

Estimating and Controlling Testing

Course Summary

Description

Unreliable estimates are a major reason managers allocate inadequate time and resources to testing. Historically, estimating has been so weak in IT that some people simply assume it is impossible to estimate IT activities accurately. Test estimates are especially prone to problems because they often depend on other (unreliable) estimates. In fact, though, it is possible to estimate accurately; and huge direct and indirect benefits can result. This interactive seminar describes key principles of effective estimating and how to apply them to the unique aspects of testing. And, rather than just being a static up-front exercise, the course shows dynamic techniques that effective estimators use throughout the project to control progress as well as to refine and improve their estimates and estimating skills. Exercises enhance learning by allowing participants to practice applying practical techniques to realistic examples.

Objectives

At the end of this course, students will be able to:

- Understand the appropriate uses and limitations of rapid top-down estimates.
- Work breakdown structure bottom-up estimating techniques and issues.
- Test planning and design methods for determining the number and nature of tests to be run.
- Identify additional testing effort required besides test execution.
- Control testing activities and refining estimates throughout the project.

Topics

- Top-Down Estimating
- Bottom-Up Estimating
- Test-Based Estimates
- Estimating During the Project

Audience

This course has been designed for QA and testing specialists, managers, analysts, designers, programmers, auditors, and users who need estimates of testing time and resources.

Prerequisites

There are no prerequisites required for this course.

Duration

One day

Estimating and Controlling Testing

Course Outline

I. Top-Down Estimating

- A. "Knowing" estimating is impossible
- B. How effective estimators differ
- C. Top-down estimating advantages
- D. Tester to developer ratio-based estimates
- E. Percentage of total budget for testing
- F. Lines of code, function points sizing
- G. Calibrating effort to size
- H. Historical testing effort as basis
- I. General and testing-specific issues
- J. Impacts rule-of-thumb methods cause
- K. Appropriateness and caveats for use

II. Bottom-Up Estimating

- A. Need to identify test activities regardless
- B. Major reason estimates are inaccurate
- C. Work-breakdown structure technique
- D. Level-by-level increase in precision
- E. Near- and far-term detail differences
- F. Naming tasks to create success
- G. Identifying effort by resource and skill level
- H. Implicit vs. explicit duration
- I. Work packet roll-up
- J. Addressing contingencies and oversights
- K. Relating to top-down estimates, fudge factor
- L. Issues with task-oriented estimates
- M. Addressing extrinsic task dependencies
- N. Testing vs. debugging, when code arrives

III. Test-Based Estimates

- A. What is a test case
- B. How many tests do we need
- C. Historical defect statistics, precision
- D. Risk-based testing
- E. IEEE Standard for test plans
- F. Master test planning
- G. Detailed test plans
- H. Test design specifications
- I. Test case specifications
- J. Exploratory and ad hoc testing
- K. Test automation factors and issues
- L. Unique factors for each type of special test

IV. Estimating During the Project

- A. Recording actuals against estimates
- B. Projecting defects remaining
- C. Statistical sampling
- D. Seeding and mutation
- E. Linking two independent measures
- F. Identifying systematic estimating errors
- G. Monitoring and projecting arrival rate
- H. Identifying and attacking bug colonies
- I. Reporting testing status, business value
- J. Monitoring defect detection efficiency
- K. Applying inspections, higher-yield methods
- L. Refining estimates and estimating skills
- M. Tracing through production