

APLICAÇÃO DE PRODUTO DE SOFTWARE UTILIZANDO A ISO/IEC 9126¹

APPLICATION OF THE SOFTWARE PRODUCT BY USING THE 9126 ISO/IEC

Silvanna Lodi²
Walkiria Cordenonzi³

RESUMO

O termo qualidade está inserido em diversos ramos empresariais, percorrendo, principalmente, a área de informática. Os compradores de *software* buscam produtos com qualidade. Por sua vez, os desenvolvedores de produtos de *software* devem implementar seus sistemas, leva em conta a satisfação do cliente. É necessário que se saiba o que é qualidade, antes de testar um produto. Para se saber se ele apresenta qualidade, é importante ter conhecimentos sobre engenharia de *software*, qualidade de processo de *software* e qualidade de produto de *software*. Para alcançar-se a qualidade total de um *software*, é preciso a observação de uma série de fatores, alguns de forma mais complexa de serem analisados. A qualidade de um produto pode ser medida por meio de um processo de avaliação baseado na norma internacional ISO/IEC 9126. A avaliação da qualidade de um produto é importante porque é tomando-a por base que o usuário pode ter certeza de que o produto adquirido apresenta qualidade. Para avaliar o produto, foi elaborado um método de avaliação denominado *MSQuality*, que apresenta um método matemático probabilístico que mostra, de forma correta, o nível de qualidade de um produto de *software*. Também foi implementada uma ferramenta fundamentada neste método estatístico, denominada *FAAPROS* - Ferramenta de Auxílio à Avaliação de Produto de *software*, que informa o valor quantitativo da qualidade do produto de *software*. O referido artigo apresenta estudos sobre as aplicações das normas internacionais ISO/IEC 9126 - Parte 1, utilizada para avaliar produtos de *software* e a ISO/IEC 15498 - Parte 5, que possui um guia de requisitos para realizar a avaliação.

Palavras-chave: *software*, engenharia de *software*, produto de *software*, qualidade.

ABSTRACT

The term 'quality' is inserted in several kinds of business, going mainly through the computing area. The software purchasers search for products having quality. On the other hand, the developers of software products

¹Trabalho Final de Graduação

²Aluna do Curso de Sistemas de Informação - UNIFRA. E-mail: silvanalodi@terra.com.br.

³Orientadora. E-mail: walkiria@unifra.br.

should implement their systems by observing their client's satisfaction. It should be known what quality means before testing a product. In order to realize if the product has quality, it is important to know about software engineering, quality of software processing and quality of software product. To achieve the total quality of software, it is necessary the observation of a series of factors, some in a more complex way of being analyzed. The quality of a product can be measured by means of an evaluation process based on the 9126 ISO/IEC international norm. The evaluation of a product quality is important because, based on this, the user can be sure of the quality shown by the product which is bought. To evaluate the product, a method named MS Quality was designed, which presents a probabilistic mathematical method showing the level of quality of a software product. It was also implemented a tool based on this statistical method, named FAAPROS - Aid Tool for the Evaluation of a Software Product, which informs the quantitative value of the quality of the software product. The present article deals with studies about the application of the international norms 9126 ISO/IEC - Part 1, used for evaluating software products and the 15498 ISO/IEC - Part 5, which has a procedure check list in order to perform the evaluation.

Key words: software, software engineering, software product, quality.

INTRODUÇÃO

Para se verificar a qualidade de um produto de *software* é necessário avaliá-lo com base em algumas normas internacionais. Entre elas, a ISO/IEC 9126, que descrevem um modelo de qualidade e alguns exemplos de métricas que podem ser utilizadas para fazerem avaliação do produto de *software* e a ISO/IEC 14598, dividida em seis partes, nela, a quinta parte é utilizada para a criação de um Método de Avaliação, que é o objetivo principal deste trabalho.

O trabalho é composto por várias partes que permitem uma melhor compreensão sobre qualidade. Tem, como objetivo principal, auxiliar na avaliação quantitativa do *software*, medindo a sua qualidade. Para se obterem conhecimentos sobre qualidade, é necessário que se tenham noções sobre engenharia de *software*, qualidade de processo de *software* e qualidade de produto de *software*, que serão apresentadas na Parte I.

