

AVALIAÇÃO DA ADOÇÃO DO CMMI CONSIDERANDO O CUSTO DE QUALIDADE DE SOFTWARE

RESUMO

Produzir software com qualidade e dentro de custos aceitáveis é uma exigência do mercado e premissa na gestão de Tecnologia da Informação nas organizações. Na busca por eficiência as empresas investem cada vez mais em processos de qualidade e modelos como o CMMI. Uma ferramenta importante para direcionar estes investimentos, embora ainda pouco explorada pelas corporações, é o Custo de Qualidade de Software. Este conceito, bastante difundido na indústria de manufatura, ainda possui poucos trabalhos na área da Tecnologia da Informação. Neste contexto foi conduzido este estudo que utilizou a teoria do Custo de Qualidade de Software para analisar os resultados e o impacto da implantação dos níveis 2 e 3 do modelo CMMI em uma empresa de desenvolvimento de software de porte médio. Adotando como metodologia o estudo de caso, os dados de uma amostra de projetos foram coletados e organizados em categorias, servindo de base para a realização de análises comparativas que permitiram verificar ganhos e oportunidades de melhoria. O estudo apresentou como resultado redução satisfatória de custos em projetos de grande porte, o mesmo não ocorrendo nos projetos de menor porte. Como parte dos resultados também ficou evidenciada a dificuldade no registro e manutenção das categorias de custo.

Palavras-chave: Qualidade de Software; Custo de Qualidade De Software; CMMI.

ASSESSMENT OF THE ADOPTION OF CMMI CONSIDERING COST OF SOFTWARE QUALITY

ABSTRACT

Develop software with quality and acceptable costs it's a market requirement and premise in IT management on the organizations. Looking for efficiency, the enterprises invest each time more in quality's process and models such as CMMI. An important tool to drive this investments is COSQ - cost of software quality. This concept is widespread in manufacturing industry, but still have a few papers in IT area. In this context we led a research using the theory of cost of software quality to analyze the results and impacts of implementation of level 2 and 3 from the CMMI model in an organization of software development. The methodology adopted was a case study, the data were collected and organized in categories and a comparative analyses was made. The results shown cost reduction in projects of great size, however it is not happens in small projects. Also, the results pointed out the difficulty in register and maintenance of the category's cost.

Keywords: Software Quality; Cost of Software Quality; CMMI.

Alex Oliveira¹
Maira Petrini²
Daniel Lanes Pereira³

¹ Pontifícia Universidade Católica do Rio Grande do Sul - PUC/RS. Brasil. E-mail: alexfgo@mail.com

² Doutora em Administração de Empresas pela Fundação Getúlio Vargas - FGV. Professora da Pontifícia Universidade Católica do Rio Grande do Sul - PUC/RS. Brasil. E-mail: maira.petrini@puccrs.br

³ Mestre em Administração e Negócios pela Pontifícia Universidade Católica do Rio Grande do Sul - PUC/RS. Brasil. E-mail: daniel.lanes@acad.puccrs.br

1 INTRODUÇÃO

Os investimentos em Tecnologia da Informação (TI), especificamente em software, constituem parte significativa do montante reservado no orçamento de muitas das empresas atuais. O desenvolvimento de aplicações customizadas está se tornando cada vez mais importante para os negócios e tem sido considerado por muitas empresas parte do seu diferencial estratégico (Chrissis, Konrad, & Shrum, 2003). Inúmeros processos corporativos estão amparados por Sistemas de Informação e a confiança na qualidade destes é crucial porque falhas ou erros, normalmente, resultam em prejuízos financeiros ou de imagem. Preservar estes investimentos e garantir a sustentação do negócio que a TI suporta são preocupações constantes para os gestores de TI.

Na indústria de software, por outro lado, a pressão por produtividade e eficiência vem aumentando significativamente, resultado da forte concorrência no setor. O resultado desta pressão, muitas vezes, provoca influência negativa sobre a qualidade dos produtos e acarreta aumento dos custos de produção por conta de falhas introduzidas ao processo em razão de fatores como prazos exíguos e falta de planejamento adequado. Manter produtividade, eficiência e qualidade em equilíbrio tornou-se vital para o sucesso das empresas do ramo.

Para atender estas demandas as empresas têm buscado na melhoria de processos baseada em modelos de qualidade uma forma de aumentar a maturidade de seus processos produtivos. O incremento de maturidade torna os mesmos previsíveis e mensuráveis e as causas mais significativas de baixa qualidade e produtividade são controladas ou eliminadas. Os modelos de qualidade são guias que orientam na adoção de boas práticas e processos comprovadamente eficientes, resultado da experiência de inúmeras corporações que colaboram para compor o corpo de conhecimento destes modelos (Ahern Dennis, Aaron, & Richard, 2008).

Um dos modelos mais conhecidos e adotados na atualidade é o modelo CMMI (*Capability Maturity Model Integration*). Reconhecidamente direcionado a grandes corporações, que executam projetos de média ou longa duração, o modelo reúne melhores práticas distribuídas em áreas-chave de processos “que endereçam aspectos das diversas fases desde a concepção até a entrega e manutenção de produtos de software” (Chrissis et al., 2003). Ao utilizar um modelo, as empresas podem modificar ou criar processos baseadas em práticas que comprovadamente melhoraram a capacidade dos processos e, assim, atingir maturidade. “O aumento da maturidade exige forte suporte gerencial e foco

consistente no longo prazo” (Ahern Dennis et al., 2008).

Embora seja imprescindível, a adoção de um modelo de qualidade implica necessariamente na incorporação de custos ao processo de produção do software. Estes são oriundos das atividades necessárias para atingir um patamar de qualidade satisfatório para o cliente. Identificar e controlar estes custos tornou-se extremamente importante para que o equilíbrio mencionado anteriormente possa ser atingido e é neste contexto que se aplica o modelo de custos de qualidade de software (CoSQ – *Cost of Software Quality*).

Na indústria de manufatura tradicional o modelo de custos de qualidade (CoQ – *Cost of Quality*) é conhecido e utilizado desde a década de 80. No âmbito da TI, entretanto, a experiência com modelos de CoSQ é mais recente, e ele ainda é pouco empregado pelas empresas do setor (Krasner, 1998). Utilizado primariamente para avaliar o retorno de investimento em qualidade, também oferece às empresas parâmetros para determinar investimentos em qualidade. Trata-se de uma ferramenta que permite identificar processos candidatos a melhoria, reduzir custos e estabelecer métricas de comparação entre projetos, entre outros benefícios.

Destá forma, este estudo tem como objetivo responder se a adoção do modelo CMMI de melhoria de processo contribuiu para aumentar a qualidade dos produtos de softwares criados e reduzir os custos com atividades relacionadas com a qualidade, gerando maior rentabilidade nos projetos. A empresa estudada é uma multinacional de porte médio, com cerca de 150 colaboradores, que desenvolve softwares sob medida para o segmento de varejo nacional e internacional. Com atuação forte no mercado Europeu, também possui negócios na África e no Brasil. Além disso, é certificada no nível 3 do CMMI e tem o investimento em qualidade como um diferencial estratégico desde sua fundação, elementos que corroboram sua escolha. A empresa está estruturada em áreas denominadas centros de competência que são focadas na solução para necessidades específicas do mercado em que atua. O tamanho dos projetos de seu portfólio varia desde os muito pequenos até os muito grandes e todos são executados sob o mesmo processo com pequenas personalizações de artefatos de acordo com a área.

Neste sentido, o objetivo desta pesquisa foi avaliar os resultados obtidos pela adoção do modelo de qualidade CMMI utilizando o modelo de CoSQ como ferramenta para tal avaliação. Inicialmente foram mapeadas e alinhadas as categorias de custo dos projetos da empresa com as categorias do modelo de CoSQ. A partir destas categorias de custo definidas foram avaliadas as variações de custos em dois períodos do tempo: (1) após a obtenção da

certificação nível 2 do CMMI e (2) após a obtenção da certificação nível 3 do CMMI.

O artigo apresenta na seção 2 a base teórica da pesquisa desenvolvendo temas sobre qualidade de software, melhorias de processos através do CMMI e o modelo do custo de qualidade de software. Na seção 3 é apresentada a metodologia e o delineamento da pesquisa e na seção 4 apresentam-se os principais resultados. A seção 5 finaliza este artigo trazendo algumas questões à discussão e considerações finais.

2 REFERENCIAL TEÓRICO

Os tópicos apresentados nesta seção abordam referências teóricas que sustentam a importância da qualidade aplicada ao desenvolvimento de software. Para suportar a pesquisa conduzida neste trabalho, foram revisitados o modelo de maturidade de processos CMMI, foco de investigação desta pesquisa, e o modelo de custos da qualidade de software, ferramenta na qual se baseou este estudo na busca da identificação de resultados mensurados em termos de qualidade.

