



UNIVERSIDADE FEDERAL DO CEARÁ
CENTRO DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO
GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO MECÂNICA

RAYANNA REBOUÇAS DE MEDEIROS

**ANÁLISE DA EFICÁCIA DO GERENCIAMENTO ÁGIL DE PROJETOS EM UMA
EMPRESA FORNECEDORA DE SOFTWARES DO ESTADO DO CEARÁ**

FORTALEZA
2017

RAYANNA REBOUÇAS DE MEDEIROS

ANÁLISE DA EFICÁCIA DO GERENCIAMENTO ÁGIL DE PROJETOS EM UMA
EMPRESA FORNECEDORA DE SOFTWARES DO ESTADO DO CEARÁ

Monografia submetida à Coordenação do Curso de Engenharia de Produção Mecânica da Universidade Federal do Ceará como parte dos requisitos necessários para a obtenção do título de Bacharel em Engenharia de Produção Mecânica.

Orientador: Prof. Me. Alysson Andrade Amorim.

FORTALEZA

2017

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos
pelo(a) autor(a)

M44a Medeiros, Rayanna.

Análise da eficácia do Gerenciamento Ágil de Projetos em uma empresa
fornecedora de software do estado do Ceará / Rayanna Medeiros. – 2017.
86 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará,
Centro de Tecnologia, Curso de Engenharia de Produção Mecânica, Fortaleza,
2017. Orientação: Prof. Me. Alysson Andrade Amorim.

1. Gerenciamento Ágil de Projetos. 2. software. 3. Scrum. I. Título.

CDD 658.5

RAYANNA REBOUÇAS DE MEDEIROS

ANÁLISE DA EFICÁCIA DO GERENCIAMENTO ÁGIL DE PROJETOS EM UMA
EMPRESA FORNECEDORA DE SOFTWARES DO ESTADO DO CEARÁ

Monografia submetida à Coordenação do Curso de Engenharia de Produção Mecânica da Universidade Federal do Ceará como parte dos requisitos necessários para a obtenção do título de Bacharel em Engenharia de Produção Mecânica.

Aprovada em: 07/12/2017.

BANCA EXAMINADORA

Prof. Me. Alysson Andrade Amorim (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. João Vitor Moccellin
Universidade Federal do Ceará (UFC)

Prof. Dr. Abraão Freires Saraiva Júnior
Universidade Federal do Ceará (UFC)

Aos meus pais, Rejane e Roosevelt, que dedicaram e ainda dedicam sua vida à nossa família.

Ao meu irmão Renato, que sempre prezou por minha educação.

E a todos aqueles que aqui buscam o conhecimento.

AGRADECIMENTO

Primeiramente agradeço a Deus por seu amor incondicional e por guiar minha vida para o caminho da luz mesmo quando disso não me torno ciente. Também agradeço a todos os amigos do plano espiritual que auxiliam em minha evolução e estão presentes mesmo quando me sinto só.

Agradeço aos meus pais, Rejane e Roosevelt, do fundo do meu coração por todo o amor a mim dedicado e todos os esforços despendidos para que eu tivesse uma vida digna. Agradeço meu irmão Renato por todo companheirismo e apoio, obrigada por sempre me entender e me ajudar a ser feliz, te amo. Agradeço também a minhas avós, Célia e Leide, por todo carinho, afeto e comidas gostosas que só uma avó pode proporcionar a uma neta.

Agradeço às tias mais corujas do mundo por sempre me lembrarem de minhas qualidades, vocês foram essenciais para que eu tivesse confiança em mim mesma. Um agradecimento especial à tia Rosane por ter sido minha professora desde que eu era criança, sem ela eu não estaria aqui hoje. Agradeço também ao meu tio preferido.

Agradeço aos meus amigos do Farias Brito que até hoje iluminam minha vida de alguma forma, em especial a Geórgia, Bebeto e Elcinho.

Agradeço à Bruna por ter estado comigo do começo ao fim dessa jornada. Obrigada amiga, você é uma das definições mais sinceras que tenho da palavra “amizade”.

Agradeço à Proativa Jr. por ter sido como um lar para mim em meus primeiros anos de faculdade. Lá amadureci mais do que palavras são capazes de definir e as amizades que formei estarão para sempre em meu coração.

Um muitíssimo obrigada a todos os amigos que conquistei na UFC, em especial Marcelo Oka, Maiara Musy, Larissa Queiroz, Lucas Damasceno, Manuelle Rodrigues e Bárbara Louise. Vocês tornaram tudo mais leve, me inspiraram e muitas vezes me impediram de desistir.

Agradeço ao Guilherme por todo o amor, companheirismo e apoio. Não sei qual o caminho que irei enfrentar daqui para frente, mas me tranquiliza saber que vocês estará nele comigo. Te amo.

Agradeço a todos os amigos da RedRocket, em especial Arthur e Thiago, que embarcaram nessa aventura junto a mim e o Guilherme.

Por fim, agradeço a todos os meu Professores, em especial ao Professor Alysson, meu orientador nesse desafio.

“Isso de ser exatamente o que se é ainda vai nos
levar além” (Paulo Leminski).

RESUMO

A velocidade de disseminação das informações evidenciada nos dias de hoje provoca na sociedade mudanças abruptas em um curto intervalo de tempo. Para atender a tais mudanças, projetos de software devem manter-se flexíveis a alterações de escopo, orçamento e cronograma, as quais podem ser viabilizadas por meio de um Gerenciamento Ágil de Projetos (GAP). Nesse contexto, o presente trabalho objetiva analisar a eficácia do emprego do Gerenciamento Ágil de Projetos no setor de desenvolvimento de uma empresa fornecedora de *softwares* por meio do framework Scrum. O trabalho trata-se de um estudo de caso, no qual por meio de uma pesquisa qualitativa foi avaliado o contexto no qual ocorreu a implementação do GAP na empresa em estudo, que anteriormente empregava o Gerenciamento Tradicional de Projetos (GTP), e uma pesquisa quantitativa foi utilizada para que pudessem ser avaliados os resultados dos requisitos de cronograma e orçamento obtidos com o emprego do GAP em comparação ao emprego do GTP. Para as pesquisas, duas amostras foram estudadas, uma contendo projetos geridos sob a abordagem do GTP e outra com projetos geridos sob a abordagem do GAP. Com os resultados obtidos, verificou-se que a adoção do GAP trouxe melhores resultados financeiros para a empresa, a qual passou a obter lucros com seus projetos após um período de prejuízos com estes. A empresa também passou a realizar seus projetos em um menor intervalo de tempo, diminuindo os atrasos apresentados durante a adoção do GTP, comprovando a eficácia do GAP e mostrando a necessidade de uma orientação às empresas desenvolvedoras de *software* sobre como adaptar seus processos à aplicação do Scrum.

Palavras-chave: Gerenciamento Ágil de Projetos, software, Scrum.

ABSTRACT

The speed of dissemination of the information evidenced in the present day causes in society abrupt changes in a short interval of time. To address such changes, software projects must remain flexible to changes in scope, budget, and schedule, which can be made via Agile Project Management (APM). In this context, the present work aims to analyze the effectiveness of the use of Agile Project Management in the development sector of a company providing software through the Scrum framework. The work is a case study in which a qualitative research was carried out to evaluate the context in which the APM implementation occurred in the company under study, which previously used the Traditional Project Management (TPM), and a research quantitative analysis was used to evaluate the results of the schedule and budget requirements obtained using the APM in comparison with the use of the TPM. In this research, two samples were studied, one containing projects managed under the TPM approach and the other with projects managed under the APM approach. With the results obtained, it was verified that the adoption of APM brought better financial results for the company, which started to obtain profits from its projects after a period of losses. The company also started to carry out its projects in a shorter period of time, reducing the delays presented during the adoption of the TPM, proving the effectiveness of APM and showing the need for guidance on software development companies on how to adapt their processes to the application of Scrum.

Keywords: Agile Project Management, software, Scrum.

LISTA DE ILUSTRAÇÕES

Figura 1 – Características de um projeto	23
Figura 2 – Interações entre os processos de um projeto	26
Figura 3 – Processos de um projeto de software	22
Figura 4 – Integração dos processos de monitoramento e controle.....	27
Figura 5 – Modelo linear de gerenciamento de projetos	30
Figura 6 - Seis Princípios para a aplicação do GAP.....	33
Figura 7 – Utilização do GTP e o GAP para cada projeto.	41
Figura 8 – Área disponível no site da empresa para cadastro da equipe externa	47
Figura 9 – Etapas do Estudo de Caso.	49
Figura 10 - Aplicação do modelo cascata na empresa em estudo.....	51
Figura 11 – Definição do fluxo de informações	53
Figura 12 – Wireframe da solicitação de orçamento	63
Figura 13 – Envio da solicitação de orçamento.....	63
Figura 14 – Índice do Guia do Cliente	65
Figura 15 – Funcionalidades no Guia do Cliente.....	66
Figura 16 – Utilização do Slack.....	67
Figura 17 – Telas detalhadas.....	68
Figura 18 – Trello Backlog.....	69
Figura 19 – Trello Development.....	70
Figura 20 – Trello EGINEERING.....	71

LISTA DE GRÁFICOS

Gráfico 1 – Mapeamento dos processos do PMBOK <i>versus</i> Scrum.....	43
Gráfico 2 – Acompanhamento da execução de um projeto sob o GTP.....	56
Gráfico 3 – Tempo de desenvolvimento estimado versus tempo de desenvolvimento realizado no GTP	58
Gráfico 4 – Margem de contribuição dos projetos sob a abordagem do GTP.....	59
Gráfico 5 – Tempo de desenvolvimento estimado versus tempo de desenvolvimento realizado no GAP.....	72
Gráfico 6 – Margem de contribuição dos projetos sob a abordagem do GAP	74
Gráfico 7 – Volume de projetos ao mês.	75
Gráfico 8 – Margem de Contribuição: Gerenciamento Tradicional de Projetos versus Gerenciamento Ágil de Projetos.....	76
Gráfico 9 – Atraso médio de projetos: GTP versus GAP.....	77
Gráfico 10 – Percepção dos desenvolvedores sobre o consumo de tempo das atividades do Scrum.....	78

LISTA DE QUADROS

Quadro 1 – Diferenças entre a abordagem tradicional de gestão de projetos e a abordagem Ágil.....	39
Quadro 2 – Confronto entre processos do PMBOK e Scrum.....	42
Quadro 3 - Requisitos do produto sob a abordagem do GTP.....	55
Quadro 4 – Gerenciamento de entrega sob a abordagem do GTP.....	56

LISTA DE ABREVIATURAS E SIGLAS

TI	Tecnologia da Informação
GAP	Gerenciamento Ágil de Projetos
GTP	Gerenciamento Tradicional de Projetos
PMI	Project Management Institute
ANSI	Instituto Nacional Americano de Padrões
PMBOK	Project Management Body of Knowledge
FCS	Fatores Críticos de Sucesso
TDD	Test Driven Development
APM	Agile Project Management

SUMÁRIO

1. INTRODUÇÃO	16
1.1. Contextualização.....	16
1.2. Objetivos.....	17
<i>1.2.1. Objetivos gerais.....</i>	<i>17</i>
<i>1.2.2. Objetivos específicos.....</i>	<i>17</i>
1.3. Justificativa.....	18
1.4. Metodologia do trabalho.....	19
1.5. Estrutura do Trabalho.....	21
2. REVISÃO BIBLIOGRÁFICA	22
2.1. Gerenciamento de projetos.....	22
<i>2.1.1. Evolução do gerenciamento de projetos.....</i>	<i>22</i>
<i>2.1.2. Definição de projeto e gerenciamento de projeto.....</i>	<i>23</i>
<i>2.1.3. Gestão de projetos segundo o PMI.....</i>	<i>24</i>
<i>2.1.4. Sucesso do Projeto.....</i>	<i>28</i>
2.2. Metodologias de gerenciamento de projetos.....	29
<i>2.2.1. Modelo linear ou “cascata” para gerenciamento de projetos.....</i>	<i>30</i>
<i>2.2.2. Gerenciamento Ágil de Projetos.....</i>	<i>31</i>
<i>2.2.3. Manifesto Ágil.....</i>	<i>34</i>
<i>2.2.4. Framework Scrum.....</i>	<i>35</i>
<i>2.2.5. Gerenciamento Ágil de Projetos X Gerenciamento Tradicional de Projetos.....</i>	<i>38</i>
2.3. Projetos de Software.....	44
3. ESTUDO DE CASO.....	46
3.1. Caracterização da empresa.....	46
3.2. Caracterização do processo.....	47
3.3. Método Proposto.....	49
<i>3.3.1. Etapa 1 - Descrição da adoção do GTP.....</i>	<i>49</i>
<i>3.3.2. Etapa 2 - Resultados obtidos com o GTP.....</i>	<i>49</i>
<i>3.3.3. Etapa 3 - Projeto de transição do GTP para o GAP.....</i>	<i>50</i>

<i>3.3.4. Etapa 4 - Descrição da adoção do GAP.....</i>	<i>50</i>
<i>3.3.5. Etapa 5 - Resultados obtidos com o GAP.....</i>	<i>50</i>
<i>3.3.6. Etapa 6 - Comparação dos resultados obtidos com o GTP e o GAP.....</i>	<i>50</i>
3.4. Aplicação do Método Proposto.....	51
<i>3.4.1. Etapa 1 - Descrição da adoção do GTP.....</i>	<i>51</i>
<i>3.4.2. Etapa 2 - Resultados obtidos com o GTP.....</i>	<i>57</i>
<i>3.4.3. Etapa 3 - Projeto de transição do GTP para o GAP.....</i>	<i>60</i>
<i>3.4.4. Etapa 4 - Descrição da adoção do GAP.....</i>	<i>62</i>
<i>3.4.5. Etapa 5 - Resultados obtidos com o GAP.....</i>	<i>71</i>
<i>3.4.6. Etapa 6 - Comparação dos resultados obtidos com o GTP e o GAP.....</i>	<i>74</i>
4. CONCLUSÃO.....	81
REFERÊNCIAS.....	83
APÊNDICE A – QUESTIONÁRIO APLICADO AOS DESENVOLVEDORES PARA COLETA DE PERCEPÇÕES DO IMPACTO DO GAP NO TEMPO DO PROJETO.....	85

1. INTRODUÇÃO

O presente capítulo introduzirá ao problema abordado, apresentando sua contextualização. Também introduzirá aos aspectos gerais do trabalho, apresentando seus objetivos, geral e específicos, sua justificativa, a metodologia utilizada e sua estrutura, conforme explanado a seguir.

