

FUNDAÇÃO GETULIO VARGAS
ESCOLA DE ADMINISTRAÇÃO DE EMPRESAS DE SÃO PAULO

ANDREZA CAVALLARI

**AVALIAÇÃO DE MELHORIA DO PROCESSO EM DESENVOLVIMENTO DE
SOFTWARE COM ADOÇÃO DE MÉTODOS ÁGEIS:**

Estudo de Caso em Empresa Brasileira

SÃO PAULO

2019

ANDREZA CAVALLARI

**AVALIAÇÃO DE MELHORIA DO PROCESSO EM DESENVOLVIMENTO DE
SOFTWARE COM ADOÇÃO DE MÉTODOS ÁGEIS: ESTUDO DE CASO EM
EMPRESA BRASILEIRA**

Dissertação apresentada à Escola de Administração de Empresas de São Paulo da Fundação Getúlio Vargas como requisito para obtenção do título de Mestre em Gestão para Competitividade.

Linha de pesquisa: Tecnologia da Informação

Orientador: Prof. Dr. Cláudio Luís C. Larieira

SÃO PAULO

2019

Cavallari, Andreza.

Avaliação de melhoria do processo em desenvolvimento de *software* com adoção de métodos ágeis: estudo de caso em empresa brasileira / Andreza Cavallari. - 2019.

41 f.

Orientador: Cláudio Luís Carvalho Larieira.

Dissertação (mestrado profissional MPGC) – Fundação Getulio Vargas, Escola de Administração de Empresas de São Paulo.

1. Software - Desenvolvimento. 2. Desenvolvimento ágil de software. 3. Scrum (Desenvolvimento de software). 4. Administração de projetos. I. Larieira, Cláudio Luís Carvalho. II. Dissertação (mestrado profissional MPGC) – Escola de Administração de Empresas de São Paulo. III. Fundação Getulio Vargas. IV. Título.

CDU 681.3.06

ANDREZA CAVALLARI

**AVALIAÇÃO DE MELHORIA DO PROCESSO EM DESENVOLVIMENTO DE
SOFTWARE COM ADOÇÃO DE MÉTODOS ÁGEIS: ESTUDO DE CASO EM
EMPRESA BRASILEIRA**

Dissertação apresentada à Escola de Administração de Empresas de São Paulo da
Fundação Getúlio Vargas como requisito para obtenção do título de Mestre em
Gestão para Competitividade.

Linha de pesquisa: Tecnologia da Informação

Data de Aprovação: ____/____/____

Banca Examinadora:

Prof. Dr. Cláudio Luís Carvalho Larieira
(Orientador)
FGV – EAESP

Prof. Dr. José Luiz Kugler
FGV – EAESP

Prof. Dr. Marcelo Novaes de Rezende
Instituto de Pesquisas Tecnológicas

Resumo

O processo de desenvolvimento de *software* envolve muitas atividades dentro de uma organização, tais como o levantamento de requisitos; análise de requisitos; implementação; testes e a implantação, neste sentido, para o desenvolvimento de projetos de *software*, torna-se fundamental o gerenciamento de projetos, que é uma das principais ferramentas utilizadas pelas organizações. Através do efetivo gerenciamento, torna-se possível melhores resultados na implementação de projetos, tornando o serviço ofertado pela organização mais produtivo e conseqüentemente trazendo melhores resultados, para tanto, é preciso definir qual o melhor método de gerenciamento de projetos, comparando os métodos tradicionais com os métodos ágeis. Desta forma, neste estudo empírico, que foi realizado em uma grande empresa do ramo financeiro, especializada na recuperação de crédito de seus clientes, avaliou-se a comparação de projetos semelhantes e em um mesmo período de tempo, que antes eram geridos no modelo tradicional de gestão de projetos baseado no PMBOK e após a adoção da transformação digital passou a gerir os projetos com métodos ágeis utilizando o *framework Scrum*. Os resultados da pesquisa mostram que, baseada na métrica de tempo de entrega do projeto em dias considerando todo o ciclo de vida de um projeto, há ganhos no processo de utilização do método ágil *Scrum*.

Palavras-chave: Desenvolvimento de *Software*. Métodos Ágeis. *Scrum*.

Abstract

The software development process involves numerous activities within an organization, such as requirements gathering; requirements analysis; Implementation; In this sense, testing and implementation for the development of software projects, project management is essential, which is one of the main tools used by organizations. Through effective management, it is possible to achieve better results in project implementation, making the service offered by the organization more productive and consequently bringing better results. Therefore, it is necessary to define the best project management method, comparing the traditional methods with the agile methods. Thus, in this empirical study, which was conducted in a financial company specialized in the recovery of its clients' credit, the comparison of similar projects over the same period, which were previously managed in the traditional model of credit, was evaluated. project management based on PMBOK and after the adoption of digital transformation started managing projects using agile methods using the Scrum framework. Research results show that, based on the project delivery time metric in days considering the entire project life cycle, there are efficiency gains and process improvement in using the Agile Scrum method.

Keywords: *Software* development. Agile Methodologies. *Scrum*.

SUMÁRIO

1. INTRODUÇÃO	06
2. REFERENCIAL TEÓRICO	09
2.1 Aspectos Gerais sobre Gestão de Projetos	09
2.2 Aspectos Gerais sobre o Método Tradicional	10
2.3 Aspectos Gerais sobre a Metodologia Ágil	13
2.4 Influência do método ágil para a melhoria do processo de desenvolvimento de software	15
3. METODOLOGIA	18
3.1 Estratégia de Pesquisa	17
3.2 Seleção do Caso	17
3.3 Coleta do Dados	18
3.3.1 Pesquisa documental e sites as organização	18
3.3.2 Observação	19
3.3.3 Entrevista semiestruturada	19
3.4 Análise do Dados	20
4. ESTUDO DE CASO E ANÁLISE DOS RESULTADOS	21
4.1 Organização	21
4.2 Metodologia de gestão de projetos da organização	21
4.3 Adoção de método ágil na organização	23
4.4 Análise dos resultados	25
5. CONCLUSÃO	33
REFERÊNCIAS	36

1. INTRODUÇÃO

A importância e o uso de tecnologia de informação e comunicação nas empresas têm crescido, com seu papel sendo considerado cada vez mais estratégico (MEIRELLES, 2016). Diante desse crescimento, a área de tecnologia de informação é vista como uma das mais poderosas influências no planejamento da organização, podendo ser decisiva para o sucesso do negócio (ALBERTIN, 2004). Com o objetivo de assimilar e utilizar essa tecnologia, as organizações realizam seus projetos de TI (ALBERTIN; ALBERTIN, 2016a).

Segundo Kerzner (2017) é considerado um projeto uma sequência de atividades ou tarefas que atendem um objetivo específico conforme uma especificação, e que também tenha datas definidas de início e fim, além de recursos financeiros limitados, humanos e não humanos.

Para o gerenciamento desses projetos é necessário a escolha de uma metodologia de desenvolvimento de *software* que se adeque ao objetivo da organização. A escolha dessas metodologias é discutida em vários aspectos para adequar a sua aplicação a necessidade organizacional diante dos projetos que tendem a ser cada vez mais complexos devido a fatores externos e internos que exigem atenção e rápida reação do mercado (MARIANO, 2008).

Segundo Meredith e Mantel (2000), na área de engenharia de *software* a literatura fomenta várias discussões sobre os fracassos resultantes de uma má aplicação ou omissão de metodologias no desenvolvimento de *software*. Os autores constataram que a maioria dos problemas relacionados aos projetos tem uma parcela de culpa derivado do mau gerenciamento do processo de desenvolvimento de *software*.

Conforme Moore (2015), apesar de mais de meio século de estudos e pesquisas, projetos de tecnologia de informação continuam falhando com frequência. Nas últimas décadas, mais da metade dos projetos de TI falharam em entregar o seu valor esperado (DOHERTY; ASHURST; PEPPARD, 2012), a exemplo dos casos de projetos de TI com orçamentos de milhões, que entregaram menos da metade do valor prometido (BLOCH; BLUMBERG; LAARTZ, 2012).

Neste sentido, existe uma grande demanda mercadológica para maior agilidade, eficiência e qualidade dos projetos de desenvolvimentos de *softwares*. Vitiello (2001) ressalta que a constante busca por rapidez nos projetos, demanda da equipe e do gerenciamento dos projetos de desenvolvimento uma maior organização e

flexibilidade para atingir os objetivos em menor tempo possível e com maior qualidade.

Neste contexto, Graham (2009) ressalta que o gerenciamento de projetos não é uma tarefa simplista, ela precisa atender não somente as especificações do escopo, mas também lidar com as expectativas de todos os envolvidos, principalmente do cliente e da qualidade esperada.

Para Sommerville (2011), o desenvolvimento de *software* é um processo criativo, com profissionais com talento, conhecimentos diferenciados e multidisciplinar com o objetivo de construir valor no produto entregue, e a gestão de projetos deve considerar estes aspectos. O fracasso nos projetos decorrentes das dificuldades da gestão no desenvolvimento de *software* surgiu desde a década de sessenta, com atrasos nos projetos, entregas com baixa qualidade que evidenciaram as falhas nas abordagens tradicionais (SOMMERVILLE, 2003). Diante deste cenário, e particularidades da área de tecnologia, a gestão de projetos foi forçada a evoluir e adaptar as suas práticas para um aumento da eficiência e qualidade dos produtos entregues.