2.1 O Modelo CMMI (Capability Maturity Model Integration)

Nas últimas décadas, o desenvolvimento de modelos de qualidade vem tentando estruturar processos de forma a buscar a garantia de qualidade, seja pela aplicação de boas práticas de mercado ou pela utilização de modelos já empregados com sucesso em outras áreas de conhecimento. O CMMI (*Capability Maturity Model Integration*) é um desses modelos e aborda a qualidade de maneira global ao longo de todos os processos produtivos.

Chrissis (2003) define o CMMI como as melhores práticas que endereçam o desenvolvimento e manutenção de produtos e serviços cobrindo o

ciclo de vida do produto da concepção até a entrega e manutenção. Afirma ainda que o CMMI oferece uma solução abrangente para o desenvolvimento e manutenção de produtos e serviços por ser “composto de corpos de conhecimento essenciais no desenvolvimento de produtos, que eram tratados em separado no passado como a engenharia de software, a engenharia de sistemas e aquisições”.

Nas bases do CMM (*Capability Maturity Model*), antecessor do atual modelo CMMI, estão os conceitos de gerenciamento total da qualidade (TQM – *Total Quality Management*) dos pioneiros Deming, Juran e Crosby. Estes conceitos permanecem na versão mais recente do modelo CMMI e em suas versões como o CMMI-DEV (Kasse, 2008), voltada para o desenvolvimento de software.

Kasse (2008) afirma que o modelo CMM, sempre indicou que para uma organização atingir um nível de maturidade maior que o inicial é necessário forte foco em qualidade. Descreve ainda que no atual modelo CMMI existe uma área de processo responsável por garantir qualidade que é chamada de Garantia de Qualidade de Processo e Produto. Essa área foi expandida de forma a incluir conceitos de qualidade para todas as disciplinas de engenharia cobrindo a qualidade de produto bem como de processo.

A ideia de que “a qualidade de um sistema ou produto é altamente influenciada pela qualidade do processo usado para desenvolvê-lo ou mantê-lo” é a premissa de gerenciamento de processo adotada pelo SEI (*Software Engineering Institute*), proponente do modelo CMMI (Kasse, 2008). Além disso, o custo da qualidade ainda hoje é uma das forças motrizes para as organizações alcançarem o nível 5 de maturidade do modelo CMMI (Kasse, 2008).

A tabela 1 apresenta o modelo no formato mais conhecido, estruturado em níveis de 1 a 5, e para cada nível as principais características do processo e suas respectivas áreas-chave.

Tabela 1 – Níveis de maturidade do modelo CMMI v 1.1 2002 (Kasse, 2008)

Nível	Características do processo	Áreas do processo
5 Otimizado	O foco do processo é na melhoria contínua quantitativa do processo.	Inovação Organizacional e Implantação - OID Análise Causal e Resolução - CAR
4 Quantitativamente Gerenciado	O processo é mensurado e controlado.	Desempenho de Processo Organizacional - OPP Gerenciamento Quantitativo de Projeto - QPM
3 Definido	O processo é caracterizado pela organização e é pró-ativo.	Desenvolvimento de Requisitos - RD Solução Técnica - TS Integração de Produto - PI

		Verificação - VER Validação - VAL Foco de Processo Organizacional - OPF Definição de Processo Organizacional - OPD Treinamento Organizacional - OT Gerenciamento Integrado de Projeto - IPM Gerenciamento de Riscos - RSKM Análise de Decisão e Resolução - DAR
2 Gerenciado	O processo é caracterizado por projetos e frequentemente reativo.	Gerenciamento de Requisitos - REQM Planejamento de Projeto - PP Acompanhamento e Controle de Projeto - PMC Gerenciamento de Acordo com Fornecedor - SAM Medição e Análise - MA Garantia da Qualidade de Processo e Produto - PPQA Gerência de Configuração - CM
1 Inicial	O processo é imprevisível, não controlado e reativo.	

Entretanto, a adoção de um modelo de qualidade implica necessariamente na incorporação de custos ao processo de produção do software. Estes são oriundos das atividades necessárias para atingir um patamar de qualidade satisfatório para o cliente. Identificar e controlar estes custos tornou-se extremamente importante para que o equilíbrio mencionado anteriormente possa ser atingido.

2.2 Custos da Qualidade de Software (*Cost Of Software Quality – Cosq*)

Ainda hoje existe certa confusão a respeito do valor da qualidade dentro e fora do contexto de desenvolvimento de software. De um lado existem aqueles que acreditam que é econômico maximizar a qualidade, pois na medida em que os custos voluntários de prevenção de defeitos aumentam, os custos involuntários de retrabalho diminuem em proporção maior que os de prevenção. No outro lado, existem aqueles que defendem o sacrifício da qualidade em favor de outros objetivos como a redução de ciclos de desenvolvimento, pois acreditam que o alcance de altos níveis de qualidade torna economicamente inviável um projeto (Slaughter, Harter, & Krishnan, 1998).

O investimento em qualidade de software possui custos imediatos e um retorno líquido esperado como qualquer outro. Este investimento tem valor na medida em que os ganhos gerados podem ser determinados e para essa avaliação métodos quantitativos se fazem necessários (Dobbings, 1999). A dinâmica exata da economia que orienta os processos de melhoria de qualidade não está completamente compreendida no mundo do desenvolvimento de software (Slaughter et al., 1998). “Ainda hoje não existe uma teoria validada da

economia da qualidade de software e que este é claramente um assunto para pesquisa multidisciplinar” (Krasner, 1998, p. 8).

O **custo da qualidade** (*Cost of Quality – CoQ*) é utilizado na indústria de manufatura para demonstrar o valor das iniciativas de qualidade e indicar aquelas que são candidatas a serem incorporadas ao programa de melhoria da qualidade (Krasner, 1998). Ou seja, são os custos que não existiriam se todos os trabalhadores fossem perfeitos em seu trabalho (Slaughter et al., 1998). Fato improvável, dada a natureza do software, conforme destacado por Krasner:

O software é uma das mais importantes e ainda desafiadoras tecnologias, do ponto de vista econômico, na era atual. Por ser um produto puramente intelectual, está entre as tecnologias mais trabalhosas, complexas e suscetíveis a erro da história da humanidade (Krasner, 1998, p. 6).

Portanto, o **custo da qualidade de software** (*Cost of Software Quality – CoSQ*) é um modelo que permite avaliar quanto custa a qualidade ou falta de qualidade em um projeto de software e tem seus princípios básicos oriundos da indústria de manufatura (Krasner, 1998). A ideia por trás desta abordagem é de que os custos das falhas de software podem ser medidos e reduzidos através de uma análise de causa-efeito apropriada (Slaughter et al., 1998).

Segundo Dobbings (1999) a análise dos custos da qualidade tem por objetivo não apenas reduzir o custo ou investimento em qualidade de software, mas assegurar que os gastos são realizados nos tipos corretos de custos e que o benefício máximo é obtido deste investimento. Os tipos e o

volume de investimento necessário são determinados por diversos fatores dos quais podem ser destacados os seguintes:

- exigências impostas pelo contratante ou por normas e padrões no caso de contratos regulados;
- tipo de tecnologia utilizada;
- tipo de produto desenvolvido;
- orçamento do projeto;
- conhecimento do negócio pelo time de desenvolvimento.

Krasner (1998) afirma que existem diversas formas de aplicação do CoSQ e neste momento as técnicas têm sido utilizadas apenas para documentar o retorno de investimento (ROI – *Return of Investment*) das iniciativas de software, pois os executivos querem saber que existe retorno dos investimentos iniciais. Sugere ainda que este tipo de aplicação do CoSQ deve aumentar na medida em que mais programas de melhoria de processo iniciarem. Além disso, destaca que o CoSQ também pode ser utilizado para:

- prover base para definir orçamento dos departamentos de gestão e garantia de qualidade;
- identificar processos específicos candidatos a melhoria de qualidade através de análise causal;
- comparar melhorias de processo propostas e identificar aquelas com melhor custo-benefício;
- prover uma unidade de medida para comparar o sucesso de vários projetos;
- reduzir os custos de qualidade em um projeto específico alterando o processo a priori ou mesmo durante sua execução;
- determinar o potencial impacto em riscos ou custo de escolhas específicas em termos de qualidade para um determinado projeto;
- determinar a potencial exposição legal da empresa em função de defeitos causados aos clientes;
- prover dados sobre custo para demonstrar a relação entre o esforço dos empregados com os resultados na qualidade do produto.

