

Uso do Scrum na Gestão do Programa de Melhoria de Processos em uma Organização CMMI 3

Teresa Maria de Medeiros Maciel¹, Ana Sofia Cysneiros Marçal², Felipe Santana Furtado Soares^{1,3}, Elizabeth Sucupira Furtado²

¹ C.E.S.A.R – Centro de Estudos e Sistemas Avançados do Recife
Rua Bione, nº 220, Cais do Apolo 50.0303-90, Recife - PE, Brasil

² Universidade de Fortaleza – Mestrado em Informática Aplicada
Av. Washington Soares 1321, 60811-341, Fortaleza - CE, Brasil

³ Centro de Informática da Universidade Federal de Pernambuco (UFPE)
Caixa Postal 7851, Cidade Universitária – 50.732-970 – Recife – PE – Brasil

{teresa, felipe.furtado}@cesar.org.br, anasofia.marc@hotmai.com, elizabet@unifor.br

Abstract. In order to meet better results in terms of quality, time and cost, software organizations have invested efforts in process improvement programs, which need to be planned and effectively managed. In project management context, agile methodologies have been adopted by software organizations. This work describes the experience with the use of Scrum, an agile methodology focused on project management, to support the management of the process improvement program in an organization CMMI level 3. Therefore, this report presents a Scrum customization, made to adequate it to improvement programs, highlighting most important lessons learned achieved in this experience.

Keywords: Scrum, CMMI, melhoria de processos, SEPG.

1 Introdução

O C.E.S.A.R, um instituto de inovação em tecnologia da informação e comunicação atestado como CMMI nível 3 em 2007, possui um programa de melhoria conduzido por um grupo de melhoria, o SEPG (do inglês, *Software Engineering Process Group*), gerenciado segundo o processo organizacional de gestão de processos.

Considerando a experiência positiva que vem tendo no uso do Scrum como base para a gerência de projetos de desenvolvimento de software, o C.E.S.A.R estendeu o seu uso também para a gerência do programa de melhoria de processos de software. Este artigo relata esta experiência, destacando customizações realizadas sobre o Scrum e lições aprendidas, estando organizado da seguinte forma: a Seção 2 apresenta uma visão geral sobre a gestão de processos no C.E.S.A.R. A Seção 3 introduz o Scrum. A Seção 4 descreve a customização do Scrum realizada para o programa de

melhoria de processos do C.E.S.A.R. A Seção 5 apresenta lições aprendidas. A Seção 6 compreende as conclusões do trabalho.

2 Gestão de Processos no C.E.S.A.R

A gestão de processos de software tem representado uma abordagem recomendada por vários modelos de qualidade. O CMMI [8], por exemplo, define duas áreas de processo focadas na definição, gestão e melhoria do processo padrão de software, ressaltando a estruturação de uma gestão efetiva de processos. A ISO/IEC 15504, norma gerada como produto do projeto SPICE, também segue a mesma linha do CMMI [5]. O modelo brasileiro MPS BR [9], por sua vez, apresenta um forte alinhamento aos modelos CMMI e ISO/IEC 15504, e também recomenda a existência de uma gestão de processo e de grupos de melhoria.

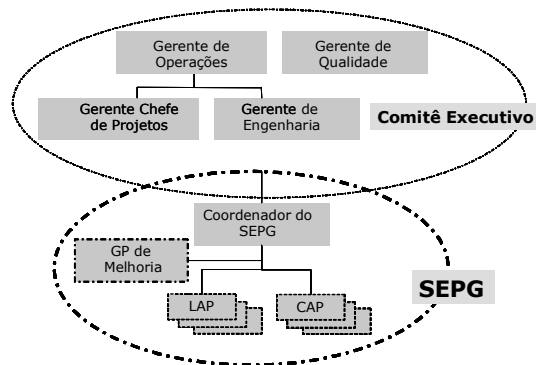


Fig. 1. Estrutura Organizacional do SEPG

O Aderente ao CMMI nível 3, a gestão de processos de software no C.E.S.A.R é conduzida por um grupo de melhoria, o SEPG, o qual é um grupo composto por uma linha de praticantes com perfis variados e representa o ponto focal para a melhoria do processo da organização [3]. O SEPG possui uma estrutura organizacional (Fig. 1) conduzida por um Comitê Executivo que é representado pelas áreas de maior envolvimento com o desenvolvimento de software e que garante o alinhamento do processo aos objetivos estratégicos da organização.

