

A qualidade do processo de desenvolvimento de software a partir da gestão de projetos: um estudo de caso

Cicero Aparecido Bezerra

Pontifícia Universidade Católica do Paraná

Avenida da União, 500, Jardim Coopagro, Toledo – PR, CEP 85902-532

Resumo

A qualidade é percebida em três dimensões: no projeto, no processo e no produto. Estas três dimensões, quando combinadas, contribuem para a qualidade total do produto. Vários estudos têm sido realizados no sentido de mensurar a qualidade do produto final, que é facilmente perceptível pelo usuário. Em relação a produtos de software, quando as necessidades do cliente não são atendidas, nada mais pode ser feito no sentido de manter sua expectativa com a qualidade, há não ser reparos no software, acarretando retrabalho e consumo extra de recursos. Neste sentido, torna-se natural focar a atenção também em aspectos existentes durante o processo de desenvolvimento, mas precisamente na gestão do projeto, visto sua persistência em todas as fases da engenharia de software. O presente artigo avalia os resultados da implantação de gestão de projetos em uma pequena software-house, sob a ótica dos engenheiros de software, nos aspectos de produtividade, controle, satisfação e inovação, fatores considerados integrantes da qualidade de processo. Esta avaliação é efetuada em dois momentos distintos, com a intenção de verificar sua continuidade, ou se trata apenas de mais uma abordagem passageira que não agrega valor ao produto final.

Palavras-chave: gestão de projetos, engenharia de software, qualidade de processo.

Abstract

The quality is perceived in three dimensions: in the project, in the process and in the product. These three dimensions, when combined, contribute to the total quality of the product. Several studies have been accomplished with the purpose of measuring the quality of the final product, which is easily perceptible by the user. In the software products, when the client's necessities aren't answered, nothing else can be done to keep the user's quality expectation, except for the software repairing, resulting in double work and consuming extra resources. This way, it becomes natural to focus the attention on existing aspects during the development process, but precisely in the project management, seeing its persistence in all phases of software engineering. This article evaluates the results of project management implantation in a small software house, under the sight of software engineers, in the aspects of productivity, control, satisfaction and innovation, factors considered constituent of the process quality. This evaluation is executed in two moments, intending to verify your continuity, or if it's just about a transitory approach which doesn't aggregate real value to the final product.

Key Words: project management, software engineering, process quality.

1. INTRODUÇÃO

Assim como na literatura, a experiência prática tem mostrado que a construção de produtos de software apresenta problemas facilmente perceptíveis, tanto pelo usuário final, como pelos desenvolvedores. No intuito de solucionar estes problemas, novos modelos, metodologias, técnicas e ferramentas são criadas e aplicadas efusivamente. Apesar disto, os problemas vêm persistindo. Pode-se concluir, ainda que superficialmente, que as ocorrências de erro não residem, exclusivamente, nos instrumentos e/ou metodologias empregados exclusivamente no desenvolvimento, devido justamente ao grande número de opções existentes. Desta forma, torna-se natural buscar novas variáveis para equacionar o problema. Ao constatar que a gestão de projetos está (ou deveria estar) presente em todas as etapas, torna-se natural voltar a atenção neste aspecto. A hipótese de que os problemas existentes no software estão mais diretamente vinculados à sua gestão, é reforçada através dos fatores apontados pelo usuário final, como problemas: entrega fora do prazo, custos excessivos e não funcionalidade.

Nesta perspectiva enquadra-se o presente estudo: avaliar o impacto (nos quesitos produtividade, satisfação, inovação e controle), sob a ótica da equipe de desenvolvimento, a partir da utilização da gestão formal de projetos em pequenas software-house's. Para isto será

necessária uma pequena revisão teórica nas questões que envolvem o gerenciamento de projetos e o desenvolvimento de sistemas informatizados em pequenas empresas. O trabalho ainda irá descrever, resumidamente, o processo de implantação e comparar os resultados imediatos (perceptíveis pelos engenheiros de software) e após 2 anos, com a finalidade de verificar se o gerenciamento de projetos, efetivamente agregou valor, provocando rupturas tecnológicas, ou se simplesmente tratou-se de um modismo passageiro, na software-house analisada.

