



GLOBAL JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY  
Volume 12 Issue 6 Version 1.0 March 2012  
Type: Double Blind Peer Reviewed International Research Journal  
Publisher: Global Journals Inc. (USA)  
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

## Processo De Estimativa De Software Com a Métrica Use Case Points, Pmbok E Rup

By Everton Castelão Tetila (UFGD), Fábio Iaione (UFGD), Ivanir Costa (USP),  
Juliana Queiroz Da Silva Tetila (UEMS) & Mauro De Mesquita Spínola (USP)

*UFGD*

*Abstract* - The competition between organizations that develop software is increasing with the growth of the market in technology of the information, as consequence the organizations has concerned more and more with the improvement of the quality of its products of software, with the effective costs and with the fulfillment of terms of its projects. To obtain these characteristics, the software development trials require a more effective management, with a well definite project plan based in accurate estimative. In that way, this work proposes through bibliographical hoist and field research, create a software estimative trial to support the management of projects during the cycle of development. This trial consists the use case points metric as a tool of estimative, in the estimative management trials proposed by the pmbok and in the estimative of development activities proposed by the rup. The results of the action research demonstrate that, in average, the software estimative trial with use case points is 4.3% more precise than the traditional estimative metrics (function points and use case points) without estimative formal trial.

*Keywords* : *metrics of software; Use Case Points; PMBOK; RUP.*

*GJCST Classification*: *D.2.8*



*Strictly as per the compliance and regulations of:*



© 2012. Everton Castelão Tetila (UFGD), Fábio Iaione (UFGD), Ivanir Costa (USP), Juliana Queiroz Da Silva Tetila (UEMS) & Mauro De Mesquita Spínola (USP). This is a research/review paper, distributed under the terms of the Creative Commons Attribution-Noncommercial 3.0 Unported License <http://creativecommons.org/licenses/by-nc/3.0/>), permitting all non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

# Processo De Estimativa De Software Com a Métrica Use Case Points, Pmbok E Rup

Everton Castelão Tetila (UFGD) <sup>α</sup>, Fábio Iaione (UFGD) <sup>σ</sup>, Ivanir Costa (USP) <sup>ρ</sup>, Juliana Queiroz Da Silva Tetila (UEMS) <sup>ω</sup>, Mauro De Mesquita Spínola (USP) <sup>¥</sup>

**Resumo** - A competição entre organizações que desenvolvem software vem aumentando com o crescimento do mercado de TI, como consequência as organizações têm se preocupado cada vez mais com a melhoria da qualidade dos produtos de software, com os custos efetivos e com o cumprimento dos prazos de seus projetos. Para obter estas características, os processos de desenvolvimento de softwares vêm exigindo um gerenciamento mais efetivo, com um plano de projeto bem definido, baseado em estimativas mais precisas. Nesse sentido, este trabalho propõe, por meio de levantamento bibliográfico e pesquisa de campo, criar um processo de estimativa de software para apoiar a gerência de projetos durante o ciclo de desenvolvimento. Este processo consiste no uso da métrica Use Case Points como ferramenta de estimativa, nos processos de gestão de estimativas do PMBOK e nas atividades de estimativas do RUP. Os resultados da pesquisa demonstraram que, em média, o processo de estimativa de software com Use Case Points é 4,3% mais preciso que as métricas de estimativas tradicionais (Function Points e Use Case Points) sem processo formal de estimativas.

**Palavras-chaves** : métricas de software; Use Case Points; PMBOK; RUP.

**Abstract** - The competition between organizations that develop software is increasing with the growth of the market in technology of the information, as consequence the organizations has concerned more and more with the improvement of the quality of its products of software, with the effective costs and with the fulfillment of terms of its projects. To obtain these characteristics, the software development trials require a more effective management, with a well definite project plan based in accurate estimative. In that way, this work proposes through bibliographical hoist and field research, create a software estimative trial to support the management of projects during the cycle of development. This trial consists the use case points metric as a tool of estimative, in the estimative management trials proposed by the pmbok and in the estimative of development activities proposed by the rup. The results of the action research demonstrate that, in average, the software estimative trial with use case points is 4.3% more precise than the traditional estimative metrics (function points and use case points) without estimative formal trial.

**Keywords** : metrics of software; Use Case Points; PMBOK; RUP.

Author <sup>α</sup> : E-mail : [evertontetila@ufgd.edu.br](mailto:evertontetila@ufgd.edu.br)

Author <sup>σ</sup> : E-mail : [fabio.iaione@ufgd.edu.br](mailto:fabio.iaione@ufgd.edu.br)

Author <sup>ρ</sup> : E-mail : [icosta11@terra.com.br](mailto:icosta11@terra.com.br)

Author <sup>ω</sup> : E-mail : [ju@uems.br](mailto:ju@uems.br)

Author <sup>¥</sup> : E-mail : [mauro@spinola.eng.br](mailto:mauro@spinola.eng.br)

## 1. INTRODUÇÃO

Conforme pesquisa da Associação para Promoção da Excelência do Software Brasileiro (SOFTEX), o setor de TI encerrou 2009 com crescimento de 9,3% e uma receita anual aproximada de R\$ 52,8 bilhões. Para 2010, a entidade que desenvolve pesquisas para o Ministério da Ciência e Tecnologia, estima um movimento de R\$ 57,7 bilhões e uma repetição do nível de crescimento, acima de 9% (SOFTEX, 2009).