O Coordenador do SEPG é responsável pelo planejamento e acompanhamento das atividades de melhoria dos processos de software. A responsabilidade pela definição, implementação e institucionalização do processo fica a cargo dos *Líderes de Área de Processo* (LAPs), os quais são especialistas em modelagem e melhoria de processos e dos *Consultores de Área de Processo* (CAPs), especialistas atuantes em projetos que prestam consultoria na definição de processos específicos.

O C.E.S.A.R conta com um processo de desenvolvimento de software, ProSCes [6], publicado na Intranet da empresa com acesso a todos os colaboradores, o qual contempla um sub-processo específico para gerência de processos. A melhoria do

ProSCes é gerenciada da mesma forma que um projeto de software, onde o Comitê Executivo é o cliente e os colaboradores da empresa são os usuários. O coordenador do SEPG atua junto ao Comitê Executivo na definição de um plano anual de melhoria, o qual define recursos, prioridades, avaliações, capacitação, responsabilidades, entre outros aspectos.

3. Visão Geral do Scrum

Alinhado aos princípios ágeis, o Scrum foi criado em 1996 por Ken Schwaber e Jeff Sutherland, como um método que aceita que o desenvolvimento de software é imprevisível e formaliza a abstração, sendo aplicável a ambientes voláteis [1].

O Scrum se destaca dos demais métodos ágeis pela ênfase dada ao gerenciamento do projeto. Há atividades de monitoramento e feedback, em geral através de reuniões rápidas e diárias com toda a equipe, visando à identificação e correção de deficiências e/ou impedimentos no desenvolvimento [7].

O Scrum assume a premissa de que o desenvolvimento de software é muito complexo para ser planejado totalmente em seu início [7]. Neste sentido, ele se utiliza de um controle de processo empírico para garantir visibilidade, inspeção e adaptação. O método baseia-se ainda em princípios como: equipes pequenas de, no máximo, 7 pessoas; requisitos pouco estáveis ou desconhecidos; iterações curtas. Ele divide o desenvolvimento em períodos de no máximo 4 semanas, denominados *sprints*.

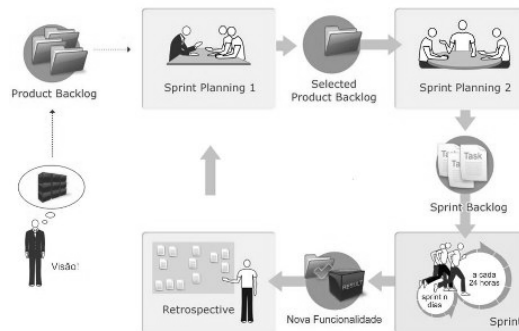


Fig. 2. Visão geral do processo do Scrum [4].

O fluxo de desenvolvimento do Scrum é mostrado na Fig. 2. Segundo Schwaber [7], um projeto no Scrum se inicia com uma visão do produto que será desenvolvido. A visão contém a lista das características do produto estabelecidas pelo cliente, além de algumas premissas e restrições. Em seguida, o *Product Backlog* é criado contendo a lista de todos os requisitos conhecidos. O *Product Backlog* é então priorizado e dividido em releases.

Cada *Sprint* se inicia com uma reunião de planejamento (*Sprint Planning Meeting*), na qual o *Product Owner* e o Time decidem em conjunto o que deverá ser implementado (*Selected Product Backlog*). A reunião é dividida em duas partes. Na

primeira parte, o *Product Owner* apresenta os requisitos de maior valor e prioriza aqueles que devem ser implementados. O Time então define colaborativamente o que poderá entrar no desenvolvimento da próxima *Sprint*, considerando sua capacidade de produção. Na segunda parte, o time planeja seu trabalho definindo o *Sprint Backlog*, composto pelas tarefas necessárias para implementar os requisitos selecionados no *Product Backlog*. A lista de tarefas pode ser modificada ao longo da *Sprint* pelo Time e duração das tarefas pode variar entre 4 a 16 horas.

Durante a execução das *Sprints*, diariamente o time faz uma reunião de 15 minutos para acompanhar o progresso do trabalho e agendar outras reuniões necessárias. Na reunião diária (*Daily Scrum Meeting*), cada membro do time responde a três perguntas básicas: O que eu fiz no projeto desde a última reunião? O que irei fazer até a próxima reunião? Quais são os impedimentos?

