

CENTRO UNIVERSITÁRIO DO PARÁ - CESUPA
ESCOLA DE NEGÓCIOS, TECNOLOGIA E INOVAÇÃO - ARGO
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

DANILO BEZERRA DE CASTRO
DAVID SANTOS CABRAL
OSCAR WALDIM SIMÕES BORGES

**GESTÃO DE PROJETOS DE SOFTWARE APLICADA AO SEGMENTO BANCÁRIO:
UMA ANÁLISE DE ESTUDO DE CASO DE IMPLEMENTAÇÃO ÁGIL E
TRADICIONAL**

BELÉM
2022

DANILO BEZERRA DE CASTRO
DAVID SANTOS CABRAL
OSCAR WALDIM SIMÕES BORGES

**GESTÃO DE PROJETOS DE SOFTWARE APLICADA AO SEGMENTO BANCÁRIO:
UMA ANÁLISE DE ESTUDO DE CASO DE IMPLEMENTAÇÃO ÁGIL E
TRADICIONAL**

Trabalho de conclusão de curso apresentado à Escola de Negócios, Tecnologia e Inovação do Centro Universitário do Estado do Pará como requisito para obtenção do título de Bacharel em Ciência da Computação na modalidade ARTIGO.

Orientador (a): MSc. Ricardo Melo

BELÉM
2022

DANILO BEZERRA DE CASTRO
DAVID SANTOS CABRAL
OSCAR WALDIM SIMÕES BORGES

**GESTÃO DE PROJETOS DE SOFTWARE APLICADA AO SEGMENTO BANCÁRIO:
UMA ANÁLISE DE ESTUDO DE CASO DE IMPLEMENTAÇÃO ÁGIL E
TRADICIONAL**

Trabalho de conclusão de curso apresentado à Escola de Negócios, Tecnologia e Inovação do Centro Universitário do Estado do Pará como requisito para obtenção do título de Bacharel em Ciência da Computação na modalidade ARTIGO.

Data da aprovação: 12 / 12 / 2022


Nota final aluno(a) I: 9,5

Nota final aluno(a) II: 9,5

Nota final aluno(a) III: 9,5


Banca examinadora

Documento assinado digitalmente

 RICARDO RODRIGO MARINHO MELO
Data: 21/12/2022 18:58:25-0300
Verifique em <https://verificador.iti.br>


Prof(a). Ricardo Rodrigo Marinho Melo
Orientador(a) e Presidente da banca

Documento assinado digitalmente

 PEDRO HENRIQUE SALES GIROTTO
Data: 22/12/2022 07:37:13-0300
Verifique em <https://verificador.iti.br>

Prof(a). Pedro Henrique Sales Giroto
Examinador(a) interno(a)

Documento assinado digitalmente

 RICARDO MELO CASSEB DO CARMO
Data: 22/12/2022 11:22:57-0300
Verifique em <https://verificador.iti.br>

Prof(a). Ricardo Melo Casseb Do Carmo
Examinador(a) interno(a)

Dados Internacionais de Catalogação-na-publicação (CIP)
Biblioteca do CESUPA, Belém – PA

Castro, Danilo Bezerra de.

Gestão de projetos de software aplicada ao segmento bancário: uma análise de estudo de caso de implementação ágil e tradicional / Danilo Bezerra de Castro, David Santos Cabral, Oscar Waldim Simões Borges; orientador Ricardo Rodrigo Marinho Melo. – 2022.

Trabalho de Conclusão de Curso (Graduação) – Centro Universitário do Estado do Pará, Ciência da Computação, Belém, 2022.

- Software – Projetos – Avaliação. 2. Projeto de sistemas – Gerência. I. Cabral, David Santos. II. Borges, Oscar Waldim Simões. III. Melo, Ricardo Rodrigo Marinho, orient. IV. Título.

CDD

23ª ed. 005.1

RESUMO

A evolução tecnológica trouxe, com o passar dos anos, cada vez mais facilidade à vida de todos, independentemente do cenário. No caso do ambiente bancário não foi diferente, o impacto gerado pela transição para o uso da tecnologia mudou muitos aspectos dos serviços oferecidos pelas instituições financeiras. Entretanto, um aspecto que não é perceptível em nosso dia a dia mas se faz de tamanha importância para a transformação no sentido de se adaptar às novas tecnologias, é o desenvolvimento do produto, o processo de criação passa por diversas etapas que muitos dos usuários podem nem imaginar que existam. O processo de desenvolvimento tem como base uma metodologia específica, dentre as mais conhecidas no mercado ágil e tradicional, sendo ágil com ideias mais atuais e a tradicional, como o nome indica, mais antigas entretanto consolidadas no mercado. No caso das instituições financeiras já consolidadas no mercado, o desenvolvimento baseado no método tradicional era muito utilizado, até a chegada da transição tecnológica, onde surgiram bancos digitais com serviços remotos, deixando as comuns agências, subutilizadas, para se manter em competitividade os ambientes bancários tiveram que se adaptar, principalmente no sentido de adotar as metodologias mais atuais empregadas no mercado, as ágeis. Assim, este trabalho de curso visa realizar um estudo de caso analítico comparativo entre dois projetos, sendo eles um ágil e um tradicional, realizados dentro do ambiente bancário, a partir de métricas, avaliando as aplicações dos métodos de gestão empregados, para assim gerar um relatório comparativo entre as duas metodologias.

Palavras-chave: Instituição financeira; Metodologias de gestão; Estudo de caso; Métricas;

ABSTRACT

Technological evolution has, over the years, made life easier for everyone, regardless of the scenario. In the case of the banking environment it was no different, the impact generated by the transition to the use of technology changed many aspects of the services offered by financial institutions. However, an aspect that is not noticeable in our daily lives but is of such importance for the transformation in order to adapt to new technologies, is the development of the product, the creation process goes through several stages that many users can not even imagine they exist. The development process is based on a specific methodology, among the best known in the agile and traditional market, being agile with more current ideas and the traditional, as the name implies, older ideas however consolidated in the market. In the case of financial institutions already consolidated in the market, development based on the traditional method was widely used, until the arrival of the technological transition, where digital banks with remote services emerged, leaving the common branches, underutilized, to keep banking environments competitive. have to adapt, mainly in the sense of adopting the most current methodologies used in the market, the agile ones. Thus, this course work aims to carry out a comparative analytical case study between two projects, one agile and one traditional, carried out within the banking environment, based on metrics, evaluating the applications of the management methods employed, in order to generate a comparative report between the two methodologies.

Keywords: Financial institution; Management methodologies; Case study; Metrics;

SUMÁRIO

1 CONTEXTUALIZAÇÃO	8
1.1 Revisão Bibliográfica	8
1.1.1 Gestão Tradicional de Projetos	8
1.1.2 Gestão Ágil de Projetos	11
1.1.3 Ponto de Função	14
1.1.4 Indicadores-chave de Desempenho na Avaliação dos Métodos de Gestão	16
1.2 Problema da Pesquisa	17
1.3 Justificativa	17
1.4 Objetivos	18
1.4.1 Objetivo Geral	18
1.4.2 Objetivos Específicos	18
1.5 Estrutura do trabalho	18
2 ESTUDO DE CASO	19
2.1 Introdução	19
2.2 Metodologia da Pesquisa	20
2.2.1 Análise de Dados	20
2.3 Resultados	22
2.3.1 Estudo de Caso com Aplicação de Gestão Ágil de Projetos (Scrum)	22
2.3.2 Estudo de Caso com Aplicação de Metodologia de Gestão Tradicional de Projetos (PMBOK)	27
2.3.3 Relatório Comparativo	30
2.4 Discussão	33
2.5 Considerações Finais	35
3 REFERÊNCIAS	37

1 CONTEXTUALIZAÇÃO

1.1 Revisão Bibliográfica

1.1.1 Gestão Tradicional de Projetos

Na década de sessenta, com a criação da engenharia de software, se iniciou um fluxo constante de atualização dentro da área do desenvolvimento de software, trazendo e adaptando formas de gerir projetos (KUHRMANN, 2022), com o objetivo de ajudar gestores a colocar em prática seus conhecimentos, nasce então o Guia do Conhecimento em Gerenciamento de Projetos (Guia PMBOK®), livro voltado para o suporte na gestão tradicional ou cascata, como também ficou conhecido. Criado pelo *Project Manager Institute* (PMI), o PMBOK trabalha com ciclos preditivos, onde os requisitos dos stakeholders, prazo, escopo e o custo são definidos no começo do projeto e em seguida cria-se um plano sequencial para realização (NERUR, MAHAPATRA, MANGALARAJ 2005; PMI, 2016). Os produtos são totalmente especificados, previsíveis e podem ser construídos por meio de planejamento meticuloso e abrangente (NERUR, MAHAPATRA, MANGALARAJ 2005, p. 75).

O PMBOK consiste em uma padronização que identifica e conceitua processos, áreas de conhecimento, ferramentas e técnicas que descrevem o "ciclo de vida" de um projeto.

