# Uso Combinado de Análise de Pontos de Função e Pontos de Casos de Uso na Gestão de Estimativa de Tamanho de Projetos de Software Orientado a Objetos

Edméia Leonor Pereira de Andrade[1], Káthia Marçal de Oliveira[2]

[1]Empresa Brasileira de Pesquisa Agropecuária
Parque Estação Biológica - Av. W3 Norte. Ed. Sede. CEP 70770-901 Brasília –DF
edmeia.andrade@embrapa.br

[2]Universidade Católica de Brasília
SGAN 916 - Módulo B - Asa Norte, CEP 70000 Brasília -DF
kathia@ucb.br

**Resumo**

A estimativa de tamanho é uma das métricas mais utilizadas na gestão de projetos de software, porque a partir dessa dimensão é possível definir o esforço, o prazo e os custos necessários para o desenvolvimento do software. Em projetos Orientados a Objetos (OO), a estimativa de tamanho é geralmente realizada através de duas métricas: Análise de Pontos de Função (APF) e Pontos de Casos de Uso (PCU). Na APF, a precisão da estimativa melhora à medida que se obtém mais informações da análise e do projeto do software. O PCU foi proposto para ser utilizado no início do ciclo de desenvolvimento, na fase de requisitos. Considerando-se que as duas métricas são adequadas em diferentes fases do processo de desenvolvimento de software, este trabalho se propõe a utilizar a APF e o PCU de forma combinada, para apoiar o gerente em relação à gestão de estimativa de tamanho de projeto de software. Este apoio consiste da definição de um processo de gestão de estimativa de tamanho, que deverá ser utilizado em paralelo aos processos de desenvolvimento e gestão de projetos de software.

**Palavras-chave:** Análise de Pontos de Função, Pontos de Casos de Uso, Gestão de estimativa de tamanho de software

**Abstract**

Size estimation is one of the most used metrics because it defines the effort, resources, and costs in software development. For Object Oriented (OO) software projects, the size estimate is usually obtained through two counting metrics: Function Points Analysis (FPA), and Use Cases Points (UCP). In FPA, the precision of the estimate improves as more information from the analysis and design is available. UCP was proposed to be used in the beginning of the development cycle, in the phase of requirement definition. Considering that these two metrics are useful in different phases of the software development process, this paper intends to indicate a combined use of APF and PCU to support the manager in the size estimate management of the software project. This support consists in the definition of a management process for size estimations that should be used along the development and management processes in software projects.

**Keywords:** Function Point Analysis, Use Case Points, Software size management.

## 1. Introdução

Para subsidiar o planejamento e a negociação de prazos, esforço, recursos e custos com o cliente, as empresas normalmente requerem estimativas de tamanho do software a ser desenvolvido no início do projeto. Em geral, a precisão de estimativas realizadas no início do projeto não é grande, necessitando ser refinada com base em informações disponibilizadas durante a implementação do projeto.

Assim, existe uma demanda por métricas que proporcionem maior precisão nas estimativas iniciais de tamanho de software a ser desenvolvido porque o tamanho tem impacto na solução técnica e na gestão do projeto, portanto, estimativas imprecisas podem resultar em fracassos [1] e [2].

A Análise de Pontos de Função (APF) é uma das métricas de estimativa de tamanho mais sedimentadas no mercado e que proporciona resultados cada vez mais precisos à medida que artefatos da fase de análise e projeto são gerados [3].

Por outro lado, Pontos de Casos de Uso (PCU) permite fazer estimativas no início do projeto com base no modelo de casos de uso, bastante utilizado no mercado, por ser mais completo que outros métodos de requisitos [4].

Portanto, para que a estimativa de tamanho de projetos Orientados a Objetos (OO) seja realizada com maior precisão desde o início do projeto, este trabalho propõe-se a utilizar essas métricas de forma combinada no momento em que elas são melhores aplicadas no processo de desenvolvimento, possibilitando assim, apoiar a gerencia de projetos.

Este apoio consiste da definição de um processo de gestão de estimativa que tem como objetivo indicar uma melhor forma de estimar, revisar e recalcular o tamanho do projeto à medida que novos artefatos estejam disponíveis durante o processo de desenvolvimento do projeto de software OO, provendo subsídios para melhorar as decisões gerenciais.

Nas seções seguintes apresentamos, uma breve descrição sobre gestão de projeto de software (seção 2), métricas de estimativa de tamanho (seção 3), a proposta de gestão de estimativa de tamanho (seção 4) as conclusões e perspectivas futuras (seção 5).

## 2. Gestão de projeto de software

O objetivo da gestão de projetos é o atendimento das demandas de clientes internos e externos, garantindo a satisfação de suas necessidades, mantendo um bom relacionamento e controlando os recursos, esforços e desempenho do projeto [17] [20].

De acordo com Pressman [18] a gestão de projetos envolve planejamento, acompanhamento e controle de processos, planos de projetos, pessoas e produtos, requerendo, segundo o PMBOK [19] a aplicação de conhecimentos, habilidades, ferramentas e técnicas para planejar e controlar atividades que visam atender as necessidades dos interessados.

O PMBOK [19] recomenda o uso dos seguintes processos para garantir a gestão eficaz de projetos: i) iniciação; ii) planejamento; iii) execução; iv) controle; e v) encerramento. Os processos de planejamento e de controle de projeto devem ser vistos como essenciais na gestão de projetos de software.

O objetivo do planejamento de projeto de software é "estabelecer planos razoáveis para desenvolver e gerenciar o projeto" [20]. O planejamento envolve o desenvolvimento de estimativas do projeto; o estabelecimento de compromissos; a definição do plano; a especificação de metas e objetivos; a elaboração de estratégias, políticas, procedimentos e regras a serem adotadas pelo projeto [20].

O principal produto do processo de planejamento é o plano do projeto. Com base no plano, o gerente inicia a execução e o controle do projeto até a geração do produto final e encerramento do projeto [17].

O objetivo do controle e rastreamento do projeto é "prover visibilidade adequada do progresso do projeto para que os gerentes possam providenciar ações corretivas quando o desempenho do mesmo desvia significativamente do plano elaborado" [20].

As decisões do gerente do projeto durante a fase de controle se baseiam em três tipos de análise que provêem informação de suporte a decisão: estimativas de tamanho, custos e

prazos; análise de viabilidade e análise de desempenho [21]. Essas análises são usualmente executadas iterativamente durante todo o processo de desenvolvimento do projeto.

Para facilitar o controle e a análise de desempenho do projeto durante o seu desenvolvimento, o PMBOK [19] propõe o uso de processos que gerenciam as principais dimensões do projeto como escopo; recursos; tempo, custos e riscos.