2. REFERENCIAL TEÓRICO

Neste tópico será apresentada a pesquisa bibliográfica no que se refere ao gerenciamento de projetos de software, além de um breve perfil do processo de desenvolvimento de software nas pequenas empresas desenvolvedoras de sistemas informatizados, com o objetivo de fornecer elementos necessários para a compreensão de seus conceitos básicos. É importante salientar que não se pretende esgotar o assunto, mas sim identificar os aspectos principais para a contextualização do tema abordado neste estudo.

2.1 Gestão de projetos de software

Nenhuma técnica de desenvolvimento de software terá resultados positivos se não forem adotados processos ordenados (formais) de gerenciamento de projetos [1]. Devido ao ciclo de vida do desenvolvimento de um produto de software, sua concepção enquadra-se no conceito de projeto que [2], é composto pela execução das fases conceptual (idéia inicial do produto, elaboração e aprovação de proposta), planejamento e organização (decomposição do projeto em etapas, planejado e organizado de forma a evidenciar os aspectos de controle e execução); implementação (desenvolvimento das etapas, sob uma coordenação formal, até a obtenção dos objetivos propostos) e; encerramento (efetivação da transferência dos resultados ao cliente, avaliação geral do projeto e desmobilização da equipe e recursos alocados ao projeto).

No caso específico de implementação de produtos de software, algumas metodologias de gerenciamento vêm sendo empregadas. Dentre elas, pode-se citar o *Unified Software Development Process* [3], que regulamenta a gestão do projeto de desenvolvimento de sistemas, partindo do pressuposto de que os desenvolvedores não dispõem de todas as exigências no início do ciclo de vida do software e, portanto devendo ser previstas mudanças em todas as fases, tornando o processo iterativo e incremental, com atividades executadas em paralelo. Porém, deve-se atentar para a diferença existente entre a engenharia (simultânea), e o “artesanato” (implementação caótica) de software.

Finalmente, sobre a gestão de projetos de software, ainda há muito o que se estudar, uma vez que, em um primeiro momento, é necessário medir os processos e, neste sentido, não existe uniformidade em padrões de medidas capazes de quantificar todos os aspectos que envolvem o desenvolvimento de sistemas informatizados, causando uma verdadeira “guerra de métricas”. Esta situação decorre do fato de que, a cada nova pesquisa, surge um novo modelo conceitual, minimizando esforços para validar os modelos existentes. Ainda que tal atitude promova riqueza e diversidade, limita o amadurecimento da área como disciplina científica. É importante salientar que, apesar da falta de uniformidade (ou de um padrão), a utilização de quaisquer métricas já é um ponto positivo (e essencial) no gerenciamento de projetos.

2.2 Desenvolvimento de software na pequena empresa

Nas empresas desenvolvedoras de software (em especial nas pequenas), o padrão de desenvolvimento começa pela programação, “isto quando não se resume somente a ela”, [1]. Além disso, percebe-se que a criatividade do engenheiro de software (essencial em quaisquer etapas no ciclo de vida) é bem vinda, porém sem restrições tende a nunca chegar a uma

conclusão satisfatória. Por outro lado, “ferramentas disciplinadoras utilizadas sem nenhum contexto criam entraves burocráticos que acabam por sufocar a inovação” [4]. Também encontram-se problemas naquelas empresas que optam por técnicas isoladas de modelagem de software, com o intuito de melhorar a qualidade de seus produtos. Adotadas neste contexto, tornam-se contraproducentes na medida em que não estão integradas em um processo maior [1]. Este problema causa na equipe de desenvolvimento, a impressão de que técnicas de modelagem (e gerenciamento) não funcionam na prática.

