

Metodologia para Implementação do MPS.BR Utilizando o Ambiente WebAPSEE

Vanderlene Covre^{1,2}, Carla Lima Reis², Eloi Luiz Favero¹

¹Universidade Federal do Pará (UFPA) – Belém – PA - Brasil
PPGEE - Programa de Pós-Graduação em Engenharia Elétrica

²Universidade Federal do Pará (UFPA) – Belém – PA - Brasil
Laboratório de Engenharia de Software (LABES)

vanderlene@gmail.com, {clima, favero}@ufpa.br

Abstract. *A number of software process improvement initiatives emerged recently aiming to improve software quality and productivity in software development organizations. Some models and standards, such as the IDEAL and MPS.BR models, are examples of these initiatives. In this context, the present work proposes a deployment methodology for the MPS.BR model based on IDEAL model, through the use of a specific tool, called WebAPSEE. Currently, this methodology is on validation process, with partial results, at a software development organization. This paper discusses both the methodology and its results.*

Resumo. *Uma série de iniciativas para melhoria de processo de software surgiu recentemente visando melhorar a qualidade e a produtividade em organizações de desenvolvimento de software. Alguns modelos e normas, tais como os modelos IDEAL e MPS.BR são exemplos dessas iniciativas. Neste contexto, o presente trabalho propõe uma metodologia para a implementação do modelo MPS.BR baseada no modelo de implantação IDEAL, através de uma ferramenta específica, chamada WebAPSEE. Atualmente, esta metodologia encontra-se em validação, com resultados parciais, em uma organização de desenvolvimento de software. O artigo apresenta a metodologia e discute seus principais resultados.*

1. Introdução

Na medida em que a sociedade depende cada vez mais de software, presente em grande parte dos produtos e serviços que consome, o desenvolvimento de software passou a ser uma atividade crítica em vários setores (Fuggetta, 2000). No entanto, aplicações de software são complexas e muitas vezes difíceis de desenvolver e testar, sem contar com as restrições de prazo, custos e exigências de qualidade. Questões como essas têm motivado organizações a adotarem modelos de melhoria de qualidade.

Estudos recentes sobre qualidade estão voltados para o aperfeiçoamento do processo de desenvolvimento de software e, como afirma Fuggetta (2000), a qualidade do produto está fortemente relacionada à qualidade do processo. O processo utilizado para desenvolver e manter o software afeta significativamente o custo, a qualidade e o prazo de entrega do produto. O impacto é tão significativo que a melhoria do processo

de software é vista por alguns como a mais importante forma para melhorar o produto de software.

Organizações que almejam melhorar seus processos têm dificuldade em analisar sua situação atual, identificar pontos fortes e pontos fracos e definir ações de melhoria. Embora a ISO 12207 (ISO/IEC 12207, 1998), o CMMI (CMU/SEI, 2002) e o MPS.BR (Softex, 2007) descrevam possíveis estratégias, estas não estão definidas com detalhes suficientes de modo a apoiar as organizações nesse processo de melhoria. A ausência de uma metodologia e o apoio adequado de ferramentas também dificulta a implantação e a evolução da melhoria de processos de software nas organizações.

Diante da necessidade das organizações em melhorar a qualidade do processo de software, o desafio que se percebe é a utilização de uma metodologia para a implantação de melhorias em processos de software que seja vinculada ao uso de ferramentas específicas, com o objetivo de potencializar os resultados da implantação e obter maior comprometimento com a evolução da maturidade da organização. A utilização do modelo MPS.BR é vista como um alicerce para tal metodologia porque representa um modelo voltado à realidade brasileira e compatível com normas internacionais, além de amplamente aceito pela indústria e academia. Considerando o trabalho que vem sendo feito na área de tecnologia de processo de software, está sendo proposta a ferramenta WebAPSEE (WebAPSEE, 2007) para apoiar a implantação da metodologia. O diferencial da ferramenta citada está em permitir modelagem visual e execução flexível de processos de software, registro de métricas, políticas de alocação de pessoas, monitoração visual, reutilização de processos de software, dentre outras funcionalidades.

Este artigo apresenta uma proposta de melhoria de processos com o auxílio de uma ferramenta de modelagem e execução de processos no apoio à implementação do MPS.BR. Esta metodologia está sendo aplicada em uma empresa do estado do Pará, que até então não possuía nenhuma experiência de implementação de processos com base em modelos de qualidade de processos de software.

O restante desse trabalho está assim organizado: na seção 2 é feito um sumário relacionado a alguns modelos/normas de melhoria de processos de software. A seção 3 apresenta a ferramenta utilizada para apoiar a implementação do processo de melhoria. Os trabalhos relacionados estão apresentados na seção 4. A seção 5 expõe a metodologia proposta para implementar o MPS.BR. Na seção 6 são descritas a definição e a implantação de melhoria de processos em uma empresa. Finalmente a seção 7 apresenta as considerações finais.

2. Modelos/Normas de Melhoria de Processos de Software

Vários modelos e normas para qualidade de software ganharam destaque devido a sua qualidade e aplicação. A seguir são apresentados aqueles que foram mais significativos para embasamento do presente trabalho.

2.1. Norma ISO/IEC 12207

A norma internacional ISO/IEC 12207 foi criada pelo esforço conjunto entre a ISO – *International Organization for Standardization* e o IEC – *International Electrotechnical*

Commission e tem como objetivo estabelecer uma estrutura comum para os processos de ciclo de vida do software.

Com terminologia bem definida, tal estrutura contém processos, atividades e tarefas para serem aplicadas durante a aquisição, fornecimento, desenvolvimento, operação e manutenção de produtos de software. Além disso, fornece um processo que pode ser usado para definir, controlar e melhorar os processos do ciclo de vida do software.