As gerências de tempo e custo dependem de estimativas iniciais para assegurar que o projeto termine dentro do prazo e do orçamento previstos. Sendo assim, no contexto deste trabalho, a gerência de tempo tem fundamental importância porque o prazo definido através do cronograma, com base nas estimativas iniciais, deverá ser gerenciado e ajustado à medida que novas estimativas de tamanho forem realizadas durante o desenvolvimento do projeto. Portanto, a medida de tamanho irá subsidiar a gerência de tempo dentro do contexto de gestão de projetos de software.

## 3. Métricas de estimativas de tamanho de software

"Estimativa de tamanho de software é um processo pelo qual uma pessoa ou um grupo de pessoas estima o tamanho de um produto de software" [5]. O tamanho do software é um indicador da quantidade de trabalho a ser executado no desenvolvimento de um projeto.

Atualmente, existem muitas métricas de estimativa de tamanho. A maioria é desenvolvida com base nas funções de software tais como: Análise de Pontos de Função; Bang; Feature Points; Boeing's ED Function Point; Mark II; Pontos de Casos de Uso e COSMIC Full Function Point [6].

A APF foi desenvolvida em 1979 por Allan Albrecht [2] e, posteriormente, refinada pelo *International Function Point Users Group* (IFPUG). Essa métrica permite uma contagem indicativa no início do desenvolvimento sem conhecer detalhes do modelo de dados. Posteriormente, na fase de projeto, essa contagem é revista com base na complexidade das funções e ao término do desenvolvimento do software, é realizada uma contagem detalhada, obtida a partir do grau de complexidade das funções levantadas no processo funcional, modelo de dados, descrição das telas e relatórios [7] e [2].

O processo de contagem da APF está bem estabelecido e é realizado em sete passos conforme mostra o Quadro 1.

**Quadro 1 - Passos do processo de contagem da APF**

| | |
|---|---|
| i) | Determinar o tipo de contagem. |
| ii) | Identificar o escopo de contagem e a fronteira da aplicação. |
| iii) | Contar as Funções de dados. |
| iv) | Contar as Funções transacionais. |
| v) | Determinar os PF não ajustados. |
| vi) | Determinar os valores dos fatores de ajustes. |
| vii) | Calcular os PF ajustados. |

São contados basicamente: i) as funções relacionadas aos dados utilizados pelo software e que englobam os arquivos lógicos internos (ALI) e os arquivos de interface externa (AIE); e ii) as funções transacionais que são as funções básicas que o software deve conter como entrada externa (EE), saída externa (SE) e consulta externa (CE). Depois são calculados os PF não ajustados com base no total de ALI, AIE, EE, SE e CE, determinado o fator de ajuste com base nas 14 características gerais do software (que indicam o grau de dificuldade para construir o software) e, finalmente, é aplicada uma fórmula para determinar os Pontos de Função (PF) ajustados.

Vários autores propuseram o mapeamento dos conceitos da APF para os conceitos OO ([8]; [7]; [9]; [6]; [10], [3] e [11]). Nesse tipo de software, os elementos básicos da contagem de funções de dados são as classes, que equivalem aos ALIs e AIEs, e os métodos que equivalem as funções transacionais, que podem ser uma EE, SE ou CE. A complexidade é atribuída de acordo com o número de itens de dados, que são os atributos da classe e dos registros lógicos, que são identificados através dos relacionamentos tipo associação, agregação e herança. Alguns autores apresentam regras para identificação de registros lógicos nos relacionamentos do tipo herança [8]; [11] e agregação [11] e [3]. Outros autores indicam regras para identificar as funções transacionais na descrição de casos de uso [11] e no diagrama de classes [9]; [10] e [6].

O Ponto de Casos de Uso (PCU) foi definido por Gustav Karner [12] para estimar projetos OO. O processo de contagem dessa métrica consiste de seis passos conforme mostra o Quadro 2. Nesse processo são contados os atores e os casos de uso. Com base nessa contagem são calculados os PCU não ajustados. Posteriormente, são determinadas as complexidades dos fatores técnicos (semelhante às características gerais da APF) e ambiental (que verifica as características da equipe e do ambiente em que será desenvolvido o software) e aplicada uma fórmula para calcular os PCUs ajustados.

**Quadro 2 - Passos do processo de contagem de PCU**

| | |
|---|---|
| i) | Contar os atores e atribuir o grau de complexidade. |
| ii) | Contar os casos de uso e atribuir o grau de complexidade. |
| iii) | Somar o total de atores com o total de casos de uso para obter o PCU não ajustado. |
| iv) | Determinar a complexidade do fator técnico. |
| v) | Determinar a complexidade do fator ambiental. |
| vi) | Calcular o PCU ajustado. |

O PCU é simples, rápido e fácil de usar [13]. Alguns dos autores que aplicaram o PCU propuseram regras para realizar a contagem de atores (como o uso de generalização) e casos de uso estendidos e incluídos ([14]; [15] e [16]).

Como a APF é a métrica mais usada no mercado e provê estimativas mais precisas à medida que se obtém mais informações do projeto, e o PCU foi criado para estimar projetos OO com base no modelo de casos de uso gerado no início do projeto, estas métricas foram selecionadas como objeto de estudo deste trabalho.

## 4. Gestão de estimativa de tamanho de projeto de software orientado a objetos

A gestão de estimativa de tamanho de projeto de software tem como objetivo indicar aos gerentes uma melhor forma de estimar, revisar e recalcular o tamanho do projeto à medida que novos artefatos estejam disponíveis durante o desenvolvimento do projeto de software OO, visando subsidiar as decisões gerenciais ao longo da gestão do projeto.

Para alcançar esse objetivo propomos um processo de gestão de estimativa de tamanho incorporando os seguintes elementos: i) uso combinado dos procedimentos de contagens de PCU e PF específicos para projetos OO; ii) uso de uma equação de relação entre a APF e PCU; e iii) uso de ações gerenciais para subsidiar o gerente na tomada de decisões [26].

Dessa forma, definimos um processo de gestão de estimativa relacionado com os processos de gestão e desenvolvimento de projetos de software OO de acordo com a NBR ISO/IEC 12.207 [22], a NBR ISO/IEC 10.006 [23] o PMBOK [19] a prática chave do CMM – nível 2 [20] e o RUP [24].

Este processo compõe-se das seguintes atividades:

i) **realizar a contagem de tamanho inicial em PCU**, na qual é realizada a primeira estimativa de tamanho do projeto em PCU;
ii) **projetar a estimativa em PF**, na qual é projetado o número de PF usando a estimativa em PCU e uma equação que descreve a relação entre PCU e APF;
iii) **realizar a contagem de tamanho em PF,** na qual é realizada a primeira contagem de PF;
iv) **comparar o número de PF projetado com o número de PF contado**, na qual são analisados os resultados projetados para PF com tamanho contado em PF;
v) **obter informações gerenciais do projeto**, na qual o gerente busca informações sobre o projeto em desenvolvimento: plano, cronograma e relatórios do status do projeto;
vi) **comparar as estimativas** com o cronograma planejado para identificar os desvios;
vii) **providenciar ações de ajuste no projeto** para corrigir os desvios do cronograma;
viii) **realizar contagem final de PF** para identificar o tamanho real do produto de software.