As pequenas equipes de desenvolvimento de software têm sido alvo freqüente de metodologias que visam garantir a qualidade do produto final (satisfação do cliente). Uma delas é a *Extreme Programming*, que vem a ser uma “metodologia ágil para equipes pequenas e médias desenvolvendo software com requisitos vagos e em constante mudança” [5]. Esta metodologia baseia-se em práticas como a incorporação do usuário final à equipe de desenvolvimento; planejamento, de modo a priorizar a produção de módulos que agregam valor ao produto; disponibilização de pequenas versões do software; testes definidos e implementados pelo cliente, projetados antes do código; simplicidade na engenharia; desenvolvimento efetuado por pares de pessoas que se revezam a cada duas horas; produção em pares; padronização e conhecimento do código por todos os envolvidos e; produção de 40 horas semanais. De um modo geral, esta abordagem não trouxe contribuições inéditas expressivas para o processo de desenvolvimento de software na pequena empresa. Na verdade, a aplicabilidade de alguns dos itens observados pode ser inviável (em algumas situações, até mesmo impraticável), como por exemplo, a incorporação *full-time* do usuário final à equipe de desenvolvimento. Outras práticas já vêm sendo realizadas corriqueiramente no cotidiano das empresas, como por exemplo, pequenas versões do software, testes isolados e efetuados (também) pelo usuário final. Algumas preconizam o óbvio (40 horas semanais, software simples). Porém, sua validade reside no fato de resgatar e realizar um *upgrade* nas boas práticas de desenvolvimento de software.

3. ESTUDO DE CASO

O processo de implantação de gerenciamento de projetos teve início em maio de 2000, em uma empresa de fornecimento, instalação e manutenção de hardware, que possuía em sua carteira de clientes, basicamente pequenas e médias empresas, atuando no mercado regional de Cascavel – PR há 8 anos, sendo que passou a oferecer serviços de desenvolvimento de sistemas de informações desde 1998. No período em que iniciou este estudo de caso, o faturamento da empresa era, aproximadamente, R\$ 10.000,00 ao mês. A equipe de desenvolvimento de sistemas compunha-se de um analista de sistemas, três programadores e um analista de suporte, cujas tarefas eram alocadas conforme a necessidade cotidiana na empresa, sendo que todos possuíam conhecimentos avançados no ambiente de programação, ainda que não conhecessem metodologias formais de desenvolvimento de software (análise, projeto e programação estruturada ou orientada a objetos, por exemplo).

3.1 Implantação de desenvolvimento por projetos

O passo decisivo para a implantação de gerenciamento de projetos na empresa estudada foi a constatação, por parte do proprietário, que os softwares desenvolvidos poderiam ter maior competitividade se distintos da concorrência. Técnica e funcionalmente, os produtos não apresentavam grandes diferenciais, visíveis ao cliente, em relação aos concorrentes. Portanto a estratégia voltou-se ao cumprimento de prazos, estimativa de custos e menor índice de manutenção após a entrega do software. Visto que os pontos críticos no processo estavam mais próximos do gerenciamento, do que em ferramentas ou metodologias de

desenvolvimento, foi estabelecido um programa de treinamento específico em técnicas de gestão de projetos.

A primeira etapa do programa de implantação constituiu-se de uma palestra sobre gerenciamento de projetos, mostrando sua importância no desenvolvimento de produtos, em termos de eficiência (razão entre recursos consumidos e objetivos alcançados) e eficácia (razão entre objetivos alcançados e objetivos atingidos). Estes aspectos foram considerados importantes para introduzir a idéia matemática de mensuração de resultados, no processo de desenvolvimento.

A seguir, ministraram-se mini-cursos específicos sobre ferramentas de gerenciamento de projetos. Adotou-se o modelo PERT/CPM, por julgá-lo suficientemente fácil para aprendizagem e prática. Dividiu-se o modelo adotado em 4 etapas (estrutura de decomposição de atividades, PERT/CPM, PERT/Custo e aceleração de atividades).

A próxima etapa foi baseada no *Unified Software Development Process*, a qual mostrava a aplicabilidade, as possibilidades e as vantagens do planejamento e (posterior) execução simultânea de processos de desenvolvimento de software, bem como a utilização do modelo PERT/CPM nesta abordagem.

