

COCOMO II - Um modelo para estimativa de custos de Gerência de Projetos

Pablo Ariel do Prado López

UNISINOS

pablo.ap.lopez@gmail.com

Resumo. Este artigo apresenta como o estabelecimento de estimativas se constitui em uma das principais atividades do planejamento do projeto de software. Propõe também, uma metodologia para a elaboração de estimativas utilizando o modelo *Constructive Cost Model II (CoCoMo II)* e apresenta o resultado da aplicação dessa metodologia considerando a comparação do custo estimado, o custo alcançado e o custo proposto pela metodologia em projetos concluídos, para justificar uma mudança de paradigma na elaboração de estimativas.

1.Introdução

Um gerente de projetos de desenvolvimento de software deve ter a capacidade de estimar e mensurar o custo, o tempo e esforço exigidos para os projetos de software a serem desenvolvidos ou em desenvolvimento. Para isto, deve conhecer a capacidade de sua equipe e os recursos com os quais pode contar para executar as atividades. Desta forma, adequando-se ao custo disponível e à qualidade desejada, o gerente poderá estabelecer *trade-offs*¹ para a realização dessas atividades.

Planejamentos com base empírica podem tornar inviável o projeto de software, por não atender requisitos básicos, como as etapas do ciclo de produção de software². As conseqüências se tornam visíveis no resultado final, em retrabalho³ na execução dos processos, o não atendimento das necessidades previstas e o atraso das atividades, resultando no atraso do projeto. A necessidade de realizar estimativas para o desenvolvimento de um projeto de software, aumentando assim a competitividade e evitando perdas torna-se, então, fator fundamental nas organizações.

Na próxima seção, é apresentada a metodologia utilizada e o perfil de empresa estudado, na seção 3, o modelo *Constructive Cost Model (CoCoMo)* é brevemente apresentado. O restante do artigo encontra-se assim organizado: na seção 4 é realizada

¹ *Trade-offs* são prioridades para a execução dos projetos; os mais utilizados são custo, tempo e qualidade, não necessariamente nesta ordem.

² Segundo Pressman (1995), as etapas básicas do ciclo de produção de software são: a análise de requisitos, projeto, desenvolvimento, testes e implantação / manutenção.

³ O retrabalho, de acordo com Pressman (1995), caracteriza-se como ciclos em torno da mesma atividade, seja de análise, projeto, ou desenvolvimento de software.

uma identificação do problema de estimativas, na seção 5 são montadas as etapas de construção do modelo para as estimativas, a seção 6 apresenta uma simulação de um problema na elaboração de estimativas para ser resolvido com o modelo CoCoMo II, na seção 7 é apresentado o produto analisado e a aplicação do modelo é feita, e na seção 8, uma análise e comparação dos resultados alcançados é realizada, e por fim uma conclusão é apresentada na seção 9.

2. Metodologia de Pesquisa

A metodologia considerada mais apropriada ao presente artigo é a do estudo de caso qualitativo segundo Yin (1994), pois é considerado um método holístico para o desenvolvimento de pesquisas, permitindo que a contextualização do problema seja bem desenvolvida.

O perfil da empresa caracteriza-se por ter sua atividade principal a área de engenharia, utilizando a Tecnologia da Informação (TI) como ferramenta de suporte às suas atividades principais, sem utilizar normas e padrões específicos de TI nos seus processos de software, assim como no desenvolvimento de seus produtos de software. Desta forma o modelo CoCoMo II foi utilizado para avaliar o planejamento de desenvolvimento de produtos da empresa, comparando os processos realizados atualmente, com o resultado da aplicação do modelo CoCoMo II.

3. Constructive Cost Model (CoCoMo)

O *Constructive Cost Model*, (CoCoMo), segundo Boehm (2000), é um método que busca medir esforço, prazo, tamanho de equipe e custo necessário para o desenvolvimento do software, desde que se tenha a dimensão do mesmo, através de um modelo de estimativa de tamanho de software, como Análise de Pontos de Função.

Devido à idade dos projetos que embasaram o modelo anterior, assim como sua incapacidade de lidar com ciclos iterativos e com a utilização de componentes *Commercial-Off-The-Shelf* (COTS)⁴, o CoCoMo 81, Boehm (1981), é atualmente considerado obsoleto, tendo sido substituído pela sua versão II, publicada inicialmente em 2000, segundo Aguiar (2004).

