

Stênio Pereira Viveiros

Um estudo para a utilização do método QFD na definição de medidas de qualidade de produtos de software

Dissertação apresentada ao Departamento de Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Belo Horizonte

Junho de 2006

*Dedico este trabalho aos meus pais,
Milton e Ruth.*

Agradecimentos

Ao meu orientador, professor Clarindo, pela competência e disponibilidade durante todo o meu mestrado. Sua ajuda foi fundamental para o meu desenvolvimento tanto pessoal quanto acadêmico.

Ao meu amor, Mariana, pela compreensão nos momentos mais difíceis e por me apoiar sempre.

Ao meu melhor amigo, Daniel Câmara, que me ajudou muito na discussão de idéias do trabalho e na revisão do texto.

Ao professor Rodolfo, pelas dicas e orientações sobre a validação do trabalho. Ao colega Márcio Barbosa, pelo material e esclarecimentos de dúvidas sobre QFD.

Ao Synergia, pela compreensão de algumas ausências na parte final do trabalho e pela bolsa concedida. Aos amigos da equipe de testes por serem tão competentes e exigirem o mínimo da minha gerência.

Ao meu pai, minha mãe e meu irmão que me ajudaram tanto neste último ano, convivendo comigo e apoiando em tudo que precisava.

Resumo

Este trabalho apresenta um método para padronização da aferição da qualidade de produtos de software. O método proposto utiliza-se de técnicas como QFD (Função de desdobramento da qualidade) e de padrões reconhecidos, como a norma ISO/IEC-9126 *Software engineering – Product Quality*. A premissa básica deste trabalho é que a expectativa do usuário com relação à qualidade de produtos de software vem crescendo e conseqüentemente enfatiza a necessidade de se identificar e padronizar parâmetros de qualidade de software. Para identificar a expectativa do usuário, o método utiliza-se da técnica QFD, desdobrando a qualidade exigida pelos usuários em atributos do software. A padronização desses atributos foi realizada por meio de sua correlação com as características de qualidade encontradas na norma ISO/IEC -9126. Com essa padronização da qualidade é possível coletar medidas quantitativas e criar uma base de dados históricos, que pode ser usada para melhorar continuamente a qualidade dos produtos desenvolvidos. Resultados iniciais experimentais mostram que o método definido pode endereçar esses aspectos e fazer com que o desenvolvimento de sistemas use um processo para a padronização de medidas de qualidade.

Abstract

This work presents a method to standardize the measurement of quality of software products. The proposed method utilizes techniques such as QFD (Quality Function Deployment) and recognized standards, like the ISO/IEC-9126 Software engineering – Product Quality. The basic premise of this work is that the expectations of the users with regard to software quality are growing, and this emphasize is the necessity of identifying and standardizing parameters for software quality. To identify the user expectations, the proposed method makes use of QFD techniques, mapping the user required quality into software attributes. The standardization of these attributes was achieved through their correlation with quality characteristics defined in the ISO/IEC -9126 standard. Another aspect of this work is that the standardization permits the use of quantitative measures to create a historical database, which can be used to continuously improve the quality of the developed products. Initial experimental results shows that the defined method can address these aspects and make use of a process for system development that standardized quality measurement.

Sumário

INTRODUÇÃO	1
1.1 MOTIVAÇÃO	3
1.2 OBJETIVOS DO TRABALHO	4
1.3 LIMITES DO TRABALHO	4
1.4 ORGANIZAÇÃO DESTE DOCUMENTO	5
REFERENCIAL TEÓRICO	6
2.1 QUALIDADE DE SOFTWARE.....	6
2.2 O MÉTODO QFD	9
2.3 MEDIÇÕES EM SOFTWARE	12
2.4 PADRONIZAÇÃO DA QUALIDADE DE SOFTWARE.....	14
2.5 CONCEITOS DO PROCESSO PRAXIS 2.1 E MERCI 1.0	17
METODOLOGIA DE PESQUISA.....	20
3.1 TIPO DE PESQUISA	20
3.2 PROCEDIMENTOS METODOLÓGICOS	21
PROPOSTA DE METODOLOGIA PARA AFERIÇÃO DA QUALIDADE DE PRODUTOS DE SOFTWARE	22
4.1 IMPORTÂNCIA DA NORMA ISO-9126 E DO MÉTODO QFD NA CONSTRUÇÃO DO MÉTODO	23
4.2 METODOLOGIA PARA AFERIÇÃO DA QUALIDADE DE PRODUTOS DE SOFTWARE	24
4.2.1 <i>Nível de Avaliação.....</i>	26
4.2.2 <i>Matriz de Qualidade do Usuário.....</i>	29
4.2.3 <i>Matriz de Qualidade em Uso.....</i>	31
4.2.4 <i>Matriz de Qualidade Externa</i>	33
4.2.5 <i>Matriz de Qualidade Interna</i>	35
CLASSIFICAÇÃO E SELEÇÃO DAS MEDIDAS DE QUALIDADE	37
INSTANCIAÇÃO DO MÉTODO DE AFERIÇÃO DE QUALIDADE.....	47
6.1 MATRIZ DE QUALIDADE DO USUÁRIO.....	47
6.2 MATRIZ DE QUALIDADE EM USO	48
6.3 MATRIZ DE QUALIDADE EXTERNA.....	49
6.4 MATRIZ DE QUALIDADE INTERNA.....	51
6.5 APLICAÇÃO DO MÉTODO PARA O PRODUTO MERCI 1.0	52
6.5.1 <i>Matriz de Qualidade do Usuário.....</i>	52
6.5.2 <i>Matriz de Qualidade em Uso.....</i>	57
6.5.3 <i>Matriz de Qualidade Externa</i>	60
6.5.4 <i>Matriz de Qualidade Interna</i>	65
CONCLUSÃO	70
7.1 TRABALHOS FUTUROS	71
REFERÊNCIAS BIBLIOGRÁFICAS	72
ANEXO I - GLOSSÁRIO.....	75

Lista de figuras

FIGURA 1 – Adaptada do modelo de Kano [KANO96]	8
FIGURA 2 – Unidades básicas do QFD	10
FIGURA 3 – Representação de uma matriz com seus elementos constituintes.....	11
FIGURA 4 – Excerto do <i>Framework de Atributos</i> de Fenton [FENTON96]	13
FIGURA 5 – Relacionamento dos tipos de métricas de acordo com a norma ISO-9126	23
FIGURA 6 – Modelo conceitual para aferição de qualidade para produtos de software.....	25
FIGURA 7 – Matriz de Qualidade do Usuário.....	29
FIGURA 8 – Matriz de Qualidade em Uso.....	31
FIGURA 9 – Matriz de Qualidade Externa.....	33
FIGURA 10 – Matriz de Qualidade Interna.....	35
FIGURA 11 – Classificação de atributos de qualidade adaptada do trabalho de Fenton.....	38
FIGURA 12 – Classificação de atributos de qualidade interna e externa	39
FIGURA 13 – Classificação de atributos de qualidade em uso	39
FIGURA 14 – Qualidade externa de acordo com a norma ISO-9126.....	50
FIGURA 15 – Excerto da planilha de priorização das funções do Merci 1.0	54
FIGURA 16 – Matriz de Qualidade do Usuário para o produto Merci 1.0.....	55
FIGURA 17 – Gráfico de importância dos atributos de qualidade	56
FIGURA 18 – Matriz de Qualidade em Uso para o produto Merci 1.0	57
FIGURA 19 – Gráfico de importância das características de qualidade em uso	58
FIGURA 20 – Distribuição da preocupação da qualidade em uso do Merci	59
FIGURA 21 – Níveis de avaliação para características de qualidade em uso.....	60
FIGURA 22 – Matriz de Qualidade Externa para o produto Merci 1.0	61
FIGURA 23 – Diagrama de casos de uso do Merci 1.0 para o ator Sistema Financeiro	62
FIGURA 24 – Gráfico de importância das subcaracterísticas priorizadas de qualidade externa.....	63
FIGURA 25 – Distribuição da preocupação com a qualidade do Merci.....	64
FIGURA 26 – Níveis de avaliação para características de qualidade externa	65
FIGURA 27 – Matriz de Qualidade Interna para o produto Merci 1.0	67
FIGURA 28 – Priorização das medidas internas de qualidade	68

Lista de tabelas

TABELA 1 – Características e subcaracterísticas de qualidade interna e externa da norma ISO-9126 .	16
TABELA 2 – Características de qualidade em uso da norma ISO-9126	16
TABELA 3 – Fases e iterações do Praxis 2.1 [PAULA03]	17
TABELA 4 – Fluxos técnicos e gerenciais do Praxis 2.1 [PAULA03]	18
TABELA 5 – Artefatos do Praxis 2.1 [PAULA03]	19
TABELA 6 – Níveis de avaliação para o aspecto de segurança humana	27
TABELA 7 – Níveis de avaliação para o aspecto econômico	27
TABELA 8 – Níveis de avaliação para o aspecto de segurança da informação	28
TABELA 9 – Níveis de avaliação para o aspecto ambiental	28
TABELA 10 – Medidas internas da norma ISO-9126	43
TABELA 11 – Algumas medidas do padrão IEEE-982	45
TABELA 12 – Medidas do processo Praxis	46

Capítulo 1

Introdução

Com o crescimento da concorrência na área de desenvolvimento de software, a preocupação com qualidade é fator preponderante, uma vez que pode ser um diferencial para as empresas. A garantia da qualidade, esperada pelo usuário, é uma atividade dispendiosa e, muitas vezes, dependente da experiência da organização para definir as medidas¹ para realizá-la. A definição de medidas de qualidade adequadas para as características da organização, por meio da utilização de um processo organizado, pode resultar em uma melhor adequação da qualidade de seus produtos. Esta dissertação tem como objetivos propor um método que visa aferir padronização da qualidade do produto esperada pelo usuário e a priorização das medidas mais adequadas a essa qualidade. Outras contribuições para o processo de desenvolvimento de software também podem ser alcançadas com a aplicação das técnicas propostas, visando uma melhoria na produção de software.

O produto de software é cada vez mais utilizado como ferramenta crítica em várias empresas. Para as quais, o sucesso do negócio pode depender da qualidade do software que utilizam. Dessa forma, a qualidade se torna um critério chave na aquisição de seus produtos.

Porém, na maioria das vezes, a qualidade não é a principal preocupação no desenvolvimento de software. Uma empresa fornecedora costuma priorizar o cumprimento do prazo de entrega e a produtividade do projeto, em detrimento da qualidade esperada pelo usuário. Ao invés de acompanhá-la durante a execução do projeto, é, muitas vezes, somente no final que se verifica se a qualidade do produto está adequada. A correção das falhas para adequar o produto causa atrasos no projeto, e conseqüentemente, a insatisfação do cliente. Portanto, é importante acompanhar a qualidade do software desde o início do seu desenvolvimento.

O acompanhamento da qualidade de produto pode ser realizado com o auxílio de um processo de definição de medidas, como, por exemplo, GQM [BASILI92], GDMS [PARK96] e PSM [DoD00]. Em linhas gerais, esses processos auxiliam na definição de medidas usadas no desenvolvimento de software a partir das metas da organização. Por exemplo, uma organização pode ter meta de desenvolver produtos de alta qualidade. Para definir medidas para essa meta, os processos acima citados desdobram a meta em perguntas, relacionadas com a qualidade do produto, e, para responder a essas perguntas, são atribuídas medidas, que serão coletadas durante o desenvolvimento do software. Contudo, o correto desdobramento dessa meta em perguntas e, em seguida, em medidas,

¹ O termo “métrica” é também usado na área de Engenharia de Software.

depende do conhecimento prévio no acompanhamento da qualidade de produtos de software.

O conhecimento adquirido no desenvolvimento de projetos de software pode ser uniformizado por meio da padronização da experiência adquirida de acordo com modelos e procedimentos. Na área de engenharia de software, alguns modelos e procedimentos são divulgados, por exemplo, pelas organizações ISO e IEEE, no formato de normas e padrões. Entre as normas estabelecidas, é possível citar algumas direcionadas para a qualidade de software, como, por exemplo, as normas ISO-9126, ISO-14598 e ISO-15504. Sendo que, dentre os padrões relacionados com qualidade de software citam-se os padrões IEEE-982, IEEE-1044 e IEEE-1061.

Em especial, a norma ISO-9126 [Iso01] categoriza a qualidade de produtos de software em características de qualidade, sendo elas divididas em dois modelos: qualidade interna e externa e qualidade em uso.

O modelo de qualidade em uso, de acordo com a norma, categoriza a qualidade em quatro características que definem a visão de qualidade do produto software em uso pelo usuário: efetividade, produtividade, segurança e satisfação. Já o modelo de qualidade interna e externa, de acordo com a norma, categoriza a qualidade do produto em seis características: funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade. Essas características são divididas em subcaracterísticas que podem ser mensuradas por medidas internas e externas.

A mensuração das subcaracterísticas do modelo de qualidade, da norma ISO-9126, auxilia no controle da qualidade do produto durante a sua construção. O controle da qualidade possibilita ao gerente tomar ações corretivas no desenvolvimento do produto e, assim, alcançar a qualidade esperada pelo usuário. O controle da qualidade, na área de engenharia de software, é assunto discutido há pouco tempo. Por exemplo, a norma ISO-9126, que padroniza a qualidade para facilitar seu controle, foi publicada em 2001.

Em outras engenharias, o controle da qualidade no desenvolvimento de produtos é assunto discutido há mais tempo. Como exemplo, pode-se citar o livro de Juran [JURAN98], que aborda o controle de qualidade na indústria, com a primeira edição publicada em 1951. Nesse livro, o autor sugere a utilização de alguns métodos e ferramentas para auxiliar o planejamento, o controle e a melhoria da qualidade. Entre esses métodos, o método QFD [AKAO96] auxilia o processo de desenvolvimento do produto, buscando, traduzindo e transmitindo as necessidades e os desejos do cliente [CHENG95].

Para organizar as necessidades do cliente, o método QFD possui um processo com passos bem-definidos que podem ser repetidos e assim, possibilitam que essas necessidades sejam mais bem geridas. Como elas variam para cada produto seria interessante que fossem organizadas de acordo com uma norma. Dessa forma, nesta dissertação, é proposta um método para definição de medidas para qualidade de produto que utiliza o método QFD em conjunto com a norma ISO-9126.

1.1 Motivação

Uma das motivações deste trabalho é a falta de formalização das necessidades de qualidade esperadas de um produto de software que são levantadas junto ao usuário. Por exemplo, uma necessidade normalmente esperada pelo usuário é que o software seja rápido e fácil de usar. Essa necessidade pode significar diferentes características de qualidade para um produto de software como: eficiência e usabilidade. Para o engenheiro de software, é necessário relacionar as necessidades de qualidade esperadas pelo usuário com as características de qualidade padronizadas. Com o relacionamento das necessidades de qualidade com as características de qualidade padronizadas por uma norma, como, por exemplo, a ISO-9126, espera-se levantar e documentar melhor a qualidade esperada pelo usuário.

Além de levantar corretamente a qualidade esperada pelo usuário, existe uma dificuldade do engenheiro de software em definir as medidas internas usadas para acompanhar a qualidade durante a construção do produto. A qualidade esperada pelo usuário precisa ser mensurada no processo de desenvolvimento do produto para que ela seja devidamente contemplada. Porém, normalmente, é difícil criar um modelo único que relacione a qualidade final do produto e as medidas de qualidade dos atributos intermediários do desenvolvimento do software, conforme discutido na ISO-9126 [Iso01]. Uma possibilidade, utilizada na engenharia, é ter um método, como o QFD, que desdobre a qualidade esperada pelo usuário, com auxílio da equipe de desenvolvimento, para o processo de produção da empresa. Com esse método são percebidos os pontos do processo de desenvolvimento que impactam a qualidade final do produto.

É importante ressaltar que a percepção de qualidade esperada pelo usuário é um conceito dinâmico, que sofre uma evolução ao longo do tempo, motivada por fatores como a evolução da tecnologia e aspectos sociais. Sendo assim, para o desenvolvimento de um novo produto, é preciso executar novamente o método QFD, pois a qualidade esperada pelo usuário pode ter se alterado. Isso requer uma nova aplicação do método QFD, utilizando a experiência de trabalhos anteriores, permitindo ao desenvolvedor garantir que a qualidade esperada pelo usuário será corretamente desdobrada e alcançada. Este quadro ilustra a importância de se ter um processo repetível, com resultados padronizados, para a gestão da qualidade do produto de software.

Como a qualidade esperada pelo usuário varia para cada projeto, é difícil criar um modelo de medição de qualidade de software único na empresa. A qualidade do produto varia com as necessidades de sua utilização, definidas pelos seus usuários. Por exemplo, para um software utilizado em uma mercearia, o usuário define como qualidade a facilidade em operá-lo. Em contrapartida, em um software científico, a qualidade para o usuário está na precisão dos resultados. Tais situações necessitam de medidas diferentes para atenderem às expectativas do cliente. Dessa forma, é importante que a empresa fornecedora do software crie um modelo específico para mensurar a qualidade esperada pelo usuário para cada projeto a ser desenvolvido.

Cada empresa possui um processo de desenvolvimento diferente e para cada processo é preciso coletar diferentes medidas internas para alcançar a qualidade esperada no produto. A qualidade final do produto depende da qualidade interna dos artefatos do processo de

desenvolvimento. A qualidade das atividades do processo influencia na qualidade dos artefatos que são construídos por elas [Iso01]. As atividades de desenvolvimento variam de uma empresa para outra dependendo do processo adotado por elas. Dessa forma, um modelo para garantir qualidade deveria ser independente do processo de desenvolvimento utilizado pela empresa.

1.2 Objetivos do trabalho

O principal objetivo deste trabalho é a criação de um método de aferição da qualidade de um produto de software, utilizando-se medidas de qualidade. A definição das medidas é realizada pelo método QFD, que desdobra a qualidade esperada pelo usuário. As medidas obtidas para aferição da qualidade são organizadas pela utilização do modelo de qualidade da norma ISO-9126.

Com o método QFD, a qualidade esperada pelo usuário é desdobrada na qualidade das atividades do processo de desenvolvimento. Dessa forma, a qualidade das atividades pode ser mensurada durante a construção do produto. A mensuração dessas atividades coleta indicadores para estimar a qualidade final do produto, que, assim, pode ser aferida conforme a qualidade inicialmente definida pelo cliente do software.

O método padroniza a qualidade do software esperada pelo cliente com o modelo de qualidade definido pela norma, que possibilita reaproveitar medidas utilizadas no ambiente da organização desenvolvedora. Essas medidas são priorizadas por meio do desdobramento da qualidade esperada pelo usuário no modelo deste trabalho. Dessa forma, a organização pode reutilizar o conhecimento de mensuração adquirido em outros projetos e ainda forma uma base histórica para futuros projetos.

Um outro objetivo específico desta pesquisa é a classificação de medidas, encontradas na literatura, utilizando o modelo de qualidade da norma ISO-9126. Essas medidas, descritas e validadas em outros trabalhos, podem ser utilizadas na mensuração da qualidade pelas empresas.

1.3 Limites do trabalho

Tão importante quanto enumerar os objetivos do trabalho é esclarecer os seus limites, visando delimitar mais precisamente o escopo que está sendo tratado.

Este trabalho trata especificamente do desdobramento da qualidade esperada pelo usuário em medidas de qualidade interna. Além da mensuração da qualidade, uma empresa poderia verificar outros aspectos do seu desenvolvimento de software. Por exemplo, uma empresa poderia verificar a produtividade na utilização de uma nova ferramenta de desenvolvimento. A mensuração da produtividade da nova ferramenta é um objetivo de medição da empresa e precisa ser desdobrada em

medidas. Para realizar essa tarefa de desdobramento, a empresa, poderia utilizar um processo de medição como GQM [BASILI92] ou GDSP [PARK96]. Esses processos pregam que a definição de medidas deve visar alcançar as metas de medição da empresa e as medidas são definidas através de perguntas. Uma meta comum em várias empresas é desenvolver produtos com alta qualidade. Esta dissertação tem como objetivo o desdobramento da qualidade esperada pelo usuário em medidas internas do desenvolvimento de software.

Esta dissertação visa criar um modelo para garantir a qualidade do produto e não a qualidade de processo do desenvolvimento. O desdobramento da qualidade é realizado a partir da qualidade esperada pelo usuário do produto de software. Além da qualidade de produto a empresa precisa verificar a qualidade do processo de desenvolvimento, e assim, garantir alcançar outros objetivos como produtividade, lucratividade, etc. Na verificação da qualidade de processo de desenvolvimento existem trabalhos como ISO-15504 [ISO03C] e CMMI [CMMI02]. Esses trabalhos visam à melhoria contínua do processo através da avaliação da maturidade do processo de desenvolvimento.

Não faz parte do escopo a definição da ferramenta para aplicação do modelo desta pesquisa. O modelo é construído desdobrando a qualidade esperada pelo usuário através do método QFD. Esse método possui unidades de trabalho simples como tabelas e matrizes, e assim, não necessitando de uma ferramenta específica. Nesta dissertação foi utilizada a ferramenta Excel[®], com algumas funções definidas como macros, para construir as tabelas, matrizes e gráficos.

1.4 Organização deste documento

O trabalho de definição do método de desdobramento da qualidade esperada pelo usuário seguiu um conjunto de atividades que facilitou sua documentação.

A primeira atividade constituiu o estudo de trabalhos relacionados com qualidade e mensuração. Esses trabalhos, que fundamentaram o estudo realizado para a criação do método, estão descritos no Capítulo 2. O método de pesquisa e o tipo de pesquisa que este trabalho se classifica estão descritos no Capítulo 3. A próxima atividade foi criar o método de aferição de qualidade de software, proposto neste trabalho, que está descrita no Capítulo 4.

Durante o levantamento bibliográfico foram encontradas várias medidas para qualidade de software. Essas medidas foram classificadas conforme a norma ISO-9126 e descritas no Capítulo 5. Esse conjunto de medidas é usado para a aplicação do método.

O método de aferição da qualidade de produto é instanciado ao processo Praxis no Capítulo 6 para mostrar a sua utilização a um processo matricial. E, como exemplo, foi realizado a aplicação do método ao produto Merci 1.0, exemplo didático do Praxis.

No Capítulo 7 deste documento, a conclusão da pesquisa, as contribuições e os trabalhos futuros são discutidos.

Capítulo 2

Referencial teórico

Este capítulo descreve as principais referências utilizadas para construção do modelo de aferição de qualidade de produto. Essas referências serão descritas sucintamente nas seguintes seções:

- A seção 2.1 descreve o termo qualidade de forma geral e os métodos de garantia de qualidade utilizados pela engenharia.
- A seção 2.2 apresenta o método QFD, utilizado para construção do método proposta.
- A seção 2.3 apresenta métodos e recomendações na área de mensuração da qualidade de software.
- A seção 2.4 mostra os padrões e as normas de qualidade, utilizados na construção do método.
- A seção 2.5 descreve o processo Praxis 2.1 e o produto Merci 1.0, utilizados na aplicação do método.

2.1 Qualidade de software

Esta seção irá apresentar o conceito de qualidade, básico para o entendimento do trabalho. Para que ele seja útil o conceito de qualidade, muitas vezes tratado subjetivamente, é preciso definir de forma mais objetiva o seu significado. Entre as várias definições de qualidade que podem ser encontradas na literatura, algumas foram destacadas.

Segundo Deming [DEMING82], qualidade é “aperfeiçoamento contínuo e a firmeza de propósitos. Compreender o que acontece, construir e interpretar estatísticas e agir aperfeiçoando. Não há respostas corretas, apenas respostas geradas pelos métodos usados para gerá-las. O objetivo devem ser as necessidades do usuário, presentes e futuras”.

O termo qualidade, de acordo com Juran [JURAN98], é “adequação ao uso”.

A norma ISO 8402 [ISO95] define qualidade de produto como “a qualidade é a totalidade das características de uma entidade que lhe confere a capacidade de satisfazer às necessidades implícitas e explícitas”.

No contexto de engenharia de software, Basili [BASILI91] ressalta que qualidade é “um conceito multidimensional. Entre suas dimensões podem ser incluídas: a entidade de interesse, o ponto de vista sobre essa entidade de interesse e os atributos de qualidade dessa entidade”.

Ainda para a área de software, a norma IEEE [IEEE90] define qualidade de software: “a qualidade de software é um conjunto de características ou fatores de software, que determinam o nível de eficiência do software em uso, em relação ao atendimento das expectativas dos clientes²”.

Dentre essas definições, existe um ponto em comum que é o conceito de que a qualidade do produto deve atender às expectativas dos clientes. Essas expectativas variam da completeza dos requisitos funcionais do produto até o seu desempenho no ambiente do usuário. Dessa forma, o método de aferição da qualidade tem como base principal a qualidade no ponto de vista do usuário e começa com o levantamento da qualidade esperada do produto para o seu desdobramento em medidas de qualidade.

Para levantar a qualidade de forma precisa e padronizada, evitando divagações e direcionando para os atributos de qualidade pode-se, por exemplo, usar a técnica 5WHY. A técnica define um conjunto de perguntas, *Why* (Por quê?), *What* (O que?), *Who* (Quem?), *When* (Quando?), *Where* (Onde?) e *How* (Como?), que tem por objetivo auxiliar o usuário a identificar os atributos de qualidade, e assim, maximizar sua satisfação.

