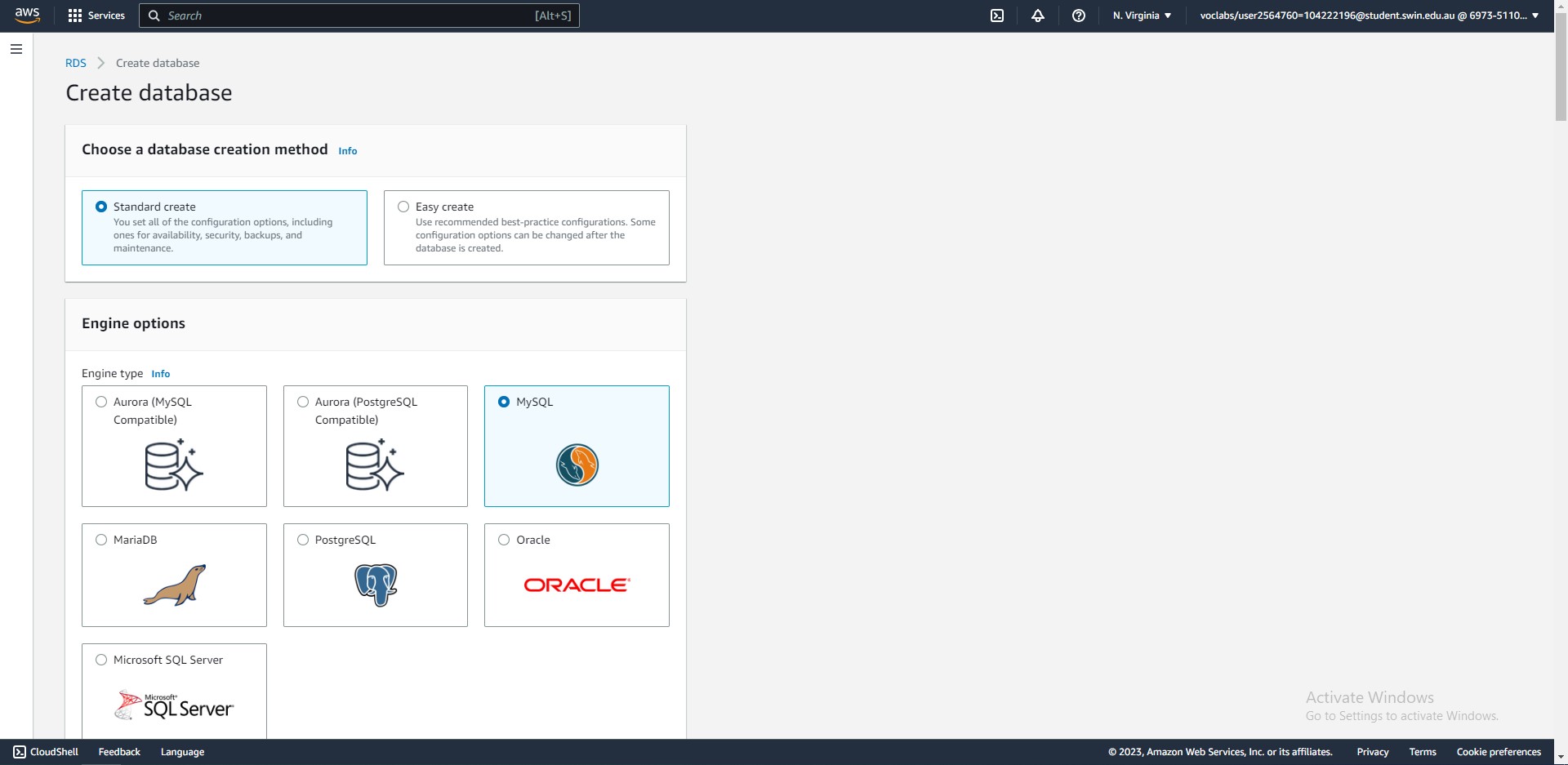


Create a DB subnet group in the VPC that spans both Availability Zones.



