

Framework para Especialização de Modelos de Qualidade de Produtos de Software*

Lizandra Bays dos Santos^{1,2}, Eduardo Pretz¹

¹Universidade Feevale

Campus II – RS 239, 2755 – 93.352-000 – Novo Hamburgo – RS – Brasil

²Serpro – Serviço Federal de Processamento de Dados

Av. Augusto de Carvalho, 1133 – 90.010-390 – Porto Alegre – RS - Brasil

lizandra.santos@serpro.gov.br, epretz@feevale.br

Abstract. *The objective of this work is to propose a framework for building models for evaluation of software product quality specialized for the Serpro's business areas. For development of the proposal have used the theoretical basis about testing and software quality and international standards such as ISO series, besides the application in a case study.*

Resumo. *O objetivo deste trabalho é propor um framework para construção de modelos de avaliação de qualidade de produtos de software especializados para as áreas de negócio do Serpro. Para desenvolvimento da proposta utilizou-se a base teórica relacionada à área de teste e qualidade de software e os modelos internacionais como normas das séries ISO, além da aplicação em um estudo de caso.*

1. Introdução

Diante da complexidade e popularização dos sistemas computacionais, muitas empresas desenvolvedoras de software têm buscado melhoria da qualidade por meio de métodos de gestão e certificações internacionais, tais como ISO 9000, CMMI (*Capability Maturity Model Integration*), além do padrão nacional MPS.BR (Melhoria de Processo do Software Brasileiro). Com este alinhamento, o Serpro (Serviço Federal de Processamento de Dados), empresa pública que desenvolve sistemas para o governo federal, conquistou em 2004 a qualificação CMMI nível 2 para o PSDS (Processo Serpro de Desenvolvimento de Soluções), que é a metodologia integrada para construção de soluções em TIC da organização (Serpro, 2009b). Atualmente, o PSDS é aderente ao nível 3 do CMMi.

Recentes movimentos da alta direção do Serpro apontam para a busca de maior produtividade no desenvolvimento de software, sem perdas na qualidade. O PSDS, assegura a qualidade do processo de desenvolvimento, porém, não garante a qualidade dos produtos. A estruturação de Centros de Testes locais em todos os departamentos de desenvolvimento aparece como uma aposta da Diretoria de Desenvolvimento na qualidade do produto, alinhada à busca da produtividade. Porém, uma área ainda imatura no processo, que é a definição dos critérios de aceitação do produto pela área de

**Este artigo foi desenvolvido no curso de Pós-Graduação em Teste e Garantia da Qualidade de Software, resultante de uma parceria entre a FEEVALE e a Hewlett-Packard Brasil Ltda. propiciada por recursos provenientes da contrapartida da redução do IPI regulamentada pela Lei n° 8.248, de 1991*

negócio, faz com que os níveis de cobertura e profundidade dos testes realizados pelos Centros de Testes, sejam influenciados por diversas variáveis, entre elas, a complexidade e o tamanho do sistema, a escassez de recursos para testes e o prazo para entrega ao cliente. Em consequência, não há um padrão de qualidade dos produtos de software entregues para homologação.

Diante deste cenário, um modelo para certificação de qualidade dos produtos do Serpro, mostra-se como uma alternativa para guiar, não só a definição dos critérios de aceitação, mas também os testes realizados pelos Centros de Testes, que poderiam emitir um selo atestando o nível de qualidade do produto.

Este artigo propõe-se a apresentar um *framework* que viabilize a especialização de modelos de certificação de produto de software. Para tal, a seção 2 traz uma revisão bibliográfica, pela qual foram estudados os principais conceitos e alguns trabalhos relacionados sobre certificação de software, bem como as bases que fundamentam o *framework* proposto. Na seção 3 é discorrido sobre a proposta e o escopo do artigo e na seção 4 relatado um estudo de caso e a metodologia aplicada. A seção 5, por sua vez, traz conclusões e sugere trabalhos futuros.

2. Revisão Bibliográfica

Segundo o IEEE (1990), uma das definições para garantia da qualidade é que é o conjunto de atividades planejadas para avaliar os processos pelos quais os produtos são desenvolvidos ou fabricados. Já o controle da qualidade é o conjunto de atividades planejadas para avaliar a qualidade dos produtos desenvolvidos ou fabricados. Desta forma, pode-se concluir que a garantia da qualidade está focada na qualidade do processo, enquanto que controle da qualidade está relacionado à qualidade do produto.

Mesmo compreendendo que o objetivo maior da disciplina de qualidade seja a qualidade do produto final entregue ao usuário, e reconhecendo a fundamental importância de processos normatizados e controlados para o desenvolvimento de produtos, a qualidade do processo, por si só, não garante a qualidade do produto, fazendo-se necessária a combinação desta com controles de qualidade do produto.

2.1. Trabalhos relacionados à certificação de produto de software

Avaliar um produto de software é atribuir um certo valor a esse produto, com base em requisitos pré-estabelecidos e sob demanda de um patrocinador (Guerra, 2008). Um método para avaliação e certificação de produto envolve um modelo de qualidade e um processo de avaliação (Santos, 2002 e Guerra, 2008).

Algumas instituições brasileiras desenvolveram métodos de avaliação da qualidade de produtos de software. Entre estes métodos, alguns inspiraram esta pesquisa e merecem destaque.

Dentro do PROIMPE (Programa de Estímulo ao Uso de Tecnologia da Informação em Micro e Pequenas Empresas), a Riosoft (2009) lançou um processo de Avaliação de Conformidade de Produtos de Software. Neste modelo a qualidade do produto também é avaliada em relação a padrões internacionais, como a NBR ISO/IEC 9126, por meio de listas de verificação, além da execução de testes previamente planejados pelo produtor do software. As empresas avaliadas com sucesso tem direito a um certificado de qualidade (Riosoft, 2009).