Na literatura é encontrada uma grande discussão entre dois pilares do gerenciamento de projetos, o primeiro baseado em princípios clássicos, e a segunda está focado na utilização de métodos ágeis para o desenvolvimento de *software*. O objetivo das duas abordagens em linhas gerais é igual, garantir que o projeto seja concluído com qualidade, no prazo combinado e com os recursos otimizados, sendo que cada método de desenvolvimento de *software* possui características e premissas específicas para sua utilização (BARROCA; SHARP; SALAH; TAYLOR; GREGORY, 2015)

Diante desse contexto, surge a necessidade de conhecer as práticas ágeis, especificamente estudada nessa pesquisa o método *Scrum* e o método tradicional de gerenciamento de projetos baseado no PMBOK, para realizar uma análise comparativa entre a utilização dos dois métodos na gestão de desenvolvimento de *software*. A partir da identificação das principais características e um comparativo da adoção das duas abordagens em projetos semelhantes, identificar os resultados e os ganhos de melhoria do processo baseado na métrica de ciclo de vida do projeto. Na literatura, não foram identificados muitos estudos práticos na comparação de empresas que passaram pela transformação digital sobre a adoção do método

Scrum e o impacto da sua utilização com relação aos aspectos de avaliação de melhoria do processo de desenvolvimento de *software*.

Desta forma, o trabalho tem o intuito de responder à pergunta de pesquisa: Qual o ganho na avaliação de melhoria do processo de desenvolvimento de *software* com a utilização do método ágil *Scrum*?

O objetivo desta pesquisa foi realizar um estudo de caso único em uma empresa brasileira do ramo financeiro especializada na recuperação de crédito de seus clientes para identificação do comparativo dos métodos e dos ganhos e melhorias no processo de desenvolvimento de *software*. Esta avaliação de melhoria no processo de desenvolvimento dos projetos foi demonstrada através do uso da métrica de *lead-time* (ciclo de vida), comparando o tempo de todo o ciclo de vida de projetos, a adoção do método ágil reduziu o tempo de entrega dos projetos da área de tecnologia, considerando entregas parciais com valor para o cliente. Após a realização de um processo de transformação organizacional os projetos de desenvolvimento de *software* que antes eram conduzidos através de método tradicional, passaram a utilizar o método ágil *Scrum*.

Esta pesquisa visa contribuir de forma geral com a discussão sobre os ganhos de melhoria do processo de desenvolvimento de *software* com adoção de métodos ágeis, e de maneira específica com a literatura que diz respeito a comparação da utilização de métodos tradicionais com os métodos ágeis em projetos de desenvolvimento de *software*, reunindo uma revisão exploratória da literatura sobre o método PMBOK e o método *Scrum*. Do ponto de vista prático, este trabalho visa contribuir para que organizações de forma geral, possam identificar e trabalhar os devidos pontos para aumentar os ganhos de melhorias no processo com adoção do método ágil na condução de seus projetos de desenvolvimento de *software*, o que poderá colaborar também para melhores resultados e sua vantagem competitiva no mercado.

Este artigo está dividido em cinco seções, iniciado por esta introdução, seguindo na segunda seção sobre a utilização e diferenciação dos métodos de gerenciamento de projetos. A terceira seção aborda a metodologia de pesquisa. A quarta seção apresenta o estudo de caso e analisa a situação encontrada na empresa foco deste estudo, finalizando na sexta seção com uma conclusão da pesquisa e sugestões para trabalhos futuros.

2. REFERENCIAL TEÓRICO

2.1 Aspectos Gerais sobre Gestão de Projetos

Segundo Maximiliano (2008), o termo projeto pode ser considerado como iniciativas transitórias, com atividades claras e definidas, criadas para resolver algum problema, oportunidade ou interesse de uma pessoa ou organização. Neste sentido, Vargas (2018), orienta que o projeto se trata de um empreendimento não repetitivo, que se caracteriza através de uma sequência clara e lógica de eventos.

Para a realização de projetos, especificamente projetos de desenvolvimento de *software* é necessário a utilização de métodos de gerenciamento de projetos de *software*. Nesse sentido, Schwaber (2014) afirma que a gestão de projetos consiste no emprego de um conjunto de conhecimento, de habilidade e de ferramentas com o objetivo de organizar as atividades de um projeto para atender as expectativas do cliente final. Dessa forma, o gerenciamento de projetos de *software* não é uma tarefa simplista, ela auxilia na prevenção de fatores de risco, permitindo o controle do que está sendo desenvolvido e antecipando as situações desfavoráveis. Segundo Vargas (2008), os ganhos e benefícios da boa gestão de projetos são notados em vários setores da organização, permitindo a tomada de decisões mais ágeis e certeiras.

Segundo Sommerville (2003), desde a década de sessenta começaram a surgir as dificuldades na gestão de projetos de *software*, onde entregas em atraso e baixa qualidade evidenciaram falhas nas abordagens tradicionais utilizadas. Especificamente na área de tecnologia e os constantes avanços tecnológicos, a gestão de projetos foi desafiada a evoluir e adaptar suas práticas para melhorar o tempo de entrega e maior qualidade dos produtos criados e desenvolvidos.

Na literatura há dois grandes pilares, um embasado nos princípios tradicionais e clássicos de gerenciamento de projetos e outro foca nos métodos ágeis de desenvolvimento de *software*. Os dois seguem o mesmo objetivo, concluir o desenvolvimento de *software* com qualidade garantida, prazo acordado e recursos otimizados. Diante disso, é necessário conhecer os processos e as práticas de cada abordagem, em específico dos métodos tradicionais de gestão de projetos escolhido o PMBOK para este estudo, e das práticas ágeis escolhido o *framework* ágil SCRUM.

2.2 Aspectos Gerais sobre o Método Tradicional

Há diversas metodologias de gerenciamento de projetos tradicionais e cada uma delas tem suas características específicas, para este estudo foi escolhido o PMBOK por sua vasta utilização nas organizações, que não pode ser considerado uma metodologia em si, mas sim um guia (PMBOK, 2013).

O *Project Management Body of Knowledge* ou PMBOK é um guia de boas práticas em gerenciamento de projetos elaborado e escrito pelos voluntários do *Project Management Institute* (PMI). Segundo Kardec (2012) o PMI é o maior disseminador de desenvolvimento de projetos através de processos, o guia descreve normas, métodos, processos e práticas reconhecidas em todo o mundo. Segundo Cruz (2013), o guia PMBOK pode ser utilizado na maioria dos projetos, independente do tempo de duração e da área de atuação, os conhecimentos do guia devem ser aplicados de forma padronizada em todos os projetos para compreensão de todos envolvidos.

Conforme PMBOK (2013) os processos são divididos em cinco grupos bem definidos para permitir a relação organizada e documentada conforme as regras do guia. Os cinco grupos de processos são: Iniciação, Planejamento, Execução, Monitoramento e Controle, e Encerramento.

Estes processos são agrupados em dez áreas de conhecimento que são utilizadas conforme cada necessidade de cada projeto.

Estas áreas de conhecimento e seus objetivos segundo PMBOK (2013) são:

1. *Gerenciamento de integração*: Inclui processos e atividades necessários para identificar, definir, combinar, unificar e coordenar os grupos de processos de gerenciamento. O gerenciamento da integração requer que sejam feitas escolhas sobre alocação de recursos, concessões entre objetivos e alternativas conflitantes, além do gerenciamento de dependências mútuas entre áreas de conhecimento e processos.

2. *Gerenciamento do escopo*: inclui processos necessários para assegurar que o projeto inclui todo o trabalho necessário e somente o trabalho necessário para concluir o projeto com sucesso. O objetivo é definir e controlar o que faz parte do projeto.
3. *Gerenciamento do tempo*: Inclui processos necessários para estimar as tarefas, seus recursos e durações, de modo a gerenciar o projeto para o término pontual.
4. *Gerenciamento de custos*: Inclui processos envolvidos em estimativas, orçamentos e controle dos custos, de modo que o projeto possa ser terminado dentro do orçamento aprovado.
5. *Gerenciamento da qualidade*: Inclui processos e atividades da organização executora que determinam as políticas de qualidade, objetivos, requisitos e responsabilidades de modo que o projeto satisfaça às necessidades para as quais foi empreendido. Implementa o sistema de gerenciamento da qualidade e atividades para a melhoria contínua dos processos.
6. *Gerenciamento de recursos humanos*: Inclui processos que organizam e gerenciam a equipe do projeto. Faz parte desta área do conhecimento descrever as necessidades de pessoal e suas respectivas capacidades e habilidades.
7. *Gerenciamento de comunicações*: Inclui todos os processos necessários para assegurar que as informações do projeto sejam geradas, coletadas, distribuídas, armazenadas, recuperadas e organizadas de maneira oportuna e apropriada
8. *Gerenciamento de riscos*: Inclui processos de planejamento, identificação, análise, planejamento de respostas, bem como o monitoramento e controle de riscos de um projeto. Os objetivos do gerenciamento de riscos são aumentar a probabilidade e o impacto dos eventos positivos e reduzir a probabilidade e o impacto dos eventos negativos no projeto.

9. *Gerenciamento das aquisições*: Inclui os processos necessários para comprar ou adquirir produtos, serviços ou resultados externos ao projeto e abrange o gerenciamento de contratos.
10. *Gerenciamento de Stakeholders*: Inclui processos de identificação, planejamento, engajamento e gerenciamento das partes interessadas. Os objetivos do gerenciamento das partes interessadas é aumentar o suporte e comprometimento dos *stakeholders* ao projeto.

Estas diretrizes estabelecem a estrutura, a relação dos processos e a interação entre agentes de um projeto. De acordo com Fernández, Garrido, Ramírez e Perdomo (2015), os processos organizados em áreas de conhecimento são a principal característica do PMBOK, escrita a partir de experiências boas e ruins dos autores do PMI, e é pela literatura considerado um método tradicional e burocrático do gerenciamento de projetos.