1.1. Contextualização

A sociedade atual é assistida por uma gama de ferramentas tecnológicas que prometem facilitar o dia dia-a-dia. Uma agenda inteligente organiza os compromissos virtualmente, um aplicativo para dietas controla as calorias consumidas, um site compara preços de passagens aéreas e oferece a melhor opção. Os *softwares* tornaram-se parte da rotina de muitas pessoas.

Porém, ao mesmo tempo em que promovem a comodidade, estes alavancam a velocidade de disseminação da informação, provocando mudanças em um espaço de tempo tão curto como jamais vivenciado na história (DANTAS, 2003).

Diante de tanto dinamismo, entre o momento em que a oportunidade de criação de um novo *software* é enxergada até a entrega final do produto, a realidade se transformou e já não é mais a mesma. Parte dos requisitos definidos no planejamento inicial de seu desenvolvimento já não se torna mais necessária, enquanto outros tornam-se imprescindíveis para que o projeto obtenha sucesso frente aos usuários.

Empresas desenvolvedoras de *software*, por muitos anos, desenvolveram os produtos demandados por seus clientes seguindo metodologias tradicionais de gerenciamento de projetos, as quais não eram capazes de acompanhar essa velocidade de transformação da realidade.

Com o crescimento do setor de Serviços em Tecnologia da Informação (TI), o qual no ano de 2016 movimentou 8,6 bilhões de dólares só no Brasil, aumentou também a concorrência entre as empresas prestadoras destes serviços. Também no ano de 2016, o Brasil já contava com 4.470 empresas dedicadas ao ramo (ABES, 2017).

Neste cenário, para garantir sua posição no mercado, empresas fornecedoras de *software* devem ser estruturadas para permitir que seus projetos se adequem às constantes mudanças de requisitos de escopo, cronograma e orçamento, seguindo processos que a

mantenham financeiramente sustentável e garantam que o valor final repassado aos clientes permaneça competitivo, gerando valor a um baixo custo.

A empresa em estudo, uma fornecedora de *software* para terceiros localizada no Estado do Ceará, apresentava um histórico de prejuízos financeiros e atrasos na entrega de seus projetos quando estes eram gerenciados seguindo metodologias tradicionais de gerenciamento de projetos. A fim de melhorar seu desempenho, a empresa implementou o Gerenciamento Ágil em seus projetos.

Diante deste cenário, o trabalho visa responder à seguinte questão: quais os resultados alcançados por uma empresa fornecedora de *softwares* localizada no estado do Ceará ao migrar o gerenciamento de seus projetos do modelo tradicional para o modelo ágil?

1.2. Objetivos

O estudo foi realizado a fim de atingir o objetivo geral descrito a seguir e mais 4 objetivos específicos.

1.1.1. Objetivos Gerais

O objetivo geral do presente trabalho é analisar a eficácia do Gerenciamento Ágil de Projetos, por meio do framework Scrum, no setor de desenvolvimento de uma empresa fornecedora de *softwares* localizada no Estado do Ceará.

1.2.2. Objetivos Específicos

1. Relatar o histórico de resultados obtidos com o emprego da metodologia de gerenciamento tradicional na empresa em estudo.
2. Descrever a aplicação do GAP no setor de desenvolvimento da empresa em estudo.
3. Avaliar os resultados obtidos com a aplicação do Scrum
4. Comparar os resultados obtidos com a aplicação do GAP com os obtidos por meio do GTP.

1.3. Justificativa

Todos os dias, novas ideias de *softwares* capazes de solucionar problemas, sejam eles problemas sociais, econômicos ou até mesmo problemas de produtividade nas empresas, surgem advindos de pessoas e organizações criativas e inovadoras.

Para sair do campo das ideias e tornarem-se produtos, esses *softwares* precisam ser desenvolvidos, programados, transformados em linhas de código capazes de executar comandos, e, para que esse desenvolvimento seja possível, tornam-se necessárias pessoas com competência técnica para tal.

Nesse cenário, as empresas fornecedoras de *softwares* auxiliam as pessoas e organizações detentoras de novas ideias no desenvolvimento de seus *softwares*, uma vez que reúnem pessoas com as capacidades técnicas necessárias para programá-los. A partir dessa criação, novas soluções digitais são disponibilizadas ao mercado, fomentando a economia ao mesmo tempo em que resolvem problemas enfrentados por pessoas e empresas em seu dia-a-dia.

Para essas empresas, cada solução a ser desenvolvida trata-se de um novo projeto, o qual necessita ser gerenciado para que seus objetivos sejam atingidos. Técnicas e metodologias de gerenciamento de projetos estão disponíveis na literatura para serem utilizadas por tais empresas, contudo, quando se trata do gerenciamento de projetos de *software*, as opções disponíveis diferenciam-se bastante entre si e leva a organização por caminhos bastante distintos.

De um lado existem as metodologias tradicionais, já conhecidas pela maioria das organizações e abrangedoras de técnicas com as quais a maioria dos clientes atendidos por tais empresas já estão adaptados. De outro lado, encontra-se o Gerenciamento Ágil de Projetos, o qual aborda técnicas relativamente novas no mercado com as quais parte dos clientes ainda não estão familiarizados.

Apesar de em um primeiro momento o Gerenciamento Tradicional de Projetos soar uma melhor opção devido sua ampla disseminação, este compromete a flexibilidade dos projetos em acompanhar a velocidade com que as mudanças nos softwares ocorrem. O Gerenciamento Ágil de Projetos surge então como uma opção mais flexível. Porém, uma vez que não está amplamente disseminada no mercado, barreiras são impostas à sua implementação, devendo a empresa interessada em adotá-lo ser capaz de adaptar seus processos para abrangê-lo.

A empresa em estudo iniciou suas atividades abordando o GTP, contudo, com os resultados obtidos, principalmente os financeiros, a empresa estava tornando-se insustentável, uma vez que os prejuízos com as alterações demandadas para que o produto se ajustasse ao dinamismo exigido pelos negócios dos clientes eram altíssimos. Nesse cenário, mudanças estruturais na execução dos projetos faziam-se necessárias.

A fim de entender como poderia adequar-se a essa realidade, a empresa realizou um *benchmark* com outras empresas do setor. Em Fortaleza não conseguiu obter resultados satisfatórios, uma vez que as demais empresas aqui localizadas mantinham o gerenciamento tradicional. Para manterem-se sustentáveis, tais empresas não realizavam alterações no produto após os seus testes internos, mesmo que isso significasse que o produto final não atenderia às necessidades atuais de seus clientes.

Foi através do *benchmark* com empresas de outros estados que a empresa em estudo resolveu adotar o Gerenciamento Ágil, principalmente por esse modelo manter o escopo do projeto em aberto, permitindo que alterações nos padrões das funcionalidades definidas, e até mesmo alterações de funcionalidades, pudessem ser atendidas.

Devido a migração do GTP para o GAP, percebe-se a necessidade de verificar a eficácia dessa mudança de abordagem considerando os resultados obtidos com ambas as aplicações. Essa necessidade motiva a realização do presente trabalho, no qual o estudo de caso a respeito da aplicabilidade do Gerenciamento Ágil de Projetos em uma empresa desenvolvedora de *software* tem como base a análise de como a empresa em estudo realizou tal migração, com enfoque em como as técnicas do Gerenciamento Ágil puderam ser adaptadas à realidade da empresa e se os resultados financeiros, de cronograma e mercadológicos desta melhoraram em comparação ao período em que o GTP foi adotado.

1.4. Metodologia do trabalho

Este trabalho segue o método indutivo, o qual assume o conhecimento como fundamentado na experiência, e não em princípios pré-estabelecidos. Segundo o método, a generalização surge de constatações particulares, partir da observação de casos reais (GIL, 1999; LAKATOS; MARCONI, 1993), tal qual ocorre no presente trabalho, no qual é estudada a realidade obtida com aplicação do Gerenciamento Tradicional de Projetos e avaliada a implementação do gerenciamento ágil, partindo da observação de um caso real em uma empresa fornecedora de *softwares*.

O presente trabalho classifica-se quanto à sua natureza como uma pesquisa aplicada, seguindo as definições de (MATIAS e PEREIRA, 2012), uma vez que objetiva gerar conhecimentos para aplicação prática e dirigidos à solução de problemas específicos.

Quanto à abordagem do problema, consiste em uma pesquisa quantitativa e qualitativa, uma vez que tanto traduz opiniões em números classificáveis e analíticos (SILVA E MENEZES, 2005), quanto privilegia o conhecimento das relações entre contexto e ação, fazendo o uso de análises fenomenológicas e subjetividade para alcançar os seus resultados (BERTO e NAKANO, 1999). De forma quantitativa, serão analisados principalmente os resultados dos requisitos de cronograma e orçamento obtidos por meio da aplicação dos gerenciamentos de projeto tradicional e ágil. Já de forma qualitativa, será avaliado o contexto no qual ocorreu a implementação do gerenciamento ágil na empresa e estudo.

Do ponto de vista de seus objetivos, classifica-se como uma pesquisa exploratória, a qual segundo Gil (1991) visa proporcionar uma maior familiaridade com determinado problema, a fim de torná-lo mais explícito.

Sob a óptica dos procedimentos técnicos aplicados, este trabalho configura-se, de acordo com Gil (1991) como uma pesquisa documental, uma vez que para suas análises utiliza materiais que ainda não receberam tratamento analítico; um estudo de caso, por envolver o estudo profundo e exaustivo da aplicação do gerenciamento ágil a fim de se permitir o seu amplo e detalhado conhecimento; e uma pesquisa-ação, devido sua concepção e realização em estreita associação com a resolução do problemas gerados pelas dificuldades que as empresas apresentam ao gerenciar seus projetos de desenvolvimento.

De acordo com Silva e Menezes (2005 p. 32) a população ou universo da pesquisa trata-se “da totalidade de indivíduos que possuem as mesmas características definidas para um determinado estudo.” e a amostra apresenta-se como “parte da população ou do universo, selecionada de acordo com uma regra ou plana. A amostra pode ser probabilística e não-probabilística”. Neste contexto, esta pesquisa utiliza amostras não probabilísticas intencionais, definidas pelos autores como os casos em que a amostra representa o “bom julgamento” da população. A população desta pesquisa pode ser compreendida como os projetos de software em geral, e duas amostras foram selecionadas, sendo ambas amostras de projetos de desenvolvimento de *software*, uma sob a ótica do gerenciamento de projetos tradicional e outra sob o gerenciamento ágil de projetos.

1.5. Estrutura do Trabalho

O presente trabalho é constituído por quatro capítulos, cada qual contendo uma ou mais subdivisões conforme explanado a seguir.

O capítulo 1 apresenta uma introdução ao trabalho. Inicia-se pela contextualização do problema abordado, seguindo pelo seu objetivo principal e mais quatro objetivos específicos. Em seguida explana-se sua justificativa e a metodologia abordada, finalizando com a estrutura de trabalho adotada.

O capítulo 2 encontra-se dividido em três principais subcapítulos. O primeiro apresenta os conceitos de projeto e gerenciamento de projeto, bem como sua evolução ao longo do tempo, as recomendações de gerenciamento de projetos sugeridas pelo PMI e as orientações para definir os critérios de sucesso em um projeto. O segundo subcapítulo abrange as metodologias de gerenciamento de projetos existentes, focando-se no modelo tradicional, representado pelo gerenciamento linear ou “cascata” e as técnicas de gerenciamento ágil, representadas pelo *framework* Scrum. Ao final do subcapítulo as duas abordagens são então comparadas. O terceiro e último subcapítulo apresenta conceitos de *software*, Engenharia de *Software* e os processos a ele inerentes.

No capítulo 3 é apresentado o estudo de caso, iniciando com a caracterização da empresa em estudo e os passos adotados para a realização da pesquisa. Em seguida, explana-se como ocorria o Gerenciamento Tradicional na empresa e quais os resultados obtidos. É também explanado como deu-se a transição para o Gerenciamento Ágil, como esse foi implementado e quais os resultados obtidos. O capítulo finaliza-se com uma comparação entre os resultados obtidos com ambos os gerenciamentos.

