

Uma Arquitetura de Processos para SW-CMM Nível 3 Baseada no RUP

Carlo Giovano Pires, Fabiana Marinho, Gabriela Telles, Márcia Sampaio
Instituto Atlântico, Rua Chico Lemos, 946, 60822-780, Fortaleza - Ceará
{cgiovano, fabiana, gabi, marcia}@atlantico.com.br

Resumo

No SW-CMM nível 3 o conceito de processo é fortemente introduzido na organização. Durante a implantação do SW-CMM nível 3, a arquitetura de processos que servirá como base para a estruturação das práticas de todas as áreas chave deve ser definida. Essa arquitetura é fundamental para a organização dos processos e para o sucesso do projeto de melhoria. No entanto, a maioria das organizações utiliza uma descrição proprietária informal na definição de seus processos, dificultando o entendimento e implementação dos mesmos. Esse trabalho mostra uma experiência prática da utilização da estrutura do RUP e UML para definição da arquitetura de processos SW-CMM.

Palavras-chave: RUP, CMM, Processos, Qualidade de Software

Abstract

In the SW-CMM level 3 the process concept is strongly introduced in the organization. During the institutionalization of the SW-CMM level 3, the architecture of processes that will serve as the foundation of all the key process areas definition. This architecture is essential for the organization of the processes and the success of the software improvement project. However, the majority of the organizations use a proprietary informal description in the definition of its processes, making it difficult the learning and implementation of them. This work shows a practical experience that uses the structure of RUP and UML for defining the architecture SW-CMM based processes.

Key-words: RUP, CMM, Process, Software Quality

1 Introdução

A preocupação com a qualidade do processo de software está se tornando tão essencial para as organizações quanto a qualidade dos produtos entregues aos clientes. Como a qualidade dos produtos de software está diretamente relacionada com a qualidade do seu processo de desenvolvimento, muitas empresas passaram a investir na melhoria de seus processos.

O Capability Maturity Model for Software (SW-CMM) é um modelo desenvolvido pelo Software Engineering Institute (SEI) que busca orientar as organizações de software na implementação de melhorias contínuas em seu processo de desenvolvimento.

Este artigo apresenta a experiência do Instituto Atlântico (IA), uma organização avaliada como SW-CMM Nível 2, na definição e documentação dos processos de software do SW-CMM nível 3. O Instituto Atlântico é uma instituição de pesquisa e desenvolvimento localizada em Fortaleza, Ceará, fundada em novembro de 2001 por iniciativa do Centro de Pesquisa e Desenvolvimento em Telecomunicações (CPqD). O IA desenvolve software em diversas áreas tecnológicas, posicionando-se como fonte inovadora de conhecimento e de geração de resultados.

Os processos do IA foram definidos a partir do mapeamento da estrutura do SW-CMM na arquitetura de processos disponibilizada pelo Rational Unified Process (RUP). O RUP foi utilizado como base para a documentação dos processos por se tratar de um de *framework* de processo adaptável que abrange as melhores práticas do desenvolvimento de software do mercado.

Este trabalho está organizado em seis seções. Na seção 2, descrevemos a estrutura da arquitetura de processos do RUP. Na seção 3, apresentamos uma visão geral da estrutura do

SW-CMM. Detalhamos o nível 3 de maturidade do SW-CMM na seção 4. Na seção 5, apresentamos o mapeamento realizado na documentação dos processos do IA. Na seção 6, apresentamos um estudo de caso do mapeamento proposto e na seção 7, apresentamos as conclusões do nosso trabalho.

2 Estrutura da arquitetura de processos do RUP

O RUP é um processo iterativo e incremental que provê uma abordagem disciplinada para o desenvolvimento de software [1]. Na engenharia de software, um processo é um conjunto de passos parcialmente ordenados com a intenção de construir um produto de software de qualidade, capaz de atender às necessidades e exigências do usuário final de acordo com planejamento e orçamento previstos [2].

No RUP, os processos são organizados em um conjunto de disciplinas para posteriormente definirem os fluxos de trabalho e outros elementos do processo [3].

