

# Load Testing Scenarios Selection

## Abstract

The purpose of Load testing is to identify and resolve all the application performance bottlenecks before they affect the application real users. If a web application's response time is more than 5 seconds then the users' interest in that application starts decreasing and they eventually abandon that application. Performance testing ensures that there isn't any major bottleneck in the system which is causing delays in the completion of users' requests. Due to the complexity, efforts and costs associated with the load testing activity, testing each and every application scenario is not possible. You need to pick only those scenarios which have greater performance impact on real users' activities. Your executed load test's results will be as useful as your scenario selection will be. Good performance testing teams are not only good in executing the selected scenarios but they are also very best when it comes to the scenario selection and they only test and fine tune those scenarios which have a greater performance impact on the overall application. Missing out even a single important scenario can greatly affect and vary the application performance in production from the test results achieved in a lab.

In this paper, we will discuss the importance of selecting the right set of application scenarios for your load test and will provide you the criteria for selection of best candidates for load testing along with the approaches to help you reach out the scenario selection decision.

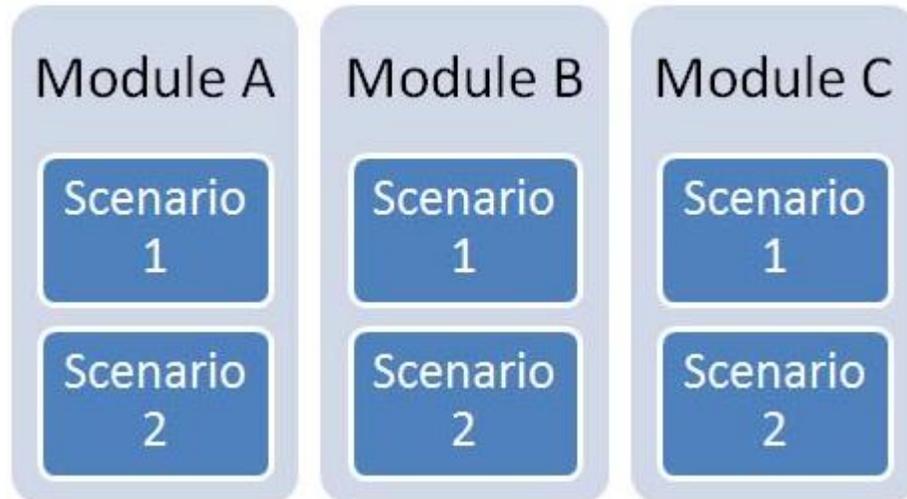
## Introduction

Load testing of a web application comprises of various activities like performance acceptance criteria, key scenarios identification, creating workload model and target load levels identification, load metrics identification, designing and executing the test and also the analysis and reporting of the test results. Only a proper execution of all these activities can produce effective test results. In order to successfully execute and complete all the above mentioned activities, a lot of planning, efforts and resources are required. Explaining all these activities in detail in one place is not easy and that's why we only chose to discuss about the "load testing key scenarios identification" in this white paper.

A Scenario is a set of user's actions on an application. Load test scenario consists of a set of user actions on application under test (AUT) which are simulated and then executed under different users load. As you know that load testing is a complex and time taking activity and you can't test each and every user action (like functional testing) under heavy loads. If you try to test all the application scenarios within a load test, you will not only enter into an unending loop but it is also something not required at all. 80-20 rule of the Pareto principle (which entails that 80% of the performance bottlenecks can be identified from 20% of the application scenarios) best describes the AUT load testing scenarios selection criteria. You only select specific AUT scenarios which have greater importance or can create more impact on the application's overall performance. Although all the application scenarios load testing is never required but your test results could be completely different from the production site performance if you miss out any single important test scenario. Similarly, if you select any extra scenario(s), it will only increase the load test cost, time and efforts without actually producing any useful outcomes. Proper planning and execution from the performance testing team is required in order to select the best set of application load test scenarios.

## Load Testing Scenario Selection Criteria

As we have already explained the importance of selecting the right set of scenarios for load testing, so in this section, a few characteristics of such scenarios will be discussed. These scenarios characteristics will provide the guidelines for all the stakeholders especially the performance testing team regarding selecting the most appropriate load test scenarios for an AUT.