No final da *Sprint*, é realizada a reunião de revisão (*Sprint Review Meeting*) para que o Time apresente o resultado alcançado na iteração ao *Product Owner*. Neste momento as funcionalidades são inspecionadas e adaptações do projeto podem ser realizadas. Em seguida, o *ScrumMaster* conduz a *Sprint Retrospective Meeting*, com o objetivo de melhorar o processo/time e/ou produto para a próxima *Sprint*.

No Scrum, o monitoramento do progresso do projeto é realizado através de dois gráficos principais: *Product Burndown* e *Sprint Burndown*. Estes gráficos mostram ao longo do tempo a quantidade de trabalho que ainda resta ser feito, sendo um excelente mecanismo para visualizar a correlação entre a quantidade de trabalho que falta ser feita (em qualquer ponto) e o progresso do time do projeto em reduzir este trabalho.

4. O Uso do Scrum para Gerenciamento de Programas de Melhoria

O uso de metodologias ágeis no C.E.S.A.R foi considerado com o objetivo de promover simplicidade ao gerenciamento de projetos sem perder a visibilidade e controle requeridos pelo grau de maturidade alcançado neste contexto.

Atualmente, pelo menos 14 projetos de desenvolvimento de software utilizam ou utilizaram o Scrum como base para seu planejamento e gerenciamento, apresentando resultados positivos identificados através de lições aprendidas. Estes resultados, alinhados aos objetivos estratégicos do C.E.S.A.R, fundamentaram a decisão de adotar o Scrum também para gestão do programa de melhoria de processos de software. Neste sentido, pequenas customizações foram realizadas para adequar o Scrum para este contexto, as quais estão descritas a seguir.

4.1 Papéis

As primeiras customizações foram relacionadas aos papéis definidos no Scrum, os quais foram atribuídos da seguinte forma:

Product Owner

São os responsáveis pelos processos específicos. Em geral, são gerentes de área responsáveis pela maior parte da execução de um processo. Os seguintes *Product Owners* foram definidos inicialmente, considerando o organograma do C.E.S.A.R:

- Gerente de Qualidade e Controles Internos, *Product Owner* de ações de melhoria em processos de garantia da qualidade, gestão de processos e medição e análise.
- Gerente de Engenharia, *Product Owner* de ações de melhoria em processos de requisitos, análise e projeto, implementação, verificação, validação e gerência da configuração.
- Gerente Chefe de Projetos, *Product Owner* de ações de melhoria em processos de planejamento e acompanhamento de projetos.

Scrum Master

O *Scrum Master* é o responsável por prover apoio ao time para o bom uso do *Scrum*, além de tratar os impedimentos. Este papel está sendo desempenhado pelo próprio coordenador do SEPG devido ao perfil do mesmo (*Certified Scrum Master*).

Time

O *time* compreende todos os responsáveis pela execução de ações de melhoria. No caso do SEPG do C.E.S.A.R (vide seção 2) há vários times atuando conjuntamente e responsáveis individualmente por grupos de processos. a saber: Gestão de Projetos, Engenharia, Design, Organizacional e Métricas.

4.2 Ciclo de Desenvolvimento de Projetos de Melhoria

Conforme os princípios do Scrum, os projetos de melhoria são desenvolvidos em iterações (*sprints*), as quais são planejadas de acordo com as prioridades da organização e a capacidade do time (vide Fig. 2). A seguir são comentados cada componente deste ciclo sob a ótica de projetos de melhoria.

Definição da Visão:

A visão compreende os objetivos reais do projeto de melhoria específico, prioridades de melhoria, assim como premissas e restrições. Entre as informações da visão, podem compreender, por exemplo, os objetivos de negócio a serem obtidos com a implementação da melhoria específica. Estes objetivos podem incluir redução de custo, aumento de produtividade, qualidade do produto entre outros.

Construção do Product Backlog

O *Product Backlog* contém os requisitos funcionais e não-funcionais do produto a ser desenvolvido. No caso de programas de melhoria, requisitos relativos aos objetivos do processo são relacionados. Por exemplo, considerando um programa de melhoria cujo

objetivo seja a aderência ao nível 3 do CMMI, um dos itens de *backlog* pode ser a estruturação do grupo de melhoria ou o estabelecimento de um sistema de medição.