Uma disciplina é uma coleção de atividades relacionadas que estão ligadas a uma área de interesse dentro do processo [4]. As disciplinas do RUP são descritas por meio de fluxos de trabalho que mostram uma seqüência significativa de grupos de atividades que produz um determinado resultado. Cada grupo de atividades é descrito por um diagrama de detalhamento de fluxo de trabalho. Esses diagramas mostram todas as atividades do grupo, os papéis envolvidos e os artefatos de entrada e saída. Um artefato é um produto de trabalho do processo. Os papéis usam os artefatos na execução das atividades. Outros elementos auxiliares ao processo também são descritos nos formatos de *templates*, relatórios, orientações de trabalho, pontos de verificação, diretriz de artefato, mentor de ferramentas e conceitos. Entre esses, os *templates* e orientações de trabalho desempenham funções de destaque em uma arquitetura de processos. O primeiro tipo representa modelos pré-formatados que auxiliam na produção de artefatos, o segundo apresenta técnicas importantes na execução de atividades e produção de artefatos.

3 Uma visão geral do SW-CMM

O SW-CMM está organizado em cinco níveis de maturidade. Cada um destes níveis apresenta fundamentos sucessivos para a melhoria contínua do processo. A Tabela 1 apresenta uma visão geral destes níveis de maturidade.

Tabela 1 - Níveis de maturidade do modelo SW-CMM

| Níveis de Maturidade | | | | | Característica Principal | |
|----------------------|----|----|----|----|--------------------------|--------------------------------------|
| | | | | ↗5 | Otimizado | Processo aperfeiçoado continuamente |
| | | | ↗4 | | Gerenciado | Processo previsível e controlado |
| | | ↗3 | | | Definido | Processo consistente e padronizado |
| | ↗2 | | | | Repetível | Processo disciplinado |
| ↗1 | | | | | Inicial | Processo imprevisível e sem controle |

O nível 1 (inicial) é caracterizado como caótico. O sucesso e a qualidade do produto dependem de esforços individuais sem qualquer nível de gerenciamento ou controle definidos explicitamente. No nível 2 (repetível) processos básicos de gerenciamento de projeto são estabelecidos e permitem acompanhar o custo, cronograma e funcionalidade. É possível repetir procedimentos em processos similares. O nível 3 (definido) é caracterizado principalmente pela existência de um processo de engenharia de software bem definido, documentado e padronizado para a organização. Todas as atividades e artefatos do processo de desenvolvimento de software são compreendidos e documentados. No nível 4 (gerenciado)

são coletadas medidas detalhadas da qualidade do processo e do produto. Tanto o processo de software quanto os produtos, são quantitativamente compreendidos e controlados. No nível 5 (otimizado) há uma melhoria contínua dos processos. Medir e avaliar continuamente propicia optar com segurança, por novas tecnologias, técnicas, políticas de treinamento e qualquer outra atividade relevante do processo que resulte em um produto de melhor qualidade.

Os níveis de maturidade do SW-CMM são compostos de áreas chaves de processo (*Key Process Area* - KPAs) que indicam o foco a ser dado pela organização para melhorar seu processo de software. A Figura 1 apresenta a estrutura interna dos níveis de maturidade do SW-CMM.

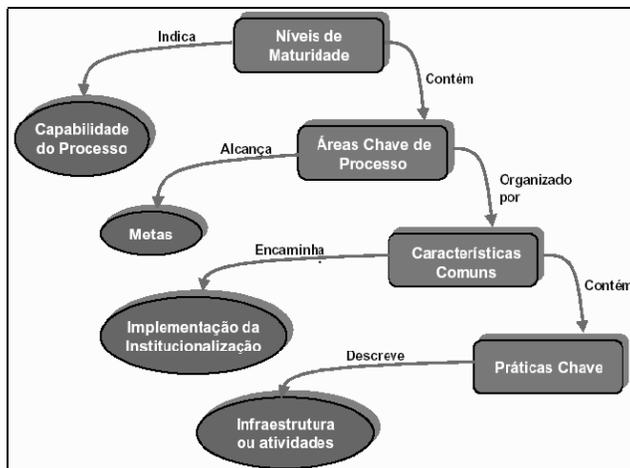


Figura 1 - Estrutura dos níveis de maturidade do SW-CMM

As KPAs são organizadas em cinco seções chamadas características comuns. Cada característica comum define, em termos de práticas chaves, as atividades e infra-estrutura necessárias para a efetiva implementação e institucionalização da KPA. Subpráticas detalhadas são fornecidas dentro das práticas chave como um guia de interpretação da implementação adequada da prática chave. Existem cinco tipos de características comuns: Compromisso para Executar, Habilidade para Executar, Atividade Executada, Medição e Análise, e Verificações da Implementação.

As Atividades Executadas descrevem o que deve ser implementado para estabelecer o processo. As demais características comuns apenas dão suporte à institucionalização das práticas descritas nas Atividades Executadas.

