

**INSTITUTO BRASILEIRO DE TECNOLOGIA AVANÇADA
ESPECIALIZAÇÃO EM ENGENHARIA DE SOFTWARE**

NELSON RODRIGO LOMBARDI BASSETTO

**Uma abordagem para um programa de medição baseado
em análise de pontos de função**

São Paulo – SP
2008

NELSON RODRIGO LOMBARDI BASSETTO

**Uma abordagem para um programa de medição baseado
em análise de pontos de função**

Trabalho de conclusão de curso apresentado como requisito final para obtenção do título de Especialista em Engenharia de Software, pelo Instituto Brasileiro de Tecnologia Avançada – IBTA, na área de concentração de Tecnologia da Informação.

Orientador: Prof. Luiz David Szilagy

Banca Examinadora - Avaliador(es):

Luiz David Szilagy
Especialista, Instituto Mackenzie

30 de outubro de 2008

Bassetto, Nelson Rodrigo Lombardi

B319a Uma abordagem para um programa de medição baseado em análise de pontos de função [manuscrito] Nelson Rodrigo Lombardi Bassetto.

096f., enc.

Orientador: Luiz David Szilagyi

Monografia – Faculdade de Tecnologia IBTA

Bibliografia: f. 113-117

1. Tecnologia da informação, Brasil I. TÍTULO II. Bassetto, Nelson Rodrigo Lombardi

005.1068

À minha mãe e avós

AGRADECIMENTOS

A Deus pelas oportunidades concedidas e caminhos que me agregam e me fazem
evoluir a cada dia;

À minha mãe e avós pelo incentivo e apoio nos estudos;

À minha família, por ter provido bases sólidas que me estruturam e servem-me de
guia todos os dias;

Aos familiares e amigos próximos, pela tolerância nos momentos de ausência;

Ao colega Renato Duran Vittoruzzo Martins pelo competente apoio técnico e
pertinentes sugestões;

À Lucy Rizzo, professora de inglês de extrema competência, por sua valiosa revisão
e sugestões quanto ao *abstract* deste trabalho;

Ao IBTA, pela oportunidade e pelo curso de alto nível ministrado, o qual agregou
significativamente em minha formação.

RESUMO

Visando proporcionar um meio que apóie no desenvolvimento de soluções sistêmicas com alto nível de qualidade, produzidas em menor tempo e custo possíveis, este estudo propõe uma abordagem de programa de medição baseado em análise de pontos de função. Tal abordagem objetiva a obtenção sistemática de dados significativos sobre a organização de *software* que o implementa, possibilitando análises, identificação de pontos positivos e falhos da organização e tomadas de decisão fundamentadas por dados quantitativos e concretos. Com o estudo realizado, verificou-se que programas de medição baseados em análise de pontos de função podem ajudar organizações a gerenciar e tomar decisões baseadas em fatos, a ter maior previsibilidade de entrega de produtos, a melhorar seu processo de desenvolvimento e seus produtos e a utilizar recursos com maior eficiência.

Palavras chave: Análise de Pontos de Função, Medição Funcional, Programa de Medição, Métricas e Medição de *Software*.

ABSTRACT

Aiming to provide an environment that supports the development of high quality systemic solutions, produced with less time and cost, this study proposes an approach to a measurement program based on function points analysis. This approach intends to systematically achieve significant data of the software organization which implements it, allowing analysis, identifying the organization's strengths and weaknesses and decision-making taking into account real and quantitative data. In this study, it was verified that measurement programs which rely on function points analysis can help organizations to manage and make decisions based on facts, to have greater predictability of products delivery, to improve its development process and its products and to use resources more efficiently.

Key Words: *Function Point Analysis, Functional Measurement, Measurement Program, Metrics and Software Measurement.*

LISTA DE ABREVIações, SIGLAS E SÍMBOLOS

AIE	Arquivo Interface Externa
ALI	Arquivo Lógico Interno
APF	Análise de Pontos de Função
<i>BFPUG</i>	<i>Brazilian Function Point Users Group</i> [Grupo Brasileiro de Usuários de Pontos de Função]
CE	Consulta Externa
CGS	Características Gerais do Sistema
CMM	<i>Capability Maturity Model</i> [Modelo de Capacidade e Maturidade]
CMMI	<i>Capability Maturity Model Integration</i> [Modelo de Capacidade, Maturidade e Integração]
EE	Entrada Externa
<i>FPA</i>	<i>Function Point Analysis</i> [Verificar APF]
<i>IFPUG</i>	<i>International Function Point Users Group</i> [Grupo Internacional de Usuários de Pontos de Função]
<i>MSF</i>	<i>Microsoft Solutions Framework</i> [<i>Framework</i> de Soluções Microsoft]
<i>PCs</i>	<i>Personal Computers</i> [Computadores Pessoais]
PF	Pontos de Função
<i>RUP</i>	<i>Rational Unified Process</i> [Processo Unificado Rational]
SE	Saída externa
TI	Tecnologia da informação
UML	<i>Unified Modeling Language</i> [Linguagem de modelagem unificada]
XP	<i>Extreme Programming</i>

LISTA DE ILUSTRAÇÕES

FIGURA 1: PROCESSO DE CONTAGEM DE PONTOS DE FUNÇÃO, ADAPTADO DE (IFPUG, 2000), PÁGINA 31	1
FIGURA 2 - PROCESSO CÍCLICO DE UM PROGRAMA DE MEDIÇÃO, ADAPTADO DE (DEKKERS, 2002), PÁGINA 163	31
FIGURA 3 - EXEMPLO DE RELATÓRIO COMPARATIVO DE MEDIÇÕES, ADAPTADO DE (RUSSAC, 2002), PÁGINA 155.....	38
FIGURA 4 - EXEMPLO DE RELATÓRIO DE MÉTRICAS ORGANIZACIONAL, ADAPTADO DE (RUSSAC, 2002), PÁGINA 156.....	40
FIGURA 5 - DEFINIÇÃO PROGRAMA MEDIÇÃO - FORMATO RELATÓRIO DETALHADO POR PROJETO.....	69
FIGURA 6 - DEFINIÇÃO PROGRAMA MEDIÇÃO - FORMATO RELATÓRIO COMPARATIVO POR PROJETO.....	71
FIGURA 7 - DEFINIÇÃO PROGRAMA MEDIÇÃO - FORMATO RELATÓRIO EVOLUCIONAL DA ORGANIZAÇÃO	73
FIGURA 8 - DEFINIÇÃO DO PROGRAMA DE MEDIÇÃO - INTEGRAÇÃO DO PROGRAMA EM UM CONTEXTO.....	1
FIGURA 9 - DEFINIÇÃO DO PROGRAMA DE MEDIÇÃO - FLUXO PARA GERAÇÃO BIMESTRAL DO RELATÓRIO EVOLUCIONAL DA ORGANIZAÇÃO.....	77

LISTA DE TABELAS

TABELA 1: FASES DO CICLO DE VIDA DE DESENVOLVIMENTO EM QUE A APF PODE SER APLICADA, ADAPTADO DE (VAZQUEZ, ET AL., 2003)	16
TABELA 2: MATRIZ COMPLEXIDADE ALI / AIE, ADAPTADO DE (IFPUG, 2000), PÁGINA 69.....	21
TABELA 3: MATRIZ DE COMPLEXIDADE EE, ADAPTADO DE (IFPUG, 2000), PÁGINA 165.....	23
TABELA 4: MATRIZ DE COMPLEXIDADE SE / CE, ADAPTADO DE (IFPUG, 2000), PÁGINA 166 ..	23
TABELA 5: MATRIZ TIPO FUNÇÃO - CONTRIBUIÇÃO EM PF, ADAPTADO DE (IFPUG, 2000), PÁGINAS 69 E 167.....	24
TABELA 6: EXEMPLO DE CONTAGEM DE PONTOS DE FUNÇÃO NÃO AJUSTADOS	25
TABELA 7 - RELAÇÃO OBJETIVOS X MEDIDAS, ADAPTADO DE (HOLMES, 2002), PÁGINA 100	33
TABELA 8 - APLICABILIDADE DE MEDIDAS PARA ENDEREÇAMENTO DE OBJETIVOS, ADAPTADO DE (DEKKERS, 2002), PÁGINA 165.....	34
TABELA 9 - RELAÇÃO OBJETIVOS, MEDIÇÕES, DADOS E RESPONSABILIDADES, ADAPTADO DE (HOLMES, 2002), PÁGINAS 103 E 104.....	36
TABELA 10 - EXEMPLOS DE TREINAMENTOS PARA DIFERENTES AUDIÊNCIAS, ADAPTADO DE (HOLMES, 2002), PÁGINA 109.....	44
TABELA 11 - RELAÇÃO DE MÉTRICAS X OBJETIVOS COM O PROGRAMA DE MEDIÇÃO	55
TABELA 12 - APURAÇÃO DO ESFORÇO GASTO NO PROJETO EM HORAS - SUMARIZAÇÃO ...	57
TABELA 13 - APURAÇÃO DO CUSTO TOTAL DO PROJETO - SUMARIZAÇÃO	59
TABELA 14 – TAMANHO DO PROJETO INICIAL EM PONTOS DE FUNÇÃO - SUMARIZAÇÃO	60
TABELA 15 - TOTAL DE PONTOS DE FUNÇÃO DE ALTERAÇÕES DE ESCOPO - SUMARIZAÇÃO	62
TABELA 16 - TOTAL DE PONTOS DE FUNÇÃO AO TÉRMINO DO PROJETO - SUMARIZAÇÃO...	63
TABELA 17 – QUANTIDADE TOTAL DE DEFEITOS ENCONTRADOS - SUMARIZAÇÃO	65

TABELA 18 – DEFINIÇÃO DO PROGRAMA DE MEDIÇÃO - RELAÇÃO MEDIDAS, DADOS, MEIOS, PONTOS E RESPONSABILIDADE PELA COLETA	66
TABELA 19 – DEFINIÇÃO DO PROGRAMA DE MEDIÇÃO – RELAÇÃO OBJETIVOS, MÉTRICAS E SUAS COMPOSIÇÕES	66
TABELA 20 - SUMARIZAÇÃO RELATÓRIO DETALHADO POR PROJETO	70
TABELA 21 - SUMARIZAÇÃO RELATÓRIO COMPARATIVO POR PROJETO	72
TABELA 22 - SUMARIZAÇÃO RELATÓRIO EVOLUCIONAL DA ORGANIZAÇÃO	74
TABELA 23 - DEFINIÇÃO PROGRAMA MEDIÇÃO - ANÁLISE E REPORTE DE RESULTADOS	75

SUMÁRIO

1. INTRODUÇÃO	1
1.1. CONTEXTUALIZAÇÃO	1
1.2. PROBLEMATIZAÇÃO.....	2
1.3. OBJETIVOS DA PESQUISA	2
1.4. JUSTIFICATIVAS	3
1.5. ENUNCIADO DA HIPÓTESE	4
1.6. MÉTODO DE PESQUISA.....	4
1.7. DELIMITAÇÃO.....	4
1.8. ORGANIZAÇÃO	5
2. MEDIÇÃO DE SOFTWARE.....	6
2.1. INTRODUÇÃO	6
2.2. MEDIDA, MEDIÇÃO, MÉTRICAS E INDICADORES.....	6
2.3. MEDIÇÃO DE <i>SOFTWARE</i>	7
2.4. MÉTRICAS PARA ESTIMATIVA DE TAMANHO DE <i>SOFTWARE</i>	10
3. PROGRAMA DE MEDIÇÃO	28
3.1. INTRODUÇÃO	28
3.2. OBJETIVOS DE UM PROGRAMA DE MEDIÇÃO	28
3.3. BENEFÍCIOS DA IMPLEMENTAÇÃO DE UM PROGRAMA DE MEDIÇÃO	29
3.4. COMPOSIÇÃO DE UM PROGRAMA DE MEDIÇÃO	30
3.5. FATORES CRÍTICOS DE SUCESSO EM UM PROGRAMA DE MEDIÇÃO	42
3.6. RECURSOS HUMANOS: TREINAMENTO E MUDANÇA CULTURAL.....	44
4. ABORDAGEM PARA UM PROGRAMA DE MEDIÇÃO BASEADO EM ANÁLISE DE PONTOS DE FUNÇÃO	46
4.1. PREMISSAS PARA O PROGRAMA DE MEDIÇÃO ABORDADO	47
4.2. DEFINIÇÃO DE OBJETIVOS E METAS.....	51
4.3. DEFINIÇÃO DE MEDIÇÕES E MÉTRICAS	53

4.4. DADOS, MEIOS E PONTOS DE COLETA E RESPONSABILIDADES	55
4.5. ANÁLISE E REPORTE DE RESULTADOS	67
4.6. INTEGRAÇÃO DO PROGRAMA DE MEDIÇÃO EM UM CONTEXTO DE DESENVOLVIMENTO DE SISTEMAS	75
5. CONCLUSÕES E TRABALHOS FUTUROS	78
5.1. CONCLUSÃO	78
5.2. RELAÇÃO COM OS OBJETIVOS INICIAIS	78
5.3. VALIDADE DA HIPÓTESE	78
5.4. TRABALHOS FUTUROS	79
REFERÊNCIAS	81

1. INTRODUÇÃO

1.1. CONTEXTUALIZAÇÃO

Com a exponencial evolução dos computadores pessoais e em seguida com o surgimento da internet, a área de tecnologia da informação passou cada vez mais a dar suporte aos negócios de corporações e não se limitando a isso, tornou-se componente chave para se atingir diferencial competitivo. Em alguns ramos de atuação tornou-se essencial até mesmo para garantir a sobrevivência das empresas no mercado. Neste mesmo sentido, Bill Gates (1999 p. 11) cita tal evolução:

Pela primeira vez, todo tipo de informação – números, texto, som, vídeo – pode ser posto numa forma digital que qualquer computador pode armazenar, processar e enviar. Pela primeira vez, um hardware padrão combinado a uma plataforma de *software* padrão está criando economias de escala que disponibilizam a empresas de todos os tamanhos, a custos baixos, poderosas soluções de informática. O termo “pessoal” na expressão “computador pessoal” significa que cada profissional do conhecimento dispõe de uma forte ferramenta para analisar e usar a informação fornecida por essas soluções. A revolução do microprocessador não só está dando aos PCs um aumento exponencial de poder, como está propiciando a criação de toda uma nova geração de dispositivos digitais inteligentes – handhelds [micros de mão], Auto PCs, smart cards [cartões inteligentes] e outros a caminho – que irão disseminar o uso da informação digital. Uma das chaves para essa disseminação é o aperfeiçoamento das tecnologias da internet, que nos permitem conectividade mundial.

Devido a esta evolução constante e a necessidade crescente de *softwares* para o apoio aos negócios, diversas tecnologias, metodologias e padrões são desenvolvidas e estudadas. Processos, linguagens de programação, conceitos, modelos arquiteturais e metodologias, são aperfeiçoados constantemente a fim de se conseguir atender a esta demanda com maior qualidade do resultado final e com o menor tempo e custo possível, a ponto de se manter o custo-benefício da solução sistêmica adquirida. As empresas que conseguem serviços de tecnologia da informação com estas características podem obter vantagens competitivas, que podem inclusive definir sua permanência ou não no mercado. Putnam, et al., (2002) cita tal cenário competitivo quando afirma que as organizações precisam cada vez mais conseguir produtos de boa qualidade em tempo, esforço e custo limitados, o que significa, em sua opinião, ter um bom nível de produtividade da equipe.

1.2. PROBLEMATIZAÇÃO

Produtividade, porém, por si só não garante produtos de qualidade em esforço e custo limitados. Nem mesmo a utilização das melhores tecnologias, processos, metodologias, linguagens de programação, etc. garantem tal resultado. Diversos fatores estão envolvidos e diversas questões devem ser respondidas, como por exemplo, como saber o quão produtiva foi a equipe no desenvolvimento de um sistema? Como saber se o produto desenvolvido tem níveis de qualidade compatíveis com o mercado do mesmo ramo, a ponto de se constatar se a organização está ou não gastando apenas o valor adequado com retrabalho? Como saber se o processo de desenvolvimento de *software* implementado pela organização tem sido eficiente e tem trazido benefícios que valham o custo investido?

Sem respostas assertivas para tais questões, nenhuma tecnologia ou processo de desenvolvimento, por mais atual e evoluído que seja, garante que os sistemas estão sendo desenvolvidos com qualidade e produtividade satisfatórios, a ponto de apoiarem na geração de diferencial competitivo para a organização.

Faz-se necessário então que se tenha um mecanismo para responder a tais questões de maneira assertiva, que permita não só saber a situação real da organização de *software*, como também a identificação de pontos fortes e fracos, buscando com base em tal análise a evolução constante. Neste mesmo sentido (Holmes, 2002) afirma que organizações de *software* precisam de uma forma para gerenciar a carga de trabalho, permitindo-se decidir o que fazer, como fazer e quando fazer e define ainda que os dias de utilização de “sentimento” como base para tomada de tais decisões e como base também para responder às questões relacionadas à sua posição, estão extintos.

1.3. OBJETIVOS DA PESQUISA

Dado tal contexto e problematização, objetiva-se com esta pesquisa, estudar métricas, medições e, por fim, programas de medição, buscando entender se tais mecanismos podem suprir a necessidade por informações assertivas, se possibilitam a identificação de pontos que devem ser aprimorados no processo de

desenvolvimento de *software* da organização e de ações e iniciativas que tem tido bons resultados e que devem ser mantidas.

Para tal, objetiva-se levantar conteúdo bibliográfico sobre medições e métricas que possibilitem a definição de um programa de medição alinhado a objetivos de negócios, a fim de se conhecer suas características e suas aplicabilidades no contexto de desenvolvimento de *software* atual.

Objetiva-se também estudar e levantar conteúdo bibliográfico a respeito da composição de um programa de medição, possibilitando entendimento das suas aplicabilidades em um processo de desenvolvimento e sua relação com as métricas estudadas.

Por fim, com base nos estudos realizados, objetiva-se estabelecer uma abordagem para um programa de medição passível de implementação por organizações, independentemente do processo de desenvolvimento de sistemas que utilizem.

1.4. JUSTIFICATIVAS

Programas de medição permitem chegar a respostas assertivas para as questões mencionadas anteriormente, o que leva a melhoria no processo de desenvolvimento de *software* e conseqüentemente nos produtos deste processo.

(Minkiewicz, 2002) afirma que cada vez mais, organizações estão se dando conta do valor que a medição em organizações de *software* agrega ao seu processo de negócios, através de estratégia de negócios de longo prazo para medição de performance e melhoria de processos de desenvolvimento de *software*. (Minkiewicz, 2002) afirma ainda que as companhias estão se dando conta também, de que o investimento em entender e implementar um programa de medição bem sucedido é pago pela melhoria na produtividade do desenvolvimento de *software* e do processo de desenvolvimento como um todo.

Tais melhorias vêm a corroborar na busca por produtos com altos níveis de qualidade, sendo produzidos em cada vez menos tempo e custo, ao mesmo tempo sendo cada vez mais aderentes ao negócio da companhia, apoiando a organização a alcançar seus objetivos estratégicos.

1.5. ENUNCIADO DA HIPÓTESE

A hipótese desta pesquisa é que um programa de medição permite a organizações gerenciarem a sua construção de *software* baseando-se em atributos quantitativos medidos ao invés de informações subjetivas ou sentimentos, conseguindo-se com tal:

- Gerenciamento e tomada de decisões baseada em fatos;
- Maior previsibilidade de entrega de produtos;
- Melhorias no processo de desenvolvimento e seus produtos;
- Melhoria na qualidade dos sistemas desenvolvidos;
- Utilização de recursos com maior eficiência;

1.6. MÉTODO DE PESQUISA

A metodologia de pesquisa utilizada neste trabalho é a pesquisa descritiva, onde conforme (Gill, 2007), dentre outras características, está à busca pelo estabelecimento de relação entre variáveis. Neste trabalho, as variáveis podem ser consideradas como as diversas métricas existentes e as diversas formas de estabelecimento de um programa de medição.

A pesquisa bibliográfica é o principal instrumento deste trabalho, onde livros e artigos científicos são analisados (Gill, 2007), com o objetivo de se adquirir embasamento teórico, possibilitando o estabelecimento das relações entre as métricas e o programa de medição.

1.7. DELIMITAÇÃO

Este trabalho está delimitado ao estudo de métricas, programas de medição e a relação entre eles. A customização de processos de desenvolvimento de *software* para a introdução do programa de medição abordado não será discutido neste trabalho, ficando esta como uma oportunidade de evolução do estudo.

Possibilidades de continuidade deste estudo são discutidas em detalhes no último capítulo deste estudo.

1.8. ORGANIZAÇÃO

Este estudo está organizado em cinco capítulos. A seguir, o conteúdo de cada capítulo é sumarizado:

O primeiro capítulo apresenta o contexto em que o trabalho está inserido, problematização, objetivos, justificativas, a hipótese de solução do problema apresentado, metodologia de pesquisa utilizada no trabalho, escopo e organização do trabalho como um todo. Seu objetivo é prover informações introdutórias sobre o assunto, bem como a motivação que levou a sua realização e a sua importância.

