**Steps of Performance Testing**



### Step 1) Identify Your Testing Environment

Know your physical test environment, production environment and what testing tools are available. Understand details of the hardware, software and network configurations used during testing before you begin the testing process. It will help testers create more efficient tests. It will also help identify possible challenges that testers may encounter during the performance testing procedures.

### Step 2) Identify the Performance Acceptance Criteria

This includes goals and constraints for throughput, response times and resource allocation. It is also necessary to identify project success criteria outside of these goals and constraints. Testers should be empowered to set performance criteria and goals because often the project specifications will not include a wide enough variety of performance benchmarks. Sometimes there may be none at all. When possible finding a similar application to compare to is a good way to set performance goals.

### Step 3) Plan & Design Performance Tests

Determine how usage is likely to vary amongst end users and identify key scenarios to test for all possible use cases. It is necessary to simulate a variety of end users, plan performance test data and outline what metrics will be gathered.

### Step 4) Configuring the Test Environment

Prepare the testing environment before execution. Also, arrange tools and other resources.

### Step 5) Implement Test Design

Create the performance tests according to your test design.

### Step 6) Run the Tests

Execute and monitor the tests.

### Step 7) Analyze, Tune and Retest

Consolidate, analyze and share test results. Then fine tune and test again to see if there is an improvement or decrease in performance. Since improvements generally grow smaller with each retest, stop when bottlenecking is caused by the CPU. Then you may have the consider option of increasing CPU power.