O aumento da satisfação do usuário depende de quais das suas expectativas foram atendidas pelo produto [KANO96]. O modelo de Kano [KANO96] descreve que todos os produtos e serviços possuem três curvas de expectativas do usuário relacionadas com sua satisfação: atrativa, linear ou básica. Nesse modelo, mostrada na FIGURA 1, o eixo X representa a presença das expectativas no produto e o eixo Y representa o grau de satisfação do usuário. Para identificar em qual curva as expectativas se encaixam no modelo de Kano, é preciso fazer duas perguntas para o usuário:

1. Como você se sente se a expectativa está ausente?
2. Como você se sente se a expectativa está presente?

A expectativa está presente na curva básica se a resposta para 1 tende a insatisfeito e para 2 tende a neutro. Por sua vez, a expectativa se enquadra na curva atrativa se a resposta para 1 tende a neutro e para 2 tende a satisfeito. Se a resposta do usuário for “depende”, a expectativa se enquadra na curva linear.

² O termo “cliente” neste contexto representa todos os interessados no produto (em inglês, *stakeholders*).

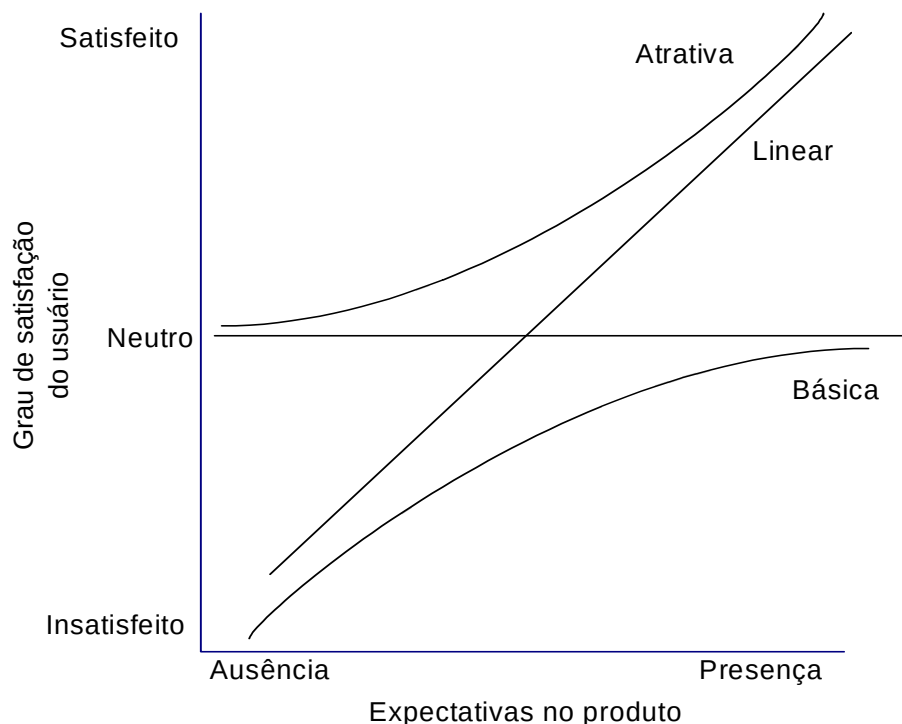


FIGURA 1 – Adaptada do modelo de Kano [KANO96]

O modelo de Kano é usado no levantamento da qualidade esperada pelo usuário, do método QFD [AKAO96] como guia na identificação dos atributos de qualidade da Matriz de Qualidade. De acordo Kano [KANO96], o usuário explicita somente as expectativas que estão na curva linear. Para identificar as expectativas que estariam na curva básica ou atrativa, o fabricante poderia usar o histórico da qualidade de outros produtos, se disponível.

Para exemplificar o modelo de Kano, pode-se utilizar como produto o automóvel e as expectativas de seus usuários. Em um automóvel, é esperado que ele seja confortável e fácil de dirigir. Esses dois atributos de qualidade não são expressos pelo usuário, mas a ausência deles causa muita insatisfação do usuário em relação ao carro. Sendo assim, esses atributos de qualidade se encontram na curva básica, do modelo de Kano.

Um dos fatores que incentivaram a utilização do QFD neste trabalho é que o método QFD vem sendo utilizado com sucesso no desdobramento da qualidade esperada pelo usuário no desenvolvimento de produtos em algumas indústrias da engenharia. De acordo com Cheng [CHENG95], a implantação do QFD objetiva duas finalidades específicas:

1. Auxiliar o processo de desenvolvimento do produto, buscando, traduzindo e transmitindo as necessidades e desejos do cliente;
2. Garantir qualidade durante o processo de desenvolvimento do produto.

Isto coincide perfeitamente com os objetivos deste trabalho, porém visando a área de desenvolvimento de software. Outro ponto extremamente atrativo do QFD é que o processo é bem definido, padronizado e permite o rastreamento entre entradas e resultados. Isto facilita sua

implantação e também ajuda na criação e manutenção de bases históricas de conhecimento das empresas.

O método QFD já foi utilizado como referência para outros trabalhos na área de engenharia de software, como por exemplo, o SQFD (*Software Quality Function Deployment*) [HAAG92], o trabalho de Pai [PAI02], o SQFD [ZULTNER90], uma extensão do QFD para aplicações de comércio eletrônico [HERZWURM03] dentre outros. Nesses trabalhos, o método QFD é utilizado para auxiliar o levantamento de requisitos de qualidade esperados pelo usuário, contudo existe pouca preocupação na padronização dessa qualidade. Outro ponto que pode ser observado nesses trabalhos é que não objetivam a definição de medidas para a qualidade levantada com o usuário.

Já o trabalho de Fehlmann [FEHLMANN02] mostra uma utilização do método QFD para definir medidas, chamadas de “métricas combinatórias”, para o desenvolvimento de software. Nesse trabalho, Fehlmann desdobra as necessidades definidas pelo usuário no processo de desenvolvimento, verificando seus relacionamentos e tentando definir suas medidas. Porém, não ficam claro como são selecionadas as medidas para acompanhar a qualidade durante o desenvolvimento. Outro ponto a observar no trabalho é que a qualidade levantada com o usuário não é padronizada, dificultando o reuso do conhecimento adquirido nos projetos.

2.2 O método QFD

O QFD é um método usado para transferir as necessidades dos clientes em requisitos de produto e de processo. Tem por fim estabelecer a qualidade no projeto, obter a satisfação do cliente, e efetuar o desdobramento das metas do referido projeto e dos pontos prioritários, em termos de garantia da qualidade, até o estágio de produção [AKAO96].

O QFD é dividido em Desdobramento da Qualidade (Quality Deployment - QD) e Desdobramento da Função Qualidade no sentido restrito. O QD consiste em “converter as exigências dos usuários em características de qualidade, definir a qualidade do projeto do produto acabado, desdobrar esta qualidade em qualidades de outros itens tais como: qualidade de cada uma das peças funcionais, qualidade de cada parte e até os elementos do processo, apresentando sistematicamente a relação entre os mesmos”. O Desdobramento da Função Qualidade, no sentido restrito, consiste no “desdobramento, em detalhes, das funções profissionais ou dos trabalhos que formam a qualidade, seguindo a lógica de objetivos e meios” [AKAO96].

Os objetivos principais do QFD são capturar a voz do cliente, reduzir a perda de informações e fornecer mecanismos a fim de que a equipe de desenvolvimento trabalhe eficientemente para atender os requisitos. Os refinamentos realizados, por meio de *modelos conceituais*, buscam realizar esses objetivos respondendo às seguintes questões:

- Quais são as qualidades que os clientes desejam?
- As quais funções o produto deveria atender e quais deveriam selecionar para prover o

produto ou serviço?

- Baseado nos recursos disponíveis, qual o melhor modo de prover o que os clientes querem?

Para operacionalizar os desdobramentos ou refinamentos, são utilizadas tabelas, matrizes e modelos conceituais que são denominados de unidades básicas de trabalho (UBTs), mostradas na FIGURA 2 [CHENG95].

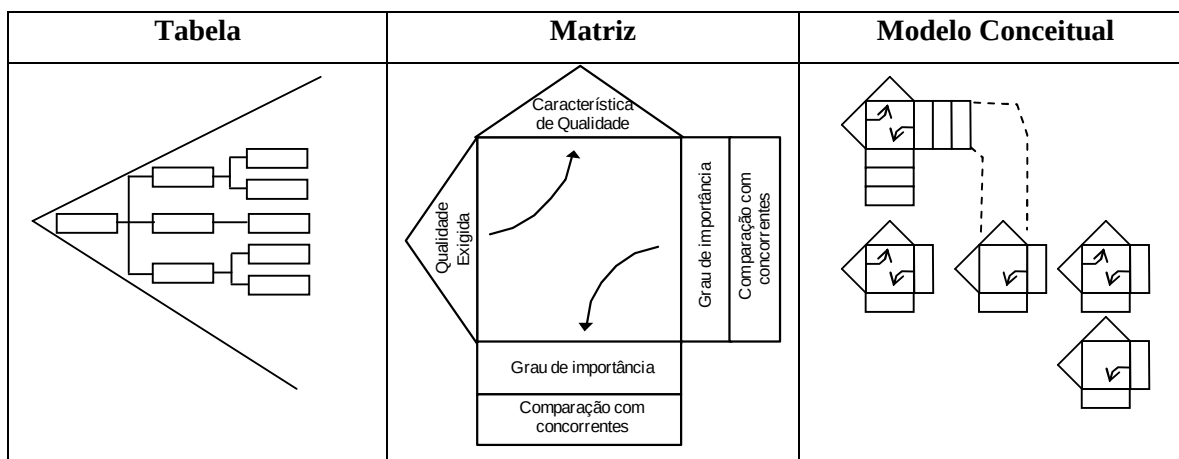


FIGURA 2 – Unidades básicas do QFD

A tabela no QFD é considerada a unidade elementar, onde se registra o detalhamento de algo de forma organizada e ordenada em níveis, semelhante a um diagrama em árvore. Essa organização hierárquica é representada graficamente por um triângulo. O seu conteúdo e a origem de informação dependem do propósito para o qual é construída. Por exemplo, na sistematização de tipos de usuários podemos definir uma tabela, cujos dados são extraídos das características dos usuários e tarefas que realizam.

Para confeccionar as tabelas, utilizam-se primeiramente, ferramentas de criatividade e participação como *Brainstorming*. Em sequência, utiliza-se o Diagrama de Afinidade [MIZUNO93], de forma a agrupar as contribuições afins sob algum critério de relação.

A partir de duas tabelas (por exemplo, A e B - FIGURA 3) elabora-se uma matriz, com a finalidade de dar visibilidade às relações entre elas. As relações podem ser de três tipos: qualitativa, quantitativa e de intensidade, cujos processos de definição são denominados *extração* (seta 1 - FIGURA 3), *conversão* (seta 2 - FIGURA 3) e *correlação* (seta 3 - FIGURA 3), respectivamente. A tabela C, mostrada na FIGURA 3, representa a importância dos itens da tabela A. Já a tabela D representa o resultado obtido através do processo de *conversão*.

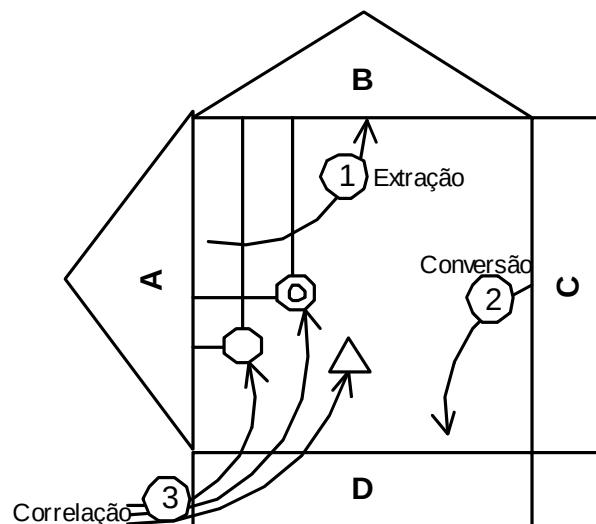


FIGURA 3 – Representação de uma matriz com seus elementos constituintes

A *extração* acontece quando elementos de uma tabela são obtidos a partir de elementos de outra tabela. A *conversão* consiste em transmitir a importância dos elementos de uma tabela para outros elementos de outra tabela. Esse processo é posterior ao processo de correlação. A *correlação* visa identificar as relações entre os elementos desdobrados do último nível das tabelas. O grau ou a intensidade da *correlação* é indicado por símbolos, tais como Forte, Fraca e Possível. Os valores normalmente usados para indicar esses critérios são $\odot = 9$, $\circ = 3$ e $\Delta = 1$ [CHENG95]. Além da análise de *correlação* é importante identificar as linhas e colunas em branco. Quando isso acontece, significa que algo foi omitido ou não é relevante.

A Matriz de Qualidade (versão japonesa) ou Casa de Qualidade (versão americana - *House of Quality*:HoQ), é a matriz mais conhecida e o ponto inicial da maioria das matrizes usadas no QFD. A matriz da Qualidade é constituída pela Tabela de Desdobramento da Qualidade Exigida e Tabela de Desdobramento de Características da Qualidade. A primeira contém as exigências do cliente, de onde se extrai os requisitos técnicos que são organizados na segunda tabela.

O *modelo conceitual* é um conjunto de tabelas e matrizes seqüenciadas de forma a permitir a visibilidade das relações existentes entre os componentes, mecanismos, processos, com a qualidade projetada para o produto. Representa o caminho por onde o desenvolvimento do projeto deve percorrer para atingir as metas estabelecidas. Um *modelo conceitual* completo contempla quatro dimensões de desdobramento: desdobramento da qualidade, da tecnologia, do custo e da confiabilidade. Entretanto, o tipo de modelo conceitual a ser construído é inteiramente dependente das metas, do tipo de empresa, da natureza do produto e da proximidade aos clientes.

2.3 Medições em software

Medição ou *mensuração* é o processo pelo qual números ou símbolos são associados a atributos de entidades no mundo real, com o objetivo de descrevê-la de acordo com um conjunto de regras claramente definidas [FENTON96]. O principal propósito da mensuração da qualidade de software é fornecer resultados quantitativos referentes aos produtos de software para que estes sejam compreensíveis, aceitáveis e confiáveis por qualquer parte interessada [ISO87]. Alguns fatores que necessitam de medição para que sejam efetivos na avaliação de qualidade são a satisfação dos usuários e o retorno econômico [KUMAR01].

Vários trabalhos sobre métricas para avaliação da qualidade de produto podem ser encontrados na literatura, como por exemplo, o trabalho de Andersson [ANDERSSON90], Riguzzi [RIGUZZI96], a norma ISO-9126 [ISO03A], [ISO03B], [ISO04] e o padrão IEEE-982 [IEEE88]. O trabalho de Andersson discute alguns conceitos relacionados com mensuração da qualidade de software e define um conjunto de métricas para os fatores de qualidade: manutenibilidade e confiabilidade. Já no trabalho de Riguzzi são apresentados conceitos gerais de mensuração de software, como por exemplo, o processo de identificação dos atributos a serem medidos, a validação das medidas selecionadas. Nesse trabalho também é apresentado um exemplo de conjunto de medidas. A norma ISO-9126 apresenta várias medidas de qualidade de software organizadas de acordo com seu modelo de qualidade e as descreve detalhadamente em tabelas. O padrão IEEE-982 define um conjunto de métricas de atributos do produto, processo e recurso para mensurar a característica confiabilidade de software.

Esses trabalhos de forma geral definem um conjunto de métricas que podem avaliar um ou mais aspectos de software. Além desses aspectos, devem-se mensurar também outras características de um software, como por exemplo, satisfação, tamanho do produto dentre outras. Contudo geralmente é inviável mensurar todas as medidas em um projeto devido ao custo associado a suas coletas. Dessa forma, torna-se necessário priorizar as medidas que sejam adequadas a cada tipo de projeto de software.

A implantação de um processo de mensuração no desenvolvimento de software deve priorizar as medidas a serem coletadas devido ao custo associado a sua coleta [PEREIRA01]. Vários mecanismos podem ser utilizados para guiar o processo de seleção das medidas, destacando-se: o paradigma GQM (*Goal-Question-Metric*) [BASILI92], o GDSM [PARK96], o QFD (*Quality Function Deployment Approach*) [AKAO96] e o IEEE-1061 [IEEE98]. Em linhas gerais, o que essas propostas pregam é que a seleção das medidas deve tomar como base os objetivos que se pretende atingir com a mensuração.

O paradigma GQM seleciona as medidas desdobrando os objetivos de medição em questões quantificáveis e depois em medidas. Esse método é genérico o suficiente para ser utilizado para qualquer objetivo de medição. Porém, a definição de medidas para alguns objetivos, como por exemplo, produzir produtos com alta qualidade, depende da experiência em desdobrá-lo em questões quantificáveis. Dessa forma, para auxiliar o desdobramento de alguns objetivos da organização seria necessário o auxílio de um outro método para definir as medidas [LUIGI05].

O processo GDSM foi definido para desdobrar as metas da organização em indicadores e depois em objetivos de medição, que possam ser utilizados pelo método GQM. O GDSM define 10 passos para desdobrar as metas da organização até nas medidas para o processo de desenvolvimento. Porém, de acordo com o próprio GDSM, esses passos não são repetíveis e dessa forma, não garante que o processo chegue sempre ao mesmo resultado.

O padrão IEEE-1061 seleciona as medidas de qualidade através da identificação de fatores e subfatores de qualidade relacionados com o produto. A identificação dos fatores de qualidade mostra a preocupação de que a qualidade do produto possa ser influenciada por mais de uma característica de qualidade. Porém, o levantamento dos fatores de qualidade não é padronizado, dependendo da experiência do engenheiro de software.

O método QFD pode ser usado para desdobrar a qualidade do produto esperada pelo usuário, no processo de desenvolvimento do software, envolvendo os responsáveis por cada fase. Esse desdobramento auxilia identificar os fatores que influenciam a qualidade do produto, e também, como deve ser mensurada essa qualidade durante a construção do software. O método QFD pode ser aplicado de maneira formalizada e padronizada para qualquer produto, inclusive, produtos de software. Na construção do método proposta neste trabalho, o método QFD foi utilizado para desdobrar a qualidade esperada pelo usuário em medidas para o desenvolvimento de software.

As medidas selecionadas por um processo de definição são utilizadas para mensuração de entidades e atributos do software. O número de entidades envolvidas em software é grande devido à quantidade e à complexidade dos fatores envolvidos no desenvolvimento de um software de grande porte [FENTON96]. De acordo com Fenton [FENTON96], existem três classes de entidades envolvidas: processo, recurso e produto, mostradas na FIGURA 4. Essas classes são subdivididas em outras e o conjunto de todas essas classes forma o *Framework de Atributos*. Nesse mesmo trabalho, Fenton mostra como esse *framework* auxilia o processo GQM [BASILI92] na definição de medidas para as metas de medição da empresa.

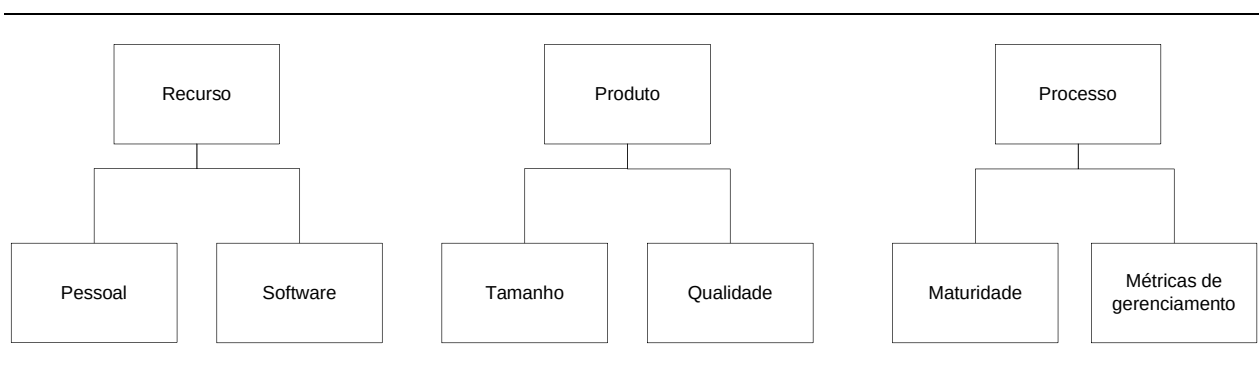


FIGURA 4 – Excerto do *Framework de Atributos* de Fenton [FENTON96]

O *Framework de Atributos* definido por Fenton foi adaptado, no Capítulo 5 deste trabalho, utilizando o modelo de qualidade definido na norma ISO-9126. A adaptação do *Framework de Atributos* de Fenton foi necessária para atualizar a entidade Produto de acordo com o modelo atual de qualidade da ISO-9126. Essa adaptação é utilizada para classificar as medidas encontradas pelo

método com objetivo de auxiliar a criação de um histórico de medidas de qualidade da organização. Esse histórico padronizado de acordo com o modelo de qualidade da norma ISO-9126 possibilita comparar a qualidade entre outros projetos.

2.4 Padronização da qualidade de software

Segundo Gilb [GILB96], já não é mais aceitável conceber projetos de engenharia de software, que lidem com alvos firmados em ambigüidades e abstrações. Por exemplo, não é mais aceitável definir a meta de um projeto como: ter um produto de boa qualidade. Dessa forma, é preciso definir de forma objetiva o termo ‘boa qualidade’ e para isso pode-se usar um modelo de qualidade de produto. Esses modelos padronizam as características de qualidade esperadas de um produto de software.

Um dos primeiros modelos a padronizar a qualidade de produto de software é o modelo de McCall [MCCALL77]. Esse modelo é referenciado pela maioria dos outros modelos de qualidade; ele propõe que os produtos de software sejam divididos em três áreas: operação, revisão e transição. A operação do produto refere-se à habilidade do produto de ser rapidamente entendido, eficientemente operado e capaz de prover resultados requeridos pelo usuário. As seguintes características de produto em operação são consideradas: modificabilidade, confiabilidade, eficiência, integridade e usabilidade. A revisão do produto está relacionada com a correção de erros e à adaptação do sistema. Esse aspecto é importante porque é geralmente considerada a parte mais cara do desenvolvimento de software. As seguintes características são cobertas em revisão: manutenibilidade, flexibilidade e testabilidade. A transição do produto pode não ser importante em todas as aplicações, contudo a tendência para processamento distribuído parece aumentar sua importância. As características desejadas na transição são: portabilidade, reusabilidade e interoperabilidade.

Uma das maiores contribuições do modelo de McCall é o relacionamento criado entre as características de qualidade e as métricas. Entretanto, nem todas as métricas do modelo são objetivas, podendo gerar dúvidas na sua aplicação. Um outro problema do modelo é que ele não considera como qualidade os aspectos funcionais do produto, como por exemplo, a adequação e acurácia das funções. Existem vários outros modelos de qualidade para o produto de software, como por exemplo, o modelo de Boehm [BOEHM78], de Ortega [MARYOLY03] dentre outros.

Diante desse cenário com vários modelos de qualidade, a ISO (*International Organization for Standardization*) define um modelo para unificar e padronizar a qualidade de produto através da publicação da norma ISO-9126 [ISO01]. O modelo da ISO-9126 define a qualidade do produto como um conjunto de características. As características que descrevem como o produto funciona no seu ambiente de desenvolvimento são chamadas de características de qualidade interna e externa. Essas características são subdivididas nas subcaracterísticas, descritas na TABELA 1.

QUALIDADE INTERNA E EXTERNA DE PRODUTO DE SOFTWARE – ISO 9126	
CARACTERÍSTICAS	DESCRIÇÃO
FUNCIONALIDADE: Refere-se à existência de um conjunto de funções que satisfazem necessidades explícitas ou implícitas e suas propriedades específicas.	Adequação: atributos de software que influenciam na adequação de um conjunto de funções para tarefas específicas e objetivos de uso; Acurácia: atributos do software que evidenciam a presença de resultados ou efeitos corretos ou conformidade acordada; Interoperabilidade: atributos de software que influenciam na habilidade de interagir com um ou mais sistemas específicos; Conformidade: atributos do software que influenciam na aderência e padrões relativos a convenções ou regulamentações legais e prescrições similares; Segurança de acesso: atributos de software que influenciam na habilidade de prevenir acessos não intencionados e resistir a ataques intencionados para se ter acesso não autorizado à informação confidencial, ou fazer modificações não autorizadas em informações ou em programa.
CONFIABILIDADE: Refere-se à capacidade do software manter seu nível de desempenho, sob condições estabelecidas, por um período de tempo.	Maturidade: atributos de software que influenciam na frequência de erros devido às falhas no software. Tolerância à falhas: atributos de software que influenciam na habilidade de um nível específico de desempenho em casos de falhas do software ou por violação de sua interface específica; Recuperabilidade: atributos de software que influenciam sua capacidade de restabelecer seu nível de desempenho e recuperar os dados diretamente afetados no caso de ocorrer uma falha e no tempo e esforços necessários. Conformidade: A capacidade do produto de software a aderir a padrões, convenções e regulamentações relacionadas com confiabilidade.
USABILIDADE: Refere-se ao esforço necessário ao uso do produto de software, bem como o julgamento individual de tal uso, por um conjunto explícito ou implícito de usuários.	Inteligibilidade: atributos de software que influenciam na capacidade do usuário entender se o software é adequado, e como ele pode ser usado para tarefas e condições de uso particulares; Apreensibilidade: atributos de software que influenciam a facilidade com a qual o usuário pode aprender suas aplicações; Operacionalidade: atributos de software que influenciam no esforço necessário para o usuário poder operar e manter o controle da operação. Atratividade: atributos de software que influenciam o interesse do usuário pelo uso do software. Conformidade: A capacidade do produto de software a aderir a padrões, convenções e regulamentações relacionadas com usabilidade.
EFICIÊNCIA: Refere-se ao relacionamento entre o nível de desempenho do software e a quantidade de recursos utilizada, sob condições estabelecidas.	Comportamento em relação ao tempo: atributos do software que influenciam no tempo de resposta de processamento e desempenho na execução de suas funções; Comportamento em relação aos recursos: atributos de software que influenciam na quantidade de recursos usados e a duração de tal uso na execução de suas funções. Conformidade: A capacidade do produto de software a aderir a padrões, convenções e regulamentações relacionadas com eficiência.
MANUTENIBILIDADE: Refere-se ao esforço necessário para fazer modificações específicas no software.	Analísabilidade: atributos de software que influenciam na necessidade de recursos para diagnóstico de deficiências ou causas de falhas, ou para identificação de partes a serem modificadas; Modificabilidade: atributos de software que influenciam na necessidade de recursos para implementar as modificações específicas; Estabilidade: atributos de software que influenciam nos riscos de efeitos inesperados das modificações;

	Testabilidade: atributos de software que influenciam na necessidade de recursos necessários para validação do software modificado. Conformidade: A capacidade do produto de software a aderir a padrões, convenções e regulamentações relacionadas com manutenibilidade.
PORTABILIDADE: Refere-se à habilidade do software para ser transferido de um ambiente para outro.	Adaptabilidade: atributos de software que influenciam na capacidade e esforço necessário para sua adaptação em ambientes diferentes especificados, sem aplicar outros meios ou ações para atingir o propósito do software; Capacidade de instalação: atributos de software que influenciam nos esforços necessários para instalar o software no ambiente especificado; Capacidade de substituição: atributos de software que proporcionam a oportunidade e influenciam no esforço requerido para usá-lo em lugar de outro software específico no ambiente de tal software; Coexistência: atributos de software que requerem esforços para existir no mesmo ambientes que outros softwares sem sofrer influências dos mesmos ou influenciá-los. Conformidade: A capacidade do produto de software a aderir a padrões, convenções e regulamentações relacionadas com portabilidade.

