



UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA

Relatórios Técnicos
do Departamento de Informática Aplicada
da UNIRIO
n°0025/2009

**PESQUISA EM ESTIMATIVAS
EM PROJETOS DE
MODELAGEM DE PROCESSOS**

**Claudia Cappelli
Flavia Santoro
José Roberto Dutra
Márcio Barros
Vanessa Nunes**

Departamento de Informática Aplicada

UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO
Av. Pasteur, 458, Urca - CEP 22290-240
RIO DE JANEIRO – BRASIL

Projeto de Pesquisa

Grupo de Pesquisa Participante



Patrocínio



PETROBRAS

PESQUISA EM ESTIMATIVAS EM PROJETOS DE MODELAGEM DE PROCESSOS *

Claudia Cappelli¹, Flávia Santoro^{1,2}, José Roberto Dutra¹,
Márcio Barros^{1,2}, Vanessa Nunes¹

¹Núcleo de Pesquisa e Prática em Tecnologia (NP2Tec)

²Departamento de Informática Aplicada (DIA) – Universidade Federal do Estado do Rio de Janeiro (UNIRIO)

claudia.cappelli@uniriotec.br, flavia.santoro@uniriotec.br, jrcludutra@gmail.com,
marcio.barros@uniriotec.br, vanunes@gmail.com

Abstract. One of the first questions in the initial phase of a process modeling project is how to estimate the effort to be spent for its implementation. A modeling project is very similar to a project of software development. This paper surveys the literature on estimates of effort to carry out projects of modeling business processes and software development seeking techniques and methods applicable to business processes modeling projects. The proposal will be applied on a real scenario, on a large brazilian oil and gas company.

Keywords: Estimate of effort, Metric, Business Process Modeling

Resumo. Uma das primeiras questões na fase inicial de um projeto de modelagem de processos é como estimar o esforço a ser gasto para sua execução. Um projeto de modelagem em muito se assemelha a um projeto de desenvolvimento de software. Este trabalho realizou um levantamento bibliográfico sobre estimativas de esforço para a realização de projetos de modelagem de processos de negócio e desenvolvimento de software buscando trabalhos contendo técnicas e métodos aplicáveis aos projetos de modelagem de processos de negócio na Petrobras.

Palavras-chave: Estimativa de esforço, Métrica, Modelagem de Processos de Negócio

* Trabalho patrocinado pela Petrobras.

Sumário

1	Introdução	8
1.1.	Motivação	8
1.2.	Objetivos	8
1.3.	Detalhamento do Problema	8
1.4.	Método de pesquisa	9
1.5.	Estrutura do Relatório	9
2	Principais abordagens relacionadas	10
2.1	A model for Software Development Effort and Cost Estimation	10
2.1.1	Descrição	10
2.1.2	Pontos positivos e pontos negativos	11
2.1.3	Contribuições para o projeto	11
2.2	A Weighted Coupling Metric for Business Process Models	11
2.2.1	Descrição	11
2.2.2	Pontos positivos e pontos negativos	11
2.2.3	Contribuições para o projeto	11
2.3	Comparison of Artificial Neural Network and Regression Models in Software Effort Estimation	12
2.3.1	Descrição	12
2.3.2	Pontos positivos e pontos negativos	13
2.3.3	Contribuições para o projeto	14
2.4	Early estimating using COSMIC-FFP	14
2.4.1	Descrição	14
2.4.2	Pontos positivos e pontos negativos	16
2.4.3	Contribuições para o projeto	16
2.5	Complexity Metrics for Business Process Models	17
2.5.1	Descrição	17
2.5.2	Pontos positivos e pontos negativos	18
2.6	Effort estimation of use cases for incremental large-scale software development	19
2.6.1	Descrição	19
2.6.2	Pontos positivos e pontos negativos	19
2.6.3	Contribuições para o projeto	19
2.7	Effort Estimation using analogy	19
2.7.1	Descrição	19
2.7.2	Pontos positivos e pontos negativos	20
2.7.3	Contribuições para o projeto	20
2.8	Estimation support by lexical analysis of requirements documents	20
2.8.1	Descrição	20
2.8.2	Pontos positivos e pontos negativos	20
2.8.3	Contribuições para o projeto	21
2.9	Estimeetings: Development Estimates and a Front-End Process for a Large Project	21
2.9.1	Descrição	21
2.9.2	Pontos positivos e pontos negativos	21
2.9.3	Contribuições para o projeto	21
2.10	Expert Judgement as an Estimating Method	22
2.10.1	Descrição	22
2.10.2	Pontos positivos e pontos negativos	22

2.10.3	Contribuições para o projeto	22
2.11	Function Points	22
2.11.1	Descrição	22
2.11.2	Pontos positivos e pontos negativos	24
2.11.3	Contribuições para o projeto	24
2.12	Functional Size Measurement Revisited	25
2.12.1	Descrição	25
2.12.2	Pontos positivos e pontos negativos	25
2.12.3	Contribuições para o projeto	25
2.13	Improving Size Estimates Using Historical Data	25
2.13.1	Descrição	25
2.13.2	Pontos positivos e pontos negativos	26
2.13.3	Contribuições para o projeto	26
2.14	Learning how to improve effort estimation in small software companies	26
2.14.1	Descrição	26
2.14.2	Pontos positivos e pontos negativos	29
2.14.3	Contribuições para o projeto	29
2.15	The application of case-based reasoning to the software development process	29
2.15.1	Descrição	29
2.15.2	Pontos positivos e pontos negativos	30
2.15.3	Contribuições para o projeto	30
2.16	Use Case Estimation – The Devil is in the Detail	30
2.16.1	Descrição	30
2.16.2	Pontos positivos e pontos negativos	31
2.16.3	Contribuições para o projeto	31
3	Conclusões	32
4	Requisitos para a proposta de como estimar projetos de modelagem de processos	33
4.1	Recuperar informações de projetos similares e suas estimativas, nas bases históricas.	33
4.2	Possibilidade de elaboração de estimativas ainda em fases iniciais do projeto	33
5	Glossário	34
	Referências Bibliográficas	36
	Anexo A – Planejamento e execução da revisão sistemática	38
A.1	Introdução	38
A.2	Planejamento da revisão	39

A.2.1	Objetivo	39
A.2.2	Formulação da Pergunta (<i>Question Formularization</i>)	39
A.2.2.1	Pergunta (<i>Question Focus</i>)	39
A.2.2.2	Abrangência e qualidade da pesquisa (<i>Question Quality and Amplitude</i>)	39
A.2.3	Seleção de fontes (<i>Sources Selection</i>)	40
A.2.3.1	Definição de critérios para seleção de fontes (<i>Sources Selection Criteria Definition</i>)	40
A.2.3.2	Idiomas (<i>Studies Languages</i>)	40
A.2.3.3	Fontes de estudo (<i>Sources Identification</i>)	41
A.2.3.3.1	Método de busca (<i>Sources search methods</i>)	41
A.2.3.3.2	Expressão de busca (<i>Search string</i>)	41
A.2.3.3.3	Relação inicial das fontes de estudo (<i>Source list</i>)	42
A.2.3.4	Seleção de fontes após validação (<i>Sources Selection after Evaluation</i>)	42
A.2.3.5	Checagem de referências (<i>References Checking</i>)	42
A.2.4	Seleção de estudos (<i>Studies Selection</i>)	42
A.2.4.1	Definição dos tipos de estudos (<i>Studies types definition</i>)	42
A.2.4.2	Procedimentos para seleção de estudos (<i>Procedures for studies selection</i>)	43
A.2.4.3	Critérios de inclusão e exclusão (<i>Studies inclusion and exclusion criteria definition</i>)	43
A.3	Execução da revisão	44
A.3.1	Lista de Publicações de Controle	44
A.3.2	Histórico das Buscas	44
	Anexo B – Relatório de estatísticas de projetos de modelagem de processos	48
B.1	Introdução	48
B.2	Estudo realizado	49
B.2.1	Condução do estudo prévio	49
B.3	Proposta de estimativa	50

1 Introdução

1.1. Motivação

Atualmente diversas iniciativas em modelagem de processos de negócio são conduzidas nas organizações. Projetos são realizados para que estas iniciativas possam ser implantadas. Para condução destes projetos são utilizadas em geral práticas de gerência de projetos já bastante conhecidas tanto na academia como no mercado.

Uma das primeiras atividades a ser realizada num projeto de modelagem é um levantamento inicial de modo a se identificar o escopo do projeto e ter um prazo para seu desenvolvimento. Este prazo será validado a partir de uma estimativa de esforço necessário para a conclusão do projeto de modelagem e de um volume estimado de alocação de profissionais para este projeto ao longo do tempo. Porém, uma das questões inerentes a este contexto, e foco deste documento, é como estimar o esforço a ser gasto para execução de um projeto de modelagem de processos.

Um projeto de modelagem em muito se assemelha a um projeto de desenvolvimento de software. Em geral parte-se sempre de um novo escopo e contexto e deseja-se uma solução única, ou seja, construída naquele projeto e que não será repetida em geral. Esta similaridade nos traz também a mesma dificuldade que temos hoje nos projetos de desenvolvimento. A maioria das técnicas existentes (pontos por função, pontos por caso de uso, complexidade de módulos, linhas de código) não atende na plenitude as medições de tamanho e derivações de esforço.

De forma a contribuir nesta linha de pesquisa, estabeleceu-se um projeto para investigar abordagens e metodologias que tenham como foco a estimativa de projetos de modelagem de processos.

1.2. Objetivos

Realizar uma busca sistemática na literatura de Ciência da Computação, em particular na área de Engenharia de Software, por trabalhos sobre estimativas de esforço para a realização de projetos de modelagem de processos de negócio e desenvolvimento de software que possam elucidar mecanismos aplicáveis aos projetos de modelagem de processos de negócio na Petrobras. No caso de desenvolvimento de software, o interesse da pesquisa está voltado para estimativas baseadas em requisitos de negócio obtidos nas fases iniciais de projetos de sistemas de informação.

1.3. Detalhamento do Problema

Em geral, são realizadas mais de uma estimativa ao longo do ciclo de vida de um projeto, na medida em que o conhecimento sobre os seus requisitos vai sendo aprofundado. Portanto, a estimativa mais crítica e difícil de ser elaborada é aquela que acontece na fase inicial do projeto, quando poucos detalhes são conhecidos e não se tem uma visão clara dos requisitos. Neste momento, a estimativa torna-se um instrumento para tomada de decisão quanto à viabilidade dos projetos.

Apesar da existência de diversas técnicas para estimar projetos de desenvolvimento de software, as estimativas ainda são muito dependentes da experiência dos estimadores e do conhecimento sobre projetos passados.

Neste sentido, a experiência na elaboração de estimativas de projetos de modelagem de processos, em diversos Escritórios de Processos da TIC/E&P, levou a equipe da UNIRIO/NP2TEC a prover um mecanismo de estimação do esforço médio necessário para a realização de um projeto desta natureza, de acordo com o tipo de projeto e o número de processos modelados. Trata-se de um estudo inicial, realizado com restrições de tempo para levantamento dos dados, organização dos mesmos e realização de análises alternativas, visando apresentar resultados iniciais que possam apoiar decisões de investimento em projetos de modelagem. Este estudo encontra-se no ANEXO B.

1.4. Método de pesquisa

O método de pesquisa seguiu as linhas gerais de uma revisão sistemática. O planejamento e execução da revisão são apresentados no ANEXO A.

O desenvolvimento de uma abordagem sistemática de revisão visa estabelecer um processo formal para conduzir este tipo de investigação, evitando a introdução de eventuais vieses da revisão de literatura informal. A revisão sistemática é um tipo de estudo secundário (KITCHENHAM, 2004), cujo processo de pesquisa segue um conjunto de passos metodologicamente bem definidos de acordo com um protocolo prévio (BIOLCHINI et al., 2005) e cuja adoção procura reduzir o viés inerente a uma revisão informal (SILVA FILHO, 2006).

1.5. Estrutura do Relatório

Este relatório está organizado em seis seções, incluindo a Introdução. Na Seção 2, são apresentados os trabalhos relacionados, provenientes das leituras e análises dos artigos listados na Introdução. Na Seção 3, são discutidas as conclusões gerais sobre os trabalhos relacionados. Na Seção 4, com base nos resultados obtidos, são descritos os requisitos da proposta deste projeto. As Seções 5 e 6 apresentam respectivamente o Glossário e Referências Bibliográficas. Há ainda o Anexo A que contém o planejamento e execução da revisão bibliográfica realizada e o Anexo B que contém o relatório de estatísticas de projetos de modelagem de processos.