Para o cálculo do custo deve-se conhecer o prazo e equipe de trabalho, para então chegar ao valor, sendo que para definir o tamanho do programa, torna-se necessário que se caracterize que medida será adotada (linhas de código, pontos por função ou pontos por caso de uso).

⁴ Comercial-off-the-shelf (COTS) são componentes de software prontos para utilização, de terceiros, comercialmente disponíveis, e que se tornam importantes durante a criação do novo software, devendo ser utilizados preferencialmente durante a fase de pré-desenvolvimento do produto (software), segundo Boehm (2000).

O CoCoMo II estima o custo e tempo baseado em pessoas/mês e meses, para a determinação do *baseline*⁵ de exigências de um produto para a conclusão de uma atividade. O modelo ainda prevê um adicional de 20% ao tempo computado, como margem de erro (análise de risco). Divide-se ainda, em três (3) sub-modelos que abordam as diferentes fases em que o projeto ou atividade que está sendo realizada se encontra, onde a utilização de cada sub-modelo aumenta a precisão e fidelidade da estimativa ao longo do processo de planejamento e execução de um projeto: O modelo da *Application Composition* é o mais apropriado para o estágio de prototipação no ciclo de vida espiral. O modelo *Early Design* é apropriado quando as exigências são conhecidas e as alternativas de arquitetura do software foram exploradas. O modelo *Post-Architecture* é o modelo o mais detalhado e envolve as etapas de construção real do software e de manutenção.

Os objetivos primários do CoCoMo II são definidos como sendo:

- Um modelo de estimativa de custo para o desenvolvimento de software de acordo com o modelo de ciclo de vida de software vista em Jones (1986), e as práticas de desenvolvimento da última década.
- Criar ferramentas de suporte capazes de fornecer melhoramentos do modelo, através de manutenção de informações sobre o desenvolvimento de software em uma estrutura de base de dados.
- Fornecer um *framework* analítico, e um conjunto de ferramentas e técnicas para avaliação dos efeitos de melhoria na tecnologia e nos custos despendidos no ciclo de vida de desenvolvimento de software.

Estes objetivos foram definidos suportar as necessidades primárias da estimativa de custo, devendo fornecer suporte para a etapa de planejamento de projetos, equipe de desenvolvimento necessária, preparação inicial do projeto, replanejamento, negociação de contratos, avaliações de propostas, necessidades e níveis de recursos (tecnológicos e humanos). A seguir serão apresentadas as características do modelo CoCoMo II.

3.1. Calibração do Modelo

A calibração do modelo para um ambiente consiste no ajuste da variável “A”, que indica o percentual de reuso de código, que será vista na seção 3.4 deste artigo, da fórmula de esforço. O objetivo da calibração é obter a distribuição da produtividade e atividade de desenvolvimento para um ambiente específico. O modelo CoCoMo II foi originalmente calibrado com dados de 161 projetos, selecionados entre 2000. Para cada um dos projetos escolhidos foram realizadas entrevistas e visitas, a fim de garantir a consistência das definições e suposições do modelo. O modelo nominal vem calibrado para esses projetos, cuja natureza pode diferir daquele que se deseja estimar.

⁵ Uma baseline é um conjunto de produtos aceitos e controlados, e que serão utilizados em atividades posteriores à sua aceitação, segundo Leite (1997)

Embora o CoCoMo II possa ser executado com os parâmetros nominais, pressupõe-se a calibração para o ambiente-alvo. Com a calibração em projetos do mesmo ramo é possível modificar o valor das constantes das fórmulas padrões. Na ausência de dados históricos disponíveis para o ambiente-alvo em questão, devem ser selecionados projetos equivalentes para efetuar a calibração. Os dados históricos selecionados devem ser validados antes de sua utilização, calculando os coeficientes calibrados e, posteriormente, verificando se a diferença percentual (estimado – real)/estimado encontra-se compatível com o nível de erro pretendido para as estimativas. Devido ao seu impacto exponencial, não é recomendável calibrar o modelo quando houver menos de 10 projetos disponíveis para a calibração, conforme Boehm (2000).

3.2. Fatores de Escala (SF)

Os modelos de estimativa de custo de software possuem um fator exponencial para considerar as despesas e economias encontradas em projetos de software de tamanhos distintos. Para determinar se o projeto apresenta despesas ou economias, uma variável é utilizada como expoente da equação de esforço, $b = 0,91 + 0,01 \times \sum SF_j$. A equação como um todo será vista com mais detalhes na seção 3.4.

