

UNIVERSIDADE DO VALE DO RIO DOS SINOS – UNISINOS
UNIDADE ACADÊMICA DE EDUCAÇÃO CONTINUADA
ESPECIALIZAÇÃO EM QUALIDADE DE SOFTWARE

CASSIA PEREIRA NINO

AVALIAÇÃO DE PRODUTO COMO FERRAMENTA PARA MELHORIA DE SOFTWARE

SÃO LEOPOLDO

2012

Cassia Pereira Nino

Avaliação do Produto como Ferramenta para Melhoria de Software

Trabalho de Conclusão de Curso de Especialização apresentado como requisito parcial para a obtenção do título de Especialista em Qualidade de Software, pelo Curso de Especialização em Qualidade de Software da Universidade do Vale do Rio dos Sinos – UNISINOS

Orientadora: Profa. Ms. Regina Maria Thienne Colombo

São Leopoldo

2012

AGRADECIMENTOS

Obrigada a você, você, vocês e vocês.

"If you have an apple and I have
an apple and we exchange apples then
you and I still each have one apple.

But if you have an idea and I have
one idea and we exchange these ideas, then
each of us will have two ideas."

– George Bernard Shaw

RESUMO

Desde a década de 1950, assistimos a um aumento considerável da demanda de sistemas baseados em computação. E este aumento não significou apenas uma quantidade maior de sistemas para se desenvolver, mas também um aumento na complexidade destes sistemas. Se antes trabalhávamos com sistemas de escopo reduzido, mono usuários e limitados a um *desktop*, hoje temos multi usuário, web, diferentes tipos de dispositivos, novas tecnologias, banco de dados, entre outros. Mas, independente do tamanho ou complexidade destes sistemas, a exigência de qualidade destes é sempre uma constante.

Entre tantas definições de qualidade, podemos dizer que, basicamente, qualidade é estar conforme os requisitos. E um dos problemas de engenharia de software atual é justamente como medir a qualidade de um produto de software de forma satisfatória. Hoje há diversas propostas e normas na área de avaliação da qualidade de software, contudo na sua grande maioria apresentam-se como métodos genéricos. Não avaliam a qualidade do ponto de vista do domínio da aplicação. Para que seja criado um método especializado, devem ser levadas em consideração toda a especificação do software.

Levando-se em consideração a importância da avaliação de produto de software como agente de evolução de sistemas, o presente trabalho visa propor um processo de avaliação contínua de um produto de software customizado, utilizando conceitos já estabelecidos e normatizados dentro de um cenário previamente selecionado. Ou seja, levando-se em consideração as particularidades deste sistema. Também é realizada a execução deste processo, fazendo uso de um modelo de qualidade também construído para as necessidades identificadas no cenário. Com isso, espera-se contribuir para a evolução e aperfeiçoamento do software destacado.

Palavras-chave: avaliação de produto de software, qualidade de produto de software, avaliação de qualidade, melhoria de software.

ABSTRACT

Since the 1950s we have seen a considerable increase in demand for computer-based systems. But the increased demand not only meant a larger amount of systems to develop, but also an increase in the complexity of these systems. Whether people previously worked with reduced-scope, single-user, desktop systems, we now have systems that have to: deal multiple concurrent users, deliver web-enabled access, fit and adapt to different types of access devices and other new technologies, integrate with databases, among others. However, regardless of the size or complexity of these systems, the demand for quality is always a constant.

Among many possible definitions of quality, we can say that it is, basically, to be adherent to requirements. One of the problems that software engineering need to deal with today is, precisely, how to measure the quality of a software product satisfactorily. Nowadays, there are several proposals in the area of standards and of software quality assessment, but mostly appear as generic methods and, hence, do not properly evaluate the quality from the viewpoint of the application domain. In order to establish a specialized method, one should take into account the whole software specification.

Aware of the importance of software product evaluation as a supporting agent of system's evolution, we propose in this work a process for continuous evaluation of a software product, using concepts already established and standardized, within a scenario previously selected and, thus, taking into account the particularities of that system. Moreover, we performed an execution of the proposed process, using a quality model also built for the needs identified in that scenario. With this, we hope to contribute to the evolution and improvement of that deployed software system

Keywords: software product evaluation, software product quality, software assessment, software improvement.

SUMÁRIO

1 INTRODUÇÃO.....	10
1.1 Motivação.....	10
1.2 Definição do Problema.....	12
1.3 Objetivos.....	13
1.4 Metodologia e Organização do Trabalho.....	14
2 FUNDAMENTAÇÃO TEÓRICA.....	15
2.1 Qualidade e Avaliação de Produto de Software.....	15
2.2 Série de Normas NBR ISO/IEC 9126.....	16
2.3 Série de Normas NBR ISO/IEC 14598.....	19
2.4 Série de Normas NBR ISO/IEC 25000.....	21
2.5 Trabalhos Relacionados.....	23
3 PROPOSTA.....	27
3.1 Cenário da Proposta.....	27
3.2 Proposta de processo de avaliação.....	32
3.3 Modelo de Qualidade.....	36
3.4 Métricas e Níveis de Pontuação.....	41
3.4.1 Métricas do tipo Lista de Verificação.....	42
3.4.2 Métricas do tipo Quantificação.....	44
3.4.3 Níveis de Pontuação.....	45
3.5 Determinação de Notas.....	46
3.6 Modelo de Plano de Avaliação.....	48
4 ESTUDO DE CASO.....	53
4.1 Uso do Processo de Avaliação.....	53
4.2 Execução do Processo de Avaliação.....	54
5 CONCLUSÃO.....	64
6 REFERÊNCIAS BIBLIOGRÁFICAS.....	67
APÊNDICE A - Questionário de Requisitos de Avaliação	69
APÊNDICE B - Roteiro de Especificação da Avaliação	71
APÊNDICE C - Guia de Avaliação	73
APÊNDICE D – Tipos de Questões de Avaliação	83

LISTA DE ILUSTRAÇÕES

Figura 1- Partes da NBR ISO/IEC 9126.....	17
Figura 2- Processo de Avaliação segundo a NBR ISO/IEC 14598.....	21
Figura 3- Arquitetura da NBR ISO/IEC 25000.....	22
Figura 4- Componentes do MEDE-PROS.....	25
Figura 5- Componentes da Lista de Verificação do MEDE-PROS.....	25
Figura 6- Processo de Desenvolvimento atual.....	28
Figura 7- Módulos do sistema GPC.....	31
Figura 8- Proposta de Processo de Avaliação.....	34
Figura 9- Estrutura do modelo de qualidade proposto.....	37
Figura 10- Hierarquia de informações na avaliação.....	46
Figura 11- Atividades da definição de notas.....	48
Figura 12- Questões da primeira parte do plano de avaliação.....	49
Figura 13- Questões do roteiro para especificação da avaliação.....	51
Figura 14- Respostas da primeira parte do plano de avaliação.....	54
Figura 15- Respostas da especificação do plano de avaliação.....	55
Figura 16- Atributos do componente Documentação do Usuário.....	56
Figura 17- Atributos do componente Documentação de Desenvolvimento.....	57
Figura 18- Atributos do componente Interface do Usuário.....	58
Figura 19- Atributos do componente Software.....	58
Figura 20- Notas dos Componentes.....	59
Figura 21- Atendimento aos Requisitos.....	61
Figura 22- Notas da avaliação por módulos do sistema.....	61

LISTA DE TABELAS

Tabela 1- Características da Qualidade de Software.....	17
Tabela 2- Subcaracterísticas da Qualidade de Software.....	18
Tabela 3- Características da Qualidade de Software.....	19
Tabela 4- Processo de Avaliação segundo a NBR ISO/IEC 14598.....	20
Tabela 5- Atividades e Responsáveis.....	30
Tabela 6- Itens do modelo de qualidade.....	38
Tabela 7- Tipos de Questões e Respostas.....	42
Tabela 8- Tipo de Questão A.....	43
Tabela 9- Tipo de Questão B.....	43
Tabela 10- Tipo de Questão C.....	43
Tabela 11- Tipo de Questão D.....	44
Tabela 12- Questões para Adequação e Acurácia (Software).....	44
Tabela 13- Notas obtidas para o Sistema.....	62

1 INTRODUÇÃO

1.1 Motivação

Desde os primeiros computadores da arquitetura de von Neumann, na década de 1940, até hoje, podemos perceber uma evolução no desenvolvimento de software. No início, o software era projetado e desenvolvido para hardwares bem específicos. Reutilização e distribuição nem eram levadas em consideração naquele contexto. A engenharia de software era inexistente. Nessas primeiras décadas, o principal desafio era desenvolver um hardware satisfatório.

Contudo, ao longo do tempo, vimos os esforços se concentrarem no desenvolvimento de software e, igualmente, no processo de desenvolver software. E, a partir da década de 1950, os sistemas baseados em computador começaram a ter grandes investimentos. Se antes, tínhamos sistemas limitados e auto-contidos, hoje trabalhamos com multiusuário, bancos de dados, tempo real, sistemas distribuídos, linguagens orientadas a objeto, redes neurais, computação paralela, entre outros (PRESSMAN, 1995). Ou seja, a complexidade de se desenvolver software aumentou bastante. Aumentou-se o escopo dos sistemas, aumentou-se o uso de sistemas e, com isso, aumentou-se também a demanda por softwares de qualidade, pois com o uso contínuo e maciço de sistemas, os problemas de qualidade de software tendem a adquirir mais importância (GOMES, 2008).

Até então, o desenvolvimento de software era tratado de maneira informal, fazendo com que o produto final geral não tivesse a qualidade exigida por seus usuários. Mas, ao longo do tempo, os problemas só foram agravando-se. Pressman (1995) cita três dos principais problemas enfrentados nos anos 1990, mas que soam atuais até hoje:

- O desenvolvimento de software não acompanha o hardware disponível, que está muito além do que nossos sistemas atuais demandam.
- Não conseguimos atender à demanda de desenvolvimento de sistemas.
- É difícil dar manutenção à sistemas existentes, em função do mau projeto de software que foi adotado.

De modo que, começar a pensar em formas otimizadas e assertivas de se

fazer software, tornou-se uma necessidade e realidade. Com isso, a engenharia de software começou a ganhar destaque.

A crescente demanda dos usuários por qualidade e uma visível insatisfação dos mesmos em relação aos sistemas apresentados (REZENDE, 2005) vem fazendo também com que a área de qualidade do software entre nos círculos de discussão não somente no meio acadêmico, mas, principalmente no meio empresarial. E faz também com que novas abordagens sejam elaboradas e implementadas no processo de desenvolvimento de software. Isso porque independentemente do tamanho ou complexidade dos sistemas, a exigência de qualidade destes pelos clientes é sempre uma constante (DÁVILA, 2006). Seja um pequeno aplicativo ou um grande software de gestão, seus usuários não desejam usar algo que apresente erros ou não atenda o que foi pedido. Ou seja, o produto utilizado deve possuir qualidade.

Mas, o que é ter qualidade? Como sabemos se um determinado sistema possui qualidade ou não? Há diversas definições do que seria qualidade de um produto de software, mas pode-se dizer basicamente que, no contexto de software, qualidade é estar conforme os requisitos (GUERRA, 2009).

Contudo, mesmo com definições formais, ainda não é trivial e simples determinar se um produto de software afinal possui ou não qualidade. Ou melhor, qual o nível de qualidade que ele apresenta. Não é suficiente simplesmente questionar o usuário se, na visão dele, o sistema utilizado possui ou não qualidade. Esta abordagem é válida, mas não é completa, tampouco fidedigna. Por exemplo, o usuário pode dar muito importância a aspectos paralelos e deixar outros mais importantes de lado (COLOMBO, 2002). É necessário realizar uma avaliação adequada para que se chegue a uma resposta verdadeira e satisfatória.

A avaliação de um produto de software envolve a atribuição de um valor para este tipo de produto, tornando-se como base seus requisitos explícitos e implícitos (GUERRA, 2009; COLOMBO, 2002). Para tanto, é necessário termos um processo de avaliação definido e um modelo de qualidade a ser seguido. Estas duas ferramentas compõem um método (SANTOS, 2010). Assim, tem-se um guia e parâmetros claros e definidos.

Hoje, já existem alguns modelos de qualidade largamente utilizados. E existem,

igualmente, alguns métodos propostos para atender, de forma genérica, o processo de avaliação de um produto de software, definido pela norma NBR ISO/IEC 14598-1 – Processo de Avaliação de Qualidade de Software (NBR ISO/IEC 14598-1, 2001). Contudo, estes métodos existentes apresentam-se justamente como métodos genéricos. Não avaliam a qualidade do ponto de vista do domínio da aplicação. Para que seja criado um método especializado, deve ser levada em consideração também toda a especificação do software (SANTOS, 2010).

1.2 Definição do Problema

É consenso que um software sem qualidade dificilmente atingirá seus objetivos e deixará seus usuários insatisfeitos. E se este software tiver sido desenvolvido dentro de uma empresa, isso ainda significa para esta empresa uma perda de recursos, de tempo, de dinheiro, é um prejuízo. Por isso a necessidade de se investir em qualidade de produto de software. E hoje não se tem mais a ideia de que qualidade é sinônimo de execução de testes; que são, na sua grande maioria das vezes realizados só no fim da esteira de desenvolvimento de um sistema. O teste é apenas uma das técnicas utilizadas para se garantir a qualidade. Dessa maneira, o verdadeiro problema está em conseguir garantir a qualidade ao longo do processo de desenvolvimento e, principalmente, avaliar corretamente a qualidade de um produto de software.

Uma das dificuldades enfrentadas hoje numa empresa gaúcha da área financeira é como determinar a qualidade de seus produtos de software desenvolvidos internamente e ainda, como garantir essa qualidade ao longo do desenvolvimento e manutenção destes sistemas. O principal sistema (o GPC - Gestão de Projetos de Crédito¹), na visão de negócio, utilizado pela empresa é do tipo Customizado ou FD (*Fully Developed Software*) (categoria definida pela IEEE 1062, 1998) e nenhum plano de avaliação formal desse produto foi jamais realizado. Sem falar que não se tem a cultura de avaliação de produto final, considerando a visão do usuário.

Assim, destaca-se o problema levantado por este trabalho:

1 Adotou-se o nome fictício de “Gestão de Projetos de Crédito” a fim de preservar a integridade do software.

- Como garantir a qualidade de um software customizado ao longo do seu desenvolvimento? E ainda, como avaliar adequadamente a sua qualidade?

1.3 Objetivos

Para este trabalho, tem-se como cenário uma empresa financeira gaúcha, que possui vários sistemas desenvolvidos internamente. Todos estes sistemas foram elaborados pela equipe interna de TI da empresa, contando com o auxílio de equipes terceirizadas de desenvolvimento e testes. O objeto de trabalho será o sistema Gestão de Projetos de Crédito (GPC), um dos sistemas mais importantes, na visão de negócio, da empresa. Este sistema tem como principal escopo o cadastro, análise e aprovação de projetos de crédito de clientes (solicitações de financiamentos). Ele é de uso interno, tendo como usuários apenas os colaboradores da empresa.

Levando-se em consideração a importância da avaliação de produto de software (sejam eles intermediários ou finais) como agente de evolução de sistemas (GUERRA, 2009) almeja-se, com este trabalho, realizar uma avaliação formal deste software, que hoje na empresa tem maiores chances de agregar valor ao negócio. Assim, espera-se contribuir para a evolução e aperfeiçoamento deste sistema.

E, visto que existem diversos métodos que definem processos de avaliação de produto de software, mas que são genéricos, deseja-se também construir, juntamente com a avaliação do software, uma sugestão de processo para a avaliação contínua deste aplicativo. Isso porque o uso de um método para um sistema customizado está intimamente ligado com a situação ao qual ele atende, ou seja, depende do contexto ao qual está.

Portanto, este trabalho tem como objetivo principal:

- Avaliar e melhorar a qualidade do software de forma contínua no seu ciclo de vida, tendo como fim a aplicação prática de uma proposta de processo para avaliação contínua de um produto de software customizado.

Destaca-se ainda como objetivos específicos:

- Realizar um estudo de modelos de avaliação de produto de software.
- Definir um escopo para a melhoria de software.
- Elaborar um plano de avaliação contínua para um software customizado.
- Propor um processo de avaliação contínua que faça uso do plano de avaliação elaborado.
- Executar o plano de avaliação proposto para um software selecionado, gerando um relatório.

1.4 Metodologia e Organização do Trabalho

Os capítulos seguintes refletem a metodologia adotada para o desenvolvimento da proposta.

Para tanto, o trabalho será dividido em três partes. A primeira será uma revisão bibliográfica dos modelos de qualidade e métodos adotados atualmente para avaliação de produto de software. A segunda, será uma adequação e adaptação dos modelos e métodos estudados para o cenário selecionado. E a terceira parte, será a aplicação da proposta em um cenário real da empresa, uma pesquisa-ação, a qual será analisada e comentada no presente trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo tem como objetivo fazer um apanhado dos principais conceitos, normas e projetos relacionados ao escopo do trabalho. Também serão apresentados algumas iniciativas e trabalhos relacionados à avaliação de produto de software.

2.1 Qualidade e Avaliação de Produto de Software

Há várias definições para a palavra qualidade, e no contexto de desenvolvimento de software isso não é diferente. Diversos autores expõem suas ideias e conceitos sobre qualidade de produto de software.

A norma ISO/IEC 9126 define qualidade de software como “A totalidade de características de um produto de software que lhe confere a capacidade de satisfazer necessidades explícitas e implícitas” (NBR ISO/IEC 9126-1, 2003). Necessidades explícitas são objetivos propostos por aqueles que produzem o software, e necessidades implícitas são requisitos subjetivos dos usuários (GOMES, 2008).

Um software com qualidade pode ainda ser dito simplesmente como aquele que não possui defeito (SUEMATSU, 2002). Mas, basicamente, podemos dizer um produto terá qualidade quando atende ao objetivos propostos (GUERRA, 2009).

Contudo, independente da definição de qualidade, um dos principais problemas na engenharia de software é a dificuldade em se medir qualidade de software (GOMES, 2008). Hoje, na área de avaliação de produto de software temos um conjunto de normas e ainda algumas iniciativas de trabalhos com propostas de métodos para se determinar a qualidade de um software. Estas ações vão desde processos completos da atividade de avaliação até recomendações de como medir a qualidade.

A forma mais comum de se expressar a qualidade de um software é através de características de qualidade (PUNTER, 1997). Estas são usadas como atributos, que são devidamente pontuados.

