SOFTWARE TESTING FUNDAMENTALS

The fundamentals of software testing are what we do as testers that are relevant to all disciplines. The fundamentals cover the analysis of systems and requirements to create test scenarios; reporting and management of defects; testing techniques and test execution.

Software testing fundamentals are so important that without developing them you cannot progress within a discipline. Just as our business domain knowledge impacts on our effectiveness to design test cases; so do our fundamental competencies impact on our ability to become skilled within any given discipline.

A software tester should review the list of skills and indicate their competency as per the Disciplined Software Tester competency guidelines.

The following table outlines each of the skills that are fundamental to software testers.

Core Skills	Level 1	Level 2	Level 3	Level 4	Level 5
Understands the role of software testing and is able to explain the who, what, where, how, why and when of software testing Concept					
Understands the difference between technical and semantic compliance and how that impacts the scope of testing. Concept					
Understands deductive reasoning and how it relates to software testing Concept					
Understands inductive reasoning and how it relates to software testing Concept					
Understands set theory from mathematic logic and how that relates to software testing Concept					
Understands proof theory from mathematic logic and how that relates to software testing Concept					
Understands the concepts and techniques of program correctness and how that relates to software testing Concept					
Use hypothesis testing used to prove or disprove an assumption Technique					
Use positive path testing techniques to derive test cases that prove requirement implementation has been successful Technique					

Core Skills	Level 1	Level 2	Level 3	Level 4	Level 5
Use negative path testing techniques to derive test cases that exercise the inverse of the specified requirements Technique					
The ability to risk asses the test suite to identify cases that should be executed first because they present the biggest risk to the organisation if they fail. Technique					
Application of the divide & conquer approach to failed tests to identify the root cause of a defect Technique					
Use root cause analysis to identify the root cause of a defect Technique					
Can analyse documented requirement and determine whether they are unambiguous Technique					
Can analyse documented requirement and determine whether they are testable Technique					
Produce test procedures that provide additional information necessary to complete a test case. Test procedures document the how of a test case. Responsibility					
Design tests that are discrete; this means that they only test one thing. Responsibility					
Design tests that consider and cater for all relevant input data combinations Responsibility					
Design tests that explicitly state the expectations of the tester Responsibility					
Design tests that can be executed by another competent tester without intervention Responsibility					
Analyse system behaviour to identify state data that will influence the expectations outcome and from this derive test cases that exercise these expectations Responsibility					
Analysis of test outcomes to identify defects Responsibility					
The ability to create both input and state data to satisfy the test case Responsibility					
The creation of defect reports that are accurate, reproducible and have an appropriate severity. Responsibility					

Core Skills	Level 1	Level 2	Level 3	Level 4	Level 5
Perform test case design reviews to ensure appropriate test coverage has been provided Responsibility					
Perform test case reviews to ensure test cases are written to the agreed standard Responsibility					
Perform test execution review to ensure test execution logs are correctly recorded Responsibility					
Fulfil the obligation of software testing by providing information to stakeholders regarding the quality of the solution Responsibility					
Communicate effectively by providing metrics to management regarding the testing effort Responsibility					
Escalate risks before they develop into issues Responsibility					