



UNIVERSIDADE DE PASSO FUNDO
INSTITUTO DE CIÊNCIAS EXATAS E GEOCIÊNCIAS
CIÊNCIA DA COMPUTAÇÃO

Qualidade de Software

SUMÁRIO

I. - O QUE É QUALIDADE	4
<i>PROPOSIÇÕES DA QUALIDADE DE SOFTWARE:</i>	4
<i>PRINCÍPIOS DE QUALIDADE:</i>	5
<i>PRINCÍPIOS DE GERÊNCIA:</i>	5
<i>PRINCÍPIOS DE ENGENHARIA:</i>	5
II. - HISTÓRICO DA QUALIDADE	5
III. - PORQUÊ SE PREOCUPAR COM A QUALIDADE DE SOFTWARE ?.....	7
<i>QUALIDADE X DEFINIÇÃO DE PRÉ-REQUISITOS:</i>	7
<i>QUALIDADE E O DESENVOLVIMENTO SOFTWARE:</i>	8
IV. - QUALIDADE E SERVIÇO DE SUPORTE AO USUÁRIO	8
<i>O QUE É UM SISTEMA DE QUALIDADE ?</i>	9
<i>QUALIDADE PRODUTO X QUALIDADE PROCESSO:</i>	10
VI. - QUALIDADE DE SOFTWARE:	11
<i>ENGENHARIA DE SOFTWARE</i>	12
<i>QUALIDADE DE PRODUTOS DE SOFTWARE - ISO 9126</i>	13
VII - MÉTRICAS DE SOFTWARE	14
VIII - GUIAS PARA AVALIAÇÃO DA QUALIDADE - ISO 14598	15
IX - QUALIDADE DE PACOTES DE SOFTWARE - ISO 12119	17
X - QUALIDADE DE PROCESSO DE SOFTWARE:	18
XI – A SÉRIE ISO 9000 :	18
<i>ESTRUTURA DO SISTEMA DE QUALIDADE</i>	19
XII - ISO 12207 – PROCESSO DE VIDA DO CICLO DE SOFTWARE.....	20
XIII – GERENCIANDO UM COMPANHIA DE QUALIDADE	22
<i>DEDICAÇÃO À SATISFAÇÃO DO CLIENTE</i>	22
<i>DAR ÊNFASE À MELHORAMENTOS CONTÍNUOS</i>	22
<i>TRATAR FORNECEDORES COMO PARCEIROS DE NEGÓCIOS</i>	22
<i>COMUNICAÇÃO E TIME DE TRABALHO</i>	22
<i>ATUALIZANDO EMPREGADOS</i>	23
<i>COMPROMISSO DA GERÊNCIA</i>	23
XIV. – IMPLMENTANDO UM SISTEMA DE QUALIDADE	23
<i>XIV.1 - UM SISTEMA DE QUALIDADE:</i>	23
<i>Aspectos Técnicos</i>	23
<i>Aspectos culturais</i>	24
<i>XIV.2 – INICIANDO UM SISTEMA DE QUALIDADE</i>	24
<i>Preparar uma política de qualidade</i>	24
<i>Estabelecer uma equipe de suporte em qualidade</i>	24
<i>XIV.3 - DEFINIR UM PROGRAMA PARA A QUALIDADE</i>	25

<i>Avaliar a organização</i>	25
<i>Projetar um sistema de qualidade.....</i>	25
<i>Planejamento e implementação do programa de qualidade.....</i>	26
<i>XIV.4. - IMPLEMENTAR UM PROGRAMA CULTURAL.....</i>	26
<i>XIV.5 - IMPLEMENTAR O PROGRAMA TÉCNICO.....</i>	27
<i>Adotar um ciclo de vida.....</i>	27
<i>Programa de métricas e medidas de software</i>	27
<i>Desenvolvimento</i>	28
<i>Suporte.....</i>	28
<i>Treinamento.....</i>	28
<i>XIV.6 – REVISÕES DE PROCESSOS E PRODUTO.....</i>	29
<i>Revisões do projeto.....</i>	29
<i>Revisões da gerência</i>	29
XV. – O FUTURO DA QUALIDADE.....	29
XVI. - BIBLIOGRAFIA	30

I. - O QUE É QUALIDADE

Qualidade hoje em dia, não é apenas um diferencial de mercado para a empresa conseguir vender e lucrar mais, é um pré requisito que a empresa deve conquistar para conseguir colocar o produto no Mercado Global.

Na área de software, há uma urgente necessidade de uma maior preocupação sobre o tema, mas afinal, o que é qualidade?

Existem diversas definições. Algumas pessoas que tentaram uma definição simples chegaram a frases como:

- Qualidade é estar em conformidade com os requisitos dos clientes
- Qualidade é antecipar e satisfazer os desejos dos clientes
- Qualidade é escrever tudo o que se deve fazer e fazer tudo o que foi escrito

Segunda a atual norma brasileira sobre o assunto (NBR ISO 8402), qualidade é:

*A totalidade das características de uma **entidade**
que lhe confere a capacidade de satisfazer
às **necessidades explícitas e implícitas***

Nota-se que esta definição formal exige alguns complementos, principalmente para definir o que são as **entidades**, as **necessidades explícitas** e as **necessidades implícitas**.

- A **entidade** é o produto do qual estamos falando, que pode ser um bem ou um serviço.
- As **necessidades explícitas** são as próprias condições e objetivos propostos pelo produtor.
- As **necessidades implícitas** incluem as diferenças entre os usuários, a evolução no tempo, as implicações éticas, as questões de segurança e outras visões subjetivas.

Por exemplo, a qualidade de um prato de comida (a entidade, o produto) está relacionada com a satisfação de necessidades (requisitos) tais como: sabor, aparência, temperatura, rapidez no serviço, preço, higiene, valor nutricional, etc... Para avaliar a qualidade de um produto, deve-se fazer uma lista destas necessidades e analisar cada uma destas necessidades.

Proposições da qualidade de software:

- 1) Qualidade é o sucesso para o negócio de softwares, como em qualquer outro;
- 2) A maneira mais barata de aumentar a produtividade é aumentar a qualidade do software;
- 3) A qualidade ao suporte do produto é tão importante quanto a qualidade do próprio software, o ambiente de suporte deve ter engenharia tanto quanto o ambiente de desenvolvimento;
- 4) Para alcançar a qualidade de software, as pessoas e a cultura são tão importantes quanto a tecnologia;

- 5) O único caminho seguro para aumentar a qualidade do software é melhorar os processos (o que inclui pessoal, facilidades, equipamentos, tecnologia e metodologia);
- 6) Aumento de processos é normalmente desnecessário a menos que o gerente demonstre compromisso e liderança;
- 7) Qualidade e melhoramento dos processos são de difíceis esforços: é sempre possível realizar algo um pouco melhor, um pouco mais rápido e um pouco mais barato;
- 8) Um sistema de qualidade compatível com ISO9000 é um bom alvo para muitas organizações, mas não para todas;
- 9) Um sistema de qualidade para uma organização deve ser medido de acordo com suas necessidades e circunstâncias ou não será eficiente
- 10) Um sistema de qualidade de software eficiente utiliza de boas práticas da engenharia de software baseado nos seguintes princípios:

Princípios de qualidade:

- Tentar prevenir defeitos ao invés de consertá-los;
- Ter certeza dos defeitos que forem encontrados, serem corrigidos o mais rápido possível;
- Estabelecer e eliminar as causas, bem como os sintomas dos defeitos;
- Auditar o trabalho de acordo com padrões e procedimentos previamente estabelecidos;

Princípios de gerência:

- Definir regras e responsabilidades;
- Planejar o trabalho;
- Trilhar o progresso através de planos e corrigir quando necessário;
- Refinar o plano sempre e progressivamente;

Princípios de engenharia:

- Analisar o problema antes de desenvolver a solução;
- Quebrar problemas complexos em problemas menores;
- Garantir que subproblemas unam-se pelo controle de seus relacionamentos;

II. -HISTÓRICO DA QUALIDADE

Conceituar qualidade se torna uma tarefa muito difícil, pois elementos intrínsecos está enraizado no intelecto de cada ser. Portanto se exercícios forem feitos dando como missão para cada grupo, várias definições são apresentadas, mas o que mostra como bem próximo de se considerar é como sendo um método gerencial que através de processos e procedimentos disseminados pôr toda a organização busca uma posição competitiva para propiciar a satisfação da sociedade ao longo do tempo.

A história do desenvolvimento da Qualidade Total como sistema administrativo ter que ser buscado na origem do modelo científico de administração F. Taylor em 1911 publicado em seu livro Princípios da Administração Científica em que citava: o aumento da eficiência , a racionalização dos métodos de trabalho, a crença no homem econômico , a divisão e a hierarquização do trabalho , a relevância da organização formal.

Nos anos 30, o Dr. W.A. Shewhart causa uma revolução à teoria científica da administração quando propõe um método voltado para gestão das organizações conhecido como Controle da Qualidade - Controle Estatístico da Qualidade (CEQ) ou Controle Estatístico de Processos (CEP) que se baseava na aplicação de gráficos de controle, na inspeção por amostragem.

A tese de Deming para os industriais do pós guerra , nos Estados Unidos, era a da produção com qualidade. Mas, como muitas vezes acontece, a verdade é que Deming não foi ouvido em sua terra, não foi profeta em sua terra e este "não ouvir Deming" vai custar caro aos americanos, porque em pouco tempo vão perder os maiores mercados do mundo.

O Japão, em 1950, convida Deming a fazer uma série de palestras para a J.U.S.E. Hoje esta é uma sigla famosa, que significa União Japonesa de Cientistas e Engenheiros. A JUSE foi responsável pela revolução que o Japão conseguiu implementar.

Deming também foi convidado pela Associação Japonesa da Alta Administração, que é composta pelos 45 maiores industriais japoneses. Isto significava que o maior poderio privado, 89% do dinheiro privado japonês, estava aí, nesses 45 industriais.

Deming ensinou seu método e, também, aperfeiçoou-o, desenvolvendo uma nova forma participativa de gerência, a qual tirava proveito dos conhecimentos e habilidades de todos funcionários, em todos os níveis, por meios de equipes e sistemas de sugestões que sempre focalizam o cliente.

A psicologia das relações humanas traz o conhecimento do comportamento dos funcionários das organizações e diferentes teorias, a da motivação e personalidade de A. Maslow, contribuem com estudos do fator humano, preocupando-se em considerar a satisfação do funcionário como um dos responsáveis pelo aumento da produtividade e de qualidade do produto. A teoria motivação-higiene de Herzberg distingue os fatores do ambiente (fisiológicos, segurança, social, estima) como fatores provenientes especificamente do serviço (automatização), mostrando serem esses últimos os motivadores.

Em 1980 começam de forma oficial, nas organizações americanas, os grandes programas de Qualidade. A primeira organização a que Deming atende é a FORD.

Os Estados Unidos buscaram, além de Deming outros grandes idealizadores deste processo: J.M. JURAN, PHILIP CROSBY, os famosos gurus da Qualidade também chamados pensadores da Qualidade, que começa a dar consultoria para os Estados Unidos sobre Qualidade Total ou Liderança pela Qualidade.

III. - PORQUÊ SE PREOCUPAR COM A QUALIDADE DE SOFTWARE ?

A qualidade, hoje em dia, é crítica para a sobrevivência e o sucesso do mercado de software que está se desenvolvendo de forma global. Uma organização não sobressairá no mercado global a menos que produza software de boa qualidade e seus clientes vejam produtos e serviços de boa qualidade.