Em consequência desta realidade, as organizações que desenvolvem software têm buscado desenvolver sistemas dentro do prazo e do orçamento previstos e com um nível de qualidade adequada. Todas estas características dependem de um gerenciamento de projeto eficiente e eficaz. Nesse gerenciamento, é essencial a adoção de guias e de um plano de projeto que englobe os requisitos de qualidade do produto exigidos pelo cliente e que seja baseado em estimativas precisas de tamanho, esforço, prazos e custos (PMBOK, 2008).

O tamanho do software é um indicador da quantidade de trabalho a ser executado no desenvolvimento de um projeto. Esta dimensão constitui a base para a derivação das estimativas de esforço, custos e prazos necessários para a definição do plano de desenvolvimento do software (CMMI, 2007). Além de subsidiar o planejamento do projeto, a estimativa de tamanho facilita o relacionamento entre cliente e fornecedor, permite o gerenciamento de riscos, o controle do cronograma e possibilita o conhecimento da produtividade da equipe – o que beneficia a gerência e a qualidade dos contratos de projetos de software (PRESSMAN, 2006; SOMMERVILLE, 2007; SWEBOOK, 2004).

A precisão de estimativas de tamanho, no entanto, é dependente de informações que nem sempre estão disponíveis no início dos projetos (exemplo: número de linhas de código, número de operadores e operandos, número de pontos de função). Essas informações são essenciais para realizar as estimativas, pois elas que vão auxiliar a discussão de contratos ou a determinação da viabilidade do projeto, em termos de análise de prazos, custos e recursos.

Uma outra questão refere-se à falta de padrões quando se aplicam as estimativas. Não se tem conhecimento na literatura sobre a existência de um modelo, guia ou processo que auxilie o trabalho de executar estimativas e avalie o seu percentual de erro em relação aos resultados obtidos e que oriente a melhoria das estimativas ao longo do tempo.

Nesse sentido, o presente trabalho define um Processo de Estimativa de Software, baseado na métrica Use Case Points, que auxilie o gerente a administrar o ciclo de desenvolvimento do projeto. Isto permite subsidiar o planejamento, comparar e avaliar estimativas, controlar o projeto com mais segurança e providenciar ações de ajustes no plano e no cronograma. Dessa forma, esperam-se reduzir os problemas de gestão, tais como: altos custos, atrasos no cronograma, insatisfação do cliente, dificuldades de medição do andamento do projeto e quebras de contratos.

## II. METODOLOGIA

Conforme Marconi (2007), a presente pesquisa caracteriza-se quanto à sua natureza como científico original, pois é uma pesquisa realizada pela primeira vez que vem contribuir com novas conquistas e descobertas para a evolução do conhecimento científico.

Um Processo de Estimativa de Software foi proposto para a pesquisa de excelência. Dez etapas foram sugeridas para a composição deste processo. As etapas foram definidas com base nos processos de gestão de estimativas do **PMBOK**, nas atividades de estimativas do **RUP**, e na métrica Use Case Points, utilizada como ferramenta do processo. Para tal, foi investigada a relação existente entre os guias e a métrica e identificadas as ações gerenciais a serem tomadas pelo gerente de projeto durante o Processo de Estimativa de Software.

Quanto aos objetivos, segundo Gil (2010), uma pesquisa pode ser classificada em: exploratória, descritiva ou explicativa.

A pesquisa exploratória, aplicada neste trabalho, envolve o levantamento bibliográfico, o que proporciona maior familiaridade com o problema, a fim de torná-lo mais explícito. “O levantamento bibliográfico é desenvolvido com base em material já elaborado, constituído principalmente de livros e artigos científicos” (GIL, 2010). Esta pesquisa é a base para a obtenção dos conhecimentos científicos e técnicos para desenvolver o Processo de Estimativa de Software.

Marconi (2007) informa que uma pesquisa quanto ao objeto pode ser classificada em: bibliográfica, de laboratório e de campo. Uma pesquisa de campo foi realizada com duas instituições de software brasileiras, uma pública e outra privada, para mostrar os conceitos, experimentar as ações do

processo e relatar a experiência. Os participantes do estudo desta pesquisa são: os gerentes de projetos das instituições e os autores deste trabalho. Os dados coletados (exemplo: número de funções) foram calculados utilizando-se planilhas eletrônicas para estimar os custos dos projetos de software. Os resultados foram analisados com base na experiência pessoal dos participantes do estudo. As instituições, no entanto, não autorizaram a divulgação de seus nomes, nem de seus projetos, por participarem de concorrências em licitações públicas.