O segundo capítulo aborda uma pesquisa bibliográfica sobre métricas, tendo como objetivo a contextualização do assunto, delineando os principais tópicos e conceitos que serão utilizados no desenvolvimento do trabalho.

O terceiro capítulo traz uma pesquisa bibliográfica sobre programas de medição, tendo como objetivo a introdução, conceituação e contextualização do assunto, servindo como referência para o estudo do capítulo posterior.

O quarto capítulo explora o estudo realizado sobre as métricas e programas de medição, estabelecendo-se, com base no levantamento bibliográfico realizado, uma abordagem para um programa de medição.

O quinto e último capítulo traz a conclusão do trabalho, apontando a relação entre os objetivos e o resultado atingido do estudo, a validade da hipótese inicial e possibilidades de evolução deste estudo.

2. MEDIÇÃO DE SOFTWARE

2.1. INTRODUÇÃO

Medição é o ato de determinar uma indicação quantitativa da extensão, quantidade, dimensão, capacidade ou tamanho de algum atributo de um produto ou processo (Pressman, 2006). Medição de *software* compreende a medição de um processo de desenvolvimento de *software*, cujo objetivo é fornecer um conjunto de indicadores de processo, que leva a aperfeiçoamentos no longo prazo (Pressman, 2006) ou de um projeto, cujo objetivo é fornecer mecanismos para o controle do projeto.

Em busca da previsibilidade de entrega, da qualidade do produto de desenvolvimento de *software*, da produtividade da equipe e da eficiência do processo de desenvolvimento, a medição é um item essencial. A medição de *software* propicia a obtenção de medidas quantitativas de um processo ou produto, permitindo-se a análise e avaliação dos valores obtidos, o que possibilita a tomada de ações visando à melhoria de qualidade como um todo. (Pressman, 2006) confirma esta análise quando afirma que a medição é uma forma de apoio à gestão, que, quando conduzida adequadamente, apóia na tomada de decisões que conduzem o projeto ao sucesso.

O objetivo deste capítulo é contextualizar a medição de *software*, apresentar tópicos relacionados a métricas para estimativa de tamanho de *software* e apresentar a análise de pontos por função, métrica utilizada como base do programa de medição abordado por este trabalho. Dentro dos objetivos deste estudo, este capítulo vem a explorar a medição de *software*, no contexto da medição em um projeto, visando prover base teórica para o estabelecimento de uma abordagem de medição para o estabelecimento de um programa de medição.

2.2. MEDIDA, MEDIÇÃO, MÉTRICAS e INDICADORES

Para que se defina medição de *software*, é essencial que cada um dos termos relacionados ao assunto estejam esclarecidos.

(Pressman, 2006) define que uma **medida** fornece uma indicação quantitativa da extensão, quantidade, dimensão capacidade ou tamanho de algum atributo de um produto ou processo. (Vazquez, et al., 2003 p. 31) reforça tal definição, quando afirma que **medida** é “a quantificação de uma característica”.

(IEEE, 1993) define **métrica** como uma medida quantitativa do grau em que um sistema, componente ou processo possui um determinado atributo.

(Pressman, 2006) define **medição** como o ato de determinar uma **medida** e um **indicador** como uma **métrica**, ou combinação de **métricas** que fornece profundidade na visão do processo de *software*, projeto de *software* ou produto em si.

(Pressman, 2006 p. 352) exemplifica cada uma das definições apresentadas:

Quando um único ponto de dados foi coletado (por exemplo, o número de erros descoberto em um único componente de *software*), uma **medida** foi estabelecida. **Medição** ocorre como resultado da coleção de um ou mais pontos de dados (por exemplo, um certo número de revisões de componentes e testes de unidade são investigados para coletar medidas do número de erros em cada um). Uma **métrica** de *software* relaciona as **medidas** individuais de algum modo (por exemplo, o número médio de erros encontrados por revisão ou número médio de erros encontrados por teste de unidade).

[...] Um **indicador** fornece profundidade de visão que permite ao gerente do projeto ou engenheiro de *software* ajustar o processo, projeto ou produto para tornar as coisas melhores.

2.3. MEDIÇÃO DE SOFTWARE

Segundo (Pressman, 2006), a medição é essencial em processos de engenharia. Através da medição, é possível obter entendimento do processo e projeto, tendo-se um mecanismo para avaliação objetiva. Tendências (boas ou más) podem ser detectadas, melhores estimativas podem ser feitas e aperfeiçoamentos reais podem ser obtidos ao longo do tempo. (Budag, 2007) confirma este raciocínio quando diz que o objetivo da aplicação da medição de *software* é fornecer aos gerentes e engenheiros de *software* um conjunto de informações tangíveis para planejar o projeto, realizar estimativas, gerenciar e controlar os projetos com maior precisão.

Segundo (Pressman, 2006) as razões para o qual se mede um *software* são:

- Caracterizar em um esforço a fim de obter entendimento de processos, produtos, recursos e ambientes, e para estabelecer referências, para comparação com futuras avaliações;
- Para avaliar, a fim de determinar o estado em relação aos planos;
- Para prever pela obtenção de entendimento de relacionamento entre processos e produtos e construção de modelos desses relacionamentos;
- Para aperfeiçoar, pela identificação de bloqueios, causas fundamentais, ineficiências e outras oportunidades para melhor a qualidade do produto e desempenho do processo.

Complementando as razões citadas por Pressman, (Vazquez, et al., 2003), entendem que as razões para se medir estão compreendidas em:

- Ter mecanismos para manter sob controle um contexto de desenvolvimento de *software* em que os requisitos tendem a expansão e que os recursos tendem a escassez – Atender ao máximo às expectativas dos clientes com a utilização do mínimo de recursos;
- Apoiar no planejamento e controle da gerência de projetos, apoiando na definição de objetivos realistas e na adoção de medidas necessárias à prevenção e correção dos desvios que causam impacto negativo ao projeto;
- Apoiar na tomada de decisões do projeto, através de indicadores que refletem uma tendência de comportamento futuro;
- Ter insumos para análises *Make-or-Buy [Construir ou Comprar]*, onde se decide entre construir um *software* internamente ou em terceirizar sua construção;
- Ter mecanismos para a medição concisa de serviços de desenvolvimento de *software* prestados por terceiros.

Dados os motivos pelo qual medir, é importante ressaltar que conforme pode ser observado, a medição pode ser aplicada sobre alguns contextos distintos, porém relacionados, como por exemplo, a medição em um processo de desenvolvimento de *software* e a medição em um projeto. Na medição em um processo, o objetivo é

obter métricas e indicadores do processo, visando sua melhoria contínua. Na medição ao longo de um projeto, têm-se métricas para auxílio na estimativa, no controle da qualidade, na avaliação da produtividade e no controle do projeto. Este controle se dá através de medições sistemáticas e avaliação com base em regras claramente definidas (Pressman, 2006). (Vazquez, et al., 2003 p. 22) enfatiza a importância das métricas na medição de projetos:

A falta de métricas de projeto prejudica de forma geral seu acompanhamento, uma vez que, apesar de o problema estar lá, ele não é percebido por aqueles que podem direcionar esforços na sua solução. O papel das métricas é permitir uma rápida identificação e correção de problemas.

Para (Demarco, 1989), os aspectos quantitativos onde as métricas podem ser usadas são no escopo, tamanho, custo, risco e tempo empregado. Uma vez que se sabe o tamanho do que deve ser construído, é possível planejar e controlar o projeto. Além disso, tendo-se uma medida padrão e comum de tamanho, é possível fazer a relação com outras variáveis, como por exemplo, o tempo gasto na construção, a quantidade de erros encontrados por medida de tamanho, conseguindo-se assim, apurar produtividade e indicadores de qualidade.

Segundo (Pressman, 2006), na maioria das vezes, em um projeto de desenvolvimento de um sistema, as métricas de projeto são aplicadas para a apuração de estimativas. Para tal, são usadas métricas para estimativa de tamanho de *software* a partir das quais, estimativas de esforço e tempo são produzidas. Com a evolução do projeto, o gerente de projetos utiliza a estimativa de esforço e tempo para controlar e monitorar o progresso da construção. Para tal, a taxa de produção, representada em termos de modelos criados, os erros descobertos durante cada tarefa de engenharia de *software* são medidos e registrados. Ao final, consegue-se comparar se as estimativas iniciais estão aderentes ao realizado, e esta informação é retro-alimentada no processo de estimativa. Consegue-se também saber, o quanto a equipe foi produtiva e o quanto foi efetiva com relação à qualidade.

Uma vez que este processo de medição é definido e utilizado, é possível identificar áreas falhas em projetos e, com o tempo, tomar ações visando sua melhoria. À medida que os pontos falhos são aperfeiçoados, (Pressman, 2006) os defeitos são minimizados e, conforme a contagem de defeitos decresce, a

quantidade de retrabalho durante o projeto é também reduzida, levando a diminuição do custo total do projeto.

Contextualizada a medição de *software*, as medições ao longo de um projeto, e estimativas de tamanho, a seguir são apresentadas métricas para estimativa de tamanho de *software*, as quais fazem parte das medições em projetos, apoiando na apuração de estimativas de esforço, em indicadores para controle de qualidade, produtividade e controle do andamento do projeto.

2.4. MÉTRICAS PARA ESTIMATIVA DE TAMANHO DE SOFTWARE

Segundo (Vazquez, et al., 2003), sempre que se tem uma necessidade de implementação de um *software* para se atender demandas de uma organização, as principais questões que são feitas são quanto ao tempo necessário para conclusão do projeto e o custo para a organização. Para responder tais perguntas, diversos fatores devem ser considerados, como as particularidades dos requisitos do projeto de *software*, da equipe envolvida e da tecnologia empregada. (Vazquez, et al., 2003 p. 155) cita os principais fatores que, quando analisados, evidenciam as dificuldades de obtenção de respostas confiáveis para as questões citadas:

Os requisitos traduzem com fidelidade as necessidades do negócio dos usuários? Já se encontram suficientemente estabilizados? Refletem características de *software* transacionais de baixa complexidade ou possuem atributos que exigem conhecimentos específicos, tais como alta performance, matemática complexa, processamentos distribuídos, etc.?

A equipe de desenvolvimento possui conhecimento na área de negócio que será atendida pelo projeto de *software*? Possui experiência na utilização das ferramentas necessárias à conclusão do projeto? Possui todos os perfis necessários impostos pelas características do projeto? Existem conflitos internos à equipe que precisam ser solucionados? É possível identificar uma liderança entre os integrantes da equipe? Qual o grau de motivação da equipe mediante o projeto?

A tecnologia já faz parte da cultura da organização? Pode ser facilmente absorvida por novos integrantes da equipe de projeto? Existe material de apoio suficiente para o aprendizado da tecnologia? Sua aplicação exige pessoal com algum tipo de especialização? Suporta a implementação de todas as características do *software*? Atende inclusive aos requisitos não-funcionais do projeto?

Dadas estas particularidades inerentes a cada projeto, apurar com exatidão o custo final e a data da conclusão é uma tarefa que só pode ser realizada quando o projeto está concluído. Antes de tal fato, só é possível obter estimativas, o que não deixa de se ser muito importante, uma vez que dados históricos sejam mantidos e utilizados para o aperfeiçoamento das estimativas, chegando a um nível de confiança cada vez maior (Vazquez, et al., 2003).

Para a apuração de tais estimativas, métricas podem ser utilizadas, dando uma abordagem sistemática ao processo, e, portanto aumentando a confiabilidade no resultado. Segundo (Aguilar), as métricas para estimativa de tamanho de *software* surgiram com o objetivo de servir como mecanismo para se estimar o esforço envolvido em projetos e o prazo associado ao desenvolvimento de sistemas.

Uma das primeiras técnicas que surgiram com este intuito foi a SLOC (*Source Lines of Code*), por volta de 1950/1960, a qual é considerada como uma medição física do tamanho do *software*, por medir o volume de código-fonte contido no mesmo. Para (Andrade, 2004), os principais benefícios desta técnica estão no fato de que ela pode ser aplicada automaticamente através da utilização de ferramentas que apurem a quantidade de linhas de código do projeto e pelo fato de ter sido usada por muito tempo, existindo grande quantidade de dados históricos de estimativas realizadas. Por outro lado, o principal problema desta técnica está no fato de que a medição do código fonte não representa propriamente o tamanho do *software* ou do esforço envolvido, uma vez que este atributo está diretamente condicionado à tecnologia utilizada e ao método de programação empregado. (Andrade, 2004) reafirma esta conclusão, quando diz que a contagem em LOC depende do grau de reuso da linguagem de programação, o que pode levar uma estimativa a ser cinco vezes superior a outra, devido às diferenças das linguagens de programação quando as linhas em branco, linhas de comentário, declaração de dados e linhas de instruções. (Andrade, 2004) afirma ainda que esta técnica penaliza projetos pequenos e bem projetados, não se adapta às linguagens não procedimentais e é de difícil obtenção na fase inicial do projeto.

Dadas as limitações da medição física, técnicas de medição baseadas em funcionalidades foram propostas, as quais procuram medir a funcionalidade disponibilizada pelo *software*, ao invés do tamanho físico. Tais técnicas são denominadas métricas funcionais de tamanho. Por medirem apenas o tamanho

funcional do sistema, é possível serem utilizadas em fases iniciais do projeto, quando normalmente ainda não se tem muitos detalhes técnicos.

Uma das primeiras técnicas criadas com este objetivo foi a Pontos de Função, criada por Allan Albrecht em 1979. Na técnica de Pontos de Função os dados e as transações envolvidas no sistema são classificadas quanto a sua complexidade e pontuadas. Um fator de ajuste é determinado com base em características específicas do sistema medido e aplicado à quantidade de pontos apurados. Ao término, tem-se a quantidade de pontos por função do sistema medido. Segundo (Aguiar), tal técnica ganhou grande popularidade com a criação do IFPUG (*International Function Point Users Group*) e com sua normatização internacional, através da norma ISSO/IEC 20926. (Aguiar) ainda afirma que esta técnica tem sido utilizada por diversas organizações com sucesso, pois o pré-requisito para sua utilização é o conhecimento dos requisitos do sistema e o da técnica propriamente dita.

Posteriormente, outras métricas funcionais de tamanho foram propostas, como *Bang*, *Mark II*, *Full Function Points* e *Cosmic-FFP*. Em 1993, Gustav Karner criou uma variação dos Pontos de Função específica para a medição de funcionalidade contida em casos de uso, denominada Pontos por Caso de Uso (Aguiar). Nesta técnica, diagramas de caso de uso são a base para a medição. Os atores e casos de uso são classificados e pontuados quanto a sua complexidade no sistema. Fatores técnicos e ambientais são apurados e aplicados aos pontos obtidos. Como resultado, tem-se a quantidade de pontos de casos de uso do sistema e a quantidade de esforço envolvido na construção de tais casos de uso.

(Aguiar) diz que o problema desta técnica reside no fato de que como não se tem padrões universais para a construção de casos de uso, a obtenção de estimativas confiáveis de esforço exigiria a padronização dos estilos de casos de uso e um extenso trabalho de calibração de um modelo de estimativas baseado em pontos por caso de uso. Além disso, a técnica só pode ser utilizada por empresas que adotem casos de uso como forma de expressão dos requisitos, o que inviabilizaria sua utilização em muitos casos e dificultaria sua utilização como métrica padrão para definição de contratos de serviços de desenvolvimento de *software* e medição de produto entregue.

Dadas as diversas métricas apresentadas, é preciso avaliar seus pontos fortes e fracos e sua aplicabilidade no contexto de desenvolvimento de sistemas atual, uma

vez que muitas delas foram desenvolvidas há mais de trinta anos, quando o contexto era completamente diferente. Devem ser levadas em consideração suas aplicabilidades durante as fases de projeto, sua difusão de utilização entre as organizações, sua facilidade de aplicação, a existência de institutos regulamentadores ou normas que regulamentem a utilização, dentre outros fatores. (Schuster, 2006), relaciona alguns itens que devem ser levados em consideração na escolha de uma métrica:

- Deve ter detalhamento dos passos a serem seguidos para a utilização da métrica;
- Deve prover resultados consistentes;
- Deve ser de fácil entendimento;
- Precisa ser objetiva, minimizando a influência de julgamentos pessoais;
- Deve ser efetiva no custo, ou seja, o resultado obtido deve compensar e exceder o esforço da utilização da métrica;
- Precisa fornecer informações que evidenciem a situação atual e sejam relevantes para a tomada de decisão;
- Deve ser apropriada para cada etapa do ciclo de vida do *software*;
- Deve servir para a realização de estimativas;
- Deve possibilitar a obtenção de séries históricas;

A análise de pontos de função atende a todos os requisitos mencionados. Segundo (Andrade, 2004) a técnica é a mais utilizada no mercado. (Aguiar) afirma que nenhuma outra técnica de medição funcional alcançou tal nível de disseminação ou investimento. Capers Jones afirma em (Jones, 2002) que a APF está rapidamente se tornando uma métrica dominante no mundo do *software*. (Aguiar) ainda lista algumas razões pela qual utilizar Pontos de Função:

- A técnica é mantida por uma organização internacional sem fins lucrativos desde 1986, o *International Function Point Users Group* – IFPUG;
- Possui suporte no Brasil através do *Brazilian Function Point Users Group* – BFPUG;
- Existe um programa de certificação mantido pelo IFPUG, com o objetivo de certificar profissionais na técnica, o que vem a aumentar sua credibilidade, uma vez que profissionais possam aplicá-la de forma normatizada;

- É padronizada internacionalmente pela ISO, através da norma ISO/TEC 20926, possibilitando uniformidade da aplicação;
- Utiliza como base requisitos em alto nível de abstração, sendo independente de artefatos, podendo ser utilizado por organizações que utilizem qualquer forma de representação de requisitos;
- A existência de grande acervo de dados sobre pontos de função armazenados por diversas organizações possibilita a realização de estudos e comparações;
- A utilização de pontos de função em contratos e licitações é uma realidade no Brasil, tendo surgido a partir de iniciativas de organizações governamentais e rapidamente alcançado o mercado em geral;

Dadas tais motivações e o contexto do trabalho, o tópico seguinte detalha a técnica de Pontos de Função, com o objetivo de explorá-la, a fim de se obter fundamentação teórica para definição da sua aplicação em um programa de medição.

2.4.1. Análise de Pontos de Função

2.4.1.1. A Análise de Pontos de Função

A Análise de Pontos de Função [APF], do inglês *Function Point Analysis* [FPA], é um método padrão para medição do desenvolvimento de *software* do ponto de vista¹ do usuário² (IFPUG, 2000). A APF mede o *software* quantificando a funcionalidade que ele provê ao usuário, baseado em um design lógico inicial (IFPUG, 2000).

¹ **Visão do Usuário:** é qualquer descrição formal de suas necessidades de negócios em sua própria linguagem. A visão do usuário pode variar em sua forma física, podendo estar representada das seguintes formas: Catalogo de transações, documento de requisitos, especificações externas, especificações detalhadas, manual do usuário, etc. (IFPUG, 2000).

² **Usuário para a APF:** qualquer pessoa que especifique requisitos funcionais ou qualquer pessoa ou coisa que interaja com o sistema. Exemplos: Ator em um caso de uso, uma aplicação, um gestor ou um usuário final (Vazquez, et al., 2003).

Sendo assim, a técnica tem como objetivo medir os produtos entregues, através da medição do que será construído. Como a construção será feita, quais processos, metodologias, tecnologias, ferramentas, linguagens de programação, etc. não são questões levadas em consideração na análise de pontos de função. Tal fato faz com que a técnica seja independente de tecnologia, possibilitando sua utilização consistente por diversas organizações (Vazquez, et al., 2003). Em resumo, os objetivos da análise de pontos de função são (IFPUG, 2000):

- Medir a funcionalidade que o usuário solicita e recebe
- Medir desenvolvimento e manutenção de *software* independentemente de tecnologia de implementação
- Ser uma medida consistente entre vários projetos e organizações
- Ser simples o suficiente para eliminar o esforço adicional com a medição

Dentre os benefícios da análise de pontos de função o principal é a possibilidade da sua utilização como um meio para se estimar custos e recursos necessários para o desenvolvimento e manutenção de *software*. Além disso, a APF pode ser utilizada como meio para medição do tamanho de pacotes, como fator de normalização para a comparação de *software*. A APF também pode apoiar na gerência de escopo de projetos de *software*, pois, uma vez que se tenha o tamanho do *software* medido, é possível medi-lo novamente a cada alteração de escopo, conseguindo-se quantificá-las. A APF também pode ser utilizada na gerência de contratos de *software*, servindo como meio para quantificação de trabalho realizado, ou a realizar, permitindo a precificação sobre uma medida comum e entendida por ambas as partes. A APF também apóia na mensuração de indicadores de qualidade e de produtividade, pois, uma vez que se tenha estabelecido o tamanho do produto, que se saiba a quantidade de tempo empregado para a construção deste produto e a quantidade de defeitos encontrados neste produto, é possível apurar tais indicadores. Sem o tamanho do produto estabelecido, tais indicadores não poderiam ser encontrados com facilidade e precisão. O (IFPUG, 2000) confirma tais benefícios quando lista as principais formas de utilização da APF pelas organizações:

- Uma ferramenta para determinar o tamanho de um pacote de aplicação adquirido, pois possibilita a contagem das funções contidas neste pacote;
- Uma ferramenta para ajudar usuários a determinar os benefícios de uma aplicação para sua organização, pela contagem de funções que atendem a seus requisitos;
- Uma ferramenta para medir unidades de um produto de *software*, suportando análise quanto à qualidade e produtividade;
- Um meio para estimar custo e recursos requeridos para um desenvolvimento de *software* ou manutenção;
- Um fator de normalização de *software* para comparações.

