

Universidade Estadual de Maringá
Centro de Tecnologia
Departamento de Engenharia de Produção

**Estudo sobre a prática dos custos da Qualidade e da Não
Qualidade no Desenvolvimento de Software na Região
Metropolitana de Maringá-PR**

Lara Raissa Bortolon

TCC-EP-48-2015

Maringá - Paraná
Brasil

Universidade Estadual de Maringá
Centro de Tecnologia
Departamento de Engenharia de Produção

Estudo do Custo da Qualidade e da Não-Qualidade no
Desenvolvimento de Software na região Metropolitana de
Maringá - PR

Lara Raissa Bortolon

Trabalho de Conclusão de Curso apresentado ao Curso de
Engenharia de Produção, do Centro de Tecnologia, da
Universidade Estadual de Maringá.

Orientadora *Prof.a. Msc. Daiane Maria De Genaro Chirolí*

**Maringá – Paraná
2015**

DEDICATÓRIA

Dedico este trabalho aos meus pais Edna e Valdeci, pelo exemplo de integridade e pela constante dedicação à família, aos meus irmãos Karen e Lucca que apesar da distância sempre presente, nunca deixaram de estar ao meu lado.

AGRADECIMENTOS

Em primeiro lugar gostaria de agradecer as pessoas mais importantes da minha vida, os meus pais Valdeci e Edna, que se sobrepuseram a qualquer percalço para facilitar meu caminho para mais esta vitória.

Aos meus irmãos Karen e Lucca que mesmo possuindo nossas diferenças se fazem muito importantes na minha vida.

A todos os colegas que por ventura passaram por esse caminho e à minha grande parceira de trabalhos Gabriela, que se tornou essencial aos anos que seguiram deixando-os mais leves e divertidos.

Finalmente gostaria de agradecer a todos os professores que fizeram parte desta jornada, pois contribuíram imensamente para a minha formação, em especial à minha orientadora Daiane, por acreditar que eu fosse capaz de executar tal trabalho e que durante todos esses anos se mostrou como uma mãe, sempre com muita paciência e amor ao seu trabalho.

RESUMO

A qualidade é um tema de crescente interesse para todas as empresas que desejam acompanhar o mercado e se preocupam com a aceitação de seu serviço e produto no mercado consumidor. Os custos da qualidade e da não qualidade devem ser observados em todos os processos das empresas para que não se torne oneroso e traga prejuízos. Este trabalho tem como objetivo verificar essa questão do custo da qualidade e não qualidade em empresas de desenvolvimento de softwares. Durante o estudo foram analisadas quarenta e seis empresas do setor de desenvolvimento de software da região metropolitana de Maringá –PR, buscando compreender seus principais interesses em desenvolvimento de software manutenção dos mesmo, como também os métodos ágeis mais utilizados. Além disso procurou identificar como são medidos os custos da qualidade e da não-qualidade e a possível utilização de indicadores para isso.

Palavras-chave: Custo; Qualidade; Não Qualidade; Software; Métodos Ágeis.

SUMÁRIO

DEDICATÓRIA.....	iii
AGRADECIMENTOS	iv
RESUMO	v
SUMÁRIO.....	vi
LISTA DE FIGURAS	viii
LISTA DE TABELAS	ix
LISTA DE QUADROS	x
LISTA DE ABREVIATURAS E SIGLAS	xi
1 INTRODUÇÃO.....	12
1.1 JUSTIFICATIVA	13
1.2 DEFINIÇÃO E DELIMITAÇÃO DO PROBLEMA	13
1.3 OBJETIVOS	14
1.3.1 OBJETIVO GERAL.....	14
1.3.2 OBJETIVOS ESPECÍFICOS	14
1.4 ESTRUTURA DO TRABALHO	14
2 REVISÃO DE LITERATURA	16
2.1 CUSTOS DA QUALIDADE.....	16
2.2 FERRAMENTAS DA QUALIDADE.....	19
2.2.1 BUSINESS PROCESS MODELING NOTATION (BPMN).....	19
2.2.2 DIAGRAMA DE CAUSA E EFEITO	21
2.2.3 GRÁFICO DE PARETO.....	22
2.2.4 FOLHA DE VERIFICAÇÃO.....	23
2.3 QUALIDADE E DESENVOLVIMENTO DE SOFTWARE	24
2.4 MODELOS DE QUALIDADE E MATURIDADE.....	27
2.4.1 SCRUM	27
2.4.2 CMMI (<i>Capability Maturity Model Integration</i>)	28
2.4.3 MODELO CASCATA	29
2.4.4 MURAL DE GANTT/GRAFICO DE GANTT	30
2.4.5 PSP (<i>Personal Software Process</i>)	31
2.4.6 TSP (<i>Team Software Process</i>).....	31
2.4.7 PMMM (<i>Project Management Maturity Model</i>).....	32
2.4.8 OPM3 (<i>Organizational Project Management Maturity Model</i>).....	32
2.5 CONCLUSÃO DO CAPÍTULO	33
3 METODOLOGIA.....	34
4 DESENVOLVIMENTO.....	36

4.1	A CIDADE DE MARINGÁ	36
4.2	DAS EMPRESAS PESQUISADAS.....	36
4.3	PROPOSTA DE USO DE FERRAMENTAS DA QUALIDADE EM AUXÍLIO AOS MÉTODOS ÁGEIS	46
5	CONSIDERAÇÕES FINAIS	48
5.1	CONTRIBUIÇÕES	48
5.2	LIMITAÇÕES	49
5.3	TRABALHOS FUTUROS	49
6	REFERÊNCIAS	51
	APENDICE A	54

LISTA DE FIGURAS

Figura 1: Custos relacionados à qualidade.	18
Figura 2: Reprodução: Diagrama de Causa e Efeito, causas principais	21
Figura 3: Reprodução: Diagrama de Causa e Efeito, causas secundárias	22
Figura 4: Gráfico de Pareto	23
Figura 5: Reprodução: Método Scrum	27
Figura 6: Cinco níveis do CMMI	29
Figura 7: Modelo cascata.....	30
Figura 8: Gráfico de Gantt.....	31
Figura 9: Fluxo de atividades para a realização da pesquisa.	35
Figura 10: Porte das empresas	37
Figura 11: Foco de atuação das empresas questionadas.....	38
Figura 12: Quantidade de empresas que possuem equipes destinadas à novas criações.....	39
Figura 13: Empresas que focam manutenção e atualização de software e possuem equipe desenvolvedora.	39
Figura 14: Empresas que focam criação de novos softwares e possuem equipe desenvolvedora.	40
Figura 15: Percentagem da empresa que forma as equipes desenvolvedoras.	41
Figura 16: Ferramentas mais utilizadas para o desenvolvimento de software.	42
Figura 17: Periodicidade de reuniões entre ciclos de trabalho.	42
Figura 18: Maneira como as empresas costumam utilizar o método Scrum.	43
Figura 19: Quantas empresas possuem um responsável pela qualidade.	44
Figura 20: Quantas empresas fazem o monitoramento dos custos da qualidade.....	45

LISTA DE TABELAS

Tabela 2: Classificação das empresas de comércio e serviço por número de funcionários37

LISTA DE QUADROS

Quadro 1:Folha de verificação para a classificação de defeitos.....	24
---	----

LISTA DE ABREVIATURAS E SIGLAS

BPMN – *Business Process Modeling Notation*

CMMI – *Software Capability Maturity Model*

CEO - *Chief Executive Officer*

OPM3 – *Organization Project Management Maturity Model*

PMI – *Project Management Institute*

PMMM – *Project Management Maturity Model*

PSP – *Personal Software Process*

SEBRAE - Serviço Brasileiro de Apoio às Micro e Pequenas Empresas

SEI – *Software Engineering Institute*

TSP – *Team Software Process*

1 INTRODUÇÃO

Com a crescente competitividade no mercado mundial, a qualidade vem para agregar valor aos produtos, sempre buscando a melhoria contínua na empresa, evitando desperdícios e retrabalhos. Com a diminuição dos desperdícios, diminui-se também os gastos decorrentes da má utilização dos recursos disponíveis, o que agrega valor à empresa e também à sua qualidade, gerando mais lucro e satisfação ao cliente (CAMPÃO *et al.*, 2012).

A busca contínua por competitividade faz com que a qualidade esteja inserida no contexto estratégico da organização, sendo grande aliada nos resultados dos negócios. Deste modo, não adianta apenas utilizar ferramentas da qualidade, se a prática do controle não for efetiva. Como estratégia de fazer negócios com o objetivo de maximizar a competitividade, a gestão da qualidade se torna intimamente relacionada com o conceito de qualidade total (TQC – *Total Quality Control*) (BALLESTERO-ALVAREZ, 2010).

Para Falconi (2004, p. 15) a Qualidade Total é “o controle exercido por todas as pessoas para a satisfação das necessidades de todas as pessoas”, também é dado por ser “todas aquelas dimensões que afetam a satisfação das necessidades das pessoas e por conseguinte a sobrevivência da empresa”. Falconi (2004, p.14) afirma que estas dimensões são: Qualidade, Custo, Entrega, Moral e Segurança.

A qualidade está diretamente ligada à satisfação do cliente interno ou externo, ou seja, qualquer meio onde a qualidade possa estar presente; O custo é visto não apenas dentro do produto final, mas também nos seus intermeios, ou seja, o seu valor agregado; A entrega se relaciona com os atrasos e erros de entrega, tanto em lugar errado quanto em quantidades erradas; A moral é a média da satisfação de um grupo de pessoas dentro da empresa que é mensurado com absenteísmo e índice de reclamações; Segurança é a dimensão que visa segurança tanto dos empregados quanto dos usuários finais (FALCONI, 2004, p.14).

A classificação de custos da qualidade desenvolvida por Feigenbaum (1994, *apud* SOUZA & COLLAZIOL, 2005) aparece com mais frequência nos estudos sobre este tema, e devido a isso ela será utilizada neste estudo. Ele classifica estes custos como: custos do controle (custos de prevenção e custos de avaliação) e custos de falha no controle (custos de falha interna, custos de falha externa). Esta classificação auxilia na transformação dos processos e da administração já citada, tendo em vista que os somatório de todos os estes custos definidos por Feigenbaum (1994, p.152 *apud* SOUZA & COLLAZIOL, 2005) são associados ao processo de busca pela qualidade nas empresas (MELO, *et al.*, 2012).

A empresa estudada será a de *software* e como os custos da qualidade são inseridos e se relacionam ao seu desenvolvimento. De acordo com Sutherland (2014) o desenvolvimento de *software* primordialmente, era um processo onde se levava muito tempo e dinheiro para ser concluído. A exemplo disso, Sutherland (2014) retrata o caso do FBI com a necessidade de modernizar o sistema vigente, já que muitos fatos não podiam ser cruzados em um sistema à base de documentos impressos. A busca por um sistema mais prático e rápido fez com que milhões de dolares e muitos anos fossem gastos, para que no final ficasse claro o fracasso destes processos de desenvolvimento. À partir de tanto fracasso, Sutherland (2014) propõe um novo método chamado Scrum que ajuda a criar um sistema de *software* para o FBI. As bases do Scrum são um menor número de pessoas para o desenvolvimento, testes regulares e apresentações aos interessados como segurança do real andamento do projeto. Este método diminuiu consideravelmente os gasto de tempo e dinheiro com o projeto, além de dar parâmetros reais do que se é produzido (SUTHERLAND, 2014).

Diante do contexto apresentado, o presente trabalho visa propor a análise dos custos da qualidade e não-qualidade em empresas de desenvolvimento de *software* na região metropolitana de Maringá-PR, tendo em vista que “Uma forma de abordar a qualidade dá-se por meio do teste de *software* (TS), que define atividades de verificação e validação com o objetivo de encontrar problemas” (Herbert, 1999, p. 17 *apud* Tosetto & Bellini, 2008).

1.1 JUSTIFICATIVA

O tema se justifica, pois diante da busca continua pela melhoria da qualidade nas empresas, se faz necessário analisar os custos decorridos da implantação e perduração da qualidade nas organizações de desenvolvimento de *software*, como também os custos ocasionados pela qualidade e não-qualidade.

Assim, como a área de desenvolvimento de software vem se tornando cada dia mais importante para o crescimento das empresas no geral, é importante analisar formas de diminuir custos decorrentes deste processo, a fim de diminuir prazos e aumentar a satisfação do cliente final.

1.2 DEFINIÇÃO E DELIMITAÇÃO DO PROBLEMA

A qualidade para o desenvolvimento de software está ligada ao custo da mudança num determinado momento do processo de desenvolvimento do mesmo. Este processo possui

várias etapas que precisam ser analisadas e revisadas constantemente em busca de possíveis erros, dado que ao avançar em sua produção as mudanças se tornam mais dispendiosas tanto em tempo quanto monetariamente.

1.3 OBJETIVOS

1.3.1 OBJETIVO GERAL

O presente trabalho objetiva avaliar empresas de *software* da região metropolitana de Maringá, com relação a preocupação da mensuração dos custos da qualidade e da não qualidade.

1.3.2 OBJETIVOS ESPECÍFICOS

- Verificar quais são as empresas no setor de desenvolvimento de *software* na cidade de Maringá-PR;
- Analisar se há o controle de custos da qualidade e não-qualidade e como é feito;
- Procurar identificar os métodos para desenvolvimento de *software* utilizados;
- Verificar ferramentas da qualidade que auxiliem no desenvolvimento de software.

1.4 ESTRUTURA DO TRABALHO

O presente capítulo serve como uma introdução ao estudo realizado, mostrando a relação das empresas com a qualidade e como a qualidade está ligada ao cliente. Apresenta a ideia inicial de classificação dos custos da qualidade e introduz o tema principal do trabalho em questão, os custos da qualidade no desenvolvimento de *software*. Expõe também a importância do trabalho e seus objetivos.

No segundo capítulo serão apresentados os conceitos de qualidade e de custo da qualidade, abrangendo os custos evitáveis e os inevitáveis e mostrando alguns conceitos de diferentes autores a respeito do custo da qualidade. Além disso serão apresentadas ferramentas da qualidade para análise e posterior verificação de sua relevância para o estudo em questão. Na continuação do capítulo é inserida a ideia da qualidade no desenvolvimento de *software* e para concluir o capítulo são apresentados modelos de qualidade e maturidade.

O terceiro capítulo é composto pela metodologia do estudo, na qual é exposta a maneira utilizada para a coleta dos dados apresentados no desenvolvimento do trabalho.

O capítulo quatro consiste na apresentação do cenário estudado, as empresas e a cidade.