## III. MÉTRICAS DE SOFTWARE

Com a necessidade de medidas que informem a eficiência do desenvolvimento de software, diversas métricas ou métodos de medição foram propostos a fim de minimizar os fracassos dos projetos obtidos, principalmente em relação às falhas no cronograma e orçamento previstos. A seção 3.1 apresenta uma das principais métricas de estimativa de tamanho de software existentes no mercado.

### a) *Use Case Points - UCP*

Os Use Case Points (UCP), ou Pontos de Casos de Uso (PCU), foram propostos em 1993, por Gustav Karner, com base nos Function Points (FP), Mark II, e no Modelo de Casos de Uso para determinar estimativas de tamanho de softwares orientado a objetos. Os UCP visam estabelecer uma medida de “tamanho” do software, em PCU, através da quantificação e complexidade das funcionalidades desempenhadas pelo software. Baseia-se na visão do usuário e tem como proposta ser utilizado logo no início do ciclo de desenvolvimento, na fase de definição dos requisitos, com base no modelo de casos de uso.

Nesta métrica, KARNER (1993) substitui alguns fatores técnicos propostos pelos FP, cria os fatores ambientais, propõe uma estimativa de produtividade de 20 homens/hora por PCU e explora a medição da funcionalidade do sistema baseado no modelo de casos de uso.

A contagem dos UCP se dá sobre dois elementos básicos de um sistema modelado: atores e casos de uso. Cada um desses elementos terá um peso na complexidade do sistema, de acordo com o seu nível de influência. O valor total desses pesos determina o valor total dos PCU não-ajustados. No entanto, esse valor deve ser ajustado em relação aos fatores de complexidade técnica e ambiental que refletem funções que afetam a aplicação de maneira geral.

Os fatores de complexidade técnica variam numa escala de 0 a 5, de acordo com o grau de dificuldade do sistema a ser construído: desempenho da aplicação, portabilidade e facilidade de manutenção, são alguns exemplos. Os fatores de complexidade ambiental indicam a eficiência do projeto, numa escala

de 0 a 5, e estão relacionados ao nível de experiência dos profissionais e às condições ambientais e de trabalho, como a capacidade do líder de projeto, a motivação da equipe e a experiência com a aplicação de desenvolvimento.

Após determinar os fatores de complexidade técnica e ambiental, esses fatores devem ser multiplicados pelos PCU não-ajustados. Dessa forma, são determinados os PCU ajustados do sistema que determinarão as estimativas de esforço, prazo e custos do projeto. Uma vantagem evidente da métrica UCP sobre os FP é que ela utiliza um documento essencial em metodologias dirigidas por casos de uso. Neste sentido, destaca-se o RUP, que usa a UML como linguagem padrão para a elaboração da modelagem de softwares orientado a objetos.

Dessa forma, é possível calcular prontamente mudanças nas estimativas do sistema a cada pequena alteração de requisitos, refazendo-se apenas alguns cálculos. Os FP, ao contrário, exigem que novos documentos para o cálculo das estimativas sejam adicionados ao sistema a cada pequena mudança no orçamento, prazo ou requisitos, sendo, dessa forma, pouco flexível às mudanças. Além disso, “os UCP contribuem para a diminuição de algumas dificuldades impostas pelo mercado em relação à resistência de adoção de métricas de estimativa, porque é um método simples, fácil de usar e rápido de se aplicar” (DAMODARAN & WASHINGTON, 2002).

Para estimar o tamanho do software, em PCU, Karner definiu os processos de contagem, disponível em (KARNER, 1993).

#### IV. PROCESSO DE ESTIMATIVA DE SOFTWARE

##### a) Introdução

Esta seção e a seção seguinte descrevem as etapas necessárias para definir o Processo de Estimativa de Software proposto por este trabalho. Dessa forma, para alcançar este objetivo, são necessários os seguintes passos :

- Mapear a relação entre os processos de gestão de estimativas do PMBOK, as atividades de estimativas do RUP e a métrica UCP.
- Identificar as ações gerenciais a serem tomadas pelo gerente de projeto durante o Processo de Estimativa de Software.

A partir do Processo de Estimativa de Software, o gerente terá informações suficientes sobre as estimativas de tempo e custos do projeto, possibilitando o rastreamento do cronograma e do orçamento previstos. Dessa forma, conhecendo a situação do projeto, o gerente poderá tomar decisões de ajustes no Plano de Desenvolvimento do Software, com base nos indicadores do andamento do projeto.

##### b) Mapeamento dos Processos de Gestão de Estimativas do PMBOK com as Atividades de Estimativas do RUP

Esta seção apresenta o mapeamento dos processos de gestão de estimativas do PMBOK com as atividades de estimativas do RUP.