Slaughter (1998), Krasner (1998), Dobbings (1999) e Galin (2004) entre outros, classificaram os custos de qualidade em dois grandes tipos: **custos de conformidade** ou **de controle** e **custos de não-conformidade** ou de **falha de**

controle. Galin (2004) propõe inclusive que estes custos podem ainda ser mensurados em separado nas unidades de desenvolvimento e de gestão ampliando a eficiência do modelo.

Os custos de conformidade “são constituídos pela quantia gasta para construir produtos de qualidade” (Slaughter et al., 1998, p. 68) e são divididos em custos de prevenção e custos de avaliação da qualidade. Os custos de prevenção são aqueles associados à prevenção dos defeitos antes que estes ocorram. Custos de treinamento de pessoal em metodologias de projeto, reuniões de melhoria de qualidade, revisões de projeto de software, coleta de métricas e análise de causas raiz são exemplos de custos de prevenção ligados ao desenvolvimento de software (Krasner, 1998). Os custos de avaliação incluem mensuração, estimativa e auditoria de produtos para garantir conformidade com padrões de qualidade e desempenho. Inspeções de código-fonte, revisões, testes e atividades de auditoria são exemplos desses custos no âmbito do desenvolvimento de software (Slaughter et al., 1998).

Já os **custos de não-conformidade**, conforme Slaughter (1998), são compostos por todas as despesas decorrentes de falhas ocorridas durante a construção do produto e são divididos em custos por falhas internas e custos por falhas externas. As **falhas internas** são aquelas ocorridas antes da entrega do produto ao cliente e, normalmente, incluem custos de retrabalho com programação, re-inspeção e re-testes. **Falhas externas** ocorrem quando o produto já está implantado no cliente. Os exemplos mais comuns ligados ao desenvolvimento são o suporte em campo ao cliente, manutenções e despesas com litígios.

A figura 1 mostra um esquema do modelo clássico de CoSQ e a tabela 2 apresenta uma lista dos custos típicos por categoria identificados nos diversos estudos realizados na área. No entanto, conforme destacou Krasner (1998, p. 10), “uma das armadilhas de um programa de CoSQ são as categorias controversas” e é comum a dificuldade em classificar o que são custos normais de operação e o que são custos da qualidade. Um dos exemplos destacados por ele é o custo para produzir um plano de projeto. Embora um plano seja claramente necessário para conduzir os custos e prazos do projeto este também influencia a qualidade do produto. Para estes casos Krasner (1998) destaca que é preciso ter em mente os seguintes pontos:

- A tendência entre os especialistas de qualidade é de identificar como custos de qualidade apenas aqueles que diretamente se relacionam a prevenção, avaliação e endereçamento de não-conformidades da baixa qualidade;

- Discussões sobre categorias controversas normalmente acabam por sabotar os programas de custo da qualidade;
- Os maiores custos da qualidade são os mais fáceis de serem identificados como revisões, avaliação de qualidade de software, testes e retrabalho. Portanto, normalmente é seguro excluir categorias controversas sem afetar significativamente o TCoSQ (*Total Cost of Software Quality* – Custo Total da Qualidade de Software);
- A consistência é importante para o CoSQ, pois permite a identificação clara de melhorias e de processos candidatos a melhoria

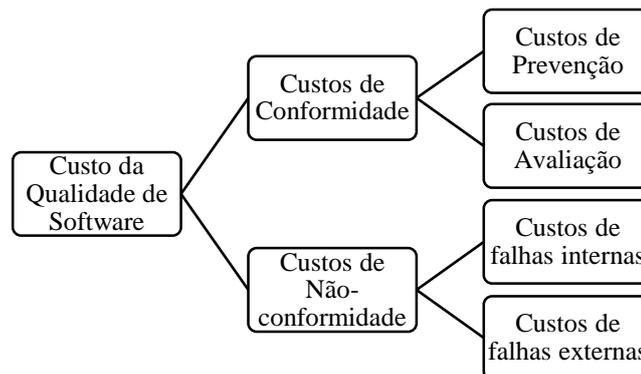


Figura 1 – Modelo Clássico do CoSQ

Krasner (1998) salienta também que conflitos podem surgir da forma como os custos devem ser categorizados e novamente afirma que o mais importante é a consistência. Seu exemplo é o custo de uma revisão ou inspeção que pode ser classificada tanto em prevenção quanto em

avaliação. Embora seja uma questão de interpretação que depende de quando o produto é considerado pronto ou não, ele afirma que as inspeções são realizadas em partes e por isso podem ser incorporadas ao processo produtivo, o que as tornaria prevenção.

Tabela 2 – Custos típicos de acordo com o modelo CoSQ (adaptado de Krasner, 1998 & Galin, 2004)

Classes	Categorias	Subcategorias	Custos típicos
Custos de Conformidade	Custos de Prevenção	Esforços para definir qualidade, objetivos de qualidade, padrões e limites. Análise de custo-benefício de ações de qualidade.	Investimentos no desenvolvimento e atualização de procedimentos de trabalho. Reuniões de melhoria de qualidade. Definição de critérios de liberação para testes de aceitação e demais padrões de qualidade. Condução de revisão de contratos. Preparação e atualização de planos de qualidade.
		Esforços para prevenir baixa qualidade de produto ou melhorar a qualidade do processo.	Treinamentos em metodologias de projeto. Certificação de profissionais. Revisões de projetos de software. Melhorias em processo, coleta de métricas e análise.
	Custos de Avaliação	Esforços para descoberta do nível de não-conformidades.	Testes (unitários, de integração, de aceitação...). Avaliações de qualidade de software (incluindo subcontratados ou agentes externos). Revisões de projeto.

		Inspeções de qualidade periódicas.	Auditorias de qualidade de produto. Medição e avaliação de conformidade e desempenho de produtos com relação a padrões de qualidade. Inspeções de código.
Custos de Não-conformidade	Custos de Falhas Internas	Problemas de qualidade detectados antes da entrega dos produtos.	Gerenciamento de defeitos pré-entrega. Retrabalho em desenvolvimento, testes e revisões.
	Custos de Falhas Externas	Problemas de qualidade detectados após a entrega dos produtos.	Investigação de reclamações e notificação de defeitos. Atualizações emergenciais e correções. Danos de responsabilidade e despesas judiciais.

Por fim, Krasner (1998) destaca ainda que existe uma relação inversa entre os custos para se atingir a qualidade (custos de conformidade) e os custos em função da falta de qualidade (custos de

não-conformidade). Na medida em que o investimento para alcançar qualidade aumenta, o custo ocasionado por falta de qualidade diminui. Esta relação pode ser observada na figura 2.

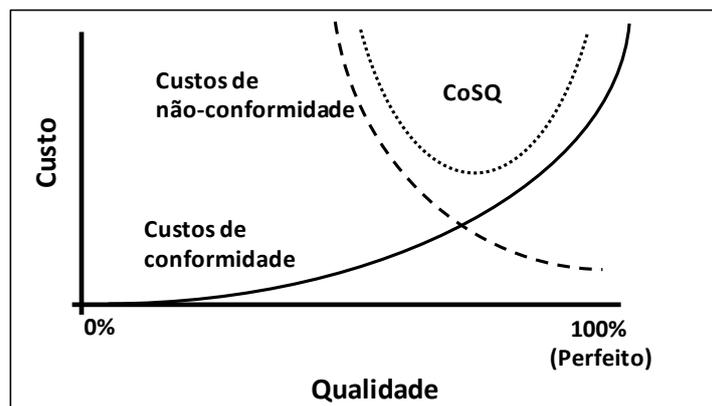


Figura 2 – Modelo econômico do CoSQ (Krasner, 1998, p. 8)

Desta forma, este estudo buscou investigar a relação ilustrada na figura 2, respondendo se a adoção de um modelo de melhoria de processo, ao contribuir para aumentar a qualidade dos produtos de *softwares* criados (o que aumenta os custos de conformidade), reduz os custos de não-conformidade de maneira a gerar maior rentabilidade nos projetos.

3 METODOLOGIA

Esta pesquisa é de natureza quantitativa e exploratória e o método utilizado foi o estudo de caso. Segundo Gil (2002) o estudo de caso pode ser definido como um “estudo profundo e exaustivo de um ou poucos objetos, de maneira que permita seu amplo e detalhado conhecimento”. O estudo de caso também é descrito por Yin (2005) como uma

investigação empírica de “um fenômeno contemporâneo dentro de seu contexto da vida real, especialmente quando os limites entre o fenômeno e o contexto não estão claramente definidos”.

