Measurement and Metrics for Test Managers

- Understand the test team's and tester's role in software estimation and measurement
- Develop the right measures for your project and organization
- Create a custom Test Metrics Dashboard
- Learn how to estimate in the face of uncertainty
- Avoid dysfunctional metrics for sustainable measurement programs

This course is two courses in one. The first half of the day will be spent discussing the test manager's role in software metrics; the second portion of the class continues the discussion on metrics by focusing on estimation.

The Test Manager's Role in Measurement
In many ways, the most important value of testing is providing timely and accurate information to project stakeholders. As a by-product of testing efforts, test managers—and lead testers—need to continually measure and report the status and quality of the product under development. They also need to measure test effectiveness as a guide for improvement. Test managers make and revise test effort estimates and help determine when to stop testing and release the product. These are all example of test metrics. Because a key component of testing is to measure the quality of the software product, test managers and testers also collect data and report metrics related to the entire software development activity. During this course the instructor addresses common metrics—measures of product quality, defect removal efficiency, defect density, defect arrival rate, and testing status. Learn the guidelines for developing a test measurement program, rules of thumb for collecting data, and ways to avoid "metrics dysfunction." Several metrics paradigms and the pros and cons of each will be discussed.

Estimation In Practice
Almost anyone who has ever attempted to develop an estimate about software realizes just how difficult the task can be. The number of factors that can affect the estimate is virtually without limit. The key to good estimates is to understand the main variables, compare them to known standards, and normalize the estimates based upon their differences. This is easy to say but difficult to accomplish because estimates are frequently required when very little is known about the project and what is known is constantly changing. Throw in a healthy dose of politics and a bit of wishful thinking and estimation can become a nightmare for software practitioners—and testers.

Who Should Attend
This course provides a background in estimation for anyone who must estimate software development or testing efforts (and that should cover almost everyone!). Analysts, developers, leads, test managers, testers, and QA personnel can all benefit from this course.

Course Outline

Introduction to Software Measurement
What is measurement?
Why is measurement important?
What makes a good metric?

Measurement Rules of Thumb
The human element
Obtaining buy-in—management and staff
The Hawthorne Effect
Use of subjective metrics

Test Manager's Dashboard

Test Estimation
What is estimation?
Who should perform estimation?
What should be estimated?

Estimating Axioms

Estimation Techniques
Work breakdown
Team estimates (Wideband Delphi)
Three-point estimates
Company standards and norms

Quality of the product
Project and test status
Test effectiveness
Resources metrics
Outstanding issues

Percent of project effort
Story point sizing
Poker estimation

Wrap-up and Discussion