

GERENCIAMENTO DO ESCOPO DOS PROJETOS DE DESENVOLVIMENTO ÁGIL DE SOFTWARE: UM ESTUDO DE CASO

Autores	Filiação	Email
Fabrcio Garcia Imbrizi	UNINOVE	fabrcio.imbrizi@gmail.com
Emerson Antonio Maccari	UNINOVE	maccari@uninove.br

Direitos de cópia - creative commons.

Recebido em: 14/11/2012

Aprovado em: 21/11/2012

Disponibilização no site 06/12/2012

Páginas: 213 - 225

ID do artigo 1591

Editor Científico: Prof. Dr. Osni Hoss, Ph.D.

RESUMO

Os projetos de desenvolvimento de *software* buscam transformar objetivos de negócios em recursos funcionais e não funcionais de *software* para atender às expectativas do cliente. O objetivo da pesquisa é identificar como o gerenciamento do escopo dos projetos de desenvolvimento ágil de *software* pode contribuir para atender efetivamente os requisitos de negócio do cliente. A pesquisa é qualitativa do tipo descritiva, baseada no método de estudo de caso único, por meio das observações realizadas com a equipe de projetos de uma organização multinacional voltada à pesquisa, desenvolvimento e integração de soluções de tecnologias de *software* e *hardware*. A coleta de dados foi feita por meio de pesquisa documental e observações do próprio autor que participou ativamente do projeto. Os resultados demonstraram que em relação aos processos de gerenciamento do escopo do projeto e de desenvolvimento ágil de *software*, especificadamente o Scrum - *framework* para gerenciar o desenvolvimento de produtos complexos, ainda há lacunas entre o que propõem a literatura e as práticas adotadas pela organização. Ao final foi sugerido um conjunto de recomendações para aumentar as chances de sucesso do projeto, gerando mais satisfação ao atender as necessidades dos *stakeholders*.

Palavras-Chave: Escopo do projeto. Desenvolvimento de software. Metodologia ágil. Scrum.

PROJECT SCOPE MANAGEMENT OF AGILE SOFTWARE DEVELOPMENT: A CASE STUDY

ABSTRACT

The software development projects try to translate business objectives into functional and non-functional features to meet customer expectations. The research aims to identify how the project scope management of agile software development can contribute to effectively meet the customer's business requirements. The research is qualitative and descriptive, based on unique study case method, by

means of observations made with the project team of a global company dedicated to research, development and integration of software and hardware technologies. Data collection was done by documental research and observations of the author who actively participated in the project. The results showed that in relation to project scope management and agile software developments process, specifically Scrum – framework to manager the development of complex products, there are still gaps between the literature and the practices adopted by the organization. At the end it was suggested a set of recommendations to increase the chances of the project success, generating more satisfaction to meet all stakeholders business requirements.

Keywords: Project scope. Software development. Agile methodology. Scrum.

1 INTRODUÇÃO

Os projetos de desenvolvimento de *software* buscam transformar objetivos de negócios em recursos funcionais e não funcionais de *software* para atender às expectativas do cliente. Para satisfazer essas necessidades, é fundamental gerenciar adequadamente o escopo do projeto de desenvolvimento de *software*.

A organização alvo desta pesquisa é uma multinacional francesa de capital aberto voltada à pesquisa, desenvolvimento e integração de soluções de tecnologias de *software* e *hardware*. A divisão em estudo é a Bancos e Finanças que contém sessenta pessoas, dentre gerentes de projeto, gerente do programa, arquitetos, consultores, analistas, desenvolvedores e testadores.

O projeto objeto da pesquisa envolve o desenvolvimento de customizações na solução de *software* proprietária da companhia e integrações desta com os sistemas legados do cliente, uma instituição bancária pública, de acordo com os requisitos de um processo de negócios definido pela instituição. O prazo do projeto é de quatro meses e a equipe é composta por quinze membros.

Os requisitos, os quais deveriam refletir os objetivos de negócio da instituição, foram colhidos ao longo de encontros realizados entre as equipes de projeto da organização e da instituição, no curto prazo disponível para tal. Nestes encontros, foram observadas discordâncias sobre a forma de implementação das customizações face às diferentes visões de negócio e experiências anteriores dos analistas de negócios da instituição. Ao final, foram geradas documentações de acordo com o nível de detalhe informado pela instituição.

Ao longo das etapas de desenvolvimento e testes das customizações, foram identificadas várias lacunas entre o que foi construído e as necessidades da instituição. Além disso, foram solicitadas várias alterações nestas etapas que impactaram substancialmente o escopo do projeto. Uma parte das solicitações foi acatada pela organização, o que exigiu esforços extras da equipe, afetando o prazo e o custo do projeto. A outra parte foi registrada e postergada para implementação no futuro. O processo de alinhamento entre a organização e a instituição, no que tange ao escopo do projeto, foi custoso, conflitante, desgastante e moroso.

Neste contexto, esse artigo busca responder a seguinte questão de pesquisa: Como o gerenciamento do escopo dos projetos de desenvolvimento ágil de *software* pode contribuir para atender efetivamente os requisitos de negócio do cliente? Recorre-se à literatura de gerenciamento do escopo do projeto e de desenvolvimento ágil de *software*, especificadamente o Scrum, para propor recomendações que suportarão o objetivo descrito acima.