No PMBOK, os processos de gestão de estimativas estão concentrados nas áreas de conhecimento: Gerenciamento do tempo do projeto e Gerenciamento dos custos do projeto. Nos grupos de processos: Planejamento e Controle, conforme destacados no Quadro 1.

GRUPOS DE PROCESSOS / ÁREAS DE CONHECIMENTO	INICIAÇÃO	PLANEJAMENTO	EXECUÇÃO	CONTROLE	ENCERRAMENTO
4. Gerenciamento da integração do projeto	4.1 Desenvolver o termo de abertura do projeto 4.2 Desenvolver a declaração do escopo preliminar do projeto	4.3 Desenvolver o plano de gerenciamento do projeto	4.4 Orientar e gerenciar a execução do Projeto	4.5 Monitorar e controlar o trabalho do projeto 4.6 Controlar mudanças e integração	4.7 Encerrar o projeto
5. Gerenciamento do escopo do projeto		5.1 Planejar o escopo 5.2 Definir o escopo 5.3 Criar EAP		5.4 Verificar o escopo 5.5 Controlar o escopo	

6. Gerenciamento do tempo do projeto		6.1 Definir atividade 6.2 Seqüenciar as atividades 6.3 Estimar os recursos da atividade 6.4 Estimar a duração da atividade 6.5 Desenvolver o Cronograma		6.6 Controlar o cronograma	
7. Gerenciamento dos custos do projeto		7.1 Estimar os custos 7.2 Desenvolver a orçamentação		7.3 Controlar os custos	
8. Gerenciamento da qualidade do projeto		8.1 Planejar a qualidade	8.2 Realizar a garantia da qualidade	8.3 Realizar o controle da qualidade	
9. Gerenciamento dos recursos humanos do projeto		9.1 Planejar os recursos humanos	9.2 Contratar ou mobilizar a equipe do projeto 9.3 Desenvolver a equipe do projeto	9.4 Gerenciar a equipe do projeto	
10. Gerenciamento das comunicações do projeto		10.1 Planejar as comunicações	10.2 Distribuir as informações	10.3 Desenvolver o relatório de desempenho 10.4 Gerenciar as partes interessadas	

Quadro 1: Mapeamento dos processos de gestão de estimativas do PMBOK. Fonte: PMBOK (2008).

Já no RUP, as atividades de estimativas estão concentradas na disciplina Gerenciamento de Projeto, conforme destacada na Figura 1.

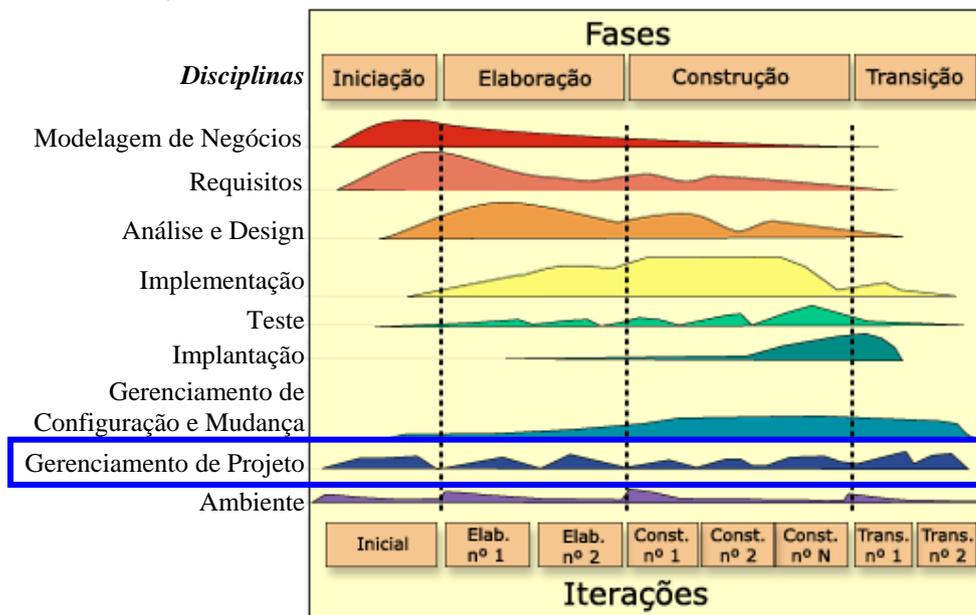


Figura 1: Mapeamento das atividades de estimativas do RUP. Fonte: RUP (2008).

Dessa forma, é possível mapear os processos de gestão de estimativas do PMBOK para as atividades de estimativas do RUP, conforme o Quadro 2. Este mapeamento foi baseado em (CHARBONNEAU, 2004).

PMBOK: Processos de Gestão de estimativas	RUP: Atividades de Estimativas
➤ Estimativa de recursos da atividade	➤ Definir Equipe e Organização do Projeto
➤ Estimativa de duração da atividade	➤ Planejar Fases e Iterações ➤ Desenvolver Plano de Iteração
➤ Estimativa de custos	➤ Planejar Fases e Iterações

Quadro 2 : Mapeamento dos processos de gestão de estimativas do PMBOK para as atividades de estimativas propostas do RUP. Fonte: Adaptado de Charbonneau (2004).