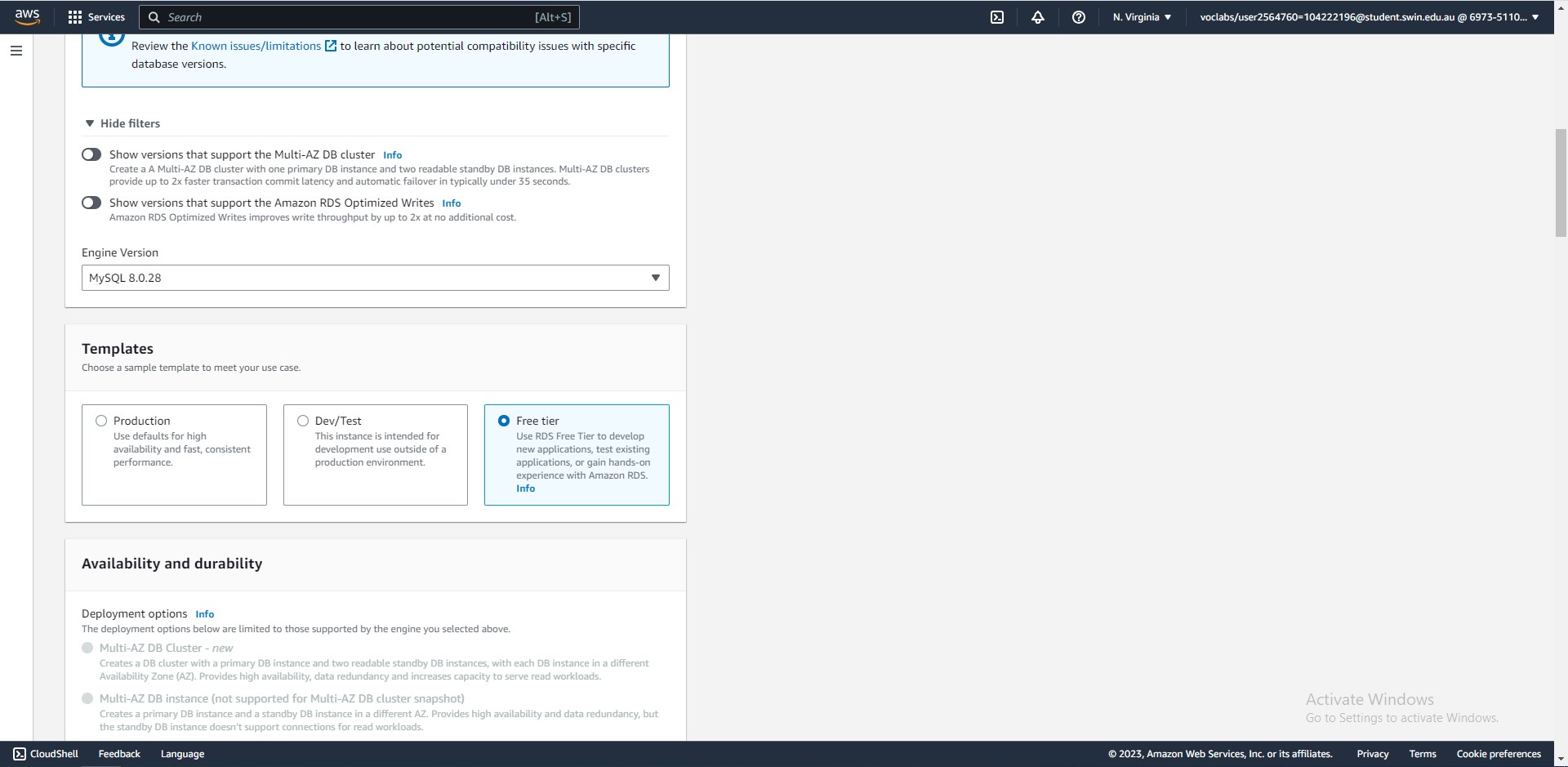


Assign both private subnets to the subnet group.