Mas, um problema encontrado é que softwares são, por si só, únicos. Há algumas características em comum, mas cada um possui as suas particularidades e

estas podem influenciar diretamente na qualidade que ele possui. Entretanto, é difícil encontrar modelos prontos que levem em consideração estas particularidades. O que se encontra nas normas e métodos existentes é um modelo genérico, que não avaliam portanto a qualidade do ponto de vista do domínio da aplicação, considerando as funcionalidades, tecnologias e ambientes envolvidos (SANTOS, 2010). Assim, cada plano de avaliação deve ficar vinculado a um software específico, devendo ser customizado para um cenário previamente definido, ou seja, define-se um método de avaliação.

A definição de um método envolve duas constantes: um processo de avaliação e um modelo de qualidade a ser seguido (SANTOS, 2010). Normas e métodos apenas nos dão modelos de partida para a avaliação de um produto de software. Para que os resultados obtidos sejam condizentes com a realidade encontrada, faz-se necessário adaptar tanto o processo quanto o modelo utilizado.

E é durante a utilização e execução do método de avaliação de produto de software que será determinado o quanto um software atende às necessidades propostas (PUNTER, 1997). Ou seja, qual a sua qualidade.

A avaliação de um produto de software pode ter, basicamente, três objetivos (GOMES, 2008):

- para aprimorar o processo de desenvolvimento;
- para avaliar a qualidade do produto;
- para adquirir um software.

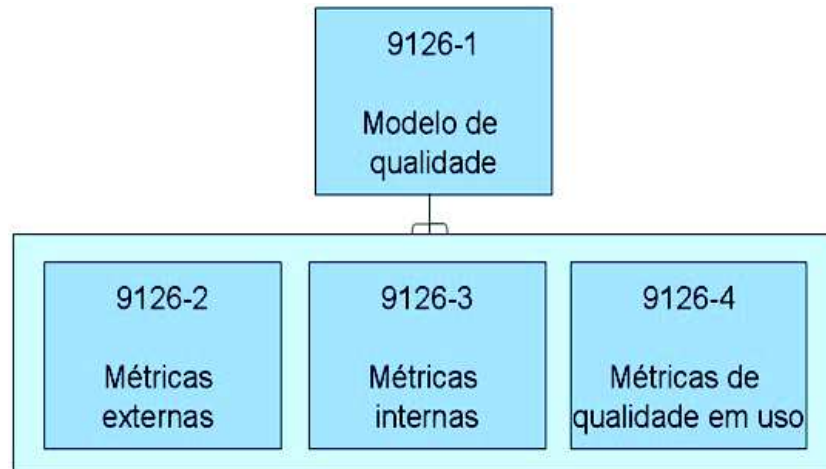
Ao se realizar uma avaliação é importante que o objetivo dela seja bem claro, escolhendo-se uma ou mais funções para tal.

2.2 Série de Normas NBR ISO/IEC 9126

A série de normas NBR ISO/IEC 9126 é nomeada como “Engenharia de Software - Qualidade do Produto” e trata de modelos de qualidade e métricas envolvidas na avaliação da qualidade de produto de software (NBR ISO/IEC 9126-1, 2003).

A série é dividida em quatro partes, como apresentado na figura 1.

Figura 1- Partes da NBR ISO/IEC 9126



Fonte: Nascimento, 2010

Na primeira parte da série, é apresentado um modelo de qualidade para produtos de software, descrito em forma de características de qualidade de software e suas respectivas subcaracterísticas. O modelo é composto por duas partes: qualidade interna e externa (com seis características) e qualidade em uso (com quatro características).

A norma recomenda que, para a avaliação de qualidade de um produto de software, seja definido um modelo de qualidade, ou seja, características de um produto de software que podem ser avaliadas e medidas. Estas características podem ser usadas para especificar requisitos funcionais e não funcionais.

O modelo padrão apresentado na norma para qualidade externa e interna possui seis características que descrevem a qualidade do software, como pode ser visto na tabela 1 (CARNEIRO, 2004).

Tabela 1- Características da Qualidade de Software

Característica	Significado
Funcionalidade	Evidencia o conjunto de funções que atendem às necessidades explícitas e implícitas para a finalidade a que se destina o produto.
Confiabilidade	Evidencia a capacidade do produto de manter seu desempenho ao longo do tempo e em condições estabelecidas.
Usabilidade	Evidencia a facilidade para a utilização do produto.
Eficiência	Evidencia o relacionamento entre o nível de

	desempenho do produto e a quantidade de recursos utilizados, sob condições estabelecidas.
Manutenibilidade	Evidencia o esforço necessário para realizar modificações no produto.
Portabilidade	Evidencia a capacidade do produto de ser transferido de um ambiente para outro.

Fonte: Carneiro, 2004

A norma define qualidade externa como as características do produto de software sob o ponto de vista externo, ou seja, percebida quando o software é executado. Enquanto que a qualidade interna é aquela percebida por quem o desenvolve. As duas são intimamente ligadas, pois a qualidade interna (definida durante a construção do software) influencia diretamente na qualidade externa (percebida pelo usuário).

O modelo apresentado na NBR ISO/IEC 9126-1 está em um formato hierárquico. Assim, as seis características padrão são subdivididas em diversas subcaracterísticas, como pode ser observado na tabela 2.

Tabela 2- Subcaracterísticas da Qualidade de Software

Característica	Subcaracterísticas
Funcionalidade	Adequação
	Acurácia
	Interoperabilidade
	Conformidade
	Segurança de acesso
Confiabilidade	Maturidade
	Tolerância a falhas
	Recuperabilidade
Usabilidade	Inteligibilidade
	Apreensibilidade
	Operacionalidade
Eficiência	Tempo
	Recursos
Manutenibilidade	Analisabilidade
	Modificabilidade
	Estabilidade

	Testabilidade
Portabilidade	Adaptabilidade
	Capacidade para ser instalado
	Conformidade
	Capacidade para substituir

Fonte: NBR ISO/IEC 9126-1, 2003

O modelo de qualidade para a qualidade em uso possui apenas quatro características, sem subdivisões, como pode ser visto na tabela 3.

Tabela 3- Características da Qualidade de Software

Característica	Significado
Eficácia	Evidencia a capacidade do produto de permitir que o usuário atinja suas metas no uso do software.
Produtividade	Evidencia a capacidade do produto de manter seu desempenho ao longo do tempo e em condições estabelecidas.
Segurança	Evidencia a facilidade para a utilização do produto.
Satisfação	Evidencia o relacionamento entre o nível de desempenho do produto e a quantidade de recursos utilizados, sob condições estabelecidas.

Fonte: NBR ISO/IEC 9126-1, 2003

A definição da norma para qualidade em uso é aquela que mede o quanto os usuários podem atingir os seus objetivos utilizando o software.

Dessa maneira temos que a qualidade interna influencia diretamente a qualidade externa, que por sua vez influencia diretamente a qualidade em uso.

A proposta deste modelo é que, através de métricas se possa medir as subcaracterísticas, definindo assim a qualidade de um produto de software.

2.3 Série de Normas NBR ISO/IEC 14598

A série de normas NBR ISO/IEC 14598 trata do processo de avaliação de um produto de software. A ideia é que, com um modelo de qualidade definido (modelo apresentado na NBR ISO/IEC 9126), se planeje e execute uma avaliação. E para

tanto, precisamos de um processo. Basicamente, a norma ISO/IEC 14598 tem seu foco nas métricas e no processo de aplicação destas sobre as características e subcaracterísticas de qualidade definidos na ISO/IEC 9126. Assim, uma define, de forma geral, o que deve ser medido e a outra como este processo de medida deve ser realizado.

Um bom processo de avaliação deve possuir algumas características, como por exemplo: deve ser objetivo e reproduzível (PUNTER, 1997). Isso para garantir a integridade das avaliações que forem executadas a partir do processo definido. Ou seja, para um mesmo produto de software, se o mesmo processo de avaliação for executado por diferentes avaliadores os resultados produzidos devem ser aceitos como semelhantes.

A série NBR ISO/IEC 14598 é dividida em seis partes. A primeira parte apresenta os conceitos envolvidos na avaliação de software e ainda um modelo de processo de avaliação genérico. A segunda e a sexta parte estabelecem itens necessários para o suporte à avaliação. E por fim, os processos de avaliação específicos para desenvolvedores, adquirentes e avaliadores está descrito nas normas 14598-3, 14598-4 e 14598-5 respectivamente.

Os processos sugeridos pela série possuem quatro passos, adaptados para cada tipo de perspectiva (desenvolvedor, adquirente e avaliador). Um resumo destes passos pode ser visto na tabela 4.

Tabela 4- Processo de Avaliação segundo a NBR ISO/IEC 14598

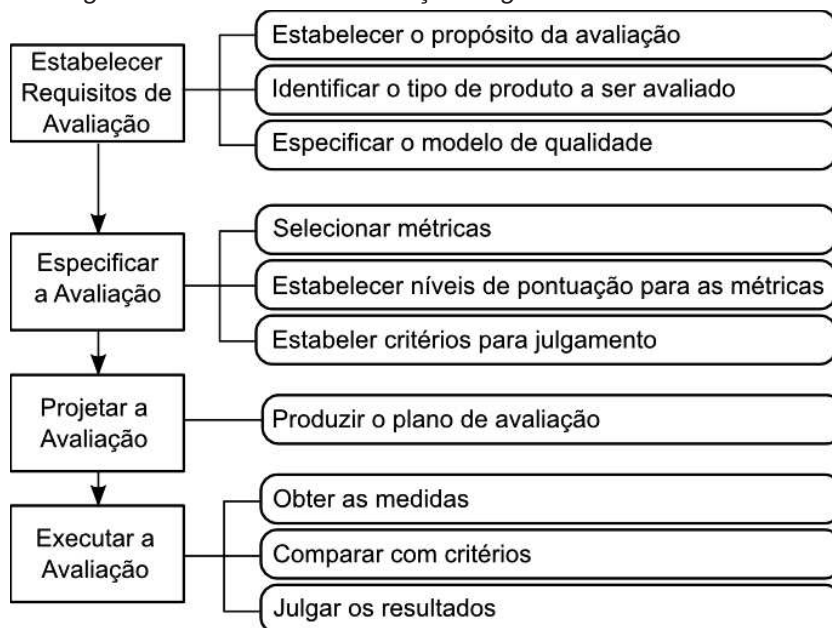
Etapas	Processos		
	Desenvolvedores	Adquirentes	Avaliadores
Estabelecer requisitos de avaliação	Definição de requisitos de qualidade e análise de sua exequibilidade	Estabelecimento do propósito e escopo da avaliação	Descrição dos objetivos da avaliação
Especificar a avaliação	Quantificação dos requisitos de qualidade	Definição de métricas externas e medições correspondentes a serem realizadas	Definição do escopo da avaliação e das medições
Projetar a avaliação	Planejamento da avaliação durante o desenvolvimento	Planejar, programar e documentar a avaliação	Documentação dos processos a serem usados pelo avaliador
Executar a avaliação	Monitoramento da qualidade e controle durante o desenvolvimento	A avaliação deveria ser realizada, documentada e analisada	Obtenção dos resultados a partir da realização de ações de medição e verificação

			do produto
--	--	--	------------

Fonte: Carneiro, 2010

O modelo de processo de avaliação genérico apresentado na primeira parte da série aborda quatro etapas divididas em atividades. A figura 2 mostra o esquema das atividades envolvidas no processo de avaliação.

Figura 2- Processo de Avaliação segundo a NBR ISO/IEC 14598



Fonte: NBR ISO/IEC 14598-1

Ressalta-se que este modelo de processo visa atender a maioria dos casos de avaliações. Contudo, um processo pensado para um cenário específico pode omitir etapas e atividades, se assim for conveniente.

2.4 Série de Normas NBR ISO/IEC 25000

Apenas para relatar o estado atual das normas de qualidade e avaliação de produto de software, será abordado brevemente a série de normas 25000.

A série de normas NBR ISO/IEC 25000 ainda está em desenvolvimento e tem como um dos objetivos absorver as normas NBR ISO/IEC 9126 e NBR ISO/IEC 14598, de modo que estas duas hoje estão válidas, mas são consideradas como “em transição” (NBR ISO/IEC 25000). A série 25000 é denominada Requisitos e

Avaliação da Qualidade de Produto de Software (SQuaRE).

A figura 3 mostra um esquema da arquitetura da série.

Figura 3- Arquitetura da NBR ISO/IEC 25000

ISO/IEC 2503n Requisitos de Qualidade	ISO/IEC 2501n Modelo de Qualidade	ISO/IEC 2504n Avaliação da Qualidade
	ISO/IEC 2500n Gestão da Qualidade	
	ISO/IEC 2502n Medição da Qualidade	

Fonte: NBR ISO/IEC 25000, 2008

Cada uma das partes acima é um conjunto de normas relacionadas. Dessa maneira as normas da divisão de Gestão da Qualidade tem por objetivo definir os modelos e conceitos utilizados pelas outras divisões da série SQuaRE.

A divisão Modelo de Qualidade tem como responsabilidade propor dois modelos de qualidade que serão utilizados nas avaliações, um para a qualidade interna e externa, e o outro para qualidade dos dados pertencentes ao sistema. Esta divisão tem correspondência direta com a norma NBR ISO/IEC 9126.

A divisão de Medição da Qualidade contém normas que apresentam um modelo de referência para medição da qualidade de produto de software. E ainda orientações para sua aplicação.

A divisão de Requisitos de Qualidade aborda a especificação de requisitos de qualidade de um produto de software, que poderão ser utilizados posteriormente como entrada para um processo de avaliação.

Por fim, as normas que fazem parte da divisão de Avaliação de Qualidade apresentam sugestões de processos de avaliação de produto de software, sob várias perspectivas (desenvolvedores, adquirentes e avaliadores). Ainda são fornecidas orientações e recomendações para a definição e uso dos processos.

Esta junção de normas sobre qualidade e avaliação de produto de software em uma só série tem como objetivo unificar e organizar de forma lógica informações que hoje são dependentes, mas que estão espalhadas. Ou seja, tem-se um modelo de referência geral, ao se falar de qualidade e avaliação de produto de software.

2.5 Trabalhos Relacionados

Há várias publicações e iniciativas na área de avaliação de produto de software descrevendo desde modelos até processos de avaliação e métodos (modelos e processos). A grande maioria usa as referências das normas de qualidade e avaliação de produto de software (descritas nas seções anteriores), adaptando-as de acordo com suas necessidades.

Franco (2011) apresenta a execução de uma avaliação de um software do tipo COTS (*Commercial Off-The-Shelf-Software*, conhecido também como “software de prateleira”). A avaliação consistiu simplesmente em verificar se o software escolhido atendia à norma NBR ISO/IEC 9126. Para tanto, foram respondidas algumas perguntas propostas por GOMES (2008), correspondentes a cada uma das seis características de qualidade descritas na norma. Ao fim da avaliação não se obteve uma resposta sobre a qualidade do produto, mas apenas indicou-se que, em função das respostas encontradas em cada uma das questões, o software atendia à norma. Este tipo de avaliação é válido, mas não serve para construir uma base histórica de dados sobre a qualidade do produto, por exemplo. Dessa maneira, não é possível acompanhar a evolução da sua qualidade.

Em Dávila (2006) é apresentado um processo para determinar as características de qualidade que serão utilizadas em uma avaliação, bem como a sua importância (através de pesos). Eles propõem uma abordagem em três passos:

- Decompor o produto de software em componentes (é sugerido utilizar a ferramenta WBS – *Work breakdown structure*) e selecionar os mais importantes.
- Definir os pesos para as características e subcaracterísticas do modelo de qualidade (em função dos componentes selecionados) usando um questionário para vários participantes (usuários e desenvolvedores).
- Definir os níveis de qualidade esperado nos atributos do produto de software. Para tanto, também são utilizados questionários com usuários e desenvolvedores.

Com isso, é possível obter um mapa de pesos que envolve todos os componentes, características e subcaracterísticas. Dessa forma, a determinação da

qualidade levará em conta as necessidades de um software em específico.

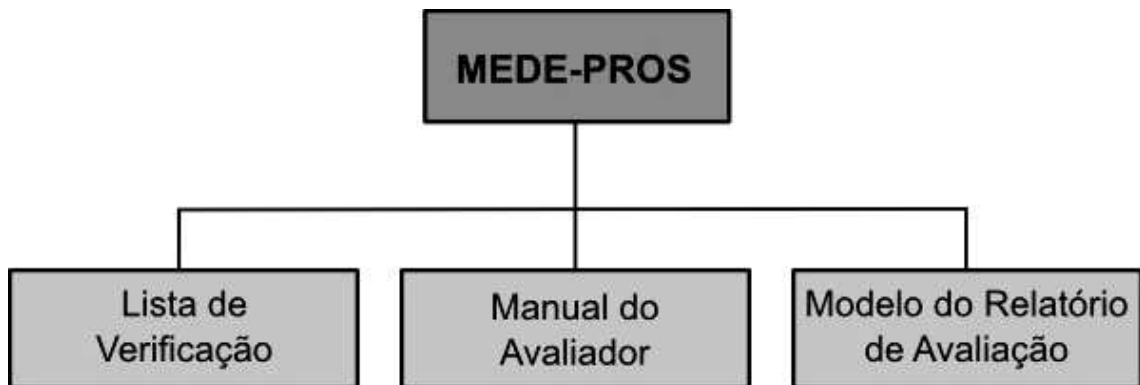
A RIOSOFT também apresenta um trabalho na área de avaliação de produto de software voltado para micro e pequenas empresas de TI. Em sua página (RIOSOFT, 2012), disponibiliza um método de avaliação genérico, que faz parte do programa Qualiteste, que visa auxiliar estas empresas no desenvolvimento de sistemas com qualidade. Este método também faz uso de normas como a NBR ISO/IEC 9126 e NBR ISO/IEC 14598. Todo o método é disponibilizado de forma gratuita para as próprias empresas realizarem uma auto-avaliação, mas é possível ainda contar com o auxílio da própria RIOSOFT para uma avaliação formal, que pode ter como produto final uma certificação de qualidade emitida pela RIOSOFT.

O método apresentado pela organização é baseado em questionários de avaliação e também leva em consideração atividades de teste realizadas ao longo do processo de desenvolvimento. Por exemplo, um item na avaliação é a entrega dos casos de testes realizados.