A partir do valor de “b”, pode-se capturar os efeitos do projeto e chegar as seguintes premissas: Se o valor de b for abaixo de 1.0 então o projeto apresenta economia. Se for igual a 1.0 então as economias e gastos estão em um nível de equilíbrio. Este modelo linear é utilizado a rigor para a estimativa de custo de projetos pequenos. Se b for maior que 1.0 então o projeto apresenta gasto.

O expoente “b” é obtido através dos fatores de escala. Cada fator tem um intervalo de níveis de valores que vão desde Muito Baixo até Extra Alto. Cada nível de valores possui um peso, SF, e o valor específico do peso é chamado Fator de Escala. Um fator de escala de um projeto é calculado realizando o somatório de todos os fatores e é utilizado para determinar o expoente “b”.

3.3 Multiplicadores de Esforço (EM)

Os drivers de custo são utilizados para capturar características do desenvolvimento de software que afetam o esforço para completar o projeto e possuem um nível de medida que expressa o impacto do driver no esforço de desenvolvimento e variam desde Extra Baixo até Extra Alto. Para o propósito de análise quantitativa, o nível de medida de cada driver de custo possui um peso associado. A este peso se dá o nome de Multiplicador de Esforço (EM). A medida atribuída a um driver de custo é 1.0 e o nível de medida associado com este peso é chamado de nominal. Se um nível de medida produz mais esforço de desenvolvimento de software, então seu correspondente EM possui valor acima de 1.0. Reciprocamente se o nível de medida reduz o esforço, então o correspondente EM possui valor abaixo de 1.0.

Os EM são utilizados para ajustar o esforço Pessoas/Mês nominal. Os multiplicadores de esforço são utilizados nos modelos *Early Design* e *Post-Architecture*,

e a sua influência dentro da fórmula de esforço dá-se da seguinte forma: $PM = A \times (Size)^b \times \Pi EM_i$ ⁶.

3.4 Realizando Medições através do CoCoMo II

A estimativa no modelo CoCoMo II é realizada a partir do resultado de um modelo de estimativa de tamanho do software. A particularidade neste modelo é que os pontos de função são considerados quando ainda não foram ajustados, ou seja, não é o resultado final da estimativa de tamanho de software, visto que na maioria dos casos é difícil determinar exatamente o número de linhas de código que o projeto ou a atividade conterá. Isto acarreta que ambos os modelos – de estimativa de tamanho e CoCoMo II, sejam perfeitamente compatíveis e complementares, de acordo com Cocomo Manual (2004).

A seguir serão apresentados os passos a serem seguidos para utilização do método CoCoMo II para se obter a estimativa do Custo, Esforço e Tempo para o projeto ou atividade a ser desenvolvida, o que, de acordo com a fase em que o projeto se encontra, utiliza-se o modelo adequado. Para este caso apresentaremos a estimativa pelo modelo *Post-Architecture*, por se tratar do modelo mais detalhado.

1º Passo – Calibrar o CoCoMo II para a base histórica de projetos obtida, salvando os coeficientes em um modelo.

2º Passo – Aplicar a fórmula para o cálculo do esforço de desenvolvimento.

$E = a \times A \times KSLOC^b \times \Pi$ (drivers de custo);

- ✓ O fator “a” é determinado pelo tipo de projeto;
- ✓ O fator “A” determina o percentual de reutilização de código; e possui o valor de 2,94, conforme a calibração do CoCoMo II em Cocomo Manual(2004),
- ✓ O expoente “b” é derivado da soma dos 5(cinco) fatores de escala utilizando a fórmula: $b = 1,01 + 0,01 \times \sum W_i$,

Onde cada fator “W” possui um intervalo de valores entre 5 (baixo) e 0 (alto). Os fatores são:

Precedência - Flexibilidade de desenvolvimento - Resolução do risco/arquitetura
- Coesão da equipe - Maturidade do processo

3º Passo – Aplicar o cálculo do tempo de desenvolvimento, e estimar o prazo.

$TDEV = C \times \{PM^F\}$

Onde:

C é uma constante multiplicativa para tempo de desenvolvimento.

(C = 3.67 conforme calibragem do CoCoMo II – Cocomo Manual(2004))

⁶ Π significa produtório, que simboliza o total de diversas mutlplicações.

$$F = [D + 0.2 \times (b - 1.01)]$$

D é uma constante multiplicativa para tempo de desenvolvimento.