Segundo (Vazquez, et al., 2003), a APF pode ser aplicada na maioria fases do ciclo de vida de um *software*, entretanto, não é aplicável em todas, como por exemplo, manutenções corretivas e manutenções perfectivas. Tal fato se dá, pois em tais fases não se tem a visão do usuário, uma vez que ele normalmente não sabe o motivo da ocorrência do erro corrigido na manutenção corretiva nem o que melhorar tecnicamente na manutenção perfectiva. (Vazquez, et al., 2003) afirma ainda que em algumas fases, dada a pouca quantidade de informações, a APF pode ser utilizada apenas no contexto de estimativa. A tabela 1 a seguir resume as fases, quando a APF pode ser aplicada ou não, e, sendo aplicável, se o resultado obtido pode ser considerado como uma estimativa ou medição.

Tabela 1: Fases do ciclo de vida de desenvolvimento em que a APF pode ser aplicada, adaptado de (Vazquez, et al., 2003)

Fase no ciclo de Vida	Estimativa	Medição
Proposta	✓	
Requisitos	✓	✓
Projeto	✓	✓
Construção	✓	✓
Implantação	✓	✓
Manutenção adaptativa	✓	✓
Manutenção corretiva		
Manutenção perfectiva		

Dado que a APF mede os requisitos do usuário em sua visão, é importante ressaltar quais tipos de requisitos que a APF se propõe a medir. A norma ISSO/IEC 14.143-1 define que existem três tipos de requisitos do usuário: Funcionais, de Qualidade e Técnicos. Os requisitos funcionais compreendem procedimentos que o *software* deve executar para atender às necessidades do usuário. Os requisitos de qualidade compreendem, conforme a norma ISO/IEC 9126, requisitos de funcionalidade, confiabilidade, usabilidade, eficiência, manutenibilidade e portabilidade. Já os requisitos técnicos compreendem a tecnologia para desenvolvimento, manutenção e suporte a execução do *software*, como por exemplo: linguagens, ferramentas de teste, OS, BD e tecnologia de interface com o usuário. Segundo (Vazquez, et al., 2003), dados tais tipos de requisitos, o único que a APF se propõe a medir são os requisitos funcionais. (Vazquez, et al., 2003) afirma ainda que para a APF, os requisitos funcionais são classificados em dois tipos: Interação do usuário com o sistema (requisitos de transações) e armazenamento dos dados manipulados em transações (requisitos de armazenamento).

2.4.1.2. Processo de contagem

O processo de contagem de pontos de função pode ser representado pela Figura 1 a seguir. Cada caixa da figura representa um passo no processo de contagem, que é executado seqüencialmente, de acordo com a ordem estabelecida. O resultado do processo é a aferição do número de pontos de função ajustados. A seguir, cada passo do processo será explicado detalhadamente.

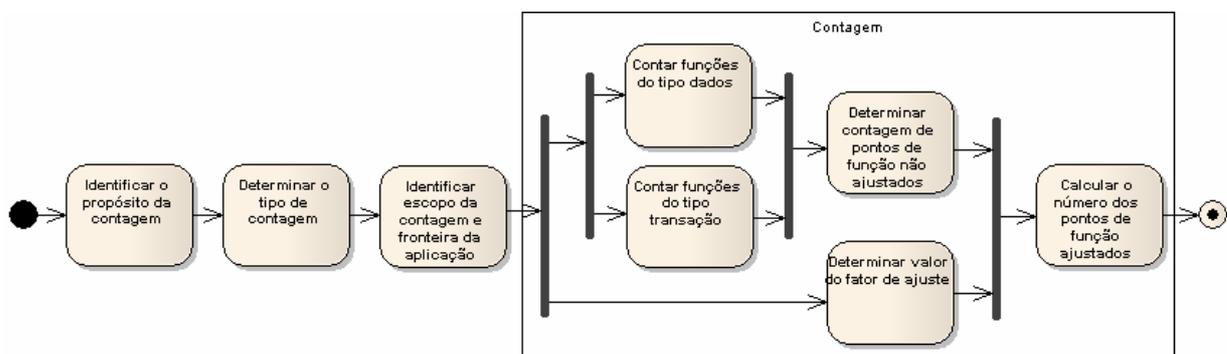


Figura 1: Processo de contagem de pontos de função, adaptado de (IFPUG, 2000), página 31

Identificar o propósito da contagem

Como o próprio nome diz, neste passo, determina-se o propósito da contagem. O propósito compreende o objetivo que se tem com a contagem, que pode variar de acordo com o contexto em que a medição está inserida. O propósito da contagem influencia na definição do tipo, escopo e fronteira da contagem (IFPUG, 2000). Se a APF está sendo aplicada para se ter uma estimativa de esforço com base em uma proposta de desenvolvimento de *software*, o objetivo da contagem é apurar o total de pontos de função estimados. Se a APF está sendo aplicada para contar uma aplicação instalada, visando verificar se o estimado está de acordo com o realizado, o propósito da contagem é apurar o tamanho do *software* em pontos de função. Uma vez que se tem o propósito identificado, sabe-se o resultado esperado da aplicação da técnica.

Determinar o tipo da contagem

A análise de pontos de função pode ser aplicada para medir três tipos de projetos: Projeto de desenvolvimento, Projeto de melhoria e Aplicação Base. No projeto de desenvolvimento, medem-se as funções fornecidas na primeira instalação do *software*, quando o projeto está completo [criação de um novo *software*]. No projeto de melhoria, medem-se modificações em aplicações já existentes, entregues quando o projeto está completo [manutenção evolutiva]. Na aplicação base, medem-se as funções fornecidas ao usuário, da aplicação atualmente instalada [medição de um projeto inteiro e funcional] (IFPUG, 2000).

Determinar a fronteira da aplicação

A fronteira da aplicação é uma interface conceitual entre aplicação interna e o mundo externo da aplicação, independente de considerações técnicas ou implementação, de acordo com a visão externa de negócio do usuário da aplicação (IFPUG, 2000). A definição da fronteira da aplicação é muito importante no processo de contagem, pois, uma vez tendo-a definida, pode-se saber o que deve ser contado como sendo interno à aplicação e o que deve ou não ser contado como sendo externo à aplicação. Para a APF, os itens que compõem a parte interna de uma aplicação têm pesos diferentes dos itens externos.

É importante ressaltar, que para a delimitação da fronteira da aplicação, deve ser levada em consideração a visão do usuário. Aspectos técnicos não devem ser levados em consideração na delimitação. Uma fronteira mal posicionada pode alterar a perspectiva da contagem de uma visão lógica para uma visão física, o que pode ter como consequência a duplicidade na identificação de funções, ou sua omissão (Vazquez, et al., 2003).

Determinar o escopo da contagem

A determinação do escopo da contagem consiste na identificação da funcionalidade que será contada no processo de contagem (IFPUG, 2000). Define-se quais as aplicações envolvidas, quais funcionalidades e tipos de funcionalidades serão contados. O escopo da contagem é determinado pela identificação do propósito da contagem. Uma vez que se tenha definido qual o propósito da contagem, sabe-se o que está ou não no escopo. O escopo pode abranger a contagem de uma aplicação inteira ou parte dela, pode abranger a contagem de várias aplicações, ou, por exemplo, de um projeto que envolve a criação de uma nova aplicação e a manutenção de outras que terão integração com a nova aplicação a ser construída.

Contar funções do tipo dados

O processo de contagem das funções do tipo dados consiste em identificar os requisitos de armazenamento de dados sob a ótica do usuário, de acordo com a sua lógica, e classificá-los quanto ao seu tipo e complexidade, determinando sua contribuição individual em pontos de função (IFPUG, 2000). Por exemplo: em um sistema de emissão de notas fiscais, o usuário identifica que são entidades que devem ser mantidas pelo sistema clientes e nota Fiscal. Uma vez identificadas tais entidades, o passo seguinte é a sua classificação.

É importante ressaltar, que as funções do tipo dados devem ser identificáveis pelo usuário, ou seja, deve ser de seu conhecimento e reconhecimento. Não são consideradas funções do tipo dado, por serem meramente questões técnicas de implementação os seguintes itens: Visões de bancos de dados, armazenamento visando maior performance, dados de domínio com intenção única de manter integridade referencial no banco de dados, dados de configuração da aplicação que não estejam relacionadas ao negócio ou que não sejam reconhecidas pelo usuário

(IFPUG, 2000). Tal restrição é dada para que não se tenha distorções entre as contagens em fases iniciais, onde não se tem detalhamentos técnicos e as fases finais, onde toda a aplicação e solução técnica já foi definida.

As funções do tipo dado podem ser classificadas de duas formas (IFPUG, 2000):

- **ALI – Arquivos lógicos internos:** Armazenamento de dados, identificável pelo usuário, mantido dentro da fronteira da aplicação contada. Exemplos:
 - Tabelas que armazenam dados da aplicação
 - Entidades de negócio
 - Entidades de referência
 - Arquivos mantidos não só pela aplicação, mas também por outra aplicação
- **AIE – Arquivo de interface externa:** Armazenamento de dados, identificável pelo usuário, mantido fora da fronteira da aplicação contada, porém, utilizado pela aplicação contada em algum momento. Exemplos:
 - Dados externos utilizados pela aplicação

É importante ressaltar que o termo “arquivo” utilizado na nomenclatura das funções do tipo dado não se refere a um arquivo físico de um sistema e sim a um agrupamento lógico de dados (IFPUG, 2000). Uma vez classificados quanto ao tipo, o próximo passo é a determinação da complexidade. Para tal, para cada arquivo lógico identificado, determina-se a quantidade de tipos de dados e tipos de registros que o arquivo lógico compreende.

Um tipo de dado é um campo único, reconhecido pelo usuário e não repetido (IFPUG, 2000). Um exemplo de tipo de dado do arquivo lógico nota fiscal é seu valor total. Já um tipo de registro é um subgrupo de dados dentro de um arquivo lógico, reconhecido pelo usuário (IFPUG, 2000), como por exemplo, os itens da nota fiscal.

Uma vez apurado o arquivo lógico, a quantidade de tipos de dados e tipos de registros compreendidos, o passo seguinte é classificá-lo quanto a sua complexidade. Para tal, a tabela 2 a seguir deve ser considerada.

Tabela 2: Matriz complexidade ALI / AIE, adaptado de (IFPUG, 2000), página 69

ALI / AIE	TR \ TD	< 20	20 - 50	> 50
1		Baixa	Baixa	Média
2 -5		Baixa	Média	Alta
> 5		Média	Alta	Alta

TR – Tipo de Registro / TD – Tipo de Dado

Contar funções do tipo transação

Para ter entendimento quanto ao processo de contagem das funções do tipo transação, é necessário que se tenha esclarecido antes um conceito chave da APF, que é o processo elementar. O processo elementar é a menor unidade de atividade com significado para o usuário, completo em si mesmo, que deixa a aplicação em estado consistente (IFPUG, 2000). Em outras palavras, um processo elementar pode ser entendido como uma funcionalidade completa do sistema que o usuário enxerga, por exemplo: Cadastrar clientes, emitir nota fiscal, etc.

O processo de contagem das funções do tipo transação consiste em identificar os processos elementares da aplicação que está sendo contada, identificar a principal intenção de cada processo elementar encontrado, classificá-los quanto ao tipo e quanto a complexidade, determinando sua contribuição individual em pontos de função (Vazquez, et al., 2003).

Uma vez identificados os processos elementares, o passo seguinte é identificar sua principal intenção, classificando-o. Os processos elementares podem ser classificados em:

- **Entrada Externa:** Processo elementar que processa dados ou informações de controle³ vindos de fora da fronteira da aplicação, mantendo um ou mais ALI (IFPUG, 2000). Exemplos (Vazquez, et al., 2003):
 - Transações que recebem dados externos utilizados na manutenção de ALI's;

³ **Informações de Controle:** Influencia processos elementares da aplicação. Especifica o que, quando ou como os dados são processados (IFPUG, 2000).

- Operações de inclusão, exclusão e alteração de registros em arquivos;
- Processamento em lotes de atualização de bases cadastrais a partir de arquivos de movimento;
- **Consulta Externa:** Processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação. Nas consultas externas não há manutenção de ALI's nem apuração de fórmulas, cálculos ou dados derivados⁴ (IFPUG, 2000). Exemplos (Vazquez, et al., 2003):
 - Geração de arquivo de movimento para outra aplicação;
 - Recuperação de dados com base em critérios informados;
 - Consultas implícitas, embutidas em outras transações;
 - Telas de login com verificação de senhas;
 - Controles tipo “*drop down*” [lista suspensa] ou semelhantes;
- **Saída Externa:** Processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação, mantendo um ou mais ALI's ou contendo apuração de fórmulas, cálculos ou dados derivados (IFPUG, 2000). Exemplos (Vazquez, et al., 2003):
 - Relatório com totalização de dados;
 - Relatórios que também atualizam arquivos;
 - Consultas contendo cálculos ou geração de dados derivados;
 - Informações em formato gráfico;
 - Telas de login com informação de senha, em que a senha é criptografada;

Uma vez identificado e classificado o processo elementar, o próximo passo é a determinação da sua complexidade, o que permite determinar sua contribuição em pontos de função. Para tal, o processo é semelhante à determinação de complexidade dos arquivos lógicos, porém, para as funções do tipo transação,

⁴ **Dados Derivados:** Dados que requerem processamento que não seja apenas ou que seja adicional à recuperação simples e validação de informações de arquivos lógicos (IFPUG, 2000).

conta-se a quantidade de arquivos referenciados e tipos de dados envolvidos no processo elementar (IFPUG, 2000).

Tipos de dados são campos, reconhecidos pelo usuário, não repetidos, que entram ou saem da aplicação e que sejam relevantes a execução do processo elementar (Vazquez, et al., 2003). Podem ser considerados tipos de dados o nome do cliente em um relatório de clientes, a ação que dispara a emissão do relatório, como por exemplo o clique no botão “emitir”, a mensagem exibida informando que o relatório foi impresso com sucesso, etc., ou seja, todas as informações relevantes, envolvidas no processo elementar. Já para a contagem dos arquivos referenciados são considerados os arquivos lógicos envolvidos no processo de alguma forma, seja sendo consultados ou mantidos (IFPUG, 2000). Por exemplo, o arquivo lógico de clientes é um arquivo referenciado na emissão de relatório de clientes.

Contados os processos elementares, classificados quanto ao seu tipo e definidas as quantidades de tipos de dados e arquivos referenciados para cada processo elementar identificado, o próximo passo é sua classificação quanto a sua complexidade. Tal classificação é dada pelas tabelas 3 e 4 a seguir.

Tabela 3: Matriz de complexidade EE, adaptado de (IFPUG, 2000), página 165

EE	AR \ TD	< 5	5 - 15	> 15
	< 2	Baixa	Baixa	Média
2	Baixa	Média	Alta	
> 2	Média	Alta	Alta	

Tabela 4: Matriz de complexidade SE / CE, adaptado de (IFPUG, 2000), página 166

SE / CE	AR \ TD	< 6	6 - 19	> 19
	< 2	Baixa	Baixa	Média
2 - 3	Baixa	Média	Alta	
> 3	Média	Alta	Alta	

AR – TIPO DE REGISTRO / TD – TIPO DE DADO

Determinar contagem de pontos de função não ajustados

A contagem dos pontos de função não ajustados se dá pela conversão da quantidade de funções do tipo dados e transação classificadas quanto à complexidade, para o número de pontos de função correspondente de acordo com cada tipo de função e complexidade (IFPUG, 2000). Tal transformação é feita de acordo com a tabela 5 a seguir.

Tabela 5: Matriz Tipo Função - Contribuição em PF, adaptado de (IFPUG, 2000), páginas 69 e

167

Tipo de Função	Complexidade		
	Baixa	Média	Alta
Arquivo Lógico Interno	7 PF	10 PF	15 PF
Arquivo Interface Externa	5 PF	7 PF	10 PF
Entrada Externa	3 PF	4 PF	6 PF
Saída Externa	4 PF	5 PF	7 PF
Consulta Externa	3 PF	4 PF	6 PF

Uma vez que se tenha a quantidade de funções de cada tipo e complexidade, e se saiba a contribuição em pontos de função de cada tipo de função e complexidade, pode-se chegar ao número de pontos de função não ajustados da aplicação contada. A tabela 6 a seguir ilustra um exemplo de uma contagem de pontos de função não ajustados de uma aplicação.

Tabela 6: Exemplo de contagem de pontos de função não ajustados

Tipo de Função	Complexidade Funcional			Totais por Tipo de Complexidade	Totais por Tipo de Função
	QTD	COMPLEX	PF		
Arquivo Lógico Interno	3	Baixa	7	21	61
	1	Média	10	10	
	2	Alta	15	30	
Arquivo Interface Externa	0	Baixa	5	0	14
	2	Média	7	14	
	0	Alta	10	0	
Entrada Externa	4	Baixa	3	12	26
	2	Média	4	8	
	1	Alta	6	6	
Saída Externa	4	Baixa	4	16	35
	1	Média	5	5	
	2	Alta	7	14	
Consulta Externa	2	Baixa	3	6	24
	3	Média	4	12	
	1	Alta	6	6	
Total Pontos Função Não Ajustados					160

Determinar o fator de ajuste

O fator de ajuste indica a funcionalidade geral fornecida pela aplicação ao usuário, ou seja, características gerais da aplicação que refletem funções que afetam a aplicação de maneira geral, não consideradas na contagem de funções do tipo dado e funções do tipo transação. O objetivo com a aplicação do fator de ajuste é ajustar o número de pontos de função por conta de características da aplicação que impactem no esforço empregado, não medido na contagem dos pontos de função não ajustados (Vazquez, et al., 2003).

O processo de determinação do fator de ajuste consiste na pontuação de zero a cinco quanto ao nível de influência em relação à aplicação medida, para quatorze características gerais do sistema [CGS] (IFPUG, 2000). Tais características são listadas a seguir:

1. Comunicação de Dados
2. Processamento Distribuído
3. Performance
4. Configuração Altamente Utilizada
5. Volume de Transações
6. Entrada de Dados On-Line
7. Eficiência do Usuário Final

8. Atualização On-Line
9. Complexidade de Processamento
10. Reusabilidade
11. Facilidade de Instalação
12. Facilidade de Operação
13. Múltiplos Locais
14. Facilidade de Mudanças

Uma vez apurado o nível de influência de cada característica geral do sistema, a sua somatória reflete o nível total de influência. Este nível é multiplicado por 0,01 e somado a 0,65, chegando-se ao fator de ajuste da aplicação, podendo este variar de 0,65 a 1,35, produzindo um ajuste de -35% ou +35% (IFPUG, 2000). A fórmula que contempla tal operação é apresentada a seguir.

$$\text{Valor Fator Ajuste} = (\text{Total Nível Influência} * 0,01) + 0,65$$

Determinar o número de pontos de função ajustados

Uma vez apurado o número de pontos de função da aplicação contada, e o fator de ajuste, pode-se chegar ao número de pontos de função ajustados. Tal número indica o tamanho funcional da aplicação em pontos de função.

A apuração do número de pontos de função ajustados varia de acordo com o tipo da contagem realizada, pois cada tipo de contagem envolve operações diferenciadas. Enquanto uma contagem de aplicação base mede a aplicação já implantada para o usuário, um projeto de desenvolvimento mede um projeto de um sistema novo, onde podem estar envolvidas criações de funções de conversão de dados, caso dados pré-existentes sejam aproveitados. Já em um projeto de melhoria, funções podem ser alteradas, inclusas e excluídas. Considerando-se tais diferenças entre os tipos de contagem, a seguir são apresentadas as fórmulas para apuração do número de pontos de função ajustados para cada tipo de contagem.