2 Principais abordagens relacionadas

2.1 A model for Software Development Effort and Cost Estimation

2.1.1 Descrição

Boehms [1] classifica modelos algorítmicos usados para estimativa de esforço da seguinte forma:

- Modelos lineares (linear models): também chamados de modelos matemáticos que traçam uma linha simples para os dados observados;
- Modelos multiplicativos (multiplicative models): expressam o esforço como um produto de constantes com vários fatores de custo;
- Modelos analíticos (analytic models): expressam o esforço através de uma função que não é nem linear nem multiplicativa;
- Modelos tabulados (tabular models): representam o relacionamento entre os fatores de custo e o esforço de desenvolvimento em forma de matriz;
- Modelos de composição (composite models): usam uma combinação de todos os modelos acima ou alguns deles. Exemplo: RCA PRICE S, Putnam's SLIM Software lifecycle management, COCOMO.

O modelo de Putnam, objeto de estudo dos autores do artigo, pode ser utilizado na análise de processos para avaliar o impacto de um cronograma apertado e prever custo em termos longos a partir da evolução do projeto. Dada a mão-de-obra (quantidade e relação homem-hora) disponível para o projeto, o modelo de Putnam estima tempos de completude através de técnicas simples de curve-fitting.

O modelo de Putnam, segundo os autores prevê de forma simples uma estimativa de custos para o desenvolvimento de um software. No entanto, sua capacidade em prever tempo de desenvolvimento e requisitos para quantidade de mão-de-obra em um estado inicial do desenvolvimento não são satisfatórios. Trata-se de uma inabilidade inerente de modelos que usam a equação de Norden/Rayleigh para representar mão-de-obra em homem/ano por unidade de tempo (normalmente representado em Homem-Ano/Ano).

Os autores apresentam o modelo de Putnam através das explicações da curva de Norden/Rayleigh, equações associadas e suas limitações. O problema principal reside no fato de que uma previsão inicial (early prediction) falha, pois modelos lineares simplificados falham em capturar a variação inicial das observações.

Os autores apontam duas formas de tentar resolver essa questão: ajustar a curva com um grau mais elevado para o conjunto de dados e/ou encontrar uma alternativa de transformação para os eixos. Essas opções, no entanto, se aplicadas ao modelo de Putnam, criariam uma complexidade enorme ao modelo e o tornariam muito mais dependente dos dados e não genérico. Os autores então propuseram um modelo chamado GAMMA para compensar essas limitações usando a segunda forma proposta.

Eles propõem novas equações para construção de uma nova curva, diferente da de Norden/Rayleigh e que levam em consideração o grau de transformação baseado em resíduos do conjunto de dados que criam desvios nas curvas. Além disso, eles levam também em consideração um fator de erro baseado no real e no estimado. Foram criadas equações para cálculo de cada uma dessas variáveis. E para cada explicação, foi provado matematicamente, a superioridade deste modelo em relação ao de Putnam.

2.1.2 Pontos positivos e pontos negativos

Como ponto positivo, o modelo leva em consideração desvios que podem acontecer nos projetos e taxa de erro na estimativa.

Como ponto negativo, o artigo não descreve quais dados são importantes para a construção das variáveis utilizadas nas equações, bem como não apresenta um exemplo de uso dos modelos e equações propostas.

2.1.3 Contribuições para o projeto

Pode ser interessante olhar os aspectos apontados pelo modelo que levam em consideração desvios, taxas de erros e a taxa de “manpower buildup” (que relaciona a quantidade de pessoas e o tipo de projeto que é realizado para identificar o quão realista o cronograma do projeto é).

2.2 A Weighted Coupling Metric for Business Process Models

2.2.1 Descrição

Em um artigo de apenas quatro páginas, os autores ressaltam a similaridade entre projetos de desenvolvimento de software e projetos de modelagem de processos de negócio. Baseados nestas similaridades, os autores sugerem a importância de se definir e utilizar métricas nos projetos de modelagem de processos. Com base nesta afirmativa, apresentam uma métrica de coesão aplicável a processos desta natureza.

2.2.2 Pontos positivos e pontos negativos

Como ponto positivo, o artigo serve como referência para nossa premissa básica no projeto: projetos de modelagem possuem similaridades com projetos de software.

Como ponto negativo, a apresentação de uma solução a procura de um problema.

2.2.3 Contribuições para o projeto

Como colocado acima, o artigo dá substância a uma premissa que usamos desde o início do projeto e que justifica a análise que estamos realizando sobre técnicas de estimação em projetos de software. Isto é particularmente importante quando redigirmos o resumo de tudo que foi lido.

2.3 Comparison of Artificial Neural Network and Regression Models in Software Effort Estimation

2.3.1 Descrição

Os autores comentam sobre a importância de gestão de projetos de desenvolvimento de software, em especial do processo de estimativas e em linhas gerais destacam as variáveis que devem ser estimadas em função da dependência entre elas como segue na figura abaixo.

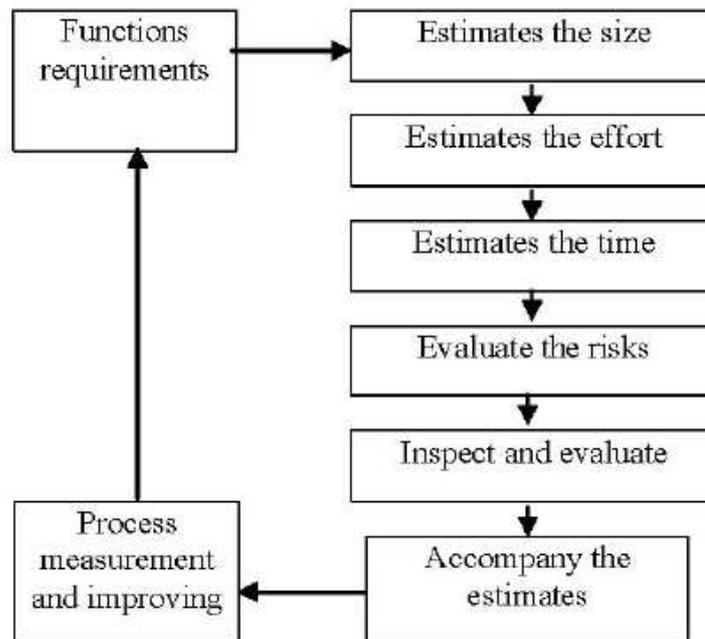


Fig. 1. The predictive process [1]

Destacam ainda que as organizações normalmente querem escolher modelos de estimativas com maior grau de acurácia no resultado, porém modelos robustos demais pedem dados mais robustos, concretos e confiáveis, e muitas vezes estas organizações não possuem essas informações de antemão.

Os autores citam algumas iniciativas de modelos e métodos de estimativas. Inicialmente as propostas tendiam a retirar a subjetividade no processo de estimativas como é o caso de Análise de pontos por função, COCOMO e Ordinal Regression Models (todos possuem referências).

Eles citam que as técnicas de análise e exploração de dados como clustering, case-based reasoning (ex.: ESTOR) e ANN tem surtido efeito na estimativa de esforço do projeto de desenvolvimento de software.

Os autores citam ainda outras abordagens e combinações entre elas, mas sempre com taxas mediana a baixa de performance dos resultados.

Foi realizado um estudo de caso baseado nos dados de um estudo prévio implementado por Boehm [7] para testar o método COCOMO e o objetivo dos autores era gerar uma estimativa de esforço para cada fase do projeto de desenvolvimento de sistemas (O método COCOMO no entanto só apresentava o esforço total).

Eles utilizaram o software STATISTICA para encontrar o melhor modelo dentre os possíveis 10 modelos existentes no software, baseado em um número de variáveis independentes disponibilizadas no appendix do artigo. Algumas delas são: linhas de código, tamanhos do banco de dados, tempo de desenvolvimento, volatilidade do hardware, experiência dos profissionais, etc...

Os autores fizeram um passo de regressão construindo um modelo de regressão onde era adicionado em cada fase do projeto a variável com a maior correlação parcial levando em consideração todas as variáveis do modelo. O objetivo era maximizar um valor F dado. Este valor F servia para verificar, na regressão, se uma determinada variável tinha impacto no esforço do projeto. Quando a variável era acrescentada e o valor diminuía, esta variável era descartada. O estudo mostrou que apenas a variável TOTKDSI (software size in source lines of code) apresentou resultado significativo.

Após esse estudo eles quiseram refinar retirando o sexto projeto, de seis em seis, dos 63 projetos (criando uma forma quase randômica). Eles queriam diminuir ou verificar o otimismo que os métodos acabam por embutir.

O resultado pode ser visto nas tabelas 1 e 2

TABLE I
REGRESSION AND ANN ESTIMATES

Esforço/T	Regression	ANN
6,753	26,026	26,5586
59,3	93,678	52,37
3705	673,847	859,7053
120,05	202,332	165,4667
1,12	21,885	25,5485
12,93	48,577	33,0107
10,58	111,104	63,0066
375,24	969,057	638,4523
3,5	26,846	26,7646
9,98	31,561	27,9878

TABLE II
THE PREDICTIVE ACCURACY

	Regress. Eq.	R- square	MMRE
ANN	-1,68+1,676*x	0,85	420
Regression	-1,71+1,623*x	0,83	462
FPA	-37 +0,96x	0,58	103
COCOMO	27,7 + 0,156x	0,70	610
SLIM	49,9 +0,082x	0,89	772

2.3.2 Pontos positivos e pontos negativos

Pontos positivos: Segundo os autores, técnicas baseadas em ANN têm sido aplicadas com sucesso em vários domínios no sentido de desenvolver soluções para estimativas de problemas, classificação, controle, etc. Estas técnicas podem ser usadas como modelos preditivos, pois são capazes de modelar funções complexas. Os autores

apontam a questão da dependência entre parâmetros e variáveis quaisquer do projeto que pode causar erros de estimativas.

Pontos negativos: Os autores não explicam como chegaram aos cálculos realizados, o que dificulta o entendimento das tabelas de resultado.

2.3.3 Contribuições para o projeto

Os autores apostam em técnicas onde já se possui o valor da variável tamanho do projeto. No caso de projetos de desenvolvimento de software, eles tratam de linhas de código, o que é perigoso. Transpondo esta questão para modelagem de processos, o perigo, ainda que menor, ainda existe, pois não se sabe ao certo o tamanho exato do projeto. O que facilita é que a técnica de modelagem é única e, portanto o tamanho médio levantado de um projeto tende a ser próximo da realidade (levando-se em consideração um bom levantamento e a não ocorrência de mudanças de escopo durante a execução). Portanto modelos de regressão podem ser interessantes de investigar.

2.4 Early estimating using COSMIC-FFP

2.4.1 Descrição

Nas fases iniciais de um projeto poucos detalhes são conhecidos mas o impacto de uma decisão em relação ao desenvolvimento de um sistema de informação é grande. Uma estimativa do tamanho do software a ser desenvolvido tem de ser feita com tão pouca informação quanto possível e tanta exatidão quanto possível. Neste sentido, várias técnicas são derivadas da experiência sobre quais aspectos de um sistema de informação podem contribuir na estimativa de tamanho em um estágio inicial de desenvolvimento como, por exemplo:

- **“Fuzzy logic”** é uma forma de sistematizar as comparações com os últimos trabalhos.
- **“Standard-component sizing”** identifica uma série de componentes-chaves que podem ser comparados com os dados históricos sobre tais componentes.
- **“Proxy-based Estimating”** atribui um intervalo de tamanho para cada método conceitual dentro do software a ser estimado usando dados históricos de tais métodos.
- **“Wideband-Delphi”**, onde um número de especialistas estima o tamanho de um sistema de informação com base nos seus requisitos. Um moderador calcula o tamanho médio estimado e devolve aos especialistas, juntamente com todas as outras estimativas, para uma nova rodada. Este processo continua até que o desvio entre as estimativas seja aceitável.
- **“FPA i”**, nesta abordagem um número de estimadores estabelece o peso de uma série de requisitos. Para cada peso é atribuído um valor mínimo, um mais provável e um máximo estimado.
- **“Early and Quick Function Point Analysis”**, método baseado em software para identificar objetos em diferentes níveis de detalhe. Para cada tipo de objeto há uma estimativa do valor mínimo, do mais provável e do máximo.

Dependendo do tipo de software que tem de ser produzido e do ambiente de desenvolvimento, todos estes métodos podem ser úteis, mas quase todos dependem da experiência dos especialistas e conhecimento sobre projetos anteriores, tornando-os de certa forma subjetivos. Outra maneira de fazer estimativas iniciais de tamanho é por simplificação da técnica detalhada de estimação.