Os itens de *backlog* foram identificados a partir das metas e prioridades estabelecidas pelo Comitê Executivo do SEPG e pelas solicitações de melhoria registradas pelos desenvolvedores na ferramenta de controle de mudanças. Os itens foram organizados por grupo da melhoria e atribuídos aos seus respectivos times responsáveis, como pode ser visto na Fig. 3. Após isso, os mesmos foram estimados (em horas) e priorizados de acordo com o seu valor de negócio agregado.

IBL #	Grupo	Subprocesso	Item de Backlog	Descrição	Origem	Time
IBL1.01	Migração para EPF	PSC	Estudo do EPF	Estudo e apresentação do EPF para o comitê executivo.	Planejamento Estratégico	Organizacional
IBL1.02	Migração para EPF	PSC	Arquitetura do ProsCes no EPF	Definir a nova arquitetura para o processo	Planejamento Estratégico	Organizacional
IBL1.03	Migração para EPF	PSC	Identidade visual do ProsCes no EPF	Definir a nova identidade visual para o processo. Logomarca + stylesheet no EPF.	Planejamento Estratégico	Design
IBL3.01	Projeto Melhoria	PP, PMC, PV	Planejamento do PMO	Início e planejamento inicial do programa de melhoria para definição e institucionalização de um escritório de projetos e gestão de portfólio.	Planejamento Estratégico	Gestão
IBL7.01	Adaptação	TODOS	Definição do Processo de Consultoria de Planejamento Estratégico		Planejamento Estratégico	Gestão
IBL1.08	Migração para EPF	MA	Migração do subprocesso de Medição e Análise		Planejamento Estratégico	MA
IBL1.13	Migração para EPF	REQ	Migração do subprocesso de Requisitos		Planejamento Estratégico	Engenharia
IBL1.14	Migração para EPF	AEP	Migração do subprocesso de Análise e Projeto		Planejamento Estratégico	Engenharia

Fig. 3. *Product Backlog* do programa de melhoria.

Sprint Planning 1

Na *Sprint Planning 1* os itens de *backlog* são selecionados de acordo com as prioridades junto aos *Product Owners* do processo. Esta seleção resulta nos itens de *backlog* que serão realizados na próxima *sprint*. As *sprints* têm duração de 4 semanas.

No programa de melhoria do C.E.S.A.R, são realizadas 5 reuniões mensais, uma para cada time do SEPG, na qual participam o coordenador do SEPG (*Scrum Master*), os LAPs e CAPs (time) e o seu respectivo *Product Owner*.

Sprint Planning 2

Similarmente à *Sprint Planning 1*, estas reuniões são realizadas mensalmente, uma para cada time do SEPG, com foco no detalhamento das atividades a serem realizadas de acordo com os itens selecionados na reunião anterior (*Sprint Backlog*). Participam destas reuniões o Coordenador do SEPG (*Scrum Master*) e o time (formado por LAPs e CAPs) dos subprocessos. A partir do detalhamento, o esforço é estimado para realização de cada atividade específica bem como um planejamento semanal para a realização das atividades ao longo da *sprint*.

Monitoramento da *Sprint* e do *Product Backlog*

Uma vez definido e estimado o escopo da *sprint*, bem como o conjunto de tarefas a ser realizado, inicia-se o acompanhamento do progresso do programa de melhoria em dois níveis: monitoramento da *sprint* e monitoramento do *Product Backlog*.

O monitoramento da *sprint* é feito semanalmente (e não diariamente como sugerido no Scrum) já que o time está alocado parcialmente ao SEPG. Uma vez por semana, cada time se reúne com o *Scrum Master* e responde a três perguntas recomendadas pelo Scrum: O que eu fiz desde a última reunião? O que irei fazer até a próxima reunião? Quais são os impedimentos? Antes da reunião, entretanto, cada membro do time atualiza o *Sprint Backlog* com o total de horas realizado desde a última reunião e o replanejamento de horas para concluir as atividades planejadas para as próximas semanas. Durante a execução da *sprint*, novas atividades podem ser inseridas no *Sprint Backlog* à medida que forem sendo identificadas pelo time.

O gráfico de *Burndown* da *sprint* dá ao time uma indicação semanal da sua velocidade e do progresso do trabalho em relação ao que foi planejado, indicando como a resolução das tarefas do *Sprint Backlog* está evoluindo e se serão concluídas ao final da mesma. A Fig. 4 exemplifica um gráfico de *Burndown* da *sprint* do programa de melhoria do C.E.S.A.R.