Quanto ao tempo, o estudo pode ser classificado como longitudinal com coleta de dados referentes a dois momentos distintos do tempo. Foram obtidos dados de projetos ocorridos após a adoção do CMMI nível 2 e após a adoção do CMMI nível 3, detalhados a seguir.

3.1 População e Amostra

A população é representada por 160 projetos de desenvolvimento e manutenção executados e finalizados pela empresa entre janeiro de 2005 e agosto de 2010, compreendendo cinco anos de trabalhos. Neste período a empresa foi submetida a duas avaliações para certificação aos

níveis 2 e 3 do modelo CMMI. A primeira ocorrida em janeiro de 2006 para obtenção do nível 2 e a segunda ocorrida em fevereiro de 2009 para obtenção do nível 3. Desta forma, os projetos desenvolvidos entre janeiro de 2006 e janeiro de 2009 foram classificados como CMMI nível 2 e os projetos desenvolvidos após fevereiro de 2009 foram classificados como CMMI nível 3.

Para realizar o estudo foi selecionado um conjunto de 32 projetos que compõem a amostra, correspondendo a 20% dos projetos realizados. Destes 32 projetos, 55% deles são projetos de CMMI nível 2 e 45% de CMMI nível 3. O conjunto foi definido com base na disponibilidade de dados categorizados que tornaram possível a comparação de métricas para identificação dos custos e avaliação dos resultados no âmbito da qualidade. Embora tenham sido criadas versões de processo durante o período estudado, todos os projetos foram executados sobre o mesmo processo base que foi evoluindo ao longo do tempo. Além disso, foram construídos utilizando tecnologia semelhante e por equipes compostas de forma similar.

Segundo Parthasarathy (2007) o tamanho, em um contexto de desenvolvimento de software, é o conjunto completo de funcionalidades de um sistema, com diferentes métodos para análise, sendo LOC (*Lines of Code*) a mais conhecida. O autor considera os meses ou o tempo para o desenvolvimento de um software de um determinado tamanho, como o esforço requerido para aquele projeto. Sendo assim, nesta pesquisa, os projetos foram classificados por esforço (em uma escala que vai de projetos muito pequenos até projetos muito grandes) e por tipo (projetos de desenvolvimento ou de manutenção). A empresa possui uma forma de classificação de esforço própria, com projetos até 400 horas sendo considerados muito pequenos, entre 401 e 3000 horas projetos pequenos, entre 3001 e 5000 projetos médios, entre 5001 e 15000 projetos grandes e acima de 15000 são projetos muito grandes. A composição da amostra manteve proporção similar entre os dois critérios de classificação tanto para projetos de CMMI nível 2 quanto para projetos de CMMI nível 3 (Figuras 5 e 6).

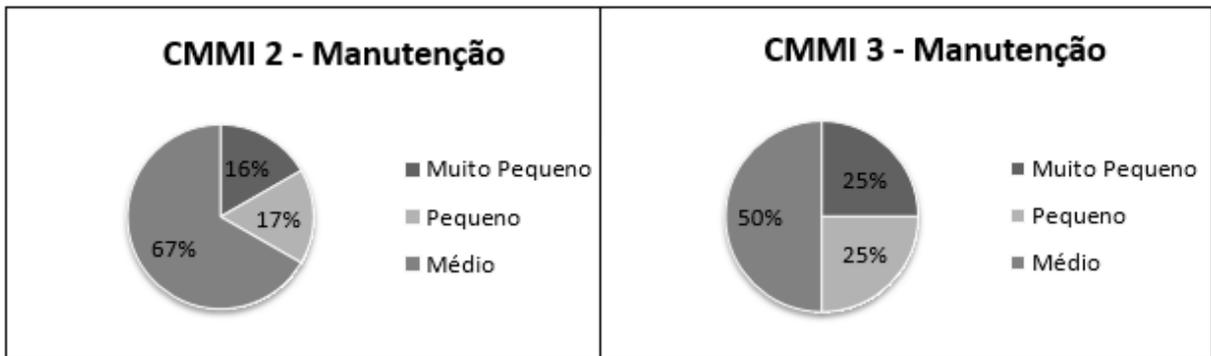


Figura 5 – Perfil por tamanho dos projetos CMMI 2 e CMMI 3 – Projetos de Manutenção

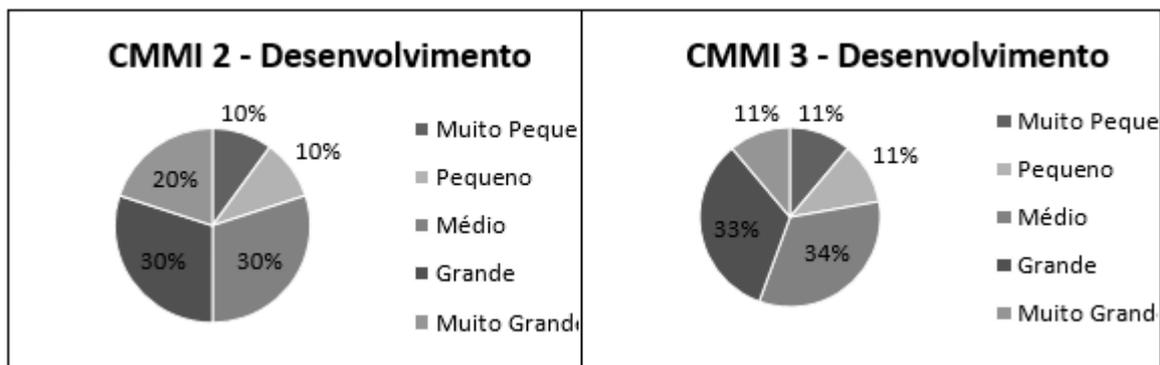


Figura 6 – Perfil por tamanho dos projetos CMMI 2 e CMMI 3 – Projetos de Desenvolvimento

3.2 Coleta de Dados

A coleta de dados utilizou a técnica de pesquisa documental e foi realizada sobre relatórios

de esforço extraídos das bases de dados históricos dos projetos. Embora razoavelmente estruturados, em decorrência das exigências de qualidade internas da empresa, os dados de esforço dos projetos não

foram categorizados ao longo do tempo exatamente de acordo com o modelo de CoSQ apresentado neste estudo. Assim sendo, inicialmente foram mapeadas as categorias originais de esforço e estabelecida a relação destas categorias com as categorias

propostas pelo modelo CoSQ (Tabela 3). Algumas categorias, correspondentes ao esforço direto na atividade de produção de software não foram mapeadas ao modelo e estão sinalizadas como não aplicável (NA).

Tabela 3 – Categorias de Esforço – Empresa x Modelo

Categorias Originais de Esforço da Empresa			Custos de Qualidade	
Categoria	Subcategoria	Descrição	Classes	Categorias
Comercial	Estimativas	Determinação de esforço para construção do projeto.	NA	NA
	Elaboração de Proposta	Criação de proposta comercial e precificação do projeto.	NA	NA
Gestão	Comunicação	Gestão de comunicação	Conformidade	Prevenção
	Gerenciamento	Gestão de pessoas, clientes, fornecedores...	Conformidade	Prevenção
	Planejamento	Planejamento de atividades, manutenção de cronogramas...	Conformidade	Prevenção
	Reuniões	Reuniões de esclarecimento, homologação e aceite de funcionalidades. Reuniões de equipe.	Conformidade	Prevenção
	Monitoramento e Controle	Coleta de métricas, atividades de acompanhamento.	Conformidade	Prevenção
Projeto	Análise de Requisitos	Levantamento e documentação de necessidades funcionais e não-funcionais do sistema. Gestão de requisitos.	NA	NA
	Análise de Impacto	Avaliação de abordagens, alternativas de desenvolvimento e demais aspectos relevantes relacionados com a solução.	NA	NA
	Projeto de Software	Projeto de componentes de software, provas de conceito...	NA	NA
Programação	Codificação	Atividades de programação, construção do código-fonte.	NA	NA
Testes	Especificação	Criação de casos de teste.	Conformidade	Avaliação