- **Contagem do Tipo Aplicação Base:** Na contagem dos pontos de função de uma aplicação já instalada para o usuário, medem-se apenas as funcionalidades já disponíveis. Para tal, a fórmula para apuração é a seguinte (IFPUG, 2000):

PFA = PFNA * VFA, onde:

PFA = Pontos de função da aplicação

PFNA = Pontos de função não ajustados

VFA = Valor do fator de ajuste

- **Contagem do Tipo Projeto de Desenvolvimento:** Já para os projetos de desenvolvimento, como mencionado anteriormente, funções de conversão podem fazer parte do projeto. Para chegar-se ao número de pontos de função ajustados para este tipo de projeto, a fórmula utilizada é (IFPUG, 2000):

PFDD = (PFNA + PFFC) * VFA, onde:

PFDD = Pontos de função projeto de desenvolvimento

PFNA = Pontos de função não ajustados

PFFC = Pontos de função de funções de conversão

VFA = Valor do fator de ajuste

- **Contagem do Tipo Projeto de Melhoria:** Nos projetos de melhoria, além das eventuais funções de conversão, novas funções podem ser adicionadas, funções existentes podem ser modificadas ou excluídas. Considerando-se tal cenário, para se apurar o número de pontos de função ajustados para este tipo de projeto, a fórmula utilizada é a seguinte (IFPUG, 2000):

PFM = [(PFAD + PFAL + PFFC) * VFAN] + (PFEX * VFAN), onde:

PFM = Pontos de função projeto de melhoria

PFAD = Pontos de função de funções adicionadas

PFAL = Pontos de função de funções alteradas

PFEX = Pontos de função de funções excluídas

PFFC = Pontos de função de funções de conversão

VFAN = Valor do fator de ajuste da aplicação após melhoria

VFAN = Valor do fator de ajuste anterior ao projeto de melhoria

3. PROGRAMA DE MEDIÇÃO

3.1. INTRODUÇÃO

Para que se melhore uma organização de TI, é necessário que se use dados para determinar boas práticas, melhorar o modelo de processos, estabelecer *benchmarks*, analisar tendências, melhorar estimativas, estabelecendo um conhecimento sobre a organização que vai de gerentes a desenvolvedores (Russac, 2002). (Holmes, 2002) afirma que o uso do “sentimento” como subsidio para tomada de decisão não é uma forma eficaz. (Dekkers, 2002) afirma ainda que não se pode gerenciar o que não se pode medir. Organizações de *software* precisam de uma forma de gerenciar a carga de trabalho e decidir o que fazer, quando fazer e onde fazer.

Tais formas de gerenciamento e meios de tomadas de decisão podem ser providos por um programa de medição, o qual pode ser definido como a aplicação continua de técnicas baseadas em medição para o processo de desenvolvimento de *software* e seus produtos, visando sua melhoria, promovendo gerenciamento de informação rápida e significativa (Dekkers, 2002).

Contextualizada a medição de *software* no capítulo anterior, este capítulo vem a apresentar conceitos de um programa de medição, seus objetivos, benefícios e composição. Dentro dos objetivos deste trabalho, este capítulo vem a explorar os conceitos de programas de medição, visando prover base teórica para o estabelecimento de uma abordagem de programa de medição baseado em análise de pontos de função.

3.2. OBJETIVOS DE UM PROGRAMA DE MEDIÇÃO

Um programa de medição tem como objetivo medir um processo de desenvolvimento de *software* ou um determinado projeto de desenvolvimento de *software* em termos de atributos quantitativos, possibilitando análises com base em dados concretos, permitindo a identificação de pontos falhos e boas práticas,

possibilitando a melhoria contínua do processo de desenvolvimento de *software* da organização que o aplica.

Tais melhorias no processo de desenvolvimento de *software* trazem como conseqüência a melhoria nos produtos de *software* desenvolvidos pela organização, o que vem a aumentar a satisfação dos clientes, melhorar o ambiente de trabalho da equipe de TI e a reduzir custos da organização de TI como um todo.

3.3. BENEFÍCIOS DA IMPLEMENTAÇÃO DE UM PROGRAMA DE MEDIÇÃO

Uma vez que se tenha estabelecido um programa de medição, métricas podem ser usadas para identificação de boas práticas, melhora no processo de desenvolvimento, estabelecimento uma linha de base para se determinar tendências, apoio em estimativas e planejamento de projetos, apoio no gerenciamento de orçamento, identificação da quantidade e a qualidade de um produto distribuído, comparação da efetividade e eficiência do processo atual e ferramentas, prover base para comparações de mercado e permitir melhor comunicação entre clientes e desenvolvedores (Russac, 2002). Além disso, as métricas ajudam ainda a clarear os detalhes quanto à definição do produto correto, a execução efetiva do projeto e a distribuição do produto no tempo correto (Dekkers, 2002). Com alguns indicadores de projetos, como produtividade, qualidade e previsibilidade de estimativas, líderes podem tomar suas decisões baseadas em informações sólidas, ao invés de apenas sentimentos ou experiências pessoais (Silveira, 2002).

Além das possibilidades de melhoria no processo e de qualidade, iniciativas do mercado atual vêm a confirmar que a medição de *software* formal leva a um melhor gerenciamento e acompanhamento do processo de desenvolvimento de *software*. Entre tais iniciativas, estão o ISSO 9000 e iniciativas de melhoria de processo definidas pelo *Software Engineering Institute's Capability Maturity Model [SEI CMM for Software]* e o *Capability Maturity Model Integration [CMMI]* (Dekkers, 2002).

3.4. COMPOSIÇÃO DE UM PROGRAMA DE MEDIÇÃO

Um programa de medição pode variar, dentre outros fatores, de acordo com a necessidade, orçamento, tempo e quantidade de pessoas disponíveis de cada organização. Independentemente de tais características, a formalização de um programa de medição deve conter um conjunto de padrões e procedimentos para coleta, armazenamento, análise e reporte de dados. Além disso, papéis e responsabilidades devem ser definidas para suportar a execução do programa (Silveira, 2002).

Tais definições por si só não são suficientes. Para ser bem sucedido, o programa de medição deve ser baseado em objetivos e metas da organização, as quais vem a definir o que se espera melhorar e em que nível, com a implementação do programa de medição (Silveira, 2002; Dekkers, 2002; Russac, 2002).

Definidos os objetivos e metas, medições que resultem em informações que os suportem devem ser definidas. O passo seguinte é a definição de como tais medições serão realizadas e posteriormente como serão analisadas e reportadas. O último passo então, é a definição de como tal programa de medição será implementado na organização, o que contempla a definição da forma de implementação, recursos envolvidos, documentação e educação necessárias.

(Holmes, 2002) e (Dekkers, 2002) definem ainda que o programa de medição deve ser implementado e melhorado de forma contínua. Uma vez que o primeiro conjunto de medições tenha sido implementado com sucesso, os objetivos podem ser revisados para determinar o próximo conjunto de medições para definir e implementar (Holmes, 2002).

(Dekkers, 2002) ilustra tal raciocínio com a Figura 2 a seguir. Tal figura representa resumidamente também, todos os passos do programa de medição descritos anteriormente. No planejamento, identifica-se os objetivos da corporação antes que o programa seja desenhado e implementado. Na implementação, selecionam-se e desenham-se as métricas apropriadas que atendam o planejado, em conjunto com treinamento, iniciação e coleta de medições piloto específicas. No passo de medição, a coleta de dados atual, análise, auditoria e reporte de dados são feitos com as medições alvo, e informações gerenciais são compiladas e apresentadas. No passo final, ação, identifica-se e constroem-se melhorias de processos apropriadas, baseadas nas informações do passo anterior. Do passo da

ação, volta-se então para o passo do planejamento, garantindo-se que o programa de medição ainda está dentro do escopo definido e coletando as métricas corretas (Dekkers, 2002).

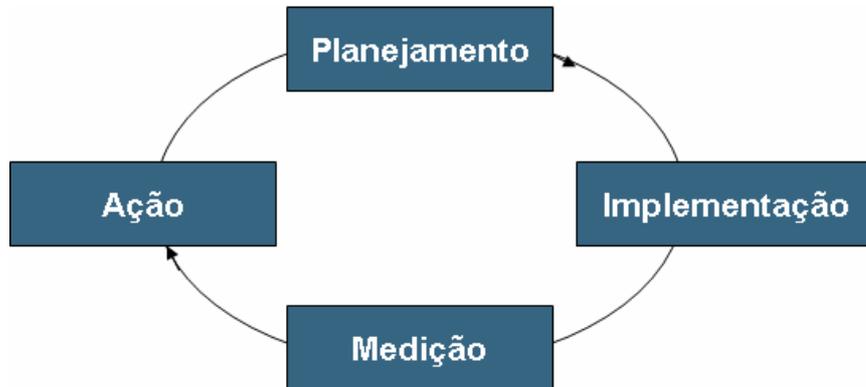


Figura 2 - Processo cíclico de um programa de medição, adaptado de (Dekkers, 2002), página 163

A seguir, cada passo do programa de medição citado é discutido em detalhes.

3.4.1. Planejamento - objetivos e metas da organização

Um programa de medição só tem sentido quanto está alinhado com objetivos e metas da organização de TI. Caso contrário, o programa poderá não sobreviver por muito tempo, pois além de depender muito do apoio de níveis gerenciais, caso não traga nenhum benefício para a organização, esforços gastos com o programa não terão sentido.

Uma das principais características de um programa de medição, é que através dele, informações para tomada de decisões podem ser obtidas. Além disso, um programa de medição pode mostrar a situação atual da organização, possibilitando o estabelecimento de metas. Tais objetivos e metas podem e devem ser estabelecidos no planejamento do programa. (Holmes, 2002) confirma tal raciocínio quando afirma que é importante que sejam estabelecidos objetivos e metas para que se tenha como produto do programa de medição exatamente o que se precisa. Caso não definidos, o produto da medição poderá ser inútil.

Os objetivos podem variar de departamento para departamento e de nível para nível, por isso, devem ser realizadas entrevistas com grupos que abranjam todos os departamentos e níveis envolvidos. (Holmes, 2002) afirma que é importante que empregados de todos os níveis sejam entrevistados, fazendo com que se sintam parte do processo de definição, facilitando-o como um todo. (Holmes, 2002) afirma ainda que é importante que o escopo de tais objetivos e metas seja bem definido, de forma a não sobrecarregar pessoas. Visando a definição apropriada de tal escopo, (Silveira, 2002) acredita que a implementação pode ter uma abordagem baseada em fases.

Para a efetiva definição de tais objetivos, (Dekkers, 2002) afirma que é importante que se leve em consideração algumas características. Para (Dekkers, 2002) os objetivos devem ser: estratégicos, mensuráveis, atingíveis, realísticos e focados. Uma vez definidos os objetivos e metas, eles devem ser consolidados e priorizados (Holmes, 2002). A seguir, (Holmes, 2002) relaciona alguns exemplos de objetivos de um programa de medição:

- Melhorar a produtividade dos projetos;
- Melhorar a qualidade dos projetos;
- Reduzir o custo dos projetos;
- Implementar inspeções formais.

3.4.2. Definição de medições

Definidos os objetivos e metas, o passo seguinte é a definição das medições que serão necessárias para endereçá-los. Devem ser definidas medidas que mostrem o status ou progresso de cada objetivo ou iniciativa em particular (Holmes, 2002).

Se um objetivo da organização é melhorar a produtividade da manutenção em 50%, algumas questões de métricas devem ser “Quais são nossos níveis atuais de suporte?”, “Quantas oportunidades de melhorias existem?”, “Como nós saberemos quando atingirmos 50% de melhora?”. As métricas requeridas para responder a tais perguntas incluem as taxas de suporte do passado e atuais (por exemplo, horas trabalhadas em manutenção por tamanho do produto), combinadas com

características das aplicações (como o tipo e linguagem de desenvolvimento) e níveis de qualidade e suporte (defeitos por tamanho do produto suportado). A seguir, (Holmes, 2002) ilustra um exemplo de objetivos e medidas que os endereçam.

Tabela 7 - Relação Objetivos X Medidas, adaptado de (Holmes, 2002), página 100

Objetivo / Iniciativa	Medida
Melhorar a produtividade dos projetos	Pontos de função por hora
Melhorar a qualidade dos projetos	Defeitos distribuídos por ponto de função
Reduzir o custo do projeto	Custo por ponto de função
Implementar inspeções formais	Taxa de eficiência de remoção de defeitos (defeitos encontrados após correções dividido pelo total de defeitos) Defeitos distribuídos por ponto de função

Na tabela 7, pode ser observado que dependendo das questões que a organização pretende responder, diferentes métricas devem se requeridas para prover respostas (Dekkers, 2002). Uma medida pode suportar mais de um objetivo, permitindo que mais objetivos sejam endereçados (Holmes, 2002).

Pode ser observado também, que a medição em pontos de função é uma medida presente em todas as métricas que endereçam objetivos. O tamanho de aplicações de *software* em pontos de função, em conjunto com outras medições, produzem métricas normalizadas por tamanho em pontos de função as quais podem ser calculadas e usadas para análises comparativas (Dekkers, 2002). Como exemplo, (Dekkers, 2002) afirma que para calcular taxas de entrega, o tamanho de cada produto de desenvolvimento em pontos de função pode ser dividido pelo esforço de trabalho empregado em horas. Tal taxa de entrega pode ser utilizada para análises comparativas entre organizações, proporcionando oportunidades de melhoria no processo.

(Dekkers, 2002) cita ainda outras aplicabilidades do tamanho em pontos de função no endereçamento de objetivos. Métricas de produtividade e entrega podem

ser calculadas pela utilização de pontos de função com outras medidas. Métricas de qualidade (densidade de defeitos) e proporcionalidade de suporte (tamanho da aplicação suportada, por pessoas envolvidas no trabalho de manutenção) são também possíveis utilizando-se pontos de função e outras medidas correlatas coletadas.

O quanto as métricas apóiam no endereçamento dos objetivos definidos e quantos objetivos uma métrica pode endereçar, deve ser um fator levado em consideração na definição das métricas utilizadas pela organização. A seguir, (Dekkers, 2002) relaciona medidas que podem ser utilizados no endereçamento de objetivos e o quanto tais medidas podem ser reaproveitadas.

Tabela 8 - Aplicabilidade de medidas para endereçamento de objetivos, adaptado de (Dekkers, 2002), página 165

Objetivos	Medidas				
	Atributos do Projeto	Pontos de Função	Esforço de Trabalho	Defeitos	Taxa de Satisfação do Cliente
Melhorar Estimativas	✓	✓	✓		
Melhorar Produtividade	✓	✓	✓		
Melhorar Qualidade	✓	✓		✓	✓
Avaliar impacto de ferramentas no suporte		✓	✓	✓	
Avaliar Construção ou Compra	✓	✓	✓		
Melhorar Testes		✓	✓	✓	✓

3.4.3. Definição de coletas de dados

Uma vez que as medidas que endereçam os objetivos definidos estejam estabelecidas, o passo seguinte é a definição das coletas de dados que serão necessárias para que se chegue a tais medidas. (Holmes, 2002) organiza a coleta de dados em quatro passos:

1. Definição de dados necessários;
2. Definição de pontos de coletas de dados;
3. Definição de responsabilidades pela coleta dos dados;
4. Meios de coletas de dados.

A seguir, cada passo citado é discutido.

3.4.3.1. Definição de dados

Definem-se quais dados são necessários para suportar as métricas definidas. Cada pedaço de dado necessário para uma medida precisa ser identificado e definido em termos que todos possam entender. Por exemplo, se medição da produtividade através de pontos de função por hora é selecionado, pontos de função e esforço gasto em horas serão dados requeridos. O esforço gasto precisará ainda ser definido baseado nas atividades que devem ser consideradas na sua composição, como por exemplo, definição de requisitos, design, codificação, etc. (Holmes, 2002).

3.4.3.2. Definição de pontos de coletas de dados

Definem-se os pontos do processo de desenvolvimento em que os dados definidos serão coletados. Para (Holmes, 2002), as atividades de coleta de dados devem ser integradas ao ciclo de vida do processo de desenvolvimento, fazendo com que a medição torne-se parte do processo e não seja percebido como algo extra. (Holmes, 2002) afirma ainda que a coleta de dados deve ser feita apenas em pontos necessários para suportar as medições selecionadas. Por exemplo, se o objetivo é melhorar a produtividade do projeto e a medição escolhida para o acompanhamento da produtividade é pontos de função por hora, significa que será

necessário fazer a contagem em pontos de função durante determinadas fases do projeto.

3.4.3.3. Definição de responsabilidades pela coleta dos dados

Uma vez definidos os dados e os pontos de coleta, definem-se os papéis responsáveis pela coleta e divulgação dos dados. Tal tarefa consiste na identificação dos papéis exercidos durante o processo de desenvolvimento de *software* executado pela organização, definindo-se quais, dentre os papéis identificados, serão responsáveis pela coleta e divulgação de cada parte de informação, de acordo com suas atribuições, habilidades e funções.

3.4.3.4. Meios de coletas de dados

A definição de meios de coletas de dados implica na definição dos métodos e formas empregadas em cada tarefa de coleta. (Holmes, 2002) afirma que para tal, é importante que os dados já existentes da organização sejam aproveitados.

Uma vez definidos os dados, pontos de coletas, responsabilidades e meios, (Holmes, 2002) ilustra um exemplo da relação entre tais informações.

Tabela 9 - Relação Objetivos, Medições, Dados e Responsabilidades, adaptado de (Holmes, 2002), páginas 103 e 104

Objetivos / Iniciativas	Medições	Definição de dados	Responsabilidades
Melhorar a produtividade dos projetos	Pontos de função por hora	Pontos de função contados durante a implementação do projeto; Esforço capturado durante o ciclo de vida do projeto	Especialista em pontos de função com especialista no assunto tratado (registrar dados em planilha); Desenvolvedor (registrar dados no sistema de apontamento de horas)
Melhorar a qualidade dos projetos	Defeitos distribuídos por ponto de função	Pontos de função contados durante a implementação do projeto; Defeitos distribuídos após a correção por três meses	Especialista em pontos de função com especialista no assunto tratado (registrar dados em planilha); Help Desk, registrando dados no sistema de controle de defeitos quando reportados pelos usuários
Reduzir o custo do projeto	Custo por ponto de função	Pontos de função contados durante a implementação do projeto; Custo da mão-de-obra baseado no esforço;	Especialista em pontos de função com especialista no assunto tratado (registrar dados em planilha); Gerente de projetos, calculando e registrado dados no formulário de conclusão do projeto
Implementar inspeções formais	Taxa de eficiência de remoção de defeitos (defeitos encontrados após correções dividido pelo total de defeitos) / Defeitos distribuídos por ponto de função	Pontos de função contados na implementação do projeto; Defeitos encontrados após implementação; Defeitos distribuídos após a implementação por três meses	Especialista em pontos de função com especialista no assunto tratado (registrar dados em planilha); Analistas de testes e de medições, registrando dados no formulário de defeitos e em planilhas; Help Desk, registrando dados no sistema de controle de defeitos quando reportados pelos usuários

3.4.4. Análise e reporte de resultados

Definidos os objetivos, métricas, dados, responsabilidades e pontos de coleta, o passo seguinte é a definição de como tal conjunto de informações será reportado, para quem será reportado e com que frequência (Holmes, 2002). Os relatórios de um programa de medição são itens chave em qualquer programa de medição. É através deles e das análises feitas sobre eles que o programa de medição se justifica, uma vez que os relatórios proverão informações necessárias quanto ao andamento na busca pelos objetivos da organização e as análises proverão insumos para melhoria no processo de desenvolvimento como todo.

(Silveira, 2002) considera que os relatórios de um programa de medição podem ser produzidos para prover informações quanto a tendências de produtividade, defeitos e tempo de resposta ao mercado. (Silveira, 2002) afirma ainda que resumo dos projetos, detalhes e relatórios de análises podem ser gerados para cada projeto finalizado.

Para a definição de como os dados serão reportados, (Holmes, 2002) afirma que é importante considerar que as várias audiências identificadas podem ter diferentes necessidades ou foco, podendo ser necessários relatórios individualizados. As audiências dos relatórios podem incluir gerentes de projetos, gerentes da área de desenvolvimento, time de medição, times de projetos e outros. (Holmes, 2002) afirma que o público alvo dos relatórios devem ser envolvidos para validação dos resultados e verificação quanto ao atendimento de expectativas e necessidades, sendo que especificamente o time de projeto deve receber análises e relatórios do projeto em que participaram, podendo utilizar os dados para identificar possibilidades de melhorias em seus próximos projetos.

Compilar e reportar os dados coletados por si só não é suficiente. (Holmes, 2002), (Silveira, 2002) e (Russac, 2002) afirmam que análises sobre os resultados devem ser feitas, permitindo a identificação de pontos positivos e falhos nos projetos, objetivando a promoção dos pontos positivos e a extinção dos pontos falhos para nos próximos projetos. (Russac, 2002 p. 154) afirma que o ponto mais crítico de todo o processo de métricas é a análise dos dados e ainda questiona: “Para que serve uma coleção de dados se não são analisados quanto aos aspectos bons e ruins do projeto?”.

Para (Silveira, 2002), o processo de análise de resultados deve incluir a análise de indicadores de performance do projeto, comparando seus resultados contra projetos passados, a performance geral da organização, corporação e indústria ou *benchmarks* de mercado. (Russac, 2002) afirma ainda que a utilização de um banco de dados de métricas torna possível tal comparação. A figura 3 a seguir ilustra um exemplo de relatório comparativo entre resultados do projeto e os resultados da organização e do mercado.

