

Assignment-1B Creating and deploying Photo Album website onto a simple AWS infrastructure Cos20019 | Cloud computing architecture

Name: Nur E Siam

Student ID: 103842784

Tutorial Time: Thursday 6.30PM

Date of Submission: 17-09-2023

Introduction

The Photo Album website leverages AWS's vast infrastructure, offering developers an array of resources from establishing a VPC for secure web application deployment to integrating an S3 bucket for photo storage. AWS's networking capabilities encompass diverse elements, such as implementing Network Access Control Lists (NACLs) and Security Groups, assimilating Amazon RDS, and orchestrating routing tables. To ensure encrypted communication within the VPC, there's an SSH linkage between instances in both public and private subnets.

This framework underlines the significance of security, scalability, and high availability, all paramount for modern cloud applications. The extensive nature of this project serves as an invaluable resource for AWS enthusiasts, shedding light on the intricacies of VPC design, strategic network division, and safeguarded data handling.

Keywords: VPC, NACL, Security Group, Routing Table, SSH, RDS.

For a hands-on understanding and visualization of this setup, the Photo Album website can be accessed via the provided link.

Website Link:

<http://ec2-44-220-13-146.compute-1.amazonaws.com/cos20019/photoalbum/photoalbum/album.php>

IMPLEMENTATION STEPS

Step-1: Create VPC

1. Create VPC select VPC only and allocate to 10.0.0.0/16.

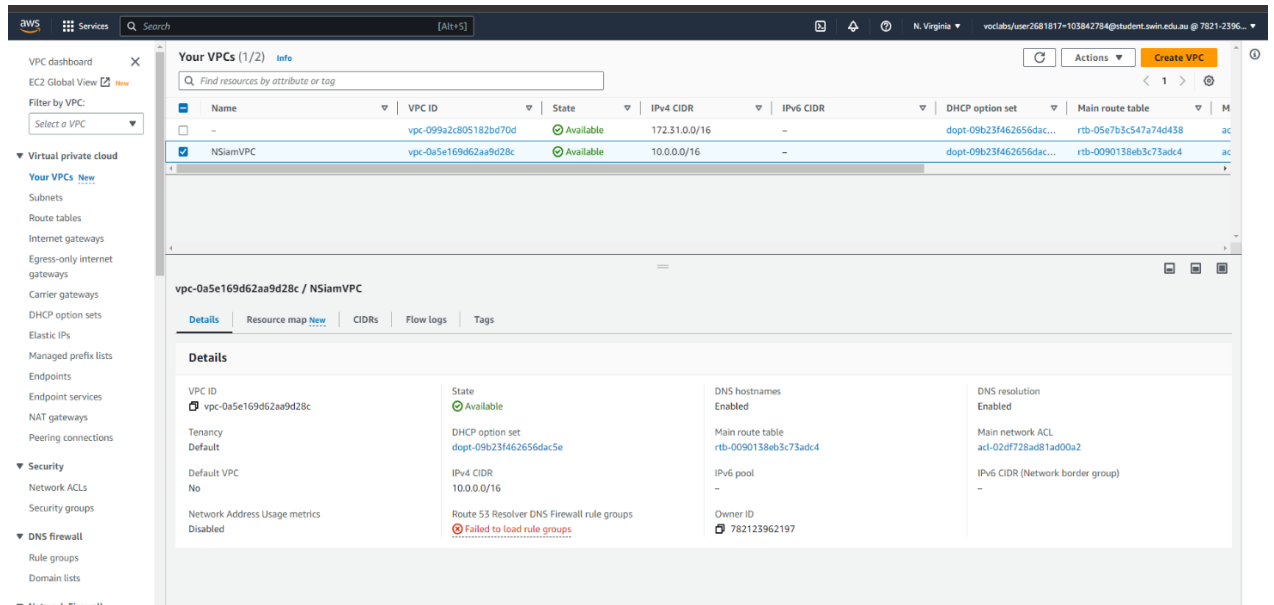


Figure 1-create VPC.

2. Setting VPC, enable DNS resolution and hostnames.

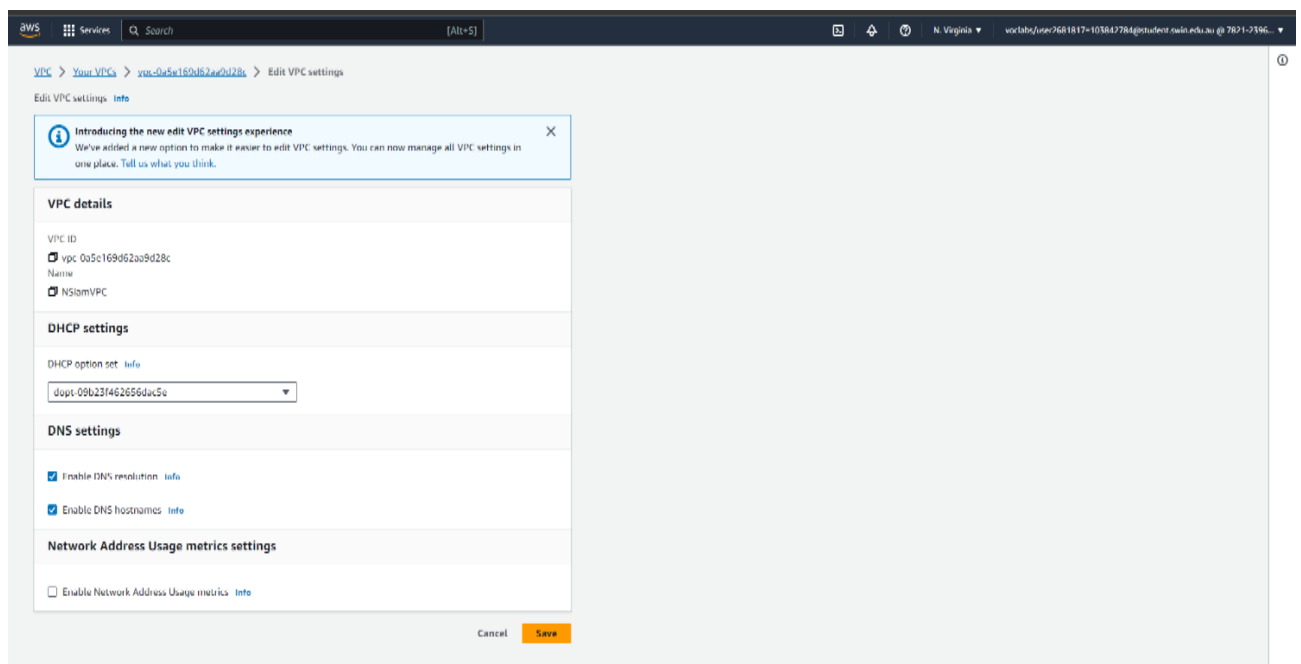


Figure 2-Adding VPC DNS

Step2: Create subnets.