A adoção do processo proposto se justifica, pois, mitiga-se o risco de atrasos no cronograma, o risco do aumento do custo na alocação de mais recursos e do não atingimento dos objetivos de negócios do cliente. Além disso, espera-se aumentar as chances de sucesso do projeto, gerando mais satisfação ao atender as necessidades dos *stakeholders*.

2 REFERENCIAL TEÓRICO

Para o desenvolvimento deste trabalho foram utilizados os seguintes fundamentos conceituais: gerenciamento do escopo do projeto, metodologia ágil de desenvolvimento de software e Scrum - *framework* para gerenciar o desenvolvimento de produtos complexos, que é utilizado no desenvolvimento de software.

2.1 GERENCIAMENTO DO ESCOPO DO PROJETO

Segundo Dinsmore e Cabanis-Brewin (2006), o principal fator crítico de sucesso para qualquer projeto é um abrangente gerenciamento do escopo. O time de projeto deve realizar todas as ações necessárias para garantir que os requisitos de trabalho atendam às expectativas dos clientes. A definição do escopo e o controle do escopo devem ser examinados cuidadosa e continuamente para minimizar o risco de extravios ao longo do projeto.

De acordo com o Project Management Institute (PMI) (2008), o termo escopo se refere ao escopo do produto (características e funcionalidades que caracterizam um produto, serviço ou resultado) e/ou ao escopo do projeto (trabalho a ser executado para entregar um produto, serviço ou resultado com as características e funcionalidades específicas).

O time de projeto deve realizar o planejamento do escopo que contém as seguintes saídas: o plano de gerenciamento do escopo, a definição do escopo, a criação da estrutura analítica de projetos (EAP) ou *work breakdown structure* (WBS), a aprovação do escopo, o controle do escopo e a verificação do escopo. O time deve garantir que cada área é endereçada com a atenção devida para atingir o sucesso do projeto (DINSMORE; CABANIS-BREWIN, 2006).

O plano de gerenciamento do escopo deve conter o processo detalhado para determinar o escopo do projeto (baseado na expectativa do cliente e experiência da equipe), o desenvolvimento da WBS e a identificação dos entregáveis (*deliverables*), o processo formal de verificação do escopo e o processo descrito de controle do escopo, acordado entre os *stakeholders* (DINSMORE; CABANIS-BREWIN, 2006).

A definição do escopo inclui o detalhamento dos objetivos do projeto. A partir desses objetivos, o time de projetos identifica as premissas e restrições do escopo. Outros documentos técnicos podem suportar o conteúdo do escopo. O desenvolvimento da especificação do escopo do projeto (EEP) ou *project scope statement* explicita os detalhes, os entregáveis e os limites do projeto (DINSMORE; CABANIS-BREWIN, 2006).

O PMI (2008) estabelece um processo anterior à definição do escopo, denominado coleta de requisitos. Esse é o processo que define e documenta as necessidades dos *stakeholders* para atingir os objetivos de negócios. Os requisitos podem ser divididos em requisitos de projeto (de negócios, de entrega, etc.) e de produto (técnicos, de segurança, de performance, etc.). Eles são a base da WBS.

De acordo com Kerzner (2009), a WBS divide o projeto em unidades menores e

suficientes para serem gerenciáveis, independentes, mensuráveis e integráveis. A principal unidade de uma WBS é o pacote de trabalho que representa uma unidade única de trabalho que deve ter prazo, recurso e custo claramente definidos. O somatório de todos os pacotes de trabalho de uma WBS representa todo o projeto.

Segundo Carvalho e Rabechini (2011), as principais vantagens do uso do WBS, além daquelas descritas acima, são: facilitar a apresentação do trabalho, identificar as fontes de riscos e incertezas, disponibilizar a visão de todo o projeto e suas entregas programadas, explicitar a responsabilidade de cada envolvido e identificar quais necessitam de informações do projeto.

Uma vez o escopo elaborado, ele é discutido entre a equipe de projetos, *stakeholders*, patrocinadores e cliente. Com a aceitação e aprovação formal do escopo, o plano de gerenciamento do escopo e os documentos que o acompanham são, assim, as referências oficiais do projeto (DINSMORE; CABANIS-BREWING, 2006).

A verificação do escopo é o processo de formalização da aceitação das entregas finalizadas do projeto. Isso inclui atividades de mensuração, exame e verificação para determinar se o trabalho e as entregas estão de acordo com os requisitos e critérios de aceitação do produto (PMI, 2008).

O gerente de projetos deve garantir que o escopo do projeto seja plenamente controlado em todas as fases do projeto. Ele deve estar atento a qualquer discrepância nas atividades do projeto e riscos potenciais que o desvie dos objetivos e cronogramas definidos. Toda e qualquer mudança de escopo deve ser explicitamente documentada e comunicada a todos os interessados e à equipe de projetos (DINSMORE; CABANIS-BREWING, 2006).

A figura 1 ilustra a visão geral do PMI (2008) sobre os processos do gerenciamento do escopo do projeto.

