

Nesta seção você encontra artigos voltados para o planejamento e gerência de seus projetos de software.

O que faz um Gerente de projetos e um Scrum Master?

Qual profissional sua organização precisa?

Definir escopo, estabelecer planos para o projeto, criar a lista de atividades e subatividades, estabelecer cronograma, identificar riscos, avaliar e orçar o impacto dos riscos no projeto, calcular os custos humanos e materiais, estabelecer os marcos do projeto, estabelecer os indicadores de acompanhamento do projeto, compor o plano de faturamento ao cliente ou ao patrocinador, administrar o orçamento, criar e manter um plano de comunicação, obter os aceites e interagir com o departamento financeiro da empresa para gerar as faturas e acompanhar a liberação dos pagamentos junto ao cliente, entre outras responsabilidades. Estes são alguns exemplos das atribuições de um Gerente de Projetos segundo o PMI (*Project Management Institute*). Se ao ler estes pontos descritos anteriormente e não pode reconhecer ou não entender o significado deles é porque talvez não esteja familiarizado com os processos de

Fique por dentro:

Tem sido uma prática constante a publicação de vagas e oportunidades de trabalho para Gerentes de Projeto requerendo conhecimentos em metodologias ágeis e, da mesma forma, os anúncios solicitando profissionais ScrumMaster os quais possuam conhecimentos sobre a metodologia PMI. Estamos abordando dois papéis fundamentais no processo de desenvolvimento de software, porém, as organizações que estão à procura de um profissional devem saber qual papel e quais são os requisitos de conhecimento mais adequados às necessidades da organização. Este artigo visa esclarecer as diferenças entre estes papéis e auxiliar na escolha do papel mais adequado para a organização.

gerenciamento de projetos. Este desconhecimento por parte dos executivos é muito maior do que se imagina.

Segundo o relatório anual do Standish Group, o Chaos Manifesto do ano de 2012, o papel do Patrocinador foi questionado, pois caso o Executivo não possua determinadas características, ele não terá condições de atuar em plenitude e poderá colocar em risco ou até condenar



Sérgio Salles Galvão Neto

ssallesinf@hotmail.com

Gerente de Projetos certificado PMP, ITIL, Microsoft entre outras. Atuando na área de informática desde 1992 e, a mais de 10 anos liderando equipes em projetos de desenvolvimento e implantação de softwares. Nos últimos cinco anos, atuando como ScrumMaster em projetos em diversas tecnologias e ramos de negócio.

os projetos da empresa ao fracasso. No relatório emitido pelo Standish Group, foi dedicado um bom espaço para abordar os principais aspectos dos quais um Executivo deva possuir para poder desempenhar plenamente suas funções e elevar o nível de sucesso dos projetos de sistemas como um todo. Em dois tópicos, o relatório do Standish Group abordou a personalidade e a conduta dos patrocinadores e/ou executivos.

Dentre todos os tópicos, um dos que chamam à atenção foi o dedicado à avaliação da conduta dos Executivos. Nesta avaliação foram considerados itens como: a personalidade do executivo (uma espécie de DNA de sua maneira de comandar), as experiências do executivo (os diferentes tipos e condições de projetos em que o executivo participou) e, o que é necessário saber para ser tornar um bom executivo (conhecimentos e habilidades). O resultado deste estudo demonstrou 50 pontos, chamados de segredos, que um executivo deve possuir para ser considerado como ótimo.

Dentre estes 50 segredos, um dos que mais chama a atenção foi o segredo de número 3 – intitulado como Entendimento dos Processos de Gerenciamento de Projetos. Neste tópico, a avaliação realizada demonstrou o nível de dificuldade de entendimento por parte dos executivos quanto aos processos de gestão de projetos. Apenas 17% dos executivos avaliados consideraram não ter dificuldades em entender os processos de gerenciamento dos projetos (**Figura 1**), ou seja, 83% dos executivos não entendem o que está sendo discutido em termos dos processos e respectivas atividades envolvidas.

Diante deste cenário, é quase impossível a interação em plenitude de alguém a qual possua a responsabilidade de decidir e ordenar sobre um assunto do qual falta o devido conhecimento.

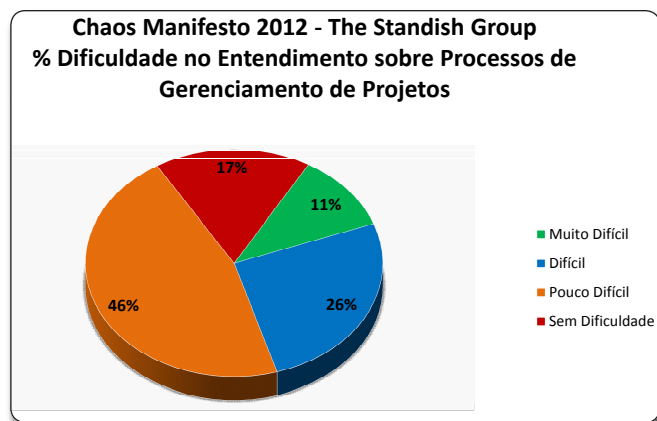


Figura 1. Percentual de Dificuldade de Entendimento sobre os Processos de Gestão de Projetos

Desta forma, se faz necessário entender o propósito dos processos de gestão de projetos, firmando o conhecimento da definição de projeto e as abordagens das metodologias que surgiram com objetivo principal de aumentar as chances de sucesso dos projetos. Em se tratando de Engenharia de Software, de desenvolvimento de sistemas, a adoção destas metodologias vem se tornando cada vez mais indispensável para que se alcancem melhores resultados.