Projeto XYZ

Categoria da Métrica	Nome da Métrica	Fórmulas	Cálculo das Métricas do Projeto	Resultado das Métricas do Projeto	Resultados Comparáveis da Organização	Resultados do Mercado
Produtividade	Taxa de Produtividade	Esforço em horas	13000	9,29	8	10
		Tamanho em pontos de função	1400			
Produtividade	Percentual de Retrabalho	Horas de retrabalho	2000	15,38%	25%	35%
		Total de horas do projeto	13000			
Qualidade	Eficiência de Remoção de Defeitos	# de defeitos antes da entrega	300	90,91%	75%	85%
		# total de defeitos descobertos (antes e depois da entrega)	330			
Qualidade	Taxa de Defeitos de Produção	Total de defeitos em produção	30	0,021	0,025	0,05
		Pontos de função instalados	1400			
Qualidade	Mudança de Escopo	PF alterações + adições + exclusões	250	17,89%	15%	20,33%
		Contagem de PF original	1400			
Custo	Eficiência de Custo	Custo Atual	\$2.500.000	\$1,786	\$1,200	1,10%
		Pontos de função instalados	1400			
Tempo	Duração do Projeto	Data do término do projeto	12/10/2000	6,02	5	4,5
		Data de início do projeto	06/10/2000			

Os resultados apresentados são fictícios.

Figura 3 - Exemplo de relatório comparativo de medições, adaptado de (Russac, 2002), página 155

(Silveira, 2002) afirma que baseado em tal análise, áreas de melhoria no processo devem ser identificadas, recomendando ações para projetos futuros. Complementando tal raciocínio, (Holmes, 2002) afirma que os dados não devem ser reportados sem contextos ou explicações. Para (Holmes, 2002) é necessário que análises sejam feitas sobre os dados e que seja assegurado que as informações serão interpretadas e usadas de maneira correta pela audiência. (Holmes, 2002) cita ainda três informações que não podem faltar nos relatórios de um programa de medição:

- **Observações:** O que foi identificado. Exemplo: “Defeitos entregues por pontos de função diminuíram no último período para algumas áreas”;
- **Conclusões:** O que a análise está dizendo. Exemplo: “A diminuição nos defeitos entregues nestas áreas pode ser atribuído a implementação das inspeções formais durante o ciclo de vida”;
- **Recomendações:** Como proceder diante de tais observações e conclusões. Exemplo: “Inspeções formais devem ser consideradas para toda a organização”.

Para que se chegue a conclusões quanto ao projeto executado, (Russac, 2002) afirma que é importante que se façam reuniões com o time do projeto, apresentando resultados parciais, apontando áreas falhas e questionando quanto aos fatores que possam ter contribuído para o resultado. Se por exemplo a taxa de produtividade foi muito abaixo de projetos similares, (Russac, 2002) afirma que é importante que se olhe as razões possíveis, como por exemplo, a utilização de uma nova ferramenta ou tecnologia sem o devido preparo. Para (Russac, 2002) tão importante quanto à identificação de áreas falhas, é a identificação das áreas positivas, identificando-se as razões do sucesso, possibilitando que o time continue com suas melhores práticas.

Após ter analisado os dados, ter se reunido com o time do projeto e ter identificado pontos positivos e falhos, bem como seus motivos, (Russac, 2002) define que um relatório deve ser divulgado não só ao time do projeto, como também a todos os times de projetos e gerentes, possibilitando que todos aprendam com as lições aprendidas.

Além dos relatórios de projetos, relatórios organizacionais também podem ser elaborados, comparando-se diferentes métricas da organização através do tempo. Gráficos para cada métrica, representando o período atual, podem demonstrar a gerentes e executivos tendências organizacionais, alertando-os para áreas que requerem melhorias futuras ou reconhecimento (Russac, 2002). A seguir é apresentado um exemplo de tal relatório.

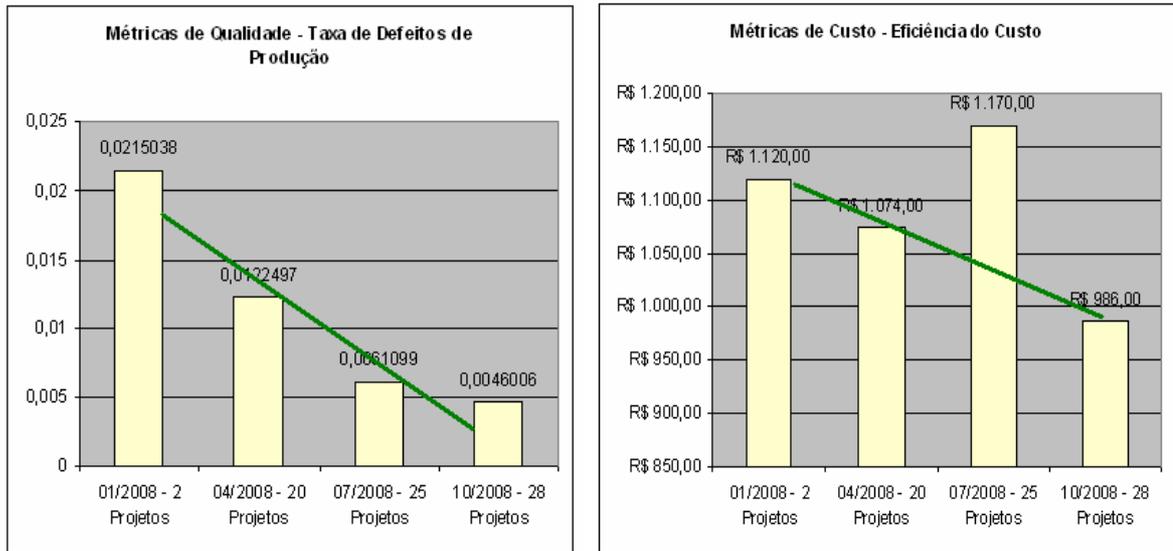


Figura 4 - Exemplo de relatório de métricas organizacional, adaptado de (Russac, 2002), página 156

3.4.5. Implementação do programa de medição

Definidos os objetivos e metas, dados, coletas, responsabilidades, formas de análise e reporte de resultados, é preciso que se planeje a implementação do programa. Tal implementação pode variar quanto à forma, tamanho, quantidade de recursos envolvidos, escopo, fases, tipo e tamanho da organização, dentre outros fatores. Para (Holmes, 2002), a implementação do programa de medição requer planejamento e desenvolvimento, bem como tomada de decisões, sendo que a abordagem seguida pode depender do nível de recursos e da estrutura da organização. Tais variantes são discutidas a seguir.

3.4.5.1. Forma de implementação

(Holmes, 2002) define que as possíveis formas de implementação de um programa de medição podem variar da seguinte forma:

- Em toda a organização de uma só vez;
- Em projetos selecionados ou aplicações em algumas áreas ou departamentos;
- Em áreas ou departamentos selecionados em fases, enquanto o processo é incorporado na organização.

Para (Dekkers, 2002), iniciar um programa de medição em toda a organização de uma só vez não é a melhor escolha, pois, corre-se o risco de se chegar a métricas desalinhadas, além do risco da introdução de um novo conceito de forma abrupta. (Dekkers, 2002 p. 166) afirma ainda que a melhor forma de se iniciar um programa de medição é através de um projeto piloto e argumenta: “Pequeno e tímido é sempre melhor do que audacioso e caótico para a implementação de novos conceitos”.

3.4.5.2. Formas de utilização de recursos

(Holmes, 2002) afirma que dado o esforço necessário de recursos para a realização das atividades do programa de medição, é preciso que se defina quais recursos serão envolvidos, considerando-se o tempo disponível de cada recurso e seu expertise no assunto. Diante de tal necessidade, (Holmes, 2002) aponta três possibilidades quando da definição da forma de utilização de recursos em um programa de medição:

- Utilizar um recurso de cada departamento;
- Utilizar recursos externos;
- Estabelecer um grupo centralizado de métricas.

(Russac, 2002) afirma ainda que mais de uma possibilidade pode ser combinada, principalmente em fases iniciais do programa, onde o conhecimento de medição e expertise no assunto são necessários. Para (Holmes, 2002), consultorias ou especialistas no assunto podem ser contratados inicialmente para apoiar no programa de medição, transferindo aos poucos seu conhecimento para a equipe interna, até que se consiga estabelecer um time centralizado de medição.

3.4.5.3. Documentação do programa e métodos

Definida a forma de implementação e os recursos envolvidos, (Holmes, 2002) afirma que a documentação do processo de medição e seus métodos deve ser feita. Para tal, (Holmes, 2002) define que deve ser documentado como o programa de medição será integrado ao ciclo de desenvolvimento atual, as atividades específicas de medição, quando e como tais atividades são executadas, formulários, ferramentas e formatos de relatórios, desta forma garantindo que o programa será implementado consistentemente, sendo claro para todos os envolvidos.

3.5. FATORES CRÍTICOS DE SUCESSO EM UM PROGRAMA DE MEDIÇÃO

Programas de medição podem falhar, podem não produzir o resultado esperado ou mesmo podem gerar sentimentos indesejáveis nas equipes de projetos, levando a conseqüências incalculáveis caso não sejam implementados adequadamente ou caso detalhes importantes não sejam levados em consideração.

Ao definir um programa de medição é muito importante que sejam levadas em consideração boas práticas e experiências vividas por pessoas com experiência no assunto. A seguir, uma lista de fatores considerados como críticos para os autores considerados nesta pesquisa são listados.

- **Precisão e consistência de dados:** Dados são suscetíveis a forma como são alimentados. Caso os dados não sejam precisos na alimentação, os resultados também não serão precisos. É muito importante que as informações utilizadas pelo programa de medição sejam consistentes. O lançamento de horas trabalhadas em projetos é um dos principais pontos de atenção. Horas extras, horas gastas em manutenção de aplicativos diversos, horas gastas em reuniões fora do projeto, etc. devem ser computadas apropriadamente, para não influenciar no resultado do projeto (Silveira, 2002; Dekkers, 2002; Russac, 2002);
- **Integração ao dia-a-dia:** Programas de medição bem sucedidos são aqueles que se tornam parte do processo diário. Para ser bem sucedida, a medição não pode ser vista como um trabalho adicional, ou como tarefas sem valor agregado no ciclo de desenvolvimento (Dekkers, 2002). Líderes seniores e gerentes de projetos precisam acreditar no programa de métricas, fazendo todos os esforços possíveis para garantir que o processo de métricas está sendo seguido e as pessoas estão tendo tempo suficiente para coletar dados das métricas (Silveira, 2002);
- **Compromisso com a implementação:** Organizações que implementam medição como “algo a ser feito” não terão sucesso. É preciso que se tenha compromisso com a mudança. O sucesso do programa será determinado pela fundição da gerência continua (Dekkers, 2002);
- **Alinhamento com os objetivos da organização:** O programa de métricas deve apoiar na melhoria do processo e deve estar alinhado com os objetivos da

organização. É importante que sejam analisados cenários antes e depois do programa de medição, estabelecendo uma correlação entre as iniciativas de melhorias do processo e dos resultados atuais, verificando como os resultados da melhoria do processo estão impactando os objetivos da organização (Silveira, 2002). Além disso, quando o programa de medição está alinhado com os objetivos e iniciativas da organização, o processo torna-se parte de algo que já acontece no dia-a-dia, sendo significativo para todos os colaboradores (Holmes, 2002);

- **Pensar grande, porém, começar pequeno:** É importante que sejam selecionadas poucas métricas para implementação inicial. Um planejamento adequado, usando uma abordagem baseada em fases, evitando implementações “*big bang*”, evitará que o escopo seja tão vasto que não possa mostrar resultado em tempo hábil ou que seja impossível de ser atingido (Silveira, 2002; Russac, 2002);
- **Utilização de padrões de mercado:** Métricas padrões de mercado (como a análise de pontos de função) facilitam comparações de resultados com outras organizações, uma vez que são difundidas e utilizadas por diversas organizações (Russac, 2002);
- **Considerar o tempo como maturidade:** Um programa de métricas precisa de tempo, projetos e observações para que seja considerado maduro. Não é uma boa prática tirar conclusões baseadas em números limitados de observações (Silveira, 2002);
- **Utilização de resultados de forma adequada:** Os resultados devem ser utilizados única e exclusivamente para a melhoria do processo de desenvolvimento de sistemas como um todo, tomada de decisões e definição de objetivos. Colaboradores não devem ser julgados (Silveira, 2002; Russac, 2002);
- **Consultoria e treinamento externo:** É importante que se tenha profissionais certificados pelo IFPUG (certified function point specialist – CFPS) na equipe de medição ou que se obtenha consultoria de tais profissionais. Transferência de conhecimento é uma chave, podendo ajudar a evitar imprevistos (Dekkers, 2002). Neste mesmo sentido (Russac, 2002) afirma ainda que é importante que todos os envolvidos no programa de medição tenham treinamento adequado.

3.6. RECURSOS HUMANOS: TREINAMENTO E MUDANÇA CULTURAL

Como na maioria dos processos de mudança ou implantação de novos conceitos, os recursos humanos têm papel fundamental e devem ser considerados a todo momento. No que diz respeito à implantação de um programa de medição, algumas questões relacionadas ao papel dos recursos humanos devem ser levadas em consideração.

Primeiramente, como discutido anteriormente, recursos humanos devem ser definidos para a execução de atividades de medição. Dada tal definição, necessidades de treinamento para tais recursos podem surgir. Uma vez definidos os recursos e o programa de medição, mudanças culturais deverão surgir e é preciso que se esteja preparado para enfrentar as conseqüências que as mudanças culturais podem trazer. A seguir, tais questões são discutidas em detalhes.

3.6.1. Necessidade de treinamento

(Holmes, 2002) afirma que educação apropriada é necessária para todos os envolvidos no programa de medição, dependendo das suas atividades dentro do processo estabelecido. (Holmes, 2002) exemplifica tipos de treinamento apropriados para os vários tipos de funções dentro de um programa de medição através da tabela 10 apresentada a seguir.

Tabela 10 - Exemplos de treinamentos para diferentes audiências, adaptado de (Holmes, 2002), página 109

Audiência	Exemplo de Treinamento
Grupo de métricas	<ul style="list-style-type: none"> • Treinamento em análise de pontos de função; • Educação em análise e reporte; • Estimativa de projeto de software; • Métodos de medição • Repositório de pontos de função e ferramentas de análise de dados
Desenvolvedores	<ul style="list-style-type: none"> • Educação no processo e envolvimento; • Treinamento em análise de pontos de função
Gerentes	<ul style="list-style-type: none"> • Educação em análise e uso de dados

3.6.2. Mudança cultural

A implantação de um programa de medição em uma organização pode trazer diversos benefícios, como visto até aqui. Porém, é preciso que se esteja atento a questões que muitas vezes passam despercebidas, e que podem causar danos incalculáveis. É o caso da mudança cultural, a qual dependendo da forma como for tratada, pode levar ao descontentamento da equipe e em última instância a perda de profissionais.

(Dekkers, 2002) define que a medição, como qualquer outra iniciativa corporativa, envolve mudança cultural, transformando uma organização que gerencia por sentimento em outra que gerencia baseada em fatos. As pessoas passam a ver seu trabalho, sua carreira e até mesmo a si próprios de forma diferente quando passam a conviver em um contexto onde se pratica a medição. Dado tal cenário, (Dekkers, 2002) afirma ainda que não há dúvidas que quando a medição de *software* é introduzida, a resistência a mudanças das pessoas é aflorada. Sendo algo inevitável e muito comum, é necessário que a gerência esteja consciente e preparada para tais resistências. (Dekkers, 2002) relaciona os principais tipos de manifestação de resistência à implantação do programa de medição:

- Questionamento quanto à consistência das informações;
- Testes do comprometimento e apoio gerencial ao programa de medição;
- Compartilhamento de percepções, rumores e mitos sobre a iniciativa.

Para reduzir o impacto da mudança cultural, (Dekkers, 2002) recomenda que sistemas de recompensa e punição sejam realinhados para promover e recompensar a coleta de dados precisa, ao invés de punir resultados insatisfatórios. (Dekkers, 2002) alerta ainda que a medição só será bem sucedida, se as pessoas não forem punidas com os dados que elas próprias reportam.

4. ABORDAGEM PARA UM PROGRAMA DE MEDIÇÃO BASEADO EM ANÁLISE DE PONTOS DE FUNÇÃO

Os capítulos anteriores introduziram e contextualizaram métricas, medições e programas de medições e apresentaram referências e embasamentos teóricos. O objetivo deste capítulo é apresentar uma abordagem simplificada para um programa de medição, usando-se como métrica base, a medição em pontos de função.

A análise de pontos de função foi selecionada como medição base, pois, como visto anteriormente, tal medição é amplamente utilizada pelo mercado, o que permite a realização de comparações entre as medições realizadas pela organização e as realizadas por outras organizações. Além disso, tal medição proporciona uma medida padrão de sistemas, podendo esta ser relacionada a diversas outras medidas / métricas, chegando-se a fatores e taxas. Tais fatores e taxas são amplamente utilizados em programas de medição.

Como foi descrito no decorrer deste estudo, existem diversas formas de implementação de um programa de medição, podendo tal implementação variar de acordo com cada organização, do processo de desenvolvimento de *software* utilizado (ou pela não utilização), pela quantidade de recursos disponíveis, pela forma adotada para implantação, de acordo com os objetivos e metas da organização, dentre outras variantes (para mais detalhes quanto às formas de variação, referenciar-se ao tópico 3.4 deste trabalho). Dada tamanha variação, a abordagem apresentada por este estudo não se prende a características específicas, ficando a customização para cada tipo de ambiente a cargo de estudos posteriores.

Este capítulo apresenta as premissas para implementação do programa de medição definido, e em seguida o programa de medição em si. A abordagem apresentada é totalmente embasada pelo estudo realizado e apresentado no decorrer deste trabalho. A estrutura da definição para o programa utilizado neste estudo segue os passos definidos por (Holmes, 2002) (discutido no capítulo 3).

4.1. PREMISSAS PARA O PROGRAMA DE MEDIÇÃO ABORDADO

Para que se implemente o programa de medição abordado por este estudo, algumas premissas são assumidas. Tais premissas são listadas a seguir.

4.1.1. Processo de desenvolvimento de *software*

A abordagem de programa de medição apresentada não está associada a nenhum processo de desenvolvimento de *software* específico, desta forma sendo passível de aplicação em organizações que não implementem nenhum processo até organizações que implementem processos rigorosos e formais.

Entretanto, conforme visto no decorrer deste estudo, um programa de medição, quando implementado por uma organização, passa a fazer parte do seu contexto de desenvolvimento de *software*, sendo que novas atividades são atribuídas a profissionais e métricas são apuradas com base em produtos específicos gerados por este contexto de desenvolvimento. Por conta de tal integração, este estudo assume que determinados itens estejam presentes no contexto de desenvolvimento de *software* da organização. Tais itens são especificados nos tópicos seguintes.

4.1.2. Atividades / Disciplinas consideradas

Este trabalho pressupõe que o contexto de desenvolvimento de sistemas esteja dividido em disciplinas⁵ onde atividades específicas sejam realizadas por papéis⁶

⁵ **Disciplinas:** (Kroll, et al., 2003) definem que disciplinas são “*containers*” utilizados para organizar atividades de um processo. Em outras palavras, disciplinas são agrupamentos de determinados papéis e atividades em áreas de conhecimento ou especialidade. Por exemplo, a disciplina de Requisitos compreende o papel do analista de sistemas para a realização da atividade de análise de requisitos.

⁶ **Papel:** Um papel define o comportamento e as responsabilidades de um indivíduo ou um grupo de indivíduos que trabalham em um time. O comportamento é expresso em termos de atividades que o papel executa e cada papel é associado a um conjunto de atividades. Papeis não representam indivíduos nem títulos de funções. Um indivíduo hora pode trabalhar desempenhando atividades de um papel, hora desempenhando atividades de outro. São exemplos de papéis: Analista de Sistemas, Arquiteto, Analista de Testes, Desenvolvedor, etc. (Kroll, et al., 2003).

específicos. Tal suposição se dá, pois, para que se defina um programa de medição, é necessário que se saiba quais atividades são realizadas pela organização, uma vez que tais atividades serão levadas em consideração em diversas medições.

Tais atividades ou mesmo a presença ou não das disciplinas podem variar de organização para organização e principalmente pelo processo de desenvolvimento de *software* utilizado ou pela sua não utilização. As disciplinas também podem variar entre organizações quanto à nomenclatura e quanto às atividades que compreendem. Dada tamanha variação, a abordagem deste trabalho não impõe a realização de atividades específicas, nem a produção de artefatos⁷ específicos, entretanto, considera apenas que as disciplinas estejam presentes no processo de desenvolvimento da organização.

A seguir são listadas e descritas as disciplinas e as atividades que compreendem, limitando-se as que são consideradas por esta abordagem de programa de medição:

- **Gerência de Projetos:** Gerenciamento, planejamento e acompanhamento de projetos.
- **Requisitos:** Definição do que o sistema deve prover, definição das fronteiras / delimitações do sistema, definição do escopo do projeto.
- **Análise e Design:** Definição arquitetural, tradução dos requisitos em projeto técnico.
- **Implementação:** Transformação do projeto técnico em codificação, gerando-se como resultado um produto de *software* executável.

