

# Essential QA Metrics

## Strategic Measurements to Improve Quality Assurance Process

Collecting test metrics allows teams to use data to improve their software quality framework, understand product quality and increase the efficiency of software testing overall. Every team is different, but these 6 key metrics can help any team measure their quality goals more effectively.

### 1 Test Coverage

The number and spread of tests across the code base. This provides insight into where your resources are being used.

#### USE FOR

- Understanding which product areas are undertested
- Determining where your team should focus their efforts

### 3 Time to Test

The amount of time it takes to run and report results for a set of tests. This helps teams understand how to make testing cycles as efficient as possible.

#### MEASURE

- How long it takes to run smoke and regression tests after a new build
- How long it takes to run new feature tests

### 2 Flakiness

Broken or unreliable tests that aren't providing useful quality feedback. Flaky tests not only waste time and resources, but also reduce overall confidence.

#### MEASURE

- Tests that pass or fail intermittently
- The root cause of failures (poorly written tests, human error, etc.)

### 4 Time to Fix

The amount of time between when something breaks and when it is fixed, giving insight into QA and developer communication.

#### MEASURE

- The process used to triage and resolve bugs
- The source of reported issues (customers, internal testing, etc.)

### 5 Escaped Defects

The number of defects that reach customers. This is one of the most direct measures of QA success!

#### MEASURE

- Number of defects that reach production
- Percentage of affected customers
- Impact on affected customers
- Product area
- Time to fix

### 6 NPS

NPS (Net Promoter Score) can be a useful indicator of overall quality, but can be challenging to map to specific quality issues.

#### USE FOR

- Assessing customer satisfaction and product performance