Para o capítulo 4, foram reservadas as conclusões gerais obtidas com a pesquisa, respondendo à questão levantada durante a definição do problema. Sugestões para trabalhos futuros também foram apresentadas bem como as considerações finais.

Ao final do trabalho, encontram-se as referências utilizadas para fundamentar o exposto.

2. REVISÃO BIBLIOGRÁFICA

O presente capítulo é composto por três subcapítulos, os quais abrangerão o gerenciamento de projetos de uma forma geral, as metodologias para o gerenciamento de projetos e as definições para os projetos de *software*.

2.1. Gerenciamento de Projetos

Aqui será abordado o histórico de evolução da área de gerenciamento de projetos, seguido das definições encontradas na literatura para projetos e gerenciamento de projetos. Será também abordado como o Project Management Institute (PMI) padroniza as técnicas e processos relativos a área e finaliza apresentando as orientações para a definição de sucesso em um projeto.

2.1.1. Evolução do gerenciamento de projetos

De acordo com Kerzner (2017) embora a abordagem científica sobre o gerenciamento de projetos seja recente, projetos vêm sendo realizados desde o início das civilizações, podendo ser considerados como “tudo aquilo que demanda do ser humano eficiência, organização e planejamento para alcançar objetivo específicos”.

Kozak-Holland (2011) nos traz que grandes projetos da antiguidade foram realizados seguindo sistemáticas semelhantes às praticadas atualmente, complementando o que já dizia BIEZUS (1978) ao afirmar que o gerenciamento de projetos sempre existiu de forma natural ou espontânea, apenas sem terminologia ou características específicas.

Kerzner (2017) apresenta que projetos da antiguidade, como as pirâmides do Egito e a Muralha da China, não possuíam rígidos controles em relação ao cumprimento de prazos e custos, sendo a partir da Revolução Industrial que, com o aumento da complexidade das relações, técnicas de gestão mais sofisticadas passaram a ser exigidas.

Em 1969 é então fundado o Project Management Institute (PMI), maior associação sem fins lucrativos do mundo para profissionais de gerenciamento de projetos, (PROJECT MANAGEMENT INSTITUTE, 2017), o qual em 1998 foi acreditado como desenvolvedor de padrões pelo Instituto Nacional Americano de Padrões (ANSI) e organiza o guia Project Management Body of Knowledge (PMBOK) o qual contém as diretrizes para o gerenciamento de projetos (PROJECT MANAGEMENT INSTITUTE, 2013).

2.1.2. Definição de projeto e gerenciamento de projeto

De acordo com o PMBOK, projeto é “um esforço temporário empreendido para criar um produto, serviço ou resultado exclusivo”. Dessa forma, o projeto deve possuir um início e um fim bem definidos, finalizando-se ao ter seus objetivos alcançados, ou quando for cancelado (PROJECT MANAGEMENT INSTITUTE, 2013). A mesma literatura também afirma que um projeto deverá produzir resultados únicos, ainda que elementos repetitivos estejam presentes em algumas atividades e/ou entregas a ele inerentes, conforme apresentado na Figura 1.

Figura 1 – Características de um projeto



Fonte: adaptado de Xavier (2008)

Dinsmore e Cabanis-Brewin (2006) trazem em contribuição que algumas características são importantes para tornar o projeto único, como ser composto por atividades interdependentes que requisitem recursos múltiplos, possuir entregáveis, e ser regido pela teoria da tríplice restrição, a qual de acordo com Rozenfeld et al (2006) afirma que a qualidade de um projeto possui como principais restrições o custo, o tempo e o escopo e a alteração em qualquer um desses fatores terá consequência nos demais. A escolha de qual restrição será alterada varia de acordo com o foco do projeto. O Guia PMBOK® exemplifica ao afirmar que, em um projeto com alterações em seu escopo fará com custo e tempo se alterem (PROJECT MANAGEMENT INSTITUTE, 2013).

Xavier (2008) diferencia um projeto de um processo, uma vez que o segundo possui procedimentos contínuos e repetitivos, contrastando com o exposto pelo Guia PMBOK®.

O Guia PMBOK® nos traz como alguns exemplos de projetos: a construção de um prédio, planta industrial ou infraestrutura; desenvolvimento de um novo produto, serviço ou resultado e desenvolvimento ou aquisição de hardwares e/ou softwares (PROJECT MANAGEMENT INSTITUTE, 2013).

Somando-se ao já exposto, tem-se a contribuição de Kerzner (2006) o qual entende um projeto como um empreendimento de objetivos bem definidos, e a sua gestão como sendo o planejamento, a programação e o controle das tarefas integradas que levarão ao cumprimento de tais objetivos.

Xavier (2008) concorda com o Kerzner e complementa que em um cenário no qual os investidores estão mais exigentes com os resultados gerados e a utilização dos recursos, executar seus projetos de maneira eficiente é essencial para as organizações modernas.

O Project Management Institute (2013), por sua vez, define o gerenciamento de projetos como “a aplicação do conhecimento, habilidades, ferramentas e técnicas às atividades do projeto para atender aos seus requisitos”

2.1.3. Gestão de projetos segundo o PMI

O Project Management Institute (2013) documentou e padronizou as técnicas comumente aceitas em gestão de projetos em seu Guia PMBOK®. Segundo o manual, o gerenciamento de projetos é composto por 47 processos logicamente agrupados em cinco macroprocessos distintos, sendo eles:

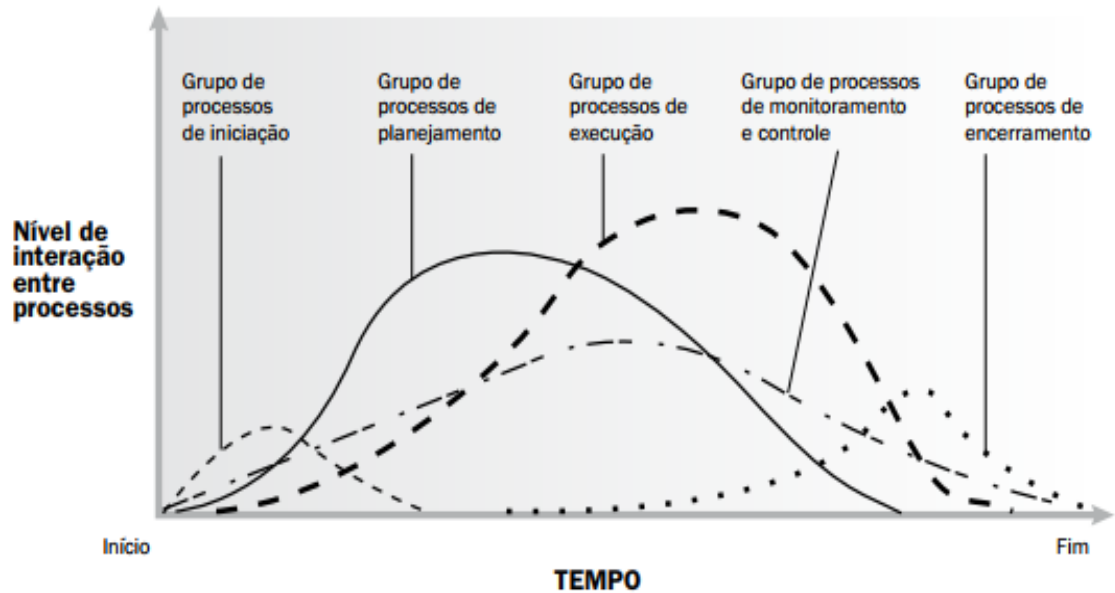
1. Processos de início: englobam os processos de definição de um novo projeto, ou nova fase do projeto. São identificadas as partes interessadas (tanto internas quanto externas), o escopo inicial é definido e o gerente de projetos identificado. Ambas informações são documentadas em um “*Termo de Abertura do Projeto*” cujo aprovação culmina no início do mesmo. O objetivo deste grupo é alinhar as expectativas entre partes interessadas e o objetivo a ser alcançado.
2. Processos de planejamento: são os processos realizados para que o escopo dos esforços a serem demandados seja definido, bem como definir e refinar os objetivos do projeto e os planos de ação necessários para alcançá-los. São desenvolvidos e documentados o plano de gerenciamento do projeto e demais documentos, os quais devem englobar aspectos de escopo, tempo, qualidade, comunicação, recursos

humanos, riscos, aquisições, e gerenciamento das partes interessadas. Caso ocorram mudanças significativas no decorrer do projeto, processos de planejamento tornam-se novamente necessários.

3. Processos de execução: são os processos necessários para que o trabalho definido no plano de gerenciamento seja concluído de acordo com as especificações do projeto, sendo necessário para isso a coordenar pessoas e recursos, gerenciar expectativas e integrar e executar as atividades do projeto. Durante a execução, alterações nas linhas de base do projeto podem fazer-se necessárias, culminando em atualizações no plano de gerenciamento e demais documentos emitidos durante os processos de planejamento.
4. Processos de monitoramento: compreende os processos demandados para acompanhar, analisar e organizar o progresso e desempenho do projeto, para isto identificando áreas que necessitam de alterações no planejamento e dando início a tais modificações. O objetivo de tais processos é identificar variações no plano de gerenciamento do projeto. O monitoramento deve ocorrer de forma contínua a fim de possa ser fornecida ao time do projeto uma visão real sobre a saúde do mesmo.
5. Processos de controle e encerramento: são os processos para que o projeto, ou uma fase deste, possa ser formalmente finalizado, incluindo suas obrigações contratuais, para isso verificando se os processos definidos foram completados de forma apropriada. Este grupo também faz-se necessário em projetos prematuramente concluídos, como, por exemplo, projetos abortados ou cancelados.

Para o Guia, apesar de tais processos serem apresentados como elementos distintos e de interfaces bem definidas, na realidade estes ocorrem sobrepondo-se uns aos outros conforme demonstrado na Figura 2.

Figura 2 – Interações entre os processos de um projeto



Fonte: PROJECT MANAGMENT INSTITUTE (2013).

Vargas(2005), exemplifica essa sobreposição utilizando a aplicação de tais fases em um projeto de desenvolvimento de *software* segundo a Figura 3:

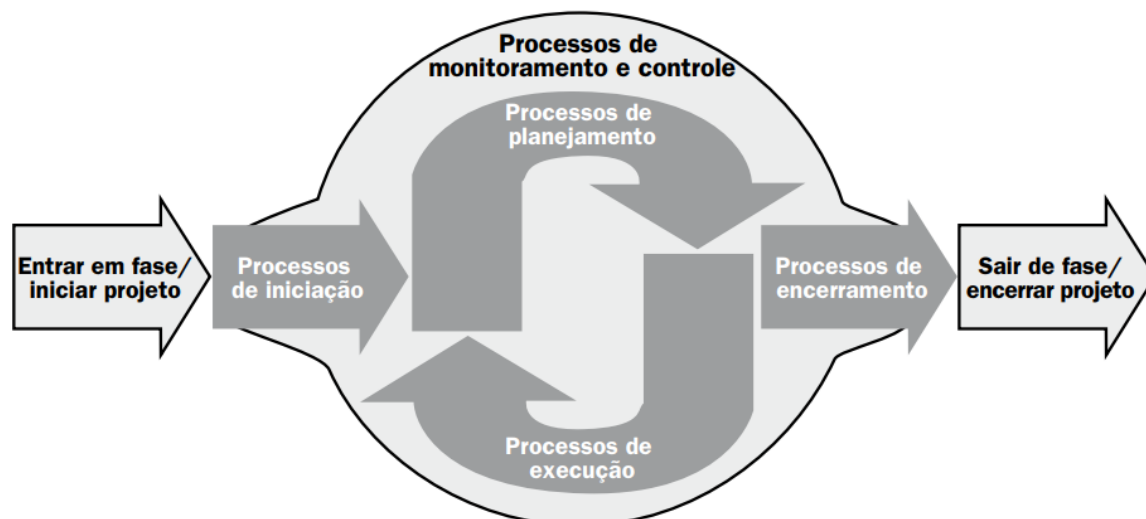
Figura 3 – Processos de um projeto de *software*



Fonte: VARGAS (2005).

O Guia PMBOK® traz ainda que o gerenciamento possui natureza integrativa entre seus processos, a qual exige que os processos de monitoramento e controle ocorram de forma conjunta aos demais 4 grupos de processos, conforme demonstrado na Figura 4.

Figura 4 – Integração dos processos de monitoramento e controle



Fonte: VARGAS (2005).

O Guia PMBOK® conceitua o Ciclo de Vida de um projeto como sendo a série de fases sequenciais que o compõe, desde seu início até seu término. Tais fases são compostas por um conjunto de atividades lógicas que resultarão em uma ou mais entregas. Importante deixar claro que para o PMBOK os grupos de processos não são fases do ciclo de vida do projeto, sendo possível que todos os 5 processos ocorram em uma única fase.

Para o guia, os ciclos de vida podem ser de 3 tipos distintos:

1. **Predeterminados:** são os ciclos nos quais o escopo do projeto, bem como o tempo e o custo exigido para a execução de tal escopo são determinados logo em seu início. Nesse modelo as fases que o compõem, as quais podem ser sequenciais ou sobrepostas, focam em um subconjunto de atividades distintas entre si, dessa forma, as habilidades exigidas para a equipe do projeto em cada uma de tais fases podem variar entre si. Estes ciclos são os mais indicados para casos em que o produto a ser entregue é bem-compreendido desde o início do projeto.
2. **Iterativos e incrementais:** são os ciclos assumidos pelos projetos nos quais a medida em que a compreensão do produto pela a equipe do projeto aumenta, as fases do projeto repetem uma ou mais atividades. Os produtos vão sendo desenvolvidos por uma série de ciclos denominadas “*iterações*” e os “*incrementos*” acrescentam ao projeto, sucessivamente, as funcionalidades do produto. Nesse modelo, um conjunto de entregas é concluído ao final de cada iteração. Esses ciclos são os mais indicados para os casos nos quais a organização precisa administrar mudanças no objetivo e escopo do projeto.