Dentro de um projeto, existem papéis e responsabilidades e estes, são a base para a execução do processo. Estamos falando das pessoas, dos membros da equipe os quais serão responsáveis por diferentes partes e em diferentes momentos no decorrer do projeto.

Depois de entendermos as metodologias, os papéis de cada um, iremos contextualizar a aplicação do papel do Gerente de Projetos e do ScrumMaster em um projeto de desenvolvimento de software. Desta maneira, os executivos de sua organização terão uma ideia clara do perfil mais adequado a ser contratado de acordo com as suas necessidades, sua realidade operacional.

O papel do gerente de projeto, qual o objetivo das metodologias ágeis, o papel do ScrumMaster e, em uma abordagem prática, verificar onde os papéis são corretamente aplicados para a execução de projetos de engenharia de software.

Definição de Projeto

Segundo o PMBOK (Project Management Body of Knowledge – livro reconhecido como o Corpo de Conhecimentos sobre Gestão de Projetos), publicado e mantido pela instituição americana PMI (Project Management Institute), em linhas gerais, projeto é um esforço temporário empreendido para alcançar um objetivo específico. Projetos são executados por pessoas, geralmente têm limitações de recursos e são planejados, executados e controlados. Um projeto cria um produto, serviço ou resultado específico.

Construir uma nova estrada, uma praça de pedágios, um novo sistema de gestão de recursos, uma escola, um novo serviço de atendimento, são bons exemplos de projetos. Neste momento, é importante ressaltar a diferença entre Projeto e Operação. Operação é uma ação em si, o ato de fazer alguma coisa de forma padronizada e repetitiva, são as ações que sustentam as atividades comerciais.

Outro exemplo de projeto, aquele que objetiva alcançar um resultado específico pode ser considerado como um projeto de melhoria. Um projeto sobre algo que já existe e que pode ser modificado de forma a gerar melhores resultados ou atender a mudanças de necessidades (requisitos).

Agora que o termo projeto está definido, precisamos entender como os projetos são executados / conduzidos. Para tanto, é preciso entender as metodologias voltadas para este tema. Metodologias são conjuntos de processos (ações) e ferramentas que auxiliam na execução de uma determinada atividade.

Visão geral sobre Gerenciamento de Projetos

De acordo com o PMBOK, Gerenciar Projetos é a aplicação de um conjunto de conhecimentos, habilidades, ferramentas e técnicas de forma que as atividades do projeto alcancem seus requisitos. Significam os itens imprescindíveis os quais aumentam as chances de sucesso dos projetos de atingirem os objetivos estabelecidos.

O PMBOK estabelece que o Gerenciamento de Projetos deve ser realizado através da correta aplicação e integração de

47 processos organizados e agrupados em cinco grupos de processos. Os cinco grupos de processos são:

- Iniciação;
- Planejamento;
- Execução;
- Monitoramento e Controle;
- Encerramento.

Gerenciar projetos comumente inclui (mas não se limitam) a:

- Identificar requisitos;
- Endereçar as diversas necessidades, preocupações e expectativas das Partes Interessadas (Stakeholders) durante o planejamento e execução do projeto;
- Estabelecer, manter e distribuir a comunicação entre as partes interessadas, de forma colaborativa e eficaz;
- Gerenciar as Partes Interessadas (Stakeholders) no sentido de cumprir os requisitos do projeto e criar os entregáveis (as entregas) do projeto;
- Balancear as limitações concorrentes do projeto, que incluem (mas não se limitam) a:
 - Escopo;
 - Qualidade;
 - Prazo;
 - Orçamento;
 - Recursos;
 - Riscos.

Nota

As limitações do projeto (constraints) estão completamente interligadas e devem estar totalmente balanceadas. Caso uma destas limitações seja afetada, automaticamente as demais também serão. Por exemplo, se o Escopo for ampliado, consequentemente, será necessário mais tempo (prazo), aumento nos processos de qualidade, aumento no custo (orçamento), aumento do uso dos recursos e, aumentam-se os riscos.

Contexto do Gerenciamento de Projetos

De acordo com o PMBOK, podemos ampliar a visão da Gestão de projetos agrupando os projetos através de Programas, Portfólios e o Escritório de Projetos, mais conhecido como PMO (Project Management Office).

Programas é uma forma de agrupar projetos relacionados gerenciados de uma maneira coordenada, com objetivo de obter benefícios e controles que não seriam possíveis caso fossem gerenciados individualmente. Para gerir os Programas, existe o conceito de Gerenciamento de Programas. Um bom exemplo de programa seria o desenvolvimento de um novo modelo de carro, onde podem ser criados projetos independentes para seus componentes como motor, cambio, freios, etc. Os programas ainda podem ser decompostos em subprogramas. Os subprogramas geram projetos mais especializados de forma que possam ser mais bem gerenciados.

Portfólios são conjuntos de projetos ou de programas os quais são agrupados para facilitar o gerenciamento eficaz do trabalho para alcançar os objetivos estratégicos da organização. Voltando ao exemplo do programa de um novo modelo

de carro, o portfólio pode ser criado para novos modelos de carro do tipo utilitários, ou novos motores, etc. Os Portfólios também podem ser organizados e subdivididos em sub portfólios, de maneira a facilitar a gestão e aumentar a eficiência de seus resultados.

