

## TECHNICAL SOFTWARE DEVELOPMENT WITH THE ADOPTION OF AGILE METHODOLOGY

T. M. R. Dias<sup>1</sup>, P. M. Dias<sup>1</sup>, G. F. Moita<sup>2</sup>, D. H. Noronha<sup>3</sup>

<sup>1</sup>INESP/FUNEDI-UEMG - Instituto de Ensino Superior e Pesquisa, Fundação Educacional de Divinópolis

<sup>2</sup>CEFET-MG – Centro Federal de Educação Tecnológica de Minas Gerais

<sup>3</sup>IFMG – Instituto Federal de Minas Gerais

**Abstract.** *In this paper we present a study of the MPS.BR and the Scrum development methodology, describing their practices and runs. Consideration was given to visit with a particular company that is currently gearing up to make sure the level of F MPS.BR. Based on these studies, some basic practices are suggested, using the Scrum methodology and MPS.BR, so that small businesses can improve their processes and forth adding agility and dynamism that Scrum provides the organization and credibility of MPS.BR of so try to achieve a higher quality in the development process of a small or medium sized businesses.*

**Keywords:** SCRUM, Software Process Improvement, Software Quality.

**Resumo:** *Neste trabalho é realizado um estudo sobre o MPS.BR e sobre a metodologia de desenvolvimento Scrum, descrevendo suas práticas e funcionamentos. Foi analisada a visita feita a uma determinada empresa que no momento está se preparando para se certificar no Nível F do MPS.BR. Com base nesses estudos, são sugeridas algumas práticas básicas, utilizando a metodologia Scrum e o MPS.BR, para que pequenas empresas possam melhorar ou adiantar seus processos somando a agilidade e o dinamismo que o Scrum oferece com a organização e a credibilidade do MPS.BR, de forma a tentar conseguir uma qualidade maior nos processos de desenvolvimento de uma pequena ou media empresa.*

**Palavras-chaves:** SCRUM, Melhoria de Processo de Software, Qualidade de Software.

## 1. INTRODUÇÃO

Há não muito tempo a construção de Software não tinha processo definido, não havia controle, o custo para construção destes sistemas era alto e o seu ciclo de vida era indefinido e curto.

Com o aumento dos custos, principalmente pelo fato dos sistemas a cada dia estarem mais complexos, as empresas com necessidades de economizar e aperfeiçoar processos viram que a criação de um modelo para desenvolvimento de software que possa minimizar estes custos e atendessem uma demanda cada vez mais crescente, seria uma alternativa interessante.

Espelhar-se na engenharia tradicional para a construção de software foi então um caminho que teoricamente deveria ser seguido. Neste cenário surge a Engenharia de Software que segundo Sommerville [13] é uma disciplina de engenharia relacionada com todos os aspectos da produção de software.

Com esse novo modo de desenvolver software alguns problemas foram resolvidos e novas necessidades foram incorporadas à construção de software como, reutilizar parte do software e utilizar componentes para construir sistemas a partir de pedaços, podendo gerar algumas facilidades e tornar o processo menos oneroso.

Após o conceito de Engenharia de Software surge também o conceito de Qualidade de Software, que segundo Vasconcelos [14], está relacionado com a normalização e melhoria de processos, medições, padrões, verificações, entre outros. Apesar dos modelos aplicados na garantia da qualidade de software atuar principalmente no processo, o principal objetivo é garantir um produto final que satisfaça às expectativas do cliente.

No cenário brasileiro um comitê criado para a melhoria do software nacional denominado SOFTEX tentou tomar uma medida que se adequasse à realidade das empresas brasileiras. Baseado em outros modelos e no próprio CMMI, nasceu o MPS.BR (Melhoria do Processo de Software Brasileiro). Esta abordagem além de conter as melhores práticas de cada uma das metodologias de avaliação citadas, tem uma melhor adaptação à realidade do Brasil, composto de empresas médias e pequenas que querem buscar qualificação a um custo acessível com valor reconhecido.

Mas para muitas empresas era inviável a utilização do MPS.BR. Essas empresas que buscavam por processos que se adequassem à sua realidade seguiram exemplos de outras empresas. Algumas optaram por utilizar Metodologias Ágeis que são mais dinâmicas e menos burocráticas, porém perdem em organização e não atendem quem precisa de certificação nacional para concorrer de igual para igual no mercado.

## 2. O MODELO DE REFERÊNCIA MPS.BR

O Modelo de Referência MPS.BR é um programa para Melhoria de Processo do Software Brasileiro coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX), contando com apoio do Ministério da Ciência e Tecnologia (MCT), da Financiadora de Estudos e Projetos (FINEP) e do Banco Interamericano de Desenvolvimento (BID). [10][11][12]

O MPS.BR tem como sua base conceitos de maturidade e capacidade de processo para a avaliação e melhoria da qualidade, produtividade de produtos de software e serviços. Dentro desse contexto, o MPS.BR possui três componentes: Modelo de Referência (MR-MPS), Método de Avaliação (MA-MPS) e Modelo de Negócio (MN-MPS).

Uma das metas do MPS.BR é definir e aprimorar um modelo de melhoria e avaliação de processo de software, visando preferencialmente as micro, pequenas e algumas médias empresas, de forma a atender suas necessidades de negócio e ser reconhecido nacional e internacionalmente como um modelo aplicável à indústria de software. O MPS.BR estabelece um modelo de referência de processos de software, um processo e um método de avaliação de processos que dá sustentação e garante que o MPS.BR está sendo empregado de forma coerente com as suas definições. O MPS.BR estabelece também um modelo de negócio que apóie sua adoção pelas empresas brasileiras desenvolvedoras de software.

O MPS.BR Melhoria de Processo de Software Brasileiro, é um movimento para a melhoria de processos, possui um modelo de referência para qualidade de processo e um modelo de negócio voltado para a realidade do mercado de pequenas e médias empresas de desenvolvimento de software no Brasil. Segundo o MPS.BR [10], o MPS.BR é baseado no CMMI, nas normas ISO/IEC 12207 e ISO/IEC 15504 e na realidade do mercado brasileiro.

A norma ISO/IEC 15504 é dividida em um framework para modelos e métodos de avaliação de processo e na 15504-5 que define um modelo exemplo de capacidade de processo.

