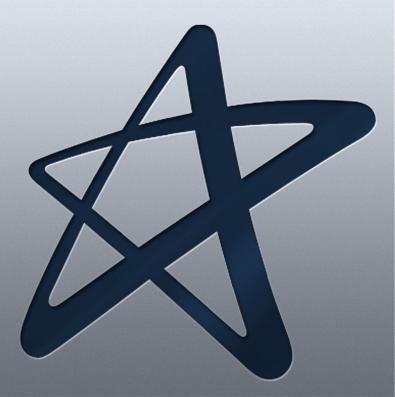


Qualidade de software





Material Teórico



Responsável pelo Conteúdo:

Prof. Ms. Douglas Almendro

Revisão Textual:

Profa. Esp. Márcia Ota

UNIDADE

Motivação de Qualidade



- Motivação de Qualidade
- Qualidade x tipo de software
- Avaliação da qualidade do produto





Nesta Unidade, será explanado o conceito de qualidade. Esse primeiro passo nos levará ao entendimento de certificações dadas às metodologias que são utilizadas no mercado de software. Então, não perca nenhum detalhe dessa unidade.

Considerando o conhecimento adquirido por meio dos conceitos iniciais de qualidade e algumas metodologias voltadas para software, a proposta desta Unidade é mostrar que a qualidade faz parte da vida cotidiana. Vale salientar que a visão popular que se tem do conceito de qualidade pode ser muito diferente de como ele é usado profissionalmente.

Por isso, não deixe de assistir, também, à apresentação narrada do conteúdo e de alguns exercícios resolvidos.

Finalmente, e o mais importante: fique atento às atividades avaliativas propostas e ao prazo de realização e envio.

Contextualização

Ao desenvolver um software, devemos estar familiarizados com os conceitos de qualidade. Entre eles, podemos citar o CMM (Capability Maturity Model), e a ISO/SPICE (Software Process Improvement & Capability Determination.

O CMM é o modelo desenvolvido pelo Instituto de Engenharia de Software (SEI) da Universidade Carnegie-Mellon, EUA, visando dar às organizações diretrizes sobre como aprimorar o processo.

A ISO/SPICE (Software Process Improvement & Capability Determination), cujo objetivo é gerar normas ISO/IEC para a avaliação de processos de software.

Norma ISO/IEC 12207, define um processo de ciclo de vida do software.

Norma ISO/IEC 9000-3, apresenta diretrizes para a aplicação da ISO 9001 (voltada para indústria), por empresas que desenvolvem software, para o processo de desenvolvimento e manutenção de software Avaliação da qualidade do produto.

Com base nos conceitos de qualidade, estaremos mais à frente, optando pela melhor metodologia a ser seguida. É assim que seguiremos uma metodologia de qualidade.



Motivação de Qualidade



Antes de iniciarmos nossos estudos, vamos primeiro entender o significado de qualidade. Veja:

O Dicionário Aurélio define qualidade como: "propriedade, atributo ou condição das coisas ou das pessoas capaz de distingui-las das outras e de lhes determinar a natureza" [Aurélio86]. Como um atributo de um item, a qualidade se refere a coisas que podem ser medidas, ou seja, comparadas com padrões conhecidos, tais como, tamanho, cor, propriedades elétricas, maleabilidade, etc. Entretanto, é mais difícil categorizarmos qualidade em software, que é uma entidade intelectual, do que em objetos físicos.

Quando falamos de qualidade, deparamo-nos com diversas situações. Veja um exemplo:

O que seria um automóvel que tem qualidade?

Vêm-nos em mente diversos tipos, modelos e marcas tão sonhadas, mas o que devemos observar?

Esses automóveis tão sonhados espelham a qualidade principalmente pela funcionalidade, segurança, fácil manutenção e diversas outras conformidades que necessitamos.

Aproveitando o nosso exemplo do automóvel, quando nos deparamos por um aviso no rádio, televisão e/ou internet, sobre um recall de um determinado automóvel, qual a sensação que é esperada a princípio por todos?