4 O Nível 3 de maturidade do SW-CMM

No SW-CMM nível 3, os processos de software para as atividades gerenciais e atividades de engenharia são documentados, padronizados e integrados em um processo de software padrão da organização, podendo ser customizado para cada projeto [6]. Esse nível também exige um grupo responsável por estudar e implementar processos cada vez mais otimizados, denominado Software Engineering Process Group (SEPG). Um programa de treinamento organizacional é implementado para garantir que a equipe técnica e os gerentes tenham o conhecimento e as habilidades necessárias para desempenharem seus papéis.

Este nível possui as seguintes KPAs: Foco no Processo da Organização, Definição do Processo da Organização, Programa de Treinamento, Gerenciamento Integrado de Software, Engenharia de Produto de Software, Coordenação Intergrupos e Revisão em Pares.

As KPAs de Foco no Processo da Organização e Definição do Processo da Organização definem a estruturação e organização da arquitetura de processos.

O objetivo da KPA de Foco no Processo da Organização é estabelecer a responsabilidade organizacional para as atividades do processo de software que melhoram, como um todo, a capacitação do processo de software da organização. Envolve criar e manter um entendimento dos processos de software dos projetos da organização, bem como coordenar as atividades para avaliar, desenvolver, manter e melhorar esses processos [6]. Metas definidas para esta KPA são:

- as atividades de desenvolvimento e melhoria de processo de software são coordenadas em toda a organização;
- os pontos fortes e as oportunidades de melhoria dos processos de software utilizados são identificados com relação a um processo padrão;
- as atividades de desenvolvimento e melhoria de processo no nível organizacional são planejadas.

O objetivo da KPA Definição do Processo da Organização é desenvolver e manter um conjunto de ativos de processo de software que sejam úteis e melhorem o desempenho dos processos utilizados nos diversos projetos. Visa também estabelecer uma base sólida capaz de gerar benefícios acumulativos e de longo prazo para toda a organização [6]. Metas definidas para esta KPA são:

- um processo de software padrão para a organização é elaborado e mantido;
- as informações, relativas ao uso do processo padrão de software da organização pelos projetos de software, são coletadas, revisadas e disponibilizadas.

5 Utilizando a estrutura do RUP para mapear e definir processos de software

Um processo de software bem definido deve descrever a seqüência de atividades a ser executada e mostrar os papéis e artefatos envolvidos na execução dessas atividades. A maioria das organizações utiliza uma descrição proprietária informal na definição de seus processos, dificultando o entendimento e implementação dos mesmos.

Nesta seção, relatamos a experiência do SEPG do IA, na definição e documentação dos processos do SW-CMM nível 3. Os processos do IA foram definidos baseados na arquitetura de processos disponibilizada pelo RUP. O RUP foi escolhido por fornecer uma notação formal para a definição de processos de software, uma abordagem disciplinada para atribuir tarefas e responsabilidades e utilizar UML. Além disso, é amplamente utilizado e difundido, facilitando o entendimento da estrutura e notações do processo. A Tabela 2 exhibe o mapeamento da estrutura do SW-CMM para a arquitetura de processos do RUP.

Tabela 2 - Mapeamento dos processos do IA

| Conceito do SW-CMM | Conceito do RUP | Processos do IA |
|--------------------|------------------------------|--|
| KPA | Disciplina/Fluxo de Trabalho | Cada KPA foi mapeada em um ou mais processos institucionais do IA, variando de acordo com as práticas cobertas pela KPA. Existem situações em que uma KPA é mapeada em mais de um processo institucional, como é o caso da KPA de Engenharia de Produto de Software (Software Product Engineering – SPE). Esta KPA foi mapeada em seis processos institucionais: Especificação de Requisitos, Análise e Projeto, Implementação, Testes de Software, Homologação/Validação, Reuso e Componentização. Cada processo é descrito como uma <i>Disciplina</i> do RUP a partir do fluxo de trabalho e representado graficamente por diagramas de atividade. |

| | | |
|-------------------------------------|-----------------------------------|--|
| Atividade Executada / Subprática | Detalhamento do Fluxo de Trabalho | Cada Atividade Executada ou Subprática é descrita como um Detalhamento de Fluxo de Trabalho com seus papéis, atividades e artefatos. |
| Subpráticas, papéis e procedimentos | Detalhamento do Fluxo de Trabalho | As subpráticas e papéis que compõem as Atividades Executadas foram mapeados respectivamente nas atividades que compõem um Detalhamento de Fluxo de Trabalho e nos papéis associados. Os procedimentos são montados como os passos e técnicas do RUP. |

6 Estudo de caso

Na Tabela 3, apresentamos uma versão simplificada da definição do processo de revisão em pares do IA. Este processo foi mapeado a partir da KPA de Revisão em Pares.