O PMBOK (2013) ressalta que o método de trabalho tradicional utilizando as boas práticas do guia, trabalha com diversos grupos de processos para abranger todas as fases do ciclo de vida de um projeto, escrevendo de forma genérica as boas práticas e sem especificar a área de conhecimento que será aplicada. Neste contexto, Cruz (2013) acrescenta que o PMBOK não diz a forma de “como fazer” o gerenciamento de projetos, ele diz “o que deve ser feito” através dos processos que contemplam as áreas de conhecimento.

Dentro do ciclo de vida do projeto utilizando o gerenciamento através do método PMBOK ainda se espera que tudo seja documentado, a documentação e o controle são essenciais para o bom gerenciamento e andamento do projeto. Temos como obrigatório o termo de abertura do projeto e o documento de escopo do projeto, a ausência desses documentos impede o início das tarefas de desenvolvimento do projeto (CRUZ, 2013). Em um cenário de constantes mudanças e rápida evolução, Cruz (2013) ainda ressalta que projetos de desenvolvimento de *software* demandam constantemente adaptações e mudanças e que estas exigências de documentação podem atrasar ou prejudicar o cronograma da execução, tornando burocrático.

2.3 Aspectos Gerais sobre a Metodologia Ágil

A existência de fatores de competitividade empresarial, maior exigência de clientes, maior flexibilidade nas empresas e ainda a demanda por prazos de entrega mais eficazes, fez com que fossem necessários novos métodos de gerenciamento de projetos para atender as exigências.

Com essa perspectiva, os métodos ágeis de gerenciamento de projetos ganhou força e se consolidou com a formação da Aliança Ágil em 2001, que foi o estabelecimento de princípios comuns de métodos de criação de *softwares* através de grandes especialistas da área, que publicaram o denominado Manifesto Ágil, que de acordo com (BECK et al., 2001) ressalta:

- a) Indivíduos e interações melhor que processos e ferramentas.
- b) O funcionamento de *Software* melhor que documentação excessiva.
- c) A colaboração do cliente melhor que negociação de contratos.
- d) Existência de resposta a modificação melhor que seguimento de um plano.

De acordo com os autores do manifesto ágil (BECK et al, 2001), o documento teve o objetivo de consolidar as boas práticas ágeis especificamente na área de tecnologia e desenvolvimento de *software*, auxiliar no suporte aos profissionais que querem utilizar deste método e fortalecer o pensamento ágil para um desenvolvimento com maior qualidade e rapidez.

O método ágil de gerenciamento de desenvolvimento de software aqui tratado é o *Scrum*, que ao contrário do que muitas pessoas pensam não é um assunto novo, a origem dos primeiros estudos data de 1986 onde já era descrita a analogia ao *rugby* para exemplificar a relação entre os membros do time da equipe de desenvolvimento de modo cíclico e contínuo. Conforme estudo de Silva e Lovato (2016) o termo *Scrum* só se consolida em 1990 através das publicações e estudos de Schwaber e Sutherland. Segundo Schwaber e Sutherland (2013), o *Scrum* é um método simples de entender e difícil de dominar, voltado para desenvolvimento de projetos complexos como o de desenvolvimento de *software*. Este *framework* permite a execução de diferentes processos e técnicas durante o seu desenvolvimento adaptativo, produtivo e criativo. Schwaber e Sutherland (2013) ressaltam que o

Scrum possibilita a divisão do desenvolvimento de *software* em iterações, o ciclo de vida do projeto é dividido em pequenas interações de uma a quatro semanas, eliminando pontos fracos, como análise de riscos e possibilitando que ao final de cada iteração, uma versão (*release*) funcional do produto, embora restrita em funcionalidades, seja liberada ao cliente.

A metodologia ágil dá destaque a aspectos humanos no desenvolvimento do projeto, permitindo a interação na equipe de desenvolvimento e o relacionamento de cooperação com o cliente. Machado e Medida (2009) complementam que a utilização do *Scrum* depende de uma total mudança cultural das empresas e das equipes para uma completa implementação do *framework*.

Segundo Schwaber e Sutherland (2013) o *framework Scrum* está dividido em rituais e papéis conforme descrito abaixo:

- ***Sprint Planning Meeting***: realizado no início de cada *sprint* para alinhar tudo que será feito para que o produto final atenda às necessidades do cliente.
- ***Daily Scrum***: uma reunião matinal diária, curta, para que todos falem sobre o dia anterior, eventuais problemas que terão que ser resolvidos e o que pretendem fazer nesse dia.
- ***Sprint Review Meeting***: a equipe envolvida no projeto apresenta ao grupo as funcionalidades que foram implementadas durante a *sprint* em foco.
- ***Sprint Retrospective***: reinício do ciclo, usando os aprendizados da *sprint* anterior para planejar o próximo ciclo.
- ***Product Owner***: é o profissional responsável por defender os interesses do cliente final. É ele quem determina as tarefas a serem realizadas a cada *Sprint*, primeiro colocando todas as tarefas necessárias ao projeto no que se chama de *Product Backlog* e, posteriormente, transferindo alguma selecionadas para o *Sprint Backlog*, ao início de cada novo ciclo.

- **Scrum Master:** uma espécie de guardião das boas práticas do método *Scrum* e que também tenta evitar que a equipe se comprometa a fazer mais do que é possível a cada *Sprint*.

Schwaber (2007) ressalta que o método *Scrum* têm conquistado no conceito a respeito de mudanças, uma vez que os mesmos oferecem respostas rápidas ao novo ambiente de desenvolvimento, onde há uma constante mutação dos requisitos, que acabam se tornando de certa forma de difícil esclarecimento. Nesse sentido, o *Scrum* é um método de gerenciamento de projetos em sua essência ágil, pois tem em seu princípio a busca constante por aperfeiçoamento e melhoramento das práticas e a constante adaptação (SCHWABER; SUTHERLAND, 2013).

2.4 Influência do método ágil para a melhoria do processo de desenvolvimento de software

De acordo com Vora (1992), a produtividade se caracteriza como a mensuração da capacidade em se realizar a produção de um bem ou serviço. Relacionado com o conceito de engenharia de *software*, Hernández-López, Colomo-Palacios e García-Crespo (2012) ressaltam que a produtividade normalmente se define como a razão entre a extensão do produto em conjunto com o esforço necessário para produzi-lo, sendo que neste sentido o esforço é determinado em horas, enquanto o tamanho pode ser dado por distintas unidades.

Os autores orientam ainda para a existência de inúmeros fatores que afetam o resultado da medição da produtividade e eficiência, sendo fundamental que as organizações identifiquem os fatores mais relevantes para poder criar estratégias e aprimorar a eficiência de entrega das equipes. Vale observar o estudo realizado por Sampaio (2010) no qual há identificação dos fatores mais ressaltados pela literatura que mais exercem impacto na eficiência e produtividade de equipes de desenvolvimento de *software*: processo, motivação, estabilidade dos requisitos, gerência de qualidade, habilidades do time, reuso, ferramentas, linguagem de programação, tamanho do produto, tamanho do time e complexidade do *software*.

Desta forma Melo (2014) orienta que os métodos ágeis trazem a proposta do desenvolvimento de *softwares* de alta qualidade com alta da eficiência de produção das equipes, melhorando o processo e transformando-se em um enorme atrativo de

atenção das organizações que buscam destaque no mercado de trabalho e que demandam cada vez mais velocidade e qualidade em seus produtos. Em suas pesquisas, Melo (2014) demonstra que o aumento da eficiência na entrega dos projetos é um dos maiores motivadores para a adoção de métodos ágeis pelas organizações.

Ainda como orienta Melo (2015), medir a eficiência em equipes ágeis de desenvolvimento de *software* ainda é de difícil contextualização, principalmente em razão da flexibilidade que existe em um time ágil. Assim, mesmo existindo inúmeras métricas para desenvolvimento de *software*, há proposta de diversas novas abordagens para mensuração dos ganhos em ambientes ágeis: qualidade, valor, eficiência e velocidade.

Dessa forma, Melo (2015) detalha os ganhos como:

- **Qualidade:** a qualidade é contabilizada pelos números de defeitos encontrados em cada iteração, entendendo por iteração cada *sprint* do método *Scrum*.
- **Valor:** o valor que aquela entrega representa para o negócio pode ser medido através de pesquisa com os clientes para entender se a funcionalidade entregue atende a expectativa do cliente.
- **Lead-Time:** é medido contabilizando o tempo total desde da solicitação do item até a sua entrega ao cliente, é o ciclo de vida do projeto.
- **Eficiência:** a eficiência é contabilizada através da identificação dos desperdícios de tempo em cada iteração, como, atividades extras, trabalhos não finalizados, defeitos e mudanças.
- **Velocidade:** a velocidade de um time é contabilizada através das estimativas de entregas do time, que podem ser medidas por: pontos de função, *story points* e etc.

É importante citar ainda Jensen (2015), que ressalta a relação existente entre eficiência e a velocidade de desenvolvimento, sendo que a velocidade possui relação direta com o custo de tempo e energia gasta para a realização das tarefas. Neste contexto, a métrica escolhida para comparação da avaliação da melhoria do processo de desenvolvimento de *software* com a adoção do método ágil foi o ciclo

de vida do projeto, pois é possível a comparação de todo tempo gasto no ciclo de vida de um projeto quando utilizado o método tradicional baseado no PMBOK com o método ágil *Scrum*, a escolha dessa métrica também é devida aos dados disponíveis para o estudo de comparação.

3. METODOLOGIA

3.1 Estratégia de Pesquisa

Esta seção abrange os caminhos metodológicos que foram seguidos para realizar a pesquisa. Para Rampazzo (2005), a pesquisa é definida como um processo de foco reflexivo que está ordenado e tem cunho crítico, ela abrange descobertas de fatos ou soluções novas em diferentes áreas de conhecimento e é realizada através de método científico.