Outra instituição que desenvolveu um selo de qualidade para produtos de software foi a Assespro (Associação das Empresas de Tecnologia da Informação, Software e Internet). Teste OK é um Selo de Certificação de Qualidade de Software com o objetivo de garantir e atestar que o software possui qualidade, sendo o mesmo aplicado em softwares com desenvolvimento finalizado ou que já estejam em uso no mercado. Neste modelo, várias técnicas são utilizadas durante o processo de certificação, entre elas: performance/carga, durabilidade, confiabilidade/recuperação, instalação/atualização, acessibilidade, funcionalidade, interface de usuário e usabilidade (Assespro, 2009).

O MEDE-PROS® é um método de avaliação genérico desenvolvido pelo então CENPRA (Centro de pesquisa Renato Archer), atual CTI/MCT (Centro de Tecnologia da Informação Renato Archer) que visa avaliar o software com base no modelo de qualidade explícito nas normas NBR ISO/IEC 9126 e NBR ISO/IEC 12119. Neste método, todos os aspectos do software são avaliados tendo o mesmo valor agregado. O modelo considera um conjunto de elementos que compõem um software e os atributos da qualidade que estão relacionados a estes elementos. Cada atributo, por sua vez, é relacionado a um conjunto de questões que compõem as Listas de Verificação (Guerra, 2008).

Uma iniciativa desenvolvida no setor público é o 5CQualiBr. Trata-se de um ambiente dedicado à qualidade de software dentro do Portal do Software Público Brasileiro (SPB). O vetor de Qualidade do Produto deste programa apresenta um modelo de qualidade de produto de software para as aplicações disponibilizadas no portal SPB, que são aplicações desenvolvidas na Administração Pública, empresas ou universidades, compartilhadas e distribuídas sob licença de software livre (SPB, 2009). O modelo é composto por grupos que representam áreas de interesse com objetivo de avaliar a qualidade. Estes grupos são divididos em características (características da qualidade) que são derivados em atributos e métricas. As métricas, por sua vez, são questões que devem ser respondidas por meio de medições.

Os métodos pesquisados citados acima, ou apresentam-se como métodos genéricos, não estando especializados para nenhuma área de domínio, ou buscam a especialização por meio da análise da documentação produzida pelos desenvolvedores.

Outra característica comum dos modelos pesquisados é que todos preconizam que avaliação seja realizada por avaliadores capacitados em laboratórios credenciados.

Modelos genéricos não avaliam a qualidade do ponto de vista do domínio da aplicação, considerando as funcionalidades, tecnologias e ambientes envolvidos. Para elaboração de um método especialista, é necessário apresentar a base teórica das normas de qualidade existentes para produto de software genérico e, além disso, contemplar toda a especificação de requisitos do software em questão (Guerra, 2008).

Dentre os trabalhos relacionados pesquisados, o método baseado no MEDE-PROS® e em requisitos especificados em um edital, que foi especializado e aplicado no estudo de caso PNAFM (Programa Nacional de Apoio à Gestão Administrativa e Fiscal dos Municípios Brasileiros), propõe-se a ser um modelo especialista. Segundo Maintinguer (2004), o modelo seguiu a estrutura de listas de verificação proposta no modelo MEDE-PROS® para a avaliação dos requisitos não funcionais. Para a avaliação dos requisitos funcionais, a proposta vale-se de um Guia de Avaliação, o qual é composto, além dos atributos de qualidade relacionados com o requisito da aplicação,

de um passo a passo para orientar o avaliador quanto às operações a serem realizadas na utilização do software. A autora destaca ainda, que a sequência de passos deve ser obtida por meio de entrevistas com os possíveis usuários do sistema e que o responsável pela elaboração do guia de avaliação deve estar capacitado na área de domínio da aplicação.

2.2. Bases que fundamentam a proposta deste artigo

A seguir, são apresentadas as referências que serviram de base para fundamentar a proposta deste artigo.

● RRBT

A estratégia RRBT (*Risk & Requirement Based Testing*) preconiza que, aos testes que cobrem os maiores riscos seja dado tratamento especial. Além disso, um inventário de requisitos é considerado. Os requisitos são relacionados aos riscos do produto e, com base nesta informação, são criados os testes que cubram os principais riscos (Pinkster, 2006).

Vale destacar a distinção entre riscos de projeto e risco de produto. Segundo Pinkster (2006), riscos do projeto são aqueles relacionados à gestão do processo de testes. A avaliação destes riscos e medidas de redução devem ser incluídas no plano de testes. Riscos do produto são aqueles que afetam diretamente o negócio. No início do projeto de testes, o gerente de testes está focado, principalmente em riscos do projeto. Com o progresso do projeto, a atenção se volta para para os riscos do produto, pois os testes devem cobrir estes riscos.

Riscos isolados não são insuficientes para as bases da avaliação do produto. Requisitos também são necessários. Uma maneira apropriada de relacionar requisitos e riscos é pela análise do funcionamento de determinada funcionalidade, e os efeitos de seu não funcionamento (Pinkster, 2006).

● NBR ISO/IEC 9126-1

A norma NBR ISO/IEC 9126-1, sob o título geral "Engenharia de software - Qualidade do produto", atualmente reestruturada na série ISO/IEC 25000 - *Software Engineering - Software product Quality Requirements and Evaluation* (SquaRE), mais especificamente na divisão de Medição da Qualidade, fornece um modelo baseado em um conjunto de características que um produto de software deve apresentar, o qual é organizado hierarquicamente em subcaracterísticas e atributos, conforme figura 1.