Segundo Soares [9] o framework é composto pelos elementos:

- Guia para utilização dos resultados de uma avaliação de processo;
- Estrutura para medição de capacidade de processo, composta por seis níveis sequenciais e cumulativos de capacidade de processo;
- Requisitos para um processo de avaliação de processo;
- Requisitos para modelos de avaliação de processo.
- Requisitos para verificação de conformidade de uma avaliação.
- Requisitos para modelos de processo.
- Exemplo de atividades de um processo de avaliação.
- Guia para avaliação de processo, incluindo orientações para qualificação de avaliadores competentes.
- Exemplo de um modelo de avaliação de processo de software.

De acordo com Dias [3], a norma ISO/IEC 12207 estabelece o ciclo de vida de software com seus processos, atividades, tarefas e resultados. E tem como objetivo definir uma estrutura comum para todos os envolvidos no desenvolvimento do software, como adquirentes, fornecedores, operadores, desenvolvedores, mantenedores, gerentes, profissionais de qualidade e usuários. A responsabilidade define que o processo deve ser de responsabilidade de uma parte (organização ou parte dela participante do contrato).

Os processos são agrupados de acordo com a sua natureza, ou seja, processos de apoio, processos fundamentais e processos organizacionais. Os processos fundamentais são aqueles que uma organização obrigatoriamente executa para que seu serviço de desenvolvimento, manutenção ou operação de software seja realizado.

Um Modelo de Referência de Processo define um conjunto de processos que representam as melhores práticas de um determinado domínio. Por sua vez, um Modelo de Avaliação de Processo deve ser baseado num Modelo de Referência de Processo e detalha os processos de forma a viabilizar uma avaliação de processo em relação a este modelo e também detalha a estrutura de medição.

O MPS.BR é composto por três partes: MR-MPS, Modelo de Referência para Melhoria do Processo de Software, que será explicado a seguir; MA-MPS, Método de Avaliação para Melhoria do Processo de Software, que tem como objetivo orientar a realização de avaliações em conformidade com a norma ISO/IEC 15504 em organizações que implementaram o MR-MPS; e MN-MPS, Modelo de Negócio, que consiste em regras para a implementação do MR-MPS pelas empresas de consultoria, de software e avaliação, o que o torna acessível a empresas de médio e pequeno porte.

O modelo de referência MR-MPS define níveis de maturidade que são uma combinação entre os processos e sua capacidade.

O MPS.BR contém sete níveis de maturidade, são eles : A (Em Otimização), B (Gerenciado Quantitativamente), C (Definido), D (Largamente Definido), E (Parcialmente Definido), F (Gerenciado) e G (Parcialmente Gerenciado). A escala dos níveis de maturidade se inicia no nível G e progride até o nível A. A possibilidade de se realizar avaliações considerando diferentes níveis de maturidade permite uma visibilidade dos resultados de melhoria de processos em prazos mais curtos. Os níveis são acumulativos, ou seja, se a empresa está no nível F, esta possui o nível de capacidade do nível F que inclui os atributos de processo dos níveis G e F para todos os processos relacionados. Isto significa que, ao passar do nível G para o nível F, os processos do nível de maturidade G passam a ser executados no nível de capacidade correspondente ao nível F. Esses processos devem ser realizados durante todo o projeto executado por uma empresa.

Neste trabalho são considerados o nível de maturidade G constituído dos processos de Gerência de Projeto e Gerência de Requisito, e o nível de maturidade F composto por Gerência de Configuração, Garantia de Qualidade e Medição, excluindo Aquisição e Gerência de Portfólio por serem muito específicos e não obrigatórios.

## **2.1 Nível G – Parcialmente Gerenciado**

O nível de maturidade G é composto pelos processos Gerência de Projetos e Gerência de Requisitos. Neste nível os processos devem satisfazer os atributos de processo AP 1.1 e AP 2.1. [10]

O propósito do processo Gerência de Projetos é estabelecer e manter planos que definem as atividades, recursos e responsabilidades do projeto, bem como prover informações sobre o andamento do projeto que permitam a realização de correções quando houver desvios significativos no desempenho do projeto. O propósito deste processo evolui à medida que a organização cresce em maturidade. Assim, a partir do nível E, alguns resultados evoluem e outros são incorporados, de forma que a gerência de projetos passe a ser realizada com base no processo definido para o projeto e nos planos integrados. No nível B, a gerência de pro-

jetos passa a ter um enfoque quantitativo, refletindo a alta maturidade que se espera da organização. Novamente, alguns resultados evoluem e outros são incorporados.

O propósito do processo Gerência de Requisitos é Gerenciar os requisitos dos produtos e componentes do produto do projeto e identificar inconsistências entre os requisitos, os planos do projeto e os produtos de trabalho do projeto.

## **2.2 Nível F – Gerenciado**

O nível de maturidade F é composto pelos processos do nível de maturidade anterior (G) acrescidos dos processos Aquisição, Gerência de Configuração, Garantia da Qualidade e Medição. Todos estes processos devem satisfazer os atributos de processo AP 1.1, AP 2.1 e AP 2.2 [10].

O propósito do processo de Configuração é estabelecer e manter a integridade de todos os produtos de trabalho de um processo ou projeto e disponibilizá-los a todos os envolvidos.

O propósito do processo Garantia da Qualidade é assegurar que os produtos de trabalho e a execução dos processos estejam em conformidade com os planos e recursos predefinidos.

O propósito do processo Medição é coletar, analisar, relatar os dados relativos aos produtos desenvolvidos e aos processos implementados na organização e em seus projetos, de forma a apoiar os objetivos organizacionais.

Os demais níveis de maturidade do modelo de referência MPS.BR não serão citados, pois não serão utilizados neste trabalho. O foco é que empresas possam começar a introduzir alguns processos mais básicos.

O nível F será até onde vamos abordar neste trabalho, pois se trata de uma iniciação para os processos mais básicos do MPS.BR utilizados juntamente com Scrum.

## **3. METODOLOGIA SCRUM**

De acordo com Soares [9], o Scrum assume a premissa de que o desenvolvimento de software é muito complexo e imprevisível para ser planejado totalmente no seu início.

A maior vantagem do Scrum e outros métodos ágeis é que, rapidamente consegue-se avaliar se o que foi feito está certo ou errado.

O Scrum destaca-se dos demais métodos ágeis pela ênfase dada ao Gerenciamento de projeto [6].