Existem muitas razões que devem ser levadas em conta, a saber:

- A) Qualidade é competitividade: a única maneira de diferenciar o produto do competidor é pela qualidade do software e do suporte que é fornecido juntamente. Como o mercado amadurece, usuários não querem apenas que a empresa fale que tem qualidade, mas que mostre a todos a sua qualidade através de Certificação internacional. Não ter certificação pode acarretar desvantagem competitiva.
- B) Qualidade é essencial para a sobrevivência: Clientes estão pedindo por qualidade. Se a empresa não tiver habilidade de sobreviver em um mercado altamente competitivo, ela está em débito com o mercado. A maioria das grandes organizações está reduzindo o número de fornecedores, e um meio de escolher os fornecedores é verificando quais deles têm certificações de qualidade.
- C) Qualidade é essencial para o mercado internacional: O mercado de software está, cada vez mais, se tornando global. A habilidade das empresas de mostrarem qualidade, eventualmente as colocam no mercado global. O mercado local é vulnerável a produtos importados que, normalmente, têm mais qualidade.
- D) Qualidade é custo/benefício: um sistema de qualidade direciona para o aumento da produtividade e permanentemente reduz custos, habilitando o gerenciamento para reduzir a correção de defeitos dando ênfase à prevenção. Todas as empresas sabem que corrigir defeitos após o desenvolvimento do software é mais dispendioso do que corrigi-los depois. Prevenir defeitos primeiramente pode resolver muita coisa depois e economizar bastante.
- E) Qualidade retém consumidores e aumenta lucros: pouca qualidade normalmente custa muito mais do que contratar mais desenvolvedores e ainda continuar sem qualidade. A maioria dos consumidores não tolerarão falta de qualidade e irão procurar outros desenvolvedores. Mais qualidade aumenta a satisfação dos consumidores e assegura os que já são clientes a mais tempo.

Qualidade x Definição de pré-requisitos

O processo de pré-requisitos deve identificar e definir as características de um produto em particular que é de necessidade do cliente e distingui-los dos menos importantes. É importantíssimo que, na entrega do produto final, o sistema tenha pouquíssimos ou nenhum erro ou falha e seja fácil de utilizá-lo deixando a performance para segundo plano.

A comunicação entre o desenvolvedor e o cliente é a chave para a definição correta. O desenvolvedor deverá trabalhar em conjunto com o cliente, nesta primeira fase, para definir corretamente as especificações do software.

No caso de a empresa não poder entrar em contato direto com o cliente (um produto para vários clientes, como um sistema que vai ser desenvolvido para o mercado SOHO – Small Office/Home Office), a função do marketing da empresa deverá ser tomada como cliente por conhecer o mercado que o produto vai ser lançado.

É aconselhável desenvolver técnicas de **prototipagem** para isolar e definir as características de qualidade.

O processo de descobrir os pré-requisitos do sistema é geralmente a fase de análise do sistema. O custo/benefício de se aplicar adequadamente os recursos para esta fase é muito grande.

“Pode ser muito caro desenvolver o software errado”

Qualidade e o desenvolvimento software

O processo de desenvolvimento do software é onde os desenvolvedores traduzem os pré-requisitos em software.

É certo que a qualidade do software está diretamente ligada à qualidade dos processos utilizados para o desenvolvimento.

Um bom desenvolvimento de software deve capacitar à organização a definição da consistência do produtos de qualidade. A comunidade de software está vendo que o desenvolvimento do produto deve ser feito de maneira muito rápida. O ciclo de vida do produto é agora um negócio crítico para muitos desenvolvedores.

Os consumidores de software necessitam de produtos cada vez melhores e mais rápidos de serem desenvolvidos para aumentar a sua competitividade no mercado global.

Se estes objetivos forem cumpridos, o desenvolvimento de software, deve:

- Utilizar as melhores práticas da engenharia de software;
- Ser operado por pessoal treinado com responsabilidades e instruções;
- Dar ênfase na prevenção de defeitos assim que forem detectados
- Gerar registro para demonstrar efetividade e eficiência;
- Utilizar destes registros para aumentar a performance no futuro.

IV. - QUALIDADE E SERVIÇO DE SUPORTE AO USUÁRIO

O suporte ao usuário é complexo e deve incluir:

- Documentação para o usuário, incluindo ajuda on-line;
- Empacotamento e distribuição organizados;

- Implementação e customização de serviços e consultas;
- Treinamento;
- Assistência help-desk;
- Relatórios de erros e correções;
- Melhoramento do software.

Para uma aplicação instalada em um local, os requisitos de suporte podem ser fornecidos via telefone e uma pessoa deve ser treinada e encarregada de receber as ligações.

No caso de ser uma aplicação de nível internacional, ela deve prover ajuda pela Internet e implementar um sistema administrativo para suportar um sistema de help-desk em nível internacional.

O que é um Sistema de Qualidade ?

Aplicar os princípios da qualidade de software é o início para o sucesso. O termo “sistema de qualidade” é utilizado internacionalmente para descrever um processo na qual garante e demonstra a qualidade dos produtos e serviços ofertados pela empresa.

A padronização ISO 9000 define e descreve o que requerido ou satisfatório em um sistema de qualidade contendo componentes de desenho e desenvolvimento. As padronizações existentes para garantir a qualidade de software serão estudadas mais à frente neste trabalho.

Além das padronizações ISO, muitas outras organizações nacionais e internacionais promovem padrões que descrevem sistemas de qualidade para serem aplicado em sistemas de desenvolvimento e suporte em certas circunstâncias, a exemplificar o CMM (Capability maturity model).

O termo “gerenciamento de sistemas de qualidade” é utilizado às vezes. Este termo dá ênfase às necessidades do processo de qualidade para serem gerenciados para garantir que continue de forma correta e eficiente.

Tão importante quanto as práticas e ferramentas é o status da pessoa que as usa. A qualidade deve garantir que as pessoas envolvidas devem ter suas habilidades certas para cada tipo de trabalho de uma maneira profissional. Se as pessoas necessitam de treinamento, então a empresa deverá treinar os seus usuários. Deve-se garantir que as pessoas entendam suas responsabilidades e como seu trabalho se relaciona com outras pessoas.

Um sistema de qualidade dá grande ênfase à correção de erros. É muito útil corrigir os erros durante o início do ciclo de vida do sistema.

Melhor ainda, é anular erros antes mesmo deles serem feitos Um sistema de qualidade de sucesso inclui maneiras de registrar os erros para determinar as causas e agir de acordo com o erro eliminando suas causas.

Em suma, um sistema de qualidade é tudo que o gerenciamento utiliza para garantir e demonstrar a qualidade do software e do serviço de suporte. O sistema de qualidade é o trabalho completo, incluindo política, procedimentos, ferramentas e recursos, incluindo humano e tecnológico.

V. - CERTIFICAÇÃO DA QUALIDADE

Um aspecto interessante da qualidade é que não basta que ela exista. Ela deve ser **reconhecida** pelo cliente. Por causa disso, é necessário que exista algum tipo de **certificação oficial**, emitida com base em um padrão. Alguns tipos de certificados são bastante conhecidos, como:

- ✓ O selo do SIF de inspeção da carne
- ✓ O selo da ABIC nos pacotes de café
- ✓ O certificado da Secretaria de Saúde para restaurantes (classe "A" são os melhores)
- ✓ A classificação em estrelas dos hotéis (hotéis com cinco estrelas são ótimos)
- ✓ Os certificados de qualidade da série ISO-9000

Ouvimos muitas propagandas de empresas falando de sua certificação ISO-9000. Isto nada mais é do que um padrão de qualidade (reconhecido mundialmente) pelo qual esta empresa foi avaliada e julgada. Para que seja possível realizar uma avaliação e um julgamento, é necessário haver um padrão ou norma. Existem alguns organismos normalizadores reconhecidos mundialmente:

ISO - International Organization for Standardization

IEEE - Instituto de Engenharia Elétrica e Eletrônica

ABNT - Associação Brasileira de Normas Técnicas

A norma ISO-9000, por exemplo, foi criada pela ISO para permitir que todas as empresas do mundo possam avaliar e julgar sua qualidade. Existindo um padrão único mundial, uma empresa do Brasil, mesmo não tendo nenhum contato com uma outra empresa na Europa, pode garantir a ela a qualidade de seu trabalho.

A Certificação em uma norma ou padrão é a emissão de um documento oficial indicando a conformidade com esta determinada norma ou padrão. É claro que, antes da emissão do certificado, é preciso realizar todo um processo de avaliação e julgamento de acordo com uma determinada norma. Embora uma empresa possa auto-avaliar-se ou ser avaliada por seus próprios clientes, o termo Certificação costuma ser aplicado apenas quando efetuado por uma empresa independente e idônea, normalmente especializada neste tipo de trabalho.

No Brasil, o INMETRO é o órgão do governo responsável pelo credenciamento destas instituições que realizam a certificação de sistemas de qualidade.

Qualidade Produto x Qualidade Processo

Uma das evoluções mais importantes no estudo da qualidade está em notar que a qualidade do produto é algo bom, mas que qualidade do processo de produção é ainda mais importante. No caso do prato de comida, por exemplo, pode-se dizer mais sobre a qualidade observando como o prato foi preparado do que analisando o produto final.

Afinal, não se consegue ter certeza da higiene ou o valor nutricional apenas comendo o prato.

Esta descoberta aconteceu durante a própria evolução dos conceitos de qualidade, ao longo dos anos. Observe na tabela abaixo como aconteceu esta evolução:

1. Inspeção pós-produção	Avalia o produto final, depois de pronto	1900
2. Controle estatístico da produção	Avalia os subprodutos das etapas de produção	1940
3. Procedimento de produção	Avalia todo o procedimento de produção	1950
4. Educação das pessoas	Avalia as pessoas envolvidas no processo	1960
5. Otimização dos processos	Avalia e otimiza cada processo	1970
6. Projeto robusto	Avalia o projeto de produção	1980
7. Engenharia simultânea	Avalia a própria concepção do produto	1990

Hoje em dia, pode-se consultar normas e padrões tanto para produtos quanto para processos. Obviamente, **os certificados mais valiosos são aqueles que certificam o processo de produção de um produto** e não aqueles que simplesmente certificam o produto.

Entretanto, é comum encontrar empresas que perseguem os dois tipos de padrão de qualidade.

VI. - QUALIDADE DE SOFTWARE:

Agora que se tem o conhecimento sobre o que é qualidade e como ela pode ser avaliada, vamos tentar aplicar estes conceitos aos produtos de software e ao processo de desenvolvimento de software. Inicialmente, vamos encontrar um grande problema: muitas pessoas acham que criar programas é uma arte que não pode seguir regras, normas ou padrões. Isto acontece principalmente porque:

- Produtos de software são complexos, até mais do que o hardware onde executam
- Software não têm produção em série. Seu custo está no projeto e desenvolvimento
- Software não se desgasta e nem se modifica com o uso
- Software é invisível. Sua representação em gráficos e diagramas não é precisa.
- A Engenharia de Software ainda não está madura, é uma tecnologia em evolução
- Não há um acordo entre os profissionais da área sobre o que é Qualidade de Software

Apesar de tudo isso, precisamos entender que **o problema não está no Software** em si, mas **na forma como as pessoas tem desenvolvido software até os dias de hoje**. Precisamos nos conscientizar que necessitamos aplicar na indústria de software os conceitos de qualidade, urgentemente.