Esta norma não está ligada a métodos, ferramentas, treinamentos, métricas ou tecnologias empregadas. Tal determinação é importante para permitir que a norma seja utilizada mundialmente e possa acompanhar a evolução da engenharia de software nas diversas culturas organizacionais. Ela pode ser utilizada com qualquer modelo de ciclo de vida, método ou técnica de engenharia de software e linguagem de programação. Sua flexibilidade é uma característica importante, ela descreve a arquitetura dos processos de ciclo de vida do software, mas não especifica os detalhes de como implementar ou executar as atividades e tarefas incluídas no processo (ISO/IEC 12207, 1998).

A utilização dos processos da ISO/IEC 12207 é de responsabilidade da organização, que dependendo de seu objetivo, pode selecionar um subconjunto de processos apropriado para satisfazê-lo. Os processos são agrupados, por uma questão de organização, de acordo com o seu objetivo principal no ciclo de vida de software. Esse agrupamento resultou em três diferentes classes de processos, que são:

- Processos Fundamentais: Aquisição, Fornecimento, Desenvolvimento, Operação e Manutenção;
- Processos de Apoio: Documentação, Gerência de Configuração, Garantia de Qualidade, Verificação, Validação, Revisão Conjunta, Auditoria e Resolução de Problema;
- Processos Organizacionais: Gerência, Infra-estrutura, Melhoria e Treinamento.

2.2. Modelo PDCA

O princípio básico da melhoria de processos está calcado no ciclo PDCA, proposto por Walter Shewart da década de 20 e difundido amplamente na indústria japonesa após a segunda guerra por um de seus seguidores, William Edwards Deming (Filho, 2006). O ciclo tem por princípio tornar mais claros e ágeis os processos envolvidos na execução da gestão e está dividido em quatro passos: planejamento (*Plan*), execução (*Do*), verificação (*Check*) e ação (*Act*). Este ciclo também inspirou modelos como CMMI, IDEAL, QIP, dentro outros.

O ciclo começa pelo planejamento, onde são definidos os objetivos de melhoria e os procedimentos necessários para atingir os resultados. Em seguida a ação ou conjunto de ações planejadas é executado, checka-se o que foi feito para verificar se está de acordo com o planejado, constantemente e repetidamente, e por fim, ações de refinamento são realizadas para eliminar ou ao menos mitigar os defeitos encontrados no produto ou na execução.

2.3. Modelo QIP

O modelo QIP - *Quality Improvement Paradigm* foi desenvolvido por Basili (1985) como resultado da aplicação do método científico ao problema de melhoria da qualidade do software. O QIP, inspirado no ciclo de Deming – PDCA, é baseado em uma abordagem cíclica e contínua para melhoria de processos e pode ser detalhado em seis etapas (Basili, 1994): *Caracterizar* o projeto e seu ambiente utilizando modelos, dados, intuição, etc. e estabelecer linhas base com os processos de negócio existentes na organização; *Estabelecer Objetivos* quantificáveis com base na caracterização inicial e nos aspectos de relevância estratégica para a organização; *Escolher o Processo* conforme a caracterização do ambiente e dos objetivos que foram determinados; *Executar* os processos nos projetos, analisar os dados obtidos em cada projeto e fornecer um retorno a respeito dos dados que estão sendo coletados; *Analisar* os dados da informação reunida para avaliar as práticas atuais, determinar os problemas, registrar os achados e realizar recomendações para projetos futuros e, por fim, *Empacotar* as experiências adquiridas em modelos atualizados para torná-las disponíveis para futuros projetos.

2.4. Modelo IDEAL

O modelo IDEAL Mcfeeley (1996) é um modelo de programa organizacional de melhoria de processos de software que serve como guia para iniciar, planejar e implementar ações de melhoria. O modelo IDEAL, desenvolvido pelo SEI (*Software Engineering Institute*), foi concebido como um modelo de ciclo de vida para o melhoramento do processo de software baseado no Capability Maturity Model (CMM), e por essa razão o modelo usa termos de melhoramento de processos (IDEAL, 2004).

O modelo fornece uma abordagem de engenharia disciplinada para o aprimoramento, foca no gerenciamento do programa de aprimoramento, e estabelece a fundação para uma estratégia de melhoramento de longo prazo. O modelo consiste em cinco fases. São elas:

1. *Initiating* (Fase de Iniciação): estabelece um alicerce para um programa de melhoria bem sucedido. Os esforços são identificados, os recursos alocados e a infra-estrutura é definida.
2. *Diagnosing* (Fase de Diagnóstico): determina o estado atual da organização e o estado futuro, aonde ela quer chegar. Diante desses estados são realizadas recomendações para conduzir as atividades subseqüentes.
3. *Establishing* (Fase de Estabelecimento): planeja os detalhes de como a organização alcançará seu objetivo. As prioridades são definidas com base no diagnóstico e uma abordagem é então desenvolvida.
4. *Acting* (Fase de Ação): executa o trabalho de acordo com o planejamento feito nas fases anteriores e testa a solução através da execução de projetos pilotos ou outros mecanismos que possibilitem identificar problemas e refinar a solução.
5. *Learning* (Fase de Aprendizagem): aprender com a experiência e melhorar a habilidade da organização em adotar novas tecnologias no futuro.