Finalmente, utilizaram-se algumas das boas práticas da atividade de programação, atualizadas pela disciplina da *Extreme Programming*, para concluir o programa de treinamento. Foram enfatizados os aspectos referentes à importância da participação do usuário final na definição e implementação de testes, desenvolvimento dos módulos que mais agregam valor final ao produto em primeiro lugar e, padronização de código (bem como demais subprodutos decorrentes de cada interação no processo de desenvolvimento).

Ao todo, o programa de treinamento levou aproximadamente 32 horas, distribuídos em 12 dias. Alguns fatores tornaram-se facilitadores do processo, tais como: a estrutura organizacional que já possuía grande foco na execução de projetos, ainda que informalmente gerenciados; a inexistência de outro projeto em andamento e; o fato de que toda a equipe (analistas, programadores e gerente de projeto) recebeu treinamento das mesmas ferramentas e técnicas, sem distinção de função, possibilitando uniformidade em aspectos referentes a como o processo seria medido e, conseqüentemente, controlado.

O projeto piloto, gerenciado formalmente, consistiu de um sistema de informação comercial, contendo os módulos clientes e fornecedores, compras, vendas e estoque, contas a receber, contas a pagar e fluxo de caixa para uma indústria de insumos agrícolas, atuando no mercado regional de Cascavel – PR desde 1984, cujo faturamento no ano de 1999 foi de aproximadamente R\$ 1.000.000,00. Esta indústria, na época, contava com 20 funcionários e 3 gerentes (financeiro, produção e administrativo / comercial) e não tinha experiência anterior na utilização de sistemas informatizados de gestão e operações.

3.2 Instrumento de validação

O instrumento utilizado para medir o resultado da implantação da metodologia de desenvolvimento por projetos, na empresa, foi desenvolvido por Torkzadeh e Doll em 1999 [6], e fundamenta-se em 4 dimensões: produtividade, inovação, satisfação e controle. Optou-se por este instrumento devido suas características de avaliação multidimensional do impacto sobre os indivíduos, facilidade de aplicação e sua capacidade em ser adaptado para diversos contextos. O modelo proposto por Torkzadeh e Doll, utilizou a pesquisa *survey*, sob a forma de um questionário, para determinar quantitativamente (em uma escala Likert, onde nada = 1, um pouco = 2, moderadamente = 3, muito = 4 e muitíssimo = 5) características de constructos das 4 dimensões representativas do impacto da adoção de determinada tecnologia na satisfação do usuário final. As dimensões, os constructos e as questões, do instrumento implementado por Torkzadeh e Doll, podem ser visualizadas no QUADRO 1:

QUADRO 1 – Instrumento de validação

Dimensão	Constructo (Em que medida...)	Questões (A tecnologia...)
Produtividade	a aplicação interfere na produção do usuário em determinada unidade de tempo	a) poupa-me tempo b) melhora minha produtividade c) permite-me melhores resultados do que seria possível executar sem ele
Inovação	a aplicação ajuda a criar ou tentar expressar novas idéias em seu trabalho	d) ajuda-me a criar novas idéias e) permite-me propor novas idéias f) coloca-me diante de idéias inovadoras
Satisfação do Usuário	a aplicação ajuda o usuário a criar valor para os clientes internos e externos à organização	g) melhora o serviço do usuário h) melhora a satisfação do usuário i) vai ao encontro às necessidades do usuário
Controle Gerencial	a aplicação ajuda a regular processos e desempenho	j) ajuda no controle gerencial do processo de trabalho k) melhora o controle do gerenciamento l) ajuda no controle do gerenciamento de performance do processo de trabalho

3.3 Resultados imediatos

Após o desenvolvimento do projeto piloto, o qual foi implantado em 11 semanas (dentro do prazo estabelecido junto ao cliente), aplicou-se o questionário, individualmente, por escrito, e sem identificação (com o intuito de minimizar tendências ou influências do grupo e manter a isenção) aos envolvidos no desenvolvimento do projeto. A análise dos resultados indica que o desenvolvimento do projeto piloto, gerenciado formalmente, trouxe índices significativos de satisfação para a equipe de desenvolvimento, conforme visualizado na TABELA 1, onde as colunas referentes à escala Likert correspondem à frequência de respostas em cada constructo; as colunas referentes aos totais se referem ao somatório do total de cada quantidade de respostas multiplicada pela escala correspondente e; as colunas referentes à porcentagem correspondem à aproximação dos totais, do valor máximo possível, para cada constructo:

TABELA 1 – Resultados 2000

Constructo	Resultados 2000							
	Escala Likert					Σ	%	
	1	2	3	4	5			
Produção	a) Poupa tempo	0	0	1	2	2	21	84
	b) Melhora a produtividade	0	1	2	1	2	19	76
	c) Melhores resultados	1	1	2	1	0	13	52
	Totais	1	2	4	4	4	53	70,6
Inovação	d) Criar novas idéias	0	1	1	1	2	19	76
	e) Propor novas idéias	0	0	1	2	2	21	84
	f) Diante de inovação	0	1	1	1	2	19	76
	Totais	0	2	3	4	6	59	78,6
Satisfação	g) Melhora o serviço	0	1	2	2	0	16	64
	h) Melhora a satisfação	1	0	1	2	1	17	68
	i) Encontrar necessidades	0	1	1	1	2	19	76
	Totais	1	2	4	5	3	52	69,3
Controle	j) Processo do trabalho	0	1	2	1	1	17	68
	k) Controle da gestão	0	1	1	2	1	18	72
	l) Performance e trabalho	0	1	1	1	2	19	76
	Totais	0	3	4	4	4	54	72

Os números apresentados mostram que o constructo “inovação” teve o maior percentual de aprovação (78,667%). A inovação percebida reside-se no fato de que o gerenciamento pôde medir matematicamente o progresso do projeto – circunstância inédita, até então, para a equipe de desenvolvimento. Com relação à “inovação”, o quesito “o sistema

permite-me propor novas idéias” surpreendeu pelo elevado índice (84%). O motivo alegado pela equipe foi a possibilidade de compartilhar idéias, promovido pelos aspectos interativos e incrementais da *Unified Software Development Process* e de uma abordagem de engenharia simultânea no processo de desenvolvimento de software. Já o quesito “o sistema permite-me melhores resultados do que seria possível executar sem ele” do constructo “produtividade” apresentou o menor índice de aprovação: pouco mais da metade da equipe de desenvolvedores confirmou este aspecto. Este fato é explicado ao analisar as resistências às mudanças intrínsecas às pessoas ao saírem, ou serem obrigadas a sair, da zona de conforto onde, até então, julgavam ser absolutas conhecedoras de suas atividades [7]. Outra razão para o resultado apresentado é a constatação de que o software, independente de técnica, metodologia de desenvolvimento, ferramenta e / ou gerenciamento do processo, irá ser concluído. Além disso, existe a crença (errada) de que os problemas decorrentes do desenvolvimento informal (custos e prazos estourados e não funcionalidade) fazem parte do processo e que, sendo assim, não existem razões para minimizar esta situação.

De um modo geral, as etapas de planejamento de projeto e as etapas do ciclo de desenvolvimento de software, principalmente levantamento de requisitos, análise e projeto, levaram um tempo maior do que aquele existente anteriormente em sistemas da mesma natureza, o que levou a equipe a ter, preliminarmente, uma idéia de baixa produtividade, uma vez que o software, propriamente dito, ainda não havia sido iniciado. Porém, após este momento, a etapa de codificação mostrou-se extremamente rápida e “mecânica” [8]. Infelizmente não se adotou nenhuma métrica científica para medição do software, sendo que este valor foi estimado de acordo com a experiência do grupo de trabalho.

Para que a implantação do gerenciamento de projetos, neste estudo de caso, obtivesse sucesso, foi necessária a quebra de alguns paradigmas [7]. O primeiro deles foi a busca por resultados imediatos (processos de transformação levam mais tempo para mostrar resultados). Evitou-se também a indução ao erro através de um canal aberto e efetivo de comunicação entre proprietário da software-house, consultor, desenvolvedores e usuários finais, de forma a tornar claro o significado de todas as ações, os pontos de controle e os objetivos esperados.