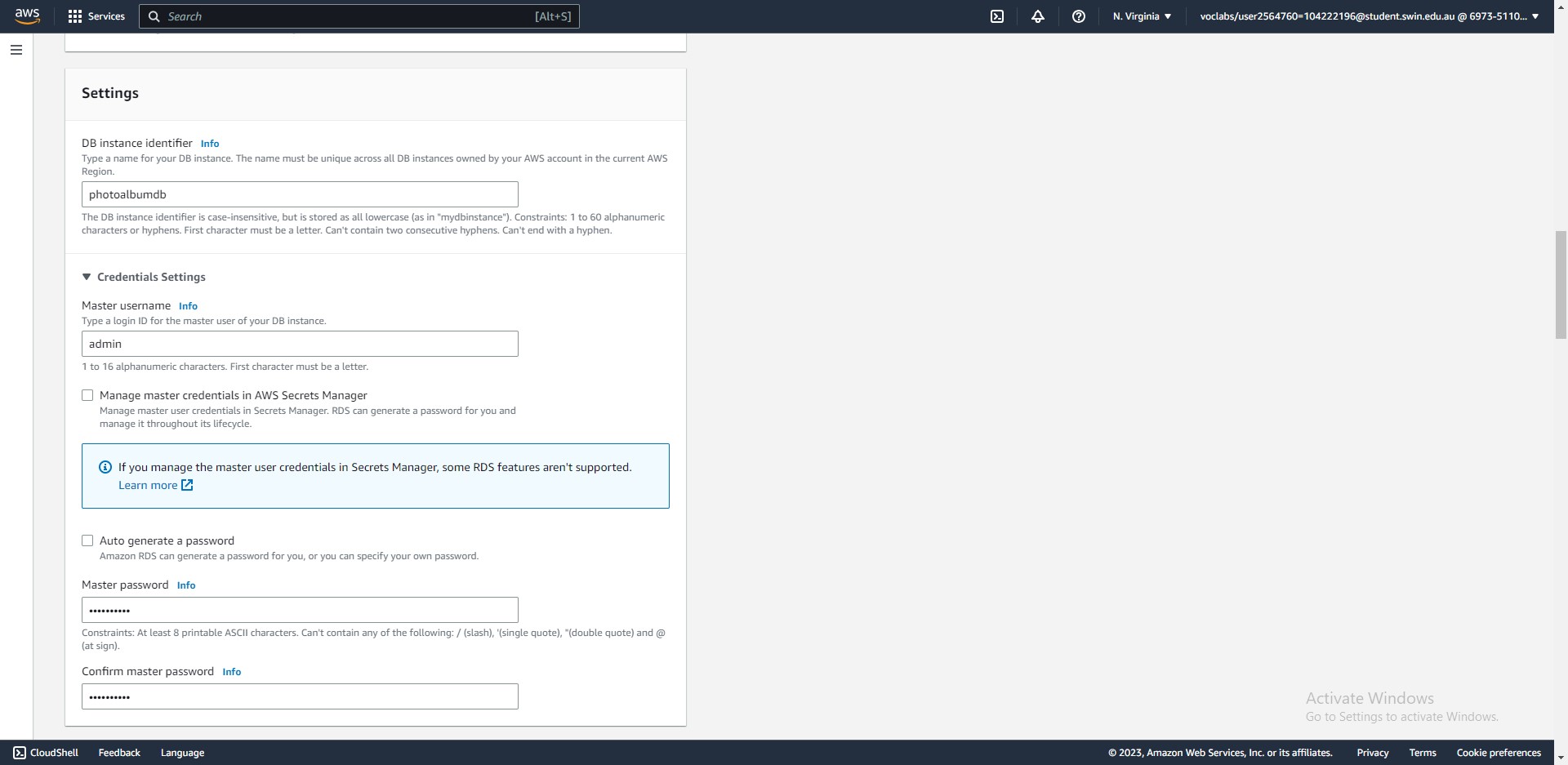


Create the RDS instance: (1) choose MySQL.