A parte II traz informação sobre a norma internacional ISO/IEC 9126, mostrando, detalhadamente, suas características e subcaracterísticas.

Já a parte III apresenta um método de avaliação denominado *MSQuality*. Sua finalidade principal é verificar todas as características de qualidade

existentes num produto de *software*, observando suas vantagens e desvantagens.

A parte IV aborda o estudo de caso, que trata da demonstração do nível de pontuação do *software* bem como do parecer de três testadores do produto.

A quinta e última parte traz informações sobre a implementação da ferramenta denominada *FAAPROS* - Ferramenta de Auxílio à Avaliação de Produto de *software*. A mesma foi desenvolvida por componentes do Grupo de Pesquisa em Engenharia de *software* - GPES. Seu objetivo principal é automatizar a execução do método estatístico, resultando na avaliação quantitativa dos *software* que forem solicitados na sua avaliação. Após, são apresentadas as conclusões.

PARTE I: QUALIDADE

Com a necessidade de se sobressair no mercado e garantir um ambiente competitivo nos negócios, muitas empresas vivem num constante processo de transformação. Diante desse contexto, o termo qualidade está sendo difundido em todos os ramos empresariais, pois, na busca do aperfeiçoamento, ela tornou-se um tema em crescente discussão.

A conceituação de qualidade deve ser de forma contínua, dinâmica e ampliada. Assim, PACHECO JR (1995) afirmou que: “Qualidade é a capacidade do contínuo planejamento e desenvolvimento de ações para sempre satisfazer as necessidades dos clientes”.

Existem outras definições do conceito de qualidade, como por exemplo:

Qualidade é antecipar e satisfazer os desejos dos clientes (WEINBERG, 1993).

Qualidade é uma rápida percepção e satisfação das necessidades do mercado, adequação ao uso dos produtos e homogeneidade dos recursos do processo (Demmin apud YSHIKAWA, 1997).

Os conceitos de qualidade não bastam para que uma empresa desenvolva produtos com tecnologia e qualidade. Para isso, é preciso que se tenham maiores informações e definições sobre engenharia de *software*, o que será apresentado a seguir (DEMING, 1990)

ENGENHARIA DE SOFTWARE

A Engenharia de *software*, na prática profissional, cresce dia após dia, por meio do somatório de experiências. O tema engenharia de *software* ainda diverge entre os estudiosos e a comunidade interessada na construção e manutenção de um produto de *software*.

A Engenharia de *software*, para PRESSMAN (1995), apresenta o seguinte conceito:

O estabelecimento e uso de sólidos princípios de engenharia, para que possa, obter economicamente um *software*, que seja confiável e que, funcione eficientemente em máquinas reais.

É necessário ter um melhor conhecimento sobre a qualidade de processo de *software* e qualidade de produto de *software*, pois os mesmos não podem ser separados quando se analisa a qualidade. Esses conceitos serão vistos, a seguir.

QUALIDADE DE PROCESSO DE SOFTWARE

A qualidade de *software* é determinada pela qualidade dos processos utilizados para o desenvolvimento. Na qualidade do processo, inicia-se com a coleta dos requisitos do usuário e termina na entrega do produto final, passando por um ciclo de vida que oferece uma análise crítica do contrato e estende-se à sua instalação e manutenção.

Existem alguns modelos de definição, avaliação e melhoria de processos de *software*, que são: Modelo CMM - o seu objetivo inicial foi estabelecer um padrão de qualidade para o *software* desenvolvido para as forças armadas americanas. O SPICE é o nome do projeto de elaboração da futura norma ISO/IEC 15504, foi lançado em 1993, com o objetivo de gerar normas para avaliação de processos, visando à melhoria contínua do processo e a determinação da sua capacitação. A ISO 9000-3 apresenta diretrizes que cobrem questões como entendimento comum entre as partes de requisitos funcionais e o uso de metodologias consistentes para o desenvolvimento de *software*. A norma ISO/IEC 12207 foi aprovada em agosto de 1995 e estabelece os processos de atividades e tarefas a serem aplicados durante a aquisição, fornecimento, desenvolvimento, operação e manutenção de *software* (ROCHA, 2001).