⁷ **Artefatos:** São pedaços de informações produzidos, modificados ou usados por um processo. São produtos tangíveis de um projeto. Exemplos: Modelo de casos de uso, diagramas de classes, documentos de definição arquitetural, etc. (Kroll, et al., 2003).

- **Testes:** Averiguação e avaliação da qualidade do produto desenvolvido, através da busca e documentação de falhas e da verificação se produto está de acordo com os requisitos estabelecidos.

4.1.3. Recursos humanos e papéis considerados

Este estudo pressupõe que a organização disponha de recursos humanos capacitados para execução de determinados papéis dentro de um contexto de desenvolvimento de sistemas. Não há restrição quanto à quantidade de recursos para a execução de tais papeis, ficando a cargo da organização sua definição.

Caso a organização não disponha de recursos capacitados para a execução de todos os papeis definidos, pode ser considerada a opção de contratação de consultorias especializadas, de novos recursos capacitados, ou de capacitação dos recursos disponíveis. Tais opções são discutidas em detalhes neste trabalho no capítulo 3.

A seguir, os papeis necessários para a implementação desta abordagem de programa de medição são listados e detalhados. Cada papel pode variar quanto às atribuições de organização para organização, por isto, esta relação considera apenas as atribuições necessárias para a implementação do programa de medição.

- **Gerente de Projetos / Líder de Projetos:** Profissional responsável pelo acompanhamento dos projetos, e por garantir que as horas despendidas nos projetos sejam contabilizadas adequadamente, bem como os custos.
- **Analista de Sistemas / Analista de Requisitos:** Profissional responsável pela definição dos requisitos sistêmicos e por sua medição em pontos de função.
- **Desenvolvedor:** Profissional responsável pelo desenvolvimento dos sistemas definidos.
- **Analista de Testes:** Profissional responsável pela elaboração e execução de planos de testes e pela contabilização de defeitos encontrados em projetos.

4.1.4. Ferramentas mandatórias e desejáveis

A abordagem deste trabalho não impõe a utilização de ferramentas específicas ou especialistas, porém, dada a necessidade de determinados controles, de aplicação de técnicas específicas e de manipulação de grandes volumes de dados conforme o programa de medição é executado, algumas ferramentas podem ajudar na produtividade dos trabalhos executados e outras se fazem necessárias. A seguir, tais ferramentas são listadas e detalhadas.

4.1.4.1. Mandatórias

- **Repositório de métricas:** Ferramenta responsável pelo armazenamento das diversas métricas coletadas e analisadas no programa de medição. Recomendável a utilização de ferramentas especialistas ou ferramentas customizadas pela organização, porém, planilhas eletrônicas podem ser consideradas quando da impossibilidade das opções anteriores.
- **Repositório de horas trabalhadas (também conhecido como “*time-tracking*”):** Ferramenta responsável por prover mecanismo para lançamento e armazenamento de horas trabalhadas de profissionais em projetos. Recomendável a utilização de ferramentas especialistas ou ferramentas customizadas pela organização, porém, planilhas eletrônicas podem ser consideradas quando da impossibilidade das opções anteriores.
- **Ferramentas para medições:** Ferramentas responsáveis pelo apoio nas tarefas de medições e de aplicações de métricas e técnicas específicas. Planilhas eletrônicas ou ferramentas especialistas podem ser consideradas.

4.1.4.2. Desejáveis

- **Gerador de gráficos:** Ferramenta responsável por análises sobre dados, medições e métricas e geração de gráficos customizados sobre tais dados. Podem ser consideradas a utilização de planilhas eletrônicas, ferramentas customizadas pela organização ou ferramentas especialistas.

- **Ferramentas estatísticas:** Ferramentas que façam análises comparativas sobre bases históricas de métricas e produzam relatórios estatísticos. É aconselhável a utilização de ferramentas customizadas pela organização ou ferramentas especialistas.
- **Ferramentas de medição integradas:** Ferramenta que integre lançamento de horas, cronograma e repositório de métricas. É aconselhável a utilização de ferramentas customizadas pela organização ou ferramentas especialistas.

Estabelecidas as premissas, a seguir o programa de medição é apresentado.

4.2. DEFINIÇÃO DE OBJETIVOS E METAS

Conforme visto nos capítulos anteriores, os objetivos e metas de um programa de medição variam de organização para organização, sendo que estas devem ser estabelecidas principalmente pelo nível executivo da organização e deve ter colaboração de todos os indivíduos atingidos pelo programa de medição.

Para a abordagem definida por este estudo, são definidos quatro objetivos e metas, os quais foram selecionados por serem objetivos comuns dentre diversas organizações de *software*. Tais objetivos são listados e detalhados a seguir.

4.2.1. Melhorar a produtividade dos projetos

Objetiva-se que o programa de medição mostre a produtividade empregada em cada projeto, e que com tal, seja parte do programa de medição a realização de análises para identificação dos fatores que levam a melhores ou piores produtividades, tendo como resultado, a relação e aplicação de ações que levem a melhoria da produtividade em todos os projetos da organização.

Tal objetivo atendido tem como benefício a entrega de produtos em tempo menor, reduzindo-se por conseqüência custos e permitindo que a organização atenda a um número maior de demandas de *software*.

4.2.2. Melhorar a qualidade dos projetos

Objetiva-se que o programa de medição mostre a qualidade dos produtos de *software*, e que com tal, seja parte do programa de medição a realização de análises para identificação dos fatores que levam a melhores ou piores qualidades, tendo como resultado a relação e aplicação de ações que levem a melhora na qualidade de todos os produtos desenvolvidos pela organização.

Tal objetivo atendido tem como benefício a entrega de produtos com menor número de defeitos e mais aderentes às necessidades do cliente, diminuindo o número de retrabalhos e aumentando a satisfação do cliente.

4.2.3. Reduzir o número de alterações de escopo

Alterações de escopo são geradas pela identificação de novos requisitos não contemplados na definição anterior, ou alteração de requisitos / regras de negócio definidas. Ambos os casos exigem trabalho adicional ou mesmo retrabalho por parte da equipe de desenvolvimento de sistemas, o que pode significar aumento no custo, diminuição da qualidade ou alteração de prazos.

Objetiva-se com o programa de medição que alterações de escopo sejam monitoradas e que, através de análises, identifiquem-se as principais causas, visando sua redução ou extinção.

4.2.4. Reduzir custos dos projetos

Objetiva-se que o programa de medição mostre o custo despendido pela organização para produzir unidades de *software*. Tal medição deverá ser comparada com medições realizadas em projetos semelhantes, procurando-se identificar o quão custoso está ou não o processo de desenvolvimento de *software* implementado pela organização.

Uma vez identificado que o custo está acima do padrão, deverá ser parte do programa de medição a realização de análises para identificação das principais causas, visando sua redução ou extinção. Tal objetivo atendido, traz como benefício a utilização eficiente de recursos da organização, permitindo-se que custos excedentes sejam revertidos para investimentos na área como um todo.

4.3. DEFINIÇÃO DE MEDIÇÕES E MÉTRICAS

Definidos os objetivos do programa de medição, devem ser definidas as medições e métricas necessárias que permitam saber o andamento atual de cada objetivo. Uma vez que se saiba o quão cada projeto está dentro ou não dos objetivos definidos, análises podem ser realizadas a fim de se reverter situações ou de se manter melhores práticas.

A seguir, as medições e métricas necessárias para cada objetivo definido no item anterior são relacionadas.

4.3.1. Medições e métricas necessárias para o objetivo “Melhorar a produtividade dos projetos”

Para que se melhore a produtividade dos projetos, o primeiro passo é saber o esforço empregado sobre determinada unidade de medida de *software* padrão. Para tal, a métrica é dada pela seguinte fórmula:

Taxa de produtividade = Esforço gasto no projeto em horas / Total de pontos de função ao término do projeto

4.3.2. Medições e métricas necessárias para o objetivo “Melhorar a qualidade dos projetos”

Para que se melhore a qualidade, o mesmo raciocínio empregado na melhoria de produtividade é válido. Deve-se saber a qualidade atual. Esta, pode ser dada pela quantidade total de defeitos encontrados dividido pelo tamanho do *software* em pontos de função ao término do projeto. A seguinte fórmula representa esta métrica:

Taxa de defeitos = Quantidade total de defeitos encontrados / Total de pontos de função ao término do projeto

4.3.3. Medições e métricas necessárias para o objetivo “Reduzir o número de alterações de escopo”

Para que se reduza o número de alterações de escopo, é preciso saber a taxa de crescimento funcional do *software* ao longo da sua construção. Para tal, a quantidade de pontos de função provenientes de alterações de escopo deve ser

obtida e considerada sobre a quantidade de pontos de função iniciais. Entende-se como pontos de função provenientes de alteração de escopo, apenas os pontos de função incluídos, alterados ou excluídos por conta de alterações de escopo. Funcionalidades não afetadas pela alteração de escopo não são considerados nesta conta. Tal métrica então é dada pela seguinte fórmula:

Taxa de mudança de escopo = Total de pontos de função de alterações de escopo / Total de pontos de função iniciais

4.3.4. Medições necessárias para o objetivo “Reduzir custos dos projetos”

Para que se saiba o quanto se tem gasto efetivamente com a construção de um *software*, é necessário que se saiba o custo por unidade de medida de *software* produzido. Tal métrica é dada pela seguinte fórmula:

Eficiência do Custo = Custo total do projeto / Total de pontos de função ao término do projeto

4.3.5. Medições necessárias para o programa de medição

Definidas as métricas para o endereçamento cada objetivo, medições deverão ser realizadas a fim de se obter os dados necessários para compor cada métrica. A seguir tais medições são listadas. Determinadas medições podem ser utilizadas para endereçar mais de uma métrica ou objetivo.

- Esforço gasto no projeto em horas
- Quantidade total de defeitos encontrados
- Custo total do projeto
- Total de pontos de função iniciais
- Total de pontos de função de alterações de escopo
- Total de pontos de função ao término do projeto

A tabela 11 a seguir resume as métricas necessárias para cada objetivo definido.

Tabela 11 - Relação de métricas X objetivos com o programa de medição

Objetivo / Meta	Métrica
Melhorar a produtividade dos projetos	Taxa de produtividade
Melhorar a qualidade dos projetos	Taxa de defeitos
Reduzir o custo dos projetos	Eficiência do custo
Reduzir o número de alterações de escopo	Taxa de mudança de escopo

Definidas as métricas e medições necessárias, o passo seguinte é a definição dos momentos em que tais medições serão realizadas e quais papéis serão responsáveis por sua realização.

4.4. DADOS, MEIOS E PONTOS DE COLETA E RESPONSABILIDADES

Conforme visto no decorrer deste estudo, a definição de coleta de dados compreende a definição de pontos chave do programa de medição, onde se definem quais são os dados necessários para compor as métricas definidas e como, quando e por quem tais dados são coletados.

No tópico anterior foram definidas as métricas necessárias para endereçar os objetivos esperados com o programa de medição, onde foi visto que tais métricas exigem que dados específicos sejam combinados. Tais dados devem ser definidos individualmente, a fim de se saber do que são compostos, como e em que pontos serão apurados e posteriormente por quem.

A seguir, cada métrica definida para o programa de medição é descrita em detalhes e definida quanto aos dados envolvidos, meios e pontos de coleta e responsabilidades pela coleta. Como produto final é definida uma tabela de relação objetivo, medição, dados e responsáveis de todo o programa de medição.

4.4.1. Esforço gasto no projeto em horas

Compreende a quantidade de horas gastas no projeto pela equipe participante, em determinadas atividades. Como este dado será relacionado posteriormente com a quantidade de pontos de função do projeto para se apurar a produtividade, deve-

se ser definido um escopo quanto ao que se deseja saber se foi ou não produtivo em comparação a outros projetos. Um escopo muito extenso na definição de esforço em horas, como por exemplo, abrangendo todo o projeto, tende a deixar a taxa de produtividade pouco objetiva, uma vez que compreende muitas atividades realizadas por diversos papéis, tornando assim pouco assertiva a identificação das possíveis causas de eventuais discrepâncias.

4.4.1.1. Dados

Dadas tais definições, é estabelecida a apuração do esforço gasto no projeto em horas como:

Esforço gasto no projeto em horas =

Quantidade de horas gastas em atividades de análise de requisitos +

Quantidade de horas gastas em atividades de análise e projeto +

Quantidade de horas gastas em atividades de implementação

Horas gastas nas demais atividades do projeto como gerência de projetos, testes, análise de negócios, distribuição, etc., não devem ser contabilizadas neste item.

4.4.1.2. Meio de coleta

Para a coleta dos dados identificados para esta medição, é necessário que cada profissional que desempenhe os papéis que compõe a medição, lance as horas em que trabalhou no projeto desempenhando tal papel no repositório de horas trabalhadas definido pela organização (*time-tracking*). Tal ferramenta é considerada como mandatória para esta abordagem de programa de medição e é discutida no item 4.1.3 deste capítulo.

4.4.1.3. Pontos de coleta

As horas deverão ser lançadas ao término das atividades ou caso a atividade ultrapasse uma semana de duração, as horas trabalhadas deverão ser lançadas no término da semana.

4.4.1.4. Responsabilidade de coleta

Analistas de sistemas e desenvolvedores são responsáveis pelo lançamento de horas semanal. O gerente de projetos é responsável por garantir o correto lançamento das horas de tais profissionais.

4.4.1.5. Relação sumarizada

A seguir a relação sumarizada das definições para apuração do esforço gasto no projeto em horas é apresentada.

Tabela 12 - Apuração do esforço gasto no projeto em horas - sumarização

Dado	Coleta		
	Meio	Ponto	Responsabilidade
Quantidade de horas gastas em análise de requisitos	Time-tracking	Ao término da atividade ou semanalmente	Analista de Sistemas / Gerente de Projetos
Quantidade de horas gastas em análise e projeto			
Quantidade de horas gastas em implementação			Desenvolvedor / Gerente de Projetos

4.4.2. Custo total do projeto

Compreende o custo total despendido com profissionais durante a sua atuação no projeto. Ao contrário do esforço para a apuração da produtividade, onde é limitado o escopo da medição, no custo total do projeto deve ser considerado o custo do projeto inteiro.

4.4.2.1. Dados

Dadas tais definições, é estabelecida a apuração do custo total do projeto como:

Custo total do projeto =

Somatória (Horas gastas por profissional X Valor/hora profissional)

4.4.2.2. Meio de coleta

Todos os profissionais envolvidos no projeto deverão ter as horas em que trabalharam no projeto devidamente contabilizadas no repositório de horas trabalhadas definido pela organização (*time-tracking*). Tal ferramenta é considerada como mandatória para esta abordagem de programa de medição e é discutida no item 4.1.3 deste capítulo.

O valor/hora por profissional também deverá estar disponível no repositório de horas trabalhadas, para posterior apuração da relação entre horas trabalhadas e valor/hora do profissional.

4.4.2.3. Pontos de coleta

Os pontos de coleta das horas trabalhadas segue a mesma regra definida pelo item 4.4.1.3, ou seja, ao término de cada atividade ou caso a atividade ultrapasse uma semana, ao término da semana. O valor/hora de todos os profissionais envolvidos no projeto deverão ser lançados / atualizados antes do início de cada projeto.

4.4.2.4. Responsabilidade de coleta

Todos os envolvidos no projeto são responsáveis pelo lançamento das suas horas trabalhadas, sendo o Gerente de Projetos o responsável pela exatidão das informações e por garantir que todos os envolvidos tiveram suas horas contabilizadas. Gerentes de projetos também são responsáveis por manter atualizado o valor/hora dos profissionais.

4.4.2.5. Relação sumarizada

A seguir a relação sumarizada das definições para apuração do custo total do projeto é apresentada.

Tabela 13 - Apuração do custo total do projeto - sumarização

Dado	Coleta		
	Meio	Ponto	Responsabilidade
Quantidade de horas gastas por profissional atuante no projeto	Time-tracking	Ao término da atividade ou semanalmente	Todos os profissionais envolvidos / Gerente de Projetos
Valor/Hora dos profissionais atuantes no projeto		Antes do início do projeto	Gerente de Projetos

4.4.3. Total de pontos de função iniciais

Compreende a medição do tamanho do projeto em pontos de função, logo após o fechamento do escopo de requisitos do projeto. Para tal, todos os requisitos do projeto devem ser considerados e sobre eles deverá ser aplicada a técnica de análise de pontos de função, conforme descrito no capítulo 2 deste estudo.

4.4.3.1. Dados

Dadas tais definições, é estabelecida a apuração do total de pontos de função iniciais como:

Total de pontos de função iniciais =

Medição de todo o projeto em pontos de função após o primeiro fechamento do escopo

4.4.3.2. Meio de coleta

Com base nos requisitos definidos para a aplicação, a técnica de análise de pontos de função deve ser aplicada, através do uso de uma planilha de cálculo ou ferramenta especialista, conforme discutido no item 4.1.3 deste capítulo. O valor apurado deve ser lançado na ferramenta repositório de métricas, também discutida no item 4.1.3 deste capítulo.

4.4.3.3. Pontos de coleta

A medição inicial em pontos de função deve ser realizada logo após o fechamento do escopo inicial do projeto, sendo este devidamente oficializado e entendido pelas partes envolvidas no projeto.

4.4.3.4. Responsabilidade de coleta

Uma vez oficializado o fechamento do escopo, uma demanda para medição em pontos de função deve ser gerada pelo Gerente de Projetos. O Analista de Sistemas é então o responsável pela atendimento de tal demanda.

4.4.3.5. Relação sumarizada

A seguir a relação sumarizada das definições para apuração do total de pontos de função iniciais é apresentada.

Tabela 14 – Tamanho do projeto inicial em pontos de função - sumarização

Dado	Coleta		
	Meio	Ponto	Responsabilidade
Total de pontos de função iniciais	Ferramenta de cálculo de pontos de função / Repositório de métricas	Após a oficialização do fechamento do escopo do projeto	Analista de Sistemas / Geração de demanda pelo Gerente de Projetos

4.4.4. Total de pontos de função de alterações de escopo

Compreende a medição dos requisitos alterados, incluídos e excluídos a cada eventual mudança de escopo que ocorra no projeto. Para tal, apenas tais requisitos do projeto devem ser considerados após a mudança do escopo, e sobre eles aplicada a técnica de análise de pontos de função, conforme descrito no capítulo 2 deste estudo.

4.4.4.1. Dados

Dadas tais definições, é estabelecida a apuração do total de pontos de função de alteração de escopo como:

Total de pontos de função de alterações de escopo =

Total de pontos de função de requisitos incluídos + Total de pontos de função de requisitos alterados + total de pontos de função de requisitos excluídos

4.4.4.2. Meio de coleta

Com base nos requisitos alterados, incluídos ou excluídos com a alteração de escopo, a técnica de análise de pontos de função deve ser aplicada, através do uso de uma planilha de cálculo ou ferramenta especialista, conforme discutido no item 4.1.3 deste capítulo. O valor apurado deve ser lançado na ferramenta repositório de métricas, também discutida no item 4.1.3 deste capítulo.

4.4.4.3. Pontos de coleta

A medição em pontos de função deve ser realizada logo após o fechamento do escopo do projeto, após mudança de escopo, sendo esta devidamente oficializada e entendida pelas partes envolvidas no projeto.

4.4.4.4. Responsabilidade de coleta

Uma vez oficializado o fechamento do escopo, após mudança de escopo, uma nova demanda para medição em pontos de função deve ser gerada pelo Gerente de Projetos. O Analista de Sistemas é então o responsável pelo atendimento de tal demanda.

4.4.4.5. Relação sumarizada

A seguir a relação sumarizada das definições para apuração do total de pontos de função de alteração de escopo é apresentada.

Tabela 15 - Total de pontos de função de alterações de escopo - sumarização

Dado	Coleta		
	Meio	Ponto	Responsabilidade
Total de pontos de função incluídos por alteração de escopo	Ferramenta de cálculo de pontos de função / Repositório de métricas	Após a oficialização do fechamento do escopo do projeto, após mudança de escopo	Analista de Sistemas / Geração de demanda pelo Gerente de Projetos
Total de pontos de função aterados por alteração de escopo			
Total de pontos de função excluídos por alteração de escopo			

4.4.5. Total de pontos de função ao término do projeto

Compreende a medição de todo o projeto em pontos de função, após a sua conclusão. Para tal, todo o projeto entregue deve ser considerado, e sobre ele aplicada a técnica de análise de pontos de função, conforme descrito no capítulo 2 deste estudo.

4.4.5.1. Dados

Dadas tais definições, é estabelecida a apuração do total de pontos de função ao término do projeto como:

Total de pontos de função ao término do projeto =

Medição de todo o projeto em pontos de função após sua conclusão

4.4.5.2. Meio de coleta

Com base no projeto entregue a técnica de análise de pontos de função deve ser aplicada, através do uso de uma planilha de cálculo ou ferramenta especialista, conforme discutido no item 4.1.3 deste capítulo. O valor apurado deve ser lançado na ferramenta repositório de métricas, também discutida no item 4.1.3 deste capítulo.

