

## MODELO PARA GESTÃO DE PROJETOS DE DESENVOLVIMENTO DISTRIBUÍDO DE *SOFTWARE* EM UMA EMPRESA DE AUTOMAÇÃO BANCÁRIA<sup>1</sup>

**Laine Aparecida TAVARES<sup>2</sup>**

Pós-graduada em Gestão da Tecnologia da Informação  
IFSP/Câmpus São Paulo

**José Braz de ARAUJO<sup>3</sup>**

Doutor em Administração/UNINOVE  
Docente do IFSP/Câmpus São Paulo

### RESUMO

O desenvolvimento distribuído de *software* tem sido uma abordagem muito utilizada por grandes organizações para o aproveitamento de mão de obra especializada e de baixo custo, entretanto o gerenciamento de projetos neste ambiente ainda é pouco estudado. Neste artigo, é apresentado um estudo de caso realizado em uma empresa do setor de automação bancária que desenvolve seus projetos utilizando equipes locais em diversos países com o propósito de avaliar os modelos de gerenciamento de projetos PMBOK e SCRUM, verificando sua aderência ao ambiente de desenvolvimento distribuído. O estudo explora o modelo de gerenciamento de projetos adotado pela empresa, identificando os processos do guia PMBOK e SCRUM empregados. A análise dos resultados demonstra a importância da inclusão de processos de ambos os modelos avaliados de forma complementar, para aumento da efetividade do gerenciamento do desenvolvimento distribuído de *software* e a consequente melhoria de desempenho dos projetos.

**Palavras-chave:** Gerenciamento de projetos. PMBOK. SCRUM. Desenvolvimento distribuído de *software*.

### Introdução

O setor de automação bancária é um dos únicos setores da economia brasileira em que o desenvolvimento tecnológico é tão avançado quanto em setores estrangeiros equivalentes, atingindo alto grau de excelência, como atesta Diniz (2004). Esse desenvolvimento da automação bancária brasileira foi consequência da demanda dos bancos por uma tecnologia cada vez mais avançada, que permitiu a oferta de serviços a

---

<sup>1</sup> Resultado de monografia apresentada para obtenção da certificação de especialista no Curso de Pós-Graduação *Lato Sensu* em Gestão da Tecnologia da Informação, do IFSP-Câmpus São Paulo, sob orientação do Prof. Dr. José Braz de Araújo.

<sup>2</sup> Endereço eletrônico: lainetavares2014@gmail.com

<sup>3</sup> Endereço eletrônico: jose.ba@ifsp.edu.br

um crescente contingente de clientes, transações e postos de atendimento, com velocidade, qualidade e segurança. (FONSECA; MEIRELLES; DINIZ, 2010).

Devido à expressividade de investimentos em tecnologia pelos bancos, com dispêndios que crescem a uma taxa de 6% ao ano desde 2010, conforme dados da FEBRABAN (2015), o mercado de automação bancária se tornou muito competitivo, principalmente, como salienta Iamonti (2014), após a entrada de empresas multinacionais que intensificaram ainda mais a competição entre as empresas do setor. Entretanto, assim como em outros setores da economia, o setor de automação bancária também sofreu pressões para se manter rentável e competitivo diante de um mercado globalizado, o que favoreceu a adoção do modelo de desenvolvimento distribuído de *software*, que, segundo Kiel (2003), consiste em se estabelecer um processo de desenvolvimento de *software* utilizando profissionais ou equipes de profissionais geograficamente dispersos.

A necessidade de baixo custo na obtenção de mão de obra, a possibilidade de trabalho ininterrupto durante as 24 horas do dia, aproveitando as diferenças de fuso horário entre países ou regiões, e a dificuldade de se encontrar todos os profissionais necessários com qualificação em uma mesma região geográfica viabilizaram a estratégia de desenvolvimento distribuído de *software*. (CARMEL; TJIA, 2005).

Essas características peculiares do desenvolvimento distribuído de *software*, ou DDS, como indicam Prikladnicki, Audy e Evaristo (2003), geram riscos que não são comuns ao desenvolvimento centralizado de *software* e que devem ser levados em consideração quando se planeja migrar projetos de *software* para este perfil de desenvolvimento. Herbsleb (2007) aponta que o problema fundamental do desenvolvimento distribuído de *software* é que muitos dos métodos estabelecidos para gerenciar o trabalho de um projeto centralizado são inexistentes ou incompletos para projetos distribuídos.

Nesse sentido, Sommerville (2007) observa que o bom gerenciamento não garante o sucesso do projeto, porém o mau gerenciamento frequentemente causa falhas no projeto, tais como aumento do prazo de entrega, aumento do custo e falha no atendimento aos requisitos. Neste contexto, a questão que surge é: qual modelo de gestão de projetos melhor se adequa ao contexto distribuído de *software* em empresas de automação bancária?