O capítulo cinco conclui o trabalho apresentando suas contribuições, limitações e fornecendo ideias para desenvolvimento de trabalhos futuros relacionados ao assunto abordado.

2 REVISÃO DE LITERATURA

O presente capítulo traz breves explicações sobre os temas abordados neste trabalho, através de introduções aos custos da qualidade, como também a algumas ferramentas da qualidade sendo elas o BPMN, o Diagrama de Causa e Efeito, Gráfico de Pareto e Folha de Verificação. Apresenta também a qualidade e o desenvolvimento de software e alguns de seus modelos de maturidades, tais como Scrum, CMMI, Modelo Cascata, Mural de Gantt, PSP, TSP, PMMM e OPM3.

2.1 CUSTOS DA QUALIDADE

A qualidade, mesmo que subjetiva, existe desde a pré-história, onde os homens para sobreviver se preocupavam com a qualidade dos alimentos que eram tirados da natureza. Com a evolução passou a cuidar daquilo que plantava e colhia, e das pedras as quais seriam feitas armas e ferramentas. Graças a essa procura da qualidade que o homem vem evoluindo até os dias atuais (FERNANDES & INMETRO, 2011).

O estudo da qualidade é um assunto muito vasto, com ampla literatura e vários trabalhos desenvolvidos em diferentes níveis e enfoques. Nascida no ambiente industrial, ela se alastrou por vários setores econômicos até chegar ao setor de prestação de serviços. A qualidade inicialmente teve sua prática direcionada somente a conferir o trabalho realizado pelos artesãos, porém com o grande crescimento do mercado, a saturação de certos produtos, a grande competitividade e a globalização, a qualidade passou a ter um novo enfoque, voltado aos clientes (MIGUEL, 2001).

Carpinetti (2010) afirma que nos dias atuais, a gestão da qualidade é tida como um fator estratégico não só no meio empresarial, mas no meio acadêmico. Este fator está ligado à melhoria da competitividade e da produtividade.

Para se ter uma visão geral do estado da qualidade, pode-se fazer uma coleta de informações sobre três aspectos: o custo da má qualidade (perdas internas devido à má qualidade); o estado da qualidade da companhia com relação à concorrência e as percepções da qualidade pelo empregado (JURAN & GRYNA, 1991).

A Gestão da Qualidade para Paladini (2008), são “regras (que) envolvem tanto diretrizes gerais que direcionam as ações da organização, quanto as normas de funcionamento de cada uma de suas partes”. Para ele a qualidade possui múltiplos conceitos o que causa muitas vezes equívoco em sua definição e implicações, tendo em vista que se estes forem adotados

erroneamente, o processo gerencial associado à sua produção poderá comprometer ações e resultados com grandes prejuízos no quesito competitividade.

Para Juran (1988, apud MATTOS E TOLEDO, 1998, p. 313) a obtenção de um determinado nível da qualidade é dividido em custos evitáveis e custos inevitáveis.

Os custos inevitáveis são associados à prevenção e avaliação (custos de desenvolvimento de fornecedores, de inspeção, de controle do processo, de revisão do projeto, etc.). Os custos evitáveis são associados aos defeitos e falhas dos produtos identificados dentro da empresa (custos de refugo, de retrabalho, de reclassificação dos produtos, etc.) e dos identificados quando o produto já se encontra no mercado (custos com o processamento de reclamações, com substituições do produto nos prazos de garantia, com processos de indenização, com perdas de clientes, etc.) (JURAN 1988, apud MATTOS E TOLEDO, 1998, p. 313).

Na gestão da qualidade se insere o conceito de custo da qualidade, que para Schiffauerova e Thomson (2003, p. 647) não existe um acordo sobre a exata definição, mas usualmente é entendida como a soma dos custos da conformidade e dos da não conformidade. O custo da conformidade é o preço pago pela prevenção da má qualidade, e os custos da não conformidade são os custos da má qualidade causada pela falha do produto.

Para Dale e Plunkett (1995 *apud* SCHIFFAUEROVA & THOMSON 2014, p. 647) o conceito de custo da qualidade é amplamente entendido como o custo incorrido no projeto, implementação, operação e manutenção no sistema de gestão da qualidade, além dos custos com pesquisas e melhoria contínua, custos da falha do sistema, produto e serviço, e todos os outros custos necessários e atividades sem valor necessários para se obter a qualidade do produto ou serviço.

Um sistema eficaz para integrar as forças de desenvolvimento, manutenção e melhoria da qualidade dos vários grupos de uma organização, permitindo levar a produção e serviços aos níveis mais econômicos da operação e que atendam plenamente à satisfação do consumidor (FEIGENBAUM 1994 *apud* BALLESTERO-ALVAREZ, 2010).

Mattos e Toledo (1998, p. 313) se perguntaram o quão abrangente é a conceituação e mensuração de custo da qualidade, desde qual o valor da qualidade que a empresa oferece e quanto ela custa, quanto custa a má ou falta de qualidade, até o quanto custa a perda de um cliente devido a problemas relacionados a qualidade e onde investir para reduzir esses custos que a falta dela traz a empresa.

Mattos e Toledo (1998, p. 313) definem custos da qualidade como “quaisquer despesas de manufatura ou de serviço que ultrapassem as que teriam havido se o produto tivesse sido feito ou o serviço tivesse sido prestado com perfeição na primeira vez.”.

Feigenbaum (*apud* SCHIFFAUEROVA & THOMSON, 2014, p. 649) propôs o método mais aceitável sobre custo da qualidade atualmente que são: “Os custos da qualidade são

associados ao processo de busca da qualidade pelas empresas, sendo formado pelo somatório dos custos de prevenção, avaliação, falhas internas e falhas externas” (MELO, SOUZA, & ARAÚJO, 2012).

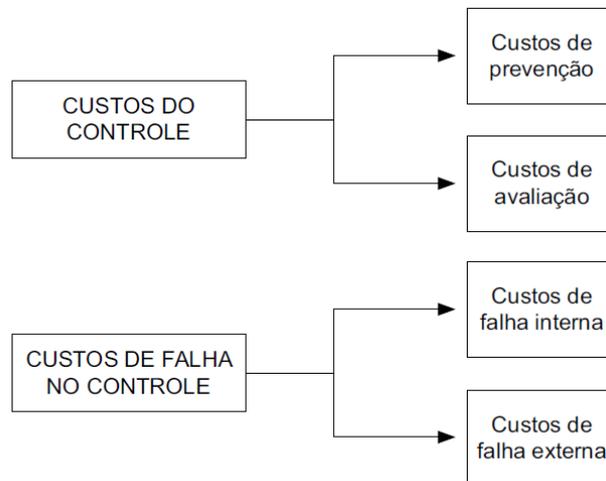


Figura 1: Custos relacionados à qualidade. Fonte: (Feigenbaum 1994, p.152 apud SOUZA & COLLAZIOL, 2005).

Para Feigenbaum (1994, *apud* MOORI & SILVA, 2003 p. 38):

- os custo de prevenção se comprometem aos custos necessários para evitar produtos ou serviços de qualidade inferior ao determinado;
- os custo de avaliação são os relacionados a teste e inspeções que tragam garantias sobre o produto estar dentro de suas especificações contando que não haja trabalho adicional para isso;
- os custos de falha internas são os custo que acontecem antes da entrega do produto aos clientes, ou seja, ligados a falhas ou defeitos que ocorreram;
- os custo de falha externa são os de produtos que já chegaram aos seus clientes e foram devolvidos, ou descontos e garatias dadas para produtos defeituosos que já foram entregues.

Os custos de controle podem ser fiscalizados pela empresa se assim for decidido, os custos de falha no controle não podem ser controlados e são resultados de falhas no processo o que acarreta em custos muito mais elevados (MOORI & SILVA, 2003 p. 39).

Este método genérico é compartilhado por outro autores, e é chama do inglês de modelo P-A-F (*Prevention + Appraisal + Failure*) que no português seriam prevenção, avaliação e falha. Há outros métodos utilizados para o estudo do custo da qualidade apresentados por Schiffauerova e Thomson (2014, p. 650) e são modelos semelhantes entre si, modelo de

Crosby, modelo de custo intangível ou de oportunidade, modelos de custo de processo e modelo ABC.

Resumidamente, Schiffauerova e Thomson (2014, p. 650) apresentam as características que cada um deles:

- modelo P-A-F possui as categorias de custo e atividades sendo prevenção, avaliação e falha;
- modelo de Crosby tem suas categorias sendo conformidade e não-conformidade;
- modelo de custo intangível ou de oportunidade tem suas atividades em prevenção, avaliação, falha e oportunidade; conformidade, não-conformidade e oportunidade; tangíveis e intangíveis; P-A-F incluindo custo de oportunidade;
- modelos de custo de processo abrangem apenas conformidade e não conformidade;
- modelo ABC se direciona para valores adicionados e não-valores adicionados.

2.2 FERRAMENTAS DA QUALIDADE

Independentemente do setor que participam, as empresas atualmente buscam cada vez mais se inserirem no mercado. Em uma teia de organizações competitivas, clientes mais exigentes, legislações restritivas, preocupação com meio ambiente e saúde do trabalhador, as ferramentas da qualidade vêm como orientação e auxílio a essa busca. Com o objetivo de melhorar a qualidade nos processos, as ferramentas são instrumentos destinados ao desenvolvimento, medição, análise e melhoria da qualidade onde estão inseridas (OLIVEIRA, J. A. *et al.* 2011).

A melhoria contínua da qualidade necessita da análise da situação atual em que a empresa se encontra, tendo o o enfoque para o planejamento e implementação de melhorias. Para isso são necessárias as ferramentas da qualidade, que ajudam a compreender os processos e modificá-los da melhor forma, já que estes princípios partem da abordagem científica e são inseridos na tomada de decisão com base em dados e fatos (CARPINETTI, 2010).

Muitas são as ferramentas da qualidade, mas para este trabalho serão estudados o BPMN, O Diagrama de Causa e Efeito, Gráfico de Pareto e Folha de Verificação.

2.2.1 BUSINESS PROCESS MODELING NOTATION (BPMN)

O BPMN tem o objetivo de dar às organizações a capacidade de compreender os seus procedimentos internos de negócio de uma forma gráfica para facilitar a melhor compreensão das transações comerciais na organização, e dá à organização a capacidade de comunicar

esses procedimentos de uma forma padrão (Object Management Group Business Process Model and Notation, 2015).

“ **BPMN** (*Business Process Model and Notation*) é uma notação gráfica que tem por objetivo prover uma gramática de símbolos para mapear, de maneira padrão, todos os processos de negócio de uma organização” (SGANDERLA, 2015).

De acordo com Sgandela (2015), esta ferramenta proporciona ao usuário um conjunto de elementos (atividades, eventos, *gateways*, conectores e *swimlanes*):

- atividades (*Activities*): Representa um trabalho realizado em uma etapa do processo de negócio. Podem ser de dois tipos, Tarefa (*task*) e Sub-processo (*subprocess*);
- tarefa (*Task*): representa uma ação no processo que pode ser executada por um sistema ou uma pessoa, é representada como um retângulo de bordas arredondadas com a descrição dentro da caixa. Ex: avaliar documentos, calcular impostos, elaborar parecer técnico, elaborar proposta comercial, cadastrar operação;
- conector de Sequência de fluxo (*Sequence flow*): É representado por uma linha sólida com uma seta preenchida na direção do destino, em um processo de negócio todos os elementos de fluxo precisam estar conectados conforme a ordem a serem realizados. Liga uma etapa à outra. Esta conexão significa que após a conclusão da atividade, a seguinte pode ser iniciada;
- gateways: controla iterações do fluxo, criando caminhos alternativos ou paralelos, ou unifica fluxos para a continuação em uma mesma sequência de atividades, é representado por um losango;
- eventos (*Events*): “Eventos podem representar a espera de que um fato aconteça para iniciar/prosseguir a execução o processo ou então sinalizar que o processo produzirá a ocorrência de um fato durante ou ao término de sua execução.” (SGANDERLA, 2015). São representados através de um círculo de linha grossa (início e fim) e um círculo de linha dupla (intermediário). Eventos que *aguardam* fatos são chamados de “*catch*”, eventos que *produzem* fatos são chamados de “*throw*” e a causa ou resultado de um evento é chamado de “*trigger*”;
 - Eventos de início (*Start events*): representa onde o processo se inicia. É sempre do tipo *catch* já que aguarda a ocorrência de um evento. Podem ser do tipo none (sem definição), timer (fato temporal, ex: primeira terça-feira do mês), message (chegada de uma comunicação, ex: documento, telefonema), conditional (quando uma determinada condição se torna verdadeira);

- Eventos intermediários (*Intermediate events*): representa ocorrência de eventos no decorrer do processo. Podem ser do tipo *catch* e do tipo *throw*. Seus tipos mais comuns são Tempo ou Prazo (timer, simbolizado por um relógio),
- Eventos de fim (*End events*): representa onde o processo termina. Sempre é do tipo *throw*. Podem ser do tipo *none* (não gera nenhum fato específico), *message* (finalizado com o envio de uma comunicação) sua representação é um envelope preto, *terminate* (processo é finalizado por completo) sua representação é um círculo preto preenchido.

2.2.2 DIAGRAMA DE CAUSA E EFEITO

As ferramentas da qualidade auxiliam na investigação das prováveis causas de um problema, com coleta de dados para auxiliar a entender melhor o processo e assim agir de forma correta. O diagrama de causa e efeito, também chamado de diagrama da espinha de peixe, diagrama da árvore ou diagrama do rio, foi criado em 1953 por Kaoru Ishikawa, e tem a finalidade de mostrar a relação entre características da qualidade e seus fatores (KUME, 1993).

Para Vieira (1999), construir o diagrama de causa e efeito exige alguns passos, tais quais: identificar o problema a ser investigado; escrevê-lo dentro de um retângulo ao final de um eixo à direita da papel; escrever as causa primárias do problema a ser investigado em retângulos dispostos em torno do eixo e ligá-los ao eixo principal por segmentos de reta.