3. Adaptativos: são ciclos de vida iterativos e incrementais nos quais as mudanças ocorrem em um intervalo de tempo bastante curto, levando cerca de 2 a 4 semanas para ocorrerem, e com tempo e recursos fixos. Nesse modelo, o escopo do projeto é desmembrado em um conjunto de requisitos a serem executados denominado de “*backlog*”. No início de cada iteração, são determinados a quantidade de itens prioritários que deverão ser entregues na próxima iteração. Ao final de cada uma dessas iterações, deve ser entregue ao cliente um conjunto de itens que possam ser testados e analisados. As entregas devem receber feedbacks dos patrocinadores do projeto à medida em que forem sendo concluídas para que o backlog responda às suas reais necessidades. Estes são os ciclos mais indicados para os casos em que o projeto está inserido em um ambiente de rápidas mutações, nos quais dificilmente é possível definir os requisitos de escopo antecipadamente.

2.1.4. Sucesso do Projeto

Chow e Cao (2008) conceituam os Fatores Críticos de Sucesso (FCS) como uma abordagem utilizada para identificar e medir o desempenho de um projeto como um todo, tendo como base os trabalhos desenvolvidos por Rockhart (1984), o qual definiu os FCS como “um número limitado de áreas com resultados satisfatórios que asseguram um desempenho competitivo de sucesso para o indivíduo, departamento ou organização”. Em complemento, Kerzner (2017) afirma que os fatores críticos de sucesso devem medir o resultado final de um projeto a partir da perspectiva do cliente.

Kerzner (2006) já havia nos trazido em seu trabalho o comentário de um porta-voz da Nortel Networks que afirmou:

A Nortel define o sucesso de um projeto de acordo com medidas relativas ao cronograma, custo e qualidade; como acordado pelo cliente, a equipe do projeto e as principais partes interessadas [...] a situação e os resultados do projeto são monitorados de perto e analisados regularmente ao longo de todo o projeto para garantir expectativas consistentes e o sucesso geral. O sucesso do projeto é medido pela a satisfação do cliente.

O Guia PMBOK® traz que o sucesso de um projeto, considerando sua natureza temporária, deve ser medido em termos de sua conclusão, referindo-se às últimas entregas aprovadas pelas partes envolvidas e autorizadas para tal. Neste cenário, a figura de um Gerente de Projetos se torna o responsável e o responsabilizável em estabelecer os limites alcançáveis

para o projeto e a realização das entregas aprovadas (PROJECT MANAGEMENT INSTITUTE, 2013).

Kerzner (2017) admite que os Fatores Críticos de Sucesso variam de acordo com o projeto e a sua intenção, porém, alguns exemplos de FCS podem ser citados, tais como:

- Envolvimento do cliente desde o início;
- Processos definidos e avaliações de fase formalizadas;
- Estrutura organizacional interfuncional das equipes;
- Controle de requerimentos e prevenção do aumento gradual de escopo.
- Comunicação entre equipes internas e com o cliente.

2.2. Metodologias de Gerenciamento de Projetos

Kerzner (2001) afirma que para que a excelência em gerenciamento de projetos possa ser alcançada, faz-se necessário um processo repetitivo que possa ser utilizado em cada projeto, sendo a metodologia este processo repetitivo.

Complementando o autor, Charvat (2003) afirma que uma metodologia trata-se de “um conjunto de orientações e princípios que podem ser adaptados e aplicados em uma situação específica [...] pode também ter uma abordagem específica, modelos, formulários e também checklists, usados durante o ciclo de vida do projeto”.

Kerzner (2001) apresentou também algumas características que uma metodologia de gerenciamento de projetos precisa possuir para ser considerada como tal, sendo elas:

- Um nível recomendado de detalhes;
- Uso de modelos;
- Técnicas padronizadas de planejamento, programação e controle;
- Formato padronizado de relato de desempenho;
- Flexibilidade na aplicação nos projetos;
- Flexibilidade para melhorias, quando necessário;
- Facilidade de entendimento e aplicação;
- Ser aceita e aplicada em toda a Organização.

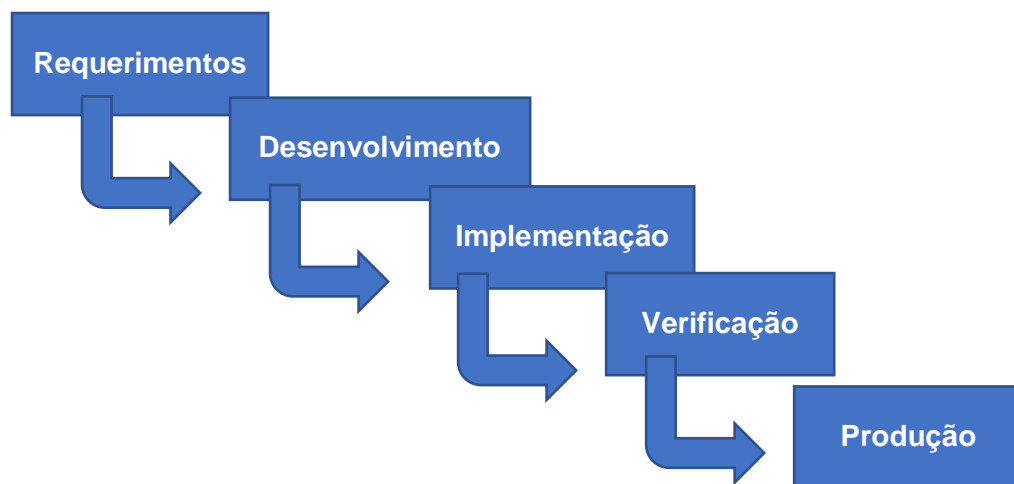
Apesar das definições, estudos apresentam que as metodologias existentes não são completamente adequadas aos projetos, por esse motivo, muitas acabam sendo desenvolvidas dentro da própria organização, ou customizadas (WHITE, FORTUNE, 2006). Dentro desta não-adequação, Joslin e Müller (2015) complementam que uma metodologia deve ser flexível,

sendo capaz de adequar-se às limitações da organização. Para Wells (2013), quando a customização de uma determinada metodologia faz-se necessária, isso torna-se responsabilidade da figura do Gerente de Projetos.

2.2.1. Modelo linear ou “cascata” para gerenciamento de projetos

Veras (2016) afirma que, em gestão de projetos, o método tradicional segue o modelo conhecido por “cascata”, no qual o projeto é composto por uma sequência linear de fases na qual o início de uma é dependente da conclusão da anterior (com exceção da primeira fase). Torres (2014) exemplificou o modelo linear, ou cascata, na Figura 5.

Figura 5 – Modelo linear de gerenciamento de projetos



Fonte: adaptado Torres (2014).

O autor nos traz que o pressuposto básico para a aplicação deste modelo é que, durante a fase de *Requerimento* a documentação criada não poderá mais ser alterada, dessa forma, não é possível durante o projeto visitar etapas anteriores, sendo o foco do gerenciamento garantir que o produto ou serviço seja entregue tal qual foi planejado. Ainda segundo o autor, tal modelo é bastante útil quando o escopo do trabalho é claramente definido.

Contudo Veras (2016) também afirma que existem casos nos quais a modelo cascata não se apresenta como a melhor solução. O autor utiliza o projeto de desenvolvimento de sistemas como exemplo à tal exceção.

Para Wysocki (2007), a abordagem adotada pelas metodologias tradicionais de gerenciamento é intolerante a mudanças e Veras (2016) complementa que por meio do modelo

cascata, o custo de uma alteração cresce exponencialmente conforme o projeto avança entre as fases da cascata. Vargas (2005) afirma ainda que o custo das mudanças ou correções de problemas pode ser pequeno nas fases iniciais do projeto, contudo, concorda com os demais autores ao trazer que com o progresso do projeto tais custos crescem exponencialmente, e complementa que tais custos podem facilmente atingir o custo total do projeto, ao até mesmo a superá-lo.

2.2.2. Gerenciamento Ágil de Projetos

Para Miller (2002) foi na década de 90 que as bases da metodologia tradicional de gerenciamento de projetos começaram a ser questionadas no ramo da Engenharia de Software. Segundo o autor, tal questionamento derivou da observância de que os projetos de software, com uma alta frequência, não conseguiam cumprir seus cronogramas de entregas e extrapolavam o orçamento delimitado.

Ainda segundo o autor, nesse contexto, havia uma grande dificuldade de utilização das metodologias tradicionais de gerenciamento de projeto, chamadas de “*metodologias pesadas*”. Soumyaditya e Singh (2012) afirmam ainda que, em 70% dos projetos de *software* nos quais o modelo tradicional é aplicado, as entregas realizadas não conseguem alcançar um ou mais objetivos definidos no planejamento.

Chin (2004) complementa que a gestão tradicional de projetos não é eficiente quando aplicada em projetos com elevado nível de incertezas, formados por equipes pequenas com muita interação entre seus membros da equipe e clientes, características típicas de projetos de *software*.

Filho (2008) traz que, frente às dificuldades de aplicação dos métodos tradicionais, alguns líderes experientes começaram a adotar métodos de trabalho que se opunham aos conceitos trazidos pelas metodologias tradicionais, e com os resultados obtidos constataram que, apesar de não seguirem os padrões aplicados pelo mercado, tais métodos de trabalho eram bastante eficientes. No decorrer dos projetos em que eram aplicados, tais métodos foram aperfeiçoados até transformarem-se em novas metodologias de gerenciamento, denominadas de “*metodologias leves*” por não utilizarem as formalidades exigidas pelos métodos de gerenciamento tradicionais.

Autores como Martins (2007) afirmam que modelagem ágil não trata-se de uma metodologia, não possuindo ciclo de vida ou processos a serem seguidos, mas sim um conjunto de orientações e princípios que auxiliam a responder com agilidade às mudanças no projeto.

Em concordância, Highsmith (2004) define gerenciamento ágil de projetos como “[...] um conjunto de princípios, valores e práticas que auxiliam a equipe de projetos a entregar produtos ou serviços de valor em um ambiente de projetos desafiador” e complementa que o gerenciamento ágil de projetos é “habilidade para flexibilidade e estabilidade”.

Augustine (2005) vai além e traz que a flexibilidade deve estar acompanhada de estabilidade, executando o gerenciamento ágil com planejamento, e expõe sua definição para esse gerenciamento como sendo: [...] o trabalho de energizar, capacitar e habilitar o time de projeto para entregas rápidas e confiantes, de valor para o negócio, através da integração dos clientes num processo contínuo de aprendizado e adaptação das mudanças, de acordo com suas necessidades e ambiente de negócios”.

Smith (2007) foca-se no gerenciamento ágil para desenvolvimento de novos produtos substituindo o termo “agilidade” por “flexibilidade” e assumindo que mudanças são fundamentais para inovação em produtos, portanto, a flexibilidade nada mais é do que estar apto a realizar mudanças no produto mesmo em estágios avançados de seu desenvolvimento, sem que isso prejudique a qualidade e demais resultados do projeto.

Highsmith (2004) defende a existência de quatro pilares para o GAP, sendo eles:

1. A revolução ocorrida na área de software e seu impacto no desenvolvimento de novos produtos.
2. Valores e princípios que norteiam o GAP.
3. Uma *framework*, que consiste em um modelo de processos propostos.
4. Práticas específicas focadas em resultados.

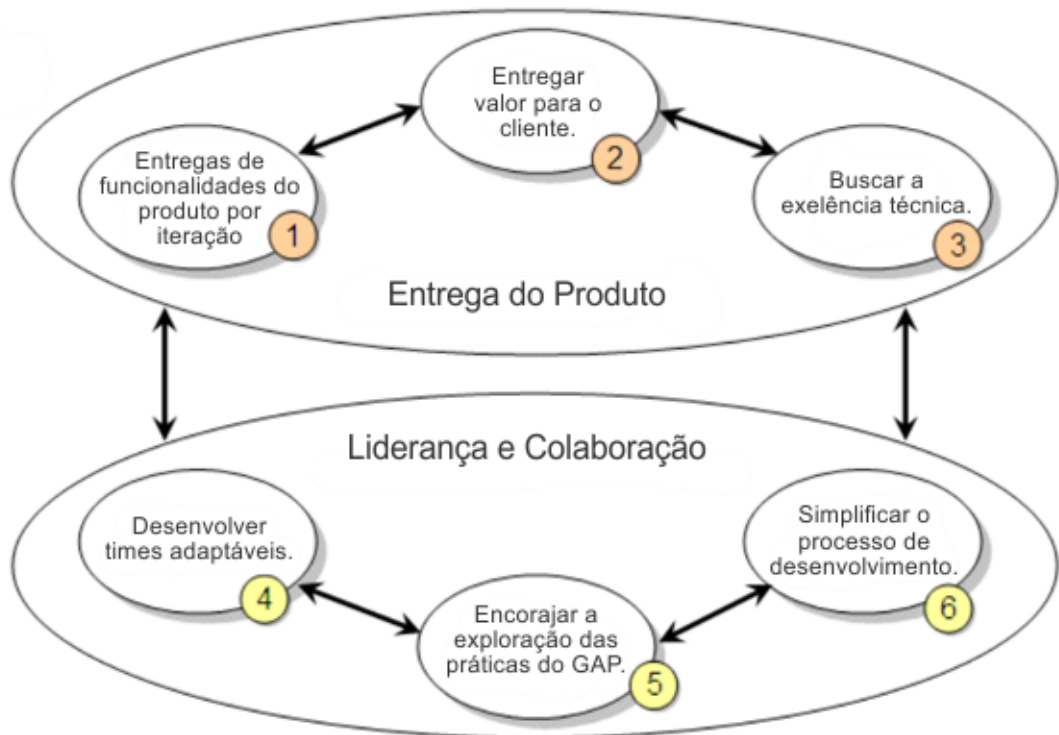
Filho (2008) traz que algumas características comuns às práticas de gerenciamento ágil são: Desenvolvimento Dirigido por Testes (Test Driven Development, TDD), desenvolvimento iterativo, desenvolvimento incremental e colaboração entre as partes.