Subnet	Ip	Availability Zone
Public Subnet 1	10.0.1.0/24	Us-east-1a
Private Subnet 1	10.0.3.0/24	Us-east- 1a
Public Subnet 2	10.0.2.0/24	Us-east-1b
Private Subnet 2	10.0.4.0/24	Us-east-1b

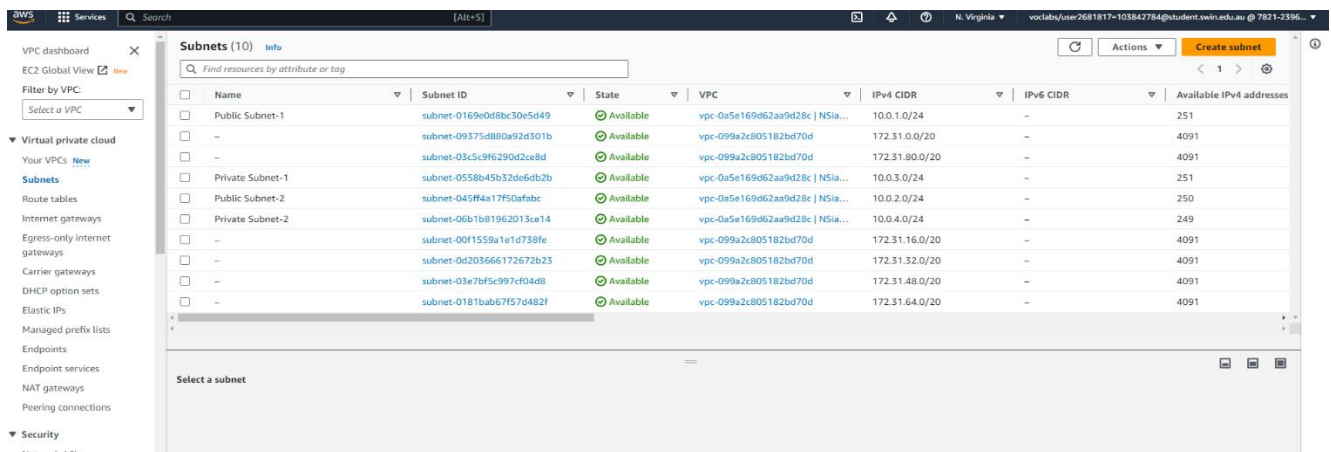


Figure 3-Public & Private Subnets

Step3: Create an internet gateway and attach it to the VPC.

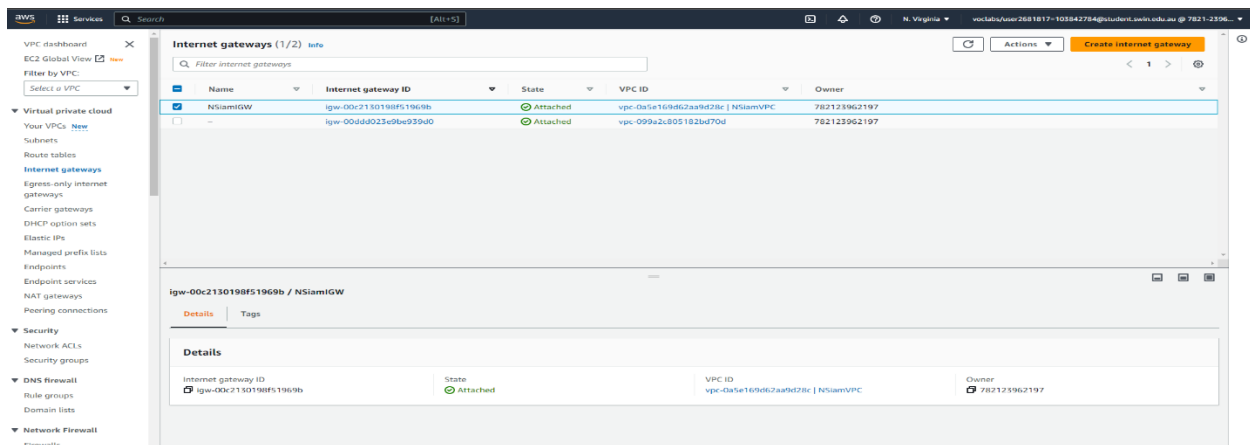


Figure 4-Internet Gateway attached to VPC.

Step4: Create 2 routing tables.

