Background:
You are recently promoted from a Cloud Engineer to a Cloud Architect and assigned a project to prepare a new environment in the cloud, to which your team will later migrate their applications.

Goal:
To architect a solution that is secure, scalable, highly available, and cost-effective using AWS.

Requirements:

- They are concerned about the security of the environment, so they have decided to virtually isolate their network from the rest of the customers and the rest of the environments in the same AWS Cloud Account

- Due to the budget issue, the company cannot afford a dedicated DB engineer, so they are willing to outsource the DB management from a Cloud provider, to store and maintain the customer information received by PHP application. You must pick the right solution from AWS, which should be a Platform as a Service. It should also provide high availability, patching, and back-ups. (hint: Create DB subnet group)

- And about disaster recovery, you should have enough backups for both the Web and Database server, so if in case the environment crashes, we can launch a new environment from the disaster recovery backups

- Design a dynamic website where the customers can enter their details, which should be stored in a database

- They are uncertain about the traffic pattern that how low or high it can be, so they have a requirement that the environment should be running at least two EC2 servers all time, and when there is a high load, they must burst up to four servers in total

- Now the company cannot afford a dedicated engineer for monitoring, so you must automate the incident management through an event notification. Anytime there is an increase and decrease in the VM's due to high or low traffic, you must receive a notification via email

- The application should be highly available, even if a VM fails to respond to queries, there should be a mechanism to shift the connection to another healthy VM automatically

- Your Dynamic Website should also be cached globally, so users worldwide can access it with less latency. The customer is okay if we get an unfriendly AWS generated URL for accessing the website

Deliverables:

- Configure all the services and create a document that includes the screenshots of the configured services. Upload the document on the LMS