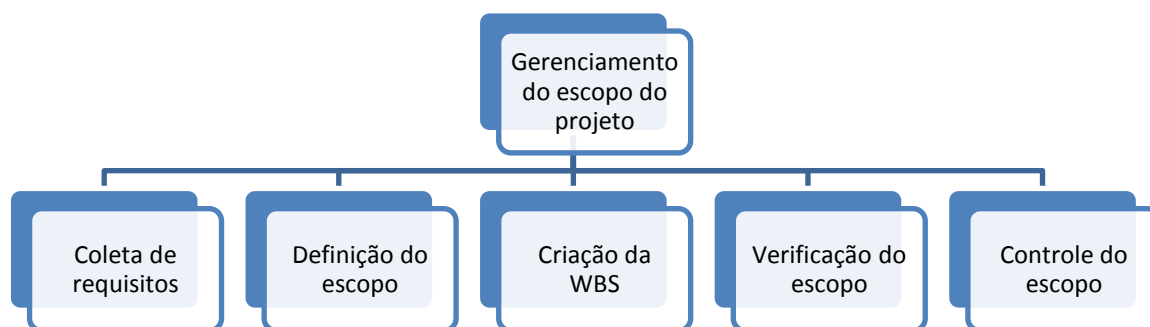


Figura 1: Visão geral do gerenciamento do escopo do projeto

Fonte: adaptado do PMI (2008).

2.2 METODOLOGIA ÁGIL

De acordo com Franco (2007), é preciso definir o que é agilidade e porque ela é tão importante para a indústria de desenvolvimento de *software*. Além disso, o termo “ágil” é usado de forma indiscriminada, levando a interpretações equivocadas e desconfiança perante o meio acadêmico. Segundo Highsmith (2002 apud FRANCO, 2007, p. 17), “Agilidade é a habilidade de criar e responder às mudanças como forma de manter a lucratividade num ambiente turbulento de negócios.”

Com o intuito de encontrar melhores maneiras de se desenvolver *software*, dezessete especialistas da área formalizaram em 2001 o “Manifesto para o

Desenvolvimento Ágil de Software” ou “Manifesto Ágil”. Esse documento contempla um conjunto de valores e princípios em que se valorizam indivíduos e interações mais que processos e ferramentas, *softwares* funcionais mais que documentação, colaboração com o cliente mais que negociaçãõ de contrato e respostas às mudanças mais que seguir um plano (BECK et al., 2001).

Segundo Ionel (2009), metodologias ágeis são cada vez mais usadas em empresas de desenvolvimento de software. Do ponto de vista da pesquisa acadêmica, ainda há poucos *papers* representativos, a maioria de estudos de casos e focados nas metodologias ágeis Scrum e XP – *Extreme Programming*.

2.3 SCRUM

Scrum é um *framework* para gerenciar o desenvolvimento de produtos complexos usado desde 1990. Scrum não é um processo ou técnica para desenvolver produtos. Ao contrário, ele é um *framework* no qual é possível empregar vários processos e técnicas. Scrum clarifica a relativa eficácia do gerenciamento de produtos e práticas de desenvolvimento. Ele é composto por equipes (e suas respectivas funções), eventos, artefatos e regras que regem o relacionamento e interação entre eles (SCHWABER; SUTHERLAND, 2011).

O Scrum é fundamentado no empirismo, o qual define que o conhecimento vem das experiências e tomadas de decisões baseadas no que é conhecido. Scrum aplica uma abordagem incremental e interativa para otimizar a previsibilidade e o controle de risco. O empirismo é baseado em três pilares: transparência, inspeção e adaptação (SCHWABER; SUTHERLAND, 2011).

De acordo com Schwaber e Sutherland (2011), transparência requer que os principais aspectos do processo estejam visíveis e padronizados para que os interessados tenham o mesmo entendimento, tal como, a definição de uma linguagem comum e o que se entende como algo concluído.

A inspeção dos artefatos do Scrum e do progresso do trabalho deve ocorrer com frequência para detectar desvios indesejados. Uma vez detectados, o processo ou o produto em desenvolvimento deve ser ajustado o quão antes para minimizar desvios futuros. (SCHWABER; SUTHERLAND, 2011).

As equipes do Scrum consistem do Dono do Produto, Time de Desenvolvimento e o Scrum Master. Elas são auto-organizáveis, ou seja, definem a melhor forma de trabalhar sem interferência de pessoas de fora da equipe, e são multifuncionais, isto é, contêm todas as competências necessárias para realizar as tarefas (SCHWABER; SUTHERLAND, 2011).

O Dono do Produto é a única pessoa responsável pelo gerenciamento do *backlog* do produto. Mesmo existindo um comitê para avaliar o *backlog*, o Dono do Produto será o responsável pelas decisões finais. O Time de Desenvolvimento contém os profissionais responsáveis por realizar a entrega. O Scrum Master é o responsável por garantir que o Scrum está entendido e sendo aplicado corretamente, apoiando o Dono do Produto, o Time de Desenvolvimento e a organização (SCHWABER; SUTHERLAND, 2011).