2.5. MPS.BR

O projeto MPS.BR – Melhoria do Processo de Software Brasileiro é uma iniciativa brasileira envolvendo universidades, grupos de pesquisa e empresas, sob a coordenação da SOFTEX (Sociedade para Promoção da Excelência do Software Brasileiro). Este projeto visa promover a qualificação de um amplo grupo de empresas compatível com os padrões de qualidade aceitos internacionalmente pela comunidade de software, a custos acessíveis, sendo adequado ao perfil e cultura da grande maioria das empresas brasileiras, (Softex, 2007).

O programa MPS.BR foi baseado nas normas ISO/IEC 12207 (ISO/IEC 12207, 1998), ISO/IEC 15504 (ISO/IEC 15507, 1998), além de ser aderente ao CMMI. Está dividido em três (3) componentes: Modelo de Referência (MR-MPS), Método de Avaliação (MA-MPS) e Modelo de Negócios (MN-MPS), e define sete níveis de maturidade: Parcialmente Gerenciado (G), Gerenciado (F), Parcialmente Definido (E), Largamente Definido (D), Definido (C), Gerenciado Quantitativamente (B) e Em Otimização (A). A divisão em estágios, embora baseada nos níveis de maturidade do CMMI, tem uma graduação diferente, com o objetivo de possibilitar uma implementação e avaliação mais gradual e adequada à realidade das micros, pequenas e médias empresas. Isso permite uma visibilidade dos resultados de melhoria em prazos mais curtos (Softex, 2007).

3. Ferramenta de Gerência de Processos WebAPSEE

Os Ambientes de Desenvolvimento de Software (ADS) desempenham um importante papel no apoio aos engenheiros de software na execução de processos e tem sido bastante utilizados para auxiliar a implementação de melhorias no processo de software pelo fato de integram diversas ferramentas de apoio. O WebAPSEE (WebAPSEE, 2007) é um PSEE (Ambiente de desenvolvimento de Software Centrado em Processo - *Process-Centered Software Engineering Environment*) (Gimenes, 1994), que permite a modelagem e execução de processos de software e tem como diferenciais:

1. Ambiente para modelagem e acompanhamento visual do processo;
2. Mecanismos para auxiliar a alocação de pessoas no processo;
3. Permite a execução flexível de processos, isto é, apóia (entre outras funcionalidades) mudanças dinâmicas no processo, garantindo a consistência do mesmo;
4. Geração automatizada de diversos relatórios que auxiliam na implementação do MR-MPS, como por exemplo: cronograma, relatório de métricas, de custos, recursos, plano da organização, etc.
5. Possui *feedback* direto entre o desenvolvedor e o gerente;
6. Permite a reutilização de processos de software utilizando *templates*, através de três funcionalidades: *Process Instantiation*, *Process Composition* e *Process Distilling*.

Para apoiar a definição e implantação de melhoria do processo de software de acordo com o MR-MPS, o WebAPSEE disponibiliza duas principais ferramentas: o *Manager Console* e a *Task Agenda*.

A imagem localizada ao fundo da figura 1 mostra o *Manager Console* com um processo modelado e sendo executado, através dele o gerente de projetos pode modelar, visualizar e gerenciar a execução de processos, visualizar relatórios, além de gerenciar as informações da organização, como por exemplo recursos humanos, cronograma, artefatos, dentre outras.

Através da Agenda, o desenvolvedor pode acompanhar a execução de suas tarefas, bem como fornecer um *feedback* sobre o andamento dessas tarefas. Além disso, a Agenda permite ao desenvolvedor fazer *download* e *upload* dos artefatos do processo. Através da imagem a frente da figura 1 é possível ver a *Task Agenda* de um desenvolvedor.