Provavelmente, a sensação de que houve um defeito no projeto do automóvel, pois o que nos surpreende é que logo em seguida mostram o lote em que se deve efetuar o recall, do número x até o número y desse veículo e modelo... Isso mostra, de fato, que houve algum erro, defeito ou bug?

Sim, mas o erro foi identificado a tempo e, mesmo assim, conseguiram saber quais veículos produzidos da linha de montagem tiveram esta anomalia.

Isso prova que as maiorias das indústrias (nesse caso, automobilísticas) conseguem ter controle de todos os processos efetuados na criação de um veículo.

Vale salientar que as indústrias têm como controlar essas anomalias, mas nem sempre foi assim, para chegar a esse grau de excelência, erram muito.

Bem, mas e quando falamos de software?

Definir qualidade de software é uma tarefa difícil. Muitas definições têm sido propostas e uma definição decisiva poderia ser debatida interminavelmente.

Ao se examinar um item baseado em suas características mensuráveis, dois tipos de qualidade podem ser encontradas: qualidade de projeto e qualidade de conformidade [Pressman97].

- Qualidade de projeto se refere a características que projetistas especificam para um item (desempenho, tolerância, etc.). O enfoque maior é nos requerimentos, na especificação e no projeto do sistema.
- Qualidade de conformidade é o grau, no qual, as especificações do projeto são seguidas durante o processo de desenvolvimento. O enfoque maior é na implementação.

Uma definição de qualidade de software que se encaixa nesse escopo é: "conformidade a requisitos funcionais e de desempenho explicitamente declarados, a padrões de desenvolvimento claramente documentados e a características implícitas que são esperadas de todo software profissionalmente desenvolvido" [Pressman95].

Hoje, temos uma dependência crescente em sistemas computacionais.

Sistemas computacionais dependem cada vez mais do software. Por isso, o mau funcionamento do software pode gerar custos altos.

Alguns exemplos do passado:

- bug do milênio;
- avião F-16: voou de cabeça para baixo ao cruzar o equador devido à falha no software de navegação;
- lançamento do ônibus espacial Columbia foi atrasado em 1981 devido à alteração errada em rotina de sincronização;
- ao menos duas mortes causadas por overdose de radiação por causa de uma falha de software no Therac-25;

http://www.cs.tau.ac.il/~nachumd/verify/horror.html

Agora, reflita:

No futuro, o que pode acontecer se bancos perderem milhões, clientes virem saldos de suas contas sumirem de repente, telefones pararem de funcionar, aviões tiverem suas rotas desviadas, vários trens/metrô forem colocados no mesmo trilho, entre muitos outros exemplos relacionados às consequências do mau funcionamento dos softwares?

Características de Qualidade

Abaixo, temos uma figura que representa as características de qualidade segundo McCall.

Cada item representa uma característica de cada processo.



Modelo de Qualidade de McCall et al, 1977



Com relação ao uso do produto (características operacionais):

- correção: o quanto um programa satisfaz a sua especificação e cumpre os objetivos visados pelo cliente;
- confiabilidade: o quanto um programa executa a função pretendida com a precisão exigida;
- **eficiência:** a quantidade de recursos computacionais e de código exigida para que um programa execute sua função;
- **integridade:** o quanto o acesso ao software ou aos dados por pessoas não autorizadas pode ser controlado;
- **usabilidade:** o quanto de esforço é necessário para aprender, preparar a entrada e interpretar a saída de um programa.

Com relação às alterações do produto (habilidade para ser alterado):

- **manutenibilidade:** o quanto de esforço é necessário para localizar e eliminar erros em um programa.
- flexibilidade: o quanto de esforço é necessário para modificar um programa.
- **testabilidade:** o quanto de esforço é necessário para testar um programa a fim de garantir que ele execute a função pretendida.

Com relação às alterações do produto (habilidade para ser alterado):

- **portabilidade:** o quanto de esforço é necessário para transferir um programa de uma plataforma de hardware e/ou software para outra.
- **reusabilidade:** o quanto um programa (ou partes dele) pode ser reutilizado em outros programas.
- **interoperabilidade:** o quanto de esforço é necessário para se acoplar um programa a outro.

Funcionalidade: o software satisfaz às necessidades explícitas e implícitas do usuário?