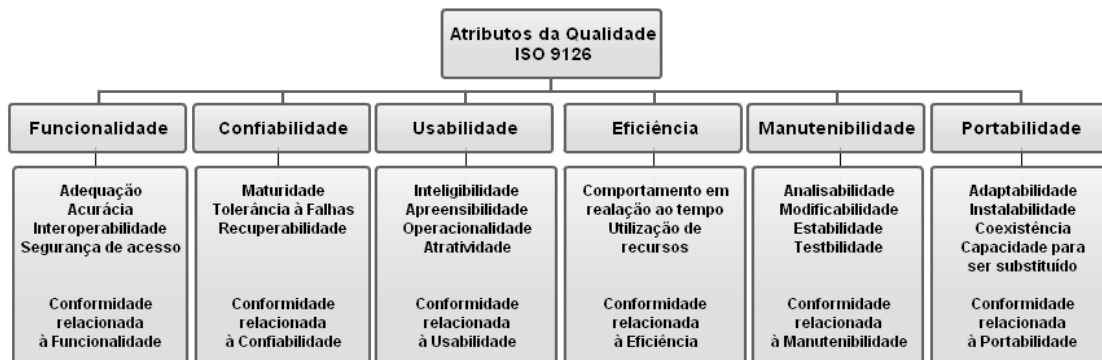


Figura 1 — Atributos da qualidade (NBR ISO/IEC 9126-1, 2003)

A qualidade do produto de software pode ser avaliada medindo-se os atributos internos (tipicamente medidas estáticas de produtos intermediários), os atributos externos (pela medição do comportamento do código quando executado) ou os atributos de qualidade em uso (NBR ISO/IEC 9126-1, 2003). Em sua parte 2, a norma NBR ISO/IEC 9126 fornece um conjunto de métricas que permitem realizar a medição das subcaracterísticas da qualidade.

A norma recomenda que, para a avaliação de qualidade de um produto de software, seja definido um modelo de qualidade e que este modelo seja usado na definição das metas de qualidade para o produto. Devem ser alocados recursos para avaliação considerando os diferentes tipos de medições, dependendo dos objetivos de negócios, da natureza do produto e dos processos utilizados no projeto (ISO/IEC TR 9126-2, 2003).

● GQM

A abordagem *Goal Question Metric* (GQM) foi definida originalmente por Basili (1994) para avaliar defeitos em uma série de projetos do Centro de Vão Espacial da NASA. Baseia-se na suposição de que um processo de medição, para ser efetivo, deve ser focado em objetivos específicos, e sua interpretação deve ser baseada no entendimento desses objetivos (Basili, 1994). Trata-se de uma abordagem orientada a objetivos para a medição de produtos e processos de software, que apóia a definição *top-down* do processo de medição e a análise *bottom-up* dos dados resultantes. Entre as vantagens observadas neste método, destacam-se, apoio na identificação de métricas úteis e relevantes, e a análise e interpretação dos dados coletados.

O modelo GQM envolve três níveis, conforme figura 2. O processo de definição de objetivos, no nível conceitual, é uma das etapas mais críticas para o sucesso desta abordagem. Um objetivo, possui um propósito e três coordenadas, a saber:

Propósito: um verbo que represente um objetivo, por exemplo "*avaliar*"

Questão: um adjetivo do objeto, por exemplo "*a maturidade*"

Objeto: o objeto em avaliação, por exemplo "*do software*"

Ponto de vista: por exemplo, "*segundo a perspectiva do cliente*".

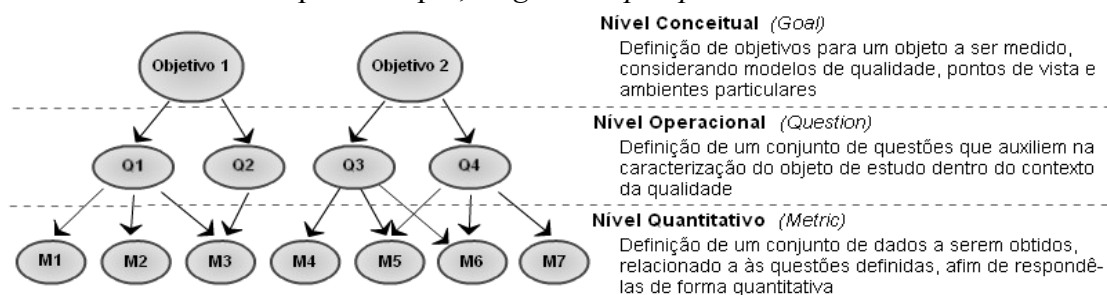


Figura 2 - Estrutura da abordagem GQM (Basili, 1994)

● NBR ISO/IEC 14598

Um processo de avaliação da qualidade de produtos de software pode ser encontrado na série NBR ISO/IEC 14598, que está incluída na série 25000 (SquaRE) na parte 2504n – Divisão de Avaliação da Qualidade.

A norma NBR ISO/IEC 14598-1 contém conceitos de como avaliar a qualidade de software e define um modelo de processo de avaliação genérico, conforme figura 3.

O processo proposto pode ser usado para avaliar produtos já existentes ou produtos intermediários, isto é, em desenvolvimento. Pode ser utilizada por laboratórios de avaliação, fornecedores de software, compradores de software, usuários e entidades certificadoras, cada qual com seu objetivo. Em termos de características de qualidade, pode ser usada a norma NBR ISO/IEC 9126-1, já citada nesta seção, por ser uma norma reconhecida internacionalmente.