Um projeto consiste em um esforço temporário para criar um produto, serviço ou resultado específico, o livro, em sua sexta edição, se organiza em 10 áreas de conhecimento (Guia PMBOK 2017, p. 553):

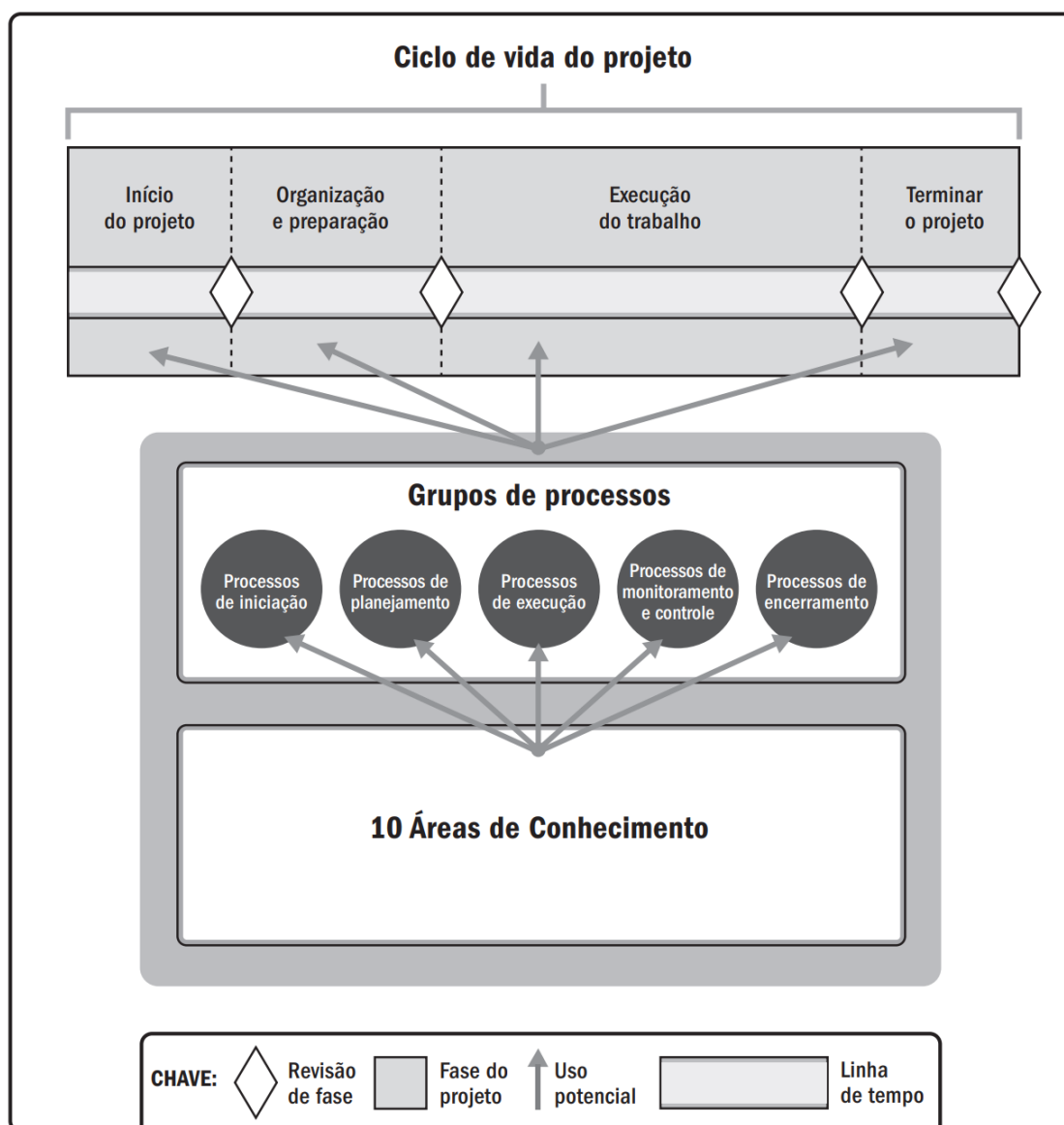
- **Gerenciamento Da Integração do projeto:** O gerenciamento da integração do projeto inclui os processos e atividades para identificar, definir, combinar, unificar e coordenar os vários processos e atividades de gerenciamento dentro dos Grupos de Processos de Gerenciamento do Projeto.
- **Gerenciamento Do Escopo do projeto:** O gerenciamento do escopo do projeto inclui os processos necessários para assegurar que o projeto inclui todo o trabalho necessário, e apenas o necessário, para terminar o projeto com sucesso.
- **Gerenciamento Do Cronograma Do Projeto:** O Gerenciamento do Cronograma do Projeto inclui os processos necessários para gerenciar o término dentro do prazo do projeto.

- **Gerenciamento Dos Custos Do Projeto:** O Gerenciamento dos Custos do Projeto inclui os processos envolvidos em planejamento, estimativas, orçamentos, financiamentos, gerenciamento e controle dos custos, de modo que o projeto possa ser terminado dentro do orçamento aprovado.
- **Gerenciamento Da Qualidade Do Projeto:** O Gerenciamento da Qualidade do Projeto inclui os processos para incorporação da política de qualidade da organização com relação ao planejamento, gerenciamento e controle dos requisitos de qualidade do projeto e do produto para atender as expectativas das partes interessadas
- **Gerenciamento dos Recursos do Projeto:** O Gerenciamento dos Recursos do Projeto inclui os processos para identificar, adquirir e gerenciar os recursos necessários para a conclusão bem-sucedida do projeto.
- **Gerenciamento das Comunicações do Projeto:** O Gerenciamento das Comunicações do projeto inclui os processos necessários para assegurar que as informações do projeto sejam planejadas, coletadas, criadas, distribuídas, armazenadas, recuperadas, gerenciadas, controladas, monitoradas e dispostas de maneira oportuna e apropriada.
- **Gerenciamento dos Riscos do Projeto:** O Gerenciamento dos Riscos do Projeto inclui processos de condução de planejamento, identificação e análise de gerenciamento de risco, planejamento de resposta, implementação de resposta e monitoramento de risco em um projeto.
- **Gerenciamento das Aquisições do Projeto:** O Gerenciamento das Aquisições do Projeto inclui os processos necessários para comprar ou adquirir produtos, serviços ou resultados externos à equipe do projeto.
- **Gerenciamento das Partes Interessadas do Projeto:** O Gerenciamento das Partes Interessadas do Projeto inclui os processos necessários para identificar todas as pessoas ou organizações impactadas pelo projeto, analisando as suas expectativas e o impacto das partes interessadas no projeto, e desenvolvendo estratégias de gerenciamento apropriadas para o engajamento eficaz das partes interessadas nas decisões e execução do projeto.

Além de reunir as práticas direcionadas às áreas de conhecimentos em gerenciamento de projetos, são definidos 5 grandes grupos de processos, Figura 1, com 49 subprocessos que podem ser aplicados ao escopo de desenvolvimento do projeto, se alinhando entre as etapas a serem realizadas para alcançar os objetivos definidos. Os grandes grupos indicam fases que devem ser cumpridas para se completar o ciclo completo de um projeto para alcançar os objetivos definidos. Os grandes grupos são:

- **Grupo de Processos de Iniciação:** O(s) processo(s) realizado(s) para definir um novo projeto ou uma nova fase de um projeto existente, através da obtenção de autorização para iniciar o projeto ou fase.
- **Grupo de Processos de Planejamento:** O(s) processo(s) realizado(s) para definir o escopo do projeto, refinar os objetivos e definir a linha de ação necessária para alcançar os objetivos para os quais o projeto foi criado.
- **Grupo De Processos De Execução:** O(s) processo(s) realizado(s) para concluir o trabalho definido no plano de gerenciamento do projeto para satisfazer os requisitos do projeto.
- **Grupo De Processos De Monitoramento E Controle:** Os processos exigidos para acompanhar, analisar e controlar o progresso e desempenho do projeto, identificar quaisquer áreas nas quais serão necessárias mudanças no plano, e iniciar as mudanças correspondentes.
- **Grupo De Processos De Encerramento:** O(s) processo(s) realizado(s) para concluir ou fechar formalmente um projeto, fase ou contrato.

Figura 1 - Aplicação das Áreas de Conhecimento e Grupo de Processos ao Ciclo de Vida de um Projeto



Fonte: Guia do Conhecimento Em Gerenciamento De Projetos, p. 18 (2017)

1.1.2 Gestão Ágil de Projetos

Em 2001 cerca de 17 profissionais que aplicavam metodologias inovadoras diversas na época, se reuniram para definir valores e princípios que todos aplicavam em comum, os quais entregavam o conceito esperado no desenvolvimento de software, o documento que surgiu naquele encontro foi o Manifesto Ágil, o qual apresentou uma nova forma de pensar o trabalho que era realizado, trazendo métodos muitos mais iterativos, buscando aproximar o cliente do

processo de desenvolvimento para se adaptar conforme fosse necessário dentro da entrega ideal (MANIFESTO ÁGIL, 2001). Nesse sentido, Grandes empresas passaram a introduzir a metodologia dentro do seu ambiente de desenvolvimento de software para usufruir dos benefícios que o método oferece, o foco na entrega da aplicação funcional e de boa qualidade gerando valor o mais rápido possível ao cliente reduz drasticamente a necessidade de um grande esforço da gestão para a criação de documentos e escopos extensos que não consigam se adaptar às necessidades do mercado a tempo, trazendo uma constante evolução ágil para seus produtos. A facilidade em absorver novas melhorias e entregar incrementos constantemente é o reflexo da necessidade do público e o que o mercado busca é o principal objetivo da metodologia ágil (MANIFESTO ÁGIL, 2001).

A principal metodologia seguida pelas grandes empresas quando falamos sobre gestão ágil é o *Scrum*, tal método foi criado por volta dos anos 90 (MORAIS, 2017, p. 109), e com o passar do tempo foi ganhando cada vez mais popularidade em meio aos gestores de projeto de *software*. Baseado em três pilares para seguir seu processo empírico, transparência, inspeção e adaptação, além de cinco principais valores, compromisso, foco, abertura, respeito e coragem, Figura 2, o *framework* é uma solução leve para problemas complexos que usa da adaptabilidade como base para a geração de valor (SCHWABER, SUTHERLAND, 2020).