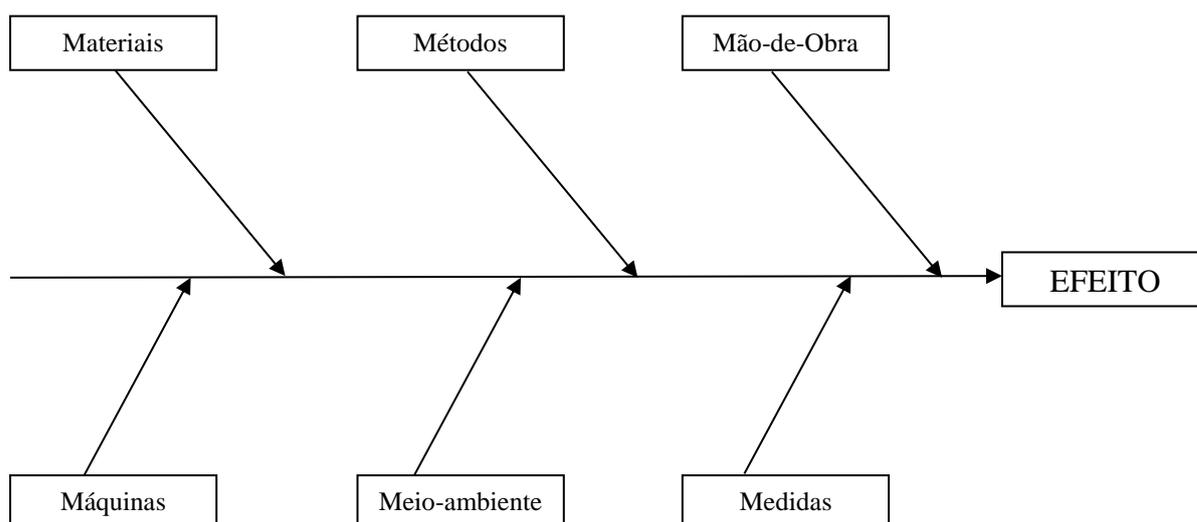


Figura 2: Reprodução: Diagrama de Causa e Efeito, causas principais. Fonte: (VIEIRA, 1999 p. 32).

Tem-se como causas principais: Materiais, Métodos, Mão de obra, Máquinas, Meio-Ambiente, Medidas, como todas as palavras começam com a letra M, este diagrama também pode ser chamado de 6M. Os termos “medidas” e “meio-ambiente” não estavam no diagrama

inicial, porém foram inseridos com o tempo, já que as organizações começaram a se preocupar com esses dois novos termos. As causas secundárias devem ser inseridas dentro de cada classe primária equivalente (VIEIRA, 1999).

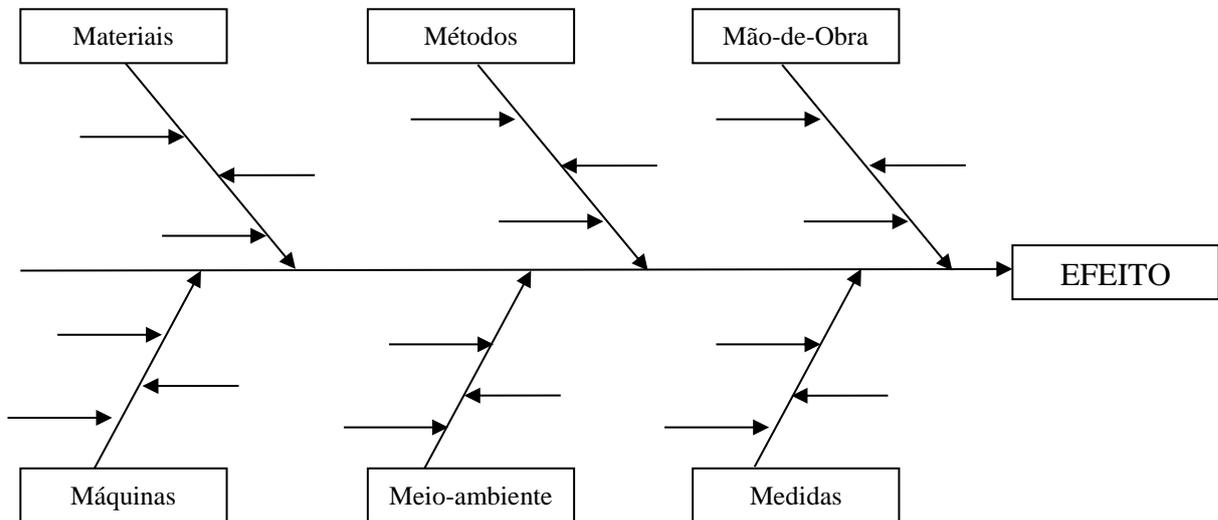


Figura 3: Reprodução: Diagrama de Causa e Efeito, causas secundárias. Fonte: (VIEIRA, 1999 p. 32).

Para fazer o diagrama de causa e efeito é preciso fazer um levantamento das causas, e a melhor forma de se conseguir boas ideias é em discussões dinâmicas através do “*brainstorming*”. Os fatores relevantes são inseridos no diagrama para serem aprofundados e analisados, não há limites para a quantidade de diagramas a serem feitos contanto que sejam mensuráveis (KUME, 1993).

Há cinco sugestões que Kume (1993) apresenta para a construção de diagramas de causa-e-efeito: Identificar os fatores relevantes através do “*brainstorming*”, expressar as características de forma concreta, elaborar quantos diagramas forem necessários, escolher características e fatores mensuráveis e descobrir os fatores que podem ser atacados. Depois de todas as etapas o importante é atacar os problemas de forma objetiva e continuamente modificar o diagrama, verificando os itens que precisam ser acrescentados ou excluídos.

2.2.3 GRÁFICO DE PARETO

“O Gráfico de Pareto consiste em organizar dados por ordem de importância, de modo a determinar as prioridades para a resolução de problemas. É um gráfico usado para classificar causas (por ordem de frequência), que podem ser defeitos, não conformidades, etc.” (MIGUEL, 2001).

Esta ferramenta foi adaptada por Juran tendo base a teoria desenvolvida Vilfredo Pareto, economista e sociólogo italiano. Este princípio parte de que a maior parte das perdas da

qualidade, são decorrentes de alguns poucos problemas vitais, ou seja, grande parte dos problemas são advindos de poucas causas, as causas principais. O princípio de Pareto é um gráfico de barras verticais, que compões o Gráfico de Pareto, que são disposto de maior importância para menor afim de facilitar sua vizualização (CARPINETTI, 2010).

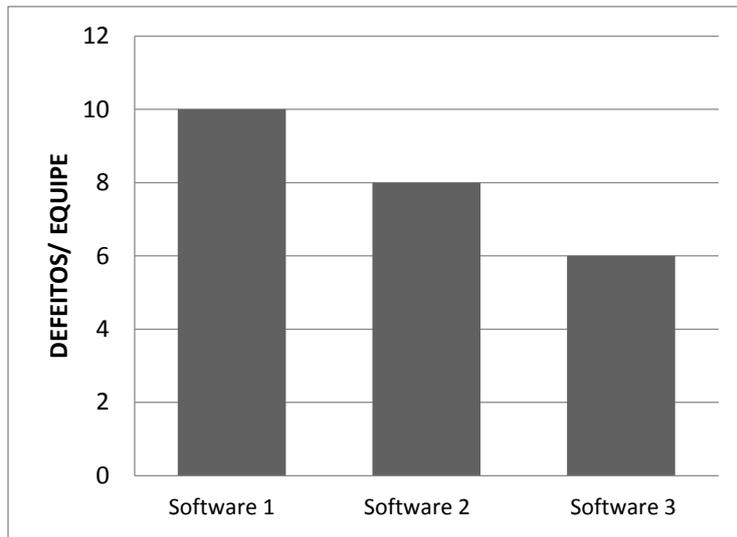


Figura 4: Gráfico de Pareto. Fonte: (CARPINETTI, 2010 p. 83)

As etapas de sua construção, de acordo com Carpinetti (2010) são:

1. selecionar os tipos de causas ou problemas à serem comparados utilizando a discussão em grupo (*brainstorming*);
2. selecionar a unidade comum que os problemas serão comparados (frequência, custos, etc);
3. definir o período de tempo que os dados serão coletados, e isso valerá para todos;
4. coletar os dados;
5. listar as categorias em ordem decrescente da esquerda para a direita;
6. desenhar o gráfico de barras;
7. iniciando ao topo do primeiro retângulo, pode-se adicionar uma linha de frequência cumulativa das caregorias.

2.2.4 FOLHA DE VERIFICAÇÃO

A folha de verificação é utilizada para fazer a coleta de dados mais simplificada e organizada sem a necessidade de posteriores rearranjos. Esta folha é um formulário onde será impresso os tipos de defeitos e estes serão anotados cada vez que ocorrerem (CARPINETTI, 2010).

desenvolvedores e em seguida envolve todo o grupo para enfim envolver toda a organização (TONINI *et al.*, 2008).

Para auxiliar o entendimento de qualidade de *software*, Pang (2002) define três problemas que podem ocorrer com o software, o bug, o erro e o defeito. O bug é “qualquer desvio entre a expectativa do cliente, a especificação, e / ou implementação”; o erro é “qualquer evento observado (incidente) ou desvio detectado entre a especificação e a implementação”; e o defeito “um desvio entre a especificação e a implementação, detectada após a liberação para o cliente (ou a próxima atividade no processo de software)”.

Podemos identificar quatro fatores que afetam a qualidade dos produtos de software: tecnologia de desenvolvimento; qualidade dos processos; qualidade das pessoas; cronograma (recursos e tempo). No que tange a qualidade dos processos estamos nos referindo ao grau de satisfação e eficiência do processo ou um conjunto de processos, utilizados por uma organização ou projeto para: planejar, gerenciar, executar, monitorar, controlar e melhorar as atividades relacionadas à construção de software (SOMMERVILLE 2001 *apud* LAHOZ & SANT'ANNA, 2003, p.2).

A busca por esses quatro fatores ficam claros no método Scrum, desenvolvido por Sutherland (2014) e implementada para a inserção de um novo sistema no FBI. A tecnologia estava avançando, enquanto o sistema do FBI de documentos físicos provava que estava obsoleto com o atentado às torres gêmeas do dia onze de setembro em 2001. Ficou claro na época que este atentado poderia ter sido evitado, já que o FBI possuía informações necessárias para prever o ocorrido, porém com o sistema lento de informações o fato não foi descoberto a tempo. A modernização era necessária e com isso foram contratadas equipes para o desenvolvimento de um software que agregasse e cruzasse todos esses dados. Uma equipe após a outra falhou, e com isso eram dispendidos tempo e dinheiro, frequentemente ultrapassando os previstos.

Após tanto fracasso, Sutherland (2014, p. 12) começou a desenvolver o *software* com o seu método Scrum que procurava entender a forma como as pessoas trabalhavam, e não como diziam que trabalhavam. Era composto de equipes menores, tendo a documentação de requisitos revisada para dar prioridade às que possuíam mais valor ao projeto (80% do valor de qualquer parte do *software* corresponde a 20% de suas funcionalidades), haviam paradas regulares para a verificação do que se tinha feito, e assim aprimorar o trabalho para obter resultados melhores e mais rapidamente, isso proporcionava às equipes um *feedback* quase que imediato do trabalho realizado. No início de cada ciclo (duas semanas), havia uma reunião para a equipe decidir a quantidade de trabalho que iriam executar no próximo ciclo, como poderiam trabalhar melhor, quais obstáculos tiveram para poder removê-los e quais deles estavam diminuindo o ritmo de trabalho. O melhor de tudo para Sutherland (2014, p.

23) era que o Scrum possibilitava demonstrações sobre o trabalho às partes interessadas, gerando um sentimento de confiança.

No início dos tempos de desenvolvimento de software, explica Bartié (2002, p.3), testar o *software* era apenas percorrer o seu código corrigindo problemas conhecidos na fase em que o produto já estava quase pronto. Com o passar das épocas, os testes foram ficando mais sofisticados, sendo feitos periodicamente trazendo aos desenvolvedores a importância da qualidade de *software* e surgindo então ferramentas de teste. Atualmente para a qualidade de *software* ser atingida afirma que existem duas dimensões fundamentais: qualidade do processo (teste de verificação) e qualidade do produto (teste de validação).

Os testes de verificação estão ligados às atividades e avaliação de todos os documentos, gráficos, código-fonte que foram gerados em todas as fases do processo para que fiquem claros os seus objetivos, evitando que dúvidas passem para as fases seguintes, e assim diminuindo possíveis retrabalhos. Os testes de validação devem ser aplicados em componentes isolados, módulos ou no sistema todo e procura verificar se o software está de acordo com os requisitos das etapas iniciais do projeto. Nesta etapa existe o software físico e para os teste são feitas simulações que são comparadas com as especificações iniciais (BARTIÉ, 2002, p.37-38).

“Planejar é útil. Seguir cegamente os planos é burrice” (SUTHERLAND, 2014, p. 28). Esta frase aponta o quão importante é revisar o *software* periodicamente sem seguir cegamente a um rumo previamente planejado, acreditando que este permanecerá o mesmo por todo o projeto. Muito dinheiro e tempo são poupados neste método, agregando à qualidade do software e aos custos dela, já que a necessidade de reparar a não-qualidade é baixa e com reuniões e revisões periódicas, a manutenção da qualidade se mantém de forma visível e contínua.

É difícil delimitar um processo de desenvolvimento de *software* padrão, já que para isso existem vários modelos de processo representando uma tentativa de colocar ordem a uma atividade (RAMOS, 2009).

Para Ramos (2009, p.4) o processo de *software* é uma sequência de atividades que produzem variedades de documentos e resultados inseridos em um programa satisfatório e executável. Como ele mesmo diz, um mesmo requisito para o desenvolvimento de um software, se feito por pessoas diferentes, retorna um software diferente.

2.4 MODELOS DE QUALIDADE E MATURIDADE

2.4.1 SCRUM

Conhecido por ser uma metodologia ágil, o Scrum (figura 5) é comumente utilizado para a gestão e planejamento de projetos de *software*, sendo eles divididos em ciclos tipicamente mensais chamados *Sprints* (conjunto de atividades a ser executada de forma iterativa). O que se deseja implementar em um projeto é mantido em uma lista chamada de *Product Backlog* onde o *Product Owner* (responsável por decidir os recursos e funcionalidades que serão construídos e a ordem que devem ser feitos) prioriza os itens contidos no *Product Backlog*. Durante o *Sprint* a *Scrum Team* (equipe de desenvolvimento) deve selecionar as atividades que será capaz de implementar, transferindo-as para o *Sprint Backlog*. Normalmente no início de cada dia de uma *Sprint* há uma breve reunião chamada de *Daily Scrum* que possui o objetivo de apresentar o que foi feito no dia anterior, identificar os obstáculos e priorizar o trabalho do dia que se inicia. Quando uma *Sprint* chega ao final é feito uma *Sprint Review Meeting* que é uma reunião onde o *Scrum Team* apresenta o que foi alcançado durante a *Sprint*, nela participam todos os interessados sendo eles o *Product Owner*, o *Scrum Team*, o *Scrum Master*, a gerencia, os clientes e engenheiros de outros projetos. Ao final de todas essas etapas é feita uma *Sprint Retrospective* para identificar o que ocorreu como o desejado, o que pode ser melhorado e quais ações serão tomadas em seguida, partindo assim para o próximo *Sprint* (PEREIRA. P. *et al*, 2007).

