

1 Dr.EVERTON CASTELAO TETILA¹ and MAURO DE MESQUITA SPANOLA²

2 ¹ UFGD

3 *Received: 9 April 2012 Accepted: 30 April 2012 Published: 15 May 2012*

4

5 Abstract

6 The competition between organizations that develop software is increasing with the growth of
7 the market in technology of the information, as consequence the organizations has concerned
8 more and more with the improvement of the quality of its products of software, with the
9 effective costs and with the fulfillment of terms of its projects. To obtain these characteristics,
10 the software development trials require a more effective management, with a well definite
11 project plan based in accurate estimative. In that way, this work proposes through
12 bibliographical hoist and field research, create a software estimative trial to support the
13 management of projects during the cycle of development. This trial consists the use case points
14 metric as a tool of estimative, in the estimative management trials proposed by the pmbok
15 and in the estimative of development activities proposed by the rup. The results of the action
16 research demonstrate that, in average, the software estimative trial with use case points is 4.3

17

18 **Index terms**— metrics of software; Use Case Points; PMBOK; RUP.

19 Processo De Estimativa De Software Com a Métrica Use Case Points, Pmbok E Rup Everton Castelão Tetila
20 (UFGD) ? , Fábio Iaione (UFGD) ? , Ivanir Costa (USP) ? , Juliana Queiroz Da Silva Tetila (UEMS) ? ,
21 Mauro De Mesquita Spínola (USP) ¥ Resumo -A competição entre organizações que desenvolvem software vem
22 aumentando com o crescimento do mercado de TI, como consequência as organizações têm se preocupado cada
23 vez mais com a melhoria da qualidade dos produtos de software, com os custos efetivos e com o cumprimento
24 dos prazos de seus projetos. Para obter estas características, os processos de desenvolvimento de softwares vêm
25 exigindo um gerenciamento mais efetivo, com um plano de projeto bem definido, baseado em estimativas mais
26 precisas. Nesse sentido, este trabalho propõe, por meio de levantamento bibliográfico e pesquisa de campo, criar
27 um processo de estimativa de software para apoiar a gerência de projetos durante o ciclo de desenvolvimento.
28 Este processo consiste no uso da métrica Use Case Points como ferramenta de estimativa, nos processos de gestão
29 de estimativas do PMBOK e nas atividades de estimativas do RUP. Os resultados da pesquisa demonstraram
30 que, em média, o processo de estimativa de software com Use Case Points é 4,3% mais preciso que as métricas
31 de estimativas tradicionais (Function Points e Use Case Points) sem processo formal de estimativas.

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

33 Abstract -The competition between organizations that develop software is increasing with the growth of the
34 market in technology of the information, as consequence the organizations has concerned more and more with
35 the improvement of the quality of its products of software, with the effective costs and with the fulfillment of
36 terms of its projects.

37 To obtain these characteristics, the software development trials require a more effective management, with a
38 well definite project plan based in accurate estimative. In that way, this work proposes through bibliographical
39 hoist and field research, create a software estimative trial to support the management of projects during the
40 cycle of development. This trial consists the use case points metric as a tool of estimative, in the estimative
41 management trials proposed by the pmbok and in the estimative of development activities proposed by the
42 rup. The results of the action research demonstrate that, in average, the software estimative trial with use case
43 points is 4.3% more precise than the traditional estimative metrics (function points and use case points) without
44 estimative formal trial.

45 I.

1 INTRODUÇÃO

onforme 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). 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.

2 II.

3 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.

4 III.

5 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.

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". 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. 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.

7 PESQUISA DE CAMPO

163

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

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

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

8 a) Análise e interpretação

177

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

184 O erro obtido pelo Processo de Estimativa de Software foi de -1,69%. Isto, em termos de custo, representa R\$
185 6.178,00. Porém, o percentual de erro poderia ser ainda menor, já que ocorreram faltas de membros da equipe
186 que não estavam previstas no planejamento do projeto. Fato que desencadeou um esforço de desenvolvimento
187 ligeiramente acima do esperado.

188 É razoável supor que o Processo de Estimativa de Software obteve menor erro que as métricas tradicionais
189 (FP e UCP) porque este considera uma possível variação que as estimativas podem sofrer ao longo do projeto
190 (exemplo: inflação), ao contrário das métricas tradicionais que estimam apenas o custo do software em si. A
191 inflação no período do desenvolvimento do Projeto foi de 2,94% (5,7% ao ano). Veja a Tabela 1. No Projeto 2,
192 realizado pela Instituição pública, os erros de custos obtidos pelas métricas tradicionais (FP e UCP) foram de
193 -1,35% e -5,48%, respectivamente. O erro de custo obtido pelo Processo de Estimativa de Software foi de -2,59%.
194 Portanto, os percentuais de erros das três métricas utilizadas foram inferiores ao limite máximo de 10% (em valor
195 absoluto), aconselhável em (TETILA et al., 2006).