QUALIDADE DE PRODUTO DE SOFTWARE

A qualidade de um produto de *software* é resultante das atividades realizadas no seu processo de desenvolvimento. Avaliar a qualidade de um produto de *software* é verificar, por meio de técnicas e atividades operacionais, o quanto os requisitos são atendidos.

A satisfação de cada usuário depende, exclusivamente, da forma pela qual ele se relaciona com o *software* como produto, levando em consideração, a facilidade de uso e a quantidade de informações disponíveis do *software*. A escolha de um produto de *software* depende do conhecimento técnico do usuário, tendo em vista o atendimento de suas necessidades mínimas (STRAFACCI, 1996).

Existem algumas normas que apresentam modelos utilizados para definir, avaliar e propor melhorias nos produtos de *software*, como as normas internacionais ISO/IEC 12119, publicadas em 1994, que trata da avaliação de pacotes de *software*, também conhecidos como "*software* de prateleira", a ISO/IEC 14598 oferece uma avaliação de produtos de *software* numa visão geral dos processos e fornece guia de requisitos para avaliação. Já ISO/IEC 9126 foi publicada em 1991. É uma das normas mais antigas da área de qualidade de *software* e já possui sua tradução para o Brasil, publicada em agosto de 1996.

Na próxima parte será apresentada a norma ISO/IEC 9126 - parte 1 que é objeto de estudo deste trabalho, pois o objetivo principal do mesmo é obter maiores informações sobre a referida norma, e com a ISO/IEC 14598, propor um novo método de avaliação para os produtos de *software*.

PARTE II: NORMA ISO/IEC 9126

A norma internacional ISO/IEC 9126 é composta por um conjunto de características que devem ser verificadas em um *software* para que ele seja considerado um "*software* de qualidade".

A ISO/IEC 9126 (ISO/IEC 9126, 1991 apud ROCHA, 2001) está dividida em quatro partes. A parte 1 da norma aborda Modelo de Qualidade é documento, basicamente de definições. Ele é composto por seis características de qualidade, em que cada uma é dividida em várias subcaracterísticas que serão o objeto de estudo deste trabalho. As características são: Funcionalidade, Confiabilidade, Usabilidade, Eficiência, Manutenibilidade e Portabilidade.

Na parte a seguir, será apresentado o método de avaliação utilizado para testar a qualidade do produto de *software*.

PARTE III: MÉTODO DE AVALIAÇÃO

Anteriormente, abordaram-se conceitos sobre qualidade, qualidade de processo, qualidade de produto e qualidade de *software*. Esses conceitos foram baseados na norma ISO/IEC 9126.

Esta parte tem como objetivo, apresentar um novo método de avaliação baseado no método proposto por FERNANDES (2001) e nas normas ISO/IEC 9126-1 e ISO/IEC 14598-5. Esse método se fundamenta em modelo de qualidade e baseia-se no processo de avaliação do *software*, respectivamente. O método de avaliação denomina-se *MSQuality* e é composto por uma série de itens como: questionário técnico utilizado pelo testador, apresentação das perguntas dos questionários, modelo estatístico para tabulação dos resultados, entre outras. Todas são especificadas a seguir.

Questionário Técnico: É composto por 52 perguntas, apresentadas num formulário que será utilizado pelo testador do produto de *software*. O questionário foi baseado em FERNANDES (2001) e foi totalmente reformulado neste trabalho. Seu objetivo é auxiliar o testador na avaliação quantitativa de todos os requisitos do *software*, ou seja, verificar a medida de qualidade existente no *software*, tendo como base a norma ISO/IEC 9126.

Entrevista do Usuário: Durante a entrevista, será aplicado ao usuário do *software* um questionário composto por 14 perguntas. Este item do método de avaliação, não foi utilizado anteriormente, tendo como contribuição neste trabalho, a verificação do grau qualitativo do produto, ou seja, dar maior atenção à opinião de todos aqueles que utilizam o produto de *software*, na realização e execução do seu trabalho diário, observando suas vantagens e desvantagens.