Figura 2 - Pilares e Princípios do Scrum

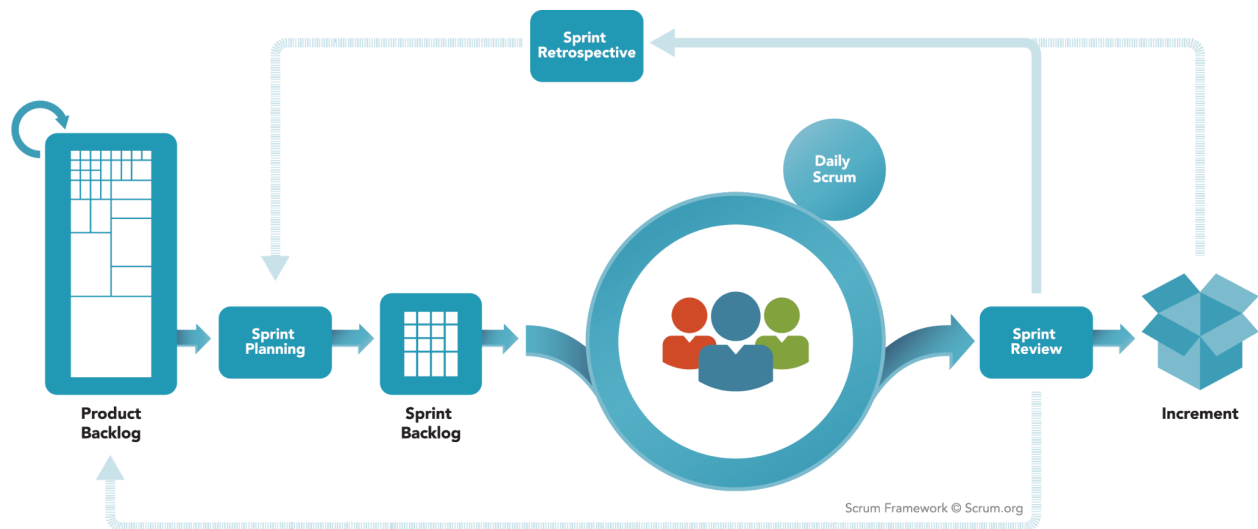


Fonte: <https://www.scrum.org/resources/what-is-scrum>

O *Scrum* apresenta um modelo de construção direcionado por pequenos ciclos de desenvolvimento, as *Sprints*. Cada *sprint* possui cerimônias que devem ser seguidas para garantir a evolução constante do produto, a primeira etapa é o Planejamento da *Sprint* (*Sprint Planning*), quando se planeja o que a equipe irá produzir no tempo delimitado, as tarefas são selecionadas a partir do *product backlog*, uma lista com todas as atividades que devem ser concluídas para atingir a definição de pronto do projeto. Como segunda cerimônia temos a *Daily Scrum*, reuniões diárias de acompanhamento com a equipe, ajustando o que está e o que será feito. Para finalizar a *sprint*, temos a Revisão de *Sprint* (*Sprint Review*) e Retrospectiva (*Retrospective*), na revisão é apresentado ao cliente as tarefas que foram concluídas até o momento, como forma de avaliar adaptações que o projeto precisa seguir, já a retrospectiva se trata de uma análise interna do grupo com o intuito de verificar quais foram os pontos negativos, positivos e a melhorar dentro da *sprint* (SCHWABER, SUTHERLAND, 2020), modelo mostrado na Figura 3.

Além da definição de um escopo de atuação, o *Scrum* alinha papéis com tarefas a tornar o trabalho fluido, como por exemplo o dono do projeto (*Project Owner*) que tem por responsabilidade básica maximizar o valor que será gerado pelo projeto, é quem fica responsável por tudo que se trata da organização do *product backlog*, desde a criação até ordenação. Para garantir que as tarefas priorizadas pelo dono do projeto sejam cumpridas da melhor forma possível, o *Scrum Master* assume o papel de garantir que todos sigam as definições do *Scrum Guide*, um guia para esclarecer tudo sobre o *Scrum*. Já os desenvolvedores assumem a construção do produto, gerando valor às atividades estipuladas para chegar na definição de pronto.

Figura 3 - Modelo do Processo de *Sprint* do *Scrum*



Fonte: <https://www.scrum.org/resources/what-is-scrum>.

1.1.3 Ponto de Função

O Ponto de Função é uma métrica usada para medir o tamanho funcional de um sistema considerando a complexidade das funções, a partir da visão do usuário (BRASIL. MINISTÉRIO DO PLANEJAMENTO, 2018), e assim realizar, a partir da Análise de Ponto de Função (APF), de forma correta a estimativa de remuneração de contratos e serviços de desenvolvimento e manutenção firmados como forma de padronizar as requisições, principalmente, entre instituições públicas e empresas prestadoras desse tipo de serviço (MORAIS, 2017, p. 53).

Como forma de definir os pontos de função de um projeto, o Manual de Práticas de Contagem (CPM), estabelece como regras a determinação do propósito, escopo e tipo de contagem, além da fronteira de aplicação (BRASIL. MINISTÉRIO DO PLANEJAMENTO, 2018). Assim, é determinada a forma de resolver o problema de negócio, explicitando o propósito, levando tal ideia como base para selecionar o tipo e escopo, onde o escopo delimita as funcionalidades que sejam levadas à contagem e se o projeto será do tipo desenvolvimento, melhoria ou aplicação instalada, assim com as três primeiras etapas definidas, chamamos a fronteira de aplicação tendo um conceito mais subjetivo, esta é definida de forma a considerar o processo de negócio, onde se estipula a interface conceitual, dada a visão do usuário (BRASIL. MINISTÉRIO DO PLANEJAMENTO, 2018).

Nesse Sentido, chegamos ao próximo passo onde o tamanho de cada Ponto de Função é baseado no tipo de função para pode ser definido o nível de complexidade de tais pontos, sendo as funções:

- **Arquivo Lógico Interno (ALI):** São os dados armazenados por meio de um processo simples realizado pela aplicação.
- **Arquivo de Interface Externa (AIE):** São dados armazenados na aplicação cotada por meio de um processo de ALI realizado por outra aplicação.
- **Entrada Externa (EE):** É o processamento de dados ou informações de controle que chegam na aplicação pela sua fronteira.
- **Consulta Externa (CE):** É o processo de exibição de dados ou informações de controle ao usuário.
- **Saída Externa (SE):** É o processo de exibição de dados, ao usuário, que passam por processamento lógico.

Partindo para o cálculo, são definidos dois tipos de requisitos, funcionais e não funcionais (BRASIL. MINISTÉRIO DO PLANEJAMENTO, 2018). Os requisitos funcionais indicam tarefas que o sistema irá realizar, representando as funcionalidades lógicas do produto em questão, onde para serem apurados utiliza-se de quatro definições ponto de função de inclusão, no caso de adicionar novas funcionalidades, de alteração, para modificação de funcionalidades pré existentes, de exclusão, voltados para a eliminação de funcionalidades pré existentes e de

conversão, visando a utilização de informações existentes para consumo em novas tabelas para novas funcionalidades. Na questão dos requisitos não funcionais, estão relacionados ao contexto mais voltado para o usuário na utilização da tecnologia, sendo aspectos qualitativos, os quais possuem como tipos de ponto de função, usabilidade, o qual fala sobre a facilidade na utilização do *software*, técnico, abordando a ideia dos tipos de tecnologias usadas para o desenvolvimento, segurança, a aplicação deve proteger o usuário e seus dados conforme as leis, acessibilidade, o produto deve ser adaptativo para pessoas com limitações, performance, exigindo que a aplicação seja otimizada e interoperabilidade, o sistema deve ser capaz de se trabalhar em conjunto com outras organizações de maneira eficiente.

1.1.4 Indicadores-chave de Desempenho na Avaliação dos Métodos de Gestão

Os Indicadores-chave de Desempenho, originalmente chamados de KPI, *Key Performance Indicators*, segundo a organização KPI, são ferramentas que utilizam da medição do nível de sucesso de uma empresa ou projeto a partir do desempenho gerado em direção às metas traçadas (KPI, s.d.), para guiar e dar foco nos pontos que necessitam de mais atenção a partir dos indicadores definidos.

Nesse sentido, são apresentados alguns tipos de indicadores-chave que abrangem um escopo grande dentro de um projeto (KPI, s.d.), entretanto tais indicadores variam dependendo do conceito que será aplicado, dentre eles temos:

- **Indicadores de Produtividade:** Medem os recursos utilizados no desenvolvimento com as entregas.
- **Indicadores de Qualidade:** Visam avaliar a satisfação dos clientes e os produtos gerados.
- **Indicadores de Efetividade:** Verificam a eficácia do produto, qualidade pelo proposto.
- **Indicadores da Capacidade:** Analisam as entregas e saídas propostas pelo tempo utilizado.
- **Indicadores Estratégicos:** Informam como estão sendo direcionadas as entregas relacionadas ao objetivo definido.

Visando a constante evolução do desenvolvimento de um projeto baseado nas metodologias ágeis ou tradicionais, se faz a necessidade da definição de indicadores que realizam a manutenção do progresso dentro do projeto, acompanhando as atividades realizadas para se manter atualizado em relação ao objetivo final e mais importante a entrega do projeto com a qualidade e esperada.

1.2 Problema da Pesquisa

Desde o surgimento dos modelos de metodologia de gestão de projetos, o modo que as empresas constroem soluções de software passa por constantes mudanças, sempre buscando evoluir e otimizar o trabalho realizado, fato que não foi diferente com a área monetária.