Este processo deve ser executado paralelamente aos processos de desenvolvimento e de gerência de projetos de software OO conforme mostra a Figura 1. Esta figura é dividida em três partes. A primeira parte apresenta as principais atividades e artefatos gerados em um processo de desenvolvimento de software [24] e utilizados no processo de gestão de estimativa de tamanho. A segunda parte apresenta o processo de gestão de estimativas de tamanho, suas atividades e os elementos necessários à execução deste processo. A terceira parte mostra o processo de gestão de projetos de software [19].

O processo de gestão de estimativa de tamanho mostra através da Figura 1, que a estimativa inicial do projeto é realizada em PCU com base no modelo e descrição de casos de uso definido na disciplina de requisitos do processo de desenvolvimento (atividade 1). Com base no PCU e na equação de relação entre APF e PCU é projetado PF (atividade 2).

Através do PCU encontrado e do PF projetado, o gerente do projeto desempenha as atividades do processo de planejamento que faz parte dos processos de gestão de projetos: i) estimar o esforço e custo do projeto; ii) elaborar o plano e o cronograma do projeto; e iii) negociar compromissos com o cliente de acordo com os recursos necessários.

A projeção de PF é gerenciada e re-estimada durante todo processo de desenvolvimento do projeto, à medida que novos artefatos forem desenvolvidos. Assim que, os diagramas de seqüência e de classes de análise e projeto forem gerados, é realizada a primeira contagem de PF (atividade 3). Em seguida, a estimativa de PF projetada é comparada com o número de PF contado (atividade 4). Nesse ponto do projeto, o gerente compara os resultados das estimativas com as informações gerenciais do projeto, obtidas através dos processos de planejamento e controle do projeto e verifica se o cronograma é possível de se realizar ou impraticável, se está dentro do prazo ou se está atrasado (atividades 5 e 6).

Deve ser verificado a completitude das atividades realizadas, o trabalho concluído, o esforço e os recursos gastos com o planejamento inicial. Com base nos resultados dessa comparação, o gerente providencia ações de ajustes no projeto (atividade 7). Em geral, as ações incluem revisões no plano de desenvolvimento do projeto, controle do desempenho da equipe e comunicação das decisões para as pessoas envolvidas. As mudanças no tamanho do projeto podem afetar as negociações realizadas anteriormente, tornando necessário novas negociações com o cliente.

O cronograma e o plano do projeto é rastreado pelo gerente durante os processos de planejamento, controle e execução do projeto, conforme proposto pelo PMBOK [19], CMM – nível 2 [20] e de acordo com as necessidades do projeto e com as diretrizes da organização.

Finalmente, após a implementação e testes do produto e com base nos subsistemas e produto final é realizada a última contagem de PF (atividade 8) para subsidiar os acertos finais com o cliente, antes de disponibilizar o produto e encerrar o processo de gestão do projeto.

A contagem inicial em PCU, a projeção de PF e todas as contagens de PF realizadas durante o desenvolvimento do projeto, devem ser armazenadas em repositórios de cada organização. Com base na contagem de diversos projetos é possível refinar a equação de relação entre PCU e APF e projetar PF com maior precisão para a organização.

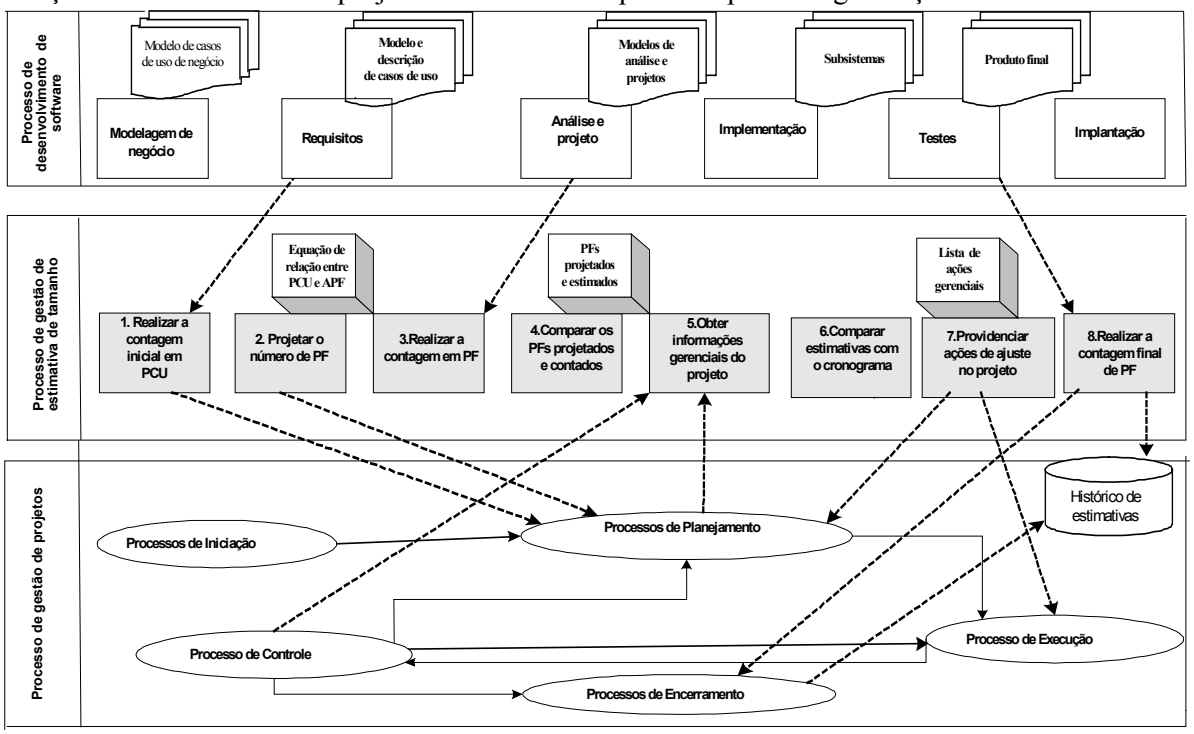


**Figura 1. Relacionamento do processo de gestão de estimativa de tamanho com os processos de desenvolvimento e gestão de projetos de software.**

Para possibilitar a gestão de estimativa de tamanho proposta é essencial, portanto, definir uma relação entre a APF e PCU, de forma a permitir o seu uso combinado. Além disso, seria importante ter um conjunto de possíveis ações gerenciais que subsidie as decisões do gerente durante a gestão do projeto.

Nas seções seguintes, apresentamos como foi encontrada a relação entre a APF e PCU (seção 4.1) e como foi identificado o conjunto de ações gerencias (seção 4.2).