Estimativas iniciais utilizando Pontos por Função:

- A primeira simplificação é usada quando os elementos são conhecidos mas não se sabe exatamente os seus respectivos tamanhos: tomar o valor médio para a EI (“External Inputs”), EO (“External Outputs”) e EQ (“External Enquiries”) porque geralmente existe um número equivalente de funções com complexidade baixa e alta. Atribuir baixa complexidade para ILF (“Internal Logical File”) e FEI (“External Interface file”) porque mais de 90% dos arquivos são de baixa complexidade. Esta simplificação, também conhecida como FPS (“Function Points Simplified”), pode ter uma precisão no intervalo de 5%.
- Outra simplificação é o fato de que geralmente há uma relação linear entre o número de arquivos lógicos e o número de funções transacionais sobre estes arquivos. Portanto, se houver um modelo de dados já disponível na terceira forma normal, é possível estimar a número de pontos por função com base neste modelo de dados. Para cada tipo de entidade ILF são contados 25 pontos por função e para cada tipo de entidade FEI são contados 10 pontos por função.
- Se apenas um modelo conceitual de dados é conhecido, uma aproximação semelhante pode ser usada para estimativa do tamanho. Para cada tipo de entidade ILF conceitual são contados 35 pontos por função e, para cada tipo de entidade FEI conceitual são contados 15 pontos por função. Desvios de até 50% têm sido registrados, de modo que este tipo de avaliação só deve ser utilizado como uma primeira estimativa.

As estimativas por pontos por função são independentes do ambiente de desenvolvimento, isto é, em todos os ambientes em que a análise de pontos de função pode ser aplicada, estas técnicas de estimativas iniciais funcionam da mesma maneira. Em ambientes onde a análise de pontos de função não se aplica, estimar o tamanho usando COSMIC-FFP pode ser uma alternativa. O manual de medição deste método apresenta duas técnicas para a estimativa do tamanho inicial: a aproximada e a aproximada refinada. Até o momento em que o artigo foi escrito, havia pouco relato de experiência prática usando essas técnicas.

COSMIC-FFP Aproximada: Nas fases iniciais do desenvolvimento de software apenas o número de processos funcionais é conhecido. Para estimar o tamanho de uma aplicação, o número de processos funcionais pode ser multiplicado pelo tamanho médio de processos funcionais de um software com características similares. O manual de medição do COSMIC-FFP fornece um exemplo de software para aviação militar onde o tamanho médio dos processos funcionais é 8 cfsu. A experiência no grupo financeiro Dutch Rabobank aponta um tamanho médio de 7,2 cfsu, o que sugere que esta técnica é dependente do ambiente.

COSMIC-FFP Refinada: Esta técnica é um refinamento da anterior, onde os processos funcionais são divididos em categorias com igual contribuição para o tamanho funcional da aplicação. O manual de medição do COSMIC-FFP fornece um exemplo em que classifica os processos funcionais em quatro categorias: pequeno, médio, grande e muito grande. Para cada uma destas categorias, são calculados os tamanhos

médios dos processos funcionais de um software local com características similares à aplicação em questão. Para estimar o tamanho da aplicação multiplica-se, para cada categoria, a quantidade de processos funcionais pelos respectivos tamanhos médios dos processos funcionais do software local. Durante a pesquisa foi encontrada uma discrepância entre a forma que o manual do COSMIC-FFP descreve o cálculo dos quartis (categorias) e como eles são efetivamente calculados nos exemplos do manual.

- Método 1 (descrito no manual): divide o tamanho total do software por quatro e encontra o tamanho médio dos quartis, contendo os processos funcionais ordenados por tamanho.
- Método 2 (usado nos exemplos): divide o número de processos funcionais por quatro e encontra o tamanho médio dos quartis, contendo os processos funcionais ordenados por tamanho.

Levantamento de estimativas iniciais de tamanho em diferentes ambientes

O final do artigo apresenta números levantados de diversos projetos que utilizaram COSMIC-FFP para estimativas de tamanho em estágios iniciais do ciclo de desenvolvimento. A intenção foi avaliar o quanto esta técnica é dependente do ambiente. Foram selecionados 47 projetos, divididos em quatro setores: bancário, governo, seguro e logístico.

Para o método aproximado, embora o tamanho médio dos projetos de cada setor fosse da mesma ordem de grandeza, o tamanho médio dos processos funcionais de cada setor mostrou-se bastante diferente. Para explicar estas diferenças os autores apontam a necessidade de futuras investigações sobre os detalhes dos projetos. Os números suportam a idéia de que existe uma relação entre setor e tamanho médio dos processos funcionais.

Para o método refinado utilizando o Método 2, foi observado que os intervalos de tamanho para as categorias pequeno, médio e grande ficam muito próximos, não sendo de utilidade para estimativas em fases iniciais de projeto.

Precisão do método

Os autores recalcularam o tamanho dos processos funcionais com base nos números encontrados nas estimativas e compararam com os valores reais fornecendo percentuais de desvio por setor. Os números variaram de 26% a 53% para o método aproximado, enquanto para o método refinado a variação foi de 7% a 19%.

2.4.2 Pontos positivos e pontos negativos

Como ponto positivo: o fato de ser possível realizar estimativas de tamanho, em fases iniciais de projeto utilizando as técnicas COSMIC-FFP.

Como ponto negativo: pouco relato de experiência prática usando essas técnicas.

2.4.3 Contribuições para o projeto

Os métodos baseados em COSMIC-FFP podem servir de base para definição de um método para estimar esforço em projetos de modelagem de processos, principalmente em relação à questão de estimativas em fases iniciais de projeto.

2.5 Complexity Metrics for Business Process Models

2.5.1 Descrição

Diversas pesquisas neste sentido têm sido realizadas sobre complexidade de software com objetivo de estimar taxa de erro, custo de manutenção e partes de software que devem ser reestruturadas. Neste artigo as idéias a respeito da complexidade de software são discutidas com vistas à aplicação nos modelos de processos de negócio. É fornecida uma visão geral dos fatores que podem influenciar a complexidade dos modelos e métricas que podem ser usadas para medir estes fatores. A discussão aqui é restrita ao fluxo de controle não envolvendo aspectos como, fluxo de dados e uso de recursos.

Tamanho do modelo: linhas de código

O dicionário de padrões do IEEE define complexidade de sistema ou componente como sendo o grau de dificuldade de entendimento e verificação de seu desenho ou implementação. A medida de complexidade de software mais simples e conhecida é a de “linhas de código” (LOC) que representa o tamanho de um programa. Normalmente refere-se ao número de expressões executáveis, desconsiderando-se comentários, quebras de linhas etc.

O número de atividades pode ser visto como o equivalente ao número de linhas de código em um software, sendo a forma mais simples e fácil para medir o tamanho de um modelo de processo de negócio (BPM). No entanto, o número de atividades não leva em conta a estrutura do modelo. Um modelo com 50 atividades pode ser desenhado de forma bem estruturada e de fácil de entendimento ou, de forma oposta, ser desestruturado e difícil de entender. Por este motivo, o artigo apresenta a seguir outras métricas com objetivo de medir a complexidade do fluxo de atividades.

Complexidade do Fluxo de Controle do Modelo: McCabe-Metric

O número ciclomático, introduzido por McCabe, é uma das métricas de software mais difundidas e usadas. Corresponde ao número de decisões binárias mais 1. Decisões não binárias com N resultados possíveis são contadas como N-1 decisões binárias. A Complexidade do Fluxo de Controle de processos (CFC), generalização do número ciclomático, é definida por Cardoso como sendo o número de estados mentais que tem que ser considerados quando um projetista desenvolve um processo.

De forma análoga ao número ciclomático, que é igual ao número de decisões binárias mais 1, a métrica CFC para aplicação em BPMs conta o número de decisões no fluxo de controle. Cada split no modelo incrementa o número de decisões possíveis da seguinte forma: AND-split soma 1, XOR-split soma N e OR-split soma $2N - 1$.

Estrutura do modelo: Aninhamento e saídas da estrutura de controle

Apesar de dois modelos possuírem o mesmo CFC, eles podem ter complexidades diferentes. Um fluxo linear tende a ser menos complexo que um com muitas decisões. Os autores consideram as métricas de aninhamento adequadas para medir esta influência na complexidade global do modelo. Quanto mais profundo o aninhamento, maior a complexidade do modelo. A definição desta métrica é simples e direta: é igual ao número necessário de decisões no fluxo de controle para execução de um processo. É sugerido o uso desta métrica de forma complementar ao CFC. No entanto, é

importante perceber que esta métrica pode levar a resultados falsos. As ferramentas gráficas de modelagem de processos de negócio não obrigam separações e junções de fluxo sejam formadas em pares casados, permitindo assim a criação de modelos com problemas semelhantes ao da programação não estruturada, com desvios (GOTO) indesejáveis no código.

Na programação de software, foi introduzida a métrica “knot count” para medir estes desvios indesejáveis do fluxo de controle.

Um modelo de processo de negócio é considerado bem estruturado se possui separações e junções apropriadamente aninhadas em pares casados. Esta afirmativa é formalmente definida por meio da terminologia das redes de Petri, onde um workflow bem estruturado é aquele que não possui “handles”. Esta métrica é aplicável em modelos do tipo workflow, tais como EPCs (Event driven Process Chains) que é o padrão adotado pela ferramenta ARIS. Algumas ferramentas de workflow possuem restrições semânticas que forçam o modelador a construir modelos bem estruturados, ou seja, com número de “handles” igual a zero.

Grau de Compreensão do modelo: Métricas de Complexidade Cognitiva

Para suprir algumas deficiências da métrica CFC, outras propostas foram elaboradas. Por exemplo, dois modelos mostram estruturas de controle com mais de um caminho sendo executado em paralelo. Neste caso, se a necessidade fosse desenvolver casos de teste, a métrica CFC seria extremamente apropriada, porque os casos de teste devem contemplar todos os caminhos possíveis de serem percorridos. No entanto, para expressar a dificuldade de entendimento do modelo esta métrica pode não ser aplicável. Em alguns casos, o entendimento do modelo independe da quantidade de caminhos existentes. O artigo cita o peso cognitivo como sendo a métrica para medir o esforço requerido para compreensão de um pedaço de software. Pesquisadores, com base em estudos empíricos, definiram pesos cognitivos para estruturas de controle básicas.

Utilizando-se a esta mesma linha de raciocínio, parece possível a definição de pesos cognitivos para BPMs. Neste caso, deve-se considerar que BPMs podem ser modelados de forma não estruturada, ou seja, levar em conta “workflow patterns” e “anti-patterns”.

Modularização do modelo

Para análise da modularização de um BPM é possível adotar as métricas fan-in e fan-out para sistemas de software modularizados, onde: Fan-in refere-se ao número de vezes que um módulo é chamado. Fan-out refere-se ao número de módulos que são chamados pelo módulo que está sendo investigado.

2.5.2 Pontos positivos e pontos negativos

Como ponto positivo, algumas das métricas apresentadas no artigo podem ser aplicadas diretamente sobre os modelos (no. de atividades, CFC, aninhamento Max./médio, no. de "handles" e fan-in/fan-out).

Como ponto negativo, pesos cognitivos, patterns e anti-patterns são métricas de software que requerem customização para aplicação em modelos de processo de negócio.

2.6 Effort estimation of use cases for incremental large-scale software development

2.6.1 Descrição

A técnica de Pontos por Caso de Uso (UCP) se baseia na identificação e classificação de atores e casos de uso de um projeto de software para estimar o esforço necessário para o desenvolvimento deste projeto. Os autores propõem algumas simplificações para aplicar a técnica e apresentam uma situação onde estas simplificações foram aplicadas.

2.6.2 Pontos positivos e pontos negativos

Como ponto positivo, os autores documentaram uma forma de calcular o esforço de manutenção que já é aplicada na prática, tanto em UCP quanto em FP. O estudo de caso apresentado é muito interessante, em especial no que diz respeito à distribuição do esforço ao longo das atividades de desenvolvimento (que poderiam ser ainda mais detalhadas).

Por outro lado, como aspectos negativos destacam-se o problema da variação de complexidade das transações não ter sido resolvido e ainda requerer a análise manual por um especialista para identificar se as transações estão no mesmo nível de granularidade e compatíveis com o esperado pela técnica UCP (mesmo este, não é bem definido).

2.6.3 Contribuições para o projeto

A grande contribuição seria para a primeira fase do nosso projeto, onde foi construída uma maneira rápida de estimar o esforço para a modelagem de processos. Nesta etapa o objetivo é exatamente o inverso: adicionar detalhes que possam ser obtidos a partir de uma análise mais profunda do processo (o que chamaríamos de uma estimativa pós-arquitetural em software) para chegar a uma estimativa mais precisa do esforço.

2.7 Effort Estimation using analogy

2.7.1 Descrição

Os autores fazem uma comparação entre as técnicas de julgamento por especialistas através da técnica de DELPHI, as técnicas de cálculo por algoritmos (COCOMO e Pontos por Função) e a técnica de analogia. Chegam à conclusão de que esta terceira seria a mais apropriada e constroem uma ferramenta para aplicá-la. Acreditam que esta terceira combina pontos positivos das outras duas técnicas. Esta técnica consiste basicamente na escolha de características de projetos (escolha feita por especialistas) que devem ser registradas ao longo do seu desenvolvimento para depois serem escolhidas e comparadas com os novos projetos (aqui usam-se alguns algoritmos para seleção e combinação).

2.7.2 Pontos positivos e pontos negativos

Como ponto positivo, podemos citar o próprio uso da técnica de analogias. Ela se apresenta como mais uma opção para o processo de estimativas e aparentemente tem vantagens sobre as demais.