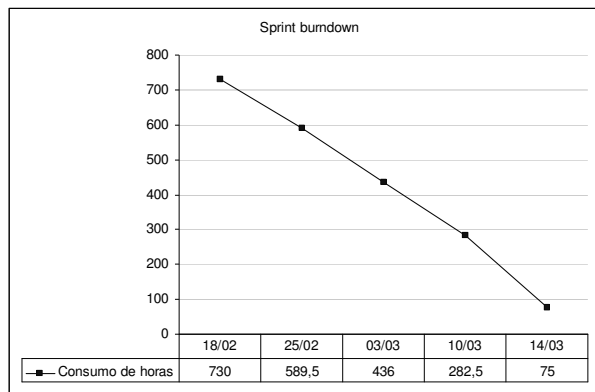


Fig. 4. Gráfico do *Sprint Burndown* do programa de melhoria

O monitoramento do *Product Backlog* é feito através de gráficos de *Burndown* que dão uma indicação de quão rapidamente o time está “queimando” o trabalho a ser realizado e entregando os requisitos do *backlog* do produto. É uma ferramenta útil para ajudar a planejar quando liberar ou remover requisitos de uma *sprint* caso o progresso do programa de melhoria não esteja ocorrendo como planejado. A Fig. 5 exemplifica os gráficos de *Burndown* do *Product Backlog* que estão sendo usados no contexto do programa de melhoria do SEPG.

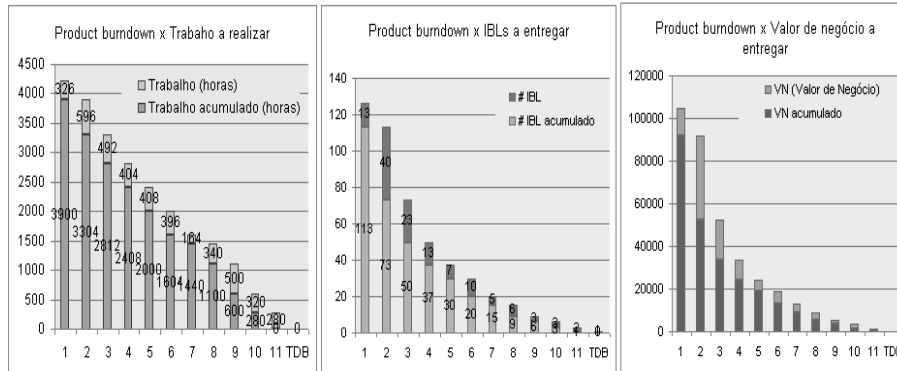


Fig. 5. Gráficos de *Product Burndown* do programa de melhoria

Revisão e Retrospectiva da *Sprint*

Ao final de cada *Sprint* é realizada uma reunião de revisão para que os times do projeto apresentem o resultado alcançado na iteração. Participam da reunião todos os membros do Comitê Executivo do SEPG (*Product Owners*), bem como o Coordenador do SEPG (*Scrum Master*). Neste momento são apresentados todos os resultados alcançados durante a execução da *sprint* (como por exemplo, apresentação de um novo processo de consultoria) e adaptações do programa de melhoria podem ser realizadas.

Em seguida, o Coordenador do SEPG (*Scrum Master*) conduz a reunião de retrospectiva da *sprint* com todo o time, com o objetivo de melhorar o processo/time e/ou produto para a próxima *sprint*.

5. Lições Aprendidas

A partir do uso do Scrum para gerenciamento do programa de melhoria de processos do C.E.S.A.R foram identificadas algumas lições aprendidas que serão resumidas brevemente a seguir.

- O planejamento ágil do Scrum, através da construção e priorização de itens de *backlog*, apresentou-se como um mecanismo efetivo de monitoramento das várias solicitações de melhoria provenientes de diversas áreas organizacionais.
- O uso do Scrum permitiu uma maior aproximação e comprometimento do comitê executivo no programa de melhoria, já que o mesmo passou a atuar como *Product Owner* dos processos de desenvolvimento de software, participando mais ativamente da construção e priorização do *product backlog*, além de se sentirem os donos dos processos aos quais são diretamente responsáveis.
- As melhorias do processo são definidas e priorizadas em conjunto com o comitê executivo garantindo que a melhoria contínua do ProSCes gere mais valor agregado para os seus usuários. As melhorias não são mais priorizadas

unilateralmente pelo SEPG. Ao final de cada *sprint* são entregues aos usuários as melhorias que possuem a melhor relação entre o valor agregado à organização e a velocidade de produção do time, ou seja, essa forma de priorização entrega os itens de maior valor de negócio produzidos em um menor espaço de tempo.