1. Public routing table

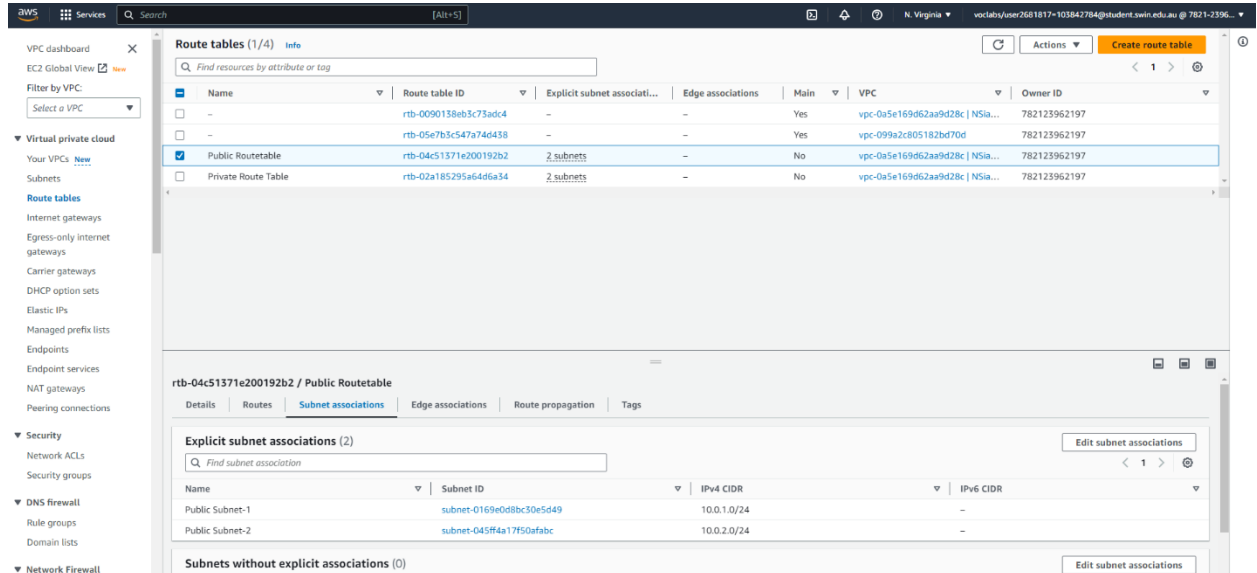


Figure 5-Public Routing table

2. Private routing table

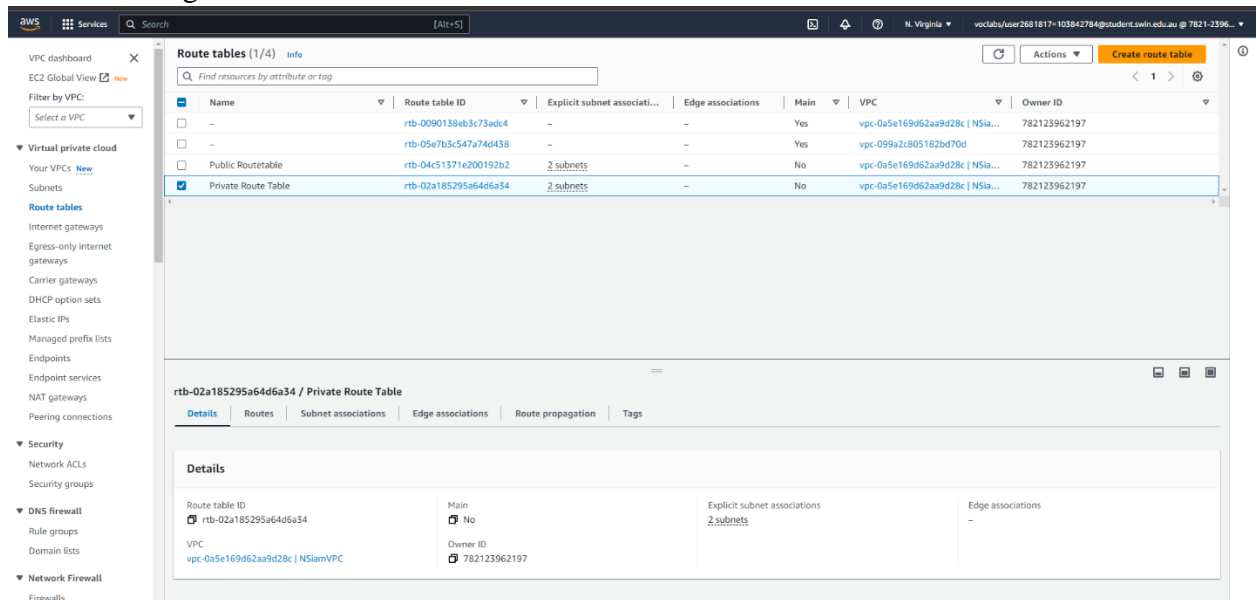


Figure 6-Private Routing table

3. Edit route of the Public route table to internet gateway and assign destination to 0.0.0.0/0

The screenshot shows the AWS Management Console interface for the 'Route tables' section. The 'Public Routetable' is selected, and its routes are displayed. The route for destination 0.0.0.0/0 is highlighted, showing it is associated with the 'igw-00c2130198f51969b' internet gateway.

Name	Route table ID	Explicit subnet associati...	Edge associations	Main	VPC	Owner ID
Public Routetable	rtb-04c51371e200192b2	2 subnets	-	No	vpc-0a5e169d62aa9d28c NSia...	782123962197

Destination	Target	Status	Propagated
0.0.0.0/0	igw-00c2130198f51969b	Active	No
0.0.0.0/16	local	Active	No

Figure 7-Edit routes on Public Route table.

Step5: Create 3 security groups.

The screenshot shows the AWS Management Console interface for the 'Security Groups' section. Three security groups are listed: DBServerSG, WebServerSG, and TestInstanceSG. The 'WebServerSG' is selected, and its details are shown, including its VPC ID and description.

Name	Security group ID	Security group name	VPC ID	Description	Owner	Inbound rules count	Outbound rules count
DBServerSG	sg-002911e28e2c31d02	DBServerSG	vpc-0a5e169d62aa9d28c	Security group for DBS...	782123962197	1 Permission entry	1 Permission entry
WebServerSG	sg-038534cff5a145304	WebServerSG	vpc-0a5e169d62aa9d28c	Security group for We...	782123962197	3 Permission entries	1 Permission entry
TestInstanceSG	sg-0b64993e20baab1cc	TestInstanceSG	vpc-0a5e169d62aa9d28c	Security group for Test...	782123962197	1 Permission entry	1 Permission entry

Security group name	Security group ID	Description	VPC ID
WebServerSG	sg-038534cff5a145304	Security group for WebServerSG	vpc-0a5e169d62aa9d28c

Figure 8-creating 3 security group.

Step 6: create resources for use when launching web server.

4. Public Elastic IP

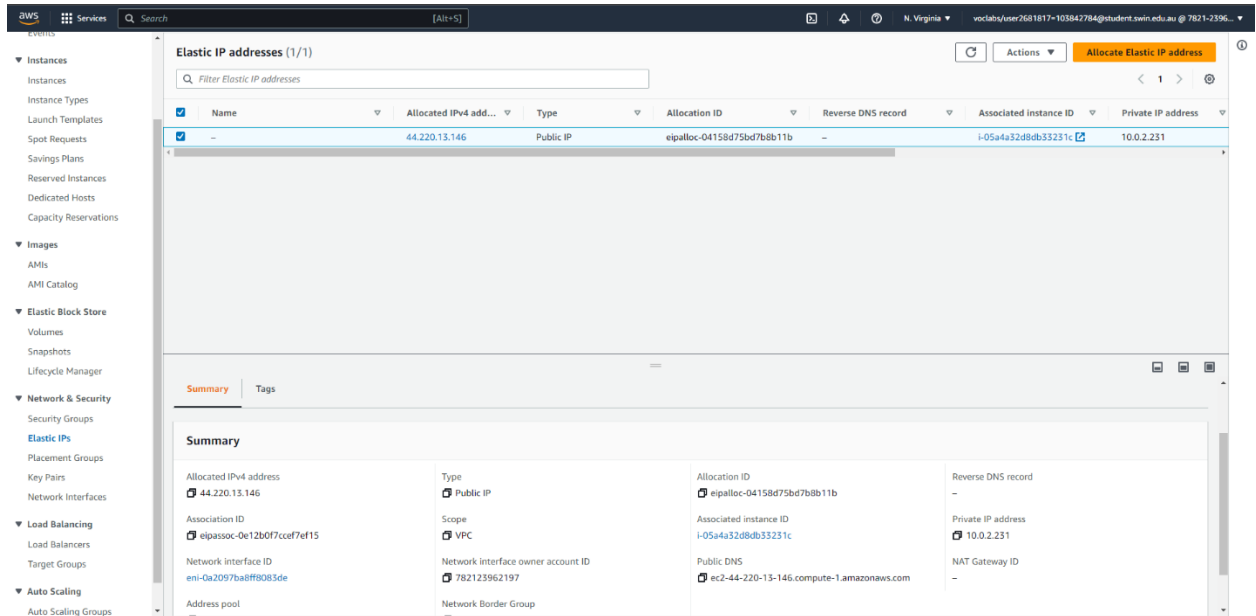


Figure 9-Elastic Ip

Step 7: launch the instances.

