

Universidade São Judas Tadeu MOOCA

**USJT-2025.1-SIN1AN-MC3-GESTÃO E QUALIDADE
DE SOFTWARE**

RA

823165544 - Alex Aldib

823213510 - Leonardo Ribeiro de Almeida

823116166 - Guilherme Figueiredo

Tema 3

Alcançando a Qualidade do Software

A qualidade do software é um elemento essencial no desenvolvimento de sistemas, garantindo que o produto final atenda às exigências dos usuários, funcione corretamente e se mantenha sustentável ao longo do tempo. A qualidade abrange não só a eliminação de falhas, mas também a capacidade do software de atender tanto às necessidades funcionais quanto às não funcionais de forma eficaz e confiável.

Elementos que Afetam a Qualidade do Software

Para que um software seja considerado de qualidade, deve atender a diversos critérios, tais como:

Funcionalidade: O sistema precisa proporcionar as funcionalidades necessárias para alcançar os objetivos de seus usuários e clientes.

Confiabilidade: O software deve oferecer um desempenho consistente em situações normais, sem apresentar falhas frequentes.

Usabilidade: A interface e a experiência do usuário precisam ser intuitivas e fáceis de compreender.

Eficiência: É crucial garantir que os recursos, como tempo de processamento e memória, sejam utilizados de maneira otimizada.

Manutenibilidade: O software deve permitir que futuras alterações sejam realizadas de forma simples para aprimoramentos e ajustes.

Portabilidade: A capacidade do software de funcionar em múltiplas plataformas e ambientes tecnológicos.

Estratégias para Assegurar a Qualidade do Software

A Engenharia de Software apresenta diversas estratégias para garantir a qualidade dos produtos desenvolvidos. Entre as mais relevantes, destacam-se:

1. Modelagem de Software

O uso de representações gráficas e estruturais para planejar o sistema antes da implementação é fundamental para prevenir problemas futuros.

Exemplos: UML (Linguagem de Modelagem Unificada), diagramas de classe, diagramas de sequência e casos de uso.

2. Métodos Ágeis

Abordagens interativas que favorecem entregas regulares e uma adaptação ágil a mudanças durante o desenvolvimento do projeto.

Exemplos: Scrum, Kanban e Programação Extrema (XP).

3. Testes de Software

A realização de testes é vital para garantir que o software esteja livre de erros e funcione conforme o esperado.

Exemplos: Testes unitários, testes de integração, testes de aceitação e testes de regressão.

4. Revisões e Inspeções de Código

O código-fonte deve ser revisado por outros programadores para detectar melhorias e corrigir falhas antes de ser implementado.

5. Gerenciamento de Configuração

O controle das alterações no código e nos requisitos do projeto proporciona um desenvolvimento mais organizado e eficiente.

Exemplos de ferramentas: Git, SVN e Mercurial.

6. Documentação e Normas

Uma documentação apropriada e a aplicação de normas contribuem para que o software seja claro e mantenha um padrão de qualidade ao longo do tempo.

Exemplos: Documentação de requisitos, recomendações de boas práticas de programação (como Clean Code) e guias técnicos.

Técnicas de Teste e Qualidade

- **CAIXA-PRETA**

- Aqui não existe contato com o código. Neste tipo de testes valida-se o resultado da compilação e/ou publicação, seja através de um *app mobile*, um *site* na internet, um sistema *desktop* ou um serviço disponibilizado. Aqui o analista de teste interage com o sistema, aplicando o que conseguiu aprender através das documentações disponíveis, da exploração da solução e do conhecimento de usuários.

- **CAIXA-CINZA**

- Normalmente, ao seguir esta estratégia, faz-se uma engenharia reversa para que se possa ver e analisar o código, para que assim, seja possível entender fatores limitantes e mensagens de erro que possam “aprimorar” a construção de casos de testes (forma comumente utilizada para a definição de testes estruturados de acordo com as melhores práticas).
- **CAIXA-BRANCA**
 - Por fim, nesta abordagem tratamos de “código”. São feitas validações de estrutura, avaliação da qualidade da estruturação e da implementação do código, complexidade do código para manutenção, coesão e acoplamento, padronização, segurança, padrão de projeto, etc.

