BEST PRACTICE
Test Metric Categories

While there are no generally accepted categories of metrics, it has proved helpful to many test organizations to establish categories for status and reporting purposes. The metric categories and metrics within those categories provide an inventory of the metrics that testers will use in status reporting and final test reports.

In examining many reports prepared by testers the following eight metric categories are commonly used:

- Metrics unique to test
- Complexity measurements
- Project metrics
- Size measurements
- Defect metrics
- Product measures
- Satisfaction metrics
- Productivity metrics

Metrics Unique to Test

This category includes metrics such as Defect Removal Efficiency, Defect Density, and Mean Time to Last Failure. The following are examples of metrics unique to test:

- Defect removal efficiency – the percentage of total defects occurring in a phase or activity removed by the end of that activity.
- Defect density – the number of defects in a particular product.
- Mean time to failure – the average operational time it takes before a software system fails.
- Mean time to last failure – an estimate of the time it will take to remove the last defect from the software
- Coverage metrics – the percentage of instructions or paths executed during tests.
- Test cycles – the number of testing cycles required to complete testing (Note: May be related to the size of the software system or complexity of the system).
- Requirements tested – the percentage of requirements tested during testing (Note: Can indicate requirements tested which are correct, and requirements tested having defects).

Complexity Measurements

This category includes quantitative values accumulated by a predetermined method, which measure the complexity of a software product. The following are examples of complexity measures:

- Size of module/unit (larger module/units are considered more complex).
- Logic complexity – the number of opportunities to branch/transfer within a single module.
- Documentation complexity – the difficulty level in reading documentation usually expressed as an academic grade level.

Project Metrics

This category includes status of the project including milestones, budget and schedule variance and project scope changes. The following are examples of project metrics:

- Percent of budget utilized
- Days behind or ahead of schedule
- Percent of change of project scope
- Percent of project completed (not a budget or schedule metric, but rather an assessment of the functionality/structure completed at a given point in time)

Size Measurements

This category includes methods primarily developed for measuring the software size of software systems, such as lines of code, and function points. These can also be used to measure software testing productivity. Sizing is important in normalizing data for comparison to other projects. The following are examples of size metrics:

- KLOC – thousand lines of code, used primarily with statement level languages.
- Function points – a defined unit of size for software.
- Pages or words of documentation

Defect Metrics

This category includes values associated with numbers or types of defects, usually related to system size, such as “defects/1000 lines of code” or “defects/100 function points,” severity of defects, uncorrected defects, etc. The following are examples of defect metrics:

- Defects related to size of software.
- Severity of defects such as very important, important, and unimportant.
- Priority of defects – the importance of correcting defects.
- Age of defects – the number of days the defect has been uncovered but not corrected.
- Defects uncovered in testing
- Cost to locate a defect

Product Measures

This category includes measures of a product's attributes such as performance, reliability, usability. The following are examples of product measures:

- Defect density – the expected number of defects that will occur in a product during development.

Satisfaction Metrics

This category includes the assessment of customers of testing on the effectiveness and efficiency of testing. The following are examples of satisfaction metrics:

- Ease of use – the amount of effort required to use software and/or software documentation.
- Customer complaints – some relationship between customer complaints and size of system or number of transactions processed.
- Customer subjective assessment – a rating system that asks customers to rate their satisfaction on different project characteristics on a scale, for example a scale of 1-5.
- Acceptance criteria met – the number of user defined acceptance criteria met at the time software goes operational.
- User participation in software development – an indication of the user desire to produce high quality software on time and within budget.

Productivity Metrics

This category includes the effectiveness of test execution. Examples of productivity metrics are:

- Cost of testing in relation to overall project costs – assumes a commonly accepted ratio of the costs of development versus tests.
- Under budget/Ahead of schedule.
- Software defects uncovered after the software is placed into an operational status.

- Amount of testing using automated tools.

References

Guide – CSTE Common Body Of Knowledge, V6.1