1. Create the Web Server instance in Public Subnet 2 and associate the Elastic IP.

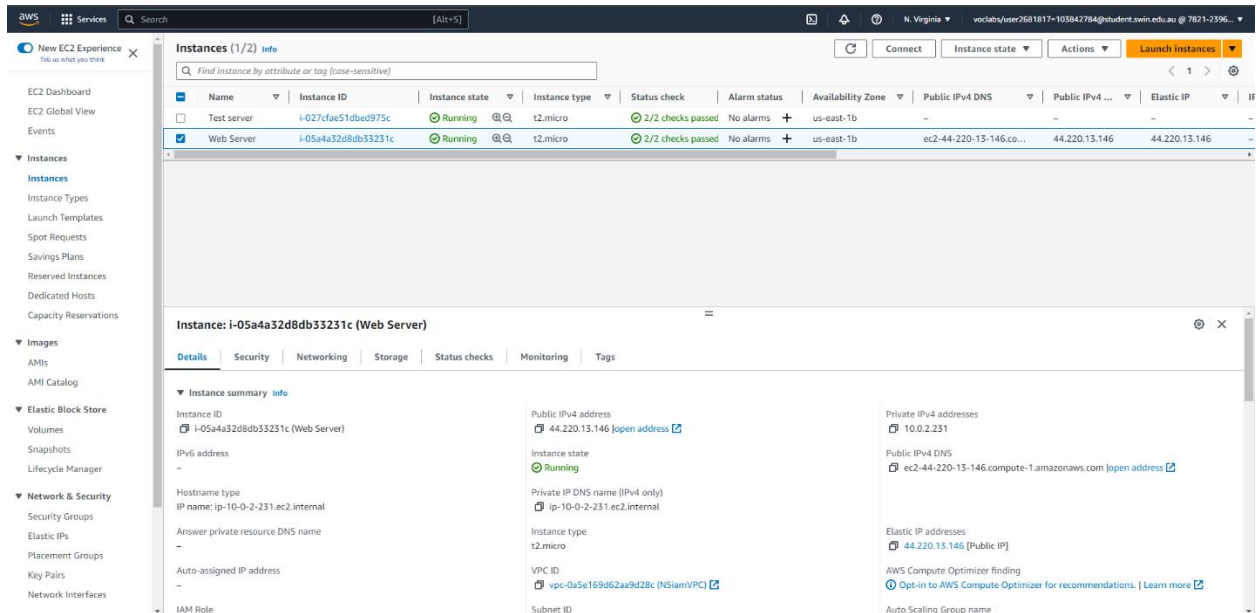


Figure 10-Web Server Instance & Elastic IP.

2. Create the Test server instance in Private Subnet 2

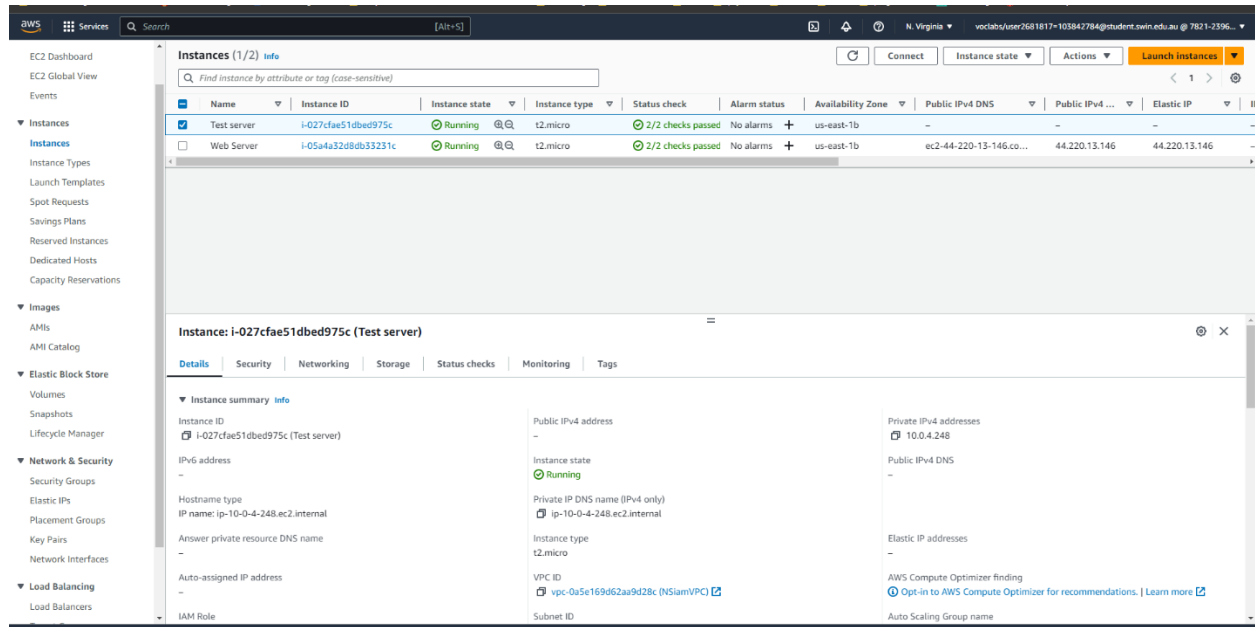


Figure 11-Test instance in Private subnet 2

3. Check that PHP is installed check that PHP is installed

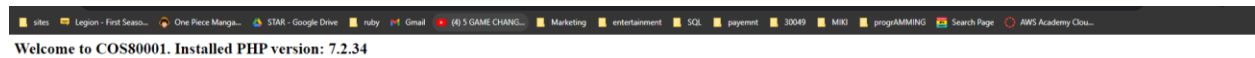


Figure 12- access phpinfo.php to test php functionality

Step 8: Login as ec2-user and SSH to Test Instance using its private IP address, and ping from Test instance to private IP address of Web server.

The screenshot displays the AWS Management Console interface. At the top, the 'Instances' page is active, showing a list of two EC2 instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
Test server	i-027cfae51dbed975c	Running	t2.micro	2/2 checks passed	0 in alarm	us-east-1b	-	-	-
Web Server	i-05a4a32d8db33231c	Running	t2.micro	2/2 checks passed	0 in alarm	us-east-1b	-	-	-

Below the table, the 'Web Server' instance is selected, and its details are shown. The 'Instance summary' tab is active, displaying the following information:

- Instance ID: i-05a4a32d8db33231c (Web Server)
- IPv6 address: -
- Hostname type: -
- IP name: ip-10-0-2-231.ec2.internal
- Answer private resource DNS name: -
- Auto-assigned IP address: -
- IAM Role: -
- Instance state: Running
- Private IP DNS name (IPv4 only): ip-10-0-2-231.ec2.internal
- Instance type: t2.micro
- VPC ID: vpc-0a5e169d62a9d28c (NSIAMVPC)
- Subnet ID: -