Nesse sentido, a utilização de métodos de gestão que já se fazem enraizados nas empresas pode ser um empecilho quando se busca concluir projetos de forma acelerada e eficiente. Logo, surgiu a necessidade de um questionamento de como avaliar e comparar a contribuição de metodologias de gestão, como ágil e em cascata, de processos e projetos de software no ambiente bancário do Estado do Pará, usando como indicadores de efetividade, as métricas: produtividade e lucratividade.

1.3 Justificativa

A criação de um produto e a entrega de resultados pode parecer algo complicado aos olhos de alguém que não acompanha metodologias de gestão e a evolução do mercado, entretanto, com a ideia concreta e o processo certo para o desenvolvimento, qualquer empresa pode alcançar os objetivos definidos.

Tal fato se justifica pelo cenário de baixa competitividade de mercado que os bancos tradicionais do Estado do Pará apresentam em relação às outras instituições financeiras. A priorização, por parte dos clientes, de empresas de fora do estado, do ramo monetário, se deve pela qualidade e rapidez na questão da inovação, ter produtos e atualizações com frequência ajuda a entregar novidades constantes para os usuários. Nesse sentido, ter uma produção de projetos constante e acelerada se faz necessária visando alcançar uma posição competitiva em relação aos concorrentes.

Com objetivo de avaliar a baixa efetividade dos projetos de software nas instituições financeiras do Estado do Pará pela má otimização dos modelos de gestão de projeto, o trabalho apresenta uma análise comparativa como forma de identificar pontos positivos e negativos em relação a aplicação das metodologias de gestão ágil e tradicional.

1.4 Objetivos

1.4.1 Objetivo Geral

Analisar dois estudos de caso sobre a aplicação dos modelos de gestão de projetos, ágil e tradicional, para o desenvolvimento de software em instituições financeiras, definindo como indicadores de efetividade, as métricas produtividade e lucratividade gerada no projeto a serem avaliadas.

1.4.2 Objetivos Específicos

- Analisar a aplicação das ferramentas das metodologias de gestão de projetos no desenvolvimento de software de uma instituição financeira;
- Realizar uma análise comparativa tomando como base as métricas, produtividade e lucratividade, para avaliar o retorno gerado pela aplicação dos tipos de gestão de projetos de software no desenvolvimento de produtos de uma instituição financeira.
- Avaliar e comentar as metodologias utilizadas na instituição observada;

1.5 Estrutura do trabalho

O trabalho consiste em um primeiro capítulo introdutório, apresentando e explicando as metodologias de gestão ágil e tradicional, baseadas em um referencial teórico apresentando a problemática e a justificativa. Um segundo capítulo contendo a explicação da pesquisa realizada sobre os estudos de casos, demonstrando sua metodologia, o ambiente e os resultados neles contidos, além de discussões gerais, sínteses retiradas dos casos listados, com suas devidas evidências e limitações do estudo.

2 ESTUDO DE CASO

2.1 Introdução

A chegada da era da informação proporcionou diversas mudanças na nossa realidade. Atualmente, o uso da tecnologia passou a fazer cada vez mais parte da nossa rotina e realidade, a constante variação faz com que todos tenham que se tornar flexíveis a alterações, e dentro dessa realidade uma das áreas que mais evoluiu e se transformou neste período foi o mercado financeiro. “A tecnologia transformou o mercado financeiro, caracterizado por ser robusto, em um mercado cada vez mais aberto, competitivo e oportuno para novas ideias” (SILVA, 2019; DE ARAÚJO JÚNIOR, 2019).

Com a transformação acelerada do setor financeiro, os bancos precisaram investir de maneira pesada na área de tecnologia e inovação, para acompanhar a evolução do setor e se manter competitivo dentro do mercado financeiro. Dessa forma, entendendo o pedido dos clientes e com tecnologia suficiente para abastecer o mercado, surgiram as *fintechs*, startups que desenvolvem produtos e serviços digitais e aplicam grandes investimentos no mercado financeiro, as quais se tornaram a principal propulsora do desenvolvimento tecnológico monetário, deixando em estado de alerta grandes instituições do ramo. Os processos ágeis, tecnológicos e eficazes, fizeram com que instituições bancárias virtuais preenchessem os espaços vazios deixados pelos grandes bancos tradicionais.

Assim, nasce a necessidade de introduzir novas metodologias para gestão de software alinhadas com o mercado e com a volatilidade das necessidades do cliente, visando cumprir os objetivos estipulados para a solução dentro do escopo definido, dentro das instituições financeiras que antes eram referência no mercado e hoje se encontram em risco.

Logo, com a finalidade de realizar um estudo de caso sobre a aplicação das metodologias de gestão de projetos dentro das instituições financeiras, serão levantados dados e estatísticas sobre o desenvolvimento de soluções tecnológicas, além de realizar uma comparação levando como base métricas como forma de evidenciar a evolução dos projetos do ponto de vista da rentabilidade e entrega de valor.

2.2 Metodologia da Pesquisa

Este trabalho consiste em um estudo de caso sobre as metodologias de gestão de projetos aplicadas no ambiente de desenvolvimento de uma empresa terceirizada que presta serviços para uma instituição financeira, analisando os tipos de gestão aplicados ao desenvolvimento de projetos de *software*, modelo ágil e tradicional.

Para realizar o estudo de caso foram escolhidos dois projetos modelo, que contam com metodologias de desenvolvimento diferentes utilizadas por duas equipes atuantes na empresa citada anteriormente. O critério de seleção dos projetos para o estudo de caso foi definido através da análise sobre a qualidade da aplicação das metodologias propostas nos projetos perante as o indicador de efetividade gerado por ambos, tomando como base as métricas: produtividade, avaliando a qualidade e quantidade de entregas pelo tempo utilizado, e a lucratividade, tomando como base o retorno financeiro gerado para a instituição financeira com o projeto.

Com a finalidade de realizar uma análise descritiva, foram coletados dados do *software* de gestão de projetos Azure Devops, utilizado pela instituição na aplicação dos trabalhos foco do estudo de caso. As principais fontes de dados tomadas como base para a pesquisa foram as documentações que abordam gráficos de medição de desenvolvimento do projeto, gráfico de *Burndown*, fluxos de trabalho. Para definir o contexto a ser estudado, a equipe analisou os dados, dando destaque aos indicadores de efetividade, tomando como pilar na comparação entre os projetos analisados as métricas lucratividade, dando ênfase ao retorno financeiro de cada projeto, e a produtividade, avaliando o resultado gerado como forma de produto pelo tempo e recursos consumidos.

2.2.1 Análise de Dados

Os dados analisados foram organizados e adicionados à ferramenta de gestão Azure DevOps durante a criação do projeto e sua ocorrência. A primeira informação necessária se trata dos recursos alocados para atuar no desenvolvimento do projeto, levando em consideração que no projeto ágil foram utilizados um total de 8 recursos entre eles: um especialista em tecnologia, um desenvolvedor sênior, três desenvolvedores juniors, um analista de requisitos, um analista de teste e um Scrum Master. Enquanto isso, no projeto que utiliza metodologia tradicional, foram

utilizados um total de 5 recursos, entre eles um desenvolvedor sênior, um analista de teste, um analista de requisitos, um líder técnico e um Gestor de projetos.

O tempo estimado para cada uma das *sprints* do projeto ágil foi de 15 dias, porém quando os itens não são finalizados dentro de uma *sprint* eram realocados na próxima *sprint*, o que afetou diretamente a duração do desenvolvimento, aumento o tempo para a entrega em todas, como mostra a tabela 1.

Tabela 1 - Tabela de Dados Projeto Ágil

Sprints	Pontos de Função	Histórias de usuário	Tempo Estimado	Duração
Sprint 1	44	11	15 dias	61 dias
Sprint 2	107,6	21	15 dias	20 dias
Sprint 3	100	30	15 dias	29 dias
Total	251,6	62	45 dias	110 dias

Fonte: Autorial(2022).

Enquanto o cronograma dos pacotes de entrega do método tradicional foi definido pelo equipe, desenvolvedor e gerente de projeto, levando em consideração o esforço para as atividades e o período de faturamento dentro de um mês, como mostra a tabela 2.

Tabela 2 - Tabela de Dados Projeto Tradicional

Entregas	Pontos de Função	Casos de Uso	Duração
pacote 1	73,3	5	21 dias
pacote 2	61,3	5	23 dias
pacote 3	7	5	22 dias
Pacote 4	76,75	5	21 dias
Pacote 5	17,85	3	20 dias
Total	236	23	107 dias

Fonte: Autorial(2022).

Durante a *Planning* do projeto que utiliza metodologias ágeis foram criadas as tarefas que precisavam ser realizadas em cada uma das histórias de usuário que foram priorizadas para o desenvolvimento, durante o *time box* de 3 *sprints*, que compõe o tempo observado de cinco meses, foram criadas um total de 312 tarefas para um total de 62 melhorias descritas em histórias

de usuário que renderam um total de 251,6 pontos de função, utilizando a contagem de pontos de função do SISP.

Já no projeto que utiliza de metodologia tradicional na fase de entendimento da demanda, também foram criadas as tarefas no Azure DevOps para acompanhamento do percentual de conclusão durante o cronograma definido. Durante o período de 5 meses observado, foram criados um total de 152 tarefas para 23 melhorias descritas em casos de uso e que renderam um total de 236 pontos de função, os quais também utilizam a contagem de pontos de função do SISP.