Como ponto negativo, podemos citar que o artigo não explicita como são feitos os cálculos para se chegar a que critérios usar e também não coloca como é feita a escolha pelos especialistas dos itens que serão pontuados nos projetos. Além disso, o artigo se preocupa muito com a ferramenta e pouco fala da técnica.

2.7.3 Contribuições para o projeto

Esta técnica pode ser bastante útil neste projeto, pois no caso de projetos de modelagem de processos na Petrobras já temos vários realizados que poderiam ser usados como base histórica. O que teria que ser feito é a escolha dos dados a serem comparados entre os projetos. Um ponto negativo neste caso é se para os projetos já realizados não existirem os dados a serem coletados registrados.

Vale ressaltar que numa primeira fase deste projeto na Petrobras fizemos algo similar ao que é proposto neste artigo. Escolhemos algumas variáveis e comparamos entre vários projetos para obter uma estimativa a ser usada para comparar com novos projetos.

2.8 Estimation support by lexical analysis of requirements documents

2.8.1 Descrição

Embora o resumo e a introdução levem a crer que trata-se de um artigo sobre estimativa de esforço para o desenvolvimento de um sistema, os autores focalizam a identificação das classes centrais para o projeto e implementação deste software. Os autores argumentam que o número de classes está diretamente relacionado com o esforço, apresentam e comparam duas técnicas baseadas na análise léxica de documentos de requisitos para identificar as classes. O artigo apresenta uma validação da proposta, onde os resultados das técnicas baseadas na análise léxica são comparados com os resultados obtidos por um especialista e um conjunto de alunos.

2.8.2 Pontos positivos e pontos negativos

Como ponto positivo, uma análise léxica mais estruturada pode ajudar a encontrar as principais classes de um projeto orientado a objetos. No entanto, há de se questionar até onde esta questão é realmente relevante. Ela foi extensamente abordada no início dos anos 90, quando do início da aplicação de técnicas orientadas a objetos na análise e projeto de sistemas, sendo praticamente esquecida desde então.

Como ponto negativo, verifica-se que não é trivial estimar o esforço para a realização de um projeto baseado apenas no número de classes chave deste.

2.8.3 Contribuições para o projeto

O artigo cumpre um papel importante no projeto de pesquisa por chamar a atenção sobre a possibilidade de extrairmos informações de uma descrição sucinta do processo. Com base em uma análise léxica, pode ser viável identificar o número de atividades, de riscos e de outros elementos que não temos quando começamos a fazer as estimativas. Algo para ser pensado à medida que a proposta do método for evoluindo.

2.9 Estimeetings: Development Estimates and a Front-End Process for a Large Project

2.9.1 Descrição

Os autores abordam as principais dificuldades relacionadas com as estimativas de projetos de desenvolvimento de software, quebrando o processo de estimação em duas fases essenciais: a estimativa do tamanho e a estimativa do esforço (a partir do tamanho). Os autores observam que a principal dificuldade envolvida nas estimativas está em identificar o trabalho que deve ser realizado para implementar uma determinada funcionalidade. Assim, ressaltam a importância de tomar as decisões sobre estimativas em grupos, onde estejam representados os principais envolvidos na arquitetura do projeto (representantes de subsistemas) e técnicos especialistas em estimação de projetos de software.

2.9.2 Pontos positivos e pontos negativos

Como pontos positivos, os autores destacam que a discussão sobre as estimativas distribui um comprometimento sobre sua execução, permite que os melhores profissionais participem das decisões sobre as estimativas e exige que as equipes funcionais preparem um material de qualidade para ser entregue e “digerido” antes da reunião.

Como ponto negativo, destaca-se que os autores não apresentam evidências claras sobre a melhoria do processo de estimação (apenas informações subjetivas sobre suas percepções sobre o método).

2.9.3 Contribuições para o projeto

O artigo segue uma linha diferente do que estamos abordando. Ao invés de destacar informações relevantes para as estimativas e buscar uma fórmula ou derivação baseada nestas informações, o artigo destaca a tomada de decisão em grupo a partir da junção e discussão de diferentes visões construídas de forma subjetiva. Considerando as divergências existentes nos dados que utilizamos na estimativa inicial que geramos para o projeto, talvez seja possível pensar em uma técnica que envolva grupos de discussão sobre o tema para gerar a estimativa de um novo projeto. É uma linha distinta das propostas até então revistas, mas talvez seja complementar de acordo com a precisão que conseguimos obter dos dados atuais que vêm sendo analisados.

2.10 Expert Judgement as an Estimating Method

2.10.1 Descrição

Os autores apresentam uma avaliação crítica dos diferentes métodos de estimativa utilizados para avaliação do esforço necessário para a realização de tarefas ligadas ao desenvolvimento e manutenção de software. Em particular, os autores criticam a classificação de Barry Boehm sobre o tema, reorganizando as classes de métodos sugeridas por Boehm em um modelo matricial (ao invés de uma lista de alternativas). De posse desta matriz, os autores identificam a necessidade de opinião subjetiva e intervenção manual em todos os métodos. Assim, eles destacam a importância de se entender a forma com que os especialistas conduzem o processo de estimação, como forma de se aprimorar os demais métodos.

2.10.2 Pontos positivos e pontos negativos

Como ponto positivo, o artigo apresenta diversas considerações interessantes sobre o uso de especialistas nos processos de estimação.

Como ponto negativo, os resultados do estudo realizado não são exatamente surpreendentes e o objetivo principal, conforme expressado na introdução do artigo, acaba se perdendo devido a características próprias da empresa onde o estudo de caso foi conduzido.

2.10.3 Contribuições para o projeto

Não aparecem muitas contribuições para o projeto. Na conclusão, os autores deixam um tema para reflexão, sugerindo que métodos de estimativa deveriam se integrar às opiniões de especialistas ao invés de tentar substituí-las, fornecendo meios para que os especialistas tenham mais informação para tomar suas decisões e não incorram em esquecimento de partes importantes do projeto. Alguns métodos baseados em IA e na representação e gerenciamento de conhecimento propõem algo nesta linha, mas nunca vi um estudo prático sobre sua aplicação.

2.11 Function Points

2.11.1 Descrição

Alguns métodos para realizar medição da funcionalidade de software vêm sendo propostos, e estes variam de acordo com o que medir e como são feitas estas medições, além de sua utilidade, aplicabilidade e aceitação na indústria. Medidas de tamanho podem focar ou no problema que o software pretende tratar ou pelo programa em si, ou seja, a solução.

O problema é expresso na especificação funcional, através dos requisitos, e a solução possui um tamanho físico (comprimento ou extensão). Assim, funcionalidade e comprimento são dois aspectos diferentes do tamanho de um software. Medir comprimento é fácil, a medida mais comum é o número de linhas de código (LOC). Porém medir funcionalidade é mais complicado, diversos métodos já foram propostos: Pontos por função do IFPUG; Pontos por função Mark II; Cosmic, entre outros.

O método pontos por função foi proposto por Albrecht em 1979. Basicamente consiste em contar certos componentes que representam manifestações visíveis do software com pesos que devem refletir seu valor para o cliente, e depois ajustar o resultado para incorporar fatores inerentes à complexidade técnica da aplicação. Uma revisão na proposta inicial foi feita em 1984.

A primeira fase do método pontos por função consiste em identificar certos componentes do sistema que provêm funcionalidade ao usuário: entradas externas, saídas externas, consultas externas, arquivos lógicos internos e arquivos externos de interface. O número de pontos associado a cada componente depende de sua complexidade funcional.

A segunda fase do método consiste em realizar um ajuste no resultado obtido com a contagem. Para isso é definido um Fator de Ajuste com base em 14 características gerais de sistema: comunicação de dados, processamento distribuído de dados, performance, configuração, taxa de transação, entrada de dados em tempo real, eficiência do usuário final, atualizações em tempo real, processamento complexo, reusabilidade, facilidade de instalação, facilidade de operação, múltiplos sites, facilidade de mudanças.

Várias experiências com o uso deste método e sua boa aceitação no mercado comprovam seu valor. No entanto, existem críticas quanto aos problemas teóricos na construção do método: a primeira fase envolve medidas em escala absoluta, mas depois passa a ser escala ordinal; na função de ajuste também existem inconsistências nas operações aplicadas; a origem e objetividade dos pesos também são questionadas. Isso resulta que a medida é inválida em várias situações. Questiona-se então o que realmente é medido (funcionalidade, tamanho), e qual deve ser sua utilização (estimar, determinar produtividade).

O fato é que a proposta é adequada para sistemas fortemente baseados em manipulação de dados, onde o processamento é relativamente simples. Em outros tipos de sistema (engenharia, científicos, tempo real), não se aplica com a mesma eficiência. Desta forma, algumas abordagens alternativas vêm sendo propostas.

Mark II Functions Points foi proposto como uma evolução da abordagem original de Pontos por Função (IFPUG). Nesta proposta, um sistema é considerado como uma coleção de transações lógicas consistentes. Com entrada, processamento e saída. O tamanho de uma transação é a soma do tamanho de suas entradas, saídas e processamento. O tamanho de um sistema é a soma dos tamanhos de suas transações.

Uma medida baseada em Mark II FP pode ser convertida para IFPUG através de uma fórmula que foi derivada empiricamente.

Além de Mark II, outras alternativas foram propostas na tentativa de resolver problemas apontados nesta abordagem: DeMarco's Function Weight (System Bang); 3D Function Points; Feature Points.

System Bang considera sistemas fortemente baseados em funções (conta o número de funções no DFD), dados (conta o número de elementos no modelo de dados), ou híbridos. 3D Function Points mede sistemas com processamento em tempo real e científicos. Feature Points mede sistemas com alta complexidade algorítmica. Nenhuma destas abordagens conseguiu alcançar o êxito pretendido, segundo os autores.

COSMIC (Common Software Measurement International Consortium) é um consórcio formado por vários pesquisadores com objetivo de prover uma nova

abordagem para medição e estimativa de software independente do tipo, prioritariamente software para apoio a negócios e tempo real (Full Function Points). Os princípios em que se baseia esta abordagem são: requisitos funcionais do software são definidos como conjunto de processos funcionais; software manipula pedaços de informação (Figura 1) chamados grupos de dados; processos funcionais envolvem sub-processos funcionais de acordo com movimento dos dados e transformações nos grupos de dados; o tamanho funcional é diretamente proporcional ao seu número de movimento de dados; o tamanho de uma aplicação é a soma dos tamanhos de seus processos funcionais.

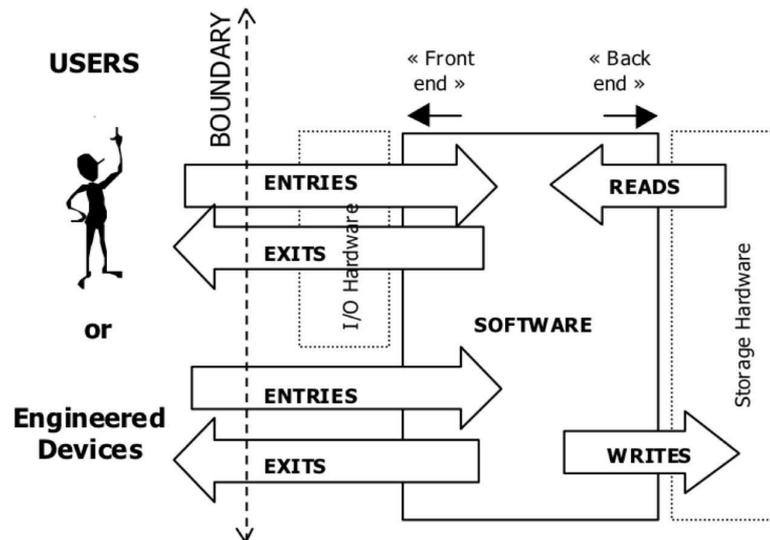


Figura 1 – Movimentos dos dados em Full Function Points.

COSMIC-FFP apresenta duas inovações importantes: camadas e pontos de vista de medidas (measurement viewpoints).

Especificamente para software orientado a objetos, várias abordagens foram propostas para adaptar os conceitos de FP para os elementos e modelos OO. Use Case Points calculam o tamanho do software com base no modelo de casos de uso, ao invés de contar transações e dados, conta-se atores e casos de uso. Além disso, o resultado obtido é um valor de esforço e não de tamanho como em IFPUG. O método propõe 3 estágios: identificação e classificação dos atores; identificação e classificação dos casos de uso e aplicação de um fator de ajuste.

2.11.2 Pontos positivos e pontos negativos

Os autores fazem uma descrição de cada método, comparam e apresentam um histórico de pesquisas na área.

Não há pontos negativos.

2.11.3 Contribuições para o projeto

Pode ser interessante olhar outros aspectos que não somente a quantidade de funções (ou atividades) para estimar projetos de modelagem de processos, tais como os

diversos métodos de FP fazem (por exemplos, quantidade de informação manipulada pode ser um bom indicativo de esforço para modelagem do processo).