### 4.1. Investigando a relação entre a APF e PCU

Os procedimentos de contagem da APF e PCU para projetos OO devem ser utilizados em diversos projetos de software de uma mesma organização para criar uma base de dados histórica de estimativas para investigar a relação entre essas duas métricas.

Como não foi encontrada uma base de dados histórica de estimativas de projetos OO em PCU e APF, decidimos aplicar essas métricas em projetos OO da academia e da indústria visando obter valores reais de PF e PCU para subsidiar a investigação da existência de relação entre essas métricas.

Foram contados ao todo dezenove projetos de software OO através da análise da documentação dos projetos disponibilizados pelos gerentes de forma impressa e eletrônica. As

características gerais do sistema foram identificadas através de um questionário enviado por meio eletrônico ou aplicado através de entrevista estruturada com os gerentes de projeto.

A APF e PCU foram aplicadas em 9 projetos da academia desenvolvidos por alunos de final de curso de Ciência da Computação de uma universidade, sendo que, 8 foram desenvolvidos como projeto final de conclusão do curso e 1 foi desenvolvido como um projeto de pesquisa. Todos os projetos seguiram o mesmo processo de desenvolvimento de software; a *Unified Modeling Language* (UML) e um formulário padrão para descrição dos casos de uso.

As informações sobre os projetos foram coletadas a partir do documento de contexto, especificação de requisitos, diagrama de casos de uso, diagrama de classes de negócio, descrição de casos de uso e protótipo de telas, diagramas de seqüência, diagramas de classes de análise e diagramas de classes de projetos e Modelo de Entidade e Relacionamento (MER).

Na indústria, as métricas foram aplicadas em três empresas (duas privadas e uma pública). As empresas privadas têm como principal negócio o desenvolvimento de software para terceiros. No contexto deste trabalho, essas empresas foram denominadas como Empresa A, B e C. Ao todo foram contados 10 projetos (em desenvolvimento) da indústria, sendo cinco da Empresa A, dois da Empresa B e três da Empresa C.

Todas as três empresas desenvolvem software orientados a objetos com base no *Rational Unified Process* (RUP) e na (UML) e utilizam as ferramentas *Rational Rose, Soda,* e as linguagens C, *Java* e *Enterprise Java Beans* (EJB).

A Empresa A é atuante no mercado do setor financeiro e no setor de saúde brasileiro. A Empresa B é uma autarquia do setor público e tem como principal negócio à coordenação e distribuição de recursos financeiros da área de educação. Essas empresas possuem uma equipe de aproximadamente cem profissionais. A Empresa C atua na área de planejamento da informação, treinamento e desenvolvimento de sistema. Essa empresa possui certificação ISO 9001 e uma equipe de mais de mil profissionais.

Todas as equipes de projetos de desenvolvimento de software das Empresas A, B e C seguem documentos padrões para definição e documentação do software tais como: documento de visão do projeto, especificações suplementares, detalhamento de casos de uso e descrição de entidades persistentes no banco de dados.

Para a contagem de PCU utilizamos os modelos e descrição de casos de uso e para a contagem de PF utilizamos os modelos de análise, de projeto e o esquema do banco de dados ou Modelo de Entidade-Relacionamento (MER).

#### 4.1.1. Análise dos resultados

Os valores de PCU e PF não ajustados e ajustados são apresentados na Tabela 1 (representados pelas letras 'A' de academia e 'I' de indústria, seguidos do número seqüencial).

O número de PF foi superior ao número de PCU em todos os projetos. Esta diferença pode ser atribuída à contagem de funções de dados que incluem os ALIs e AIEs definidos ou consultados pelo sistema.

O projeto "A9" era um projeto de pesquisa, que foi posteriormente implantando na Universidade e devido a isso, tornou-se maior e mais complexo que os outros projetos da academia. Mas, para fins do estudo da relação foi considerado um projeto acadêmico por seguir os mesmos padrões definidos para a academia. Observou-se, ainda, que este projeto teve 57 casos de uso de baixa complexidade, com uma a duas transações cada um. Em projetos como esse, a contagem de Casos de uso pode superestimar o tamanho do software, já que cada Caso de uso vale no mínimo 5 PCUs quando se tem até 3 transações. Portanto, é de

fundamental importância que a equipe tenha claro entendimento de como definir e descrever os casos de uso e suas respectivas transações.

Os resultados da aplicação das métricas nos projetos desenvolvidos pela indústria seguiram a mesma tendência dos resultados apresentados para a academia, diferenciando apenas no tamanho e na complexidade dos projetos, principalmente em relação às funções de dados, ou seja, são definidos muitos ALIs e, em alguns casos, identificados a existência de AIEs (projetos I1 a I5 e I9) (Tabela 1).

**Tabela 1 – Resultados da contagem de PCU e PF na academia e indústria**

| ACADEMIA | | | | | INDÚSTRIA | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | PCU | | APF | | | PCU | | APF | |
| PROJETOS | PCU - NA | PCU-A | PF-NA | PF-A | PROJETOS | PCU - NA | PCU- A | PF-NA | PF - A |
| A1 | 62 | 52,28 | 120 | 102,00 | I1 | 60 | 40,95 | 135 | 159,30 |
| A2 | 176 | 174,74 | 230 | 220,80 | I2 | 158 | 153,85 | 365 | 416,10 |
| A3 | 200 | 178,89 | 217 | 232,19 | I3 | 232 | 231,98 | 652 | 652,00 |
| A4 | 125 | 97,34 | 160 | 148,8 | I4 | 258 | 177,26 | 362 | 372,86 |
| A5 | 76 | 58,17 | 96 | 91,02 | I5 | 103 | 86,34 | 183 | 194,47 |
| A6 | 139 | 112,36 | 159 | 162,18 | I6 | 63 | 34,26 | 109 | 89,38 |
| A7 | 131 | 96,62 | 267 | 240,3 | I7 | 216 | 156,05 | 494 | 503,88 |
| A8 | 36 | 27,55 | 38 | 36,01 | I8 | 116 | 85,41 | 325 | 344,50 |
| A9 | 331 | 296,86 | 492 | 432,96 | I9 | 303 | 223,1 | 634 | 672,04 |
| | | | | | I10 | 169 | 124,43 | 447 | 473,82 |

Os projetos "I6 e I7" da Empresa B e os projetos "I8, I9 e I10" da Empresa C também tiveram a contagem de PF bem maior que PCU devido ao alto número de ALIs definidos nos projetos, apesar da contagem de PF dos projetos da Empresa B serem baseadas na descrição de casos de uso e nos diagramas de classes e MER já que os diagramas de seqüência não são adotados nessa empresa.