(D = 0.28 conforme calibragem do CoCoMo II – Cocomo Manual(2004))

b assume o mesmo valor da fórmula para cálculo de esforço

PM é o esforço calculado

“TDEV” é expresso em quantidade de meses.

4º Passo – A Equipe Média é obtida através da divisão do Esforço pelo Prazo. O CoCoMo II considera um mês de trabalho equivalentes a 152 horas de trabalho, excluídos finais de semana e feriados.

$$P = PM/TDEV$$

“P” é expresso em quantidade de homens

5º Passo – Calculo do custo.

Com o prazo (**TDEV**) e Equipe Média (**P**) é possível estimar o custo do software, conhecendo, o valor hora de cada integrante envolvido no projeto.

4. Identificação do Problema

Os principais problemas enfrentados por um Gerente de Projetos de Software estão relacionados aos processos de estimativa dos custos e prazos de entrega dos projetos.

O problema que se apresenta é o fato que, com relação aos produtos de software e das atividades que fazem parte de um projeto, as mensurações e estimativas são realizadas seguindo um modelo empírico, baseado na experiência dos responsáveis pela fase de análise dos requisitos e também pelo conhecimento de quem realiza a atividade. Isso dificulta a obtenção de informações precisas sobre os projetos realizados, para efeitos de comparação do que foi estimado com o que foi realizado, bem como para a realização de estimativas para os novos projetos que serão realizados.

Para estes novos produtos, supõe-se a utilização de uma estimativa por semelhança, seguindo a percepção dos gerentes e analistas; de maneira totalmente empírica o gerente ou analista, ao receberem as especificações do projeto, levam-nas ao projetista e conjuntamente avaliam e discutem a melhor maneira de realizá-las, sendo que a definição da estimativa é realizada nesse momento.

5. Etapas de Construção do Modelo

O primeiro obstáculo a ser superado para que as métricas de software e as métricas de projetos sejam utilizadas, é a efetiva implementação dessas métricas nas empresas que desenvolvem projetos de software para gerarem os dados necessários.

O segundo obstáculo a superar é a obtenção de uma ferramenta que auxilie o Gerente de Projetos de Software na compilação e análise dos dados gerados, que não são

poucos. Tudo isso seria impossível se não existissem métricas de software para o projeto software que está sendo desenvolvido.

Para realizar a modelagem do projeto, a melhor maneira é utilizar as ferramentas e metodologias apropriadas, que nos permitem realizar essas estimativas de forma mais precisa. Assim será possível obter melhores resultados e contar com um histórico para atividades subseqüentes ou futuras.

6. Descrição do Caso – Metodologia de Elaboração de Estimativas

Nesta seção é apresentada a maneira como são realizadas as estimativas atuais de custo e prazo dos projetos e atividades desenvolvidas na empresa “case”.

1) O Gerente de Projetos de Software recebe uma solicitação de um novo projeto, ou funcionalidade, ou de uma alteração.

2) Após analisar a solicitação, repassa-a ao Analista de Sistemas, para elaborar uma especificação funcional mais detalhada, repassando-a ao Projetista de Softwares.

3) O Gerente de Projetos, o Analista e o Projetista verificam quem dos recursos humanos disponíveis teria capacidade de realizar a atividade, definindo assim o número de pessoas que realizará a atividade.

4) Analista, Projetista e desenvolvedor(es), discutem as premissas necessárias para a atividade, que foram especificadas e definidas na etapa 2.

5) A estimativa de tempo é passada para o(s) desenvolvedor(es), que podem ou não concordar, ou estipular um novo prazo de acordo com a sua percepção, repassando ao Gerente de Projetos e Analista até que se chegue a um acordo.

6) Finalizando, há a definição do custo do desenvolvimento, o qual pode ser definido seja por algum projeto ou atividade anteriormente realizado e que possa ter alguma similaridade, mas que geralmente é definido através da percepção do Gerente de Projetos, sendo utilizado para realização de orçamento para o cliente.

6.1. Escopo das Estimativas Realizadas

Partindo da premissa de que se deve ter em mente quando se considera a qualidade do produto é que a mesma se torna intangível sem um bom gerenciamento da qualidade do projeto, segundo Prado (2000), a estimativa será referente ao **custo do produto**, e não da solução completa. O custo da solução completa engloba outros aspectos que estão além do propósito deste artigo, como por exemplo, as visitas realizadas, as etapas de análise, testes, implantação e manutenção do projeto.