2.12 Functional Size Measurement Revisited

2.12.1 Descrição

Os autores relacionam uma série de métodos para medir tamanho e estimar software baseados em pontos por função e afirmam que na medida em que vinham sendo apresentados, inconsistências entre eles eram detectadas. Desta forma, a International Organization of Standard (ISO) passou a definir padrões e atualmente 4 métodos são certificados: IFPUG, Mark II FP, COSMIC-FFP e NESMA FSM. Eles se diferenciam entre si pelo que contam e como contam, mas todos são baseados em funcionalidades disponibilizadas para usuários. Os autores realizaram dois estudos de caso para avaliar os métodos baseados em FP.

O primeiro estudo de caso teve como objetivo avaliar a aplicabilidade dos métodos na medição de projetos em diferentes domínios funcionais e por isso envolveu 3 casos.

O segundo estudo teve caráter exploratório e o objetivo foi avaliar a aplicação de vários métodos em diferentes fases de um projeto de software.

2.12.2 Pontos positivos e pontos negativos

Os autores descrevem em detalhes as questões que não funcionam nos métodos em seus estudos.

Não há pontos negativos.

2.12.3 Contribuições para o projeto

As questões descritas são basicamente conceituais. Ao tomar estes métodos como base, será importante verificá-las também.

2.13 Improving Size Estimates Using Historical Data

2.13.1 Descrição

Os autores comentam sobre a necessidade de se ter um processo ou método de estimativas de tamanho para desenvolvimento de sistemas. Porém mesmo sem um processo de estimativas definido, as equipes de desenvolvimento de software, no geral, já desenvolveram diversos softwares e portanto possuem uma grande quantidade de informações acerca do tempo para realizar cada desenvolvimento que pode ser utilizado para estimar desenvolvimento futuros.

Os autores então realizaram um estudo em cima de um grande projeto de um sistema desenvolvido em C++ voltado para geociência, o qual continha todos os dados de tamanho e tempo de desenvolvimento de cada fase e parte do software.

Eles estudaram se existia algum elemento específico que impactava no tempo de desenvolvimento de software como, por exemplo, os seguintes elementos mais

utilizados no projeto: Elementos GUI em um componente, mudanças de evento e estado tratados, número de funções por componente, número de componentes reutilizados.

Concluíram que sim e que ainda estes elementos impactam uns nos outros.

Foram construídos gráficos mostrando a correlação entre os elementos elencados e os estudos mostram que o tamanho do software está relacionado a diversos níveis de abstração do sistema e de quais elementos se adota em sua construção.

2.13.2 Pontos positivos e pontos negativos

Como ponto positivo, propõe a identificação da relação entre diferentes elementos do sistema.

Como ponto negativo, realiza a pesquisa baseado em um único projeto e nos elementos usualmente utilizados por uma equipe específica restringindo o estudo. Uma pequena equipe bastante uniforme.

2.13.3 Contribuições para o projeto

Não há contribuições expressivas para o projeto. A não ser por reforçar o fato de que existem variáveis que ao serem adicionadas em um projeto impactam em outras já existentes.

2.14 Learning how to improve effort estimation in small software companies

2.14.1 Descrição

As estimativas de esforço para desenvolvimento de software requerem o entendimento e conhecimento de fatores peculiares à organização e ao seu processo de desenvolvimento. Por exemplo, os métodos utilizados e a equipe de desenvolvimento influenciam no cálculo do esforço. É considerado pouco realista a existência de um modelo matemático ou sistema estatístico que determine com exatidão o esforço requerido.

Na prática, o uso da memória das pessoas (conhecimento), a avaliação e suposição dos estimadores continuam sendo a abordagem mais comumente utilizada para estimar esforço. Desta forma as estimativas são subjetivas. Mesmo usando alguma técnica conhecida, as estimativas de esforço produzidas são inerentemente subjetivas. Neste sentido, é proposto que a melhoria na exatidão das estimativas seja alcançada através da abordagem Bayesiana.

A inferência Bayesiana pode fornecer uma estatística da tendência de subjetividade do estimador durante a estimativa de atributos que são medidos em uma escala ordinal. Esta abordagem pode habilitar a produção uma distribuição de erros individual, para cada estimador, sobre uma escala ordinal de medida de esforço. Esta distribuição de erros pode ser usada pelos estimadores para que eles verifiquem o quanto suas estimativas diferem de uma estimativa de consenso de um grupo ou do esforço real de desenvolvimento dos sistemas. Este processo de aprendizado pode ser usado para melhoria da exatidão das estimativas. Entretanto, para determinar a

distribuição de erros é necessário coletar dados das estimativas, dos fatores relacionados e dos esforços reais.

O princípio básico deste artigo é que as características de uma pequena empresa de desenvolvimento de software requerem uma abordagem de suporte às estimativas de esforço que seja compatível com suas práticas de trabalho. O segundo princípio é que o suporte deve ser dado considerando a abordagem atual da organização para elaboração das estimativas, mesmo que não seja a mais apropriada. Mesmo existindo um bom método de estimativas apoiado por sistema é provável que haja restrições de recursos e tempo em pequenas organizações que impeçam a sua plena utilização. Ou seja, qualquer tipo de suporte deve permitir que as pequenas organizações utilizem seus próprios métodos de estimativas.

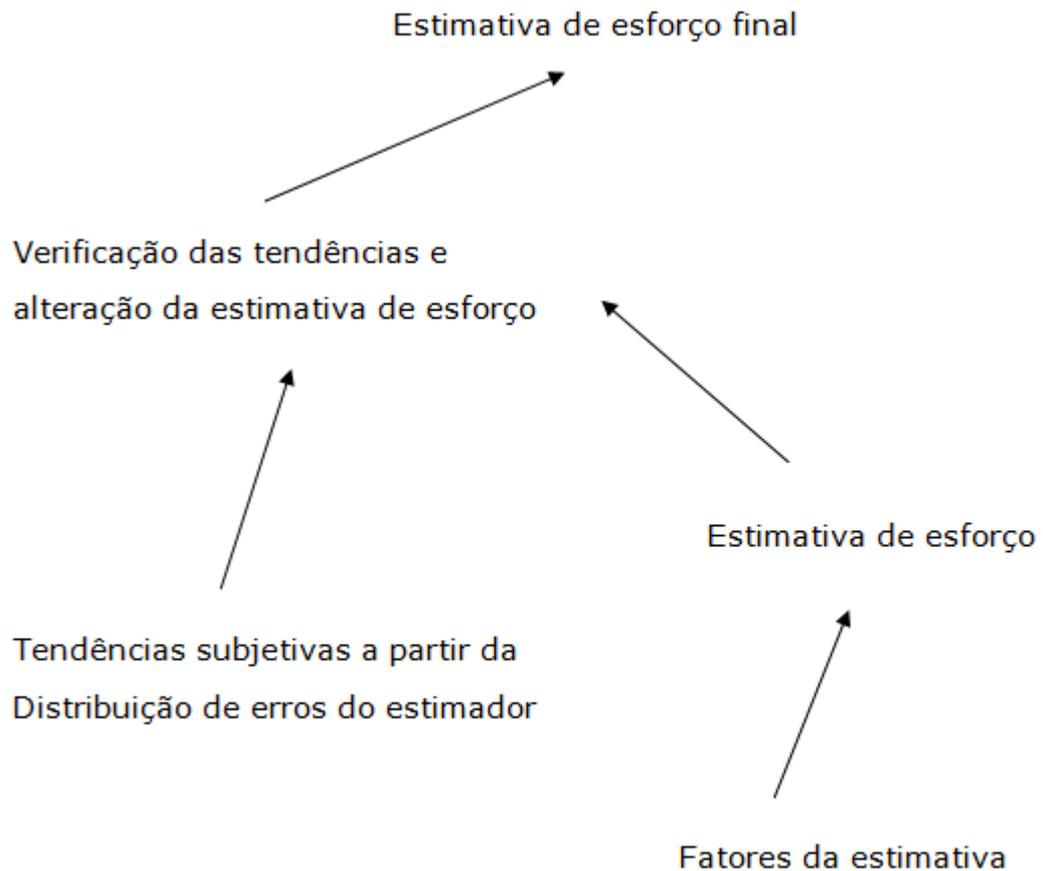
Pequenas empresas utilizam estimativas que incluem tendências subjetivas, a experiência e o conhecimento de seus desenvolvedores. Além disso, com frequência estão envolvidas com soluções e domínios de problemas completamente novos.

O suporte às estimativas de esforço em pequenas empresas é necessário em duas áreas: Primeiro para melhorar a exatidão das estimativas quando o domínio do problema e solução é conhecido. Segundo, para quando o domínio é desconhecido e neste caso, o conhecimento das tendências de subjetividade do estimador e sua classificação de valores de fatores do diferente domínio do problema e solução podem ajudar a melhorar futuras estimativas para domínios novos e desconhecidos. Coleta de dados e análise estatística podem reduzir o número de domínios desconhecidos.

Outro ponto de destaque é que as estimativas devem ser consistentes. Tendo em vista que elas são medidas subjetivas e não muito acuradas, sugere a utilização de uma escala de medida ordinal como base para os estimadores realizarem suas estimativas de comum acordo. A escolha de fronteiras/patamares apropriados para os intervalos da escala ordinal pode fornecer medidas consistentes.

O artigo faz referência a um manual (Gibbs Sampling) que pode ser usado para definir os parâmetros de distribuição de probabilidade que represente a distribuição dos erros dos estimadores quando o esforço real e os dados da estimativa são fornecidos para um modelo estatístico apropriado. Desta forma, julgamentos baseados na intuição e experiência são incorporados sistematicamente ao modelo em conjunto com as observações dos dados obtidos, a fim de obter estimativas mais confiáveis.

A abordagem Bayesiana para inferência da distribuição de erros pode ajudar os estimadores no aprendizado e melhoria de suas habilidades em domínios conhecidos. Isto pode ser alcançado através da leitura de suas próprias tendências subjetivas, medidas sobre uma gama de estimativas de esforço. Com base nas suas prováveis tendências de erro, os estimadores podem ajustar suas estimativas, conforme representado no desenho abaixo:



A prática corrente nas empresas que participaram do levantamento envolvia de uma a duas estimativas distintas: a primeira no estudo de viabilidade e a segunda após o levantamento de requisitos.

Nestas empresas, os fatores e atributos utilizados pelos estimadores para elaborarem suas estimativas são:

- Domínio do problema
 - Setor de negócio
 - Tipo de sistema
- Domínio da solução
 - Hardware
- Software (linguagens de programação)
 - Ferramentas
 - Métodos de desenvolvimento
- Nível de conhecimento do cliente/usuário sobre a aplicação/domínio do problema, medido numa escala ordinal.
- Conhecimento da equipe/estimador sobre:
 - Aplicação/domínio do problema

- Domínio da solução: linguagem, hardware e outras restrições técnicas, como por exemplo, velocidade de execução.
- ID do estimador.
- Numero de módulos que provavelmente serão necessários.
- Complexidade de cada módulo – avaliados em uma escala ordinal (simples, médio, complexo, muito complexo, extremamente complexo: 1, 2, 3, 4, 5).
- Dados adicionais podem ser coletados:
 - Esforço da fase de design
 - Esforço da fase de implementação
 - Esforço da fase de testes
 - “Overheads”

Cada estimador interpreta estes fatores usando sua experiência e ponto de vista pessoal. As estimativas de um projeto são frequentemente elaboradas por mais de um estimador.

No caso de domínios de problemas e soluções desconhecidos, a distribuição de erros existente pode mostrar como um estimador adapta suas estimativas como contingência em novos domínios. Os estimadores podem isolar esta informação usando a distribuição de erros do domínio antigo e voltar para reavaliar os fatores de ajuste e contingências usadas na estimativa para novos domínios quando os valores reais estiverem disponíveis.

2.14.2 Pontos positivos e pontos negativos

Os autores fazem uma proposta para melhoria das estimativas de forma independente do método ou técnica utilizada para estimar o esforço. A idéia é que os estimadores tenham um instrumento para entender o seu comportamento e ajustar as próximas estimativas.

Não há pontos negativos.

2.14.3 Contribuições para o projeto

A abordagem apresentada permite que o conhecimento prévio (histórico) seja considerado na realização/ajuste das estimativas, o que pode ser de interesse para o projeto.

2.15 The application of case-based reasoning to the software development process

2.15.1 Descrição

O artigo, apoiado em uma ferramenta comercial, demonstra um método em que o raciocínio baseado em casos (CBR) pode ser aplicado ao processo de desenvolvimento de software. Definição de requisitos, estimativa de esforço, desenho de software, localização de defeitos e manutenção são discutidos como candidatos à tecnologia

CBR, que é explorada como um mecanismo para melhoria da produtividade e qualidade do desenvolvimento de software.