Atualmente, muitas instituições se preocupam em criar normas para permitir a correta avaliação de qualidade tanto de produtos de software quanto de processos de desenvolvimento de software. Apenas para ter uma visão geral, observe o quadro a seguir, com as principais normas nacionais e internacionais nesta área:

Norma	Comentário
ISO 9126	Características da qualidade de produtos de software.
NBR 13596	Versão brasileira da ISO 9126
ISO 14598	Guias para a avaliação de produtos de software, baseados na utilização prática da norma ISO 9126
ISO 12119	Características de qualidade de pacotes de software (software de prateleira, vendido com um produto embalado)
IEEE P1061	Standard for Software Quality Metrics Methodology (produto de software)
ISO 12207	Software Life Cycle Process. Norma para a qualidade do processo de desenvolvimento de software.
NBR ISO 9001	Sistemas de qualidade - Modelo para garantia de qualidade em Projeto, Desenvolvimento, Instalação e Assistência Técnica (processo)
NBR ISO 9000-3	Gestão de qualidade e garantia de qualidade. Aplicação da norma ISO 9000 para o processo de desenvolvimento de software.
NBR ISO 10011	Auditoria de Sistemas de Qualidade (processo)
CMM	Capability Maturity Model. Modelo da SEI (Instituto de Engenharia de Software do Departamento de Defesa dos USA) para avaliação da qualidade do processo de desenvolvimento de software. Não é uma norma ISO, mas é muito bem aceita no mercado.
SPICE ISO 15504	Projeto da ISO/IEC para avaliação de processo de desenvolvimento de software. Ainda não é uma norma oficial ISO, mas o processo está em andamento.

Engenharia de Software

A disciplina que nos ajuda a entender o processo de desenvolvimento de software é a *Engenharia de Software*. É através dela que poderemos chegar à qualidade. Existe, entretanto, um grande problema a ser resolvido: tecnicamente, ela não existe.

O problema é que, para que uma disciplina seja considerada realmente uma Engenharia, é necessário atender a alguns requisitos básicos que a Engenharia de Software, pelos menos até agora, não atende. A seguir a definição de Engenharia:

"A Engenharia deve criar soluções com uma relação custo-benefício adequada para problemas práticos, pela aplicação de conhecimentos científicos, para construir coisas a serviço da humanidade"

Dentro destes conceitos, a Engenharia de Software falha principalmente no que diz respeito à adequação do custo-benefício e à aplicação, em toda a sua extensão, de conhecimentos científicos. Atualmente, estes requisitos são atendidos apenas em parte.

É necessário definir, portanto, o que é exatamente a Engenharia de Software. A seguir algumas tentativas de definição:

"...é a disciplina que integra métodos, ferramentas e procedimentos para o desenvolvimento de software para computadores."

"...é uma coleção de processos de gerenciamento, ferramental de software e atividades de projeto para o desenvolvimento de software. "

"...é um termo usado para referir-se a modelos de ciclo de vida, metodologias de rotina, técnicas de estimativa de custo, estruturas de documentação, ferramentas de gerenciamento de configuração, técnicas de garantia de qualidade e outras técnicas de padronização da atividade de produção de software."

Qualidade de Produtos de Software - ISO 9126

Quando se pensa em qualidade de um "produto físico", é fácil imaginar padrões de comparação, provavelmente ligado às dimensões do produto ou alguma outra característica física. Quando se trata de software, como podemos definir exatamente o que é a qualidade? Parece difícil...

Felizmente, para nós, a **ISO (Organização Internacional de Padrões)** já pensou bastante sobre o assunto. O suficiente para publicar uma norma que representa a atual padronização mundial para a qualidade de produtos de software. Esta norma chama-se ISO/IEC 9126 e foi publicada em 1991. Ela é uma das mais antigas da área de qualidade de software e já possui sua tradução para o Brasil, publicada em agosto de 1996 como NBR 13596.

Mas, afinal de contas, o que está escrito nesta norma ISO/IEC 9126 ou na NBR 13596? Bem, estas normas listam o conjunto de características que devem ser verificadas em um software para que ele seja considerado um "software de qualidade". São seis grandes grupos de características, cada um dividido em algumas subcaracterísticas.

Os nomes dados pelo ISO/IEC para as características e subcaracterísticas são um pouco .Entretanto, uma pessoa que trabalha com software não terá dificuldade em entendê-las. Observe na tabela abaixo a lista completa:

Característica	Sub-característica	Pergunta chave para a subcaracterística
Funcionalidade (satisfaz as necessidades?)	Adequação	Propõe-se a fazer o que é apropriado?
	Acurácia	Faz o que foi proposto de forma correta?
	Interoperabilidade	Interage com os sistemas especificados?
	Conformidade	Está de acordo com as normas, leis, etc.?
	Segurança de acesso	Evita acesso não autorizado aos dados?
Confiabilidade (é imune a falhas?)	Maturidade	Com que frequência apresenta falhas?
	Tolerância a falhas	Ocorrendo falhas, como ele reage?
	Recuperabilidade	É capaz de recuperar dados em caso de falha?
Usabilidade (é fácil de usar?)	Intelegibilidade	É fácil entender o conceito e a aplicação?
	Apreensibilidade	É fácil aprender a usar?
	Operacionalidade	É fácil de operar e controlar?
Eficiência (é rápido e "enxuto"?)	Tempo	Qual é o tempo de resposta, a velocidade de execução?
	Recursos	Quanto recurso usa? Durante quanto tempo?
Manutenibilidade (é fácil de modificar?)	Analisabilidade	É fácil de encontrar uma falha, quando ocorre?
	Modificabilidade	É fácil modificar e adaptar?
	Estabilidade	Há grande risco quando se faz alterações?
	Testabilidade	É fácil testar quando se faz alterações?
Portabilidade (é fácil de usar em outro ambiente?)	Adaptabilidade	É fácil adaptar a outros ambientes?
	Capac. para ser instalado	É fácil instalar em outros ambientes?
	Conformidade	Está de acordo com padrões de portabilidade?
	Capac. Para substituir	É fácil usar para substituir outro?

VII - MÉTRICAS DE SOFTWARE

Embora a atual norma ISO 9126/NBR 13596 enumere as características e sub-características um software, ela ainda não define como dar uma nota a um software em cada um destes itens. A não familiarização com o processo de avaliação de software, pode-se ter dificuldades em tentar utilizar a norma. Se pretende avaliar um software segundo esta norma, deve tentar atribuir valores (como se fossem notas ou conceitos) a cada uma das sub-características.

Algumas características podem ser realmente medidas, como o tempo de execução de um programa, número de linhas de código, número de erros encontrados em uma sessão de teste ou o tempo médio entre falhas. Nestes casos, é possível utilizar uma técnica, uma ferramenta ou um software para realizar medições. Em outros casos, a característica é tão subjetiva que não existe nenhuma forma óbvia de medi-la.

Ficam, portanto, as questões: como dar uma nota, em valor numérico, a uma característica inteiramente subjetiva? O que representa, por exemplo, uma "nota 10" em termos de "Segurança de Acesso"? Quando se pode dizer que a "Intelegibilidade" de um software pode ser considerada "satisfatória"? Criou-se, então, uma área de estudo à parte dentro da Qualidade de Software conhecida como **Métricas de Software**. O que se pretende fazer é definir, de forma precisa, como medir numericamente uma determinada característica.

Para avaliar uma determinada subcaracterística subjetiva de forma simplificada, por exemplo, pode-se criar uma série de perguntas do tipo "sim ou não". Crie as perguntas de forma tal que as respostas "sim" sejam aquelas que indicam uma melhor nota para a característica. Depois de prontas as perguntas, basta avaliar o software, respondendo a cada pergunta. Se conseguir listar 10 perguntas e o software obtiver uma resposta "sim" em 8 delas, terá obtido um valor de 80% nesta característica.

Obviamente, a técnica acima não é muito eficiente. Para melhorá-la, entretanto, pode-se garantir um número mínimo perguntas para cada característica. Além disso, algumas perguntas mais importantes podem ter pesos maiores. É possível, ainda, criar perguntas do tipo ABCDE, onde cada resposta indicaria um escore diferenciado. Alguns estudiosos sugerem formas diferentes de medir uma característica, baseada em conceitos do tipo "não satisfaz", "satisfaz parcialmente", "satisfaz totalmente" e "excede os padrões". Estes conceitos, embora parecem muito subjetivos, não deixam de ser uma forma eficiente de medir uma característica.

Em todos os casos, um fato fica claro: nada ajuda mais a avaliar características de um software do que um avaliador experiente, que já realizou esta tarefa diversas vezes e em diversas empresas diferentes. Afinal, medir é comparar com padrões e um avaliador experiente terá maior sensibilidade do que um profissional que acaba de ler uma norma pela primeira vez.

Atualmente, a norma ISO/IEC 9126 está sendo revisada. A revisão, que deverá estar pronta nos próximos anos, não deverá modificar nenhuma das características básicas da 9126. A maior modificação será a inclusão de dois documentos adicionais para descrever métricas externas (relativas ao uso do produto) e métricas internas (relativas à arquitetura do produto). A seguir algumas das modificações previstas para esta revisão:

- Algumas novas subcaracterísticas: **Conformidade** fará parte de todas as características. **Atratividade** será uma subcaracterística de **Usabilidade**. Capacidade de coexistir será uma subcaracterística de portabilidade.
- A norma será dividida em três partes. A primeira (9126-1) incluirá definições e características. As duas seguintes descreverão métricas externas (9126-2) e internas (9126-3).
- A versão brasileira da revisão desta norma deverá ser chamada de NBR 9126-1, 9126-2 e 9126-3, segundo a numeração original da ISO/IEC.

VIII - GUIAS PARA AVALIAÇÃO DA QUALIDADE - ISO 14598

Nota-se a necessidade de mais detalhes sobre como avaliar a qualidade de um software. As características e subcaracterísticas da norma ISO/IEC 9126 apenas começaram o trabalho. Faltava definir, em detalhes, como atribuir um conceito para cada item. Afinal, sem uma padronização, que valor teria uma avaliação?

A ISO, consciente deste problema, está finalizando o trabalho em um conjunto de Guias para a Avaliação da Qualidade segundo a norma ISO/IEC 9126. Estes guias descrevem, detalhadamente, todos os passos para que se avalie um software. Embora o trabalho nesta norma ainda não esteja totalmente pronto, já existem informações detalhadas sobre o que será esta norma, quando for oficialmente publicada.

Esta nova norma trará muitos recursos interessantes aos avaliadores, já que trata o processo de avaliação em grande detalhe. Ela leva em conta a existência de três grupos interessados em avaliar um software, o que define os três tipos básicos de certificação:

Certificação	Quem realiza	Finalidade
de 1a. parte	Empresas que desenvolvem software	Melhorar a qualidade de seu próprio produto
de 2a. parte	Empresas que adquirem software	Determinar a qualidade do produto que irão adquirir
de 3a. parte	Empresas que fazem certificação	Emitir documento oficial sobre a qualidade de um software

Esta norma se constituirá, na verdade, de seis documentos distintos, relacionados entre si, como demonstrado a seguir:

Norma	Nome	Finalidade
14598-1	Visão Geral	Ensina a utilizar as outras normas do grupo
14598-2	Planejamento e Gerenciamento	Sobre como fazer uma avaliação, de forma geral
14598-3	Guia para Desenvolvedores	Como avaliar sob o ponto do vista de quem desenvolve

14598-4	Guia para Aquisição	Como avaliar sob o ponto de vista de quem vai adquirir
14598-5	Guia para Avaliação	Como avaliar sob o ponto de vista de quem certifica
14598-6	Módulos de Avaliação	Detalhes sobre como avaliar cada característica

Em resumo, esta nova norma complementar a ISO/IEC 9126 e permitirá uma avaliação padronizada das características de qualidade de um software. É importante notar que, ao contrário da 9126, a 14598 vai a detalhes mínimos, incluindo modelos para relatórios de avaliação, técnicas para medição das características, documentos necessários para avaliação e fases da avaliação. Como um exemplo, observe um modelo de relatório de avaliação, segundo um anexo da norma 14598-5:

Seção	Itens
1 - Prefácio	Identificação do avaliador Identificação do relatório de avaliação Identificação do contratante e fornecedor
2 - Requisitos	Descrição geral do domínio de aplicação do produto Descrição geral dos objetivos do produto Lista dos requisitos de qualidade, incluindo - Informações do produto a serem avaliadas - Referências às características de qualidade - Níveis de avaliação
3 - Especificação	Abrangência da avaliação Referência cruzada entre os requisitos de avaliação e os componentes do produto Especificação das medições e dos pontos de verificação Mapeamento entre a especificação das medições com os requisitos de avaliação
4 - Métodos	Métodos e componentes nos quais o método será aplicado
5 - Resultado	Resultados da avaliação propriamente ditos Resultados intermediários e decisões de interpretação Referência às ferramentas utilizadas

As normas 14598-1, 14598-4 e 14598-5 já foram publicadas. As demais estão em processo de finalização. Está sendo feito pela ABNT um trabalho de tradução desta norma (tanto dos itens já publicados quanto das versões preliminares dos itens restantes). Com isso, esta norma terá sua versão brasileira pouco tempo depois do final de sua publicação pela ISO.