4.4.5.3. Pontos de coleta

A medição em pontos de função deve ser realizada logo após a disponibilização final do projeto para área cliente, sendo esta devidamente oficializada e entendida pelas partes envolvidas no projeto.

4.4.5.4. Responsabilidade de coleta

Uma vez oficializada a entrega do projeto para a área cliente, uma nova demanda para medição em pontos de função deve ser gerada pelo Gerente de Projetos. O Analista de Sistemas é então o responsável pelo atendimento de tal demanda.

4.4.5.5. Relação sumarizada

A seguir a relação sumarizada das definições para apuração do total de pontos de função ao término do projeto apresentada.

Tabela 16 - Total de pontos de função ao término do projeto - sumarização

Dado	Coleta		
	Meio	Ponto	Responsabilidade
Total de pontos de função do projeto entregue	Ferramenta de cálculo de pontos de função / Repositório de métricas	Após a oficialização da distribuição do projeto	Analista de Sistemas / Geração de demanda pelo Gerente de Projetos

4.4.6. Quantidade total de defeitos encontrados

Compreende a contabilização da quantidade de defeitos encontrados durante a realização de testes de validação sobre os produtos de *software* gerados. Defeitos encontrados em testes de verificação, caso realizados pela organização, não são contabilizados nesta medição. A quantidade de defeitos encontrados deve ser contabilizada a cada iteração de testes realizada, caso aconteça mais do que uma, chegando-se ao total de defeitos encontrados em toda a realização dos testes.

4.4.6.1. Dados

Dadas tais definições, é estabelecida a apuração da quantidade de defeitos encontrados como:

Quantidade total de defeitos encontrados =

Quantidade total de defeitos de validação encontrados no produto de *software*.

4.4.6.2. Meio de coleta

A quantidade de defeitos encontrados no produto de *software* testado deverá ser lançada na ferramenta repositório de métricas, discutida no item 4.1.3 deste capítulo.

4.4.6.3. Pontos de coleta

Logo após a conclusão de cada iteração de realização de testes. Entende-se como iteração de testes a conclusão de uma etapa inteira de testes, sendo o motivo da conclusão a impossibilidade de continuar com os testes por algum defeito crítico ou a realização efetiva de todos os testes planejados. Iterações de testes em produtos de *software* podem ocorrer por diversas vezes até que o produto esteja estável. Sempre que concluída uma iteração, a quantidade de defeitos deverá ser contabilizada.

4.4.6.4. Responsabilidade de coleta

O analista de testes responsável pela realização dos testes é o responsável pela contabilização da quantidade de defeitos encontrados em seus testes.

4.4.6.5. Relação sumarizada

A seguir a relação sumarizada das definições para apuração do custo total do projeto é apresentada.

Tabela 17 – Quantidade total de defeitos encontrados - sumarização

Dado	Coleta		
	Meio	Ponto	Responsabilidade
Quantidade total de defeitos encontrados	Repositório de métricas	Após conclusão de cada iteração de testes	Analista de Testes

4.4.7. Relação objetivo, medição, dados e responsáveis

Definidos os objetivos no tópico 4.2, as medições para endereçar tais objetivos no tópico 4.3, os dados necessários para apurar tais medições e os meios de coleta e responsabilidades pelas coletas de tais dados no tópico 4.4, este tópico traz a relação sumarizada de todas as definições. A matriz a seguir apresenta a relação medida, dado, meio, ponto e responsabilidade pela coleta.

Tabela 18 – Definição do Programa de Medição - Relação Medidas, Dados, Meios, Pontos e Responsabilidade pela coleta

Medida	Dado	Coleta		
		Meio	Ponto	Responsabilidade
A - Esforço total em horas	Quantidade de horas gastas em análise de requisitos	Time-tracking	Ao término da atividade ou semanalmente	Analista de Sistemas / Gerente de Projetos
	Quantidade de horas gastas em análise e projeto			Desenvolvedor / Gerente de Projetos
	Quantidade de horas gastas em implementação			
B - Custo total do projeto	Quantidade de horas gastas por profissional atuante no projeto	Time-tracking	Ao término da atividade ou semanalmente	Todos os profissionais envolvidos / Gerente de Projetos
	Valor/Hora dos profissionais atuantes no projeto		Antes do início do projeto	Gerente de Projetos
C - Total de pontos de função iniciais	Total de pontos de função iniciais	Ferramenta de cálculo de pontos de função / Repositório de métricas	Após a oficialização do fechamento do escopo do projeto	Analista de Sistemas / Geração de demanda pelo Gerente de Projetos
D - Total de pontos de função de alterações de escopo	Total de pontos de função incluídos por alteração de escopo	Ferramenta de cálculo de pontos de função / Repositório de métricas	Após a oficialização do fechamento do escopo do projeto, após mudança de escopo	Analista de Sistemas / Geração de demanda pelo Gerente de Projetos
	Total de pontos de função alterados por alteração de escopo			
	Total de pontos de função excluídos por alteração de escopo			
E - Total de pontos de função ao término do projeto	Total de pontos de função do projeto entregue	Ferramenta de cálculo de pontos de função / Repositório de métricas	Após a oficialização da distribuição do projeto	Analista de Sistemas / Geração de demanda pelo Gerente de Projetos
F - Quantidade total de defeitos encontrados	Quantidade total de defeitos encontrados	Repositório de métricas	Após conclusão de cada iteração de testes	Analista de Testes

A seguir, é apresentada a matriz de relação objetivos / metas, com as métricas que as endereçam e suas composições, definidas pela relação entre medições apresentadas na matriz anterior.

Tabela 19 – Definição do Programa de Medição – Relação Objetivos, Métricas e suas Composições

Objetivo / Meta	Métrica	Composição Métrica (ref. Tabela 18)
Melhorar a produtividade dos projetos	Taxa de produtividade	A ÷ E
Melhorar a qualidade dos projetos	Taxa de defeitos	F ÷ E
Reduzir o custo dos projetos	Eficiência do custo	B ÷ E
Reduzir o número de alterações de escopo	Taxa de mudança de escopo	D ÷ C

4.5. ANÁLISE E REPORTE DE RESULTADOS

Definidas as medições necessárias e seus responsáveis, este tópico aborda a definição de como tais dados, após coletados, serão analisados e reportados. Conforme visto nos capítulos anteriores, esta definição abrange a especificação do formato, audiência, frequência de entrega dos relatórios, bem como os responsáveis por análises sobre os resultados.

Esta abordagem de programa de medição define a apuração de três relatórios, os quais são os meios definidos para reporte dos resultados quanto às medições definidas, fornecendo desde dados detalhados de cada projeto até sumarizados de toda a organização de *software*, provendo assim dados necessários para análises quanto ao resultado de projetos individualizados e dados necessários para análises quanto ao resultado da organização como um todo.

Tais relatórios são detalhados a seguir e em seguida é apresentada uma sumarização geral das definições quanto ao reporte e análise de resultados.

4.5.1. Relatório detalhado por projeto

4.5.1.1. Objetivo

Prover informações detalhadas de cada projeto, permitindo análises minuciosas sobre cada resultado. Apresenta o resumo de todo o ciclo de vida do projeto, em termos de medições, métricas e apurações.

Permite que o time envolvido no projeto, gerência da organização, gerentes de projetos e demais times da organização visualizem resultados gerais do projeto, possibilitando o reconhecimento de bons resultados e análise e tomada de ações quando de resultados ruins.

4.5.1.2. Audiência

Todo o time envolvido no projeto, todos os gerentes de projetos, gerência da organização e demais times de desenvolvimento de sistemas da organização.

4.5.1.3. Formato

O relatório detalhado por projeto é composto pela identificação do projeto, onde constam dados de identificação e classificação do projeto, pelos dados do ciclo de

vida do projeto, onde constam as medições realizadas, dados relacionados ao custo do projeto, envolvidos nas atividades contabilizadas para apuração da taxa de produtividade e suas respectivas contabilizações de horas trabalhadas no projeto, histórico de iterações de testes e quantidade de defeitos encontrados. O relatório também é composto pelos resultados do projeto, onde constam apurações de métrica definida para o programa de medição e a comparação com médias de resultados da organização em formato gráfico. Por fim, o relatório apresenta a análise dos resultados. A Figura 5 a seguir ilustra o formato descrito. Os dados apresentados na ilustração são meramente ilustrativos.

Dados do Projeto

Identificação 1865 **Assunto** Sistema de Gestão de Descontos **Plataforma** Microsoft NET / Web Application
Nível Conhecimento Equipe no Assunto / Sistema Baixo **Tipo** Construção de novo sistema

Histórico de Medições em Pontos de Função

Medição	Tamanho Funcional em PF
A - FECHAMENTO ESCOPO	120
Alteração Escopo 01	25
Alteração Escopo 02	34
B - TAMANHO FINAL	150
C - TOTAL ALT ESCOPO	59

Custo do Projeto

Total de Recursos Envolvidos 12
Total Horas Recursos 1120
C - CUSTO TOTAL PROJETO R\$ 50.450,00

Horas Dedicadas por Categoria de Atividades

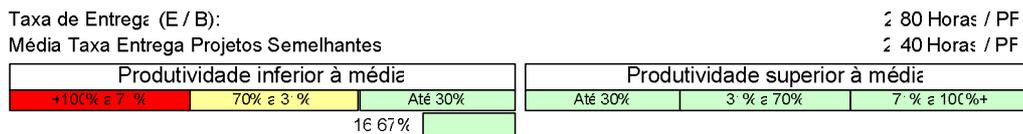
Nome	Categoria	Horas Dedicadas
João da Silva	Análise e Projeto	100
João da Silva	Análise e Projeto	100
José Silveira	Construção	55
Maria Santos	Construção	55
Jorge da Silva	Construção	110
E - TOTAL DE ESFORÇO EM HORAS		420

Histórico de Defeitos

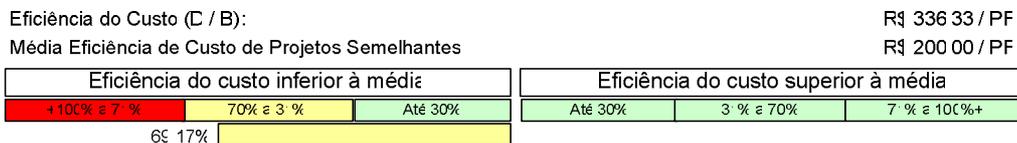
Fase	Qtd Defeitos	
1º Iteração Testes	20	
2º Iteração Testes	7	
3º Iteração Testes	1	
F - TOTAL DE DEFEITOS DO PROJETO		28

Resultados do Projeto

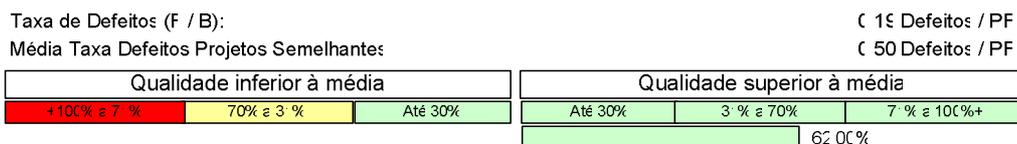
Produtividade



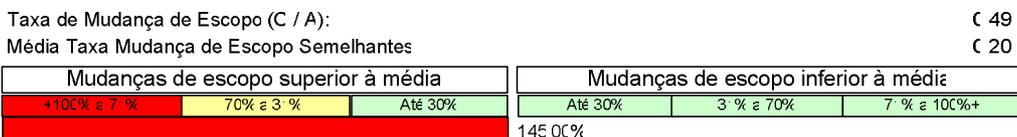
Custo



Qualidade



Mudança de Escopo



Conclusões / Análise dos Resultados

Observações: Apesar da altíssima taxa de mudança de escopo, observa-se que o projeto foi entregue com taxa de produtividade pouco abaixo da média e com qualidade muito superior. O custo, por outro lado, foi um dos pontos fracos do projeto, estando muito abaixo da média.

Conclusões: A alta taxa de mudança de escopo prejudicou o projeto como um todo, fazendo com que esforço fosse desperdiçado com retrabalho, aumentando assim o custo total do projeto e diminuindo a produtividade. Mesmo neste cenário, a equipe surpreendeu conseguindo manter ótimo nível de qualidade.

Recomendações: Recomenda-se que a equipe aperfeiçoe as atividades de elicitação de requisitos, visando a diminuição das alterações de escopo. Tal atividade por si só não será suficiente, uma vez que é a área cliente quem tem o domínio sobre o escopo do projeto, porém, ajudará a reduzir a alta taxa atual consideravelmente. Recomenda-se que a equipe mantenha as práticas de testes unitários, o que tem se demonstrado muito eficiente no aumento da qualidade do produto entregue.

Figura 5 - Definição Programa Medição - Formato Relatório Detalhado por Projeto

4.5.1.4. Responsável pela apuração, análise e divulgação

O Analista de Sistemas é o responsável pela apuração e divulgação. Analista de Sistemas em conjunto com Gerente de Projetos e time envolvido são responsáveis pela análise de resultados, sendo o Analista de Sistemas o responsável pela análise final embasada pela análise conjunta.

4.5.1.5. Frequência de entrega

Após a publicação em produção de cada projeto.

4.5.1.6. Sumarização

A seguir, é apresentada a sumarização das definições para o relatório detalhado por projeto.

Tabela 20 - Sumarização Relatório Detalhado por Projeto

Relatório	Objetivo	Audiência	Responsável Apuração, Análise e Divulgação	Frequência Entrega
Relatório Detalhado por Projeto	Prover informações detalhadas de cada projeto, permitindo análises minuciosas sobre cada resultado	Todo o time envolvido no projeto, todos Gerentes de Projetos, Gerente da Organização de Software e demais times de desenvolvimento de sistemas da organização	Analista de Sistemas: Apuração e Divulgação; Analista de Sistemas, Gerente de Projetos e Time do Projeto: Análise de Resultados	Após a publicação em produção de cada projeto

4.5.2. Relatório comparativo por projeto

4.5.2.1. Objetivo

Prover resultados sumarizados do projeto, em comparação com médias de projetos semelhantes. Permite que seja avaliado se os resultados do projeto estão melhores do que a média, se estão dentro dos padrões da organização ou se estão abaixo da média da organização.

Além de apontar os resultados do projeto, este relatório mostra o quanto o projeto colaborou ou não com os objetivos e metas da organização, mostrando resultados do projeto em associação com os objetivos definidos com o programa de medição.

4.5.2.2. Audiência

Todo o time envolvido no projeto, todos os gerentes de projetos, gerência da organização e demais times de desenvolvimento de sistemas da organização.

4.5.2.3. Formato

O relatório comparativo por projeto é composto pela identificação do projeto, onde constam dados de identificação e classificação do projeto, pelos resultados do projeto, onde constam os resultados em relação aos objetivos definidos para o programa de medição e a média dos resultados de projetos semelhantes, e por fim, pela análise dos resultados. A Figura 6 a seguir ilustra o formato descrito. Os dados apresentados na ilustração são meramente ilustrativos.

Dados do Projeto
Identificação 1865 **Assunto** Sistema de Gestão de Descontos **Plataforma** Microsoft .NET Web Application
Nível Conhecimento Equipe no Assunto Sistema Baixo **Tip** Construção de novo sistema

Objetivo / Meta	Métrica	Composição Métrica	Resultados do Projeto		Resultados da Organização (Projetos Semelhantes)
			Resultado da Medição	Resultado da Métrica	Resultado da Métrica
Melhorar a produtividade dos projetos	Taxa de Produtividade	Total de pontos de função ao término do projeto	150	2,8	2,4
		Esforço total em horas	420		
Melhorar a qualidade dos projetos	Taxa de Defeitos	Total de pontos de função ao término do projeto	150	0,19	0,50
		Quantidade total de defeitos encontrados	28		
Reduzir o custo dos projetos	Eficiência do Custo	Total de pontos de função ao término do projeto	150	R\$ 336,33	R\$ 200,00
		Custo total do Projeto	R\$ 50.450,00		
Reduzir o número de alterações de escopo	Taxa de Mudança de Escopo	Total de pontos de função iniciais	120	0,49	0,20
		Total de pontos de função de alterações de escopo	59		

Legenda
Ponto favorável do projeto
Ponto que necessita de atenção
Ponto falho do projeto

Conclusões / Análise dos Resultados

Observação: Apesar da altíssima taxa de mudança de escopo, observa-se que o projeto foi entregue com taxa de produtividade pouco abaixo da média e com qualidade muito superior. O custo, por outro lado, foi um dos pontos fracos do projeto, estando muito abaixo da média.

Conclusões: A alta taxa de mudança de escopo prejudicou o projeto como um todo, fazendo com que esforço fosse desperdiçado com retrabalho aumentando assim o custo total do projeto e diminuindo a produtividade. Mesmo neste cenário, a equipe surpreendeu conseguindo manter ótimo nível de qualidade.

Recomendações: Recomenda-se que a equipe aperfeiçoe as atividades de elicitação de requisitos, visando a diminuição das alterações de escopo. Tal atividade por si só não será suficiente, uma vez que é a área cliente quem tem o domínio sobre o escopo do projeto, porém, ajudará a reduzir a taxa atual consideravelmente. Recomenda-se que a equipe mantenha as práticas de testes unitários, o que tem se demonstrado muito eficiente no aumento da qualidade do produto entregue.

Figura 6 - Definição Programa Medição - Formato Relatório Comparativo por Projeto

4.5.2.4. Responsável pela apuração, análise e divulgação

O Analista de Sistemas é o responsável pela apuração e divulgação. Analista de Sistemas em conjunto com Gerente de Projetos e time envolvido são responsáveis pela análise de resultados, sendo o Analista de Sistemas o responsável pela análise final embasada pela análise conjunta.

4.5.2.5. Frequência de entrega

Após a publicação em produção de cada projeto.

4.5.2.6. Sumarização

A seguir, é apresentada a sumarização das definições para o relatório comparativo por projeto.

Tabela 21 - Sumarização Relatório Comparativo por Projeto

Relatório	Objetivo	Audiência	Responsável Apuração, Análise e Divulgação	Frequência Entrega
Relatório Comparativo por Projeto	Prover resultados sumarizados do projeto, em comparação com médias de projetos semelhantes	Todo o time envolvido no projeto, todos Gerentes de Projetos, Gerente da Organização de Software e demais times de desenvolvimento de sistemas da organização	Analista de Sistemas: Apuração e Divulgação; Analista de Sistemas, Gerente de Projetos e Time do Projeto: Análise de Resultados	Após a publicação em produção de cada projeto

4.5.3. Relatório evolucional da organização

4.5.3.1. Objetivo

Prover informações históricas da organização quanto ao seu desempenho. Mostra tendências (onde a organização estava e para onde está indo), resultado de tomada de ações (melhorias, aplicação de práticas de sucesso identificadas em análises sobre resultados ao longo da utilização do programa de medição, utilização de novas ferramentas, aplicação de treinamento, etc.) ao longo do tempo e posição da organização em relação aos objetivos e metas definidas inicialmente.

Permite a tomada de ações, uma vez identificadas tendências negativas, a fim de revertê-las. Possibilita a análise quanto a boa ou má sucessão de ações tomadas e a tomada de ações corretivas á tempo. Permite que se identifique se a

organização está ou não no caminho certo rumo a atingir objetivos e metas definidas.

4.5.3.2. Audiência

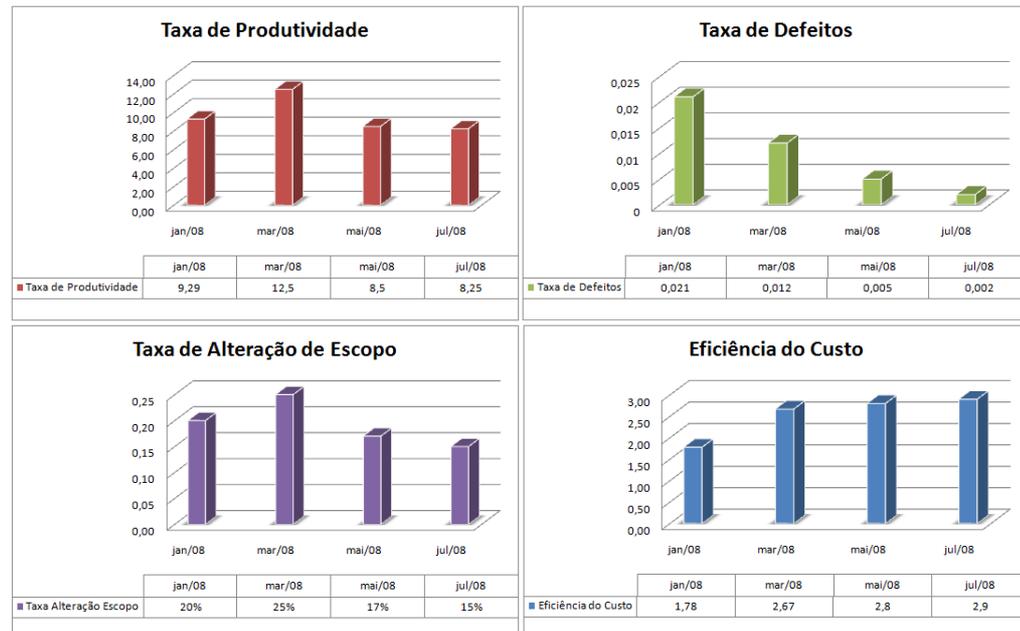
Toda a organização de *software*.