Fases e nomenclaturas utilizadas no mercado

- **CAIXA-BRANCA**
 - *Teste Unitário*
 - Validação de sub-rotinas, métodos, classes e trechos isolados de código;
- ***Teste de Integração***
 - Em resumo é a validação da transição de dados entre componentes de um mesmo sistema e partes complementares.
- **CAIXA-PRETA & CAIXA-CINZA**
 - *Teste de Sistema*

- Aqui testamos o sistema, colocando-se no lugar do usuário final, replicando-se as possíveis ações do mesmo, validando as funcionalidades do sistema em um ambiente espelho do ambiente real, seja em estrutura, em dados, ou em perfis de acesso.

- **CAIXA-PRETA**

- *Teste de Aceite*

- Neste tipo de validação o sistema deve ser testado por um grupo de usuários reais, podendo o teste ser acompanhado pelos desenvolvedores, para que eles possam anotar possíveis falhas, caso ainda existam. O foco é a validação das funcionalidades do sistema em um ambiente espelho do ambiente real, seja em estrutura, em dados, regras de negócios, requisitos ou em perfis de acesso.

- *Teste de Aceite Operacional*

- Neste caso são testes que devem ser executados no ambiente final de publicação, devendo cobrir contingência e todos os demais aspectos – funcionais e não-funcionais – necessários para garantia da qualidade do sistema, no contexto em que este funciona (por ex.: segurança é crucial para um sistema financeiro).

- *Teste de Regressão*

- Este teste é feito para identificar, a cada nova versão do software, a manutenção da qualidade das funcionalidades anteriormente implementadas e as novas funcionalidades inseridas. Neste tipo de teste é comum a identificação de “efeitos colaterais” – problemas causados em outras partes já existentes no sistema – quando da implementação ou manutenção de uma nova funcionalidade ou módulo do sistema.

Garantia da Qualidade x Controle da Qualidade

(QC x QA)

O que é Garantia de Qualidade?

A Garantia de Qualidade (QA) é um conjunto de práticas que assegura que os softwares atendam aos padrões necessários, evitando falhas e defeitos. Trata-se de um processo essencial para empresas que desejam entregar produtos confiáveis e eficientes. Além de prevenir problemas, a QA estabelece métodos para identificar, medir e analisar a qualidade dos produtos. Também envolve o monitoramento contínuo e a otimização dos processos, garantindo que os padrões sejam mantidos ao longo do tempo.

O que é Controle de Qualidade?

O Controle de Qualidade (QC) foca na identificação e correção de defeitos em softwares já desenvolvidos. Ele envolve inspeções e testes rigorosos para assegurar que os produtos atendam às especificações exigidas. Durante o desenvolvimento, o QC atua para eliminar falhas e garantir um produto final livre de problemas que possam comprometer sua funcionalidade e a experiência do usuário. Além disso, contribui para a consistência e confiabilidade dos produtos entregues.

Diferenças entre Garantia de Qualidade e Controle de Qualidade

Embora estejam relacionados, QA e QC possuem diferenças fundamentais:

- **Prevenção vs. Correção:** A Garantia de Qualidade previne falhas antes que ocorram, enquanto o Controle de Qualidade identifica e corrige defeitos após o desenvolvimento.
- **Proatividade vs. Reatividade:** QA é um processo proativo, que envolve planejamento e implementação de padrões de qualidade. Já QC é um processo reativo, focado na verificação e testes após a produção.
- **Abrangência:** QA é um conceito mais amplo, que engloba o próprio Controle de Qualidade e outras ações voltadas à gestão da qualidade. QC, por sua vez, está mais restrito à inspeção e testes de produtos finalizados.
-

Compreender essas diferenças é essencial para adotar práticas eficazes de gestão da qualidade dentro de uma organização.

Benefícios da Garantia e do Controle de Qualidade

A implementação de QA e QC traz inúmeras vantagens para as empresas, tais como:

- **Qualidade consistente**, resultando em maior satisfação e fidelização dos clientes.
- **Redução de custos**, minimizando gastos com falhas e retrabalho.
- **Fortalecimento da reputação**, demonstrando compromisso com a entrega de produtos confiáveis.
- **Maior eficiência operacional**, reduzindo o tempo necessário para produção e testes.
- **Melhoria contínua**, permitindo ajustes constantes nos processos para aperfeiçoar a qualidade dos produtos.

Como Aplicar em Projetos de Software

A garantia da qualidade é um conjunto de atividades planejadas e sistemáticas que asseguram que um produto ou serviço atenda aos requisitos estabelecidos pelo cliente e pelas normas técnicas. Esse processo abrange desde a definição dos padrões de qualidade até o monitoramento contínuo e a melhoria dos resultados.

