**Zachary Linton Capstone Report**

The following are the steps I followed to complete the capstone project, given the mandatory requirements:

1. In order to start the planning phase, I examined the provided environment. I inspected the VPC, Subnets, and Security Groups.
2. I created a MySQL database that would be secure. This database was created in Private Subnet 1. It was a multi-AZ database, so an additional database was created in Private Subnet 2. In this step, a subnet group was created, directing the database creation to the private subnets.
3. I created a t2.micro EC2 instance that would act as a web server. This instance was created in Private Subnet 1. This web server needed to be secure for administrative access, while providing anonymous access to web users. Since the instance was created in a private subnet, a NAT gateway would be necessary in providing the instance access to the internet for possible updates. A NAT gateway was created in each public subnet. Before the creation of this instance, a new key pair was created to attach to this instance. For Secure Shell access to the web server, the administrative user would need two separate key pairs. The first, provided, key pair would allow access to the bastion host and the second, newly created, key pair would allow access to the web server from the bastion host.
4. Once connected to the web server instance, via the bastion host, it was configured as a LAMP server. The project assets were imported (wget <https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/ILT-TF-200-ACACAD-20-EN/capstone-project/Example.zip>), then unzipped. In this step the SQL dump was also done (wget <https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/ILT-TF-200-ACACAD-20-EN/capstone-project/Countrydatadump.sql>).
5. In the Systems Manager Parameter Store, parameters had to be created with values attached. The parameter created were as follows:

* /example/endpoint
* /example/username
* /example/password
* /example/database

1. An AMI of the web server instance was created.
2. I created an Application Load Balancer. A target group was created in this step. The ALB would provide high availability to the web server.
3. I created a launch template that would be used for auto scaling. The Auto Scaling group would be created in the private subnets and attached to the ALB created in the previous step.
4. I went through the security groups and edited rules, limiting access where required. In Inventory-AppSG, modifications to inbound only allowing HTTP access from ALBSG.