Nessa perspectiva, para delimitar o escopo do estudo, foi realizada a comparação entre o guia PMBOK (PMI, 2013) – que possui uma abordagem tradicional de gerenciamento de projetos e possui mais de 400.000 profissionais certificados em 204 países (PMI, 2016) – e o guia SCRUM (SCHWABER; SUTHERLAND, 2013) – que possui uma abordagem ágil e é o *framework* ágil mais utilizado mundialmente (SCRUM ALLIANCE, 2013).

Na próxima seção, abordamos o gerenciamento de projetos de *software*; na sequência, apresentamos o Guia PMBOK e o Guia SCRUM; em seguida, os procedimentos metodológicos, resultados e discussão, encerrando com nossas considerações finais e referências utilizadas.

### **Gerenciamento de projetos de *software***

Segundo Sommerville (2007), o gerenciamento de projetos de *software* envolve maiores desafios em relação ao gerenciamento de outros tipos de projeto, pois o *software* é um produto intangível, que não pode ser visto ou tocado, logo a visualização do progresso do andamento do projeto deve ser feita por meio de documentação. Não existe, portanto, padronização nos processos de desenvolvimento de *software*, visto que os processos variam entre as empresas, o que torna difícil a previsão de problemas de desenvolvimento e as lições aprendidas em projetos anteriores podem não ser aplicadas em novos projetos devido à unicidade de projetos de *software* de grande porte. Dessa maneira, por causa da inovação inerente aos projetos de *software*, os atrasos de cronograma se tornam comuns neste tipo de projeto.

De acordo com Pressman (2011), o gerenciamento efetivo de desenvolvimento de *software* deve ter foco em 4 Ps: pessoas, produto, processo e projeto. As pessoas envolvidas no projeto de *software* podem ser divididas em cinco grupos: os gerentes seniores, que definem os itens de negócio; os gerentes de projeto, que tem a função de planejar, motivar e controlar os programadores; os programadores, cuja responsabilidade é desenvolver a engenharia do produto de *software*; os clientes, que têm interesse no produto final e especificam os requisitos; e os usuários finais, que operam o *software*.

O produto diz respeito ao estabelecimento e delimitação do escopo do produto que não deve ser ambíguo e ser compreensível por todos os grupos de pessoas interessadas do projeto. O escopo do produto considera o contexto de negócio, quais dados são necessários como entrada e quais dados serão produzidos como saída, qual a função do *software* e qual o desempenho desejado. (PRESSMAN, 2011).

Quanto ao processo de desenvolvimento de *software*, este consiste em um roteiro de passos necessários para a criação de um produto de *software* de qualidade dentro do prazo definido (SOMMERVILLE, 2007). Conforme Pressman (2011), o emprego de projetos com planejamento e controle é a única forma de administrar a complexidade do desenvolvimento de *software*.

Herbsleb (2007) indica que o gerenciamento de projeto no ambiente de desenvolvimento distribuído de *software* (DDS) necessita de uma abordagem diferenciada com relação ao gerenciamento de projetos centralizados. Existem, na literatura, alguns modelos de gerenciamento de projetos para DDS que basicamente propõem a sincronização como principal característica. A sincronização dos processos requer marcos comuns bem definidos e critérios de entrada e saída claros e objetivos (HERBSLEB; MOITRA, 2001). No modelo DDS, o trabalho de desenvolvimento é realizado simultaneamente por muitos indivíduos e equipes, e, em seguida, integrados em um único produto. Peças de *software* precisam integrar e interoperar corretamente, e cronogramas de produção precisam ser sincronizados, criando dependências entre tarefas e pessoas. Sob o ponto de vista técnico, a integração de várias peças de *software* precisa ser coordenada através de mecanismos como a comunicação da equipe, especificações e gerenciamento de configuração. Sob a perspectiva da gestão de projetos, a gestão do tempo é particularmente importante, já que pequenos atrasos em algumas partes do *software* podem gerar atraso em todo o projeto. Do mesmo modo, do ponto de vista do processo de desenvolvimento de *software*, a falta de conformidade com os processos estabelecidos pode ter efeitos negativos sobre os resultados do projeto, tais como erros de requisitos e conflitos de prioridade. (ESPINOSA *et al.*, 2007).

Outro desafio crítico para a gestão de DDS é a construção e manutenção da confiança e coesão dos membros da equipe. (LEMMEX, 2005). Quando os membros da equipe são separados geograficamente, existe a dificuldade do indivíduo para se

familiarizar com as habilidades e hábitos de trabalho de seus colegas, identificar e transferir conhecimentos, quando necessário, principalmente conhecimentos tácitos e desenvolver a consciência contextual. (HERBSLEB; GRINTER, 1999).

Além disso, existe também a restrição à capacidade de conexão emocional, que faz com que as pessoas gostem umas das outras, se ajudem e trabalhem umas pelas outras, desenvolvendo um espírito de equipe. Mesmo em equipes centralizadas, a construção de companheirismo e confiança demanda tempo e ainda mais em equipes distribuídas, gerando um paradoxo: é necessário confiança para trabalho eficiente à distância e é muito difícil desenvolver confiança à distância. Outro fator negativo, na visão de Carmel (1999), é a diferença cultural, uma vez que algumas culturas tendem a confiar mais lentamente que outras.