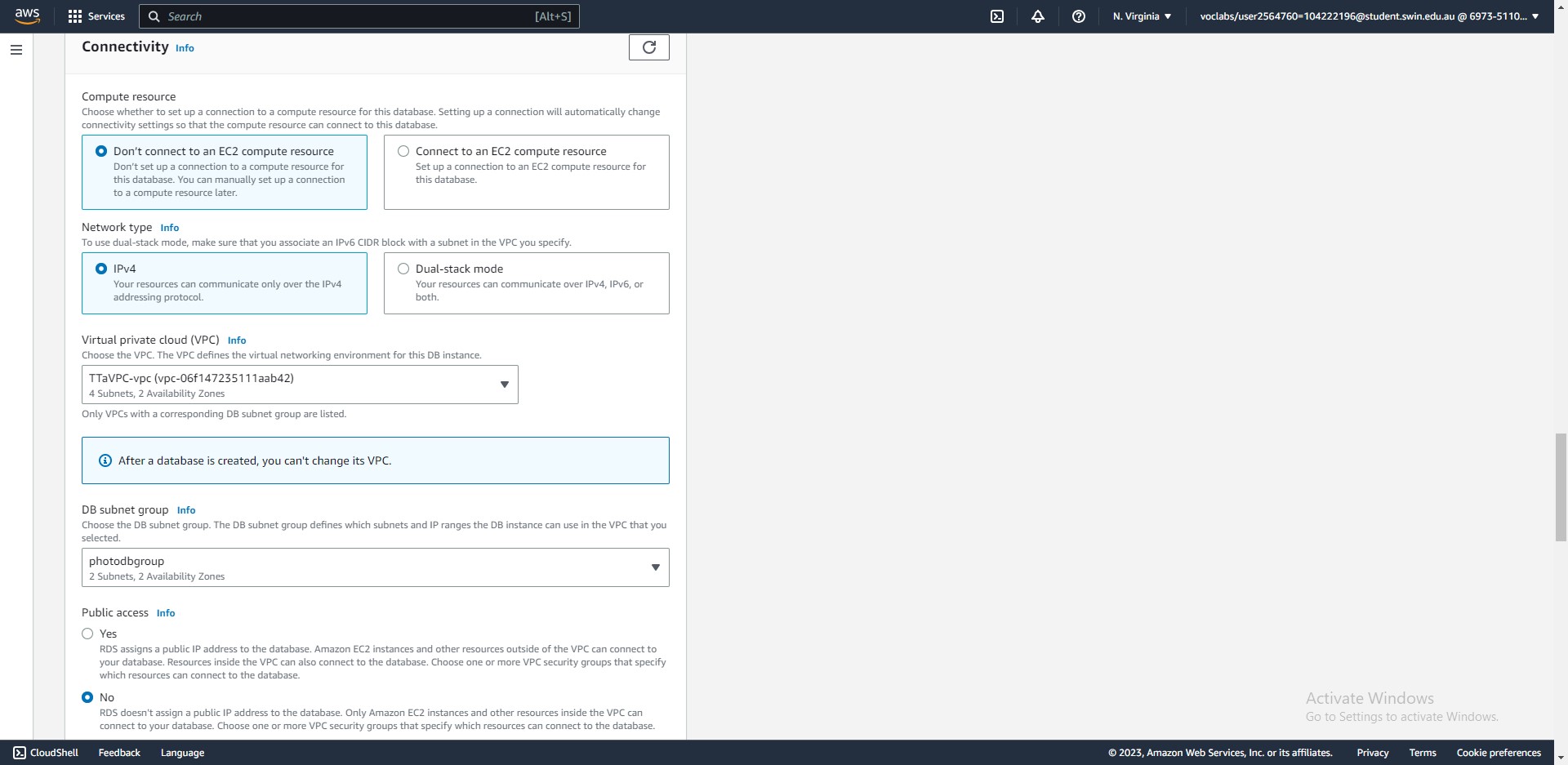


Create the RDS instance: (2) choose engine version 8.0.28, (3) choose the free tier template.





Create the RDS instance: (4) configure DB credentials.





Create the RDS instance: (5) place the DB in the VPC, (6) assign it the private subnet group, (7) block public access.

A screenshot of a computer

Description automatically generated



Create the RDS instance: (8) assign the DB Server security group, (9) choose Availability Zone A.

A screenshot of a computer

Description automatically generated

The RDS instance has successfully been created.

A screenshot of a computer

Description automatically generated with medium confidence



After setting up phpMyAdmin on the Bastion/Web Server instance, access the database server and create a table named ‘photos’ with the above structure. The SQL statement to create the table is:

CREATE TABLE photos (

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

title VARCHAR(255), description VARCHAR(255),

date DATE, keywords VARCHAR(255), s3\_photo VARCHAR(255)

);

* 1. Network ACL

A screenshot of a computer

Description automatically generated

Create the Network ACL for public subnet 2 with the above inbound rules.

Rule 1, 2, 3 allow HTTP, HTTPS, and SSH traffic from anywhere to reach the subnet.

Rule 5 allows ICMP traffic from the Test Instance to reach the subnet.

Rule 6 allows all TCP traffic from the Database Server to reach the subnet. This lets the database reach the web server.

Rule 7 allows all TCP traffic from the Test Instance to reach the subnet. This lets the Test Instance reach the bastion/web server instance.

A screenshot of a computer

Description automatically generated

Allow all outbound traffic because this does not affect security.

1. Functional requirements of Photo Album website
   1. Photo storage (S3)

A screenshot of a computer

Description automatically generated

Create an S3 bucket in the us-east-1 region to store photos.

