PERFORMANCE TESTER

The performance tester is concerned with how well a system or system component behaves under a variety of controlled circumstances. The performance tester analyses the system architecture and runtime environment to devise scenarios and then execute them against the system. After execution has completed the performance tester will analyse the results to determine whether or not there is a potential or realised performance issue.

The performance tester is interested in the whole solution including architectural design, software implementations and infrastructure provisions.

Core Skills

	Level 1	Level 2	Level 3	Level 4	Level 5
Understanding why performance testing is done *Concept*					
Understanding the observer effect and its impact in your performance test *Concept*					
Understanding controllers and agents and their usage in performance testing *Concept*					
Understanding protocols and how they are relevant for different types of performance problems and solution architectures. *Concept*					
Has a solid understanding concepts of software development and can apply them in development performance test cases or a performance test framework *Concept*					
Understands the implications of virtualisation on performance testing *Concept*					
Understand the difference between clustering and load balancing and the implications each has on performance testing *Concept*					
Understands background noise and implications it has on performance test results *Concept*					
Understands the implications of different operating systems on performance *Concept*					
Understands the implications of virtual machines on performance *Concept*					
Understands how infrastructure components communicate and can therefore devise test scenario based on this information *Concept*					

	Level 1	Level 2	Level 3	Level 4	Level 5
Core Skills					
Is aware of the available profiling counters and knows which ones to use to diagnose a particular performance problem. Concept					
Understands the difference between throughput (transactions per interval) and latency (responsiveness) and can identify when performance tests should measure either. Concept					
Understands what a resource bound component is how that impacts performance. Is able to articulate why a resource bound component can be an issue. Concept					
Understands what a CPU bound component is and how that impacts performance. Is able to articulate why a CPU bound component can be an issue. Concept					
Understands what CPU starvation is and how that impacts performance. Is able to articulate why a CPU starved component is an issue for the current architecture. Concept					
Understands what handshake issues are and how they impact performance. Is able to identify handshake issues. Concept					
Understands the difference resources that are available within an environment and how these are used (e.g. threads, handles) Concept					
Understands how different resource allocation policies can impact or benefit performance. Concept					
Understands how difference caching policies can impact or benefit performance. Concept					
Understands the concept of thrashing and can identify it within performance testing results. Concept					
Understands the difference between message size policies and how that can impact on performance. Concept					
Understands the horizontal and vertical scaling models, tradeoffs between both and can perform scalability testing on either. Concept					
Understands IP Switching or IP Spoofing and how that impacts on performance tests. Concept					

Core Skills	Level 1	Level 2	Level 3	Level 4	Level 5
Understands the current and intended growth to be used in load and volume scenarios *Concept*					
Use the organisation's performance testing tool set to execute performance tests *Responsibility*					
Use programming languages associated with the automation tool. *Responsibility*					
Debug automation test code during development and fix broken performance tests. *Responsibility*					
Create or identify state and input data needed to satisfy a performance testing scenario and is aware of the implications of data creation during performance testing *Responsibility*					
Understand the difference between identity and anonymous data and the impacts it has on automated data creation *Responsibility*					
Analyse system architecture design to identify potential bottlenecks *Responsibility*					
Read network topologies and devise test scenarios from them *Responsibility*					
Create end to end test scenarios that better mimic production usage and understands the difference between isolated and integrated performance tests *Responsibility*					
Make recommendations for improving architectural design with respect to performance issues *Responsibility*					
Is able to evaluate multiple potential performance testing tools and evaluate them against criterion to determine which tool is best for a given organisation. *Responsibility*					
Produce the performance test report at the completion of all performance testing activities for a project *Responsibility*					
Produce the performance test plan using available knowledge of project and current architectural design. *Responsibility*					
Produce a 'state of the system' benchmark document that identifies each infrastructure component and their 'load' levels during standard, peak and trough times. *Responsibility*					

Core Skills	Level 1	Level 2	Level 3	Level 4	Level 5
Establish and manage a performance testing environment so that it best replicates production whilst allowing the isolation of new components. *Responsibility*					
Test the system failure evident capabilities using performance testing techniques. *Responsibility*					
Performance Test Plan – The performance test plan d*etails what performance testing is going to be undertaken on a project, where it will occur and who is doing it.* *Artefact*					
Performance Report – The performance report is produced after all performance testing has been completed. It outlines how the testing went and if there are any pending issues or future performance constraints. *Artefact*					
State of the System Report – The state of the system report should be compiled after each production deployment. It documents each infrastructure component and the amount of 'load' that it experiences. The load should be separated into min, peak and average. This is used with performance testing to determine issues that may occur when the system is deployed to production. *Artefact*					

Aspects	Level 1	Level 2	Level 3	Level 4	Level 5
Performance – *the fundamentals of performance tests are to measure two things; throughput and latency. This is the first step where basic timings are recorded without load.*					
Load – *how does the system operate under difference usage scenarios? Load testing looks at three basic tests; the spike test where many users access the system at once; the soak test where there is a high load activity over an extended period of time and the expected usage scenario.*					
Volume – *volume testing is the inverse of load testing; where rather than changing the concurrency of users we alter scope of their effort. A search test may be performed with both one million and ten million records to see how the volume change impacts the performance*					
Stress – *stress test is done to determine the maximum capacity of the system. Aside from pushing volume and load to their maximum possible levels there are other ways to stress the system: resource restrictions, resource competition and bandwidth throttling.*					
Reliability – *similar to the soak test, the reliability test is performed to determine whether or not the solution meets its uptime requirements given expected usage.*					
Scalability – *here we look at how well the system can scale out. We can do this by throttling server capacity and seeing how easily the system scales horizontally or vertically.*					