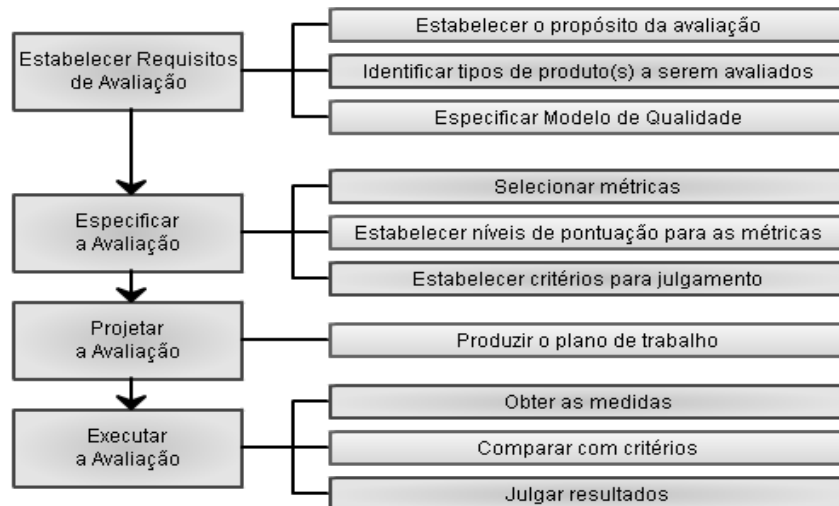


Figura 3 — Processo de avaliação segundo NBR ISO/IEC 14598-1.

A parte 3 da norma NBR ISO/IEC 14598, que trata do processo de avaliação para desenvolvedores de software, aplica-se a todo software e em todas as fases do ciclo de vida de desenvolvimento. A norma NBR ISO/IEC 14598-4, que trata do processo de avaliação para adquirentes de software, deve ser usada na aceitação ou seleção de um produto.

● Técnicas de Teste

Técnicas de testes são os meios utilizados pelos testadores para realizarem os objetivos de testes e há vasta bibliografia disponível abordando o assunto. As técnicas são dependentes dos tipos de testes, e muitas vezes se confundem com eles. Tipos de testes são relacionados aos objetivos dos testes, ou seja, "o que testar", enquanto as técnicas dizem respeito a "como testar". A seguir são apresentadas, de forma resumida, algumas técnicas e tipos de testes que são citadas no estudo de caso realizado para este artigo.

Tabela 1 — Tipos e técnicas de teste

Tipo/Técnica	Descrição
Teste Baseado na Especificação	Conjunto de técnicas pelas quais os casos de testes são derivados de modelos, formais ou informais, que especificam um problema a ser resolvido, o software ou seu componente. Testes podem ser especificados a partir de casos de uso ou cenários de negócios (ISTQB, 2007).
Análise de Valores Limites	Técnica de teste baseada em especificação, pela qual casos de testes são desenvolvidos com foco nos valores fronteira para entradas e saídas de determinada função (QAI, 2006).
Indução de Erros	Técnica que visa determinar a capacidade da aplicação de processar transações incorretas. Consiste em testar a introdução de erros, seu processamento, controles e reteste após a correção (QAI, 2006).

Teste de <i>stress</i>	Tipo de teste o qual valia um sistema ou componente em relação e além dos limites de seus requisitos especificados (ISTQB, 2007).
Teste de Carga	Tipo de teste que mede o comportamento de um componente ou sistema por meio do aumento de carga, por exemplo, número de usuários paralelos e/ou número de transações para determinar qual o tamanho de carga que pode ser suportada pelo componente ou sistema (ISTQB, 2007).
Teste de Longa Duração	Também conhecidos como testes de longevidade, saturação, imersão ou resistência, são técnicas que procuram por problemas de performance ou segurança que só aparecem depois de um longo período de operação do software. Diferem dos testes de estresse típicos, que aplicam uma carga pesada ao sistema, na medida em que verificam se o <i>buffer</i> de dados e as filas de mensagens operam corretamente durante longos períodos (Woody, 2009).
Teste de Penetração	Método que avalia a segurança de um sistema ou de uma rede, simulando ataques maliciosos. O processo envolve uma análise das vulnerabilidades do sistema (QAI, 2006).

Na seção seguinte é apresentada a proposta do artigo e a metodologia utilizada.

3. *Framework* para especialização de modelos de qualidade do Serpro

Nesta seção é apresentada a proposta de um *framework*, pelo qual o Serpro possa especializar modelos de qualidade para suas diferentes áreas de negócio, visando, não só a avaliação e certificação da qualidade dos produtos de software, mas também a definição dos critérios de aceitação destes produtos, conforme ilustrado na figura 4.



Figura 4 - Objetivos do modelo de qualidade especializado para o Serpro

Desta forma, entende-se que o modelo completo deve ser composto de três processos, conforme ilustrado na figura 5.

O processo de especialização do modelo de qualidade visa apoiar a definição dos atributos da qualidade específicos para o domínio da aplicação, baseado nos riscos e requisitos do produto. O processo de medição da qualidade, contempla as métricas e técnicas para mensuração das evidências dos atributos da qualidade. Já o processo de avaliação da qualidade, por sua vez, deve contemplar uma sistemática de pontuação e critérios de julgamento para as medidas coletadas, de modo que o produto possa ser certificado e comparado com outros.