	Gestão	Gestão dos testes.	Conformidade	Avaliação
	Execução	Execução dos testes.	Conformidade	Avaliação
Revisões	Revisão de Testes	Revisão dos cenários de teste.	Conformidade	Prevenção
	Revisão Técnica	Atividades de revisão de código, peer review, homologação de padrões.	Conformidade	Prevenção
Retrabalho	Correção de Defeitos	Correção de defeito em código-fonte e documentação de projeto.	Não-conformidade	Falha Interna
	Correção de não-conformidades	Correção de erros ou pendências no produto, em documentação ou em instrumentos de controle do processo de desenvolvimento.	Não-conformidade	Falha Interna
Implantação	Formação de suporte	Treinamento de equipe para manutenção dos sistemas em fase de transição e estabilização.	Conformidade	Prevenção
	Estabilização	Acompanhamento do produto pós entrega durante o período que sucede a entrada em produção da solução.	Não-conformidade	Falha Externa
	Documentação	Documentação de estrutura lógica e física do projeto.	Conformidade	Prevenção
	Treinamento	Treinamento de usuários finais e de suporte.	Conformidade	Prevenção
	Implantação	Atividades de instalação e configuração do sistema.	NA	NA
Formação	Capacitação de Equipe	Treinamento de equipe em tecnologia específica ou metodologia para desenvolvimento de projeto.	Conformidade	Prevenção

Uma vez que as categorias de custos originais foram identificadas e alinhadas às categorias propostas pelo modelo CoSQ, definiu-se qual a abordagem a ser adotada para o cálculo do custo da qualidade de software. Existem algumas abordagens possíveis para demonstrar o CoSQ em uma empresa e o tipo de informação disponível influencia diretamente na escolha da mais adequada. A abordagem adotada nesta pesquisa - a mesma de grande parte dos estudos na área (Galim, 2004) apresenta os custos de qualidade **como percentual**

sobre o custo total de desenvolvimento. Krasner (1998) lista os principais motivos que sustentam esta escolha:

Demonstrar o CoSQ como um percentual do custo total de desenvolvimento é apropriado para o software por diversas razões. Vendas e lucro podem não ter relação direta com o custo de um produto de software, pois o preço normalmente é ditado por forças do mercado. Todo o custo de desenvolvimento de software,

exceto um pequeno percentual, pode ser medido em horas de trabalho, logo os custos podem ser facilmente exibidos tanto em horas ou dólares. O estado da arte no desenvolvimento de software é tal que comparar os custos de qualidade com os custos de desenvolvimento ilustra bem a magnitude do problema atual. (Krasner, 1998, p. 10)

4 ANÁLISE DOS RESULTADOS

Este capítulo apresenta os resultados do estudo através de uma série de análises sobre os dados comparativos obtidos entre os projetos utilizando CMMI nível 2 e CMMI nível 3. As análises exploram a representatividade do custo de qualidade de forma global, por porte e tipo dos projetos. Depois de efetuados os mapeamentos descritos na metodologia de pesquisa, os dados foram agrupados em gráficos que apresentam os custos de qualidade como um percentual em horas do esforço total dos projetos selecionados como amostra. Na primeira parte é apresentada a evolução geral dos custos de qualidade entre os períodos estudados. Em seguida os dados são avaliados separadamente por tipo e porte dos projetos.

4.1 Evolução Global dos Custos de Qualidade

Considerada a totalidade da amostra obtida os dados consolidados **revelaram redução do Custo Total de Qualidade da empresa estudada**. Foi verificada pequena diminuição percentual de 1,8% entre os períodos avaliados no estudo. Embora tenha se observado redução de 9% nos custos de não-conformidade, o resultado foi afetado por um incremento de 7,2% nos custos de conformidade. Ainda que seja difícil comprovar, em virtude das inúmeras variáveis envolvidas em projetos de *software*, é bastante provável que este incremento tenha sido influenciado pela implantação de 11 novas áreas-chave de processo, exigidas pelo CMMI nível 3. Estas áreas não estavam completamente formalizadas ou desenvolvidas no nível anterior e o investimento nestas exigências se traduz diretamente no peso que as atividades de gestão, planejamento e acompanhamento tem sobre os projetos. Algumas destas áreas estão fortemente relacionadas aos custos de prevenção e avaliação como Verificação, Validação e Treinamento Organizacional (Gráfico 5).

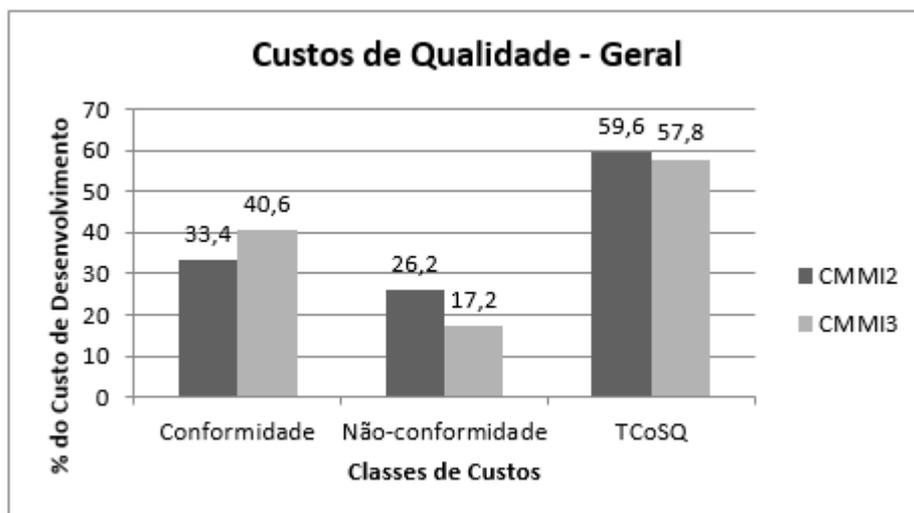


Gráfico 5 – Custos de Qualidade consolidados

A decomposição dos custos nas subcategorias das categorias Conformidade e Não-conformidade contribui para sustentar estes resultados, mostrando que os custos de prevenção e

avaliação sofreram incremento de cerca de 5% e 2,5% respectivamente. Já os custos com falhas internas e externas tiveram redução em torno de 3% e 6% (Gráfico 6).

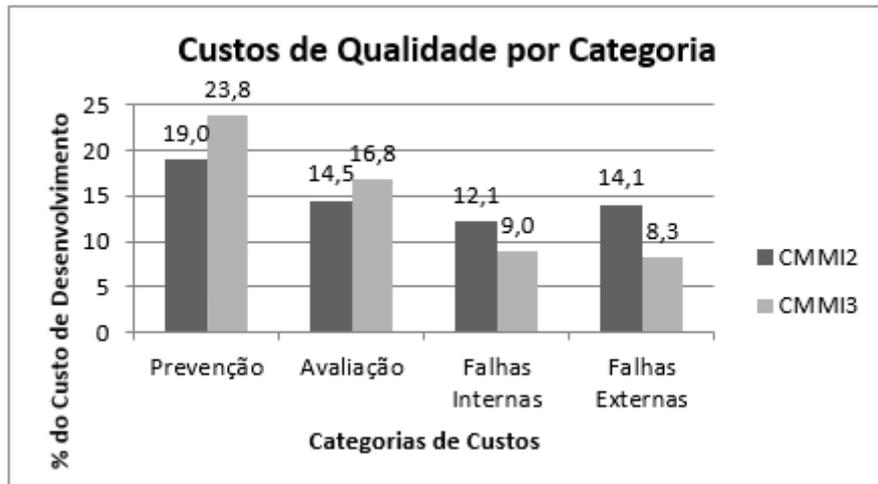


Gráfico 6 – Custos de Qualidade por categoria

Assim, observa-se que embora tenham reduzido em comparação ao período anterior, os custos de não-conformidade somados ainda representam quase 18% do custo de desenvolvimento dos projetos em geral. Patamar dentro do padrão para projetos CMM conforme os estudos de Knox (1993 como citado em Krasner, 1998) que apontam para um valor em torno de 35% no nível 3 do modelo. No entanto, os custos de conformidade ultrapassam os 40% valor acima dos cerca de 20% apresentados pelo mesmo estudo. Infelizmente não foram encontrados estudos ou modelos teóricos mais recentes que permitam aproximar melhor os resultados obtidos, mas isto permite pelo menos concluir que os investimentos ou custos com atividades de qualidade na empresa estudada superam os valores previstos em teoria. Esse investimento pode não ter sido totalmente

revertido na qualidade dos produtos e isso normalmente caracteriza o investimento excessivo ou superinvestimento.

Os tópicos a seguir mostram o comportamento dos custos quando os dados são segmentados por tipo e porte dos projetos.