O artigo descreve os passos do processo de raciocínio baseado em casos: avaliação da situação, pesquisa do caso, recuperação do caso, avaliação do caso e manutenção da base de casos. O protótipo apresentado consiste de casos que descrevem projetos de software. Cada projeto é identificado por um número único, nome, descrição, ambiente de hardware, linguagem ou pacote usado no desenvolvimento, número de pontos de função e cinco características (semelhantes aos fatores de ajuste da análise de pontos por função). O número de pontos de função é útil para limitar a pesquisa a projetos que provavelmente tenham esforços de desenvolvimento semelhantes.

Em relação a métodos ou técnicas de estimativas em projetos de software, o assunto praticamente não é explorado. São feitos apenas alguns comentários, tais como:

- CBR tem o potencial de reduzir o esforço, e conseqüentemente o custo, para produzir pontos por função através da facilidade de reutilização de módulos, objetos, desenhos etc.
- Com a retenção das informações das estimativas nos casos, a tecnologia CBR pode afetar de maneira positiva o processo de estimativas de esforço e custo, oferecendo uma forma de estimar baseada em um gama de experiência bem mais abrangente.

2.15.2 Pontos positivos e pontos negativos

Como ponto positivo, a proposta de uso de CBR para recuperação de históricos de estimativas.

Como ponto negativo o fato do artigo não ter foco em algum método ou técnica de estimativas que é o assunto de interesse do projeto.

2.15.3 Contribuições para o projeto

A sugestão de uso da tecnologia CBR pode ser interessante caso o projeto caminhe para uma solução baseada em estimativas por analogia com consultas intensivas em históricos de estimativas.

2.16 Use Case Estimation – The Devil is in the Detail

2.16.1 Descrição

Os autores fazem um resumo breve sobre as técnicas de Pontos de Função, Pontos por Caso de Uso e Estrutura Analítica do Projeto (EAP). E indicam que estas técnicas que hoje só são utilizadas, em geral, em estágios mais avançados do projeto deveriam ser usadas em conjunto desde o início do projeto, sempre mantendo a rastreabilidade entre estes artefatos. À medida que o projeto vá evoluindo cada um dos artefatos também vai. Com isso, poderia ser obtido um nível de detalhe de custo maior desde o início do projeto.

2.16.2 Pontos positivos e pontos negativos

Como ponto positivo, a construção da EAP pode dar uma visão melhor principalmente de relacionamentos e agrupamentos entre os requisitos. Também definir os casos de uso sem um nível de detalhe ainda maior pode ajudar na elaboração das estimativas iniciais, pois amplia a visão do engenheiro de software.

Como ponto negativo, podemos citar que numa fase ainda muito inicial os casos de uso podem tender a ficar muito superficiais e pouco detalhados não agregando muito valor ou fazendo muita diferença dos próprios requisitos funcionais e de infraestrutura em si. Outro ponto negativo é a falta de exemplos.

2.16.3 Contribuições para o projeto

O artigo cumpre um papel importante no projeto de pesquisa, pois chama a atenção sobre a possibilidade de elaborarmos estimativas ainda em fases iniciais do projeto. O que teria que ser feito é algum tipo de relação entre requisitos, casos de uso e EAP e artefatos produzidos em projetos de modelagem de processos.

3 Conclusões

Apesar da existência de artigos sobre métricas de qualidade para modelos de processos de negócio, não foi encontrada nenhuma publicação que abordasse ou fizesse referência a algum método ou técnica para realizar estimativas em projetos de modelagem de processos. Alguns autores [Gruhn, V. and Laue, R.,2006] ressaltam a similaridade entre projetos de desenvolvimento de software e projetos de modelagem de processos de negócio e, baseados nesta similaridade, propõe uma customização das métricas de software para área de modelagem de processos de negócio.

Esta constatação vem confirmar a expectativa do grupo, comprovando que a decisão de incluir na busca técnicas de estimação em projetos de software foi acertada.

4 Requisitos para a proposta de como estimar projetos de modelagem de processos

Conforme demonstrado na conclusão do relatório e nas análises dos artigos da revisão bibliográfica, os seguintes requisitos são relevantes para uma proposta de como elaborar estimativas em projetos de modelagem de processos:

4.1 Recuperar informações de projetos similares e suas estimativas, nas bases históricas.

Para viabilizar esta recuperação é necessário classificar os projetos de acordo com um conjunto de características que permita isolar uma classe com características e objetivos similares ao projeto que será desenvolvido. Existe um grande número de fatores de projeto que afetam o seu desenvolvimento, entre elas:

- Fatores relacionados à pessoal, tais como: experiência dos usuários, experiência da equipe de desenvolvimento na ferramenta, metodologia e diretrizes de modelagem de processos, bem como no domínio do negócio;
- Fatores relacionados ao problema, tais como: complexidade e frequência de alteração de requisitos;
- Fatores relacionados ao produto, tais como: tamanho e complexidade do modelo, quantidade de processos e/ou atividades envolvidas, nível de detalhamento das atividades;
- Fatores relacionados a recursos, tais como: disponibilidade de pessoas e facilidade de acesso aos usuários;
- Fatores relacionados ao desenvolvimento, tais como: objetivo do modelo (mapeamento do processo “as-is” e/ou “to-be”, derivação de requisitos, simulação etc.), a possibilidade de entregar o produto em partes, o paradigma adotado e o nível de riscos técnicos.

4.2 Possibilidade de elaboração de estimativas ainda em fases iniciais do projeto

5 Glossário

3D Function Points: Métrica para sistemas com processamento em tempo real e científicos.

Artificial Neural Network (ANN): Redes neurais artificiais representam um conceito da computação que visa trabalhar no processamento de dados de maneira semelhante ao cérebro humano.

Case-Based Reasoning (CBR): Raciocínio Baseado em Casos é uma tecnologia emergente em representação e processamento de conhecimento que utiliza a experiência passada para resolver problemas.

Common Software Measurement International Consortium (COSMIC): Consórcio formado por vários pesquisadores com objetivo de prover uma nova abordagem para medição e estimativa de software independente do tipo, prioritariamente software para apoio a negócios e tempo real (Full Function Points).

Control-flow Complexity (CFC): Métrica para análise da complexidade do fluxo de processos.

COSMIC-FFP Size Unit (CFSU): Unidade de tamanho para COSMIC-FFP.

COSMIC-FFP: Método proposto em 2000, na prática foi um refinamento do método FFP. Ainda não é uma técnica tão disseminada no mundo quanto à do IFPUG, porém observa-se que muita pesquisa está sendo realizada sobre este método.

Cost Constructive Model (COCOMO): método que busca estimar esforço, prazo, custo e tamanho de equipe, necessários ao desenvolvimento do software, desde que se tenha como premissa a sua dimensão.

Diagrama de Fluxo de Dados (DFD): representa o fluxo de dados num sistema de informação, assim como as sucessivas transformações que estes sofrem. O DFD é uma ferramenta gráfica que transcreve, de forma não técnica, a lógica do procedimento do sistema em estudo, sendo usada por diferentes métodos e principalmente pelos classificados como orientados a processos.

Estimeeting: termo definido pelos autores para denominar uma reunião estruturada onde técnicos com grande experiência se reúnem para estimar o esforço necessário para implementar e testar uma funcionalidade em um projeto de software. Estas decisões são tomadas com base em um documento funcional e um documento técnico, gerados para cada funcionalidade por uma equipe destacada para demonstrar a importância e viabilidade desta funcionalidade.

Estrutura Analítica do Projeto (EAP): “Uma decomposição hierárquica orientada à entrega do trabalho a ser executado pela equipe do projeto para atingir os objetivos do projeto e criar as entregas necessárias. Ela organiza e define o escopo total do projeto”. (PMBOK)

Feature Points: Métrica para sistemas com alta complexidade algorítmica.

Function Points (FP): Método de análise de pontos por função.

Functional Size Measure (FSM): Medida do tamanho funcional.

Graphical User Interface (GUI): Interface gráfica com o usuário que permite a interação homem-computador.

Handle: O número de handles é um termo das redes de Petri utilizado para medir a quantidade de pares de nós mal estruturados. Esta métrica aplicada em modelos de workflow mede o número de construções “split/join” mal-estruturadas.

International Function Point Users’ Group (IFPUG): é uma organização sem fins lucrativos, cuja missão é promover e incentivar a gestão eficaz das atividades de desenvolvimento e manutenção de software através da utilização da Análise de Pontos por Função Análise e outras técnicas de medição de software.

Inferência Bayesiana: é um tipo de inferência estatística que descreve as incertezas sobre quantidades invisíveis de forma probabilística. Incertezas são modificadas periodicamente após observações de novos dados ou resultados. A operação que calibra a medida das incertezas é conhecida como operação bayesiana e é baseada na fórmula de Bayes. A fórmula de Bayes é muitas vezes denominada Teorema de Bayes.

Lines of Code (LOC): Número de linhas de código.

Mark II FP: Variação mais antiga do método de pontos de função padrão do IFPUG, foi apresentada em Londres por Charles Symons em 1983. Atualmente é um método de domínio público mantido pela associação de métricas do Reino Unido - UKSMA.

NESMA FSM: Método para medida do tamanho funcional que utiliza a mesma filosofia, conceitos, termos e regras do IFPUG, com algumas diretrizes diferentes.

Netherlands Software Metrics Association (NESMA): Associação de métricas da Holanda mantém seu próprio manual de contagem, cuja primeira versão em 1990 foi baseada no manual do IFPUG.

Ordinal Regression Model: Modelo de regressão ordinal. É uma abordagem probabilística para realização de estimativas.

Workflow pattern / anti-pattern: Coleção de padrões de workflow, referentes a problemas recorrentes e soluções demonstradas, que fornecem um exame completo das várias perspectivas (fluxo de controle, dados, recursos e manipulação de exceções) que precisam ser suportadas por linguagens de workflow ou de modelagem de processos de negócio.

Referências Bibliográficas

BIELAK J. "Improving size estimates using historical data" In: IEEE Software, Vol.17, Issue 6, p. 27-35, Nov/Dec 2000.

BIOLCHINI, J.; MIAN, P. G.; NATALI, A. C. C.; TRAVASSOS, G. H. "Systematic Review in Software Engineering". In: Technical report RT-ES 679/05. PESC - COPPE/UFRJ, 2005

BOWDEN P., HARGREAVES M., LANGENSIEPEN C.S. "Estimation support by lexical analysis of requirements documents" In: The Journal of Systems and Software 51, Elsevier, p. 87-98, 2000.

CARDOSO J. , VANDERFEESTEN I., REIJERS H.A. "A weighted coupling metric for business process models" In: Proceedings of CAiSE, 2007

GENCEL C., DEMIRORS O. "Functional size measurement revisited" In: ACM Transactions on Software Engineering and Methodology, Vol. 17, No. 3, Article 15, Jun. 2008.

GRUHN, V. AND LAUE R. "Complexity Metrics for Business Process Models" In: 9th International Conference on Business Information Systems, Klagenfurt, Austria, 2006

GRUPE F.H., URWILER R., RAMARAPU N.K., OWRANG M. "The application of case-based reasoning to the software development process" In: Information and Software Technology 40, Elsevier, p. 493-499, May 1998

HUGHES R.T. "Expert judgement as an estimating method" In: Information and Software Technology 38, Elsevier, p.67-75, 1996

KITCHENHAM, B. A. "Procedures for Performing Systematic Reviews", Software Engineering Group - Keele University - United Kingdom and Empirical Software Engineering, National ICT Australia Ltd., 2004

LOKAN, C.J. "Function Points" In: Advances in Computers, Vol. 65, Chapter 7, Academic Press, p. 298-349.

MOHAGHEGHI P., ANDA B., CONRADI R. "Effort estimation of use cases for incremental large-scale software development" In: 27th International Conference on Software Engineering, St Louis, MO, p.303-311, May 2005.

MOSES, J. CLIFFORD, J. "Learning how to improve effort estimation in small software development companies" In: Computer Software and Applications Conference (COMPSAC), The 24th Annual International, Taipei, Taiwan, p. 522-527, Oct. 2000

PILLAI K., SUKUMARAN NAIR V.S. "A model for software development effort and cost estimation" In: IEEE Transactions on Software Engineering, vol. 23, n^o.8, Aug. 1997

SHEPPERD M., SCHOFIELD C., KITCHENHAM B. "Effort estimation using analogy" In: Proceedings of the 18th International Conference on Software Engineering, Berlin, Germany, p. 170-178, Mar. 1996

TAFF L.M., BORCHERING J.W., AND JR., HUDGINS W.R. "Estimeetings: Development Estimates and a Front-End Process For a Large Project" In: IEEE Transactions on Software Engineering, vol. 17, no. 8, august 1991

TRONTO I., SILVA J., SANT'ANNA N., "Comparison of Artificial Neural Network and Regression Models in Software Effort Estimation" In: Proceedings of International Joint Conference on Neural Networks, Orlando, Florida, USA, August 12-17, 2007

VINSEN, K., JAMIESON, D. & CALLENDER, G. "Use case estimation - the devil is in the detail" In: IEEE Joint International Requirements Engineering Conference RE'04, IEEE Computer Society, Kyoto, Japan, 2004.