Este estudo é do âmbito qualitativo, com objetivo de responder da melhor forma os pontos ressaltados acerca do tema da pesquisa, que por sua vez não é quantificado e não pode ser traduzida em números (CHIZZOTTI, 2003).

Entre os tipos de pesquisas qualitativas, o estudo de caso único é a estratégia adequada quando se analisa eventos contemporâneos onde não se pode manipular os comportamentos relevantes (YIN, 2001). Segundo Collis e Hussey (2009) eles complementam que é uma metodologia para explorar um fenômeno em seu contexto para obtenção de conhecimento em profundidade. O estudo de caso é um método de pesquisa amplamente utilizado na área de sistemas de informação (SHANKS; BEKMAMEDOVA, 2018). Este estudo ainda pode ser classificado como descritiva, pois visou descrever as características do objeto de estudo e estabelecer relação de comparação dos resultados.

3.2 Seleção do Caso

O universo da pesquisa foi composto pelas equipes de gerentes de projetos e equipes de desenvolvimento de *software* de uma grande organização do setor financeiro especializada em recuperação de crédito. Dentro dessas equipes incluem os papéis de gerentes de projetos, coordenador de times, desenvolvedores de *software*, analistas de testes, *scrum masters*, *product owners* e analistas de portfólio

de projetos. Vale ressaltar que os integrantes do universo da pesquisa participaram ativamente em projetos nos dois tipos de métodos aqui estudados, o tradicional baseado no PMBOK e no método ágil utilizando o *Scrum*. As amostras foram não probabilísticas (DAVIES; HUGHES, 2014) pelos critérios de acessibilidade e tipicidade (VERGARA, 2003), dado que o pesquisador teve acesso às equipes de projeto e conhecimento dos papéis exercidos pelas pessoas dentro dessas equipes. Importante ressaltar que a organização pesquisada tem um alto nível de maturidade na condução dos seus projetos de desenvolvimento de *software* baseado no PMBOK, em 2012 foi eleita uma das três melhores empresas do Brasil em gestão de projetos pela Mundo PM, revista da área de gestão de projetos.

3.3 Coleta do Dados

Nesta parte do estudo, foram utilizados conceitos da triangulação de dados e da triangulação metodológica, utilizando diferentes fontes de dados e diferentes técnicas e métodos de coleta de dados, com o objetivo de se alcançar a validade do estudo e obter diferentes perspectivas do mesmo fenômeno (VERGARA, 2005). A coleta dos dados da pesquisa foi realizada em janeiro de 2019 referente ao período de apuração do ano de 2018, foram utilizadas as técnicas de pesquisa documental, sites da organização, observação e coleta de dados qualitativos por meio de entrevistas semiestruturadas.

3.3.1 Pesquisa documental e sites as organização

Informações sobre a organização, sua estrutura e seus processos de gestão de projetos foram obtidos por meio de pesquisa documental e acesso aos sites da internet e da intranet da organização, assim como dados oficiais dos resultados dos projetos de desenvolvimento de *software*, visando a coleta de dados para apoiar a análise das entrevistas e a análise do resultado do questionário.

3.3.2 Observação

O pesquisador agiu como um observador na coleta de dados contextuais sobre a organização, sua estrutura, papéis e processos, que foram utilizados para ajudar na interpretação dos dados extraídos nas entrevistas e nas pesquisas documentais (SAUNDERS; LEWIS; THORNHILL, 2012).

3.3.3 Entrevista semiestruturada

A coleta de dados qualitativos foi realizada com a técnica de entrevistas individuais, semiestruturadas e realizadas frente a frente dos entrevistados (WILLIAMSON, 2013), uma forma legítima de coleta para estudos de natureza exploratória (SAUNDERS; LEWIS; THORNHILL, 2012), buscando identificar os conhecimentos, as atividades da gestão de projetos de desenvolvimento de *software* e os fatores e percepção dos entrevistados quanto a condução dos projetos nas duas formas de métodos utilizados, dado que por meio dessa técnica é possível conforme (SIMONS, 2009) capturar a perspectiva do entrevistado, promover comprometimento e aprendizado dos envolvidos na questão, permitir alterações no rumo da entrevista caso surjam tópicos e dialogar com entrevistados.

O roteiro que orientou a entrevista foi criado com o objetivo de entender as percepções dos entrevistados quanto a mudança de metodologia de desenvolvimento de *software* na organização, além das atividades exercidas anteriormente e atualmente, assim como as mudanças de papéis, e os fatores mais relevantes que influenciaram na forma de trabalho e na melhoria do processo de desenvolvimento de *software* das equipes.

As entrevistas foram realizadas em local reservado, a identidade, a privacidade e a confidencialidade dos entrevistados foram mantidas. Ao todo foram realizadas cinco entrevistas com diferentes níveis e papéis dentro dos projetos, conforme identificado no quadro abaixo:

Quadro 1 – Características dos entrevistados

Entrevista Semiestruturada			
Entrevistado	Papel na Organização	Experiência na Organização	Tempo de Entrevista
1	Gerente de Projetos	Mais de 5 anos	30 minutos
2	Gerente de TI	Mais de 10 anos	40 minutos
3	Gerente de Portfólio	Mais de 10 anos	40 minutos
4	<i>Product Owner</i>	Mais de 10 anos	30 minutos
5	<i>Scrum Master</i>	Mais de 2 anos	30 minutos

Fonte: Elaboração Própria

3.4 Análise do Dados

A análise dos dados da pesquisa foi realizada com base nos dados qualitativos coletados nas entrevistas e com o apoio das informações obtidas na pesquisa documental e na observação. Os dados qualitativos foram analisados por meio da análise de seu conteúdo, que tem o objetivo exploratório (VERGARA, 2005). Para a análise do conteúdo foi feita uma pré-análise, que é uma fase de organização onde foi selecionado o material e definido os passos a serem executados. Na fase de exploração do material, foram seguidos os passos definidos anteriormente, e na última fase de tratamento dos dados, os dados foram tratados e interpretados por meio de inferências. Conforme orienta Bardin (1977), a análise do conteúdo foi realizada em três fases: pré-análise, exploração e tratamento dos dados.

Os dados foram transcritos, analisados e tratados, contudo, o pesquisador foi o responsável pela interpretação. O procedimento básico na realização da análise do conteúdo foi a definição de categorias relacionadas aos propósitos da pesquisa (VERGARA, 2005). As categorias para análise foram definidas conforme a grade mista de análise, na qual se define preliminarmente as categorias pertinentes ao objeto de pesquisa, porém admite-se a inclusão de categorias surgidas durante o processo de análise (VERGARA, 2005).

Por fim, a análise foi feita através de categorias de comparação entre os projetos feitos nos dois métodos de gestão de desenvolvimento de *software*, tendo como objetivo principal comparar os resultados de tempo de ciclo de vida do projeto para verificar a avaliação de melhoria no processo.

4. ESTUDO DE CASO E ANÁLISE DOS RESULTADOS

4.1 Organização

A organização escolhida para o estudo de caso foi uma empresa brasileira de grande porte do ramo financeiro de análise crédito, a empresa tem um total de 3.000 funcionários, sendo 500 da área de tecnologia da informação. O foco deste trabalho foi em projetos de desenvolvimento de *software* de uma das principais e mais rentáveis área de negócio da organização. Esta área de negócio é responsável por 40% de todo o faturamento da organização, entre os produtos desenvolvidos por esta área, cinco deles estão em primeiro lugar na lista dos produtos mais rentáveis da organização.

A organização desenvolvia seus projetos de tecnologia internamente, com mão de obra própria e com auxílio de fábricas de *software*, em média o desenvolvimento interno representa 50% e a mão de obra com fábricas de *softwares* os outros 50%. Na área de negócio estudada havia em média vinte projetos em desenvolvimento simultâneo que abrangiam em torno de trinta sistemas em manutenção ou evolução. Ao todo a área de negócio dessa pesquisa tem cem aplicações e sistemas em uso, que representam 30% de toda a área de tecnologia de sistemas da organização.

Para desenvolver os projetos, eram formados grupos de habilidades diferentes conforme a necessidade tecnológica, normalmente era composto por: um gerente de projetos, um líder técnico, um líder de qualidade de *software* e de seis a oito desenvolvedores de *software* e analistas de qualidade de *software*, e após o término do projeto a equipe era desmobilizada. Após a adoção aos métodos ágeis, foram formadas três *squads* para atender esta área de negócio, estas *squads* são fixas e compostas por seis a oito desenvolvedores de *software* e analistas de qualidade de *software* e mais um *scrum master* e um *product owner*.

4.2 Metodologia de gestão de projetos da organização

A organização possuía uma diretoria específica para gestão de portfólios e projetos de desenvolvimento de *software*, onde toda a gestão dos projetos de *software* era adaptada e realizada baseado no PMBOK do PMI.

Os projetos eram divididos em cinco fases: iniciativa, priorização, planejamento, execução e transição, em cada fase existiam pontos de controle. As fases estabeleciam as atividades que deveriam ser executadas em cada momento e cada ponto de controle determinava os requisitos e os documentos necessários para avançar as fases. O projeto era gerido pelo papel do gerente de projetos que assumia a gestão dos custos, riscos, comunicação, cronograma, mudanças e aquisições. A equipe do projeto era composta pelo gerente de projetos, desenvolvedores de *software*, analistas de qualidade de *software*, apoio de líder técnico, analista de negócios, e ainda contavam com um escritório de projetos e um patrocinador que possuíam papéis e responsabilidades estratégicas quanto ao projeto e seus produtos.