Tabela 3: Mapeamento do processo de revisão em pares

| Prática Chave do SW-CMM | Descrição da Prática Chave do SW-CMM | Grupo de Atividades RUP | Papel | Atividade | Artefato |
|-------------------------|--|--|---|--|--|
| Hab1 | Recursos e fundos adequados são providos para executar revisões em pares em cada produto de trabalho de software a ser revisado | | <ul style="list-style-type: none"> – Líder de revisão – Equipe de revisão – Responsável pelo documento | | <ul style="list-style-type: none"> – Checklists de revisão |
| Hab2 | Líderes de revisão em pares recebem treinamento requerido em como liderar revisões em pares | | <ul style="list-style-type: none"> – Líder de revisão | | |
| Hab3 | Revisores que participam das revisões em pares recebem treinamento requerido nos objetivos, princípios e métodos de revisão em pares | | <ul style="list-style-type: none"> – Equipe de revisão – Equipe de projeto | | |
| Ativ1 | Revisões em pares são planejadas e os planos são documentados | <ul style="list-style-type: none"> – Planejar atividades de revisão em pares | <ul style="list-style-type: none"> – Coordenador | <ul style="list-style-type: none"> – Definir documentos para revisão – Elaborar cronograma de revisões | <ul style="list-style-type: none"> – Plano de desenvolvimento – Cronograma detalhado |
| Ativ2 | Revisões em pares são executadas de acordo com um procedimento documentado | <ul style="list-style-type: none"> – Executar revisões em pares – Acompanhar resolução de defeitos | <ul style="list-style-type: none"> – Líder de revisão | | <ul style="list-style-type: none"> – Checklists de revisão – Documentação do projeto – Ferramenta de registro de defeitos |
| Ativ3 | Dados da condução e resultados das revisões em pares são armazenados | <ul style="list-style-type: none"> Executar revisões em pares | | | |

O fluxo de trabalho da KPA de revisão em pares é mostrado na Figura 2. Os grupos de atividades mostrados no fluxo de trabalho foram detalhados nos diagramas de detalhamento de fluxo de trabalho (veja Figura 3, Figura 4 e Figura 5).

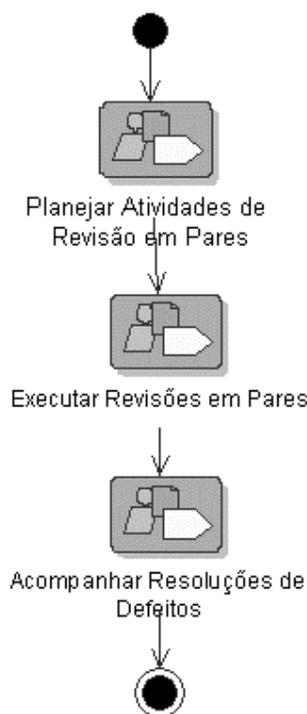


Figura 2 -Fluxo de trabalho do processo institucional de revisão em pares

Para cada diagrama de detalhamento de fluxo de trabalho, foram descritos os papéis responsáveis e envolvidos na execução e os artefatos de entrada e saída. Os papéis responsáveis pela execução das atividades foram mostrados dentro do grupo de atividades, representado pela caixa que envolve as atividades. Uma seta da atividade para o artefato representa que a atividade modifica ou cria o artefato, ou seja, o artefato é uma saída da atividade. O sentido inverso indica que o artefato serve de entrada para a atividade. Os papéis envolvidos ou interessados na execução foram mostrados fora do grupo.

O diagrama de detalhamento *Planejar atividades de revisão em pares* é mostrado na Figura 3. A atividade *Definir documentos para revisão* foi mapeada a partir da subprática 1 da Atividade 1 (*Identificar os produtos de trabalho de software que serão submetidos à revisão em pares*). A atividade *Elaborar cronograma de revisões* foi mapeada a partir da subprática 2 da Atividade 1 (*Especificar o cronograma das revisões em pares*). As subpráticas 1 e 2 mencionam a necessidade de elaborar um plano e um cronograma para as atividades de revisão em pares. O artefato *Plano de desenvolvimento*, foi utilizado para documentar os produtos de trabalho que serão revisados e o artefato *Cronograma detalhado* contém o planejamento destas revisões.