IX - QUALIDADE DE PACOTES DE SOFTWARE - ISO 12119

Esta norma foi publicada em 1994 e trata da avaliação de pacotes de software, também conhecidos como "software de prateleira". Além de estabelecer os requisitos de qualidade para este tipo de software, ela também destaca a necessidade de instruções para teste deste pacote, considerando estes requisitos. A norma divide-se em itens, da seguinte forma:

Item	Descrição
1. Escopo	
2. Definições	
3. Requisitos de qualidade	
3.1. Descrição do Produto	Descreve o produto, de forma a ajudar o comprador em potencial, servindo como base para testes. Cada declaração deve ser correta e testável. Deve incluir declarações sobre funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade.
3.2. Documentação do usuário	Deve ser completa, correta, consistente, fácil de entender e capaz de dar uma visão geral do produto.
3.3. Programas e dados	Descreve em detalhes cada uma das funções do software, incluindo declarações sobre funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade.
4. Instruções para teste	
4.1. Pré-requisitos de teste	Lista de itens necessários ao teste, incluindo documentos incluídos no pacote, componentes do sistema e material de treinamento.
4.2. Atividades de teste	Instruções detalhadas sobre os procedimentos de teste, inclusive instalação e execução de cada uma das funções descritas.
4.3. Registro de teste	Informações sobre como os testes foram realizados, de tal forma a permitir uma reprodução destes testes. Deve incluir parâmetros utilizados, resultados associados, falhas ocorridas e até a identidade do pessoal envolvido.
4.4. Relatório de teste	Relatório incluindo: identificação do produto, hardware e software utilizado, documentos utilizados, resultados dos testes, lista de não conformidade com os requisitos, lista de não conformidade com as recomendações, datas, etc.

Um dos grandes méritos desta norma está na profundidade com que são descritas cada uma das características e subcaracterísticas mencionadas na norma 9126. A norma inclui detalhes que devem estar presentes no produto, tais como:

- Documentação do usuário de fácil compreensão
- Um sumário e um índice remissivo na documentação do usuário
- Presença de um Manual de instalação com instruções detalhadas
- Possibilidade de verificar se uma instalação foi bem sucedida

- Especificação de valores limites para todos os dados de entrada, que deverão ser testados
- Operação normal mesmo quando os dados informados estão fora dos limites especificados
- Consistência de vocabulário entre as mensagens e a documentação
- Função de auxílio (help) com recursos de hipertexto
- Mensagens de erro com informações necessárias para a solução da situação de erro
- Diferenciação dos tipos de mensagem: confirmação, consulta, advertência e erro
- Clareza nos formatos das telas de entrada e relatórios
- Capacidade de reverter funções de efeito drástico
- Alertas claros para as conseqüências de uma determinada confirmação
- Identificação dos arquivos utilizados pelo programa
- Identificação da função do programa que está sendo executada no momento
- Capacidade de interromper um processamento demorado

Outras características importante são a ênfase nos testes e os modelos de relatórios incluídos. Tudo isso facilita grandemente o trabalho do avaliador. Uma versão traduzida desta norma será publicada em breve ABNT.

X - QUALIDADE DE PROCESSO DE SOFTWARE:

Os estudos sobre qualidade mais recentes são na sua maioria voltados para o melhoramento do processo de desenvolvimento de software. Não é que a qualidade do produto não seja importante, ela é. Mas o fato é que, ao garantir a qualidade do processo, já se está dando um grande passo para garantir também a qualidade do produto.

O estudo da Qualidade do Processo de Software é uma área ligada diretamente à Engenharia de Software. O estudo de um ajuda a entender e aprimorar o outro. Em ambas as disciplinas, estuda-se modelos do processo de desenvolvimento de software. Estes modelos são uma tentativa de explicar em detalhes **como** se desenvolve um software, quais são as etapas envolvidas. É necessário compreender cada pequena tarefa envolvida no desenvolvimento.

XI – A SÉRIE ISO 9000 :

Esta série é um conjunto de normas da ISO que define padrões para garantia e gerenciamento da qualidade. Algumas destas normas estão descritas abaixo:

Norma	Trata de
ISO 9001	Modelo para garantia da qualidade em projeto, desenvolvimento, produção, instalação e assistência técnica.
ISO 9002	Modelo para garantia da qualidade em produção e instalação
ISO 9003	Modelo para garantia da qualidade em inspeção e ensaios finais
ISO 9000-1	Diretrizes para escolher entre as normas ISO 9001, 9002 e 9003
ISO 9000-3	Orientação para a aplicação da ISO 9001 em Software

Entre as normas 9001, 9002 e 9003, a primeira é a que mais se adequa ao desenvolvimento e manutenção de software. Como toda norma deste grupo, ela é usada para garantir que um fornecedor atende aos requisitos especificados nos diversos estados do desenvolvimento.

Estes estágios incluem projeto, desenvolvimento, produção, instalação e suporte.

A norma ISO 9000-3 (não confundir com a ISO 9003) traz os roteiros para aplicar a ISO 9001 especificamente na área de desenvolvimento, fornecimento e manutenção de software. Todas as orientações giram em torno de uma "situação contratual", onde uma outra empresa contrata a empresa em questão para desenvolver um produto de software.

Na tabela abaixo seguem os processos definidos na ISO 9000-3:

Grupo	Atividade
Estrutura do Sistema de Qualidade	Responsabilidade do fornecedor Responsabilidade do comprador Análise crítica conjunta
Atividades do Ciclo de Vida	Análise crítica do contrato Especificação dos requisitos do comprador Planejamento do desenvolvimento Projeto e implementação Testes e validação Aceitação Cópia, entrega e instalação Manutenção
Atividades de Apio	Gerenciamento de configuração Controle de documentos Registros da qualidade Medição Regras, convenções Aquisição Produto de software incluído Treinamento

O processo de certificação de uma empresa de software segundo as normas ISO 9001 / 9000-3 segue um conjunto de passos bem definidos:

- A empresa estabelece o seu sistema de qualidade.
- A empresa faz uma solicitação formal a um órgão certificador, incluindo detalhes do negócio da empresa, escopo da certificação solicitada e cópia do manual de qualidade. O órgão certificador faz uma visita à empresa, colhe mais dados e explica o processo de certificação. O órgão certificador verifica se a documentação do sistema de qualidade está de acordo com a norma ISO. O órgão certificador envia uma equipe à empresa com fins de auditoria. Nesta visita, será verificado se todos na empresa cumprem o que está documentado no manual de qualidade.
- O órgão certificador emite o certificado de qualidade.

- O órgão certificador realiza visitas periódicas à empresa para assegurar que o sistema continua sendo efetivo

XII - ISO 12207 – PROCESSO DE VIDA DO CICLO DE SOFTWARE

Este padrão formaliza a arquitetura do ciclo de vida do software, que é um assunto básico em Engenharia de Software e também em qualquer estudo sobre Qualidade do Processo de Software. Esta norma possui mais de 60 páginas e detalha os diversos processos envolvidos no ciclo de vida do software.

Estes processos estão divididos em três classes: Processos Fundamentais, Processos de Apoio e Processos Organizacionais.

Segue a lista completa dos processos na tabela abaixo:

Processos Fundamentais	Início e execução do desenvolvimento, operação ou manutenção do software durante o seu ciclo de vida.
Aquisição	Atividades de quem um software. Inclui: definição da necessidade de adquirir um software (produto ou serviço), pedido de proposta, seleção de fornecedor, gerência da aquisição e aceitação do software.
Fornecimento	Atividades do fornecedor de software. Inclui preparar uma proposta, assinatura de contrato, determinação recursos necessários, planos de projeto e entrega do software.
Desenvolvimento	Atividades do desenvolvedor de software. Inclui: análise de requisitos, projeto, codificação, integração, testes, instalação e aceitação do software.
Operação	Atividades do operador do software. Inclui: operação do software e suporte operacional aos usuários.
Manutenção	Atividades de quem faz a manutenção do software.
Processos de Apoio	Auxiliam um outro processo.
Documentação	Registro de informações produzidas por um processo ou atividade. Inclui planejamento, projeto, desenvolvimento, produção, edição, distribuição e manutenção dos documentos necessários a gerentes, engenheiros e usuários do software.
Gerência de Configuração	Identificação e controle dos itens do software. Inclui: controle de armazenamento, liberações, manipulação, distribuição e modificação de cada um dos itens que compõem o software.
Garantia da Qualidade	Garante que os processos e produtos de software estejam em conformidade com os requisitos e os planos estabelecidos.
Verificação	Determina se os produtos de software de uma atividade atendem completamente aos requisitos ou condições impostas a eles.
Validação	Determina se os requisitos e o produto final (sistema ou software) atendem ao uso específico proposto.
Revisão Conjunta	Define as atividades para avaliar a situação e produtos de uma

	atividade de um projeto, se apropriado.
Auditoria	Determina adequação aos requisitos, planos e contrato, quando apropriado.
Resolução de Problemas	Análise e resolução dos problemas de Qualquer natureza ou fonte, descobertos durante a execução do desenvolvimento, operação, manutenção ou outros processos. .
Processos Organizacionais	Implementam uma estrutura constituída de processos de ciclo de vida e pessoal associados, melhorando continuamente a estrutura e os processos.
Gerência	Gerenciamento de processos.
Infra-estrutura	Fornecimento de recursos para outros processos. Inclui: hardware, software, ferramentas, técnicas, padrões de desenvolvimento, operação ou manutenção.
Melhoria	Atividades para estabelecer, avaliar, medir, controlar e melhorar um processo de ciclo de vida de software.
Treinamento	Atividades para prover e manter pessoal treinado.

A norma detalha cada um dos processos acima. Ela define ainda como eles podem ser usados de diferentes maneiras por diferentes organizações (ou parte destas), representando diversos pontos de vista para esta utilização. Cada uma destas visões representa a forma como uma organização emprega estes processos, agrupando-os de acordo com suas necessidades e objetivos.

As Visões têm o objetivo de organizar melhor a estrutura de uma empresa, para definir suas gerências e atividades alocadas às suas equipes. Existem cinco visões diferentes: contrato, gerenciamento, operação, engenharia e apoio. Na figura abaixo se encontra como estas visões se relacionam aos processos.

A ISO/IEC 12207 é a primeira norma internacional que descreve em detalhes os processos, atividades e tarefas que envolvem o fornecimento, desenvolvimento, operação e manutenção de produtos de software. A principal finalidade desta norma é servir de referência para os demais padrões que venham a surgir. Lançada em agosto de 1995, ela é citada em quase todos os trabalhos relacionados à Engenharia de Software desde então, inclusive aqueles relativos à qualidade. A futura norma ISO 15504 (SPICE), por exemplo, organiza seu trabalho segundo o que está descrito na 12207.

A versão brasileira da norma foi encaminhada para votação na ABNT em junho de 1997 e a expectativa da comissão encarregada da tradução é que ela se transforme em norma brasileira ainda em 1997.

