P1.a) For ABC company’s cloud migration of their web servers, here are two non-negotiables:

1. **Data Security and Compliance**: The cloud provider must offer robust security features that comply with industry standards for handling sensitive data, such as PCI DSS, to protect customers' credit card information and personal data.
2. **System Scalability**: Given the nature of e-commerce, the cloud service must provide seamless scalability options to handle traffic surges, especially during peak shopping seasons, without service interruption or degradation.

P2.a) Connect your user to the web server with the arrow tool.

A screenshot of a computer

Description automatically generated

P2.b) Specifications of RDS  
MySQL Engine, Instance type: M5 Standard/Intel, Large size  
  
A screenshot of a computer

Description automatically generated  
  
  
P2.c)

Amazon ElastiCache is a managed caching service that provides scalable, in-memory data stores in the cloud, supporting caching engines such as Redis and Memcached. It is designed to retrieve data from high-throughput, low-latency in-memory data stores, significantly improving the performance of web applications by reducing the load on databases for read-intensive workloads. ElastiCache helps maintain application responsiveness by allowing frequently accessed data to be served quickly without repeated database queries. Additionally, it offers features like data partitioning, replication, and failover to enhance reliability and scalability of application caching mechanisms.  
  
A screenshot of a computer

Description automatically generated

P2.d)

Create two more instances of your EC unit above and add Auto Scaling

A screenshot of a computer

Description automatically generated

P2.e)

Elastic load balancer

A screenshot of a computer

Description automatically generated

P2.f)

The classic load balancer is to sit between your elastic compute and the VPCs

A screenshot of a computer

Description automatically generated

P2.g)

I began by adding a user icon to the edge of my Cloudcraft design to represent the client-side interaction with the e-commerce platform. Then, I added an Elastic Compute (EC2) instance with a Linux-based configuration similar to a T-type instance, which serves as the web server for the e-commerce site.

Next, I connected the user to the web server using an arrow to illustrate the communication pathway. After that, I introduced an Amazon RDS with a MySQL engine and chose an M5-standard large size for a robust and scalable database service without the need to manage database software.

I made sure the specifications of the RDS included features for high availability and automated backups, considering the e-commerce operations' demands. Following that, I added ElastiCache with a Redis engine to cache frequent queries and reduce the load on the RDS, which enhances the performance of the web application.

To isolate and protect the network, I created a VPC named “Cached Database” for the database and cache unit. I also included a customer gateway for secure access to the VPC, ensuring that the web traffic passes through a secure entry point.

I rerouted the compute instance’s connection through the customer gateway before it reached the database, adding an additional layer of security. I expanded the architecture by creating two more EC2 instances and incorporating them into Auto Scaling to handle variable loads. I included a classic load balancer with 10 GB data processing capability to distribute incoming traffic evenly across the server resources.

To ensure high availability, I replicated the entire setup into a second VPC named “Replica” and managed the traffic between the "Cached Database" and "Replica" VPCs using another classic load balancer. This redundancy means that if one VPC encounters issues, the other can seamlessly take over, ensuring uninterrupted service.

Through these steps, I designed a solution that ensures operational continuity, optimizes application response times, securely handles storage, maintains data integrity, and adjusts dynamically to traffic with robust elasticity.

P2.h)

US-EAST-2 (Ohio)  
In Ohio the yearly charges are   
Compute: $2186.50/yr  
Networking: $227.76/yr  
Database: $5729.04/yr

Total: 8143.30

For 3 year term the cost would be 24429.9

US-WEST-1 (California)  
In Ohio the yearly charges are   
Compute: $2606.98/yr  
Networking: $238.27/yr  
Database: $6605.04/yr

Total: 9450.29

For 3 year term the cost would be 28350.87

We can see that in North California the cost of database is higher that Ohio, but the Networking is lower in North California than Ohio.

The cost variations between regions like North California and Ohio can be attributed to several factors including the cost of real estate, power, and local infrastructure which affect how AWS prices its services. Databases may be more expensive in North California due to higher operational costs in this region, while networking costs might be lower potentially due to more efficient or less expensive networking infrastructure available there.

For the ABC company, choosing between these regions should consider not only these cost differences but also factors like proximity to customers, latency, and data residency requirements. If most of their user base is closer to one region or if they require faster data access, the benefits of choosing a region based on these criteria might outweigh the cost differences. Additionally, considering overall business strategy, including long-term cost efficiency and operational performance, is crucial.

P3.

For AWS, Elastic Load Balancing (ELB) is the service that offers several types of load balancers, and one of the prominent products is the Application Load Balancer (ALB). The ALB operates at the application layer, directing traffic across multiple targets like EC2 instances and containers. It can handle complex load balancing requirements, such as routing traffic based on content type, and it can also accommodate changes in incoming traffic, automatically adjusting its capacity to handle various loads. This makes it particularly suitable for modern application architectures like microservices.

For Azure, Azure Load Balancer is a core network offering that includes built-in high availability and network performance for your applications. It works at layer 4 of the OSI model, providing distribution of inbound flows that arrive at the load balancer's front end to backend pool instances. These flows are according to configured load-balancing rules and health probes. The service is zone-redundant, ensuring high availability across the Azure regions.

P4.

Amazon Managed Blockchain is a fully managed service that lets you set up and manage a scalable blockchain network with just a few clicks. It eliminates the overhead required to create the network, and ensures that it easily scales to meet the demands of thousands of applications running millions of transactions. It currently supports the Ethereum and Hyperledger Fabric frameworks, providing a robust infrastructure for building versatile and secure blockchain applications.