## **Guia PMBOK**

O guia PMBOK (*Project Management Body Of Knowledge* ou Guia do Conhecimento em Gerenciamento de Projetos) é um conjunto de processos e boas práticas que tem o objetivo de fornecer diretrizes para o gerenciamento de projetos individuais. Entende-se por boas práticas o conjunto de técnicas, ferramentas e habilidades que, quando aplicadas corretamente, propiciam um aumento das chances de sucesso de um projeto. O guia foi produzido pelo PMI (*Project Management Institute* ou Instituto de Gerenciamento de Projetos), uma associação para profissionais de gerenciamento de projetos, sem fins lucrativos e fundada em 1969 nos Estados Unidos. (PMI, 2013).

Segundo Saladis e Kerzner (2009), o guia PMBOK proporciona uma base sólida para planejamento de projetos e introduz processos-chave e um *framework* para atender ao gerenciamento de projeto. Há muitos pontos de vista e perspectivas sobre o PMBOK e há provavelmente uma quantidade significativa de gerentes de projeto que discordam sobre como ele deve ser usado e sobre seu conteúdo. Para os autores, apesar de diferentes pontos de vista, o guia é adição valiosa para qualquer biblioteca de gerentes de projeto.

No guia PMBOK (PMI, 2013), as pessoas ou organizações diretamente envolvidas no projeto são chamadas de partes interessadas, cujos interesses são afetados

pela execução do projeto e podem exercer influência positiva ou negativa sobre o mesmo. Normalmente, as partes interessadas incluem: clientes ou usuários, patrocinador, gerentes de portfólio ou comitê de análise de portfólio, gerente de programas, escritório de projetos, gerentes de projetos, equipe do projeto, gerentes funcionais, gerentes de operações e fornecedores.

O gerenciamento de projeto, de acordo com o PMBOK (PMI, 2013), é dividido em processos cuja aplicação e integração proporcionam o atendimento aos requisitos do projeto. Os processos de gerenciamento de projeto interagem com os processos da organização, portanto, os projetos não devem ser considerados como um sistema fechado e independente, eles recebem dados de fontes internas e externas e entregam funcionalidades à organização. Os processos podem gerar informações tanto para aprimorar o gerenciamento de projetos quanto para aprimorar ativos de processos organizacionais futuros. (MARTINS, 2007).

O Guia PMBOK (PMI, 2013) divide os processos de gerenciamento de projetos em 5 grupos: processos de iniciação, processos de planejamento, processos de execução, processos de monitoramento e controle e processos de encerramento.

Os processos são também agrupados em 10 áreas de conhecimento, que representam as melhores práticas de gestão em cada área especializada do gerenciamento de projeto (PMI, 2013). As áreas de conhecimento gerenciadas são: recursos humanos, integração, escopo, tempo, custo, qualidade, comunicações, riscos, aquisições e partes interessadas.

Kerzner (2014) acentua que o guia PMBOK ainda é apenas um guia e não o conjunto absoluto de todo o conhecimento sobre gerenciamento de projeto. O desenvolvimento desse conjunto de conhecimentos perfeito deveria ser baseado no tamanho, natureza e complexidade dos projetos de uma organização. Por esse motivo, segundo o autor, embora as empresas tendam a criar metodologias de gerenciamento de projetos baseados no guia PMBOK, essas metodologias raramente o implementam completamente.

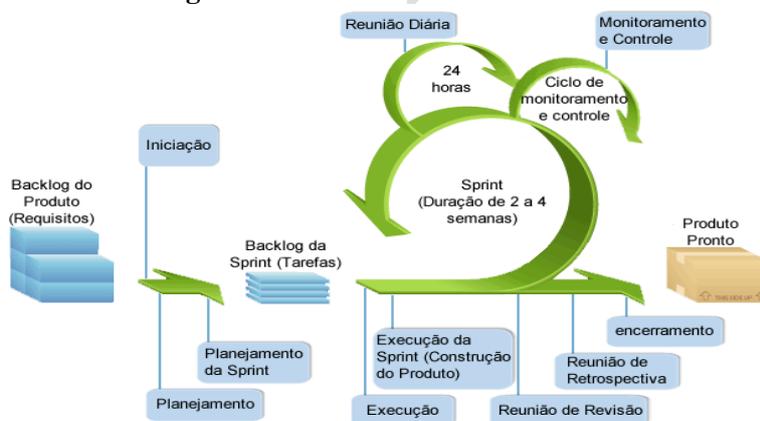
## **Guia SCRUM**

O SCRUM é um *framework*<sup>4</sup> de processos, usado para gerenciar o desenvolvimento de projetos complexos, desde o início da década de 1990. O *framework* SCRUM consiste em equipes e seus papéis, eventos, artefatos e regras. Cada componente dentro do *framework* serve a um propósito específico e é essencial para seu sucesso e uso, como indicam Schwaber e Sutherland (2013).