A terminal window is open, showing a successful SSH login as 'ec2-user' to the 'Web Server' instance. The terminal output includes the Amazon Linux 2023 logo and a series of ping commands from the 'Test server' to the 'Web Server' private IP address (10.0.2.231), all of which are successful.

```

Warning: Permanently added '10.0.2.231' (ECDSA) to the list of known hosts.
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-10-0-2-231 ~]$ ping 10.0.2.231
PING 10.0.2.231 (10.0.2.231): 56(84) bytes of data:
64 bytes from 10.0.2.231: icmp_seq=1 ttl=255 time=0.509 ms
64 bytes from 10.0.2.231: icmp_seq=2 ttl=255 time=0.545 ms
64 bytes from 10.0.2.231: icmp_seq=3 ttl=255 time=0.405 ms
64 bytes from 10.0.2.231: icmp_seq=4 ttl=255 time=0.536 ms
64 bytes from 10.0.2.231: icmp_seq=5 ttl=255 time=0.566 ms
64 bytes from 10.0.2.231: icmp_seq=6 ttl=255 time=0.587 ms
64 bytes from 10.0.2.231: icmp_seq=7 ttl=255 time=0.584 ms
64 bytes from 10.0.2.231: icmp_seq=8 ttl=255 time=0.596 ms
64 bytes from 10.0.2.231: icmp_seq=9 ttl=255 time=0.587 ms
64 bytes from 10.0.2.231: icmp_seq=10 ttl=255 time=0.643 ms

```

Figure 13- login as EC2 user and Ping

Step 9: Configuring RDS

- In RDS, create a subnet group and allocate two private subnet and create the RDS data base.
- The username = admin & password = admin1234
- RDS END Point = rds-instance.cfg1su8spy97.us-east-1.rds.amazonaws.com
- Install PhpMyAdmin on EC2 and change the file name from config.sample.inc.php to config.inc.php and then from inside the file change the localhost to the RDS end point.

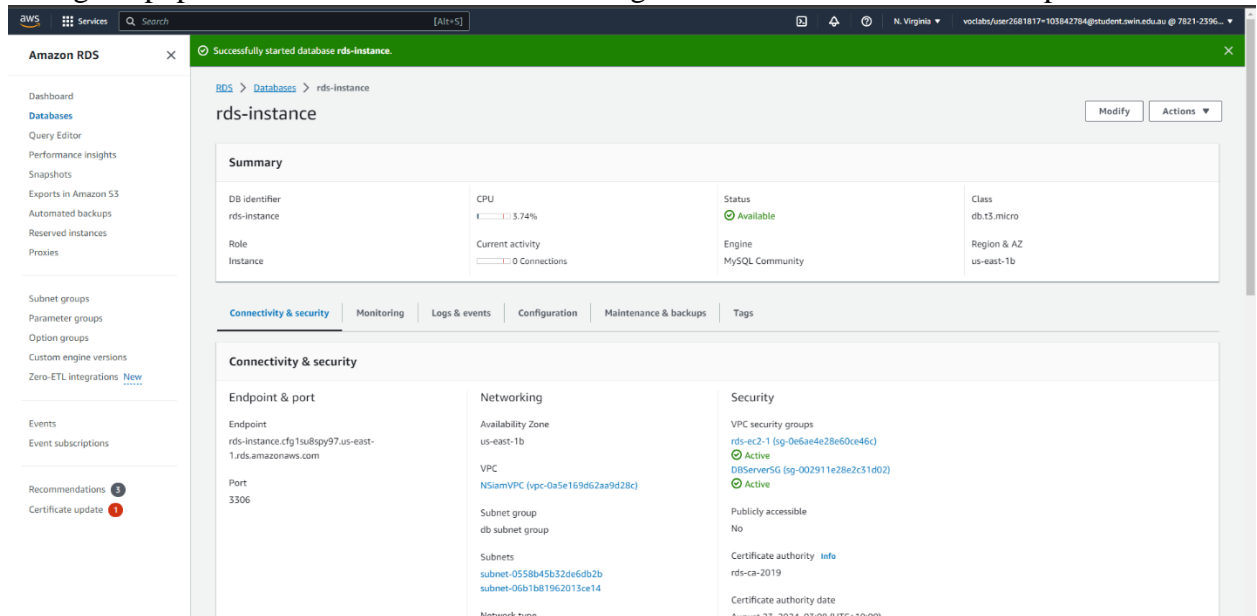


Figure 14- Database RDS



Figure 15-replacing the local host to RDS end point

Step 10: Go to <http://ec2-44-220-13-146.compute-1.amazonaws.com/phpmyadmin>.

- Create a table named “photos”. Add 5 columns to the table.

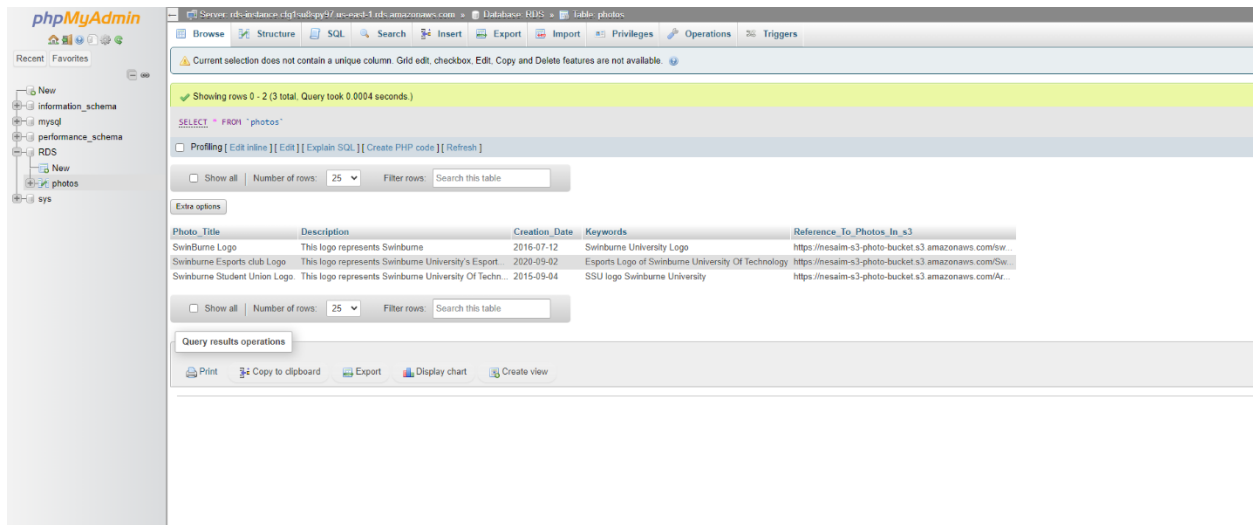


Figure 16-Table creation

Step 11: Creating Network ACL

- In Network ACL Inbound rules and Outbound rules should be the same and attach it to public subnet 2.