Beck (2002) explica que o TDD trata-se de um conjunto de técnicas que encorajam projetos simples com aplicação de testes que inspiram confiança. Filho (2008) explica que, diferentemente do aplicado em metodologias tradicionais o TDD não trata a fase de testes como uma fase distinta das demais presente no projeto, mas sim como uma técnica capaz de criar os testes antes de sua implementação, uma vez que Boehm et al. (2000) apontaram que o custo para detecção e correção erros após a entrega do software, pode chegar a ser 100 vezes maior que durante as fases de exploração e design. Jeffries e Melnik (2007) reuniram vários relatos de uso de TDD destacando os ganhos de qualidade conquistados com sua aplicação.

Highsmith (2004) traz que, de acordo com os fatores capazes de impactar as entregas do produto e os fatores relacionados à equipe do projeto, sua liderança e a colaboração

entre as partes, seis princípios para a aplicação do GAP podem ser propostos de acordo com a Figura 6.

Figura 6 - Seis Princípios para a aplicação do GAP



Fonte: Highsmith (2004).

Augustine (2005) também traz princípios para a aplicação do GAP, porém focado para a área de *software*. Segundo ele, pode-se listar:

- Entregas parciais: por meio de iterações, o projeto pode ser dividido em pequenas entregas a fim de diminuir a complexidade do gerenciamento e obter um *feedback* constante dos clientes e usuários finais do produto.
- Co-localização: o autor defende que o time do projeto e o representante do cliente no projeto devem compartilhar um mesmo espaço de trabalho, a fim de que as iterações necessárias possam ser executadas com maior facilidade.
- Plano de entregas: em colaboração constante com o cliente ao longo do ciclo de vida do projeto, a equipe deve elaborar um plano inicial, definindo os recursos de tempo estimados para essas entregas. O plano deve ser elaborado a partir das prioridades de negócio definidas pelo o cliente.
- Plano de iterações: uma vez definidas as entregas, estas devem ser distribuídas entre as iterações do projeto. Tais iterações, bem como suas durações, devem delimitadas,

para que o trabalho possa ser executado e suas entregas sejam finalizadas dentro de uma organização e um arranjo adequado.

- Equipes auto-organizadas: para que o GAP seja flexível e responda a um ambiente dinâmico, o time de desenvolvimento deve ser capaz de se auto organizar a fim que as entregas ocorram de forma cíclica e contínua, sem que para isso faça-se necessário rígidos controles gerenciais.

Também com enfoque na área de *software*, Cohn (2006), concorda com Augustine (2005) e Highsmith (2004) ao apontar o trabalho a ser desenvolvido por iterações como um dos princípios para o GAP, porém complementa acrescentando o trabalho sendo realizado como um só time, entregas de valor em cada iteração e inspeção e adaptação constante como também princípios do gerenciamento ágil.

Highsmith (2004) defende ainda que, para aplicar o GAP corretamente, é recomendado à empresa definir cinco objetivos comuns e alinhados aos princípios do gerenciamento ágil, sendo eles:

1. Inovação contínua.
2. Adaptabilidade do produto
3. Entregas contínuas em um curto intervalo de tempo.
4. Adaptabilidade do processos e pessoas envolvidas.
5. Resultados confiáveis.

2.2.3. Manifesto Ágil

Para Filho (2008) em 2001, 17 desses líderes citados no item 2.1.1 como sendo os primeiros a aplicarem técnicas que se opunham às metodologias tradicionais existentes no mercado, se reuniram em Utah para debater sobre as técnicas que estavam empregando em suas empresas e o resultado foi o que hoje é conhecido como “Manifesto Ágil”, o qual define as principais características de uma metodologia ágil.

De acordo com o Manifesto Ágil:

-“[...]começamos a valorizar: Indivíduos e interações mais do que processos e ferramentas; Software funcional mais do que documentação abrangente; Colaboração com o cliente mais do que negociação contratual; Responder à mudança mais do que seguir um plano”

Para Kalermo e Rissanen (2002), a metodologia ágil ainda enxerga a importância de processos, contratos, documentação e planejamento, contudo, estes itens assumem uma importância menor do que saber lidar com pessoas, atuar em conjunto com o cliente para que a melhor solução possa ser encontrada, ser capaz de entregar um *software* de qualidade e ser flexível para adaptar-se às mudanças.

O Manifesto também aponta que, os líderes que o compuseram afirmam seguir os seguintes princípios:

1. Entrega contínua e adiantada de software com valor agregado para satisfazer o cliente.
2. Adaptação à mudanças nos requisitos, mesmo que essas sejam tardiamente demandadas.
3. Entrega de software funcionando em uma curta escala de tempo.
4. Trabalho conjuntos entre os desenvolvedores do projeto e as pessoas de negócios.
5. Equipe motivada, em um ambiente de qualidade, recebendo o suporte necessário e com autonomia para realizar seu trabalho.
6. Priorização da comunicação de forma pessoal, “*cara-a-cara*”.
7. *Software* funcionando como a medida primária do projeto.
8. Desenvolvimento sustentável com patrocinadores, desenvolvedores e usuários trabalhando em ritmo constante.
9. Atenção à excelência técnica e ao design.
10. Simplicidade, dispensando a realização de trabalhos que não ferem necessários ao projeto.
11. Equipes auto-organizáveis.
12. Reflexão constante da equipe sobre como tornar-se mais eficaz, e a aplicação das modificações necessárias para atingir esse objetivo.

2.2.4 Framework SCRUM

Utilizado desde o início dos anos 1990 para resolver problemas complexos e adaptativos, o *SCRUM* trata-se de um *framework* para a construção de produtos, dentro do qual diversas técnicas e processos podem ser aplicados (SCHWABER & SUTHERLAND, 2016). Definido a partir dos trabalhos de Nonaka e Takeuchi (1986) e formalizado por Schwaber (1995), o *SCRUM* destaca-se dos demais métodos e *frameworks* ágeis pela maior ênfase dada

ao gerenciamento de projetos, reunindo atividades constantes de monitoramento e feedback (SCHWABER & BEEDLE, 2002).

Soumyadipta e Singh (2012) resumem o SCRUM ao afirmar que sua ideia central trata-se em dividir o projeto em várias iterações de curto período de desenvolvimento (aproximadamente um mês), iterações essas denominadas *sprint*. Para cada *sprint* definida, cabe à equipe do projeto planejar, analisar os requisitos do projeto executar, testar o realizado e apresentar o resultado ao cliente a fim de obtenção do feedback necessário. O objetivo é que, a cada *sprint*, o produto seja incrementado até que atinja sua completude. Espera-se com isso que o risco da entrega final do produto não ser condizente com o esperado pelo cliente seja minimizado, uma vez que o projeto é o resultado de múltiplas interações, permitindo que desvios de resultado sejam precocemente identificados durante tais interações.

Schwaber e Sutherland (2016) complementa a definição de *sprint* ao delimitar que durante a execução destas:

- Não é permitido realizar mudanças no projeto que possam prejudicar os objetivos definidos para aquela *sprint*.
- Não é permitido diminuir as metas de qualidade já definidas.
- É permitido ter seu escopo discutido e renegociado entre o time do projeto e o cliente, conforme vai-se aprendendo mais sobre o projeto.

Os autores também organizam as técnicas e processos do Scrum no *Guia do SCRUM*, conforme os padrões desenvolvidos e sustentados por Jeff Sutherland e Ken Schwaber há mais de 20 anos. Para o autor, o SCRUM é fundamentado nas teorias empíricas de controle de processo, as quais possuem sua implementação apoiada por 3 principais pilares:

1. **Transparência:** permitir que aspectos significativos do processo estejam visíveis aos responsáveis pelos resultados.
2. **Inspeção:** garantir que os usuários frequentemente, mas não com uma frequência tão grande a ponto de atrapalhar o desenvolvimento, inspecionem os resultados e os processos aplicados a fim de detectar variações.
3. **Adaptação:** caso variações sejam detectadas, os ajustes necessários devem ser realizados o mais breve possível a fim de minimizar os danos causadas por tais variações.

Para o Guia, o time da organização a ser alocado para a aplicação de SCRUM em um determinado projeto denomina-se “*Time Scrum*” e deve ser composto pelos seguintes atores:

- *Product Owner*: responsável por realizar o gerenciamento do *backlog* do produto, o qual inclui: expressar de forma clara o itens que compõem o *backlog* definido para o projeto e ordená-los a fim de que as metas e missões definidas sejam alcançadas, garantir o valor do trabalho a ser realizado pelo time de desenvolvimento, permitindo que o *backlog* esteja visível ao time e que este entenda todos os seus itens no nível necessário. Importante ressaltar que o *Product Owner* (PO) deve ser uma pessoa, e não um comitê, e suas decisões devem ser respeitadas por toda a organização.
- *Time de desenvolvimento*: são os profissionais responsáveis por realizar o trabalho de entregar uma versão utilizável do produto, com potencial de incremento. É estruturado para possuir a autonomia de organizar e gerenciar o seu próprio trabalho, além de apresentar algumas características próprias, como: ser multifuncional, não reconhecer títulos internos, sendo todos do time de desenvolvimento tratados apenas por *desenvolvedor*, não possuir um único responsável pelo trabalho a ser executado, uma vez que por mais que cada membro possua uma área de especialização, a entrega é de responsabilidade de todo o time, e não possuem sub-times dedicados a áreas específicas de conhecimento.
- *Scrum Master*: considerado como um servo-líder para o time de SCRUM, o *Scrum Master* trabalha para o PO, o time de desenvolvimento e para a Organização como um toda, com trabalhos específicos para cada um, a fim de garantir que o SCRUM seja entendido e aplicado por todos. Ele deve ajudar até mesmo aqueles que são externos ao time de SCRUM a entender como podem interagir com o *Time de Scrum* a fim de maximizar o valor por este criado.

Para a aplicação do SCRUM em projeto, Sutherland (2014), um dos co-criadores da *framework*, traz que as 10 fases para a aplicação do SCRUM são:

1. Selecionar o *Product Owner*.
2. Selecionar o *Time de Desenvolvimento*.
3. Selecionar o *Scrum Master*.
4. Definir e priorizar o *backlog* do produto.
5. Redefinir e estimar o *backlog* do produto por meio da análise constante dos itens a serem desenvolvidos.
6. Planejar a *Sprint* por meio de uma reunião inicial de SCRUM na qual o PO em conjunto com o *Scrum Master* devem consultar o backlog e prever a quantidade de atividades possíveis de serem realizadas no intervalo de cada *sprint*.

7. Permitir que o trabalho realizado esteja visível à organização, por meio de um *Quadro Scrum* onde estejam disponíveis as atividades “*A fazer*”, “*Fazendo*”, e “*Feito*”.
8. Realizar diariamente reunião de SCRUM, com duração máxima de 15 minutos, e na qual três perguntas principais devem ser respondidas: “*O que você fez ontem para ajudar a equipe a concluir o Sprint?*”, “*O que você fará hoje para ajudar a equipe a concluir o Sprint?*” E “*Há algum obstáculo impedindo o alcance da meta do Sprint?*”
9. Realizar reunião de revisão de *sprint* ao final de cada iteração e com a participação de todo o time SCRUM.
10. Realizar retrospectiva da *Sprint* ao receber do cliente o feedback do resultado entregue, a fim de identificar as possíveis falhas ocorridas e corrigi-las.

Ainda segundo o autor, durante a quarta fase, para se realizar a criação e priorização do *backlog*, faz-se necessário o uso de uma ferramenta denominada *Planning Poker*. Por meio do *Planning Poker*, o esforço demandado para a realização de cada item a ser realizado é estimado com o uso de um jogo de cartas. Utilizando como escala de pontos a *Sequência de Fibonacci*, cada membro envolvido no projeto, utilizando sua opinião pessoal e a experiência acumulada com demais projetos, estima o nível de dificuldade das tarefas a serem realizadas. Para isso, toma-se a tarefa mais fácil como tarefa de base, à qual é atribuída um ponto. A partir disso, tem-se uma base de comparação para a pontuação das demais atividades. Nem sempre haverá concordância sobre quantos pontos devem ser atribuídos à cada tarefa, porém, o objetivo é que haja uma discussão entre a equipe para que ao final tenha-se cada atividade pontuada com seu grau de dificuldade.