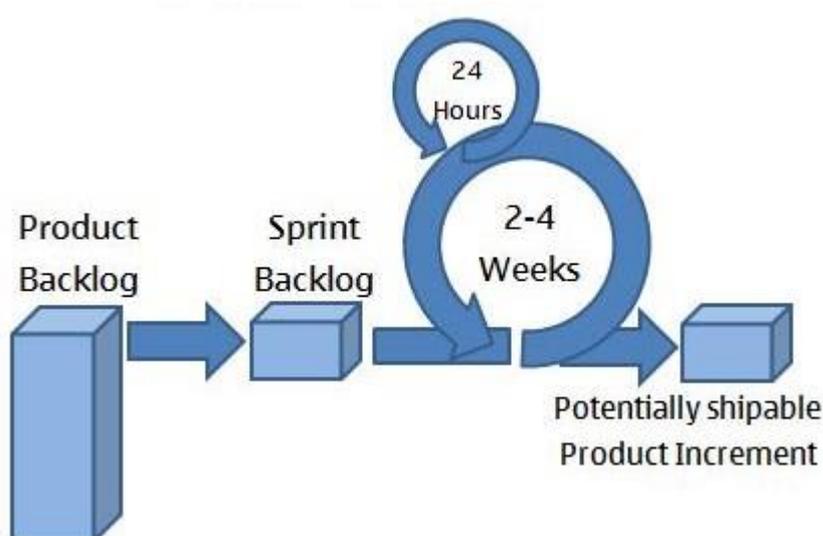


Figura 5: Reprodução: Método Scrum. Fonte: (MACDONALD, 2016).

2.4.2 CMMI (*Capability Maturity Model Integration*)

O CMMI (figura 6) tem a missão de aperfeiçoar a qualidade final dos produtos das organizações, buscando a melhoria da qualidade no processo de desenvolvimento de *software*, é baseado em um modelo evolutivo de maturidade partindo da completa falta de controle e adquirindo novas competências gradativamente, aumentando seu nível de maturidade relacionados aos seus processos críticos (BARTIÉ, 2002). É um método criado pelo SEI (*Software Engineering Institute*) com o intuito de que o Departamento de Defesa dos EUA pudesse acessar a capacidade de desenvolvimento de produtos de software. Primeiro foi criado o SW-CMM (*Software Capability Maturity Model*) que não era efetivo na utilização de processos específicos no desenvolvimento de software e assim foi criado o CMMI (*Capability Maturity Model Integration*) que trouxe como entendimento o fato de que nem em todos os processos se pode obter melhorias, além de nem todas as empresas desenvolvedoras de software executam todos os processos do ciclo de desenvolvimento. Em sua versão estagiada o SW-CMM apresenta cinco níveis de maturidade destinada aos requisitos estruturais nas áreas-chave no desenvolvimento de software que são (HUMPHREY, 1989; PAULK *et al* 1997; CMMI-1 E 2, 2002 apud TONINI *et al.*, 2008): Nível 1: Inicial; Nível 2: Repetitivo; Nível 3: Definido; Nível 4: Gerenciado; Nível 5: Otimizado (BARTIÉ, 2002).

No nível inicial (1) o processo é imprevisível e pouco controlado e o mais difícil de se alcançar, pois trata-se de mudanças radicais no âmbito empresarial, no nível repetitivo (2) as tarefas mestras podem ser repetidas continuamente, tendo novos projetos baseados na experiência adquirida com processos similares anteriormente executados, no nível definido (3) o processo é caracterizado e bem entendido partindo para a padronização, possuindo os processos bem documentados e integrados ao processo de engenharia de software, auxiliando os gerentes e profissionais de software no ganho de produtividade, no nível gerenciado (4) o processo torna-se mensurável, sendo assim medido e controlado aumentando a previsibilidade do desempenho de diferentes processos, no nível otimizado (5) a busca pelo aperfeiçoamento do processo deve ser contínua, identificando pontos fracos e defeitos efetuando ações preventivas nas causas (BARTIÉ, 2002); (SILVA, 2016).

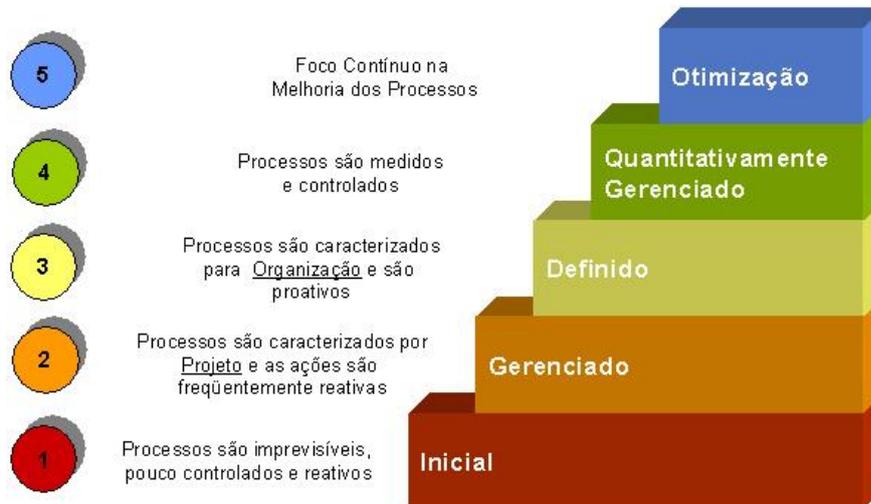


Figura 6: Cinco níveis do CMMI. Fonte: (ISD BRASIL, 2016).

2.4.3 MODELO CASCATA

É o modelo (figura 7) mais antigo no desenvolvimento de *software* e por isso caiu em desuso. É um modelo que busca a abordagem sequencial e sistemática no desenvolvimento de *software*, iniciando o levantamento de dados à partir das necessidades do cliente, avançando para as fases de planejamento, modelagem, construção, emprego e enfim chegando à manutenção contínua do software concluído (PRESSMAN, 2011, p. 59).

A primeira etapa chamada comunicação, é onde se inicia o projeto buscando a necessidade dos clientes. A segunda etapa é o planejamento onde há as estimativas de prazos, execução de cronograma e do acompanhamento. A terceira etapa é a Modelagem onde ocorre a análises do projeto para a execução do mesmo. A quarta etapa é a construção, se iniciando a codificação e os testes necessários. A quinta etapa é o Emprego tendo a entrega do software ao cliente, disponibilizando manutenção e recebendo o feedback (PRESSMAN, 2011, p. 60).

Há também a Cascata Modelo V, sendo uma variação do modelo tradicional porém com algumas melhorias inseridas, sendo elas testes que devem acontecer paralelamente ao modelo e em todas as etapas de forma a garantir a melhor funcionalidade do projeto (PRESSMAN, 2011, p.60).

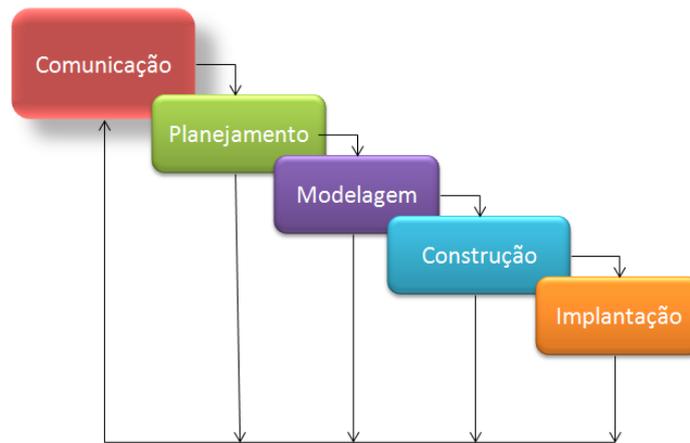


Figura 7: Modelo cascata. Fonte: (PRESSMAN, 2011).

2.4.4 MURAL DE GANTT/GRAFICO DE GANTT

É um modelo visual de gestão de projetos que pode ser adequado para a utilização de cada empresa e permite definir um conjunto de tarefas e organiza-las cronologicamente, dando suas respectivas durações e relações de precedência. É feita à partir de uma planilha relacionando datas e duração dos processos em questão, ajudando a otimizar o tempo e alcançar melhores resultados (TERESO, 2008).

O gráfico de gantt (figura 8) cria uma lista de atividades que devem ser completadas durante o projeto, estimando todas as ações que devem ser executadas. As atividades são inseridas de forma sequencial, e podem ou não acontecer ao mesmo tempo, porém muitas atividades executadas em paralelo podem acabar prejudicando a execução do projeto (BASTIANI, 2016).

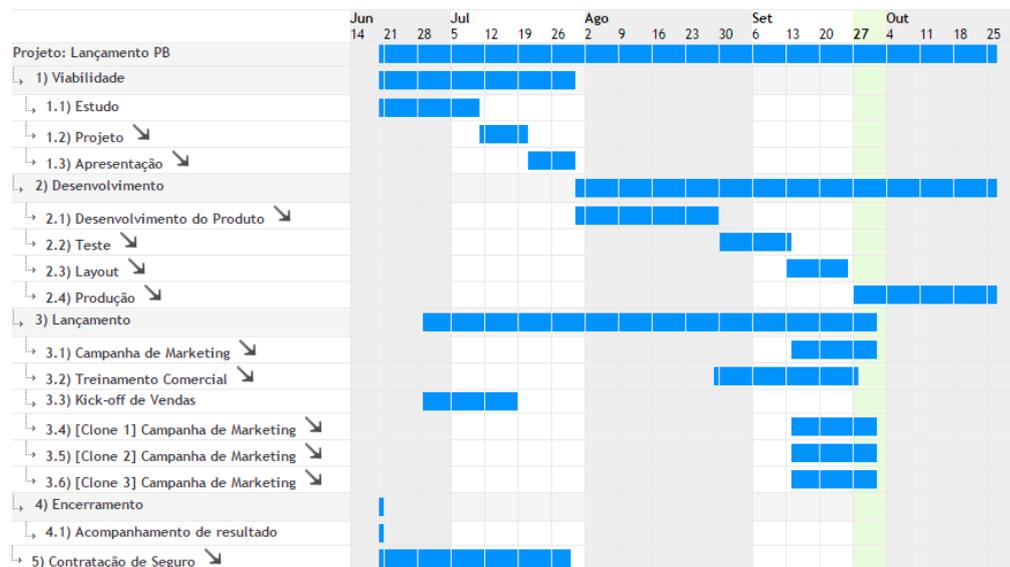


Figura 8: Gráfico de Gantt. Fonte: (REIS, 2016)

2.4.5 PSP (*Personal Software Process*)

O modelo PSP (*Personal Software Process*) foi elaborado pelo SEI (*Software Engineering Institute*) para abrir possibilidades de ser aplicado em pequenas empresas e no indivíduo, ao contrário do CMMI que é destinado à empresa. É direcionado às pessoas e às equipes de desenvolvimento e seu primeiro nível busca tornar os desenvolvedores aptos a elaborarem medições básicas do seu trabalho como tempo gasto e defeitos encontrados. No nível seguinte os desenvolvedores devem amplificar suas habilidades em planejamento pessoal, priorizando o planejamento do tempo ideal para a realização de suas atividades. O terceiro nível busca a melhoria da qualidade pessoal, sendo feitas revisões de trabalhos para assim prever quantos erros podem ser cometidos em cada fase do ciclo de desenvolvimento, levando em conta o histórico do desenvolvedor. O último nível do PSP visa tratar de projetos maiores, subdividindo-o em pequenos projetos que possam ser tratados de forma pessoal (HUMPRHEY, 2000).

2.4.6 TSP (*Team Software Process*)

O modelo TSP é como uma extensão do PSP, porém é focado no gerenciamento de equipes. O PSP foca na melhora das habilidades individuais, o TSP procura a melhora da performance do grupo e o CMM busca a da organização. Os três podem trabalhar em conjunto, sendo complementares uns aos outros. O TSP deve ocorrer em ciclos incrementais de forma a suavizar os riscos a cada ciclo. Assim, antes de iniciar o desenvolvimento, a equipe deve

delimitar estratégicas, analisando suas forças, fraquezas e disponibilidades, para então iniciar o planejamento e execução do desenvolvimento (TONINI *et al.*, 2008).

2.4.7 PMMM (*Project Management Maturity Model*)

É baseado nos cinco níveis do CMMI e nos processos de gerenciamento do PMBOK (*Project Management Maturity Model*). Ele possui cinco níveis de desenvolvimento sendo eles: linguagem comum, processos comuns, metodologia singular, *benchmarking* e melhoria contínua. Inicialmente o PMMM utiliza as nove áreas de conhecimento do PMBOK nos diversos níveis de maturidades, essas nove áreas são: integração, escopo, tempo, qualidade, recursos humanos, comunicações, riscos, aquisições, custos. O primeiro nível é chamado de linguagem comum e nesta fase é avaliado o grau que a organização compreende os conceitos fundamentais do gerenciamento de projetos. O Nível dois, chamado de processos comuns, visa a concentração de esforços na gestão de projetos para desenvolver processos e metodologias que possuam finalidades definidas na organização. O nível três ou a metodologia única, é a etapa onde a organização busca o desenvolvimento de uma metodologia única à partir do profundo conhecimento e controle dos processos. A análise comparativa é o quarto nível, sendo onde a organização insere a coleta de informações e executando o *benchmarking*, sendo ele um método de abordagem de avaliação comparativa que deve ser contínuo na empresa. O último nível é a melhoria contínua que a organização deve buscar, avaliando o que se foi aprendido e implementando mudanças necessárias para o processo de gestão de projetos, fazendo uma avaliação para identificar se a organização conseguiu alcançar a maturidade na gestão de projetos, o que aumenta a probabilidade de sucesso dentro da empresa (TRENTIM, 2016).

2.4.8 OPM3 (*Organizational Project Management Maturity Model*)

É um guia de maturidade multidimensional, que direciona os usuários a entenderem qual o estado organizacional de gerenciamento de projetos que a empresa se encontra relacionando este ao padrão. O OPM3 foi projetado para ajudar as empresas alinharem suas operações com as estratégias de negócio, para isso há a divisão desta fase em duas etapas, sendo a primeira a revisão das melhores praticas presentes ou não, feita por uma ferramenta de autoavaliação constituída por 151 perguntas de sim/não a fim de proporcionar a separação de seus três domínios: Projeto, Programa e Portfólio e dentro de cada uma há a separação por etapas de

aprimoramento: Padronização, Medição, Controle e Aprimoramento Contínuo (PMI, 2013); (SANTOS NETO, *et al.* 2011).