A equipe de projeto tinha a sua disposição um conjunto de *templates* de documentos que deveriam fazer a cada fase do projeto, com destaque para alguns:

- Fase de Iniciativa: Criação de documentos de conceitos da demanda, utilizados pela área de negócio para propor e detalhar novos produtos.
- Fase de Priorização: Criação de documentos de estimativas de horas que serão gastas no projeto e ata do comitê de priorização do projeto.
- Fase de Planejamento: Documentos da solução adotada, cronograma, termo de abertura do projeto e declaração oficial do início do projeto para todos os envolvidos.
- Fase de Execução: Documentos de requisitos de *software*, casos de testes, cenários de testes, diagramas diversos que remetem ao funcionamento do sistema.
- Fase de Transição: Documentos de manuais dos sistemas que remetem a sua utilização para disponibilizar para clientes e equipe de manutenção dos sistemas.

No âmbito dos custos do projeto, o valor que seria gasto para cada projeto era determinado na fase de iniciativa, onde era coletada a informação de estimativa de tempo e custo pelos especialistas do projeto antes de seu início. Neste momento também era determinado o prazo do projeto baseado na estimativa de horas e no

escopo apresentado. Dessa forma, com o custo, prazo e escopo definidos, o projeto era iniciado.

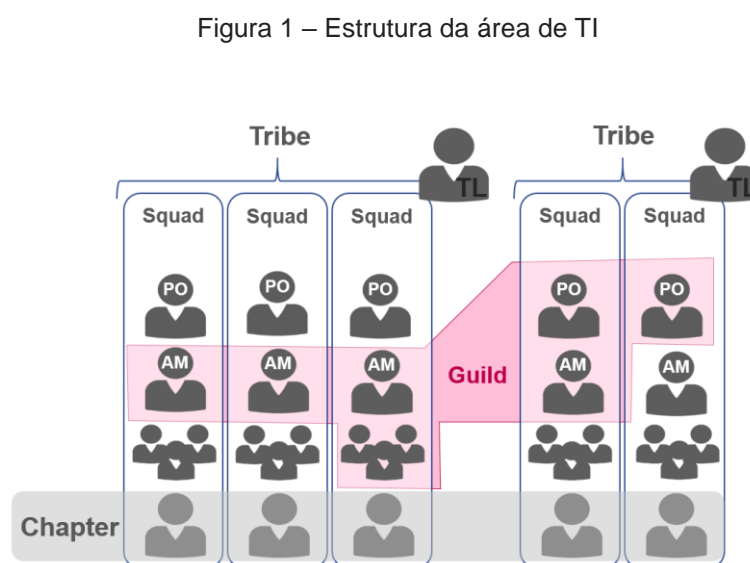
Em 2018, a organização passou por uma mudança de estratégia e passou a adotar métodos de gerenciamento de *software* ágil, o principal *framework* escolhido foi o *Scrum*.

4.3 Adoção de método ágil na organização

A organização passou por uma grande mudança em 2018 com a chegada de um novo vice-presidente de tecnologia da informação, o seu objetivo era transformar a área de tecnologia da informação propondo uma transformação digital. O principal agente que desencadeou essa necessidade de mudança foi o crescimento da concorrência, *fintechs* e *startups* que prometem entregar em menor tempo as necessidades de seus clientes. Uma das frentes da transformação digital foi a adoção de métodos ágeis para o gerenciamento de projetos de tecnologia.

Com a adoção do *framework Scrum* no gerenciamento de projetos de TI da organização, houve uma reestruturação da área de TI e novas definições de papéis.

Os papéis da área de TI foram definidos conforme ilustração abaixo:



Fonte: Elaboração Própria

- *Tribe* – Antes chamado de Gerência de TI, agrupa os integrantes do projeto de forma multidisciplinar e trata de assuntos de uma área de negócio específica.
- *TL (Tribe Leader)* – É o papel do antigo gerente de TI, que cuida das questões administrativas e processuais dos integrantes da *Tribe*.
- *Squad* – Formada por colaboradores multidisciplinares que atendem a evolução e manutenção de determinados produtos que fazem parte da sua *Tribe*.
- *PO (Product Owner)* – É o papel responsável pelo produto, sua evolução e manutenção e que vai guiar a *Squad* quando as atividades que serão executadas. Ele é responsável pelo projeto, exercendo a liderança sobre o produto que está sendo desenvolvido, assumindo a gestão das seguintes disciplinas: benefícios, engajamento das partes interessadas, mudanças, impactos organizacionais, e detalhamento do escopo
- *AM (Agile Master)* – É o papel responsável pela gestão do funcionamento da *Squad*, garantindo que o método ágil seja seguido e os eventos e artefatos do *framework Scrum* sejam utilizados e adaptados da melhor maneira.
- *Chapter* – São áreas de apoio que exercem contribuição com papéis específicos. Não fazem parte da equipe fixa do projeto, são alocados conforme a necessidade, como arquiteto de solução, analista de infraestrutura e analista de dados.
- *Guild* – Comunidade formada por especialidades para troca de informação, conhecimento e padrões, garantindo a disseminação do conhecimento entre todos e novos aprendizados.

No caso dos projetos de desenvolvimento de *software*, durante a fase de execução, para elaboração do sistema de informação, começaram a ser utilizados eventos adaptados do *framework Scrum* (SCRUM.ORG, 2019) como: *Sprint Planning*, *Daily Scrum*, *Sprint Review* e *Sprint Retrospective*.

O ciclo de vida do projeto é dividido em *releases* *plannings* e *sprints*. A *release planning* tem duração de até cinco dias dedicados para o planejamento do projeto, nela são analisadas as demandas no âmbito de refinamento dos itens a serem

feitos, priorização do escopo, análise de impactos com outras áreas como riscos, jurídico, *compliance* e estipulado a estimativa de prazo para execução.

Após isso, o projeto é executado em *sprints*, ciclos de três semanas para desenvolvimento. A cada fim de ciclo a forma de trabalho é revista através da retrospectiva para avaliar como foi o andamento desse ciclo e o que pode ser feito para melhorar o próximo ciclo, através dessa inspeção é feita a melhoria contínua do processo.

4.4 Análise dos resultados

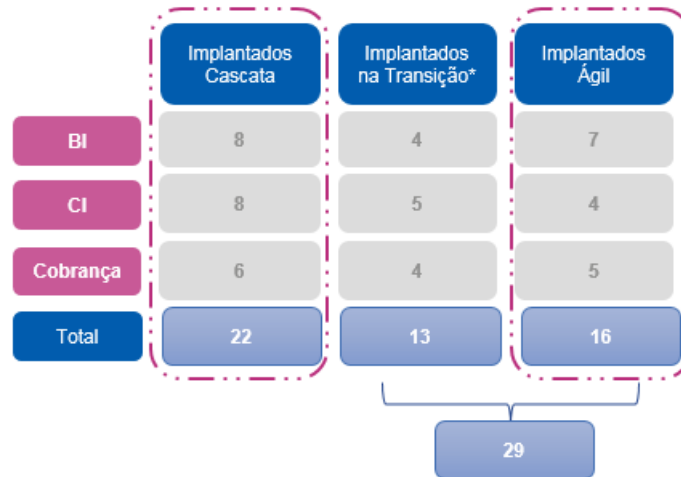
Em vista dos objetivos gerais e específicos da pesquisa, a análise do resultado levou em consideração algumas premissas:

- O período de abrangência da coleta de dados utilizada foi considerado de abril de 2017 até agosto de 2018 para projetos geridos no modelo tradicional de gestão de projetos baseado no PMBOK e o período de abril de 2018 até agosto de 2018 para projetos geridos com o *framework Scrum* no modelo ágil de gestão de projetos.
- A escolha dos projetos para comparação foi baseada em custo, prazo e escopo semelhantes e com considerável representatividade financeira para a organização.
- Não foram considerados na análise final projetos em fase de transição, onde iniciaram no modelo tradicional de gestão de projetos e foram finalizados já no modelo ágil de gestão de projetos.
- Considerados dois tipos de projetos, o projeto denominado “*Features*” que é a criação de relatórios em um produto já existente, e o projeto denominado “Novo Produto” que parte da criação de um novo produto do zero.

A área de negócio onde foram analisados os projetos é denominada *Credit Services*. Nela compõe projetos de três categorias de produtos aqui denominadas BI, CI e

Cobrança. Considerando o período da coleta de dados, temos conforme ilustrado na figura abaixo a quantidade de projetos que foram implantados neste período.

Figura 2 – Quantidade de projetos finalizados



Fonte: Elaboração Própria

Como visto na figura acima, se comparado apenas a quantidade de projetos implantados no modelo tradicional de projetos denominado na figura como “cascata” com a quantidade de implantados no modelo ágil, a quantidade é menor, visto que boa parte dos projetos analisados no período foram híbridos, ou seja, começaram no modelo cascata e terminaram no modelo ágil. Se considerarmos todos projetos que foram implantados no modelo ágil independente se começaram no ágil ou não, temos um valor maior que no modelo cascata.