3.4 Resultados posteriores

Após a análise dos resultados, uma grande dúvida persistiu: projetos posteriores seriam gerenciados formalmente pela empresa? Se sim, que resultados poderiam ser percebidos? Se não, qual o motivo que levou a desistência de se utilizar esta abordagem? A partir destas questões, passados 2 anos após o estudo, procurou-se avaliar a empresa novamente, utilizando-se a mesma ferramenta, com o intuito de verificar se o gerenciamento de projeto, naquele contexto, tratou-se de uma metodologia passageira, que não representou nenhum avanço, ou um *change trigger* (avanço tecnológico provocador de rupturas) [7].

A empresa, no momento em que efetuado o novo estudo, contava com o mesmo número de desenvolvedores, apesar de terem substituído 2 colaboradores no ano de 2001. Quando da primeira avaliação, aproximadamente 75% de seu faturamento correspondia à venda, manutenção e instalação de hardware, sendo que transcorridos 2 anos, a comercialização e manutenção de sistemas de informação já representava até 50%, em média, do faturamento total, que atingia até R\$ 25.000,00 ao mês.

Verificou-se, nesta ocasião, a utilização maciça da gestão de projetos, não somente em desenvolvimento de software, mas também na forma como a empresa passou a comercializar, nas palavras do proprietário, “soluções e não mais produtos”. Neste cenário, aplicou-se novamente o questionário e seus resultados podem ser visualizados na TABELA 2:

TABELA 2 – Resultados 2002

Constructo	Resultados 2002							
	Escala Likert					Σ	%	
	1	2	3	4	5			
Produção	a) Poupa tempo	0	0	1	2	2	21	84
	b) Melhora a produtividade	0	0	1	2	2	21	84
	c) Melhores resultados	0	0	1	1	3	22	84
	Totais	0	0	3	5	7	64	85,33
Inovação	d) Criar novas idéias	0	1	1	1	2	19	76
	e) Propor novas idéias	0	1	1	2	1	18	72
	f) Diante de inovação	0	1	1	2	1	18	72
	Totais	0	3	3	5	4	55	73,33
Satisfação	g) Melhora o serviço	0	0	1	2	2	21	84
	h) Melhora a satisfação	0	1	1	2	1	18	72
	i) Encontrar necessidades	0	0	1	2	2	21	84
	Totais	0	1	3	6	5	60	80
Controle	j) Processo do trabalho	0	0	1	2	2	21	84
	k) Controle da gestão	0	0	0	2	3	23	92
	l) Performance e trabalho	0	0	1	2	2	21	84
	Totais	0	0	2	6	7	65	86,67

Em um primeiro momento, pode-se constatar o aumento expressivo na satisfação observada pela equipe de desenvolvimento de software, principalmente nos constructos “controle”, que passou de um percentual de 72% para 86,67% de aprovação; “produtividade”, de 70,667% para 85,33%; e “satisfação”, de 69,333% para 80%. Infelizmente, a empresa procurou agregar poucas novas técnicas e metodologias no desenvolvimento do processo de engenharia de software, o que explica a queda (de 78,667% para 73,33%) no constructo “inovação”. Os processos que levam à qualidade possuem a característica da “melhoria contínua” e a estagnação (ou acomodação), neste contexto, contribui para a percepção de não estarem praticando a inovação.

3.5 Limitações e recomendações ao estudo

A implantação da gestão de projetos na empresa analisada foi relativamente fácil devido à sua característica de já trabalhar com equipes alocadas a cada projeto individual. Desta forma, recomendam-se estudos dos impactos e dos resultados obtidos naquelas empresas que possuem uma estrutura organizacional já formalizada, levando em consideração o surgimento de conflitos devido à quebra de modelos mentais e organizacionais. Nas software-house's onde os colaboradores estejam envolvidos em mais de um projeto, uma grande dificuldade será a mensuração do tempo de alocação de cada um a cada tarefa de cada projeto. Talvez o principal desafio seja balancear a criatividade com uma maneira disciplinada de controle da atividade de desenvolvimento.

