Performance Testing

Lesson 2: Types of Performance Testing

Lesson Objectives

- To understand the following topics:
  - Load Testing
  - Stress Testing
  - Endurance(Soak) Testing
  - Spike Testing
  - Volume Testing
  - Scalability Testing
  - Summary

Load Testing

Load Testing - Overview

- Load Testing is non functional testing type and is a subset of Performance Testing
- Purpose of Load Test is to determine how the application behaves when multiple users access it simultaneously.
- Goal of Load Testing is to ensure smooth functioning of the software under real-life load conditions.

This testing usually identifies -

- The maximum operating capacity of an application
- Determine whether the current infrastructure is sufficient to run the application
- Sustainability of application with respect to peak user load
- Number of concurrent users that an application can support, and scalability to allow more users to access it.

Need of Load Testing - Examples

- Some extremely popular sites have suffered serious downtimes when they get massive traffic volumes.
- Popular toy store Toysrus.com, could not handle the increased traffic generated by their advertising campaign resulting in loss of both marketing dollars, and potential toy sales.
- An Airline website was not able to handle 10000+ users during a festival offer.
- Encyclopedia Britannica declared free access to their online database as a promotional offer. They were not able to keep up with the onslaught of traffic for weeks.

Load Testing

Load Testing Examples

- **Airline Website During Promotion Period**. Evaluating an airline's website that will be launching a flight promotion offer and is expecting 10,000+ users at a time.
- **Government Web Portal**. Evaluating a govt. websites particularly during the filing date of income tax returns when traffic will spike.
- **Running numerous concurrent requests on a server** subjecting the server to a massive volume of traffic.
- **The nonstop transfer of multiple files to and from a hard disk**. This could take the form of measuring the speed of laptop with the transfer of 8GB-15GB files to and from a laptop with speeds of 1 mbps.
- **Downloading huge volume of large files** from a company website.
- **Requesting various jobs** on printers in a queue simultaneously.

Stress Testing

Stress Testing - Overview

- Stress Testing is non functional testing type.
- Purpose of Stress Testing is to make sure that the system would not crash under extreme conditions.
- Under Stress Testing, AUT is stressed for a short period of time to know its withstanding capacity.
- **Example :**
- The application under testing will be stressed when 5GB data is copied from the website and pasted in notepad. Notepad is under stress and gives 'Not Responded' error message.