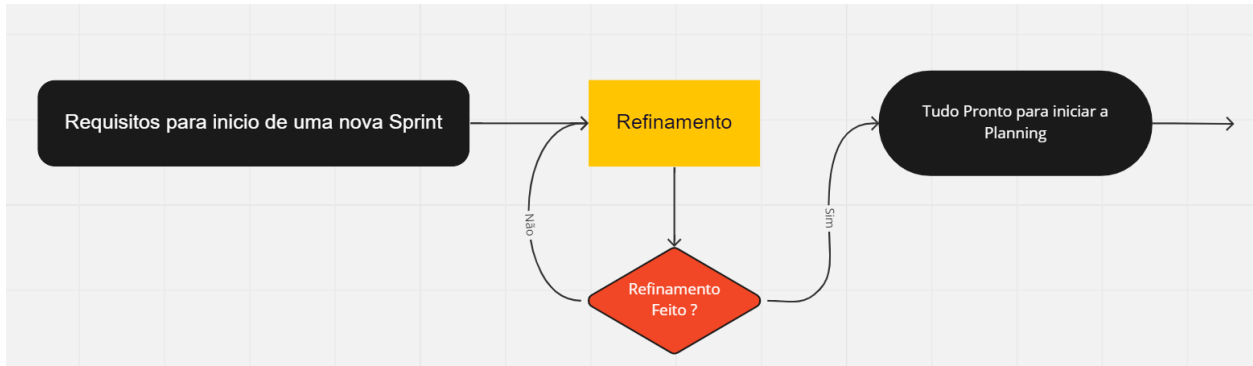
2.3 Resultados

2.3.1 Estudo de Caso com Aplicação de Gestão Ágil de Projetos (Scrum)

Com a tentativa de acompanhar as novas metodologias utilizadas para o desenvolvimento de software do mercado e ajudar a melhorar a transparência das atividades durante as solicitações de melhoria da instituição financeira, surgiu a necessidade de iniciar contratos que utilizam de metodologias ágeis de desenvolvimento de software. Assim iniciou-se o Projeto Ágil, com uma equipe dedicada para tal atividade, formada por um analista de requisitos, quatro desenvolvedores *fullstack*, um analista de teste, um *scrum master* e o *project owner*.

O fluxo de trabalho utilizando a metodologia *Scrum* iniciou-se com o refinamento e entendimento das histórias de usuários pelo analista de requisitos, além da criação do protótipo do produto pelo *designer* de experiência de usuário, o qual foi homologado pelos *stakeholders*. Seguindo o processo, se deu a criação do levantamento de riscos, a finalização dos impedimentos de arquitetura, e da priorização das histórias de usuário pelo *project owner* à serem selecionadas para o *Sprint Backlog*, como mostrado na Figura 5.

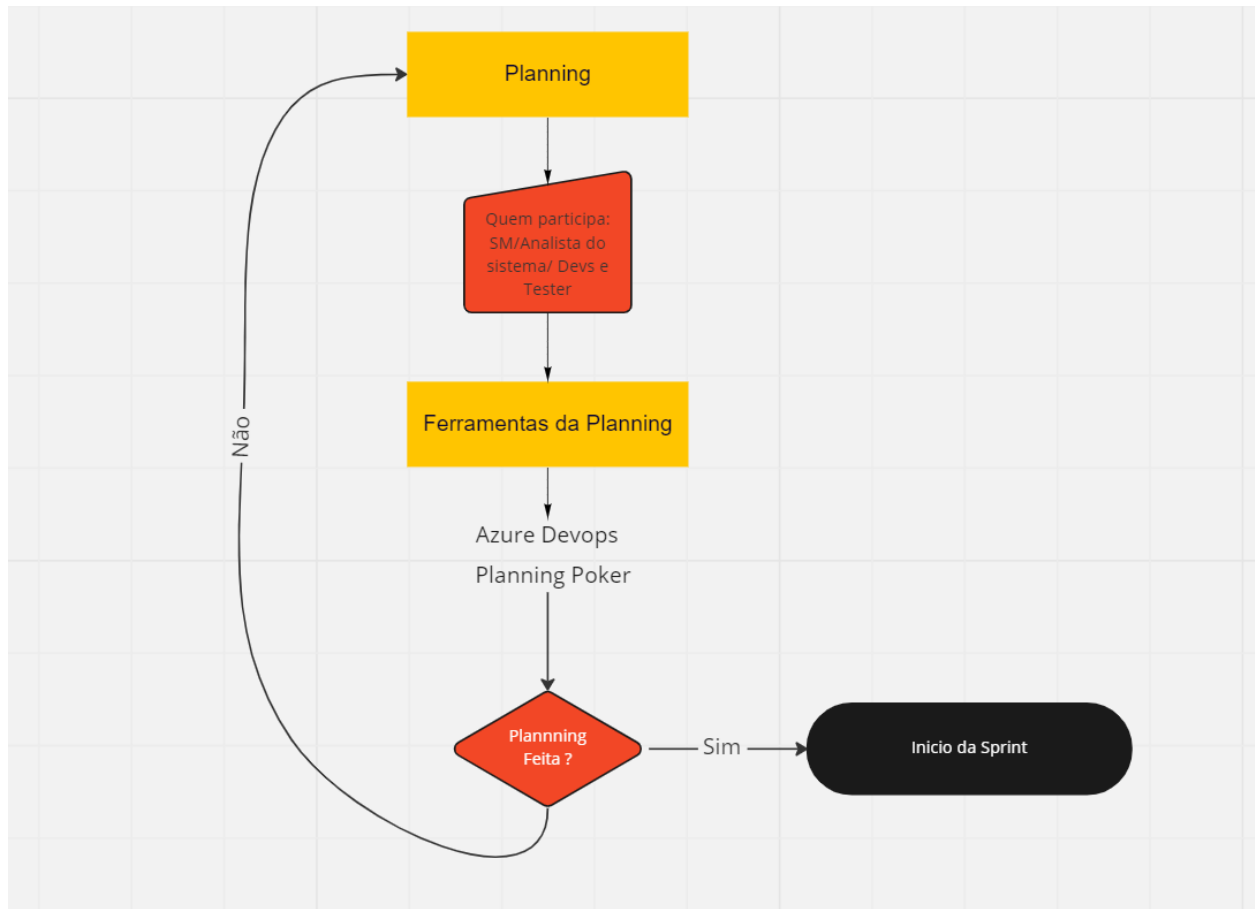
Figura 5 - Fluxo de Abertura de Projeto do Modelo Analisado



Fonte: Autoral(2022).

Seguindo para a próxima etapa, a *planning*, os analistas de requisitos e *project owners* apresentaram ao time de desenvolvimento a documentação e o *sprint backlog* priorizado, sem impedimentos para que os itens definidos na priorização fossem desenvolvidos e testados. Durante esse evento, o time de desenvolvimento esclareceu o que seria realizado, discutiu de maneira técnica formas para aplicar o que foi definido. As atividades foram mapeadas através da ferramenta *Azure DEVOPS* para o acompanhamento das atividades durante a *sprint*, Figura 6, e, ao final, quando a maioria das dúvidas haviam sido sanadas, foi feito o *planning poker* da atividade para que fosse medido o esforço e atender a demanda em questão.

Figura 6 - Fluxo de Planejamento das Atividades da *Sprint*



Fonte: Autoral(2022).

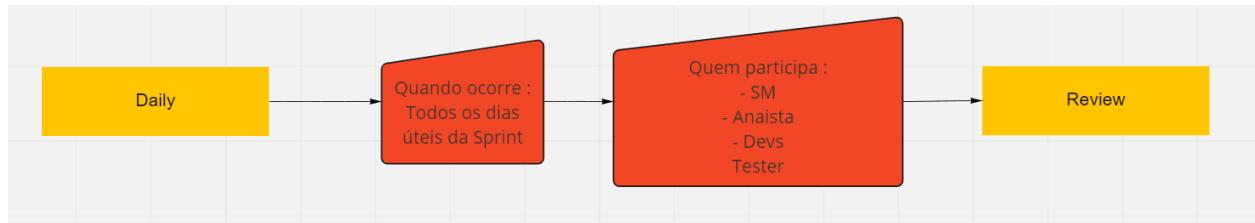
Com as duas etapas iniciais finalizadas, iniciou-se a *sprint*. Durante o desenvolvimento, ocorreram as *daily*s, reunião diária de acompanhamento das atividades, na qual o time de desenvolvimento alinhou o que estava sendo desenvolvido das demandas, respondendo as principais questões, Figura 7:

1. O que foi feito ontem?
2. O que estou fazendo agora?
3. Tenho algum impedimento ou bloqueio?

As *sprints* inicialmente foram definidas com duração de quinze dias úteis seguindo as cerimônias definidas no *scrum guide* e utilizando o *kanban* para a gestão visual das atividades.

As atividades foram acompanhadas diariamente por meio da utilização do gráfico de *burndown* das tarefas.

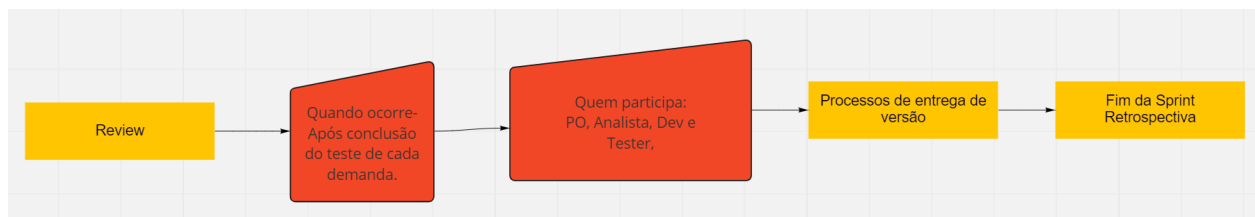
Figura 7 - Fluxo de Realização das Reuniões Diárias



Fonte: Autoral(2022).