### Measurable Scenarios

Fundamental point in any scenario selection is that you can set its specific performance goals and it should be measurable. Your selected load testing scenario will be useless if either you can't set its pass/fail criteria or it's not completely measurable. So, before selecting any scenario make sure you can set its performance goals and all its parameters will be available in quantified manner for analysis once the testing activity is complete.

### Most Frequently Accessed Scenarios

Best performance of the most visited areas of your application is very essential. These are the areas which receive the most users' hits and most of the application traffic is generated from such areas. Bad performance of the most commonly accessed scenarios will cause you to lose maximum users especially when such application scenarios lead to business critical scenarios like E-commerce web applications. Browsing product catalog is the most common scenario for an E-commerce application. Most users go through the product catalog before performing the next required action which could be either searching the desired product or product selection if it is found while browsing through the catalog. In such situations, performance of these scenarios becomes even more critical as they help users to complete their business transactions.

Now comes the question, how to identify AUT most commonly accessed scenarios? Following is a list of different techniques that could be used to identify the most commonly used scenarios of the application.

- If it's a live application then web server log files analysis can provide you with the information of application areas that are most frequently accessed.
- Analyzing similar existing applications can provide you with the information of most used application scenarios if the application is not yet in production.
- Another approach could be to ask the beta testers or prototype users for this information.
- Explore the application and use your testing and relevant domain skills to identify application's most frequently accessed scenarios.

### **Business Critical Scenarios**

Any application's core scenarios are called its Business Critical scenarios. Testing and optimizing most frequently accessed scenarios is not all about load testing. In fact performance of the application's business critical scenarios is more important. These are the application's core areas and they generate significant amount of revenue for the company. If the users are unable to complete application business processes effectively, it will create a huge frustration among them. In case of an E-commerce application, purchasing a product will be an example of business critical scenario.

Different approaches could be adopted to identify any application's business critical scenarios. Some of them are as follows:

- You can consult with application's major stakeholders especially the marketing department and ask them to provide their input on these scenarios.
- You can also read the marketing material to identify AUT's business critical scenarios.
- Another technique is to browse through the application and use your experience to figure out the business critical scenarios own your own.

### **Resource Intensive Scenarios**

There are always a few scenarios in AUT which require more resources as compared to others. Proper testing of such scenarios is always required otherwise they can affect the system even at a very low user load. These scenarios may not be used most frequently but they are still very important due to their impact on overall application performance. Database operations (read, write, update and delete) are normally considered as the most resource intensive scenarios in any application and they should be thoroughly tested. In an E-commerce web application, order placement will be the most resource intensive scenario as it will be accessing the database during its execution. Processor, memory, Network I/O and Disk I/O are the primary sources of resource usage in such scenarios.

Identification of most resource intensive scenarios could be done through following techniques:

- You can figure out the most resource intensive scenarios by reading the application design documents.
- Consulting with the developers could be another approach to identify these scenarios.
- You can also use your experience for identifying the most resource intensive scenarios by exploring the application.

### **Technology Specific Scenarios**

There can be a few technology specific scenarios in an AUT that might not be executed very frequently but they should be a part of your load test. These special scenarios performance can be entirely different from the AUT's other scenarios and testing them thoroughly is very important. Uploading of files through FTP could be an example of technology specific scenarios.

Technology specific scenarios can be identified using the following approaches:

- You can figure out these scenarios by reading the AUT design documents.
- Another approach could be to consult the application developers about such scenarios.

### **Stakeholder Concerning Scenarios**

There could be a few stakeholders' related scenarios which might not be very crucial for AUT's overall performance but they are directly related to the stakeholders. AUT's stakeholders will be greatly concerned about the performance of such scenarios and they will be their most desired test scenarios. New added features could be an example of the stakeholders' concerned scenarios.

