LOAD TESTING, PERFORMANCE TESTING AND STRESS TESTING EXPLAINED

Are you a looking for ways to simulate expected pressure on your software, website or application? Load testing allows you to know if your software can handle the eventual demands from the numerous end users who will be using it.

Load testing is one form of performance testing that involves applying ordinary stress on a software, application or IT system. The aim is to establish if what you have can perform under normal conditions. Related to load testing is the more brutal stress testing.

So how do you ensure that under peak demand, your systems remain accessible? You have to take your system through performance testing, load testing, and stress testing.

What is Performance Testing?

When you want to find out how the components of a system are performing under a given situation, performance testing is the way to go. You will also be interested in finding out the scalability, resource usage, and reliability of the final product.

Since it's a subset of performance engineering, this type of testing is meant to address performance issues in the architecture and design of the system. Load testing and stress testing are merely subsets of performance testing. As such, performance testing is a wide term that encompasses spike testing, endurance testing, volume testing, and scalability testing.

Want to establish the benchmark behavior of your system? What you need is performance testing. What you are looking for during performance testing is to meet a number of industry-defined benchmarks. It is not in any way aimed at finding any defects with the application.

Rather, it's meant to set the benchmark and standard for the application. The key is to ensure that utmost accuracy is observed during performance testing. That's why this process involves close monitoring of how the system or application is performing.

During the setting of the benchmark, ensure you examine attributes such as throughput, response time, speed, stability, and resource usage. Analysis of these attributes is the essence of a performance test.

What is Load Testing?

Of course, load testing ensures a given function, system, or program can handle what it's designed for. It is the process that involves putting pressure on a software, system or computing device so as to measure its response. That's how you find out the behavior of the system under normal and expected peak conditions. Creating production simulations within a system or application is what load testing is about. Usually, it's done on a system that is as close as possible to becoming a finished product ready for deployment to the masses. The team that's working on developing the software should establish if the system is doing as expected under given conditions.

How Does Load Testing Work?

During this process, the system is taken through a steadily and constantly increasing load until it reaches the threshold. The weight limit is usually at what is expected to be the peak load. Load testing should help you identify page load issues, system lag, and anything else arising from multiple users accessing the system.

Of course, you want to know what will happen to the system when the traffic increases suddenly. You may not be able to detect some of these issues during development and testing. At this time, the code is tested only with the individual users in mind. So you want to know what will happen when you have hundreds or thousands of people trying to use the software or run it almost at the same time.

Who knows, you might just end up detecting problems that you can't see during solo use.

How Can Load Testing Be Done?

There are quite a number of suitable automation tools that can be used to perform load testing. Some of the best load testing tools include LoadRunner and WAPT. No matter what tool you use, you might as well call load testing endurance testing or volume testing. The only distinction is that volume testing is mainly focused on databases.

As for endurance testing, the system is kept under a significant load for a long period of time.

Where Can You Use Load Testing?

Load testing is usually done on a software program by trying to mimic the actions of multiple concurrent users of the program. Such a program could be a multi-user platform such as a web server. But that doesn't mean you can't test other software systems.

To test a graphics editor or word processor, for example, you can force it to read a very large document. You may also want to test a financial software by making it produce a report on the basis of various years of data. Load testing simulates actual use rather than testing with an analytical or theoretical model.

If you need to know the quality of service performance of your website, you need to take it through performance testing. Of course, the test will be on the basis the expected behavior of the actual customer. You need load testing tools to do this. Customers will visit your site, the communication will be recorded by a script recorder creating related interaction scripts. The work of a load generator is to replay the recorded scripts and even modify them with different parameters before rerun.

It is a repeated procedure that involves the collection and monitoring of both the software and hardware statistics. Of interest are the CPU, disk IO, and memory of the physical servers. The test will also show the response time and throughput of the system among other things. After analysis, the result is the generation of a load testing report.

What Are the Goals of Load Testing?

Load testing is done with a number of goals in mind. The following are some of the objectives of load testing:

- Determining the upper limit of all the components of an application which include the hardware, database, and network. This test ensures that the application has the ability to manage any load it will be exposed to in the future.
- Detecting defects in applications that relate to buffer overflow, mismanagement of memory, and memory leaks. The process of load testing is likely to expose a number of issues including bandwidth issues, load balancing problems, and the carrying capacity of the system.

When is the Time For Load Testing?

Load testing as a process doesn't start till a project is close to its end cycle. It is at this time when system performance and actual user engagement can undergo accurate simulation and testing. Think of a car. It's possible to repair it and test the engine.

However, without installing the engine can you really run a road test on the car? But that doesn't mean you should wait till the software development process is completed before testing. You are free to test a specific component in a focused way for the load at any stage of the development process.

It could be backend performance issues, the endurance of input over prolonged periods of time or simultaneous user input. However, checking for things that could cause lag, broken functionality or memory leaks before completing development is load testing in a limited form.

But it's advisable that you wait till the system is nearly complete before you can test the demand on a software by thousands of users.

What is Stress Testing?

Stress testing is concerned with applying unrealistic load scenarios to the point of overloading the system till it breaks. With this, you can be sure to know how well a piece of software performs. You can use it on a software, website or a backend system such as a server hosting the site.

Through stress testing, you can deliberately induce failures to enable you to analyze the risks at the breaking points. You can then tweak the programs to make the breaking more graceful. Through stress testing, you can find out how far you can push a given system apart from exploring the outer limits of the system's performance capacity.

It involves various activities that are used to overload the system with lots of jobs so as to break down the system. Stress testing may also involve the removal of system components. It also goes by the name fatigue testing.

The aim is to find out how stable an application is by stretching it beyond its bandwidth capability. You can, therefore, say that stress testing evaluates how an application will behave beyond normal conditions and peak load.

What is the Purpose of Stress Testing?

Stress testing aims at ascertaining the failure of the system and monitoring how it will gracefully recover back. But you have to ensure that you set up a controlled environment before you launch the test. That way, you can precisely capture the system behavior over and over again under scenarios that you cannot predict.

Some of the issues that can be exposed during stress testing include memory leaks, race conditions, and synchronization issues.

What Are the Types of Stress Tests?

- A spike test involves checking the behavior of a system when the number of users suddenly increase.
- A soak test checks the sustainability of a system over a period of time through a gradual increase in the number of users.

What is the Goal of Stress Testing?

Stress testing has its goal set on analyzing post-crash reports to understand how the application behaves after failure. Only that you must ensure that the system doesn't compromise the security of sensitive data after failing.

Under normal circumstances, the system should return to normality along with its components following the most terrible breakdown.

Load testing is not merely a development need but rather a business requirement because key stakeholders demand fast and reliable digital experiences. This means that your applications must be able to support thousands or even hundreds of thousands of users seamlessly. Knowing what test to do at what time and for what purpose will help you achieve this goal.