O escritório de Projetos, o PMO, é uma estrutura gerencial cuja função é a de padronizar os processos de governança relacionados a projetos. Esta estrutura facilita o compartilhamento de recursos, metodologias, ferramentas e técnicas. Dentre as responsabilidades do PMO estão, desde o suporte aos gerentes de projeto, fornecer treinamento e capacitação, gerir conflitos e até mesmo, responder pela gestão de projetos, portfólios ou programas.

Vantagens em se trabalhar com Projetos

A pressão comercial para que as empresas tenham mais lucratividade, reduzindo custos e aumentando a eficiência, potencializado pelos fatores da globalização remetem a necessidade das empresas serem mais ágeis. A estrutura tradicional é altamente burocrática e a experiência tem mostrado que essa estrutura não pode responder de modo suficientemente rápido a um ambiente em constante mudança. Por isso, a estrutura tradicional deve ser substituída pelo gerenciamento de projetos, ou outra estrutura temporária de gestão que seja altamente orgânica e que possa responder muito rapidamente conforme as situações se apresentarem internamente e externamente à empresa. O gerenciamento de projetos tem sido amplamente discutido por executivos e acadêmicos como uma das possibilidades exequíveis para modelos organizacionais futuros que poderiam integrar esforços complexos e reduzir a burocracia.

Uma empresa tem como opção trabalhar de forma projetizada (orientada a projetos). Originalmente, as empresas trabalhavam de forma funcional. Uma vez que definimos o conceito de projetos e a maneira com que eles devem ser tratados pelas organizações, cabe o entendimento de como as organizações estão estruturadas e qual deve ser a abordagem para a adoção de uma estrutura orientada a projetos.

Saiba identificar o tipo da Organização onde você trabalha

As organizações estabelecem suas estruturas de forma a obterem uma melhor maneira de administrar seus recursos humanos, materiais e financeiros possibilitando atingir de forma mais eficiente seus objetivos comerciais. Basicamente, existem três estruturas organizacionais: funcional, projetizada e a matricial.

A estrutura organizacional funcional prega que cada funcionário possui um superior e que as equipes estão organizadas por função exercida (por exemplo: finanças, produção, jurídico). Empresas de menor porte estão mais suscetíveis a adotarem esta estrutura funcional como um posto de gasolina, borracharia ou um pequeno comércio.

Na estrutura organizacional projetizada, existem os departamentos e cada um deles possui um gestor de projetos para

conduzir seus próprios projetos contando com o apoio de outros departamentos. Esta abordagem traz à tona a questão da concorrência dos recursos dos departamentos de apoio, sendo requisitados simultaneamente pelos gerentes de projetos dos demais departamentos da empresa. Por exemplo, a pessoa responsável pelas aquisições de uma empresa, inevitavelmente, receberá solicitações de compra de diversos gerentes de projetos. Por vezes, receberá solicitações de compras de um mesmo material porém com dias de diferença, fazendo com que o comprador repita, desnecessariamente, todo o processo de compra.

Já a estrutura organizacional matricial propõe a mistura entre as estruturas funcional e projetizada. Desta forma, o projeto será conduzido conforme seu grau de relevância, alocando os recursos funcionais necessários.

A estrutura matricial pode ainda ter três aspectos: matricial fraca, matricial forte e a matricial balanceada. Para entender melhor cada uma delas, vamos explicar como estão organizadas cada uma delas a seguir.

A estrutura matricial fraca mantém o Gerente Funcional (responsável por um setor, unidade organizacional ou grupo de pessoas) com o maior nível de autoridade. O gerente de projetos está subordinado ao gerente funcional e depende dele e dos demais gerentes funcionais para disponibilizarem os recursos humanos e materiais para a execução dos projetos.

Já na estrutura matricial forte, o Gerente de Projetos é que tem maior nível de autoridade, alocando os recursos humanos e materiais necessários ou até contratá-los externamente a fim de garantir a execução e entrega dos resultados dos projetos.

A estrutura matricial balanceada busca, como o próprio nome diz, balancear as necessidades funcionais e dos projetos dentro da empresa, ou seja, o Gerente Funcional e o Gerente de Projetos possuem o mesmo grau de autoridade. Esta estrutura é pouco utilizada dada a complexidade em sua gestão.

O contexto dos projetos

O PMBOK não é considerado como uma metodologia. Apesar do mercado já ter se habituado a chamar de Metodologia de Gestão de Projetos PMI, o foco do PMBOK é concentrar um conjunto de boas práticas e processos, de forma genérica. Fica sob responsabilidade de cada empresa construir sua própria metodologia a qual pode ou não seguir as recomendações que constam no PMBOK. Para ser considerada uma metodologia, a abordagem do PMBOK deveria estabelecer princípios e regras, estabelecendo como exatamente os processos deveriam ser executados quando na verdade, o PMBOK sugere processos, ferramentas e técnicas de forma que a empresa estabeleça as suas próprias regras para executá-los.

Considerando o exposto, passamos a estabelecer como premissa que as empresas, aquelas que optaram em trabalhar por projetos, devam manter uma estrutura Matricial Forte. Geralmente, a alta direção da empresa determina uma necessidade interna ou de um cliente, algo precisa ser criado objetivando um resultado específico. Pode ser construir uma nova loja, ampliar instalações, criar uma nova área, implantar

um sistema de gestão (ERP), construir um novo sistema, elaborar um novo site ou uma grande mudança visando atender a objetivos estratégicos da empresa. Ao solicitante ou cliente do projeto dá-se o nome de patrocinador. É o patrocinador que irá usufruir dos resultados alcançados no projeto e, aquele que paga pelo projeto.