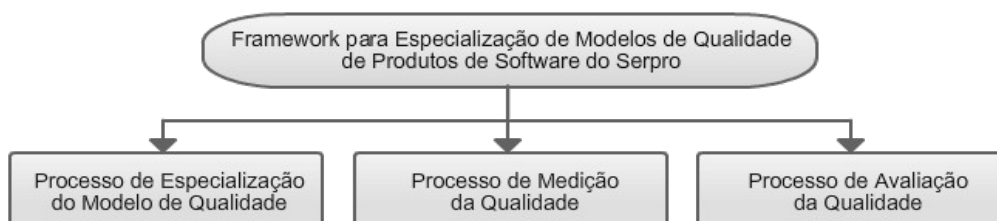


Figura 5 - *Framework* para Especialização de modelos de qualidade de software

O escopo deste trabalho limita-se aos dois primeiros processos propostos: especialização do modelo e medição da qualidade. A idéia principal desta proposta pode ser resumida como uma rastreabilidade desde o requisito até a técnica de teste que permitirá a medição da qualidade, considerando os riscos e atributos da qualidade associados. Uma representação gráfica pode ser observada na figura 6.

No processo de desenvolvimento do Serpro (PSDS), as áreas de processo do CMMI são agrupadas em macroatividades. As áreas de desenvolvimento e gestão de requisitos são contempladas na macroatividade de Requisitos, cujo principal produto é o Modelo de Requisitos (MR). O MR é um conjunto de artefatos e diagramas que contextualizam e especificam as necessidades, funcionalidades e requisitos que o produto de software se propoe a atender. É o MR que fornece subsídios para todo o desenvolvimento, da análise e projeto, passando por implementação e testes, até a homologação (Serpro, 2009a).

A área chave relacionada à gestão de riscos, é atendida no PSDS na macroatividade de Gestão de Projetos de Software (GPS), da mesma forma que atividades de planejamento, monitoramento e controle de projetos (Serpro, 2009a). Porém, no âmbito de GPS, é comum que a equipe de desenvolvimento se preocupe com riscos de projeto, ficando a gestão de riscos do produto a cargo da área de negócios, de responsabilidade da macroatividade de Negócio.

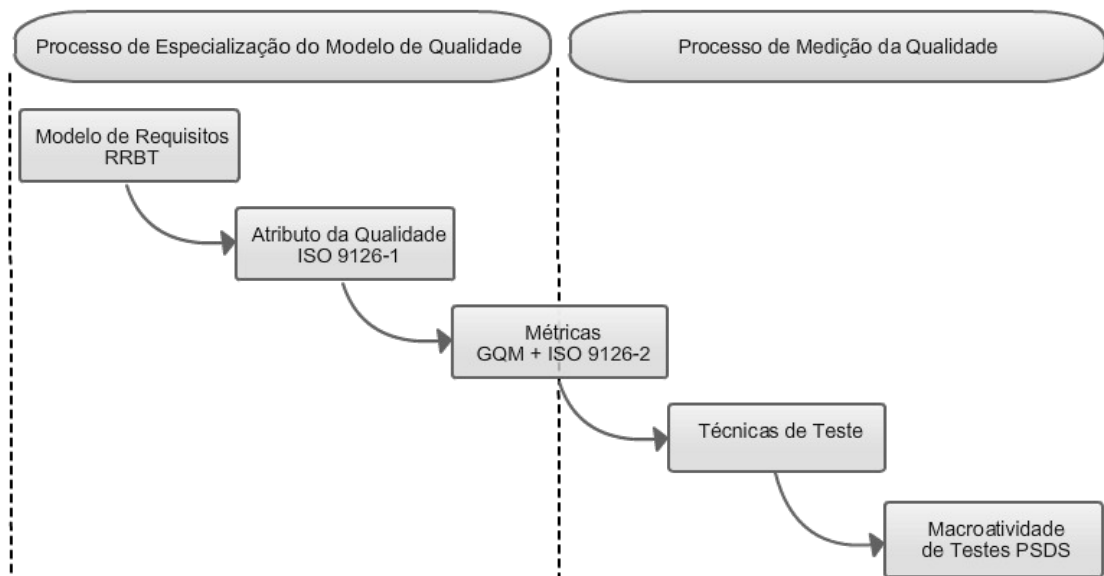


Figura 6 - Processos de especialização do modelo e de medição da qualidade

Para o processo de especialização do modelo da qualidade, a proposta é que a partir do MR seja aplicada a técnica RRBT para associação dos riscos e priorização dos testes. James Bach (2003), citou em seu artigo *The Challenge of "Good Enough" Software* que, em um contexto particular, alguns fatores da qualidade podem ser mais importantes do que outros. Segundo o autor, alguns fatores prejudicam (diminuem) os outros e existem maneiras nem tão simples e nem tão baratas de medi-los. Alinhado à esta idéia, a técnica de teste baseado em riscos e requisitos (RRBT) apresenta vantagens pois permite priorizar os testes e distribuir seu esforço de forma coerente. Esta proposta sugere que, para o contexto do Serpro, a análise de riscos seja realizada pela área de negócios com participação do líder do projeto de software.

Associados os devidos riscos aos requisitos, faz-se necessário associar os atributos da qualidade, segundo a NBR ISO/IEC 9126-1, para completar o modelo especialista. Para apoiar a definição das metas de qualidade e seleção de métricas correspondentes, este trabalho propõe o método GQM. As metas ou objetivos (*goals*) são baseadas nos riscos já associados aos requisitos. As questões (*questions*), assim como as métricas (*metrics*) baseiam-se na NBR ISO/IEC 9126-2.

De posse das métricas inicia-se o processo de medição da qualidade, associando técnicas de testes que permitirão coletar os dados necessários para medição. Neste momento, já tem-se subsídios para dar entrada na macroatividade de Testes do PSDS, a qual preconiza planejamento de testes e projeto de casos de testes, prioritariamente com base em riscos e requisitos. As medidas associadas à cada métrica, podem ser obtidas como resultado da execução dos testes.