TABELA 1 – Características e subcaracterísticas de qualidade interna e externa da norma ISO-9126

Na norma ISO-9126, o modelo de qualidade interna e externa é utilizado para mensurar tanto os atributos externos quanto internos da qualidade de um produto de software.

As características relacionadas com o desempenho do produto no seu uso pelo usuário são chamadas de características de qualidade em uso, descritas na TABELA 2.

QUALIDADE EM USO DE PRODUTO DE SOFTWARE – ISO 9126	
CARACTERÍSTICAS	DESCRIÇÃO
EFETIVIDADE	Refere-se à capacidade do produto de software em possibilitar aos usuários atingir metas especificadas com acurácia e completeza, num contexto de uso especificado.
PRODUTIVIDADE	Refere-se à capacidade do produto de software em possibilitar aos usuários utilizar uma quantidade adequada de recursos em relação à efetividade alcançada, num contexto de uso especificado.
SEGURANÇA	Refere-se à capacidade do produto de software em propiciar níveis aceitáveis de risco de danos a pessoas, negócios, software, propriedade ou ambiente, num contexto de uso especificado.
SATISFAÇÃO	Refere-se à capacidade do produto de software em satisfazer usuários, num contexto de uso especificado.

TABELA 2 – Características de qualidade em uso da norma ISO-9126

Uma das vantagens do modelo de qualidade da norma ISO 9126 é que ele identifica as características de qualidade em uso, interna e externa. Entretanto, o modelo tem a desvantagem de não apresentar claramente como esses aspectos podem ser mensurados para alcançar a qualidade esperada pelo usuário.

Para resolver esse problema a ISO definiu a norma ISO 14598 [Iso99] que demonstra o processo de avaliação de um produto de software. Essa norma é dividida em três processos de

avaliação: aquisição, avaliação e desenvolvimento. Cada parte da norma descreve os passos para avaliar a qualidade do produto e utiliza como referência o modelo de qualidade da norma ISO 9126. Porém, o processo de avaliação de qualidade no desenvolvimento, descrito na norma ISO 14598, não demonstra como desdobrar a qualidade esperada pelo usuário em medidas nem como priorizá-las.

2.5 Conceitos do processo Praxis 2.1 e Merci 1.0

O Praxis (PRocesso para Aplicativos eXtensíveis e InterativoS) constitui um processo de desenvolvimento de software desenhado para projetos com duração de seis meses a um ano, realizados individualmente ou por pequenas equipes [PAULA03]. É voltado para o desenvolvimento de aplicativos gráficos interativos, baseados na tecnologia orientada a objetos.

Seguindo a arquitetura definida no Processo Unificado [JACOBSON98], o Praxis propõe um ciclo de vida composto por fases, divididas em uma ou mais iterações, e fluxos, que correspondem a disciplinas da Engenharia de Software. A TABELA 3 apresenta a divisão em fases e iterações, enquanto a organização em fluxos é descrita na TABELA 4.

Fase	Iteração	Sigla	Descrição
Concepção	Ativação	AT	Levantamento e análise das necessidades dos usuários e conceitos da aplicação, em nível de detalhe suficiente para justificar a especificação de um produto de software.
	Levantamento dos Requisitos	LR	Levantamento das funções, interfaces e requisitos não funcionais desejados para o produto em nível de detalhe suficiente para determinar a viabilidade de desenvolvimento de um produto.
Elaboração	Análise dos Requisitos	AR	Modelagem conceitual dos elementos relevantes do domínio do problema, validação dos requisitos e definição da arquitetura, em detalhe suficiente para o planejamento da fase de Construção.
	Desenho Implementável	DI	Implementação de um subconjunto crítico de funções do produto, para validar a arquitetura e tecnologias escolhidas, determinando a produtividade de desenvolvimento.
Construção	Liberação 1	L1	Implementação de um subconjunto de funções do produto que será avaliado pelos usuários.
	Liberação ...	Ln	Idem.
	Testes Alfa	TA	Realização dos testes de aceitação, no ambiente dos desenvolvedores, juntamente com elaboração da documentação de usuário e possíveis planos de Transição.
Transição	Testes Beta	TB	Realização dos testes de aceitação, no ambiente dos usuários.
	Operação Piloto	OP	Operação experimental do produto em instalação piloto do cliente, com a resolução de eventuais problemas através de processo de manutenção.

TABELA 3 – Fases e iterações do Praxis 2.1 [PAULA03]

Natureza	Fluxo		Descrição
Técnicos	Requisitos		Fluxo que visa a obter um conjunto de requisitos de um produto, acordado entre cliente e fornecedor.
	Análise		Fluxo que visa a detalhar, estruturar e validar os requisitos, em termos de um modelo conceitual do problema, de forma que estes possam ser usados como base para o planejamento e acompanhamento detalhados da construção do produto.
	Desenho		Fluxo que visa a formular um modelo estrutural do produto que sirva de base para a implementação, definindo os componentes a desenvolver e a reutilizar, assim como as interfaces entre si e com o contexto do produto.
	Implementação		Fluxo que visa a detalhar e implementar o desenho através de componentes de código e de documentação associada.
	Testes		Fluxo que visa a verificar os resultados da implementação, através do planejamento, desenho e realização de baterias de testes.
	Engenharia de sistemas		Fluxo que abrange atividades relativas ao desenvolvimento do sistema no qual o produto de software está contido; por exemplo, modelagem de processos de negócio, implantação, usabilidade e criação de conteúdo.
	Fluxo	Subfluxo	Descrição
Gerenciais	Gestão de projetos	Gestão de requisitos	Controle das alterações e rastreamento dos requisitos.
		Planejamento de projetos	Elaboração de planos de projetos, por meio de estimativas de tamanho, esforço, prazo e riscos.
		Controle de projetos	Acompanhamento do progresso e dos riscos dos projetos, com execução de procedimentos corretivos, quando necessários.
		Contratação de projetos	Especificação e acompanhamento de contratos de desenvolvimento por parte de terceiros.
	Gestão da qualidade	Garantia da qualidade	Conjunto planejado e sistemático de ações necessárias para estabelecer um nível adequado de confiança na qualidade de um produto.
		Gestão de revisões	Planejamento, convocação e avaliação de revisões técnicas e inspeções.
		Gestão de configurações	Conjunto de procedimentos técnicos e gerenciais para identificação de artefatos e gestão de alterações destes.
		Gestão da manutenção	Conjunto de procedimentos para a manutenção dos produtos em Transição e Produção.
	Engenharia de processos	Gestão de processos	Guarda, manutenção e personalização do patrimônio de processos da organização.
		Gestão do treinamento	Gestão das atividades de treinamento relacionadas com processos de software.
		Melhoria de tecnologia	Execução das atividades de evolução tecnológica relacionadas com processos de software.
		Melhoria de processos	Aferição, controle quantitativo e evolução dos processos de software.

TABELA 4 – Fluxos técnicos e gerenciais do Praxis 2.1 [PAULA03]

As atividades desenvolvidas pelos fluxos técnicos e gerenciais do Praxis geram e consomem artefatos, descritos na TABELA 5.

Espécie	Sigla	Significado
Modelos	CRSw	Cadastro dos requisitos
	MASw	Modelo de análise
	MPPSw	Memória de planejamento de projeto
	MDSw	Modelo de desenho
	BTRSw	Bateria de testes de regressão
	CFSw	Código fonte
	CESw	Código executável
Relatórios	LCSw	Listas de conferência
	RRSw	Relatórios de revisão
	RTSw	Relatório dos testes
	RAPSw	Relatório de acompanhamento de projeto
	RAQSw	Relatório de auditoria da qualidade
Documentos para o cliente	PESw	Proposta de especificação
	ERSw	Especificação dos requisitos
	PDSw	Plano de desenvolvimento
	PQSw	Plano da qualidade
	DDSw	Descrição do desenho
	DTSw	Descrição dos testes
	MUSw	Manual do usuário

TABELA 5 – Artefatos do Praxis 2.1 [PAULA03]

Apesar de ter sido originalmente destinado à utilização em projetos didáticos em disciplinas de Engenharia de Software, o processo vem sendo usado com sucesso também em projetos maiores, alguns envolvendo equipes de até dezenas de pessoas.

O produto Merci 1.0 é utilizado como um exemplo de referência no material de ensino do Praxis, assim sendo, ele contém os artefatos técnicos e gerenciais, produzidos em cada iteração do processo, utilizados para aplicar e analisar o método proposto. O Merci está disponível para baixar no site [PAULA06] do autor do processo Praxis.

Capítulo 3

Metodologia de pesquisa

Este capítulo descreve e classifica o tipo de pesquisa utilizada neste trabalho e apresenta a metodologia de pesquisa utilizada. Um leitor atento ao método e em busca a uma forma de reproduzir ou dar continuidade à investigação, deve encontrar de forma clara o processo metodológico utilizado no desenvolvimento de uma pesquisa. Desta forma este capítulo tem por objetivo apresentar a metodologia utilizada no desenvolvimento do trabalho.

3.1 Tipo de pesquisa

De acordo com Jung [JUNG04], um trabalho de pesquisa pode ser classificado quanto a sua natureza (básica ou fundamental / aplicada ou tecnológica), quanto aos seus objetivos (exploratória / descritiva ou / explicativa) e quanto aos procedimentos (experimental / operacional / estudo de caso). Este trabalho de pesquisa é de natureza aplicada ou tecnológica, com objetivos de caráter explicativos, utilizando procedimentos de pesquisa experimental.

O trabalho é uma pesquisa aplicada ou tecnológica, pois objetiva a aplicação de conhecimentos básicos na geração de novos produtos. Nesse contexto, o trabalho utiliza o método QFD e a norma ISO-9126 com intuito de criar um novo método para a aferição da qualidade de produtos nos processos de desenvolvimento de software.

O método proposta neste trabalho é de caráter explicativo, pois tenta ampliar a utilização do método QFD na área de desenvolvimento de software. O método QFD já foi utilizado para o levantamento de requisitos do produto de software, por exemplo, o SQFD [ZULTNER90], porém pouco utilizado na definição de medidas para acompanhar a qualidade durante o desenvolvimento do software.

Este trabalho utiliza-se de procedimento experimental para definir um método para mensurar a qualidade esperada pelo usuário durante o desenvolvimento de software.

3.2 Procedimentos metodológicos

No primeiro ano deste trabalho, foram realizadas algumas disciplinas que serviram de base para a criação do método de aferição da qualidade de software. Duas dessas matérias foram realizadas no Departamento de Produção da UFMG, sendo elas: Sistema de Desenvolvimento de Produtos (SDP) e Gestão da Qualidade Industrial (GQI).

Na matéria de SDP, foram apresentados vários conceitos relacionados com o desenvolvimento de produtos de engenharia. Dentre esses conceitos, o método QFD se destacou para formalizar o método proposto neste trabalho.

A matéria GQI foi importante no ensino de vários conceitos gerais sobre a qualidade. Nesta disciplina também foram estudados alguns aspectos do processo de garantia de qualidade utilizados até hoje, como por exemplo, QFD [AKAO96], PDCA (*Plan-Do-Check-Act*), Six Sigma dentre outros.

Na área de qualidade da engenharia de software. Para a definição do trabalho foi importante a experiência adquirida nos projetos de desenvolvimento com clientes externos, desenvolvidos no laboratório Synergia do Departamento de Ciência da Computação. Nesse laboratório, foram vivenciados vários aspectos da qualidade de um produto de software, como por exemplo, o processo de testes e também pode ter contato direto com o processo Praxis.

Depois de concluídas as disciplinas foi realizada uma busca intensa para definição do tema da dissertação. Nessa busca foram estudados alguns processos de seleção de medidas, como por exemplo, IEEE-1061, GQM e GDSM. Esses processos auxiliaram na definição do escopo do trabalho.

Também foram estudados vários trabalhos de definição de medidas para qualidade. Nesses trabalhos foi identificada a necessidade de uma forma de organizá-las. A classificação e seleção de medidas foram realizadas no segundo semestre de 2005 e encontram-se descritas no Capítulo 5 deste trabalho.

A construção do método proposto iniciou-se no período de final de 2005 e alcançou o início de 2006, com auxílio do Márcio, aluno de mestrado do departamento de produção da UFMG. O método de aferição da qualidade de produto foi instanciada para o processo Praxis 2.1 no mês de Janeiro deste ano. Nessa instanciación foram personalizados algumas atividades e artefatos do processo Praxis.

Então, o método foi aplicado ao produto Merci 1.0 no período de Fevereiro/2006 a Abril/2006. Para realizar essa aplicação foram utilizadas algumas planilhas automatizadas no aplicativo Excel 2003 ®. Os resultados obtidos foram analisados nos meses de Maio e Junho de 2006, descritos na seção 6.5 deste trabalho.

Capítulo 4

Proposta de metodologia para aferição da qualidade de produtos de software

De acordo com a norma ISO-14598 [Iso99], o termo modelo de qualidade é definido como “o conjunto de características e as relações entre elas, que provê a base para especificar os requisitos de qualidade e avaliar a qualidade”. Metodologias baseadas em modelos de qualidade são amplamente utilizadas para, antecipadamente, avaliar e controlar a qualidade de software. Esta seção propõe um método para aferição da qualidade de produtos de software baseada no modelo de qualidade da norma ISO-9126 e no método QFD.

O método proposto neste trabalho utiliza-se do método QFD para o mapeamento da qualidade do produto em medidas utilizadas para o desenvolvimento do software. O desdobramento da qualidade esperada pelo usuário foi definido através da construção de um *modelo conceitual* para aferição de qualidade de produto de software. O *modelo conceitual* utiliza quatro matrizes: Matriz de Qualidade do Usuário, Matriz de Qualidade em Uso, Matriz de Qualidade Externa e Matriz de Qualidade Interna. As matrizes são utilizadas para mapear a qualidade esperada pelo usuário em medidas para o controle e o acompanhamento dessa qualidade durante o desenvolvimento do software.

O método proposto também utiliza o modelo de qualidade definida na norma ISO-9126 para transferir uma padronização à qualidade do produto esperada pelo usuário. A norma ISO-9126 divide a qualidade do produto de software em dois modelos: qualidade interna e externa, e qualidade em uso, utilizados na construção do *modelo conceitual*. De acordo com a norma ISO-9126, a qualidade interna e externa possuem as mesmas características de qualidade, porém a qualidade interna é mensurada através dos resultados intermediários do produto e a qualidade externa é mensurada através da execução do seu código.

A próxima seção apresenta os aspectos da norma ISO-9126 e do método QFD que foram utilizados na construção do método de aferição de qualidade.

4.1 Importância da norma ISO-9126 e do método QFD na construção do método

O modelo de qualidade definido na norma ISO-9126 [ISO01] pode ser aplicado para todo tipo de produto de software. O modelo é dividido em duas partes: qualidade interna e qualidade externa e qualidade em uso. A qualidade interna representa a qualidade de produtos intermediários do desenvolvimento de software e pode ser um indicativo da qualidade externa do produto. A qualidade externa é a qualidade do produto em execução, por exemplo, o produto sendo executado na fase de testes de aceitação e pode ser um indicativo da qualidade em uso. A qualidade em uso representa a qualidade que o usuário percebe em determinados contextos de uso do produto [ISO01].

Na literatura existem alguns modelos de qualidade de produto de software como os trabalhos de McCall [MCCALL77], Boehm [BOEHM78], FURPS [GRADY87], Dromey [DROMEY96] e ISO-9126 [ISO01]. O modelo de qualidade da norma ISO-9126 foi escolhido para este trabalho por ser um padrão estabelecido, pela qualidade e clareza em sua definição e sua abrangência, pois se trata de um padrão internacional. Contudo, outro modelo, possivelmente algum já definido em um processo de desenvolvimento de software, pode ser utilizado, desde que apresente categorias de qualidade bem definidas.

A qualidade interna, coletada durante o desenvolvimento, pode fornecer uma indicação da qualidade externa esperada do produto. Contudo, as medidas de qualidade interna, para serem utilizadas como indicativo da qualidade externa, devem ser relacionadas com a qualidade externa do produto [ISO01], mostrado na FIGURA 5. Esse relacionamento da qualidade externa com a qualidade interna é realizado no método definido neste trabalho através do método QFD [AKAO96].

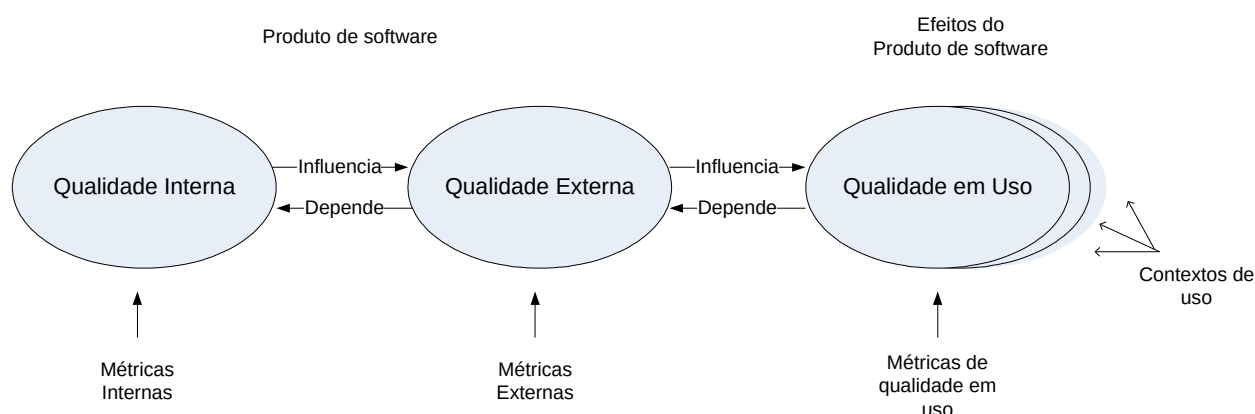


FIGURA 5 – Relacionamento dos tipos de métricas de acordo com a norma ISO-9126

O método QFD foi escolhido para guiar o desdobramento da qualidade de software, por se tratar de um método consolidado, flexível e comprovadamente eficiente para desdobrar a qualidade

exigida pelo usuário nos processos de desenvolvimento de produtos [CHENG95]. Este método também vem sendo utilizado com grande sucesso na área de software [SHINDO99], mesmo que não ainda de forma completamente integrada ao processo de desenvolvimento das empresas.

De acordo com Cheng [CHENG95], o método QFD foi criado para auxiliar o processo de gestão de desenvolvimento do produto, denominado ação gerencial do planejamento da qualidade. Também segundo Cheng [CHENG95], o método QFD pode ser aplicado tanto a produto (entendido como bens ou serviços) da empresa quanto a produto intermediário entre cliente e fornecedor interno. A implantação do método QFD objetiva duas finalidades específicas; auxiliar o processo de desenvolvimento do produto, buscando, traduzindo e transmitindo as necessidades e desejos do cliente; garantir qualidade durante o processo de desenvolvimento do produto. Dessa forma, o método QFD se mostra adequado no planejamento e controle da qualidade no desenvolvimento de software.

Na seção seguinte será apresentada, em detalhes, o método desenvolvido neste trabalho para aferir a qualidade durante o desenvolvimento de um produto de software.

4.2 Metodologia para aferição da qualidade de produtos de software

O método para aferição da qualidade de produtos de software tem o objetivo de prover medidas para mensuração da qualidade durante o desenvolvimento do software, por meio do desdobramento da qualidade esperada pelo usuário. A mensuração da qualidade durante o desenvolvimento do produto, antecipa a descoberta de falhas e quanto mais cedo elas são identificadas, menor o custo de sua correção [NIST02]. Através da correção dessas falhas, espera-se que o desenvolvimento do produto possa obter a qualidade esperada pelo usuário, contribuindo para o custo e o prazo sob controle.

O método tem por objetivo também auxiliar a padronização da qualidade levantada com o usuário de acordo com a norma ISO-9126. Essa padronização auxilia a criação de uma base histórica de projetos da empresa, visando à reutilização de conhecimento, como por exemplo, medidas, soluções técnicas, arquitetura dentre outros. A reuso de conhecimento é um passo em direção a melhoria contínua da qualidade do produto e também do aumento da produtividade nas atividades do processo de desenvolvimento.

A FIGURA 6 apresenta um *modelo conceitual*, conforme proposto na metodologia QFD, que utiliza quatro matrizes correlacionadas, sendo elas, respectivamente, Matriz de Qualidade do Usuário, Matriz de Qualidade em Uso, Matriz de Qualidade Externa e Matriz de Qualidade Interna. O modelo é utilizado para desdobrar a qualidade esperada pelo usuário em medidas e também para conferir uma padronização aos atributos de qualidade de acordo com a norma ISO-9126.

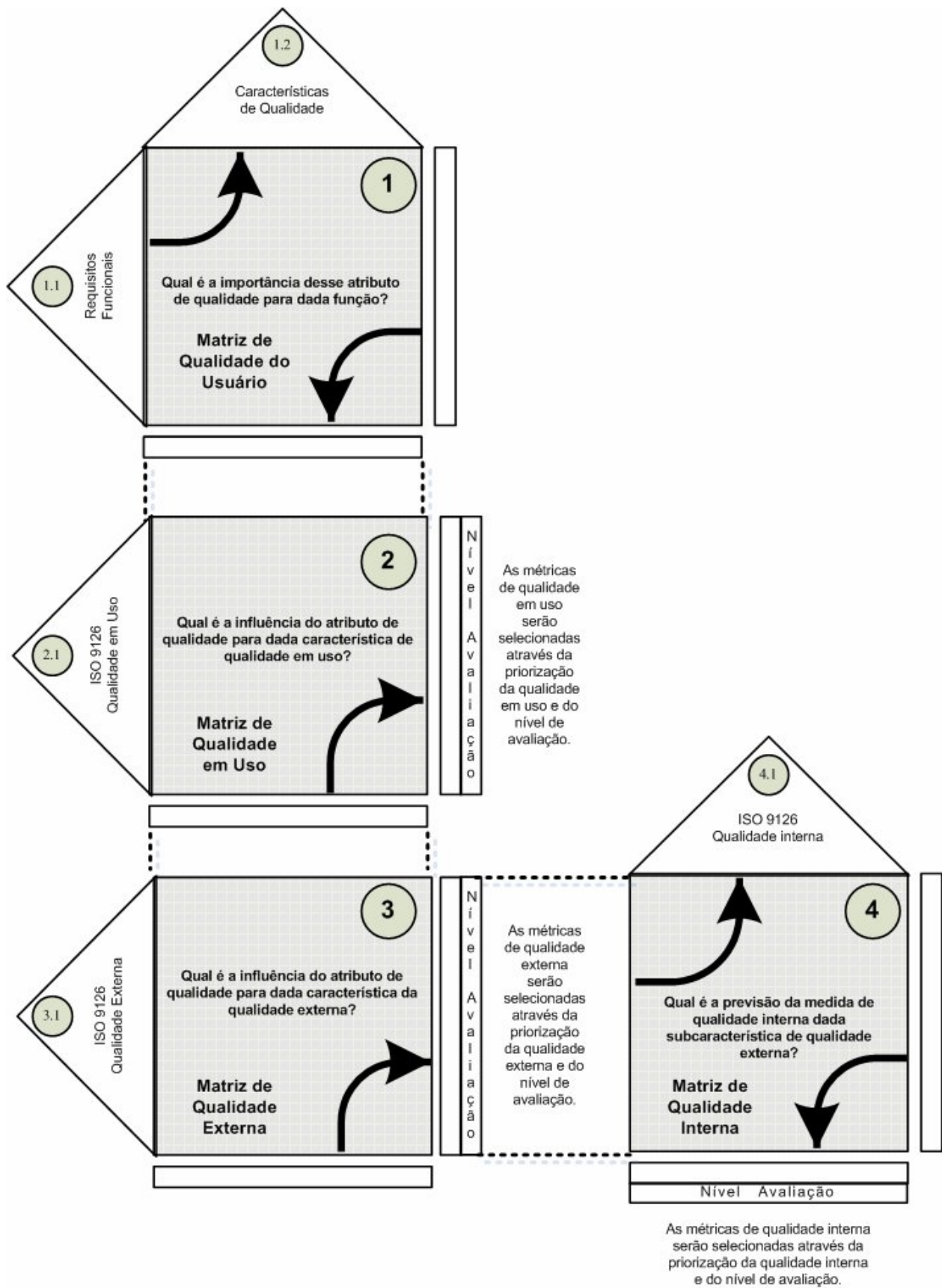


FIGURA 6 – Modelo conceitual para aferição de qualidade para produtos de software

A Matriz de Qualidade do Usuário (matriz 1) correlaciona as funções do produto com os atributos de qualidade esperados pelo usuário. Os atributos de qualidade são também conhecidos como requisitos não-funcionais na área de Engenharia de Software. O objetivo da matriz de qualidade do usuário é determinar, desde o início do projeto, os atributos de qualidade mais importantes para o usuário, o que é feito através de sua correlação com as funções do produto. A Matriz de Qualidade em Uso (matriz 2) correlaciona os atributos de qualidade esperados pelo usuário com as características de qualidade em uso da norma ISO-9126. O objetivo dessa atividade é priorizar as características de qualidade em uso e transferir a padronização de qualidade em uso do produto de software aos atributos de qualidade, do ponto de vista do usuário. A Matriz de Qualidade Externa (matriz 3) correlaciona os atributos de qualidade esperados pelo usuário com as características de qualidade externa da norma ISO-9126, visando à transferência da padronização da qualidade externa do produto de software aos atributos de qualidade levantados com o usuário. A Matriz de Qualidade Interna (matriz 4) correlaciona as características de qualidade externa da ISO-9126 com as medidas de qualidade interna do desenvolvimento do produto. O objetivo dessa matriz é priorizar as medidas internas que possam auxiliar na previsão da qualidade externa do produto.