O Portal do Software Público Brasileiro também tem uma iniciativa para avaliação e certificação de qualidade, baseado nas normas vigentes de qualidade de software. É apresentado um processo de avaliação com um Modelo de Qualidade de Produto de Software (SPB, 2012). A avaliação é realizada através do preenchimento de questões e medições em função das características e atributos do modelo. O objetivo desse trabalho é avaliar os softwares apresentados no portal (sistemas desenvolvidos na Administração Pública e softwares públicos desenvolvidos por terceiros), de modo a qualificar estas soluções disponibilizadas. Dessa maneira, o processo proposto visa atender especificamente este tipo de software, com suas particularidades.

Outro trabalho que merece destaque é o MEDE-PROS (Método de Avaliação de Qualidade de Produto de Software), desenvolvido no CenPRA (Centro de Pesquisas Renato Archer) – uma instituição do Ministério da Ciência e Tecnologia, que também é baseado nas normas ISO de qualidade de software (MARTINEZ, 1999). Este trabalho apresenta um método de avaliação genérico, não especializado para nenhuma área de domínio nem fazendo distinção do tipo de software avaliado, por exemplo. O método é dividido em três partes, como mostra a figura 4.

Figura 4- Componentes do MEDE-PROS



Fonte: GUERRA, 2009.

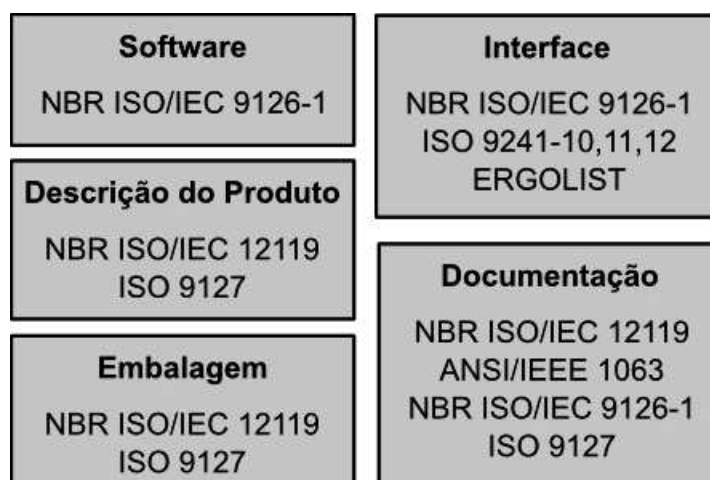
A Lista de Verificação contém um roteiro para a avaliação do software. Essa lista possui todas as questões que serão utilizadas na avaliação e deve ser seguida pelos avaliadores.

O Manual do Avaliador contém instruções para os avaliadores executarem corretamente a avaliação. Este é um mecanismo utilizado para tentar eliminar a subjetividade da avaliação, garantindo que qualquer avaliador tenha um mesmo entendimento das questões envolvidas na avaliação.

Por fim, o Modelo do Relatório de Avaliação apresenta um modelo para o relatório final de avaliação, que deve ser elaborado pelo avaliador ao término do processo de avaliação.

O método MEDE-PROS construiu sua lista de verificação observando as melhores práticas e normas, definindo cinco componentes, como pode ser visto na figura 5.

Figura 5- Componentes da Lista de Verificação do MEDE-PROS



Fonte: GUERRA, 2009

Em um processo de avaliação, cada um destes componentes é levado em consideração e possui um conjunto específico de questões relacionadas.

3 PROPOSTA

Este capítulo apresentará o cenário no qual o trabalho foi baseado, a proposta de processo de avaliação contínua e o plano para avaliação para um software customizado.

3.1 Cenário da Proposta

A proposta apresentada no presente trabalho foi pensada para um cenário específico: o setor de TI de uma instituição financeira. Para um bom entendimento da aplicação da proposta será descrito aqui o cenário atual encontrado na empresa.

A empresa foi criada em 1997 e sempre contou com uma equipe reduzida de colaboradores na área de TI. Hoje, a empresa tem um quadro de cerca de 200 colaboradores, e sua equipe de sistemas conta com quatro Analistas de Negócios que também têm o papel de Analista de Sistemas.

O papel principal da equipe de TI é manter os sistemas internos da empresa, visto que a maioria destes sistemas têm desenvolvimento próprio. Assim, há um trabalho grande de atualizações e melhorias nos sistemas já existentes.

Este desenvolvimento sempre foi realizado por equipes terceirizadas em função do baixo número de recursos na área de sistemas. Hoje, existem uma fábrica de desenvolvimento e uma fábrica de testes alocadas fisicamente na empresa, que correspondem a duas empresas externas diferentes. Este modelo está sendo aplicado desde julho de 2010. Anteriormente, o trabalho era realizado também por uma empresa terceirizada, mas no formato de *body shop* (onde a empresa contratante fica responsável pela gerência de cada um dos profissionais alocados).

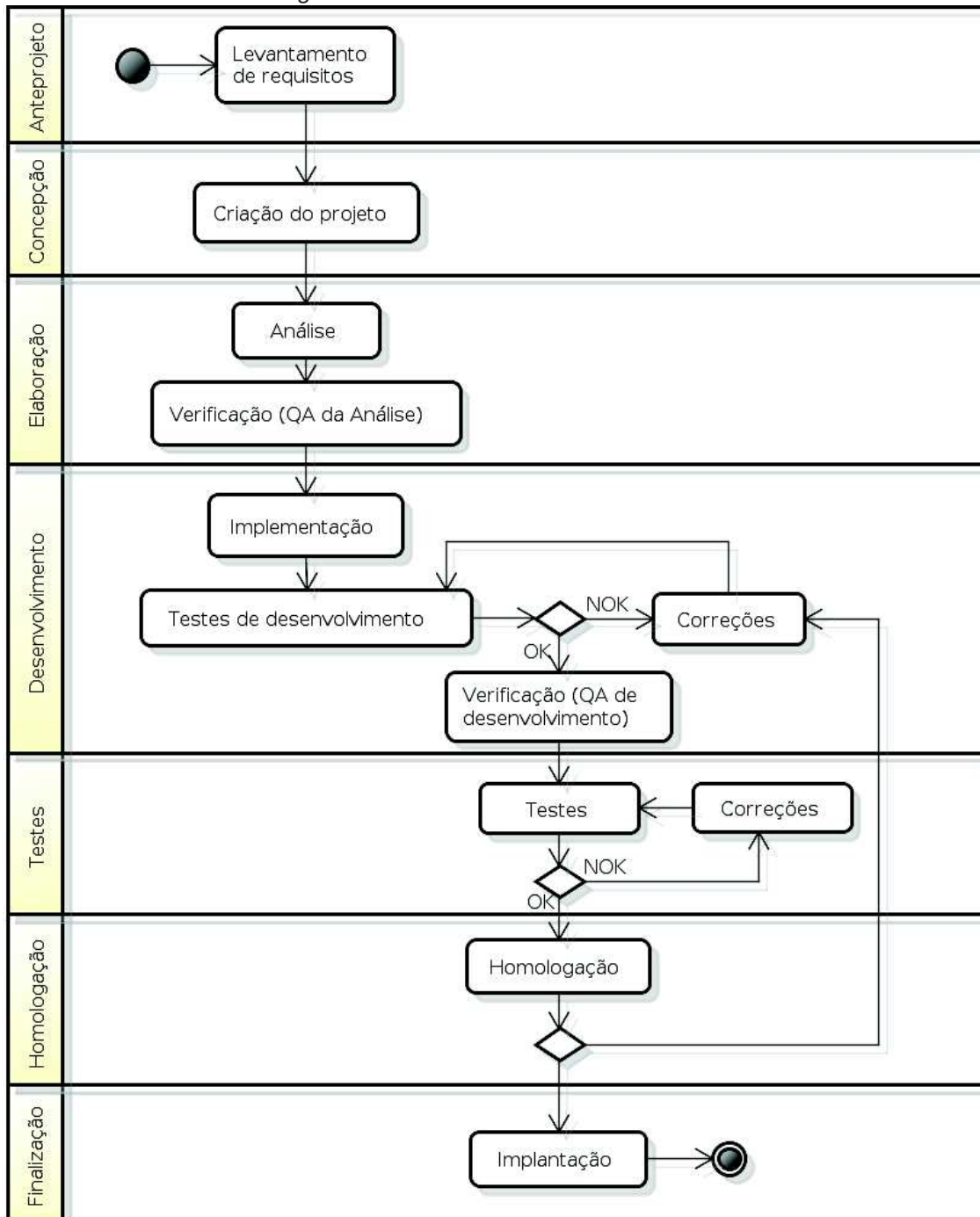
A fábrica de desenvolvimento conta com uma equipe composta pelos seguintes papéis: Gerente de Fábrica, Analistas de Sistema, Desenvolvedores e Testadores. A fábrica de testes possui uma equipe composta por Analistas de Teste e Testadores.

O modelo de desenvolvimento é por projetos de escopo reduzido, com o objetivo de se ter velocidade nas entregas. Praticamente todos os projetos podem ser classificados como de melhoria em sistemas já existentes. Estes projetos têm origem nas demandas feitas pelos usuários e sempre têm um Analista de Negócios

responsável. Este ator deve encaminhar o projeto para a fábrica de desenvolvimento e acompanhá-lo até sua implantação, servindo como canal de comunicação entre a fábrica e os usuários.

A figura 6 abaixo mostra um diagrama simplificado deste processo de desenvolvimento, dividido por etapas.

Figura 6- Processo de Desenvolvimento atual



Fonte: Elaborada pela autora

Ressalta-se o fato de que este processo foi documentado e institucionalizado pela primeira vez há pouco mais de um ano, em abril de 2011. Anteriormente, não haviam processos documentados ou ao menos definidos.

Abaixo estão descritas brevemente as macro-atividades apresentadas no processo:

- Levantamento de requisitos: atividade onde os Analistas de Negócios realizam reuniões com os usuários para levantar requisitos e propor soluções de sistema. Eventualmente, Analistas de Sistemas da fábrica de desenvolvimento acompanham os Analistas de Negócios nas reuniões.
- Criação do projeto: aqui os Analistas de Negócios definem o escopo do projeto que será encaminhado para a fábrica de desenvolvimento. Este projeto é encaminhado junto com uma especificação das funcionalidades envolvidas.
- Análise: primeira atividade realizada pela fábrica de desenvolvimento, deve se basear nas especificações repassadas. Esta atividade tem como produtos uma proposta de cronograma com estimativa de esforço e ainda uma especificação detalhada do que será realizado, em forma de casos de uso (com definições de regras de negócio e protótipos).
- Verificação (QA da Análise): após a finalização da atividade de análise, o Analista de Negócios responsável pelo projeto deve revisar a documentação gerada. Somente após o aceite do Analista de Negócios é que a implementação é iniciada.
- Implementação: com base na especificação gerada, os desenvolvedores projetam e implementam a solução.
- Testes de desenvolvimento: esta atividade é realizada pela fábrica de desenvolvimento também e consiste em testes funcionais feitos por um testador em conjunto com o desenvolvedor.
- Correções: esta atividade é realizada pelos desenvolvedores da fábrica de desenvolvimento sempre quando inconformidades forem encontradas após as atividades de teste de desenvolvimento, teste e homologação.

- Verificação (QA do Desenvolvimento): mais uma atividade de controle onde ocorre a revisão dos produtos da fase de desenvolvimento pelos Analistas de Negócios. Somente após o aceite do Analista de Negócios é que a fase de testes é iniciada.
- Testes: esta atividade, na verdade agrega mais de uma tarefa: elaboração e execução de casos de teste. Os testes realizados são funcionais e são baseados na documentação gerada ao longo do projeto e foca-se nas alterações realizadas.
- Homologação: a atividade de homologação é dividida em duas partes. Numa primeira parte, somente o Analista de Negócios responsável pelo projeto homologa a nova versão do sistema. Após a aprovação do Analista de Negócio é que este irá homologar junto com os usuários envolvidos as novas funcionalidades.
- Implantação: após o aceite dos usuários, a nova versão do sistema é implantada pela equipe de infra-estrutura da empresa, com o auxílio da fábrica de desenvolvimento.

Na tabela 5 pode ser visto um breve resumo dos responsáveis das atividades.

Tabela 5- Atividades e Responsáveis

Atividade	Responsável
Levantamento de requisitos	Analistas de Negócio
Criação do projeto	Analistas de Negócio
Análise	Fábrica de Desenvolvimento
Verificação (QA da Análise)	Analistas de Negócio
Implementação	Fábrica de Desenvolvimento
Testes de desenvolvimento	Fábrica de Desenvolvimento
Correções	Fábrica de Desenvolvimento
Verificação (QA de Desenvolvimento)	Analistas de Negócio
Testes	Fábrica de Testes
Homologação	Analistas de Negócio
Implantação	Fábrica de Desenvolvimento e Equipe de Infra-Estrutura

Fonte: Elaborada pela autora

Por se ter o desenvolvimento todo terceirizado, optou-se por ter pontos de verificação ao longo do processo (após o término da fase de análise e de desenvolvimento). Assim, consegue-se ter um acompanhamento regular dos projetos desenvolvidos. E ainda, facilita-se a passagem de conhecimentos.

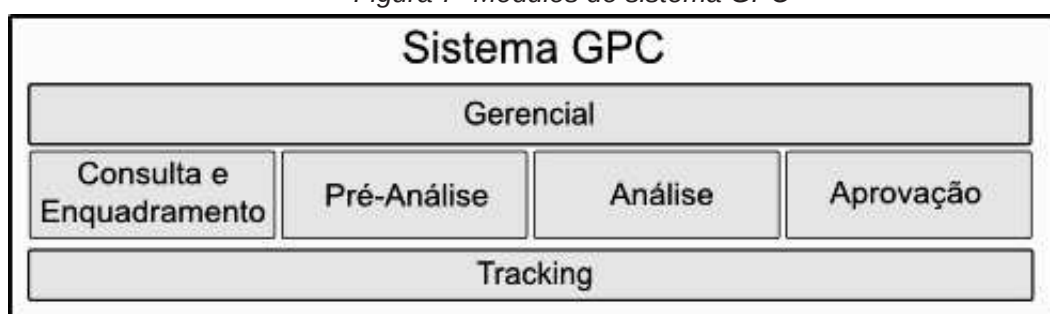
Contudo, apesar de terem sido criados pontos de verificação ao longo do processo de desenvolvimento, ainda observa-se uma qualidade abaixo das expectativas nos produtos de software apresentados. Por vezes algumas funcionalidades não são devidamente implementadas ou ainda, há uma discrepância entre o que é entregue e o que o usuário espera de resultado.

Desta maneira, surge a necessidade de haver um plano de avaliação contínua dos sistemas desenvolvidos, que não sejam apenas verificações pontuais ou testes isolados.

Como já comentado anteriormente, a proposta de processo sugerida por este trabalho tem como cenário uma instituição financeira e basear-se-á no seu sistema Gestão de Projetos de Crédito (GPC). Assim, tanto o plano de avaliação, quanto a proposta de processo de avaliação contínua são direcionados para este sistema em específico. Contudo, ressalta-se o fato de que a proposta de processo de avaliação pode ser adaptada a qualquer sistema da empresa.

O sistema GPC tem como escopo o cadastro e análise de projetos de crédito, até a sua contratação. Possui basicamente seis módulos (figura 7).

Figura 7- Módulos do sistema GPC



Fonte: Elaborada pela autora

O módulo de "Consulta e Enquadramento" permite que sejam cadastradas as propostas dos clientes para financiamentos. Neste módulo também é realizado o enquadramento das propostas, ou seja, a sugestão de produtos (linhas de financiamentos) que melhor atendem ao cliente.

No módulo de “Pré-Análise” é realizada uma análise preliminar do projeto que o cliente apresentar. Caso o projeto seja aceito, são iniciadas as atividades no módulo de “Análise”. Neste módulo, um grupo de Analistas (das mais diversas áreas, como por exemplo: Contabilidade, Economia, Engenharia, Jurídica) realizam uma avaliação profunda do projeto considerando seus objetivos, justificativa, análise econômico-financeira, etc.

Após o término das atividades de “Análise”, os projetos passam por alçadas de aprovações, os Comitês. Para tanto, o módulo de “Aprovação” gerencia o registro das reuniões dos Comitês de aprovação e os certificados gerados durante as reuniões, onde constam os pareceres do Comitê (aprovação ou não dos projetos).

Junto com estas funcionalidades ainda há o módulo gerencial que provê informações gerenciais através de relatórios e o módulo de *Tracking*, que tem como responsabilidade armazenar históricos de alterações dos registros do sistema.

O desenvolvimento do sistema iniciou em 2006 e conta hoje com 109 requisitos registrados, 117 casos de uso documentados e ainda está em evolução. Nem todos os requisitos já levantados para o sistema foram desenvolvidos em sua totalidade e tampouco se sabe exatamente o percentual de atendimento do sistema a estes requisitos. Isso porque nunca foi realizada uma avaliação completa do sistema a fim de obter estes dados. A quantidade de requisitos também não é imutável, uma vez que, por ser uma empresa financeira, esta é regida por normas de diversos órgãos (como por exemplo, o Banco Central², o BNDES³, etc). Sem falar das novas funcionalidades percebidas pelos usuários ao longo da utilização do sistema.

É interessante registrar também o fato de que os casos de uso começaram a ser documentados juntamente com o início da institucionalização do processo de desenvolvimento. Ou seja, cerca de cinco anos após o início do desenvolvimento do sistema. Hoje, estima-se que 90% do sistema já esteja documentado em casos de uso.