XIII – GERENCIANDO UM COMPANHIA DE QUALIDADE

Gerenciar uma companhia de qualidade é mais do que implementar um sistema de qualidade consistindo de um conjunto de técnicas do padrão Iso.

É a criação da “cultura da qualidade” que permeia toda a organização.

A seguir, mostraremos algumas idéias práticas que podem vir a ajudar na realização de trabalho com mais qualidade.

Dedicação à satisfação do cliente

Qualidade fala sobre a satisfação do cliente. Produzir um produto de qualidade é uma parte importante para garantir a satisfação e atenção do cliente. Isto inclui:

- Esforçar-se ao máximo para entender as necessidades do cliente no produto e no suporte;
- Escrever um contrato que reflita estas necessidades;
- Prever suporte pós-venda para igualar requisitos ou expectativas;
- Vigiar os contatos com os clientes. A primeira impressão é a que fica.

Dar ênfase à melhoramentos contínuos

O fim da qualidade total nunca será alcançado, será sempre possível fazer as coisas um pouco melhor ou mais rápido. A melhor coisa a ser feita é fazer a organização se tornar uma “Organização do aprendizado” que constantemente pesquisa e renova recursos e experiências de todo o pessoal envolvido para aumentar a qualidade, reduzir custos e responder rapidamente às necessidades dos clientes. Tendo se tornado uma organização do aprendizado, aprenda como aprender mais rápido do que os concorrentes.

Tratar fornecedores como parceiros de negócios

Esta filosofia de parceiros de negócios pode ser estendida a outros com uma participação no sucesso do seu negócio. As empresas de desenvolvimento devem considerar que seus agentes e distribuidores devem ser considerados como parceiros do negócio para benefício mútuo. E o mesmo pode ser aplicado a grandes consumidores, que estão procurando relacionamentos.

Decisões em negócios de parcerias não tem razões idealistas, mas têm no gerenciamento do negócio a luz da realidade estratégica e comercial. Eles refletem o fato de o negócio não ser uma coisa qualquer e que é geralmente em seu próprio interesse promover o sucesso dos outros. Em outras palavras é criada uma situação “Vencedor-Vencedor”.

Comunicação e time de trabalho

É importante criar uma cultura na empresa no qual indivíduos e departamentos pensem que eles mesmos são seus próprios consumidores dentro da organização. Este

supridores internos tentam entender e suprir as necessidades de seus consumidores internos e estes trabalham com seus supridores internos para ajudá-los.

Responsabilidades e linhas de comunicação deve ser regularmente revistos, e relacionamentos tradicionais entre departamentos devem ser alterados quando for necessário.

Atualizando empregados

O pessoal da empresa não será capaz de produzir um sistema de qualidade e fornecer um suporte de qualidade a menos que eles tenham as habilidades necessárias, conhecimento e recurso. O gerente deve selecionar, em primeiro lugar, o pessoal certo para cada tarefa a ser executada de acordo com suas habilidades e treinamento que eles venham a precisar para conseguir realizar o seu melhor. Isto motivará o pessoal e gerará um compromisso para um contínuo melhoramento do seu conhecimento e do seu trabalho.

Compromisso da gerência

A gerência pode ter uma melhor responsabilidade se:

- Tratar a qualidade como um negócio estratégico;
- Desenvolver uma política para a qualidade;
- Garantir que a política da qualidade será comunicada e implementada através da organização
- Regularmente rever a política e suas implementações.

XIV. – IMPLANTANDO UM SISTEMA DE QUALIDADE

Como alcançar a qualidade? Alguns anos atrás, John Ruskin ditou alguns pontos quando disse que a qualidade nunca é um acidente é sempre o resultado de grandes esforços. Os dizeres de Ruskin foram feitos durante uma idade mais vagarosa, mais ainda são aplicáveis. Para os negócios modernos, resultados inteligentes devem ser direcionados através de um sistema de qualidade.

XIV.1 - UM SISTEMA DE QUALIDADE

Um programa de aumento de qualidade conduz ao estabelecimento de um sistema de qualidade que deve ter aspectos técnicos e culturais. É fácil de ver a razão disto: a entidade a ser melhorada consiste em tecnologia e pessoas.

Aspectos Técnicos

Envolve o desenvolvimento de padrões e técnicas para implementar a qualidade em todas as atividades.

Aspectos culturais

A prática da qualidade deve ser aceita por todas as pessoas da empresa sem exceção. Cada pessoa deve estar consciente da sua responsabilidade pela qualidade.

Um programa de educação e treinamento envolvendo a empresa toda deve ser necessário para conseguir conscientizar a todos na empresa.

XIV.2 – INICIANDO UM SISTEMA DE QUALIDADE

Preparar uma política de qualidade

A gerência da empresa deverá começar um projeto de qualidade formulando uma política de qualidade. Esta política deve ser claramente declarada pela “comissão de organização da qualidade” e gerentes que estarão envolvidos no projeto.

A política deverá ser publicada e comunicada para que seja entendida e implementada em todos os setores da empresa.

Estabelecer uma equipe de suporte em qualidade

Uma equipe de suporte à qualidade normalmente compreende um comitê de direção e uma equipe de aumento da qualidade. A gerência sênior da companhia deve estar no comitê de direção. A tarefa do comitê é verificar o programa de qualidade.

As tarefas principais do comitê de direção são:

- Definir uma direção estratégica e metas a longo prazo;
- Estabelecer uma equipe de aumento e melhoramento da qualidade e rever sua performance;
- Autorizar e aprovar gastos para o programa de qualidade;
- Prover suporte de alto nível para o programa de qualidade.

O comitê de direção deverá adotar uma equipe de melhoramento da qualidade que será responsável por estabelecer e manter um programa de qualidade em toda a organização. As atividades específicas são:

- Avaliar as necessidades da organização;
- Desenvolver um sistema de qualidade para a empresa;
- Planejar e monitorar a implementação;

Outras atividades poderão estar sendo desenvolvidas na organização, a saber:

- Comunicar, durante todo o processo, o programa de qualidade ao pessoal da empresa;
- Treinamento e outras atividades para ajudar no suporte ao programa de qualidade;
- Preparar e rever procedimentos e padrões;
- Adotar um ciclo de vida para o desenvolvimento do programa;
- Adotar métodos e ferramentas;
- Estabelecer um programa para medir os processos do software, produtos e serviços.

XIV.3 - DEFINIR UM PROGRAMA PARA A QUALIDADE

A seguir, mostraremos alguns passos que podem ser utilizados para construir e definir corretamente um programa de implementação da qualidade na empresa.

Avaliar a organização

Para se avaliar a organização, chama-se um órgão respeitado internacionalmente para poder avaliar os processos da empresa de software.

Esta organização vai entrar dentro da empresa e começar a verificar e definir padrões e regras que devem ser cumpridas para se conseguir o certificado.

Esta organização que estará sendo avaliada, vai reorganizar-se de acordo com as normas a serem ditadas.

As normas podem variar de acordo com a necessidade da empresa e ser aplicado padrões internacionais, tais como ISO 9000 ou CMM ,

Projetar um sistema de qualidade

Nesta parte da implementação, a empresa vai definir quais os seus objetivos perante a qualidade que se quer alcançar.

Alguns dos objetivos mais comuns que as empresas querem para se adaptar ao mercado, é:

- Reduzir o número de erros e defeitos encontrados na fase de teste e na implantação do software;
- Aumentar a produtividade;
- Reduzir o tempo de desenvolvimento;
- Diminuir o tempo de resposta às requisições do cliente;
- Melhorar as estimativas de custo e agendamento de tarefas para se entregar ao usuário;
- Alcançar a certificação necessária.

Todos os objetivos e partes do projeto devem ser colocados em um manual que, mais tarde deverá conter todas as resoluções que foram feitas durante todo o projeto. Este “Manual da Qualidade” deverá, depois de pronto, ser publicado para todas as pessoas da empresa para entendimento e retirada de dúvidas.

Neste manual, poderá conter:

- A política de qualidade e objetivos da empresa;
- A estrutura da organização mostrando as responsabilidades de todos que gerenciam, fazem e verificam o trabalho que afetará a qualidade;
- Uma descrição do ciclo de vida da qualidade na empresa (será visto mais à frente);
- Uma visão geral do sistema de qualidade;
- relacionamento do sistema de qualidade com o padrão internacional adotado;
- Referência completa a procedimentos e padrões detalhados.

Planejamento e implementação do programa de qualidade

Após a conclusão do “Manual de Qualidade” e sua aprovação pelo pessoal da empresa, a comissão de organização da qualidade determinará a quantidade de serviço que deverá ser necessário para implementar o programa de qualidade.

A comissão deverá desenvolver um plano de implementação detalhando tarefas, atividades, marcos e recursos para a implantação.

Normalmente, as tarefas mais comuns a serem realizadas são as seguintes:

- implementar um programa cultural;
- Adotar um ciclo de vida;
- Desenvolver um sistema de controle de qualidade;
- Desenvolver e documentar procedimentos e padrões para todas as atividades de cada ciclo da implementação;
- Definir e implementar um programa de medidas;
- Rever e, se necessário revisar o manual de qualidade;
- Apreciação da qualidade e treinamento do sistema de qualidade;
- Um programa de auditoria da qualidade;
- Revisões gerenciais;
- Avaliação ISO 9000.

XIV.4. - IMPLEMENTAR UM PROGRAMA CULTURAL

O programa cultural deverá ser implementado em conjunto com o programa técnico.

O programa cultural consiste em conscientizar as pessoas da empresa da necessidade de se ter qualidade de alto nível. Normalmente, se utiliza do treinamento de qualidade e profissional (aprendizado de novas e melhores ferramentas de desenvolvimento) e realização de Workshop para discutir qualidade e necessidades da empresa e das pessoas que estão envolvidas.

Também faz parte do programa cultural, as iniciativas da gerência e do pessoal da empresa para poder melhorar e aumentar o incentivo das pessoas para realizarem uma tarefa melhor e mais bem feita.

As iniciativas da gerência dão ênfase a análise de defeitos e suas causas e a definição de marcos que devem ser alcançados em certo período de tempo.

Já as iniciativas dos funcionários da empresa são voltadas para sugestões de como realizar tarefas e procedimentos de forma a retornar mais qualidade e mais serviço.

XIV.5 - IMPLEMENTAR O PROGRAMA TÉCNICO

O programa técnico consiste em:

Adotar um ciclo de vida

Toda empresa de software, grande ou pequena, deve possuir um ciclo de vida para o desenvolvimento do produto. Muitos tipos de ciclo de vida têm sido implementados e testados para atender a determinadas circunstâncias.

A empresa deverá adotar um ciclo de vida e, então adequá-lo às suas necessidades.

O ciclo de vida é dividido em etapas. Em cada etapa poderá ser feito uma revisão do que já foi feito e rever o que vai ser feito na próxima etapa.

Desenvolver procedimentos e padrões

Algumas empresas trabalham com o máximo possível de procedimentos e padrões definidos rigorosamente.

É comum aos desenvolvedores não se acostumarem facilmente com estes novos padrões. No caso da empresa estar implantando pela primeira vez um sistema de qualidade, os desenvolvedores estarão sujeitos a não aceitarem o que foi dito e a empresa vai demorar muito tempo tentando conscientizá-las.

Os procedimentos e padrões devem ser identificados da maneira como são utilizados e, depois, alterá-los, se necessários e divulgá-los pela empresa toda. Estes, normalmente, sofrem a interferência de padrões internacionais como ISO e CMM e devem ser desenvolvidos e utilizados pelas pessoas que já os utilizavam.

Caso a empresa venha implantar novos procedimentos e padrões, eles devem ser feitos de maneira gradual. Se implementar tudo de uma vez, a resistência será maior e a empresa poderá perder por causa disto.

Selecionar métodos e ferramentas.