Ao final do *time box*, tempo definido para conclusão da *sprint*, foi realizada a revisão da *sprint*, Figura 8, na qual o analista de teste e os desenvolvedores apresentaram ao *project owner* o que foi desenvolvido, além de testar para que fosse homologado, validando assim a entrega do item da *sprint*. O propósito foi inspecionar o resultado da demanda, para garantir a entrega de valor esperada ao cliente, após isso todas as tarefas concluídas foram disponibilizadas no ambiente de homologação do cliente.

Figura 8 - Fluxo de Realização das Reuniões de Revisão do Projeto

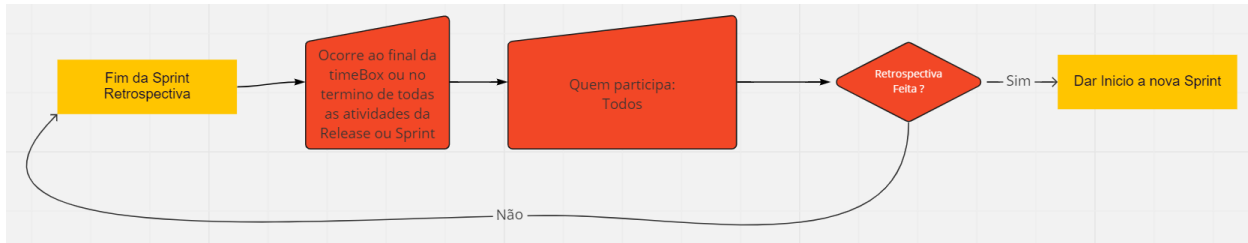


Fonte: Autoral(2022).

Para conclusão da *sprint*, foi realizada a retrospectiva, Figura 9, com finalidade de avaliar as ações desenvolvidas, pela equipe do projeto, até esta cerimônia. O time de desenvolvimento se reuniu para responder às seguintes perguntas:

- O que a equipe deve começar a fazer?
- O que a equipe deve parar de fazer?
- O que a equipe deve continuar fazendo?

Figura 9 - Fluxo de Realização da Reunião de Retrospectiva



Fonte: Autoral(2022).

Ao final da primeira *sprint*, onde eram esperados quinze dias, transformou-se em sessenta e um dias de desenvolvimento, observou-se que por alguns fluxos não inicialmente compreendidos, levantados na fase de requisitos, surgiram impedimentos durante o desenvolvimento que geraram atraso nas entregas, além da necessidade de mudança de escopo durante a *sprint*, o que prejudicou a entrega de valor e a revisão da *sprint*, esses itens foram citados na retrospectiva para serem melhorados.

Os resultados alcançados na primeira *sprint* não foram suficientes, das histórias de usuário planejadas apenas 30% foram entregues, o valor desejado pelo PO não foi alcançado e os itens pendentes tiveram que ser realocados na próxima *sprint*, o número de PF (Pontos de Função) não foi suficiente para cobrir o custo da equipe, logo não houve lucro para a empresa que prestou serviço e sim uma margem negativa de 40%. Um ponto que colaborou com esse resultado negativo foi a curva de aprendizado dos desenvolvedores, pelo fato da equipe ter separado um período para estudos aplicados à linguagem, além da dificuldade e demora na configuração do ambiente de desenvolvimento, esse tempo foi desconsiderado na *planning*, o que prejudicou ainda mais o cronograma.

Ao final da segunda *sprint* de quinze dias, a quantidade de histórias de usuários entregues dobrou já que algumas pendências haviam sido sanadas, porém ainda assim, não foi possível alcançar o número mínimo de pontos de função para cobrir o custo da equipe e o percentual negativo de faturamento continuou acumulando.

Ao final dos 5 meses de desenvolvimento com um total de 3 *sprints* parcialmente finalizadas, foi observado o surgimento de novas atividades à ser desenvolvidas, além das atividades que foram repassadas nas *sprints* anteriores por não terem sido entregues a tempo, o

que culminou com o atraso da entrega de valor ao cliente. Outro grande impacto foi o faturamento da empresa prestadora de serviço para a instituição financeira, a qual teve muitos prejuízos pelo fato do cronograma não ter sido cumprido.

2.3.2 Estudo de Caso com Aplicação de Metodologia de Gestão Tradicional de Projetos (PMBOK)

O Projeto Tradicional ocorreu em um cenário de manutenções evolutivas de um sistema em ambiente de produção, muitas dessas solicitações eram demandas legais onde o banco central determinou um prazo fixo que não deveria ser excedido, a quebra desse prazo poderia gerar multas à instituição financeira.

A equipe foi constituída de 1 desenvolvedor, um analista de requisitos, um gestor de projetos, um analista de teste e um líder técnico. Os requisitos para que a solicitação fosse iniciada foram os casos de uso das funcionalidades atingidas, mapeados e modificados com as alterações da análise arquitetural, documento que mapeia todas as alterações técnicas a serem seguidas e prazo definido, Figura 10.

Figura 10 - Fluxo de Abertura de Escopo do Projeto



Fonte: Autorial(2022).

Os analistas de requisitos mapearam as modificações solicitadas através dos casos de uso, documento que descreve as funcionalidades dos sistemas, e passaram para o líder técnico avaliar os riscos envolvidos para a criação da análise arquitetural. Após a conclusão da análise arquitetural, a equipe de desenvolvimento seguiu à risca as modificações descritas na documentação, a equipe se reuniu diariamente para mostrar a evolução e apresentar os impedimentos conforme eles surgiam.

Durante a solicitação, foi utilizado o *azure devops* para mapear as atividades e acompanhar a evolução das tarefas através das entregas, Figura 11, a porcentagem de conclusão de cada atividade foi acompanhado diariamente pelo gestor de projetos para que, se necessário, fossem criados planos de ação com o objetivo de se manter dentro do prazo, a comunicação entre

a equipe foi a principal ferramenta para manter o alinhamento. Ao final da solicitação notou-se que para que o prazo fosse cumprido algumas atitudes foram necessárias, como o acordo com o desenvolvedor para atuar em regime de hora extra para dar vazão a atividades não mapeadas.

Figura 11 - Fluxo de Desenvolvimento e Entregas do Projeto



Fonte: Autoral(2022).

Os desenvolvedores não atuaram exclusivamente em uma solicitação, o que flexibilizou a movimentação e alocação de profissionais que possuíam mais experiência em determinada atividade, o que ajudou na celeridade do desenvolvimento além da possibilidade de atuação em mais de uma demanda ao mesmo tempo, o que impactou positivamente o lucro para a empresa prestadora de serviço, já que o contrato e o pagamento utilizaram da métrica de pontos de função e não alocação em horas.

Outro ponto positivo para a metodologia foi que com o maior esforço na fase de documentação sendo mais rebuscada e analisada como um todo, e nem sempre focando em apenas uma alteração ou melhoria, quando disponibilizado para desenvolvimento, tornou-se mais fácil de ser seguida pelo desenvolvedor e com o risco reduzido de alteração de escopo durante a fase de desenvolvimento.

Ao final dos 5 meses de desenvolvimento todas as atividades mapeadas das melhorias propostas foram finalizadas sem nenhuma pendência, todas foram homologadas pelo cliente e as correções necessárias foram analisadas e corrigidas.

2.3.3 Relatório Comparativo

A escolha de uma abordagem focada em um relatório comparativo se faz necessária para evidenciar pontos principais relacionando assuntos distintos, os quais apresentam dados e estatísticas a serem comparadas de forma descritiva, levando uma informação a ser analisada pelo leitor com a finalidade de gerar uma reflexão sobre o assunto.

Sendo assim, os resultados obtidos dentro do estudo dos projetos apontaram vários pontos de coerência e também de discordância em relação a aplicação das duas metodologias abordadas, levando em consideração o período de desenvolvimento e as entregas dos projetos. Para fins de uma comparação mais detalhada, definimos o indicador de efetividade a ser utilizado na análise dos métodos de gestão a partir das métricas, produtividade e lucratividade, sendo referente à entrega do produto proposto analisando a qualidade pelo tempo e recursos consumidos e o retorno financeiro gerado pelos projetos levando em conta o tamanho da equipe e lucro final da instituição, respectivamente, como mostra a Tabela 3.

Tabela 3: Comparação entre os Projetos

	Tempo Estimado (meses)	Tamanho da Equipe	Quantidade de Itens de Requisitos	Quantidade de Tarefas	Taxa de Lucro
Projeto Ágil	5	8	62	312	-58,64%
Projeto Tradicional	5	4	23	152	20,81%

Fonte: Autoral(2022)

A análise sobre produtividade busca evidenciar questões logísticas do projeto, assuntos relacionados ao desenvolvimento, tempo de entrega, a qualidade final do produto e recursos consumidos. O tempo estimado para a entrega do projeto ficou avaliada em 5 meses para ambos, entretanto o ponto que mais chama a atenção entre as duas metodologias é o tamanho da equipe,

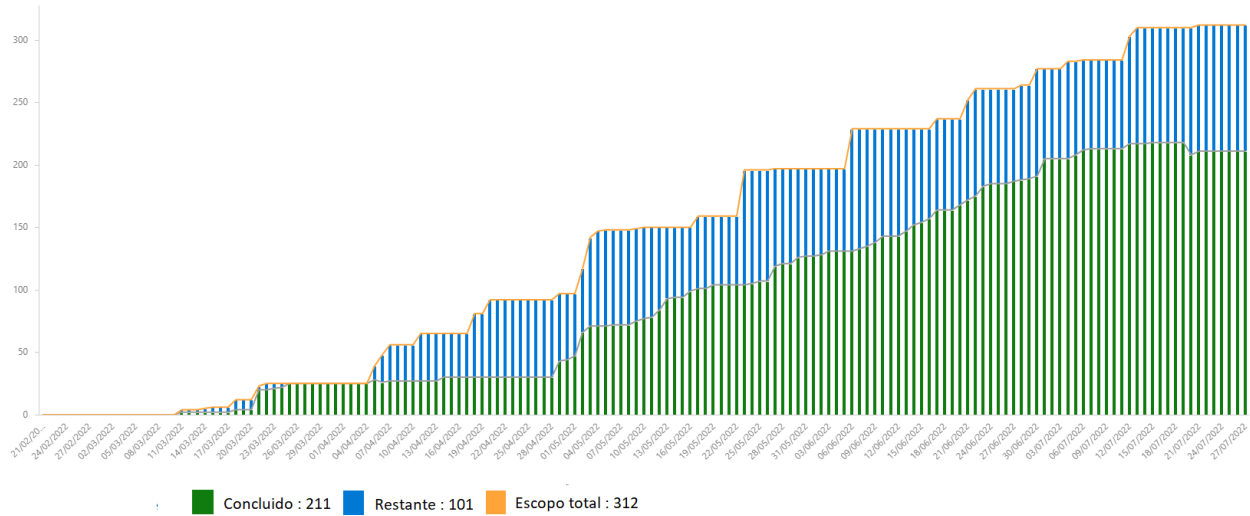
essa diferença deve-se ao fato de que em projetos ágeis a equipe deve ser multidisciplinar e focada no processo que foi designada, já na questão do tradicional, mesmo que participando deste projeto, a equipe como um todo fazia-se presente em outras atividades.