O Scrum reúne atividades de monitoramento e feedback, em geral, reuniões rápidas e diárias com toda a equipe, visando à identificação e correção de quaisquer deficiências ou impedimentos no processo de desenvolvimento.

A metodologia Scrum originou-se através de um estudo dos processos de fabricação de automóveis na Toyota em 1991 e tornou-se público como Scrum em 1995. Seu nome vem de uma característica jogada de uma partida de Rugby.

Trabalho em equipe, concentração, foco curto. São os principais pontos que norteiam o Scrum.

A idéia do SCRUM é justamente definir papéis bem específicos para as pessoas envolvidas no projeto, seria como se cada pessoa jogasse, ou seja, o que cada uma vai ter que fazer para que o time alcance seu objetivo: que neste caso é o próprio desenvolvimento do software.

Para ser considerada ágil a metodologia deve aceitar a mudança ao invés de tentar prever o futuro. O problema não é a mudança em si, mesmo por que todo projeto está sempre passível a mudanças. O problema é como receber, avaliar e responder às mudanças. Enquanto as metodologias ágeis variam em termos de práticas e de ênfases, elas compartilham algumas características consideradas básicas, como desenvolvimento iterativo e incremental, comunicação e redução de documentação extensiva. Desta maneira existem maiores possibilidades de atender aos requisitos do cliente, que muitas vezes são mutáveis [9].

O Scrum introduz os seguintes artefatos principais usados no fluxo de desenvolvimento: Product Backlog, Sprint Backlog e um incremento de funcionalidade do produto, mas há alguns outros artefatos importantes a serem citados. Em geral essa metodologia funciona como mostra figura 1:



FIGURA 1. Ciclo da metodologia Scrum.

É importante ressaltar que as práticas do Scrum podem ser aplicadas em qualquer contexto onde pessoas precisem trabalhar juntas para atingir um objetivo comum. O Scrum é recomendado para projetos de praticamente todas as áreas além de software, principalmente para projetos de pesquisa e inovação. [2]

Esse modelo se aplica ao caso da empresa visitada, já que esta é de inovação tecnológica.

## **4. PROPOSIÇÃO DE ADEQUAÇÃO E ADOÇÃO**

Algumas proposições de adaptações de processos Scrum/MPS.BR são propostas em um nível mais superficial e inicial, ou seja, para empresas que não realizam processos de desenvolvimento e querem utilizar algum, ou empresas que já trabalham com Scrum e precisam preparar os seus processos para receber os processos do MPS.BR. Algumas das adequações neste capítulo foram baseadas no caso de uma empresa do sul de Minas Gerais que já utilizava boa parte do Scrum e está preparando para se adequar ao MPS.BR.

Por opção da empresa visitada, o mínimo do MPS.BR será executado a fim de conseguir uma certificação de nível F, mas o Scrum por funcionar muito bem para eles será utilizado, ou pelo menos uma parte dele, como metodologia de desenvolvimento. O foco neste trabalho é o meio termo, se possível, entre a agilidade dos processos Scrum e a confiabilidade e organização dos processos do MPS.BR.

No MPS.BR existem processos a serem cumpridos para gerar um resultado satisfatório no final de cada processo, para conseguir uma certificação ou mesmo para o caso onde uma empresa queira melhorar a organização e execução dos seus processos, podendo gerar melhoria.

De acordo com Alvim [1], “Comunicação por escrito é boa para criação de memória, mas o canal menos eficaz para obtenção de entendimento, quando comparada a meios interativos”. Isso pode fazer essa adequação e adoção interessante.

### **4.1 Processos de Nível G (Parcialmente Gerenciado)**

O nível de maturidade G é composto pelos processos de Gerência de Projetos e Gerência de Requisitos. Neste nível a implementação dos processos deve satisfazer os atributos de processo AP 1.1 e AP 2.1. [10]

AP 1.1 É para definir se o processo é executado. Essa AP pode ser realizada utilizando o Scrum ou qualquer outra metodologia desde que exista um processo definido e que o que foi planejado nesse processo gere o resultado definido.

AP 2.1 Se o processo é gerenciado. A empresa precisa de política organizacional estabelecida e mantida para o processo até o final do mesmo, a parte do desenvolvimento do software pode ser o Scrum ou até mesma outra desde que seja seguida.

Para atingir essas duas APs o próprio guia MPS.BR utiliza de alguns processos referentes aos níveis de maturidade. No nível G são as Gerências de projetos (GPR) e Gerências de requisitos (GRE).

#### **4.1.1 Gerência de Projetos – GPR**

“Gerenciamento de Projetos (GP) é uma área de atuação e conhecimento que tem ganhado cada vez mais reconhecimento e importância. Um dos principais difusores do Gerenciamento de projetos e da profissionalização do gerente de projetos é o Instituto de Gerenciamento de Projetos”. [7]

Seria interessante para uma empresa ter um Gerente de Projetos que também tenha pleno conhecimento de como funciona o Scrum nesse caso.

O Gerenciamento de Projetos talvez seja o processo de maior relevância para se adaptar ao Scrum, pois está diretamente ligado ao desenvolvimento e parcialmente ligado a todos os outros processos até o nível F. Com base na visita a empresa e de acordo com Santesso [8].

Os processos devem cumprir as 17 GPR para atingir o nível de maturidade G e F, esses processos são:

GPR1: O escopo do trabalho para o projeto é definido;

Realizando uma coleta de requisitos correta com o Product Owner, deixando claro os objetivos, os limites e fazendo um bom Product Backlog o Scrum pode atender muito bem essa condição.

GPR2: As tarefas e os produtos de trabalho do projeto são dimensionados utilizando métodos apropriados;

O principal objetivo desse processo é estabelecer o tamanho do projeto. No Scrum as estimativas de tamanho do Product Backlog é realizado utilizando StoryPoints e pode ser acompanhado pelo Burn-down chart.

GPR3: O modelo e as fases do ciclo de vida do projeto são definidos;

Ciclo de vida define um conjunto de fases, onde cada uma destas gera atividades a serem atingidas para alcançar a fase posterior. No Scrum, o ciclo de vida pode ser dividido em quatro fases ou funções: Planejamento, Preparação, Desenvolvimento e Entrega.