Os métodos e ferramentas a serem utilizados devem ser padronizados. Esta padronização deverá levar em consideração dicas dos desenvolvedores e diretores que estão envolvidos diretamente com o processo de desenvolvimento do software.

Estes devem ser utilizados em todos os projetos da empresa e ter uma variação pequena de um projeto para outro.

Aconselha-se que as ferramentas de desenvolvimento devem ser de um mesmo fornecedor para evitar conflitos, atrasos, defeitos inesperados e correções mal feitas para se adaptar as ferramentas diversas ao sistema.

Programa de métricas e medidas de software

É dito que, se você não pode medir, então você não pode gerenciar.

Esta frase pode não estar muito certa, mas se a empresa conseguir “medir” a funcionalidade e a qualidade de seu produto e dos concorrentes, poderá saber se o seu produto está melhor ou pior do que o outro.

Neste caso, as medidas são muito importantes. Quando adotado nas empresas, elas procuram as medidas que devem:

- Refletir os requisitos ou processos dos clientes que direcionam a escolha de pré-requisitos: qualidade não pode ser melhorada otimizando medidas que não relatam as necessidades dos clientes;
- Ser claramente definida. Todos os resultados e medidas devem estar utilizando as mesmas medidas;
- Não utilizar ferramentas caríssimas para coletar dados: às vezes, é melhor coletar dados de uma maneira mais fácil e barata e obter ótimos resultados do que utilizar de ferramental caro e obter dados que não são concisos e não obter resultados satisfatórios;

As medidas podem ser utilizadas tanto no projeto quanto nos processos para diferentes propósitos, por exemplo:

- Na fase de projeto, identificar componentes não usuais para um estudo mais detalhado (componentes que são mais fáceis de darem erros ou defeitos);
- No nível de processos: calibrar uma estimativa do modelo, comparar o uso de diferentes ferramentas e métodos em diferentes projetos, monitorar os avanços progressivos

Nas empresas, costuma-se, ainda, medir outras atividades relacionadas com o desenvolvimento de software, a exemplificar:

Desenvolvimento

- Taxas de erros e defeitos;
- Mudanças nos requisitos e especificações de desenvolvimento;
- Produtividade;
- Aderência ao tempo determinado para cada tarefa;
- Aderência aos gastos previstos.

Suporte

- Responsabilidade sobre as necessidades do usuário;
- Frequência com que o usuário necessita de mudanças.
- Geral
- Custo de baixa qualidade.

Treinamento

Uma grande necessidade da empresa que vai melhorar a qualidade do serviço que ela presta está em treinar o seu pessoal para eles terem consciência do que estão fazendo e conhecer todo o processo da empresa para saberem de suas necessidades e dos outros.

Este treinamento não consiste em treinar as pessoas apenas para receberem novas ferramentas. Elas devem ser treinadas em qualidade, processos novos e antigos, em comunicação, em novas ferramentas e em muitas outras que devem ser especificadas pela empresa de acordo com a necessidade do funcionário e da própria empresa.

XIV.6 – REVISÕES DE PROCESSOS E PRODUTO

As revisões são muito importantes. Elas definem o que foi feito certo, o que foi feito errado e o que precisa ser reparado ou não. É, talvez, a parte do projeto mais importante. É nesta etapa que os desenvolvedores vão descobrir a maioria dos erros e defeitos e que vão ser corrigido depois. Há, ainda, o teste final que é feito antes de enviar o produto para a implementação no usuário. É importante que não se tenha erros e defeitos quando o software estiver funcionando no cliente, pois ele pode reclamar muito e o projeto todo ficar comprometido.

Revisões do projeto

Todos os projetos, grandes ou pequenos deve ser revistos em sua totalidade para se descobrir o que foi aprendido e o que pode ser melhorado.

Esta revisão pode acontecer uma única vez no final do projeto ou pode acontecer no final de cada fase concluída. Este último pode ser o melhor porque se aprende mais sobre o projeto e já resolve os problemas desta etapa prevendo alguns erros para a próxima etapa.

Revisões da gerência

A gerência tem a máxima responsabilidade para a boa realização do plano de qualidade a ser implantado na empresa. Ela deve rever os processos regularmente, pelo menos uma vez ao ano. Estas revisões devem levantar todas as fontes de informações incluindo:

- Resultado das auditorias, internas ou externas;
- Relatórios das revisões do projeto;
- Relatórios dos serviços ao cliente;
- Mudanças de tecnologia no projeto;
- Avaliações dos clientes sobre a qualidade do serviço;
- Análises de medidas.

Os resultados devem ser colocados em uma listagem, anexados ao manual e entregue a todos na empresa para verem a evolução do projeto.

XV. – O FUTURO DA QUALIDADE

Pensadores sobre qualidade afirmam que esta terá várias mudanças daqui para frente.

Estas mudanças serão muito grandes porque as empresas hoje pensam em qualidade como apenas aquisição de respeito internacional através de certificações e procedimentos considerados corretos e qualitativos.

Mas será que estas empresas apenas conseguem seus certificados e não olham para o produto em si? Elas podem ter certificações mil e continuar desenvolvendo software da mesma maneira que antes e ainda ter várias reclamações de clientes sobre a qualidade do produto em questão.

O que se pensa pelo futuro da qualidade é que as empresas deverão se preocupar em criar um cargo para a gerência da qualidade (CQO[chief quality officer]). Gerentes de hoje, se tornaram gerentes de qualidade amanhã. Estes gerentes não deverão se preocupar em apenas definir regras e procedimentos que devem ser adotados por todos na empresa que trabalham, o que já é feito há muito tempo. Eles deverão concentrar seus esforços nos sistemas de marketing, vendas e suporte porque estes terão mais impacto sobre a satisfação do cliente do que os procedimentos adotados para o desenvolvimento de softwares.

XVI. - BIBLIOGRAFIA

SANDERS, Joc e CURRAN, Eugene. *Software Quality*. AddisonWesley, 1994.

Glossário de Termos da Qualidade

Ação Corretiva

Ação implementada para eliminar as causas de uma não-conformidade, de um defeito ou de outra situação indesejável existente, a fim de prevenir sua repetição. *[NBR ISO 8402]*

Ação Preventiva

Ação implementada para eliminar as causas de uma possível não-conformidade, defeito ou outra situação indesejável, a fim de prevenir sua ocorrência. *[NBR ISO 8402]*

Analisador de Código

Software que percorre um trecho de código, uma rotina ou um programa, com a finalidade de coletar métricas de complexidade ou de elaborar um grafo ou outra descrição da lógica do código percorrido.

Análise Crítica (Review)

Avaliação profunda e global de um projeto, produto, serviço, processo ou informação com relação a requisitos, objetivando a identificação de problemas e a proposição de soluções. *[Critérios de Excelência da Fundação para o Prêmio Nacional da Qualidade – FPNQ]*

Análise Crítica de Contrato

Atividades sistemáticas executadas pelo fornecedor, antes da assinatura do contrato, para garantir que os requisitos para a qualidade estão adequadamente definidos, sem ambigüidade e documentados, e que os mesmos possam ser atendidos pelo fornecedor. *[NBR ISO 8402]*

Análise Crítica de Projeto

Exame documentado completo e sistemático de um projeto para avaliar sua capacidade de atender os requisitos para a qualidade, identificar problemas, se houver, e propor o desenvolvimento de soluções. *[NBR ISO 8402]*

Análise Crítica de Requisitos

Processo ou reunião durante o qual os requisitos para um sistema, item de hardware ou item de software são apresentados aos desenvolvedores, gerentes, usuários, clientes, ou outros interessados para comentários e aprovação. Aqui também estão incluídos análise crítica de sistema e análise crítica de software. *[IEEE Std 610.12]*

Análise de Pontos por Função

Técnica de avaliação de um sistema, conhecida como FPA – Function Point Analysis, baseada na medição do valor das funções executadas pelos programas, ao invés de utilizar como base o volume ou a complexidade do código dos programas. A técnica está baseada na visão externa do usuário, sendo portanto, independente da linguagem utilizada, permitindo calcular o esforço de programação e auxiliando o usuário final a melhorar o exame e avaliação de projetos.

Análise de Requisitos

Conjunto de atividades que permite identificar as necessidades do usuário de modo a obter uma definição clara das características (requisitos) de um sistema. Essas características descrevem o sistema em termos de funcionalidades, desempenho esperado, restrições de projeto, níveis de qualidade esperados, interface com outros elementos do sistema. Processo de estudar as necessidades do usuário para se chegar a uma definição dos requisitos de sistema, hardware ou software. *[IEEE Std 610.12]*

ASQ - American Society for Quality

Entidade norte-americana que congrega profissionais interessados na engenharia da qualidade e na gestão da qualidade. Oferece diversas certificações profissionais, entre as quais a de engenheiro da qualidade (Certified Quality Engineer - CQE), engenheiro de confiabilidade (Certified Reliability Engineer - CRE), auditor da qualidade (Certified Quality Auditor - CQA), administrador da qualidade (Certified Quality Manager - CQM) e engenheiro da qualidade em software (Certified Software Quality Engineer - CSQE). No Brasil, os exames para certificação são aplicados pela Associação Brasileira de Controle da Qualidade - ABCQ.

Auditoria

Exame sistemático e independente, para determinar se as atividades da qualidade e seus resultados estão de acordo com as disposições planejadas, se estas foram implementadas com eficácia e se são adequadas à consecução dos objetivos. *[NBR ISO 8402]*

Avaliação

Exame sistemático do grau em que um produto, processo ou serviço atende aos requisitos especificados.

Avaliação de Terceira Parte ou Independente

Avaliação feita por pessoa ou organismo reconhecido como independente das partes envolvidas.

CASE - Computer Aided Software Engineering

Ferramenta de apoio ao desenvolvimento de software. Em linhas gerais, apóia a execução de atividades do desenvolvimento do software de forma automatizada. Em alguns casos, implementa um ambiente relativamente refinado no qual várias atividades de especificação ou codificação são apoiadas por recursos computacionais. Dependendo do tipo de atividade suportada podem ser classificados em Lower CASE, provendo suporte à codificação, teste, depuração e manutenção do código ou Upper CASE, suportando diversas tarefas de análise e projeto de sistemas. Eventualmente, ferramentas CASE podem ser integradas em ambientes de desenvolvimento de software. Neste caso, apoiando parte das atividades previstas em um processo de desenvolvimento de software.

Certificação

Modo pelo qual uma terceira parte dá garantia escrita de que um produto, processo ou serviço está em conformidade com os requisitos especificados. [ABNT ISO/IEC GUIA 2]

Certificação de Software

Emissão de um certificado de conformidade de um software a um certo conjunto de normas ou especificações, comprovada por testes de conformidade e por testes de campo.

CMM - Capability Maturity Model

Modelo para avaliação da maturidade dos processos de software de uma organização e para identificação das práticas chave que são requeridas para aumentar a maturidade desses processos. O CMM prevê cinco níveis de maturidade: inicial, repetível, definido, gerenciado e otimizando.

O modelo foi proposto por Watts S. Humphrey, a partir das propostas de Philip B. Crosby, e vem sendo aperfeiçoado pelo Software Engineering Institute - SEI da Carnegie Mellon University. [<http://www.sei.cmu.edu/cmm/cmm.html>]

Confiabilidade

Conjunto de atributos que evidenciam a capacidade do software de manter seu nível de desempenho sob condições estabelecidas durante um período de tempo estabelecido. [NBR 13596].

Tem como subcaracterísticas: maturidade, tolerância a falhas e recuperabilidade.