Stakeholder concerning scenarios can be figured out in the following way:

- You can interview the stakeholders about their concerning scenarios

### **Time Dependent Frequently used Scenarios**

Some times AUT contains one or more such scenarios which are executed frequently on Holy specific times only. Although such scenarios impact is never visible at early stages of the production system but they can create a huge impact on the time of these special scenarios execution. Such scenarios should be load tested to get an idea of their performance and sort out all their performance bottlenecks before they create problems on production system. 'Viewing the monthly pay slips on an online payroll application' is an example of such scenarios which is executed very frequently i.e. once in a month.

The identification of the time dependent frequently accessed scenarios that can be done by the following techniques:

- For a live application, you can figure out such scenarios by viewing the AUT's web server log files.
- Another approach could be to read the application's complete requirements and its use cases for identification of time dependent frequently used scenarios.

## Contractually Obligated Scenarios

There could be certain AUT scenarios that might not be accessed very frequently by the application users but the companies are contractually obligated for such scenarios and they are always keen to load test them. Failure of such scenarios can cost heavily to the companies and that is the main reason they insist to load test these scenarios before going live. Listing down all the digital contracts at one place to be viewed by their customers at any time that might be compulsory for a brokerage application is an example of contractually obligated scenarios. These scenarios must be thoroughly load tested before they are made live to save the company from losing the contract.

Following approaches could be used to identify the contractually obligated scenarios:

- You can find such scenarios by reading the contract document.
- By reading the use cases and application requirements.
- By reading the marketing material.
- Another approach could be to interview the stakeholders.

## Load Testing Scenarios Selection Approaches

We have discussed above the different principles regarding the selection of your load testing scenarios. Further in this section, we will discuss different approaches which a performance testing team could follow to affectively select load testing scenarios. We will define a mechanism here for scenarios selection.

### Identify AUT's all Scenarios

You will select your load test scenarios based on the above mentioned criteria. So it's necessary to have a complete list of application scenarios before making your choice. Start the activity by developing a complete list of all the features of the AUT.

This approach will make sure that you are not missing even a single application scenario which should be a part of your load test. In a typical E-commerce web application, most common scenarios could be,

- Browsing the Catalog
- Product Searching
- Order Placement

### Identification of Scenario Activities

Once you have figured out all the application scenarios, next step is to identify users' activities within every scenario. This activity will help you to dig more into the AUT and get more application insights to make a wise selection. For example, following activities would be involved in Order Placement scenario of an E-commerce web application:

- Login to application

- Browse the product catalog
- Searching for the desired product
- Select the product and its quantity
- Add selected product to your shopping cart
- Validate your payment method
- Place the order

## Scenarios Selection

Comparing the application scenarios (based on the above mentioned criteria) can be the next action once you have list down the details of all the activities of AUT scenarios. You can assign certain weight-age to every criterion (based on its importance for load testing) and give appropriate weight to every scenario and compare your scenarios' importance based on their aggregate score/weight. The top score scenarios will be the best candidates for load testing.

Share your selected scenarios with all the application stakeholders and get their approval before formally start working on them. Load testing is a very complex and costly activity and any missing or additionally picked scenario can not only invalidate your test results in production environment but can also be the cause of waste of lots of money and efforts.

## Conclusion

Performance testing involves a set of interconnected activities and proper planning and execution of all of these is essential in order to get the desired test outcomes. You can't test every scenario of an AUT for load testing due to the associated complexity, costs and efforts. Good performance testing teams spend decent amount of time on planning and selection of load testing scenarios because they are fully aware that any missed scenario can completely invalidate their test results and an extra selected scenario will unnecessarily increase the load test efforts and costs. We always learn from experiences and performance testing teams have figured out (on the basis of their experiences from different projects) different criteria of application scenarios which should be a part of their load testing activity. Some of these principles are that the load test scenarios should be measurable, most frequently accessed application scenarios, application business critical scenarios, most resource intrusive scenarios, stakeholders concerning scenarios, technology and time dependent scenarios etc. The approach to identify all the scenarios which fulfill the above mentioned criteria could be to list down all the application scenarios initially, then list down all the activities involve in any scenario completion and finally select the best candidates for load test by comparing them based on the above mentioned criteria.