Aplicação dos Questionários: Todas as perguntas existentes no formulário foram elaboradas com base nas normas ISO/IEC 9126, parte 1 e da ISO/IEC 14598, parte 5. O questionário é mostrado em forma de questões objetivas ou a serem respondidas em forma de gráficos.

A figura 1 mostra a tabela de respostas em forma de gráfico e possui cinco divisões que indicam o nível de pontuação existente em cada pergunta

do questionário. O primeiro critério inicia com o valor 0 (zero), representando “Não Se Aplica” e o último termina com o valor 4 (quatro), informando que “Possui Total Presença”, ou seja, representa o valor máximo de pontuação. Cada um dos critérios existente será apresentado com mais detalhes a seguir:

- | | |
|-------------------------------|---|
| (0) Não Se Aplica ... | sem grau de relevância, representa 0% de pontuação. |
| (1) Possui Valor Mínimo ... | baixo grau de relevância em relação ao <i>software</i> analisado, representa 25% do nível de pontuação. |
| (2) Possui Valor Moderado ... | moderno grau de relevância em relação ao <i>software</i> analisado, apresentando 50% do nível de pontuação. |
| (3) Possui Valor Alto ... | alto grau de relevância em relação ao <i>software</i> analisado, resultando cerca de 75% do nível de pontuação. |
| (4) Possui Total Presença ... | o grau de relevância é muito importante em relação ao <i>software</i> analisado, significando 100% do nível de pontuação. |

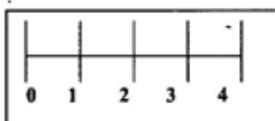


Figura 1 - Tabela de respostas de múltiplas escolhas.

O questionário apresenta algumas questões objetivas do tipo “Sim”, “Não” ou “Não Se Aplica”. A escolha da opção “Sim” corresponde à máxima satisfação e apresenta seu valor integral correspondente a 4; a opção “Não” recebe o valor 0 (zero), pois não satisfaz às necessidades do produto de *software*. Já a opção “Não Se Aplica” indica que o critério não será aplicado na referida pergunta.

As questões de múltipla ou simples escolha apresentam um peso que corresponde ao Grau de Importância (GI). O GI identifica o valor de cada pergunta, dependendo do produto de *software* a ser avaliado, o seu Grau de Importância difere de uma questão para outra, por memio da concepção do testador. O Grau de Importância - GI de cada questão varia em uma escala de 0 a 1,0, onde 0 corresponde a ausência de importância da questão e 1,0 corresponde à máxima importância da questão. A escala de valores do GI está melhor representada a seguir: (0): Sem importância para questão; (0,1

a 0,4): Baixa importância para a questão; (0,5): Média importância para a questão; (0,6 a 0,9): Boa importância para a questão e (1,0): Máxima importância para questão.

Entrevista realizada ao usuário: A entrevista realizada aos usuários do *software* apresenta um questionário com 14 questões, que são apresentadas em LODI(2002).

APRESENTAÇÃO DAS PERGUNTAS DO QUESTIONÁRIO TÉCNICO

Todas as perguntas do questionário foram elaboradas com base nas características da norma internacional ISO/IEC 9126. A seguir, será abordada, cada uma destas características, com suas subcaracterísticas. Inicialmente, temos como característica a Funcionalidade, a qual se entende por "Satisfaz as necessidades explícitas e implícitas para a finalidade a que se destina o *software*". As questões que não apresentarem valor resposta deverão ser respondidas conforme a figura 1, respondendo também o grau de importância. Como subcaracterísticas tem-se:

•Adequação:

- 1.O *software* apresenta um conjunto de funções para realizar o trabalho para o qual foi proposto?
- 2.O *software* apresenta login para cada usuário?
() Sim () Não GI=
- 3.O *software* realiza todas as tarefas para o qual foi criado, corretamente?

•Acurácia:

- 1.O *software* impede de abortar?
() Sim () Não GI=
- 2.Os resultados do *software* são corretos?
- 3.A saída dos dados está correta?
- 4.A saída dos dados é clara?