3.2 Proposta de processo de avaliação

Com base nos modelos de qualidade estudados e no processo atual de

2 <http://www.bcb.gov.br>

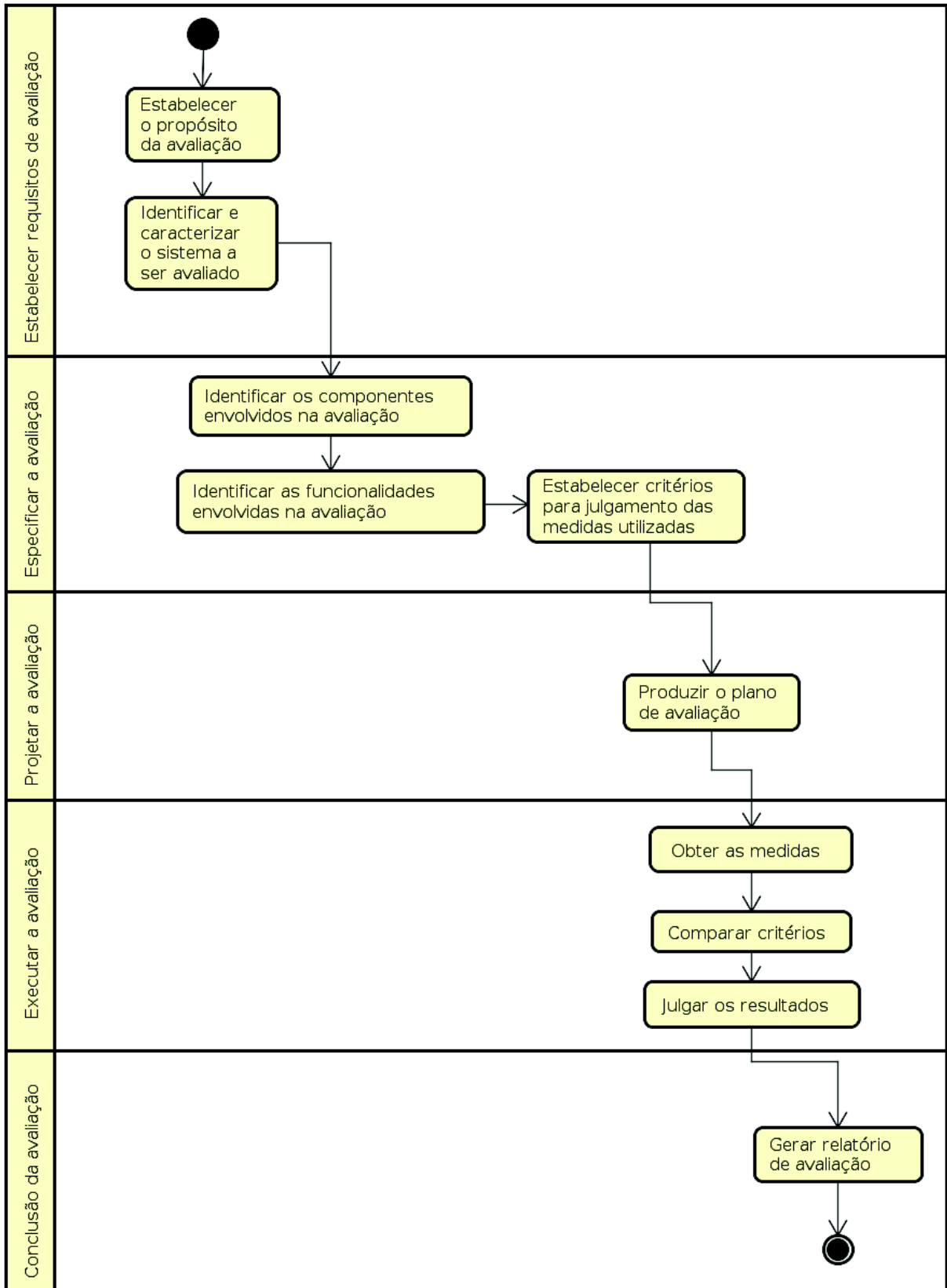
3 <http://www.bndes.gov.br>

desenvolvimento de software da empresa, propõe-se um plano de avaliação contínua para seus sistemas. Este processo foi desenhado baseado na NBR ISO/IEC 14598-5 e no MEDE-PROS (GUERRA, 2009), levando-se ainda em consideração a realidade da empresa. A figura 8 mostra as etapas e tarefas do processo proposto.

Abaixo estão descritas as etapas (E) e tarefas (T) envolvidas no processo:

- (E) Estabelecer requisitos da avaliação: nesta etapa a meta é definir o escopo da avaliação e caracterizá-la. Como saída a etapa tem o questionário apresentado no Apêndice A.
- (T) Estabelecer o propósito da avaliação: esta tarefa corresponde aos itens 8, 9 e 10 do questionário e define o objetivo da avaliação.
- (T) Identificar e caracterizar o sistema a ser avaliado: embora a proposta do trabalho seja específica para um produto, destaca-se que o processo pode ser aplicado a qualquer sistema da empresa, adaptando-se poucos itens. Assim, nesta atividade, o objetivo é somente identificar o sistema que será objeto da avaliação. Esta tarefa corresponde aos itens 2 a 7, 11 e 12 do questionário
- (E) Especificar a avaliação: nesta etapa são especificados os itens que irão compor a avaliação. Há um conjunto básico de informações que é comum a todas as avaliações (ver Apêndice A), contudo é necessário definir os componentes que serão avaliados e ainda os itens que são específicos do sistema em questão. Ou seja, são levados em consideração os requisitos do sistema. Como produto final desta atividade, será preenchido o roteiro apresentado no Apêndice B.
- (T) Identificar os componentes envolvidos na avaliação: o modelo de qualidade utilizado pelas avaliações (descrito na seção 3.3) é composto por quatro componentes. Nesta atividade do processo, devem ser selecionados os componentes que farão parte do escopo da avaliação. É importante frisar que a seleção dos componentes está fortemente ligada aos objetivos já definidos para a avaliação, atividade realizada na etapa anterior.

Figura 8- Proposta de Processo de Avaliação



Fonte: Elaborada pela autora

- (T) Identificar as funcionalidades envolvidas na avaliação: nesta tarefa, devem ser listadas as funcionalidades do sistema que serão levadas em consideração na avaliação. Não é regra que uma avaliação seja feita para todo o sistema, por exemplo. A medida de tamanho de funcionalidades adotada é “caso de uso”.
- (T) Estabelecer níveis de pontuação para as medidas: não basta realizar avaliações e obter medidas, é importante que se estabeleça quais são os valores aceitáveis. Assim, nesta tarefa, o avaliador deve identificar estes níveis de pontuação (mais informações estão descritas na seção “3.4.3 Níveis de Pontuação”).
- (E) Projetar a avaliação: a principal saída desta etapa é o plano de avaliação que será executado posteriormente.
- (T) Produzir o plano de avaliação: nesta tarefa, é elaborado o plano de avaliação que deverá ser seguido. O plano deve ser construído em função das informações levantadas nas etapas anteriores (quais os componentes selecionados para a avaliação). O Guia de Avaliação padrão pode ser visto no Apêndice C.
- (E) Executar a avaliação: nesta etapa o plano de avaliação é executado. É importante observar que a execução do plano não é uma atividade ininterrupta, mas sim, pequenas atividades que são realizadas ao longo da execução de um projeto de desenvolvimento.
- (T) Obter as medidas: em função do que foi definido no plano de avaliação, as medidas devem ser coletadas. Cada tipo de medida possui uma forma de ser coletada. O avaliador deve verificar o guia de preenchimento da avaliação para planejar esta tarefa.
- (T) Comparar critérios: nesta tarefa, o papel do avaliador é coincidir as medidas obtidas para cada uma das características avaliadas com as escalas definidas no plano de avaliação.
- (T) Julgar resultados: nesta tarefa, os dados coletados devem ser interpretados em função dos critérios estabelecidos no plano de avaliação.

- (E) Conclusão da avaliação: nesta última etapa do processo, o objetivo é compilar as informações e resultados encontrados num documento.
- (T) Gerar relatório de avaliação: a geração do relatório de avaliação é realizada ao longo do processo de avaliação, mas nesta tarefa é realizado o fechamento do relatório. Também é feita aqui a publicação do relatório para os envolvidos.

Apesar de os modelos estudados sugerirem uma tarefa para especificação do modelo de qualidade a ser seguido na avaliação, esta não foi considerada para esta proposta, uma vez que haverá apenas um modelo a ser seguido pela avaliações realizadas. Isso com objetivo de manter uma consistência nas avaliações, independente do sistema que está sendo avaliado.

É igualmente interessante frisar que a norma NBR ISO/IEC 14598-1 define três perfis de interessados na avaliação (operadores, desenvolvedores e gerentes), e que no momento da elaboração do escopo da avaliação esta informação é definida. Contudo, a presente proposta está considerando que todas as avaliações realizadas devem ter dois tipos de interessados:

- Usuário: ator que está interessado nas funcionalidades que o sistema apresenta. Para o usuário, o sistema deve dar apoio às suas atividades, e isso de forma fácil, eficiente e confiável.
- Analista de Negócios: ator que também tem interesse nas funcionalidades que o sistema possui, mas com uma visão de sistema. Ou seja, em relação à performance, otimização de recursos, bem como com documentações de sistema (por exemplo, casos de uso ao invés de manuais de utilização do sistema).

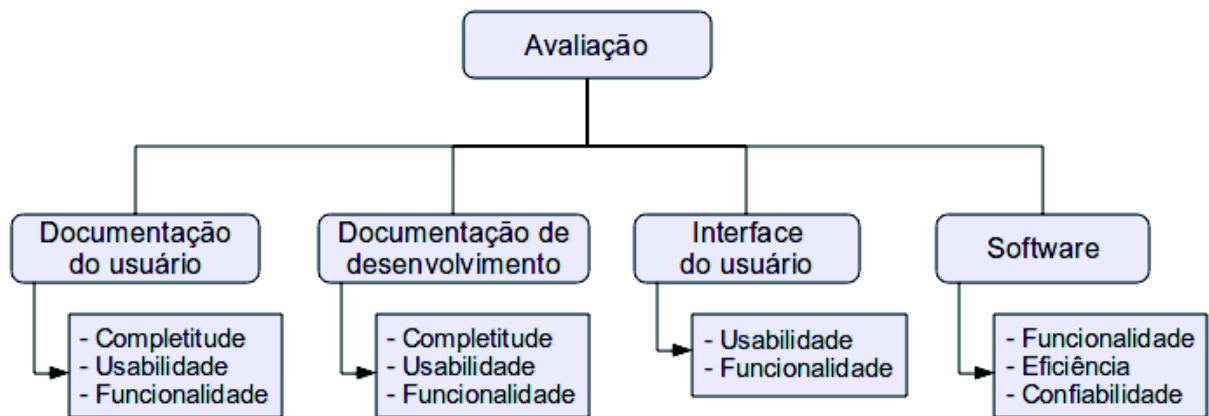
Recomenda-se que a execução desse processo seja realizada por uma pessoa denominada avaliador responsável, que deverá conduzir a avaliação do início ao fim. Este avaliador poderá ser um analista de negócios ou de sistemas, preferencialmente já familiarizado com o sistema objeto da avaliação.

3.3 Modelo de Qualidade

Por se tratar de uma proposta para um cenário específico, optou-se por criar um modelo único de qualidade, que será utilizado por todas as avaliações realizadas. Este modelo foi definido observando-se as melhores práticas apresentadas em Normas da ABNT (como por exemplo, NBR ISO/IEC 14598-1 e NBR ISO/IEC 9126-1), propostas existentes (MEDE-PROS) e adaptadas para o ambiente de execução das avaliações.

Foram escolhidos e especificados quatro componentes que poderão ser incluídos em uma avaliação, a saber: documentação do usuário, documentação de desenvolvimento, interface do usuário e software. A estrutura do modelo pode ser vista na figura 9:

Figura 9- Estrutura do modelo de qualidade proposto



Fonte: Elaborada pela autora

Alguns modelos (por exemplo, a norma NBR ISO/IEC 9126-1) sugerem como componentes de uma avaliação a instalação e desinstalação do software, a embalagem, entre outros. Contudo, considera-se que para este cenário, a avaliação destes componentes não traria muitos benefícios. Por exemplo, não existe a instalação de software, uma vez que ele é desenvolvido internamente e implantado nos próprios servidores da empresa.

Também foi criado um novo componente não considerado em nenhum modelo, o “Documentação de desenvolvimento”. Este novo componente é proposta em função do cenário da empresa, visto que o desenvolvimento de software é terceirizado e, portanto, deve-se manter uma boa base de documentação realizada durante o desenvolvimento dos sistemas. Dessa maneira, optou-se por trabalhar com os quatro componentes acima.

Cada um destes componentes poderá ser avaliado segundo as seguintes características e atributos:

Tabela 6- Itens do modelo de qualidade

Componente	Característica	Atributo
Documentação do usuário	Compleitude	Identificação do produto
		Cobertura da documentação
	Usabilidade	Apresentação visual
		Operacionalidade
		Conformidade
		Inteligibilidade
	Funcionalidade	Apreensibilidade
Acurácia		
Documentação de desenvolvimento	Compleitude	Identificação do produto
		Cobertura da documentação
	Usabilidade	Apresentação visual
		Operacionalidade
		Conformidade
		Inteligibilidade
	Funcionalidade	Acurácia
Interface do usuário	Usabilidade	Inteligibilidade
	Funcionalidade	Padrões internos
		Adequação
Software	Funcionalidade	Segurança de acesso
		Adequação
		Acurácia
	Eficiência	Tempo de resposta
	Confiabilidade	Registro de log
		Maturidade

Fonte: Elaborada pela autora

As características e atributos selecionados para este modelo procuraram abordar também aspectos da qualidade em uso, visto que um dos objetivos da avaliação deve ser a visão do usuário.

Abaixo estão descritas cada uma das características e atributos dos

componentes.

a) Documentação do usuário: composto pelo conjunto de documentos disponíveis para o usuário como fonte de consulta sobre o sistema. Por exemplo: manual impresso e online.

- Completitude
 - *Identificação do produto*: capacidade de a documentação identificar de forma clara o produto e sua versão.
 - *Cobertura da documentação*: capacidade da documentação apresentar as funcionalidades disponibilizadas no produto, explicando-as e abordando a forma de utilizá-las.
- Usabilidade
 - *Apresentação visual*: capacidade da documentação de exibir de forma clara o seu conteúdo.
 - *Operacionalidade*: capacidade da documentação de possibilitar ao usuário operá-la.
 - *Conformidade*: capacidade da documentação de estar de acordo com os padrões definidos na empresa.
 - *Inteligibilidade*: capacidade da documentação de possibilitar ao usuário compreender se o conteúdo apresentado é apropriado e como ele pode ser usado para tarefas e condições de uso específicas.
 - *Apreensibilidade*: capacidade da documentação de possibilitar ao usuário aprender sua aplicação.
- Funcionalidade
 - *Acurácia*: capacidade da documentação de prover, com o grau de precisão necessário, conteúdos corretos ou conforme acordados.

b) Documentação de desenvolvimento: composto pelos documentos gerados após a etapa de análise realizada pela fábrica de desenvolvimento. Por exemplo: casos de uso, diagramas ER, etc.

- Completitude
 - *Identificação do produto*: capacidade de a documentação identificar de forma clara o produto e sua versão.

○ *Cobertura da documentação*: capacidade da documentação apresentar as funcionalidades disponibilizadas no produto, especificando-as.

- Usabilidade

- *Apresentação visual*: capacidade da documentação de exibir de forma clara o seu conteúdo

- *Operacionalidade*: capacidade da documentação de possibilitar ao usuário operá-la.

- *Conformidade*: capacidade da documentação de estar de acordo com os padrões definidos na empresa.

- *Inteligibilidade*: capacidade da documentação de possibilitar ao usuário compreender se o conteúdo apresentado é apropriado e como ele pode ser usado para tarefas e condições de uso específicas.

- Funcionalidade

- *Acurácia*: capacidade da documentação de prover, com o grau de precisão necessário, conteúdos corretos ou conforme acordados.

c) Interface do usuário: representa a interface gráfica do sistema, a qual o usuário fará uso como canal de interação com o aplicativo.

- Usabilidade

- *Inteligibilidade*: capacidade do produto de software de possibilitar ao usuário compreender se o software é apropriado e como ele pode ser usado para tarefas e condições de uso específicas.

- Funcionalidade

- *Padrões internos*: capacidade do produto de software estar de acordo com os padrões de interface e comportamento do sistema definidos na empresa.

- *Adequação*: capacidade do produto de software de prover um conjunto apropriado de funções para tarefas e objetivos do usuário especificados.

d) Software: diz respeito ao produto de software em si, ao sistema executável.

- Funcionalidade

- *Segurança de acesso*: capacidade do produto de software de proteger informações e dados, de forma que pessoas ou sistemas não autorizados não possam lê-los nem modificá-los e que não seja negado o acesso às pessoas ou sistemas autorizados.
- *Adequação*: capacidade do produto de software de prover um conjunto apropriado de funções para tarefas e objetivos do usuário especificados.
- *Acurácia*: capacidade do produto de software de prover, com o grau de precisão necessário, resultados ou efeitos corretos ou conforme acordados.
- Eficiência
 - *Tempo de resposta*: capacidade do produto de software de fornecer tempos de resposta e de processamento, além de taxas de transferência, apropriados, quando o software executa suas funções, sob condições estabelecidas.
- Confiabilidade
 - *Registro de log*: capacidade do produto de software de registrar as alterações realizadas nos dados que mantém, construindo um histórico de ações que os usuários executam no sistema.
 - *Maturidade*: capacidade do produto de software de evitar falhas decorrentes de defeitos no software.

3.4 Métricas e Níveis de Pontuação

Em função do modelo de qualidade utilizado, é necessário definir as métricas que serão utilizadas para avaliar cada um dos atributos. Para o modelo de qualidade proposto, sugere-se duas metodologias de medição: através de uma lista de verificação (onde são atribuídos conceitos para as proposições) e de quantificação simples (onde são relacionadas as ocorrências relacionadas a uma métrica).

Contudo, independentemente do formato da medição, é importante também definir uma escala de valores para converter os conceitos informados em notas numéricas, ou seja, os níveis de pontuação. Estas notas devem ser conferidas aos atributos e para possibilitar uma comparação, as mesmas também devem ser

normalizadas, ou sejam, estejam entre o intervalo de 0 a 1 (COLOMBO, 2004).

A seguir serão detalhadas as duas metodologias diferentes de medição bem como os tipos de questões utilizada e os níveis de pontuação correspondentes.

No Apêndice D pode ser visto um resumo de todas as questões utilizadas no guia de avaliação e seus respectivos tipos, seguindo a definição dada nas próximas seções.

3.4.1 Métricas do tipo Lista de Verificação

As métricas do tipo lista de verificação consistem em um conjunto de questões que deve ser respondido pelo avaliador através de conceitos. Em função das respostas que uma questão pode ter, foram definidos quatro tipos diferentes de questão de avaliação com seus respectivos valores numéricos para conversão. Na tabela 7 podem ser vistos os quatro tipos diferentes de questão do tipo lista de verificação (as respostas estão em ordem crescente de valor, ou seja, do ideal ao indesejado):

Tabela 7- Tipos de Questões e Respostas

Tipo de Questão	Respostas possíveis
Tipo A	S = Sim N = Não
Tipo B	S = Sempre A = Algumas vezes N = Nunca
Tipo C	N = Nunca A = Algumas vezes S = Sempre
Tipo D	T = Todos Q = Quase todos A = Alguns N = Nenhum

Fonte: Elaborada pela autora

Além dos tipos de resposta específicos para cada tipo de questão, ainda são possíveis de serem informados os conceitos AP (Avaliação Prejudicada) e NA (Não se Aplica). Entretanto, estes conceitos não possuem uma pontuação, visto que não devem interferir no cálculo da nota do atributo.