Confiabilidade: o software, durante um período de tempo, funciona de acordo com as condições preestabelecidas?

Usabilidade: o software é fácil de usar?

Eficiência: o software não desperdiça recursos?

Manutenibilidade: o software é fácil de alterar?

Portabilidade: o software é facilmente adaptável a diferentes plataformas?

Características e subcaracterísticas

Funcionalidade: o software satisfaz às necessidades explícitas e implícitas do usuário?

- Adequação: propõe-se a fazer o que é apropriado?
- Acurácia: gera resultados corretos ou conforme acordado?
- Interoperabilidade: é capaz de interagir com os sistemas especi
 [cados?
- Conformidade: está de acordo com normas e convenções previstas em leis, normas e descrições similares?
- Segurança de acesso: evita o acesso não autorizado, acidental ou deliberado acesso a programa e dados?

Confiabilidade: o software, durante um período de tempo, funciona de acordo com as condições preestabelecidas?

- Maturidade: com que frequência apresenta falhas?
- Tolerância a falhas: ocorrendo falhas, como ele reage?
- Recuperabilidade: é capaz de recuperar dados após uma falha?

Usabilidade: o software é fácil de usar?

- Inteligibilidade: é fácil entender os conceitos utilizados?
- Apreensibilidade: é fácil de aprender a usar ?
- Operacionalidade: é fácil de operar e controlar a operação ?

Eficiência: o software não desperdiça recursos?

- Comportamento em relação tempo: qual é o tempo de resposta e de processamento?
- Comportamento em relação aos recursos: quanto recurso usa? Durante quanto tempo?

Manutenibilidade: o software é fácil de alterar?

- Analisabilidade: é fácil encontrar um erro quando ocorre?
- Modificabilidade: é fácil modificar e remover erros?
- Estabilidade: há grandes riscos de erros quando se faz alterações?
- Testabilidade: é fácil testar quando se faz alterações?

Portabilidade: o software é facilmente adaptável a diferentes plataformas?

- Adaptabilidade: é fácil adaptar a outras plataformas sem aplicar outras ações ou meios além dos fornecidos para esta finalidade no software considerado?
- Capacidade para instalar: é fácil instalar em outras plataformas?
- Capacidade para substituir: é fácil substituir por outro software?
- Conformidade: está de acordo com padrões e convenções de portabilidade?



Qualidade x tipo de software



Cada tipo de software tem seu próprio requisito de qualidade, a importância de cada característica depende diretamente do tipo de software por exemplo.



No exemplo acima, temos a importância que cada processo em cada sistema, cada um exige mais do que o outro de acordo com seus requisitos, em uns mais itens em outros menos itens.

A figura abaixo mostra os diferentes pontos de vista das pessoas envolvidas no processo de desenvolvimento de software.

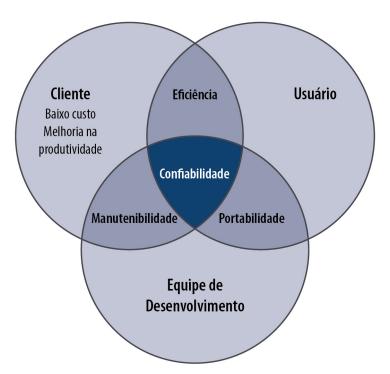


Figura: Diferentes pontos de vista

Quais são os principais objetivos da qualidade?

 Aprimorar o processo de desenvolvimento e, em consequência, melhorar a qualidade do produto resultante.

- Avaliar a qualidade do produto, visando emitir documento o cial sobre a qualidade de um software e sua conformidade em relação a uma norma ou padrão.
- Adquirir um software, com o intuito de escolher o produto mais adequado dentre um conjunto de produtos selecionados.

Aprimoramento do processo de software

Algumas iniciativas visando melhorias do processo de software:

SEI/CMM (Capability Maturity Model), modelo desenvolvido pelo Instituto de Engenharia de Software (SEI) da Universidade Carnegie-Mellon, EUA, visando dar às organizações diretrizes sobre como aprimorar o processo.