4.2 O Custo de Qualidade por Tipo de Projeto

Quando observadas as variações dos custos segmentadas por tipo de projeto é possível perceber que no geral **os ganhos foram maiores para os projetos de Manutenção**, onde se pode perceber redução de quase 3% no Custo Total de Qualidade (Gráfico 7). Observa-se também que o custo total manteve-se praticamente no mesmo patamar do custo total geral apresentado no tópico 4.1

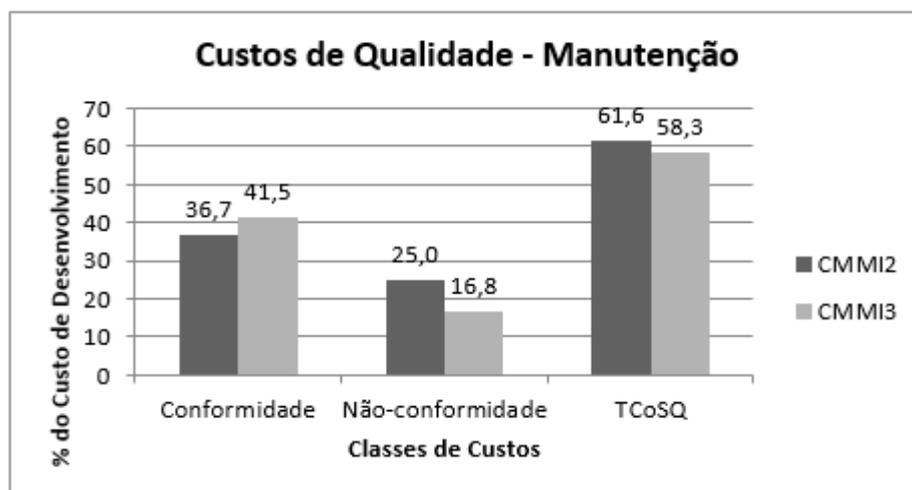


Gráfico 7 – Custos de Qualidade – Projetos de Manutenção

Já nos projetos de Desenvolvimento o mesmo resultado não pode ser observado. O Custo Total manteve-se praticamente estável e a redução percentual foi pouco perceptível conforme ilustra o gráfico 8. A diferença está na influência maior dos

custos de conformidade: nos projetos de Manutenção tiveram aumento de pouco menos de 5% enquanto os projetos de Desenvolvimento tiveram aumento de mais de 9% nos mesmos custos.

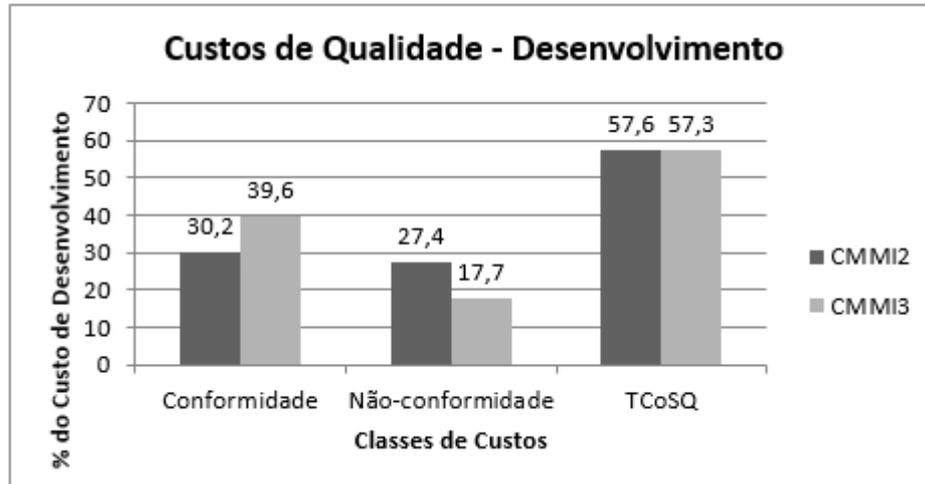


Gráfico 8 – Custos de Qualidade – Projetos de Desenvolvimento

Uma possível explicação para essa diferença está no porte dos projetos, visto que os projetos de manutenção variam de muito pequenos até médios e os projetos de desenvolvimento podem variar de muito pequenos até muito grandes e estes últimos normalmente utilizam um número maior de processos de gestão que influenciam os custos médios de conformidade para cima. Nos projetos menores normalmente são realizadas sessões de *tailoring* para adequar o processo aos projetos. O *tailoring* consiste na remoção ou adequação de processos o que normalmente reduz o nível de exigência nos projetos e consequentemente os custos de conformidade.

4.3 O Custo de Qualidade por Porte e Tipo de Projeto

A análise segmentada por porte mostra que o processo em utilização na empresa estudada

favorece projetos de grande porte. Conforme apresentado nos itens a seguir, foi possível perceber reduções significativas dos custos na medida em que os projetos aumentam de tamanho.

4.3.1 Projetos Muito Pequenos

Nos projetos de porte muito pequeno foram observados os maiores custos de qualidade total do estudo, tanto em projetos de Manutenção quanto em projetos de Desenvolvimento. Os valores encontrados foram superiores a 70%, conforme ilustrado pelos gráficos 9 e 10, bastante acima dos 57,8% da média geral. Nos projetos de desenvolvimento pode ser observado que, mesmo pequeno, houve aumento do custo total de qualidade. Em ambos os casos boa parte deste aumento é oriunda dos custos de conformidade que também estão entre os maiores encontrados no estudo.

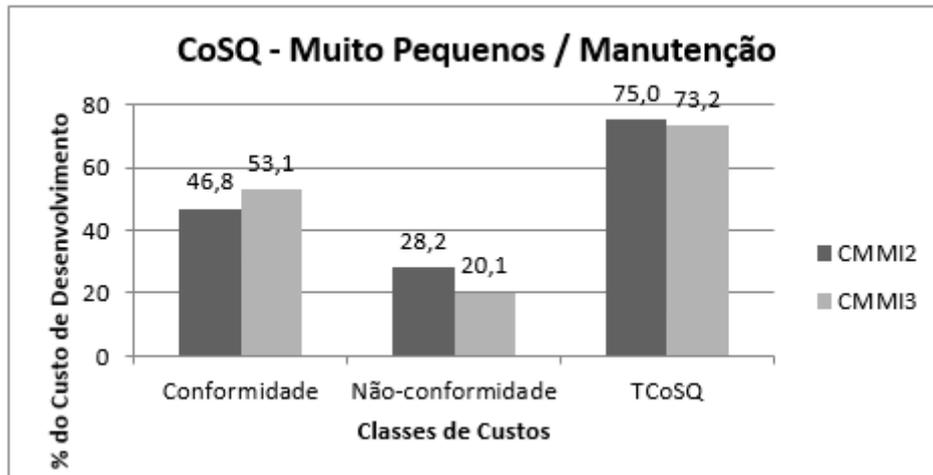


Gráfico 9 – Custos de Qualidade – Projetos de Manutenção Muito Pequenos

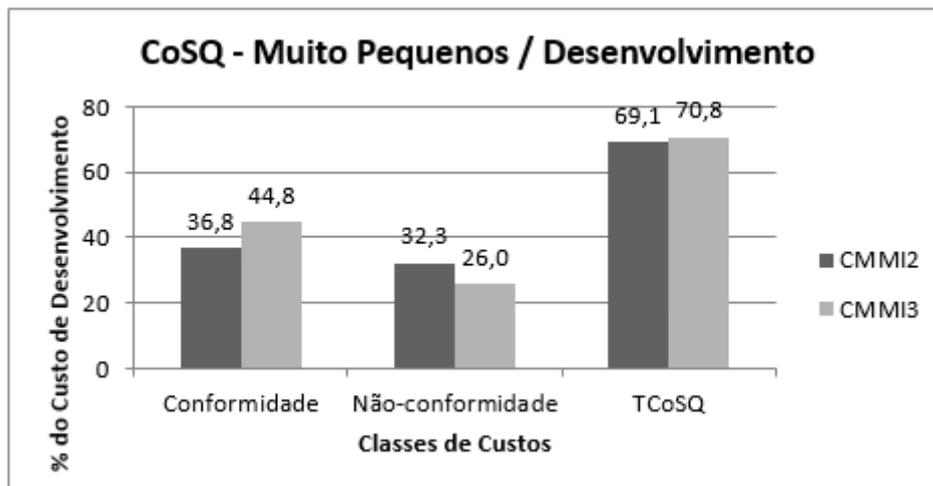


Gráfico 10 – Custos de Qualidade – Projetos de Desenvolvimento Muito Pequenos

Os fatores prováveis para esta diferença com relação à média podem estar relacionados ao *tailoring* mal executado do processo de desenvolvimento ou ainda ao próprio processo que pode estar mal dimensionado para este porte de projetos. Isto pode ser inferido a partir das análises apresentadas mais a frente, pois se observou que no geral, independentemente do seu tipo, os custos tendem a diminuir percentualmente na medida em que o porte dos projetos aumenta, o que pode induzir a ideia de que o processo está mais adequado para projetos de maior porte.