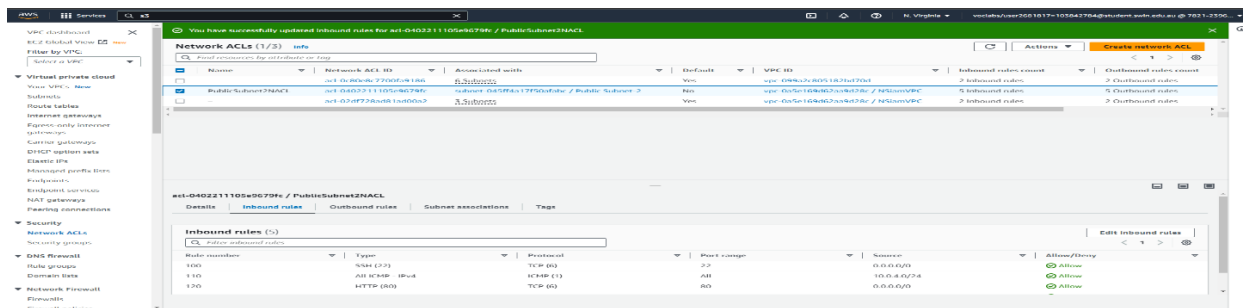


Figure 17-Network ACL with Inbound rules

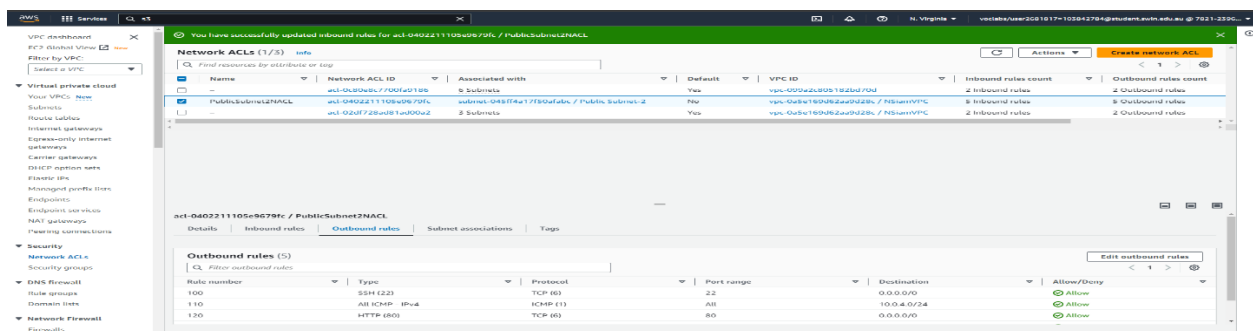


Figure 18-Newtwork ACL with Outbound rules

Step 12: Create S3 Bucket

- Specify Public access and configure the bucket policy.
- Upload the images and copy the URL

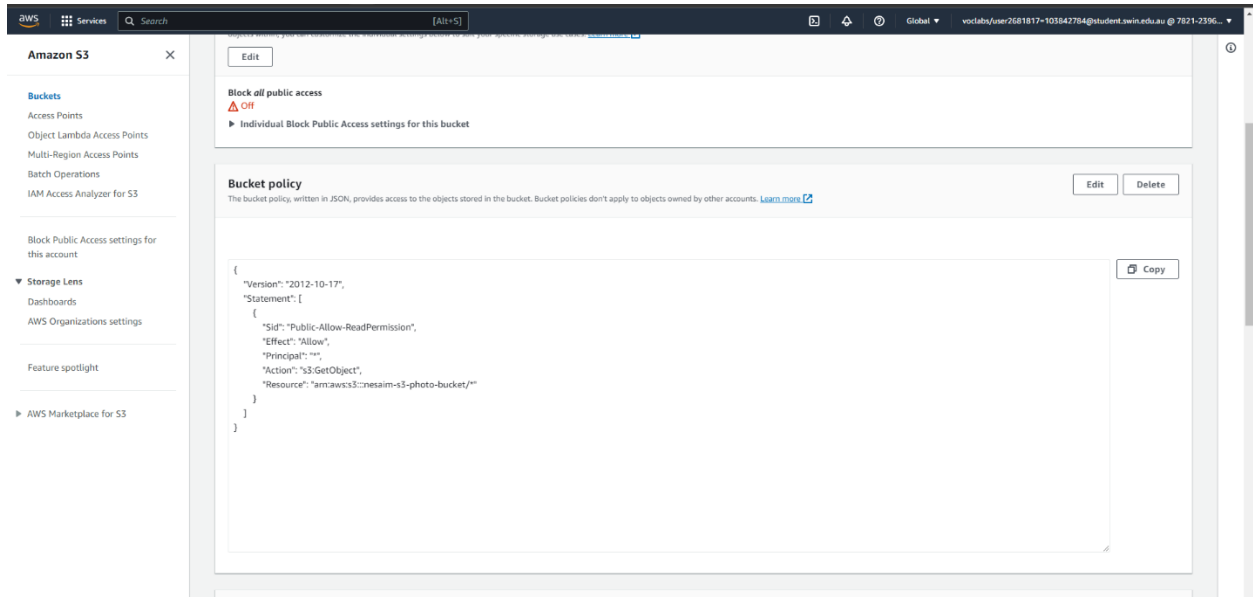


Figure 19-S3 bucket policy and public access enable.