Mas, não basta saber o que se quer, ter em mente o objetivo esperado é apenas parte do projeto. A partir deste momento, existe uma pessoa a qual deverá transformar este objetivo em um projeto e o profissional mais indicado é o gerente de projetos.

Como nosso enfoque é a área de desenvolvimento de software, iremos assumir que trataremos de um projeto de um novo sistema (software). Desta forma, do ponto de vista prático, a alta direção da empresa precisará saber, basicamente, para entrega deste novo sistema a ser desenvolvido, quanto tempo irá levar, quanto custará, como saberá se o novo sistema atenderá às expectativas, quantas etapas ou fases, como e quando serão informados do avanço do projeto, quais e quantas pessoas serão envolvidas, como os problemas serão encaminhados e resolvidos, enfim, serão necessários documentos que definirão todos estes pontos, um informe executivo periódico entre outros documentos.

Digamos que este é o modo formal, ou melhor, profissional de se gerenciar projetos conforme consta no PMBOK. O patrocinador e todas as partes interessadas (pessoas afetadas direta ou indiretamente pelo projeto) saberão de todo o andamento, das suas atividades e responsabilidades, a partir das informações, documentos e relatórios emitidos pelo Gerente de Projetos. Percebam que esta forma de condução de projetos é generalista ao ponto de poder ser adotada em projetos de qualquer natureza, desde a construção de uma ponte, de criar uma nova linha de montagem em uma indústria automotiva e, é neste ponto que se destacam as boas práticas sugeridas no PMBOK (PMI) e a necessidade de se criar uma metodologia dentro da empresa a qual estabeleça esta cultura de projetos para que todos os envolvidos estejam integrados em todos os momentos do projeto, entendendo o que está sendo, o que será feito e, principalmente, o que está sendo dito e informado sobre o projeto. Este é o contexto de um projeto.

Entendendo os conceitos de Engenharia de Software

Bom, já sabemos o que é um projeto e como ele será conduzido, mas em se tratando de desenvolvimento de software, existe ainda a necessidade de se saber construir software e esta, é uma área bastante abrangente, pois existem diversas maneiras de se fazer software. Digamos que no início da informática, a construção de software estava focada em resolver questões muito científicas e específicas tais como cálculos balísticos, para fins militares. Os computadores não estavam acessíveis ao público em geral, apenas em centros governamentais e científicos como universidades.

Com a popularização da informática, o mundo percebeu a capacidade em utilizá-la de forma a reduzir as atividades repetitivas e, nesta ocasião, surgiram os chamados programadores,

pessoas as quais detinham o conhecimento para instruir (programar) um computador, através de uma linguagem específica para executar as tarefas determinadas. Nesta ocasião, não existiam métodos ou normas as quais orientasse a construção destes programas.

Dada a falta de métodos e de profissionais frente a crescente demanda, no final da década de 1960, instaurou-se a chamada Crise do Software. Para normalizar a prática de desenvolvimento de sistemas, surge a Engenharia de Software trazendo consigo os princípios da engenharia tais como: criação, construção, análise, desenvolvimento e manutenção, com um tratamento mais sistemático e controlado. Com a engenharia de software torna-se possível especificar, projetar, planejar, construir, aferir e garantir a qualidade dos resultados.

Com a evolução da engenharia de software, foram criados os chamados modelos de processo de software. Estes modelos visam a melhor representação do gerenciamento do processo de software, trazendo maior visibilidade. Os modelos basicamente determinam o ciclo de vida. Ciclo de vida, como o próprio nome indica, significa a representação de todas as fases desde a concepção até a finalização do software entregue, quando este passa a ser tratado como um produto e será mantido em um regime de operação de manutenção. São exemplos de modelos de processo:

- Sequencial ou cascata;
- Desenvolvimento iterativo e incremental;
- Evolucionar ou prototipação;
- Modelo em "V";
- Espiral;
- Componentizado;
- Formal;
- Ágil;
- RAD.

Podemos verificar que os modelos foram evoluindo na medida em que o tempo e a maturidade foram passando. Conforme foi dito anteriormente, estes modelos foram criados para melhorar a visibilidade do gerenciamento do processo de software bem como do gerenciamento do projeto. Esta trajetória de evolução dos modelos passa por diversos níveis de exigência, como geração de documentos, processos de aprovação e validações entre fases. O melhor exemplo é o modelo sequencial (ou cascata) onde uma fase do processo de desenvolvimento de software, só pode ser iniciada após a conclusão e aceite da fase anterior. Este modelo sequencial (cascata) está representado na **Figura 2**.

Já no modelo iterativo e incremental a proposta é a de que as fases que não possuam dependências diretas podem ser iniciadas em paralelo desde que as aprovações (aceites) ocorram formalmente. A melhor representação deste modelo é demonstrada através do Processo Unificado (Unified Process). O Processo Unificado propõe a construção do software em partes menores que, ao final, serão integrados para formar um grande sistema. Este padrão de desenvolvimento permite possuir mais equipes trabalhando em diferentes partes do software e de forma integrada. A **Figura 3** representa o processo unificado (PU).