Os benefícios do OPM3 são de aumentar a a confiança nos projetos com planejamento estratégico fortemente ligado à sua execução, disponibilizando medidas de correlação entre a gestão de projetos, programas e portfólios e o sucesso organizacional. É um método que identifica as melhores práticas organizacionais (PMI, 2003).

2.5 CONCLUSÃO DO CAPÍTULO

Este capítulo trouxe de forma abrangente alguma das ferramentas da qualidade que possam vir a auxiliar as empresas na manutenção dos custos da qualidade e não qualidade de forma linear, para complementar os métodos ágeis expostos também neste mesmo capítulo. Existem muito mais ferramentas da qualidade e métodos ágeis, porém os que foram pesquisados e que primeiramente se mostraram mais interessantes acabaram sendo estudados para serem melhor compreendidos.

3 METODOLOGIA

De acordo com Silva e Menezes (2005) este trabalho se trata de uma pesquisa, pois procura respostas para o inquirido assunto. É uma pesquisa aplicada do ponto de vista da sua natureza já que procura gerar conhecimentos sobre sua aplicação prática para solucionar problemas específicos, neste caso dos custos da qualidade e da não-qualidade, direcionadas às empresas em questão. Trata-se de uma pesquisa qualitativa já que não requer o uso de métodos e técnicas estatísticas. Quanto aos seus objetivos, este trabalho se trata de uma pesquisa descritiva pois busca descrever as características das empresas desenvolvedoras de software, com coleta de dados por meio de questionário; serão realizadas pesquisas e desenvolvidos questionários baseados nos estudos bibliográficos de outros casos. Do ponto de vista de desenvolvimento técnico, é uma pesquisa bibliográfica já que busca material já publicado em livros, artigos e matérias na internet.

Para o trabalho serão efetuados os seguintes passos:

- encontrar empresas de desenvolvimento de software na cidade de Maringá-PR e verificar se aceitam participar do estudo;
- enviar questionários para melhor identificar as empresas;
- analisar os resultados e compará-los entre si;
- encontrar meios para melhorar o processo do desenvolvimento de *software*.

A pesquisa foi iniciada no mês de setembro de 2015 com o intuito de analisar os custos da qualidade e da não-qualidade em empresas do ramo de *software* situadas na região metropolitana de Maringá-PR. Para isso foi desenvolvido um questionário de para melhor conhecimento das empresas.

Inicialmente fez-se pesquisa em livros sobre possíveis perguntas que poderiam agregar informações úteis ao questionário juntamente com leituras sobre o assunto em questão procurando entender as necessidades envolvidas no desenvolvimento de *software*, mesmo como indicadores de custos da qualidade e a importância que o cliente possui no processo. Em uma planilha foram inseridas as possíveis questões e depois foram analisadas e escolhidas as melhores, fazendo modificações necessárias e inserindo novas questões. Foi desenvolvido o questionário (Apêndice A) no aplicativo *google docs* e em seguida iniciou-se a pesquisa de possíveis empresas participantes. Foi obtido um portfólio da APL Software Maringá onde obteve-se os e-mails das 58 empresas cadastradas, sendo assim enviados os questionários.

Foi feito um questionário a fim de compreender melhor as empresas a respeito de seu interesse na qualidade do processo produtivo de um *software* e assim obter dados sobre as empresas em interesse. De início houve pesquisa em livros e artigos na *internet* para identificar quais questões seriam mais pertinentes para o assunto.

Inicialmente os questionários (Apêndice A) foram enviados para os e-mails obtidos no portfólio da APL de Maringá, obtendo-se apenas quatro respostas. Com isso foram feitos telefonemas nas cinquenta e oito empresas do site da APL de Maringá em busca do e-mail dos responsáveis do setor de qualidade ou pelo desenvolvimento de *software*. Duas dessas empresas eram escolas de informática e não desenvolviam *software*, cinco delas o telefone indicado no site da APL ou no próprio site da empresa não existiam, em outras cinco não houve respostas nas ligações mesmo sendo feitas em dias e horários diferentes. Das quarenta e seis restantes esperou-se as respostas por um mês e meio, e obteve-se apenas dezoito respostas totalizando em 39% de retorno.

Na figura 9 é apresentado um fluxograma para melhor entendimento das atividades que foram efetuadas.

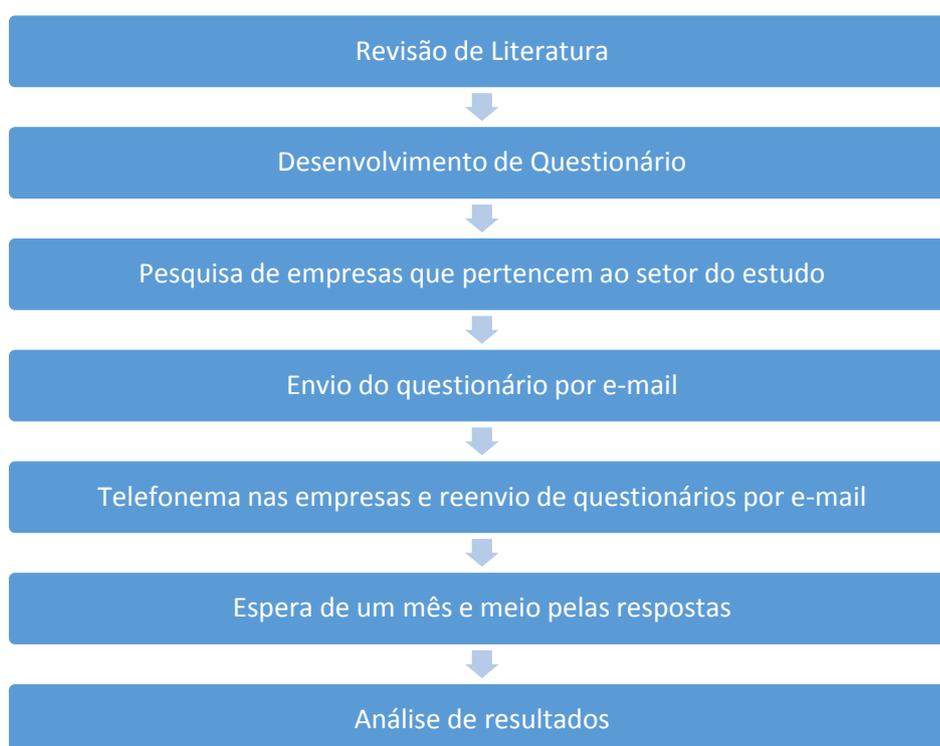


Figura 9: Fluxo de atividades para a realização da pesquisa.

4 DESENVOLVIMENTO

4.1 A CIDADE DE MARINGÁ

O gerente do Sebrae em 2011, Luiz Carlos da Silva, afirmou no 9º Encontro de Líderes de APLs de Tecnologia de Informação do Estado do Paraná que o estado subiu de décimo lugar no *ranking* nacional em número de certificações da MPS.BR¹, para o quarto lugar em 2012, e de acordo com o site Software By Maringá (2016) passou a ser o primeiro lugar em 2015, mostrando a dedicação das empresas paranaenses em serem exemplos no ramo de TI (APL SOFTWARE MARINGÁ, 2016); (HERRERO, 2016).

De acordo com dados de 2012 do Sebrae, de 2007 à 2012 o município de Maringá-PR se tornou um grande polo de Tecnologia da Informação, com empresas no ramo de desenvolvimento de *software* crescendo 81% de 2008 à 2012. Os dados também mostram que empresas ligadas ao Arranjo Produtivo Local (APL) de Software de Maringá aumentaram em 134% o número de funcionários, uma quantidade muito mais elevada que a do país sendo de 46% e a do estado com 75%. O enfoque nesse tipo de empresa aumenta a cada ano, possuindo um crescimento do faturamento anual entre 20% e 30% segundo o Jornal Gazeta do Povo em 2011 (GAZETA DO POVO, 2016).

4.2 DAS EMPRESAS PESQUISADAS

O trabalho procurou identificar o relacionamento das empresas de desenvolvimento de *software* com o custo da qualidade e da não qualidade neste setor. Das quarenta e seis empresas, apenas quatro não são especificamente da cidade de Maringá, mas se encontram inseridas na região metropolitana. Todas as empresas atuam na área de desenvolvimento de *software*.

O questionário possui vinte e nove questões sendo elas dezoito questões objetivas e onze discursivas. A primeira questão procurou identificar qual cargo a pessoa que está respondendo o questionário ocupa. Obteve-se como respondentes seis pessoas relacionadas ao desenvolvimento de *software*, quatro da área de projetos, três sócios, um CEO e quatro da área administrativa. Não houve qualquer profissional que tenha respondido o questionário que seja diretamente relacionado à área da qualidade

¹ MPS.BR: (Melhoria do Processo de Software Brasileiro) é um programa para melhorar a capacidade de desenvolvimento de software nas empresas brasileiras

Para identificar o porte da empresa, perguntou-se a quantidade de funcionários e para classificá-las utilizou-se a classificação do SEBRAE (Tabela 2) para comércio e serviços.

Comércio e Serviços	
Classificação	Quantidade de funcionários
Micro	até 9 funcionários
Pequena	de 10 a 49 funcionários
Média	de 50 a 99 funcionários
Grande	mais de 100 funcionários

Tabela 1: Classificação das empresas de comércio e serviço por número de funcionários

Das empresas avaliadas, sete são classificadas como microempresas, quatro como empresas de pequeno porte, uma empresa de médio porte e seis empresas de grande porte. Na Figura 10 é ilustrado o percentual referente ao porte das empresas estudadas.

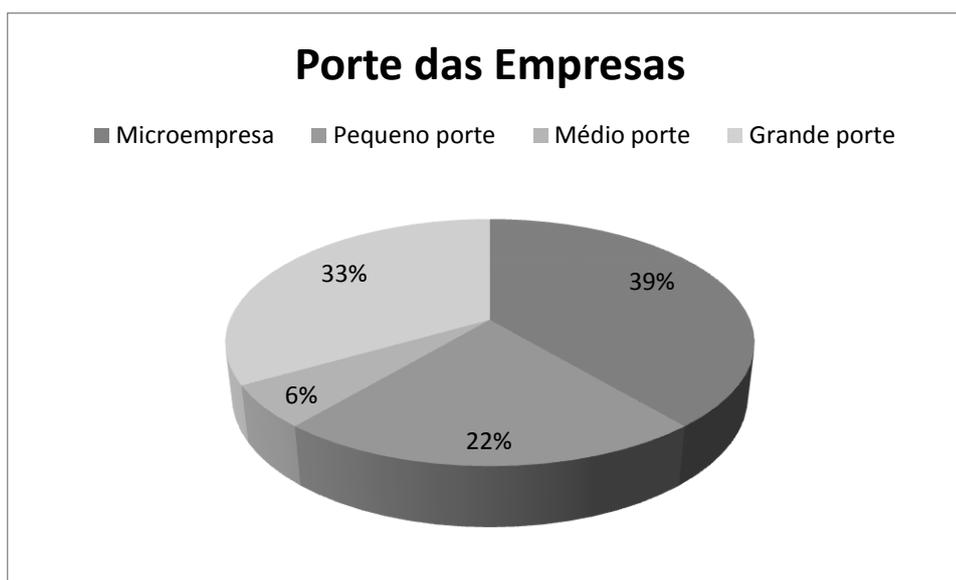


Figura 10: Porte das empresas

Para saber se há um melhor controle das atividades e possível inserção de departamento, procurou-se saber se havia a setorização da empresa. Percebeu-se que das dezoito empresas que participaram apenas uma não possuía setorização. Esta única empresa que não possuía divisão de setores era uma microempresa. É importante frisar que 94,4% das empresas possui setorização que possibilita agilidade nos processos mostrando interesse em crescimento.

Para entender melhor em que ramo a organização se insere, procurou-se saber qual o foco das empresas, se em manutenção e atualização de *softwares* ou na criação de novos *softwares*.

Ficou visível que 72%, ou seja, a maioria das empresas de *software* da região metropolitana de Maringá buscam a manutenção e atualização de *softwares* como seus principais serviços, mas isso não significa que elas não desenvolvem *software* paralelamente aos serviços de manutenção (Figura 11).



Figura 11: Foco de atuação das empresas questionadas.

Apesar do maior foco das empresas em manutenções e atualizações de *software*, muitas delas possuem equipe destinada a criação de novos softwares. Com isso pode-se perceber que mesmo não sendo sua base, a criação de *software* trabalha paralelamente aos outros interesses. Deste modo, conforme a Figura 12, 78% das empresas possuem equipe destinada às novas criações e apenas 22% não possuem.



Figura 12: Quantidade de empresas que possuem equipes destinadas às novas criações.

Dentre as treze empresas que focam na manutenção e atualização de *software*, apenas duas não possuíam uma equipe destinada às novas criações (Figura 13), e dentre as cinco que buscam o desenvolvimento de *software* como atividade principal, duas não possuem esse tipo de equipe (Figura 14). Para melhor compreensão procurou-se saber o porte das empresas que não possuem uma equipe destinada a novas criações e obteve-se uma microempresa, uma pequena empresa, uma média empresa e uma grande empresa, mostrando assim que a quantidade de funcionários da organização não está relacionada com a presença deste tipo de equipe.

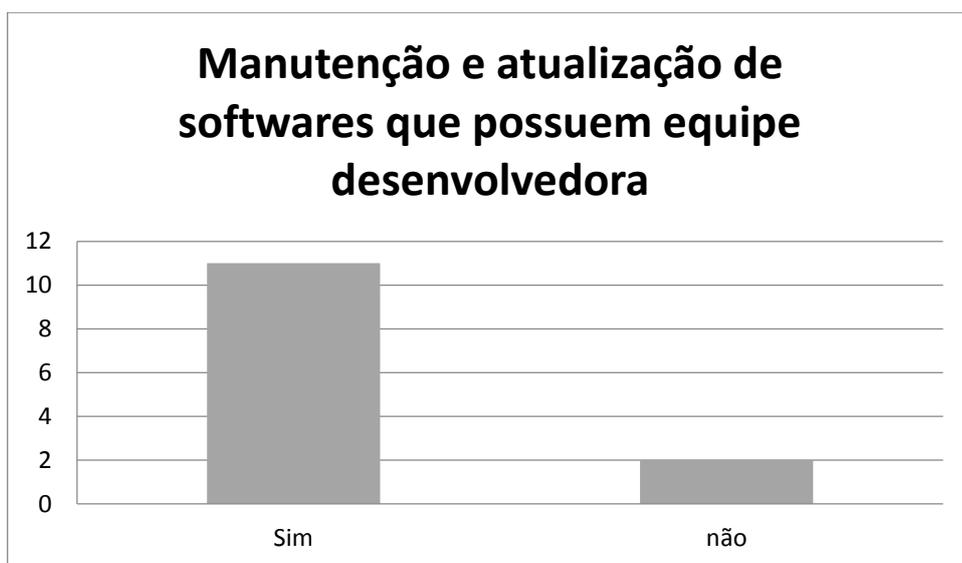


Figura 13: Empresas que focam manutenção e atualização de software e possuem equipe desenvolvedora.