O Scrum usa eventos de tempos pré-definidos (*time-box*) para criar regularidade e evitar perda de tempo. O coração do Scrum é o Sprint, um *time-box* de um mês ou menos em que uma entrega concluída e utilizável é criada. O Sprint consiste de cinco eventos: Reunião de Planejamento do Sprint, *Daily Scrums*, trabalho de desenvolvimento, Reunião de Revisão do Sprint e Retrospectiva do Sprint. A figura 2 ilustra esses eventos que são fundamentais para permitir uma

adequada inspeção e adaptação do Sprint (SCHWABER; SUTHERLAND, 2011).

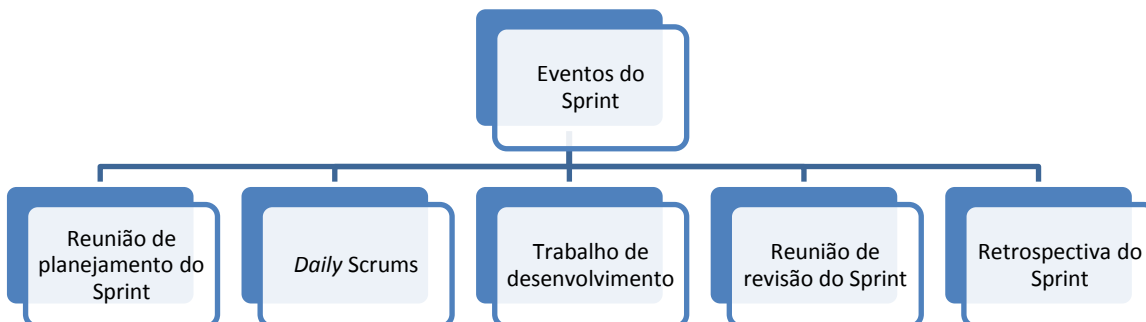


Figura 2: Visão geral dos eventos do Sprint

Fonte: elaborada pelos autores.

Durante o Sprint, nenhuma mudança que afete o objetivo do Sprint é feita, a composição do Time de Desenvolvimento e os objetivos de qualidade são mantidos constantes e o escopo deve ser clarificado e renegociado entre o Dono do Produto e o Time de Desenvolvimento à medida que surjam novas informações. Um Sprint possa ser cancelado, mas não é indicado (SCHWABER; SUTHERLAND, 2011).

Na Reunião de Planejamento do Sprint, o Dono do Produto apresenta os itens do *backlog* do produto priorizados de acordo com os objetivos organizacionais. Em seguida, o Time de Desenvolvimento estima o que poderá ser desenvolvido no Sprint e como o trabalho será executado. Os itens do *backlog* selecionados mais o plano para desenvolvê-los são chamados de Sprint Backlog. O Daily Scrum é uma reunião diária de 15 minutos para inspecionar a evolução das atividades e criar um plano para as próximas 24 horas. A figura 3 ilustra resumidamente a visão geral do *framework* Scrum (SCHWABER; SUTHERLAND, 2011).

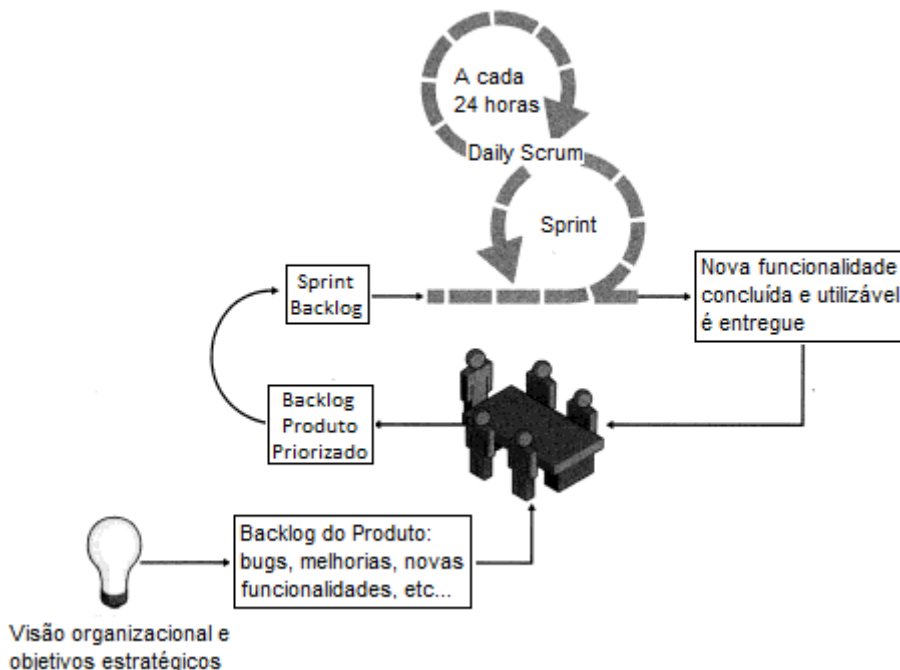


Figura 3: Visão geral do framework Scrum

Fonte: adaptado de Schwaber (2004).