Note que os seguintes processos de gestão do PMBOK: Definir a atividade, Seqüenciar as atividades, Desenvolver o cronograma, Desenvolver a orçamentação, Controlar o cronograma e Controlar os custos não foram mapeados para o Processo de Estimativa de Software (veja o Quadro 1). Isto se deve ao fato de que estes processos não são foco do processo técnico de executar o trabalho de estimativas. Portanto, não serão detalhados neste trabalho.

c) *Processo de Estimativa de Software*

Com base no mapeamento da seção 4.2, esta seção apresenta o Processo de Estimativa de Software. Para a ilustração do mesmo, foi utilizada a representação gráfica do modelo de transformação, disponível em (SLACK et al., 2008).

O modelo de transformação é o método utilizado para representar a produção de bens e ou serviços. Qualquer operação produz bens e ou serviços e isto ocorre através do processo de transformação.

Pode-se descrever toda e qualquer operação através do sistema entrada-transformação-saída. A produção envolve vários recursos chamados de input, ou entrada, que são usados para transformar ou serem transformados em algo que são os outputs, ou saída, de bens e serviços.

O Processo de Estimativa de Software é representado graficamente pelo modelo de transformação, conforme ilustra a Figura 2.



Figura 2 : Representação gráfica do Processo de Estimativa de Software. Fonte: Adaptado de Slack et al. (2008).

Segundo o PMBOK (2008), o escopo do projeto é o input para o processo de estimativa. O escopo do projeto define o trabalho necessário, e apenas o trabalho necessário, para que o projeto seja concluído com sucesso. O modelo de casos de uso é o artefato gerado pelo escopo. Por meio do modelo de

casos de uso e da métrica UCP é possível obter a estimativa de tamanho do software. O tamanho do software, por sua vez, irá derivar as estimativas de esforço, prazo, e custos necessários para o desenvolvimento do projeto (processo de transformação), oferecendo subsídios para distribuir as

atividades entre os membros da equipe, desenvolver o cronograma e determinar o orçamento do projeto.

Contudo, antes de determinar o orçamento do projeto é necessário estimar os recursos (pessoas, equipamentos ou material) utilizados para a realização das atividades. Além disso, uma previsão das variações que possam ocorrer no decorrer do projeto, como uma previsão para a inflação do período, deve ter sido realizada. Somado esses itens, é possível determinar a estimativa do custo global do projeto.

Em relação às restrições (exemplo: o custo não pode exceder R\$ 100.000,00) e prioridades (exemplo: a urgência de um produto) que estão sujeitos a maioria dos projetos, uma variedade de cenários de estimativas podem ser criados à medida que a relação entre tamanho da equipe e prazo é não-linear. Dessa forma, o tamanho da equipe ou o cronograma podem ser ajustados de acordo com restrições e prioridades pré-estabelecidas. Com a variedade de cenários de estimativas possíveis, o gerente de projeto deve selecionar o cenário que melhor se adapta às necessidades do projeto.

No final do projeto, os dados obtidos (exemplo: o tamanho da aplicação em PCU, o esforço de desenvolvimento, os defeitos encontrados) devem ser registrados em uma base de dados mantida pela organização. Estes dados serão úteis para compará-los com outras medidas obtidas de projetos anteriores. Com isso, índices de produtividade, como homens/hora por PCU, R\$ por PCU; e de qualidade, como defeitos por PCU, erros por PCU, podem ser avaliados e tendências podem ser geradas.

Por fim, é preciso verificar se as estimativas realizadas no início do projeto estão dentro de um percentual de erro aceitável. Para isto, o gerente deve comparar as estimativas realizadas no início do projeto com os resultados obtidos no final do projeto (output). “Um percentual de erro inferior a 10% (dez por cento) é aceitável para essa relação” (TETILA et al., 2006). Caso o percentual de erro seja maior que 10% (em valor absoluto), avaliações devem ser tomadas para que estimativas com maior precisão possam ser realizadas nos próximos projetos. Consistência do Índice de Produtividade, padronização dos PCU, distribuição de atividades e conversão de PCU em prazo, são os principais itens de avaliação.

## V. PESQUISA DE CAMPO

Esta seção apresenta a pesquisa de campo realizada com duas conceituadas instituições brasileiras, uma pública e outra privada.

Na instituição privada, um projeto de software foi estimado em dezembro de 2004 por Ivanir Costa. Já na instituição pública, dois outros projetos foram estimados por Everton Castelhão Tetila – um em fevereiro de 2006 e outro em setembro de 2006.