A figura 2 apresenta um exemplo de ciclo de vida utilizado por uma pequena empresa no ramo de inovação tecnológica no sul de Minas Gerais.

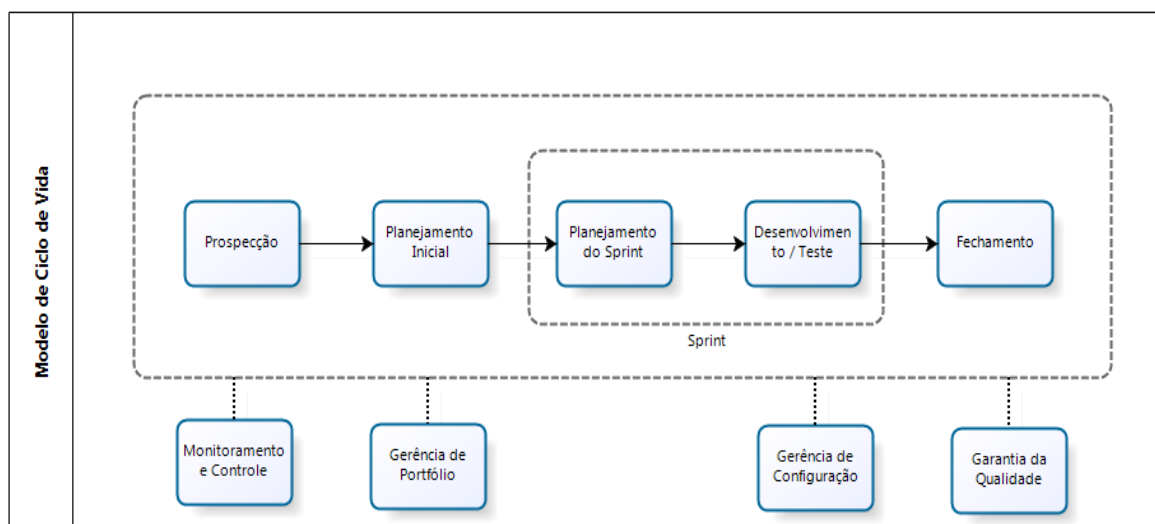


FIGURA 2. Ciclo de vida do projeto

A primeira etapa é a prospecção, onde ocorre o atendimento ao cliente e a coleta dos requisitos é feita com base nas expectativas do cliente, ou no caso de o projeto ter origem na própria empresa é realizado uma descrição formal da oportunidade de negócio, é avaliado os recursos que serão gastos e estimado o esforço gasto no projeto – essa estimativa é feita de



acordo com a experiência que a própria empresa adquiriu em outros projetos, caso a empresa não tenha nenhum projeto concluído cabe a algum funcionário que conheça toda a equipe e seja capaz de mensurar a capacidade da mesma de estimar o esforço gasto.

No planejamento inicial O Product Backlog é realizado e documentado pelo Product Owner, onde as prioridades são definidas.

Já definidas as prioridades, o Scrum Master planeja todos os processos tendo por base Sprint, depois do decorrer do projeto e já planejado sprint por sprint, pode ser alterado quando necessário.

As Sprints de três semanas – no caso dessa empresa determinada – vão sendo concluídas e novas são iniciadas até o fechamento do projeto.

O Monitoramento, a Gerência de Portfólio, Gerência de Configuração e a Garantia de Qualidade são realizadas em paralelo a outras atividades, podem ser realizadas por pessoas que participam do Scrum Team ou por pessoas fora da equipe. No caso da empresa citada, cada um do Scrum Team é responsável por uma das Gerências devido ao envolvimento com o projeto e ao número limitado de funcionários na empresa.

GPR4: O esforço e o custo para a execução das tarefas e dos produtos de trabalho são estimados com base em dados históricos ou referências técnicas;

Nessa GPR é necessário determinar esforço e custo, o Scrum não pratica estimativas de custos com base em dados históricos e o cronograma deve ser atualizado à medida que o projeto e as funcionalidades se desenvolvem, mas a própria empresa pode criar um registro com base nas Sprints já realizadas e o Scrum Master atribuir valores.

GPR5: O orçamento e o cronograma do projeto, incluindo a definição de marcos e pontos de controle, são estabelecidos e mantidos;

Aqui se desenvolve orçamento e cronograma, estimativa de esforço + velocidade = cronograma + orçamento, o Burn-down Chart pode dar à empresa uma visão geral se o cronograma será cumprido a tempo e o Burn-down de outros projetos podem gerar dados para estimar esforços com base na capacidade da equipe.

GPR6: Os riscos do projeto são identificados e o seu impacto, probabilidade de ocorrência e prioridade de tratamento são determinados e documentados;

A identificação dos riscos e suas prováveis consequências devem ser analisadas e classificadas por prioridade. Esses registros serão levantados durante as reuniões diárias e armazenados no Impediments Backlog.

GPR7: Os recursos humanos para o projeto são planejados considerando o perfil e o conhecimento necessário para executá-los;

São determinadas relações hierárquicas e funções para cada envolvido no projeto, sendo internos ou externos à organização, pode haver funcionários com mais de uma função caso elas sejam compatíveis e não prejudiquem o desempenho do mesmo.

GPR8: As tarefas, os recursos e o ambiente de trabalho necessário para executar o projeto são planejados;

As tarefas e os recursos necessários para atendê-las, como ferramentas, serviços, componentes ou despesas, devem ser especificados e planejados. O Scrum Master e o Product Owner são responsáveis por garantir esses recursos obtidos das reuniões feitas no início de

cada interação e também pelos métodos de remoção dos impedimentos levantados pela equipe.

GPR9: Os dados relevantes do projeto são identificados e planejados quanto à forma de coleta, armazenamento e distribuição. Um mecanismo é estabelecido para acessá-los, incluindo, se pertinente, questões de privacidade e segurança;

Uma coleta de dados precisa ser realizada, o Scrum Team, Scrum Master e o Product Owner definem no início do projeto como será realizado.

GPR10: Planos para a execução do projeto são estabelecidos e reunidos no Plano de Projeto;

Plano de projeto é um grupo de documentos com informações definidas e reunidas, como ciclo de vida, tamanho, esforço, orçamento, cronograma e riscos. O documento de visão e o Product Backlog do Scrum se encaixam perfeitamente nesta categoria.

