**Steps to Complete**:

1. Provision the environment through the terraform modules uploaded to github.
2. Terraform commands are mentioned in the README.md file.
3. Environment provisioned:
   1. VPC
      1. VPC
      2. DHCP Option Set (amazon provided DNS)
      3. Subnets (6 private subnets - based on product roles (web, utility and db) and 2 public subnets)
      4. 1 IGW
      5. 2 NAT Gateways – for each public subnets
      6. 1 VGW
      7. 2 EIPs and its association - for each NAT GW
      8. NACLS for each product roles.
      9. Route tables (1 private route table which is main and 2 route tables for each public subnets)
      10. Route table association (all private subnets and nat gw to main RT and igw to public RT)
   2. Active Directory
   3. Wildcard certificate
   4. Target Group for load balancer -ASG will put the instances in this TG.
   5. Application load balancer
      1. Http listener
      2. Https listener (with target group attached)
         1. Host headed for the DNS will point to this TG.
   6. Route 53 – Publich hosted Zone which has the DNS (CNAME Record) of the website pointing to the DNS of the Application Load Balancer.
   7. Key\_pair – For the EC2 instances.
   8. IAM\_Roles – To attach to the eC2 instances.
   9. RDS – For Oracle Database.
   10. ASG
       1. Launch Configuration
       2. Auto Scaling Group (which scales between 2 and 4 instances based on CPU load)
       3. Auto Sclaing Group Policies
       4. Cloud watch Metric - To monitor the CPU load of the EC2 instances and scale between 2 and 4 based on the threshold met.
       5. Autoscaling Notification – Notification will sent to SNS which will trigger the scaling policies to execute.
   11. SSM\_Doc -To auto join the EC2 instances to domain (created by the AWS Active Directory Service).
   12. Security Group: A standard SG for both EC2 instances and RDS.