O encadeamento das matrizes, acima citadas, define o *modelo conceitual* de aferição de qualidade proposto neste trabalho. Esse *modelo conceitual* irá guiar o desdobramento da qualidade esperada pelo usuário auxiliando a definição de medidas para o desenvolvimento do produto de software. Além da definição de medidas, o desdobramento da qualidade pode auxiliar no planejamento das atividades de construção do produto de software.

A definição das medidas para cada característica de qualidade, definida na norma ISO-9126, é auxiliada pelo seu Nível de Avaliação. Cada característica de qualidade é priorizada de acordo com a qualidade esperada pelo usuário e pode ser mensurada através da coleta de medidas durante o desenvolvimento do produto. O Nível de Avaliação foi combinado com o *modelo conceitual* proposto para permitir uma melhor priorização das medidas da qualidade.

As próximas seções apresentam o conceito de Nível de Avaliação utilizado para seleção de medidas no método, o processo de construção das matrizes e o relacionamento entre essas matrizes, visando à criação do *modelo conceitual* de aferição de qualidade.

4.2.1 Nível de Avaliação

O método proposto utiliza o Nível de Avaliação da norma ISO-14598 em três matrizes: Qualidade em Uso, Qualidade Externa e Qualidade Interna, com o objetivo de auxiliar a priorização das medidas para o processo de desenvolvimento. A seleção das medidas nessas matrizes considera o grau de importância, calculado através do *modelo conceitual*, e o Nível de Avaliação de cada característica de qualidade da norma ISO-9126.

De acordo com a norma ISO-14598 [Iso99], o Nível de Avaliação está relacionado com a importância dada para uma característica de qualidade da norma ISO-9126. A importância de cada característica influencia a profundidade e completude da sua avaliação. As técnicas de avaliação são utilizadas para coletar as medidas relacionadas com as características de qualidade do modelo da

norma ISO-9126.

A norma ISO-14598 propõe quatro níveis de avaliação, designados: A, B, C e D. Esses níveis constituem uma hierarquia, sendo A o nível mais alto e D o mais baixo. No nível A, as técnicas de avaliação mais estritas (levando em consideração uma quantidade razoável de esforço e tempo) são aplicadas dando uma maior confiança nas medidas coletadas. Até o nível D, gradualmente, técnicas menos estritas são utilizadas, e conseqüentemente, menos esforço é dedicado para a avaliação das medidas.

O Nível de Avaliação é selecionado conforme a conseqüência de um erro no software para os aspectos pertinentes ao seu uso. Para selecionar o Nível de Avaliação de cada característica de qualidade é preciso considerar vários aspectos, como por exemplo, aspecto de segurança humana, econômico, segurança da informação, ambiente ou marketing. A norma ISO-14598 sugere que a seleção dos níveis de avaliação das características de qualidade da norma ISO-9126 considere alguns aspectos de uso do software como segurança humana, econômico, segurança da informação e ambiente. O relacionamento do Nível de Avaliação com esses aspectos é mostrado a seguir.

Para o aspecto de segurança humana, os seguintes níveis de avaliação podem ser selecionados conforme a conseqüência de uma não conformidade no software, mostrados na TABELA 6.

Nível de Avaliação	Conseqüências
Nível D	Pequeno dano à propriedade; nenhum risco para população.
Nível C	Dano à propriedade; ameaça de ferimento para população.
Nível B	Risco para vidas humanas.
Nível A	Muitas pessoas mortas.

TABELA 6 – Níveis de avaliação para o aspecto de segurança humana

Às vezes, um erro encontrado depois que o produto já foi entregue pode ter uma grande conseqüência econômica. Por exemplo, um erro de cálculo em um software bancário pode trazer um prejuízo incalculável para o banco. Porém, por outro lado um mesmo erro em um software de uma pequena mercearia não causaria problemas, ou até mesmo passaria sem ser percebido. No aspecto econômico, os níveis de avaliação, mostrados na TABELA 7, podem ser selecionados conforme a conseqüência de uma não-conformidade no produto.

Nível de Avaliação	Conseqüências
Nível D	Insignificante perda econômica.
Nível C	Significante perda econômica. (Empresa afetada)
Nível B	Grande perda econômica. (Empresa posta em perigo)
Nível A	Desastre financeiro. (Empresa não irá sobreviver)

TABELA 7 – Níveis de avaliação para o aspecto econômico

Para o aspecto de segurança da informação deve ser avaliada a conseqüência de uma falha no produto diante do risco da perda ou violação dos dados da empresa. Os níveis de avaliação

descritos na TABELA 8 podem ser considerados para essa avaliação.

Nível de Avaliação	Consequências
Nível D	Nenhum risco específico identificado.
Nível C	Proteção diante o risco de erro.
Nível B	Proteção de dados e serviços críticos.
Nível A	Proteção de dados e serviços estratégicos.

TABELA 8 – Níveis de avaliação para o aspecto de segurança da informação

Outro aspecto a ser considerado para o Nível de Avaliação é o aspecto ambiental. Deve ser considerado o impacto ambiental que um erro no produto de software pode causar. Os níveis e consequências para o aspecto ambiental estão descritos na TABELA 9.

Nível de Avaliação	Consequências
Nível D	Nenhum risco ambiental.
Nível C	Poluição local.
Nível B	Dano ambiental recuperável.
Nível A	Dano ambiental irrecuperável.

TABELA 9 – Níveis de avaliação para o aspecto ambiental

A seleção do Nível de Avaliação deve considerar o nível mais estrito de cada aspecto quando são necessários vários aspectos para o software. Por exemplo, o nível A deve ser selecionado para uma característica de qualidade, caso esse nível tenha sido selecionado para o aspecto ambiental, mesmo que o nível D tenha sido selecionado para outros aspectos.

De acordo com a norma ISO-14598 [Iso99], as técnicas de avaliação devem ser escolhidas de acordo com a característica de qualidade da norma ISO-9126 e o Nível de Avaliação. Como exemplo, a técnica de avaliação utilizada para característica de qualidade Funcionalidade pode ser teste de caixa-preta, já para Usabilidade, outra característica de qualidade, pode ser avaliação com o usuário. O Nível de Avaliação vai determinar o critério na avaliação de cada característica de qualidade. Por exemplo, se o Nível de Avaliação para característica Funcionalidade for A, a técnica de avaliação seria teste de unidade com critério de cobertura, e se for C, para a mesma característica de qualidade, a técnica de avaliação seria testes funcionais. Isso acontece, porque o Nível de Avaliação C é menos exigente na avaliação da característica de qualidade.

Com as técnicas de avaliação é possível avaliar a qualidade do produto, para cada característica de qualidade, através da coleta de medidas. A qualidade do produto, de acordo com a norma ISO-9126, é categorizada em características de qualidade bem distintas, que podem ser avaliadas de forma independente.

No método proposto neste trabalho, o Nível de Avaliação é utilizado para ajustar o grau de importância, utilizado para priorizar as medidas para a Matriz de Qualidade em Uso, da Matriz de Qualidade Externa e da Matriz de Qualidade Interna. O ajuste do grau de importância é feito através dos pesos associados aos Níveis de Avaliações, sendo que os pesos considerados para o nível A, B, C

e D são, respectivamente, 4, 3, 2 e 1. Os pesos dos níveis de avaliação são valores sugeridos, que podem ser alterados de acordo com a percepção dos envolvidos. Dessa forma, as características de qualidade que, por exemplo, estão associadas ao nível A tem seu grau de importância aumentado.

A unificação do Nível de Avaliação e o grau de importância das características, obtida nas matrizes, é apresentada a seguir.

4.2.2 Matriz de Qualidade do Usuário

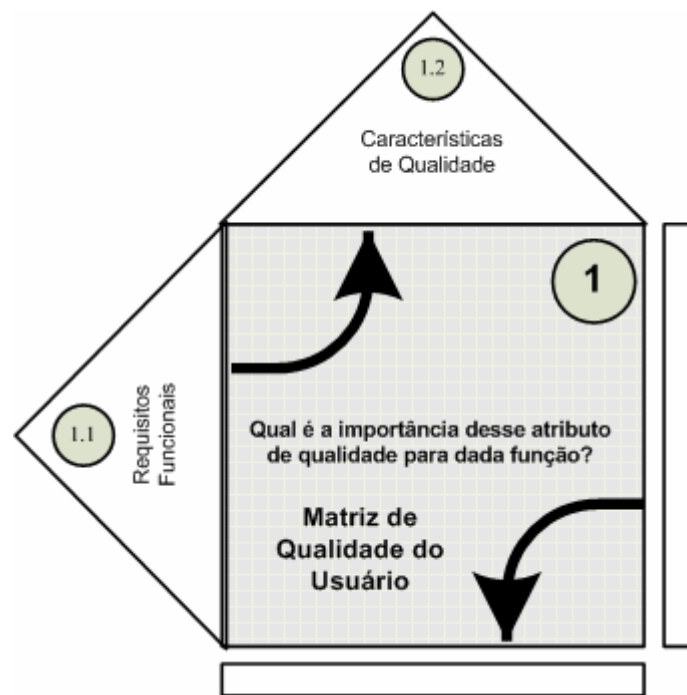


FIGURA 7 – Matriz de Qualidade do Usuário

A Matriz de Qualidade do Usuário, mostrada na FIGURA 7, relaciona as funções do produto e os atributos de qualidade esperados pelo cliente. As funções do produto são levantadas juntamente com o usuário e registradas na Tabela de Requisitos Funcionais, tabela 1.1 da matriz. Os atributos de qualidade esperados pelo cliente são extraídos também do usuário e registrados na matriz de características de qualidade, tabela 1.2 da matriz. As duas tabelas são relacionadas, para a construção dessa matriz, com o objetivo de priorizar os atributos de qualidade de acordo com o entendimento do usuário do produto.

A Tabela de Requisitos Funcionais é formada por uma lista priorizada das funções do produto de software. Para identificar as funções do produto com o usuário podem ser utilizadas algumas das técnicas conhecidas para o levantamento de requisitos tradicional, como por exemplo, a técnica JAD (*Joint Application Design*) [MCCONNELL96], a própria técnica QFD, o método SQFD dentre outros. Com as funções identificadas é preciso priorizá-las conforme a importância que essas representam para o usuário em relação ao produto.

Para alguns métodos de levantamento de requisitos como, por exemplo, SFQD, a priorização das funções já faz parte do seu processo. O usuário é responsável por priorizar as funções críticas para o produto de software. Essa priorização deve seguir critérios, como por exemplo, as funções mais utilizadas ou àquelas que representam uma atividade importante para o usuário. Cada função deve receber um valor numérico que representa o seu grau de importância em relação às outras funções.

A Tabela de Características de Qualidade é formada pelos atributos de qualidade que o usuário espera do produto de software. Esses atributos são obtidos usando um processo de *extração*, exemplificado na seção 2.1 deste trabalho, das expectativas do usuário a partir das funções listadas na tabela de requisitos funcionais. A expectativa do usuário expressa nas Características de Qualidade devem buscar contemplar suas necessidades intrínsecas, observar seus valores, visando maximizar a satisfação do usuário [KANO96]. O termo “usuário” neste contexto representa não somente os usuários diretos do produto de software, mas também todos os interessados no produto (em inglês, *stakeholders*), incluindo representantes da organização cliente. Para extrair as necessidades do usuário podem ser usadas as técnicas como: 5W1H, entrevistas com usuário ou questionários de satisfação de versões anteriores.

Depois da construção da Tabela de Requisitos Funcionais e da Tabela de Características de Qualidade, o relacionamento dessas permitirá priorizar os atributos de qualidade esperados pelo usuário. O cliente do produto é consultado para realizar a *correlação* dessas tabelas da Matriz de Qualidade do Usuário através da pergunta “Qual é a importância desse atributo de qualidade para dada função?”.

A *correlação* da Tabela de Requisitos Funcionais com a Tabela de Características de Qualidade na Matriz de Qualidade do Usuário pode trazer benefícios ao desenho da solução do produto. O primeiro benefício é que o relacionamento possibilita desenhar as funções do produto conforme os atributos de qualidade prioritários. Cada função é correlacionada com os atributos de qualidade e, dessa forma, o arquiteto do software pode desenhar uma solução mais adequada às expectativas de qualidade do usuário. O segundo benefício é priorizar os atributos de qualidade que mais preocupam os usuários, e assim, os programadores podem desenvolver o produto focando nos atributos prioritários.

Com a *conversão* realizada na Matriz de Qualidade do Usuário os atributos de qualidade esperados pelo cliente são priorizados, sendo que eles serão utilizados para construção da Matriz de Qualidade em uso e da Matriz de Qualidade Externa, as próximas matrizes do *modelo conceitual*.

4.2.3 Matriz de Qualidade em Uso

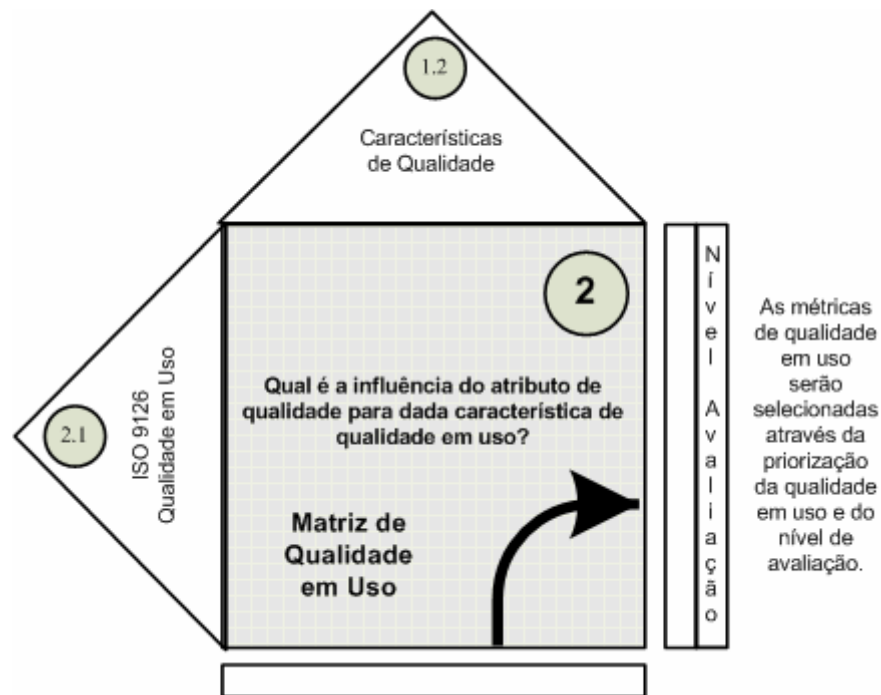


FIGURA 8 – Matriz de Qualidade em Uso

A Matriz de Qualidade em Uso do *modelo conceitual*, exibida na FIGURA 8, tem por objetivo conferir padronização aos atributos de qualidade esperados pelo usuário de acordo com o modelo de qualidade em uso definido na norma ISO-9126. Os atributos de qualidade utilizados na Matriz de Qualidade em Uso são provenientes da Tabela de Características de Qualidade, tabela 1.2, da Matriz de Qualidade do Usuário. O modelo de qualidade em uso define um conjunto de características de qualidade, sendo utilizado para formar a Tabela de Qualidade em Uso, tabela 2.1. As informações dessas tabelas são correlacionadas pelo engenheiro de software para construir a Matriz de Qualidade em Uso do *modelo conceitual*.

A Tabela de Características de Qualidade contém os atributos de qualidade levantados e priorizados na construção da Matriz de Qualidade do Usuário, ilustrada na FIGURA 7. Esses atributos foram priorizados através do relacionamento com as funções do produto e receberam seus graus de importância. Essa mesma priorização é utilizada para calcular o grau de importância da Tabela de Qualidade em Uso.

A utilização da Matriz de Qualidade em Uso nesta parte do modelo possibilita avaliar a capacidade do produto em alcançar os objetivos dos usuários em determinados contextos de uso. O modelo definido na norma categoriza a qualidade em uso, em quatro características: eficácia, produtividade, segurança e satisfação. Essas características são priorizadas através do relacionamento com os atributos de qualidade da Tabela de Características de Qualidade.

As informações da Tabela de Características de Qualidade e da Tabela de Qualidade em

Uso são correlacionadas para construção da Matriz de Qualidade em Uso. A *correlação* dessas tabelas é realizada pelo engenheiro de software nessa matriz através da pergunta “Qual é a influência do atributo de qualidade para dada característica de qualidade em uso?”.

Depois do relacionamento das características de qualidade em uso e os atributos de qualidade na Matriz de Qualidade em Uso, o grau de importância de cada característica de qualidade em uso é calculado através da *conversão*. Cabe observar que essa *conversão* nessa matriz é realizada no sentido horizontal, diferente da primeira matriz do *modelo conceitual*.

O relacionamento realizado na Matriz de Qualidade em Uso auxilia na priorização das medidas para as características de qualidade definidas na norma ISO-9126. De acordo com o grau de importância e o Nível de Avaliação das características de qualidade em uso, o engenheiro de software pode selecionar as medidas de qualidade adequadas para a qualidade esperada pelo usuário.

Essa seleção das medidas deve considerar quais características prioritárias para usuário, porém não deve deixar de selecionar medidas para outras características. Com essa seleção das medidas, realizada na Matriz de Qualidade em Uso, o engenheiro de software consegue mensurar a qualidade em determinados contextos de uso do produto.

Além da priorização das medidas de qualidade em uso, a *correlação* realizada na Matriz de Qualidade em Uso proporciona padronização aos atributos de qualidade esperados pelo usuário de acordo com a norma ISO-9126. Os atributos de qualidade são levantados com o usuário na sua linguagem própria, dificultando comparações com os atributos de qualidade de outros produtos desenvolvidos. A padronização da qualidade de acordo com a norma possibilita a empresa construir uma base histórica de projetos. A construção dessa base histórica possibilita aproveitar e até reutilizar alguns conhecimentos, como medidas, listas de conferência, soluções de desenho dentre outros.

Essa base histórica também pode ser utilizada para comparar a qualidade de um produto com outros. A comparação entre projetos permite a empresa alcançar uma melhoria contínua na qualidade dos seus produtos. A melhoria dos produtos mantém os clientes satisfeitos e possibilita a captação de novos clientes.

A próxima matriz do *modelo conceitual*, a Matriz da Qualidade Externa, exibida na FIGURA 9, é semelhante à Matriz de Qualidade em Uso, porém utiliza as características de qualidade externa, ao invés, das características de qualidade em uso. Nessa próxima matriz, a qualidade esperada pelo usuário é padronizada de acordo com o modelo de qualidade externa definido na norma ISO-9126.

4.2.4 Matriz de Qualidade Externa

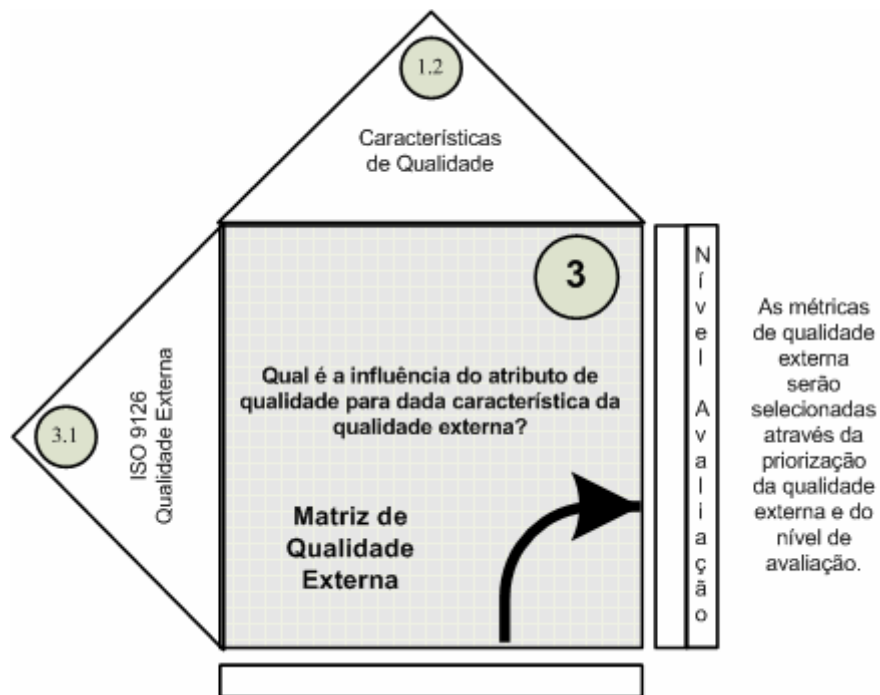


FIGURA 9 – Matriz de Qualidade Externa

A Matriz de Qualidade Externa do *modelo conceitual*, exibida na FIGURA 9, relaciona os atributos de qualidade esperados pelo usuário às características de qualidade definidas no modelo de qualidade externa da norma ISO-9126. Os atributos de qualidade utilizados na Matriz de Qualidade Externa são provenientes da Tabela de Características de Qualidade, tabela 1.2, da Matriz de Qualidade do Usuário. As características de qualidade externa são definidas no modelo de qualidade externa da norma ISO-9126 e formam a tabela 3.1 da Matriz de Qualidade Externa. As informações dessas tabelas são relacionadas pelo engenheiro de software para transferir padronização à qualidade esperada pelo cliente de acordo com o modelo de qualidade externa da norma ISO-9126.

A Tabela de Características de Qualidade contém os atributos de qualidade levantados e priorizados na construção da Matriz de Qualidade do Usuário, ilustrada na FIGURA 7. Esses atributos foram priorizados através do relacionamento desses com as funções do produto e receberam seus graus de importância. A priorização dos atributos de qualidade é utilizada para calcular o grau de importância da Tabela de Qualidade Externa.

A Tabela de Qualidade Externa é formada pelas características definidas no modelo de qualidade externa da norma ISO-9126. O modelo de qualidade externa possibilita avaliar a capacidade de um produto de software alcançar a qualidade esperada da execução do produto. Esse modelo de qualidade externa é formado por seis características: funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade. Sendo que cada característica é subdividida em subcaracterísticas que padronizam a qualidade do produto e podem ser mensuradas por medidas

externas.

As informações da Tabela de Características de Qualidade e da Tabela de Qualidade Externa são correlacionadas para construção da Matriz de Qualidade Externa. A *correlação* dessas tabelas é realizada pelo engenheiro de software nessa matriz através da pergunta “Qual é a influência do atributo de qualidade para dada característica da qualidade externa?”.

Depois do relacionamento das subcaracterísticas de qualidade externa com os atributos de qualidade, o grau de importância de cada subcaracterística de qualidade externa é calculado através da *conversão*.

O relacionamento realizado na Matriz de Qualidade Externa auxilia na priorização das medidas para as subcaracterísticas de qualidade externa da norma ISO-9126. Essa priorização é proveniente do relacionamento realizado nessa matriz e do grau de importância dos atributos de qualidade esperados pelo usuário. De acordo com as subcaracterísticas de qualidade externa priorizadas e os seus níveis de avaliação, o engenheiro de software pode selecionar as medidas de qualidade externa adequadas para o usuário.

As medidas de qualidade externa são utilizadas para controlar e acompanhar a qualidade do produto do código executável do software. Essas medidas podem ser coletadas em várias fases do desenvolvimento do projeto, principalmente na fase de testes, para verificar a qualidade externa antes de o produto ser entregue ao usuário. Essa verificação da qualidade permite o gerente do projeto antecipar a descoberta de falhas do produto.

Além da priorização das medidas de qualidade externa, o relacionamento realizado na matriz confere padronização aos atributos de qualidade esperados pelo cliente de acordo com a qualidade externa da norma ISO-9126. Essa padronização da qualidade esperada pelo usuário de acordo com a norma possibilita a empresa construir uma base histórica dos projetos realizados. A base histórica da Matriz de Qualidade Externa é unida com a base histórica da Matriz de Qualidade em Uso para completar a padronização da qualidade esperada pelo usuário.

A priorização das subcaracterísticas de qualidade externa, realizada através *conversão* realizada na Matriz de Qualidade Externa, será utilizada na construção da Matriz de Qualidade Interna, a próxima matriz do *modelo conceitual*.

