

ONE PAGE SUMMARY

IT & Software > Software Testing > Quality Metrics

Introduction to Quality Metrics

Quality Metrics: Measuring Software Testing Success

DEFINITION

Quality metrics are standards used to evaluate the effectiveness and efficiency of software testing. They help teams assess product quality and improve testing processes.

KEY CONCEPTS

Defect Density

This metric measures the number of defects per unit of software size, helping identify areas needing improvement.

Test Coverage

Test coverage indicates the percentage of the software tested, ensuring all features are evaluated.

Test Execution Rate

This measures the percentage of planned tests that have been executed, reflecting testing progress.

Mean Time to Detect

This metric tracks the average time taken to identify defects, highlighting the responsiveness of the testing process.

Customer Found Defects

This metric counts defects reported by users after release, indicating the effectiveness of the testing phase.

EXAMPLES

- A software project has a defect density of 0.5 defects per 1000 lines of code.
- Test coverage is reported at 85%, meaning 85% of the software features are tested.
- The mean time to detect defects is reduced from 5 days to 2 days.

MEMORY TIPS

- ★ Remember 'DCT' for key metrics: Defect Density, Coverage, Time to Detect.
- ★ Use 'CFT' to recall customer-focused metrics: Customer Found Defects, Test Execution Rate.
- ★ Think of 'TDC' for testing success: Test Density, Coverage, Execution.

COMMON MISTAKES

- ✗ Confusing test coverage with defect density.
- ✗ Neglecting to track customer found defects post-release.
- ✗ Overlooking the importance of timely defect detection.

QUICK RECAP

Quality metrics are essential for assessing software testing effectiveness. Key metrics include defect density, test coverage, and mean time to detect. Understanding these helps improve software quality and testing processes.