Nesse sentido, outro fator de extrema importância a ser abordado nesta comparação é a quantidade de requisitos levantados em cada projeto, a metodologia ágil apresenta 62 histórias de usuário criadas pelos analistas de requisitos, com um acúmulo de 312 tarefas para sua conclusão, enquanto a tradicional possui 23 requisitos sendo desmembrados em 152 atividades, isso é um fator que varia de acordo com o trabalho requisitado pela empresa, mas como parâmetro de comparação, percebe-se que os requisitos se fazem diretamente proporcionais ao tamanho da equipe, o que visa garantir a entrega do produto, o que não foi possível no desenvolvimento do produto que utilizou a gestão ágil, a Figura 12 mostra que neste período grande parte do esperado foi entregue, entretanto devido às constantes alterações que são realizadas influenciadas pela metodologia, algumas tarefas acabaram sendo bloqueadas, ou não foram concluídas no período estipulado. Fato este que não acontece na utilização do método tradicional, o escopo fechado garante que não haverá mudanças nas tarefas, levando a um desenvolvimento mais concreto e focado, o que trouxe melhores resultados em comparação ao outro projeto analisado, Figura 13.

A métrica de lucratividade se volta para o lado financeiro do projeto, visando analisar o retorno gerado pelas entregas do produto, além de avaliar os custos e o investimento geral realizado. Assim, um dado chama muita atenção, o lucro do projeto ágil está negativo, mesmo entregando muitas tarefas das histórias de usuário no mesmo período que o tradicional, gerou um *déficit* alto para o projeto, o fato de possuir uma equipe maior e especializada torna o custo para manter o desenvolvimento rodando muito alto, outro fator decisivo é a volatilidade das entregas, ao ficar pendente em uma *sprint* uma tarefa é levada para a próxima o que acaba sobrecarregando o período de desenvolvimento subsequente, levando a muitos atrasos e diminuindo a receita esperada por *sprint*, já na visão do tradicional o objetivo final foi atingido dentro do prazo estipulado com sucesso, mesmo enfrentando atrasos, como observado na Figura 13, o que ajudou a garantir a entrega do projeto foram os documentos de iniciação, os quais por terem sido feitos de maneira detalhada e minuciosa para que o escopo não mudasse durante o desenvolvimento.

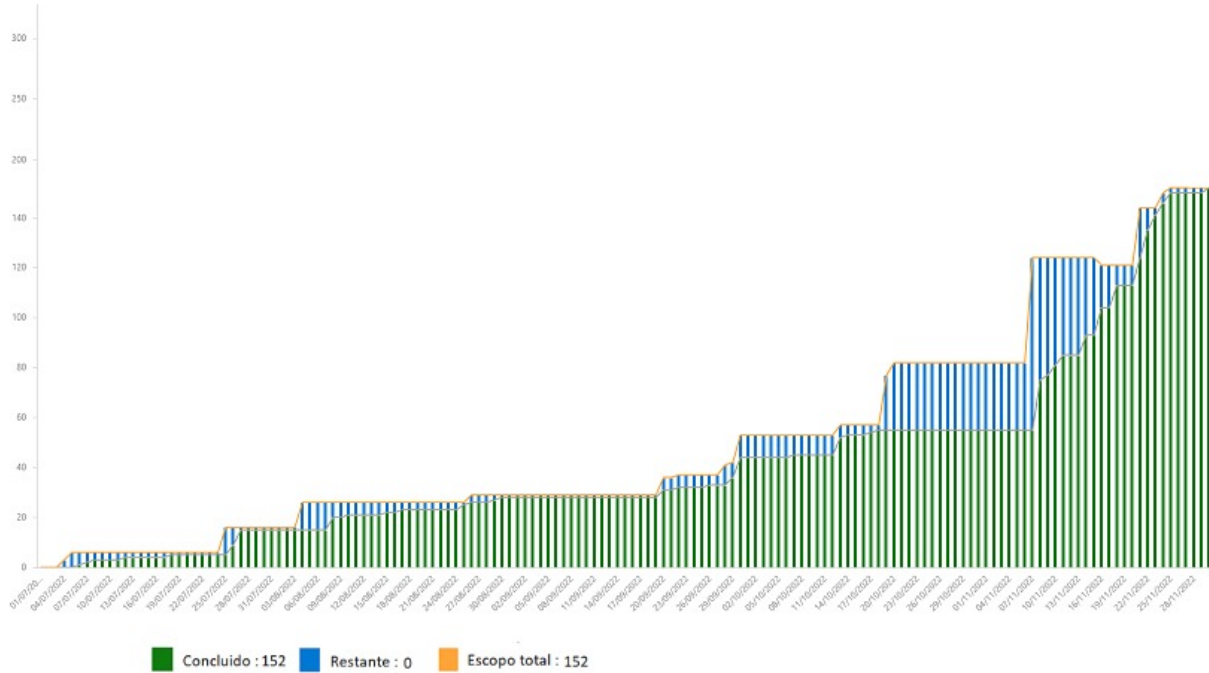
Além disso, outro ponto que contribui para o tradicional se manter positivo, é a questão da divisão da equipe entre outras tarefas, o que faz com que a despesa reduza, sendo dividida entre outros projetos, assim mesmo com um valor total menor, ao final da entrega ainda permanece dentro da margem de lucro.

Figura 12: Gráfico *Burndown* de Entregas por Tarefa do Projeto Ágil



Fonte: Autorial(2022).

Figura 13: Gráfico *Burndown* de Entregas por Tarefa do Projeto Tradicional



Fonte: Autoral(2022).

2.4 Discussão

A partir da comparação realizada entre a aplicação das metodologias ágil e tradicional em projetos distintos, citados anteriormente, foi possível evidenciar uma grande discrepância entre os retornos gerados, tanto na visão financeira quanto na qualidade entregue, pelos dois projetos no ambiente bancário tradicional analisado.

Na questão do modelo tradicional, é visível a experiência apresentada em sua utilização, na comparação pode-se visualizar uma maior facilidade para adaptação da equipe em relação aos problemas enfrentados. Houve um período, como mostrado na Figura 13 em que as entregas estagnaram em relação ao cronograma, entretanto a equipe conseguiu compensar para cumprir o prazo estipulado, o que demonstra que mesmo com a queda na produtividade, o esforço aplicado nas semanas posteriores compensou um obstáculo que apareceu anteriormente. Além disso, a linha de tarefas concluídas permanece próximo do escopo total em praticamente todos os dias, representado na Figura 13, evidenciando a constância nas evoluções das entregas, mantendo sempre o produto o mais atualizado possível em relação à demanda.

Já do ponto de vista de lucratividade, o modelo tradicional se obteve vantagem também por adquirir uma produtividade melhor, mas, além disso por manter uma equipe constituída de uma menor quantidade de desenvolvedores, os quais conseguiram alcançar os resultados estimados, cumprindo o prazo e se mantendo em atividade mesmo fora do projeto estudado, ao mesmo tempo que faziam parte do desenvolvimento do projeto, também se dividiram em outras tarefas terceiras, fazendo assim, a divisão de seus custos dentro da instituição financeira.

De maneira concreta, a instituição financeira onde foram aplicados os estudos de caso, apresenta um cenário desfavorável à aplicação de um modelo de gestão ágil de software, dado que seu rendimento em questão de desenvolvimento se mostrou muito abaixo quando comparado com o modelo tradicional, o *déficit* das entregas estimadas evidencia uma certa inexperiência em relação a utilização de ferramentas mais recentes, como o *scrum*, um fator decisivo para tal adaptação é a mudança de cultura, a base para que seja empregado o modelo ágil é o ambiente estar em sintonia com a metodologia empregada, algo que após a leitura dos dados se vê claramente distorcido. Além disso, a desorganização é mostrada também na Figura 12 pelo fato de que as *sprints* foram organizadas em jornadas de 15 dias, entretanto com o atraso nas tarefas os períodos de entregas variam drasticamente, tornando assim um projeto simples de 3 escopos de entregas, em um longo desenvolvimento de 5 meses e ainda assim, sem concluir todas as atividades definidas ao início do projeto.