4.2.5 Matriz de Qualidade Interna

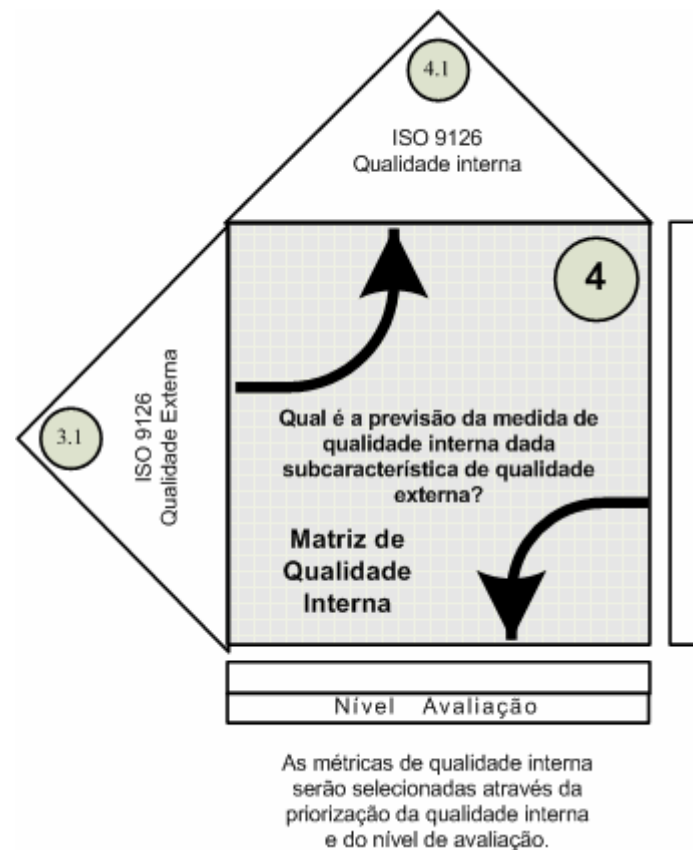


FIGURA 10 – Matriz de Qualidade Interna

A Matriz de Qualidade Interna do *modelo conceitual*, exibido na FIGURA 10, relaciona as subcaracterísticas de qualidade externa com as medidas de qualidade interna do produto de software. As subcaracterísticas de qualidade externa são provenientes da Tabela de Qualidade Externa, tabela 3.1, da Matriz de Qualidade Externa. As medidas de qualidade interna do desenvolvimento de software da empresa devem ser listadas na Tabela de Qualidade Interna, tabela 4.1. Essas tabelas são correlacionadas pelo engenheiro de software com objetivo de verificar quais medidas, e em que grau, de qualidade interna auxilia na previsão da qualidade externa do produto.

A Tabela de Qualidade Externa contém as subcaracterísticas de qualidade externa priorizadas, proveniente da Matriz de Qualidade Externa da FIGURA 9. As subcaracterísticas de qualidade externa, definidas no modelo da norma ISO-9126, foram priorizadas através da *correlação* com os atributos de qualidade esperados pelo usuário. Essas subcaracterísticas receberam seus respectivos graus de importância e são utilizados para calcular o grau de importância das medidas de qualidade interna do produto de software.

A Tabela de Qualidade Interna é formada pelas medidas de qualidade interna do produto de software da empresa. Essas medidas são coletadas com objetivo de acompanhar a qualidade durante a construção desse produto. Essa coleta das medidas tem por objetivo auxiliar a previsão da

qualidade externa esperada pelo usuário. Para alcançar esse objetivo, essas medidas internas são correlacionadas com as subcaracterísticas de qualidade externa da Matriz de Qualidade Externa, ilustrada na FIGURA 9.

As informações da Tabela de Qualidade Externa e da Tabela de Qualidade Interna são correlacionadas para construção da Matriz de Qualidade Interna. A *correlação* dessas tabelas é realizada pelo engenheiro de software nessa através da pergunta “Qual é a previsão da medida de qualidade interna dada subcaracterística de qualidade externa?”.

Depois do relacionamento das subcaracterísticas de qualidade externa com as medidas de qualidade interna, o grau de importância de cada medida de qualidade interna é calculado através da *conversão*. Essa *conversão* é realizada no sentido vertical, semelhante à Matriz de Qualidade do Usuário, para priorizar as medidas conforme a qualidade esperado pelo usuário do produto.

O relacionamento realizado na Matriz de Qualidade Interna possibilita priorizar as medidas para acompanhar as atividades do processo de desenvolvimento do software. Essa priorização é proveniente da *correlação* das medidas interna e dos graus de importância das subcaracterísticas da qualidade externa. Com as medidas priorizadas e o Nível de Avaliação dessas, o engenheiro de software pode mensurar as atividades do processo que influenciam a qualidade externa esperada pelo usuário. Essas medidas de qualidade interna podem ser coletadas em várias fases do desenvolvimento do projeto, desde o início da sua construção. A coleta dessas medidas auxilia a previsão da qualidade externa do produto final desenvolvido.

Além da seleção das medidas para acompanhar a qualidade interna, a priorização dessas medidas possibilita o gerente dimensionar melhor o esforço nas atividades de qualidade do projeto. Essas atividades podem ser dimensionadas focando alcançar as medidas de qualidade interna do produto. Dessa forma, o esforço das atividades do projeto estará priorizado de acordo com a qualidade interna e conseqüentemente com a qualidade final do produto esperada pelo usuário.

A seleção e a classificação das medidas de qualidade interna para mensuração da qualidade de produto serão detalhadas no capítulo seguinte. Essa classificação deve ser realizada com intuito de possibilitar a reutilização das medidas de projetos anteriores da mesma organização ou de outras medidas publicadas na literatura.

Capítulo 5

Classificação e seleção das medidas de qualidade

A classificação dos atributos de software possibilita organizar e priorizar as várias propriedades que, potencialmente, podem ser medidas no desenvolvimento de software. O número de propriedades envolvidas na construção de um software de grande porte é consideravelmente alto, devido à quantidade e a complexidade dos fatores envolvidos no seu processo de desenvolvimento. A coleta de medidas para cada fator envolve um custo e torna-se necessário restringir as medições a um conjunto limitado. Uma classificação de atributos é útil na seleção do subconjunto de referência de medidas a ser coletada para, de forma eficiente diminuir os custos e a complexidade do processo de desenvolvimento de software. As medidas classificadas neste capítulo são utilizadas na aplicação do método proposto neste trabalho, descrita no Capítulo 6.

Uma classificação de atributos de software padronizada auxilia na organização das medidas a serem utilizadas no processo de mensuração. A escolha adequada das medidas a serem coletadas é um fator importante para o sucesso de uma política de mensuração [FENTON96]. Além disso, a classificação dos atributos de software possibilita que outras medidas, ainda não utilizadas, sejam organizadas e potencialmente utilizadas na mensuração dos atributos de um projeto.

Para classificar as medidas de software é preciso categorizar os atributos do desenvolvimento em grupos bem definidos. Por exemplo, o *Framework de Atributos* definido por Fenton [FENTON96], descrito no Capítulo 2, identifica três classes de atributos de software: processo, recurso e produto. Cada uma dessas classes é subdividida em outras classes, por exemplo, a classe de produto é subdividida em qualidade, tamanho, complexidade entre outras. Essas classes têm uma descrição clara que possibilita classificar cada medida de acordo com seu propósito e suas entradas.

A classificação das medidas em uma estrutura de atributos de software possibilita a padronização entre os vários projetos de desenvolvimento de software de uma empresa. Dessa forma, a classificação serve de agente facilitador na criação da base histórica de projetos da instituição. Os dados coletados referentes às medidas auxiliam na avaliação dos atributos de software durante um projeto, mas também, podem servir para a construção de uma base histórica para a organização. A base histórica possibilita, por exemplo, uma melhor estimativa de preço de novos projetos.

Existem vários atributos do desenvolvimento de software que precisam ser mensurados para avaliar a qualidade desejada pelo usuário. A qualidade do software é influenciada por vários

fatores que variam desde a adequação das funcionalidades às necessidades do usuário até a facilidade de uso do produto. As informações coletadas sobre a qualidade do software podem ser organizadas através de um modelo de qualidade padronizado, como por exemplo, o modelo descrito na norma ISO-9126.

No método de aferição de qualidade proposta neste trabalho, é utilizada uma adaptação do *Framework de Atributos* [FENTON96], para padronizar a qualidade esperada pelo usuário. Essa classificação permite à empresa organizar as medidas de qualidade do produto utilizadas em projeto anteriores, bem como as medidas encontradas na literatura. O uso dessa classificação das medidas possibilita uma uniformização na aferição da qualidade. Através dessa uniformização, a empresa pode comparar a qualidade de um projeto com outros, e assim, ter informações sobre a evolução do seu processo de qualidade.

A classificação aqui proposta foi construída através da unificação da árvore de atributos de produtos, descrita por Fenton [FENTON96], com o modelo de qualidade da norma ISO-9126. A classificação definida por Fenton prevê uma subclasse de qualidade associada à classe de produto. A subclasse de qualidade foi dividida em qualidade interna e externa e qualidade em uso de acordo com o modelo da norma ISO-9126, conforme mostrado na FIGURA 11. Esta divisão é interessante no escopo deste trabalho, pois classifica as medidas para mensurar a qualidade do produto de software, selecionadas nas matrizes do método de aferição da qualidade.

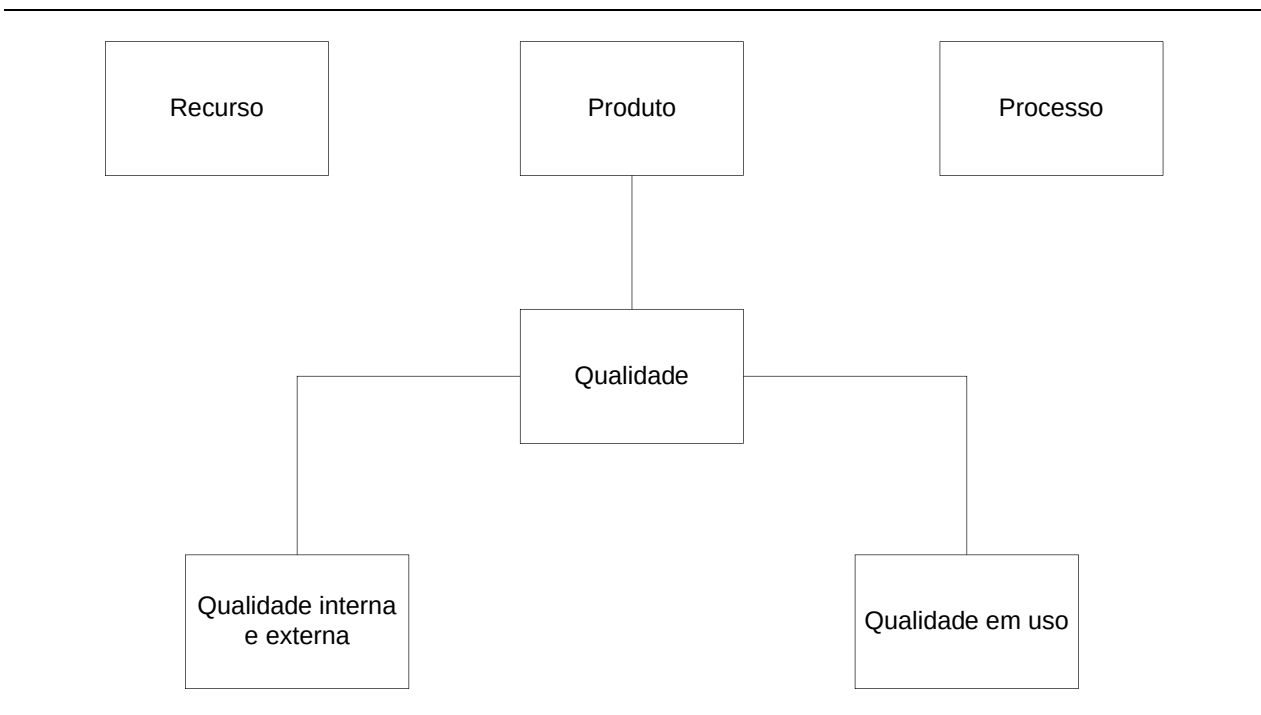


FIGURA 11 – Classificação de atributos de qualidade adaptada do trabalho de Fenton

A qualidade interna e externa é subdividida em seis características: funcionalidade, usabilidade, confiabilidade, eficiência, manutenibilidade e portabilidade; e essas são subdivididas em subcaracterísticas de acordo com o modelo da norma ISO-9126, conforme exibido na FIGURA 12.

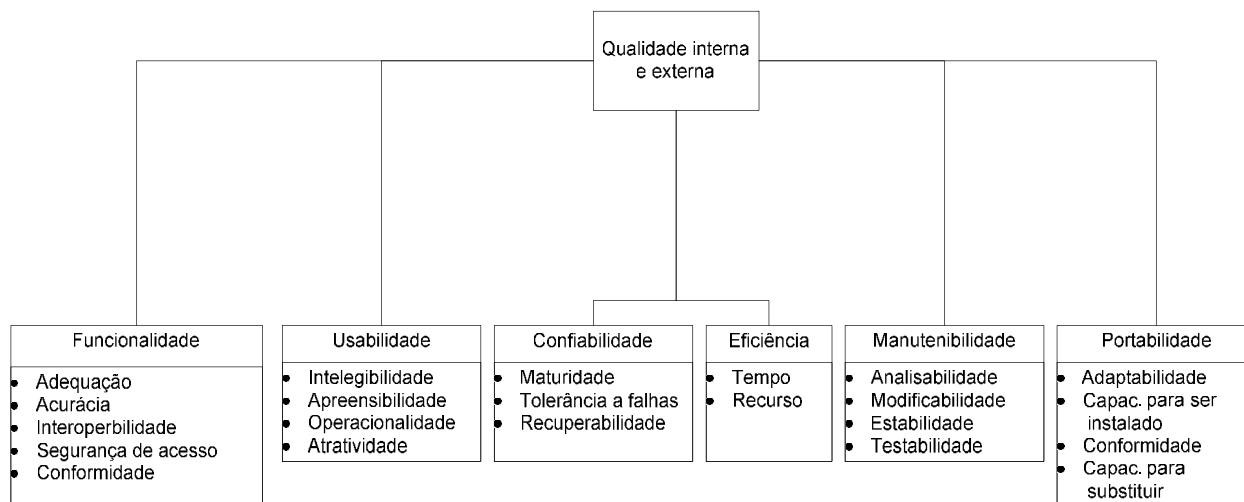


FIGURA 12 – Classificação de atributos de qualidade interna e externa

A qualidade em uso é subdividida em quatro características: efetividade, produtividade, segurança e satisfação, conforme exibido na FIGURA 13.

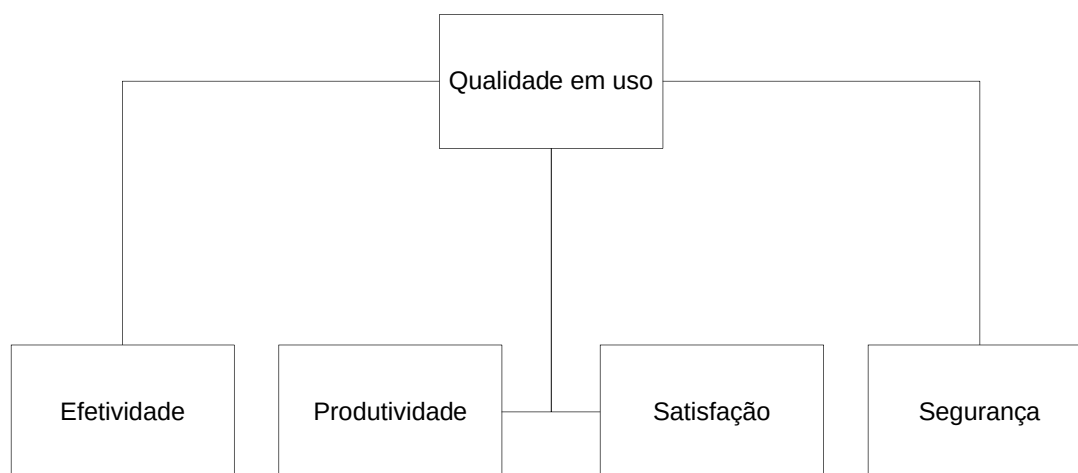


FIGURA 13 – Classificação de atributos de qualidade em uso

De acordo com a norma ISO-9126 as medidas para qualidade interna são aquelas mensuradas apenas avaliando o produto, sem se preocupar com sua execução. As medidas de qualidade externa, por sua vez, são aquelas mensuradas quando o produto está sendo executado [Iso01]. As medidas de qualidade em uso são aquelas mensuradas de acordo com a visão de qualidade do ponto de vista das necessidades dos usuários. Dessa forma, é possível classificar as medidas de qualidade de um produto de acordo com as características e as subcaracterísticas e também em medidas internas e externas ou em uso.

Além do modelo de qualidade da norma, o padrão dos parâmetros das tabelas de medidas

da norma ISO-9126 foi utilizado para descrever as medidas classificadas neste trabalho. Cada medida exemplificada na norma é descrita conforme os seguintes campos: nome, propósito, método de aplicação, fórmula, interpretação do valor, tipo de escala, tipos, entradas e audiência alvo. O campo nome é uma referência para medida que possibilita uma fácil comunicação entre as pessoas do projeto. O campo propósito descreve em poucas palavras o objetivo da coleta a medida. O campo método de aplicação define o processo de coleta da medida. O campo fórmula descreve a equação matemática para calcular o valor da medida. O campo interpretação do valor serve de referência para leitura do valor coletada da medida. O campo tipo de escala classifica a medida de forma que ela possa ser comparada com medidas de outros projetos. O campo tipos define o tipo de cada valor para o cálculo da medida. O campo entradas descreve os artefatos do processo de desenvolvimento envolvidos no cálculo da medida, e o campo audiência alvo define as pessoas que devem utilizar o valor coletado da medida. A descrição das medidas com esses campos permite documentá-las de forma clara e objetiva, e assim, melhorar a compreensão de tal medida pelo engenheiro de software que desejar utilizá-las.

Neste trabalho, foram classificadas algumas medidas da norma ISO-9126, do padrão IEEE-982 e do processo Praxis de acordo com a classificação proposta. Essas medidas foram escolhidas para demonstrar como classificar e descrever uma medida de qualidade a ser utilizada na metodologia usada neste trabalho. Como existe um número muito grande de medidas na literatura, essa classificação não tem o objetivo de ser extensiva e completa. Além disso, cada empresa de desenvolvimento pode utilizar suas medidas próprias que não estariam classificadas neste trabalho. As medidas selecionadas aqui são utilizadas na aplicação do método proposto neste trabalho.

A norma ISO-9126 fornece um conjunto de medidas já classificadas conforme seu modelo de qualidade de software e dessa forma, essas já estão classificadas de acordo com o *Framework de Atributos* adaptado. A norma ISO-9126 também é utilizada na construção do método de aferição da qualidade, descrita no Capítulo 4. A classificação proposta neste trabalho segue o modelo de qualidade da norma ISO-9126 e o padrão de campos para descrever suas medidas. Dessa forma, foram selecionadas algumas medidas internas da norma ISO-9126, somente para exemplificar, sendo elas:

- *Response time* (Eficiência)
- *Functional adequacy* (Funcionalidade)
- *Fault removal* (Confiabilidade)
- *Input Validity checking* (Usabilidade)
- *User operation cancellability* (Usabilidade)

As medidas de qualidade interna selecionadas estão descritas na TABELA 10 de acordo com o padrão de campos para classificação.

Nome	Propósito	Método de aplicação	Fórmula	Interpretação do valor	Tipo de escala	Tipos	Entradas	Audiência alvo
Response time (Eficiência)	Qual o tempo estimado para completar uma tarefa específica?	Avaliar a eficiência das chamadas do sistema operacional e da aplicação. Estimar o tempo de resposta baseada nisso. O que deve ser mensurado: - toda ou parte das especificações - testar caminhos de transação completos - testar módulos completos/partes do produto de software - o produto completo durante a fase de testes	X=tempo (calculado ou simulado)	Quanto menor melhor.	Razão	X=tempo	Sistema operacional conhecido Tempo estimado nas chamadas do sistema.	Desenvolvedor Analista de requisito

Functional adequacy (Funcionalidade)	Quão adequadas são as funções verificadas?	Contar o número de funções implementadas que são adequadas para desempenhar as tarefas especificadas, para medir a sua razão para as funções implementadas. O que deve ser mensurado: - toda ou parte das especificações - módulos completos/partes do produto de software	$X=1-A/B$ A = Número de funções em avaliação nas quais os problemas são detectados B = Número de funções verificadas.	$0 \leq X \leq 1$ Quanto mais próximo de 1, mais adequado.	Absoluta	X=contador/ contador A=contador B=contador	Especificação de requisitos Código fonte Relatório de revisão	Desenvolvedor Analista de requisito
Fault removal (Confiabilidade)	Quantas falhas foram corrigidas? Qual é a proporção de falhas removidas?	Contar o número das falhas removidas durante a codificação e compará-la com o número de falhas detectadas a revisão da codificação.	$X=A$ A=Número de falhas corrigidas durante a codificação. $Y=A/B$ A=Número de falhas corrigidas durante a codificação B= Número de falhas detectadas na revisão	$0 \leq X$ Um alto valor de X implica que menos falhas permanecem $0 \leq Y \leq 1$ Quanto mais próximo de 1, melhor. (mais falhas removidas)	Razão Absoluta	X=contador A=contador Y=contador/ contador B=contador	Valor A é proveniente do relatório de remoção de falhas. Valor B é proveniente do relatório de revisão.	Desenvolvedor Analista de requisito

Input Validity checking (Usabilidade)	Qual proporção de campos de entrada fornece checagem para dados válidos?	Contar o número de campos de entrada, o qual checa dado válido e compara com número de campos de entrada que poderia checar dado válido.	$X=A/B$ A=Número de campos de entrada com checagem de dado válido B=Número de campos de entrada os quais poderia ter checagem de dado válido	$0 \leq X \leq 1$ Quanto mais próximo de 1, melhor.	Absoluta	X=contador/ contador A=contador B=contador	Especificação de requisitos Relatório de revisão	Desenvolvedor Analista de requisito
User operation cancellability (Usabilidade)	Qual proporção de funções pode ser cancelada antes de completar?	Contador o número de funções implementadas, que podem ser canceladas pelo usuário antes de completar e compará-lo com o número de funções que requerem a capacidade de serem canceladas.	$X=A/B$ A=Número de funções implementadas que podem ser canceladas pelo usuário. B=Número de funções que requerem a capacidade de serem canceladas.	$0 \leq X \leq 1$ Quanto mais próximo de 1, melhor.	Absoluta	X=contador/ contador A=contador B=contador	Especificação de requisitos Relatório de revisão	Desenvolvedor Analista de requisito

TABELA 10 – Medidas internas da norma ISO-9126

O padrão IEEE-982 [IEEE88] também foi utilizado para exemplificar a classificação de diferentes medidas de acordo com a classificação adaptada de Fenton. O padrão da IEEE tem o objetivo de fornecer um conjunto de medidas utilizadas como indicador da característica Confiabilidade da qualidade de um software. As medidas descritas nesse padrão são organizadas conforme uma classificação própria. Algumas medidas do padrão IEEE-982 foram selecionadas para demonstrar a sua classificação de acordo com *Framework de Atributos* proposto aqui, sendo elas:

- *Cumulative Failure Profile* (descrita no item 4.3 do padrão)
- *Requirements Traceability* (descrita no item 4.7 do padrão)

A medida “*Cumulative Failure Profile*” pode ser aplicada para prever a confiabilidade de um produto através do uso de perfis de defeitos, ou para identificar os módulos ou os subsistemas que requerem testes adicionais. Dessa forma, essa medida pode ser utilizada para prever a maturidade ou para mensurar a testabilidade de um produto. Dessa forma, essa medida é classificada na subcaracterística maturidade da característica confiabilidade e também classificada na subcaracterística testabilidade da característica manutenibilidade.

A medida “*Requirements Traceability*” pode ser aplicada para identificar os requisitos que estão faltando ou fora do escopo dos requisitos definidos inicialmente. Portanto, essa pode ser utilizada como indicativo da adequação das funcionalidades de um produto. Dessa forma, essa medida é classificada na subcaracterística Adequação da característica Funcionalidade do modelo de qualidade interna e externa do *Framework de Atributos*.

As medidas escolhidas do padrão IEEE-982 foram descritas aqui de acordo com os campos da tabela do *Framework de Atributos*. As informações foram extraídas do padrão IEEE e colocadas na TABELA 11.

Nome	Propósito	Método de aplicação	Fórmula	Interpretação do valor	Tipo de escala	Tipos	Entradas	Audiência alvo
<i>Cumulative Failure Profile</i>	- Prever a confiabilidade através do uso de perfis de falhas. - Identificar módulos ou subsistemas que requerem testes adicionais	Estabelecer os níveis de severidade a designação de falhas.	Gerar gráfico de falhas acumuladas versus a base de tempo adequada. A curva pode ser gerada para o sistema como um todo, subsistemas ou módulos.	O valor de $F(i)$ deve ser interpretado conforme base histórica da organização e nível de confiabilidade desejado.	Absoluta	$F(i) =$ Contador	$F(i) =$ número de falhas para certa severidade para um dado intervalo de tempo, $i=1, \dots$	Gerente Desenvolvedor Testador
<i>Requirements Traceability</i>	Auxilia a identificação de requisitos que estão faltando, ou além, dos requisitos originais.	Um conjunto de mapeamento dos requisitos na arquitetura do software para os requisitos originais é criado. Contar cada requisito mapeado pela arquitetura (R1) e contar cada um dos requisitos originais (R2).	$TM = R1/R2 \times 100\%$	$0\% \leq TM \leq 100\%$ Quanto mais próximo de 100% melhor.	Absoluta	R1 = Contador R2 = Contador TM = Contador/ Contador (%)	R1 = Número de requisitos mapeados pela arquitetura R2 = Número de requisitos originais	Gerente Desenvolvedor

TABELA 11 – Algumas medidas do padrão IEEE-982

Algumas medidas do processo Praxis foram escolhidas para exemplificar a classificação de medidas de acordo com o *Framework de Atributos*. Como este trabalho se preocupa fundamentalmente com a mensuração do produto, as medidas escolhidas do Praxis 2.1 foram àquelas relacionadas ao produto. Essas medidas são utilizadas na aplicação do método proposto neste trabalho, descrita no Capítulo 6.