Na Reunião de Revisão do Sprint, as equipes do Scrum e os *stakeholders* se reúnem para verificar o que foi e o que não foi entregue, os desafios enfrentados, as lições aprendidas, bem como, redefinir o *backlog* do produto, a partir dos itens antigos e dos novos que venham a surgir. A Retrospectiva do Sprint é uma reunião feita pela equipe do Scrum para inspecionar o último Sprint e propor adaptações para buscar melhorias no próximo Sprint. (SCHWABER; SUTHERLAND, 2011).

3 METODOLOGIA

Para a realização desta pesquisa, foram adotadas as estratégias de pesquisa bibliográfica, pesquisa documental e estudo de caso.

A pesquisa bibliográfica é fundamental para realizar qualquer pesquisa científica. Procura-se conhecer, analisar e discutir um assunto ou problema a partir de um referencial teórico. A pesquisa documental tem como base o uso de documentos como fonte de dados, informações e evidências, colhidos pelo próprio autor, a fim de contribuir para a análise dos problemas (MARTINS; THEÓPHILO, 2009).

Segundo Yin (2001), “um estudo de caso é uma investigação empírica que investiga um fenômeno contemporâneo dentro de seu contexto da vida real”. O estudo de caso abrange do planejamento à coleta de dados e análise dos dados. Esta é uma pesquisa que utilizou a metodologia de estudo de caso único, ou seja, apenas um único projeto da organização foi alvo de estudo.

Esta pesquisa foi executada com base na experiência profissional e prática de um dos autores do artigo, bem como da interação deste com o gerente do projeto, o gerente da solução de *software*, os membros da equipe de desenvolvimento de *software* da organização e os analistas de negócios da instituição bancária. É uma pesquisa qualitativa do tipo descritiva.

O objetivo do trabalho é identificar como o gerenciamento do escopo dos projetos de desenvolvimento ágil de *software* pode contribuir para atender efetivamente os requisitos de negócio do cliente, usando como referência a literatura em gerenciamento do escopo do projeto e desenvolvimento ágil de *software*, especificadamente o Scrum.

A construção deste trabalho seguiu o seguinte roteiro:

- a) elaboração da questão de pesquisa;
- b) descrição do referencial teórico que suporta todo o artigo;
- c) coleta de dados, a partir da documentação gerada no projeto, sejam documentos eletrônicos em formato texto, planilhas de controle e acompanhamento e e-mails gerados pela organização e pela instituição;
- d) análise dos resultados obtidos a partir dos processos macros descritos no referencial teórico; e
- e) conclusão do trabalho, composto ainda por recomendações e restrições.

A organização não autorizou a divulgação de seu nome e dos documentos usados para a coleta de dados, uma vez que há termos de confidencialidade firmados com a instituição financeira.

4 ANÁLISE DOS RESULTADOS OBTIDOS

A análise dos resultados foi realizada a partir dos macros processos que foram descritos no referencial teórico. A tabela 1 resume os resultados encontrados para

aos macros processos descritos na literatura.

Planejamento do gerenciamento do escopo: não foi definido nenhum plano de gerenciamento do escopo pelas equipes de projeto da organização e da instituição bancária para estabelecer os processos de coleta de requisitos, definição de escopo, criação da WBS, verificação e controle do escopo.

Coleta de requisitos: o processo inicial de coleta dos requisitos foi realizado em reuniões com os profissionais da organização e da instituição. De um lado houve a alocação de consultores de projetos em tecnologia da informação (TI) com experiência no segmento bancário. Do outro lado, ocorreu a alocação de analistas de negócios da instituição com conhecimento no modelo de negócio objeto do projeto, mas sem nenhuma experiência em práticas de gerenciamento de projetos, mapeamento de processos e criação de especificações de requisitos de negócios para o desenvolvimento de *software*.

Durante as reuniões, foram constatadas divergências de opiniões sobre a forma mais adequada de como seriam implementadas as funcionalidades do *software*. Os consultores de projetos sugeriram soluções simples e práticas de acordo com experiências passadas, ao passo que, os analistas de negócios mantiveram a posição sobre uma solução mais complexa que pudesse atender vários cenários futuros, mesmo que naquele momento ainda não estivessem claramente definidos.

O resultado destas reuniões foi a geração de um documento de entendimento sobre os requisitos da instituição. Esse foi escrito numa linguagem informal e não estruturada pelos consultores de negócio da instituição. Ele continha uma descrição macro das necessidades e imagens ilustrativas das interfaces sugeridas. As imagens sugeridas foram construídas a partir de outras aplicações existentes no mercado e não refletiam a interface do *software* em uso pela instituição.

A equipe de TI da instituição foi pouco envolvida neste processo. Ela atuou principalmente na elaboração dos documentos de especificação de integração entre a solução de *software* da organização e os sistemas legados da instituição. As versões dos documentos entregues estavam incompletas e não forneceram subsídios necessários para interpretar corretamente quais os dados e tipos de dados seriam trocados entre os sistemas da organização e da instituição.

Definição do escopo: o escopo foi definido usando como referência o plano preliminar de implantação (documento elaborado pela organização para traçar de forma macro o objetivo do projeto), o documento de entendimento e os documentos de especificação de integração elaborados pela instituição. Não foram definidas as premissas, as restrições e os limites do escopo.