Stress Testing

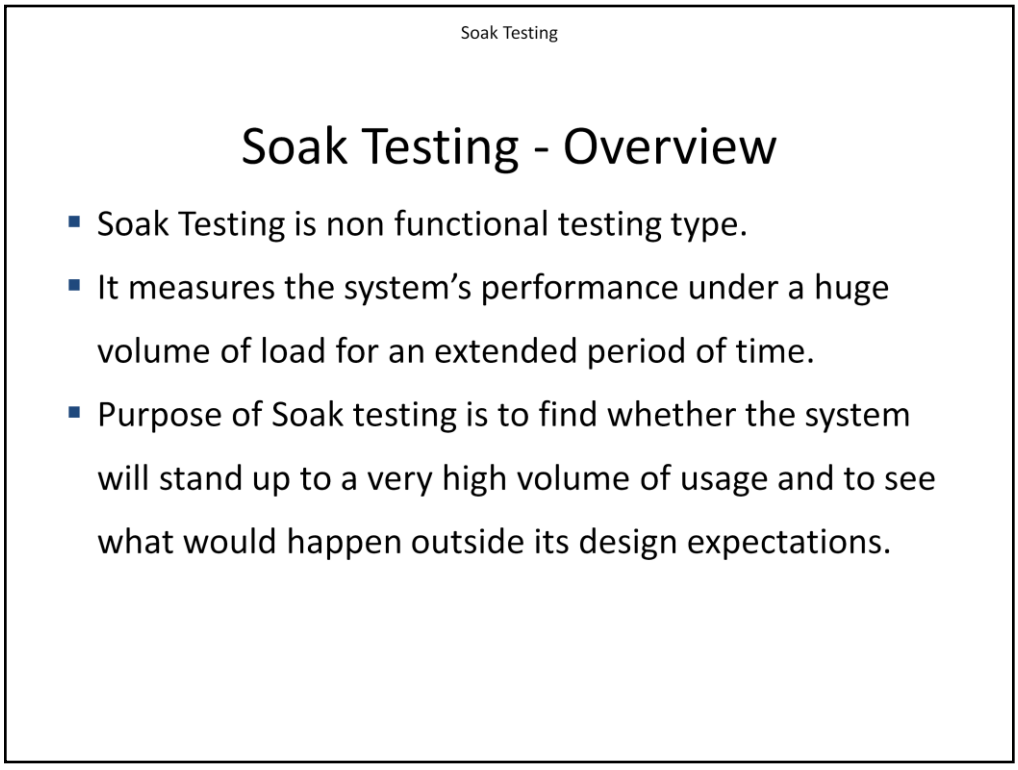
Need of Stress Testing

- Stress testing is also extremely valuable for the following reasons :
- To check whether the system works under abnormal conditions.
- Displaying appropriate error message when the system is under stress.
- System failure under extreme conditions could result in enormous revenue loss
- It is better to be prepared for extreme conditions by executing Stress Testing.

Stress Testing

Stress Testing Examples

- During festival time, an online shopping site may witness a spike in traffic, or when it announces a sale.
- When a blog is mentioned in a leading newspaper, it experiences a sudden surge in traffic.



A standard Soak Testing Method should have the following characteristics: -

- The duration of most Soak Test is often determined by the time available.
- Any application must run without any interruption if it requires an extended period of time.
- It should cover all the scenarios that are agreed upon by the stakeholders.
- Mostly every system has a regular maintenance window time period and the time between such window periods is a key driver for determining the scope of a Soak Test.

Soak Testing

Need of Soak Testing

- A system may behave normally when used for 2 hours, but when the same system is used continuously for 10 hours or more than that then it may fail or behave abnormally/randomly/it may crash.
- To predict such failure Soak Testing is performed.

Soak Testing

Soak Testing Scenarios

- Before the built is deployed to the client i.e. prior to the release of any application on a specific platform, it needs to go through a successful series of load tests at high or equivalent traffic levels. After that soak testing is performed.
- It helps us to determine how to run any particular application for an extended period. If issues like memory leaks/memory corruption are found during the period i.e. when it is on Soak, then it should be immediately reported.
- The best time to do a soak testing is over the weekends as an application need to be in a running state for as long as over a day or night. It totally depends on the limitations of the testing situation.