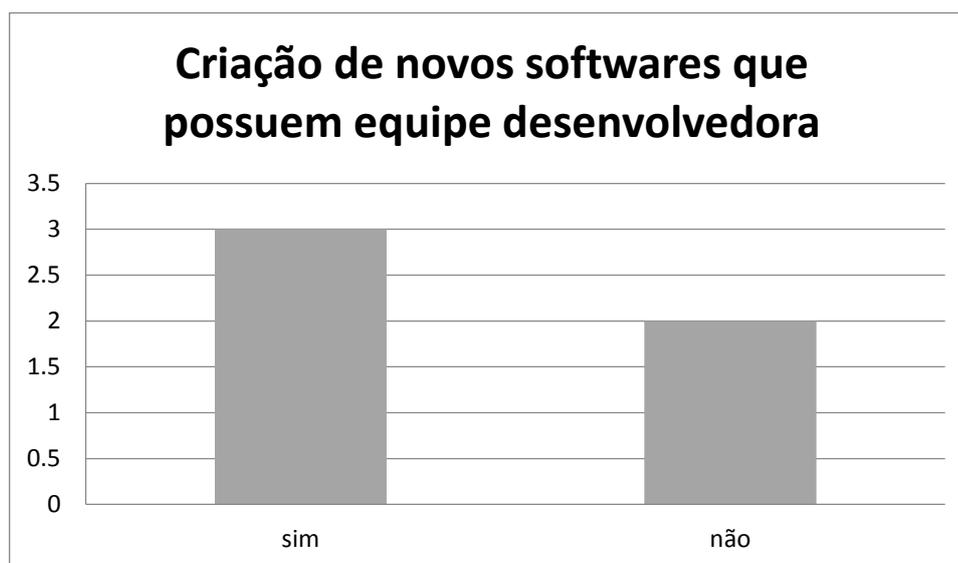


Figura 14: Empresas que focam criação de novos softwares e possuem equipe desenvolvedora.

Procurou-se identificar o tamanho da equipe destinada ao desenvolvimento de *software*, para saber se uma grande porcentagem dos funcionários fazia parte das mesmas. Percebe-se que independentemente do tamanho da empresa, a porcentagem da equipe desenvolvedora está entre 32% e 49% da quantidade de funcionários (Figura 15). Comparando com outras questões já avaliadas, percebe-se que apesar de algumas empresas não possuírem uma equipe exclusivamente destinada a esse tipo de atividade, por questões de necessidade não há a opção de não existir colaboradores com essa função.

Também se avaliou a relação do porte da empresa com o número de funcionários na equipe desenvolvedora. É interessante observar, que as microempresas possuem o maior percentual de funcionários na equipe desenvolvedora, isso se dá pois com nesta empresa o número de funcionários é reduzido, um mesmo acaba ocupando mais de uma função (Figura 15). Ao analisar as pequenas, médias e grandes empresas, observa-se que o percentual de equipe desenvolvedora cresce de forma gradual e com isso pode se perceber que quanto maior a empresa, maior a equipe de desenvolvedora, esses percentuais não necessariamente correspondem a uma única equipe, mas sim a quantidade de pessoas que possuem essa função.

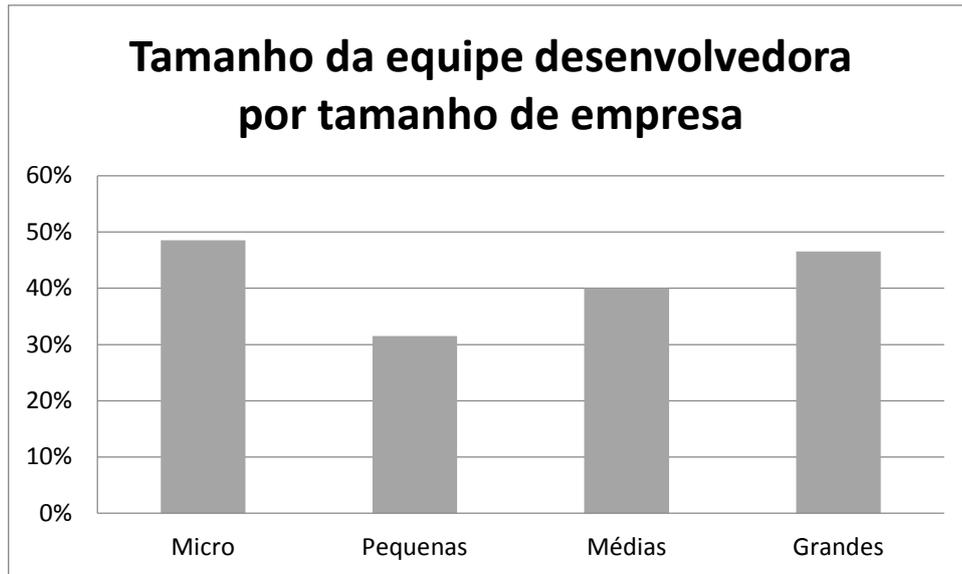


Figura 15: Percentagem da empresa que forma as equipes desenvolvedoras.

Para melhor compreender os métodos utilizados pelas empresas no desenvolvimento de *software*, procurou-se saber quais ferramentas eram mais utilizadas por elas, dentre as questionadas estavam: Scrum, Mural de Gantt, Cascata, SW-CMM/CMMI, PSP, TSP, PMMM, OPM3. Além dessas surgiu a ferramenta Kanban e Extreme Programming.

O Scrum é o método mais utilizado entre as empresas por ser um método ágil e abrangente, porém ele não descreve o que fazer em cada situação, gerando algumas vezes a necessidade da utilização de outras ferramentas para agregar informações. Seis das empresas que responderam o questionário utilizam mais de uma ferramenta e são cinco grandes empresas e duas micro empresas, duas não utilizam ferramenta alguma sendo ambas microempresas. O modelo cascata foi um dos pioneiros no auxílio ao desenvolvimento de *software* e fica visível o seu desuso atualmente, já que os métodos ágeis estão sendo mais fortemente inseridos nas empresas já que seus resultados são melhores (Figura 16).

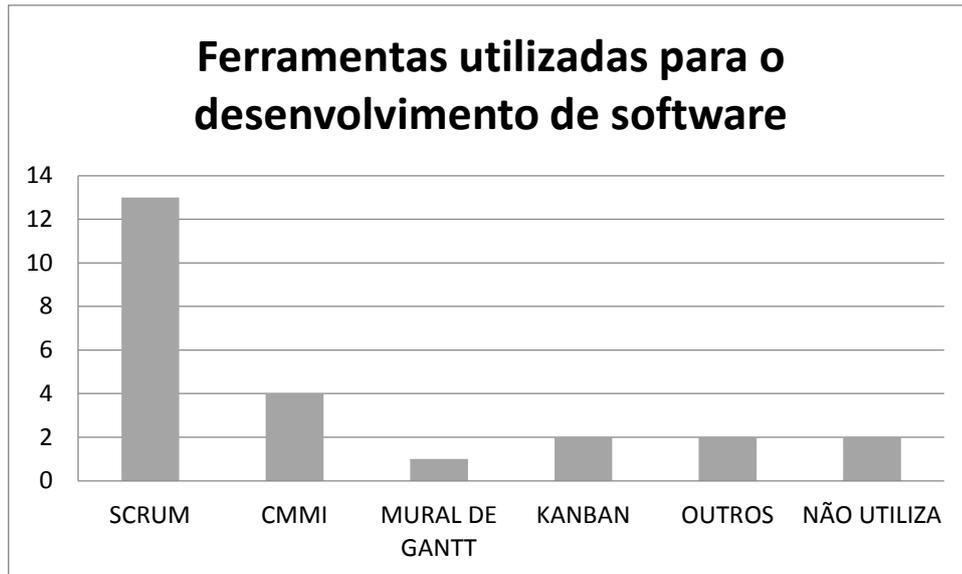


Figura 16: Ferramentas mais utilizadas para o desenvolvimento de software.

A utilização de reuniões periódicas entre ciclos de trabalho faz parte de alguns dos métodos em questão auxiliando na análise do que está sendo desenvolvido e nos planejamentos futuros. Dentre as dezoito empresas que responderam o questionário, a maioria possui reuniões semanais, não havendo resposta positiva para reuniões mensais ou a não execução de reuniões, além de duas microempresas que executam reuniões conforme necessidade e três que fazem reuniões diárias, sendo elas uma micro, uma pequena e uma grande empresa (Figura 17).

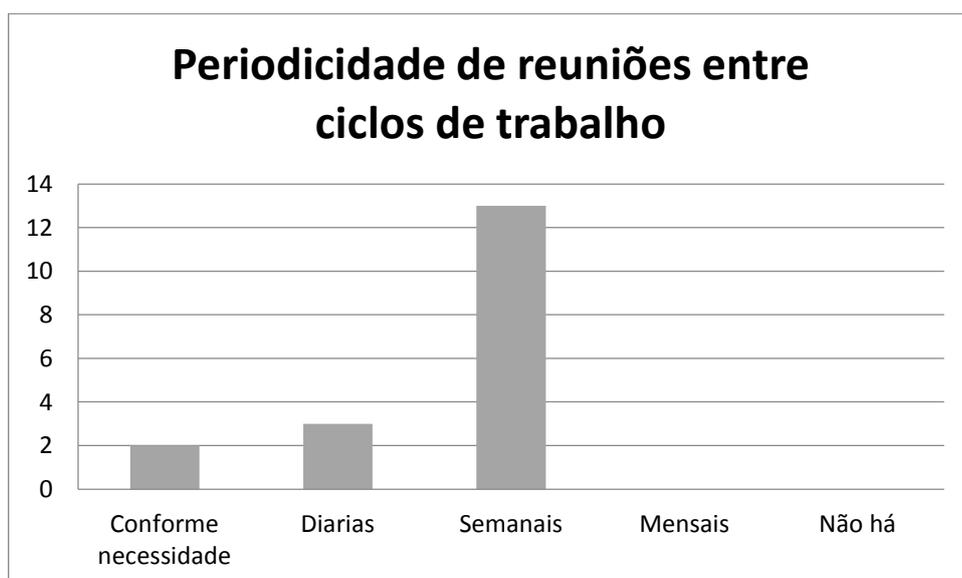


Figura 17: Periodicidade de reuniões entre ciclos de trabalho.

Para entender melhor como o método Scrum é utilizado nas empresas, procurou-se saber se este método é mais utilizado para desenvolvimento de *software* ou para gestão de desenvolvimento. Tem-se como gestão do desenvolvimento a organização, planejamento e monitoramento do que se foi planejado para a execução de um *software*, já o desenvolvimento de *software* é a etapa da elaboração e implementação do sistema computacional transformando a necessidade do usuário em um produto final.

Como mostra a Figura 18, 33% dos entrevistados utiliza o método Scrum tanto para o desenvolvimento de *software* quanto para a gestão de desenvolvimento, isso mostra que a abrangência dos métodos ágeis se torna presente em várias etapas do processo dando um maior controle da execução a empresa sobre a execução de suas tarefas.

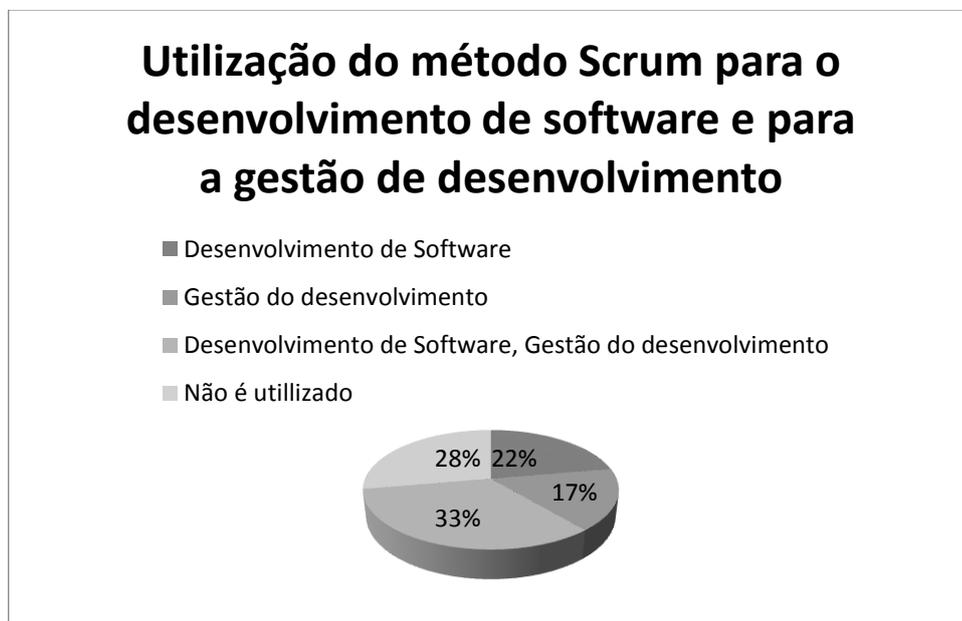


Figura 18: Maneira como as empresas costumam utilizar o método Scrum.

Para saber a percepção dos gestores, foi questionado se estavam satisfeitos com o que a empresa utiliza, dentre as dezoito respostas, quinze se dizem satisfeitos e três insatisfeitos. Das três empresas que não estão satisfeitas com o método que utilizam, uma utiliza o método Scrum, uma o Kanban e a outra não utilizava nenhum método.

Com isso procurou saber se havia alguma sugestão de outros métodos que poderiam ser utilizados, independente se estavam satisfeitos ou não com o que era usado na empresa. As respostas obtidas foram bem divididas, metade acredita ser necessário implantar novos métodos e metade acredita que não. Fazendo uma comparação com a questão anterior percebe-se que alguns querem mudanças na empresa mas não sabem como fazê-las e como

decidir o melhor método para ser utilizado. Das nove respostas obtidas a favor da implantação de novos métodos, uma delas acreditar ser interessante a introdução do Kanban e a outra procura encontrar método para colocá-lo em prática. Outra empresa busca o uso do CMMI para ter o foco maior no processo e não nas pessoas, as outras respostas não deixaram claras as intenções da empresa.