O guia SCRUM, dentro de sua abordagem incremental, prega a divisão do projeto em pequenos incrementos e cada incremento é desenvolvido em um espaço de tempo pré-definido, que é chamado *sprint*, o principal processo do guia SCRUM. Uma nova *sprint* começa imediatamente após a conclusão da *sprint* anterior. Dentro de uma *sprint*, são previstos eventos, que são agendamentos durante esse período de tempo em que devem ser executadas atividades de gerenciamento de projeto. Os eventos são: planejamento da *sprint*, reunião diária, revisão da *sprint* e retrospectiva da *sprint* (SCHWABER; SUTHERLAND, 2013).

Para Schwaber (2004), o guia SCRUM baseia todas as suas práticas em um processo iterativo e incremental conforme demonstrado na Figura 1.

Figura 1 – Processo Base do SCRUM



Fonte: MARTINS, 2013.

O círculo superior representa a inspeção diária que ocorre durante a iteração, em que os membros da equipe se reúnem para inspecionar as atividades uns dos outros e torná-las versões adequadas. Conduzir a iteração é uma lista de requisitos. Esse ciclo se repete até que o projeto entregue todos os requisitos. No início de uma iteração, a

<sup>4</sup> *Framework* ou arcabouço conceitual é um conjunto de conceitos usado para resolver um problema de um domínio específico (KERZNER, 2014)

equipe analisa o que deve fazer. Em seguida, a equipe seleciona o que pode se transformar em uma entrega (incremento de funcionalidade) até o final da iteração, por meio da divisão dos requisitos do produto (*product backlog*) em unidades menores chamadas estórias do usuário (*user stories*). A equipe conduz o ciclo de interação sem intervenção de gerentes de projeto. No final da iteração, a equipe apresenta o incremento de funcionalidade que foi construído de modo que as partes interessadas possam avaliar o atendimento aos requisitos e as adaptações do projeto possam ser feitas. (SCHWABER, 2004).

O guia SCRUM propõe uma equipe multifuncional e auto-organizada. A equipe é multifuncional porque cada membro participa de todo o ciclo de vida do projeto e é auto-organizada à medida que não há um líder que decide a tarefa de cada membro ou como um problema será resolvido. Essas questões são decididas por toda a equipe. De acordo com o guia SCRUM, a equipe é formada pelo *product owner* (proprietário do produto), o *scrum master* (mestre *scrum*) e a equipe de desenvolvimento. (COHN, 2010).

Os projetos SCRUM requerem menos planejamento do que projetos tradicionais, pois existe visibilidade e transparência durante todo o seu progresso através da apresentação e acompanhamento pelas partes interessadas a cada final de *sprint*. Os projetos SCRUM são também muito complexos e empíricos para serem descritos com muitos detalhes na sua iniciação; em vez disso, o foco desse gerenciamento é a monitoração e orientação da equipe para que sejam entregues os melhores resultados possíveis. (SCHWABER, 2004).

### **Procedimentos metodológicos, resultados e discussão<sup>5</sup>**

A base de nossa pesquisa é exploratória, realizada por meio do estudo de caso em uma empresa multinacional de automação bancária, denominada Organização, tendo instituições financeiras como principais clientes. O gerenciamento de projetos em

---

<sup>5</sup> A apresentação integral dos resultados consta da monografia apresentada para obtenção da certificação de especialista (Pós-Graduação *Lato Sensu* em Gestão da Tecnologia da Informação, do IFSP-Câmpus São Paulo), sob orientação do Prof. Dr. José Braz de Araújo, disponível em <https://spo.ifsp.edu.br/menu/68-menu-principal-pos-graduacao/1053-especializacao-em-gestao-da-tecnologia-da-informacao-producoes-2016>.

ambientes de desenvolvimento distribuído de *software* é um fenômeno relativamente novo e ainda pouco estudado. A pesquisa exploratória configura-se como a melhor opção para estudar esse fenômeno, visto que os estudos exploratórios procuram descobrir variáveis significativas na situação real (GIL, 2002). Segundo Yin (2005), o estudo de caso permite compreender os acontecimentos da vida real na sua totalidade e globalidade, o que o torna uma importante estratégia metodológica para a pesquisa em ciências humanas.

Para realização do estudo de caso, foi executada a análise da documentação gerada pelo departamento de desenvolvimento de *software* global da Organização, para avaliação do modelo de gerenciamento de projetos utilizado pela Organização. Foram analisados os relatórios de desempenho dos projetos e realizadas entrevistas, utilizando-se roteiros aplicados aos membros da equipe de desenvolvimento e gerentes de projeto para a coleta de dados, com vistas à validação da eficácia do modelo de gestão aplicado.

A entrevista foi usada principalmente para identificar a percepção da equipe de gerenciamento de projetos e da equipe de desenvolvimento com relação às dificuldades do gerenciamento de equipes distribuídas. Para este fim, foi utilizado o modelo de entrevista aberta, visando à extração de fatos e opiniões. As questões foram elaboradas de forma a colher informações sobre os seguintes assuntos: comunicação, colaboração, confiança, treinamento e reconhecimento no contexto da equipe distribuída. Objetivando a coleta de informações sob diferentes perspectivas, as entrevistas foram conduzidas, separadamente, com oito gerentes de projetos, sendo sete profissionais locados no Brasil e um profissional locado nos Estados Unidos, todos com experiência em gerenciamento de um número superior a cinco projetos globais de desenvolvimento de *software*, e dez desenvolvedores de *software*, todos locados no Brasil e com experiência em, no mínimo, dois projetos globais de desenvolvimento de *software*.