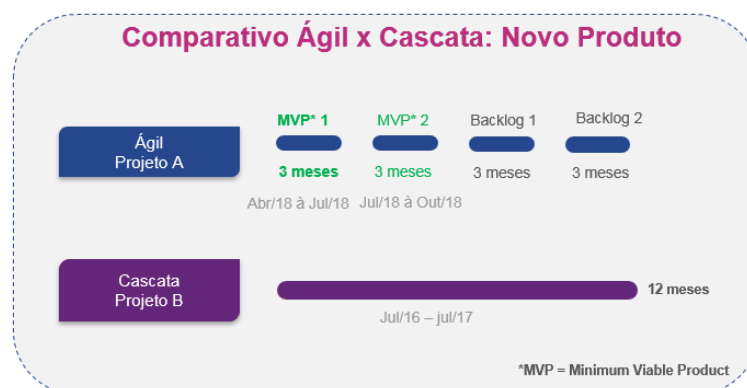
Comparando um projeto do tipo “*features*” que são criações de relatórios em um produto já existente, e considerando projetos da mesma natureza e similares em escopo, custo e prazo, temos uma redução de 81% na quantidade de dias que estes projetos levaram e um ganho de redução em 113 dias na sua entrega, conforme ilustra a figura abaixo:

Figura 3 – Redução no ciclo de vida dos projetos do tipo *features*

Fonte: Elaboração Própria

Comparando um projeto do tipo “Novo Produto” que é a criação de um novo produto do zero, e considerando projetos da mesma natureza e similares em escopo, custo e prazo, nota-se que em questão de prazo, levou o mesmo tempo de 12 meses, porém, utilizando de entregas de MVP (*Minimum Viable Product*), ou Produto Mínimo viável em português, onde com ciclos de entregas mais curtos já é possível liberar uma parte do produto desenvolvido para o cliente de forma que esta entrega tenha um valor agregado esperado pelo cliente, foi notado um ganho na entrega de valor faseada, maior agilidade na integração com as pessoas e o cliente, e respostas rápidas para mudanças de escopo e imprevistos. No modelo cascata levou 12 meses para o cliente ter o seu sistema entregue e começar a lucrar com ele, já no modelo ágil, em 3 meses já tinha uma entrega de valor onde parte do produto já poderia ser comercializada, conforme ilustra a figura abaixo:

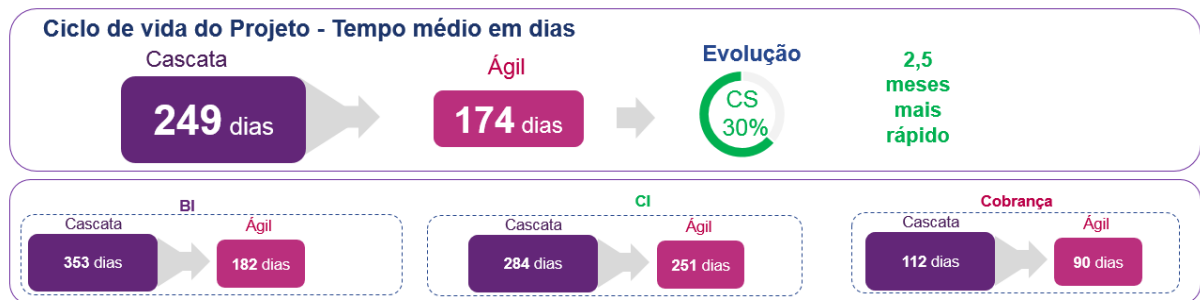
Figura 4 – Redução no ciclo de vida dos projetos do tipo novo produto



Fonte: Elaboração Própria

Considerando os projetos abordados na pesquisa, relacionados a área de negócio denominada *Credit Services* (CS), no período analisado, foi constatado uma redução no tempo médio do ciclo de vida dos projetos (tempo total do seu início até o seu fim) de 2,5 meses, em forma mais detalhada o tempo em dias da duração dos projetos divididos nas três categorias que abrange a área de CS, conforme ilustra a figura abaixo:

Figura 5 – Visão geral da redução em dias dos projetos



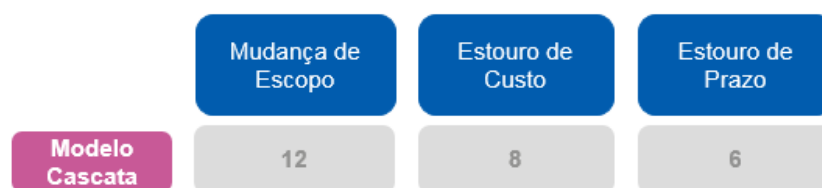
Fonte: Elaboração Própria

Diante do ciclo de vida de projetos de desenvolvimento de *software* geridos no modelo cascata e no modelo ágil, temos como ponto de distinção destacado pela literatura ao que abrange a parte de documentação. Conforme PMBOK (2013), o controle e elaboração de documentações são essenciais para o andamento do projeto. Já para o modelo *Scrum*, Schwaber e Sutherland (2013) visam um trabalho com menor documentação que registre apenas alguns elementos de suma importância para a conclusão do projeto. Nesse contexto, ao trabalhar com o modelo cascata baseado no PMBOK temos alguns documentos obrigatórios como o termo de abertura do projeto e a definição de escopo do projeto que se não forem concluídas impedem que o projeto prossiga (CRUZ, 2013). Já na utilização do modelo ágil *Scrum*, não há obrigatoriedade de formalização. Cruz (2013) ainda destaca que o que presa é a comunicação verbal, documentação simples e os combinados com os envolvidos. Dessa forma, o escopo e planejamento vão se moldando ao longo do ciclo de vida do projeto, e pode iniciar o projeto utilizando o mínimo de requisitos e definições na análise inicial. Com isso, nota-se que a definição do escopo durante os projetos que foram geridos pelo modelo ágil contribuiu para uma entrega com menor tempo de conclusão por não ser uma barreira e ponto de controle.

No âmbito de custos dos projetos, Kardec (2012) ressalta que no PMBOK deve ser calculado após a formalização e aprovação do escopo e da análise de requisitos, ao contrário do *Scrum*, que trabalha com itens do *backlog* para estimar o custo e distribuir as atividades. Os itens são estimados conforme a necessidade de priorizá-los para o desenvolvimento e não mais todo o escopo do projeto. Essas diferenças afetam diretamente o cronograma de execução do projeto. Nota-se nos projetos do estudo que é uma característica dos projetos de desenvolvimento de *software* da organização a sua constante mudança de escopo que demanda adaptações constantes, o que leva ao estouro do custo estimado dos projetos.

Em 22 projetos analisados no modelo cascata, 12 projetos tiveram mudança do escopo, sendo necessário reavaliar o custo versus benefício. Já em 8 desses projetos houve estouro do orçamento original onde os principais motivos foram, mudança de escopo, estimativa inicial incorreta devido a longevidade do projeto e necessidades não estimadas anteriormente. Em 6 dos projetos houve estouro do prazo de entrega, entre os principais motivos estão, escopo não entendido corretamente, mudanças no meio do caminho onde elevaram o retrabalho, estimativa incorreta devido ao tamanho do escopo, mudanças de tecnologia ao longo do projeto, entre outros.

Figura 6 – Mudanças de escopo, custo e prazo



Fonte: Elaboração Própria

Em 16 projetos analisados no modelo ágil, não houve constatação de mudanças não planejadas dentro do ciclo de iteração de três semanas, visto que o planejamento de escopo, custo e prazo são definidos dentro de cada ciclo.

Conforme Cruz (2013), o escopo é o principal elemento e demanda maior dificuldade no seu gerenciamento de projetos de *software*, devido a própria natureza tecnológica de constantes avanços. Dessa forma, o PMBOK (2013) diz que o

escopo é definido na fase de planejamento do projeto e deve ser aprovado pelo cliente e que, qualquer mudança no escopo precisa de um processo de gerenciamento de mudanças para reavaliar o custo versus o benefício dessa alteração, impactando assim no custo predefinido do projeto. Já com a utilização do método *Scrum*, o controle do escopo é iterativo. A cada *sprint* são revisadas as necessidades do projeto e seus requisitos e definidos os itens que serão desenvolvidos. O planejamento desses ciclos, que nos projetos analisados teve duração de três semanas cada ciclo, permite uma maior adequação do escopo, alterações e mudanças de forma ágil.

Para Schwaber e Sutherland (2013), a flexibilidade no planejamento e nas mudanças de escopo é uma necessidade da área de desenvolvimento de *software*, e nessa linha, o método ágil considera estas mudanças como natural e parte do processo de evolução da equipe. Já para o PMBOK (2013), o escopo deve sofrer o mínimo de mudanças, e se ocorrer, devem ser revistas, replanejadas, analisadas, documentadas, aprovadas e por fim, aceitas no projeto.

O ciclo de vida de um projeto gerido com *Scrum* é um evento com tempo pré-determinado, nos projetos analisados tinham duração de três semanas cada ciclo. O time de desenvolvimento que define este tempo no começo do projeto e deve manter até sua conclusão, dessa forma, o ciclo é iterativo, com eventos sequenciais que a cada ciclo revisam e planejam o próximo desenvolvimento. Nos projetos que utilizaram o modelo do PMBOK, foi utilizado o ciclo de vida preditivo, onde cada etapa do projeto só é realizada se a anterior foi finalizada.

Outro ponto de distinção dos projetos analisados que vale ressaltar é que os dois métodos tratam como importante o aprendizado e as lições aprendidas. No modelo *Scrum*, ao final de cada ciclo é realizado a reunião de retrospectiva para avaliar os pontos aprendidos e tomar ações para que o próximo ciclo melhore. Já nos projetos geridos pelo modelo baseado no PMBOK, a reunião de lições aprendidas é feita apenas no final do projeto, durante o processo de encerramento do projeto. A discussão dos pontos a melhorar e das lições aprendidas dentro de cada *sprint* (ciclos de três semanas) permite uma melhoria mais rápida dentro do próprio projeto, focando na qualidade do produto. Quando feito apenas no final do projeto, pode perder oportunidades de melhoria dentro do projeto, gerando assim retrabalho.