Partindo para a qualidade de *software* procurou saber se havia algum responsável pelo monitoramento da qualidade. Em onze empresas há o responsável pela qualidade e nas outras seis não. Isso mostra que mesmo possuindo setorização na empresa muitas vezes o setor de qualidade é inexistente, pois apenas uma das empresas participantes não possui setorização (Figura 19).



Figura 19: Quantas empresas possuem um responsável pela qualidade.

Procurou saber se há o monitoramento dos custos da qualidade nas empresas, doze não fazem esse tipo de controle (67%) mostrando que mesmo algumas delas possuindo o responsável pela qualidade, esse não executa a função do controle dos custos da qualidade (Figura 20).

Procurou-se saber de que maneira as empresas que fazem o monitoramento dos custos da qualidade executavam tal função. Elas utilizam indicadores, fazem reuniões, executam pesquisa de qualidade, utilizam rateios coletando dados e cruzando-os.

Alguns dos indicadores utilizados pelos questionados estão relacionados às horas gastas no desenvolvimento do *software* e horas de retrabalho. Também levam em consideração o cliente

e se este encontra erros e *bugs* no *software*, caso encontre o erro é mais grave que se encontrado antes de ser entregue.

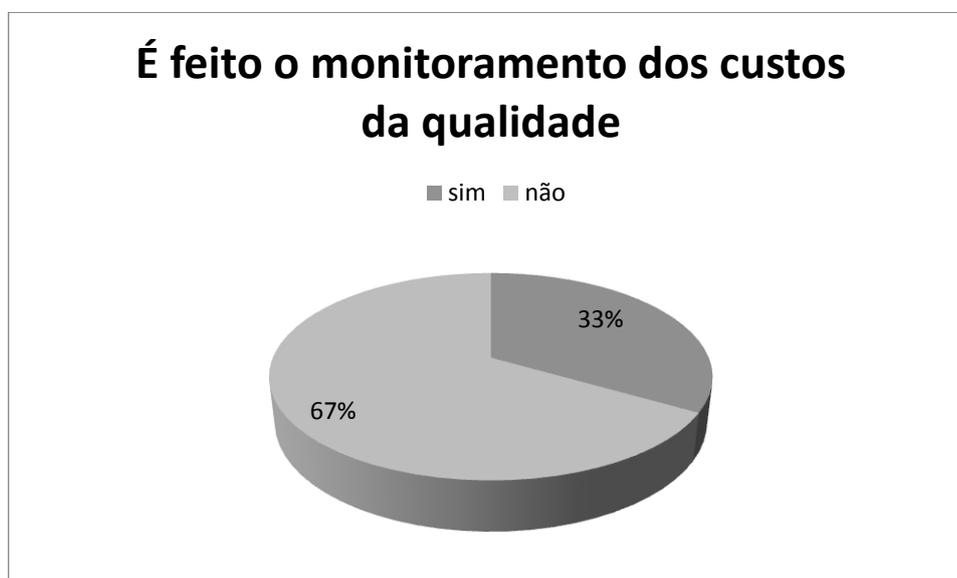


Figura 20: Quantas empresas fazem o monitoramento dos custos da qualidade.

Procurou-se saber de que maneira as empresas que fazem o monitoramento dos custos da qualidade executavam tal função. Elas utilizam indicadores, fazem reuniões, executam pesquisa de qualidade, utilizam rateios coletando dados e cruzando-os.

Os indicadores da qualidade utilizados são referentes à eficiência da qualidade, satisfação do cliente e prazo.

Foram feitas as mesmas perguntas sobre se é feito o monitoramento dos custos da não qualidade e como é feito, se há a utilização de indicadores e quais são estes. Para a não qualidade mensuram a quantidade de horas de desenvolvedores utilizados em retrabalho e correções de bugs, cada etapa do processo de criação possui um peso de mensuração da não qualidade.

Alguns dos indicadores utilizados pelos questionados estão relacionados às horas gastas no desenvolvimento do *software* e horas de retrabalho, comparando o previsto com o realizado, taxa de abertura de *bugs*, taxa de incidentes e validação dos coletores de dados (ferramentas automatizadas). Também levam em consideração a satisfação do cliente e se este encontra erros e bugs no *software*, caso encontre o erro é mais grave e tem mais peso que se encontrado antes de ser entregue. Relacionam a quantidade de horas de desenvolvedores utilizadas em retrabalho e correção de bugs versus horas disponíveis da equipe.

Percebe-se que é mais fácil de mensurar a não qualidade do que a própria qualidade, as empresas mostraram várias formas de visualizar a não qualidade, mas poucas forma de ver como a qualidade está sendo aplicada na empresa.

Quando o monitoramento da qualidade não é feito com indicadores, são utilizados ferramentas de controle que extraem dados sobre os esforços da equipe em novas funcionalidades, retrabalho e melhorias. Melhorias e correções necessárias são utilizadas para contabilizar horas de desenvolvimento utilizadas em atividades que não agregam valor aos clientes além de utilizarem gráficos para melhor visualização de dados.

A preocupação com a satisfação do cliente é constante, tanto no desenvolvimento de software quanto em sua manutenção, assim procurou saber se existe uma equipe destinada à manutenção dos softwares já desenvolvidos e entregues ao cliente. Obteve-se quatorze respostas positivas e quatro negativas. Das quatro que não possuem essa equipe, três são micro empresas e uma é grande empresa. Todas fazem modificações conforme solicitação do cliente e sempre há custo significativo para a empresa caso sejam feitas, mas duas empresas não repassam esses custos ao cliente. Das empresas que cobram o cliente pela manutenção, doze cobram por hora, quatro cobram pela quantidade de modificações no software, três não repassam os custos aos clientes e dois cobram pela avaliação da modificação que deverá ser feita. Com isso percebe que as empresas possuem estilos mistos de cobrança, podendo variar conforme o produto ou a preferência do cliente já que em alguns casos há a possibilidade de pagar uma mensalidade pela assistência no software. Os clientes participam do feedback de novos produtos em dez das empresas o que retorna criticas construtivas ajudando a encontrar quais os próximos caminhos a se seguir.

4.3 PROPOSTA DE USO DE FERRAMENTAS DA QUALIDADE EM AUXÍLIO AOS MÉTODOS ÁGEIS

Com a análise das respostas das empresas, percebeu-se que há ainda muitas ações a serem realizadas na área da qualidade, podendo estas investir em melhorias, bem como mensurar seus custos de qualidade e não qualidade.

Deste modo, algumas ferramentas da qualidade podem vir a se somar aos métodos ágeis como forma de orientar esforços e adquirir conhecimento dos pontos importantes do desenvolvimento de *software*. É interessante frisar que se pode utilizar essas ferramentas já nas etapas iniciais do processo de desenvolvimento de *software*, já que é onde podem ocorrer

menores perdas quando um problema é encontrado, ou seja, quanto antes resolvido e encontrado, menos custos e tempo serão utilizados para solucioná-lo.

Em primeiro lugar a ferramenta BPMN auxilia a mapear o processo presente na empresa deixando de forma mais visível as etapas da organização de forma sequencial. Com isso há uma maior facilidade de visualizar o contexto do desenvolvimento de software ou de sua manutenção, conseguindo assim mapear em que etapa do processo o software se encontra e quais suas etapas seguintes e anteriores. Também, por meio do mapeamento de processo é possível identificar quais são as atividades que agregam e que não agregam valor aos processos, podendo estes serem melhorados. Com a identificação das principais atividades, consegue-se enxergar centros de custos e estabelecer indicadores, tanto da qualidade como da não qualidade, o que favoreceria para melhor cobrança dos resultados por parte dos funcionários, como na formação de preços do projeto de software.

Propõe-se a utilização da folha de verificação, que permite a coleta e análise de dados de problemas que ocorrem na organização, proporcionando evidências objetiva dos problemas que envolvem a produção nas diferentes etapas do processo. Com a folha de verificação feita o passo seguinte é colocar os dados no Gráfico de Pareto de forma a deixar mais visível os dados coletados. No Gráfico de Pareto os defeitos e problemas devem ser inseridos de forma sequencial partindo dos de maior ocorrência para os de menor, assim permite a visualização dos principais problemas que acontecem nas etapas de desenvolvimento de software.

Buscando aprofundar mais ainda nas ocorrências negativas do processo, o Diagrama de Ishikawa permite encontrar os pontos críticos do processo e buscar o real motivo para os problemas estarem ocorrendo, permitindo assim o desmembramento da causa principal em causas menores obtidas através do *brainstorming*, facilitando a atuação de melhorias de forma pontual.

Cruzando os dados do Diagrama de Ishikawa com o mapeamento do processo, pode-se encontrar onde ocorre a maior quantidade de problemas, e assim proporcionando aos gestores uma melhor visualização da etapa que devem ser dedicados maiores esforços para que o software seja feito com o mínimo de defeitos.

Essas quatro ferramentas da qualidade são simples e de fácil utilização, permitindo que todos da organização participem e exponham suas ideias para o crescimento da empresa. Podem ser utilizadas para auxiliar

5 CONSIDERAÇÕES FINAIS

5.1 CONTRIBUIÇÕES

Ao final da pesquisa é importante comentar suas contribuições pois nela pode-se entender a forma das empresas lidarem com o tema custos da qualidade, e com o levantamento bibliográfico as informações pertinentes para o desenvolvimento do trabalho foram obtidas. Percebe-se que o tema de custos da qualidade no desenvolvimento de software não é um assunto muito difundido neste tipo de empresa, tendo em vista que 67% delas não fazem qualquer monitoramento de custos da qualidade no desenvolver de um software.

Fica claro que para as empresas com menos de cinquenta funcionários (micro e pequenas empresas) há uma concentração de 75% dos resultados negativos para o controle de custos da qualidade, sendo apenas três empresas de médio e grande porte que não possuem este tipo de controle. Possivelmente pela pouca quantidade de funcionários e muitas vezes o mesmo ter de efetuar mais de uma função, este tipo de controle é negligenciado.

Á partir das respostas obtidas, pode se verificar que o Scrum é o método mais aceito pelos programadores, pois se trata de um método ágil, definido pelas suas reuniões periódicas onde procura ter conhecimento do que já foi feito, do que se está fazendo e do que se irá fazer, deixando o ambiente de trabalho mais bem gerenciado.

O método Kanban, como foi mencionado, é algo que deve ser pensado na utilização de desenvolvimento de software pois se trata de uma ferramenta adaptada da indústria de manufatura, implementá-la significa tornar o modelo de trabalho visível, incentivando as pessoas a trabalharem e colaborarem para que o produto chegue ao cliente em modo contínuo além de ser um modelo puxado que permite visualizar o processo de forma mais rápida e assim suas anormalidades, permitindo antecipar medidas necessárias para a sua correção. Apesar de apenas duas empresas utilizarem este método, ele se mostra muito interessante na utilização para desenvolvimento de manutenção de software, onde outras empresas deveriam procurar implementá-los na melhoria do processo.

Ao fim da pesquisa fica visível a importância do tema de custos da qualidade no desenvolvimento de software, percebe-se que muitas das empresas não utilizam métricas para controlar estes custos, o que pode ocasionar em perdas para a organização. Essas perdas podem ser em moeda ou clientes. A perda monetária é palpável e mensurável, porém a perda de um cliente é difícil de ser medida, já que além do cliente se perde indicações que poderiam

trazer novos clientes. Isso torna a qualidade de um processo uma etapa importante a ser adquirida pela empresa e fazer o controle dos seus custos traz o melhor controle de esforços em todo o processo.

Pode-se concluir que a forma de desenvolver um *software* é muito particular de cada empresa, o que pode ser bom para uma não necessariamente funciona em outras, porém o controle dos custos da qualidade só vem a somar em qualquer uma delas, pois auxilia no apontamento das falhas, mensurando quais as mais graves e assim podendo direcionar melhorias à uma determinada etapa além de a comparação por histórico de custos direcionar se as medidas tomadas estão favorecendo a empresa.

5.2 LIMITAÇÕES

Houve algumas dificuldades na execução do trabalho e a primeira encontrada foi na coleta de dados. Ao início da pesquisa, foram mandados questionários via e-mails obtidos através do portfólio da APL Software Maringá, obtendo assim quatro respostas. A fim de obter mais participações, entrou-se no site da APL Software Maringá onde possuía cinquenta e oito empresas registradas, destas, duas eram escolas de informática e não desenvolviam software, cinco estavam com o número de telefone errado e em outras cinco não houve respostas das ligações. Restaram-se então quarenta e seis empresas das quais nos telefonemas procurou-se conversar com o responsável pelo desenvolvimento de software ou pela qualidade, algumas das vezes conversando com o sócio ou responsável de TI. Esperou-se um mês e meio e assim obteve-se dezoito respostas sendo apenas 39% da quantidade de questionários enviados.

Outra dificuldade foi a falta de estudos diretamente relacionados aos custos da qualidade no desenvolvimento de software. Encontrava-se assuntos relacionados à custo da qualidade, qualidade no desenvolvimento de software e garantia da qualidade de software mas poucos eram encontrados no tema propriamente estudado.

5.3 TRABALHOS FUTUROS

O tema de custos da qualidade no desenvolvimento de *software* traz um grande cenário de possibilidades. Percebe-se que os custos da qualidade e suas metodologias possuem muita relação à gestão de projetos, seria interessante o maior aprofundamento relacionando ambos os temas, para assim obter a melhor resposta das questões efetuadas a fim de conseguir

identificar métricas e indicadores para o possível desenvolvimento de parâmetros que auxiliem a identificação dos custos da qualidade na área do desenvolvimento de *software*. A gestão de projetos está intimamente relacionada ao tema estudado e os custos da qualidade no desenvolvimento de software é apenas uma fração do que se pode adquirir na utilização dos métodos ágeis.

Uma das empresas sugeriu que a pesquisa fosse feita verbalmente, o que agregaria mais conhecimento já que algumas das questões possuem resposta limitada, abrangendo assim uma maior gama de informações.