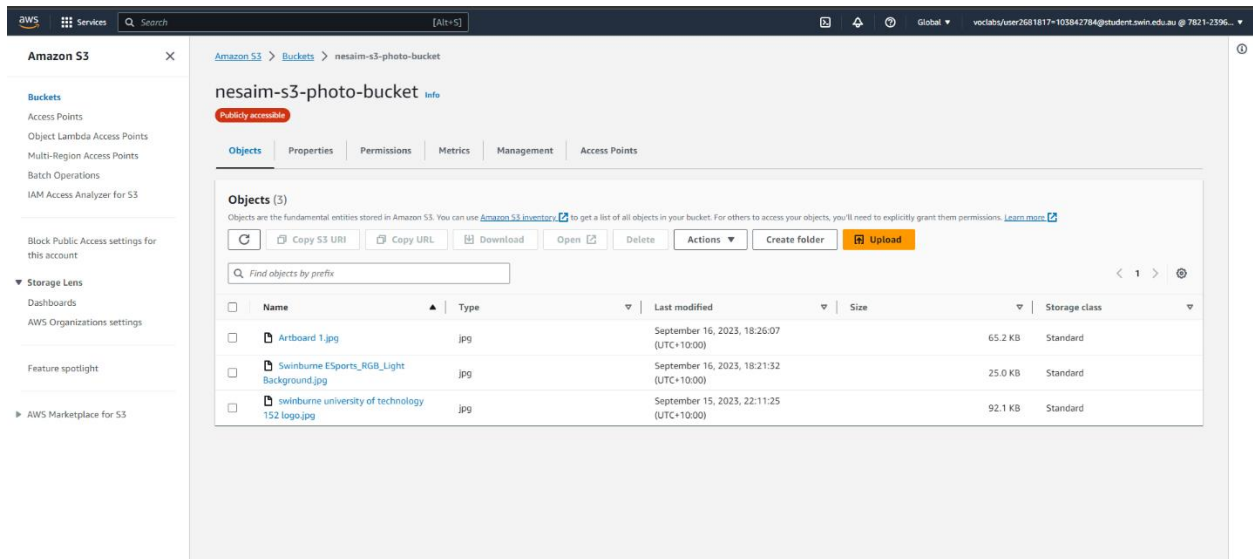
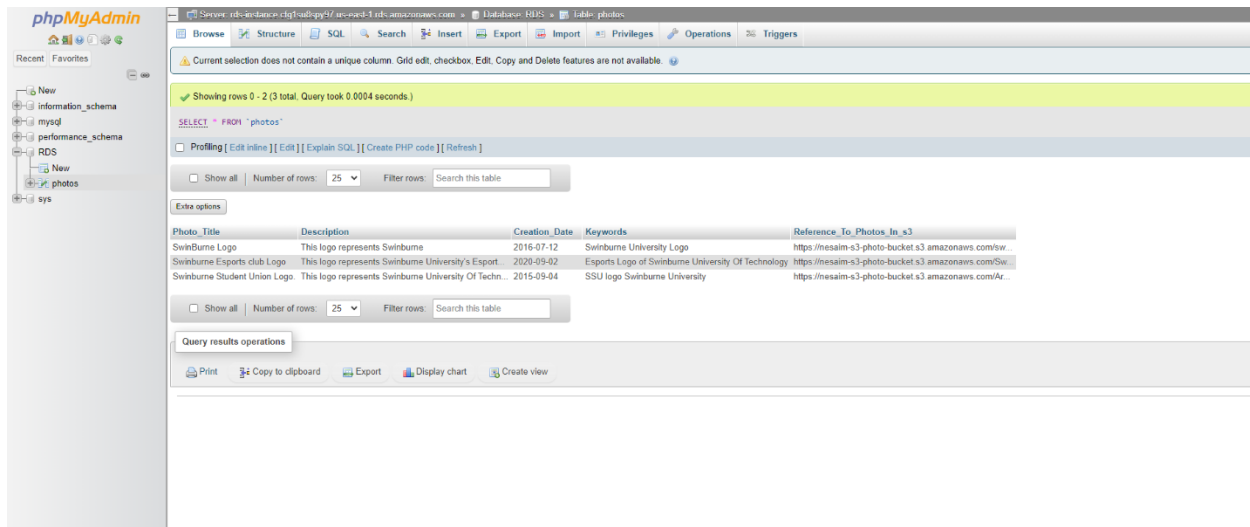


Figure 20-upload the photos

Step 13: In PhpMyAdmin, insert new roles with photos' URLs in photos table.



Photo_Title	Description	Creation_Date	Keywords	Reference_To_Photos_In_s3
Swinburne Logo	This logo represents Swinburne	2015-07-12	Swinburne University Logo	https://nesaim-s3-photo-bucket.s3.amazonaws.com/sw...
Swinburne Esports club Logo	This logo represents Swinburne University's Esport...	2020-05-02	Esports Logo of Swinburne University Of Technology	https://nesaim-s3-photo-bucket.s3.amazonaws.com/Sw...
Swinburne Student Union Logo	This logo represents Swinburns University Of Techn...	2015-09-04	SSU logo Swinburne University	https://nesaim-s3-photo-bucket.s3.amazonaws.com/Ar...

Figure 21-Table with the picture links

Step 14: In photoalbum_v3, enter the correct code based on the resources in AWS.

```
constants.php
He > 20019] > assignment 1b > photoalbum_v3.0 > photoalbum_v3.0 > constants.php
26 * The values of the constant variables with "[ACTION REQUIRED]" in the comment must be updated. The current values are just examples.
27 * You need to replace the values of those constant variables with values specific to your setup.
28 *
29 *
30 * ----- READ THE ABOVE !!! -----
31 * ----- READ THE ABOVE !!! -----
32 * ----- READ THE ABOVE !!! -----
33 */
34
35 // [ACTION REQUIRED] your full name
36 define('STUDENT_NAME', 'Nur E Siam');
37 // [ACTION REQUIRED] your Student ID
38 define('STUDENT_ID', '103842784');
39 // [ACTION REQUIRED] your tutorial session
40 define('TUTORIAL_SESSION', 'Thursday 6:30PM');
41
42 // [ACTION REQUIRED] name of the S3 bucket that stores images
43 define('BUCKET_NAME', 'nesaim-s3-photo-bucket');
44 // [ACTION REQUIRED] region of the above bucket
45 define('REGION', 'us-east-1');
46 // no need to update this const
47 define('S3_BASE_URL', 'https://'.BUCKET_NAME.'.s3.amazonaws.com/');
48
49 // [ACTION REQUIRED] name of the database that stores photo meta-data (note that this is not the DB identifier of the RDS instance)
50 define('DB_NAME', 'RDS');
51 // [ACTION REQUIRED] endpoint of RDS instance
52 define('DB_ENDPOINT', 'rds-instance.cf8is08py97.us-east-1.rds.amazonaws.com');
53 // [ACTION REQUIRED] username of your RDS instance
54 define('DB_USERNAME', 'admin');
55 // [ACTION REQUIRED] password of your RDS instance
56 define('DB_PWD', 'admin1234');
57
58 // [ACTION REQUIRED] name of the DB table that stores photo's meta-data
59 define('DB_PHOTO_TABLE_NAME', 'photos');
60 // the table above has 5 columns:
61 // [ACTION REQUIRED] name of the column in the above table that stores photo's titles
62 define('DB_PHOTO_TITLE_COL_NAME', 'Photo_Title');
63 // [ACTION REQUIRED] name of the column in the above table that stores photo's descriptions
64 define('DB_PHOTO_DESCRIPTION_COL_NAME', 'Description');
65 // [ACTION REQUIRED] name of the column in the above table that stores photo's creation dates
66 define('DB_PHOTO_CREATIONDATE_COL_NAME', 'Creation_Date');
67 // [ACTION REQUIRED] name of the column in the above table that stores photo's keywords
68 define('DB_PHOTO_KEYWORDS_COL_NAME', 'Keywords');
69 // [ACTION REQUIRED] name of the column in the above table that stores photo's links in S3
70 define('DB_PHOTO_S3REFERENCE_COL_NAME', 'Reference_To_Photos_In_s3');
71 >>
```

Figure 22-Fixing the codes in constants.php

Step 15: In WINSCP, create a path of directory where photoalbum is hosting the website.