O PU foi evoluído e popularizado através da empresa Rational (agora pertencente à empresa IBM) através da criação do chamado Rational Unified Process (RUP). A Rational desenvolveu um conjunto de ferramentas as quais representavam todas as fases de forma iterativa com a equipe de desenvolvimento de software onde, além de visualizar os processos era possível armazenar os artefatos gerados como um repositório e ainda navegar por todo seu conteúdo tal qual um website. Este padrão adota o processo de levantamento e gestão de requisitos (as necessidades do solicitante), ou as funcionalidades que serão promovidas pelo software na forma

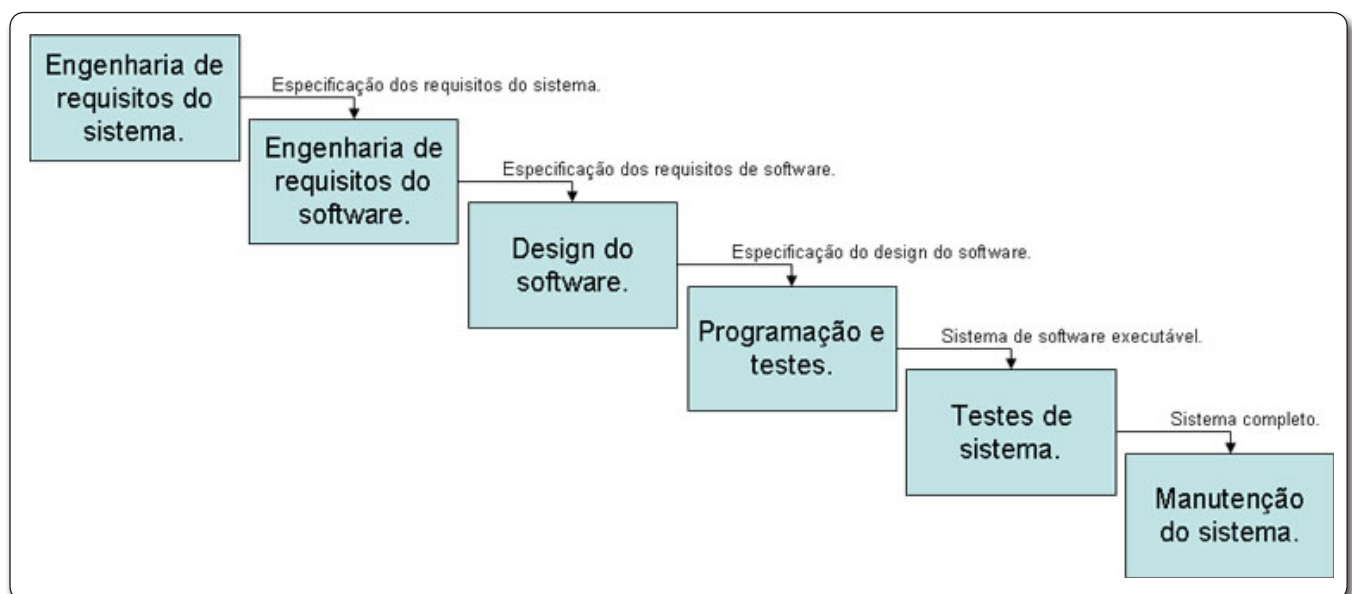


Figura 2. Modelo Sequencial (Cascata)

de casos de uso. Casos de uso são documentos criados para descrever estes requisitos e todas as suas validações como regras de negócio, atores (pessoas envolvidas na utilização do software), cenários (como a funcionalidade deverá ser tratada), fluxos, prototipação, relacionamento com casos de uso, entre outros assuntos que auxiliem no entendimento do que está sendo solicitado.

Sendo assim, o ponto central do modelo PU está justamente na correta descrição das funcionalidades através dos requisitos e casos de uso gerados. São documentos que precisam de levantamentos realizados por um analista o qual irá traduzir as necessidades dos solicitantes em um formato adequado para a equipe de desenvolvimento do software.

Outro fator interessante foi que o Rational Unified Process (RUP) inseriu a abordagem de gerenciamento de projetos conforme estabelecida pelo Project Management Institute (PMI). É neste momento que a gestão de projetos e a área de Engenharia de Software se unem para oferecer, cada vez mais, visibilidade para as partes interessadas e os patrocinadores do projeto. Em resumo, tornou-se claro que o desenvolvimento do software fazia parte de um projeto.

Os modelos de processos e os padrões associados aos modelos de desenvolvimento de software estavam concentrados em guiar as atividades e normatizar o desenvolvimento em si. Por outro lado, um projeto pode possuir mais de um software a desenvolver e ainda cuidar de outras atividades relacionadas ao projeto como a aquisição de um imóvel, mobiliário e infraestrutura para receber a nova equipe que irá operar o novo software que está sendo desenvolvido no projeto.

Ainda em se tratando de Modelos Iterativos, existem também os modelos Ágeis (Agile) nos quais são oferecidos um conjunto de práticas que permitem a produção de software de maneira mais rápida e eficaz. A base do desenvolvimento ágil consiste em um manifesto, chamado de Manifesto Ágil. A partir deste

manifesto ágil surgiram várias metodologias tais como Scrum, Extreme Programming (XP), entre outras.

O conteúdo deste Manifesto diz:

“Manifesto para Desenvolvimento Ágil de Software

Estamos descobrindo maneiras melhores de desenvolver software, fazendo-o nós mesmos e ajudando outros a fazerem o mesmo. Através deste trabalho, passamos a valorizar:

- **Indivíduos e interações** mais que processos e ferramentas
- **Software em funcionamento** mais que documentação abrangente
- **Colaboração com o cliente** mais que negociação de contratos
- **Responder a mudanças** mais que seguir um plano

Ou seja, mesmo havendo valor nos itens à direita, valorizamos mais os itens **à esquerda**.”