A screenshot of a computer

Description automatically generated

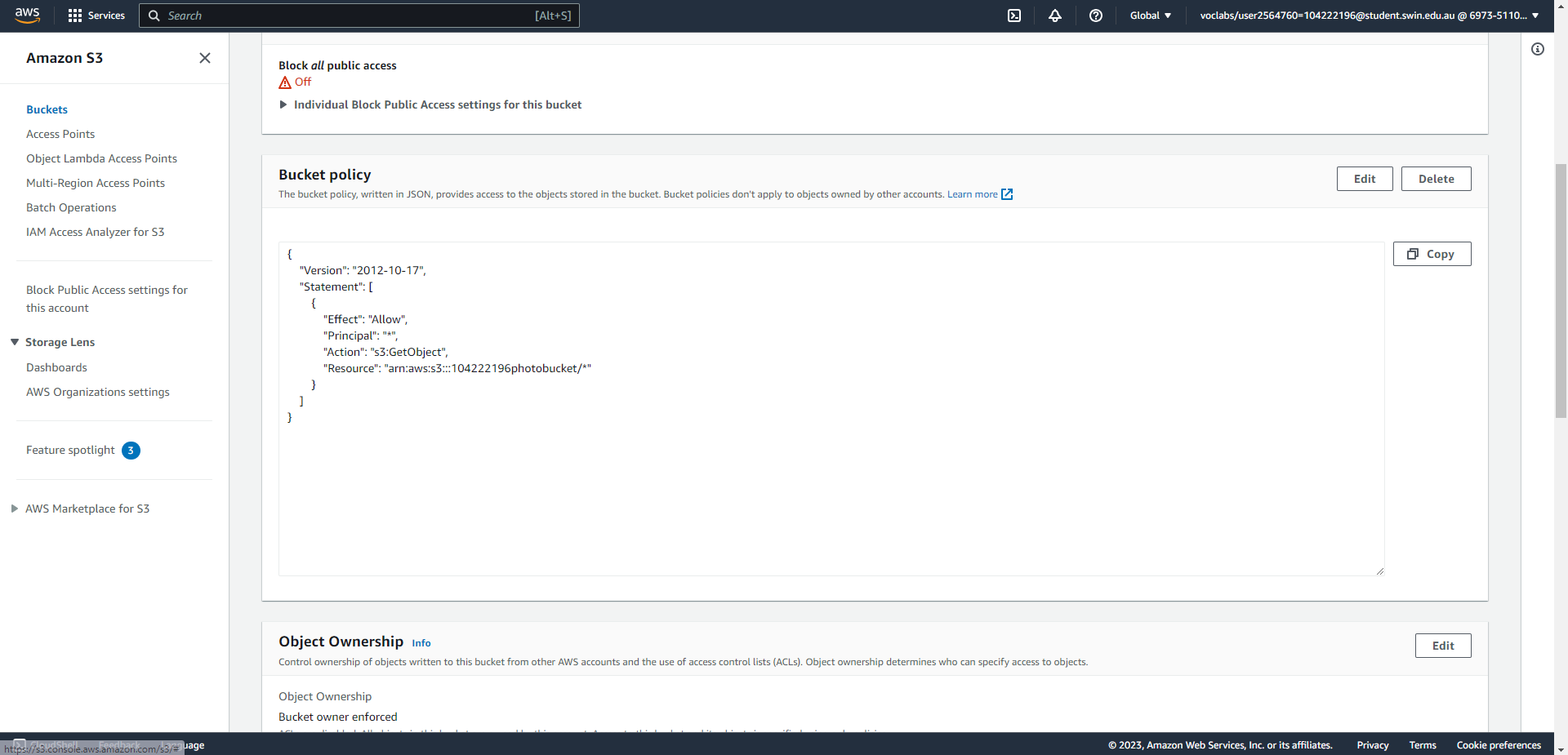


Allow public access to the bucket and its content.

A screenshot of a computer

Description automatically generated

Upload 2 photos to the S3 bucket.





Create an access policy that allows anyone to access the bucket objects. Effect “Allow” grants permission to perform the action. Principle “\*” lets anyone perform the action. Action “s3:GetObject” restricts the action to only reading the objects. Resource “arn:aws:s3:::104222196photobucket/\*” indicate that every object in this bucket can be read.

* 1. Photo meta-data in RDS Database

A screenshot of a computer

Description automatically generated with medium confidence

Add 2 records corresponding to the images on S3 into the ‘photos’ table. This is performed with the INSERT INTO query.

* 1. Photo Album website functionality

A screenshot of a computer

Description automatically generated with medium confidence

After making the necessary modifications in the constants.php file and uploading the source code to the Web Server, the page is accessible at <http://ec2-44-216-98-59.compute-1.amazonaws.com/cos80001/photoalbum/album.php>