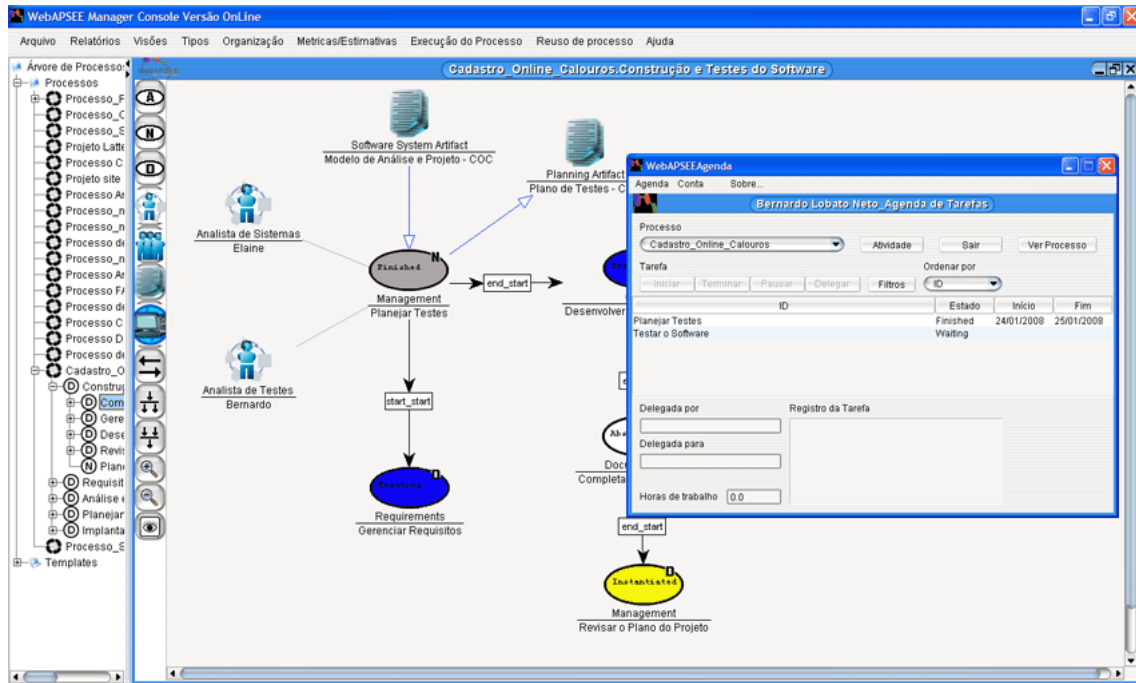


Figura 1. Execução do WebAPSEE

4. Proposta de Metodologia para Implementação do MR-MPS

Para definir a metodologia em questão foi realizada uma pesquisa dentre os principais modelos de melhoria QIP (Basili, 1985), PDCA (Deming, 1990) e IDEAL (Mcfeeley, 1996). Diante das características apresentadas por cada um, o modelo IDEAL foi o selecionado, por estar diretamente relacionado à melhoria do processo de software, por ser baseado no CMM, que por sua vez tem compatibilidade com o MPS.BR e por sugerir o uso de projetos-piloto para avaliar a efetividade das ações em prol da melhoria. Independentemente das características de cada modelo, cabe ressaltar quão desafiador é implementá-los, preservando os valores e características de cada caso. Isso é evidenciado pela necessidade de disponibilidade de recursos (humanos, financeiros e

de tempo), e no levantamento bibliográfico, pela escassez de relatos detalhados sobre o processo de implementação, dado o valor confidencial dos procedimentos e resultados.

Outra dificuldade encontrada em implementações de processos está relacionada à mudança da cultura organizacional. Há uma grande dificuldade em customizar o processo padrão de acordo com as necessidades da organização quando já existe uma cultura não completamente correta sobre os procedimentos de Engenharia de Software (modelagem, testes, documentação, etc.). Sabe-se que mudanças na cultura organizacional principalmente nas atividades tradicionalmente executadas de forma *ad hoc* são difíceis de serem aceitas pelos desenvolvedores. (Rocha et al., 2005).

Para (El-Emam et al., 2001), a falta de aplicação de uma metodologia adequada para orientar a implementação de melhoria pode trazer sucesso limitado em projetos de melhoria de processo. Dessa forma, a necessidade de uma metodologia embasada nas normas e modelos de qualidade utilizados atualmente motivou o desenvolvimento da metodologia proposta neste trabalho.

A metodologia foi definida através da linguagem visual de modelagem de processos do ambiente WebAPSEE. Ela é instanciada a cada novo projeto de implementação de melhoria de processos MPS.BR. A seguir são mostradas as etapas da metodologia proposta e como ela deve ser aplicada na organização.

4.1. Iniciação

Nesta etapa são realizadas reuniões com a alta direção para aprovar o projeto de melhoria de processos da empresa e garantir recursos para iniciar e concluir o projeto. Também são feitas atividades de planejamento, como a definição do escopo do projeto, as estimativas, a infra-estrutura e os custos são planejados, os recursos humanos para implementar a melhoria são definidos dentro da equipe de implementação e a viabilidade do projeto é avaliada.

Algumas palestras e reuniões de motivações são feitas para estimular as mudanças que irão ocorrer evitando assim, uma das dificuldades relatada por Rocha et al. (2005) - a falta de motivação da organização em implantar processo, o que traz resultados poucos satisfatórios, pois as pessoas não se empenham o suficiente para aprender sobre as práticas novas introduzidas pelos processos. Isto ocorre muitas vezes devido ao fato de as pessoas darem menos prioridade às tarefas importantes da implantação do processo, pois não compreendem os potenciais benefícios da implantação do processo.

4.2. Diagnóstico

O objetivo desta fase é avaliar a situação atual da organização, para isso são realizadas entrevistas, são passados questionários aos membros da empresa, os projetos desenvolvidos são analisados e se já existir um processo de desenvolvimento este deve ser analisado. Em organizações com baixa maturidade de capacitação em software, os processos geralmente são informais, existem apenas na cabeça de seus praticantes (Natali, 2006). As informações levantadas são avaliadas e as principais falhas e pontos a serem melhorados devem ser explorados.

Os resultados obtidos são apresentados para a alta gerência da organização e é possível diagnosticar a maturidade da organização e traçar os pontos de melhoria.

4.3. Estabelecimento

Nesta fase as prioridades e estratégias são definidas claramente, tendo como base os resultados da fase de diagnóstico e as metas de negócio da organização. É criado um plano de ação para a melhoria a ser implementada. Esse plano deve conter as ações necessárias para o sucesso da mudança. O projeto de implementação é planejado e os riscos são identificados.

A estratégia de acompanhamento do trabalho de melhoria deve ser definida, como por exemplo, definir a periodicidade das reuniões para análise e discussão dos resultados obtidos em relação às atividades de implementação de melhoria.

Depois de planejadas as ações, é preciso que se definam os recursos humanos necessários para a execução do projeto de implementação. Um ciclo de vida do processo de implementação também é definido.

4.4. Ação

Nesta fase o trabalho contextualizado e planejado nas fases anteriores é implementado, testado e refinado. As atividades previamente planejadas na fase de estabelecimento são cumpridas.

Ainda segundo (Rocha et al., 2005) as maiores dificuldades encontradas na implementação dos processos estão relacionadas às competências da equipe da empresa, daí a necessidade de realizar treinamentos, tais como elaboração de descrições de casos de uso, diagramas de classes e especificações de requisitos, planejamento de projetos, rastreabilidade de requisitos, dentre outros tópicos de engenharia de software. De acordo com as necessidades levantadas na fase de diagnósticos, treinamentos devem ser realizados para nivelar os recursos humanos da organização.