Configuração

Relação entre versões de um objeto composto, ou seja, configuração é uma instância do sistema composta da união de uma versão específica de cada objeto componente. Arranjo de um sistema computacional ou de seus componentes como definidos pelo seu número, natureza e interconexão de suas partes constituintes. [IEEE Std 610.12]

Controle de Versão

Procedimento de gestão do ciclo de vida de um produto. Consiste na identificação formal de modificações solicitadas ou efetuadas e no seu agrupamento, de modo a que fiquem incorporadas, todas elas, em uma determinada configuração do produto, num certo momento. Essa configuração recebe o nome de versão.

Custos da Qualidade

Custos relacionados com as perdas em função da qualidade insuficiente de processos, produtos ou serviços (custos da não-conformidade) ou com os investimentos em atividades que eliminem falhas ou elevem a qualidade de processos, produtos ou serviços (custos da conformidade). A identificação e a apropriação contábil desses custos permite que o administrador possa fazer uma análise do nível de qualidade de sua produção e possa tomar decisões para melhorar esse nível.

Declaração de Conformidade

Declaração, emitida pelo fornecedor ou pelo produtor de um software, assegurando que este opera em conformidade com certas normas ou especificações preestabelecidas.

Depurador Interativo

Software para apoio a testes, cuja função é permitir a visualização passo a passo da execução de uma rotina ou programa e do comportamento de seus elementos antes, durante e após a execução.

Dicionário de Dados

Listagem organizada de todos os elementos de dados que são pertinentes ao sistema, com definições precisas e rigorosas, de forma que tanto o usuário como o analista de sistemas tenham uma compreensão comum das entradas, das saídas, dos componentes dos depósitos de dados. [Pressman R.S., *Engenharia de Software*, 1995]

Driver de Teste

Software que permite a ativação de determinadas partes do software (módulos) com o intuito de testá-las. Normalmente, utilizam-se massas de teste previamente definidas e produzem resultados do teste, que podem ser verificados através da documentação de teste construída pelo analista ou engenheiro de software responsável pelos testes.

Eficiência

Conjunto de atributos que evidenciam o relacionamento entre o nível de desempenho do software e a quantidade de recursos usados, sob condições estabelecidas. [NBR 13596] Tem como subcaracterísticas: comportamento em relação ao tempo e comportamento em relação aos recursos.

Engenharia da Informação

Popularizada por James Martin, é um caminho direcionado a dados para desenvolvimento de sistemas de informação, oposto à visão de direcionamento a processo de análise estruturada. [Marciniak J.J., *Encyclopedia of Software Engineering*]

Entidade Certificadora

Órgão que realiza a certificação de conformidade, a partir da existência da garantia adequada de que um produto, processo ou serviço, devidamente identificado, está em conformidade com uma norma ou outro documento normativo especificado.

Estimação da Confiabilidade

Procedimento de estimação da confiabilidade do software a partir de dados sobre erros ou falhas conhecidas e outros dados, tais como resultados de teste e número de horas de uso.

Funcionalidade

Conjunto de atributos que evidenciam a existência de um conjunto de funções e suas propriedades especificadas. As funções são as que satisfazem as necessidades explícitas ou implícitas. [NBR 13596]

Tem como subcaracterísticas: adequação, acurácia, interoperabilidade, conformidade, segurança de acesso.

Garantia da Qualidade

Conjunto de atividades planejadas e sistemáticas, implementadas no sistema da qualidade e demonstradas como necessárias para prover confiança adequada de que uma entidade atenderá os requisitos para a qualidade. [NBR ISO 8402]

Gerador de GUI

O processo de projeto de interfaces com o usuário é iterativo. Ou seja, um modelo de projeto é criado, implementado como protótipo, examinado pelos usuários e modificado, baseado em seus comentários. O jogo de ferramentas (toolkit) de interfaces com usuário ou sistema de desenvolvimento de interfaces com o usuário (User-Interface Development Systems – UIDS), essas ferramentas oferecem módulos ou objetos que facilitam a criação de janelas, menus, interação de dispositivos, mensagens de erro, comandos e muitos outros elementos de um ambiente interativo. Os sistemas de desenvolvimento de interfaces com o usuário (User Interface Development Systems – UIDS) combinam ferramentas CASE individuais para interação humano computador com uma biblioteca de componentes de programa que possibilita que o desenvolvedor construa uma interface humano computador rapidamente. O UIDS oferece componentes de programa que gerenciam dispositivos de entrada, validam entradas do usuário, manipulam condições de erro, processam "undos" e "aborts", oferecem *feedback* visual, *prompts* e socorro, atualizam o *display*, gerenciam dados de aplicação, manipulam *scrolling* e *editing*, isolam a aplicação das funções de gerenciamento da tela e suportam características de customização para o usuário final. [Pressman R. S., *Engenharia de Software*, 1995]

Gerência de Projetos

Processo que tem início antes do trabalho técnico, prossegue à medida que o software se desenvolve do modelo conceitual para o modelo computacional e encerra somente quando o software se torna obsoleto.

Gerência de Requisitos

Estabelecimento e manutenção de um entendimento/acordo com o cliente sobre os requisitos para o projeto de software. Este acordo refere-se aos requisitos do sistema alocados para o software. O cliente pode ser interpretado como o grupo de engenharia do sistema, o grupo de marketing, outra organização interna, ou um cliente externo. O acordo compreende requisitos técnicos e não técnicos. O acordo forma a base para a estimativa, planejamento, execução e acompanhamento das atividades do projeto de software através do ciclo de vida do software. [Key Practices of the Capability Maturity Model, versão 1.1, Feb. 1993]

Gerenciamento de Configuração (Software Configuration Management – SCM)

Atividade abrangente que é aplicada em todo o processo de engenharia de software, podendo ser vista como uma atividade de garantia da qualidade de software. Uma vez que uma mudança pode ocorrer a qualquer tempo, as atividades de SCM são desenvolvidas para identificar a mudança; controlar a mudança; garantir que a mudança esteja sendo adequadamente implementada; e relatar a mudança a outras pessoas que possam ter interesse nela. O gerenciamento de configuração de software é um conjunto de atividades que foi desenvolvido para administrar as mudanças em todo o ciclo de vida do software.

Gestão da Qualidade

Todas as atividades da função gerencial que determinam a política da qualidade, os objetivos e as responsabilidades, e os implementam por meios tais como: o planejamento da qualidade, o controle da qualidade, a garantia da qualidade e a melhoria da qualidade dentro do sistema da qualidade. [NBR ISO 8402]

Gestão de Configuração

Atividade técnica e gerencial aplicada ao longo de todo o ciclo de vida do software que tem por objetivo garantir a integridade do produto sendo desenvolvido, mantendo um histórico de todos os itens sob controle. Basicamente, é composta de quatro tarefas: identificação da configuração, controle da configuração, balanço da configuração e auditoria da configuração.

Guia ABNT ISO/IEC GUIA 2

Termos gerais e suas definições relativas à normalização e atividades correlatas.

Guia ABNT ISO/IEC GUIA 25

Requisitos gerais para a capacitação de laboratórios de calibração e de ensaios.

Inspeção Formal

Técnica de revisão sistemática do software ou de alguns de seus componentes, executada, sistematicamente, ao final de cada fase do projeto, com o objetivo único de encontrar erros. A inspeção formal é executada por uma equipe na qual cada membro tem papel preestabelecido. O projetista participa mas não coordena a reunião. Todo o material gerado é lido, os erros anotados e uma estatística dos erros encontrados é mantida, para fins de posterior estudo da eficácia do procedimento.

Interface Humano Computador

Mecanismo por meio do qual se estabelece um diálogo entre o programa e o ser humano. Se os fatores humanos tiverem sido levados em conta, o diálogo será harmonioso e um ritmo será estabelecido entre o usuário e o programa. Se os fatores humanos tiverem sido ignorados, o sistema quase sempre será visto como "não-amigável". [Pressman R.S., Engenharia de Software, 1995]

ISO

International Organization for Standardization (Organização Internacional de Normalização).

ISO 8402

Quality management quality assurance – vocabulary.

ISO 9000-3

Quality management and quality assurance standards – Part 3: Guidelines for the application of ISO 9001 to the development, supply and maintenance of software. Norma internacional da série ISO 9000. Estabelece diretrizes para aplicação da norma ISO 9001 às atividades de desenvolvimento, fornecimento e manutenção de software.

ISO 9001

Quality systems – Model for quality assurance in design, development, production, installation and servicing.

Norma internacional da série ISO 9000. Modelo para garantia da qualidade em projetos, desenvolvimento, produção, instalação e serviços associados. É mais completa que as normas ISO 9002 e ISO 9003, da mesma série.

ISO 9002

Quality systems – Model for quality assurance in production, installation and servicing. Norma internacional da série ISO 9000. Modelo para garantia da qualidade na produção, instalação e serviços associados.

ISO 9003

Quality systems – Model for quality assurance in final inspection and testing. Norma internacional da série ISO 9000. Modelo para garantia da qualidade em inspeção e ensaios finais.

ISO/IEC 9126

Information technology - Software quality characteristics and metrics. Norma que define as características da qualidade de software, para fins de sua avaliação. Será complementada com outras normas que definirão guias para avaliação do software, hoje na forma de drafts. A norma brasileira correspondente é a NBR 13596.

ISO 9241

Ergonomic requirements for office work with visual display terminals (VDTs). Norma que define requisitos ergonômicos para o trabalho de escritório com computadores (VDT – Visual Display Terminals), objetivando promover a saúde e a segurança de usuários de computadores e garantir que eles possam operar esses equipamentos com eficiência e conforto.

ISO/IEC 12119

Information technology - Software packages - Quality requirements and testing. Norma que estabelece os requisitos da qualidade e testes em pacotes de software. Seu escopo refere-se a pacotes de software, na forma oferecida no mercado, e não aos processos de desenvolvimento e fornecimento de software. A norma brasileira correspondente é a NBR ISO/IEC 12119.

ISO/IEC 12207

Information technology – Software life cycle process.

ISO/IEC 14598

Information technology – Software product evaluation. Família de normas que tratam do processo de avaliação de um produto de software e complementam o modelo apresentado na norma ISO/IEC 9126, hoje na forma de drafts.

ISO/IEC 15504

Information technology – Software process assessment.

Futura norma internacional para avaliação de processos de software, em desenvolvimento pelo projeto SPICE (Software Process Improvement and Capability dEtermination), o que a torna conhecida também como Modelo SPICE. Atualmente está publicada como um relatório técnico (ISO/IEC TR 15504) da ISO/IEC com previsão de ser publicada como norma em 2002. Define um modelo de referência com processos e níveis de capacidade, orientações sobre como utilizá-lo para melhoria contínua ou determinação da capacidade, e um modelo exemplo compatível.

JAD - Joint Application Design

Conjunto de sessões intensivas e mediadas entre usuários e analistas de um sistema, com o objetivo de explicitar os seus requisitos. A técnica, desenvolvida nos anos setenta pela IBM do Canadá, voltou a ficar em voga com o uso do RAD - Rapid Application Development, metodologia que combina o JAD (para definir rapidamente a especificação do sistema) com o uso de ferramentas CASE e de metodologias de prototipação, para chegar a um produto final em menor tempo.

Lead Assessor

Certificação que qualifica um auditor a atuar na avaliação de empresas segundo as normas ISO 9000.

A obtenção desse título depende da participação em cursos e da realização de um número de horas de auditoria, acompanhando auditores já certificados.

Manutenibilidade

Conjunto de atributos que evidenciam o esforço necessário para fazer modificações especificadas no software. [NBR 13596]. Tem como subcaracterísticas: analisabilidade, modificabilidade, estabilidade e testabilidade.

Medição

Ação de aplicar uma métrica de qualidade de software a um produto de software específico. [NBR 13596]

Medição de Linhas de Código (LOC)

É a métrica de código mais básica. A definição mais comum de LOC estabelece que qualquer linha do programa que não seja comentário ou linha em branco, independente do número de sentenças (lógicas ou operações) estão presentes naquela linha. [Marciniak J.J., *Encyclopedia of Software Engineering*]

Melhoria de Processos de Software (Software Process Improvement)

Uma abordagem (SPI) para melhoria das organizações que desenvolvem e mantêm software. É baseada na melhoria da capacidade de processos fundamentais para organizações de software. Utiliza como referência um modelo de processo, como por exemplo, o CMM e a ISO/IEC 15504-SPIICE.