Três formas de aplicação de métricas de estimativas foram utilizadas para estimar o custo de cada projeto de software. Primeiro, cada projeto foi estimado utilizando a métrica Function Points. Logo após, cada projeto foi estimado utilizando a métrica Use Case Points. Finalmente, cada projeto foi estimado utilizando o Processo de Estimativa de Software. As estimativas de cada uma das métricas utilizadas foram comparadas com os resultados obtidos nos finais dos projetos. Assim, o percentual de erro entre a estimativa inicial e resultado obtido foi estabelecido. Os erros calculados para cada uma das métricas utilizadas foram comparados entre si. Isto permitiu compreender melhor o comportamento de cada projeto e avaliar o desempenho das métricas utilizadas.

### a) Análise e interpretação

No Projeto 1, realizado pela Instituição privada, os erros de custos obtidos pelas métricas tradicionais (FP e UCP) foram de -9,91% e -4,47%, respectivamente. Portanto, são aceitáveis para a relação entre estimativas iniciais e resultados obtidos “um percentual de erro inferior a 10% (dez por cento) é aceitável para essa relação” (TETILA et al., 2006). Todavia, o erro calculado pode ser expressivo quando se trata de um projeto de grande porte. Isto porque as estimativas realizadas pelas métricas FP e UCP foram inferiores ao custo final do projeto em R\$ 36.828,00 e R\$ 16.630,00, respectivamente.

O erro obtido pelo Processo de Estimativa de Software foi de -1,69%. Isto, em termos de custo, representa R\$ 6.178,00. Porém, o percentual de erro poderia ser ainda menor, já que ocorreram faltas de membros da equipe que não estavam previstas no planejamento do projeto. Fato que desencadeou um esforço de desenvolvimento ligeiramente acima do esperado.

É razoável supor que o Processo de Estimativa de Software obteve menor erro que as métricas tradicionais (FP e UCP) porque este considera uma possível variação que as estimativas podem sofrer ao longo do projeto (exemplo: inflação), ao contrário das métricas tradicionais que estimam apenas o custo do software em si. A inflação no período do desenvolvimento do Projeto foi de 2,94% (5,7% ao ano). Veja a Tabela 1.

Tabela 1 : Perspectivas para a inflação.

Ano	Inflação ao ano (%)
2004	7,6
2005	5,7
2006	3,1
2007	4,1

Fonte : BCB (2006).

No Projeto 2, realizado pela Instituição pública, os erros de custos obtidos pelas métricas tradicionais (FP e UCP) foram de -1,35% e -5,48%, respectivamente. O erro de custo obtido pelo Processo de Estimativa de Software foi de -2,59%. Portanto, os percentuais de erros das três métricas utilizadas foram inferiores ao limite máximo de 10% (em valor absoluto), aconselhável em (TETILA et al., 2006).

O esforço previsto pelas métricas UCP e Processo de Estimativa de Software, estiveram ligeiramente abaixo da realidade nesse projeto, o que acarretou estimativas menos precisas que a métrica FP. Isto ocorreu porque o Índice de Produtividade (IP) da equipe foi um pouco superior ao IP de 20 homens/hora por PCU proposto por Karner (KARNER, 1993).

Contudo, o erro de custo obtido pelo Processo de Estimativa de Software foi menor que o erro de custo obtido pela métrica UCP. Dois fatores foram determinantes para isso: (1) o Processo de Estimativa de Software considera a variação das estimativas ao longo do projeto: a inflação no período foi de 2,4%; (2) o Processo de Estimativa de Software prevê o custo dos recursos alocados: ocorreu treinamento em J2EE (Java 2 Enterprise Edition) durante a realização do projeto.

Finalmente, no Projeto 3, realizado pela Instituição pública, os erros de custos obtidos pelas métricas UCP e Processo de Estimativa de Software foram de -3,24% e 2,45%, respectivamente. Limites esses também dentro da margem de erro aceitável de 10% (em valor absoluto).

Por outro lado, o erro de custo obtido pela métrica FP foi de -14,22%, ou seja, superior ao tolerado pela margem de erro. Neste caso, quatro itens devem ser avaliados para que estimativas com maior precisão possam ser realizadas nos próximos projetos. Os itens de avaliação são: consistência do IP, padronização dos PCU, distribuição de atividades, e conversão de PCU em prazo (veja a Figura 2).

Em relação ao erro de custo obtido pela métrica FP, três fatores foram determinantes para que esse erro fosse maior que o recomendado: (1) o esforço estimado ficou bem abaixo da realidade: o IP da equipe foi superior ao IP utilizado na métrica; (2) a métrica FP não considera a variação das estimativas ao longo do projeto: durante o desenvolvimento do projeto houve reajustes salariais de três membros da equipe. Além disso, a estimativa de custo deveria ser reajustada em relação à inflação do período, que foi de 2,1%; (3) a métrica FP não prevê o custo dos recursos alocados: para a realização do projeto, foi necessária a aquisição da ferramenta Rational Functional Tester.

#### b) *Discussão dos resultados*

Três formas de aplicação de métricas foram utilizadas para estimar o custo de cada projeto de software. Isto permitiu comparar o desempenho do Processo de Estimativa de Software com o

desempenho das métricas tradicionais (FP e UCP). Esta seção discute os resultados analisados na seção anterior (5.1).