ISO/SPICE (Software Process Improvement & Capability Determination), cujo objetivo é gerar normas ISO/IEC para a avaliação de processos de software.

Norma ISO/IEC 12207, define um processo de ciclo de vida do software.

Norma ISO/IEC 9000-3, apresenta diretrizes para a aplicação da ISO 9001 (voltada para indústria), por empresas que desenvolvem software, para o processo de desenvolvimento e manutenção de software Avaliação da qualidade do produto.

Algumas normas:

ISO/IEC 9126 (NBR 13596), define as características de qualidade de software que devem estar presentes em todos os produtos.

ISO/IEC 12119, estabelece os requisitos de qualidade para pacotes de software e instruções para teste, considerando esses requisitos.

ISO/IEC 14598-5, define um processo de avaliação da qualidade de produto de software.

Avaliação da qualidade do produto



Como fazer?

- organismos de certificação:
- No Brasil, para fornecer o certificado ISO 9000, existem empresas credenciadas pelo INMETRO.
- avaliar in-house:
- Utilizar equipe multidisciplinar com especialistas da área de tecnologia e especialistas da área que se utilizará do software (ie, que vão olhar para o software a partir do ponto de vista do cliente), grupo de Garantia da Qualidade do Software.



- contratar empresas para avaliação:
- Existem empresas que fazem avaliação do software mas, por não serem credenciadas pelo INMETRO, não emitem certificado oficial. São, no entanto, mais acessíveis e ágeis que os organismos credenciados.

Alguns entraves à qualidade segundo o IEEE610.12-1990:

Erro(engano)	Ação humana que produz um resultado incorreto.	Mistake
Falha	Incorreção em um passo, processo ou definição de dados; manifestação no software de um engano cometido pelo desenvolvedor.	
Erro	Diferença entre o valor obtido e o valor esperado, ou seja, qualquer estado intermediário incorreto ou resultado inesperado na execução do software.	
Defeito	Incapacidade de fornecer o serviço, conforme especificado.	Failure

Por que surgem as falhas?

Alterações:

Alterações degradam a estrutura do software, tornando-o cada vez mais difícil de alterar.

Tempo:

 Com o tempo, os custos das implementações das alterações aumentam, e a capacidade do sistema em prestar os serviços esperados diminui.

Complexidade:

- difícil de desenvolver: um único desenvolvedor não é capaz de entender o sistema como um todo;
- difícil de usar;
- difícil de entender: código incompreensível, falta de documentação.

Garantia da Qualidade do Software

Definição de um arcabouço para se atingir a qualidade do produto de software [Sommerville01, 24.1]

Padrão sistemático e planejado de ações que são exigidas para garantir a qualidade do software. [Pressman92, 17.1.2]

Visa responder às seguintes questões:

O software atende às características de qualidade desejadas ?

- O desenvolvimento do software foi conduzido conforme os padrões preestabelecidos?
- As disciplinas técnicas cumpriram adequadamente seus papéis como parte da atividade de Garantia da Qualidade?

Norma ISO/IEC 14598 pode ser usada para definir o processo de Avaliação.

Atividades da Garantia de Qualidade segundo Pressman

Aplicação de métodos, técnicas e ferramentas, uso pelos desenvolvedores de métodos e ferramentas que ajudem a conseguir especificações, projetos, etc., de maior qualidade.

Realização de revisões técnicas, o objetivo é avaliar a qualidade do artefato de software (especificação, projeto, ...) produzido ao longo do desenvolvimento.

Atividades de testes em complemento às revisões e outras técnicas.

Aplicação de padrões

- padrões podem ser usados: para documentos, documentação do código e estilo de codificação (como usar linguagem de programação).
- padrões podem ser determinados pelo cliente, por normas internacionais ou pela empresa de desenvolvimento.

Atividades de Garantia de Qualidade

Controle de alterações, toda mudança no software tem potencial para introduzir erros ou criar efeitos colaterais, que propagam erros no controle de mudanças durante desenvolvimento e manutenção, sendo essencial para garantir a qualidade do software.

Medição, obtenção de métricas a fim de rastrear a qualidade do software e avaliar o impacto de mudanças nos métodos e procedimentos usados para desenvolvimento e manutenção.