GPR11: A viabilidade de atingir as metas do projeto, considerando as restrições e os recursos disponíveis, são avaliados. Se necessário, ajustes são realizados;

As reuniões diárias (Daily Meeting) dispõem a visibilidade necessária para acompanhamento das metas e suas restrições.

GPR12: O Plano do Projeto é revisado com todos os interessados e o compromisso com ele é obtido;

A equipe, Scrum Master e Product Owner revisam os planos no início e no fim de cada Sprint.

GPR13: O progresso do Projeto é monitorado com relação ao estabelecido no Plano de Projeto e os resultados são documentados;

Nas reuniões diárias é possível prever imprevistos de aderência e monitorar o andamento do projeto. O Product Burn-down e o Sprint Burn-down também podem ser utilizados sendo de grande ajuda.

GPR14: O envolvimento das partes interessadas no projeto é gerenciado;

As práticas de Scrum são ideais para comunicação e colaboração entre a equipe e os usuários do projeto utilizando reuniões diárias ou o Sprint Review por exemplo.

GPR15: Revisões são realizadas em marcos do projeto e conforme estabelecido no planejamento;

Ao final de cada Sprint o progresso do projeto e o cumprimento das atividades são revisados durante o Sprint Review.

GPR16: Registros de problemas identificados e o resultado da análise de questões pertinentes, incluindo dependências críticas, são estabelecidos e tratados com as partes interessadas;

Deve-se realizar revisões em marcos do projeto, no Scrum essas revisões são feitas na Sprint Review e na reunião diária, onde todos os interessados participam. Um registro deve ser realizado. Na empresa visitada pretende-se realizar um documento padrão onde tudo será registrado e um membro do time será designado como gerente de projetos.

GPR17: Ações para corrigir desvios em relação ao planejado e para prevenir a repetição dos problemas identificados são estabelecidas, implementadas e acompanhadas até a sua conclusão;

É preciso gerenciar os problemas identificados. Nas reuniões diárias todo problema é conhecido pelo time, cabe ao Scrum Master e ao Product Owner identificá-los e resolvê-los.

#### **4.1.2 Gerência de Requisitos – GRE**

Os requisitos podem ser subdivididos em funcionais e não funcionais. Os funcionais são declarações das funcionalidades do sistema, das entradas específicas, como deve ser seu comportamento em algumas situações e também o que não deverá ser feito. Os não funcionais são as restrições sobre os serviços ou as próprias funções oferecidas pelo sistema. Podem ser as restrições de tempo, processo de desenvolvimento, padrões, custos, confiabilidade, segurança, normas organizacionais, dentre outros. [13]

O propósito do processo Gerência de Requisitos é Gerenciar os requisitos dos produtos e componentes do produto do projeto e identificar inconsistências entre os requisitos, os planos do projeto e os produtos de trabalho do projeto. [10]

Abaixo estão os processos a serem cumpridos de acordo com o MPS.BR:

GRE1: O entendimento dos requisitos é obtido junto aos fornecedores de requisitos;

Toda essa informação obtida junto ao fornecedor pode ser controlada pelo Product Owner, ele mesmo pode ser o fornecedor de requisitos quando não há um cliente para o projeto, sendo responsável pelas funcionalidades que serão implementadas.

GRE2: Os requisitos de software são aprovados utilizando critérios objetivos;

Durante as reuniões diárias é possível monitorar os requisitos. Caso exista alguma consequência, cabe ao Scrum Master corrigir qualquer imprevisto ao projeto.

GRE3: A rastreabilidade bidirecional entre os requisitos e os produtos de trabalho é estabelecida e mantida;

Se for encontrada alguma inconsistência, ela deve ser registrada no Impediments Backlog e discutida em uma reunião com o Scrum Team.

GRE4: Revisões em planos e produtos de trabalho do projeto são realizados visando identificar e corrigir inconsistências em relação aos requisitos;

A avaliação das tomadas corretivas é feita durante a Sprint Review: período de revisão no final de cada Sprint. A empresa visita pretende nomear um gerente de requisitos que participe do Scrum Team que está desenvolvendo esse determinado projeto.

GRE5: Mudanças nos requisitos são gerenciadas ao longo do projeto;

Quando algum requisito com consequência foi resolvido pelo time, esse deve ser excluído da lista de Impediments Backlog.

#### **4.2 Nível F Parcialmente Gerenciado**

Alguns desses processos independem da metodologia de desenvolvimento. Serão citados todos os processos mínimos obrigatórios para tentar uma certificação, mesmo que não tenha a influência do Scrum.

O nível de maturidade F é composto pelos processos do nível de maturidade anterior acrescidos dos processos de Aquisição, Garantia da Qualidade, Configuração, Gerência de

Projetos e Medição. Neste nível a implementação dos processos deve satisfazer os atributos de processo AP 1.1, AP 2.1 e AP 2.2. [10]

Neste trabalho não abordaremos Gerência de Portifólio e Aquisição por não serem obrigatórios e por não estarem definidos na empresa visitada.

AP 1.1 É para definir se o processo é executado. Essa AP pode ser realizada utilizando Scrum ou qualquer outra metodologia desde que exista um processo definido e que o que foi planejado nesse processo gere o resultado que foi definido.

AP 2.1 Se o processo é Gerenciado. A empresa precisa de política organizacional estabelecida e mantida para o processo até o final do mesmo. A execução do processo precisa ser planejada, monitorada e medida através de um processo de medição.

No caso do Scrum uma métrica pode ser realizada com base nos Sprints que vão de duas a quatro semanas geralmente. É preciso que recursos necessários para a execução do processo sejam identificados e disponibilizados e que os funcionários sejam preparados para realizar a tarefa tendo formação, treinamento e experiência.

A comunicação entre as partes interessadas no processo precisa ser gerenciada de forma a garantir o seu envolvimento. No Scrum isso funciona perfeitamente, pois todo o Scrum Team está sempre envolvido com o projeto inteiro graças ao Daily Scrum, cabe à empresa realizar uma forma de documentação simples para registrar esse controle.

AP 2.2 Os Produtos de Trabalho do Processo são gerenciados. Todos os requisitos dos produtos de trabalho devem ser identificados, definidos e documentados e um controle deve ser estabelecido pela empresa.

A empresa define como será seu processo padrão, incluindo diretrizes de adaptação.