Para Fernandes et al (2007), muitas das dificuldades observadas nas empresas poderiam ser minimizadas com a existência de uma ferramenta para apoiá-los na condução das melhorias, principalmente neste estágio inicial de maturidade. A existência dessa ferramenta iria auxiliar a definição e execução dos processos e traria ao processo de melhoria uma produtividade maior. Na metodologia proposta essa dificuldade é sanada com a implantação e treinamentos na ferramenta WebAPSEE que irá apoiar o processo de melhoria.

Concluída a fase de diagnóstico e estabelecimento e com base nas práticas e sugestões dos principais modelos de qualidade mencionados (ISO/IEC 12207, CMMI, MPS.BR) é possível definir um processo padrão para a organização e aprimorá-lo de acordo com as necessidades da mesma. A existência de um processo padrão permite que a organização tenha um “*modus operandi*” padronizado e reproduzível. Isto facilita a capacitação das pessoas, e torna o funcionamento da organização menos dependente de determinados indivíduos. Para Weber et al. (2005) é importante definir o processo dentro do ambiente em que ele será executado. Para isso, é preciso conhecer bem a organização envolvida (características, tipos de software que são desenvolvidos, paradigmas de desenvolvimento e cultura), de forma a conceber processos que atendam as suas necessidades, suas metas de negócio e contenham aspectos-chaves de qualidade.

O processo padrão deve ainda ser utilizado por todos os projetos da organização, adaptando-o e customizando-o às características dos mesmos.

Para testar se a melhoria trará o resultado esperado pela organização e para cumprir um dos requisitos da avaliação do MA-MPS, são definidos e executados os projetos pilotos, onde, em um contexto mais restrito, a implementação MR-MPS é colocada em prática para avaliação. De acordo com Silva Filho (2006), a execução de projetos pilotos pode servir a vários propósitos, dentre eles: (i) Identificar questões relacionadas ao conhecimento tácito, tornando-o explícito à medida que o piloto é executado, servindo como uma importante fonte de lições aprendidas; (ii) Avaliar a conformidade do processo antes que qualquer esforço significativo de replicação seja realizado; e (iii) Avaliar mudanças significativas ou melhorias inovadoras que não foram experimentadas e possuem alto risco, antes de serem amplamente disseminadas na organização.

A execução dos pilotos é monitorada e controlada e ajustes são feitos a partir disso. São realizadas correções e planejamentos são alterados. Nesse momento é necessário um monitoramento eficiente para detectar problemas e solucioná-los com rapidez para evitar futuras falhas.

4.5. Aprendizagem

Nesta fase as lições aprendidas com a implementação são analisadas para evitar erros parecidos no futuro. O modelo precisa ser executado várias vezes, de modo que o processo evolua dentro da organização. Nesta fase algumas métricas podem ser coletadas. É preciso pensar em ações futuras constantemente para que a melhoria se torne contínua.

Depois de executado, o processo definido é avaliado e institucionalizado na organização. Para isso, deve-se garantir que todos os envolvidos conheçam e utilizem o processo e, sejam coletados dados que ofereçam informações sobre os resultados obtidos.

5. Implementação do Nível G do MR-MPS em uma Empresa Paraense

Para experimentação e validação da metodologia a mesma está sendo aplicada na implantação do nível G do MR-MPS em uma empresa paraense. Esta empresa é uma unidade estratégica, onde a qualidade da informação e a agilidade para responder as demandas dos usuários são fatores críticos de sucesso. O Grupo de Melhoria de Processos de Software da UFPA (Universidade Federal do Pará) foi criado para ajudar a aumentar a maturidade em software da empresa, com possibilidade de efeitos positivos nas outras áreas de atuação da mesma (como suporte técnico, por exemplo). Isso, por si só, já demonstra uma preocupação com a questão da qualidade, que pode trazer retornos em forma de reconhecimento e em forma de aumento da capacidade para atendimento a mais serviços importantes.

O projeto de implementação do nível G do MR-MPS teve início em março de 2007. A empresa conta com 44 servidores sendo que a maioria tem nível médio ou graduação, 6 possuem mestrado e 7 especialização. Desses funcionários, 14 estão afastados de suas atividades por estarem cedidos a outros órgãos ou cursando Doutorado. Devido à baixa quantidade de funcionários frente às demandas que surgem,

a empresa conta com vários bolsistas de graduação para auxiliarem na realização das tarefas.

O projeto de implementação do nível G do MR-MPS do Grupo de Melhoria de Processos de Software da UFPA seguiu as cinco fases propostas pela metodologia apresentada anteriormente para atender os objetivos da implementação.

Na fase de iniciação foi estabelecida uma infra-estrutura preliminar de melhoria. Foram feitas reuniões com o diretor da empresa para garantir recursos e disponibilização de funcionários para realização da melhoria. Algumas palestras sobre os benefícios da implementação de um processo de melhoria foram ministradas para os funcionários e direção e o planejamento da melhoria foi iniciado. Essas reuniões e palestras foram importantes para contornar um dos riscos descritos por Mendes et al. (2007), que é a falta de envolvimento da equipe da empresa. Através das palestras foi possível mostrar à equipe participante do programa Melhoria de Processos de Software (MPS) os benefícios, custos, e riscos do projeto, bem como, mostrar que práticas adequadas de engenharia de software são interessantes para a empresa, mesmo que para isso ela precise abandonar as práticas até então adotadas.