Criação da WBS: não foi definida nenhuma WBS para o projeto. Os entregáveis foram, a priori, entendidos como tudo que estava descrito nos documentos de entendimento e integrações com os sistemas legados.

Verificação do escopo: a verificação foi realizada a partir das entregas efetuadas na etapa posterior ao desenvolvimento, denominada testes em ambiente de desenvolvimento da instituição. Os únicos documentos que suportaram a verificação foram o detalhamento do processo e as atas de reuniões realizadas

durante a etapa de desenvolvimento. N3o existia nenhum documento que descrevesse qual a regra para validar cada requisito de neg3ocio.

V3arias entregas foram recusadas, pois, segundo a institui33o, n3o atenderam 3s suas expectativas, parcial ou integralmente. Ainda, segundo a institui33o, alguns requisitos funcionais n3o precisariam ser explicitados, pois, se tratavam de obviedades inerentes a qualquer *software* e que dispensariam maiores detalhes. Parte destes requisitos estava relacionada 3 navegabilidade, usabilidade e controle de eventos de componentes de interface do *software*.

Os analistas de neg3ocios da institui33o que conduziram as reuni33es durante a etapa de desenvolvimento diferiram dos analistas que realizaram a verifica33o do escopo. Como consequ3ncia, algumas das sugest33es e orienta33es realizadas pelos primeiros foram recusadas pelos 3ltimos. Foi poss3vel identificar que havia tamb3m diverg3ncias sobre a forma como o processo de neg3ocio era interpretado por um e por outro.

Controle do escopo: durante a etapa de desenvolvimento, foram realizadas reuni33es de acompanhamento, uma vez por semana, entre a equipe de projetos da organiza33o e os analistas de neg3ocios da institui33o. Em cada reuni33o foi gerada uma ata que contemplava as solicita33es de altera33es do *software*, inclus3o de novas funcionalidades e exclus3o de outras. N3o foi descrita, em momento algum, uma declara33o expl3cita de mudan3a de escopo do projeto, muito embora, tenha sido relatado que a organiza33o iria avaliar o impacto das novas requisi33es da institui33o.

Durante a etapa de testes em ambiente de desenvolvimento da institui33o, as entregas que foram recusadas, mas que na vis3o da organiza33o se tratavam de mudan3a de escopo, foram identificadas e comunicadas como mudan3a de escopo. Estas declara33es geraram v3arias discuss33es, entre r3plicas e tr3plicas, que demandou excessivos esfor3os de avalia33o das documenta33es. O relacionamento entre as partes foi afetado e no final, embora tenha se chegado a um acordo, o projeto teve seu prazo, escopo original e custo comprometidos.

Em ambas as etapas de desenvolvimento e testes, tamb3m ocorreram diversas trocas de e-mails e contatos exclusivamente verbais. Segundo Khan (2006), mudan3as n3o autorizadas podem inflar o escopo do projeto como resultado de instru33es verbais, por e-mail ou escritas, e que n3o foram formalmente avaliados os impactos da mudan3a no projeto.

Nestas etapas, foi necess3ria a atualiza33o, em v3arios momentos, do documento de entendimento e das especifica33es de integra33es com os sistemas legados. A equipe de TI da institui33o foi inserida tardiamente no processo para realizar os desenvolvimentos cab3veis 3 institui33o no que tange 3s integra33es. Isso pressionou o prazo dispon3vel para a organiza33o realizar a sua parte no processo de integra33o, o que afetou a aloca33o dos profissionais especialistas.

Scrum: A equipe de desenvolvimento da organiza33o adotou parcialmente o Scrum. As fun33es da equipe do Scrum e os Sprints iniciais foram definidos e o *Daily Scrum* foi regularmente aplicado. Os Sprints variaram entre uma e duas semanas. Dado o escopo original, do qual os desenvolvedores tinham conhecimento, os objetivos tra3ados foram atendidos. O *Daily Scrum* mostrou-se uma ferramenta fundamental para efetuar os alinhamentos de entendimento e ajustes necess3rios na aplica33o.

Não existiu uma definição formal e documental de uma linguagem comum e o que se entende como algo concluído, seja entre os desenvolvedores ou entre a equipe de projetos da organização e os analistas de negócios da instituição.

Durante a etapa de testes em ambiente de desenvolvimento da instituição, os *Daily Scrums* não foram praticados regularmente. Os eventos Reunião de Revisão de Sprint e Retrospectiva do Sprint não ocorreram, conseqüentemente, não foram registradas as lições aprendidas e as sugestões de melhorias para os próximos Sprints, comprometendo, a inspeção e adaptação dos Sprints.

