

# 10 BEST PRACTICES FOR AN EFFECTIVE TESTING & QA IMPLEMENTATION

## Course Objectives

After years of polishing and fine-tuning a division-wide testing & QA effort, we thought it would be a good idea to share the top ten lessons learned for what we now consider as the key to a successful outcome. We hope you find this short list useful as a source of validation or ideas.

## Topics:

**1) Process:** It is critical that the organization defines a process that is robust and certified by experts in order to initiate the software assurance quality culture. The process will serve as a guideline that may evolve over time. Most importantly, it should be made official and should be followed through. Improvements will be made until a mature process is established.

## 2) Requirements Traceability

**3) Personal Experience:** Hiring someone as a tester that lacks necessary experience is a common mistake. It is vital to acknowledge that the position requires experience in both the business and in software development in general.

**4) Deliverables:** As part of the software development and testing processes, it is necessary to define deliverables, such as requirements, a testing plan, and testing cases. These will guarantee that testers can effectively follow-up throughout the project from the software quality perspective.

**5) Tool Usage:** Both the use of tools for tracking and managing defects, as well as the creation of test cases and execution, are essential for increasing the maturity of the testing & QA process. The process may begin without tools, but they are a requisite for increasing execution maturity.

**6) Metrics:** Developing and creating metrics to track the software quality in its current state, as well as to compare the improvement with previous versions, will help increase the value and maturity of the testing process (e.g. the number of components with errors in the software/the total number of components in the software; or the number of errors detected in the testing phase/total number of errors detected).

**7) Testing Environment:** Implementation of appropriate testing environments that allow developers to reproduce the system execution in production environments is crucial to the creation and execution of the corresponding test cases.

**8) Test Data:** The testing environment required for day-to-day operation should provide or ensure availability of the necessary data to enable the corresponding test execution.

Even if you have developed the appropriate testing environments, developers need to access specific data required to execute the associated test cases.

**9) Change Management:** Like any other production environment, the testing environment should properly track changes in configuration, ensuring not only controlled results, but that the tests are run in environments that closely resemble those of the real production environment.

**10) Developer Awareness:** It is critical to have an awareness process that includes management commitment at each and every business unit and for associated developers. The goal is to demonstrate that testing activities add value to their daily work.

**Length of Course** – 1 day

**Course Author**

Gary Mogyorodi, B.Math., M.B.A., Certified Tester, Foundation Level (CTFL)  
Certified Tester, Advanced Level – Functional Tester (CTAL-FT)  
Certified Tester, Advanced Level – Test Manager (CTAL-TM)  
President - Canadian Software Testing Board

President  
Software Testing Services  
Toronto, Ontario, Canada  
Phone: 647 692-5040  
Email: [garym@softestserv.ca](mailto:garym@softestserv.ca)  
Web: [www.softestserv.ca](http://www.softestserv.ca)