Tentou-se identificar as funções transacionais através dos métodos definidos no diagrama de classes para os projetos que não foram definidos diagramas de seqüência (Empresa C). Observamos que esta não é uma tarefa simples, pois não há indicação clara de quais classes são responsáveis pela realização de cada caso de uso se essas classes não forem descritas explicitamente na descrição de casos de uso ou no diagrama de seqüência. Além disso, não há garantia que todos os métodos são descritos nesse diagrama e se todos os métodos indicados estão relacionados à implementação do negócio. Este tipo de contagem pode levar a valores sub ou super estimados das funções transacionais. Portanto, deve-se analisar esta forma de contagem com mais cuidado.

### 4.1.2. Equação de relação entre PF e PCU

Com o intuito de se utilizar PCU e APF de forma combinada foi investigado a existência de relação entre essas duas métricas com base nos dados das contagens dos projetos da academia e indústria (Tabela 1) para definir uma equação que representasse esta relação através da

análise estatística. O desenvolvimento de uma equação de relação permitirá a projeção de PF a partir de PCU obtidos na fase inicial dos projetos.

No entanto, como os projetos foram desenvolvidos em ambientes diferentes, ou seja, na academia e em três empresas distintas, levantou-se a questão se os resultados das contagens poderiam ser agrupados ou não. Diante desta questão, decidimos fazer a análise estatística dos dados das contagens de PCU e PF nos nove projetos da academia e nos dez projetos da indústria separadamente de acordo com as seguintes hipóteses de estudo: i) **Hipótese de estudo 1 –** Existe uma relação entre PCU e APF; e ii) **Hipótese de estudo 2 –** Não existe diferença entre as equações que descrevem a relação entre PCU e APF nos projetos da academia e da indústria.

A hipótese de estudo 1 é a principal suposição deste trabalho, pois a existência de uma relação entre PCU e APF é uma condição essencial para que estas duas métricas sejam utilizadas de forma combinada no processo de gestão de estimativa. Neste caso, usando o pacote estatístico SAS, foram testadas equações linear ($y = b_0 + b_1x$) e quadrática ($y = b_0 + b_1x + b_2x^2$ ) [27] como descritoras da relação entre PCU e APF. O termo *"y"* indica a variável dependente, no caso APF e o termo *"x"* representa a variável independente, no caso, PCU. Nestas equações, $b_0$, $b_1$ *e* $b_2$ representam respectivamente, o intercepto, o coeficiente de efeito linear e o coeficiente de efeito quadrático. De acordo com os resultados identificamos que uma equação quadrática não é adequada para descrever a relação entre APF e PCU na academia e indústria e que equações lineares são as mais adequadas para descrever esta relação devido aos valores significativos ($p < 0,05$) obtidos para o coeficiente linear (Tabela 2). No entanto, foi verificado que o intercepto não apresentou significância estatística indicando que zero era um valor possível para esse parâmetro da equação linear.

**Tabela 2- Valores dos níveis de significância (*p*) obtidos pelo "teste *t*" nas equações de regressão quadrática e linear e linear sem intercepto**

| Modelos | Teste *t* – níveis de significância (p) obtidos nas equações de regressão quadrática e linear | | |
|---|---|---|---|
| | Intercepto $b_0$ | Coeficiente linear $b_1$ | Coeficiente quadrático $b_1^2$ |
| **Quadrática** | | | |
| indústria | 0,849 | 0,1373 | 0,8297 |
| academia | 0,514 | 0,139 | 0,8600 |
| **Linear** | | | |
| indústria | 0,482 | **0,0001** | - |
| academia | 0,340 | **0,0001** | - |

| Modelo | Teste *t* – níveis de significância (p) obtidos na equação de regressão linear sem intercepto | | |
|---|---|---|---|
| | Intercepto $b_0$ | Coeficiente linear $b_1$ | Coeficiente quadrático $b_1^2$ |
| **Linear sem intercepto** | | | |
| indústria | - | **<0,0001** | - |
| academia | - | **<0,0001** | - |

Assim foi investigada a possibilidade de equações lineares sem intercepto para descrever a relação entre PCU e APF. Os baixos valores de *p* calculados para o coeficiente linear (Tabela 2) indicam adequação das equações lineares sem intercepto para descrever a relação entre APF e PCU. De fato, ao considerar que PCU e PF são variáveis quantitativas de medida de tamanho de software, sempre que PCU for zero, obrigatoriamente, PF também será zero. Assim, as equações lineares sem intercepto e que descrevem a relação entre PCU e PF nos projetos avaliados são as seguintes: i) projetos da academia: PF = 1,4615 PCU; ii) projetos da indústria: PF = 2,8905 PCU.

A hipótese de estudo 2, "Não existe diferença entre as equações que descrevem a relação entre PCU e PF dos projetos da academia e da indústria" foi testada através da comparação entre as equações obtidas para a indústria e para a academia conforme metodologia de NETER e WASSERMAN [27]. A aceitação de uma ou outra hipótese é realizada pela comparação do valor de *F* calculado pela formula "*F = (SQE © – SQE ©/2) /*

*(SQE © /$n_1$+ $n_2$ – 4)"* com o valor de *F* encontrado em tabelas estatísticas. O valor de *F* da tabela é encontrado, considerando-se um nível de significância e os graus de liberdade do numerador (2) e do denominador ($n_1$+ $n_2$ – 4) da formula acima. Aplicando esta formula temos um *F* calculado com valor de 24,07, superior ao valor de *F* da tabela que é de 4,54. Assim, como o valor de *F* calculado é maior que o valor de F da tabela concluímos que existe diferença entre as equações definidas para representar a relação entre PCU e PF na indústria e na academia.

A Figura 2 mostra a representação gráfica das equações encontradas para academia e indústria. A reta indica os valores estimados a partir da equação encontrada para a academia (PF=1,4615 PCU) e para a indústria (PF=2,8905 PCU). O nível de significância *p* indica a possibilidade de erro ao aceitar-se que existe uma relação entre as variáveis PCU e PF e que essa relação é descrita por uma equação linear. No caso, o valor de *p* é menor que 0,0001. Outra estatística apresentada nesta Figura é o coeficiente de determinação $R^2$. Os valores de $R^2$ inicialmente encontrados para a academia ($R^2$ = 0,9697) e para a indústria ($R^2$ = 0,9723), foram arredondados para 0,97. Devido a isso, os valores de $R^2$ foram iguais para os dois casos. Esse coeficiente indica quanto da variabilidade existente em PF é explicado com a utilização da variável PCU. Um valor alto como o encontrado para $R^2$ ($R^2$ = 0,97) e o baixo nível de significância *p* indicam um excelente ajuste das equações definidas.