Na fase de diagnóstico foram realizadas entrevistas diretas ou através de questionários com a gerência e com os funcionários da organização. Essas entrevistas apontaram inicialmente, a existência de espaço para melhorias na gerência de projetos e de requisitos, tendo em vista que na empresa não havia uma definição clara sobre funções e papéis, os projetos não estavam sendo gerenciados adequadamente e através do “esforço heróico” algum sucesso era obtido. Tais melhorias são cobertas pelo nível G do MR-MPS (que tem como propósitos a implantação da gerência de projetos e da gerência de requisitos). A organização em questão não possuía um processo de software definido, os processos eram informais, no entanto os pontos fortes e fracos dos mesmos foram analisados e levados em consideração na definição do processo padrão para a organização. Também foram levados em consideração os projetos já finalizados e as expectativas dos envolvidos no programa MPS.

Na fase de estabelecimento, os aspectos que a organização decidiu considerar foram priorizados e as estratégias foram estabelecidas. Durante essa fase foram deixadas bem claras as necessidades e vantagens na definição de um processo padrão, tendo em vista que um processo bem descrito permite entendimento, gerência e melhoria e que organizações maduras conhecem detalhadamente seu processo e o gerenciam através de indicadores inseridos em suas diversas fases. No caso desta empresa, depois de feitas as devidas análises, decidiu-se por definir um processo padrão baseado no modelo cascata. Este processo foi modelado e executado no WebAPSEE (ferramenta escolhida para apoiar o programa MPS), o que o torna um modelo flexível e adaptável, de tal maneira que o processo padrão pode ser instanciado para projetos de diferentes tipos e tamanhos, um exemplo disso é a existência de uma atividade decomposta, na fase de “Construção e Testes de Software” do processo, composta de sub-atividades de implementação para cada Caso de Uso, tendo em vista que essas sub-atividades deverão ser criadas pelo gerente de projetos durante a execução do processo, logo após a fase de planejamento.

O processo foi definido levando em consideração a cultura da organização, além de atender os resultados esperados do Nível G do MR-MPS. O processo padrão definido para a empresa, após alguns refinamentos, possui cinco fases (Planejamento, Análise dos Requisitos, Projeto de Arquitetura do Software, Construção e Testes do Software, Implantação e Encerramento do Projeto).

Nesta fase também foram realizados diversos treinamentos e reuniões a fim de alinhar todos os envolvidos com os objetivos do projeto de melhoria. Os colaboradores foram treinados antes de executar suas atividades e a execução das mesmas foi acompanhada para verificar se estavam sendo feitas de forma correta. Foram oferecidos treinamentos em gerência de requisitos, engenharia de software, gerência de projetos (muitas vezes através de *mentoring*), curso oficial de Introdução ao MPS.BR (em agosto de 2007), curso de especialização em gerência de projetos de software (para 3 colaboradores), além de treinamentos no processo padrão e na ferramenta WebAPSEE.

O preenchimento dos artefatos também foi acompanhado pela equipe de implementação, sendo que para cada artefato foi proposto um *template* que define uma estrutura padrão para os documentos, além de informações importantes que eles precisam ter.

Na fase de ação foram escolhidos dois projetos pilotos para testar o processo, sendo que um deles já foi finalizado, o projeto A (nome fictício). A utilização do WebAPSEE foi de grande importância para o projeto, visto que essa ferramenta permitiu a gerência de projetos com a modelagem e acompanhamento da execução do projeto feitas através do *Manager* e da *Agenda*. Alguns relatórios gerados pelo WebAPSEE, como cronograma, *gantt chart*, plano de recursos, plano de custos, plano de recursos humanos, organograma da empresa, a estrutura analítica do projeto, além do plano de gerência de documentos, ajudaram a automatizar ainda mais o processo. A figura 2 exemplifica um dos relatórios gerados pelo WebAPSEE, o *gantt chart*. Outra grande vantagem da ferramenta é a possibilidade de alteração dinâmica, isto é, durante a execução do processo novas atividades puderam ser adicionadas e/ou excluídas, sendo que para isso existem regras para checar automaticamente a consistência do modelo de processo em relação ao seu estado e de suas atividades, o fluxo de atividades, os estados dos recursos, dentre outras. O gerente de projetos pôde acompanhar o desenvolvimento do projeto em tempo real através do *Manager* e os desenvolvedores receberam automaticamente as suas tarefas e artefatos necessários pela *Task Agenda*.



Figura 2. Gantt chart de um processo executado no WebAPSEE

Outro benefício observado foi a facilidade de acesso aos artefatos e *templates* de artefatos. O processo modelado no WebAPSEE permite que os mesmos possam ser baixados e atualizados diretamente pela *Task Agenda* ou pelo *Manager*. Um agente (pessoa responsável pela atividade) pode facilmente carregar o *template* de um artefato em seu computador, preenchê-lo e postá-lo pela *Task Agenda*, como mostra o item 1 da figura 3. A ferramenta também guarda as versões dos artefatos através da integração com o CVS (*Concurrent Versions System*), facilitando o acesso à versões anteriores.

Cada atividade modelada no processo possui um roteiro, cadastrado pelo gerente de projetos no *Manager*, que inclui toda informação necessária para que um agente possa realizá-la de forma eficaz, como mostra o item 2 da figura 3. O desenvolvedor também pode ver o início e o fim estimados de suas atividades.

Através da *Task Agenda* o agente pode iniciar, pausar e finalizar uma atividade. O WebAPSEE armazena o tempo gasto pela execução da atividade, com isso o gerente verifica a produtividade dos agentes e pode traçar comparativos sobre estimado *versus* realizado. Quando o status de uma atividade muda na *Task Agenda* o mesmo muda automaticamente no *Manager* e a execução do processo pode ser acompanhada visualmente.