É relevante ressaltar, também, que o custo estimado, não é o que está sendo repassado ao cliente final, e sim o custo que a empresa terá para o desenvolvimento do produto, ao custo do produto será adicionado aos demais custos das diversas fases do projeto.

7. Produto de Software Analisado

Para realizar as estimativas com o modelo CoCoMo II, foi utilizado um produto que já estava com seus requisitos e arquitetura totalmente concluídos, desta forma foi estimado utilizando o sub-modelo *Post-Architecture*, visto que por ser o mais detalhado dos sub-modelos, nos gerará uma estimativa mais precisa. O produto consiste em uma solução de gerenciamento de um sistema de automação, portanto pode ser considerado de complexidade alta.

7.1. Aplicação Prática do Modelo CoCoMo II

Para realizarmos o cálculo da estimativa, consideraremos os seguintes dados:

Total de Pontos de Função não Ajustados (PFNA): 1029

Número de KDSI através do ajuste de PFNA com o valor do peso da linguagem utilizada, Visual Basic® = 32: $(1029 * 32) / 1000 = 32,928$.

Encontrando o Esforço de Desenvolvimento

Valor de PM nominal através da equação $A \times (SIZE)^b$: 145 meses

Valor de PM ajustado através da equação $PM_{(nominal)} \times \Pi EM$: 133 meses

Encontrando o Tempo de Desenvolvimento

Valor de TEDV através da equação $3.67 \times PM_{(ajustado)}^{(0,28 + 0,2 \times (b - 1,01))}$: 16 meses

Encontrando a Estimativa de RH

Valor de P através da equação $PM_{(ajustado)} / TDEV$: 8 pessoas

Encontrando o Custo do Produto

TEDV = 16 meses - 152 horas mensais (de acordo com Cocomo Manual (2004))

8 horas diárias - 19 dias úteis mensais, descontando finais de semana e eventuais feriados.

Total de dias = 304

Total de horas = 2432 horas por integrante

2432×8 integrantes = 19.456

Custo aproximado da hora de desenvolvimento de cada integrante = 5

$19.456 \times 5 = 97.280$

Custo estimado = R\$ 97.280 (Noventa e sete mil e duzentos e oitenta reais)

8. Análise dos Resultados

Para realizarmos esta análise, devemos considerar o resultado estimado da forma atual, e o resultado encontrado, para que possa haver uma comparação dos resultados (estimado modelo X estimado atual X realizado).

A estimativa atual é realizada da seguinte forma:

Nº de dias estimados x nº de pessoas utilizadas na atividade x valor hora dos recursos x 8 horas diárias.

A tabela 1 apresenta um comparativo das estimativas realizadas.

Tabela 1 – Estimativa do produto de software através do modelo Post-Architecture

	Estimado Empresa	Estimado CoCoMo II	Realizado
Status produto			Concluído
Tempo (dias)	400	304	325
Esforço (nº pessoas)	8	8	8
Tamanho (KDSI)	Não aplicado	33.09	56.8502
Custo (em reais)	128.000	97.280	104.000

De acordo com as estimativas realizadas, a previsão da empresa era que 400 dias úteis e 8 pessoas seria possível concluir o desenvolvimento, a um custo de R\$ 128.000,00.

Pela estimativa do modelo, o ideal para o desenvolvimento poderia ter sido realizado com 304 dias úteis, 8 pessoas a um custo de R\$ 97.280,00. Visto que é um produto que já está concluído, é possível ter uma idéia mais concreta de como a utilização do modelo pode beneficiar as estimativas:

A estimativa de esforço teve a precisão mais acertada, visto que os resultados não produziram diferenças. Em relação ao tempo de desenvolvimento, a diferença entre estimado e realizado pela empresa foi de 22,07%, visto que o produto levou menos tempo do que o previsto, entretanto na comparação com a estimativa pelo CoCoMo II, a diferença entre estimado e realizado é de 6,46%, tempo este a mais que pode ter sido causado, por exemplo, pela alocação de recurso em outro produto, problemas pessoais, enfim, alguma eventualidade não prevista no cronograma.

No tamanho do produto, visto que a empresa não estima atualmente este item, em relação à estimativa elaborada pelo modelo, a diferença é de 41%, e uma das razões principais é que há repetição de código nos diversos módulos do produto, o que justifica o fato de que não foi estimada a reutilização de código.

