Final Project Instructions

Open note, open tech, but not open friend. Do the project on your own. Earn your grade.

For the Final Project, you will **build out a solution like the one you (should have) designed for your Group Midterm Project and ones we've examined in the lab exercises**.

To accomplish this, you will work in the AWS Management Console from your AWS Educate account. See the announcement [I'm creating each of you an individual AWS account (with $100 credit in it) that you can use all semester long for this class or another class](#) on Canvas.

I want you in the **N. Virginia region**, so if your account puts you in a different region, select **N. Virginia** from the drop down to the right of your userid in the black banner across the top of the AWS Management Console.

You will upload a collection of screenshots through Canvas that documents what you were able to accomplish. **You can upload a single file with all the screenshots in it or a zip file containing individual screenshots in it. Each screenshot <u>must</u> include the black banner across the top of the AWS Management Console which includes your unique login ID. NO CREDIT FOR SCREENSHOTS THAT DO NOT INCLUDE YOUR LOGIN ID.**

Alternatively, you can create a well-documented CloudFormation template to build the environment. Be sure to read the instructions to make sure you've considered everything, not just the diagram. If you choose to complete the assignment this way, you will submit your JSON or YAML template instead of the screenshots.

This is the environment you are mostly going to build. **If you can't do all of it, please do some of it so you earn partial credit. If you've been doing the lab exercises, this will have the feel of a long lab exercise.**
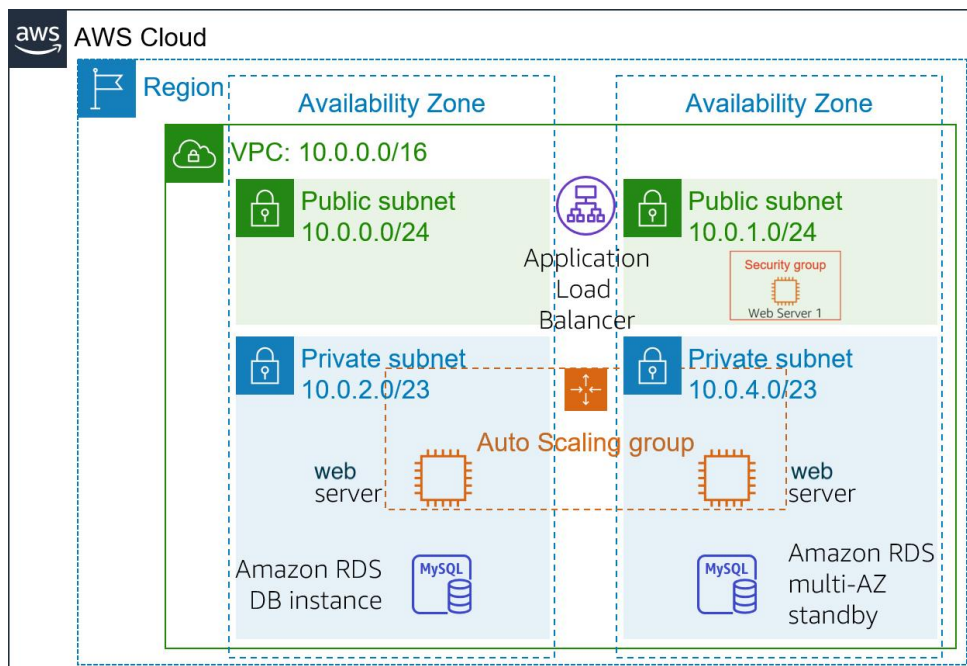
To receive full credit for your project, you should accomplish and document the following:

- **IAM**: Assign IAM users to proper groups
    - Create the users Alice, Bob, Charlie, Debbie, Earl, Frank
    - Create the groups:
        - SysAdmins (AdministratorAccess policy)
        - DBAdmins (AmazonRDSFullAccess policy)
        - DevOps (No policy yet)
    - Assign Alice and Bob to SysAdmins
    - Assign Charlie and Debbie to DBAdmins
    - Assign Earl and Frank to DevOps
    - **For each group, screenshot the Users and Permissions tabs**

