Question 1:

Using the load balancer in our website

Question 2:

Put your website on multiple web servers and behind load balancer. Then push update into your web servers one by one, remember to config balancer to not to route request into one being updated. After finish pushing update then config balancer to route request back into it

Question 3:

While all DDoS attacks aim to overwhelm a system with too much activity, hackers have different strategies they rely on to cause a distributed denial of service.

The **three main types** of attack are:

- Application-layer attacks.
- Protocol attacks.
- Volumetric attacks.

The three approaches rely on different techniques, but a skilled hacker can employ all three strategies to overwhelm a single target.

These are the 4 trick I used to save my side

Ensure High Levels of Network Security
Have Server Redundancy
Look Out for the Warning Signs
Continuous Monitoring of Network Traffic

Question 4:

I will use the NoSQL Databases from the unstructured data because there is not proper schema of the unstructured data

Question 5:

For a website handling critical transactional data that requires robust security measures, the following technology stack is often considered one of the best choices:

LAMP Stack with Enhancements:

Linux (Operating System)

Apache (Web Server)

MySQL or MariaDB (Database Management System)

PHP (Server-side Scripting Language)

Enhancements and additional security measures, such as SSL/TLS for encryption, strong authentication, and security-focused coding practices, would be integrated into this stack to ensure robust security. This stack is widely used and has a strong track record for handling secure transactions.

Question 6:

Geographic Analysis: Use geolocation data to analyze the distribution of your audience. Identify regions with high user density in both Pakistan and the USA.

Content Delivery Network (CDN): Utilize a global CDN service like AWS CloudFront, Akamai, or Cloudflare. These CDNs have a vast network of servers worldwide and can automatically route users to the nearest edge server.

DNS Load Balancing: Implement DNS-based load balancing using a service like Amazon Route 53 or a similar DNS provider. Configure geo-routing policies that direct users to the nearest server based on their geographic location.

Server Deployment: Set up server clusters in data centers strategically located in Pakistan and the USA. Ensure redundancy for high availability and failover.

Latency Monitoring: Continuously monitor network latency and server response times from different geographic regions to fine-tune routing policies.

Content Synchronization: Ensure that data and content are synchronized between servers in both locations to maintain consistency.

Question 7:

For handling the static content on your website I will used the react because this is the best framework for the frontend