Conforme PMBOK (2013) os processos são divididos em cinco grupos bem definidos. Os cinco grupos de processos são: Iniciação, Planejamento, Execução, Monitoramento e Controle, e Encerramento. Dentro destes grupos existem 10 áreas de conhecimento divididas em gerenciamento de integração, escopo, tempo, custos, qualidade, recursos humanos, comunicações, riscos, aquisições e partes interessadas. Com base nestas categorias levantadas no referencial teórico e categorizadas na pesquisa semiestruturada, foi criado um resumo das principais mudanças no processo de ciclo de vida de desenvolvimento de *software* e na forma de trabalho:

Quadro 2 – Visão geral das mudanças no processo de adoção do ágil

Grupos de Processos	Área de Conhecimento	PMBOK	Scrum
Iniciação	Integração	Foco no planejamento. Para iniciar o projeto precisa de definições bem claras e precisas de custo, prazo e escopo.	Foco na entrega de valor do produto. Trabalha com incertezas e priorização durante o trabalho.
Planejamento	Escopo	O escopo é bem definido e detalhado, e serve de base para as estimativas de custo e tempo.	Utiliza a abstração de requisitos, com escopo inicial e detalhamento no decorrer do ciclo de vida do projeto.
Planejamento	Escopo/ Documentação	Documentação rigorosa, muitos <i>templates</i> e pontos de controle. Fator impeditivo para mudança de etapa. A documentação é extensa e desenvolvida pelo papel do Gerente de Projetos.	A documentação é secundária, produzida conforme a necessidade e quando os ciclos são realizados. Documentação simples, resumida e com linguagem fácil.
Planejamento	Custos	Os custos são estimados com base no escopo e análise dos requisitos do projeto. Existem processos específicos que gerenciam os custos do projeto. A inexistência do escopo inviabiliza a estimativa de custo do projeto.	Os custos são estimados através do <i>Product Backlog</i> , agilizando o início da execução pois os detalhamentos são realizados a cada iteração. Não há um processo específico que trate do assunto.

Planejamento	Tempo	Ciclo de vida definido com as etapas de: iniciação, planejamento, execução, controle e encerramento.	Ciclo de vida baseado em ciclos de sprints, o projeto tem seu escopo detalhado ao longo do desenvolvimento. As reuniões de planejamento e revisão da Sprint definem o detalhamento do escopo do período e as possíveis mudanças necessárias. O ciclo tem duração fixa de 3 semanas.
Execução	Recursos Humanos	Processos são bem definidos e devem ser executados obrigatoriamente para garantir o desenvolvimento de qualidade. O gerente deve distribuir as atividades e montar o cronograma.	As pessoas são tratadas como indivíduos e não recursos. O ritmo de trabalho é respeitado, cada membro da equipe escolhe quais tarefas irá realizar e estimam o tempo a ser gasto das tarefas que irá desenvolver na próxima iteração.
Execução	Comunicações	A comunicação é planejada através da identificação das necessidades das partes e determinação dos meios adequados. Devem ser gerados documentos formais de comunicação para registrar e divulgar as conclusões e informar as partes interessadas.	A comunicação é interpessoal, realizada nas reuniões de planejamento e revisão da Sprint, e reuniões diárias. São reuniões objetivas, rápidas e com pauta preestabelecida.
Execução	Partes Interessadas	O cliente não está presente em todas as etapas. Participando apenas da definição do escopo e na entrega final.	Participação ativa do cliente (<i>Product Owner</i>) como parte do time do projeto.
Monitoramento e controle	Integração/mudanças	As mudanças são consideradas de grandes riscos, e permitidas após análise de impacto, de custos e de aprovação.	As mudanças são consideradas inerente ao processo de amadurecimento da equipe e são aceitas e implementadas nos ciclos iterativos.
Encerramento	Integração	Ao final do projeto é realizada reunião para arquivamento de documentos e leitura das lições aprendidas durante o projeto.	O processo de lições aprendidas é realizado ao final de cada Sprint durante a reunião de retrospectiva. Sendo possível implementar as melhorias já no próximo ciclo.

Fonte: Elaboração Própria

Contudo, nota-se também que a organização possuía uma robusta estrutura e processos bem definidos para o desenvolvimento de projetos utilizando o PMBOK e que foi evoluído durante anos através de políticas e práticas internas. Já com a adoção para o *framework Scrum* e pouco tempo para essa adaptabilidade, é possível notar diversos gaps como: conflito de papéis e responsabilidades, maturidade das pessoas na função, o mesmo profissional dividindo várias funções, conflitos de interesse nas prioridades dos projetos, responsabilidade pelos custos e controle financeiro de responsabilidade indefinida, utilização de processos antigos do modelo cascata, indefinição do processo de aquisições para contratação de empresas terceirizadas para atuar no projeto e principalmente a falta de adoção do modelo ágil para toda a empresa e não somente na área de tecnologia da informação.

Sem dúvida, há espaço para melhorias, principalmente quanto aos processos de gerenciamento de custos, priorização dos projetos e mudança da cultura organizacional.

5. CONCLUSÃO

O desenvolvimento de *software* tem uma natureza complexa e inovadora que exige a utilização de processos para o gerenciamento do desenvolvimento de projetos que acompanhem este nível de inovação sem burocratizar ou inibir a criatividade inerente à esta área. O gerenciamento de projetos com metodologias ágeis surge para suprir esta necessidade, propondo desenvolver com qualidade e de forma flexível. Os diferentes *frameworks* ágeis são derivados do Manifesto Ágil, e o *Scrum* é atualmente o mais conhecido e utilizado devido a sua estrutura e eficiência dos resultados. A proposta do *framework Scrum* torna naturalmente compatível com a complexidade dos projetos de *software*, e a característica de planejar conforme o desenvolvimento evolui vai de encontro com as mudanças tecnológicas constantes, porém, tal mudança pode ser de difícil aceitação para organizações tradicionais, pois existe uma necessidade natural de controle e conseqüentemente existe uma resistência para adoção do método em toda empresa e do entendimento dos ganhos reais da sua utilização. Baseado neste contexto, surge a pergunta de pesquisa “Qual o ganho na avaliação de melhoria do processo de desenvolvimento de *software* com a utilização do método ágil *Scrum* na empresa estudada?”, o objetivo geral deste

estudo foi identificar através da métrica de tempo de ciclo de vida dos projetos o ganho no prazo de entrega dos projetos e as suas influências nesse processo em comparação com a utilização do método tradicional de gestão de projetos baseado no PMBOK.

Para atingir este objetivo, em uma primeira etapa, foi realizada uma pesquisa exploratória da literatura sobre as duas abordagens de gerenciamento de projetos, o que incluiu autores e obras relevantes. Por meio dessa pesquisa, foi possível estabelecer uma comparação entre as formas de trabalho.

Na sequência foi realizado um estudo de caso único em uma organização de grande porte que passou recentemente por mudanças na condução de seus projetos de desenvolvimento de *softwares*. Este estudo envolveu principalmente a condução de entrevistas abertas que capturaram as visões dos diversos membros das equipes de projetos de desenvolvimento de *software* que participaram de projetos na organização nas duas formas de condução: tradicional e ágil, de forma secundária também foram utilizados base de conhecimento oficiais dos projetos da empresa.

De acordo com os resultados da pesquisa, de forma geral, observa-se ganhos no processo de desenvolvimento de *software* ao que se refere ao tempo de ciclo de vida do projeto e entrega antecipada de valor ao cliente. Notou-se uma redução em média de 30% do tempo gasto anteriormente para a entrega de um projeto ao cliente final. No âmbito da criação de um novo produto notou-se que não houve mudança no tempo total do ciclo de vida do projeto, porém, com a utilização do Produto Mínimo Viável, onde com ciclos de entregas mais curtos já é possível liberar uma parte do produto desenvolvido para o cliente de forma que esta entrega tenha um valor agregado esperado pelo cliente, foi notado um ganho na entrega faseada, maior agilidade na integração com as pessoas e com o cliente, e respostas rápidas para mudanças de escopo e imprevistos.

Ademais, também no estudo de caso, foi possível avaliar a percepção das equipes com relação às mudanças nos papéis, processos e atividades nos projetos. Nessa avaliação, além da comparação da forma de trabalho e das atividades que eram feitas no modelo tradicional e como são feitas no modelo ágil, foi possível observar os pontos que ainda não estão maduros e passíveis de evolução.

Como a adoção de métodos ágeis para o gerenciamento de desenvolvimento de *software* está em alta nas empresas que estão passando pela transformação digital, este trabalho contribui de forma geral com a discussão sobre os ganhos na adoção

desse método e de maneira específica com a literatura que versa sobre a comparação de modelos tradicionais com modelos ágeis. No âmbito da contribuição prática, o estudo de caso foi realizado em uma grande empresa que passou recentemente pela mudança na metodologia de condução de seus projetos e precisa mostrar os seus ganhos, dessa forma o estudo contribui na forma de identificação das principais mudanças de papéis e atividades, identificação dos ganhos na melhoria dos processos adotados, e identificação dos pontos que ainda mostram falhas de processos e definições e que precisam ser trabalhados pelos gestores da organização para colaborar com a melhoria de seus resultados. Existem limitações neste trabalho que podem ser exploradas como oportunidades para outros pesquisadores da área na condução de novas pesquisas.

Por se tratar de uma pesquisa qualitativa e bibliográfica há uma limitação natural frente ao universo literário existente sobre cada uma das abordagens e sua aplicabilidade nas organizações.

Com o objetivo de capturar uma visão ampla do assunto, este estudo não se aprofundou em cada uma das fases e atividades do gerenciamento de projetos.