4. Estudo de caso de aplicação do processo de especialização do modelo de qualidade

A metodologia utilizada para o estudo de caso foi baseada em riscos e requisitos conhecidos para um sistema em desenvolvimento pelo Serpro. Foram escolhidos requisitos relevantes e identificados os riscos associados com apoio da documentação do sistema e informações obtidas com a equipe de desenvolvimento.

Trata de um sistema *stand-alone* que dá suporte à ações em campo de fiscalizações de tributos federais. Os requisitos abordados neste estudo de caso, referem-se a funcionalidades de recepção e envio de dados a bases centralizadas.

Neste caso, a primeira etapa foi considerada cumprida, pois as áreas de negócio, de tecnologia, de suporte, de segurança e a equipe de desenvolvimento do sistema, já haviam produzido um documento relacionando riscos aos requisitos, o qual foi utilizado como base para o estudo de caso. Sendo assim, passou-se para a etapa seguinte, que foi a identificação das características e subcaracterísticas da qualidade para cada risco. O resultado é apresentado na tabela 2.

Tabela 2 — Especialização do Modelo de Qualidade para o estudo de caso

RRBT		ISO 9126	
Requisito	Risco	Característica	Subcaracterística
Envio e recepção de nova versão à base centralizada.	R001 - Indisponibilidade do sistema para o usuário.	Confiabilidade	Tolerância a Falhas
			Maturidade
Envio e recepção de informações dos sistemas relacionados.	R002 - Insuficiência dos recursos envolvidos com a produção do sistema, causando indisponibilidade.	Eficiência	Utilização de Recursos
	R003 – Interceptação de informações sigilosas no tráfego de rede utilizado pelo sistema.	Funcionalidade	Segurança de Acesso
R004 – Acesso liberado, aos usuários da aplicação, às informações que ficam inseridas no banco local instalado na estação de trabalho do usuário.			

Para exemplificar a aplicação da proposta neste estudo de caso, foram escolhidos riscos relacionados com confiabilidade, eficiência e segurança. Vale destacar que, o primeiro risco tem relação com a disponibilidade do sistema. A norma NBR ISO/IEC 9126-1, define como disponibilidade a capacidade de um produto de software de estar pronto para executar uma função requisitada num dado momento, sob condições

especificadas de uso e que a mesma deve ser avaliada como uma combinação da maturidade (a qual controla a frequência de falhas), tolerância a falhas e recuperabilidade (a qual controla o período de tempo inativo após cada falha).

Dando continuidade à aplicação do processo, a próxima etapa foi definir os objetivos e métricas. Nas tabelas 3, 4, 5 e 6 são apresentadas a aplicação do método GQM e sugestão de tipos e técnicas de teste, para os riscos R001, R002, R003 e R004, respectivamente.

Para avaliação da Confiabilidade, é necessário que o sistema seja submetido a um certo período de tempo em operação e que sejam simuladas situações que causem falhas e quedas no mesmo.

Tabela 3 - Aplicação de GQM para o Risco 001

R001 - Confiabilidade / Maturidade		
Objetivo		
Propósito: Avaliar		
Questão: a capacidade de prevenção de falhas		
Objeto: do sistema		
Ponto de vista: conforme usuário		
Questão	Métrica	Tipo/Técnica de Teste
Quantas falhas foram detectadas durante um período definido de experimentação?	Número de falhas detectadas / Número de casos de testes	Teste de longa duração Testes de indução de falhas Teste baseados na especificação
R001 - Confiabilidade / Tolerância a falhas e Recuperabilidade		
Objetivo		
Propósito: Avaliar		
Questão: a disponibilidade		
Objeto: do sistema		
Ponto de vista: conforme usuário		
Questão	Métrica	Tipo/Técnica de Teste
Quanto padrões de faltas são mantidos sob controle para evitar falhas críticas e sérias?	Número de ocorrências de falhas sérias e críticas evitadas conforme os casos de testes de indução de falhas / Número de casos de testes de indução de falhas executados	Teste de longa duração Testes de indução de falhas
Quão disponível é o sistema para uso durante um período de tempo específico?	a) { Tempo de operação / (Tempo de operação + Tempo de reparo) } b) Total de casos em que o sistema estava disponível e foi utilizado com sucesso pelo usuário / Número total de casos em que o usuário tentou usar o software durante um período de tempo	Teste de longa duração Testes de indução de falhas
Qual é o tempo médio em que o sistema fica indisponível quando uma falha ocorre, antes da inicialização?	Tempo ocioso total (indisponível) / Número de quedas do sistema	Teste de longa duração Testes de indução de falhas
Qual o tempo médio que o sistema leva para completar a recuperação desde o início?	Soma de todos os tempos de recuperação do sistema inativo em cada oportunidade / Número total de casos em que o sistema entrou em recuperação	Teste de longa duração Testes de indução de falhas

Para o risco R002, que tem relação com a eficiência na utilização de recursos de modo que não resulte em indisponibilidade do sistema, aparecem as técnicas de teste de stress e carga, e as medidas são com base nas mensagens de erro apresentadas pela aplicação durante as situações de simulação de carga máxima e valores limites. Para

este tipo de teste é fundamental a utilização de ferramentas automatizadas para as simulações necessárias. A capacidade de transmissão deve ser medida como uma relação da quantidade de bits por uma unidade de tempo ou por usuário, funcionalidade, etc.