•Interoperabilidade:

- 1.O *software* ao trabalhar com outros programas, apresenta um desempenho de ... %?
- 2.Ao se comunicar, o *software* impede de ocorrer falhas?
() Sim () Não GI=

•Conformidade:

- 1.O *software* está de acordo com as leis em vigor?
() Sim () Não GI=

•Segurança de Acesso:

- 1.O *software* apresenta a opção de fazer backup?
() Sim () Não GI=
- 2.O backup é realizado corretamente?
() Sim () Não GI=
- 3.As *backups* do *software*, são guardadas em lugares seguros?
() Sim () Não GI=
- 4.Os *backups* do *software* estão guardados em lugares
() Sim () Não () Não se Aplica GI=
- 5.O *software* possui a capacidade de evitar o acesso de pessoas não autorizadas?
() Sim () Não GI=
- 6.O *software* possui um registro de login? () Sim () Não GI=

Por Confiabilidade entende-se: “Evidenciam a capacidade do *software* de manter seu nível de desempenho e se mantém ao longo do tempo em condições estabelecidas”. As questões que não apresentarem valor resposta deverão ser respondidas conforme a figura 1, respondendo também o grau de importância. Como subcaracterísticas tem-se:

•Maturidade:

- 1.Qual é o grau de não ocorrer falhas?
- 2.Caso ocorram falhas, os dados e informações são perdidos?
() Sim () Não GI=

•Tolerância a Falhas:

- 1.Caso, o *software* seja multiusuário, o mesmo, mantém um nível de desempenho em caso de falhas?
() Sim () Não () Não se Aplica GI=
- 2.Caso ocorra uma falha, o *software* continua trabalhando?
() Sim () Não () Não se Aplica GI=
- 3.Mesmo ocorrendo falhas no *software*, seu desempenho não é prejudicado?
() Sim () Não () Não se Aplica GI=

1. Caso ocorra uma falha, o *software* apresente capacidade de restabelecer?
 Sim Não Não se Aplica GI=
2. Caso ocorra uma falha, o *software* é capaz de recuperar todos os dados e informações?
 Sim Não Não se Aplica GI=

Pela característica Usabilidade, entende-se: “Conjunto de atributos que evidenciam o esforço necessário para se poder utilizar o *software* bem como o julgamento individual deste uso, por um conjunto explícito ou implícito de usuários”. As questões que não apresentarem valor resposta deverão ser respondidas conforme a figura 1, respondendo também o grau de importância. Como subcaracterísticas tem-se:

•Inteligibilidade:

1. O *software* apresenta interface de fácil entendimento?
 Sim Não GI=
2. O *software* possui facilidade no entendimento dos seus conceitos?

•Apreensibilidade:

1. É fácil treinar um usuário para utilizar o *software*?
 Sim Não GI=
2. O *software* apresenta manual do usuário?
 Sim Não GI=
3. Qual é o grau de compreensão do manual do usuário?
4. As informações são encontradas com facilidade no manual do usuário?
 Sim Não GI=
5. O *software* apresenta informações sobre seu fabricante?
 Sim Não GI=
6. O *software* apresenta informações na tela, que ajuda na sua navegação?
 Sim Não GI=

•Operacionalidade:

1. O *software* mostra informações (menus) no vídeo, que facilitam sua operação?
 Sim Não GI=
2. O *software* é de fácil operação?

- 3.O *software* apresenta help?
() Sim () Não GI=
- 4.O grau de esclarecimento do help é satisfatório?
- 5.Qual é o grau de facilidade para se trabalhar com o *software*?

Pela característica Eficiência, entende-se: “Conjunto de atributos que evidenciam o relacionamento entre o nível de desempenho do *software* e a quantidade de recursos usados, sob condições estabelecidas”. As questões que não apresentarem valor resposta deverão ser respondidas conforme a figura 1, respondendo também o grau de importância. Como subcaracterísticas tem-se:

●Comportamento em relação ao tempo:

- 1.Quando o *software* realiza algum tipo de operação, o tempo de resposta (velocidade) é satisfatória?
() Sim () Não GI=
- 2.O processamento das informações do *software* é eficiente?
() Sim () Não GI=

●Comportamento em relação aos recursos:

- 1.O *software* é munido de outro recurso, como por exemplo: *word*, *excel*, *internet*, entre outros?
() Sim () Não GI=
- 2.Caso exista outro recurso utilizado no *software*, é fácil de operá-lo?
() Sim () Não () Não se Aplica GI=

Por uma das características da ISO/IEC 9126 chamada de Manutenibilidade entende-se: “Conjunto de atributos que evidenciam o esforço necessário para fazer modificações especificadas do *software*”. As questões que não apresentarem valor resposta deverão ser respondidas conforme a figura 1, respondendo também o grau de importância. Como subcaracterísticas, tem-se:

●Analisabilidade:

- 1.Caso ocorra uma falha, é fácil recuperar esses dados?
() Sim () Não () Não se Aplica GI=

●Modificabilidade:

1.É fácil de modificar o *software*?

() Sim () Não () Não se Aplica GI=

●Estabilidade:

1.Quando o *software* passa por alterações, as informações são mantidas corretamente?

() Sim () Não () Não se Aplica GI=

2.Estas alterações não prejudicam o desempenho do *software*?

() Sim () Não () Não se Aplica GI=

●Testabilidade:

1.É fácil testar o *software* quando se fazem alterações?

() Sim () Não () Não se Aplica GI=

Por Portabilidade entende-se: “Conjunto de atributos que evidenciam a capacidade do *software* ser transferido de um ambiente para outro”. As questões que não apresentarem valor resposta deverão ser respondidas conforme a figura 1, respondendo também sobre o grau de importância. Como subcaracterísticas tem-se:

●Adaptabilidade:

1.É fácil adaptar o *software* a outros ambientes?

●Capacidade de ser instalado:

1.O *software* apresenta manual de instalação?

() Sim () Não GI=

2.O *software* faz sua própria instalação automática?

3.É fácil instalar o *software* em outros ambientes?

●Conformidade:

1.O *software* está de acordo com padrões ou convenções de portabilidade?

●Capacidade para substituir:

1.É possível substituir o *software*, por uma nova versão?

() Sim () Não () Não se Aplica GI=

2.Caso seja possível substituir o *software*, qual é o menor grau de facilidade?

Apresentação do questionário aplicado ao usuário do *software* durante a entrevista: O questionário aplicado ao usuário do *software* não foi tabulado e nem computado, pois o *software* foi fornecido pelo seu fabricante, impossibilitando a presença de algum usuário. O referido questionário servirá para verificar a satisfação do usuário que utiliza diariamente o *software* na realização e execução do seu trabalho. Todas as questões são apresentadas em LODI (2002).

Modelo Estatístico para Tabulação dos Resultados: por meio de um modelo estatístico, pode-se medir a qualidade e encontrar o resultado da avaliação quantitativa de todos os requisitos do produto de *software*. Para isso, com base em SPIEGEL (1978) & TOLEDO (1991), busca-se o auxílio de fórmulas que são mostradas a seguir. As mesmas estão mais detalhadas e explicadas em LODI (2002).

Fórmulas utilizadas para efetuar o cálculo do número de pontos totais do teste

$$\#PT = 4(n_1p_1) + 3(n_2p_2) + 2(n_3p_3) + 1(n_4p_4) + 0(n_5p_5)$$

$$\#PT = \sum C_k n_1 p_1$$

Fórmula utilizada para encontrar o Valor Máximo do produto de *software*

$$\#PM = \sum 4n_1 p_1$$

Obtenção do Resultado Final do Nível de Qualidade: Para obter-se o resultado final, indicando o nível de qualidade do produto de *software* em relação à ISO/IEC 9126, o avaliador ou testador deve desconsiderar todas as questões que receberam a opção "Não Se Aplica" simbolizada por "NA".

Para facilitar a aplicação das fórmulas e obter o nível de qualidade em que o produto de *software* testado se encontra, é necessário seguir todos os passos mostrados na figura 2, a seguir.

Resultado Final: Após conhecer o PM, que significa o valor máximo de pontos a que o *software* pode atingir, será aplicada uma regra de três, entre o PM e o PT, que indica o número de pontos totais que o *software* atingiu durante a realização da avaliação.