No Projeto 1, realizado pela Instituição privada, presume-se que o Processo de Estimativa de Software obteve menor percentual de erro que as métricas tradicionais (FP e UCP) porque este considerou a inflação do período ao longo do desenvolvimento do projeto, ao contrário das métricas FP e UCP.

O índice de inflação mede, entre outras coisas, a variação geral dos preços e do custo de vida. No período de desenvolvimento do Projeto 1 a inflação foi de 2,94% (5,7% ao ano). Esse índice foi importante para calcular a variação dos preços dos bens consumidos (exemplos: aluguel, energia, telefone, salário) ao longo do projeto.

Já no Projeto 2, realizado pela Instituição Pública, o percentual de erro da métrica FP foi menor que o percentual de erro do Processo de Estimativa de Software. Isto porque o esforço estimado pela métrica FP foi mais preciso que o esforço estimado pelo Processo de Estimativa de Software (o IP estava mais consistente). Todavia, o percentual de erro do Processo de Estimativa de Software foi menor que o percentual de erro da métrica UCP, à medida que o primeiro considerou tanto a inflação do período (3,5% ao ano) quanto o treinamento (J2EE) que ocorreram ao longo do projeto.

Finalmente, no Projeto 3, realizado pela Instituição Pública, é lícito supor que o Processo de Estimativa de Software obteve menor erro que as métricas tradicionais (FP e UCP) porque este considerou, além da inflação do período (3,1%), o reajuste salarial de membros da equipe que ocorreram no decorrer do projeto.

Na ocasião, ocorreu o reajuste salarial de três membros da equipe devido ao plano de cargos e carreira da instituição. Um dos membros da equipe obteve o reajuste salarial de 25% por ter completado um ano de trabalho. Os outros dois membros da equipe obtiveram os reajustes de 12,5% por terem completado três anos trabalhados. Os reajustes, no entanto, não ocorreram no mesmo mês para cada membro da equipe. Situação em que o salário de cada membro foi calculado separadamente. Em média, os salários dos três membros obtiveram um reajuste de 20,83%, ao longo de 4,7 meses do projeto.

Em relação aos projetos pesquisados, é possível concluir que o erro médio obtido pelo Processo de Estimativa de Software é de 2,24%. Logo, a sua precisão média é de 97,76%. Pode-se afirmar com 95% de confiança que o erro percentual médio do Processo de Estimativa de Software está entre o intervalo [1,04%; 3,45%]. Veja a Tabela 2. O Processo de Estimativa de Software com UCP é 4,3% mais preciso que as métricas de estimativas tradicionais (FP e UCP) sem processo formal de estimativas.

*Tabela 2* : Valor absoluto do percentual de erro obtido pelas métricas Function Points, Use Case Points e Processo de Estimativa de Software nos projetos 1, 2 e 3.

9	N	Média	Desvio Padrão	Erro Padrão	Intervalo de confiança para a média de 95%		Mínimo	Máximo
					Limite superior	Limite inferior		
FP	3	8,4933	6,55091	3,78217	-7,7800	24,7667	1,35	14,22
UCP	3	4,3967	1,12180	0,64767	1,6100	7,1834	3,24	5,48
PES	3	2,2433	0,48429	0,27960	1,0403	3,4464	1,69	2,59
<b>Total</b>	9	5,0444	4,31997	1,43999	1,7238	8,3651	1,35	14,22

Admite-se, porém, que o considerável índice de precisão das estimativas obtido pelo Processo de Estimativa de Software pode ter sido influenciado pelas informações obtidas pelos gerentes de projetos.

O Processo de Estimativa de Software foi o único a ser realizado após a conclusão dos projetos. Isto permitiu calcular com exatidão todos os recursos consumidos (exemplos: treinamento, aquisição de ferramenta, reajuste salarial) ao longo do projeto. A situação ideal, no entanto, seria realizar as estimativas nas fases iniciais dos projetos. Nessa situação, certamente tais recursos não seriam previstos com exatidão, ou ainda, poderiam ter sido negligenciados na fase de planejamento do projeto, o que diminuiria, em certo ponto, a precisão das estimativas realizadas pelo Processo de Estimativa de Software.

## VI. CONCLUSÕES E PERSPECTIVAS FUTURAS

Este trabalho apresentou um Processo de Estimativa de Software para padronizar o trabalho de executar estimativas. O Processo de Estimativa de Software foi definido com base nos guias PMBOK, RUP e na métrica UCP. A partir do cruzamento dos guias com a métrica UCP, verificou-se que ambos são capazes de se relacionar, podendo ser usados de forma combinada.