Conforme proposto por Yin (2005), a validação dos dados coletados foi efetuada por meio da triangulação de dados obtidos de análise documental, entrevistas e descrição dos eventos observados. A coleta e análise dos dados ocorreram concomitantemente e, para isso, foi desenvolvida uma base de dados com todos os documentos coletados, anotações e gravações de áudio das entrevistas realizadas.

A Organização é sediada nos Estados Unidos e possui filiais em vários países da América Latina, Europa, Ásia e África, sendo, na sua maioria, pequenas unidades de

assistência técnica dos equipamentos produzidos. Em alguns países, existem também unidades fabris e centros de engenharia e desenvolvimento de *software* e *hardware*, além das unidades de controle de qualidade de *software* e *hardware* distribuídos em diversos países. Ela iniciou o desenvolvimento distribuído de *software* em 2003 e possui 35 projetos desenvolvidos globalmente ainda ativos. O uso do DDS foi motivado pela diminuição dos custos por meio do uso de mão de obra qualificada e mais barata que a mão de obra disponível em sua sede.

A Organização, ao longo dos anos, efetuou melhorias em seu modelo de gerenciamento de desenvolvimento de *software*. Iniciou as operações de DDS utilizando a metodologia de gestão de projetos locais que se baseava no uso do guia PMBOK. Diante de dificuldades inerentes ao DDS e a dificuldade de comunicação, alguns ajustes foram necessários e a documentação que já era extensa ficou ainda maior devido à necessidade de documentação de todas as comunicações, mesmo as informais, de forma a garantir o perfeito entendimento dos requisitos do projeto por toda a equipe e o cumprimento dos prazos e responsabilidades.

A partir de 2013, iniciou-se a implantação de algumas funcionalidades do *framework* SCRUM no desenvolvimento em ambiente DDS, como o emprego da técnica de *sprint* (iteração) na qual o *software* é dividido em partes menores que devem ser entregues em até 30 dias. Utilizou-se também a técnica de reuniões diárias de curta duração.

A Organização desenvolveu uma coleção de documentos que são guias oficiais para desenvolvimento, documentação, testes e controle que devem ser utilizados e respeitados por todas as equipes globais em todo e qualquer desenvolvimento de *software*. Todas as etapas previstas nos guias são seguidas sem exceções. Os atrasos ou erros de requisitos são tratados com extrema seriedade e burocracia, sujeitos a análises e autorizações em níveis hierárquicos superiores, caracterizando uma forma de evitá-los já que causam um imenso transtorno e custo, pois envolvem uma série de equipes dispersas com fusos horários diferentes e agendas diferentes.

Toda a definição dos processos de gestão de projetos é centralizada na sede da empresa, que é responsável pelas revisões da documentação, treinamento e distribuição para toda a corporação. Nenhuma outra unidade da corporação participa da definição e

melhorias dos processos de gestão de desenvolvimento. Já os papéis de gerentes de projetos são distribuídos entre as equipes globais, conforme a necessidade do projeto.

Para a análise dos dados de projetos coletados, foram consideradas as dez áreas de conhecimento do PMBOK que representam as áreas especializadas do gerenciamento de projeto: recursos humanos, integração, escopo, tempo, custos, qualidade, comunicações, riscos, aquisições e partes interessadas. A partir do estudo de caso dos projetos de DDS da Organização, foram identificadas as melhores práticas de gestão do guia PMBOK e do guia SCRUM utilizados pela Organização. A seguir, apresentamos um quadro-resumo e considerações acerca dos resultados.