Tabela 4 - Aplicação de GQM para o Risco 002

R002 – Eficiência /Utilização de Recursos		
Objetivo		
Propósito: Avaliar Questão: a eficiência na utilização de recursos Objeto: de produção Ponto de vista: conforme usuário		
Questão	Métrica	Tipo/Técnica de Teste
Qual é o limite absoluto de transmissões necessárias para cumprir uma função?	Número máximo de mensagens de erro e falhas relacionadas a transmissão do primeiro ao último item avaliado / Máximo requerido de mensagens de erro e falhas relacionadas a transmissão	Teste stress Valores limites de usuários simultâneos Valores limites de dados trafegados (simular carga máxima)
O sistema é capaz de desempenhar tarefas dentro da capacidade de transmissão esperada?	Capacidade de transmissão / Capacidade de transmissão específica projetada para ser usada pelo software durante sua execução	Teste stress Valores limites de usuários simultâneos Valores limites de dados trafegados (simular carga máxima)

Tanto para Confiabilidade como para Eficiência, é necessário conhecer os valores mínimos necessários, desejáveis e aceitáveis para os requisitos de modo que as medidas possam ser comparadas e a devida avaliação seja realizada. Neste estudo de caso, o modelo de requisitos especifica que são estimados 900 usuários simultâneos com carga aproximada de 40 MB trafegados por usuário, na funcionalidade de receber uma nova versão da aplicação da base centralizada. Para a funcionalidade de integração com outros sistemas, a estimativa é igualmente de 900 usuários simultâneos.

Para avaliações da integridade dos dados e segurança de acesso, são sugeridos testes de penetração. São testes que simulam ataques às vulnerabilidades do sistema. Estes tipos de testes são tipicamente executados por especialistas em segurança e baseados em padrões de ataques e de vulnerabilidades.

Tabela 5 - Aplicação de GQM para o Risco 003

R003 – Funcionalidade / Segurança de Acesso		
Objetivo		
Propósito: Avaliar Questão: a integridade dos dados Objeto: do sistema Ponto de vista: conforme usuário		
Questão	Métrica	Tipo/Técnica de Teste
Qual é a frequência de eventos de corrupção de dados?	a) 1 - (Número de vezes que o maior evento de corrupção de dados ocorreu / Número de casos de testes executados que causaram eventos de corrupção de dados) b) 1- (Número de vezes que o menor evento de corrupção de dados ocorreu / Número de casos de testes executados que causaram eventos de corrupção de dados)	Testes de penetração Simulação de padrões de ataques às vulnerabilidades da conexão

A segurança de acesso, foi derivada em duas questões, uma relacionada a integridade dos dados (tabela 5) e outra ao controle de acesso (tabela 6). Isto porque os riscos R003 e R004 sugeriam estas questões.

Tabela 6 - Aplicação de GQM para o Risco 004

R004 – Funcionalidade / Segurança de Acesso		
Objetivo		
Propósito: Avaliar Questão: o controle de acesso Objeto: ao sistema Ponto de vista: conforme usuário		
Questão	Métrica	Tipo/Técnica de Teste
Quão completa é a trilha de auditoria sobre o acesso do usuário ao sistema e dados?	Número de acessos do usuário ao sistema e dados gravados no log de acesso / Número de acessos do usuário ao sistema e dados realizados durante a avaliação	Testes de penetração; Simulação de padrões de ataques às vulnerabilidades da conexão
Quão controlável é o acesso ao sistema?	Número (tipos diferentes) de operações ilegais detectadas/ Número (tipos diferentes) de operações ilegais especificadas	Testes de penetração Simulação de padrões de ataques às vulnerabilidades da conexão

O estudo de caso limitou-se à realização dos processos de especialização do modelo de qualidade e ao início do processo de medição da qualidade, o qual só pode ser completado com a execução dos testes. De posse das métricas e técnicas de testes é possível iniciar o planejamento e projeto de testes, para execução da macroatividade de testes do PSDS.

5. Conclusão

Entende-se como principal contribuição deste trabalho, a consolidação de modelos e metodologias reconhecidas e amplamente aplicadas, em uma proposta de *framework* que pode ser utilizado para elaboração de modelos especialistas de avaliação de qualidade de produtos de software.

Sutilmente, este artigo contribui ainda, para a discussão quanto à importância da qualidade do produto de software, na medida em que foram estudados e são elencados alguns métodos de avaliação da qualidade de produto desenvolvidos por instituições brasileiras.

A aplicação dos dois processos do *framework* cobertos pelo escopo deste artigo no estudo de caso evidenciou que é viável a utilização da técnica RRBT com base nos artefatos preconizados pelo PSDS. A única dificuldade encontrada foi a falta de um pouco mais de detalhamento nos requisitos não funcionais, principalmente na especificação explícita das características de performance e disponibilidade. Esta dificuldade é muito natural na engenharia de software, pois os usuários costumam ter dificuldade em especificar requisitos não funcionais devido à falta de conhecimento técnico.

Vale destacar, neste momento, que a norma ISO 9126 não fornece atributos e métricas para avaliação de características relacionada à tecnologia, tais como restrições tecnológicas, características de teleprocessamento ou conectividade, etc. No estudo de caso realizado, foram selecionados riscos relacionados à aplicação, de modo que fosse possível aplicar a proposta tal qual a mesma foi delineada, com base nas características da qualidade e métricas da ISO 9126. Porém, no modelo de requisitos que subsidiou este estudo, foi observada a existência de vários riscos relacionados a tecnologia, tais

como, configuração do serviço de mensageria, ausência de conexão segura, vulnerabilidades no código, possibilidade de descompilação do programa no desktop, entre outras. Neste sentido, avalia-se que a proposta pode ser aplicada com uma pequena adaptação: relacionar características tecnológicas, em vez de características da qualidade e, na aplicação do GQM, indicar métricas alternativas que permitam responder as questões relativas aos riscos de tecnologia. As técnicas de testes conhecidas, certamente permitirão a medição.