Métricas de Complexidade

Grandezas coletadas através do exame da especificação do código de um sistema, programa com rotina e que refletem o seu tamanho e a sua complexidade lógica. Diversos modelos existem para relacionar métricas de complexidade com tempo ou esforço de desenvolvimento e com o número de erros embutidos no produto.

Métrica de Qualidade de Software

Método e uma escala quantitativa que podem ser usados para determinar o valor que uma particularidade (feature) recebe em um produto de software específico. [NBR 13596]

NBR ISO 8402

Gestão da qualidade e garantia da qualidade – Terminologia, Brasil.

NBR ISO 9000-3

Normas de gestão da qualidade e garantia da qualidade - Parte 3: Diretrizes para a aplicação da NBR 19001 (ISO 9001) ao desenvolvimento, fornecimento e manutenção de software, Brasil.

NBR ISO 9001

Sistemas da qualidade - Modelo para garantia da qualidade em projetos, desenvolvimento, produção, instalação e serviços associados, Brasil.

NBR ISO 9002

Sistemas da qualidade – Modelo para garantia da qualidade em produção e instalação e serviços associados, Brasil.

NBR ISO 9003

Sistemas da qualidade – Modelo para garantia da qualidade em inspeção e ensaios finais, Brasil.

NBR ISO/IEC 12119

Tecnologia de informação – Pacotes de software – Testes e requisitos de qualidade, Brasil. Norma que estabelece os requisitos de qualidade para pacotes de software e instruções de como testar um pacote de software com relação aos requisitos estabelecidos.

NBR ISO/IEC 12207

Tecnologia de informação – Processos de ciclo de vida de software, Brasil. Norma que estabelece uma estrutura comum para os processos de ciclo de vida de software, com terminologia bem definida, que pode ser referenciada pela indústria de software.

NBR 13596

Tecnologia de informação – Avaliação de produto de software – Características de qualidade e diretrizes para o seu uso, Brasil. Versão brasileira da norma ISO/IEC 9126.

Otimizador

Software, usualmente embutido no compilador que otimiza o código gerado a partir do exame do programa a ser compilado, eliminando redundâncias, código inacessível, etc.

Peer-review

Técnica de revisão de um produto, na qual um colega (*peer*) do projetista ou do programador revisa o produto desenvolvido, buscando encontrar erros ou oferecer sugestões de melhoria.

Política da Qualidade

Intenções e diretrizes globais de uma organização relativas à qualidade, formalmente expressas pela alta administração. [NBR ISO 8402]

Portabilidade

Conjunto de atributos que evidenciam a capacidade do software de ser transferido de um ambiente para outro. [NBR 13596]. Tem como subcaracterísticas: adaptabilidade, capacidade para ser instalado, conformidade e capacidade para substituir.

Processo

Conjunto de recursos e atividades inter-relacionadas que transformam insumos (entradas) em produtos (saídas). [NBR ISO 8402]. Agrupamento em seqüência de todas as tarefas destinadas a obter um determinado resultado. É a combinação de equipamentos, instalações, mão-de-obra, métodos, técnicas, ferramentas, procedimentos e outros fatores, com a finalidade de elaborar um produto ou alcançar um resultado preestabelecido.

Processo de Software

Conjunto de atividades, métodos, práticas e transformações que as pessoas empregam para desenvolver e manter software e os produtos associados (por exemplo, planos de projeto, documentos de projeto/design, código, casos de teste, manual do usuário).

Programação Orientada a Objetos

Técnica de programação que enfatiza a descrição dos conceitos envolvidos com o domínio do problema (objetos) através de seus dados e operações, encapsulados e representados através de classes. Cada objeto é criado como pertencendo a uma classe. A utilização de um objeto, e sua eventual mudança de estado, se dá a partir de mensagens enviadas a ele, representadas pelas operações encapsuladas na classe. Novas classes podem ser criadas a partir de classes existentes e organizadas através de um processo de classificação e hierarquização, explorando o conceito de herança. Os programas são construídos como organizadores da ativação de mensagens para os objetos, desta forma fazendo com que as funcionalidades de um sistema sejam obtidas através da cooperação dos objetos.

Projeto da Interface com o Usuário

O processo global para projetar uma interface com o usuário inicia-se com a criação de diferentes modelos de função do sistema. Quatro diferentes modelos entram em cena quando uma HCI vai ser projetada. O engenheiro de software cria um modelo de projeto; um engenheiro humano estabelece um modelo de usuário, o usuário final desenvolve uma imagem mental que muitas vezes é chamada modelo do usuário ou de percepção do sistema e os implementadores do sistema criam uma imagem do sistema. [Pressman R. S., Engenharia de Software, 1995]

Projeto de Software

Envolve tipicamente análise, especificação, projeto (design), desenvolvimento, teste e/ou manutenção dos componentes de software e da documentação associada. [Mark Paulk, 1995]

Prototipação

Método de desenvolvimento que prevê a execução de vários ciclos de análise, especificação e codificação de um sistema. No primeiro ciclo, gera-se um produto simplificado em pouco tempo, de modo que o usuário possa examiná-lo e refinar as suas demandas. Nos ciclos seguintes, o produto é aperfeiçoado e novas funções são sucessivamente implementadas, até se chegar ao produto final.

Prova de Correção

Exame de uma especificação descrita segundo regras formais preestabelecidas, de modo a provar matematicamente a sua correção, através do uso de axiomas, teoremas e procedimentos algébricos.

QFD - Quality Function Deployment

Técnica de planejamento e de especificação de requisitos que consiste em reuniões com técnicos e clientes, nas quais são elaboradas matrizes em que se cruzam informações sobre "o que" é desejado (requisitos) e "como" implementar. É composta por quatro etapas - projeto, componentes, processo e produção, sendo gerada a cada etapa uma matriz, a partir da matriz anterior. As matrizes explicitam relações, conflitos, níveis de dificuldade, estágio tecnológico. Por seu formato peculiar, a matriz do QFD é conhecida como "casa da qualidade".

Qualidade (Quality)

Totalidade de características de uma entidade que lhe confere a capacidade de satisfazer as necessidades explícitas e implícitas. [NBR ISO 8402]. Entidade pode ser uma atividade ou um processo, um produto, uma organização ou uma combinação desses.

Reengenharia de Software

Técnica de reestruturação ou modificação de um código existente, ou de desenvolvimento de um novo código, preservando-se inalterada a especificação ou o projeto do software.

Requisitos (Requirements)

Necessidades básicas do cliente, geralmente explicitadas como condição de negócio no contrato com o fornecedor. São características, tais como especificações técnicas, prazo de entrega, garantia, que o cliente "requer" do produto.

Uma condição ou capacidade necessitada por um usuário, para resolver um problema ou alcançar um objetivo. [IEEE 83]

Requisitos para a Qualidade

Expressão das necessidades ou sua tradução num conjunto de requisitos, explicitados em termos quantitativos ou qualitativos, objetivando definir as características de uma entidade a fim de permitir sua realização e seu exame. [NBR ISO 8402]

Reuso de Código

Técnica de utilização de um programa, de uma rotina ou de uma biblioteca de rotinas em mais de uma aplicação, ou em diferentes implementações de uma aplicação. Para tal, o código deve ser escrito prevendo-se a sua reusabilidade e enfatizando aspectos de sua abrangência funcional e de independência de ambiente.

SPICE - Software Process Improvement and Capability dEtermination

Projeto estabelecido em junho de 1993 pela ISO/IEC JTC1/SC7 (Subcomitê de Engenharia de Software) com três objetivos principais: auxiliar o desenvolvimento de uma Norma Internacional para avaliação de processos de software; coordenar e analisar utilizações desta futura Norma para subsidiar revisões antes de sua publicação como norma (esta atividade é chamada de SPICE Trials); e disseminar a futura norma no mercado. A norma desenvolvida é a ISO/IEC 15504 que está publicada atualmente com relatório técnico da ISO/IEC. Já foram realizadas duas fases do SPICE Trials, com dados sobre cerca de 100 utilizações da futura norma e a fase 3 está em andamento.

Teste de Aceitação

Teste formal conduzido para determinar se um sistema satisfaz ou não seus critérios de aceitação e para permitir ao cliente determinar se aceita ou não o sistema. [IEEE 83]. Validação de um software pelo comprador, pelo usuário ou por terceira parte, com o uso de dados ou cenários especificados ou reais. Pode incluir testes funcionais, de configuração, de recuperação de falhas, de segurança e de desempenho.

Teste de Campo

Verificação de um software ou um trecho de software durante a sua utilização real (em campo), de modo a detectar, confirmar ou examinar falha ou erro relatado.

Teste de Integração

Técnica sistemática para a construção da estrutura de programa, realizando-se ao mesmo tempo, testes para descobrir erros associados a interfaces. O objetivo é, a partir dos módulos testados no nível de unidade, construir a estrutura de programa que foi determinada pelo projeto. O teste de integração cuida das questões associadas aos duplos problemas da verificação e construção de programas.

Teste de Sistema

Processo de testar um sistema integrado de hardware e software para verificar se o sistema satisfaz seus requisitos especificados. [IEEE 83]

Teste de Unidade

Verificação de um componente de um software, através de teste funcional, desenvolvido a partir da especificação das funções previstas para o componente, ou de teste estrutural, desenvolvido a partir da descrição da estrutura do componente.

Teste de Usabilidade

Verificação e homologação individual do uso por um conjunto de usuários.

Teste Funcional

Teste conduzido para demonstrar a operacionalidade das funções que foram especificadas. O teste funcional (ou caixa-preta) é realizado olhando-se o software apenas através de suas interfaces, portanto testando sua funcionalidade.

Tick IT

Esquema para a avaliação e registro de sistemas da qualidade de atividades de desenvolvimento, fornecimento e manutenção de software. Baseia-se no uso da norma ISO 9000-3 e de um guia, Tick IT Guide, para que os auditores apliquem adequadamente as normas ISO 9000. No esquema Tick IT, os auditores são submetidos a critérios adicionais aos do Lead Assessor para serem certificados. Essa iniciativa limita-se, por enquanto, à Grã-Bretanha e ainda não tem recebido aceitação unânime.

Usabilidade

Conjunto de atributos que evidenciam o esforço necessário para se poder utilizar o software, bem como o julgamento individual desse uso, por um conjunto explícito ou implícito de usuários. [NBR 13596].

Tem como subcaracterísticas: inteligibilidade, apreensibilidade e operacionalidade.

Validação

Confirmação, por exame e fornecimento de evidência objetiva, de que os requisitos específicos para um uso pretendido são atendidos. [NBR ISO 8402]. Informações cuja veracidade pode ser comprovada com base em fatos obtidos através da observação, medição, ensaios ou outros meios constituem evidência objetiva.

Verificação

Confirmação, por exame e fornecimento de evidência objetiva, do atendimento aos requisitos especificados. [NBR ISO 8402]. Processo de avaliação de um sistema (ou componente) com o objetivo de determinar se o produto de uma dada fase do desenvolvimento satisfaz às condições impostas no início dessa fase.

Walkthrough Estruturado

Técnica de análise estática na qual um projetista ou programador apresentam aos membros do grupo de desenvolvimento e outros profissionais interessados uma parte de documentação ou código, e os participantes fazem perguntas e comentários sobre possíveis erros, violação de padrões de desenvolvimento ou sobre outros problemas.

Fonte: <http://www.mct.gov.br/temas/info/Dsi/qualid99/99anexo2.htm>