Context-1

Q). How would you test if the cart counter is syncing correctly across different pages?

Answer:

I would like to test following-

- Open the website and add one item to the cart on a product listing page. Verify the cart counter updates to "1"
- Navigate to several other key pages
  - ✓ Homepage
  - ✓ Product Detail Page
  - ✓ Checkout page
  - ✓ Contact Us page

  And observe the cart counter on each page to confirm it still displays "1".
- On a different page add another item. The counter should update to "2". Repeat the cross-page verification process to ensure the updated count is reflected everywhere.
- Log in to the same account on a different browser or device. The cart count should immediately reflect the existing items.
- Add items while logged out, then log in. The cart should merge and the count should be correct.

Q). What API or network validations would you do in this scenario?

Answer:

- Open browser Developer Tools → **Network tab**
  1. Check the "Add to Cart" API request trigger when click on that button
  2. Check request payload includes required properties/objects (e.g., product Quantity, Product Id etc.)
  3. Check the API response and also Confirm frontend updates cart counter based on API response.
  4. Check for caching issues or stale responses.

5. Validate cookies, local storage, or session storage updates.

Q). If the cart works in staging but fails in production, how would you approach debugging?

Answer:

1. Analyze Network & Logs
    a. Check server logs for errors, timeouts, or failed requests
2. Check Data & all Configurations
3. Compare Environments
4. Verify same build version is deployed.
5. Inspect Frontend Behavior
6. Check the caching, local storage, or session storage if need clear these.

Context-02:

Q) Perform an impact analysis if this issue were found in production?

Answer:

Below is my clear **impact analysis** for a broken **"Forgot Password"** link found in production –

- User Impact
    - ✓ Users cannot reset their passwords if they forget them and users may be locked out of their accounts. So, it Leads to frustration and poor user experience.
- Business Impact
    - ✓ If the "Forgot Password" link is broken, customers will contact support instead, which increases the business costs.
    - ✓ If the users cannot access the application, then application loss of active users that leads to potential revenue loss
    - ✓ Damages brand trust and credibility.

Analytical Solutions:

1. What comes next in the series? 2, 6, 12, 20, 30, ___

A) 36 **B) 42** C) 44 D) 46

Answer: B) 42

2. Which word does not belong to the group? Apple, Banana, Mango, Carrot

A) Apple B) Banana C) Mango **D) Carrot**

Answer: D) Carrot

3. A person walks 5 km north, then 3 km east, then 5 km south. How far is the person from the starting point?

A) 0 km **B) 3 km** C) 5 km D) 8 km

Answer: B) 3 km

4. Statement: All engineers are intelligent.

Conclusion 1: Some intelligent people are engineers.

Conclusion 2: All intelligent people are engineers. Which conclusion(s) follow?

A) Only 1 B) Only 2 C) Both D) Neither

Answer: A) Only 1

Load Testing Concepts:

Q) What is the difference between Load, Stress, Spike, and Soak testing?

Answer:

Load Testing:

Testing the speed and response time of an application by gradually increasing user load which is equal to desired numbers of users load or less than desired number of users. For Verifying the speed of an application, we do load testing.

Example:

let's, the desired number of users is 1000 as per NFR's (Non-functional Requirements) document. So, we are applying load <=1000 users and check the response time of the application.

Stress Testing:

Stress testing is primarily used to determine the robustness (stability and reliability) of an application by gradually increasing user load with some additional volume of users load than the desired number of users' load.

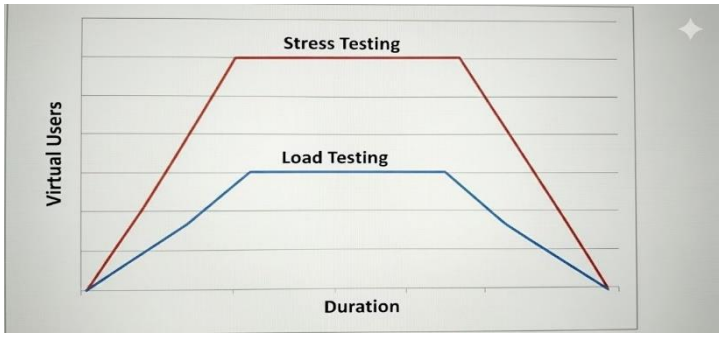
In Stress testing the objective is not to break the system, Our Object is to test an application with the additional volume of users to determine the robustness (stability and reliability) of an application but during this process our application may break down or may not break down.

Example:

let's, the desired number of users is 1000 as per NFR's (Non-functional Requirements) document. So, we are gradually applying load more than 1000 users and check the Scalability of the application.