2.2.5. Gerenciamento Ágil de Projetos X Gerenciamento Tradicional de Projetos

Uma vez concebidas as definições para o Gerenciamento Ágil e o Gerenciamento Tradicional, Shenhar e Dvir (2007), a partir suas pesquisas e experiência na área de gerenciamento de projetos, compararam o GAP e o GTP sob a ótica de nove abordagens, sendo elas:

- Metas do Projetos;
- Plano no Projeto;
- Planejamento;
- Abordagem gerencial;

- Trabalho/Execução;
- Influência da organização;
- Controle do projeto;
- Aplicação da metodologia;
- Estilo de gestão.

Os autores utilizam em seu texto original o termo “abordagem adaptativa”, porém, assim como na pesquisa de Conforto (2009), o presente texto adota o termo “abordagem ágil”. As diferenças estão sintetizadas no Quadro 1.

Quadro 1 – Diferenças entre a abordagem tradicional de gestão de projetos e a abordagem ágil

Abordagem	Tradicional	Ágil
Metas do projeto	Enfoque na finalização do projeto seguindo os requisitos de tempo, custo e qualidade levantados	Enfoque nos resultados de negócio para atingir múltiplos critérios de sucesso
Plano do projeto	Definição de um conjunto de atividades a serem executadas conforme o planejado para atender à tripla restrição	Definição de organização e processos para atingir os resultados esperados para o negócio.
Planejamento	Realizado ao início do projeto	Realizado ao início do projeto e revisado sempre que necessário
Abordagem gerencial	Rígida, focada em manter o planejamento inicial	Flexível, variável e adaptativa
Trabalho/Execução	Previsível, mensurável, linear e simples	Imprevisível, não-mensurável, não-linear, complexo
Influência da organização	Mínima e imparcial a partir do início do projeto	Capaz de afetar o projeto ao longo de sua execução
Controle do projeto	Orientado a identificar desvios do plano inicial e corrigir o trabalho para seguir o planejamento	Orientado a identificar mudanças no ambiente e ajustar o plano adequadamente a tais mudanças
Aplicação da metodologia	Aplicação genérica e padronizada em todos os projetos	Processos adaptados conforme o tipo do projeto

Estilo de gestão	Modelo de gestão padronizado a todos os projetos	Modelo de gestão adaptativo, compreendendo as diferenças entre os projetos
------------------	--	--

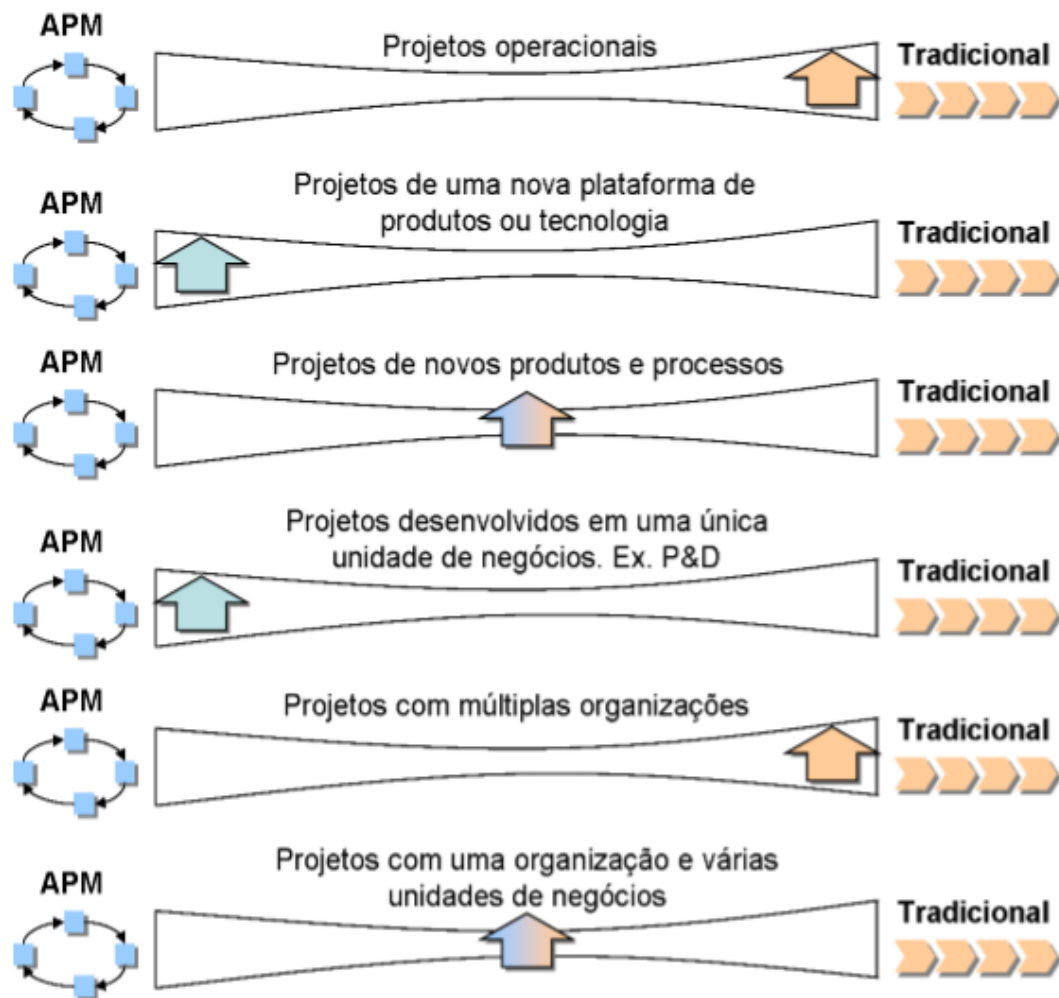
Fonte: adaptado de Shenar e Dvir.

Uma vez que para cada abordagem haverá pontos fortes e fracos a ela inerentes, Wesley (2010) propôs um modelo para que, a partir das especificidades de determinado projeto, pudesse ser avaliado quais as técnicas mais apropriadas à sua gestão. O autor avalia a complexidade do projeto por meio da análise dos seus requisitos e da tecnologia envolvida, sendo os requisitos avaliados em uma escala de “certos” a “incertos”, e a tecnologia variando em uma escala de “conhecida” a “desconhecida”.

Para os autores, quanto mais complexo for um projeto, apresentando requisitos incertos e tecnologia desconhecida, mais recomendado é o seu gerenciamento por meio de processos empíricos, como o Scrum. Quando trata-se de projetos simples, com requisitos certos e tecnologia conhecida, os autores indicam o uso das boas práticas contidas no PMBOK.

Chin (2004) apresenta uma classificação dos projetos considerando o GTP e o GAP, conforme exposto na Figura 7. Para o autor, projetos de desenvolvimento de uma nova plataforma de produtos ou uma nova tecnologia e projetos desenvolvidos em uma única unidade de negócios, no qual a própria unidade assume o papel de cliente, são melhores atendidos pelo GAP. Já para projetos onde várias unidades de negócios estão envolvidas, o autor recomenda o uso combinado de técnicas de GTP e GAP, e, por fim, para projetos operacionais, com baixo nível de incertezas, o GTP apresenta-se como a melhor opção.

Figura 7 – Utilização do GTP e o GAP para cada projeto



Fonte: Chin (2004).

Visualizando o confronto entre PMBOK e Scrum, Rodrigues (2012) realizou o trabalho de mapear os 42 processos descritos no PMBOK e compará-los com as técnicas abrangidas pelo Scrum. Para isso, o autor utilizou uma escala de 0 a 4 pontos para avaliar o grau de atendimento do processo pelo Scrum, sendo 0 o não atendimento e 4 o atendimento completo. Parte dos resultados de sua pesquisa encontra-se na Quadro 2.

Quadro 2 – Confronto entre processos do PMBOK e Scrum

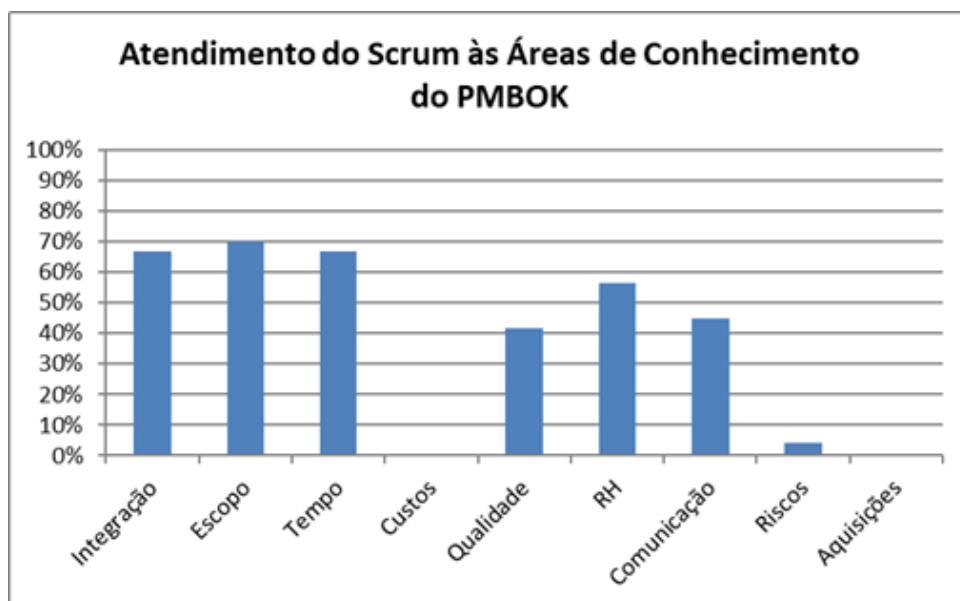
Disciplina	Processo	Atendimento do processo pelo Scrum
Integração	Desenvolver o termo de abertura do projeto	1
Integração	Desenvolver o plano de gerenciamento do projeto	3
Integração	Orientar e gerenciar a execução do projeto	4
Integração	Monitorar e controlar o trabalho do projeto	3
Integração	Realizar o controle integrado de mudanças	3
Integração	Encerrar o projeto ou fase	2
Escopo	Coletar os requisitos	2
Escopo	Definir o escopo	4
Escopo	Criar a EAP	0
Escopo	Verificar o escopo	4
Escopo	Controlar o escopo	4
Tempo	Definir as atividades	4
Tempo	Sequenciar as atividades	2
Tempo	Estimar os recursos das atividades	4
Tempo	Estimar as durações das atividades	2
Tempo	Desenvolver o cronograma	2
Tempo	Controlar o cronograma	2
Custos	Estimar custos	0
Custos	Determinar orçamento	0
Custos	Controlar os custos	0
Qualidade	Planejar a qualidade	3
Qualidade	Realizar a garantia da qualidade	0
Qualidade	Realizar o controle da qualidade	2
RH	Desenvolver o plano de recursos humanos	3
RH	Mobilizar a equipe do projeto	0

RH	Desenvolver a equipe do projeto	4
RH	Gerenciar a equipe do projeto	2
Comunicação	Identificar as partes interessadas	0
Comunicação	Planejar as comunicações	3
Comunicação	Distribuir as informações	2
Comunicação	Gerenciar as expectativas das partes interessadas	2
Comunicação	Reportar o desempenho	2
Riscos	Planejar o gerenciamento dos riscos	0
Riscos	Identificar os riscos	1
Riscos	Realizar a análise qualitativa dos riscos	0
Riscos	Realizar a análise quantitativa dos riscos	0
Riscos	Planejar respostas aos riscos	0
Riscos	Monitorar e controlar os riscos	0
Aquisições	Planejar as aquisições	0
Aquisições	Conduzir as aquisições	0
Aquisições	Administrar as aquisições	0
Aquisições	Encerrar as aquisições	0

Fonte: adaptado de Rodrigues (2013).

O autor verificou que o Scrum atende a 42% dos processos do PMBOK, estando mais evidente nas disciplinas de Integração, Escopo e Tempo, porém não atendendo às áreas de Custos e Aquisições, conforme Gráfico 1.

Gráfico 1 – Mapeamento dos processos do PMBOK *versus* Scrum



Fonte: Rodrigues (2013).

2.3. Projetos de Software

Para Crowston et al. (2003), uma vez que as mudanças do mundo ocorrem de forma cada vez mais rápida e frequente, estando os produtos por nós consumidos cada vez mais globalizados e internacionais, o desenvolvimento de softwares tornou-se uma tarefa nada trivial. As mudanças ocorridas no mundo afetam diretamente os softwares que devem atingir um maior grupo de usuários em lugares bastante distintos entre si.

Tratando-se de softwares, Sommerville (2008) traz que não abrange apenas programas capazes de executar instruções em computadores, mas sim todo o conteúdo que o caracteriza, tais como dados de documentação e configuração.

Pressman (2005) destaca que a área de Engenharia de Software abrange três componentes básicos:

- Métodos, os quais oferecem detalhes sobre como construir um software, englobando tarefas de planejamento e estimativa de projeto, análise de requisito de software e sistemas, projeto da estrutura de dados, arquitetura de programa e algoritmo de processamento, codificação, teste e manutenção.
- Ferramentas, que devem sustentar cada um dos métodos a serem utilizados.
- Procedimentos, os quais definem a sequência nos quais os métodos serão aplicados, constituindo um elo entre esses e as ferramentas abordadas.