A seqüência de processos deve ser definida junto com as competências requeridas para eles, ocorrem também os processos que serão realizados em paralelo ao processo padrão. A infra-estrutura e o ambiente de trabalho requerido também fazem parte do processo padrão definido pela empresa.

#### **4.2.1 Gerência de Configuração – GCO**

O propósito do processo Gerência de Configuração é estabelecer e manter a integridade de todos os produtos de trabalho de um processo ou projeto e disponibilizá-los a todos os envolvidos. [10]

GCO1: Um Sistema de Configuração é estabelecido e mantido;

É necessário que a empresa tenha um sistema de configuração definido. A empresa estudada nesse trabalho elaborou seu diagrama de gerência de configuração que pode servir de exemplo para outra empresa.

Quando uma nova Requisição de Mudança é requisitada, ela passa por um processo de análise onde o Scrum Master e o Product Owner vão avaliar se essa mudança é necessária e se acrescenta algum valor ou funcionalidade ao produto. Após analisado o impacto desse requisito é decidido pelo Product Owner ou por um gerente de configuração se a mudança poderá ser autorizada.

A figura 3 mostra em forma de diagrama como é feito o gerenciamento de mudanças de acordo com a empresa estudada.

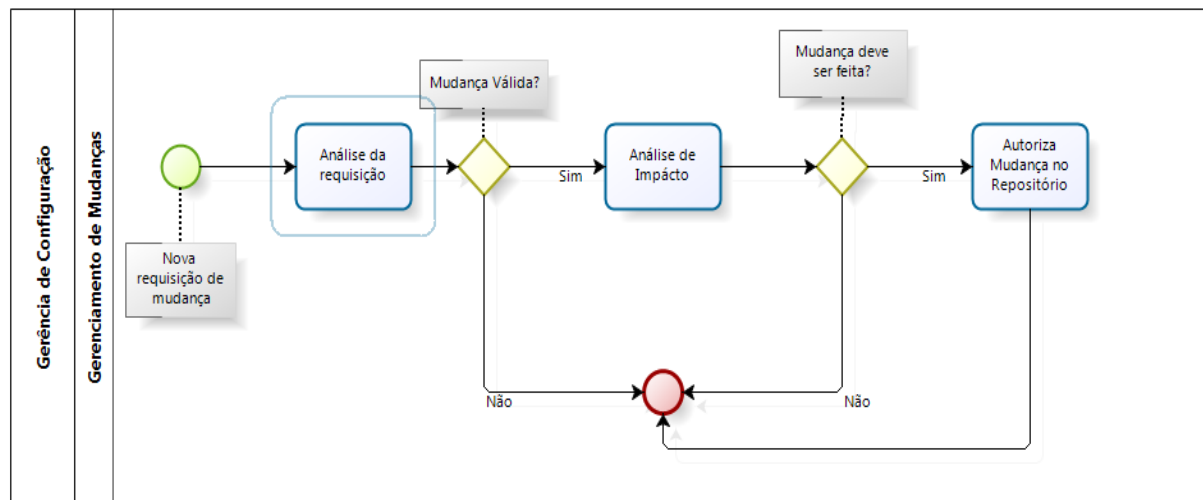


FIGURA 3. Configuração de mudanças

GO2: Os itens de configuração são identificados com base em critérios estabelecidos;

Independente da metodologia, cada empresa determina seus critérios, mas devem ter esse critério registrado e seguido até o final do projeto. O StoryPoint pode ser uma opção.

GCO3: Os itens de configuração sujeitos a um controle formal são colocados sob Baseline;

A empresa citada nesse trabalho fixou suas Baseslines nas Story Points e no final de cada Sprint, assim criaram um documento simples para o registro.

GCO4: A situação dos itens de configuração e das Baseslines é registrada ao longo do tempo e disponibilizada;

É registrado de acordo com o modelo de documento que a empresa adotou como padrão.

GCO5: Modificações em itens de configuração são controladas;

São controladas pelo Scrum Master ou outro membro do time designado como gerente de configuração.

GCO6: O armazenamento, o manuseio e a liberação de itens de configuração e base-lines são controlados;

São controladas pelo Scrum Master ou outro membro do time designado como gerente de configuração.

GCO7: Auditorias de configuração são realizadas objetivamente para assegurar que as baselines e os itens de configuração estejam íntegros, completos e consistentes.

Fica a cargo da empresa fazer essas auditorias independentes da metodologia de desenvolvimento.

#### 4.2.2 Garantia da Qualidade GQA

O propósito do processo Garantia da Qualidade é assegurar que os produtos de trabalho e a execução dos processos estejam em conformidade com os planos, procedimentos e padrões estabelecidos.

Ao ser executada qualquer atividade do processo, o responsável por ela deverá encaminhar os produtos de trabalho gerados pela atividade ao auditor que realizará a auditoria

da Garantia da Qualidade sobre os produtos de trabalho e ao Scrum Master caso essas duas funções não seja da mesma pessoa.

O auditor deverá verificar se existem não conformidades no produto de trabalho auditado usando o checklist para o produto de trabalho em questão, caso exista, ele deve comunicar as não conformidades encontradas no produto de trabalho e estipular um prazo para que o autor faça as correções necessárias.

Quando o autor do produto de trabalho realizar as correções necessárias ele deverá reenviar o produto de trabalho para o auditor para que ele faça uma nova verificação, se todas as não conformidades foram corrigidas ele deverá registrar em um documento.

GQA1: A aderência dos produtos de trabalho aos padrões, procedimentos e requisitos aplicáveis é avaliada objetivamente antes dos produtos serem entregues e em marcos predefinidos ao longo do ciclo de vida do projeto;

Se os processos anteriores já foram cumpridos, essa etapa também será cumprida automaticamente. Nesse ponto o projeto já deve estar padronizado, com as baselines definidas seja em Story Points ou no final da Sprint, o ciclo de vida já deve estar definido.

GQA2: A aderência dos processos executados às descrições de processo, padrões e procedimentos é avaliada objetivamente;

O Scrum pode ser definido como metodologia padrão de desenvolvimento e o Scrum Master avalia se os processos executados estão de acordo com a descrição do processo e o padrão definido para empresa.

GQA3: Os problemas e as não-conformidades são identificados, registrados e comunicados;

Informações do Impedimentos Backlogs podem ser usadas como registro das não conformidades e as reuniões diárias servem para comunicar ao time.