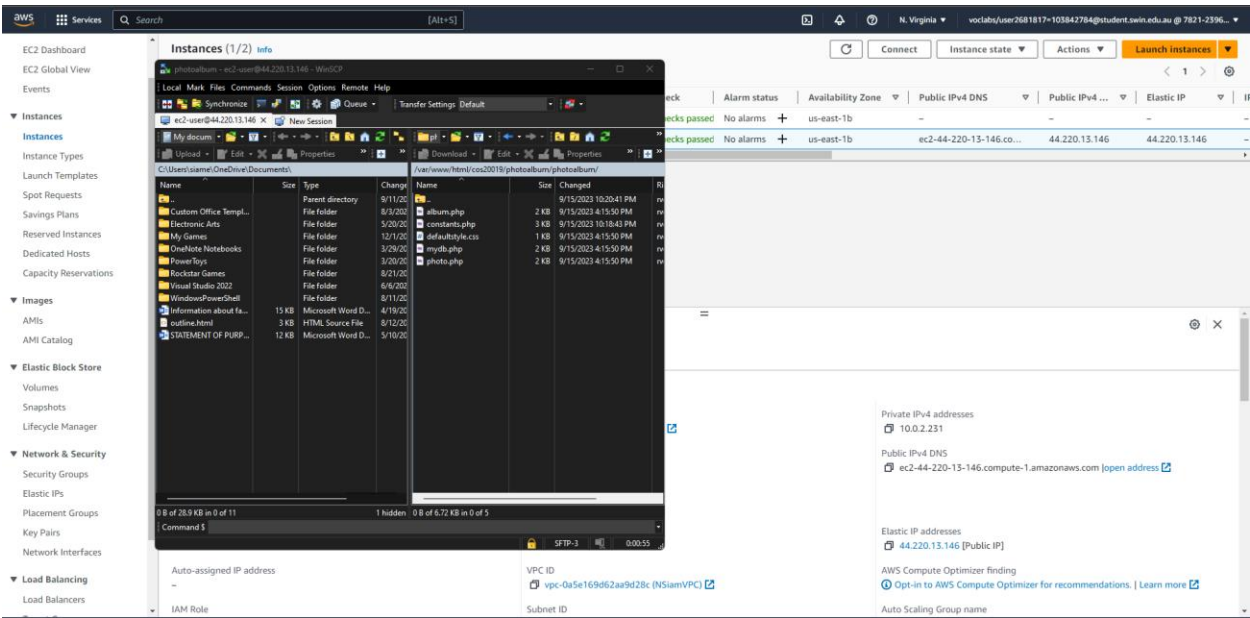


Figure 23-WINSCP uploading the photoalbum file with corrcet constants.php code

Step 16: Access my newly created website which is its database in cloud.

Website link:

<http://ec2-44-220-13-146.compute-1.amazonaws.com/cos20019/photoalbum/photoalbum/album.php>

Student name: Nur E Siam

Student ID: 103842784

Tutorial session: Thursday 6:30PM

Uploaded photos:


Photo	Name	Description	Creation date	Keywords
	SwinBurne Logo	This logo represents Swinburne	2016-07-12	Swinburne University Logo

Figure 24-Final product the Website

Step 17: Testing

- Add more photos.
- And do a ping to web server again to test Network ACL.

Student name: Nur E Siam

Student ID: 103842784

Tutorial session: Thursday 6:30PM

Uploaded photos:




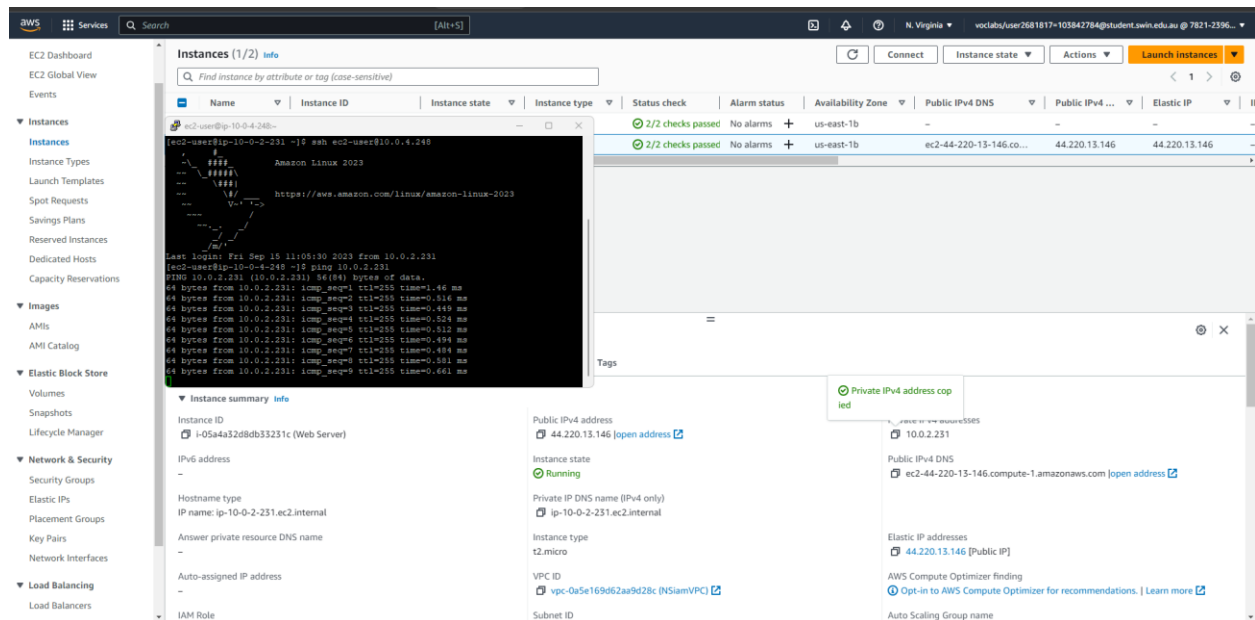
Photo	Name	Description	Creation date	Keywords
	SwinBurne Logo	This logo represents Swinburne	2016-07-12	Swinburne University Logo
	Swinburne Esports club Logo	This logo represents Swinburne University's Esports club	2020-09-02	Esports Logo of Swinburne University Of Technology
	Swinburne Student Union Logo	This logo represents Swinburne University Of Technology's Student Union	2015-09-04	SSU logo Swinburne University

Figure 25- Final website with more photos.



The screenshot displays the AWS Management Console interface. On the left, the navigation menu includes sections for EC2 Dashboard, Events, Instances, Images, Elastic Block Store, Network & Security, and Load Balancing. The main content area shows the 'Instances (1/2)' page. A terminal window is open, displaying the output of a ping command from an EC2 instance to the public IP address 44.220.13.146. The terminal output shows successful ping results with response times ranging from approximately 0.146 ms to 0.661 ms. The instance details on the right indicate that the instance is in a 'Running' state, has a public IPv4 address of 44.220.13.146, and is using the t2.micro instance type.

Figure 26- Ping after Network ACL is done with more photos being uploaded.