



[Home](#) ▶ [Technical Training](#) ▶ [Testing Articles](#)

Metrics Used In Testing

Author : Exforsys Inc. Published on: 5th Dec 2005 | Last Updated on: 2nd Feb 2011

Metrics Used In Testing

In this tutorial you will learn about metrics used in testing, The Product Quality Measures - 1. Customer satisfaction index, 2. Delivered defect quantities, 3. Responsiveness (turnaround time) to users, 4. Product volatility, 5. Defect ratios, 6. Defect removal efficiency, 7. Complexity of delivered product, 8. Test coverage, 9. Cost of defects, 10. Costs of quality activities, 11. Re-work, 12. Reliability and Metrics for Evaluating Application System Testing.

Ads

The Product Quality Measures:

1. Customer satisfaction index



This index is surveyed before product delivery and after product delivery (and on-going on a periodic basis, using standard questionnaires). The following are analyzed:

- Number of system enhancement requests per year
- Number of maintenance fix requests per year
- User friendliness: call volume to customer service hotline
- User friendliness: training time per new user
- Number of product recalls or fix releases (software vendors)
- Number of production re-runs (in-house

information systems groups)

2. Delivered defect quantities

They are normalized per function point (or per LOC) at product delivery (first 3 months or first year of operation) or Ongoing (per year of operation) by level of severity, by category or cause, e.g.: requirements defect, design defect, code defect, documentation/on-line help defect, defect introduced by fixes, etc.

3. Responsiveness (turnaround time) to users

- Turnaround time for defect fixes, by level of severity
- Time for minor vs. major enhancements; actual vs. planned elapsed time

4. Product volatility

- Ratio of maintenance fixes (to repair the system & bring it into compliance with specifications), vs. enhancement requests (requests by users to enhance or change functionality)

5. Defect ratios

- Defects found after product delivery per function point.
- Defects found after product delivery per LOC
- Pre-delivery defects: annual post-delivery defects
- Defects per function point of the system modifications

6. Defect removal efficiency

- Number of post-release defects (found by clients in field operation), categorized by level of severity
- Ratio of defects found internally prior to release (via inspections and testing), as a percentage of all defects
- All defects include defects found internally plus externally (by customers) in the first year after product delivery

7. Complexity of delivered product

- McCabe's cyclomatic complexity counts across the system
- Halstead's measure

- Card's design complexity measures
- Predicted defects and maintenance costs, based on complexity measures

8. Test coverage

- Breadth of functional coverage
- Percentage of paths, branches or conditions that were actually tested
- Percentage by criticality level: perceived level of risk of paths
- The ratio of the number of detected faults to the number of predicted faults.

9. Cost of defects

- Business losses per defect that occurs during operation
- Business interruption costs; costs of work-arounds
- Lost sales and lost goodwill
- Litigation costs resulting from defects
- Annual maintenance cost (per function point)
- Annual operating cost (per function point)
- Measurable damage to your boss's career

10. Costs of quality activities

- Costs of reviews, inspections and preventive measures
- Costs of test planning and preparation
- Costs of test execution, defect tracking, version and change control
- Costs of diagnostics, debugging and fixing
- Costs of tools and tool support
- Costs of test case library maintenance
- Costs of testing & QA education associated with the product
- Costs of monitoring and oversight by the QA organization (if separate from the development and test organizations)

11. Re-work

- Re-work effort (hours, as a percentage of the original coding hours)
- Re-worked LOC (source lines of code, as a percentage of the total delivered LOC)
- Re-worked software components (as a percentage of the total delivered components)

12. Reliability

- Availability (percentage of time a system is available, versus the time the system is needed to be available)
- Mean time between failure (MTBF).
- Mean time to repair (MTTR)
- Reliability ratio (MTBF / MTTR)
- Number of product recalls or fix releases
- Number of production re-runs as a ratio of production runs

Metric = Formula

Test Coverage = Number of units (KLOC/FP) tested / total size of the system. (LOC represents Lines of Code)

Number of tests per unit size = Number of test cases per KLOC/FP (LOC represents Lines of Code).

Acceptance criteria tested = Acceptance criteria tested / total acceptance criteria

Defects per size = Defects detected / system size

Test cost (in %) = Cost of testing / total cost *100

Cost to locate defect = Cost of testing / the number of defects located

Achieving Budget = Actual cost of testing / Budgeted cost of testing

Defects detected in testing = Defects detected in testing / total system defects

Defects detected in production = Defects detected in production/system size

Quality of Testing = No of defects found during Testing/(No of defects found during testing + No of acceptance defects found after delivery) *100

Effectiveness of testing to business = Loss due to problems / total resources processed by the system.

System complaints = Number of third party complaints / number of transactions processed

Scale of Ten = Assessment of testing by giving rating in scale of 1 to 10

Source Code Analysis = Number of source code statements changed / total number of tests.

Effort Productivity = Test Planning Productivity = No of Test cases designed / Actual Effort for Design and Documentation

Test Execution Productivity = No of Test cycles executed / Actual Effort for testing



Like

9

0

Read Next: [Software Cost Estimation](#)

This tutorial is part of a Testing Articles tutorial series. Read it from the beginning and learn yourself.

Testing Articles

1. [Software Testing Best Practices](#)
2. [Bug Reporting – Art and Advocacy](#)
3. [Software Testing Myths](#)

4. [Challenges in Testing Web Based Applications](#)
5. [Auditing Software Testing Process](#)
6. [Testing for Agile Software Development](#)
7. [What is User Acceptance Testing?](#)
8. [System Testing: Why? What? & How?](#)
9. [What is Regression Testing?](#)
10. [Integration Testing: Why? What? & How?](#)
11. [Unit Testing: Why? What? & How?](#)
12. [Effective Software Testing](#)
13. [Software Quality Management](#)
14. [Software Project Planning](#)
15. [Software Cost Estimation](#)
16. [Metrics Used In Testing](#)
17. [Risk Analysis](#)
18. [Syllabus for Certified Software Tester](#)
19. [Technical Terms Used in Testing World](#)
20. [Bug Life Cycle & Guidelines](#)
21. [So you want to be a Software Tester?](#)
22. [Automated Testing Best Practices](#)
23. [Database Checkpoints in WinRunner](#)
24. [Data Driven Tests using WinRunner](#)
25. [Why Choose LoadRunner for Performance testing](#)
26. [Testing : Introduction to CMM](#)
27. [Automated Testing Advantages, Disadvantages and Guidelines](#)
28. [Testing Types](#)
29. [WinRunner - TSL Functions](#)
30. [Life Cycle of Testing Process](#)
31. [Best Practices in Automated Testing](#)
32. [A Unit Testing Framework for the Oracle PL/SQL Language](#)
33. [Usability Engineering - Usability Testing](#)
34. [Functional Testing Tools](#)
35. [SDLC : Software Development Life Cycle](#)
36. [QuickTest Professional 8.0 CBT Tutorial and Evaluation Copy](#)
37. [XML Unit Testing tools Series 1](#)
38. [Microsoft .NET Unit testing Tools](#)
39. [SQL Unit Testing Tools](#)

Comments