4. CONCLUSÕES

A abordagem de desenvolvimento por projetos na pequena empresa encontra algumas barreiras de caráter empírico, por exemplo: é comum adotar a postura de que um bom programador será um bom analista de sistemas e que, um bom analista de sistemas será um bom gerente de projetos. Esta situação até pode ocorrer (e quando acontece, é bastante saudável), mas não como via de regra. Este é um paradigma a ser quebrado em desenvolvimento de software por projetos. As competências devem ser alocadas conforme a atividade a ser desenvolvida. Isto é importante no sentido de determinar uma pessoa plenamente apta a gerenciar o projeto, no sentido literal da palavra gerente, possuindo conhecimentos organizacionais e técnicos, habilidades de comando e relacionamento pessoal,

atitudes de disciplina, interesse por questões administrativas, entrosamento com pessoal externo à organização e ser um estrategista [2]. Além disso, há que se concordar com Watts Humphrey, quando afirma que “a ausência de práticas administrativas é a principal causa de sérios problemas enfrentados pelas organizações: atraso em cronogramas, custo maior do que o esperado e presença de defeitos, ocasionando uma série de inconveniências para os usuários e enorme perda de tempo e de recursos dos desenvolvedores” [9]. Em uma pequena software-house, a produção e o gerenciamento dos projetos devem estar situadas no mesmo nível hierárquico dos gerentes funcionais caracterizando uma estrutura organizacional do tipo matricial balanceada [2].

Evidentemente, a empresa que adota a postura do gerenciamento de projetos, deve estar em um nível de maturidade organizacional elevado ou, no mínimo, estar consciente desta necessidade. Isto é determinante pelo fato de que problemas irão existir, porém serão mais facilmente equacionados e resolvidos. Outra postura que irá contribuir positivamente é a consciência de que a gestão formal de projetos é apenas uma das etapas para o objetivo maior da empresa, que é a qualidade total e, neste sentido, é importante observar que deverá ocorrer uma evolução para técnicas, metodologias e modelos mais específicos.

REFERÊNCIAS

- [1] BELLOQUIM, Átila. Modelagem de Software: Ontem, Hoje e Amanhã. **Developers' Magazine**. Rio de Janeiro: Axcel Books, ano 6, n.70, p.10-13, jun.2002.
- [2] VALERIANO, Dalton L. **Gerência em Projetos**. São Paulo: Makron Books, 1998.
- [3] KRUCHTEN, Philippe. **Introdução ao RUP Rational Unified Process**. Rio de Janeiro: Ciência Moderna, 2003.
- [4] QUATRANI, Terry. **Visual Modeling with Rational Rose and UML**. 3. ed. New York: Addison-Wesley, 2002.
- [5] BECK, Kent; FOWLER, Martin. **Planning Extreme Programming**. New York: Addison-Wesley, 2000.
- [6] PEREIRA, Maria Tereza Flores; BECKER, João Luiz. **O Impacto da Tecnologia de Informação (TI) sobre o Processo de Trabalho Individual**: Estudo em um Grande Banco Brasileiro. Atibaia, 2003. Anais do XXVII Encontro da Associação Nacional dos Programas de Pós-Graduação em Administração. Rio de Janeiro: ANPAD, 2003. CD-ROM.
- [7] HEHN, Herman F. **Peopleware: Como Trabalhar o Fator Humano nas Implementações de Sistemas Integrados de Informação (ERP)**. São Paulo: Editora Gente, 1999.
- [8] PRESSMAN, Roger. **Software Engineering: a Practitioner's Approach**. 5. ed. New York: McGraw-Hill, 2003.
- [9] TEIXEIRA JÚNIOR, Waine; SANCHES, Rosely. **Proposta de um Modelo de Processo de Planejamento de Projeto de Software Para Melhoria de Gerenciamento de Projetos**. Curitiba, 2001. Anais do XII Congresso Internacional de Tecnologia de Software. Curitiba: CITS, 2001. CD-ROM.