Tomando como base o que descrevemos anteriormente sobre o PU, percebemos a ruptura na necessidade de tantos processos e iterações, documentos e aprovações quando na realidade o objetivo é a entrega de um software funcionando. Construir um software o qual atenda às expectativas deve ser feito através do contato direto entre a equipe de desenvolvimento e o próprio solicitante (cliente).

É importante lembrar que esta forma de construir software só é possível graças à evolução e maturidade das tecnologias e da cultura das pessoas envolvidas. Nas décadas de 90 e início dos anos 2000, podemos dizer que a tecnologia até viabilizaria a adoção de uma metodologia ágil mas a cultura das pessoas, principalmente daquelas pessoas que contratavam a construção de um software, estava muito longe. Tomemos como exemplo a compra de um carro. Seria ótimo poder chegar numa concessionária e dizer todos os requisitos desejados em

um novo carro e, numa linha de montagem, estes requisitos iriam sendo adicionados como rádio, bancos de couro, ar condicionado, entre outros mas, e quando chegasse ao ponto da escolha do motor? Câmbio? Sistema de Freios? Tração? Suspensão? Neste ponto seriam exigidos conhecimentos muito mais profundos e técnicos os quais, nem todas as pessoas possuem. Em se tratando de software, seria a mesma situação onde o solicitante poderia ser questionado sobre temas que ele não conhece e nem pretende conhecer devido a sua própria formação ou profissão.

Retornando ao ponto sobre os processos e modelos, quando abordamos o tema dos processos

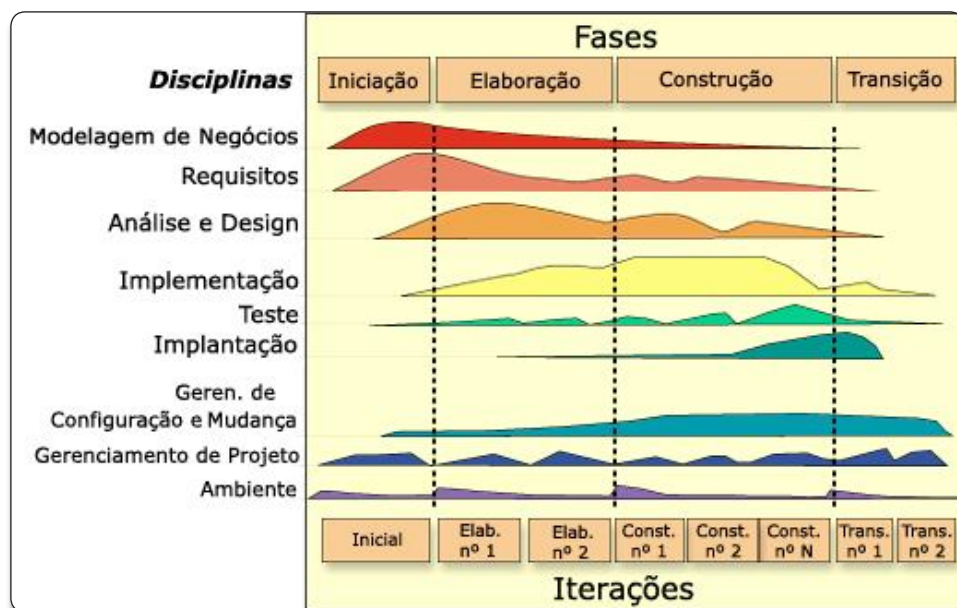


Figura 3. Modelo Iterativo (Unified Process)

ágeis, mais uma vez, temos a clara visão de que a construção do software possui necessidades específicas para garantir mais flexibilidade e qualidade na entrega do software funcionando, que o software funcione conforme as expectativas de quem o solicitou. E o projeto? Qual seria o papel do projeto sendo que na verdade o item desejado é o software entregue e funcionando. O papel do projeto é de disponibilizar controles e envolvimento das partes interessadas a fim de garantir que todo o processo de desenvolvimento de software, seja ele ágil ou não, funcione perfeitamente.

Um projeto formal pode ser ágil?

A resposta é sim, definitivamente. A base de um método ágil é o de abolir burocracias que dificultam atingir um determinado objetivo. Em nosso caso, estamos falando de construir software e é totalmente possível entregar software, de qualidade, funcionando conforme as expectativas de forma ágil e documentada de maneira mínima e necessária.

A Engenharia de Software, com seus processos e modelos permitiu o surgimento dos chamados modelos de maturidade. Maturidade se resume na adoção das reconhecidas boas práticas da engenharia de software (metodologias, processos, ferramentas e técnicas) aplicadas pelas empresas que produzem software. O modelo de maturidade mais reconhecido mundialmente é o CMMI (*Capability Maturity Model Integration*) criado pelo Software Engineering Institute – SEI.

Como referência em termos da maturidade no desenvolvimento de software, o modelo CMMI aceita plenamente a adoção de metodologias ágeis desde que estas não interfiram ou deixem de lado o cumprimento das boas práticas como registros e controles.

O modelo CMMI fornece um conjunto de boas práticas geralmente aplicáveis em grandes projetos que possuam altos níveis de risco e ainda, oferece um conjunto de práticas de gestão e suporte para implantação dos métodos ágeis nas empresas, independentemente do tamanho dos projetos. Não cabe neste artigo o detalhamento sobre o Modelo CMMI. Recomendamos o artigo “CMMI: Uma visão Geral” publicado na edição número 44. O importante é ter em mente o fato de que a maturidade pode e deve andar em conjunto com a agilidade.