Para Pressman (2006), o software é um produto desenvolvido por profissionais da computação da área de Engenharia de Software, ramo responsável por gerenciar os processos de desenvolvimento de softwares computacionais, sendo para isso demandados menos custos possíveis com uma alta qualidade do produto.

Fuggeta (2000) define o processo de software como sendo um conjunto coerente de políticas, estruturas organizacionais, tecnologias, procedimentos e artefatos necessários para a concepção, desenvolvimento, implantação de um produto de software, sendo um processo bem-definido aquele que é capaz de indicar as atividades a serem executadas, os recursos requeridos, os artefatos consumidos e produzidos e demais procedimentos a serem adotados, tais como métodos, técnicas, modelos de documentos, entre outros.

Ainda segundo Fuggeta (2000), a concepção dos processos de software quanto disciplina é algo recente, tendo surgido a partir dos anos 80 e, ao longo dos anos, esforços tem sido demandados para a criação de padrões de qualidade de processo. Para Berger (2003), um processo padrão como base para o planejamento do processo de software em uma organização, bem como os artefatos, procedimentos, ferramentas e papéis envolvidos, permite aos gerentes

de um projeto definir planos em conformidade com os padrões de qualidade e procedimentos da organização.

Utilizado inicialmente em 1968, o conceito de Engenharia de Software surgiu, de acordo com Pfleeger & Atlee (2006) com o intuito de propor soluções mais sistematizadas e controladas para um conjunto de problemas enfrentados no desenvolvimento de sistemas de software, estando baseado nos conhecimentos de práticas e métodos da computação.

Sommerville (2008) afirma que, embora diversas metodologias para os processos de software estejam disponíveis, ambas compartilham algumas características em comum, tais como:

- Especificação do software, abrangendo a definição das funcionalidades que irão compor o software;
- Projeto de implementação, durante o qual o produto de software deve atender às especificações definidas para o mesmo;
- Validação do software. como garantia de que este cumpre as funções requisitadas pelo o cliente;
- Evolução do software, para que este cresça e adapte-se às necessidades levantadas pelo o cliente.

3. ESTUDO DE CASO

Neste capítulo será apresentada a empresa em estudo, com sua caracterização seguida pela caracterização do processo de desenvolvimento de *softwares*. São também apresentadas as etapas propostas para o presente estudo de caso, bem como a aplicação de tais etapas.

3.1. Caracterização da empresa

O estudo abrange uma microempresa do ramo de Tecnologia da Informação, a qual trabalha desenvolvendo *softwares* sob encomenda para terceiros. Cada *software* desenvolvido é projetado exclusivamente para o cliente que o encomendou, não possuindo a empresa propriedade intelectual sobre o mesmo.

Dentre os *softwares* que a empresa desenvolve, estão aplicações *mobile*, as quais podem ser instaladas em *tablets* e *smartphones*, plataformas *web*, que podem ser acessadas em um navegador de internet por meio de um endereço eletrônico (por exemplo, www.exemplo.com), e aplicações *desktop*, as quais são instaladas em computadores específicos e apenas podem ser por eles utilizadas.

A empresa trabalha principalmente para Startups, Pequenas e Médias empresas, as quais buscam *softwares* tanto para otimizar seus processos internos quanto para serem por elas comercializados. Atualmente a empresa atende clientes localizados em Fortaleza, Belo Horizonte, Recife e Rio de Janeiro.

Trata-se de uma empresa jovem, com atividades iniciadas em abril de 2016, mas que apresenta um rápido crescimento, tendo executado 21 projetos nesse período. Atualmente, 11 profissionais trabalham com a empresa, divididos entre programadores, designers, vendedores e responsáveis administrativos.

Atualmente, 7 programadores trabalham para o time de desenvolvimento, sendo dois deles pertencentes à chamada “*Equipe Externa*”. A Equipe Externa foi criada para que a empresa possa manter-se flexível à flutuações na demanda de seus clientes, uma vez que seus membros são adeptos do chamado “*Home Office*”, ou seja, seu trabalho não precisa ser executado no local físico da empresa, podendo ser realizado de sua própria casa, o que permite à empresa, em épocas de aumento da demanda, poder expandir sua equipe sem para isso precisar expandir seu local físico.

A Equipe Externa é contratada por tempo determinado, apenas para executar o projeto no qual está alocada, podendo ter seu contrato renovado para projetos futuros. Essa configuração permite à empresa poder adequar-se a momentos de redução da demanda de seus clientes, uma vez que não precisa manter altos custos fixos em sua operação.

A equipe externa é selecionada apenas após a assinatura do Contrato de Prestação de Serviço entre a empresa e o cliente. Para tal seleção, a empresa mantém consigo um banco de dados de programadores, cujas informações são coletadas por meio de um cadastro prévio de tais programadores no site da empresa, conforme ilustrado na Figura 8.

Figura 8 – Área disponível no site da empresa para cadastro da equipe externa



Explicação ao programador, orientando este a preencher o formulário de cadastro.

Explicação sobre como chegarão ao programador os convites para participar de projetos.

Explicação sobre como o programador será remunerado. No caso, o pagamento ocorre de acordo com as entregas relativas ao projeto.

Fonte: disponibilizado por a empresa.

3.2. Caracterização do processo

O trabalho de desenvolvimento de softwares na empresa inicia-se a partir do momento em que o *software* começa a ser planejado, quando o cliente entra em contato com o setor comercial apresentando a necessidade de desenvolvimento.

Nesse primeiro momento, ainda não há um Contrato de Prestação de Serviços formalizado entre empresa e cliente, contudo, por muitas vezes o cliente solicita à empresa, logo em um contato inicial, a assinatura de um Termo de Confidencialidade, no qual a empresa compromete-se a não divulgar informações sobre o seu projeto à terceiros e nem as utilizar em proveito próprio sem o prévio consentimento do cliente. A solicitação de tal assinatura decorre do perfil de cliente atendido, o qual trabalha com ideias inovadoras no mercado e teme que estas sejam adquiridas pela concorrência antes do desenvolvimento de seu produto.

Após o período de negociação entre empresa e cliente, ambos assinam o Contrato de Prestação de Serviços, o qual formaliza o início da execução do projeto bem como as diretrizes que irão reger a relação entre os envolvidos.

Após a assinatura, a equipe responsável por executar o projeto é definida, selecionando os integrantes da Equipe Interna e Equipe Externa que irão compô-la, tendo como critério de escolha a afinidade de cada desenvolvedor com as linguagens de programação demandadas para o projeto. Os membros da Equipe Externa não são autorizados a trabalharem sozinhos em um projeto, devendo sempre estar acompanhados de alguém da Equipe Interna, além de não poderem assumir o papel de *Scrum Master*, atribuição essa a ser explicada no decorrer do presente trabalho.

O desenvolvimento dos *softwares* da empresa divide-se em desenvolvimento *front-end* e desenvolvimento *back-end*.

O Desenvolvimento front-end abrange a construção das áreas visuais e gráficas do software, tais como suas cores, os botões que irão compô-lo, a disposição dos ícones, gravuras, textos, menu... As linguagens de programação utilizadas na empresa durante o desenvolvimento front-end são *Ionic 2* e *Ionic 3*, quando o software trata-se de uma plataforma mobile, *Angular 4*, quando o software trata-se de uma plataforma web e *React*, *HTML*, *CSS* e *JavaScript*, os quais são utilizados tanto em aplicações mobile quanto plataformas web.

Já o desenvolvimento back-end abrange a construção da lógica que rege o software, sendo para isso utilizado a linguagem *Ruby on Rails*, tanto para a construção de aplicações mobile quanto plataformas web.

Durante o desenvolvimento, os programadores são responsáveis por aplicar no código de programação por eles escrito os chamados *Testes Automáticos*, os quais compreendem um conjunto de comandos para que a aplicação em construção esteja sempre testando se suas funcionalidades estão operando dentro do tecnicamente esperado.

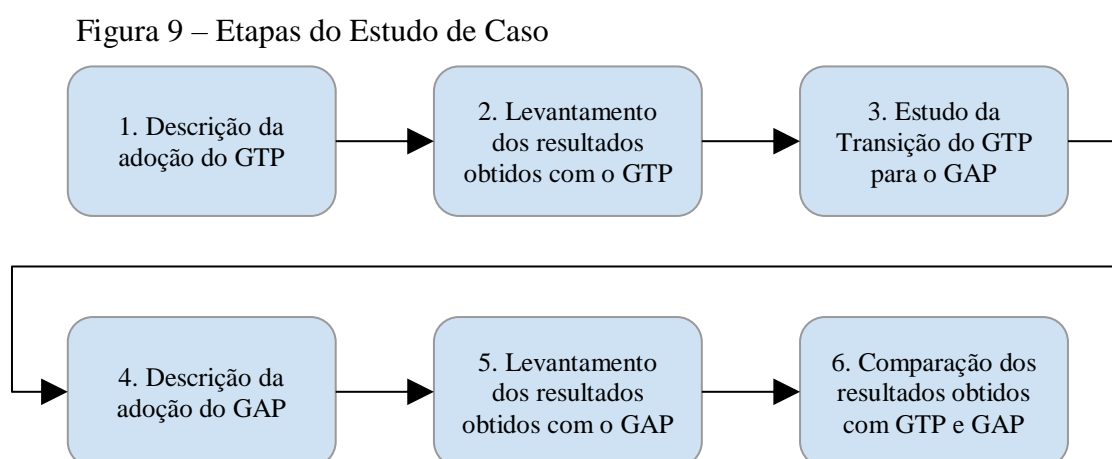
Além dos Testes Automáticos também são aplicados os chamados *Testes Manuais*, estes, porém, são realizados utilizando usuários reais que irão coletar suas percepções sobre as

funcionalidades desenvolvidas. Os Testes Manuais são realizados somente após o desenvolvimento das funcionalidades.

O processo de acompanhamento dos projetos e a validação dos softwares desenvolvidos são objeto de estudo deste trabalho, os quais variam de acordo com a abordagem de gerenciamento de projetos adotada, como será explorado no decorrer do estudo.

3.3. Método Proposto

O presente estudo de caso tem sua aplicação dividida em seis etapas conforme Figura 9.



Fonte: autora.

3.3.1. Etapa 1 - Descrição da adoção do GTP

Nesta etapa, foram levantadas informações sobre como a empresa em estudo aplicava o Gerenciamento Tradicional de Projetos em sua realidade, descrevendo os processos utilizados, bem como os seus envolvidos. Essa fase visa compreender quais os processos e atividades que levaram a empresa a atingir os resultados apresentados.

3.3.2. Etapa 2 - Resultados obtidos com o GTP

Nesta etapa foram levantadas informações a respeito dos indicadores de cumprimento dos requisitos de cronograma e orçamento dos projetos, os quais medem o tempo de desenvolvimento previsto em relação ao tempo realizado e à lucratividade do projeto, respectivamente. Para isso, foram considerados 4 projetos executados sob o GTP. O cumprimento dos requisitos de escopo será apresentado de forma implícita. Essa fase visa

apresentar as lacunas existentes entre os resultados esperados e os resultados obtidos, para que então possa ser compreendido onde a empresa deve concentrar seus esforços a fim de melhorar seu desempenho.

3.3.3. Etapa 3 - Projeto de transição do GTP para o GAP

Nesta etapa é estudado como a empresa preparou-se para a migração do GTP para o GAP, abrangendo a necessidade de adaptação das técnicas ágeis à realidade da empresa, considerando as limitações impostas por seus clientes, time, e o mercado como um todo, bem como a necessidade de adequação da empresa à nova abordagem de gerenciamento de seus projetos. Essa fase visa compreender como os processos e atividades da empresa foram alterados para a implementação do GAP.

3.3.4. Etapa 4 - Descrição da adoção do GAP

Nesta etapa é apresentado como o Gerenciamento Ágil foi adotado por a empresa, descrevendo os processos, atividades e ferramentas utilizados bem como e interações entre os envolvidos. Essa fase visa compreender como a empresa implementou o GAP para que os resultados apresentados no tópico seguinte fossem alcançados.

3.3.5. Etapa 5 - Resultados obtidos com o GAP

Nesta etapa foram coletados os dados relativos ao cumprimento de requisitos orçamentários e de prazo dos projetos gerenciados sob a ótica do GAP, bem como demais indicadores da empresa, sendo para isso considerados os resultados obtidos a partir de 3 projetos executados sob a abordagem ágil. Essa fase visa levantar os resultados adquiridos por a empresa com a adoção do GAP.

3.3.6. Etapa 6 - Comparação dos resultados obtidos com o GTP e o GAP

Nesta etapa, os resultados obtidos na segunda etapa foram comparados com os obtidos na quinta etapa a fim de que possa verificar se o emprego do GAP realmente alavancou o desempenho da empresa, preenchendo as lacunas entre os resultados esperados e os resultados obtidos gerados pelo GTP. Ao final, será identificado qual abordagem foi mais benéfica ao desempenho da empresa.