Tabela 1: Resultados encontrados para cada macro processo

Itens	Resultados
Planejamento do gerenciamento do escopo	✓ Não foi elaborado um plano de gerenciamento de escopo.
Coleta de requisitos	<ul style="list-style-type: none"> ✓ Os analistas de negócios da instituição desconheciam alguma metodologia de gerenciamento de projetos; ✓ Documento de entendimento em linguagem informal; ✓ Equipe de TI da instituição foi pouco envolvida nesta fase; ✓ Documentação de integração entregue parcialmente e incompleta.
Definição do escopo	<ul style="list-style-type: none"> ✓ Documentos de entendimento e integrações estavam incompletos; ✓ Premissas, restrições e limites do escopo indefinidos.
Criação da WBS	✓ Não foi criada uma WBS.
Verificação do Escopo	<ul style="list-style-type: none"> ✓ Sem regra de validação definida; ✓ Vários requisitos foram recusados face aos conflitos de interpretações; ✓ Os analistas de negócios que verificaram a entrega foram diferentes daqueles que definiram o escopo inicial.
Controle do escopo	<ul style="list-style-type: none"> ✓ Reuniões de acompanhamento com registro em atas; ✓ Sem declaração explícita de mudança de escopo; ✓ Discussões custosas, conflitantes, desgastantes e morosas; ✓ Mudanças constantes no documento de entendimento e nas especificações de integrações.
Scrum	<ul style="list-style-type: none"> ✓ Funções da equipe do Scrum e Sprints foram definidos; ✓ Daily Scrum foi regularmente aplicado. ✓ Não foi definido uma linguagem comum e o que se entende como algo concluído; ✓ Os eventos Reunião de Revisão do Scrum e

5 CONCLUS3O

O referencial te3rico fornece um conjunto de processos-chave para nortear os passos da equipe de projetos no gerenciamento do escopo do projeto. 3 possvel identificar similaridades e completudes entre as refer4ncias descritas que facilitaram a leitura e interpreta33o do projeto objeto da pesquisa.

A verifica33o e o controle (ou inspe33o) do escopo do projeto foram percebidos como um dos processos cr3ticos definidos na literatura de gerenciamento do escopo do projeto e de desenvolvimento 3gil de *software*. Isso tamb3m foi identificado de acordo com os impactos observados nas an3lises dos resultados.

3 possvel concluir que apesar da organiza33o adotar algumas pr3ticas, ainda h3 lacunas entre o que prop3em a literatura de gerenciamento do escopo de projetos e de desenvolvimento 3gil de *software* e as pr3ticas adotadas no projeto.

Portanto, a quest3o de pesquisa pode ser respondida com as seguintes recomenda33es a serem consideradas pela organiza33o para que o gerenciamento do escopo dos projetos de desenvolvimento 3gil de *software* possa contribuir para atender efetivamente os requisitos de neg3cio do cliente. A tabela 2 resume as recomenda33es propostas para aos macros processos descritos na literatura.

- a) Realizar um *workshop* introdut3rio para os analistas de neg3cios da institui33o banc3ria sobre os processos de gerenciamento de projetos alvos do projeto, permitindo melhor equalizar o conhecimento de todos sobre os conceitos inerentes desta 3rea, com 4nfase nos processos de gerenciamento do escopo do projeto;
- b) Elaborar um plano de gerenciamento de projetos que explicita claramente os processos de coleta de requisitos, defini33o de escopo, cria33o da WBS, verifica33o e controle do escopo do projeto;
- c) Incentivar o envolvimento da equipe de TI da institui33o desde o in3cio do projeto para obter o alinhamento adequado entre essa equipe e a equipe de analistas de neg3cios da institui33o. Uma alternativa 3 obter a formaliza33o documental conjunta das equipes sobre as necessidades da institui33o, principalmente, no que tange aos requisitos de integra33o;
- d) Definir um processo formal de modelagem dos processos de neg3cio. Segundo Pinna (2008), as notaa3es mais utilizadas atualmente s3o o SADT/IDEF0 (que descreve os processos e sub-processos de neg3cios e suas respectivas entradas, sa3idas e regras de neg3cio) e UML Use Case (que descreve os fluxos de interaa3o entre o usu3rio e o *software* e suas respectivas pr3-condi33es, p3s-condi33es e regras de neg3cios);
- e) Exigir que a documentaa3o de especificaa3o de integra33es seja entregue completa e revisada antes do in3cio da etapa de desenvolvimento das customiza33es de *software*. Al3m disso, essa documentaa3o deve vir acompanhada dos exemplos de modelos de dados e tipos de dados que ser3o trocados entre os sistemas da organiza33o e da institui33o;
- f) Formalizar via documentaa3o e assinaturas dos *stakeholders*, o escopo do projeto, esclarecendo quais s3o os entreg3veis, os limites, as exclus3es, as premissas e restri33es do escopo;

- g) Criar a WBS do projeto para nortear a verificação e o controle do escopo do projeto, bem como usá-la como ferramenta de apoio na estruturação dos Sprints. A WBS deve ser compartilhada com toda a equipe de projetos da organização e da instituição;
- h) Definir, previamente, quais os critérios serão utilizados para validar as entregas, incluindo o resultado esperado, a criticidade, as restrições funcionais, técnicas e soluções alternativas. A criticidade deve explicitar claramente o tipo de impacto para o projeto, como, por exemplo: alta (impede a utilização do *software*), média (dificulta ou acarreta prejuízos para o sucesso da utilização do *software*) e baixa (não impede ou prejudica a utilização do *software*);
- i) Em caso de mudança de um ou mais membros da equipe de TI ou de consultores de negócio da instituição, visitar os processos de gerenciamento do escopo do projeto e o escopo do projeto definidos anteriormente, em reunião presencial, com o intuito de manter todos alinhados e evitar retrocessos;
- j) Formalizar cada solicitação de mudança de escopo, única e exclusivamente, em documento próprio intitulado, por exemplo, “Documento de Requisição de Mudanças”, evitando-se qualquer outro meio de comunicação, formal ou informal;
- k) Realizar *Daily Scrums* mesmo na etapa de testes em ambiente de desenvolvimento da instituição para manter toda a equipe de desenvolvimento ciente das mudanças e ajustes requisitados.
- l) Realizar as Reuniões de Revisão de Sprint e Retrospectiva do Sprint, objetivando registrar as lições aprendidas e sugestões de melhorias para os próximos Sprints.