6 REFERÊNCIAS

- APL SOFTWARE MARINGÁ**, 2011. Disponível em: <<http://www.sub100.com.br/empresas/comercial/apl/layout.php?codigo=39>>. Acesso em: 06/01/2016.
- AYRES M., **Jornal Gazeta do Povo**, 2012. Disponível em: <<http://www.gazetadopovo.com.br/vida-e-cidadania/maringa/com-crescimento-acima-da-media-maringa-se-firma-como-polo-regional-de-ti-3xskoz9efdn9q44vt775y8ocu>>. Acesso em: 07/01/2016.
- BALLESTERO-ALVAREZ, M. E. **Gestão da Qualidade, Produção e Operações**. São Paulo: Editora Atlas. 2010.
- BARTIÉ, A. **Garantia da Qualidade de Software: as melhores práticas de engenharia de software aplicadas à sua empresa**. 13ª triagem. Rio de Janeiro: Editora Elsevier, 2002. 291 p.
- BASTIANI, J. A. **Por que utilizar o gráfico de Gantt?**. Disponível em: <<http://www.blogdaqualidade.com.br/por-que-utilizar-o-grafico-de-gantt/>>. Acesso em: 23/01/2016.
- CAMPÃO, C. A., *et al.* **Análise dos custos da qualidade: um estudo de caso em uma empresa alimentícia**. Fonte: Revista Espacios. Disponível em: <<http://www.revistaespacios.com/a12v33n03/123303261.html>>. Acesso em: 20/04/2015.
- CARPINETTI, L. C. **Gestão da Qualidade: Conceitos e Técnicas**. São Paulo: Atlas. 2010.
- FALCONI, V. C. **TQC - Controle da Qualidade Total no estilo japonês**. Nova Lima, MG: 8ª edição. Editora INDG Tecnologia e Serviços Ltda. 2014.
- FERNANDES, W. A., & INMETRO. **O Movimento da Qualidade no Brasil**. São Paulo: Essencial Idea. 2011.
- HERRERO D., **Software by Maringá**, 2015. Disponível em: <<https://ciranda.me/sbm/software-by-maringa/post/parana-e-lider-em-certificacoes-mps-br>>. Acesso em: 24/01/2016.
- HUMPHREY, W.S. **The personal software processSM (PSPSM)**. Carnegie Mellon. Software Engineering Institute. Pittsburgh, PA. CMU/SEI-200-TR-022. ESC-TR-2000-022. Novembro 2000. Disponível em <<http://www.sei.cmu.edu/reports/00tr022.pdf>>. Acesso em: 25/01/2016.
- ISD BRASIL. **O que é CMMI?**. Disponível em <<http://www.isdbrasil.com.br/o-que-e-cmmi.php>>. Acesso em: 15/01/2016.
- JURAN, J. M., & GRZYNA, F. M. **Controle da Qualidade: Componentes Básicos da Função Qualidade Volume II**. São Paulo: Editora McGraw-Hill Ltda. 1991.
- KUME, H. **Métodos estatísticos para melhoria da qualidade**. São Paulo: Editora Gente. 1993.

LAHOZ, C., & SANT'ANNA, N. **Os padrões ISSO/IEC 12207 e 15504 e a modelagem de processos da qualidade de software.** *IAE, CTA, INPE*, p. 6. 2003.

MACDONALD, R. **Getting things 'Done Done'.** 2015. Disponível em <<https://www.madetech.com/blog/getting-things-done-done>>. Acesso em: 15/01/2016.

MATTOS, J. C., & TOLEDO, J. C. **Custo da qualidade: diagnóstico nas empresas com certificação ISO 9000.** *Gestão & Produção*. Departamento de Engenharia de Produção, Universidade Federal de São Carlos, UFScar de dez de 1998. pp. v.5, n.3, p. 312-324.

MELO, M. M., SOUZA, F. J., & ARAÚJO, A. O. Custos da qualidade: um estudo em um restaurante de grande porte da cidade de natal-rn. In: *XIX CONGRESSO BRASILEIRO DE CUSTOS*, p. 16. 12 A 14 de NOVEMBRO de 2012. Bento Gonçalves. **Anais...** Bento Gonçalves: UNB; UFPB; UFRN, 2012.

MIGUEL, P. A. **Qualidade: enfoque e ferramentas.** São Paulo: Artliber. 2001.

MOORI, R. G., & SILVA, R. V. **Gestão do Custo da Qualidade nas Empresas Químicas do Brasil.** *Gestão de Operações e Logística, ERA*, Vol 43. pp. p. 36 - 49. jul. set de 2003.

Object Management Group Business Process Model and Notation. Disponível em: <<http://www.bpmn.org>>. Acesso em: 22/04/2015.

OLIVEIRA, J. A., *et al.* **Um estudo sobre a utilização de sistemas, programas e ferramentas da qualidade em empresas do interior de São Paulo.** SCIELO. UNESP, Bauru-SP Produção, v. 21, n.4, p. 708-723. Out-Dez de 2011.

PALADINI, E. P. **Gestão Estratégica da Qualidade: Princípios, Métodos e Processos.** São Paulo: Editora Atlas. 2008.

PEREIRA, P., TORREÃO, P., MARCAL, A.S., **Entendendo Scrum para gerenciar Projetos de Forma Ágil.** C.E.S.A.R. versão 1.8. Mundo PM. 2007. 11p. Disponível em <<http://www.siq.com.br/DOCS/EntendendoScrumparaGerenciarProjetosdeFormaAgil.pdf>>. Acesso em: 21/01/2016.

PRESSMAN, R. S. **Engenharia de Software: Uma abordagem profissional.** 7ª ed. Porto Alegre: Editora Afiliada, 2011.

PMI - PROJECT MANAGEMENT INSTITUTE, INC. **Organizational Project Management Maturity Model (OPM3).** Knowledge Foundation. Newtown Square, Pennsylvania, USA, 2003. Disponível em: <http://www.dphu.org/uploads/attachements/books/books_5912_0.pdf>. Acesso em 26/01/2016.

RAMOS, R. A. **Processo de Desenvolvimento de Software.** Univasf - Engenharia de Software. 2009. Disponível em <http://www.univasf.edu.br/~ricardo.aramos/disciplinas/ESI2009_2/Aula02.pdf>. Acesso em 24/04/2015.

REIS, T. **Gráfico de Gantt: como e por que utilizá-lo para gerenciar projetos?.** Project Builder. 2016 Disponível em: <<http://www.projectbuilder.com.br/blog->

pb/entry/conhecimentos/grafico-de-gantt-como-e-por-que-utiliza-lo-para-gerenciar-projetos>. Acesso em 15/01/2016.

SANTOS NETO, N. F., ANTONIO, L. M., NOVASKI, O. **Análise da metodologia de avaliação OPM3® aplicada ao gerenciamento dos riscos**. Revista Tecnológica. Artigo 8. Maringá, v. 20, p. 75-82, 2011.

SCHIFFAUEROVA, A. THOMSON, V. **A review of research on cost of quality models and best practices**. Department of Mechanical Engineering, McGill University, Montreal, Canada. International Journal of Quality & Reliability Management, Vol. 23 Iss 6 pp. 647 - 669

SEBRAE. **Critérios de classificação de empresas: MEI – ME – EPP**. Disponível em: <<http://www.sebrae-sc.com.br/leis/default.asp?vcdtexto=4154>>. Acesso em: 12/01/2016.

SGANDERLA, K. **IPROCESS**. Fonte: Site da iprosses soluções em tecnologia: <<http://blog.iprocess.com.br/2012/11/um-guia-para-iniciar-estudos-em-bpmn-i-atividades-e-sequencia/>>. Acesso em: 22/04/2015.

SILVA, E. L., & MENEZES, E. M. **Metodologia da Pesquisa e Elaboração de Dissertação**. UFSC. Florianópolis. 4 ed. rev. Atual. 2005

SILVA, M.C. **Linha de código**. Disponível em: <<http://www.linhadecodigo.com.br/artigo/1401/cmmi-para-iniciantes.aspx>>. Acesso em: 25/01/2016.

SOUZA, M. A., & COLLAZIOL, E. PLANEJAMENTO E CONTROLE DOS CUSTOS DA QUALIDADE: UMA INVESTIGAÇÃO DA PRÁTICA EMPRESARIAL. XXIX *EnANPAD*, 2005, Brasília. **Anais...** Brasília: USP, 2005 p. 38 - 55. Maio-Agosto de 2006.

SUTHERLAND, J. **A Arte de Fazer o Dobro do Trabalho na Metade do Tempo, Título Original: The art of doing twice the work in half the time**. Traduzido por Natalie Gerhardt, São Paulo: Editora Leya. 2014.

TERESO, D. R. A. **Utilização de ferramentas informáticas na gestão de projectos**. 5º Congresso Luso-Moçambicano de Engenharia. 2º Congresso de Engenharia de Moçambique. Maputo, 2-4 de Setembro de 2008.

TONINI, A. C., CARVALHO, M. M., & SPINOLA, M. D. **Contribuição dos modelos de qualidade e maturidade na melhoria dos processos de software**. *PRODUÇÃO*, v.18, n. 2, p. 275-286. Maio-Agosto 2008.

TOSETTO, M., & BELINNI, C. G. **Gestão sociotécnica do teste de software em projetos de sistemas de informação**. JISTEM - Journal of Information Systems and Technology Management. 2008

TRENTIM, M.H. **Blog Revista Mundo Project Management**. Maturidade em gerenciamento de projetos. 29 de fevereiro de 2012. Disponível em <<http://blog.mundopm.com.br/2012/02/29/maturidade-em-gerenciamento-de-projetos/>>. Acesso em 26/01/2016

VIEIRA, S. *Estatística para a Qualidade*. Rio de Janeiro: Editora Campus. 1999.

APENDICE A

APENDICE A: Questionário enviado às empresas.

Questionário sobre custos da qualidade e da não qualidade no desenvolvimento de software.

O presente questionário faz parte de uma pesquisa relacionada a custos de qualidade em empresas de software de Maringá. É destinado às empresas inseridas no ramo de Software da região metropolitana de Maringá-PR nos desenvolvimentos e assistência dos mesmos. Tem como objetivo verificar se as empresas possuem controle do Custo da Qualidade e da Não-Qualidade nas etapas do processo de criação e manutenção de Software e posteriormente propor a utilização de algumas ferramentas da qualidade como auxílio aos métodos utilizados. É uma pesquisa destinada ao desenvolvimento de um trabalho de conclusão de curso da Universidade Estadual de Maringá no Departamento de Engenharia de Produção da Graduanda Lara Bortolon com a Orientadora Daiane de Genaro. Todas as respostas são sigilosas e apenas para efeito de pesquisa, não serão divulgados dados individuais.

Qual cargo você ocupa na empresa?

A pessoa que está respondendo o questionário é de que cargo

Quantos funcionários a empresa possui?**Ha divisão de setores na empresa?**

Existe uma equipe desenvolvedora, uma de marketing, RH, vendas, etc.

- Sim
- Não

O que é visto como mais importante na empresa?

- Criação de novos softwares
- Manutenção e atualização dos softwares existentes

Há uma equipe destinada às novas criações de software?

- Sim
- Não

Qual o tamanho da equipe desenvolvedora?

A periodicidade de reuniões entre ciclos de trabalho são:

- Inexistentes
- Semanais
- Mensais
- Outros:

Utiliza algum desses métodos para o desenvolvimento de software?

- Scrum
- Mural de Gantt
- Cascata
- SW-CMM/CMMI
- PSP
- TSP
- PMMM
- OPM3
- Outros:

Scrum é utilizado método para desenvolvimento de software ou para a gestão do desenvolvimento?

- Desenvolvimento de Software
- Gestão do desenvolvimento
- Não é utilizado
- Outros:

Está satisfeito com o método utilizado na empresa?

- Sim
- Não

Se está SATISFEITO com o método utilizado, qual o motivo?

Pontos positivos do método utilizado. Se está insatisfeito, escreva "INSATISFEITO"

Acredita ser necessário utilizar outros métodos? Quais?

Se está satisfeito ou não com o método utilizado, acredita que deveriam implementar outros ou trocar para um que seja mais eficiente? Se a resposta for negativa, escreva "NÃO", se positiva escreva quais os métodos

Existe alguém responsável por monitorar a qualidade no desenvolvimento do software?

- Sim
- Não

É feito o monitoramento dos custos da qualidade no desenvolvimento de um software?

- Sim
- Não

Como é feito o monitoramento dos custos da qualidade?

Explique como é feito esse monitoramento. Se não é feito, escreva "NÃO"

Há a utilização de indicadores para obtenção dos custos da qualidade?

- Sim

- Não

Quais os indicadores de custo da qualidade são utilizados?

Caso não utilize escreva "NENHUM"

É feito o monitoramento dos custos da não-qualidade no desenvolvimento de um software?

- Sim
- Não

Como é feito o monitoramento dos custos da não-qualidade?

Explique como é feito esse monitoramento. Se não é feito, escreva "NÃO"

Há a utilização de indicadores para obtenção dos custos da não-qualidade?

- Sim
- Não

Quais os indicadores de custo da não qualidade são utilizados?

Caso não utilize escreva "NENHUM"

De que forma é feita o monitoramento da qualidade se não é feita com indicadores?

Se há o monitoramento da qualidade, ele é feita com planilhas, anotações em tabelas ou outras formas. Se não há monitoramento escreva "INEXISTENTE"

Há uma equipe destinada à manutenção dos softwares já desenvolvidos e entregues ao cliente?

Fixar bugs e problemas relacionados ao software quando solicitado pelo cliente

- Sim
- Não

Faz modificações em um software já existente conforme solicitação do cliente?

Se o cliente quiser algo diferente, acrescentar ou tirar um item do software, isso é feito?

- Sim
- Não

Se as modificações em um software já existente forem feitas, isso gera custos significativos para a empresa?

- Sim
- Não

Esses custos são repassados para os clientes?

É feita a cobrança dos clientes quando ha solicitação de modificações ou o custo ja está incluso no valor base do software

- Sim
- Não

Como é feito o cálculo do custo das modificações solicitadas pelo cliente?

Na resposta: Solicitações seriam a quantidade de vezes que o cliente pede por modificações, independente de quantas ele queira (um valor pré-determinado por encontro com o cliente). O cliente pede uma mudança ou dez mudanças e é cobrado o mesmo valor sempre que solicitá-las

- É cobrado por hora
- É cobrado pela quantidade de modificações no software
- É cobrado pela quantidade de solicitações à empresa
- Não há custo para o cliente
- Outros:

Cliente participa do feedback de novos produtos?

O cliente faz o pedido de um novo produto e acompanha o seu desenvolvimento dando o seu feedback

- Sim
- Não

Considerações

Algum comentário que queira acrescentar

Av. Colombo 5790, Maringá-PR CEP 87020-900
Tel: (044) 3011-4196/3011-5833 Fax: (044) 3011-4196