



MINISTÉRIO DA EDUCAÇÃO  
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA  
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO TRIÂNGULO MINEIRO

**Alunos: Gabriel Euripedes Vieira Correa, Gustavo Garcia Santana,  
João Vitor Lemes de Almeida, Lucca Henrique de Sousa**

## **1) INTRODUÇÃO**

O campo de desenvolvimento de software é complicado e dinâmico, onde a necessidade por produtos de qualidade superior e desempenho está sempre em alta. Como a sociedade moderna depende cada vez mais da tecnologia, falhas ou erros no software podem ter implicações importantes, desde perdas financeiras até riscos à segurança e até mesmo prejuízos nos negócios. Para mitigar tais riscos e garantir que o software seja seguro e eficaz, um papel importante é desempenhado pelos testes de software.

Ele não se resume apenas à descoberta de bugs ou falhas operacionais; trata-se do processo total de controle da qualidade que tem como objetivo assegurar que o programa satisfaça os requisitos comerciais, seja fácil para os usuários compreenderem, tenha uma ótima performance e seja forte o suficiente para resistir a uma série de cenários. No entanto, é um desafio avaliar a conhecimento sobre eficácia dos testes no ambiente de desenvolvimento. Métricas são valiosas na avaliação do teste do software.

As métricas de teste de software são ferramentas essenciais que permitem quantificar e qualificar o processo de teste. Elas fornecem uma maneira objetiva de medir diversos aspectos do processo de teste, desde a cobertura do código até a eficácia na detecção e correção de defeitos. Ao analisá-las, as equipes de desenvolvimento podem identificar áreas para melhorias, tomar decisões informadas e garantir a entrega de software de alta qualidade e desempenho.

Neste artigo, exploraremos em detalhes o conceito de métricas de teste de software, os diferentes tipos disponíveis e exemplos de como essas métricas são aplicadas na prática. Ao compreender o papel fundamental das métricas de teste de software as equipes de desenvolvimento podem aprimorar seus processos de teste e garantir a entrega de software confiável e eficaz.

## 2) MÉTRICAS DE TESTE DE SOFTWARE

Desempenham um papel fundamental no processo de desenvolvimento de software fornecendo uma maneira objetiva de avaliar e quantificar a qualidade do software em desenvolvimento. Elas são ferramentas quantitativas que permitem às equipes de desenvolvimento e teste entenderem melhor o processo de teste, identificarem áreas de melhoria e tomarem decisões informadas para garantir a entrega de um produto final de alta qualidade.

Ao usar métricas de teste, as equipes podem obter insights valiosos sobre diversos aspectos do processo de teste, como eficiência na detecção de defeitos, cobertura de código, qualidade do software e desempenho dos testes. Por exemplo, ao medir a cobertura de código, as equipes podem determinar quais partes do software estão sendo testadas e identificar áreas que não estão recebendo testes adequados. Isso pode ajudar a garantir uma ampla cobertura do código e reduzir o risco de defeitos não detectados.

Além disso, as métricas de teste também permitem que as equipes avaliem a eficácia dos testes em encontrar e corrigir defeitos. Ao medir a taxa de detecção de defeitos e o tempo médio de detecção e reparo, as equipes podem avaliar quão rapidamente os defeitos estão sendo identificados e corrigidos, o que é crucial para manter o cronograma do projeto e garantir a satisfação do cliente.

Podem ser categorizadas em diferentes tipos, cada uma focada em aspectos específicos do processo de teste. Essas categorias incluem métricas de cobertura, que avaliam a extensão em que o código e os requisitos do software estão sendo testados; métricas de eficiência de teste, que medem a eficácia do processo de teste em encontrar defeitos; métricas de qualidade, que avaliam a qualidade geral do software; e métricas de desempenho, que avaliam o desempenho do software em termos de tempo de resposta, utilização de recursos e escalabilidade.

Ao compreender e utilizar adequadamente as métricas de teste de software, as equipes de desenvolvimento podem melhorar a qualidade do software, aumentar a eficiência dos testes e garantir a entrega de um produto final que atenda aos requisitos de qualidade e desempenho estabelecidos.

### **3) TIPOS DE MÉTRICAS DE TESTE**

#### **1. Métricas de Cobertura**

As métricas de cobertura avaliam a extensão em que o código-fonte e os requisitos do software estão sendo testados, que são:

**Cobertura de Código:** Mede a porcentagem do código-fonte exercida pelos testes, uma cobertura de código alta indica que a maioria das instruções do programa foi testada.

**Cobertura de Requisitos:** Avalia a porcentagem dos requisitos do software que foram testados, uma cobertura de requisitos abrangente garante que todos os aspectos funcionais e não funcionais do software sejam testados.

**Cobertura de Caminhos:** Analisa quantos caminhos possíveis através do código foram exercidos pelos testes, uma cobertura de caminhos completa garante que todas as ramificações do código sejam testadas.

#### **2. Métricas de Eficiência de Teste**

As métricas de eficiência de teste medem a eficácia do processo de teste em encontrar defeitos e validar o software. que são:

**Taxa de Detecção de Defeitos:** Calcula a porcentagem de defeitos identificados durante o processo de teste, uma taxa de detecção alta indica que o processo de teste está encontrando muitos defeitos.

**Tempo Médio de Detecção:** Avalia quanto tempo leva para identificar um defeito durante o processo de teste, um tempo médio de detecção curto indica que os defeitos estão sendo encontrados rapidamente.

**Tempo Médio de Reparo:** Mede quanto tempo leva para corrigir um defeito após sua identificação, um tempo médio de reparo curto indica que os defeitos estão sendo corrigidos rapidamente.