O primeiro tipo de questão trabalha com dois tipos de resposta, considerando que “Sim” seja uma situação desejável para a questão. A tabela 8 mostra os valores numéricos correspondentes:

Tabela 8- Tipo de Questão A

Tipo de Resposta	Valor Numérico
S = Sim	1,00
N = Não	0,00

Fonte: Elaborada pela autora

Os valores utilizados para o segundo tipo de questão podem ser vistos na tabela 9 abaixo. Para este tipo de questão, a resposta “S” (Sempre) é o desejado para a situação. Por exemplo, a questão “A documentação do usuário online possui tipo de letra de fácil leitura” da característica de Usabilidade do componente de Documentação do usuário utiliza esta escala.

Tabela 9- Tipo de Questão B

Tipo de Resposta	Valor Numérico
S = Sempre	1,00
A = Algumas vezes	0,50
N = Nunca	0,00

Fonte: Elaborada pela autora

O terceiro tipo de questão é semelhante ao anterior, contudo aqui a resposta desejada é “N” (Nunca). Por exemplo, a questão “A documentação do usuário online apresenta erros gramaticais” utilizada esta escala. Os valores utilizados podem ser vistos na tabela 10 abaixo.

Tabela 10- Tipo de Questão C

Tipo de Resposta	Valor Numérico
N = Nunca	1,00
A = Algumas vezes	0,50
S = Sempre	0,00

Fonte: Elaborada pela autora

Por fim o quarto tipo faz uso dos valores apresentados na tabela 11:

Tabela 11- Tipo de Questão D

Tipo de Resposta	Valor Numérico
T = Todos	1,00
Q = Quase todos	0,66
A = Alguns	0,33
N = Nenhum	0,00

Fonte: Elaborada pela autora

Os valores utilizados para os conceitos das questões foram inicialmente definidos com base no estudo realizado por Colombo (2004). Contudo, o objetivo é que, após a execução das avaliações na empresa, estes valores sejam revistos e ajustados para a sua realidade.

3.4.2 Métricas do tipo Quantificação

Neste tipo de métrica tem-se uma questão e a resposta desta é um valor numérico obtido ao longo da avaliação. A métrica do tipo quantificação é utilizada para os atributos da característica de adequação e acurácia, para o componente Software e pode ser calculada de duas formas. A tabela abaixo mostra as questões envolvidas nos dois atributos e os seus dois tipos:

Tabela 12- Questões para Adequação e Acurácia (Software)

Atributo	Questão	Tipo
Adequação	Quantos requisitos presentes no sistemas são considerados úteis pelo usuário?	E
	Quantos requisitos presentes no sistemas são considerados inúteis pelo usuário?	F
	Quantos requisitos não estão presentes no sistema, mas são importantes para o usuário?	F
Acurácia	Quantos requisitos presentes no sistemas são considerados úteis pelo usuário e estão corretos?	E
	Quantos requisitos presentes no sistemas são considerados úteis pelo usuário, mas precisam de modificação?	F

Fonte: Elaborada pela autora

A resposta de cada uma das questões é um número absoluto que deverá ser comparado com o total de requisitos envolvidos na avaliação.

Para as questões do tipo E, considera-se que, quanto mais ocorrências forem encontradas, mais próximo do desejado. Dessa maneira, a nota do atributo é obtida através da fórmula abaixo:

$$v = T_o / T_r$$

onde:

T_o = número total de ocorrências encontradas

T_r = número total de requisitos envolvidos na avaliação

Nas questões do tipo F considera-se que, quanto menos ocorrências forem encontradas, mais próximo se está do desejado. Por exemplo, a quantidade de requisitos que precisam se revistos ou modificados no sistema, quanto menos ocorrências forem encontradas neste caso, mais próximo do resultado ideal se está. Assim, o valor final da métrica deve seguir a fórmula:

$$v = 1 - T_o / T_r$$

onde:

T_o = número total de ocorrências encontradas

T_r = número total de requisitos envolvidos na avaliação

O número total de requisitos envolvidos na avaliação é encontrado através dos casos de uso envolvidos na avaliação, que são identificados e relacionados durante a etapa de “Especificação da avaliação” (ver seção 3.2, que mostra o processo de avaliação).

Todas as características de qualidade, bem como seus atributos e questões relacionadas estão descritos no Apêndice C, que apresenta o Guia da Avaliação de produto de software.

3.4.3 Níveis de Pontuação

Como falado anteriormente, não é suficiente realizar avaliações e obter ao final

das atividades um valor para atributos, características ou componentes. É necessário que se estabeleça quais são os valores aceitáveis.

O modelo proposto no presente trabalho leva em consideração uma escala de quatro níveis, seguindo a norma NBR ISO/IEC 14598-1:

- Ultrapassa os requisitos
- Intervalo-alvo
- Mínimo aceitável
- Inaceitável

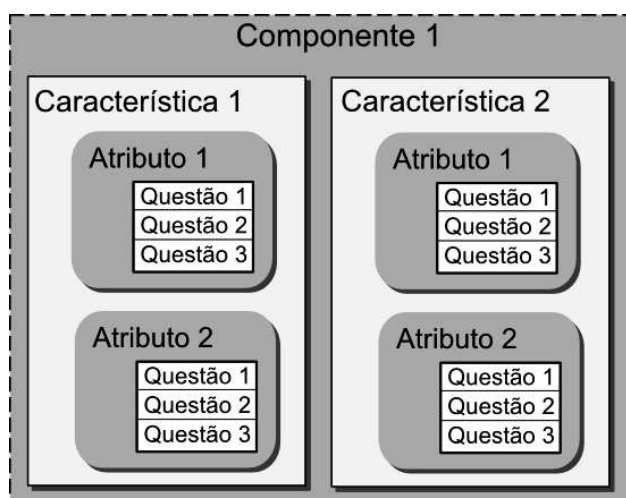
Propõe-se a utilização desta escala por característica utilizada na avaliação. A definição dos intervalos de valores para cada nível deve considerar o intervalo válido das notas (de 0 a 1).

Esta escala deve existir por sistema avaliado. E sugere-se que estes valores sejam estabelecidos após a realização de, no mínimo, cinco avaliações de um sistema. Assim, consegue-se ter um parâmetro consistente para comparação.

3.5 Determinação de Notas

Fazendo uso dos conceitos dados para cada uma das questões e dos valores numéricos correspondentes (apresentados na seção anterior), pode-se calcular a nota dos atributos, características de qualidade e componentes. Na figura 11 é exibido um pequeno esquema da hierarquia de informações na avaliação.

Figura 10: Hierarquia de informações na avaliação



Fonte: Elaborada pela autora

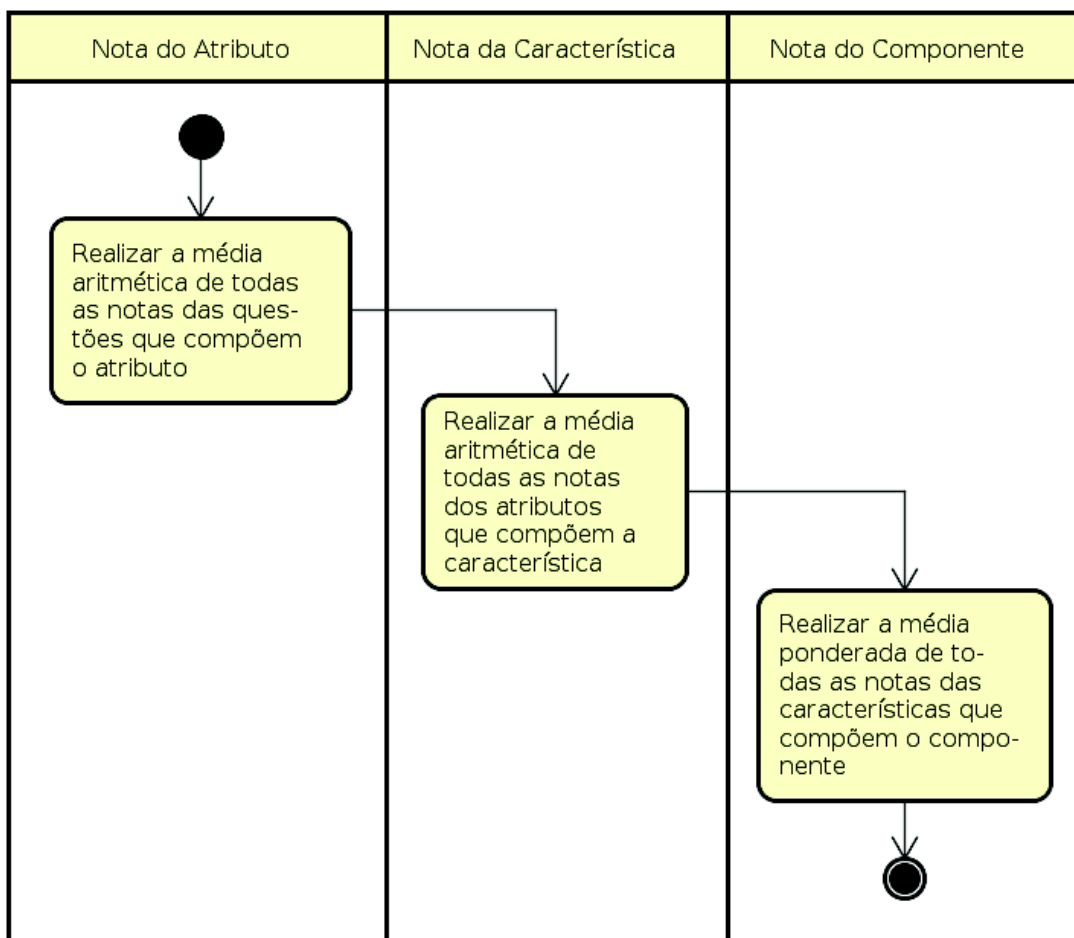
Como pode ser observado na figura acima, há uma organização de informações, fazendo com que tenhamos que trabalhar com composições de notas. Por exemplo, para se definir a nota de um atributo preciso computar as questões, para definir a nota da característica preciso das notas já computadas dos atributos. Assim, pode-se dizer que trabalha-se com uma abordagem *bottom-up*.

No presente trabalho, é sugerida uma forma de definição das notas para cada um das partes que constituem a avaliação. São elas:

- Atributos: para a nota dos atributos, utiliza-se uma fórmula simples que faz uso da média aritmética das notas das questões (já normalizadas) que compõem a avaliação do atributo.
- Característica: com base nas notas dos atributos, pode-se definir a nota da característica, que também é calculada através da média aritmética das notas dos atributos que compõem esta característica.
- Componente: a nota do componente é calculada utilizando-se a média ponderada das suas características. Os pesos utilizados para cada característica são aqueles que foram definidos nos requisitos da avaliação (Apêndice A).

A figura 11 apresenta um pequeno resumo das atividades envolvidas no cálculo das notas dos itens da avaliação.

Figura 11- Atividades da definição de notas



Fonte: Elaborada pela autora

Adicionalmente, pode-se ainda calcular a nota geral da aplicação realizando-se uma média aritmética simples das notas dos componentes envolvidos na avaliação.

3.6 Modelo de Plano de Avaliação

Nenhum dos sistemas existentes hoje na empresa (sistemas desenvolvidos internamente) passou em algum momento por uma avaliação formal. Assim, é de grande importância que o plano de avaliação seja pensado tanto para atender a primeira execução de uma avaliação completa de um produto de software, quanto para avaliações contínuas. Ressalta-se que, a princípio, o plano atenderá ao sistema de Gestão de Projetos de Crédito, mas que terá um nível de generalidade para que possa ser utilizado nos demais sistemas da empresa.

O plano foi pensado para ter três partes. E, assim como a proposta de

processo de avaliação, o plano foi elaborado com base no MEDE-PROS (GUERRA, 2009), mas também no modelo de qualidade da NBR ISO/IEC 9126-1.

A primeira parte contém a identificação do plano (questionário apresentado no Apêndice A). Estas informações são preenchidas nas etapas iniciais do processo de avaliação descrito na seção anterior, e tem como objetivo definir o escopo e o objeto da avaliação, sendo portanto comum a todos os planos de avaliação que forem produzidos. As questões propostas podem ser vistas na figura 12.

Figura 12- Questões da primeira parte do plano de avaliação

1) Avaliador:
2) Nome do sistema e versão:
3) Módulos envolvidos:
4) Usuário chave:
5) Áreas que utilizam o sistema:
6) Tecnologia utilizada:
7) Comunicação com outros sistemas:
8) Objetivos da avaliação: <input type="checkbox"/> Assegurar a qualidade do sistema <input type="checkbox"/> Obter um laudo técnico da sua qualidade <input type="checkbox"/> Indicar pontos para melhoria no produto
9) Priorização dos aspectos de qualidade do produto: <input type="checkbox"/> Funcionalidade. Ênfase: <input type="checkbox"/> Confiabilidade. Ênfase: <input type="checkbox"/> Usabilidade. Ênfase: <input type="checkbox"/> Eficiência. Ênfase: <input type="checkbox"/> Completitude. Ênfase:
10) Quais funções merecem maior dedicação durante a avaliação?
11) Existe massa de dados disponível para a avaliação?
12) Existe um manual de padrão de interface para o produto?

Fonte: Elaborada pela autora

Pode-se observar que através destes dados tem-se uma ideia geral do propósito da avaliação.

Os dados requisitados na primeira parte procuram atender a duas situações: quando será realizada uma avaliação completa de uma versão do sistema e quando apenas algumas funcionalidades/módulos serão avaliados. Através do campo “Módulos envolvidos” o avaliador⁴ pode discriminar este escopo da avaliação. A determinação de quais módulos devem ser indicados não é dúvida, uma vez que o

4 Aqui cabe uma observação: está considerando-se que o avaliador é o mesmo ator que especifica e projeta a avaliação. Mas isso não é regra, duas pessoas diferentes poderiam assumir o papel de elaborador e executor.

plano de avaliação começará a ser elaborado juntamente com o projeto de desenvolvimento. No projeto de desenvolvimento já são identificados quais os casos de uso que sofrerão alterações (ou serão criados). Assim, de posse destas informações, o avaliador consegue saber quais módulos deverão ser inclusos.

Outra questão interessante é que na parte de identificação do plano também deve ser informado o usuário-chave. Caso esteja sendo feita uma avaliação de todo o sistema, este é o próprio usuário-chave do sistema. Caso contrário, indica-se o usuário que já está indicado no projeto de desenvolvimento como principal contato.

Na questão 9 o avaliador pode identificar e caracterizar os requisitos de qualidade envolvidos, de modo a orientar as posteriores atividades de avaliação.

Sugere-se que o preenchimento dos dados de identificação seja sempre realizado consultando-se o usuário-chave, uma vez que um dos focos das avaliações é o próprio usuário.

Para este cenário, os conceitos dos aspectos de qualidade utilizados têm como referência as normas NBR ISO/IEC 9126-1 e NBR ISO/IEC 12119, descritos abaixo:

- Funcionalidade: capacidade do sistema de prover funções que atendam às necessidades explícitas e implícitas, quando o mesmo estiver sendo utilizado sob condições especificadas.
- Confiabilidade: capacidade do sistema de manter um nível de desempenho especificado, quando usado em condições especificadas
- Usabilidade: capacidade do sistema de ser compreendido, aprendido, operado e atraente ao usuário, quando usado sob condições especificadas.
- Eficiência: capacidade do sistema apresentar desempenho apropriado, relativo à quantidade de recursos usados, sob condições especificadas.
- Completitude: capacidade do sistema ser completo, ou seja, conter todas as informações necessárias para o seu bom funcionamento.

A segunda parte do plano contém um conjunto de questões comuns que serão acrescentadas nos planos de avaliação dos sistemas em função dos componentes selecionados para o escopo da avaliação (ver Apêndice B). Este conjunto de questões tem como objetivo a especificação da avaliação e é voltado para as funcionalidades que estão sendo avaliadas. Um resumo destas questões pode ser

visto na figura 13.

Figura 13- Questões do roteiro para especificação da avaliação

1) Casos de uso envolvidos: _____

2) Perfis de usuários envolvidos: _____

3) Manual do usuário _____

4) Componentes e características utilizados na avaliação:

- Documentação do usuário
 - Completitude
 - Usabilidade
 - Funcionalidade

5) Nível de pontuação:

Característica Nível	Intervalo	
	De	Até
Ultrapassa os requisitos		
Intervalo-alvo		
Mínimo aceitável		
Inaceitável		

Fonte: Elaborada pela autora

A definição dos casos de uso envolvidos é realizada a partir da descrição do projeto de desenvolvimento (caso a avaliação esteja sendo realizada em conjunto com o processo de desenvolvimento, ou seja, não seja uma avaliação completa de todo o sistema).

É importante destacar a questão sobre os perfis dos usuários. Não existe sistema hoje, na empresa, que seja utilizado somente por um tipo de usuário. Há sempre um conjunto de perfis, cada qual com a sua responsabilidade e função dentro do sistema. Assim, é interessante destacar na especificação da avaliação quais os perfis envolvidos nas funcionalidades que serão avaliadas. Isso para que as expectativas da avaliação sejam melhor direcionadas.

Na especificação deve-se identificar também se há algum manual do usuário (seja impresso ou online) e onde ele pode ser encontrado.

A questão 4 identifica quais os componentes e características (definidos no modelo de qualidade) serão avaliados. A partir dessa seleção são definidos os níveis de pontuação para cada uma das características, na questão 5. Os níveis de pontuação nortearão a interpretação das medidas obtidas para cada uma das

características.

Por fim, a terceira parte é a customização do Guia de Avaliação (Apêndice C) considerando a especificação da avaliação. Em função do que foi especificado para a avaliação, são selecionadas as questões que dizem respeito aos componentes e características consideradas na avaliação.

4 ESTUDO DE CASO

A proposta apresentada, como já comentado, ambientou-se no processo de desenvolvimento de sistemas internos de uma instituição financeira. Para o estudo de caso, a proposta será aplicada no desenvolvimento de um dos principais sistemas de negócio da empresa: o sistema de Gestão Projetos de Crédito (GPC).

No capítulo anterior foram apresentados uma proposta de processo de avaliação, bem como o plano de avaliação sugerido. Neste capítulo será apresentada a aplicação deste processo e plano de avaliação, bem como os resultados obtidos, considerando-se o cenário inicialmente proposto: o software GPC escolhido como estudo de caso.