Com o resultado da regra de três, adquirida após a avaliação obtida em pontos, deve-se verificar, na figura 3, o nível do produto de *software* analisado. Este pode ter, como resultado, um índice considerado "Excelente", "Bom" ou "Regular", que indicam uma "Satisfação" no produto de *software* ou por meio do índice "Insuficiente" que demonstra "Insatisfatório" o

Tabela 1 - Etapas de aplicação das fórmulas para obter o nível de qualidade.

ETAPAS	DESCRIÇÃO
Passo 1	Somar o peso das questões que receberam, após a aplicação do teste do produto de <i>software</i> , conceito 0.
Passo 2	Somar o peso das questões que receberam, após a aplicação do teste ao pacote de <i>software</i> , conceito 1.
Passo 3	Somar o peso das questões que receberam, após a aplicação do teste ao pacote de <i>software</i> , conceito 2.
Passo 4	Somar o peso das questões que receberam, após a aplicação do teste ao pacote de <i>software</i> , conceito 3.
Passo 5	Somar o peso das questões que receberam, após a aplicação do teste ao pacote de <i>software</i> , conceito 4.
Passo 6	O testador deverá multiplicar o valor obtido de cada item anterior por seus respectivos valores de 0 a 4 e após somá-los: $a \times 0 + b \times 1 + c \times 2 + d \times 3 + e \times 4$

produto de *software*.

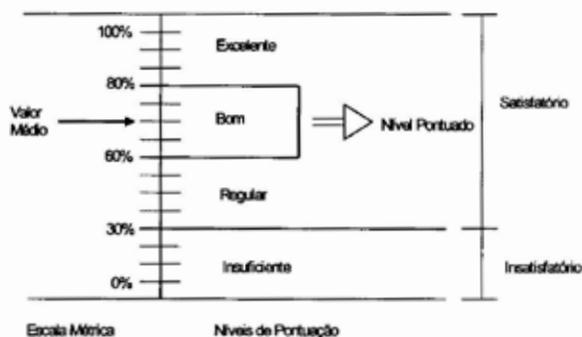


Figura 2 - Valor Medido e Nível de Pontuação, segundo a ISO/IEC 9126.

Parecer do Testador: O testador, de posse do resultado final de avaliação do produto de *software*, deve redigir um relatório, no qual constará o seu parecer sobre todos os itens analisados, ressaltando principalmente, os fatores negativos e dará sugestões para o melhoramento do *software*.

Na próxima parte, são mostradas avaliações e pareceres realizados por três testadores, em um produto de *software*, com base na aplicação do método *MSQuality*.

PARTE IV: ESTUDO DE CASO

Para se verificar a aplicação do *MSQuality*, método proposto neste trabalho, fez-se o estudo de caso de um produto de *software*. Com ele, deseja-se demonstrar o nível de pontuação do *software*, com base na ISO/IEC 9126, bem como o parecer do testador, em que constam os fatores positivos e negativos do produto de *software* e citando sugestões para sua melhoria.

Com o objetivo de averiguar a veracidade e confiança do método *MSQuality*, o estudo de caso foi realizado por três testadores diferentes, em um produto de *software* denominado Ábacus 4.0, que apresenta várias funções como: atualização de valores individuais, fórmulas financeiras, gráficos profissionais, entre outras. A *New Soft Informática*, empresa criadora do *software*, foi responsável, pelo seu desenvolvimento, para a plataforma *Windows*.

A seguir, serão apresentadas as avaliações dos testadores. O parecer e as sugestões de cada testador encontram-se em (LODI, 2002).

Para o 1º, 2º e 3º testador, o *software* obteve o valor de 87,50%, 86,91% e de 86,47% respectivamente. De posse dos valores obtidos no método estatístico e com base na Tabela de Valor Medido e Nível de Pontuação da ISO/IEC 9126, conclui-se que o *software* Ábacus 4.0 é considerado “Excelente” e enquadra-se como um *software* Satisfatório. Observa-se que ocorreram semelhanças nos resultados das avaliações do produto de *software*, certificando-se que o método de avaliação proposto, neste trabalho, é adequado e correto.