VOGELEZANG F.W., "Early estimating using COSMIC-FFP" In: Proceedings of the 2nd Software Metrics European Forum (SMEF), Roma, 2005

Anexo A – Planejamento e execução da revisão sistemática

A.1 Introdução

O objetivo deste documento é apresentar o planejamento da revisão sistemática que será realizada no contexto do projeto de Governança.

Motivação: Necessidade de validar os prazos estabelecidos em propostas de fornecedores para projetos em modelagem de processos de negócio. As propostas são elaboradas após reuniões de levantamento com o cliente para definição do tipo e quantidade de processos que fazem parte do escopo do projeto.

Fora do escopo de estudo: a) Levantamento de dados para cálculo de produtividade da equipe de modelagem. Estes dados serão extraídos do histórico de projetos de modelagem desenvolvidos pela UNIRIO/NP2Tec dentro da Área de Exploração e Produção da Petrobras; b) Estimativas de custo.

Este trabalho faz parte das iniciativas de pesquisa do Termo de Cooperação para modelagem de processos entre a NP2Tec/UNIRIO e a Petrobras/TIC-E&P/GDIEP.

A.2 Planejamento da revisão

A.2.1 Objetivo

Procurar trabalhos sobre estimativas de esforço para a realização de projetos de modelagem de processos de negócio e desenvolvimento de software que possam ser aplicados aos projetos de modelagem de processos de negócio na Petrobras. No caso de desenvolvimento de software, o interesse da pesquisa está voltado para estimativas baseadas em requisitos de negócio obtidos nas fases iniciais de projetos de sistemas de informação.

A.2.2 Formulação da Pergunta (*Question Formularization*)

A.2.2.1 Pergunta (*Question Focus*)

Como estimar o tamanho, estimar ou derivar o esforço e prazo de projetos de modelagem de processos de negócio?

A.2.2.2 Abrangência e qualidade da pesquisa (*Question Quality and Amplitude*)

- **Problema/Questão** (*Problem/Question*)

Pesquisar e elaborar um método específico de estimativas para projetos de modelagem de processos de negócio.

- **Palavras chave e sinônimos** (*Keywords and synonyms*)

Português: Estimar, estimativa, estimação, medir, medição, métrica, tamanho, esforço, prazo, tempo, modelo, modelagem de processo, BPM, business process management, processo de negócio, desenho de processo, pontos por função, pontos por caso de uso.

Inglês: Estimate, estimation, calculate, calculation, measure, measurement, metric, size, sizing, effort, deadline, term, time, model, business modeling (modelling), BPM, process management, business process, process design, function points, use case points.

- **Intervenção** (*Intervention*)

Metodologias de estimativas orientadas à modelagem de processos e fases de desenho e análise de soluções.

- **Controle**

Volker Gruhn and Ralf Laue: Complexity Metrics for Business Process Models. Chair of Applied Telematics / e-Business, Computer Science Faculty, University of Leipzig, Germany.

- **Resultados esperados** (*Effect*)

Lista de metodologias para estimativas de projetos de modelagem de processos

Lista de metodologias para estimativa de projetos de desenvolvimento de sistemas com ênfase na fase de desenho.

Lista de ferramentas para realização de estimativas de projetos.

- **Comparação** (*Outcome measure*)

Não se aplica.

- **População** (*Population*)

Trabalhos publicados em conferências e periódicos relatando estudos e propostas de metodologias para estimativa de projetos de modelagem de processos e desenvolvimento de software e/ou exemplos de suas aplicações e ferramentas que implementam as propostas.

- **Forma de condução da experimentação**

Não se aplica.

A.2.3 Seleção de fontes (*Sources Selection*)

A.2.3.1 Definição de critérios para seleção de fontes (*Sources Selection Criteria Definition*)

- Disponibilidade das referências, não necessariamente em sua íntegra, na Internet ou em formato impresso.
- Utilização de portais considerados relevantes para os pesquisadores da área estudada.
- Existência de mecanismos de pesquisa para levantamento de referências
 - Desejável: Possuir mecanismo de busca que permita o uso de expressões lógicas;
 - Os mecanismos de busca deverão permitir a busca no resumo (abstract) e/ou texto completo da publicação.

A.2.3.2 Idiomas (*Studies Languages*)

Português e inglês.

A.2.3.3 Fontes de estudo (Sources Identification)

A.2.3.3.1 Método de busca (Sources search methods)

Via Web, através de expressões de busca pré-estabelecidas.

As publicações oriundas de fontes que não possuam mecanismos de busca serão analisadas manualmente, quando disponíveis, considerando as expressões de busca definidas.

As publicações não-digitais serão analisadas manualmente considerando as expressões de busca pré-estabelecidas.

A.2.3.3.2 Expressão de busca (Search string)

Para publicações em português deve-se utilizar a expressão abaixo:

(técnica OR método OR metodologia OR mecanismo OR modelo)

AND

(estimar OR estimativa OR estimativas OR estimação OR medir OR medição OR medições OR métrica OR métricas)

AND

(tamanho OR esforço OR prazo OR tempo OR “pontos por função” OR “pontos por caso de uso”)

AND

(“modelagem de processo” OR “modelagem de processos” OR “BPM” OR “processo de negócio” OR “processos de negócio” OR “desenho de processo” OR “desenho de processos” OR “modelagem de negócio”)

Para publicações em inglês deve-se utilizar a expressão abaixo:

(technic OR technique OR method OR methodology OR mechanism OR model OR models)

AND

(estimate OR estimation OR measure OR measurement OR metric OR metrics)

AND

(size OR sizing OR effort OR deadline OR term OR time OR “function point” OR “use case point” OR UCP OR AFP OR NAFFP)

AND

("process modeling" OR BPM OR "business process" OR "process design" OR "business modeling")

A.2.3.3.3 Relação inicial das fontes de estudo (Source list)

Fonte
Compendex – http://www.engeneeringvillage.com/
Citeseer - http://citeseer.ist.psu.edu/
Editora Springer - http://www.springer.com/
Google Scholar
Portal da Association for Computing Machinery - http://portal.acm.org/dl.cfm
Portal da IEEE - http://www.ieee.org/portal/site
Scopus – http://www.scopus.com/

A.2.3.4 Seleção de fontes após validação (Sources Selection after Evaluation)

À princípio, todas as referências com texto completo obtidas serão analisadas pela equipe.

A.2.3.5 Checagem de referências (References Checking)

Não se aplica

A.2.4 Seleção de estudos (Studies Selection)

A.2.4.1 Definição dos tipos de estudos (Studies types definition)

Não se aplica

A.2.4.2 Procedimentos para seleção de estudos (*Procedures for studies selection*)

O analista aplicará a estratégia de busca para a identificação de potenciais publicações. As publicações identificadas serão selecionadas (1º filtro) através da leitura das seções de abstract e conclusão e verificação dos critérios de inclusão e exclusão estabelecidos.

Posteriormente a lista de publicações excluídas será avaliada pelo grupo. Em caso de conflito, a publicação será incluída. Após a identificação das publicações, através do critério dos mecanismos de busca, os documentos selecionados serão lidos e resumidos (2º filtro) para extração de informações sobre metodologias para estimativas de projetos de modelagem de processos de negócio. Nesta etapa, algumas publicações podem ser excluídas considerados os critérios estabelecidos. Para esta etapa, vale a mesma regra anterior, ou seja, caso haja impasse entre os pesquisadores, a publicação deve ser mantida na lista.

A.2.4.3 Critérios de inclusão e exclusão (*Studies inclusion and exclusion criteria definition*)

Critérios de exclusão:

- **CE1:** Não serão selecionadas as publicações sem referência a um método para estimar tamanho, esforço e/ou prazo de projetos de modelagem de processos ou desenvolvimento de software.
- **CE2:** Não serão selecionadas as publicações cuja abordagem das estimativas não mencione projetos de modelagem de processos de negócio ou desenvolvimento de software.
- **CE3:** Não serão selecionadas as publicações em que a palavra BPM não significar Business Process Management.

Critérios de inclusão:

- **CI1:** Podem ser selecionadas publicações que apresentem um método para estimativa ou medição de tamanho, esforço e prazo, qualquer que seja a fase do projeto de modelagem de processos.
- **CI2:** Podem ser selecionadas publicações que descrevam protótipos de métodos ou ferramentas relacionados à estimativa de projetos.
- **CI3:** Devem ser consideradas as referências bibliográficas que apareçam nas referências das publicações e sejam consideradas relevantes apesar de não terem sido identificadas pelas palavras chave do estudo.

A.3 Execução da revisão

A.3.1 Lista de Publicações de Controle

Na fase de planejamento, o artigo abaixo [Gruhn, V. and Laue R.] foi citado como controle, mas na verdade era uma referência a ser utilizada no projeto independente dela aparecer ou não na revisão. A intenção não era utilizá-lo apenas para refinamento da expressão de busca. No momento em que os testes preliminares foram iniciados, a segunda publicação foi incorporada ao grupo de controle:

- i. GRUHN, V. AND LAUE R. "Complexity Metrics for Business Process Models" In: 9th International Conference on Business Information Systems, Klagenfurt, Austria, 2006
- ii. VOGELEZANG, F.W., "Early estimating using COSMIC-FFP", In: Proceedings of the 2nd Software Metrics European Forum (SMEF 2005), Roma, Mar. 2005

A.3.2 Histórico das Buscas

Os primeiros testes das máquinas de busca foram realizados com a expressão planejada (item A2.3.3.2). Em relação às fontes de estudo foram verificados os seguintes problemas:

- Compendex: Não foi possível acessar o sistema, uma vez que era necessário informar conta de usuário e senha.
- Google Scholar e CiteSeer: Não foi possível executar a pesquisa, provavelmente devido a restrições em relação ao tamanho da expressão de busca.
- Portal da ACM: Não obteve resultados. Como é conhecida por ter problemas no seu mecanismo de busca, esta base foi descartada.

Optou-se então pela Scopus, que apresentou boas condições para realização das buscas, oferecendo a facilidade de exportação dos resultados da pesquisa para uma planilha no formato CSV e retornando artigos de outras bases inclusive da IEEE e Springer.

Com a expressão planejada (item A2.3.3.2), a base da Scopus retornou 181 resultados, com vários artigos fora do escopo da pesquisa. A partir daí foram realizadas diversas rodadas com variações da expressão de busca obtendo-se os seguintes resultados:

TITLE-ABS-KEY((technic OR technique OR method OR methodology OR mechanism OR model OR models OR metric OR metrics) AND ({estimate size} OR {size estimation} OR {estimate effort} OR {effort estimation} OR {function point} OR

{use case point} OR UCP OR AFP OR NAFF) AND ({process modeling} OR BPM OR {business process} OR {process design} OR {business modeling} OR {information system} OR {information systems})) AND SUBJAREA(comp)

Resultado: 26 artigos.

TITLE-ABS-KEY((technic OR technique OR method OR methodology OR mechanism OR model OR models OR metric OR metrics) AND ({estimate size} OR {size estimation} OR {estimate effort} OR {effort estimation} OR {function point} OR {function points} OR {use case point} OR {use case points} OR UCP OR AFP OR NAFF) AND ({process modeling} OR BPM OR {business process} OR {process design} OR {business modeling} OR {information system} OR {information systems})) AND SUBJAREA(comp)

Resultado: 33 artigos.

TITLE-ABS-KEY(early AND estimation AND size AND (requirement OR requirements)) AND SUBJAREA(comp)

Resultado: 33 artigos.

TITLE-ABS-KEY((earl* OR initial) AND estimat* AND (size OR cost OR effort) AND (project OR requirement OR business OR process)) AND SUBJAREA(comp)

Resultado: 595 artigos.

Expressão de busca final

TITLE-ABS-KEY((requirement OR functionality) AND estimat* AND (size OR effort) AND (project OR business OR process)) AND SUBJAREA(comp)

Resultado: 249 artigos.