4.1 Uso do Processo de Avaliação

O objetivo de se ter um processo de avaliação é que este esteja alinhado com o próprio processo de desenvolvimento. No processo atual de desenvolvimento da empresa nota-se que existem algumas atividades que são realizadas com o objetivo de garantir a qualidade do software implementado. Contudo, o escopo destas atividades sempre acaba sendo focado em pontos muito específicos do sistema (por exemplo, somente verificando se os itens solicitados foram implementados sem erros). Não há uma preocupação em ver o sistema como um todo.

Assim, sugere-se que o processo de avaliação proposto pelo trabalho seja integrado às atividades do processo de desenvolvimento atual, desde o seu início até a finalização. Essa preocupação também pode ser vista na NBR ISO/IEC 14598-5, quando afirma-se que para avaliações em produtos que software que ainda estão em desenvolvimento, o processo de avaliação deve estar sincronizado com o processo de desenvolvimento de software.

Para a aplicação do processo de avaliação, portanto, as seguintes macro-atividades do processo de desenvolvimento serão alteradas:

- Criação do projeto: ao elaborar o escopo do projeto, também deve ser elaborado o plano para a avaliação. Assim, o Analista de Negócios tem como responsabilidade especificar e projetar a avaliação neste instante.

Além disso, a especificação da avaliação utiliza muitas informações que estão no escopo do projeto., por isso a importância destas atividades serem realizadas o mais próximo possível.

- Homologação: aqui deverá ser realizada a execução da avaliação. Esta é uma atividade que deve ser realizada por um Analista de Negócios, preferencialmente um Analista que não seja o responsável pelo projeto de desenvolvimento. Na execução da avaliação, usuários poderão ser consultados também.

Dessa maneira, o processo de avaliação de qualidade do produto proposto está e deve ser executado paralelamente ao processo de desenvolvimento do sistema.

Destaca-se ainda que no caso de avaliações completas de um sistema (onde todo o sistema é avaliado), não há esta integração, visto que o produto a ser avaliado já está finalizado.

4.2 Execução do Processo de Avaliação

Como pensado no início deste trabalho, pensou-se em criar um plano de avaliação específico para um dos sistemas da empresa do cenário selecionado, mas que este plano pudesse ser estendido aos outros sistemas existentes na empresa. Também foi definido que o primeiro passo da execução neste estudo de caso seria uma avaliação de todo o sistema. Espera-se, com isso, estabelecer o início de uma base histórica de dados para as próximas avaliações, item necessário para um bom ambiente de avaliação (MARTINEZ, 1999).

A primeira etapa da avaliação foi definir os requisitos de avaliação, preenchendo o questionário de avaliação (Apêndice A). Ao fim desta atividade, foram obtidos os resultados apresentados na figura 14.

Figura 14- Respostas da primeira parte do plano de avaliação

- 1) Avaliador: *Cassia Nino*
- 2) Nome do sistema e versão: *GPC (v. 2.8.1)*
- 3) Módulos envolvidos: *Gerencial, Consulta e Enquadramento, Pré-Análise, Análise, Aprovação e Tracking*
- 4) Usuário chave: *João Silva*
- 5) Áreas que utilizam o sistema: *S1, S2, S3, S4, S5 e S6*
- 6) Tecnologia utilizada: *Java web e MS SQL Server.*
- 7) Comunicação com outros sistemas:
 - * *A: busca de dados de clientes*
 - * *B: busca de regras de cálculos de produtos*
 - * *C: busca de dados de cadastros básicos (configurações)*
- 8) Objetivos da avaliação: Assegurar a qualidade do sistema
 Obter um laudo técnico da sua qualidade
 Indicar pontos para melhoria no produto
- 9) Priorização dos aspectos de qualidade do produto:
 - Funcionalidade. Ênfase: 5
 - Confiabilidade. Ênfase: 3
 - Usabilidade. Ênfase: 2
 - Eficiência. Ênfase: 2
 - Completitude. Ênfase: 4
- 10) Quais funções merecem maior dedicação durante a avaliação? *Não há.*
- 11) Existe massa de dados disponível para a avaliação? *Não.*
- 12) Existe um manual de padrão de interface para o produto? *Sim.*

Fonte: Elaborada pela autora

Como esta é a primeira avaliação do sistema e ainda terá como escopo todo o software, o objetivo da avaliação é o de obter um laudo técnico da sua qualidade. Assim, estes primeiros resultados poderão servir como balizadores para as futuras avaliações do sistema.

Após definir alguns requisitos, foi preenchido o roteiro para especificação da avaliação, que corresponde ao Apêndice B. Na figura 15, pode-se observar um resumo das informações obtidas ao fim desta atividade.

Figura 15- Respostas da especificação do plano de avaliação

- 1) Casos de uso envolvidos: *Todos disponíveis.*
- 2) Perfis de usuários envolvidos: *Administrador, Cadastrador, Analista de Projeto, Superintendente, Relator*
- 3) Manual do usuário: *Manual impresso e online*
- 4) Componentes e características utilizados na avaliação: *Todos os componentes e características serão utilizados.*
- 5) Nível de pontuação: *Não se aplica.*

Fonte: Elaborada pela autora

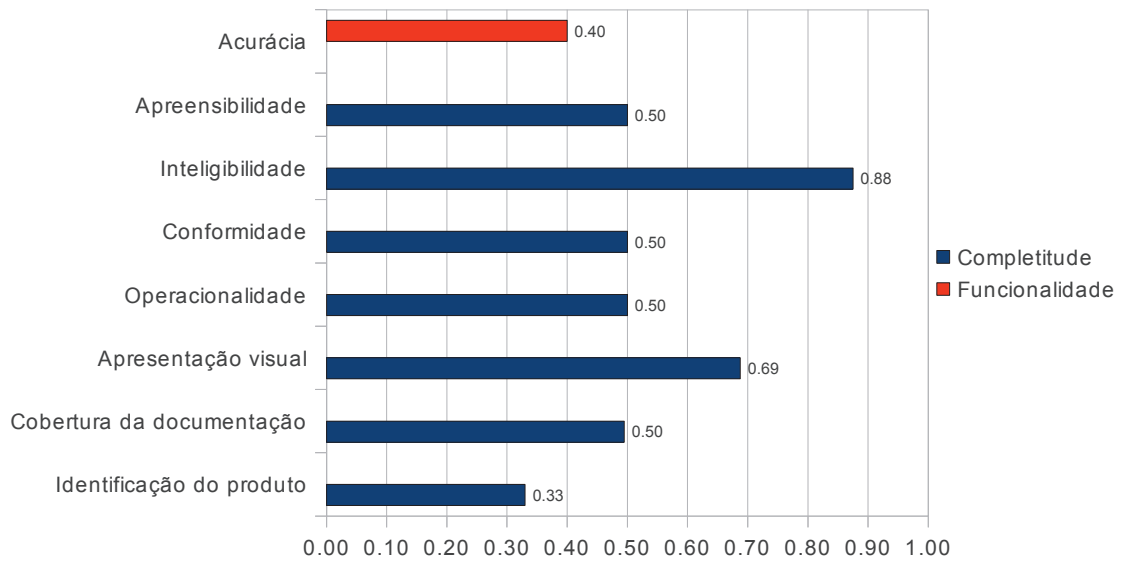
Novamente, como se trata de uma avaliação completa, todos os casos de uso serão considerados na avaliação, bem como todos os perfis que o sistema possui. Igualmente, todos os componentes e características foram selecionados para compôr a avaliação.

Não foram estabelecidos níveis de pontuação para as características uma vez que esta é a primeira avaliação a ser realizada. Assim, não se tem parâmetros para definição de faixas de valores ainda.

Como todos componentes e características foram selecionados para a avaliação, o Guia de Avaliação (Apêndice D) foi utilizado em sua totalidade. Após um teste exploratório no sistema e verificação nas documentações disponíveis, as questões propostas no Guia foram devidamente preenchidas. Com base nas respostas, foram realizadas as apurações para definição das medidas para atributos, características e componentes. Para atributos e características, os valores variam de 0 a 1 onde quando mais próximo de 1, mais perto da situação desejada.

A figura 16 mostra as notas obtidas nos atributos do componente “Documentação do usuário”.

Figura 16- Atributos do componente Documentação do Usuário

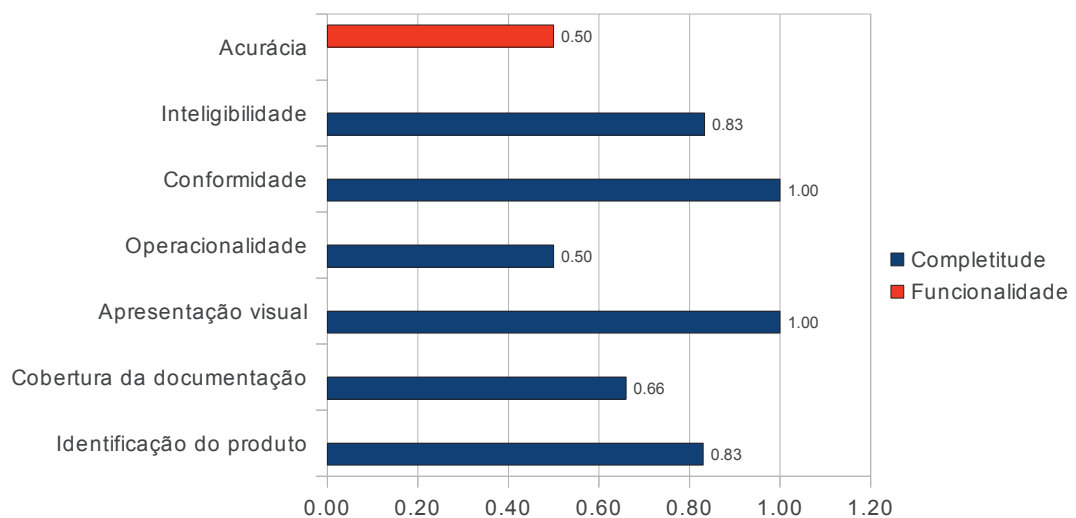


Fonte: Elaborada pela autora

Com estas medidas encontradas para os atributos, temos uma média de 0,4 para a característica de Funcionalidade e 0,56 para a característica de Compleitude, na Documentação do Usuário.

A figura 17 mostra as notas obtidas nos atributos do componente “Documentação de desenvolvimento”.

Figura 17- Atributos do componente Documentação de Desenvolvimento

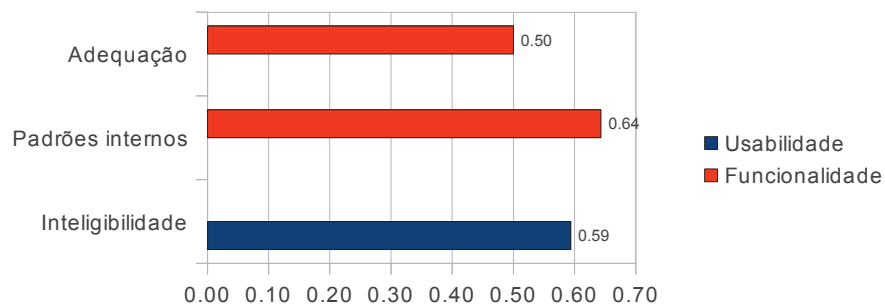


Fonte: Elaborada pela autora

Com estas medidas encontradas para os atributos, temos uma média de 0,5 para a característica de Funcionalidade e 0,8 para a característica de Completitude na Documentação de Desenvolvimento.

A figura 18 mostra as notas obtidas nos atributos do componente “Interface do Usuário”.

Figura 18- Atributos do componente Interface do Usuário

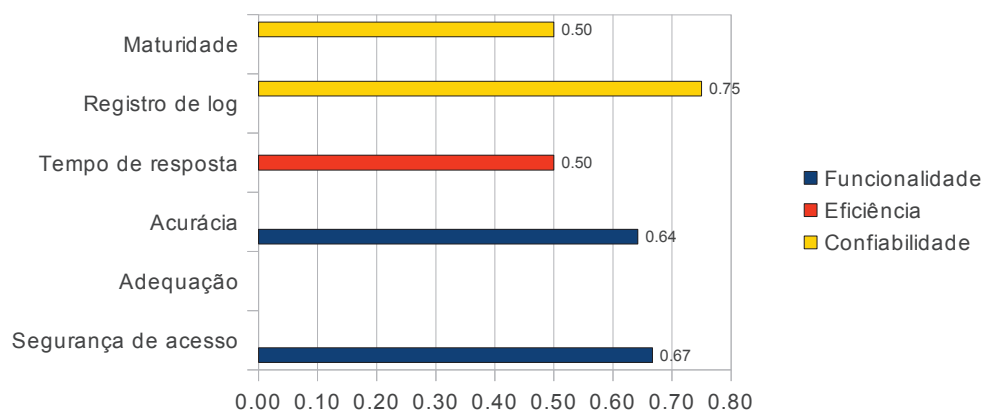


Fonte: Elaborada pela autora

Com estas medidas encontradas para os atributos, temos uma média de 0,59 para a característica de Usabilidade e 0,64 para a característica de Funcionalidade na Interface do Usuário.

Por último, a figura 19 mostra as notas obtidas nos atributos do componente “Software”.

Figura 19- Atributos do componente Software



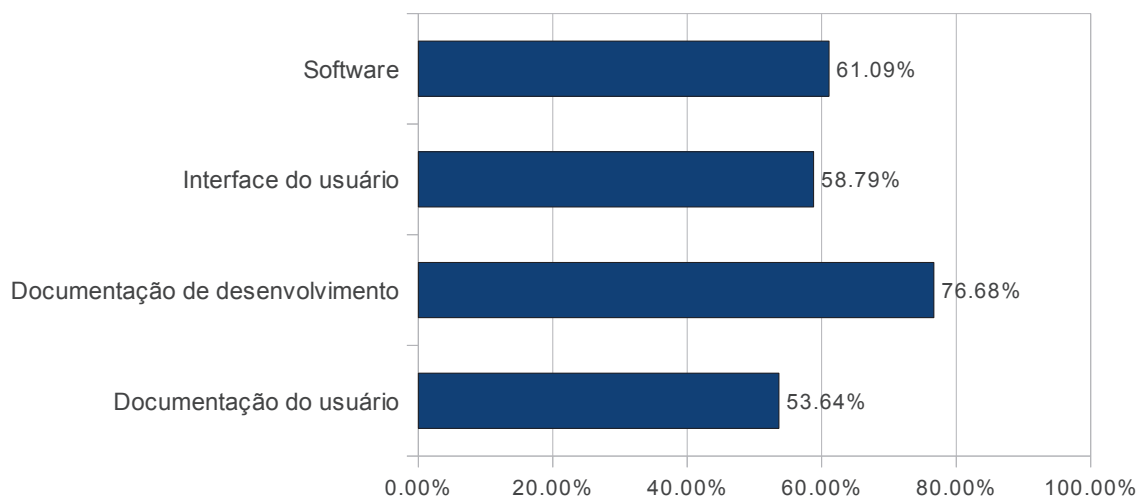
Fonte: Elaborada pela autora

O atributo “Adequação” não foi apurado por requerer uma interação muito forte com o usuário. E, para esta primeira avaliação, optou-se por não envolver todos os usuários. O atributo “Acurácia” foi apurado pelos Analistas de Negócio responsáveis atualmente pelo sistema.

Com estas medidas encontradas para os atributos, temos uma média de 0,65 para a característica de Funcionalidade, 0,5 para Eficiência e 0,63 para a característica de Confiabilidade no Software.

Utilizando as notas das características, calculou-se as notas para cada um dos componentes, através de uma média ponderada. Para este tipo de média os pesos considerados foram aqueles definidos nos objetivos da avaliação. Um resumo das notas de cada um dos componentes pode ser visto na figura 20. A fim de facilitar a visualização e comparação das notas, estas foram normalizadas.

Figura 20- Notas dos Componentes



Fonte: Elaborada pela autora

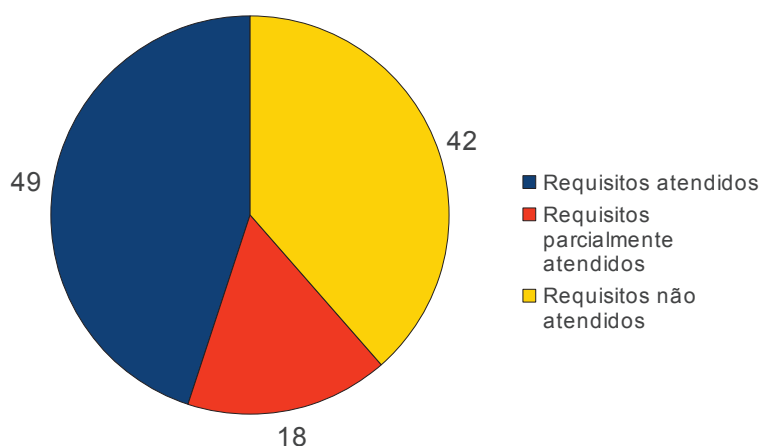
O pior desempenho é o do componente “Documentação do usuário”. Durante a avaliação observou-se uma falta muito grande deste tipo de artefato. Os manuais impressos são escassos e incompletos (não tratam de todo o sistema). O manual online não corresponde ao manual impresso, também trazendo algumas informações desatualizadas. Contudo, ambos estavam numa linguagem clara e adequada.

O componente “Interface do usuário” também obteve um índice baixo. Observou-se várias faltas na interface durante a avaliação. Como por exemplo, ícones sem identificação ou funcionalidades semelhantes que não correspondem em todas as telas (por exemplo, em uma listagem é necessário clicar no ícone de edição para editar um item, em outra listagem é necessário clicar sobre o item para realizar a mesma ação), não há um padrão, apesar de haver um manual de padrões de interface.

O melhor desempenho ficou com o componente “Documentação de desenvolvimento”. A explicação para tal foi o constante investimento no setor de TI da empresa para organizar seus processos de desenvolvimento no último ano.

O componente software foi avaliado através de testes exploratórios. Aqui observou-se a conformidade dos requisitos já levantados do sistema, e encontrou-se uma quantidade muito grande de requisitos que não foram implementados. (cerca de 39%) A figura 21 mostra uma visão geral dos requisitos já implantados, os que necessitam alguma alteração por estarem incompletos e os que não foram implementados.