Da mesma forma que o modelo CMMI prevê e apoia a adoção das metodologias ágeis, o PMI (Project Management Institute) também as reconhece. Em 2011 o PMI lançou uma nova certificação voltada aos praticantes dos métodos ágeis chamada de PMI-ACP (*Agile Certified Practitioner* ou Praticante de Agile Certificado). Esta certificação visa demonstrar ao mercado (empregadores) o nível de profissionalismo em práticas Ágeis na gestão de projetos pelo praticante (profissional), aumentar a versatilidade do profissional em ferramentas e técnicas de gestão de projetos, aliada com a capacidade de liderança de equipes ágeis e a disponibilização de uma estrutura voltada à treinamentos e iniciativas de desenvolvimento profissional.

FÓRUM DEVMEDIA

O lugar perfeito para você ficar por dentro de tudo o que acontece nas tecnologias do mercado atual



No fórum da DevMedia você irá encontrar uma equipe disponível e altamente qualificada com consultores e colaboradores prontos para te ajudar a qualquer hora e sobre qualquer assunto. Temos as salas de Java, .NET, Delphi, Banco de Dados, Engenharia de Software, PHP, Java Script, Web Design, Automação comercial, Ruby on Rails e muito mais!

ACESSE AGORA
www.devmedia.com.br/forum



Em resumo, as práticas ágeis podem e devem ser associadas a processos formais de gestão de projetos. Isto não significa abrir mão da rapidez que as práticas ágeis oferecem sem perder o controle, qualidade e visibilidade oferecidas pela gestão de projetos formal. Porém, tanto as práticas ágeis bem como o gerenciamento de projetos formal, por si só, não garantem a eficiência e eficácia do resultado. Precisamos separar as áreas e demonstrar o funcionamento destas práticas para elucidar o contexto de desenvolvimento de software na vida real.

Gestão de Projetos e Métodos Ágeis na prática

Falamos anteriormente da existência das empresas e como estas estão normalmente estruturadas. Agora é necessário que o leitor identifique se a empresa em que está trabalhando é do tipo funcional, projetizada ou matricial para enxergar os paralelos e comparações a seguir. Vamos estabelecer um cenário onde apresentaremos a aplicação dos conceitos de gestão de projetos e práticas ágeis.

Você trabalha numa empresa voltada ao desenvolvimento de software, é do tipo matricial forte onde o gerente de projetos possui a autoridade necessária para acionar os demais gerentes funcionais e garantir que os projetos sejam executados seguindo a metodologia definida pela empresa. Esta metodologia inclui a gestão de projetos formal (segundo as recomendações do PMBOK do PMI) bem como as boas práticas da área de engenharia de software, entre elas o uso das práticas ágeis do Scrum conforme sugeridas pelo modelo CMML. Não entraremos no mérito da empresa estar ou não certificada em um dos níveis de maturidade do CMML.

Sua empresa venceu uma concorrência para desenvolver um software de gestão para um cliente do ramo de mineração. Este projeto foi concebido a partir da apresentação de uma solicitação de proposta (RFP – *Request For Proposal*) onde foram recebidas as informações como escopo, prazo, qualidade. Esta solicitação foi recebida pela área comercial da sua empresa e, foi encaminhada ao time de orçamento. Este time de orçamento é composto por membros da equipe comercial, do gerente de projetos e até da alta direção da sua empresa.

O Gerente de Projetos em conjunto com as equipes funcionais fizeram a análise das solicitações da empresa de mineração, definiram um plano de projeto contendo: escopo, premissas, restrições, custos, riscos, controle e garantia da qualidade, controle de solicitações de mudanças, plano de comunicação, gestão das partes interessadas, controle dos contratos de terceirizadas, gestão dos recursos, enfim, todos os tópicos descritos na metodologia da empresa que tomaram como base o PMBOK. E além disso, irão adotar a metodologia ágil Scrum. Neste documento de plano de projeto, já se sabe, em linhas gerais, o que precisa ser feito e como, quando será feito e por quem. De posse destas informações, a equipe comercial pode gerar uma proposta contendo valores e condições.

Chega a notícia de que a sua empresa foi escolhida pela empresa de mineração para desenvolver o sistema de gestão solicitado por ela. Neste momento, o Gerente de Projetos entra em cena para alocação dos recursos e dar início ao projeto.

Faz-se uma revisão do plano do projeto para garantir que tudo o que estava na proposta comercial está contemplado no plano e então é feita a reunião de início (Kick-off). As providências administrativas são iniciadas, os recursos alocados e, o time de desenvolvimento faz a revisão das histórias liderados pelo ScrumMaster, organizam o backlog de histórias e planejam as Sprints.

Para iniciar o desenvolvimento do sistema, a primeira coisa que o time de desenvolvimento precisará é do envolvimento do cliente (as pessoas elencadas pela empresa de mineração) para refinar e desenvolver as histórias. Neste meio tempo, o gerente do projeto está providenciando contratos, contratações de pessoas e materiais necessários para a execução do projeto.

Durante o planejamento da Sprint, o ScrumMaster e a equipe de desenvolvimento não conseguem total disponibilidade do cliente. Um impedimento é registrado na reunião e o ScrumMaster aciona o Gerente do Projeto por telefone. O Gerente do Projeto, verifica no plano do projeto e identifica no plano de comunicação e na matriz de responsabilidades quem é a pessoa responsável do lado do cliente em resolver este problema. O Gerente de Projeto entra em contato com esta pessoa, explica a situação e solicita um encaminhamento. Passado um breve período, o cliente assume que não havia entendido corretamente e que os recursos estarão disponíveis no próximo dia.