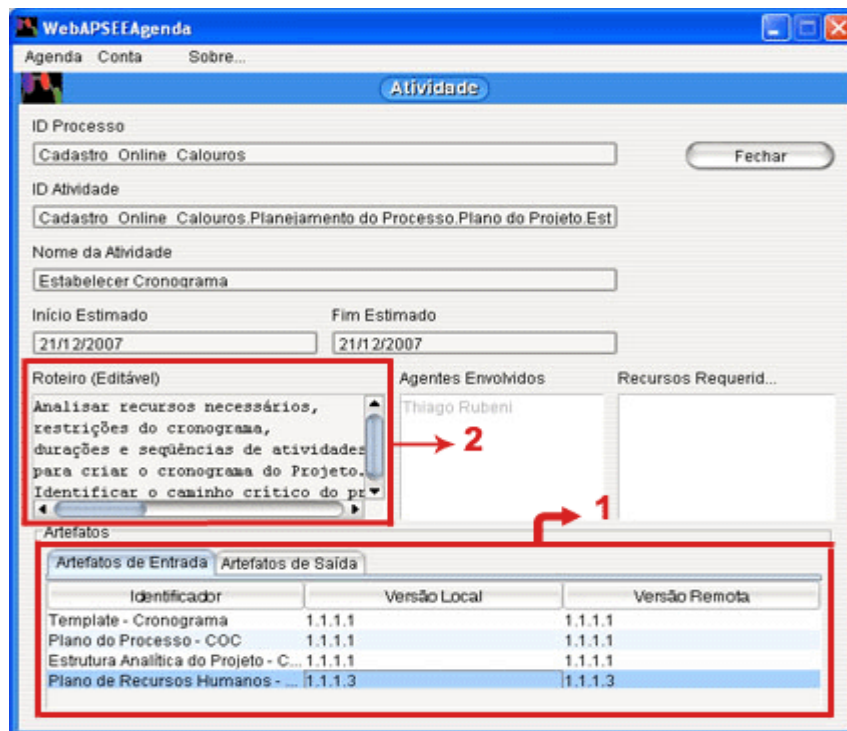


Figura 3. Detalhes da Atividade mostrados pela Agenda

Na fase de aprendizagem ficaram comprovados os benefícios do programa MPS, o projeto A foi muito elogiado pelos seus clientes, alcançando seu objetivo e atendendo as expectativas da patrocinadora do projeto. Durante a execução deste primeiro projeto foram identificadas as seguintes lições aprendidas:

- A presença freqüente da equipe de implementação favorece a disseminação das vantagens e dos benefícios do programa de melhoria na organização.
- Quando os gerentes ou líderes de projetos responsáveis pela execução dos processos na organização possuem conhecimento em engenharia de software, há uma considerável rapidez na implementação da melhoria e uma necessidade bem menor por acompanhamento da equipe de implementação.
- O primeiro projeto a utilizar o processo definido apresenta dificuldades, já que a equipe está se adaptando às novas atividades e a ferramenta de apoio, o que exige uma mudança na maneira como os projetos são executados.
- A obtenção dos comprometimentos ao longo do projeto teve certa resistência, pois a empresa não estava habituada a registrar o comprometimento formal tanto interna como externamente.
- Adotar uma ferramenta de rastreabilidade de requisitos, para diminuir a complexidade de criar uma matriz de rastreabilidade usando planilha.

Os resultados alcançados com os projetos pilotos serão estendidos para outros projetos e o processo será institucionalizado. O próximo passo da empresa é se preparar para a avaliação preliminar do nível G do MR-MPS.

6. Trabalhos Relacionados

Vários relatos têm sido publicados sobre o uso de modelos de qualidade e os resultados que as iniciativas de melhoria baseadas nestes modelos têm obtido em diversas organizações. Fernandes, 2007; Rocha, Montoni, Santos et al. 2005, Macedo et al., 2006 relatam implantações do MPS.BR, tanto em empresas individuais como em organizações cooperadas. Esses trabalhos apontam as necessidades e vantagens da utilização de ferramentas de apoio, e, mostram, de maneira simplificada, a utilização de alguma metodologia. No entanto, nenhum desses trabalhos sugere a execução da metodologia como um processo de melhoria executado em um ambiente centrado em processos.

A metodologia discutida é tratada como um processo que foi modelado e executado utilizando o WebAPSEE. Apoiando a equipe de implementação desde o planejamento até as lições aprendidas, essa metodologia possibilita, além de outras vantagens, a gerência e redução do tempo de implementação e guarda os históricos que poderão ser úteis em outros projetos.

7. Considerações Finais

Neste trabalho foi apresentada uma metodologia de implementação do nível G do MR-MPS, orientada pelo modelo IDEAL, que está sendo utilizada por uma empresa paraense para conseguir a avaliação nível G do MR-MPS. Esta metodologia engloba desde a definição da estratégia de implementação até a avaliação final.

Esta metodologia específica inclui procedimentos, *templates* de documentos, material de treinamento e outros, voltados para a implementação do MR-MPS. Desta forma, espera-se ao final do projeto, ter esta metodologia de implementação de programa de melhoria revisada e totalmente testada, e a empresa com o nível G do MR-MPS implementado e apta para avaliação formal.

Para colher os benefícios esperados com o programa MPS, deve haver a conscientização da organização em adotar a melhoria de processos de software como uma estratégia, treinando os seus gerentes e colaboradores e criando um ambiente favorável. A organização está bastante motivada, principalmente depois dos benefícios observados com a execução do projeto A.