Figura 21- Atendimento aos Requisitos



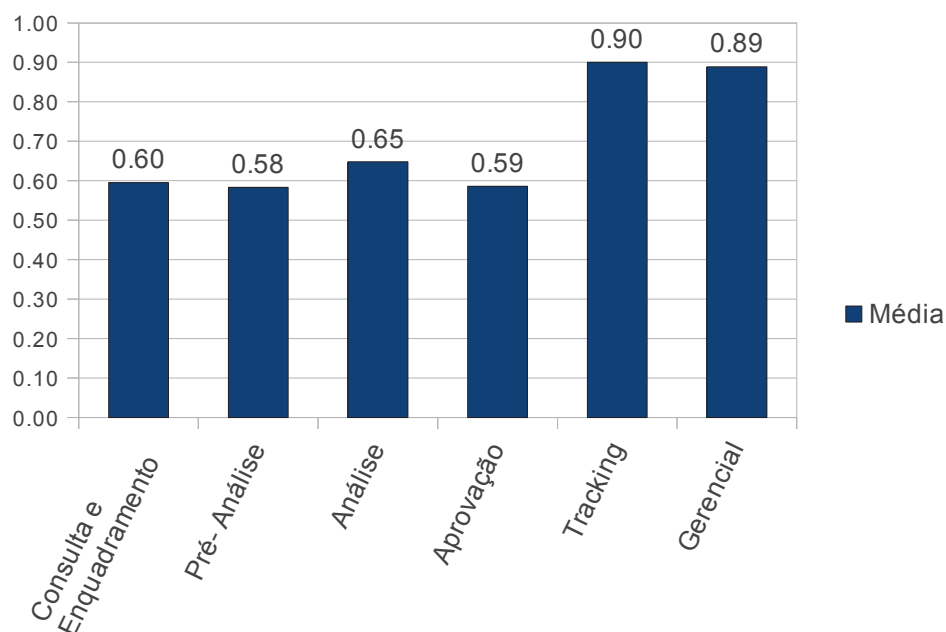
Fonte: Elaborada pela autora

Em função deste desempenho nos testes, a nota do componente “Software” não foi muito alta.

Por ser um componente importante na avaliação, analisaram-se os resultados que compuseram a nota do componente de Software de forma detalhada. Por

exemplo, para as questões do item 4.1.3 do guia de avaliação (componente de Software, característica de Funcionalidade e atributo Acurácia, ver Apêndice C), as notas foram, primeiramente, atribuídas por módulos do sistema. A figura 22 mostra as notas detalhadas para cada um destes módulos.

Figura 22- Notas da avaliação por módulos do sistema



Fonte: Elaborada pela autora

As notas apresentadas no gráfico estão normalizadas, tendo como intervalo de valores possíveis [0; 1].

Através destes dados é possível perceber que o módulo que está mais bem atendido no sistema não é de negócio. O módulo de negócio que se destaca é o Gerencial, que só possui funcionalidades para relatórios, ou seja extração de dados. O restante dos módulos, que tem um valor grande de negócio para a empresa, estão com uma pontuação média bem abaixo dos dois primeiros.

Em função das notas obtidas em cada um dos componentes e as notas máximas que poderiam ser atribuídas para cada um deles, calculou-se também a nota geral do sistema, como pode ser visto com mais detalhes na tabela 13 abaixo.

Tabela 13- Notas obtidas para o Sistema

Nota do Sistema calculada	Maior nota possível para o Sistema	Percentual da nota obtida
11,99	19,13	62,7%

Assim, determina-se que numa escala de 0 a 100 a nota para o sistema avaliado é 62,7.

Como já foi comentado anteriormente, não é possível fazer algum tipo de julgamento sobre esta pontuação. Somente com as próximas avaliações do sistema poderemos ter uma base histórica de informações que permitirá uma comparação de resultados.

É importante ressaltar que, por mais que a escala de valor utilizada tenha como valor máximo 100, é difícil observar uma situação onde se tenha um sistema deste porte e todos os requisitos funcionais e de qualidade estejam atendidos. Sempre haverá pontos de melhoria a serem atendidos. Mas, o essencial é que estes pontos sejam amplamente conhecidos pela equipe responsável pelo sistema, para que possa planejar suas versões futuras de maneira consistente.

5 CONCLUSÃO

Um dos principais problemas encontrados em ambientes de desenvolvimento de software é como obter um bom nível de qualidade no seu produto final. Tendo isso em mente e ainda, com o apoio de modelos de qualidade existentes, este trabalho se propôs a estudar um cenário particular em uma empresa e definir um plano de avaliação contínua para um software desenvolvido.

Em função do cenário encontrado, foi elaborado um processo de avaliação de produto, juntamente com um modelo de qualidade e demais artefatos necessários para a sua execução. Este método foi pensado para ser válido tanto para a avaliação de todo um sistema, quanto para pequenas partes, bem como para poder ser utilizado em outros sistemas da empresa. O modelo de qualidade definido procurou trazer as melhores práticas adotadas em trabalhos estudados e ainda foi baseado em um conjunto de normas amplamente conhecidas e seguidas. Para o trabalho, procurou-se adequar este modelo com o ambiente encontrado. Dessa forma, é importante destacar a definição da característica “Documentação de Desenvolvimento” que foi incluída no modelo de qualidade. Esta característica não foi observada em nenhum modelo ou norma existente, mas é de grande valia no cenário escolhido.

O processo proposto foi executado da forma completa (tendo como escopo todo o sistema) pela autora e seu resultado foi apresentado no presente trabalho também. Esta execução foi realizada com o objetivo principal de obter uma situação atual da qualidade do software selecionado. Foram avaliados todos os componentes do modelo de qualidade e assim, obteve-se uma pontuação também para o sistema.

Ao longo da execução do trabalho, percebeu-se a importância de se realizar formalmente uma avaliação do produto que está sendo desenvolvido. Assim como em muitas empresas, quase sempre o foco das validações e verificações fica limitado ao que se está alterando no sistema e não no sistema como um todo. O plano de avaliação contínua proposto surgiu justamente para tentar minimizar este tipo de carência.

Através dos resultados da avaliação, pôde-se ter uma visão geral e inicial da situação do sistema. A partir destes resultados, como o presente trabalho já

sinalizou, os dados encontrados servirão de parâmetro para as próximas avaliações do sistema.

Um ponto positivo do relatório de avaliação gerado ao fim da avaliação é que ele traz de forma clara o cenário atual do sistema. Por exemplo, qual a situação do sistema sob a visão de características de qualidade de software ou ainda, quais são as taxas de atendimento dos módulos de funcionalidades. Isso auxiliará a equipe de Analistas do sistema a planejar suas melhorias futuras.

Além dos objetivos propostos, também foram obtidas algumas melhorias como consequência do que foi inicialmente pensado, como por exemplo a criação do catálogo de produtos. Este catálogo tem como objetivo listar todos os sistemas que a empresa utiliza, identificando apropriadamente para cada um dados como: tecnologia utilizada, escopo, módulos, comunicação com outros sistemas, etc. Este tipo de informação encontrava-se anteriormente disperso ou então inexistente, em muitos casos. Em função da necessidade de se ter este tipo de informação para a construção do plano de avaliação, já optou-se por manter estes dados sempre disponíveis e atualizados.

Como passos futuros para este trabalho destaca-se dois pontos que podem ser colocados em prática em breve:

- Avaliação total do sistema por outros analistas do sistema: outros analistas do sistema do cenário utilizado podem realizar o mesmo processo documentado pelo presente trabalho. Assim, já se tem uma base histórica mais consistente para as futuras avaliações.
- Avaliação total do sistema com usuários: este é um trabalho de campo mais longo, visto que deverá levar em consideração todo o sistema. Por sua característica modular, deverão ser definidos grupos de usuários por perfis (pois cada um acessar um conjunto diferente de funcionalidades no sistema), e deverão ser selecionados alguns usuários por grupos. Estes usuários serão entrevistados, utilizando-se as questões do atributo de Adequação do componente Software (segundo o modelo de qualidade proposto).

Além disso, os próximos projetos de alterações do sistema já deverão utilizar este processo de avaliação, que será integrado ao processo de desenvolvimento.

Outro passo futuro é a utilização deste plano para o restante dos sistemas da empresa, que também são desenvolvidos internamente. Assim, será possível ter um panorama geral da situação dos sistemas que o setor de TI dá manutenção. Com isso, poderão ser realizados planos de melhoria com um escopo bem claro, que terão grandes chances de trazer benefícios reais para os usuários.

6 REFERÊNCIAS BIBLIOGRÁFICAS

CARNEIRO, Marcelo Renê. **Uma Análise Crítica do MEDE-PROS**. 2004. 67 f. Trabalho de Conclusão de Curso de Especialização (Especialização em Tecnologia da Informação) – Universidade Federal de Pernambuco, Recife, PE, 2004.

COLOMBO, Regina Maria Thienne. **Processo de Avaliação da Qualidade de Pacotes de Software**. 2004. 184 f. Dissertação (Mestrado Profissional) – Programa de Pós Graduação da Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas, Campinas, SP, 2004.

COLOMBO, Regina, GUERRA, Ana Cervigni. **The Evaluation Method for Software Product**. ICSSEA'2002 - International Conference "Software & Systems Engineering and their Applications", Paris, 2002.

DÁVILA, Abraham, MELENDEZ, Karin, FLORES, Luis. **Determinación de los Requerimientos de Calidad del Producto Software Basados em Normas Internacionales**. IEEE, Latin America Transactions, vol. 4, April, 2006.

GOMES, Nelma da Silva Gomes. **Qualidade de Software - Uma Necessidade**. Ministério da Fazenda, Programa Nacional de Apoio à Administração Fiscal para os Estados Brasileiros – Documentos de Referência sobre Melhores Práticas. Disponível em: <http://www.fazenda.gov.br/ucp/pnafe/cst/arquivos/Qualidade_de_Soft.pdf>. Acesso em 27 abr. 2012.

GUERRA, Ana Cervigni, COLOMBO, Regina Maria Thienne. **Qualidade de Produto de Software**. PBQP Software, 2009.

MARTINEZ, Márica Regina M., et al. **The Software Product Evaluation Data Base - Supporting MEDE-PROS**. In: ISESS - International Software Engineering Standards Symposium - Best Software Practices for the Internet age, 4. Curitiba, 1999.

NASCIMENTO, Thiago Alexandre do. **Avaliação da Qualidade de um Produto de Software**. 85 f. Trabalho de Conclusão de Curso – Curso de Ciência da Computação, Universidade Federal de Pernambuco, Recife, PE, 2010.

NBR ISO/IEC 9126-1. Engenharia de software - Qualidade de produto. Parte 1: Modelo de qualidade. Associação Brasileira de Normas Técnicas, 2003.

NBR ISO/IEC 14598-1. Tecnologia de informação – Avaliação de Produto de Software. Parte 1: Visão Geral. Associação Brasileira de Normas Técnicas, 2001.

NBR ISO/IEC 14598-2. Tecnologia de informação – Avaliação de Produto de Software. Parte 2: Planejamento e gestão. Associação Brasileira de Normas Técnicas, 2003.

NBR ISO/IEC 14598-5. Tecnologia de informação – Avaliação de Produto de Software. Parte 5: Processo para avaliadores. Associação Brasileira de Normas Técnicas, 2001.

NBR ISO/IEC 25000-1. Engenharia de software - Requisitos e avaliação da qualidade de produtos de software (SQuaRE) - Guia do SquaRE. Associação Brasileira de Normas Técnicas, 2008.

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

PUNTER, Teade, SOLINGER, Rini Van, TRIENEKENS, Jos. **Software Product Evaluation - Current status and future needs for customers and industry**. 4th IT Evaluation Conference (EVIT-97), The Netherlands, 1997.

REZENDE, Denis Alcides. **Engenharia de Software e Sistemas de Informação**. Rio de Janeiro: Brasport, 2005.

RIOSOFTE. Avaliação de Conformidade de Produtos de Software. Riosoft, agente Softex – PROIMPE (Programa de Estímulo ao Uso de Tecnologia de Informação em Micro e Pequenas Empresas). Disponível em: <<http://www.riosoft.softex.br/cgi/cgilua.exe/sys/start.htm?sid=53>>. Acesso em: 04/03/2012.

SAMADHIYA, Durgesh, WANG, Su-Hua, CHEN, Dengjie. **Quality Models: Role and Value in Software Engineering**. 2nd International Conference on Software Technology and Engineering (ICSTE), Porto Rico, 2010.

SANTOS, Lizandra Bays dos, PRETZ, Eduardo. **Framework para Especialização de Modelos de Qualidade de Produtos de Software**. IX Simpósio Brasileiro de Qualidade de Software, 2010.

SPB, 5CQualiBr. Vetor Qualidade de Produto SPB. Portal do Software Público Brasileiro. Disponível em <<http://www.softwarepublico.gov.br/5cqualibr/>>. Acesso em 04/03/2012.

SUEMATSU, Rhonda L., BROWN, David W. **Software quality depends upon the development process**. Laser Focus World - Software & Computing, Jan 2002.

APÊNDICE A – Questionário de Requisitos de Avaliação

Questionário de Avaliação de Produto

Requisitos de Avaliação

- 1) Avaliador: _____
- 2) Nome do sistema e versão: _____
- 3) Módulos envolvidos: _____
- 4) Usuário chave: _____
- 5) Áreas que utilizam o sistema: _____
- 6) Tecnologia utilizada: _____
- 7) Comunicação com outros sistemas: _____
- 8) Objetivos da avaliação:
 - Assegurar a qualidade do sistema
 - Obter um laudo técnico da sua qualidade
 - Indicar pontos para melhoria no produto
- 9) Priorização dos aspectos de qualidade do produto:
 - Funcionalidade. Ênfase (1 a 5): _____
 - Confiabilidade. Ênfase (1 a 5): _____
 - Usabilidade. Ênfase (1 a 5): _____
 - Eficiência. Ênfase (1 a 5): _____
 - Completitude. Ênfase (1 a 5): _____
- 10) Quais funções merecem maior dedicação durante a avaliação? _____
- 11) Existe massa de dados disponível para a avaliação? _____
- 12) Existe um manual de padrão de interface para o produto? _____

Instruções de Preenchimento

Abaixo encontram-se instruções de preenchimento das questões:

- 1) Nome do avaliador.
- 2) Indicar o nome, sigla e versão do sistema que será avaliado.
- 3) Caso não seja feita uma avaliação do sistema todo, deverão ser indicados os módulos que serão levados em consideração. Sugere-se que, ao realizar uma avaliação apenas de algumas funcionalidades (casos de uso) do sistema, indique-se o módulo ao qual elas fazem parte. Assim, pretende-se garantir uma consistência maior das alterações realizadas.
- 4) Indicar o usuário-chave do sistema, quando estiver sendo realizada uma avaliação de todo o sistema. Caso seja uma avaliação por funcionalidades (ou seja, por projeto de desenvolvimento), o usuário-chave será o principal contato do projeto.
- 5) Relacionar as áreas que usam o sistema que será avaliado (ou os módulos).
- 6) Anotar a tecnologia utilizada pelo sistema (linguagem e banco).
- 7) Caso o sistema (ou módulos) que será avaliado tenha uma comunicação com outros aplicativos, listá-los com uma breve explanação dessa comunicação. Importante: só serão listados os aplicativos que impactam diretamente no escopo da avaliação.
- 8) Nesta questão devem ser marcados o(s) objetivo(s) da avaliação. Pode ser marcada mais de uma opção.
- 9) Os aspectos de qualidade do produto que serão levados em consideração na avaliação deverão ser marcados. E ainda, deve ser definida uma ênfase para cada um dos itens selecionados, onde 1 indica o menor interesse e 5, o maior.
- 10) Caso haja necessidade, nesta questão podem ser sinalizados quais funcionalidades devem ter uma atenção especial durante a avaliação.
- 11) Verificar se existem informações históricas sobre a avaliação deste sistema. Caso afirmativo, destacar este fato nesta questão, listando que tipo de dados estão disponíveis e se serão utilizados nesta avaliação e como.
- 12) Como nem todos os sistemas da empresa possuem um manual de padrões de interface e esta é uma das características levadas em consideração durante a avaliação, deve-se verificar se este sistema possui um manual específico. Caso afirmativo, indicar onde se encontra.

APÊNDICE B – Roteiro de Especificação da Avaliação

Roteiro de Especificação da Avaliação

1) Casos de uso envolvidos: _____

2) Perfis de usuários envolvidos: _____

3) Manual do usuário: _____

4) Componentes e características utilizados na avaliação:

- Documentação do usuário
 - Completitude
 - Usabilidade
 - Funcionalidade

- Documentação de desenvolvimento
 - Completitude
 - Usabilidade
 - Funcionalidade

- Interface de usuário
 - Usabilidade
 - Funcionalidade

- Software
 - Funcionalidade
 - Eficiência
 - Confiabilidade

5) Nível de pontuação:

Característica _____ Nível	Intervalo	
	De	Até
Ultrapassa os requisitos		
Intervalo-alvo		
Mínimo aceitável		
Inaceitável		

Instruções de Preenchimento

Abaixo encontram-se instruções de preenchimento das questões:

- 1) Neste item devem ser listados quais os casos de uso serão considerados na avaliação. Esta resposta é principalmente importante no caso de avaliação de uma parte do produto de software.
- 2) Em função dos casos de uso identificados, o avaliador deve listar os perfis de usuários envolvidos. Assim, tem-se de modo claro o tipo de usuário que deverá ser considerado no momento da avaliação.
- 3) Caso exista o manual do usuário (ou manuais), deve ser indicado neste item. É importante relacionar também as seções do manual que serão utilizadas na avaliação (em função dos casos de uso envolvidos também).
- 4) Neste item são sinalizados os componentes que serão levados em consideração na avaliação. Em uma avaliação completa de uma versão pronta do sistema, por exemplo, o componente “Documentação de desenvolvimento” não seria utilizada.
- 5) Nesta questão devem ser definidos os níveis de pontuação. Para cada característica selecionada, deve existir uma escala que caracteriza a pontuação (medida) obtida na característica. As medidas das características estão no intervalo [0; 1].

APÊNDICE C – Guia de Avaliação

Guia de Avaliação de Produto de Software

Este documento serve como guia para a avaliação de um produto de software.

Após preenchimento do questionário para levantamento de requisitos da avaliação e do roteiro para especificação da avaliação, este guia deverá ser executado de acordo com as atividades descritas no processo de avaliação.

A execução deste guia deverá ser realizada paralelamente com atividades de testes do tipo exploratório no sistema, bem como verificações na documentação disponível. Ao longo do teste e verificação deverão ser considerados com mais atenção as funcionalidades especificamente destacadas para a avaliação.