No Praxis, existem três situações típicas em que é possível coletar medidas de defeitos: procedimentos de revisão e auditoria e relatórios de teste. Nessas atividades, os defeitos são registrados nos relatórios RRSw (Relatório de Revisão de Software), RISw (Relatório de Inspeção de Software) e RTSw (Relatório de Teste de Software). Esses relatórios do Praxis são consolidados no relatório RAPS w (Relatório de Acompanhamento de Projeto de Software), visando o

acompanhamento da medida de defeito do projeto. A contagem de defeitos constitui a base de diversos indicadores de qualidade comumente utilizados, como confiabilidade, correção, completeza, eficiência e usabilidade [IEEE90]. Dessa forma, a medida de defeito do Praxis influencia vários atributos do *Framework de Atributos* envolvidos com a qualidade de produto de software.

Para exemplificar a classificação das medidas do Praxis foram consideradas as seguintes medidas de defeitos:

- Defeitos de Testes de aceitação
- Defeitos de Avaliação pelo usuário

A medida “Defeitos de Testes de aceitação” é coletada na atividade de teste de aceitação do Praxis. Os testes de aceitação têm por objetivo validar o produto, ou seja, verificar se ele atende aos requisitos especificados [PAULA03]. O produto deve atender tanto aos requisitos funcionais quanto aos requisitos não-funcionais. Dessa forma, a medição em testes de aceitação pode envolver praticamente todas as características de qualidade do produto, dependendo de o que o cliente especificou como requisito a ser atendido. Uma vez que os testes de aceitação são realizados a partir do código executável do produto, a medição de defeitos desses testes é considerada como uma medida de qualidade externa ou qualidade em uso.

A medida “Defeitos de Avaliação pelo usuário” é coletada na atividade de avaliação de uso, que formaliza a avaliação do produto por parte dos usuários. De acordo com a lista de conferência utilizada pela atividade de avaliação de uso, ela tem por objetivo verificar os aspectos que influenciam a característica de usabilidade da qualidade do produto. As medidas de defeitos do processo Praxis são descritas na TABELA 12 de acordo com os campos definidos no *Framework de atributos*.

Nome	Propósito	Método de aplicação	Fórmula	Interpretação do valor	Tipo de escala	Tipos	Entradas	Audiência alvo
Defeitos de Testes de aceitação	Validar os requisitos do produto	Essa medida é coletada na atividade de testes de aceitação.	F = número de defeitos encontrados	Quanto menor melhor	Absoluta	F = Contador	Defeitos registrados no RTSw.	Auditor de qualidade Gerente do projeto
Defeitos de Avaliação pelo usuário	Avaliar o uso do produto	Essa medida é coletada na atividade de avaliação de uso.	F = número de defeitos encontrados	Quanto menor melhor	Absoluta	F = Contador	Defeitos registrados no RAU.	Auditor de qualidade Gerente do projeto

TABELA 12 – Medidas do processo Praxis

As medidas de qualidade selecionadas e classificadas neste capítulo são utilizadas na aplicação do método de aferição da qualidade do produto, descrita no próximo capítulo.

Capítulo 6

Instanciação do método de aferição de qualidade

Este capítulo exemplifica como o método de aferição de qualidade proposta pode ser aplicada a um processo de desenvolvimento concreto. Isso é feito através da sua instanciação para o processo Praxis 2.1, descrito em [PAULA03].

Foram dois os fatores principais que motivaram a instanciação do método para o processo Praxis:

- Exemplificar como as matrizes do método, apresentadas na seção 4.2, pode ser definido para adequá-la ao contexto de um processo real;
- Fornecer os subsídios necessários para a elaboração de uma personalização do Praxis que contemple as práticas de mensuração de qualidade de produto, já que o processo atualmente não possui uma política de medição bem definida.

Uma descrição do processo Praxis pode ser encontrada na seção 2.5. As seções seguintes, de 6.1 a 6.4, descrevem a instanciação do método ao processo Praxis. A seção 6.5 apresenta a aplicação do método ao produto Merci 1.0, que é um sistema didático utilizado em [PAULA03] para exemplificar o processo Praxis.

6.1 Matriz de Qualidade do Usuário

A instanciação da Matriz de Qualidade do Usuário é relacionada no Praxis com os artefatos e as atividades da fase de Elaboração. Esta relação existe, pois esta fase tem como objetivo entender o problema do cliente e criar artefatos que expressem as necessidades do cliente. A Matriz de Qualidade de Usuário tem, por sua vez, exatamente o objetivo de auxiliar a padronização da qualidade do produto esperada pelo usuário. Os artefatos e as atividades envolvidos no levantamento da qualidade do produto no processo Praxis são: CRSw (Cadastro de Requisitos do Software) e MASw (modelo de Análise do Software).

O CRSw é o artefato responsável por registrar os casos de uso identificados juntamente

com o usuário do produto. Esses casos de uso são utilizados para formar a lista de funções na Tabela de Requisitos Funcionais, da Matriz de Qualidade do Usuário. Essa tabela irá auxiliar na *extração* dos atributos de qualidade da Tabela de Características de Qualidade.

Os atributos de qualidade da Tabela de Características de Qualidade são obtidos a partir da *extração* das funções em atributos de qualidade e a partir dos requisitos não-funcionais registrados no MASw e no CRSw. No processo Praxis, os requisitos não-funcionais são levantados com os usuários durante o fluxo de Análise no subfluxo Oficina de detalhamento dos requisitos. Após a identificação dos atributos de qualidade é possível fazer a *correlação* na Matriz de Qualidade do Usuário.

A *correlação* da Tabela de Requisitos Funcionais com a Tabela de Características de Qualidade tem por objetivo priorizar os atributos de qualidade. Essa priorização dos atributos deve ser realizada na Iteração de Análise de Requisitos do processo Praxis com a participação do usuário. O fluxo de análise visa detalhar, estruturar e validar os requisitos de um produto, em termos de um modelo conceitual do problema. Desta forma os requisitos podem ser usados como base para o planejamento e controle detalhado do respectivo projeto de desenvolvimento [PAULA03]. Nessa Iteração a lista de funções do produto está completa e o usuário pode identificar os atributos de qualidade desejados. Sendo assim, a identificação e a priorização dos atributos de qualidade se encaixam como atividades do fluxo de Análise e devem ser registradas utilizando o artefato MASw.

Os atributos de qualidade da Matriz de Qualidade do Usuário servem de entrada para o planejamento e o controle das atividades da fase de Construção e Transição do Praxis. A identificação e a documentação dos atributos de qualidade melhoram o entendimento do analista sobre a qualidade, e dessa forma, possibilitam a construção um produto mais adequado ao usuário. Além disso, a documentação dos atributos pode ser utilizada no planejamento e controle mais detalhado das atividades de garantia da qualidade.

Os atributos de qualidade, registrados no MASw, serão utilizados para construir a próxima matriz do método de aferição da qualidade, que é a Matriz de Qualidade em Uso.

6.2 Matriz de Qualidade em Uso

A Matriz de Qualidade em Uso tem o objetivo de conferir padronização, usando a norma ISO-9126, à qualidade esperada pelo usuário. A qualidade esperada pelo usuário foi priorizada pela *conversão* da Matriz de Qualidade de Usuário. Nesta seção será apresentada a instanciação dessa matriz para o Praxis.

A Matriz de Qualidade em Uso é formada pela *correlação* da Tabela de Características de Qualidade com a Tabela de Qualidade em Uso. A Tabela de Características de Qualidade é proveniente da Matriz de Qualidade do Usuário. A Tabela de Qualidade em Uso é formada pelas características de qualidade em uso definidas na norma ISO-9126. Essas duas tabelas são relacionadas com objetivo de identificar quais atributos de qualidade têm influência na qualidade em uso do

produto.

A Tabela de Características de Qualidade contém os atributos de qualidade priorizados através da *conversão* na Matriz de Qualidade de Usuário. Essa priorização é utilizada para fazer o cálculo da *conversão* das características de qualidade em uso.

As características de qualidade em uso, definidas na norma ISO-9126, são usadas para construir a Tabela de Qualidade em Uso, sendo elas: efetividade, produtividade, segurança e satisfação.

Com as duas tabelas construídas é possível correlacionar seus itens na Matriz de Qualidade em Uso. Essa *correlação* deve seguir os passos do método de aferição de qualidade, descritos na seção 4.2.3 deste documento. Para executar esses passos foi inserida uma atividade no fluxo de Desenho do Praxis. A atividade deve ser realizada em uma reunião com a participação de, pelo menos, um responsável de cada fluxo do Praxis, permitindo a interação de várias visões na *correlação* da matriz. Nessa reunião, os atributos de qualidade e as características de qualidade em uso são correlacionados utilizando a experiência dos participantes e a definição da norma ISO-9126.

Depois da *correlação*, realizada na Matriz de Qualidade em Uso, deve ser obtido o grau de importância das características de qualidade em uso conforme processo do método proposto neste trabalho, descrito na seção 4.2.3. As características de qualidade em uso priorizadas devem ser armazenadas em uma nova seção intitulada “Qualidade em Uso” do documento PQSw (Plano de Qualidade do Software) do processo Praxis. As informações contidas nessa seção podem ser utilizadas como insumo para as atividades da fase Construção e Transição do Praxis. As informações podem, por exemplo, auxiliar no planejamento dos casos de testes, que é uma das primeiras atividades do fluxo de Teste da fase de Construção.

Com o grau de importância das características de qualidade em uso e o Nível de Avaliação, descrito na seção 4.2.1, é possível priorizar as medidas de qualidade em uso mais adequadas para mensurar a qualidade esperada pelo usuário. A seleção das medidas deve ser realizada no subfluxo Planejamento do fluxo de Gestão de Projeto e devem ser registradas no artefato MPPSw (Memória de Planejamento de Projeto de Software). Dessa forma, a qualidade em uso esperada pelo usuário pode ser acompanhada durante a execução do projeto de desenvolvimento.

6.3 Matriz de Qualidade Externa

A Matriz de Qualidade Externa tem por objetivo proporcionar padronização à qualidade esperada pelo usuário de acordo com a norma ISO-9126 como referência. A qualidade esperada pelo usuário está representada com uma lista de atributos priorizada, proveniente da Matriz de Qualidade do Usuário. Nesta seção será apresentada a instanciação da Matriz de Qualidade Externa para o Praxis.

A Matriz de Qualidade Externa é formada pela *correlação* da Tabela de Características

de Qualidade, proveniente da Matriz de Qualidade do Usuário, com a Tabela de Qualidade Externa. A Tabela de Qualidade Externa é formada pelas subcaracterísticas de qualidade externa definidas na norma ISO-9126. Essas subcaracterísticas são correlacionadas com os atributos de qualidade para construir a Matriz de Qualidade Externa.

Para construir a Tabela de Qualidade Externa são utilizadas as subcaracterísticas de qualidade externa definidas na norma ISO-9126, que estão associadas às características de qualidade, como ilustrado na FIGURA 14.

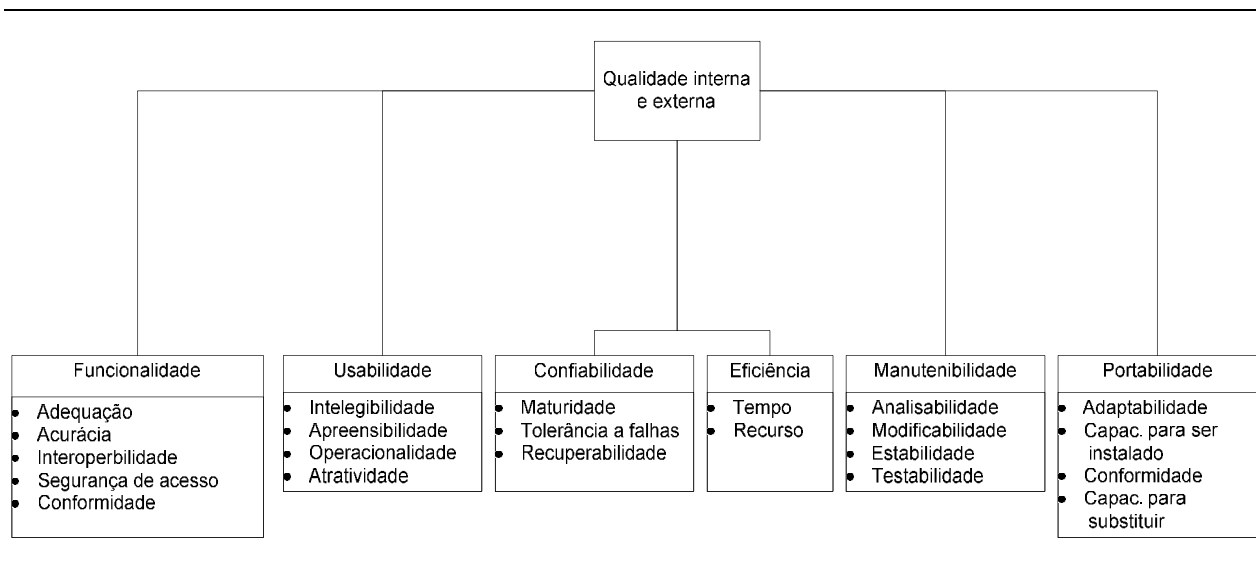


FIGURA 14 – Qualidade externa de acordo com a norma ISO-9126

A *correlação* realizada na Matriz de Qualidade Externa deve seguir os passos do método de aferição da qualidade, descritos na seção 4.2.4 deste documento. Para executar esses passos deve-se inserir uma atividade no fluxo de Desenho do Praxis. A atividade deve seguir o processo semelhante ao realizado na Matriz de Qualidade em Uso.

Depois de realizada a *correlação* na Matriz de Qualidade Externa deve-se calcular o grau de importância das subcaracterísticas de qualidade externa. As subcaracterísticas de qualidade externa priorizadas são armazenadas em uma nova seção intitulada “Qualidade Externa” no documento PQSw do Praxis. As informações contidas nessa seção podem auxiliar o planejamento das atividades das fases de Construção e Transição do Praxis. Por exemplo, essas informações podem guiar o planejamento dos casos de testes de unidade, identificando os testes adequados às necessidades do usuário.

Com o grau de importância das subcaracterísticas de qualidade externa e o Nível de Avaliação é possível selecionar medidas de qualidade externa adequadas para mensurar a qualidade esperada pelo usuário. A seleção das medidas deve ser realizada no subfluxo Planejamento do fluxo de Gestão de Projeto do Praxis e devem ser registradas no artefato MPPSw. Dessa forma, a qualidade em uso esperada pelo usuário pode ser acompanhada durante a execução do projeto de desenvolvimento.

Depois da *conversão* das subcaracterísticas de qualidade externa da Matriz de Qualidade Externa é possível construir a próxima matriz do método de aferição de qualidade, que é a Matriz de Qualidade Interna.

6.4 Matriz de Qualidade Interna

A Matriz de Qualidade Interna tem por objetivo auxiliar na priorização das medidas interna de qualidade, que são coletadas durante o desenvolvimento de um projeto de software. A priorização dessas medidas será realizada através da sua *correlação* com as subcaracterísticas de qualidade externa definidas na norma ISO-9126.

A Matriz de Qualidade Interna correlaciona a Tabela de Qualidade Externa com a Tabela de Qualidade Interna. A Tabela de Qualidade Externa é formada pelas subcaracterísticas de qualidade externa priorizadas, proveniente da Matriz de Qualidade Externa. Já a Tabela de Qualidade Interna é formada pelas medidas internas da qualidade do produto.

As subcaracterísticas de qualidade externa estão priorizadas e registradas na seção “Qualidade Externa” do PQSw do Praxis. Essa priorização será utilizada para realizar a *conversão* das medidas internas de qualidade.

Para construir a Tabela de Qualidade Interna são utilizadas as medidas de qualidade interna de produto. A qualidade interna é o total das características de um produto de software a partir da visão interna. No Praxis, as características da qualidade interna são formadas pelos artefatos produzidos durante a fase de Elaboração e Construção. As atividades dessas fases têm por objetivo modelar e implementar um produto que atenda à qualidade esperada pelo usuário. Para mensurar a qualidade o Praxis prevê algumas medidas internas, como por exemplo, defeitos de revisões, defeitos de testes de aceitação e defeitos de avaliação pelos usuários. Geralmente, essas medidas não englobam todas as características esperadas do produto pelo usuário; sendo assim, a Tabela de Qualidade Interna deve ser completada com medidas de qualidade interna, julgadas importantes. A tabela pode ser completada, por exemplo, com as medidas classificadas de acordo com o modelo de qualidade da norma ISO-9126.

A construção da Matriz de Qualidade Interna deve seguir os passos do método de aferição da qualidade, descritos na seção 4.2.5 deste documento. A priorização das medidas de qualidade interna deve ser realizada no subfluxo Planejamento do fluxo de Gestão de Projeto do Praxis e devem ser registradas no artefato MPPSw. Essas medidas serão coletadas durante o desenvolvimento do produto para o acompanhamento da qualidade interna dos artefatos produzidos pelo processo Praxis.

A coleta das medidas de qualidade deve ser realizada no processo Praxis pelo subfluxo Garantia da Qualidade do fluxo de Gestão da Qualidade. Os resultados da coleta devem ser armazenados em uma nova seção intitulada “Medidas de Qualidade Interna” no relatório RAPS w (Relatório de Acompanhamento de Projeto de Software) do Praxis. As informações contidas nessa

seção são utilizadas para acompanhar a qualidade interna do produto e para contemplar a qualidade externa esperada pelo usuário.

6.5 Aplicação do método para o produto Merci 1.0

Esta seção exemplifica os resultados obtidos com a aplicação do método de aferição da qualidade proposta em um produto de software. Durante a aplicação são realizadas algumas avaliações qualitativas dos benefícios esperados do método. Contudo, consideramos que avaliações adicionais são necessárias para validar o método de aferição de qualidade, proposta neste trabalho.

O método deve ser validado através de um experimento de engenharia de software, preferencialmente, em um ambiente real de desenvolvimento de software [TRAVASSOS02]. A validação do método depende da disponibilidade de recursos e pessoas chaves, envolvidas nos projetos. Mesmo com os recursos, ainda seria preciso planejar o experimento do método podendo demandar meses para sua conclusão.

Diante da falta dos recursos necessários, principalmente tempo, para uma validação mais detalhada do método, decidiu-se por aplicá-la ao produto Merci 1.0 com objetivo de identificar possíveis melhorias e eventuais problemas. Essa análise possibilitou a identificação de melhorias no trabalho definido inicialmente, permitindo seu aprimoramento.

Nas seções seguintes será descrita e discutida a aplicação do método proposto ao produto Merci.

6.5.1 Matriz de Qualidade do Usuário

A Matriz de Qualidade do Usuário tem por objetivo auxiliar a padronização da qualidade do produto esperada pelo usuário. Essa matriz é formada pela Tabela de Requisitos Funcionais e pela Tabela de Características de Qualidade. O preenchimento dessas tabelas com os dados do produto Merci, para a construção da Matriz de Qualidade do Usuário, será descrito a seguir.

Para preencher a Tabela de Requisitos Funcionais foi utilizada a lista de CUA (casos de uso de análise) do modelo CRSw e MASw. Essa lista é composta por:

- Gestão Manual de Estoque
- Gestão de Mercadorias
- Gestão de Fornecedores
- Gestão de Pedidos de Compra
- Operação de Venda
- Relatório de Estoque Baixo

- Relatório de Mercadorias
- Relatório de Fornecedores
- Relação de Pedidos de Compra
- Emissão de Nota Fiscal

Cada caso de uso representa uma função que agrega valor ao produto, do ponto de vista do usuário. Dessa forma, essa lista representa o total de valor esperado pelo usuário do Merci. Esse valor agregado está documentado na Matriz de Qualidade do Usuário e será correlacionado com a Tabela de Características de Qualidade.

No Merci, a Tabela de Características de Qualidade é obtida usando os atributos de qualidade obtidos a partir do modelo MASw e do documento PESw (proposta de especificação de software). A seção 2 do documento PESw fornece uma lista dos benefícios esperados pelo usuário, utilizados para fazer a *extração* dos atributos de qualidade. No caso do Merci, os atributos de qualidade foram completados pela lista dos requisitos não-funcionais e as restrições de memória, obtidos a partir do modelo MASw. A junção das informações desses dois artefatos compõe a lista de atributos de qualidade:

- Economia de mão-de-obra na operação de venda
- Agilidade na compra e venda de mercadorias.
- Diminuição de erros nas vendas.
- Diminuição do tempo de venda.
- Diminuição dos prejuízos na operação de venda
- Diminuição dos erros nas notas fiscais.
- Eliminação da duplicidade de pedidos.
- Tempo de operação de venda deverá gastar no máximo 2s
- Tempo de operação de pesquisa de 10s
- Desenhado de forma que possa ser expandido para mais de um terminal de caixa.
- Leiaute do relatório Nota Fiscal deve seguir padrão da Secretaria da Receita.
- Um operador de caixa deverá ser capaz de aprender a operar o Merci com 1 dia de treinamento.
- Restringir o acesso dos usuários às funções através de senhas, conforme o respectivo grupo.
- O produto deve ocupar no máximo 200 MB (sem considerar as bases de dados).
- O produto deve executar em 128 MB.

A lista de atributos representa a qualidade esperada pelo usuário do produto Merci. Essa qualidade deverá ser acompanhada durante o desenvolvimento do projeto com o objetivo de garantir

que o produto final esteja dentro das expectativas do usuário. O acompanhamento da qualidade é auxiliado pelo método, pois documenta a qualidade de forma clara para consulta de todos envolvidos no projeto.

Para correlacionar a Tabela de Características de Qualidade e a Tabela de Requisitos Funcionais foram utilizadas as informações disponíveis nos artefatos do produto Mercí. Do artefato MASw foi utilizada a aba “Influências” da planilha “Priorização das funções”, exibida na FIGURA 15, que representa a relação das funções com os benefícios esperados do Mercí. No relacionamento da planilha, os valores 10, 4 e 1 representam, respectivamente, uma influência alta, média e baixa. O valor de cada influência foi utilizado para realizar a *correlação* dos atributos de qualidade extraídos dos benefícios do Mercí.

Influências

Número	Benefício	Peso do benefício	Gestão Manual de Estoque	Relatório de Estoque Baixo	Relatório de Mercadorias	Relatório de Fornecedores	Relação de Pedidos de Compra	Gestão de Mercadorias	Gestão de Fornecedores	Gestão de Pedidos de Compra	Operação de Venda	Emissão de Nota Fiscal
1	Diminuição de erros na venda de mercadorias.	10						10	1			
2	Qualidade na emissão da nota fiscal e ticket de venda, em relação à emissão manual.	10									10	10
3	Identificação de distorções entre o vendido e o estoque.	10	10	4				4			10	
4	Agilidade na compra de mercadorias.	4		1	4	4	4	4		4	4	
5	Economia de mão-de-obra.	4	4	1	1	1	1	1	1	4	1	10
6	Diminuição do custo de estocagem.	4	10	4	1			4		1		
7	Identificação de produtos mais e menos vendidos.	4	1				4	4			4	
8	Conhecimento do mercado de fornecedores.	1			1	4		4	10			
9	Indicação de promoções.	1	10		1			4			4	

FIGURA 15 – Excerto da planilha de priorização das funções do Mercí 1.0

Para realizar a *correlação* dos requisitos não-funcionais verificaram-se quais funções eram influenciadas por esses requisitos. Por exemplo, o requisito não-funcional “Tempo de operação de venda deverá gastar no máximo 2s” tem uma forte correlação com a função “Operação de Venda”, sendo, assim, registrado como tendo uma correlação alta na Matriz de Qualidade do Usuário. Para construir o restante da matriz, mostrada na FIGURA 16, foi considerado a *correlação* de todas as funções com os atributos de qualidade do produto.

Matriz de Qualidade do Usuário

Características de Qualidade

Requisitos Funcionais	Importance	Tempo de operação de pesquisa de 10s	Diminuição de erros nas vendas.	Diminuição do tempo de venda.	Diminuição dos prejuízos na operação de venda	Economia de mão-de-obra na operação de venda	Tempo de operação de venda deverá gastar no máximo 2s	Um operador de caixa deverá ser capaz de aprender a operar o Mercú com 1 dia de treinamento	Agilidade na compra e venda de mercadorias.	Diminuição dos erros nas notas fiscais.	Restringir o acesso dos usuários às funções através de senhas, conforme o respectivo grau de importância.	O produto deve ocupar no máximo 200 MB (sem considerar as bases de dados).	O produto deve executar em 128 MB.	Eliminação da duplicidade de pedidos.	Leiaute do relatório Nota Fiscal deve ser aprovado pela Sec. de Receita.	Desenhado de forma que possa ser expandido para mais de um terminal de caixa.	Total
Gestão de Mercadorias	4	A	M	B	B				M	B	B	B	B	B			88
Gestão Manual de Estoque	4	A									B	B	B				48
Operação de Venda	4		A	A	A	A	A	A	M	M	B	B	B				252
Emissão de Nota Fiscal	2		A	A	A	A	M	B		A	B	B	B		A		122
Gestão de Fornecedores	1	A							B		B	B	B	M			16
Gestão de Pedidos de Compra	1	A							M		B	B	B	A			24
Relação de Pedidos de Compra	1								M		B	B	B	B			7
Relatório de Estoque Baixo	1								B		B	B	B				4
Relatório de Fornecedores	1								B		B	B	B	B			5
Relatório de Mercadorias	1								M		B	B	B	B			7
Total		90	66	58	58	54	42	38	36	34	20	20	20	19	18	0	

FIGURA 16 – Matriz de Qualidade do Usuário para o produto Mercú 1.0

Nessa matriz, o requisito não-funcional “Desenhado de forma que possa ser expandido para mais de um terminal de caixa” não foi relacionado, pois no produto Mercú esse requisito está adiado, como mostrado na seção “Requisitos Adiadados” do MASw.