GQA4: Ações corretivas para as não-conformidades são estabelecidas e acompanhadas até as suas efetivas conclusões. Quando necessário, o escalamento das ações corretivas para níveis superiores é realizado, de forma a garantir sua solução;

De acordo com um modelo seguido em uma empresa no Sul de Minas Gerais, todos os processos executados pelo projeto deverão ser auditados pela gerência de qualidade. Ao final do processo, haverá uma auditoria verificando se o processo foi seguido de maneira correta e se todos os produtos de trabalho descritos foram criados.

O auditor deverá verificar se existem não conformidades no processo auditado usando o checklist para o processo em questão, caso exista, ele deverá comunicar as não conformidades encontradas no processo e estipular um prazo para que o responsável faça as correções necessárias.

A figura 4 mostra em forma de diagrama como pode ser realizado uma auditoria de processo.

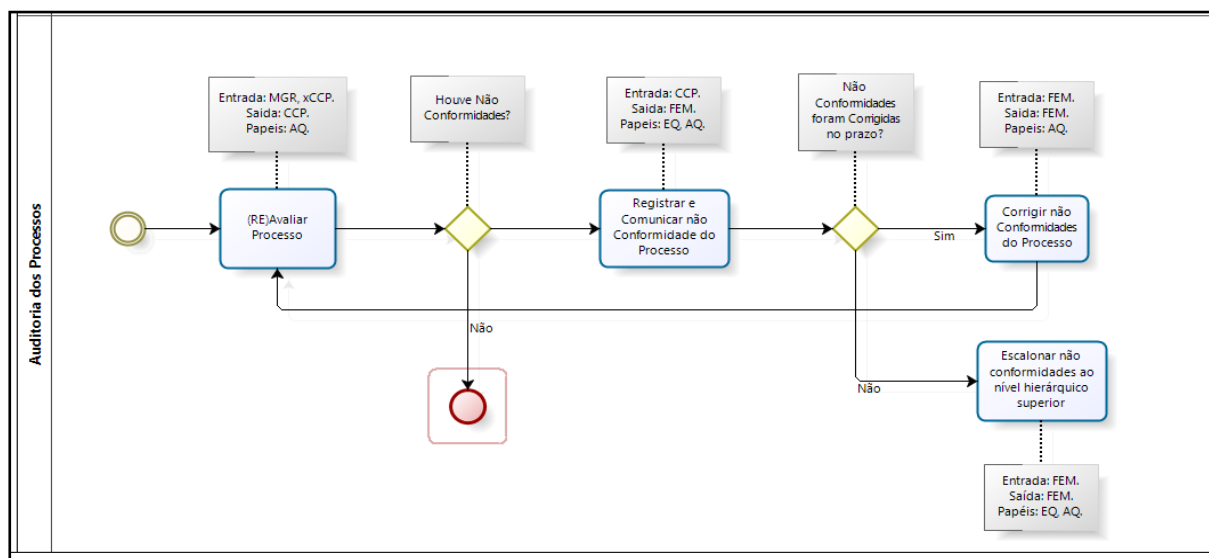


FIGURA 4. Auditoria de processos

Quando o responsável pelo processo realizar as correções necessárias ele deverá reenviar e comunicar ao auditor para que ele faça uma nova verificação. Se todas as não conformidades foram corrigidas ele deverá comunicar ao Product Owner, caso contrário o auditor deverá comunicar ao Scrum Master ou ao responsável pelo processo, que ainda existem não conformidades, reforçando a necessidade de correção das mesmas e estipulando um novo prazo para a entrega da correção. Este processo é cíclico até que todas as não conformidades do processo sejam corrigidas.

#### 4.2.3 Medição – MED

O propósito do processo Medição é coletar, armazenar, analisar e relatar os dados relativos aos produtos desenvolvidos e aos processos implementados na organização e em seus projetos, de forma a apoiar os objetivos organizacionais. [10]

MED1: Objetivos de medição são estabelecidos e mantidos a partir dos objetivos de negócio da organização e das necessidades de informação de processos técnicos e Gerenciais;

A empresa deve decidir e analisar o que é melhor para ela. No caso da empresa visitada as métricas foram montadas com base nos Sprints da metodologia Scrum justamente porque eles acreditaram que seria a melhor forma, pois já haviam utilizado o modelo. Nem tudo do Scrum é seguido, mas tudo que se julga importante sim. Cada caso de uso gera um Story Point a ser analisado e medido.

MED2: Um conjunto adequado de medidas, orientado pelos objetivos de medição, é identificado e definido, priorizado, documentado, revisado e, quando pertinente, atualizado;

É necessário definir as medidas que serão usadas. No caso da empresa visitada o Story Points e as baselines são definidas após cada sprint, um membro do time ou um funcionário em paralelo com a equipe de desenvolvimento deve gerenciar, analisar, documentar e postar no servidor as métricas. As prioridades são definidas pelo Product Owner, a empresa visitada atribui suas Story Points com base em Pontos por Caso de Uso. Valores e pesos são definidos para cada tipo de atividade dependendo da dificuldade exigida.

MED3: Os procedimentos para a coleta e o armazenamento de medidas são especificados;

No final de cada Sprint, Story Point e no fim do projeto, será armazenado um documento desenvolvido pela empresa e disponibilizado no servidor.

MED4: Os procedimentos para a análise das medidas são especificados;

As medidas podem ser estabelecidas de acordo com os Sprints e Story Points, os procedimentos serão gerados a partir dessas medidas pré-estabelecidas, cada empresa pode especificar quais procedimentos serão seguidos.

Um exemplo básico de medida planejada pela empresa está exibido abaixo na tabela1.

