Project Name: Deploy-high-availability-web-app-using-CloudFormation

There will be two parts to this project:

- **Diagram**: You'll first develop a diagram that you can present as part of your portfolio and as a visual aid to understand the CloudFormation script.
- **Script (Template and Parameters)**: The second part is to interpret the instructions and create a matching CloudFormation script.

First part Diagram:

High availability (HA) is a component of a technology system that eliminates single points of failure to ensure continuous operations

Another meaning for that my application is running in 2 Availability zones as if one failed the application is still up and running

1-Think what we need from scratch to make the application up & running

You need

1- VPC
2- Internet gateway
3- Routing table
4- 2 availability zones to achieve high availability as I said If anyone failed due to any reason there is another one to serve the traffic
5- Subnets our application will be in a private subnet but how this will make requests to the internet to say install apache or make any upgrade to make this we need public subnet which will do this by configuring nat gateway in it , till now we make the application running but if any issue happened In the subnet in a private or public one this will cause great issues and our users will be impacted so we need to achieve high Availability by creating another 2 subnets one for private and another one for public so we needs 4 subnet
6- Nat gateway

For app level we need:

1- load balancer to distribute the traffic between machines
2- autoscaling group

2-go to www.lucidchart.com and draw the diagram with the above mentioned components

---

The second target is implementing the above components with cloudformation. For this one try to use parameters as you can , don't hardcode the values and use the resources as in the lessons

Requirements from project page:
Launch configuration specs:

- You'll need two vCPUs and at least 4GB of RAM. The Operating System to be used is Ubuntu 18. So, choose an Instance size and Machine Image (AMI) that best fits this spec.
- Be sure to allocate at least 10GB of disk space so that you don't run into issues.

Load balancer:

- Should be public and In public subnet with security group accept traffic from port 80 inbound only from `(0.0.0.0/0)`
- And outbound will be port 80 only to the servers

Servers:

- The application needs to be deployed into private subnets
- Since you will be downloading the application archive from an **S3 Bucket**, you'll need to create an **IAM Role** that allows your instances to use the S3 Service.
- Open port 80 inbound and all outbound in server security group

Output:

- One of the output exports of the **CloudFormation** script should be the public URL of the **LoadBalancer**. **Bonus points** if you add `http://` in front of the load balancer **DNS Name** in the output, for convenience.