**Quadro – Relação de Processos PMBOK e SCRUM utilizados pela Organização**

Área de Gerenciamento	Processos PMBOK Utilizados	Processos SCRUM Utilizados
Recursos Humanos	<ul style="list-style-type: none"> <li>- Desenvolver o plano dos recursos humanos</li> <li>- Mobilizar a equipe do projeto</li> <li>- Desenvolver e gerenciar a equipe do projeto</li> </ul>	<ul style="list-style-type: none"> <li>- Estabelecer times de desenvolvimento com tamanho entre três e nove membros.</li> <li>- Aplicar reconhecimento por equipe</li> </ul>
Integração	<ul style="list-style-type: none"> <li>- Desenvolver o termo de abertura do projeto</li> <li>- Desenvolver o plano de gerenciamento do projeto</li> <li>- Orientar e gerenciar o trabalho do projeto</li> <li>- Monitorar e controlar o trabalho do projeto</li> <li>- Realizar o controle integrado de mudanças</li> <li>- Encerrar o projeto ou fase</li> </ul>	<ul style="list-style-type: none"> <li>- Planejar <i>sprints</i></li> <li>- Revisar <i>sprints</i></li> <li>- Realizar a retrospectiva da <i>sprint</i></li> <li>- Gerenciar o <i>backlog</i> do Produto</li> <li>- Garantir transparência para todas as partes interessadas</li> </ul>
Escopo	<ul style="list-style-type: none"> <li>- Planejar o gerenciamento do escopo</li> <li>- Coletar os requisitos</li> <li>- Definir o escopo</li> <li>- Validar o escopo</li> <li>- Controlar o escopo</li> </ul>	<ul style="list-style-type: none"> <li>- Desenvolver <i>backlog</i> do produto</li> <li>- Desenvolver <i>backlog</i> da <i>sprint</i></li> <li>- Decompor requisitos em estórias</li> <li>- Dividir o projeto em iterações com tempo de duração máxima de 30 dias.</li> </ul>
Tempo	<ul style="list-style-type: none"> <li>- Planejar o gerenciamento do cronograma</li> <li>- Definir as atividades</li> <li>- Sequenciar as atividades</li> <li>- Estimar os recursos e duração das atividades</li> <li>- Desenvolver o cronograma</li> <li>- Controlar o cronograma</li> </ul>	<ul style="list-style-type: none"> <li>- Planejar <i>sprints</i></li> <li>- Decompor requisitos em estórias</li> <li>- Atribuir pontos de estória</li> <li>- Utilizar <i>planning poker</i> para estimativa de esforço</li> <li>- Dividir o projeto em <i>sprints</i> com duração pré-definida</li> <li>- Realizar reuniões diárias do tipo <i>stand up</i></li> </ul>

Área de Gerenciamento	Processos PMBOK Utilizados	Processos SCRUM Utilizados
Custos	<ul style="list-style-type: none"> <li>- Planejar a gestão dos custos</li> <li>- Estimar os custos</li> <li>- Determinar o orçamento do projeto</li> <li>- Controlar os custos</li> </ul>	- - -
Qualidade	<ul style="list-style-type: none"> <li>- Planejar a gestão da qualidade</li> <li>- Realizar a garantia da qualidade</li> <li>- Realizar o controle da qualidade</li> </ul>	<ul style="list-style-type: none"> <li>- Planejar <i>sprints</i></li> <li>- Definir as entregas com consenso sobre os padrões de qualidade</li> <li>- Revisar <i>sprints</i></li> <li>- Realizar reuniões diárias</li> <li>- Realizar a retrospectiva da <i>sprint</i></li> </ul>
Comunicações	<ul style="list-style-type: none"> <li>- Planejar a gestão das comunicações</li> <li>- Gerenciar as comunicações</li> <li>- Controlar as comunicações</li> </ul>	<ul style="list-style-type: none"> <li>- Planejar <i>sprints</i></li> <li>- Realizar reuniões diárias do tipo <i>stand up</i></li> <li>- Revisar <i>sprints</i></li> <li>- Realizar a retrospectiva da <i>sprint</i></li> </ul>
Riscos	<ul style="list-style-type: none"> <li>- Planejar a gestão dos riscos</li> <li>- Identificar os riscos</li> <li>- Analisar os riscos qualitativamente</li> <li>- Planejar as respostas aos riscos</li> <li>- Controlar os riscos</li> </ul>	<ul style="list-style-type: none"> <li>- Planejar <i>sprints</i></li> <li>- Revisar <i>sprints</i></li> </ul>
Aquisições	<ul style="list-style-type: none"> <li>- Planejar a gestão das aquisições</li> <li>- Conduzir as aquisições</li> <li>- Controlar as aquisições</li> <li>- Encerrar as aquisições</li> </ul>	- - -
Partes Interessadas	<ul style="list-style-type: none"> <li>- Identificar as partes interessadas</li> <li>- Planejar a gestão das partes interessadas</li> </ul>	<ul style="list-style-type: none"> <li>- Planejar <i>sprints</i></li> <li>- Revisar <i>sprints</i></li> </ul>

Fonte: Dados da pesquisa

Os dados apresentados demonstram que os processos recomendados pelo SCRUM são insuficientes para atender à demanda por gerenciamento e controle de todas as áreas de gerenciamento de projeto no ambiente de desenvolvimento distribuído de forma a garantir a confiabilidade, integridade e segurança dos produtos de *software*, especialmente em relação à criticidade dos projetos de *software* do setor de automação bancária, que prevê um controle rigoroso de qualidade e riscos.

Entretanto, a característica do SCRUM de orientação a eventos com duração curta e pré-determinada se mostra muito aderente ao gerenciamento de tempo no desenvolvimento distribuído de *software*, já que contribui para a eficácia da sincronização dos trabalhos, principal característica do bom gerenciamento do DDS (HERBSLEB; MOITRA, 2001).

A utilização de processos recomendados pelo SCRUM, adaptados ao modelo DDS, proporcionou melhoria no desempenho dos projetos como consequência dos seguintes fatores: aumento da produtividade da equipe devido à decomposição e distribuição dos trabalhos com entregas mais rápidas e frequentes; melhoria na comunicação e interação entre as partes interessadas devido às reuniões periódicas; melhoria no controle de qualidade dos incrementos propiciada pelas revisões constantes; e melhoria contínua dos processos através das reuniões de retrospectiva ao final das iterações, o que proporcionou ganhos diretos de desempenho nas áreas de gerenciamento de escopo, tempo e qualidade.