TABELA1: Exemplo de especificação de medidas

Nome	Mnemônico	Descrição	Procedimento de coleta
Tamanho estimado	EST_SIZE	Calculo do tamanho estimado do projeto utilizando-se da metodologia de pontos por caso de uso	Coletar executando os processos da metodologia de pontos por caso de uso
Tamanho real	REAL_SIZE	tamanho real do projeto no seu término	Executar a metodologia de pontos por caso de uso no final do projeto.
Esforço estimado	EST_EFFORT	Calculo do esforço estimado acumulado das atividades.	coletar medida na planilha do sprint Backlog ou na ferramenta MS Project
Esforço estimado acumulado	EST_ACUM_EFFORT	Calculo do esforço estimado acumulado das atividades desenvolvidas até um determinado momento.	Somatório das estimativas de esforço de todas as atividades necessárias para o desenvolvimento do projeto
Esforço real	REAL_EFFORT	Mostra o esforço real para o trabalho já realizado por um recurso ou atividade.	Coletar medida na planilha do sprint Backlog ou na ferramenta MS Project
Esforço real acumulado	REAL_ACUM_EFFORT	Mostra o esforço real para o trabalho já realizado por um recurso ou atividade, até a data de <i>status</i> , ou data atual do projeto.	Somatório do esforço das atividades realizadas até um determinado ponto do projeto.
Custo real	REAL_COST	Custo real de uma atividade do projeto	Coletar a medida do custo da atividade, no MS Project.
Custo real acumulado - AC	REAL_ACUM_COST	Custo real do projeto até um determinado momento	Somatório dos custos das atividades realizadas até um determinado momento do projeto. Coletada no MS Project

MED5: Os dados requeridos são coletados e analisados;

O Scrum Master designa alguém da equipe para se tornar responsável por coletar os dados de medição, a empresa cria um tipo de documento padrão para guardar as métricas do projeto.

MED6: Os dados e os resultados das análises são armazenados;

O responsável pela coleta das medidas terá que armazená-las em algum local. No Sul de Minas Gerais esses resultados são registrados em um documento da própria empresa e disponibilizados no servidor da mesma.

MED7: Os dados e os resultados das análises são comunicados aos interessados e são utilizados para apoiar decisões.

Nessa empresa, e-mails são enviados para comunicar aos interessados sendo, o chefe executivo, o Scrum Master ou um membro da equipe responsável por essa gerência.



## 5. CONCLUSÕES

Algumas empresas podem não se adaptar ao MPS.BR na sua totalidade, o custo é elevado, muito tempo será gasto ou simplesmente há empresas que podem não se adaptar a burocracia ou aos processos.

Há processos do MPS.BR que se encaixam no Scrum sem precisar de alterações. O MPS.BR serve como um modelo a ser seguido, não obriga a empresa fazer tudo arbitrariamente como descrito, tem uma margem para adaptação e uma liberdade para que a própria empresa crie seus processos e suas documentações com base nas requisições que o MPS.BR exige.

Na empresa visitada para este trabalho, não são utilizadas todas as práticas do Scrum, mas utilizam o que decidiram ser necessário. Do MPS.BR estão começando a aplicar o mínimo exigido de documentação e métricas. Ambos estão sendo desenvolvidos em um modelo próprio utilizando o Scrum.

As dificuldades ocorrem na aceitação dos funcionários. No caso de uma empresa que já trabalhe com SCRUM o time acha burocrático e pode não gostar de documentar o que está fazendo. Para quem está introduzindo o MPS.BR pode parecer estranho as práticas do Scrum.

Neste trabalho há algumas práticas do Scrum com MPS.BR que podem funcionar em outras empresas que já trabalham com Scrum e querem se preparar para receber uma certificação MPS.BR ou para empresas que não adotaram nenhuma metodologia de desenvolvimento e julgarem interessante essa combinação.

Então uma proposta de adoção e adequação de processos do MPS.BR e do Scrum pode ser útil para essas empresas que desejam adaptar o Scrum ao MPS.BR ou para empresas que não seguem processos definidos analisarem se algo nesse trabalho pode ser utilizado, aproveitado ou até mesmo adaptado para o uso de uma determinada empresa. Este trabalho exhibe superficialmente algumas práticas que possam ser utilizadas na inicialização de adoção ou adaptação de processos.

## Agradecimentos

Os autores agradecem a **FAPEMIG** pelo auxílio financeiro.

## 6. REFERENCIAS

- [1] ALVIN, P. SCRUM Certificado (MPS.BR F). PowerLogic, Recife, 2008.
- [2] CHAPIEWSKI, G. Como estamos indo com a adoção de SCRUM na Globo.com. Disponível em: <<http://gc.blog.br/2008/05/27/como-estamos-indo-com-a-adocao-descrum-na-globocom/>>.
- [3] DIAS, T.M.R. Estudo da viabilidade de praticas e valores do extreme programming no processo de desenvolvimento de software baseado no modelo MPS.BR, LAVRAS, 2007.

- [4] FIGUEIREDO S. C. Melhoria e Avaliação de Processo de Software com ISO/IEC 15504-5:2006. Lavras: UFLA/FAEPE, 2006.
- [5] KNIBERG, H. Scrum e XP direto das Trincheiras. InfoQ Disponível em: <<http://www.infoq.com/br/minibooks/Scrum-xp-from-the-trenches>>.
- [6] MARÇAL, A. S. C; SCRUMMI: Um processo de gestão ágil baseado no SCRUM e aderente ao CMMI;Universidade de Fortaleza, 2009.
- [7] PMBOK®, A Guide to the Project Management Body of Knowledge – PMBOK® Guide – Fourth Edition, Project Management Institute (PMI®), USA, 2008.
- [8] SANTESSO, P.H.C.Utilização de Métodos Ágeis Scrum com MPS.BR de Nível G. Londrina: Universidade Estadual de Londrina, 2009.
- [9] SOARES, S.S. Metodologias Ágeis Extreme Programming e SCRUM para o Desenvolvimento de Software, 2009. Disponível em: <<http://revistas.facecla.com.br/index.php/reinfo/article/view/146/38>>. Acesso em: 27 out.2010.
- [10] SOFTEX, Melhoria de Processo de Software Brasileiro (MPS.BR) – Guia Geral: Melhoria de Processo de Software Brasileiro 2009.
- [11] SOFTEX, Melhoria de Processo de Software Brasileiro (MPS.BR) – Guia de Implementação – Parte 1: Nível G. versão 1.1.,2009.
- [12] SOFTEX. Melhoria de Processo de Software Brasileiro(MPS.BR) – Guia de Aquisição:2009, 2009.
- [13] SOMMERVILLE, I. Engenharia de Software. 8ª ed.São Paulo. Pearson Addison Wesley,2007.
- [14] VASCONCELOS, et al., Introdução à engenharia de software e aos princípios de qualidade. Lavras: UFLA/FAEPE. (2005).