Tabela 2: Recomendações propostas para cada macro processo

Itens	Recomendações
Planejamento do gerenciamento do escopo	<ul style="list-style-type: none"> ✓ Realizar um workshop introdutório sobre os processos de gerenciamento de projetos para os analistas de negócios da instituição; ✓ Elaborar um plano de gerenciamento de projetos.
Coleta de requisitos	<ul style="list-style-type: none"> ✓ Envolver a equipe de TI da instituição desde o início; ✓ Definir um processo formal de modelagem dos processos de negócio.
Definição do escopo	<ul style="list-style-type: none"> ✓ Formalizar o escopo do projeto; ✓ Definir os entregáveis, os limites, as exclusões, as premissas e restrições do escopo; ✓ Fornecer previamente toda documentação das integrações de software.
Criação da WBS	<ul style="list-style-type: none"> ✓ Criar a WBS.
Verificação do	<ul style="list-style-type: none"> ✓ Definir os critérios usados para validar as entregas.

Escopo	
Controle do escopo	<ul style="list-style-type: none"> ✓ Revisitar os processos de gerenciamento do escopo do projeto, em caso de mudança de membros da equipe; ✓ Formalizar cada solicitação de mudança de escopo em documento próprio para este fim.
Scrum	<ul style="list-style-type: none"> ✓ Realizar Daily Scrum também na etapa de testes. ✓ Realizar as Reuniões de Revisão de Sprint e Retrospectiva do Sprint.

Fonte: elaborada pelos autores.

O envolvimento direto de um dos autores no projeto é um fator limitante e enviesado. Sugere-se para pesquisas futuras, explorar as estratégias de mapeamento de processos e requisitos de *software* com o objetivo de adotar instrumentos eficazes de definição de escopo do projeto.

REFERÊNCIAS

- BECK, M.; BEEDLE, M.; BENNEKUM, A. VAN.; COCKBURN, A.; CUNNINGHAM, W.; FOWLER, M.; GRENNING, J.; HIGHSMITH, J.; HUNT, A.; JEFFRIES, R.; KERN, J.; MARICK, B.; MARTIN, R. C.; MELLOR, S.; SCHWABER, K.; SUTHERLAND, J.; THOMAS, D. **The manifesto for agile software development**. [Wasatch]: [s.n.], 2001. Disponível em: <<http://www.agilemanifesto.org/>>. Acesso em: 25 jun. 2012.
- CARVALHO, M. M.; RABECHINI, R., Jr. Fundamentos em gestão de projetos: construindo competências para gerenciar projetos. 3. ed. São Paulo: Atlas, 2011.
- DINSMORE, P. C.; CABANIS-BREWIN, J. The AMA handbook of project management. 2. ed. New York: AMACOM, 2006.
- FRANCO, E. F. Um modelo de gerenciamento de projetos baseado nas metodologias ágeis de desenvolvimento de software e nos princípios da produção enxuta. Dissertação de Mestrado, Universidade de São Paulo, São Paulo, 2007.
- IONEL, N. Agile software development methodologies: an overview of the current state of research. The Journal of the Faculty of Economics, v. 4, n. 1, p. 381-385, mai. 2009.
- KERZNER, H. Project management: a system approach to planning, scheduling, and controlling. 10. ed. New York: John Wiley & Sons, 2009.
- KHAN, A. Project scope management. Cost Engineering, v. 48, n. 6, p. 12-16, jun. 2006.
- MARTINS, G. A.; THEÓPHILO, C. R. Metodologia da investigação científica para ciências sociais aplicadas. 2. ed. São Paulo: Atlas, 2009.
- PINNA, C. C. A.; CARVALHO, M. M. Gestão de escopo em projetos de aplicações web. Revista Produção On Line, v. 8, n. 1, mar. 2008.
- PROJECT MANAGEMENT INSTITUTE. A guide to the project management body of knowledge (PMBOK Guide). 4. ed. Newtown Square: Project Management Institute, 2008.
- SCHWABER, K. Agile project management with scrum. 1. ed. Redmond: Microsoft Press, 2004.
- SCHWABER, K.; SUTHERLAND, J. The scrum guide. Disponível em: <http://www.scrum.org/storage/scrumguides/Scrum_Guide.pdf>. [S.l.: s.n.], 2011. Acesso em: 24 jun. 2012.
- YIN, R. K. Estudo de caso: planejamento e métodos. 2. ed. Porto Alegre: Bookman, 2011.