Em relação a perspectivas futuras, entende-se que esta pesquisa apresenta-se como um instrumento importante para contribuir com as metas de qualidade e produtividade do Serpro. É sabido que a qualidade do software é uma integração entre qualidade do processo e qualidade do produto. Além disso, o Serpro vem, há alguns anos, investindo na qualidade do processo com o PSDS. A estruturação de Centros de Testes regionais demonstra o interesse da empresa em investir na qualidade do produto. Combinando o *framework* proposto por este artigo com o modelo de Centros de Testes quem vem sendo implantado pela Diretoria de Desenvolvimento, seria possível não só medir, quantitativamente a qualidade dos produtos, como compará-los. Para tal, faz-se necessária a continuação da pesquisa e elaboração do terceiro processo do *framework*, que é processo de avaliação da qualidade, o qual deve conter uma sistemática de pontuação e critérios de julgamento para as métricas.

Ainda como sugestão de trabalhos futuros, o modelo pode ser validado em estudos de casos de outras organizações. Apesar da motivação para este artigo ter partido do contexto do Serpro, o *framework* pode ser aplicado para especialização de modelos de qualquer área de domínio de aplicações de software.

Outra consideração importante é que há um esforço dispendido para utilização desta proposta. Um certo esforço é necessário, tanto para este, quanto para qualquer método de avaliação e teste de software. Neste sentido, uma avaliação da relação custo-benefício deve ser considerada na tomada de decisão de avaliar ou não a qualidade de um software, sendo o modelo recomendado para projetos estratégicos e para sistemas de vida longa com muitos projetos de manutenção evolutiva, que é uma característica dos sistemas do Serpro.

6. Referências

Assespro (2009), Teste OK! Associação das Empresas Brasileiras de Tecnologia da Informação. Disponível em: <<http://www.testeok.com.br/>>. Acesso em: 13/09/2009.

Bach, J. (2003), The Challenge of "Good Enough" Software. Disponível em: <<http://www.satisfice.com/articles/gooden2.pdf>>. Acesso em: 04/12/2009.

Basili, V. R., Caldiera, G. e Rombach, H. D. (1994), The Goal Question Metric Approach. Disponível em: <<ftp://ftp.cs.umd.edu/pub/sel/papers/gqm.pdf>>. Acesso em: 02/12/2009.

Chrissis, M. B., Konrad, M. e Shrum, S. (2004), CMMI – Guidelines for Process Integration and Product Improvement. Addison-Wesley.

Guerra, A. C. e Colombo, R. M. T. (2008), Qualidade de Produto de Software. PBQP/MCT. Disponível em: <<http://www.mct.gov.br/index.php/content/view/2867.html#lista>>. Acesso em: 13/09/2009.

IEEE (1900), Standard Glossary of Software Engineering Terminology. Institute of Electrical and Electronics, Inc.

ISO/IEC TR 9126-2 (2003). Software engineering - Product quality - Part 2: External metrics. ISO/IEC.

ISTQB (2007), Base de Conhecimento para Certificação em Teste. ISTQB Foundation Level Syllabus. Versão 2007br. Brazilian Software Testing Qualifications Board. Disponível em: <http://www.bstqb.org.br/uploads/docs/syllabus_2007br.pdf> Acesso em: 01/12/2009.

Maitinguer, S. T. (2004), Um Método de Avaliação Especialista para produtos de Software, desenvolvido a partir dos requisitos de um Edital. Trabalho Final de Mestrado Profissional. Universidade Estadual de Campinas, Faculdade de Engenharia Mecânica. Disponível em: <<http://libdigi.unicamp.br/document/?code=vtls000332940>>. Acesso em: 09/11/2009.

Softex (2007), MPS.BR - Melhoria de Processo do Software Brasileiro. Versão 1.1.. Disponível em: <<http://www.softex.br/mpsbr/>>. Acesso em 04/12/2009.

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

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

NBR ISO/IEC 25000 (2008). Engenharia de Software – Requisitos e avaliação da qualidade de produtos de software (SquaRe) – Guia do SquaRe. Associação Brasileira de Normas Técnicas.

Pinkster, I., Burg, B. V., Jansen, D. e Veenendaal, E. V. (2006), Successful Test Management: An Integral Approach. Springer. Berlin.

QAI (2006), Guide to the CSTE COMMON BODY OF KNOWLEDGE. Software Certifications. Quality Assurance Institute Worldwide. Version 6.2.

Riosoft (2009), Avaliação de Conformidade de Produtos de Software. Riosoft, agente Softex. Disponível em: <<http://www.riosoft.softex.br/cgi/cgilua.exe/sys/start.htm?sid=53>>. Acesso em: 13/09/2009.

Santos, A. D. dos . (2002), Processo de certificação de qualidade de produto de software na Embrapa : apostila de curso. Embrapa Informática Agropecuária. Disponível em: <<http://www.cnptia.embrapa.br/files/doc27.pdf>>. Acesso em: 23/08/2009.

Serpro (2009a), PSDS - Processo Serpro de Desenvolvimento de Soluções. Release 6.7 de novembro/2009. Disponível em: <<http://psds.portalcorporativo.serpro/>>. Acesso em: 01/12/2009.

Serpro (2009b), Serpro - A Instituição: Quem Somos. Disponível em <<http://www.serpro.gov.br/instituicao/quem>>. Acesso em: 01/12/2009.

SPB (2009), 5CQualiBr. Vetor Qualidade de Produto SPB. Portal do Software Público Brasileiro. Disponível em <<http://www.softwarepublico.gov.br/5cqualibr/xowiki/Qualidade>> Acesso em 18/12/2009.

Woody, S. (2009), Software Longevity Testing. Better Software Magazine. Outubro/2009. Disponível em: <<http://www.stickyminds.com/BetterSoftware/magazine.asp?fn=cifea&ac=422>>. Acesso em: 30/11/2009.