Em termos de custo, verificando os dados obtidos, verifica-se que a diferença entre o estimado pela empresa e o efetivamente gasto com o produto foi de 23,07%, a empresa gastou menos do que estimou para o desenvolvimento, entretanto na comparação com a estimativa pelo CoCoMo II, a diferença entre estimado e o que foi gasto é de 6,46%, que seria uma economia considerável de custo se o projeto tivesse seu curso normal estimado, esta diferença de custo é em decorrência dos fatores tempo e tamanho do produto, que influenciaram de forma diretamente proporcional.

9. Conclusão

No decorrer deste artigo procurou-se explorar o modelo CoCoMo II, bem como as características e variáveis necessárias e que influenciam no sucesso da mudança de paradigma e implantação do modelo CoCoMo II. Foram formulados os aspectos necessários para elaboração de estimativas com embasamento, que foram investigadas e aplicadas em um estudo de caso em uma empresa com perfil pré-definido e, finalmente, relatadas e discutidas na seção de análise de resultados.

Sobre o estudo de caso, pode-se inferir que grande parte das diferenças nos valores encontrados para as variáveis estudadas entre os produtos nas estimativas realizadas e nos resultados encontrados, justifica-se pelo fato que os processos de desenvolvimento de software ainda estão sofrendo os efeitos da padronização ora sendo adotada. Assim, por exemplo, reutilização e modularização, podem acarretar a repetição de instruções escritas no produto estudado, ao invés de um encapsulamento de rotinas comuns aos módulos do produto.

Os fatores levantados nas referências bibliográficas e as evidências colhidas no estudo de caso levam à conclusão que o sucesso na mudança de paradigma e implantação de uma metodologia para elaboração de estimativas com embasamento teórico, neste caso o modelo estudado, o CoCoMo II, depende, dentre outros aspectos, das seguintes variáveis:

- Comparação de custo estimado pela empresa, o custo realmente alcançado e o custo proposto pela metodologia em projetos concluídos. Tal comparação é importante, pois, desta forma existe a possibilidade de verificar se há e onde há problemas nos processos de desenvolvimento de software, e desta forma ter a possibilidade de encontrar maneiras de melhorá-los ou atualizá-los.
- No aspecto tecnológico e de projetos, comparação do custo estimado de projetos em andamento ou em planejamento, não apenas em projetos concluídos, para justificar a mudança de paradigma na elaboração de estimativas. Isso viabiliza a possibilidade de aplicar a metodologia e o modelo CoCoMo II como forma de comparação e em projetos que estejam em andamento, possibilitando que haja uma noção real de como a estimativa pode trazer benefícios aos processos de estimativa e elaboração do custo dos produtos e projetos, pois estando em andamento é possível direcionar o andamento do desenvolvimento em relação à estimativa, ou seja, se for detectado algum desvio, pode-se solucionar em meio ao desenvolvimento (iteração), e não trabalhar apenas com hipóteses, que é o resultado da aplicação em projetos concluídos.

Essas foram as variáveis que se apresentaram como as mais relevantes na elaboração do estudo de caso. Entretanto, é possível haver outras variáveis, como por exemplo, a receptividade da equipe à mudança, que podem influenciar nos resultados de planejamento e desenvolvimento de software, apresentando desta forma resultados diferentes e estimativas alteradas.

10. Referências Bibliográficas

- Aguiar, Maurício (2004) “Estimando os Projetos com COCOMO II”. TIMétricas. <http://www.timetricas.com.br>, Maio.
- Boehm, Barry W. (1981) “Software Engineering Economics”. Prentice Hall.
- Boehm, Barry et al. (2000) “Software Cost Estimation With COCOMO II”. Prentice Hall.
- Cocomo Manual (2004) “Cocomo II Model Definition Manual”. <http://sunset.usc.edu/research/COCOMOII>, Agosto.
- Jones, Capers (1986) “A Short History of Functions Points and Feature Points Software Productivity Research”. Inc. Beurlington MA
- Leite, J.C.S.P. et al. (1997) “Enhancing a Requirements Baseline with Scenarios. Requirements Engineering”, Springer-Verlag London
- Prado, Darcy (2000) “Gerenciando Projetos nas Organizações”, Belo Horizonte MG, EDG.
- Pressman, Roger S. (1995) “Engenharia de Software”. São Paulo. Atlas.
- Yin, Robert K. (1994) “Case Study Research, Design and Models”. California. Sage.