- O uso do *Sprint Backlog* como instrumento do planejamento detalhado da *sprint* deu maior visibilidade do trabalho a ser realizado bem como do andamento do progresso do mesmo. Houve ganho de produtividade e maior comprometimento do time na hora da execução e cumprimento das tarefas já que as estimativas foram realizadas colaborativamente pelo próprio time.
- O monitoramento do programa foi baseado nos princípios do Scrum, os quais muito contribuem com a comunicação do time e com a visibilidade das tarefas. Esta forma trouxe resultados positivos ao SEPG, cujos membros possuem altas taxas de rotatividade, são fisicamente distribuídos e com alocações parciais, cenários cada vez mais constantes em grupos de melhoria de processo.
- O uso dos gráficos de *Burndown* permitiu um monitoramento efetivo do andamento da *sprint* e do *Product Backlog*, promovendo visibilidade do progresso do programa de melhoria ao time e ao comitê executivo.

Além de aspectos positivos, foram identificadas práticas não-recomendadas, as quais estão relatadas a seguir, assim como as soluções de mitigação adotadas.

- O uso de *Story Points* [2] para estimativas dos itens de *backlog* de melhoria de processos não se mostrou ser uma técnica adequada à realidade de programas de melhoria. Ao invés disso o time faz as estimativas usando o conhecimento e bases históricas.
- A grande quantidade de *Product Owners* pode causar conflitos no momento da priorização das atividades que requerem integração entre os grupos de processo. A reunião de acompanhamento do comitê executivo é fundamental para minimizar esses conflitos.

6. Conclusões

Este trabalho apresentou um relato de experiência sobre a customização e utilização do Scrum como base para o gerenciamento do programa de melhoria no C.E.S.A.R, um instituto de inovação em TIC, CMMI nível 3, sediado no Recife, Brasil.

Apesar de apresentarem propostas antagônicas para muitos, o uso de metodologias ágeis na implementação de boas práticas no desenvolvimento de software pode ser uma boa estratégia para organizações CMMI. Particularmente no estudo de caso apresentado, os resultados com o uso do Scrum têm sido positivos tanto em projetos de desenvolvimento de software como na gestão da melhoria de processos.

Analisando as lições aprendidas até o momento, pode-se concluir que o Scrum permitiu uma maior aproximação e comprometimento das equipes e dos responsáveis pelos processos específicos. A entrega dos itens de maior valor de negócio,

produzidos em um menor espaço de tempo fornece uma maior visibilidade do progresso da melhoria. O trabalho colaborativo promoveu um ganho na produtividade e maior comprometimento do time na hora da execução e cumprimento das tarefas.

Considerando os resultados positivos encontrados até o momento, ações de melhoria estão sendo executadas sempre visando produtividade e motivação do time de melhoria de processos do C.E.S.A.R.

Referências

1. ADM - Advanced Development Methods.: Controlled chaos: living on the edge, <http://www.controlchaos.com/old-site/ap.htm>
2. Cohn, M.: Agile Estimating and Planning, Prentice Hall, 330 p. (2006)
3. Fowler, P., Rifkin, S.: “Software Engineering Process Group Guide”, CMU/SEI-90-TR-024, SEI - Software Engineering Institute. (1990)
4. Gloger, B.: The Zen of Scrum, <http://www.glogerconsulting.de>.
5. ISO: “ISO/IEC 15504 – Information Technology – Process Assessment”. (2004)
6. PROSCES: “Processo de Software do C.E.S.A.R” Centro de Estudos e Sistemas Avançados do Recife, Versão 3.0. (2007)
7. Schwaber, K.: Agile Project Management With Scrum, Microsoft Press, Redmond, Washington, USA. (2004)
8. SEI, Software Engineering Institute: CMMI-DEV: CMMI for Development, V1.2 model, CMU/SEI-2006-TR-008, <http://www.sei.cmu.edu/cmmi/general/> (2006)
9. SOFTEX: “MPS.BR – Melhoria de Processo de Software: Guia Geral” (2005)