Diante da natureza do estudo de caso, os resultados da pesquisa não podem ser generalizados, contudo eles podem servir como base na avaliação de contextos similares. Além do mais, cabe salientar que os resultados desta pesquisa foram baseados na interpretação do pesquisador dos dados coletados. Este estudo capturou de forma ampla a correlação das fases e atividades das duas abordagens em projetos de desenvolvimento de *software*, dessa maneira, sugere-se que novas pesquisas em profundidade sejam realizadas com o foco em cada uma das fases e etapas do gerenciamento de projetos, com o objetivo de compreender e esclarecer detalhes e as relações entre si.

Sugere se também a realização de um estudo mesclando as duas abordagens, de forma que elas sejam complementares e não contrárias, visto que na abordagem ágil ainda há processos que não são detalhados como conduzi-los.

Sugere-se ainda a realização de outros estudos de caso em diferentes contextos de forma a complementar e consolidar o conhecimento aqui estruturado.

Ademais, com o objetivo de apresentar relações e causas entre os diversos elementos da pesquisa, sugere-se também a realização de estudos de natureza quantitativa com o estabelecimento de variáveis para medir o impacto de cada abordagem com a avaliação de melhoria nos processos.

REFERÊNCIAS

- ALBERTIN, A. L. Administração de informática: funções e fatores críticos de sucesso. 5. ed. São Paulo: Atlas, 2004.
- ALBERTIN, A. L.; ALBERTIN, R. M. de M. Tecnologia de informação e desempenho empresarial: as dimensões de seu uso e sua relação com os benefícios de negócio. 3. ed. São Paulo: Atlas, 2016a.
- BARDIN, L. Análise de conteúdo. Lisboa, Portugal: Edições 70, 1977.
- BARROCA, L.; SHARP H.; SALAH, D.; TAYLOR, K.; GREGORY, P. Bridging the gap between research and agile practice. *International Journal Systems Assurance Engineering Management*, p. 01-12, 2015.
- BECK, K., et.al. "Manifesto for Agile *Software Development*." 2001. Disponível em: <<http://AgileManifesto.org>>. Acesso em fev. 2019.
- BLOCH, M.; BLUMBERG, S.; LAARTZ, J. Delivering large-scale IT projects on time, on budget, and on value. McKinsey & Company, 2012. Disponível em: <<https://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/delivering-large-scale-it-projects-on-time-on-budget-and-on-value>>. Acesso em fev.2019.
- COLLIS, J.; HUSSEY, R. Business research: a practical guide for undergraduate and postgraduate students. 3. ed. New York: Palgrave Macmillan, 2009.
- CHIZZOTTI, A. Pesquisa qualitativa em ciências humanas e sociais. Petrópolis: Vozes, 2003.
- CRUZ, F. *Scrum* e Guia PMBOK unidos no gerenciamento de projetos. Rio de Janeiro: Brasport, 2013.
- DA SILVA, E., & LOVATO, L. *Framework Scrum: Eficiência em Projetos de Software*. Revista de Gestão e Projetos GeP, p. 01-15, 2016.
- DAVIES, M. B.; HUGHES, N. Doing a successful research project: using qualitative or quantitative methods. 2. ed. New York: Palgrave Macmillan, 2014.
- DOHERTY, N. F.; ASHURST, C.; PEPPARD, J. Factors affecting the successful realisation of benefits from systems development projects: findings from three case studies. *Journal of Information Technology*, v. 27, n. 1, p. 1–16, 2012.
- FERNANDEZ, K., GARRIDO, A., RAMINEZ, Y., & PERDOMO, I. PMBOK y PRINCE 2, similitudes y diferencias. *Revista Científica*, 23, p. 111-123, 2015.
- GHOSH, S. "Enhance PMBOK by Comparing it with P2M, ICB, Prince2, APM and *Scrum* Project Management Standards", *PM World Today*, 2012.
- GRAHAM, N. PRINCE2 for Dummies. (2a. ed.) EUA: John Wiley & Sons, 2009.

HERNÁNDEZ-LÓPEZ, A.; COLOMO-PALACIOS, R.; GARCÍA-CRESPO, Á. “Productivity in *software* engineering: a study of its meanings for practitioners: Understanding the concept under their standpoint”. In: II Workshop Europeu sobre informática e TIC Profissionalismo, p. 1-6, 2012.

JENSEN, J. R. Sensoriamento remoto do ambiente: uma perspectiva em recursos terrestres (tradução da segunda edição). São José dos Campos: Parêntese, p. 598, 2015.

KARDEC, M. S. Estudo de Compatibilidade entre PMBOK e *Scrum*. Revista Tecnologias e Projeção, p. 1-7, 2012.

KERZNER, H. Gestão de Projetos: As Melhores Práticas. Porto Alegre: Bookman, 2006.

KERZNER, H. Project management: a systems approach to planning, scheduling, and controlling. 12. ed. New Jersey: John Wiley & Sons, 2017.

MACHADO, M., & MEDINA, S. G. *Scrum*–Método Ágil Uma mudança cultural na gestão de projetos de desenvolvimento de *software*. Revista Científica Intraciência, p. 58-71, 2009.

MARIANO, A. D. “Gerenciamento de Portfólio de Projetos: Fatores Críticos de Sucesso e Impactos sobre os resultados organizacionais”, Universidade de São Paulo, 51p, 2008.

MAXIMIANO, A.C.A. Administração de projetos: como transformar ideas em resultados (3a. ed.). São Paulo: Atlas, 2008.

MEIRELLES, F. de S. Indicadores em pesquisas sobre uso de TIC na gestão das empresas. In: BARBOSA, A. F. (Ed.). Pesquisa sobre o uso de tecnologias de informação e comunicação nas empresas brasileiras: TIC Empresas 2015. São Paulo: Comitê Gestor da Internet no Brasil, p. 67–74, 2016.

MELO, C. O. “Produtividade e adaptabilidade de times ágeis: conceitos e paradoxos”. Thoughtworks Antologia Brasil: Histórias de aprendizado e inovação, Casa do código, p. 305, 2014.

MELO, C. O. “Productivity of agile teams: an empirical evaluation of factors and monitoring process”, Tese de Doutorado, Instituto de Matemática e Estatística, USP, p. 195, 2015.

MEREDITH, J.R., & MANTEL, S.J., Jr. Project Management: a managerial approach (4nd ed.) New York: Wiley, 2000.

MILES, M. B. & HUBERMAN, M. A. Qualitative data analysis. (2nd.ed.) London: Sage, 1994.

MOORE, S. IT projects need less complexity, not more governance. Gartner, 2015. Disponível em: <<https://www.gartner.com/smarterwithgartner/it-projects-need-less-complexity-not-more-governance/>>. Acesso em 6 abr. 2018.

PRESSMAN, R. S. Engenharia de *Software*, Ed. McGraw, 2006.

RAMPAZZO, L. Metodologia Científica: Para alunos dos cursos de graduação e pós-graduação. São Paulo: Loyola, 2005.

SAMPAIO, S. C. “A Review of Productivity Factors and Strategies on *Software Development*”. In: The Fifth International Conference on *Software Engineering Advances of ICSEA*, p. 196–204, 2010.

SAUNDERS, M.; LEWIS, P.; THORNHILL, A. Research methods for business students. 6. ed. Harlow: Pearson, 2012.

SCHWABER, K. The enterprise and *Scrum*. Ed. Microsoft, 2007.

SCHWABER, K. Information technology project management (7a. ed. rev.) EUA: Cengage Learning, 2014.

SCHWABER, K. & SUTHERLAND, J. Guia *Scrum*, 2015.

SCRUM.ORG. The *Scrum Framework* Poster. *Scrum.org*, 2019. Disponível em: <<https://www.Scrum.org/resources/Scrum-framework-poster>>. Acesso em 20 jun. 2019.

SHANKS, G.; BEKMAMEDOVA, N. Case study research in information systems. In: WILLIAMSON, K.; JOHANSON, G. (Ed.). Research Methods: Information, Systems, and Contexts. 2. ed. Cambridge: Elsevier, p. 193–208, 2018.

SIMONS, H. Case study research in practice. Los Angeles: SAGE, 2009.

SOMMERVILLE, I. Engenharia de *Software* (A. Ribeiro, trad.). São Paulo: Addison Wesley, 2003.

SOMMERVILLE, I. Engenharia de *Software* (I. Bosnic & K. Gonçalves). São Paulo: Pearson Prentice Hall, 2011.

VARGAS, R. Gerenciamento de Projetos. Estabelecendo diferenciais competitivos. Ed. Brasport. 9ª edição. 2018.

VASCO, Carlos G.; VITHOFT, Marcelo H.; ESTANTE, Paulo R. C. Comparação entre metodologias RUP e XP. Curitiba, 2004. Disponível em: <http://www.ppgia.pucpr.br/~alcides/Teaching/mestrado/FundamentosEngenhariaSoftware/artigos/ResumosApresentacoes/RUPvsXP_draft.pdf>. Acesso em fev. 2019.

VERGARA, S. C. Projetos e relatórios de pesquisa em administração. 4. ed. São Paulo: Atlas, 2003.

VITIELLO, J. Fast Track into Management. Computerworld, 2001 Disponível em <http://www.computerworld.com/article/2582357/it-skills-training/fast-track-into-management.html>. Acesso em fev.2019.

VORA, J. A. "Productivity and Performance Measures: Who Uses Them". Production and Inventory Management Journal, Vol 33, p. 46–49, 1992.

YIN, R. K. Estudo de caso: planejamento e métodos. 2. ed. Porto Alegre: Bookman, 2001.

WILLIAMSON, K. Questionnaires, individual interviews and focus group interviews. In: WILLIAMSON, K.; JOHANSON, G. (Ed.). Research methods: information, systems and contexts. 1. ed. Prahran: Tilde University Press, p. 349–372, 2013.