Todavia, também foram identificadas dificuldades relacionadas à adoção das práticas recomendadas pelo SCRUM no contexto DDS, destacando-se: dificuldade no agendamento de reuniões diárias com todos os membros necessários por causa das diferenças de fuso horário e dificuldade em cumprir a regra de reuniões de 15 minutos em razão das dificuldades com diferenças de idioma.

Com relação ao guia PMBOK, constatou-se maior aderência de seus processos ao gerenciamento de projetos de DDS da Organização na maioria das áreas de gerenciamento, pois sua natureza preditiva e rígida, com base em extensa documentação, permite a mitigação de riscos associados aos pontos críticos do DDS relacionados por Carmel (1999) que são: falhas de comunicação, dificuldade para transferência de conhecimento, dificuldade para desenvolvimento de consciência contextual e dificuldade de desenvolvimento de confiança e companheirismo entre os membros da equipe.

Os benefícios percebidos pela utilização dos processos indicados no PMBOK no gerenciamento de DDS são: aumento da confiabilidade, devido à padronização dos processos de gerenciamento de projetos; melhor atendimento aos padrões internacionais de segurança impostos pelas instituições financeiras; e melhor controle de custos como consequência da definição de requisitos estáveis.

Os problemas encontrados na aplicação dos processos do PMBOK foram o excesso de responsabilidade sobre o gerente de projeto e o acúmulo de processos na fase de testes de qualidade.

## Considerações finais

Esta pesquisa apresentou o estudo sobre modelos de gerenciamento de projetos de *software* em ambientes distribuídos considerando duas abordagens muito utilizadas atualmente, com o objetivo de auxiliar organizações para melhor aproveitamento das vantagens do desenvolvimento global.

Os resultados obtidos indicam que é possível unir os modelos PMBOK e SCRUM para um gerenciamento efetivo de projetos de desenvolvimento de *software* distribuído, sendo que o bom desempenho do modelo de gerenciamento da Organização pode ser observado por meio dos dados de desempenho dos projetos desenvolvidos pela Organização durante o período de 2011 a 2015. Durante este período, foram realizados 186 projetos, sendo 182 projetos concluídos no prazo previsto.

Nesse ponto, ressaltamos que os dados de desempenho dos projetos mostraram que ocorreram atrasos no final de iterações em 47 projetos, correspondendo a 25,26% do total de projetos concluídos. Esses atrasos foram corrigidos durante as iterações subsequentes, o que demonstra que a divisão do projeto em iterações com revisões constantes e pré-agendadas são eficientes para controle de cronograma.

Em relação aos custos dos projetos da Organização, há uma variação média entre 4% e 10% do orçamento planejado no início dos trabalhos, enquanto que 84,5% das empresas pesquisadas pela Standish Group (2014) responderam que a variação média de custo dos projetos é maior que 20% do orçamento inicial.

Considerando os resultados obtidos nesta pesquisa, pode-se concluir que o modelo de gestão de projetos que melhor se adequa ao contexto de desenvolvimento distribuído de *software* em empresas de automação bancária é um modelo híbrido, composto por elementos do PMBOK e do SCRUM, visto que o modelo híbrido prevê processos preditivos, estáveis e com documentação detalhada, de acordo com o preconizado no guia PMBOK, o que permite melhor controle, sincronização e integração das atividades do projeto, além de maior garantia de controle de riscos e segurança da informação; além disso, processos adaptativos, flexíveis e empíricos, de acordo com o recomendado no guia SCRUM, o que permite aumento de produtividade e desempenho.

Considerando o caso de uso apresentado no estudo, o modelo híbrido foi utilizado pela Organização para gerenciamento de projetos de sistemas críticos que demandam confiabilidade e integridade da informação e que são desenvolvidos sob o modelo de desenvolvimento distribuído de *software*, demonstrando ser efetivo para a entrega de produto final com qualidade e desempenho esperados.

### Referências bibliográficas

CARMEL, E. *Global software teams: Collaborating across borders and time-zone*. Upper Saddle River: Prentice Hall, 1999.

CARMEL, E.; TJIA, P. *Off shoring information technology: Sourcing and outsourcing for a global workforce*. Cambridge: Cambridge University Press, 2005.

COHN, M. *Succeeding with agile software development using scrum*. New Jersey: Addison-Wesley, 2010.

DINIZ, E. H. *Evolução e segmentação no perfil dos serviços bancários pela internet*. São Paulo: FGV, 2004.

ESPINOSA, J. A. *et al.* Team Knowledge and coordination in geographically distributed software development. *Journal of Management Information Systems*, v. 24, n. 1, p. 135-169, 2007.

FEBRABAN - FEDERAÇÃO BRASILEIRA DE BANCOS. *Pesquisa FEBRABAN de tecnologia bancária 2014*. Febraban.org, 2015. Disponível em: <<http://www.febraban.org.br>>. Último acesso em: 14 mar.2016.