O guia está dividido em quatro partes, que representam os quatro componentes que podem estar no escopo da avaliação.

1) Componente Documentação do Usuário

1.1) Completitude

1.1.1) Identificação do produto

A documentação do usuário impressa está identificada com:

nome do produto

versão do produto e data correspondente

T=Todas; Q=Quase todas; A=Algumas; N=Nenhuma

A documentação do usuário online está identificada com:

nome do produto

versão do produto e data correspondente

T=Todas; Q=Quase todas; A=Algumas; N=Nenhuma

1.1.2) Cobertura da documentação

A documentação do usuário impressa disponível cobre o quanto de funcionalidades do escopo da avaliação? ()

T=Todas; Q=Quase todas; A=Algumas; N=Nenhuma

A documentação do usuário online disponível cobre o quanto de funcionalidades do escopo da avaliação? ()

T=Todas; Q=Quase todas; A=Algumas; N=Nenhuma

1.2) Usabilidade

1.2.1) Apresentação visual

A documentação do usuário impressa:

() possui tamanho de letra de fácil leitura

() possui tipo de letra de fácil leitura

() apresenta recursos gráficos com uma diagramação bem distribuída

() utiliza recursos para destaque nas informações relevantes

S=Sempre; A=Algumas vezes; N=Nunca

A documentação do usuário online:

() possui tamanho de letra de fácil leitura

() possui tipo de letra de fácil leitura

() apresenta recursos gráficos com uma diagramação bem distribuída

() utiliza recursos para destaque nas informações relevantes

S=Sempre; A=Algumas vezes; N=Nunca

1.2.2) Operacionalidade

A documentação do usuário impressa:

() apresenta um índice geral

() permite uma busca por informações

S=Sempre; A=Algumas vezes; N=Nunca

A documentação do usuário online:

() apresenta um índice geral

permite uma busca por informações

utiliza o recurso de hipertexto através de links

S=Sempre; **A**=Algumas vezes; **N**=Nunca

1.2.3) Conformidade

O formato da documentação impressa está de acordo com os padrões de apresentação definidos?

S=Sempre; **A**=Algumas vezes; **N**=Nunca

O formato da documentação online está de acordo com os padrões de apresentação definidos?

S=Sempre; **A**=Algumas vezes; **N**=Nunca

1.2.4) Inteligibilidade

O texto da documentação impressa:

claro e preciso, não dando margem a interpretações ambíguas

utiliza termos e explicações considerando o tipo de usuário a que se destina o produto

S=Sempre; **A**=Algumas vezes; **N**=Nunca

O texto da documentação impressa:

apresenta erros gramaticais

apresenta erros ortográficos

N=Nunca; **A**=Algumas vezes; **S**=Sempre

O texto da documentação online:

claro e preciso, não dando margem a interpretações ambíguas

utiliza termos e explicações considerando o tipo de usuário a que se destina o produto

S=Sempre; **A**=Algumas vezes; **N**=Nunca

O texto da documentação online:

apresenta erros gramaticais

apresenta erros ortográficos

N=Nunca; A=Algumas vezes; S=Sempre

1.2.5) Apreensibilidade

A documentação impressa:

apresenta exemplos de uso do sistema para auxiliar na compreensão

os exemplos mostrados são claros e precisos, não dando margem a interpretações ambíguas

S=Sempre; A=Algumas vezes; N=Nunca

1.3) Funcionalidade

1.3.1) Acurácia

Na documentação do usuário impressa:

foram encontradas evidências de informações incorretas

foi observada alguma contradição entre os dados apresentados e o observado no sistema

N=Nunca; A=Algumas vezes; S=Sempre

Na documentação do usuário online:

foram encontradas evidências de informações incorretas

foi observada alguma contradição entre os dados apresentados e o observado no sistema

N=Nunca; A=Algumas vezes; S=Sempre

As informações apresentadas no material impresso são as mesmas da documentação online?

S=Sempre; A=Algumas vezes; N=Nunca

1.4) Observações

2) Componente Documentação de Desenvolvimento

2.1) Completitude

2.1.1) Identificação do produto

A documentação está identificada com:

nome do produto

versão do produto e data correspondente

T=Todas; Q=Quase todas; A=Algumas; N=Nenhuma

2.1.2) Cobertura da documentação

A documentação disponível cobre o quanto de funcionalidades do escopo da avaliação?

T=Todas; Q=Quase todas; A=Algumas; N=Nenhuma

2.2) Usabilidade

2.2.1) Apresentação visual

A documentação:

possui tamanho de letra de fácil leitura

possui tipo de letra de fácil leitura

apresenta recursos gráficos com uma diagramação bem distribuída

S=Sempre; A=Algumas vezes; N=Nunca

2.2.2) Operacionalidade

A documentação:

apresenta um índice geral

permite uma busca por informações

S=Sempre; A=Algumas vezes; N=Nunca

2.2.3) Conformidade

O formato da documentação está de acordo com os padrões de apresentação definidos? ()

S=Sempre; A=Algumas vezes; N=Nunca

2.2.4) Inteligibilidade

O texto da documentação:

() é claro e preciso, não dando margem a interpretações ambíguas

S=Sempre; A=Algumas vezes; N=Nunca

O texto da documentação:

() apresenta erros gramaticais

() apresenta erros ortográficos

N=Nunca; A=Algumas vezes; S=Sempre

2.3) Funcionalidade

2.3.1) Acurácia

Na documentação:

() foram encontradas evidências de informações incorretas ou dúbias

() foi observada alguma contradição entre os dados apresentados e o observado no sistema

S=Sempre; A=Algumas vezes; N=Nunca

2.4) Observações

3) Componente Interface do Usuário

3.1) Usabilidade

3.1.1) Inteligibilidade

O texto apresentado na interface:

- está organizada em grupos segundo uma forma lógica facilmente compreendida pelo usuário (menus)
- informa ao usuário sobre o que um componente faz (nomes claros, hints, etc)
- utiliza um mesmo identificador para uma dada função em todo o sistema
- possibilita a realização da tarefa desejada com um número reduzido de passos
- exibe mensagens de orientação ao usuário
- as mensagens exibidas orientam o usuário de forma efetiva e eficiente na execução da tarefa desejada

S=Sempre; **A**=Algumas vezes; **N**=Nunca

A interface e o texto apresentado na interface:

- apresenta erros gramaticais
- apresenta erros ortográficos
- utiliza códigos e terminologia não significativa
- dá margem à interpretações ambíguas

N=Nunca; **A**=Algumas vezes; **S**=Sempre

As telas:

- apresentam uma distribuição uniforme de seu conteúdo, levando em consideração o espaço disponível
- apresentam somente informações necessárias e utilizáveis, sensíveis ao contexto
- seguem um padrão na distribuição dos objetos, facilitando o entendimento dos mesmos
- exibem as mensagens com bom aspecto visual, utilizando, com moderação, negrito, itálico e sublinhado
- utilizam tipos e tamanhos de letras de fácil visualização
- apresentam contrastes de cores, facilitando a leitura

S=Sempre; **A**=Algumas vezes; **N**=Nunca

3.2) Funcionalidade

3.2.1) Padrões internos

A interface mantém uma padronização em relação:

- à configuração das janelas
- à apresentação de ícones
- às mensagens exibidas
- à apresentação de resultados ao usuário
- à posição de um determinado botão que executa uma mesma função em janelas ou caixas de diálogo semelhantes
- aos formatos dos campos de entrada de dados
- ao formato do cursor para uma mesma situação

S=Sempre; A=Algumas vezes; N=Nunca

3.2.2) Adequação

A interface:

- possui funções de interface bem definidas, de modo a não deixar dúvidas sobre o que fazem (ex.: botões, menus)
- apresenta as funções do sistema numa estrutura de fácil acesso ao usuário

S=Sempre; A=Algumas vezes; N=Nunca

3.3) Observações

4) Componente Software

4.1) Funcionalidade

4.1.1) Segurança de acesso

O software:

- faz uso de acesso autenticado por usuário
- impede a entrada de usuários não autorizados
- utiliza permissões por usuário para controle de acesso

S=Sim; N=Não

4.1.2) Adequação

Quantos requisitos presentes no sistemas são considerados úteis pelo usuário?
(_____)

Quantos requisitos presentes no sistemas são considerados inúteis pelo usuário?
(_____)

Quantos requisitos não estão presentes no sistema, mas são importantes para o usuário? (_____)

4.1.3) Acurácia

Quantos requisitos presentes no sistemas são considerados úteis pelo usuário e estão corretos? (_____)

Quantos requisitos presentes no sistemas são considerados úteis pelo usuário, mas precisam de modificação? (_____)

4.2) Eficiência

4.2.1) Tempo de resposta

Há telas no software que são acessadas e tornam-se funcionais:

() em menos de 10 segundos

S=Sempre; A=Algumas vezes; N=Nunca

Há consultas realizadas em alguma tela onde o resultado apareça:

() em menos de 10 segundos

S=Sempre; A=Algumas vezes; N=Nunca

4.3) Confiabilidade

4.3.1) Registro de log

Há registros de logs nas funcionalidades para:

() inserção de dados

() alteração de dados

exclusão de dados

consulta de dados

S=Sempre; **A**=Algumas vezes; **N**=Nunca

4.3.2) Maturidade

O software:

exibiu mensagens claras de orientação após a ocorrência de falhas

S=Sempre; **A**=Algumas vezes; **N**=Nunca

O software:

apresentou falhas durante a sua execução

N=Nunca; **A**=Algumas vezes; **S**=Sempre

APÊNDICE D – Tipos de Questões de Avaliação

Abaixo encontra-se um resumo das questões utilizadas na avaliação e os seus respectivos tipos para definição da medida, segundo especificação apresentada na seção “3.4 Métricas e Níveis de Pontuação”.

Componente	Característica de Qualidade	Atributo	Questões	Tipo de Questão
Documentação do usuário	Compleitude	Identificação do produto	A documentação do usuário impressa está identificada com: <input type="checkbox"/> nome do produto <input type="checkbox"/> versão do produto e data correspondente A documentação do usuário online está identificada com: <input type="checkbox"/> nome do produto <input type="checkbox"/> versão do produto e data correspondente	D
		Cobertura da documentação	A documentação do usuário impressa disponível cobre o quanto de funcionalidades do escopo da avaliação? <input type="checkbox"/> A documentação do usuário online disponível cobre o quanto de funcionalidades do escopo da avaliação? <input type="checkbox"/>	D
	Usabilidade	Apresentação visual	A documentação do usuário impressa: <input type="checkbox"/> possui tamanho de letra de fácil leitura <input type="checkbox"/> possui tipo de letra de fácil leitura <input type="checkbox"/> apresenta recursos gráficos com uma diagramação bem distribuída <input type="checkbox"/> utiliza recursos para destaque nas informações relevantes A documentação do usuário online: <input type="checkbox"/> possui tamanho de letra de fácil leitura <input type="checkbox"/> possui tipo de letra de fácil leitura <input type="checkbox"/> apresenta recursos gráficos com uma diagramação bem distribuída <input type="checkbox"/> utiliza recursos para destaque nas informações relevantes	B

		Operacionalidade	<p>A documentação do usuário impressa:</p> <p><input type="checkbox"/> apresenta um índice geral</p> <p><input type="checkbox"/> permite uma busca por informações</p> <p>A documentação do usuário online:</p> <p><input type="checkbox"/> apresenta um índice geral</p> <p><input type="checkbox"/> permite uma busca por informações</p> <p><input type="checkbox"/> utiliza o recurso de hipertexto através de links</p>	B
		Conformidade	<p>O formato da documentação impressa está de acordo com os padrões de apresentação definidos? <input type="checkbox"/></p> <p>O formato da documentação online está de acordo com os padrões de apresentação definidos? <input type="checkbox"/></p>	B
		Inteligibilidade	<p>O texto da documentação impressa:</p> <p><input type="checkbox"/> claro e preciso, não dando margem a interpretações ambíguas</p> <p><input type="checkbox"/> utiliza termos e explicações considerando o tipo de usuário a que se destina o produto</p> <p>O texto da documentação online:</p> <p><input type="checkbox"/> claro e preciso, não dando margem a interpretações ambíguas</p> <p><input type="checkbox"/> utiliza termos e explicações considerando o tipo de usuário a que se destina o produto</p>	B
			<p>O texto da documentação impressa:</p> <p><input type="checkbox"/> apresenta erros gramaticais</p> <p><input type="checkbox"/> apresenta erros ortográficos</p> <p>O texto da documentação online:</p> <p><input type="checkbox"/> apresenta erros gramaticais</p> <p><input type="checkbox"/> apresenta erros ortográficos</p>	C
		Apreensibilidade	A documentação impressa:	B

			<input type="checkbox"/> apresenta exemplos de uso do sistema para auxiliar na compreensão <input type="checkbox"/> os exemplos mostrados são claros e precisos, não dando margem a interpretações ambíguas	
	Funcionalidade	Acurácia	Na documentação do usuário impressa: <input type="checkbox"/> foram encontradas evidências de informações incorretas <input type="checkbox"/> foi observada alguma contradição entre os dados apresentados e o observado no sistema Na documentação do usuário online: <input type="checkbox"/> foram encontradas evidências de informações incorretas <input type="checkbox"/> foi observada alguma contradição entre os dados apresentados e o observado no sistema As informações apresentadas no material impresso são as mesmas da documentação online? <input type="checkbox"/>	C
Documentação de desenvolvimento	Compleitude	Identificação do produto	A documentação está identificada com: <input type="checkbox"/> nome do produto <input type="checkbox"/> versão do produto e data correspondente	D
		Cobertura da documentação	A documentação disponível cobre o quanto de funcionalidades do escopo da avaliação? <input type="checkbox"/>	D
	Usabilidade	Apresentação visual	A documentação: <input type="checkbox"/> possui tamanho de letra de fácil leitura <input type="checkbox"/> possui tipo de letra de fácil leitura <input type="checkbox"/> apresenta recursos gráficos com uma diagramação bem distribuída	D
		Operacionalidade	A documentação: <input type="checkbox"/> apresenta um índice geral <input type="checkbox"/> permite uma busca por informações	D
		Conformidade	O formato da documentação está de acordo com os padrões de	D

			apresentação definidos? ()	
		Inteligibilidade	O texto da documentação: () é claro e preciso, não dando margem a interpretações ambíguas	B
			O texto da documentação: () apresenta erros gramaticais () apresenta erros ortográficos	C
	Funcionalidade	Acurácia	Na documentação: () foram encontradas evidências de informações incorretas ou dúbias () foi observada alguma contradição entre os dados apresentados e o observado no sistema	C
Interface do usuário	Usabilidade	Inteligibilidade	O texto apresentado na interface: () está organizada em grupos segundo uma forma lógica facilmente compreendida pelo usuário (menus) () informa ao usuário sobre o que um componente faz (nomes claros, hints, etc) () utiliza um mesmo identificador para uma dada função em todo o sistema () possibilita a realização da tarefa desejada com um número reduzido de passos () exibe mensagens de orientação ao usuário () as mensagens exibidas orientam o usuário de forma efetiva e eficiente na execução da tarefa desejada	B
			A interface: () apresenta erros gramaticais () apresenta erros ortográficos () utiliza códigos e terminologia não significativa	C

			<p>O texto apresentado na interface:</p> <p><input type="checkbox"/> dá margem à interpretações ambíguas</p>	
			<p>As telas:</p> <p><input type="checkbox"/> apresentam uma distribuição uniforme de seu conteúdo, levando em consideração o espaço disponível</p> <p><input type="checkbox"/> apresentam somente informações necessárias e utilizáveis, sensíveis ao contexto</p> <p><input type="checkbox"/> seguem um padrão na distribuição dos objetos, facilitando o entendimento dos mesmos</p> <p><input type="checkbox"/> exibem as mensagens com bom aspecto visual, utilizando, com moderação, negrito, itálico e sublinhado</p> <p><input type="checkbox"/> utilizam tipos e tamanhos de letras de fácil visualização</p> <p><input type="checkbox"/> apresentam contrastes de cores, facilitando a leitura</p>	B
	Funcionalidade	Padrões internos	<p>A interface mantém uma padronização em relação:</p> <p><input type="checkbox"/> à configuração das janelas</p> <p><input type="checkbox"/> à apresentação de ícones</p> <p><input type="checkbox"/> às mensagens exibidas</p> <p><input type="checkbox"/> à apresentação de resultados ao usuário</p> <p><input type="checkbox"/> à posição de um determinado botão que executa uma mesma função em janelas ou caixas de diálogo semelhantes</p> <p><input type="checkbox"/> aos formatos dos campos de entrada de dados</p> <p><input type="checkbox"/> ao formato do cursor para uma mesma situação</p>	B
		Adequação	<p>A interface:</p> <p><input type="checkbox"/> possui funções de interface bem definidas, de modo a não deixar dúvidas sobre o que fazem (ex.: botões, menus)</p> <p><input type="checkbox"/> apresenta as funções do sistema numa estrutura de fácil acesso ao usuário</p>	B

Software	Funcionalidade	Segurança de acesso	O software: <input type="checkbox"/> faz uso de acesso autenticado por usuário <input type="checkbox"/> impede a entrada de usuários não autorizados <input type="checkbox"/> utiliza permissões por usuário para controle de acesso	A
		Adequação	Quantos requisitos presentes no sistemas são considerados úteis pelo usuário? <input type="checkbox"/> Quantos requisitos presentes no sistemas são considerados inúteis pelo usuário? <input type="checkbox"/> Quantos requisitos não estão presentes no sistema, mas são importantes para o usuário? <input type="checkbox"/>	E F F
		Acurácia	Quantos requisitos presentes no sistemas são considerados úteis pelo usuário e estão corretos? <input type="checkbox"/> Quantos requisitos presentes no sistemas são considerados úteis pelo usuário, mas precisam de modificação? <input type="checkbox"/>	E F
	Eficiência	Tempo de resposta	Há telas no software que são acessadas e tornam-se funcionais: <input type="checkbox"/> em menos de 10 segundos Há consultas realizadas em alguma tela onde o resultado apareça: <input type="checkbox"/> em menos de 10 segundos	B
	Confiabilidade	Registro de log	Há registros de logs nas funcionalidades para: <input type="checkbox"/> inserção de dados <input type="checkbox"/> alteração de dados <input type="checkbox"/> exclusão de dados <input type="checkbox"/> consulta de dados	B
		Maturidade	O software: <input type="checkbox"/> exibiu mensagens claras de orientação após a ocorrência de falhas	B
			O software: <input type="checkbox"/> apresentou falhas durante a sua execução	C