Uma pesquisa de campo foi realizada com três projetos de software de duas instituições brasileiras, uma pública e outra privada. Isto foi fundamental para comparar o desempenho do Processo de Estimativa de Software com o desempenho das métricas tradicionais (FP e UCP). Os resultados da pesquisa de campo demonstram que, em média, o Processo de Estimativa de Software com UCP é 4,3% mais preciso que as métricas de estimativas tradicionais (FP e UCP) sem processo formal de estimativas.

Em relação aos resultados obtidos nos projetos, três elementos foram determinantes para o aumento da precisão nas estimativas: (1) a Previsão das Variações de Estimativas, que foi fundamental para calcular a variação geral dos preços ao longo do projeto; (2) a Estimativa dos Recursos: que conseguiu prever os preços dos recursos alocados para a

realização das atividades; (3) o Índice de Produtividade (IP), que, quando consistente, garantiu que a estimativa de esforço fosse mais precisa, contribuindo para o menor percentual de erro nas medições.

Como trabalhos futuros, este trabalho visa adaptar ações da gestão de estimativas ao Processo de Estimativa de Software apresentado. Para isto, será necessário mapear os processos de gestão propostos pelo PMBOK: Controlar o cronograma e Controlar os custos, para as atividades de gestão de estimativas propostas pelo RUP. Além disso, um método de avaliação e melhoria de estimativas para descobrir não-conformidades nos processos utilizados pela organização seria útil ao processo.

## REFERÊNCIAS BIBLIOGRÁFICAS

1. BCB. BANCO CENTRAL DO BRASIL. Setor externo, Inflação e Atividade: Perspectivas para 2006. *Banco Central do Brasil*. 06 abr. 2006. Disponível em: <<http://www.bcb.gov.br/?APRES2006012>>. Acesso em: 30 maio 2007.
2. CHARBONNEAU, S. Software Project Management - A Mapping between RUP and the PMBOK. *Xelation Software Corporation*, [S.l.], May. 2004. Disponível em: <<http://www.ibm.com/developerworks/rational/libra-ry/4721.html>>. Acesso em: 17 agosto 2007.
3. CMMI. CAPABILITY MATURITY MODEL INTEGRATION. CMMI: *Integração dos Modelos de Capacitação e Maturidade de Sistemas*. 6. ed., Rio de Janeiro: Ciência Moderna, 2007. 292 p.
4. DAMODARAN, M.; WASHINGTON, A. *Estimation using use case points*. University of Houston-Victoria, Victoria, 2002.
5. GIL, A. C. *Como elaborar projetos de pesquisa?* 5.ed., São Paulo: Atlas, 2010. 200 p.
6. KARNER, G. Use Case Points: resource estimation for Objectory projects. *Objective Systems SF AB*, [S.l.], Set. 1993.
7. MARCONI, M. A. *Metodologia do Trabalho Científico*. 7. ed., São Paulo: Atlas, 2007. 255 p.
8. PMBOK. PROJECT MANAGEMENT BODY OF KNOWLEDGE. *A guide to the project management body of knowledge (PMBOK Guide)*. 4th, Newtown

- Square: Project Management Institute, Dez. 2008. 386 p.
9. PRESSMAN, R. S. *Engenharia de software*. 6. ed., São Paulo: McGraw-Hill, 2006. 720 p.
  10. RUP. RATIONAL UNIFIED PROCESS. IBM *Rational Unified Process Reference and Certification Guide: Solution Designer*. [S.l.]: IBM Press, 2008. 336 p.
  11. SLACK, N.; CHAMBERS, S.; JOHNSTON, T.; *et al.* *Gerenciamento de Operações e de Processos: Princípios e Práticas de Impacto Estratégico*. Porto Alegre: Bookman, 2008. 552 p.
  12. SOFTEX. ASSOCIAÇÃO PARA PROMOÇÃO DA EXCELÊNCIA DO SOFTWARE BRASILEIRO. Software e Serviços de TI: A Indústria Brasileira em Perspectiva. *Associação para promoção da excelência do software brasileiro*, Campinas, nov. 2009. Disponível em: [http://publicacao.observatorio.softex.br/\\_publicacoes/arqui-vos/resumo/Resumo\\_Executivo.PDF](http://publicacao.observatorio.softex.br/_publicacoes/arqui-vos/resumo/Resumo_Executivo.PDF). Acesso em: 5 janeiro 2010.
  13. SOMMERVILLE, I. *Engenharia de software*. 8. ed., São Paulo: Addison-Wesley, 2007. 568 p.
  14. SOWEBOK. *Guide to the software engineering body of knowledge*. Los Alamitos: IEEE Computer Society, v. 2004. 202 p.
  15. TETILA, E. C.; COSTA, I.; SPÍNOLA, M. M. Estimativa de software combinando métricas: um estudo de caso. In: SIMPÓSIO DE ADMINISTRAÇÃO DA PRODUÇÃO, LOGÍSTICA E OPERAÇÕES INTERNACIONAIS (SIMPOI), 9, 2006. São Paulo. *Anais*. São Paulo: Fundação Getúlio Vargas. 29-31 ago. 2006.