4.3.2 Projetos Pequenos e Médios

Os projetos pequenos apresentaram redução gradual dos custos de não-conformidade conforme ilustram os gráficos 11 e 12, acompanhando a média encontrada nos custos de manutenção por tipo de projeto. Os projetos de manutenção pequenos tiveram aumento percentual de 2,3% no custo total de qualidade, mesmo com o custo de conformidade tendo aumentado menos em comparação com os projetos menores, o que se explica pelo maior aumento nos custos de conformidade.

Já os projetos de desenvolvimento pequenos, o custo total de qualidade teve uma pequena redução percentual de 0,8%, mesmo com aumento de mais de 12% dos custos de conformidade.

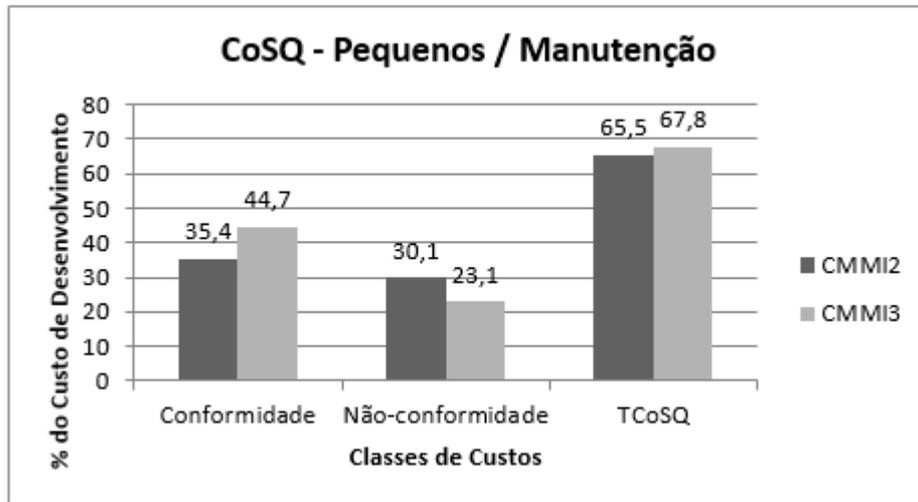


Gráfico 11 – Custos de Qualidade – Projetos de Manutenção Pequenos

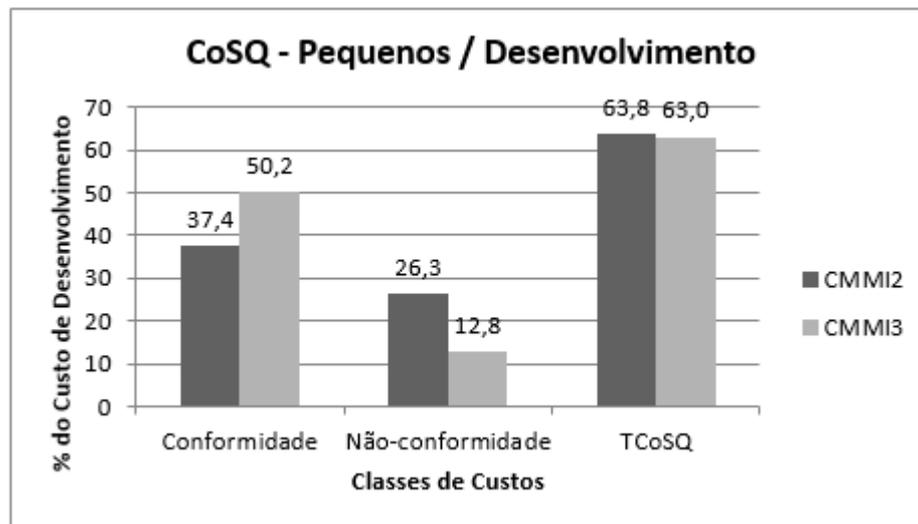


Gráfico 12 – Custos de Qualidade – Projetos de Desenvolvimento Pequenos

Nos projetos de porte médio foi possível perceber que a tendência de redução de todas as classes de custos se manteve. Tanto os projetos de manutenção quanto os de desenvolvimento tiveram

aumento dos custos de conformidade e redução dos custos de não-conformidade que proporcionaram redução no custo total de qualidade conforme apresentam os gráficos 13 e 14.

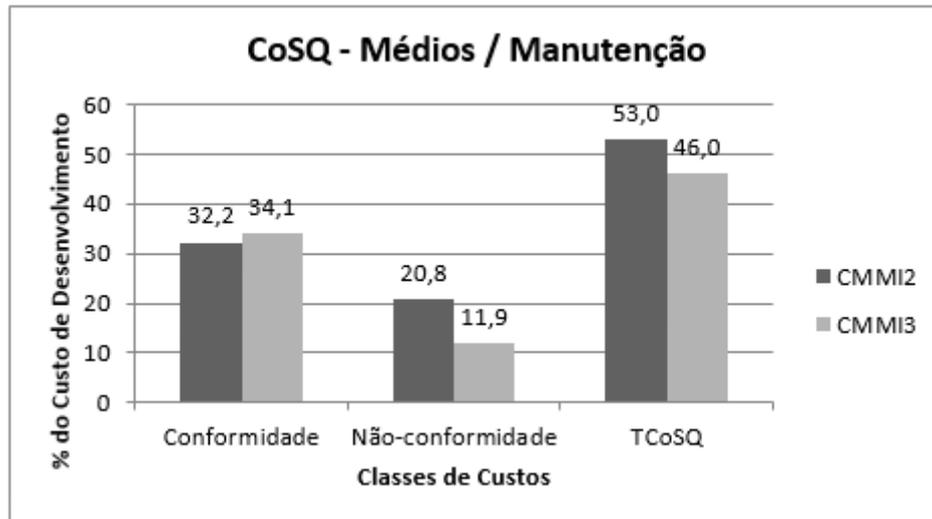


Gráfico 13 – Custos de Qualidade – Projetos de Manutenção Médios

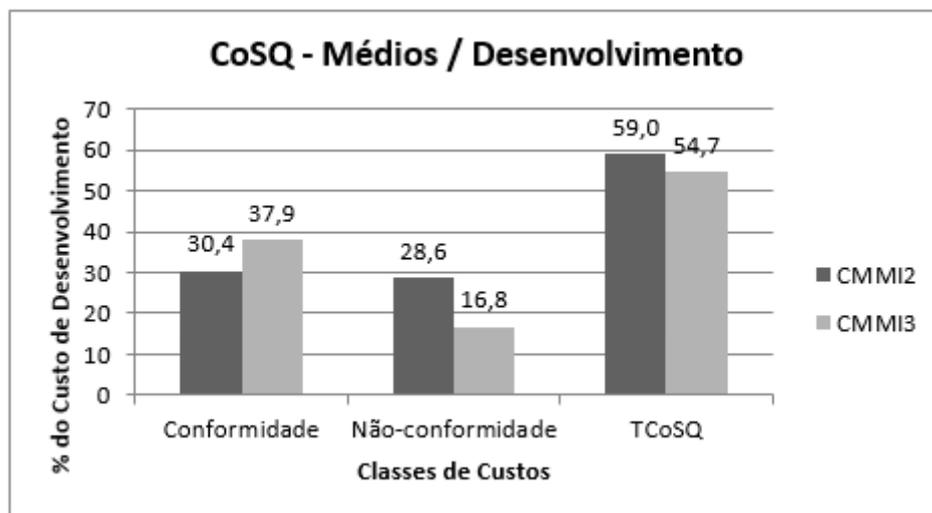


Gráfico 14 – Custos de Qualidade – Projetos de Desenvolvimento Médios

4.3.3 Projetos Grandes e Muito Grandes

A redução dos custos de qualidade observada nos projetos de menor porte manteve-se nos projetos de grande porte e foram observadas reduções nos custos totais de qualidade de 1,3% nos projetos grandes e de 5,9% nos projetos muito grandes (Gráfico 15 e 16). Foi possível observar que o comportamento dos custos de conformidade e de não-conformidade apresentou tendência à estabilidade quando comparados entre si os dois

portos de projetos. Assim, embora esta análise isoladamente não seja suficiente para confirmar uma hipótese, poderia se considerar que os custos mínimos de qualidade da empresa, neste momento, estão situados na faixa entre 36% e 38% para os custos de conformidade e por volta de 17% para os custos de não-conformidade. Estes seriam os níveis mínimos de gastos em qualidade com que os projetos da empresa são produzidos hoje e são os valores que poderiam ser utilizados para apurar progressos em futuros projetos de melhoria de processo.