- **VPC**: Allocate an Elastic IP address from Amazon's pool of IPv4 addresses and **screenshot the list of Elastic IP addresses**

- **VPC**: Create a VPC named **FinalVPC** that looks like the diagram. Pay attention to the subnet address ranges. YOU WON'T BE ABLE TO USE THE VPC WIZARD SINCE IT TRIES TO ALLOCATE A NAT GATEWAY AND THAT ISN'T SUPPORTED YET IN AWS EDUCATE ACCOUNTS.
    - Create the VPC
    - Create an Internet Gateway
    - Create the 4 subnets. Name them sensibly.
    - Create a public route table (hint: that will include an Internet Gateway route), associate with public subnets
    - Create a private route table, associate with private subnets

- **VPC**: **Screenshot the list of VPCs and the list of subnets**

- **EC2**: Launch an EC2 instance into your second PUBLIC subnet and tag it with the name **TigerEC2**
    - Instance type t2.micro, Amazon Linux 2 AMI (HVM), SSD Volume Type
    - Enable Auto-assign Public IP
    - Use this user data to bootstrap your instance:

```
#!/bin/bash
# Install Apache Web Server and PHP
yum install -y httpd mysql php
# Download Lab files
wget https://aws-tc-largeobjects.s3.amazonaws.com/AWS-TC-AcademyACF/acf-lab3-vpc/lab-app.zip
unzip lab-app.zip -d /var/www/html/
# Turn on web server
chkconfig httpd on
service httpd start
```

    - Create 17 GiB storage volume, type gp2, 100 IOPS, encrypted using default aws/ebs keys
    - Create a security group named **Web Security Group** which allows SSH and HTTP traffic
    - You won't need SSH access for this project, but you can create a new key pair and check the acknowledgement box in the keypair dialog box when launching the instance
    - **Screenshot the list of running EC2 instances with the TigerEC2 instance selected**. I want to see the description information for that instance in your screenshot
    - **Screenshot the list of EBS volumes with the TigerEC2 volume selected**. I want to see the description information for that volume in your screenshot

- **EC2**: After your EC2 instance is running, visit the public IP of your EC2 instance using your browser and screenshot the page

- **EC2**: The EC2 instance you created will serve as your baseline web server. Create an image (AMI) named **TigerEC2-Baseline** from your EC2 instance. You will use this AMI to create additional EC2 instances so that they look identical to your baseline web server.

- **EC2**: Now you are going to create TWO (2) new EC2 instances **using the TigerEC2-Baseline image** you created. You will put one of your new EC2 instances in your first PRIVATE subnet and the other one in your second PRIVATE subnet.

- **EC2**: Create an Application Load Balancer with both new EC2 instances in the target group
    - Name it **TigerELB**
    - Be sure to specify the **FinalVPC** VPC

- Include both your PRIVATE EC2 instances in the target group
- **Screenshot the list of load balancers with TigerELB selected**
- **Screenshot the target group and show the Targets information**
- Using your browser, visit the public IP of your Load Balancer and **screenshot the page**
- If you've done everything correctly, you should see the same web page you did when you reached your baseline web server
- If you keep refreshing the page, you should see that your load balancer is routing traffic to both your private EC2 instances

- **DB**: Create a new RDS instance (MySQL) named **ClemsonDB**
  - Create a DB Subnet Group named **CU-Subnets** which includes your two private subnets
  - DB Instance Size: Burstable db.t2.micro; Storage Type: General Purpose
  - Multi-AZ deployment using the **FinalVPC** and the **CU-Subnets** subnet group in the additional connectivity configuration
  - Additional configuration information: No Enhanced Monitoring
  - After your database is up, click its name and <u>**screenshot the Connectivity & Security information and also the Configuration information**</u>

- **EXTRA CREDIT**: Configure EC2 Auto-Scaling behind your load balancer.

Good luck!