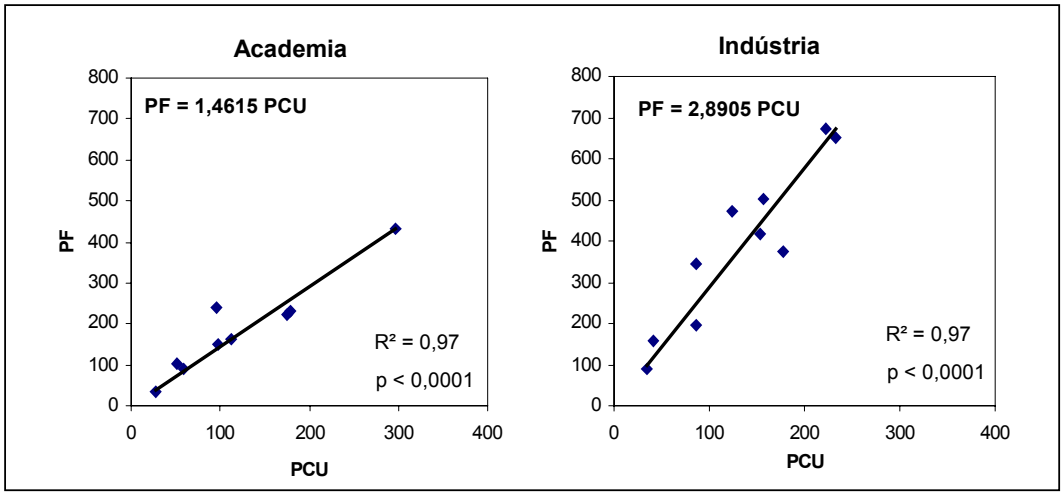


**Figura 2 – Relação entre PCU e APF em projetos da academia e indústria**

As diferenças entre as equações da academia e indústria podem ser atribuídas ao nível de complexidade dos projetos e ao número de casos de uso definidos. Os projetos de conclusão de curso desenvolvidos na academia não utilizam, na maioria das vezes, AIEs e possuem um número relativamente baixo de ALIs, o que implica em uma pontuação menor de PF. Por outro lado, são definidos muitos casos de uso de baixa complexidade (com apenas 1 ou 2 transações como no projeto "A9"). Isto contribuiu para que a contagem de PCU ficasse superior e, conseqüentemente diminuísse a diferença entre o total de PCU e PF.

Já os projetos da indústria, além de serem mais complexos em relação ao número de funções de dados e de funções transacionais, são melhores definidos, ou seja, os números de casos de uso representam as funções do software de forma mais adequada.

É importante ressaltar que estas equações serviram para mostrar que existe uma relação entre APF e PCU e que essas métricas podem ser utilizadas de forma combinada.

Para obter maior precisão na equação a ser utilizada para projetar PF, é importante que cada empresa encontre a sua equação de relação entre a APF e PCU através das contagens de

seus próprios projetos, desenvolvidos com base em processos, padrões e tecnologias específicas ao ambiente de desenvolvimento de software da empresa.

### 4.2. Identificando as ações gerenciais

A partir da gestão de estimativa de tamanho associada à gerência de projetos, o gerente terá informações mais precisas sobre o projeto e isso possibilitará o rastreamento do cronograma, o controle das atividades planejadas e realizadas e os ajustes no plano.

No entanto, para facilitar a tomada de decisão em relação aos ajustes no plano e no cronograma, foi necessário identificar as ações mais relevantes que poderão subsidiar os gerentes de projeto na execução do processo de gestão de estimativa descrito na seção 4.

Para identificar essas ações, foi realizada uma pesquisa de campo entre gerentes de projetos de software da academia e indústria.

O objetivo do estudo foi identificar um conjunto de ações gerenciais sob o ponto de vista dos gerentes de projeto no contexto de gestão de estimativa de tamanho de projetos de software. Considerando este objetivo e o contexto da gestão de estimativa que é permitir ao gerente de projeto uma forma de estimar, revisar e recalcular o tamanho do projeto para que possa realizar uma melhor gestão do prazo de desenvolvimento do projeto (considerando a medida de tamanho), foi analisado então quais os momentos em que o gerente de projeto precisa providenciar ações gerenciais relacionadas ao prazo de desenvolvimento.

Um primeiro momento é logo no início do projeto quando o gerente faz uma primeira estimativa para definir prazos para o cliente. Com a estimativa de tamanho inicial, o gerente pode perceber que o prazo que o cliente deseja receber o produto é impraticável diante do tamanho do projeto a ser desenvolvido. Dessa forma, para alcançar o objetivo deste estudo é necessário identificar **(1) quais ações devem ser tomadas quando o cronograma do projeto é impraticável**. Assim, através da pesquisa de campo, foi solicitado aos gerentes de projeto à indicação das ações "mais recomendadas", "recomendadas", "pouco recomendadas" ou "não recomendadas" para essa situação.

Uma vez definido um prazo com o cliente, é necessário que o gerente acompanhe o cronograma durante todo o processo de desenvolvimento do software, buscando mantê-lo dentro do prazo estabelecido. Dessa forma, para alcançar o objetivo deste estudo, é necessário identificar **(2) quais ações devem ser tomadas para manter o cronograma do projeto dentro do prazo**. Na pesquisa de campo, foi solicitado que os gerentes de projeto identificassem a periodicidade em que essas ações devem ser tomadas na prática, ou seja, no "início do desenvolvimento do projeto", "semanalmente", "mensalmente", "após o final de cada fase de desenvolvimento", as que são tomadas "esporadicamente", "sem periodicidade definida" e as "não praticadas pelos gerentes".

No entanto, à medida que o gerente de projeto acompanha continuamente a execução do projeto, mesmo tomando ações para mantê-lo no prazo, pode ocorrer atrasos no cronograma por diferentes fatores. Assim, decidimos identificar **(3) quais ações devem ser tomadas para cada fato identificado como causador do atraso no cronograma do projeto**, de forma a melhor alcançar o objetivo deste estudo. Os fatos que provocam atraso no cronograma foram identificados no estudo realizado por Farias [25] e as possíveis ações a serem tomadas pelo gerente foram identificadas na literatura ([17]; [18]; [19]; [20] e [21].

A coleta de dados foi realizada através da aplicação de um questionário [26] entre os gerentes de projetos de software da indústria (empresas públicas e privadas) e alunos do curso de um mestrado na área, cujo perfil são de profissionais que trabalham na indústria na área de TI. Do total de 48 questionários distribuídos, obteve-se 37 respostas (36 da indústria e 1 da Academia).

A primeira parte do questionário identificou as características dos profissionais que participaram da pesquisa como: a formação profissional, área de atuação e experiência na gestão de projetos. Em relação à formação profissional, identificamos que 23 dos respondentes possuem especialização, 18 possuem mestrado, 3 possuem doutorado e 8 possuem certificações. Quanto a experiência em gestão de projetos de software, 13 gerentes possuem de 5 a 10 anos, e 12 de a 1 a três anos, sendo que a maioria dos respondentes gerenciou de 1 a 5 projetos de software.

Os resultados da pesquisa foram analisados de acordo com as questões definidas.As respostas para a **questão 1: "Quais ações devem ser tomadas quando o cronograma do projeto é impraticável?"** foram sintetizadas por ordem decrescente de indicação das ações pelos respondentes do questionário conforme mostra o Quadro 3.