O executor da pesquisa analisou individualmente os campos de “título” e “abstract” dos 249 artigos, considerando os critérios de inclusão e exclusão, até chegar a um resultado de 79 artigos. Em seguida, os “abstracts” destes 79 artigos foram lidos por 3 dos integrantes do grupo. Cada integrante fez sua avaliação individual indicando se deveria ser lido integralmente o artigo ou não. Numa reunião da equipe estas avaliações foram discutidas e os conflitos existentes foram resolvidos através de consenso e votação, tendo como resultado os 28 artigos listados a seguir:

- How to use COSMIC functional size in effort estimation models? Gencil C. 2008
- Improving quality of functional requirements by measuring their functional size Trudel S., Abran A. 2008

- Project sizing and estimating: A case study using PSU, IFFUG and COSMIC Buglione L., Cuadrado-Gallego J.J., De Mesa J.A.G. 2008
- Uncertainty in ERP effort estimation: A challenge or an asset? Daneva M., Wettflower S., De Boer S. 2008
- **A metamodeling approach to estimate software size from requirements specifications** Abrahao S., Insfran E. 2008
- Impact of base functional component types on software functional size based effort estimation Buglione L., Gencel C. 2008
- Integrating portfolio management and simulation concepts in the ERP project estimation practice Daneva M. 2008
- **Functional size measurement revisited** Gencel C., Demirors O. 2008
- **Comparison of artificial neural network and regression models in software effort estimation** De Barcelos Tronto I.F., Simoes Da Silva J.D., Sant'Anna N. 2007
- A flexible method for software effort estimation by analogy Li J., Ruhe G., Al-Emran A., Richter M.M. 2007
- A functional size measurement method for object-oriented conceptual schemas: Design and evaluation issues Abrahao S., Poels G., Pastor O. 2006
- A requirement-based project estimation approach Chang C.-P., Wu C.-S., Chu C.-P., Lv J.-L. 2005
- **Effort estimation of use cases for incremental large-scale software development** Mohagheghi P., Anda B., Conradi R. 2005
- **Function Points** Lokan C.J. 2005
- **Use case estimation - The devil is in the detail** Vinsen K., Jamieson D., Callender G. 2004
- Fuzzy modeling for function points analysis De Souza Lima Jr. O., Farias P.P.M., Belchior A.D. 2003
- Measuring Effort Estimation Uncertainty to Improve Client Confidence Moses J. 2002
- **Learning how to improve effort estimation in small software development companies** Moses John 2000
- **Improving size estimates using historical data** Bielak J. 2000
- **Estimation support by lexical analysis of requirements documents** Bowden P., Hargreaves M., Langensiepen C.S. 2000
- A Rule-Based Approach to Developing Software Development Prediction Models Chatzoglou P.D., Macaulay L.A. 1998
- **The application of case-based reasoning to the software development process** Grupe F.H., Urwiler R., Ramarapu N.K., Owrang M. 1998
- **Estimating software project effort using** Shepperd M., Schofield C. 1997

analogies

- | | | |
|--|--|-------------|
| • A model for software development effort and cost estimation | Pillai K., Sukumaran Nair V.S. | 1997 |
| • Expert judgement as an estimating method | Hughes R.T. | 1996 |
| • A structured methodology for software development effort prediction using the analytic hierarchy process | Lee H. | 1993 |
| • Estimeetings: Development estimates and a front-end process for a large project | Taff Louis M., Borchering James W., Hudgins Jr. W.Richard | 1991 |
| • A model for measuring information system size | Wrigley C.D., Dexter A.S. | 1991 |

Iniciou-se então o trabalho de recuperação dos artigos para que fossem lidos e resumidos. Neste momento, foi observado que através da rede da UNIRIO, alguns downloads não traziam o conteúdo dos artigos, baixando apenas o respectivo “abstract”. Sendo assim, no final desta etapa foram obtidos os 14 artigos que aparecem em destaque na relação acima.

Um dos artigos obtidos (Abraham S., Insfran E., 2008, “A metamodeling approach to estimate software size from requirements specifications”) foi considerado como falso positivo após sua leitura, descartando-se a necessidade de elaboração de seu resumo.

Neste momento, o artigo de Irene Vanderfeesten, Jorge Cardoso, Hajo A. Reijers, “A weighted coupling metric for business process models” foi incluído na relação para leitura, por indicação de um dos integrantes do grupo. Este artigo não foi encontrado pelo processo de revisão sistemática, mas se mostrou de interesse para a pesquisa.

Os 14 artigos resultantes foram então lidos integralmente e resumidos, permitindo a identificação dos pontos fortes e fracos de cada um e suas contribuições para o projeto em questão.

Anexo B – Relatório de estatísticas de projetos de modelagem de processos

B.1 Introdução

O objetivo deste documento é apresentar o estudo realizado com base nos projetos de modelagem de processos já realizados pela UNIRIO/NP2Tec em diversos Escritórios de Processos da TIC/E&P para prover um mecanismo de estimação do esforço médio necessário para a realização de um projeto desta natureza, de acordo com o tipo de projeto e o número de processos modelados. Trata-se de um estudo inicial, realizado com restrições de tempo para levantamento dos dados, organização dos mesmos e realização de análises alternativas, visando apresentar resultados iniciais que possam apoiar decisões de investimento em projetos de modelagem. Estão sendo organizados estudos mais profundos no sentido de prover uma estimativa mais confiável, ainda que dependente de dados mais específicos sobre o projeto em análise.

B.2 Estudo realizado

B.2.1 Condução do estudo prévio

Durante o estudo foram levantadas informações sobre os projetos de modelagem de processos realizados pela UNIRIO/NP2TEc dentro do E&P. Estas informações incluíram, para cada projeto, o tempo total gasto no projeto (em dias úteis de trabalho), o número de processos modelados e o número de participantes, bem como seus perfis.

Para realizar este estudo inicialmente coletamos os dados supracitados dos projetos que foram realizados junto ao E&P. Selecionamos 40 (quarenta) projetos de modelagem de processos, dentre os quais dois foram excluídos por conta dos dados existentes estarem muito fora da curva padrão o processo de coleta. Os dados dos 38 (trinta e oito) projetos restantes foram tabulados, calculando-se o esforço ponderado exigido por cada projeto, de acordo com o número de participantes envolvidos.

Em seguida, os projetos selecionados foram classificados em três categorias, de acordo com informações fornecidas por pessoas do NP2TEc que participaram dos projetos. Os tipos de projetos definidos foram: ADM (processos administrativos), TGE (processos técnicos com foco em gestão) e TOP (processos técnicos operacionais). Analisando-se graficamente os dados destes projetos, eliminamos mais um projeto, que se tratava de um caso extremo (outlier) em relação a seus pares.

Depois, os projetos de cada grupo foram classificados como pequenos, médios ou grandes, utilizando-se um critério que visava reduzir o desvio padrão intra-grupo. Devido ao número relativamente reduzido de projetos (em relação aos nove subgrupos pretendidos) e à concentração destes projetos em determinados tamanhos, optamos por uma divisão de cada grupo em dois subgrupos: pequeno/médio e grande.

Finalmente, calculamos a média e o desvio padrão de cada grupo, estabelecendo intervalos de confiança para os subgrupos. Em casos onde tivemos subgrupos com apenas um projeto, sendo impossível calcular o desvio padrão, fizemos uma estimativa baseada em experiência e no desvio padrão de outros subgrupos.

Foram realizadas análises rápidas levando em consideração outros aspectos dos projetos. No entanto, estas análises apontaram falta de correlação entre os fatores escolhidos, de modo que nada se pode concluir a respeito até o presente momento. Novas análises estão sendo planejadas, junto a uma revisão sistemática de mecanismos de estimativa similares ao que propomos, no sentido de refinar as estimativas em um horizonte de médio prazo.

B.3 Proposta de estimativa

Baseado nos estudos realizados foram consideradas para a proposta inicial de estimativas as informações que certamente são levantados durante o levantamento inicial do projeto: Domínio e processos (em números) a serem modelados. Qualquer outra informação adicional pode não estar disponível e varia bastante dependendo de cada projeto e por este motivo, além dos já citados na seção anterior, não foram consideradas neste estudo.

Tipos de processos:

TOP - Técnico operacional	Processos do E&P característicos de práticas operacionais de exploração e produção onde há necessidade de descrição e modelagem detalhada dos procedimentos de execução das atividades.
TGE - Técnico de gestão	Processos de acompanhamento, análise e gestão das operações de exploração e produção onde a descrição e modelagem dos procedimentos de execução das atividades são detalhadas em um nível mais alto, pelas características destes tipos de processos onde ocorrem análises e negociações.
ADM - Administrativo	Processos executados por áreas administrativas e de apoio à exploração e produção onde a descrição e modelagem dos procedimentos de execução das atividades possuem detalhamento que varia de acordo com a necessidade do Gestor de processos.

Características dos projetos:

A classificação de um projeto leva em consideração os tipos e a quantidade de processos modelados.

Os projetos podem ser classificados em:

PP - Pequeno/Médio Projeto	GP - Grande Projeto
-----------------------------------	----------------------------

Classificação proposta:

Nº Processos	Tipo do processo	Classificação
Até 5	Técnico operacional	PP
6 em diante	Técnico operacional	GP
Até 6	Administrativo	PP
7 em diante	Administrativo	GP
Até 4	Técnico de gestão	PP
5 em diante	Técnico de gestão	GP

Esforço médio de realização dos projetos:

Classificação	Tipo de processo	Número de Processos Analisados	Média do esforço total (em homens/dia)	Desvio padrão (em homens/dia)
PP	TGE	14	56	31
PP	ADM	13	84	20
PP	TOP	6	160	60
GP	TGE	1	100	40
GP	ADM	1	162	30
GP	TOP	2	230	80

A média do esforço leva em consideração a atuação de 1 analista/hora (8hrs/dia) dedicado a esta atividade.

Para todos os projetos levou-se em consideração a participação (não contabilizados na tabela acima) de:

- 1 líder de projetos em tempo parcial (4hrs/dia) → 0,5 líder de projeto
- 1 gerente de projetos dedicado em tempo parcial (3,2hrs/dia) → 0,4 gerente

Perfis:

Analista modelador	Responsável pelo levantamento e modelagem de processos de negócio. Este trabalho inclui promover a discussão e servir como facilitador da mesma junto aos stakeholders coletando as informações necessárias e representando em modelos de processo de acordo com padrões internos de modelagem.
---------------------------	---

	<p>Deve possuir as seguintes habilidades:</p> <ul style="list-style-type: none"> - Capacidade de pensamento estruturado/algorítmico; - Capacidade de abstração e classificação em grupos; - Facilidade de comunicação; - Bons conhecimentos do pacote Microsoft Office; - Bons conhecimentos na ferramenta de modelagem de processos a ser utilizada no projeto; - Boa capacidade de redação; - Boa capacidade de relação interpessoal e negociação; <p>As formações preferenciais incluem: engenharia de produção e áreas correlatas a TI ou administração.</p>
Líder de projeto	<p>Responsável por coordenar o levantamento e modelagem de processos com relação a estratégia de condução das entrevistas junto aos stakeholders e a estratégia de modelagem de processos escolhida, garantir a aderência dos modelos de processos aos padrões internos de modelagem, gerenciar o plano de projeto controlando o cronograma do projeto.</p> <p>Deve possuir as seguintes habilidades:</p> <ul style="list-style-type: none"> - Capacidade de pensamento estruturado/algorítmico; - Capacidade de abstração e classificação em grupos; - Facilidade de comunicação; - Bons conhecimentos do pacote Microsoft Office; - Conhecimento na ferramenta de modelagem de processos a ser utilizada no projeto; - Boa capacidade de redação; - Boa capacidade de relação interpessoal e negociação; - Conhecimentos em gestão de projetos; - Aptidão para gestão de pessoas e conflitos; <p>As formações preferenciais incluem: engenharia de produção e áreas correlatas a TI ou administração.</p>
Gerente	<p>Responsável por gerenciar o projeto junto ao contratante com relação à gestão de tempo, esforço, custo e escopo do projeto de modelagem, bem como gerir a expectativas dos gestores do projeto e servir como ponte de comunicação do gerente contratante com a equipe de modelagem de processos.</p> <p>Deve possuir as seguintes habilidades:</p> <ul style="list-style-type: none"> - Capacidade de pensamento estruturado/algorítmico; - Conhecimento de ferramentas de modelagem de processos; - Capacidade de abstração e classificação em grupos;

	<ul style="list-style-type: none"> - Facilidade de comunicação; - Boa capacidade de redação; - Boa capacidade de relação interpessoal e negociação; - Conhecimentos em gestão de projetos; - Aptidão para gestão de pessoas e conflitos; <p>As formações preferenciais incluem: engenharia de produção e áreas correlatas a TI ou administração.</p>
--	---

A incorporação de mais analistas modeladores na equipe de modelagem de processos pode reduzir o esforço médio total do projeto da seguinte forma:

Inclusão de analista modelador	Total de analistas modeladores	Porcentagem média de diminuição do esforço médio
1	2	80% do esforço médio total
2	3	75% do esforço médio total
3 em diante	4 em diante	70% do esforço médio total

Essa relação é explicada pela:

- Grande integração existente entre os processos modelados o que requer um tempo de dedicação dos analistas em realizar análises em conjunto;

- A baixa disponibilidade dos stakeholders (participantes e interessados nos processos) para realizar reuniões de levantamento de processos;

- A necessidade de aumentar o esforço do Líder de projeto e do Gerente em função do aumento na demanda por mentoring e auditoria interna de processos bem como do aumento da própria gestão do mesmo.

No entanto esse porcentual médio de diminuição do esforço tende a diminuir em maior grau considerando o paralelismo de modelagens que requerem entrevistas com stakeholders distintos.