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

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

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

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

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

9 b) Discussão dos resultados

219

220 Três formas de aplicação de métricas foram utilizadas para estimar o custo de cada projeto de software. Isto
221 permitiu comparar o desempenho do Processo de Estimativa de Software com o desempenho das métricas
222 tradicionais (FP e UCP). Esta seção discute os resultados analisados na seção anterior ??5.1).

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

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

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

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

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

244 Em relação aos projetos pesquisados, é possível concluir que o erro médio obtido pelo Processo de Estimativa
245 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
246 percentual médio do Processo de Estimativa de Software está entre o intervalo [1,04%; 3,45%]. Veja a Tabela 2.
247 O Processo de Estimativa de Software com UCP é 4,3% mais preciso que as métricas de estimativas tradicionais
248 (FP e UCP) sem processo formal de estimativas. Admitese, porém, que o considerável índice de precisão das
249 estimativas obtido pelo Processo de Estimativa de Software pode ter sido influenciado pelas informações obtidas
250 pelos gerentes de projetos.

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

257 10 VI.

258 11 CONCLUSÕES E PERSPECTIVAS FUTURAS

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

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

268 Em relação aos resultados obtidos nos projetos, três elementos foram determinantes para o aumento da precisão
269 nas estimativas: (1) a Previsão das Variações de Estimativas, que foi fundamental para calcular a variação geral
270 dos preços ao longo do projeto; (2) a Estimativa dos Recursos: que conseguiu prever os preços dos recursos
271 alocados para a realização das atividades; (3) o Índice de Produtividade (IP), que, quando consistente, garantiu
272 que a estimativa de esforço fosse mais precisa, contribuindo para o menor percentual de erro nas medições.

273 Como trabalhos futuros, este trabalho visa adaptar ações da gestão de estimativas ao Processo de Estimativa
274 de Software apresentado. Para isto, será necessário mapear os processos de gestão propostos pelo PMBOK:
275 Controlar o cronograma e Controlar os custos, para as atividades de gestão de estimativas propostas pelo RUP.
276 Além disso, um método de avaliação e melhoria de estimativas para descobrir nãoconformidades nos processos
277 utilizados pela organização seria útil ao processo. ^{1 2 3}

¹© 2012 Global Journals Inc. (US) Global Journal of Computer Science and Technology Volume XII Issue VI
Version I

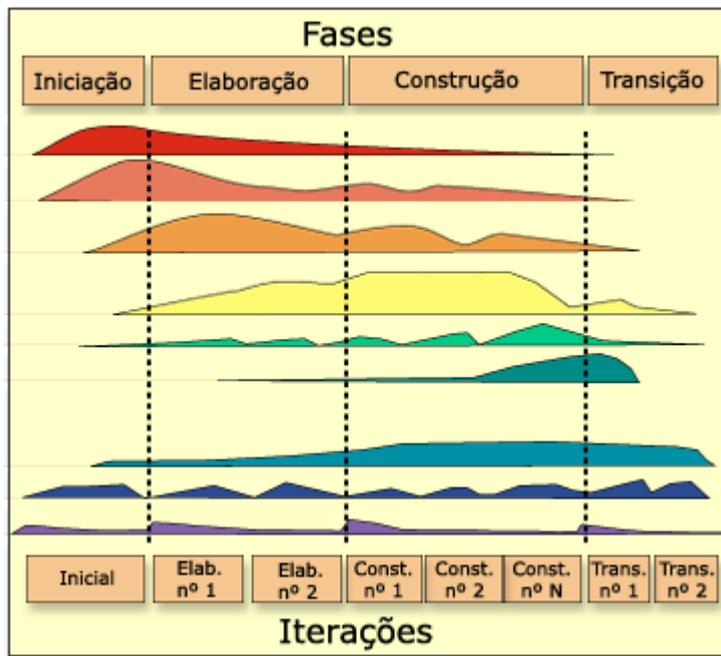
²© 2012 Global Journals Inc. (US)

³March



2

Figure 1: Figura 2 :



1

Figure 2: Tabela 1 :



Figure 3:

6. Gerenciamento do

6.1 Definir atividade

IV. PRO DE ES-TI-MA-TIVA DE

tempo do projeto

6.2 Se-qüen-ciar as

6.6 Con-tro-lar o SOF cronogram

Quadro 2 :

7. Gerenciamento dos custos do projeto

atividades 6.3 Estimar os recur-a) Introdução Esta seção e a seção seguinte Cronogramam mapear a relação entre os p