4.5.3.3. Formato

O relatório evolucionar da organização é composto pelo período de análise considerado no relatório, pelos resultados de cada indicador considerado no programa de medição ao longo dos meses analisados e, por fim, pela análise dos resultados. A figura 7 a seguir ilustra o formato descrito. Os dados apresentados na ilustração são meramente ilustrativos.

Relatório Bimestral Evolucionar da Organização

Período de jan/2008 até jul/2008



Observações

À medida que a taxa de produtividade de defeitos e de alteração de escopo demonstram tendência de queda a eficiência do custo tem aumentado. Em março a taxa de produtividade demonstra ter sido prejudicada pelo crescimento na taxa de alteração de escopo. A partir de março a eficiência do custo demonstra estar se estabilizando o que pode significar uma tendência de que se manterá neste patamar.

Conclusões

A organização demonstrou evolução significativa no semestre avaliado. O custo apesar de ter elevado justifica-se pela melhora na qualidade e produtividade.

Recomendações

Como observado altas taxas de alteração de escopo influenciam significativamente na taxa de produtividade da organização o que pode aumentar como consequência o custo e taxa de defeitos. Recomenda-se que esforços sejam voltados para que a taxa de alteração de escopo se estabilize e se possível decresça ao longo do tempo. Recomenda-se também que uma vez estabilizados os demais indicadores sejam voltados esforços para redução da eficiência do custo.

Figura 7 - Definição Programa Medição - Formato Relatório Evolucionar da Organização

4.5.3.4. Responsável pela apuração, análise e divulgação

O Analista de Sistemas é o responsável pela apuração e divulgação. Analista de Sistemas em conjunto com Gerentes de Projetos e Gerente da Organização são responsáveis pela análise dos resultados.

4.5.3.5. Frequência de entrega

Bimestral, porém, podendo ser apurado a qualquer momento quando da necessidade.

4.5.3.6. Sumarização

A seguir, é apresentada a sumarização das definições para o relatório detalhado por projeto.

Tabela 22 - Sumarização Relatório Evolucionar da Organização

Relatório	Objetivo	Audiência	Responsável Apuração, Análise e Divulgação	Frequência Entrega
Relatório Evolucionar da Organização	Prover informações históricas da organização quanto ao seu desempenho	Toda a organização de software	Analista de Sistemas: Apuração e Divulgação; Analista de Sistemas, Gerentes de Projetos e Gerente da Organização: Análise de Resultados	Bimestral

4.5.4. Sumarização das Definições quanto a Análise e Reporte de Resultados

Detalhados todos os relatórios quanto a sua audiência, responsáveis e frequência de entrega, a seguir a sumarização geral é apresentada.

Tabela 23 - Definição Programa Medição - Análise e Reporte de Resultados

Relatório	Objetivo	Audiência	Responsável Apuração, Análise e Divulgação	Frequência Entrega
Relatório Detalhado por Projeto	Prover informações detalhadas de cada projeto, permitindo análises minuciosas sobre cada resultado	Todo o time envolvido no projeto, todos Gerentes de Projetos, Gerente da Organização de Software e demais times de desenvolvimento de sistemas da organização	Analista de Sistemas: Apuração e Divulgação; Analista de Sistemas, Gerente de Projetos e Time do Projeto: Análise de Resultados	Após a publicação em produção de cada projeto
Relatório Comparativo por Projeto	Prover resultados sumarizados do projeto, em comparação com médias de projetos semelhantes	Todo o time envolvido no projeto, todos Gerentes de Projetos, Gerente da Organização de Software e demais times de desenvolvimento de sistemas da organização	Analista de Sistemas: Apuração e Divulgação; Analista de Sistemas, Gerente de Projetos e Time do Projeto: Análise de Resultados	Após a publicação em produção de cada projeto
Relatório Evolucionário da Organização	Prover informações históricas da organização quanto ao seu desempenho	Toda a organização de software	Analista de Sistemas: Apuração e Divulgação; Analista de Sistemas, Gerentes de Projetos e Gerente da Organização: Análise de Resultados	Bimestral

4.6. INTEGRAÇÃO DO PROGRAMA DE MEDIÇÃO EM UM CONTEXTO DE DESENVOLVIMENTO DE SISTEMAS

Nos tópicos anteriores deste capítulo foram definidos os objetivos, dados, métricas, relatórios e responsabilidades do programa de medição, concluindo assim a sua definição. Uma vez definido o programa, este tópico aborda a sua integração em um contexto de desenvolvimento de sistemas, resumindo e ilustrando (através de diagramas de atividades da UML 2.0) o programa de medição definido.

Como descrito anteriormente neste estudo, a abordagem de programa de medição apresentada não é restrita para implementação em um processo de desenvolvimento de sistemas em específico, sendo passível de aplicação a qualquer organização de *software* que se encaixe nas premissas definidas por esta abordagem. Tais premissas consideram que determinados papéis, atividades e ferramentas estejam presentes na organização de *software*, o que define um contexto de desenvolvimento de sistemas.

Para ilustrar a integração do programa de medição proposto em um contexto de desenvolvimento de sistemas, a Figura 8 a seguir apresenta todas as macro-

atividades provenientes do programa de medição definidas ao longo da sua especificação feita nos tópicos anteriores, em conjunto com as demais macro-atividades realizadas pela organização ao longo de um fluxo de construção de *software*. O objetivo é apresentar, em linhas gerais, como o programa de medição abordado se integra em tal fluxo já realizado pela organização.

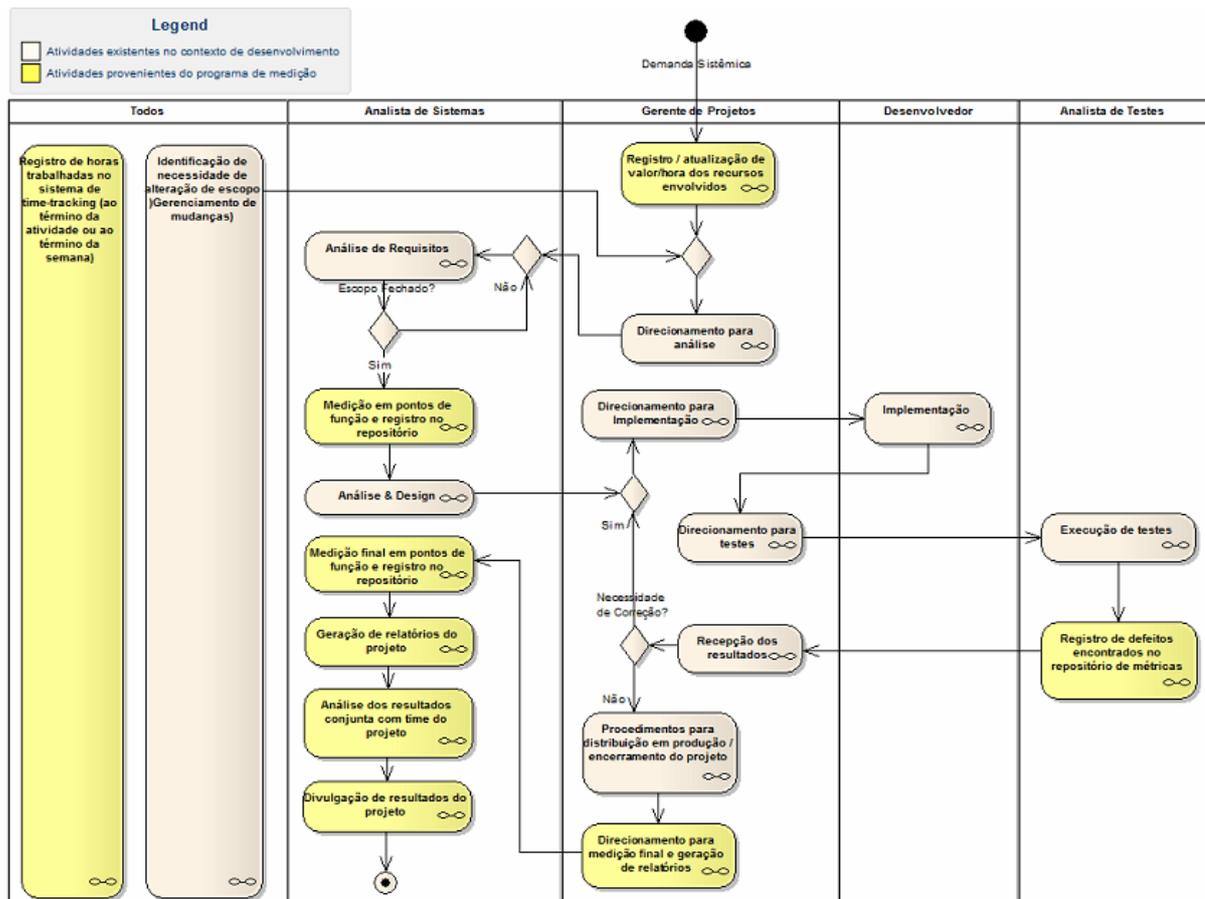


Figura 8 - Definição do Programa de Medição - Integração do programa em um contexto de desenvolvimento de sistemas

Por não fazer parte do fluxo de desenvolvimento de sistemas da organização, a apuração bimestral do relatório evolucionar da organização, o qual é definido no tópico 4.5.3, é ilustrado separadamente do fluxo principal, através da figura 9 a seguir. A ilustração apresenta o fluxo das macro-atividades envolvidas no processo de geração do relatório e as responsabilidades por cada atividade.

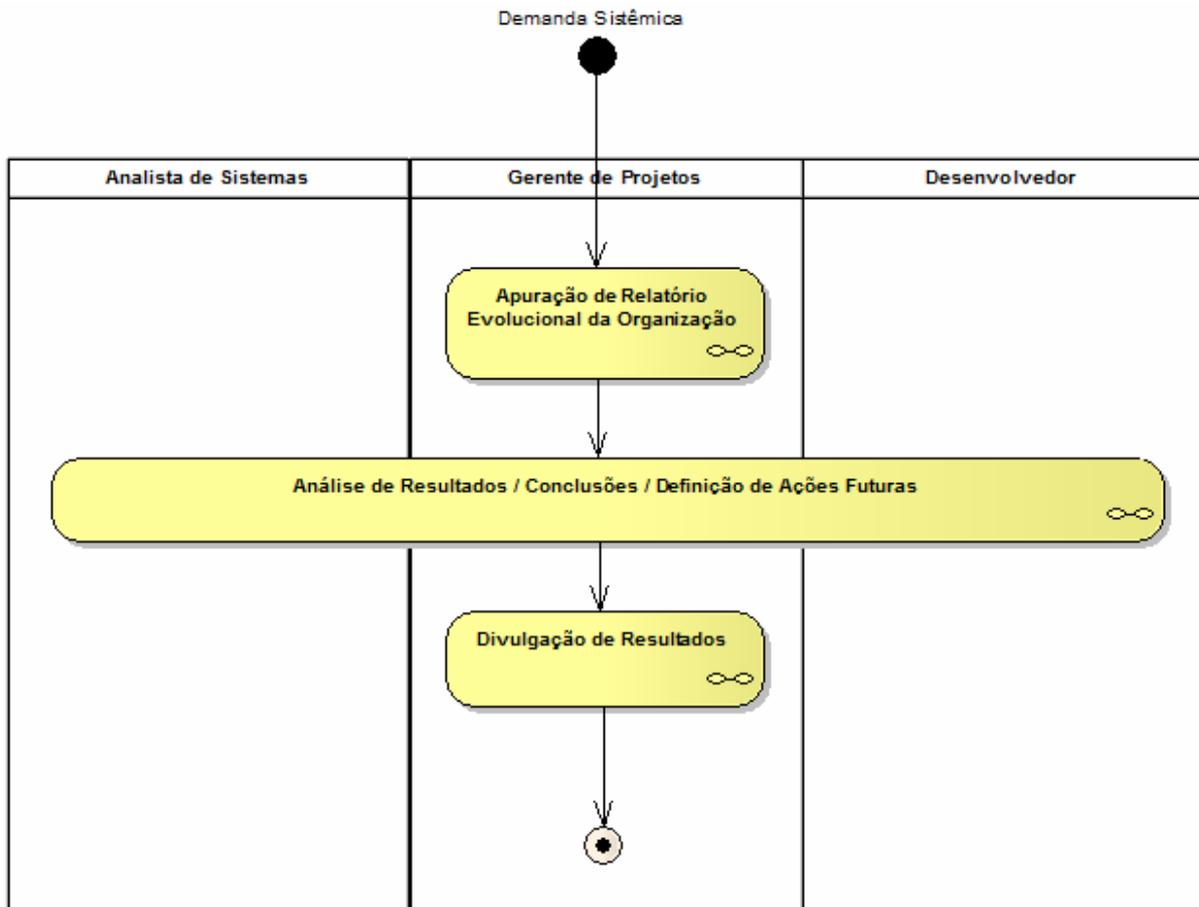


Figura 9 - Definição do Programa de Medição - Fluxo para geração bimestral do relatório evolucionar da organização

5. CONCLUSÕES E TRABALHOS FUTUROS

5.1. CONCLUSÃO

Este estudo abordou o cenário da crescente necessidade por soluções sistêmicas com cada vez mais qualidade e em menor tempo e custo possível, fatores estes atualmente decisivos para o apoio estratégico de organizações ou mesmo sua sobrevivência. Visando suprir tal necessidade, objetivou-se com este trabalho, o estudo de medições, métricas e programas de medição, a fim de se verificar se poderiam apoiar no fornecimento de informações quantitativas e consistentes sobre a organização de *software* e seus produtos, a fim de possibilitar a avaliação de resultados e tomada de decisões baseada em fatos, visando o aperfeiçoamento contínuo do processo de desenvolvimento de sistemas e seus produtos, possibilitando assim, suprir as necessidades descritas.

5.2. RELAÇÃO COM OS OBJETIVOS INICIAIS

Visando o estabelecimento de um programa de medição baseado em análise de pontos de função, o estudo dividiu-se em três etapas. A primeira etapa consistiu no estudo de métricas e medições e dentre elas a análise de pontos de função. A segunda etapa consistiu no estudo de programas de medição e a terceira na proposição de um programa de medição, atingindo assim os objetivos iniciais enunciados deste trabalho.

5.3. VALIDADE DA HIPÓTESE

Pôde-se concluir com este estudo que a análise de pontos de função é uma medição extremamente adequada para composição em programas de medição, pois, é independente de tecnologia, processo ou metodologia, o que possibilitou que fosse amplamente utilizada pelo mercado. Tal fato permite que análises comparativas de resultados entre organizações sejam feitas, sendo este um fator

muito relevante para um programa de medição, uma vez que possibilita saber a posição da organização em relação a organizações semelhantes. Além disso, verificou-se que a análise de pontos de função é normatizada por um órgão internacional, o IFPUG, o qual é responsável por garantir a continuidade e evolução da técnica.

Verificou-se também com este estudo, que programas de medição podem ajudar organizações a coletar e analisar dados sobre si próprias, permitindo saber o quão efetivo é o seu desenvolvimento de sistemas. Tais análises não só permitem que um gerenciamento e tomada de decisões baseada em fatos seja implantado na organização, como também permitem que pontos falhos sejam identificados e corrigidos e que pontos positivos sejam mantidos e promovidos para toda a organização. Concluí-se que tais atividades possibilitam a organização melhorar seus processos e produtos, desenvolvendo-os cada vez com mais qualidade e produtividade, o que por consequência permite a utilização de recursos com maior eficiência. Tais características vêm a confirmar a hipótese enunciada deste estudo.

5.4. TRABALHOS FUTUROS

A seguir, são listadas possíveis evoluções deste trabalho.

5.4.1. Estudo de caso da implementação do programa de medição proposto

Este trabalho limitou-se a propor um programa de medição. Uma possível evolução é a implementação efetiva do programa proposto, a análise e documentação de resultados obtidos e, se necessário, a sua adequação.

5.4.2. Estudo de ferramentas disponíveis para apoio no programa de medição proposto

O trabalho também limitou-se a descrever os tipos de ferramentas necessárias, não apresentando ferramentas específicas, ou mesmo realizando estudos sobre elas. Outra possível evolução é a análise de ferramentas de medições, métricas e repositórios de métricas existentes no mercado ou mesmo a proposição de uma ferramenta customizada para apoio em programas de medição.

5.4.3. Customização de um processo de desenvolvimento de sistemas com o programa de medição proposto

O estudo não se restringiu a um processo de desenvolvimento de sistemas específico. Uma possível evolução é a customização de um determinado processo de desenvolvimento de sistemas, como o RUP, MSF, XP, Scrum, etc., ou mesmo um processo já customizado para uma organização, integrando o programa de medição proposto.

REFERÊNCIAS

Aguar, Mauricio. BFPUG. [Online] [Citado em: 2008 de 04 de 21.] [www.bfpug.com.br/artigos/UCP/Aguar-Pontos de Funcao ou Pontos por Caso de Uso.pdf](http://www.bfpug.com.br/artigos/UCP/Aguar-Pontos%20de%20Funcao%20ou%20Pontos%20por%20Caso%20de%20Uso.pdf).

Andrade, Ediméia Leonor Pereira de. 2004. *Pontos de Caso de Uso e Pontos de Função na gestão de estimativa de tamanho de projetos de software orientados a objetos.* Universidade Católica de Brasília. Brasília : s.n., 2004. Master Thesis.

Budag, Mônica. 2007. *Desenvolvimento de um Processo Baseado em Métrica para Estimar Esforço em um Projeto de Implantação de Software.* Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau. Blumenau : s.n., 2007.

Dekkers, Carol A. 2002. How and When Can Functional Size Fit with a Measurement Program? [A. do livro] International Function Point Users Group. *IT measurement: practical advice from the experts.* Indianapolis : Pearson Education, Inc., 2002, pp. 161-169.

Demarco, Tom. 1989. *Controle de Projetos de Software.* Rio de Janeiro : Campus, 1989.

Gates, Bill. 1999. *A empresa na velocidade do pensamento: com um sistema nervoso digital.* São Paulo : Companhia de letras, 1999. p. 11.

Gill, Antonio Carlos. 2007. *Como elaborar projetos de pesquisa.* 4. São Paulo : Atlas, 2007.

Holmes, Lori. 2002. Measurement Program Implementation Approaches. [A. do livro] International Function Point Users Group. *IT measurement: practical advice from the experts.* Indianapolis : Pearson Education, Inc., 2002, pp. 97-110.

IEEE. 1993. *Standard Glossary of Software Engineering Terminology.* s.l. : IEEE, 1993.

IFPUG. 2000. *Function Point Counting Practices Manual*. Princeton Junction : International Function Point Users Group, 2000.

Jones, Capers. 2002. The Expanding Roles of Function Point Metrics. [A. do livro] International Function Point Users Group. *IT measurement: practical advice from the experts*. Indianapolis : Pearson Education, Inc., 2002, pp. 3-30.

Kroll, Per e Kruchten, Philippe. 2003. *The Rational Unified Process Made Easy: a practioner's guide to the RUP*. Boston : Pearson Education, 2003.

Minkiewicz, Arlene F. 2002. Benchmarking. [A. do livro] International Function Point Users Group. *IT Measurement: practical advice from the experts*. Indianapolis : Pearson Education Inc., 2002, pp. 113-124.

Pressman, Roger S. 2006. *Engenharia de Software*. 6. São Paulo : Mc Graw-Hill, 2006.

Putnam, Lawrence H. e Ware, Myers. 2002. The Core of Software Planning. [A. do livro] International Function Point Users Group. *IT Measurement: practical advice from the experts*. Indianapolis : Pearson Education Inc, 2002, pp. 53-66.

Russac, Janet. 2002. Cheaper, Better, Faster: A Measurement Programa That Works. [A. do livro] International Function Point Users Group. *IT measurement: practical advice from the experts*. Indianapolis : Pearson Education, Inc., 2002, pp. 147-157.

Schuster, Carlos. 2006. *Métricas de software com ênfase em análise de pontos de função apresentando estudo de caso na empresa Data Sul S.A.* Sociedade Educacional de Santa Catarina, Instituto Superior Tupy. Joinville : s.n., 2006.

Silveira, Mário Luiz Barroso da. 2002. EDS Brazil Metrics Program: Measuring for Improvement. [A. do livro] International Function Point Users Group. *IT measurement: practical advice from the experts*. Indianapolis : Pearson Education, Inc., 2002, pp. 85-96.

Vazquez, Carlos Eduardo, Simões, Siqueira Guilherme e Albert, Machado Renato. 2003. *Análise de Pontos de função: medição, estimativas e gerenciamento de projetos de software*. 6. São Paulo : Érica, 2003.