**Quadro 3 – Ações a serem tomadas quando o cronograma é impraticável**

| **Questão 1: Quais ações devem ser tomadas quando o cronograma do projeto é impraticável?** | |
|---|---|
| **Ações mais recomendadas** | ▪ **Ação 1.** Reunir com o cliente, apresentar a estimativa detalhada em relação ao esforço e a duração estimada para o projeto e negociar novo prazo |
| **Ações recomendadas** | ▪ **Ação2.** Definir uma estratégia de desenvolvimento incremental que entregue a funcionalidade crítica na data prevista e adie as funções restantes;<br>▪ **Ação 4.** Contratar consultoria especializada para resolver problemas técnicos;<br>▪ **Ação 5.** Substituir os técnicos que não tem apresentado resultados satisfatórios. |
| **Ações pouco recomendadas** | ▪ **Ação 3.** Negociar o aumento do orçamento e contratar mais recursos humanos. |
| **Ações não recomendadas** | ▪ **Ação 6.** Ignorar a realidade e esperar completar o prazo determinado no cronograma – 100% respondentes |
| **Outras ações indicadas** | ▪ Medir a satisfação do cliente, Motivar a equipe e Oferecer treinamento |

A **questão 2: "Quais ações devem ser tomadas para manter o cronograma do projeto dentro do prazo"** visa identificar as ações que permitem controlar ou manter o cronograma dentro do prazo. Identificamos que a maioria das ações recomendadas para manter o cronograma no prazo, deve ser tomadas no início do desenvolvimento, semanalmente ou sem periodicidade definida conforme mostra o Quadro 4.

**Quadro 4- Ações a serem tomadas quando o cronograma está dentro do prazo**

| **Questão 2: Quais ações devem ser tomadas para manter o cronograma do projeto dentro do prazo?** | |
|---|---|
| **No início do desenvolvimento do projeto** | ▪ **Ação 10.** Identificar as atividades que não são do projeto ou que dependem de fornecedores para serem executadas;<br>▪ **Ação 3.** Incluir reservas orçamentárias para resolver problemas ou executar o plano de contingência;<br>▪ **Ação 5.** Verificar se todas as atividades e tarefas que visam alcançar o objetivo do projeto constam no cronograma e foram entendidas pela equipe;<br>▪ **Ação 18.** Identificar se há algum membro da equipe que possui habilidades únicas, difíceis de serem substituídas ou está comprometido com outros projetos;<br>▪ **Ação 2.** Avaliar o tempo necessário para implementar cada atividade;<br>▪ **Ação 6.** Identificar se os marcos de referência do projeto foram estabelecidos de modo que o progresso possa ser acompanhado;<br>▪ **Ação 8.** Verificar se as interdependências entre as atividades foram adequadamente indicadas;<br>▪ **Ação 4.** Identificar e documentar as restrições no plano do projeto que limitarão a realização das atividades e elaborar um plano de contingência;<br>▪ **Ação 7.** Verificar se todo o esforço e duração do projeto foram atribuídos corretamente a cada atividade do cronograma. |

**Quadro 4- Ações a serem tomadas quando o cronograma está dentro do prazo (cont.)**

| Questão 2: Quais ações devem ser tomadas para manter o cronograma do projeto dentro do prazo? | |
|---|---|
| **Semanalmente** | ▪ **Ação 14.** Conduzir reuniões periódicas para que cada membro do projeto possa relatar o progresso e os problemas atuais;<br>▪ **Ação 12.** Identificar e coordenar as atividades e tarefas realizadas em paralelo;<br>▪ **Ação 19.** Controlar a produtividade da equipe<br>▪ **Ação 13.** Identificar as datas do cronograma que foram alcançadas e as que não foram;<br>▪ **Ação 9.** Identificar questões que podem causar problemas futuros, providenciar ações corretivas e alertar a equipe do projeto;<br>▪ **Ação 16.** Atualizar o plano, os documentos técnicos e informar a equipe e clientes;<br>▪ **Ação 17.** Ter membros representativos nas reuniões de revisões técnicas para garantir o entendimento comum das necessidades do cliente e evitar futuras surpresas. |
| **Mensalmente** | ▪ **Ação 15.** Analisar os relatórios de desempenho do projeto e avaliar os resultados das revisões conduzidas ao longo do processo de desenvolvimento. |
| **Após o final de cada fase de desenvolvimento** | ▪ **Ação 20.** Refinar as estimativas realizadas no início do projeto durante todo o ciclo de desenvolvimento do projeto. Esta indicação coincide com a proposta do processo de gestão de estimativa, definido neste trabalho (ver Figura 1). |
| **Sem periodicidade definida** | ▪ **Ação 21**. Contratar consultoria externa para auxiliar no uso da tecnologia;<br>▪ **Ação 1.** Avaliar o plano de projeto, o ambiente de desenvolvimento, as datas impostas pelo cliente e os fatores externos que podem interferir no cronograma; **Ação 22.** Facilitar a comunicação entre os membros da equipe;<br>▪ **Ação 11.** Identificar possíveis mudanças que influenciam no cronograma (ex: na legislação, no escopo ou tecnologia);<br>▪ **Ação 16.** Atualizar o plano, os documentos técnicos e informar a equipe e clientes;<br>▪ **Ação 17.** Ter membros representativos nas reuniões de revisões técnicas para garantir o entendimento comum das necessidades do cliente e evitar futuras surpresas. |
| **Não praticadas** | ▪ **Ação 6.** Ignorar a realidade e esperar completar o prazo determinado no cronograma. |

A **questão 3: "Quais ações devem ser tomadas para cada fato identificado como causador do atraso no cronograma do projeto?"**identificou as ações a serem tomadas pelo gerente quando o cronograma está atrasado de acordo com os fatos que provocam atraso no cronograma. As ações mais indicadas pelos respondentes para esta questão (independente do fato causador do atraso) são as apresentadas no Quadro 5 em ordem de decrescente de indicação.