O diagrama de detalhamento *Executar revisão em pares* é mostrado na Figura 4. A atividade *Executar revisões* foi mapeada a partir das subpráticas 1 (*Revisões em pares são planejadas e lideradas por líderes de revisão treinados*), 2 (*Materiais de revisão são distribuídos para os revisores antecipadamente para que eles possam se preparar adequadamente para a revisão em pares*), 3 (*Revisores possuem papéis atribuídos nas revisões em pares*), 4 (*Critérios de conclusão para as revisões em pares são especificados e reforçados*) e 5 (*Checklists são usados para identificar critérios para revisão do produto de trabalho de software de forma consistente*) da Atividade 2. De acordo com estas subpráticas, as revisões devem ser executadas pela equipe de revisão, utilizando checklists e baseadas no planejamento contido no plano e no cronograma de revisões. A atividade *Executar resolução de defeitos* foi mapeada a partir da subprática 7 da Atividade 2.

A atividade *Realizar reunião de resultados* foi estabelecida a partir de uma prática institucional do IA. Foi identificada a necessidade de realizar uma reunião para discutir os problemas encontrados durante a revisão. Neste caso, não existe o mapeamento para nenhuma prática da KPA.

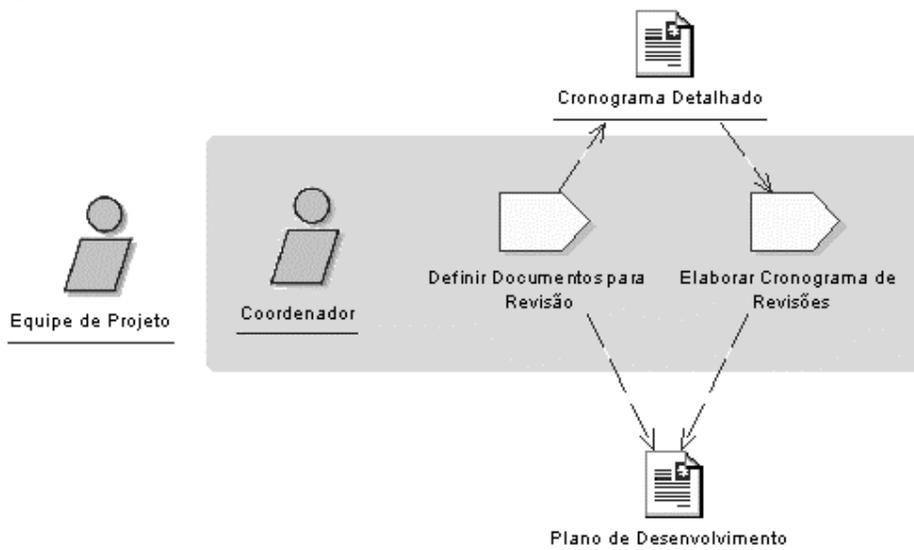


Figura 3 - Detalhamento do grupo planejar atividades de revisão em pares

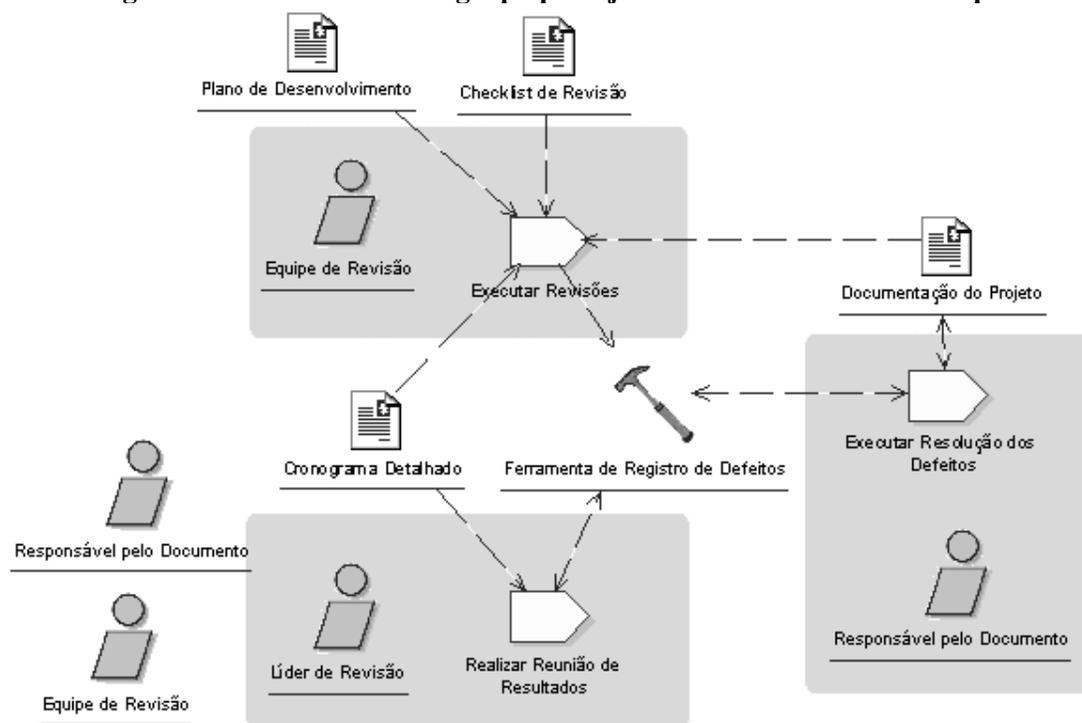


Figura 4 - Detalhamento do grupo executar revisão em pares

O diagrama de detalhamento *Acompanhar resolução de defeitos* é detalhado na Figura 5. A atividade *Acompanhar resolução de defeitos* foi mapeada a partir da subprática 6 da Atividade 2 (*Ações identificadas nas revisões em pares são acompanhadas até serem*

resolvidas) . O *Líder de revisão* acompanha a resolução dos defeitos a partir da *Ferramenta de registro de defeitos*.

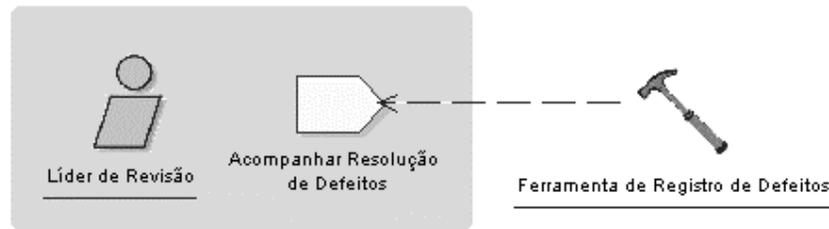


Figura 5 - Detalhamento do grupo acompanhar resolução de defeitos