Neste caso, na visão de lucratividade o projeto ágil acaba tendo impactos piores devido à produtividade, manter a equipe de 8 pessoas, focadas exclusivamente em um projeto, com custos altos, acarretou um prejuízo grande tanto para a terceirizada que presta serviços para a instituição financeira, no sentido de custear a equipe, quanto para a instituição, pois seu produto entra com atraso em ambiente de produção, onde geraria retorno financeiro, atrasando também futuras entregas, como possíveis atualizações da entrega, desenvolvimento de outros projetos por centralizar uma equipe grande em entregas que não estão se custeando e ainda pior, a perda de possíveis cliente que poderiam já estar utilizando o produto da instituição, entretanto acabam optando por terceiros, tanto no sentido de qualidade, visto que nem todas as funcionalidades desejadas foram entregues quando na visão de negócio, pois passa uma visão negativa a partir do

momento em que a empresa disponibiliza produtos que já apresentam competitividade no mercado.

Levando em conta a aplicação direta da metodologia, o projeto tradicional conseguiu cumprir com o esperado, por exigir apenas a entrega final dentro do prazo estipulado, a equipe se manteve dentro das expectativas do modelo utilizado. Já na questão do projeto ágil, foram especuladas 3 *sprints* de 15 dias, o que não foi cumprido, quebrando a expectativa inicial, além de quebrar o prazo, as entregas de valor ao usuário foram descumpridas, algo essencial para a correta aplicação da metodologia ágil. Fora a metodologia, um ponto comum utilizado nos dois projetos foi o cálculo de ponto de função como forma de estimar o valor do projeto, entretanto as duas aplicações acabaram sofrendo muitas alterações em questão de escopo, muitas tarefas foram adicionadas em relação ao inicial, isso pode ser algo comum para a metodologia ágil, uma vez que a iteratividade causa mudanças por estar em constante contato com o cliente, já no tradicional mostra que seu cálculo não foi tão preciso quando é esperado no modelo.

2.5 Considerações Finais

Consideramos que este trabalho conseguiu atingir seus objetivos de realizar uma análise comparativa entre os projetos selecionados, ágil e tradicional, de maneira a se destacar os pontos positivos e negativos dentro de suas aplicações no ambiente bancário, mantendo a coesão com as métricas, produtividade e lucratividade. Elas foram definidas de início, de forma que os dados utilizados foram abordados concretizando o conceito definido no escopo da metodologia, mostrando descrição detalhada sobre as metodologias usadas nos projetos e sobre suas ferramentas utilizadas durante o processo de desenvolvimento.

Nesse sentido, damos destaque à rica fonte de conhecimento gerada para a instituição financeira, por, mais adiante do esperado, ter mostrado de maneira minuciosa e clara quais foram os erros e acertos dentro da aplicação das metodologias utilizadas nos projetos de estudo, o que também pode ser considerado para outros ambientes de desenvolvimento que desejam realizar a transição da cultura tradicional de gestão de produção de *softwares* para uma ideia mais atual que é o caso do modelo ágil, tendo em vista um escopo de pontos que devem ser tomados como importantes ao se realizar este processo.

Diante disso, este estudo apresenta uma projeção que pode ser aplicadas em diferentes áreas que utilizam do desenvolvimento de *software*, como em órgãos públicos, indústrias, e instituições relacionadas ao governo, dentre outros, por apresentar uma perspectiva voltada à ambientes de desenvolvimento que apresentam alta burocracia por necessitar de uma minuciosidade maior, casos em que o modelo ágil não se encaixa por não possuir escopo fechado. Além disso, como abordagem futura seria interessante uma investigação sobre como realizar a transição cultural destes ambientes evidenciando alternativas para a implantação de forma adequada de outras metodologias.

3 REFERÊNCIAS

ALMEIDA, Renan. **Comparativo entre o Gerenciamento Tradicional e o Ágil de Projetos de Desenvolvimento de Software**. Santa Catarina: Repositório Universitário da Ânima. 2020.

Disponível em:

<https://repositorio.animaeducacao.com.br/bitstream/ANIMA/12211/4/AD6_artigo_cleder_adria_no_rossato_versão_final.pdf>

BERKANI, Akim; CAUSSE, Dominique; THOMAS, Laurent. **Triggers analysis of an agile transformation: the case of a central bank**. Procedia Computer Science, v. 164, p. 449-456,

2019. Disponível em:

<<https://reader.elsevier.com/reader/sd/pii/S1877050919322525?token=11AB414C9028A7018D51E9D8ABB3C465E363B2D5FC84430D990EEEECFEC0FAE36270F59C608810767B9DF839BF6AD804&originRegion=us-east-1&originCreation=20220927014301>>.

BORGES, Julia; DE CARVALHO, Marly. **Sistemas de Indicadores de Desempenho em Projetos**. 2011. Revista de Gestão e Projetos, 2(1), 174-207. Disponível em:

<<https://periodicos.uninove.br/gep/article/view/9418/4187>>.

BRASIL. Ministério do Planejamento, Desenvolvimento e Gestão. Secretaria de Tecnologia da Informação e Comunicação. **Roteiro de Métricas de Software do SISP: versão 2.3** / Ministério do Planejamento, Desenvolvimento e Gestão. Secretaria de Tecnologia da Informação e Comunicação - Setic. – Brasília: MP, 2018.

DA ROSA, Marcos Roque; PEREIRA, Eliane Nascimento. **Metodologias ágeis no contexto da administração pública: análise de estudos de caso de implementação ágil**. 2021. Disponível em: <<https://revista.enap.gov.br/index.php/RSP/article/view/4310#:~:text=A%20pesquisa%20aponta%20que%20o,todos%20os%20envolvidos%20no%20processo>>.

DA SILVA, Jessica Rodrigues; MARIZ FILHO, Marcus Vinícius Arruda Plaisant. **Uma análise dos métodos ágeis e sua importância no processo de transformação dos bancos tradicionais**. Boletim do Gerenciamento, [S.l.], v. 22, n. 22, p. 1-10, jan. 2021. ISSN 2595-6531. Disponível em: <<https://nppg.org.br/revistas/boletimdogerenciamento/article/view/473>>.

DE OLIVEIRA, Izadora Ribeiro e Garcia; SIQUEIRA, Leticia Bettoni; SANTOS, Lauriene Teixeira; MARTINS, Augusto Chaves; OLIVEIRA, José Augusto. **Aplicação de Metodologias de Gerenciamento de Projetos em uma Equipe de Projeto Estudantil**. 2020. Disponível em: <<http://pdf.blucher.com.br.s3-sa-east-1.amazonaws.com/engineeringproceedings/viisimep/315731.pdf>>.

DE SOUZA, Luciana Cristina; PINTO, Sandra Lúcia Aparecida. **Tecnologia e trabalho na era da informação**. Scientia Iuris, v. 21, n. 3, 2017. ISSN 1415-6490. Disponível em: <<http://www.uel.br/revistas/uel/index.php/iuris/article/view/28248>>.

KPI.org. **What is a Key Performance Indicator (KPI)?**. KPI.org. Disponível em: <<https://www.kpi.org/kpi-basics/>>.

KUHRMANN, Marco. **What Makes Agile Software Development Agile?**. IEEE Transactions on Software Engineering, vol. 48, no. 9, pp. 3523-3539. 2022, Disponível em: <<https://ieeexplore.ieee.org/abstract/document/9496156>>.

MORAIS, Izabelly Soares de. **Engenharia de software**. Porto Alegre : SAGAH, 2017. 213p. ISBN: 9788543025902. Disponível em: <<https://plataforma.bvirtual.com.br/Acervo/Publicacao/184098>>

NERUR, Sridhar; MAHAPATRA, RadhaKanta; MANGALARAJ, George. **Challenges of migrating to agile methodologies**. *Communications of the ACM*, 5(48): 73–78. 2005. Disponível em: <http://www.umsl.edu/~sauterv/analysis/challenges_of_migrating_to_agile_methodologies.pdf>

PERIDES, Maria Paula; BARROTE, Erika; SBRAGIA, Roberto. **As Competências de Gestores de Projetos que Atuam com Métodos Ágeis e Tradicionais: Um Estudo Comparativo**. 12(1). São Paulo: Revista de Gestão e Projetos. 2021. Disponível em: <<https://periodicos.uninove.br/gep/article/view/17534#:~:text=São%20elas%3A%20Integridade%20e%20Confiabilidade,Estratégia%20e%20Relacionamentos%20e%20Engajamento>>

Project Management Institute. **Guia PMBOK**. 6a. ed. EUA: Project Management Institute. Management Institute. 2017. Disponível em: <<https://www.pmi.org/pmbok-guide-standards/foundational/pmbok>>.

SCHWABER, Ken; SUTHERLAND, Jeff. **The 2020 Scrum Guide**. 2020. Disponível em: <<https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>>.

SILVA, Gabriel Martins da Costa; DE ARAÚJO JÚNIOR, Alain Viana. **FINTECH: A disruptura do mercado financeiro e a reação dos bancos**. 2019. Disponível em:

<<https://dspace.doctum.edu.br/bitstream/123456789/2237/1/FINTECH%20A%20disruptura%20do%20mercado%20financeiro%20e%20a%20reação%20dos%20bancos.pdf>>.

SOMMERVILLE, Ian. **Engenharia de software**. 8. ed. São Paulo: Addison Wesley, 2019. 768p. ISBN: 9788543024974 Disponível em:

<<https://plataforma.bvirtual.com.br/Acervo/Publicacao/168127>>.

SUTHERLAND, Jeff . **Manifesto para Desenvolvimento Ágil de Software**. Disponível em:

<<http://agilemanifesto.org/>>.

VALENTE, Marco Tulio. **Engenharia de Software Moderna: Princípios e Práticas para Desenvolvimento de Software com Produtividade**, Editora: Independente, 395 páginas, 2020.

Disponível em: <<https://engsoftmoderna.info/cap2.html>>.