FONSECA, C. E. C.; MEIRELES, F. S.; DINIZ, E. *Tecnologia bancária no Brasil: uma história de conquistas, uma visão de futuro*. São Paulo: FGV/RAE, 2010.

GIL, A. C. *Como elaborar projetos de pesquisa*. 4ª ed. São Paulo: Atlas, 2002.

HERBSLEB, J. D. Global software engineering: the future of socio-technical coordination. *IEEE Computer Society*, p. 88-198, 2007.

HERBSLEB, J. D.; GRINTER, R. E. Architectures, coordination, and distance: Conway's law and beyond. *IEEE Software*, n. 16, v. 5, p. 63-70, 1999.

HERBSLEB, J. D.; MOITRA D. Global software development. *IEEE Software*, v. 16, n. 2, p. 16-20, 2001.

IAMONTI, V. Z. Automação bancária no Brasil: O circuito espacial de produção e a difusão dos terminais de autoatendimento (ATMs) pelo território brasileiro no período

da globalização. São Paulo: *Congresso Ibero-americano de Estudos Territoriais e Ambientais*, 2014.

KERZNER, H. *Project management: best practices. achieving global excellence*. New Jersey: John Wiley & Sons, Inc., 2014.

KIEL, L. Experiences in distributed development: A case study. *International Workshop on Global Development at ICSE*, 2003, Oregon. Proceedings... EUA, p. 44-47, 2013.

LEMMEX S. Successfully managing remote teams. In: *Expert reference series of white papers. cary: global knowledge network*, 2005. Disponível em: <[www.globalknowledge.com](http://www.globalknowledge.com) or >. Último acesso em 01 ago.2015.

MARTINS, D. *SCRUM e PMBOK*. Trabalhando juntos na gestão de projetos – parte 03. 5ª ed.: Project Prime, 2013. Disponível em: <http://projectprime.com.br/>. Último acesso em 26 fev.2016.

MARTINS, J. C. C. *Gerenciando projetos de desenvolvimento de software com PMI, RUP e UML*. Rio de Janeiro: Brasport, 2007.

PMI. *Guia PMBOK - Um guia do conhecimento em gerenciamento de projetos*. 5 ed. Newtown Square: Project Management Institute, Inc., 2013.

PMI. *PMI fact file*. PMI Today, Newtown Square, PA, Fev/2016, p. 4. Disponível em <<http://www.pmitoday-digital.com/pmitoday>>. Último acesso em 15 mar.2016.

PRESSMAN, R. *Engenharia de software: Uma abordagem profissional*. 7ª ed., São Paulo: McGraw-Hill, 2011.

PRIKLADNICKI, R.; AUDY, J.; EVARISTO, R. Distributed software development: toward an understanding of the relationship between project team, users and customers. *International Conference on Enterprise Information Systems*, 5 Anais. ICEIS Press, v. 3, p. 417-423, 2003.

SANTOS, B. O. *Utilização de métricas nos projetos de desenvolvimento de sistemas de informação: um survey com gerentes de projetos*. 2009. Dissertação (Mestrado em Informática) – Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, 2009.

SALADIS, F. P.; KERZNER H. *Bringing the PMBOK guide to life*. New Jersey: John Wiley and Sons, 2009.

SCRUM ALLIANCE. *The state of scrum: Benchmarks and Guidelines*. Scrum Alliance.org, jun. 2013. Disponível em: <<https://www.scrumalliance.org>>. Último acesso em 11 abr.2015.

SCHWABER, K. *Agile project management with scrum*. Washington: Microsoft Press, 2004.

SCHWABER, K.; SUTHERLAND, J. *The scrum guide: The definitive guide to scrum: the rules of the game*. 2013. Disponível em <<http://www.scrumguides.org>>. Último acesso em 01 nov.2014.

SOMMERVILLE, I. *Engenharia de software*. 8ª ed., São Paulo: Addison Wesley Brasil, 2007.

SOTILLE, M. A. *et al. Gerenciamento do escopo em projetos*. Rio de Janeiro: Editora FGV, 2014.

YIN, R. K. *Estudo de caso - Planejamento e métodos*. Porto Alegre: Bookman, 2005.

## **MODEL FOR MANAGING DISTRIBUTED SOFTWARE DEVELOPMENT PROJECTS IN A BANKING AUTOMATION COMPANY**

### **ABSTRACT**

*Distributed software development has been a widely used approach by large organizations for the use of specialized and low-cost manpower, although project management in this environment is still poorly studied. This work presents a case study carried out in a company of the banking automation sector that develops its projects using teams distributed in several countries with the purpose of evaluating PMBOK and SCRUM project management models, verifying their adherence to the distributed development environment. The study explores the project management model adopted by the company, identifying the use of PMBOK and SCRUM processes. The results demonstrates the importance of including processes of both models evaluated in a complementary way, to increase the effectiveness of distributed software development management and project performance improvement.*

**Keywords:** *Project management. PMBOK. SCRUM. Distributed software development.*