Difference between Load and Stress Testing

- The duration is the same for load testing and stress testing only difference is the number of users.



Spike Testing:

- If there is suddenly increase the volume of users load on our application and that users load stay on our application for a short duration of time (non-existent ramp-up time. In the same way, the ramp-down is very fast), that is called spike testing.
- A spike test verifies whether the system survives and performs under sudden and massive rushes of utilization.

Example:

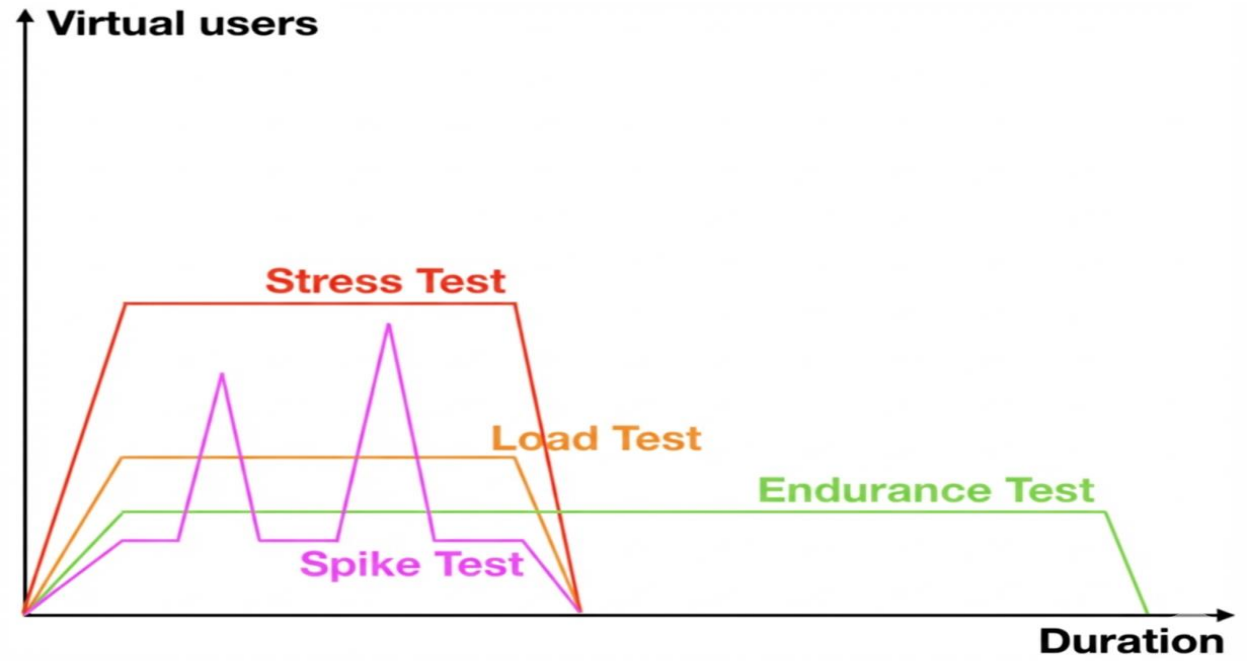
Spike tests are useful when our application may experience events with exceptional traffic volumes. Examples of such events include ticket sales (Bangladesh Vs India), product launches, seasonal sales (Black Friday) etc.

Soak Testing:

- To verify our application behavior and reliability by continuously testing an application and steady state extends very long duration of time with constant volume of users load on the application.

The following table provides comparisons of the 4 types of load testing:

Type	VUs/Throughput	Duration
Load Testing	Equal or less than expectation	Mid (5-60 minutes)
Stress Testing	High (above expectation)	Mid (5-60 minutes)
Spike Testing	Very high	Short (a few minutes)
Soak Testing	Average	Long (hours/days)



Q.) What are throughput and latency in load testing?

Answer:

Throughput:

- The number of requests process by the server per unit of time is known as Throughput. The unit of time could be seconds, minutes, hours.
- In case of load testing, this usually hits per second also known as request per seconds.
- Throughput also known as TPS (Transaction per seconds).

Latency:

- The time delay between the request being sent and the first byte of the response being received from the server.
- In latency, server processing time is not included.

Q.) What metrics do you monitor during a load test?

Answer:

The following metrics I do monitor during load testing-

1. Response Time
2. Throughput
3. Concurrency
4. Error Rates
5. Time to First Byte (TTFB)
6. CPU Utilization
7. Memory Usage
8. Network Throughput and Latency
9. Disk I/O