**Quadro 5 - Ações a serem tomadas pelos gerentes quando o cronograma está atrasado**

| Questão 3: Ações a serem tomadas pelos gerentes quando o cronograma está atrasado |
|---|
| ▪ **Ação 1.** Rever o planejamento do projeto e identificar desvios no cronograma;<br>▪ **Ação 8.** Fazer ajustes no plano e documentar as ações para contemplar as revisões de tempo, custo, seqüência de atividades e análise de alternativas de respostas a riscos;<br>▪ **Ação 6.** Rever o cronograma e negociar com as partes envolvidas, caso não haja possibilidade de recuperação do atraso;<br>▪ **Ação 10.** Registrar os desvios do cronograma, identificar quais atividades ocorreram e o motivo de sua ocorrência;<br>▪ **Ação 9.** Verificar se as atualizações do cronograma requerem ajustes em outros aspectos do plano geral do projeto;<br>▪ **Ação 2.** Agregar novas pessoas na equipe do projeto;<br>▪ **Ação 11.** Contratar consultoria especializada;<br>▪ **Ação 7.** Controlar a comunicação entre os membros do projeto;<br>▪ **Ação 5.** Rever o cronograma e buscar alcançar o próximo marco do projeto na data prevista para garantir o término do projeto;<br>▪ **Ação 4.** Rever as funções do software, priorizá-las e deixar as menos importantes para futuras revisões;<br>▪ **Ação 3.** Redistribuir as pessoas na equipe. |

## 5. Conclusão

No mercado competitivo atual, uma preocupação constante das organizações é o cumprimento de cronogramas e o custo efetivo de seus projetos de desenvolvimento de software. Todas estas características dependem de um gerenciamento adequado e de um plano efetivo do projeto baseado em estimativas mais precisas de tamanho.

Diante desta necessidade, este trabalho propôs um processo de gestão de estimativa de tamanho que acompanha a evolução do projeto do início ao fim porque o tamanho do projeto influencia na estimativa de custos e prazos do projeto. Nessa proposta foi identificada uma equação de relação entre as métricas de análise por pontos de função e análise de casos de uso. Foi identificada ainda, uma lista de ações gerenciais mais recomendadas para controlar o cronograma.

A equação que descreve a relação entre PCU e APF na indústria (PF = 2,8905 PCU) já foi utilizada em uma empresa de desenvolvimento de software para verificar a diferença entre a estimativa obtida através de PCU e APF em um projeto iterativo [28]. Nesse trabalho, quando dividido o total de PF por PCU, o resultado foi 2,8911, o qual é muito próximo a 2,8905, valor encontrado na nossa pesquisa através da análise de regressão.

Como continuidade deste trabalho, sugerimos implantar o processo de gestão de estimativa de tamanho de projetos em uma das empresas que participaram desta pesquisa para verificar a adequação da equação de relação encontrada entre a APF e CPU na projeção de PF de novos projetos e definir uma ferramenta para apoiar o processo.

## Referências Bibliográficas


[1] ROSS, M. Size does matter: continuous size estimating and tracking. **Quantitative Software Management.** S.d. 16 p.

[2] LONGSTREET, D. **Fundamentals of Function Point Analysis**. Blue Springs: Longstreet Consulting Inc., 2002.

[3] CALDIERA, G et al. Definition and experimental evaluation of Function Points for object-oriented systems. In: International Symposium on Software Metrics , 5. **Proceedings.** IEEE. March 20 – 21, 1998, Bethesda, Maryland. P. 167 – 179.

[4] DEKKERS, C. Uses Cases and Function Points: Where´s the Fit? In: **Metrics Strategies**. Jan.1999.6 p

[5] McPHEE, C. **SENG 621**: Software process management: software size estimation. University of Calgary. 1999. 11p.

[6] GARMUS, D., HERRON, D. Function Point Analysis: Measurement Practices for successful software projects. Addison-Wesley: EUA. 2000. 363 p.

[7] IFPUG. Function Point Counting Practices Manual: Release 4.1. Ohio: IFPUG. 2000. 1 v.

[8] _____. **Function Point Counting Practices:** Case Study 3:Analysis, Construction. Release 2.0. Princeton Junction: IFPUG. 2001. 246 p

[9] RAM, J.; RAJU, S. Object oriented design function points. In: Asia-Pacific Conference on Quality Software, 1. **Proceedings.** IEEE, 2000. 6 p.

[10] UEMURA, T.; KUSUMOTO, S.; INOUE, K. Function point measurement tool for UML design specification. In: International Symposium on Software Metrics , 6. **Anais.** IEEE, 1999, p. 62 –69.

[11] FETCKE, T.; ABRAN, A; NGUYEN, T. **Mapping the OO-Jacobson approach into Function Point Analysis.** 1997. 11 p.

[12] KARNER, G. **Use Case Points**: resource estimation for Objectory projects. Objective Systems SF AB (copyright owned by Rational/IBM), 1993.

[13] DAMODARAN, M; WASHINGTON, A. Estimation using use case points. **Computer Science Program.** Texas –Victoria: University of Houston. S.d. 4 p.

[14] SCHNEIDER, G.; WINTERS, J. Use case and project plan. In: ____. **Appying use cases:** a practical guide. 2. ed. New York: Addison Weslwy, 2001. capítulo 10, p. 143 – 159.

[15] RIBU, K. **Estimating object-oriented software projects with use cases**. Oslo: **University of Oslo, 2001.132 p. Tese (Mestrado– Department of Informatics).**

[16] ANDA B. et al. Estimating software development effort based on use cases: experiences from industry. In: International Conference on the Unified Modeling Language (UML2001), 4. **Proceedings.** Toronto, Oct. 1 – 5, 2001, p. 487 – 502.

[17] JURISON, J. Software project management: the manager's view. **Communications of the association for Information Systems.** V. 2, article 17, Sep. 1999. 56 p.

[18] PRESSMAN, R. **Engenharia de Software.** Rio de Janeiro: McGraw-Hill, 2002. 843 p.

[19] PMBOK. **Project management body of knowledge.** Belo Horizonte: PMIMG, 2000.

[20] PAULK, M. et al. **The capability maturity model**: guidelines for improving the software process. Boston: Addison Wesley. 2000. 441p. Carnegie Mellon University: SEI.

[21] McGARRY, J. et al. **Practical software measurement:** objective information for decision makers. Boston: Addison-Wesley. 2001. 277p.

[22] ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS (ABNT). **NBR ISO/IEC 12.207**: Tecnologia de Informação:processos do ciclo de vida de software.Rio de Janeiro : ABNT.1997. 38

[23] _____. **NBR ISO/IEC 10.006**: Gestão da qualidade: diretrizes para a qualidade de gerenciamento de projetos. Rio de Janeiro: ABNT. 2000. 17 p.

[24] RATIONAL UNIFIED PROCESS. **Rational Unified Process*:*** best practices for software development teams**.** 2001.

[25] FARIAS, L. D. **Planejamento de riscos em ambientes de desenvolvimento de software orientados a organização**. Rio de Janeiro: UFRJ/COPPE. 2002. Tese (Mestrado).

[26] ANDRADE, E. L. P. **Pontos de Casos de Uso e Pontos de Função na gestão de estimativas de tamanho de projetos de software orientados a objetos**. Brasília: UCB. 2004. Tese (Mestrado).

[27] NETER, J; WASSERMAN, W. **Applied linear statistical models.** Homewood, Illinois: Richard D. Irvin, 1974.

[28] D'OLIVEIRA, F.M.; SOUZA, S. V. **Utilizando PCU e PF para estimar projetos de software em um processo de desenvolvimento iterativo.** UCB; POLITEC. 2004. 6 p.