7.1 estimativas do PMBOK, as a

Esti-mar os cus-orçamentação

7.2 De-sen-volver a tos

8. Gerenciamento da qualidade do projeto

8.1 Planejar a quali- A partir do dade o gerente terá informações su estimativas de tempo e custos

se aplicar” 9.1 Planejar os re-(DAMODARAN & WASHINGTON, 2002). 9. Gerenciamento dos recursos hu Para estimar o tamanho do software, em PCU, Karner definiu os processos de contagem, disponível

atividades de estimativas do l

No PM-BOK os pro-ces-sos de gestã de 10.4 Gere-ciar

em (KARNER, 1993).

estimativas estão concentrado conhecimento: Gerenciamento Gerenciamento dos custos do processos: Planejamento e Co

2

| 9 | N | Média | Desvio Padrão | Erro Padrão | Intervalo de confiança para a média de 95% superior | Limite superior | Mínimo | Máximo |
|-------|---|--------|------------------|----------------|---|--------------------|--------|--------|
| 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 |
| Total | 9 | 5,0444 | 4,31997 | 1,43999 | 1,7238 | 8,3651 | 1,35 | 14,22 |

Figure 5: Tabela 2 :

-
- 278 [Pmbok et al.] *A guide to the project management body of knowledge (PMBOK Guide)*, Pmbok , Project , Body
279 , Knowledge . Newtown. 4.
- 280 [Gil ()] ‘Como elaborar projetos de pesquisa? 5’. A C Gil . *São Paulo: Atlas*, 2010. 200 p.
- 281 [Sommerville ()] *Engenharia de software*, I Sommerville . 2007. São Paulo: Addison-Wesley. 8.
- 282 [Pressman ()] *Engenharia de software. 6*, R S Pressman . 2006. São Paulo: McGraw-Hill. 720.
- 283 [Damodaran and Washington ()] *Estimation using use case points*, M Damodaran , A Washington . 2002.
284 Victoria. University of Houston-Victoria
- 285 [Tetila et al. ()] *Estimativa de software combinando métricas: um estudo de caso*, E C Tetila , I Costa , M. M ;
286 Simpósio De Administração Da Spínola , Logística E Operações Produção , Internacionais . 2006.
- 287 [Sowebok ()] *Guide to the software engineering body of knowledge*, Sowebok . 2004. Los Alamitos: IEEE Computer
288 Society. 202.
- 289 [Rational and Process ()] *IBM Rational Unified Process Reference and Certification Guide: Solution Designer*,
290 Rup Rational , Process . 2008. IBM Press. 336. (S.l.)
- 291 [Bcb et al. ()] *Inflação e Atividade: Perspectivas para*, Bcb , Banco , Do Brasil , Setor . <<http://www.bcb.gov.br/?APRES2006012>>. Acesso em 2006. 2006. p. . (Banco Central do Brasil. 06 abr)
- 293 [Marconi and Metodologia Do Trabalho Científico ()] M A Marconi , Metodologia Do Trabalho Científico . *São*
294 *Paulo: Atlas*, 2007. 255.
- 295 [Maturity et al. ()] Cmmi Maturity , Integration , Cmmi . *Integração dos Modelos de Capacitação e Maturidade*
296 *de Sistemas. 6*, 2007. 292. (Rio de Janeiro: Ciência Moderna)
- 297 [Slack et al. ()] N Slack , S Chambers , T Johnston . *Gerenciamento de Operações e de Processos: Princípios e*
298 *Práticas de Impacto Estratégico*, (Porto Alegre) 2008. 552.
- 299 [Softex et al. ()] *Software e Serviços de TI: A Indústria Brasileira em Perspectiva. Associação para promoção*
300 *da excelência do software brasileiro*, Softex , Associação , Promoção Da Excelência , Do , Brasileiro
301 . <[http://publicacao.observatorio.softex.br/_publicacoes/arqui-vos/resumo/Resumo_](http://publicacao.observatorio.softex.br/_publicacoes/arquivos/resumo/Resumo_Executivo.PDF)
302 [Executivo.PDF](http://publicacao.observatorio.softex.br/_publicacoes/arquivos/resumo/Resumo_Executivo.PDF)>. Acesso em: 5 janeiro 2009. 2010. Campinas, nov.
- 303 [Charbonneau] *Software Project Management -A Mapping between RUP and the PMBOK*, S Charbonneau .
- 304 [Paulo and Anais ()] *São Paulo: Fundação Getúlio Vargas. 29-31 ago*, São Paulo , Anais . 2006.
- 305 [Karner ()] ‘Use Case Points: resource estimation for Objectory projects’. G Karner . *Objective Systems SF AB*
306 1993. (S.l.), Set)