#### **3. Métricas de Qualidade**

As métricas de qualidade avaliam a qualidade geral do software com base em critérios como estabilidade, usabilidade e confiabilidade. que são:

**Total de Defeitos Encontrados pelo Cliente:** Conta o número de defeitos relatados pelos usuários após o lançamento do software, uma alta quantidade de defeitos encontrados pelo cliente pode indicar baixa qualidade do software.

**Eficiência na Correção:** Calcula a porcentagem de defeitos corrigidos em relação ao total de defeitos identificados, uma alta eficiência na correção indica que a equipe de desenvolvimento está respondendo rapidamente aos defeitos.

**Tempo Médio Entre Falhas:** Mede o intervalo de tempo médio entre falhas no software, um tempo médio entre falhas longo indica que o software é estável e confiável.

#### **4. Métricas de Desempenho**

As métricas de desempenho avaliam o desempenho do software em termos de tempo de resposta, utilização de recursos e escalabilidade. Elas incluem:

**Tempo de Resposta Médio:** Mede quanto tempo o sistema leva para responder a uma solicitação do usuário, um tempo de resposta médio curto indica que o software é responsivo e eficiente.

**Taxa de Erro durante Picos de Carga:** Avalia quantos erros ocorrem quando o sistema está sob carga máxima, uma baixa taxa de erro durante picos de carga indica que o software é escalável e robusto.

### **4) EXEMPLO DE MÉTRICAS DE TESTE**

#### **1. Contextualização**

Imagine que uma equipe de desenvolvimento está trabalhando em um sistema de gestão de uma loja online. Eles estão se preparando para realizar testes no software antes do lançamento para garantir que atenda aos requisitos de funcionamento e ofereça uma experiência satisfatória aos usuários.

#### **2. Escolha de Métricas de Teste**

Para avaliar a eficácia do processo de teste e a qualidade do software, a equipe decide escolher métricas relevantes que possam fornecer insights valiosos, optam as que abrangem diferentes aspectos do teste, incluindo cobertura de código, eficiência de teste, defeitos encontrados, tempo e esforço, e qualidade de código.

### **3. Métricas Utilizadas**

Após analisar as necessidades do projeto, a equipe decide utilizar as seguintes métricas:

- Cobertura de Código: Porcentagem de linhas de código testadas.
- Eficiência de Teste: Taxa de defeitos encontrados em relação ao total de testes realizados.
- Defeitos: Classificação dos defeitos encontrados por gravidade.
- Tempo e Esforço: Tempo médio para corrigir defeitos.
- Qualidade de Código: Complexidade ciclomática do código.

### **4. Primeira Etapa**

Na primeira etapa dos testes, a equipe executa uma bateria de testes para cobrir o máximo possível do código, utilizando ferramentas de análise estática e dinâmica para medir a cobertura de código e garantir que uma porcentagem significativa do código seja testada.

### **5. Segunda Etapa**

Na segunda etapa, a equipe concentra-se em avaliar a eficácia dos testes em encontrar defeitos no software, e registram todos os defeitos encontrados durante os testes e calculam a taxa de defeitos encontrados em relação ao total de testes realizados.

### **6. Terceira Etapa**

Na terceira etapa, a equipe analisa a gravidade dos defeitos encontrados e o tempo necessário para corrigi-los, e eles classificam os defeitos como críticos, importantes ou menores e calculam o tempo médio para corrigir cada categoria de defeito.

### **7. Resultado das Métricas**

Após a conclusão dos testes, a equipe analisa os resultados das métricas:

- Cobertura de Código: 85% das linhas de código foram testadas.
- Eficiência de Teste: Taxa de defeitos encontrados foi de 15%.
- Defeitos: 5 defeitos foram classificados como críticos, 10 como importantes e 5 como menores.
- Tempo e Esforço: Tempo médio para corrigir defeitos foi de 3 dias.
- Qualidade de Código: Complexidade ciclomática média do código foi de 10.

Com base nessas métricas, a equipe identifica áreas de melhoria no processo de teste e no código-fonte e toma medidas para otimizar a qualidade do software antes do lançamento.

## 5) CONCLUSÃO

As métricas de teste de software desempenham um papel crucial no desenvolvimento de produtos de alta qualidade e desempenho. Este relatório explicou detalhadamente o conceito de métricas de teste de software, destacando sua importância na avaliação objetiva da eficácia e qualidade dos processos de teste.

Ao longo do texto discutimos os diferentes tipos de métricas de teste, incluindo métricas de cobertura, eficiência de teste, qualidade e desempenho, fornecendo exemplos práticos de como elas são aplicadas em um cenário de desenvolvimento de software. Vimos como as métricas podem oferecer insights valiosos para as equipes de desenvolvimento, permitindo a identificação de áreas de melhoria e a tomada de decisões informadas para garantir a entrega de um produto final de alta qualidade.

O exemplo fornecido ilustra como uma equipe de desenvolvimento pode utilizar métricas de teste para avaliar a eficácia do processo de teste e a qualidade do software em desenvolvimento. Ao analisar métricas como cobertura de código, eficiência de teste, defeitos encontrados e qualidade de código, a equipe pode identificar áreas de melhoria e implementar medidas corretivas para otimizar a qualidade do software antes do lançamento.

No geral, as métricas de teste de software são ferramentas essenciais para garantir a entrega de software confiável, eficaz e de alta qualidade. Ao compreender e utilizar adequadamente essas métricas, as equipes de desenvolvimento podem melhorar seus processos de teste, reduzir riscos e garantir a satisfação dos usuários finais.