Depois da *correlação* realizada na Matriz de Qualidade do Usuário, é calculado o grau de importância dos atributos de qualidade. Na matriz exibida na FIGURA 16, os atributos de qualidade já estão ordenados pelo grau de importância calculado.

Através da *correlação* das funções com os atributos de qualidade podem-se verificar quais os atributos de qualidade são prioritários para o usuário do produto. O gráfico de importância obtido para todos os atributos de qualidade do Mercú está mostrado na FIGURA 17. Nesse gráfico do

Merci, o atributo de qualidade mais importante é “Tempo de operação de pesquisa de 10s” e o menos importante é “Leiaute do relatório Nota Fiscal deve seguir padrão da Secretaria da Receita”. Isso demonstra que o usuário do Merci está mais preocupado com a rapidez que os resultados das pesquisas são retornados, e assim, esse atributo de qualidade deve ser priorizado na fase de Construção do produto.

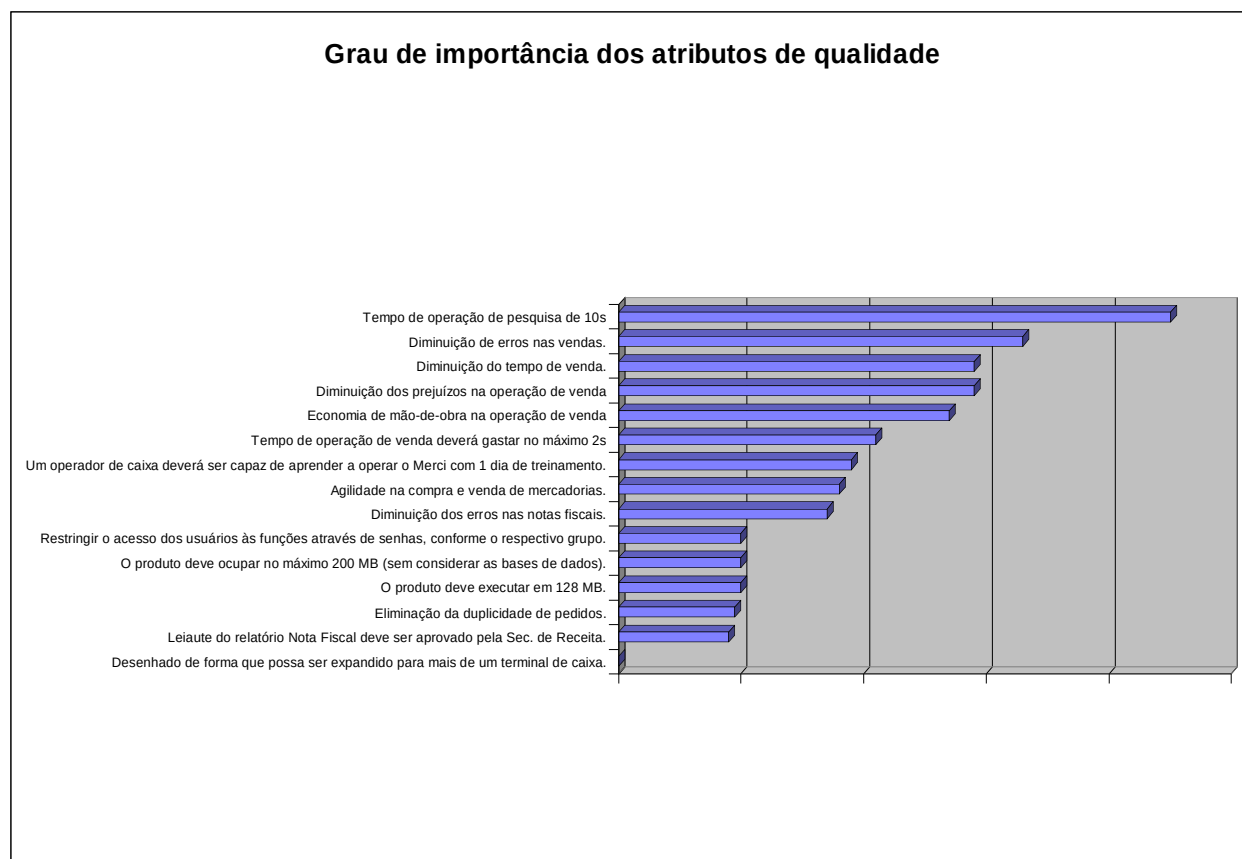


FIGURA 17 – Gráfico de importância dos atributos de qualidade

A priorização dos atributos de qualidade esperados pelo usuário é utilizada no planejamento e no controle das atividades do desenvolvimento de um produto. Por exemplo, nas atividades do fluxo de Gestão da Qualidade do Praxis pode ser considerada a priorização dos atributos no dimensionamento do esforço e da cobertura desejada para a qualidade do produto. No Praxis, o esforço e cobertura planejados da qualidade são registrados no documento PQSw e são acompanhados durante o desenvolvimento do produto.

Os atributos de qualidade priorizados serão utilizados para construir a Matriz de Qualidade em Uso e a Matriz de Qualidade Externa, discutidas, respectivamente, nas seções 6.5.2 e 6.5.3.

6.5.2 Matriz de Qualidade em Uso

A Matriz de Qualidade em Uso foi construída para o Merci com o objetivo de conferir padronização à qualidade esperada pelo usuário. A Tabela de Características de Qualidade e a Tabela de Qualidade em Uso foram correlacionadas exemplificando a construção da Matriz de Qualidade em Uso para o produto Merci.

A Tabela de Características de Qualidade é composta pela lista priorizada dos atributos de qualidade, proveniente da Matriz de Qualidade do Usuário. Já a Tabela de Qualidade em Uso é formada pelas características de qualidade em uso definidas na norma ISO-9126, sendo elas: efetividade, produtividade, segurança e satisfação.

A correlação dos atributos de qualidade do Merci com as características de qualidade em uso está mostrada na FIGURA 18.

Alguns atributos de qualidade do produto não estão relacionados com a qualidade em uso de acordo com a norma ISO-9126. Por exemplo, o atributo “Um operador de caixa proficiente em máquina registradora deverá ser capaz de aprender a operar o Merci com 1 dia de treinamento” não se relaciona com nenhuma das características de qualidade em uso da norma ISO-9126. Portanto, não é necessário que todos os atributos estejam relacionados com a qualidade em uso, pois alguns desses estão relacionados com a qualidade externa definida na norma.

Matriz de Qualidade em uso		Características Qualidade em Uso				
Características de Qualidade	Importância	Produtividade	Efetividade	Segurança	Satisfação	Total
Tempo de operação de pesquisa de 10s	90	B				90
Diminuição de erros nas vendas.	66	B	M	B		330
Diminuição do tempo de venda.	58	M				174
Diminuição dos prejuízos na operação de venda	58		M	B		232
Economia de mão-de-obra na operação de venda	54	A				486
Tempo de operação de venda deverá gastar no máximo 2s	42	M				126
Um operador de caixa deverá ser capaz de aprender a operar o Merci com 1 dia de treinamen	38					0
Agilidade na compra e venda de mercadorias.	36	A				324
Diminuição dos erros nas notas fiscais.	34		M	B		136
Restringir o acesso dos usuários às funções através de senhas, conforme o respectivo grupo.	20			A		180
O produto deve ocupar no máximo 200 MB (sem considerar as bases de dados).	20					0
O produto deve executar em 128 MB.	20					0
Eliminação da duplicidade de pedidos.	19		M	B		76
Leiaute do relatório Nota Fiscal deve ser aprovado pela Sec. de Receita.	18					0
Desenhado de forma que possa ser expandido para mais de um terminal de caixa.	0					0
Total		1266	531	357	0	

FIGURA 18 – Matriz de Qualidade em Uso para o produto Merci 1.0

A construção da Matriz de Qualidade em Uso possibilita verificar quais características de qualidade em uso da norma ISO-9126 são mais importantes para o usuário. Com o resultado da matriz, exibido no gráfico da FIGURA 19, percebe-se que a construção do Merci deve priorizar a Produtividade e Efetividade. Mas isso não quer dizer que, por exemplo, a característica de segurança deve ser desconsiderada no desenvolvimento do produto.

No Merci não havia nenhum atributo de qualidade relacionado explicitamente com a característica Satisfação. Isso também pode acontecer com outros produtos, pois a satisfação geralmente faz parte das expectativas do usuário que estão na curva básica, do modelo de Kano. Dessa forma, o usuário não levanta claramente um atributo de qualidade, mesmo esperando que a característica de qualidade esteja contemplada no produto. Portanto, é importante a empresa fornecedora avaliar as características de qualidade zeradas e, se necessário, selecionar medidas de qualidade.



FIGURA 19 – Gráfico de importância das características de qualidade em uso

Outro resultado obtido da *conversão* realizada na Matriz de Qualidade em Uso é a distribuição da qualidade esperada pelo usuário, ilustrada na FIGURA 20. Nesse gráfico, a área destacada demonstra a qualidade em uso priorizada pelo usuário do Merci. Essa priorização possibilita o gerente do projeto distribuir os esforços nas atividades da fase de Construção do produto relacionadas com a qualidade esperada pelo usuário.

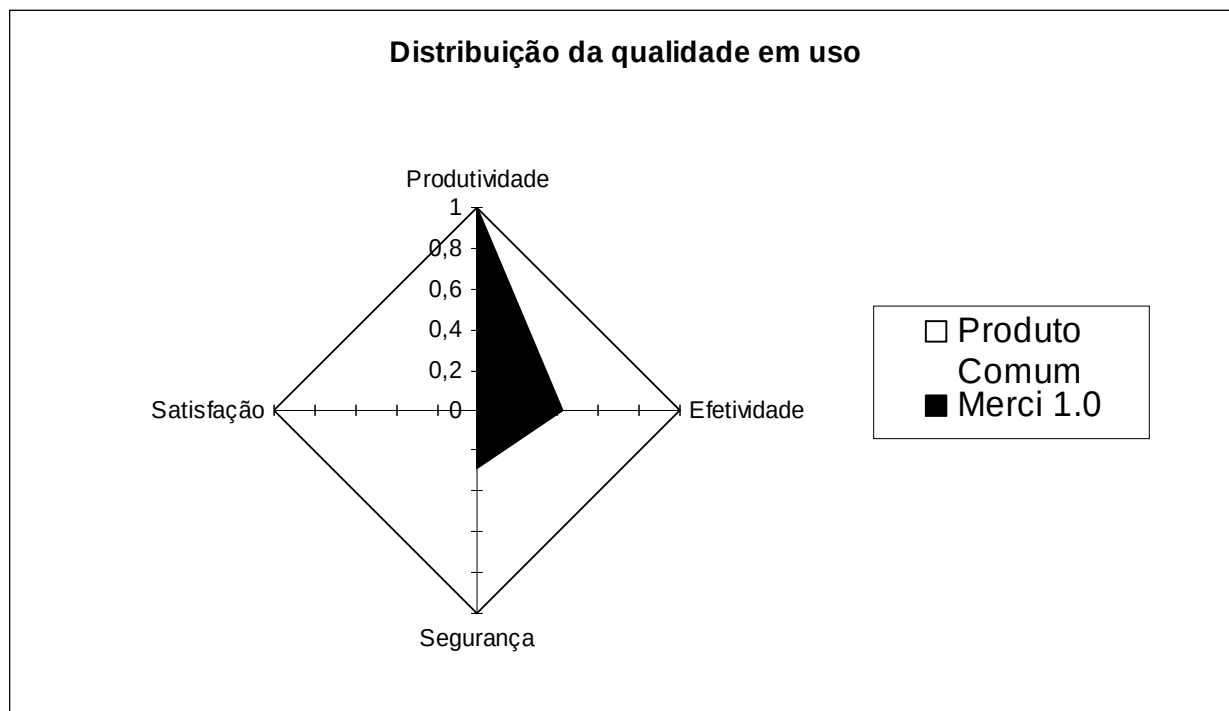


FIGURA 20 – Distribuição da preocupação da qualidade em uso do Merci

A priorização das características da Tabela de Qualidade em Uso também é utilizada para selecionar as medidas da qualidade esperada pelo usuário. Com o grau de importância calculado pela *conversão* da Matriz de Qualidade em Uso e o Nível de Avaliação dessas características é possível priorizar as medidas de qualidade em uso. Por exemplo, a característica Produtividade tem o grau de importância maior que a característica Segurança, sendo assim, devem-se selecionar mais medidas para sua avaliação.

Para identificar o Nível de Avaliação das características de qualidade em uso é avaliado o impacto de uma consequência de uma falha em cada característica. Como o Merci é um produto que controla a movimentação do caixa de uma mercearia, a característica Efetividade e Segurança tem o nível C associado, sendo os níveis para as demais características D, como mostrado na FIGURA 21.

Características Qualidade em Uso	Nível de avaliação
Efetividade	C
Produtividade	D
Segurança	C
Satisfação	D

FIGURA 21 – Níveis de avaliação para características de qualidade em uso

Os atributos de qualidade priorizados, da Tabela de Características de Qualidade, também serão utilizados na construção da Matriz de Qualidade Externa, descrita na seção seguinte.

6.5.3 Matriz de Qualidade Externa

A Matriz de Qualidade Externa é construída para o produto Merci com o objetivo de exemplificar a padronização na qualidade esperada pelo usuário. Essa matriz é composta pela Tabela de Características de Qualidade, proveniente da Matriz de Qualidade do Usuário, e pela Tabela de Qualidade Externa.

A Tabela de Qualidade Externa é formada pelas subcaracterísticas de qualidade externa definidas na norma ISO-9126. Essa tabela é organizada colocando a letra inicial da característica na frente de cada subcaracterística na tabela para mostrar o relacionamento das subcaracterísticas com sua característica.

A *correlação* das tabelas foi realizada executando os passos da instanciação do método, descritos na seção 6.3 deste documento. A *correlação* de todas subcaracterísticas de qualidade para o produto Merci 1.0 está ilustrada na FIGURA 22.

Matriz de Qualidade Externa

Características Qualidade Externa

[illegible]

FIGURA 22 – Matriz de Qualidade Externa para o produto Merci 1.0

O atributo de qualidade “Desenhado de forma que possa ser expandido para mais de um terminal de caixa” está relacionado com três subcaracterísticas de qualidade, porém seu grau de importância é zero, por ser um requisito adiado. Dessa forma, as subcaracterísticas de qualidade correlacionadas com esse atributo ficam com seu grau de importância zero também e não são priorizadas para qualidade externa do produto.

Na Matriz de Qualidade Externa, nenhum dos atributos de qualidade está relacionado com a subcaracterística Interoperabilidade, apesar de que o Merci 1.0 possui integração com o software Sistema financeiro, como demonstrada na FIGURA 23. Nos artefatos disponíveis no Merci, não fica claro se faltou levantar com o usuário os atributos de qualidade para a subcaracterística de interoperabilidade. Porém, a identificação da falta desse atributo correlacionado com a subcaracterística possibilita o arquiteto de um produto como o Merci solicitar esclarecimento dessa dúvida com o usuário. Dessa forma, o método auxiliaria o desenvolvedor a lembrar de características de qualidade importantes para o usuário do produto.

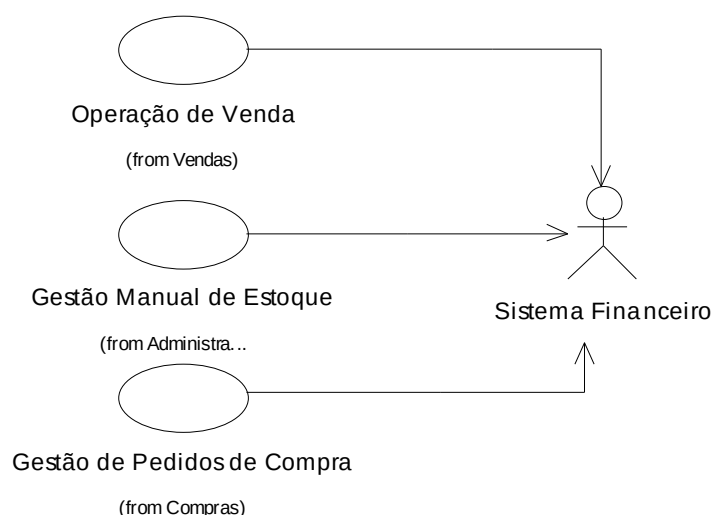


FIGURA 23 – Diagrama de casos de uso do Mercı 1.0 para o ator Sistema Financeiro

Também na matriz da FIGURA 22, nenhum atributo de qualidade é relacionado com as subcaracterísticas da característica Manutenibilidade. Normalmente, usuários de produtos de software não levantam os atributos de qualidade relacionados com essa característica. Essa subcaracterística está mais relacionada com a qualidade desejada pela empresa desenvolvedora do produto, pois problemas com essa característica impactam, por exemplo, o custo de um projeto de manutenção. Nesse caso, o método poderia ser estendido para incorporar a qualidade do produto esperada pela empresa desenvolvedora. Para a incorporação da qualidade esperada pela empresa desenvolvedora no método, no entanto, seria necessário ponderar os atributos de qualidade esperados pelo usuário com os atributos de qualidade esperados pela empresa.

Na tabela da FIGURA 22 pode-se observar que vários atributos de qualidade do produto Mercı não foram relacionados com a subcaracterística Adequação da característica Funcionalidade, semelhante com que aconteceu com a característica Satisfação na Matriz de Qualidade em Uso. Portanto, devem-se ser selecionadas medidas para avaliar a subcaracterística Adequação, mesmo que não tenha nenhum relacionamento na Matriz de Qualidade Externa.

As subcaracterísticas de qualidade foram priorizadas de acordo com sua relação com os atributos de qualidade de software. A priorização dessas subcaracterísticas é mostrada no gráfico apresentado na FIGURA 24 e é utilizada para priorizar as medidas de qualidade coletadas no desenvolvimento. Contudo, é importante salientar que a priorização não significa que as subcaracterísticas de qualidade menos importantes não devam ser mensuradas.

Importância das subcaracterísticas de qualidade externa

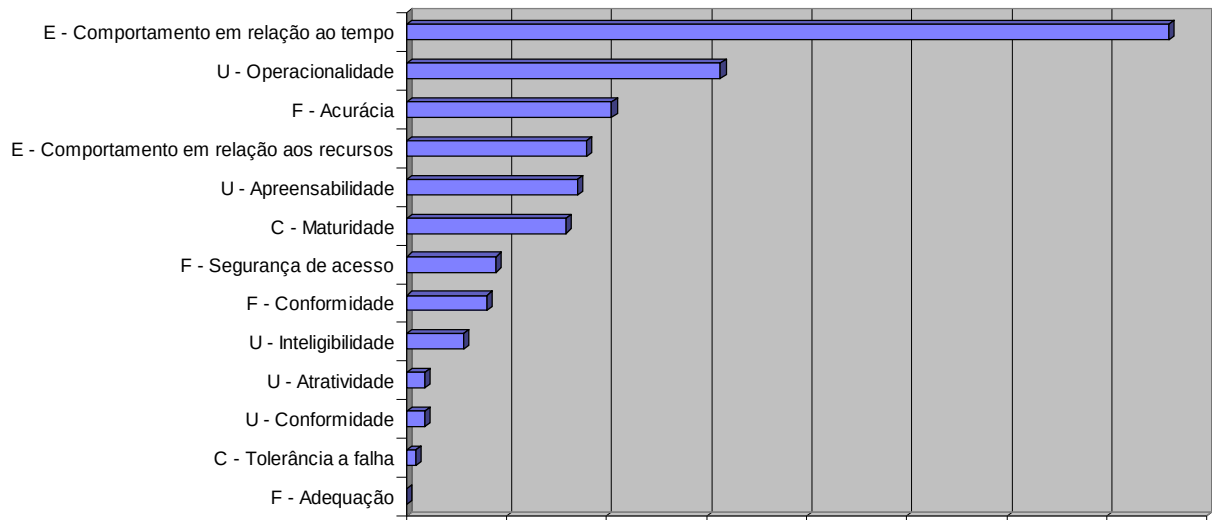


FIGURA 24 – Gráfico de importância das subcaracterísticas priorizadas de qualidade externa

Outro resultado obtido do relacionamento realizado na matriz de qualidade externa é a forma de distribuição da qualidade esperada pelo usuário do Merci. Essa distribuição, ilustrada no gráfico da FIGURA 25, pode-se perceber que a preocupação com qualidade é concentrada em uma área menor do que a área que abrange a qualidade total de um produto de software. Através desse gráfico, o gerente do projeto pode alocar esforço nas atividades de garantia de qualidade que abrangem essa área. É importante lembrar que o gerente não deve esquecer completamente das outras características de qualidade do produto.

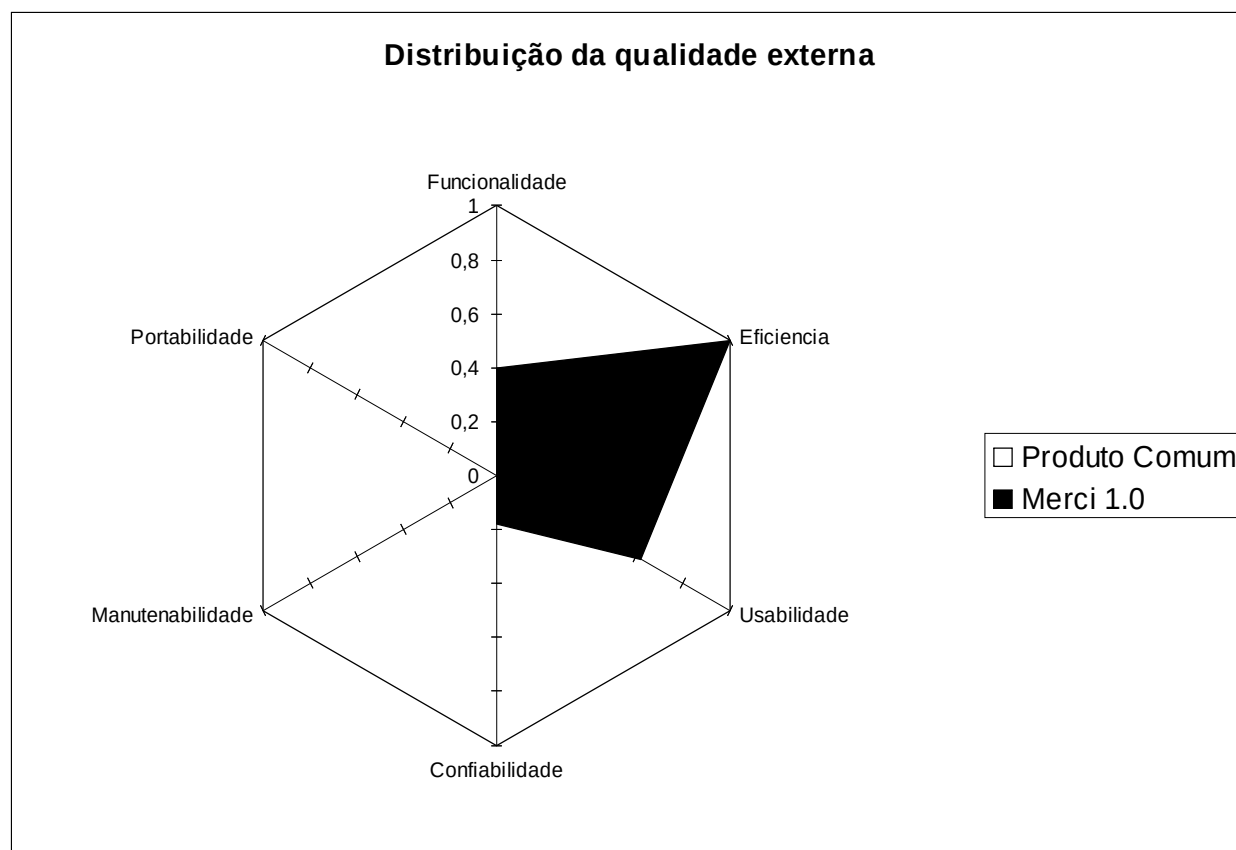


FIGURA 25 – Distribuição da preocupação com a qualidade do Merci

A priorização das subcaracterísticas da Tabela de Qualidade Externa também é utilizada para selecionar as medidas da qualidade esperada pelo usuário. Com o grau de importância calculado pela *conversão* da Matriz de Qualidade Externa e o Nível de Avaliação dessas subcaracterísticas é possível escolher as medidas de qualidade externa. Por exemplo, a característica Eficiência tem o maior grau de importância para o Merci, e assim, deve-se selecionar um maior número de medidas para essa característica. Essas medidas são utilizadas para verificar a qualidade externa do produto Merci no seu desenvolvimento.

Como o Merci é um produto com uma grande interação com seu usuário, as características Funcionalidade, Usabilidade e Eficiência estão associadas com o Nível de Avaliação C, sendo que os níveis para as demais características são D, como mostrado na FIGURA 26.