Na parte V, serão abordados Linguagem de Programação e Banco de Dados utilizados para a criação da *FAAPROS*. A ferramenta tem como finalidade automatizar o método de avaliação e fornecer com rapidez e exatidão os resultados da avaliação do produto de *software*, mostrando se o mesmo apresenta características de qualidade.

PARTE V: *FAAPROS*

A implementação da Ferramenta de Auxílio à Avaliação de Produto de *software* - *FAAPROS*- tem como objetivo principal propiciar maior facilidade para que o testador realize a avaliação do *software*, pois, de posse do questionário, devidamente preenchido, o testador deverá transcrever os resultados das questões para o sistema. O método estatístico existente na ferramenta executará todas as fórmulas matemáticas e dará com mais rapidez e exatidão o valor do quantitativo do produto de *software*.

A conexão entre o banco de dados e a ferramenta é feita por meio de alias denominada *Quality*. A ferramenta foi implementada com o apoio da linguagem de programação *Delphi 6.0*, utilizando como banco de dados, o *InterBase 6.0*. O acesso ao sistema é restrito e controlado pelo uso de senha, impossibilitando a entrada de usuários não autorizados.

CONCLUSÕES

Com a busca incessante da perfeição e com a finalidade de sobressair no mercado competitivo dos negócios, busca-se obter a qualidade, que é uma das características utilizadas desde o desenvolvimento até o acabamento de um produto de *software*.

Com este trabalho, apresenta um método de avaliação, que auxilie os desenvolvedores de *software*, a verificar se as características de qualidade existentes na norma ISO/IEC 9126 estão presentes no produto de *software*. No referido método, pode-se, também, verificar a aceitação do produto por parte do seu usuário, obtendo-se o grau qualitativo do *software*.

O método de avaliação denominado *MSQuality* é correto e eficiente, pois o estudo de caso realizado pelos testadores, no *software* denominado *Ábacus*, ocasionou em um resultado praticamente igual e a *FAAPROS* foi implementada com a finalidade de automatizar o processo da avaliação do *software*.

Como trabalhos futuros, é proposta a implementação da Classe Testes, verificando o tipo de teste a ser utilizado para as questões do questionário pertencente à norma do *software*. A ferramenta também poderá ser implementada para *internet*, propiciando uma avaliação mais rápida.

REFERÊNCIAS BIBLIOGRÁFICAS

DEMING, William Edwards. 1990. **Qualidade: a revolução da administração**. Rio de Janeiro: Brasport.

FERNANDES, Claudio Piccolo. 2001. **Aplicação de normas internacionais para avaliação de qualidade de produto de software**. UNIFRA, Trabalho de Final de Graduação.

ISO 9000-3. 1993. **Normas de gestão da qualidade e garantia da qualidade**. Rio de Janeiro.

ISO/IEC 12119. 1997. **International standard. information technology -software packages - Quality requirements and testing**.

ISO/IEC 14598-5. 1997. **International standard. information technology - software product evaluation - part 5: process for evaluators.** May.

ISO/IEC 9126-1. 1997. **International standard. information technology - software Quality characteristics and metris - part 1: Quality characteristics and sub-characteristics.** Jan.

PACHECO JR, Walter, 1995. **Qualidade na segurança e higiene do trabalho.** São Paulo: Atlas.

PRESSMAN, Roger. 1995. **Engenharia de software.** São Paulo: Makron Books.

ROCHA, Ana Regina Cavalcanti da. 2001. **Qualidade de software: teoria e prática.** Rio de Janeiro.

SPIEGEL, Murray R. 1978. **Probabilidade e estatística.** São Paulo: Makron Books. (Coleção Schaun).

STRAFACCI, Valter. 1996. **Relevância da gestão de engenharia de software para a garantia da qualidade.** São Paulo: ITA.

TOLEDO, Geraldo Luciano & OVALLE, Ivo Izidoro. 1991. **Estatística Básica.** 2.ed. São Paulo: Atlas.

WEINBERG, Geral M. 1993. **Software com qualidade: pensando e idealizando sistemas.** São Paulo: Makron Books.