Soak Testing

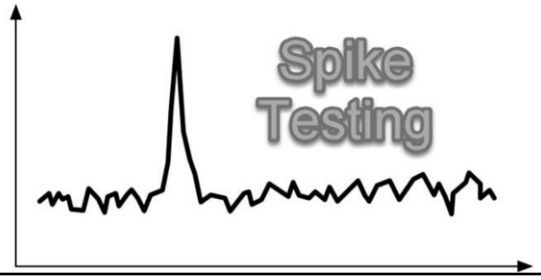
Soak Testing Examples

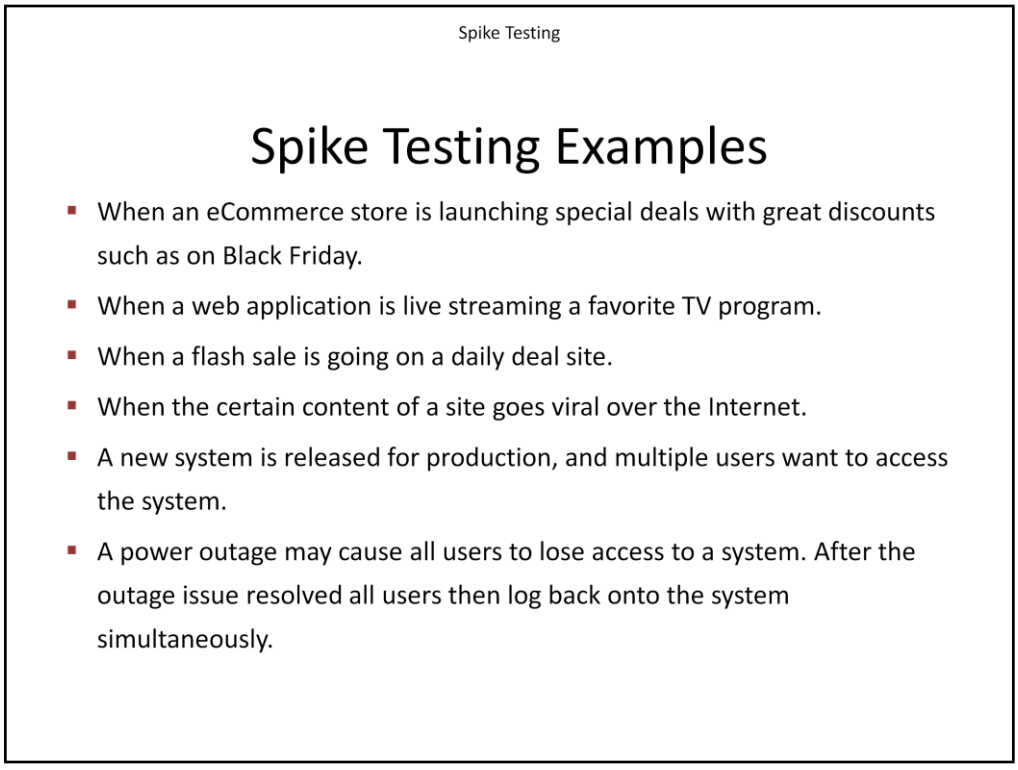
- In a banking domain when there is a large amount of data from merchants, the tester will put the system under load continuously for 70hrs to 150 hrs to check how the application behaves during this loading period.
- Suppose there are 33,000 logins, which needs to be put through the system. In this case, a 60- 70 hours of Soak Test can be started by Friday evening around 6 pm which can be completed by Monday morning at 6 am. Only with such a test, it will be possible to observe any degradation of performance under the controlled conditions.
- In the case of Video games, Mobile applications, etc. involve leaving the game or application in a running state for a prolonged time period, in various modes of operation- such as idling, paused at the title screen and so on to find out whether an application can handle the continuously expected load.

Spike Testing

Spike Testing - Overview

- Soak Testing is non functional testing type.
- It measures the system's performance with extreme increment and decrements in the load.
- Purpose of Spike testing is to determine the recovery time after a spike of user load.
- It is performed to estimate the weakness of an application.





Recovery scenarios on Spike Loads

Three main recovery scenarios can be configured to guard against spikes :

1. Use cloud platforms like AWS, Azure to dynamically increase server capacity in tandem with the user load
2. Do not allow the application-access to some users, so that system does not face heavy load. This stops people above the maximum designed load from entering into the system. Thus protects the system from the threat of an excessive load.
3. The site admin allows users to join the system. However with warning that they may face slow response because of heavy load. This may result in the adverse effect on the system performance. However, the user will be able to work with the system.

Volume Testing

Volume Testing - Overview

- Volume Testing is non functional testing type.
- It a.k.a. Flood Testing.
- It measures the system's performance by increasing the volume of data in the database.
- **Example :**
- Testing the music site behavior when there are millions of user to download the song.

volume Testing

Need of Volume Testing

- To check system performance with increasing volumes of data in the database
- To identify the problem that are likely to occur with large amount of data
- To figure out the point at which the stability of the system degrades
- To identify the capacity of the system or application - normal and heavy volume

Best practices for high volume testing

- Stop all servers and check all logs
- Before the load test manually execute the application scenario
- For most useful results stagger the number of users
- To overcome license constraints, balance think time
- Be cautious with the new build
- Analyze the use case for improvement once a baseline has been established
- A repetition of particular parts of volume testing becomes inevitable in case there is a performance bottleneck

Scalability Testing

Scalability Testing - Overview

- Scalability Testing is non functional testing type.
- It measures the system's performance in terms of its ability to scale up or scale down the number of user requests or other such performance measure attributes.
- **Example :**
- If scalability testing determines the maximum load to be 10,000 users, then for the system to be scalable, developers need to take measures on factors such as decreasing response time after 10,000 user limit is reached or increasing the RAM size to accommodate the growing user data.

Scalability Testing

Need of Scalability Testing

- Scalability testing lets you determine how your application scales with increasing workload.
- Determine the user limit for the Web application.
- Determine client-side degradation and end user experience under load.
- Determine server-side robustness and degradation.

Summary

- In this lesson, you have learnt:
  - What is Performance Testing?
  - Objectives of Performance Testing
  - Performance Testing Process
  - Performance Testing Metrics
  - Examples Performance Test Cases