Conforme o entendimento telefônico, o gerente de projetos registra a ocorrência no formato previsto da metodologia da empresa e acompanha a solução do impedimento junto ao ScrumMaster. No dia seguinte as pessoas definidas pelo cliente chegam à empresa e desempenham o papel necessário para iniciar as atividades de desenvolvimento junto ao ScrumMaster e a equipe. O impedimento é removido, registrado na reunião diária e o processo de desenvolvimento segue, sem maiores problemas.

De acordo com a evolução das sprints, o ScrumMaster envolve e reporta o Gerente de Projetos quanto ao andamento e, o Gerente de Projetos por sua vez emite os relatórios e os documentos de aceite obrigatórios na liberação dos pagamentos e início das próximas fases. O ScrumMaster e a equipe de desenvolvimento também registram as mudanças no backlog, as reuniões de retrospectiva, planejamento das sprints seguintes.

Ao final do prazo acordado, a equipe de desenvolvimento da sua empresa entrega o sistema de gestão em funcionamento, conforme as expectativas definidas pela empresa de mineração e o projeto é dado como finalizado com sucesso, entrando em produção e em operação gerando os benefícios esperados. Enfim, temos um caso de projeto bem sucedido utilizando o processo formal e o ágil de desenvolvimento de software em harmonia.

Entendendo os Papéis do Gerente de Projetos e do ScrumMaster

Com o sucesso comprovado de resultados ao se adotar as metodologias ágeis, algumas empresas passaram a entender que o papel do ScrumMaster seria suficiente para executar os projetos. Da mesma forma que outras empresas também

passaram a entender que um Gerente de Projetos também poderia assumir a função de ScrumMaster. Diante da novidade e, por vezes, da falta de informações podemos considerar natural que esta confusão entre papéis ocorram.

Segundo o PMBOK (PMI), o papel do gerente de projetos é a pessoa definida por uma organização responsável por garantir que os resultados (objetivos) do projeto sejam alcançados. Diferentemente dos gerentes funcionais (responsáveis por equipes funcionais ou unidades de negócio específicas) ou dos gerentes operacionais (responsáveis por garantir a eficiência das operações comerciais). Um Gerente de Projetos possui responsabilidades e competências próprias. Destacamos entre as responsabilidades do Gerente de Projetos a responsabilidade de atender às necessidades das tarefas, das equipes e dos indivíduos, funcionando como um elo entre a equipe e a estratégia pois, é através dos projetos que as empresas podem crescer e se manterem atendendo as crescentes e variadas demandas do mercado.

Um Gerente de Projeto precisa deter muito mais do que conhecimentos sobre ferramentas e técnicas, deve aliar a prática e ainda possuir habilidades gerenciais (soft skills) tais como: liderança, motivação, possuir boa comunicação, influenciador, tomada de decisão, confiabilidade, gestão de conflitos, orientação, entre outros.

O papel do ScrumMaster, diferentemente do Gerente de Projeto, é o de facilitar o caminho da equipe para que esta possa concluir os objetivos da Sprint, removendo impedimentos, impedindo que influencias externas atrapalhem o andamento das atividades, mantendo o foco da equipe. Além disso, o ScrumMaster é o responsável pela aplicação correta das práticas do Scrum, capacitando e motivando a equipe, promovendo as adaptações necessárias ao processo de desenvolvimento de software. Não precisa necessariamente ser um Líder da equipe de desenvolvimento, mas por outro lado, assume para si o papel de ponto focal entre a equipe e os demais envolvidos no projeto como o Gerente de Projetos e o Cliente.

Percebemos que algumas empresas, ao publicarem vagas e oportunidades de emprego vem exigindo, cada vez mais, que tanto o Gerente de Projetos possua conhecimento e experiência com Scrum e práticas ágeis bem como, exigindo do

ScrumMaster que possua conhecimentos e experiência sobre gestão de projetos.

Entretanto, é importante deixar claro que os papéis do Gerente de Projetos e do ScrumMaster são bem distintos e não deveriam ser confundidos. Do ponto de vista prático, conforme expusemos neste artigo, um Gerente de Projetos possui muitas responsabilidades as quais, vão muito além de cuidar e garantir que o software seja entregue conforme o combinado e, não pode ficar junto da equipe aguardando que impedimentos ocorram para poder resolvê-los.

Da mesma forma, não seria justo que um ScrumMaster deixasse de resolver os impedimentos apresentados pela equipe enquanto produz relatórios ou está em reunião junto ao cliente para aprovação de pagamentos.

Links:

The Standish Group – responsável pela emissão do relatório anual “Chaos Manifesto”

<http://blog.standishgroup.com/>

Relatório Chaos Manifesto 2011

http://versionone.com/assets/img/files/ChaosManifest_2011.pdf

Relatório Chaos Manifesto 2012

<http://versionone.com/assets/img/files/CHAOSManifesto2012.pdf>

Relatório Chaos Manifesto 2013

<http://versionone.com/assets/img/files/ChaosManifesto2013.pdf>

Project Management Institute (PMI)

www.pmi.org

Sobre a Certificação PMI-ACP (Agile Certified Practitioner)

http://www.pmi.org/Certification/~/_media/Files/PDF/Agile/PMI_Agile_Certification_Content_Outline.ashx

Você gostou deste artigo?

Dê seu voto em www.devmedia.com.br/esmag/feedback

Ajude-nos a manter a qualidade da revista!