Anotação e manutenção de registros para manter histórico com resultados de revisões, auditorias, controle de alterações e outras atividades de garantia de qualidade, que devem ser levados ao conhecimento dos desenvolvedores.

Com isso, conseguimos ter uma noção de qualidade e chegar a uma conclusão de qualidade de software que devido ao aumento da competitividade e a preocupação em oferecer ao mercado softwares que atendam às expectativas de seus clientes, os desenvolvedores de software têm buscado aplicar os conceitos de qualidade em seus produtos. Dada a história e os fatos mencionados, notamos que, nos dias de hoje, a qualidade dos softwares desenvolvidos devem aumentar significativamente. Algumas certificações existentes no mercado têm dado uma atenção especial nos processos de qualidade e mobilizado as organizações a adotarem este tipo de processo em todos seus produtos. Espera-se que as organizações comecem a incluir em seu processo de desenvolvimento, o processo de qualidade de software, não apenas no momento que o produto foi finalizado ou desenvolvido, mas desde o início de sua concepção, para não ser surpreendido com a possibilidade de ocorrerem falhas no ciclo de vida do software.



Material Complementar

Para aprofundar seus estudos sobre Qualidade, temos abaixo os sites e as seguintes referências:

- http://www.sei.cmu.edu/
- http://www.cin.ufpe.br/~in953/olds/relatorios/fabrica1.pdf
- http://www.spinsp.org.br/
- http://ibpi.org/standard/isoiec-15504/
- GAMMA, Erich. Padrões de projeto: soluções reutilizáveis de software orientado a objetivos. Porto Alegre: Bookman, 2000.
- SOMMERVILLE, Ian. **Engenharia de software.** 8ª ed. São Paulo: Pearson Addison-Wesley, 2007.

Referências

J. McCall, P. Richards and G. Walters. **Factors in Software Quality** (3 vols.), NTIS AD-AO49-014, 015, 055, Nov. 1977.

R.S.Pressman. Software Engineering, A Practicioners Approach, McGraw-Hill.

Steven R. Rakitin. "Software Verification and Validation: a Practitioner's Guide". Artech House, 1997.

Nelma S. Gomes. "Qualidade de Software - Uma Necessidade". Artigo obtido em fev/2003 em: www.esaf.fazenda.gov.br/cst/arquivos/Qualidade de Soft.pdf

PRESSMAN, Roger S. Engenharia de software. 6ª ed. Porto Alegre: Bookman, 2006.

GAMMA, Erich. Padrões de projeto: soluções reutilizáveis de software orientado a objetivos. Porto Alegre: Bookman, 2000.

SOMMERVILLE, Ian. **Engenharia de software.** 8ª ed. São Paulo: Pearson Addison-Wesley, 2007.

RESENDE, Denis Alcides. **Engenharia de software e sistemas de informação.** 3ª Ed. Rio de Janeiro: Brasport, 2005.

FILHO, Wilson de Pádua Paula. **Engenharia de software: fundamentos, métodos e padrões.** 3ª ed. Rio de Janeiro: LTC, 2005.

FERNANDES, Aguinaldo Aragon.

 $http://www.redepro.rs.gov.br/docs/11177116862Seminario_redepro_palestra_1.pdf (22/07/2010, 17:35h)$

[SEI2000] Sei, **An Overview of Capability Maturity Model Integration (CMMI)** – Version 1.0, Tutorial presented at SIMPROS 2000 [23], 2000.

[SEI2002a] Sei, **Web Site do software Engineering Institute** – SEI, http://www.sei.cmu.edu/ (CMMI nodels available at www.sei.cmu.edu/cmmm).

[ISO9001:2000] International Standard Organization Certification for IMS Company.

[ISO12207:2000] **International Standard Organization.** ISO/IEC 12207 Amendement: Information Technology – Amendement to ISO/IEC 12207, versão PDAM 3, novembro 2000.



Anotações	



www.cruzeirodosulvirtual.com.br Campus Liberdade Rua Galvão Bueno, 868 CEP 01506-000 São Paulo SP Brasil Tel: (55 11) 3385-3000