No contexto de projetos de software, a garantia da qualidade desempenha um papel crucial, garantindo conformidade, funcionalidade, usabilidade, segurança e confiabilidade dos sistemas. Além disso, contribui para a satisfação do cliente, a redução de custos, a prevenção de defeitos e a entrega de valor agregado.

Etapas da Garantia da Qualidade em Projetos de Software

1. **Definição dos Requisitos de Qualidade** – Identificar as expectativas do cliente e os critérios técnicos que o projeto deve atender.
2. **Estabelecimento de Critérios de Aceite** – Determinar os indicadores e métodos de verificação e validação da qualidade.
3. **Planejamento das Atividades de Qualidade** – Definir responsabilidades, recursos, prazos e riscos envolvidos.
4. **Execução das Atividades** – Realizar auditorias, inspeções, testes e revisões para validar a qualidade do produto ou serviço.
5. **Monitoramento e Controle** – Coletar e analisar dados de desempenho, identificar e corrigir não conformidades, além de implementar ações corretivas e preventivas.
6. **Avaliação e Melhoria Contínua** – Analisar as lições aprendidas, reconhecer boas práticas e propor aperfeiçoamentos contínuos nos processos de qualidade.

Certificação: Reconhecimento e Valorização da Qualidade

Uma das formas mais eficazes de validar a qualidade de um software é por meio da certificação. Esse processo, conduzido por entidades independentes, assegura que um produto ou serviço atende a padrões previamente definidos, conferindo credibilidade, confiança e competitividade no mercado.

No Brasil, a **Certificadora Brasileira em Qualidade de Software (CBQS)** é referência nesse segmento. Mantida pela **Academia Brasileira de Certificações e Treinamentos (ACerT)**, a CBQS promove a excelência tecnológica por meio de diversas certificações, tanto empresariais quanto profissionais.

A garantia da qualidade não é apenas um requisito técnico, mas um fator estratégico para o sucesso dos projetos de software. Com certificações como as oferecidas pela CBQS, empresas brasileiras podem se destacar pela inovação e pela competitividade, garantindo produtos mais confiáveis e alinhados às demandas do mercado.

Conclusão do grupo

A qualidade do software é um dos pontos mais importantes de um projeto, pois é por meio dela que um cliente avalia e atribui uma nota ao produto. A percepção do usuário influencia diretamente a reputação do projeto, e quanto maior a qualidade, maiores serão as oportunidades e o reconhecimento profissional.

A aplicação de boas práticas de **Garantia de Qualidade (QA)** e **Controle de Qualidade (QC)** permite a prevenção e rápida correção de falhas, reduzindo custos e garantindo uma experiência positiva para o usuário final.

Neste trabalho, discutimos a importância da qualidade de software, os principais fatores que a determinam e as metodologias utilizadas para garanti-la. Além disso, diferenciamos QA e QC, destacando suas funções complementares na busca por um produto final livre de defeitos.

Como mencionado no início, investir na qualidade do software não é apenas uma necessidade técnica, mas um diferencial competitivo para empresas que buscam excelência em seus produtos e serviços. O compromisso com a qualidade resulta em maior satisfação do cliente, otimização de processos e um impacto positivo no mercado.

Referências Bibliográficas

Métodos de Engenharia de Software

DEVMEDIA. *Qualidade de Software: Conceitos e Características*. Disponível em: <https://www.devmedia.com.br/qualidade-de-software-engenharia-de-software-29/18209>. Acesso em: 19 mar. 2025.

Técnicas de Gerenciamento de Software

ACERTBR. *Qualidade de Software: Entenda as Técnicas, Processos, Metodologias e Ferramentas de Testes*. Disponível em: <https://acertbr.com.br/qualidade-de-software-entenda-as-tecnicas-processos-metodologias-e-ferramentas-de-testes/>. Acesso em: 19 mar. 2025.

Controle de Qualidade

LINKEDIN. *Garantia da Qualidade x Controle da Qualidade*. Disponível em: <https://pt.linkedin.com/pulse/garantia-da-qualidade-x-controle-testing-company-qualidade-software-3prxf>. Acesso em: 19 mar. 2025.

Garantia da Qualidade

CBQS. *Garantia da Qualidade: O Que é e Como Aplicar em Projetos de Software*. Disponível em: <https://cbqs.com.br/2024/03/18/garantia-da-qualidade-o-que-e-e-como-aplicar-em-projetos-de-software/>. Acesso em: 19 mar. 2025.