7 Conclusão

Esse artigo apresentou a experiência do SEPG do Instituto Atlântico na definição da arquitetura de processos no contexto do projeto de melhoria do SW-CMM Nível 3. O RUP e a UML foram adotados como base para a representação e estruturação dos processos. Essa abordagem apresentou os seguintes benefícios:

- representação formal dos processos com base em UML;
- reutilização de estrutura de processos comprovada;
- fácil leitura e entendimento dos processos, atividades, papéis e artefatos pelos colaboradores existentes e pelos colaboradores novatos. Os colaboradores antigos já utilizavam fortemente o RUP e UML, colaboradores contratados como recém formados já tinham conhecimento acadêmico de RUP e UML;
- reutilização de processos já existentes no próprio RUP;
- uniformidade de estrutura e representação entre os processos (disciplinas), já existentes e aproveitadas do próprio RUP e os processos específicos do SW-CMM nível 3.

A proposta deste trabalho não pretende ser uma metodologia formal para a definição e documentação de processos, mas oferecer diretrizes gerais para auxiliar os engenheiros de software na estruturação de uma arquitetura de processos para o SW-CMM nível 3 utilizando o RUP e a UML.

Referências

- [1] Pollice, Gary. Using the Rational Unified Process for Small Projects: Expanding Upon extreme Programming. Rational Software White Paper.
- [2] Sommerville, Ian. Software Engineering, 6th Edition, Addison-Wesley Publishers Ltd., 2001. ISBN 0-201-39815-X.
- [3] Rational Unified Process Tutorial. Versão 2002 05 00.
- [4] Kruchten, Philippe. Rational Unified Process made easy: A practioner´s guide to the RUP, Addison-Wesley Publishers Ltd. 2003.
- [5] Rumbaugh, J., Booch, G., Jacobson, I. UML Guia do Usuário. Editora Campus. 2000.
- [6] Paulk, M. C. et al. Capability Maturity Model for Software, Version 1.1, Technical Report SEI-CMU-93-TR-24, Software Engineering Institute, 1993.