Características Qualidade Externa	Nível de Avaliação
Funcionalidade	C
Confiabilidade	D
Usabilidade	C
Eficiência	C
Manutenabilidade	D
Portabilidade	D

FIGURA 26 – Níveis de avaliação para características de qualidade externa

Além disso, a Tabela de Qualidade Externa é utilizada para a construção da Matriz de Qualidade Interna do método, descrita na seção seguinte.

6.5.4 Matriz de Qualidade Interna

A Matriz de Qualidade Interna tem o objetivo de auxiliar na priorização das medidas de qualidade interna, que serão coletadas durante o desenvolvimento do produto. Essa matriz é composta pela Tabela de Qualidade Externa, proveniente da Matriz de Qualidade Externa, e a Tabela de Qualidade Interna.

Esta última tabela é formada pelas medidas de qualidade do produto Merci 1.0 e pelas medidas de qualidade interna da norma ISO-9126. As medidas de qualidade do produto Merci são obtidas da aba “Defeitos” do relatório RAPS_w. As medidas de qualidade interna da norma ISO-9126 foram selecionadas de forma que cada subcaracterística de qualidade externa possuísse, pelo menos, uma medida de qualidade interna.

As medidas de qualidade interna selecionadas do produto Merci são:

- Defeitos de Revisões
- Defeitos de Testes de aceitação
- Defeitos de Avaliação pelo usuário
- Defeitos de auditoria de qualidade

Algumas medidas do produto Merci 1.0 são acompanhadas durante o desenvolvimento, porém não estão ligadas diretamente com a qualidade interna, como por exemplo, medidas de tamanho e esforço. Por exemplo, as medidas de tamanho são utilizadas para normalizar os dados das medidas de qualidade coletados entre projetos, não sendo utilizadas para construção da Matriz de Qualidade Interna. Como exemplo para o produto Merci, algumas dessas medidas de tamanho são:

- Tamanho do Código
- Tamanho (Derivadas) Linhas lógicas aplicação por PF
- Tamanho (Derivadas) Casos de testes por PF

- Ponto de Função

Para completar a Tabela de Qualidade Interna foi escolhida pelo menos uma medida de qualidade interna da norma ISO-9126 para cada subcaracterística. O nome dessas medidas foi deixado em inglês para facilitar a busca dessas nas tabelas de definição de medidas da norma ISO-9126, sendo elas:

- E - *Response time*
- U - *Input Validity checking*
- F - *Computational Accuracy*
- E - *Memory utilization*
- U - *Completeness of description*
- C - *Fault removal*
- F - *Access auditability*
- F - *Functional compliance*
- U - *Completeness of user documentation and/or help facility*
- U - *Attractive interaction*
- U - *Usability compliance*
- C - *Failure avoidance*
- F - *Functional adequacy*

As medidas de qualidade interna e as subcaracterísticas de qualidade externa foram correlacionadas para construir a Matriz de Qualidade Interna. As medidas selecionadas do MercI foram relacionadas com as subcaracterísticas de qualidade externa utilizando as informações das suas listas de conferência. Essas listas possibilitam perceber o que foi verificado durante as revisões realizadas no desenvolvimento do MercI 1.0, sendo elas: LCIDESw (Lista Conferência de Inspeção do Desenho Externo de Software), LCIDTSw (Lista Conferência de Inspeção dos Testes de Software), LCIDISw (Lista Conferência de Inspeção do Desenho Interno de Software), LCIIISw (Lista Conferência de Inspeção da Implementação e Desenho Detalhado de Software), LCAUSw (Lista Conferência de Inspeção da Avaliação do Usuário de Software), LCAQSw (Lista Conferência da Auditoria de Qualidade de Software). O relacionamento das medidas de qualidade interna da ISO-9126 com as subcaracterísticas de qualidade externa foi realizado através da sua classificação, descrita na norma.

A relação entre as medidas de qualidade interna e as subcaracterísticas de qualidade externa está demonstrada na Matriz de Qualidade Interna, ilustrada na FIGURA 27.

Matriz de Qualidade Interna

Métricas de qualidade interna

Características Qualidade Externa	Importância	Métricas de qualidade interna																
		E - Response time	Defeitos de Avaliação pelo usuário	Defeitos de Testes de aceitação	U - Input Validity checking	F - Computational Accuracy	E - Memory utilization	U - Completeness of description	C - Fault removal	F - Access audibility	F - Functional compliance	U - Completeness of user doc. and/or help facility	Defeitos de Revisões	U - Attractive interaction	U - Usability compliance	C - Failure avoidance	F - Functional adequacy	Defeitos de auditoria de qualidade
E - Comportamento em relação ao tempo	1524	A		M														
U - Operacionalidade	628		A		A													
F - Acurácia	411			M		A							B					
E - Comportamento em relação aos recursos	360			M			A											
U - Apreensabilidade	342		A					A										
C - Maturidade	319			B					A									
F - Segurança de acesso	180			M						A			B					
F - Conformidade	162			M							A		B					
U - Inteligibilidade	114		A									A						
U - Atratividade	38		A											A				
U - Conformidade	38		A															
C - Tolerância a falha	19			B												A		
C - Conformidade	0																	
C - Recuperabilidade	0																	
E - Conformidade	0																	
F - Adequação	0			M									M				A	
F - Interoperabilidade	0			B														
M - Analisabilidade	0												B					
M - Modificabilidade	0												B					
M - Estabilidade	0																	
M - Testabilidade	0																	
M - Conformidade	0																	
P - Adaptabilidade	0																	
P - Capacidade para ser instalado	0			B														
P - Co-existência	0			B														
P - Capacidade para substituir	0																	
P - Conformidade	0																	
Total		13716	10440	8249	5652	3699	3240	3078	2871	1620	1458	1026	753	342	342	171	0	

FIGURA 27 – Matriz de Qualidade Interna para o produto Merci 1.0

A conversão, realizada na Matriz de Qualidade Interna, e o Nível de Avaliação auxilia na priorização das medidas de qualidade interna. Por simplificação, foram utilizados os mesmos Níveis de Avaliação da Matriz de Qualidade Externa. Essa priorização, ilustrada no gráfico da FIGURA 28, permite o acompanhamento e controle da qualidade com objetivo de alcançar a qualidade externa do produto esperada pelo usuário. Porém, a seleção das medidas internas deve considerar também outros aspectos da organização, como por exemplo, o custo envolvido na coleta das medidas.

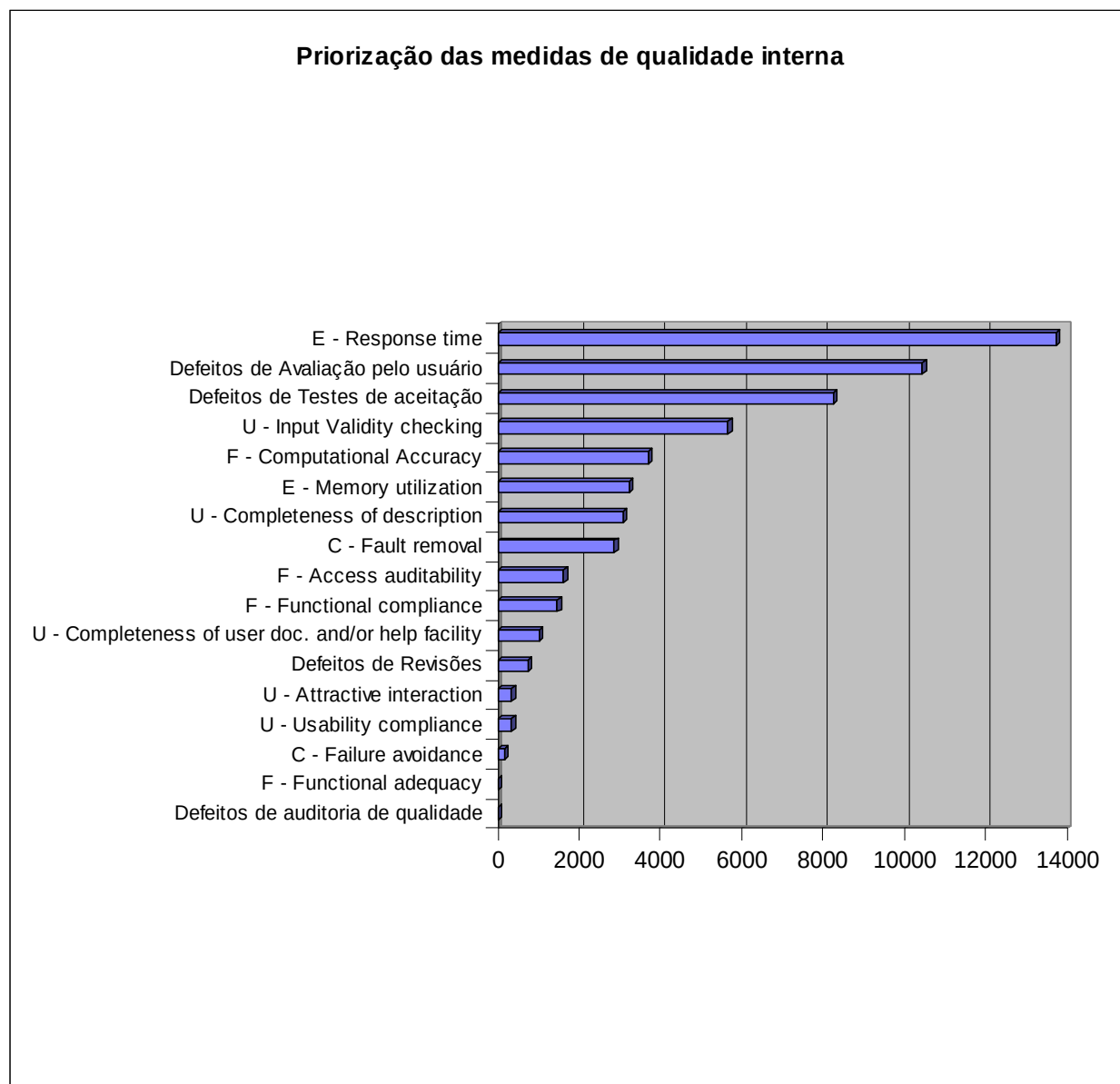


FIGURA 28 – Priorização das medidas internas de qualidade

A *conversão* realizada na Matriz de Qualidade Interna também pode auxiliar a confecção das listas de conferências dos artefatos do Mercú. As listas de conferência são utilizadas para verificar padrões de qualidade dos artefatos confeccionados durante a fase de Construção do produto. Dessa forma, as listas também devem incluir itens que verifiquem a qualidade interna dos artefatos, visando alcançar a qualidade esperada pelo usuário.

Além da confecção das listas de conferência, essa *conversão* pode auxiliar na criação dos casos de testes de unidade. Esses testes de unidade são utilizados para verificar a implementação dos casos de uso do produto. Com a *conversão* realizada na matriz, por exemplo, os testes de unidade podem priorizar a característica de qualidade Eficiência.

A medida “Defeitos de auditoria de qualidade” não foi relacionada com nenhuma

subcaracterística de qualidade, pois ela é uma medida de processo e não uma medida de produto. Contudo, medidas de processo também devem ser coletadas, pois a qualidade do produto depende da qualidade do processo. Porém, neste trabalho de pesquisa o foco de mensuração é a qualidade do produto e não qualidade do processo.

Capítulo 7

Conclusão

A união do método QFD e do padrão de qualidade ISO-9126 possibilitou a criação de um método de aferição de qualidade, que desdobra a qualidade esperada pelo usuário em medidas. Com essas medidas é possível acompanhar a qualidade do produto desde o início da sua construção até a entrega ao cliente. O acompanhamento, através das medidas, auxilia na identificação de falhas no produto e por consequência aumenta a satisfação do seu usuário. Além disso, a mensuração da qualidade desde o início do desenvolvimento antecipa a descoberta de erros, diminuindo o custo de suas correções.

O desdobramento da qualidade esperada pelo usuário através do método QFD possibilita a empresa fornecedora priorizar as atividades de garantia da qualidade. Isto é importante, pois no desenvolvimento de software o investimento em qualidade precisa ser planejado para não extrapolar custos e alcançar seus objetivos de satisfazer as expectativas do usuário. O método proposto auxilia na definição das medidas adequadas a qualidade esperada pelo usuário possibilitando assim uma melhor priorização das atividades de garantia da qualidade.

O modelo de qualidade da norma ISO-9126, utilizado na definição do método deste trabalho, possibilitou uma padronização da qualidade exigida pelo usuário. A qualidade padronizada facilita a criação de uma base histórica de projetos desenvolvidos na empresa. Essa base histórica pode auxiliar na previsão e planejamento de esforços de qualidade para desenvolvimento de projetos futuros.

A utilização da norma ISO-9126 para adaptar o *Framework de Atributos* de Fenton [FENTON96], auxiliou a classificação das medidas utilizadas pelo Praxis e também outras medidas encontradas na literatura. Essa classificação facilitou a sua seleção de medidas realizada pelo método proposto neste trabalho.

A instanciação do método de aferição de qualidade ao processo Praxis, descrita no Capítulo 6, mostra que o trabalho pode ser facilmente incorporado a um processo de desenvolvimento. A instanciação do método ao processo Praxis foi simples, precisando apenas algumas poucas atividades e artefatos. Essas alterações tiveram como objetivo complementar a identificação e acompanhamento da qualidade esperada pelo usuário durante o desenvolvimento, nenhuma alteração conceitual no processo Praxis foi necessária. Portanto acredita-se que a aplicação em grande parte dos outros processos de desenvolvimento seja também simples e facilitada pela exemplificação da aplicação do método que fizemos aqui no Praxis.

Outra contribuição deste trabalho é a unificação do Nível de Avaliação da norma ISO-14598 com a priorização do método QFD. O Nível de Avaliação é utilizado pelo método de aferição de qualidade na seleção das medidas de qualidade do produto. Dessa forma, a seleção das medidas considera a priorização da qualidade esperada, realizada pelo método QFD, e a consequência de uma falha no sistema, calculado pelo Nível de Avaliação. Com o Nível de Avaliação e o grau de importância, o método definido neste trabalho consegue refinar a seleção das medidas de qualidade adequadas ao usuário do produto.

7.1 Trabalhos futuros

Como foi ressaltada na aplicação do método ao Praxis, assim como ocorre com processos de desenvolvimento de software, o método de aferição de qualidade deve ser experimentada, avaliada e continuamente ajustada e melhorada. Por isso, é importante aplicar o método proposto neste trabalho em projetos reais de desenvolvimento, com o objetivo de validar as práticas propostas e identificar possibilidades de melhoria.

Para aplicação do método foram utilizadas planilhas em Excel[®], porém para projetos maiores pode ser necessária a utilização de ferramenta automatizada. As matrizes criadas pelo método podem ficar grandes e de difícil acompanhamento para projetos maiores. Dessa forma, a especificação e a construção de uma ferramenta para o método de aferição de qualidade pode ser uma natural extensão deste trabalho.

Quanto à classificação de medidas, vários outros trabalhos de medidas podem ser classificados e organizados de acordo com o *Framework de Atributos*. Essa extensão do trabalho pode ser realizada com os exemplos realizados no Capítulo 5 e assim aumentar a abrangência de medidas de qualidade de produto de software.

Além da classificação de novas medidas, os resultados do método proposto poderiam ser comparados com outros processos de definição de medidas, como por exemplo, GQM, GQIM ou ISO-14598. Essa comparação poderia avaliar o custo de implantação das metodologias e também a adequação do conjunto de medidas selecionadas.

Referências bibliográficas

- [AKAO96] AKAO, Y. Introdução ao Desdobramento da qualidade. Belo Horizonte: Fundação Christiano Ottoni, 1996. 187 p.
- [ANDERSSON90] ANDERSSON, THORBJÖRN. A survey of software quality metrics. 1990 (citeseer.ifi.unizh.ch/andersson90survey.html).
- [BASILI91] BASILI, V.R. AND MUSA, J.D.. The Future Engineering of Software: A Management Perspective. IEEE Computer Society Press Los Alamitos, CA, USA, 1991
- [BASILI92] BASILI, Victor R. *Software Modeling and Measurement: The Goal/Question/Metric Paradigm*. Technical Report UMIACS-TR-92-96. University of Maryland, 1992.
- [BOEHM78] BOEHM, B.W., BROWN, J.R., KASPAR, H., LIPOW M., MCCLEOD, G.J., AND MERRITT M.J. Characteristics of Software Quality. Amsterdam, North-Holland, 1978.
- [CHENG95] CHENG, Lin C. et. Al. QFD: planejamento da qualidade. Belo Horizonte: UFMG, Escola de Engenharia, Fundação Christiano Ottoni. Editora Littera Maciel Ltda., 1995. 262 p.
- [CMMI02] CMMI PRODUCT TEAM. CMMI for Software Engineering, Staged Representation – CMMI-SW®, version 1.1, Staged. Technical Report CMU/SEI-2002-TR-029, SEI – Software Engineering Institute, Carnegie Mellon, Pittsburg, August 2002.
- [DEMING82] DEMING, W.E. Quality, productivity, and competitive position. Massachusetts Institute of Technology, Center for Advanced Engineering Study Cambridge, MA, 1982.
- [DoD00] USA. Department of Defense. *Practical Software and Systems Measurement; A Foundation for Objective Project Management (P1045/D5.0)*. Washington, D.C.: Department of Defense and US Army, 2000. Available from World Wide Web: <<http://www.psmc.com>>.
- [DROMEY96] DROMEY, G. Cornering the chimera, IEEE Software (January): 33–43, 1996.
- [FEHLMANN02] FEHLMANN, T. Combinatory metrics for software development. QFD Institut Deutschland, 2002.
- [FENTON96] FENTON, Norman E., PFLEEGER, Shari L. *Software Metrics; A Rigorous and Practical Approach*. 2nd ed. London: International Thonson, 1996.
- [GILB96] GILB, TOM. Level 6: Why We Can't Get There from Here. Journal IEEE Software - IEEE Computer Society Press, 1996. pp 97-98.
- [GRADY87] GRADY, R. AND CASWELL, D. Software Metrics: Establishing a Company-Wide Program. Englewood Cliffs, NJ, Prentice-Hall, 1987.
- [HAAG92] HAAG, S.T. (1992), A Field Study of the Use on Quality Function Deployment (QFD) as Applied to SoftwareDevelopment, University of Texas at Arlington, TX.

- [HERZWURM03] HERZWURM, G., SCHOCKERT, S. The leading edge in QFD for software and electronic business. The international Journal of Quality & Reliability Management, v. 20(1), p. 35-55, 2003.
- [IEEE88] INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS. IEEE STD 982.1, IEEE Standard Dictionary of Measures to Produce Reliable Software, 1988.
- [IEEE90] INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS. IEEE STD 610.12, IEEE Standard Glossary of Software Engineering Terminology, 1990.
- [IEEE98] INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS. IEEE STD 1061, IEEE Standard for a Software Quality Metrics Methodology, 1998.
- [ISO01] THE INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. *ISO/IEC 9126; Software engineering - Product quality. Part 1 - Quality Model*, 2001.
- [ISO03A] THE INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. *ISO/IEC 9126; Software engineering - Product quality. Part 2 - External metrics*, 2003.
- [ISO03B] THE INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. *ISO/IEC 9126; Software engineering - Product quality. Part 3 - Internal metrics*, 2003.
- [ISO03C] THE INTERNATIONAL ORGANIZATION FOR STANDARDIZATION - ISO/IEC 15504. Information technology. process assessment. part 1: Concepts and vocabulary, 2003.
- [ISO04] THE INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. *ISO/IEC 9126; Software engineering - Product quality. Part 4 - Quality in use metrics*, 2004.
- [ISO87] THE INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO-9000-9003: Quality Management Systems. International Standards Organization: United Kingdom, 1987.
- [ISO95] THE INTERNATIONAL ORGANIZATION FOR STANDARDIZATION - ISO/IEC 8402. Quality management and quality assurance--vocabulary, 1995.
- [ISO99] THE INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. *ISO/IEC 14598; Information technology - Software product evaluation* . [s.l.], 1999.
- [JACOBSON98] JACOBSON, IVAR, BOOCH, GRADY, RUMBAUGH, JAMES. The Unified Software Development Process. Reading, MA: Addison-Wesley, 1998.
- [JUNG04] JUNG, C. F. Metodologia para pesquisa & desenvolvimento: aplicada a novas tecnologias, produtos e processos. Rio de Janeiro/RJ: Axcel Books do Brasil Editora, 2004.
- [JURAN98] Juran, J. M., Godfrey, A. B. *Juran's Quality Handbook*. 5th ed. McGraw-Hill Professional, 1998. 1872p.
- [KAFURA85] KAFURA, DENNIS. A survey of software metrics. ACM '85: Proceedings of the 1985 ACM annual conference on The range of computing: mid-80's perspective, 1985. pp 502-506.
- [KANO96] KANO, N., SERAKU, N., TAKAHASHI, F. AND TSUJI, S. Attractive quality and must-be quality. In The best on quality, edited by John D. Hromi. Volume 7 of the Book Series of the International Academy for Quality. Milwaukee:ASQC Quality Press, 1996.
- [KUMAR01] KUMAR , AGARWAL, MANISH; YOGESH, S. MALLICK; BRARADWAJ, R. M., ET ALL, Estimating Software projects, ACM SIGSOFT, p.60, 2001

- [LUIGI05] LUIGI LAVAZZA E GIANCARLO BARRESI. Automated support for process-aware definition and execution of measurement plans. ICSE, 2005.
- [MARYOLY03] ORTEGA, MARYOLY, PÉREZ, MARÍA E ROJAS, TERESITA. Construction of a Systemic Quality Model for Evaluating a Software Product. Journal Software Quality Control. Kluwer Academic Publishers Hingham, MA, USA, 2003.
- [MCCALL77] MCCALL, J.A., RICHARDS, P.K., AND WALTERS, G.F. Factors in Software Quality, Vols. I–III, AD/A-049-014/015/055. Springfield, VA, National Technical Information Service, 1977.
- [MCCONNELL96] MCCONNELL, S. Rapid Development: Taming Wild Software Schedules. Microsoft Press, 1996.
- [MIZUNO93] MIZUNO, S. Gerência para Melhoria da Qualidade: As Sete Novas Ferramentas de Controle da Qualidade.:LTC, RJ, 1993.
- [NIST02] THE NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. The Economic Impacts of Inadequate Infrastructure for Software Testing. Maio, 2002. <http://www.nist.gov/>
- [PAI02] PAI, WENG C. A Quality-enhancing software function deployment model. Journal International Journal of Business Information Systems, 2002.
- [PARK96] PARK, Robert E., GOETHERT, Wolfhart B., FLORAC, William A. *Goal-Driven Software Measurement: A Guidebook* (CMU;SEI-96-HB-002). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1996. Available from World Wide Web: <<http://www.sei.cmu.edu/publications/documents/96.reports/96.hb.002.html>>.
- [PAULA03] PAULA FILHO, Wilson de Pádua. *Engenharia de software*; fundamentos, métodos e padrões. 2.ed. Rio de Janeiro: Editora LTC, 2003. 602 p.
- [PAULA06] PAULA FILHO, Wilson de Pádua. Site com material do processo Praxis 2.1 e do produto didático Merci 1.0. (<http://www.wppf.uaivip.com.br/praxis/apresentacao.htm>). Acessado em abril 2006.
- [PEREIRA01] PEREIRA, EDUARDO. Um modelo de medição para processos de desenvolvimento de software. Master's thesis, Universidade Federal de Minas Gerais, 2001.
- [RIGUZZI96] RIGUZZI, FABRIZIO. A survey of software metrics. Universita degli Studi di Bologna, 1996.
- [SHINDO99] HIZAKAZU, SHINDO. “Application of QFD to software and QFD software tools.” Proceedings of the 5ht International Symposium on QFD, 1999
- [TRAVASSOS02] TRAVASSOS, GUILHERME H., DMYTRO GUROV, EDGAR AUGUSTO GURGEL DO AMARAL. Introdução à Engenharia de Software Experimental. Relatório Técnico 2002. (<http://cronos.cos.ufrj.br/publicacoes/reltec/es59002.pdf>) Acessado Maio 2006.
- [ZULTNER90] ZULTNER, R.E. “Software Quality [Function] Deployment. Applying QFD to software”, in QFD Institute (Ed.), Transactions from the 2nd Symposium on Quality Function Deployment, QFD Institute, Novi, MI, pp. 1990 132-49.

Anexo I - Glossário

CUA	Casos de uso de análise
LCAQSw	Lista Conferência da Auditoria de Qualidade de Software
LCAUSw	Lista Conferência de Inspeção da Avaliação do Usuário de Software
LCIDESw	Lista Conferência de Inspeção do Desenho Externo de Software
LCIDISw	Lista Conferência de Inspeção do Desenho Interno de Software
LCIDTSw	Lista Conferência de Inspeção dos Testes de Software
LCIISw	Lista Conferência de Inspeção da Implementação e Desenho Detalhado de Software
MASw	Modelo de análise de software.
Modelo Conceitual	Modelo conceitual é um conjunto de tabelas e matrizes seqüenciadas de forma a permitir a visibilidade das relações existentes entre os componentes, mecanismos, processos, com a qualidade projetada para o produto.
MPPSw	Memória de planejamento de projeto de software
PDSw	Plano de desenvolvimento do software
PESw	Proposta de especificação de software.
PQSw	Plano de qualidade de software
Produto de Software	Um produto de software [IEEE90] é o conjunto completo, ou qualquer parte individual dos itens desse conjunto, de programas de computadores, procedimentos, e a documentação associada e os dados designados para entrega ao cliente ou usuário final.
RAPSw	Relatório de acompanhamento de projeto de software
RISw	Relatório de inspeção de software
RRSw	Relatório de revisão de software
RTSw	Relatório de testes de software