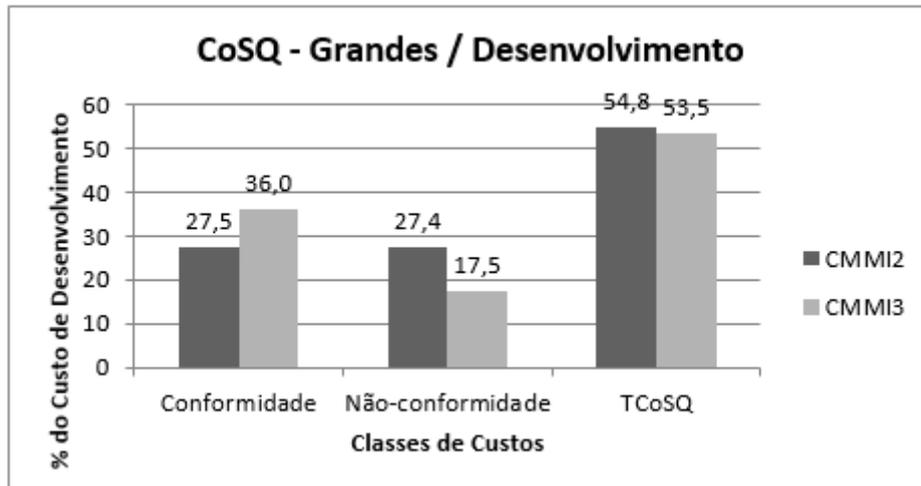


Gráfico 15 – Custos de Qualidade – Projetos de Desenvolvimento Grandes

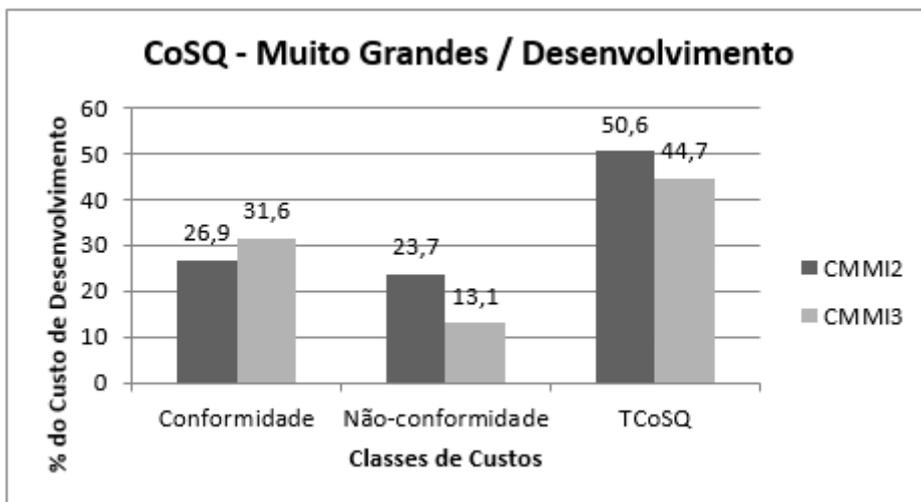


Gráfico 16 – Custos de Qualidade – Projetos de Desenvolvimento Muito Grandes

5 CONSIDERAÇÕES FINAIS

Como na maioria dos trabalhos existentes na área de qualificação de custos, esta pesquisa evidenciou a dificuldade existente na organização em registrar, manter e analisar com regularidade os dados oriundos dos projetos. Durante o processo de levantamento de dados, foi possível observar que a qualificação e a categorização dos custos não são tarefas simples e um processo de acompanhamento eficiente requer disciplina e sistemática para que os resultados apurados realmente representem uma visão da realidade e possam ser utilizados para direcionar os investimentos em qualidade. Ainda durante o período de apuração dos dados para o estudo, foi possível perceber que existem muitos problemas na correta identificação e registro das atividades que são fundamentais para tradução do

esforço e custos dos projetos. Muitos projetos possuíam dados incompletos e grandes distorções no registro de informações.

Com relação ao CMMI, foi possível perceber que ele atende a projetos de maior porte, sendo que empresas com projetos de menor porte poderiam analisar a adoção de outras abordagens de modelos de qualidade e desenvolvimento de software. Comprovadamente orientado para projetos grandes e complexos, o CMMI onera bastante os projetos pequenos e de curta duração. Algumas iniciativas no formato de projetos piloto já experimentam abordagens alternativas ao processo formal da empresa, mas ainda não existe uma versão final contemplando o resultado destas experiências. Os resultados sugerem que nos projetos de menor porte os custos ainda se mantêm muito acima do que

seria razoável e em alguns casos chegam a quase 80% do total do projeto.

No levantamento observou-se que os custos de qualidade variam muito de acordo com o porte dos projetos. Hoje, poderia se afirmar que o processo está ajustado para os projetos de grande porte, vista a melhor relação entre os custos destes projetos nas comparações realizadas. Assim mesmo, foi possível perceber resultados positivos que demonstram melhoria dos custos de não-conformidade de forma geral. Outro aspecto a ser destacado é a boa institucionalização do processo na empresa, que permitiu a obtenção da amostra utilizada no estudo.

Trabalhos futuros poderiam explorar a segmentação mais detalhada dos dados categorizados bem como ampliar o número de métricas adotadas para suportar a tomada de decisão. Informações como as estatísticas de defeitos e outras métricas de desenvolvimento poderiam ser combinadas a estas análises permitindo a empresa compreender melhor os resultados de seus esforços diretamente com base em seus produtos. Importante ressaltar que este trabalho foi resultado de um único estudo de caso e que trabalhos futuros poderiam trabalhar a mesma temática em empresas de menor e maior porte, analisando desta forma a sua relação com o porte de projeto. Apesar de buscar-se um entendimento ao invés de uma generalização nesta pesquisa, um único estudo de caso pode ser considerado uma limitação.

Outro ponto importante a ser destacado, com relação aos dados sobre custos de qualidade, foi a ausência de ações estruturadas para utilização desta informação. Não foi possível identificar qualquer iniciativa semelhante à proposta neste estudo dentro da empresa. Desta forma, uma das principais contribuições deste estudo é auxiliar no direcionamento nos investimentos realizados em qualidade pela empresa. Iniciativas de acompanhamento são fundamentais para que as empresas possam calibrar o mais rápido possível seus esforços em atividades de prevenção e avaliação na medida em que a qualidade aumenta. Isto contribui para reduzir os custos de qualidade e evitar o investimento excessivo em áreas já consolidadas do processo de desenvolvimento.

Assim, o gasto em qualidade de software deve ser considerado sempre uma forma de investimento. E como qualquer investimento, precisa ser acompanhado para obtenção de melhores resultados. Não existem limites para investir na melhoria de processos de software, por isso é importante observar que todo o custo associado à criação de um produto com qualidade acaba sendo

transferido ao custo pago pelos clientes. No universo competitivo atual a habilidade e a velocidade com que as empresas ajustam seus esforços certamente determinam maiores ou menores custos produtivos e isso define o sucesso das iniciativas adotadas neste sentido.

REFERÊNCIAS

- Ahern Dennis, Aaron, C., & Richard, T. (2008). *CMMI® Distilled: A Practical Introduction to Integrated Process Improvement, Third Edition*. Australian family physician (Vol. 39).
- Chrissis, M. B., Konrad, M., & Shrum, S. (2003). *CMMI: Guidelines for Process Integration and Product Improvement*. Engineering.
- Dobbings, J. H. (1999). The Cost of Software Quality. In Prentice Hall (Ed.), *Handbook of Software Quality Assurance* (pp. 195–216). New Jersey, NY: Artech House Publishers.
- Galín, D. (2004). Toward an Inclusive Model for the Costs of Software SOFTWARE DEVELOPMENT. *Software Quality Professional*, 1(2).
- Gil, A. (2002). *Como elaborar projetos de pesquisa*. São Paulo.
- Kasse, T. (2008). *Practical insight into CMMI*. Artech House computing library.
- Krasner, H. (1998). Using the cost of quality approach for software. *The Journal of Defense Software Engineering*, 11(11), 6–11.
- Parthasarathy, M. A. (2007). *Practical Software Estimation: Function Point Methods for Insourced and Outsourced Projects*. Addison-Wesley Professional.
- Slaughter, S. A., Harter, D. E., & Krishnan, M. S. (1998). Evaluating the cost of software quality. *Communications of the ACM*. doi:10.1145/280324.280335
- Yin, R. K. (2005). *Estudo de Caso Planejamento e Métodos*. Estudo de Caso Planejamento e Métodos (Vol. 15).