Ficou evidenciado que o apoio do WebAPSEE facilitou o treinamento, desenvolvimento, a definição, execução e está sendo de grande importância na institucionalização dos processos, ajudando a reduzir o tempo de implementação dos mesmos. A ferramenta ajudou a equipe de implementação tanto na execução do processo de implementação como na execução da metodologia e os benefícios obtidos pela organização cliente com o uso da ferramenta também puderam ser observados pela equipe de implementação, já que a metodologia é modelada e executada em tal ferramenta.

Referências

Basili, V.R. (1985) Quantitative Evaluation of Software Engineering Methodology, Proceedings of First Pan Pacific Computer Conference, Melbourne, Australia, September.

- Basili, V. R.; Caldeira, G.; Rombach, H. D. (1994) The Experience Factory, In: MARCINIAK, John J. (Ed.). Encyclopedia of Software Engineering. New York: John Wiley & Sons, pp. 469-476.
- Chrissis, Mary Beth et al., (2003) CMMI: Guidelines for Process Integration and Product Improvement. Boston, Addison Wesley.
- CMU/SEI, (2002), Capability Maturity Model Integration (CMMI), Version 1.1, Pittsburgh, Software Engineering Institute, Carnegie Mellon University. URL: <http://www.sei.cmu.edu>. Acesso em janeiro de 2008.
- Deming, W. E. (1990) Qualidade: A Revolução da Administração. Rio de Janeiro: Marques Saraiva.
- El-Emam, K., Goldenson, D., McCurley, J. and Herbsleb, J. (2001) Modelling the Likelihood of Software Process Improvement: An Exploratory Study," Empirical Software Engineering, vol. 6, pp. 207-229.
- Fernandes, P.G.; Oliveira, J. L.; Mendes, F.F.; Souza, A.S. (2007) Resultados de Implementação Cooperada MPS.BR. ProQualiti - Qualidade na Produção de Software, pages 11-17.
- Filho, R.C.S. (2006) Uma Abordagem para a Avaliação de Propostas de Melhoria em Processos de Software. Dissertação de Mestrado. Rio de Janeiro – Brasil.
- Fuggetta, (2000) A. Software Process: A Roadmap. The Future of Software Engineering.
- Gimenes, I.M. (1994) Uma Introdução ao Processo de Engenharia de Software: Ambientes e Formalismos. Trabalho apresentado na Jornada de Atualização em Informática, Caxambu - MG: SBC.
- IDEAL, (2004) The IDEALSM Model. Software Engineering Institute (SEI). URL: <http://www.sei.cmu.edu/ideal>. Acesso em maio de 2007.
- ISO/IEC 12207, (1998) Tecnologia de Informação - Processos de ciclo de vida de Software, ABNT - ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, Rio de Janeiro: ABNT.
- ISO/IEC TR 15504, (1998) Information technology – software process assessment. International Organization for Standardization.
- Macedo, C. C.; Lima, S. H.; Rocha, A. R. et al. (2006) Implantação de Melhoria de Processo de Software no Tribunal Superior Eleitoral. V Simpósio Brasileiro de Qualidade de Software (SBQS'06). Vila Velha, Brasil.
- Mcfeeley, B. (1996) IDEALSM: A User's Guide for Software Process Improvement. Pittsburgh, Software Engineering Institute.
- Mendes, F. F.; Oliveira, J. L.; Fernandes, P. G.; Souza, A. S. (2007) Análise de Riscos na Implementação de Melhorias de Processo de Software. ProQualiti – Qualidade na Produção de Software, pages 25 – 31, Recife, Brasil.
- Natali, A. (2006) Engenharia de Software: Introdução e uma Visão do Processo de Software. Apostila de curso.

- Rocha, A. R. et al. (2005) Fatores de Sucesso e Dificuldades na Implementação de Processos de Software Utilizando o MR-MPS e o CMMI, In: I Workshop de Implementadores MPS.BR, Brasília, Brasil.
- Rocha, A. R.; Montoni, M.; Santos, G. et al. (2005) Estação TABA: Uma Infraestrutura para Implantação do Modelo de Referência para Melhoria de Processo de Software. IV Simpósio Brasileiro de Qualidade de Software (SBQS'05). Porto Alegre, Brasil.
- Silva Filho R. C.; Rocha, A. R.; Travassos, G. H. (2006) O Uso de Projetos-Piloto para Avaliação da Efetividade da Melhoria de Processos. V Simpósio Brasileiro de Qualidade de Software (SBQS'06). Vila Velha, Brasil.
- Softex (2007) – Sociedade para Promoção da Excelência do Software Brasileiro, MPS.BR - Melhoria de Processo do Software Brasileiro, Guia Geral (v. 1.2), URL: http://www.softex.br/mpsbr/_guias/MPS.BR_Guia_Geral_V1.2.pdf. Acesso em agosto de 2007.
- Softex A (2007) – Sociedade para Promoção da Excelência do Software Brasileiro, MPS.BR - Melhoria de Processo do Software Brasileiro, Guia Geral (v. 1.1), URL: http://www.softex.br/mpsbr/_guias/MPS.BR_Guia_de_Implementacao_Parte_1_V1.1.pdf. Acesso em agosto de 2007.
- WebAPSEE (2007). Documento de Referência da Ferramenta – Versão 1.0. URL: <http://www3.ufpa.br/webapsee/images/documentacao/doc%20de%20referencia.pdf>. Acesso em julho 2007.
- Weber, S.; Hauck, J.C.R.; Von Wangenheim, C.G. (2005) Estabelecendo Processos de Software em Micro e Pequenas Empresas. IV Simpósio Brasileiro de Qualidade de Software (SBQS'05). Porto Alegre, Brasil.