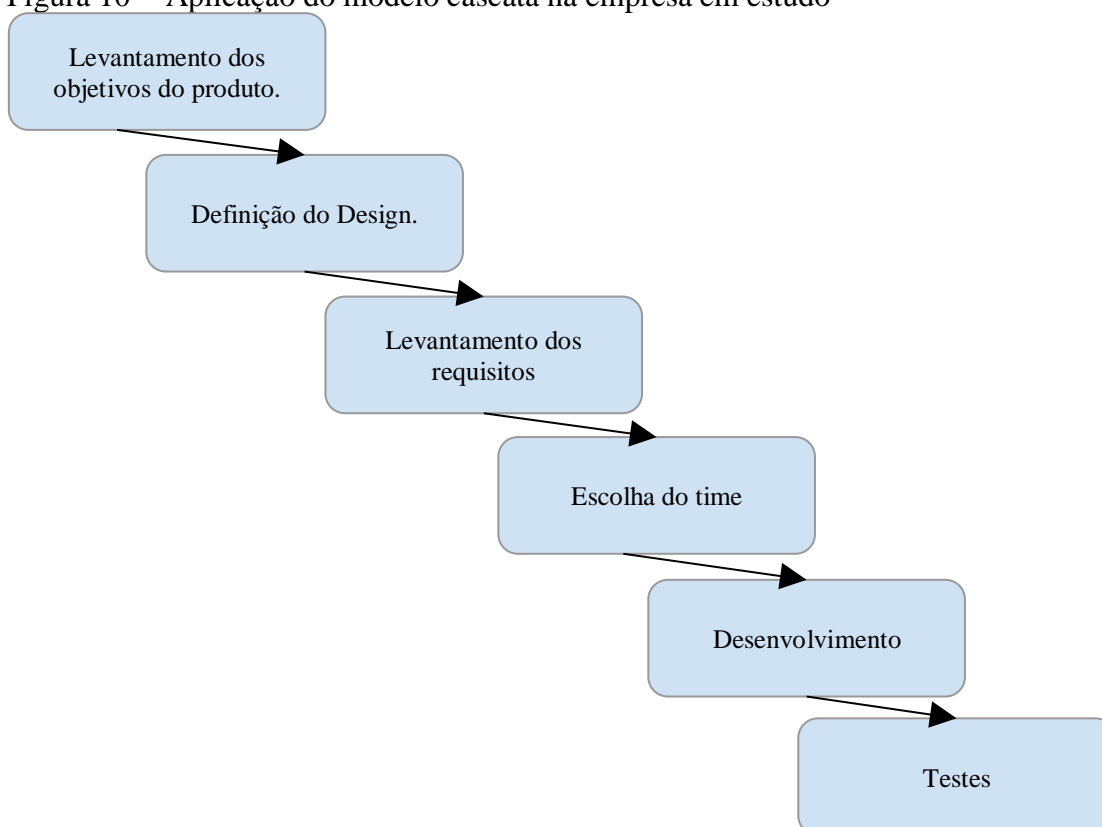
3.4. Aplicação do Método Proposto

As etapas sugeridas no item “3.3. Etapas do Estudo de Caso” foram aplicadas na empresa em estudo. Os resultados desta aplicação serão explanados a seguir.

3.4.1. Etapa 1 - Descrição da adoção do GTP

Durante a aplicação do gerenciamento tradicional, a empresa utilizou o modelo linear, ou cascata, representado na Figura 10.

Figura 10 - Aplicação do modelo cascata na empresa em estudo



Fonte: autora.

Em um primeiro momento, os interessados em desenvolver sua plataforma digital, denominados também como clientes, procuravam o responsável comercial da empresa para uma primeira reunião, denominada “Reunião de *Briefing*”. Nesse momento, o objetivo era captar do cliente as respostas de cinco principais perguntas:

1. Qual a natureza da plataforma, *web* ou *mobile*?
2. Quais serão os usuários do sistema?
3. Quais as permissões de acesso que cada usuário possui?
4. Quais as funções que a plataforma deverá desempenhar?

5. Há algum modelo parecido no mercado que sirva de referência?

Ao longo da reunião, outras informações além das listadas poderiam ser obtidas.

Os dados eram então armazenadas em um documento, o qual não precisava seguir um determinado padrão, apenas apresentar as respostas das citadas perguntas. Tal documento era enviado aos Designers da empresa, os baseados nas informações obtidas, desenhavam a primeira versão de como se comportaria o fluxo de informações na plataforma, bem como todos os itens que iriam compor cada uma de suas telas. A primeira versão costumava seguir o padrão exposto na Figura 11.

o cliente aprovasse, partia-se então para a etapa de definição dos requisitos do necessários ao desenvolvimento da plataforma.

Baseado no desenho de como o fluxo de informações deveria se comportar, bem como quais elementos deveriam compor a plataforma, eram então definidas quais as tecnologias que, dentro do portfólio da empresa, melhor atenderiam às necessidades do projeto. Em seguida era definido o *Backlog do Produto* o qual continha descrito todas as funcionalidades que o sistema deveria conter. Após as funcionalidades levantadas, três programadores da empresa eram convidados a realizar o *Planning Poker* para a determinação do tempo necessário ao desenvolvimento de cada uma das funcionalidades listadas, e, a partir do tempo de desenvolvimento definido, era calculado o valor total do projeto.

Destaca-se que, apesar do *Planning Poker* ser uma ferramenta do Gerenciamento Ágil, ela já era de conhecimento da empresa e, por adequar-se às suas necessidades, já era utilizada em seus projetos,

O Quadro 3 apresenta o modelo de tabela abrigando o backlog, o tempo de desenvolvimento de cada funcionalidade e o seu valor monetário correspondente, adotado por a empresa.

Quadro 3 - Requisitos do produto sob a abordagem do GTP

Funcionalidade	Dias úteis	Valor
Lista de Pacientes	1	R\$111,00
Adicionar novo paciente	1	R\$222,00
Ver Paciente	1	R\$111,00
Novo SAE	2	R\$555,00
Definir tipo de SAE	1	R\$148,00
Buscar necessidades empíricas	1	R\$296,00
Enviar necessidades de um SAE	1	R\$296,00
Buscar Diagnósticos de um SAE	2	R\$370,00
Enviar Diagnósticos do SAE	1	R\$222,00
Buscar Evidencias de um SAE	2	R\$407,00
Enviar evidencias do SAE	1	R\$222,00
Buscar Fatores Relacionados de um SAE	2	R\$370,00
Enviar fatores relacionados do SAE	1	R\$222,00
Buscar ações para um SAE	1	R\$222,00
Enviar ações para um SAE	1	R\$222,00
Buscar SAEs	1	R\$148,00
Emitir PDF	1	R\$296,00
TOTAL APROXIMADO	21	R\$4.440,00

Fonte: Empresa.

Até esta fase, o projeto ainda estava sendo negociado entre o cliente e a empresa em estudo. Somente após a definição do valor do projeto é que o cliente aprovava ou não o desenvolvimento do produto, portanto, os processos de início e alguns processos de planejamento apresentados no item 2.1.5 deste trabalho ocorriam em fase anterior à assinatura de Contrato entre a empresa e o cliente, e os demais processos de planejamento e os processos de execução, monitoramento e controle ocorriam em fase posterior.

O cliente aprovando o orçamento e os requisitos, iniciava-se então a escolha do time responsável por executar o projeto, composto geralmente por um Gerente do Projeto, um Designer e dois programadores, um responsável pelo o *back-end* e outro pelo o *front-end*.

Considerando as funcionalidades a serem desenvolvidas, o Gerente do Projeto as organizava em pacotes de entrega, determinando quais as datas em que cada uma deveria estar pronta, considerando as relações de dependência entre elas. Eram também definidas a datas em que itens a serem enviados pelo o cliente, como o endereço de domínio da plataforma, precisariam estar disponíveis ao time do projeto.

O Quadro 4 apresenta um exemplo de *Entrega* a ser controlada pelo Gerente.

Quadro 04 – Gerenciamento de entrega sob a abordagem do GTP

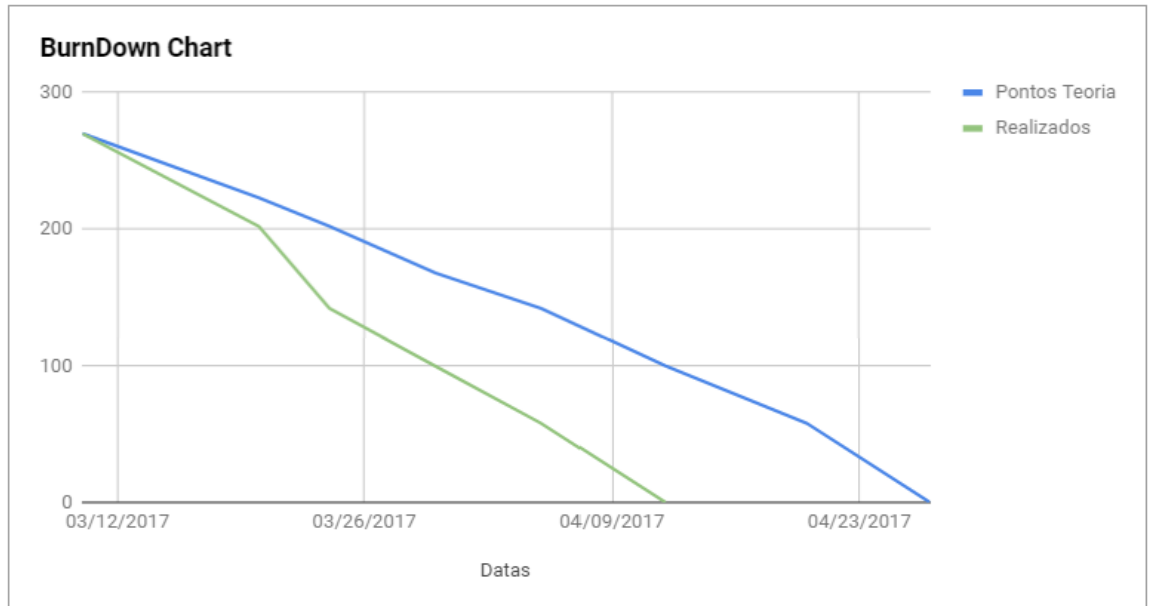
Entrega:	2	Data de início:	20/03/2017	Data de término:	24/03/2017	Pontos de dificuldade:	21	Observações:
Feature	Duração em horas	Dificuldade:	Status:	4,2 dias úteis. Terminou no dia 20/03.				
Novo SAE	0	1	Finalizado					
Definir tipo de SAE	4	2	Finalizado					
Buscar necessidades empíricas	12	8	Finalizado					
Enviar necessidades empíricas	9	5	Finalizado					
Escolher necessidade humanas básicas	5	5	Finalizado					

Fonte: Empresa.

Um sistema de pontos de dificuldade era utilizado para que, havendo impedimento para o desenvolvimento de determinada funcionalidade, o programador pudesse então trabalhar em uma outra que possuísse pontos de dificuldades semelhantes. Os pontos utilizados eram os mesmos definidos durante o *Planning Poker*.

O Gerente do Projeto comparava os pontos de dificuldade planejados com os executados para uma determinada data a partir do Gráfico 2.

Gráfico 2 – Acompanhamento da execução de um projeto sob o GTP



Fonte: disponibilizado pela a empresa.

No início do gráfico vê-se a quantidade total de pontos de dificuldade a serem desenvolvidos durante o projeto. Conforme o tempo avança, a quantidade total diminui, até o ponto 0 do eixo horizontal, o que representa o projeto concluído. A linha azul representa o desenvolvimento previsto e a linha verde o desenvolvimento real. A linha verde abaixo da linha

azul representa o projeto adiantado, acima representa atrasado, e ocupando o mesmo espaço, representa o projeto em dia.

Durante a execução, o papel do Gerente do Projeto era então garantir que o desenvolvimento ocorresse tal qual o planejado.

Durante o desenvolvimento das funcionalidades, testes automáticos do código de programação eram criados pelos programadores. Após o desenvolvimento de todas as funcionalidades, iniciava-se o período de testes manuais, realizados não apenas pelo o time do projeto, mas também por pessoas da empresa que pudesse oferecer uma visão externa ao produto. Havendo correções a serem realizadas, estas eram repassadas ao programadores e quando o Gerente do Projeto considerasse que tudo estava conforme, o produto era repassado ao cliente.

O cliente então inspecionava o produto entregue a fim de verificar se ele correspondia às suas necessidades. Havendo necessidades de alterações, estas eram repassadas à equipe, desde que estivessem dentro do escopo das funcionalidades definidas, e, quando o produto estivesse de acordo com as expectativas, o projeto era considerado como encerrado pelas as partes.

3.4.2. Etapa 2 - Resultados obtidos com o GTP

Como a empresa em estudo trabalha com o desenvolvimento de novos produtos para o mercado, alguns de seus clientes a solicitam a assinatura de um *Termo de Confidencialidade* sobre o seu projeto, sendo assim, as informações provenientes do relacionamento da empresa com tais clientes não podem aqui ser divulgadas. Dessa forma, para obtenção de resultados, considerar-se-á os dados obtidos a partir de quatro dos oito projetos que a empresa desenvolveu no formato acima descrito.

Dentre tais projetos, dois são relativos ao desenvolvimento de um aplicativo, um relativo a uma plataforma web, e um relativo tanto a um aplicativo quanto uma plataforma web.

Tempo

A empresa apresenta que apenas um dentre os quatro projetos em questão apresentou problemas de cumprimento dos prazos durante a fase de execução, estando a maioria sempre adiantada em relação ao prazo a ser cumprido. Porém, ao finalizar a etapa de realização dos testes manuais no produto e este ser repassado para a inspeção do cliente, na maioria dos

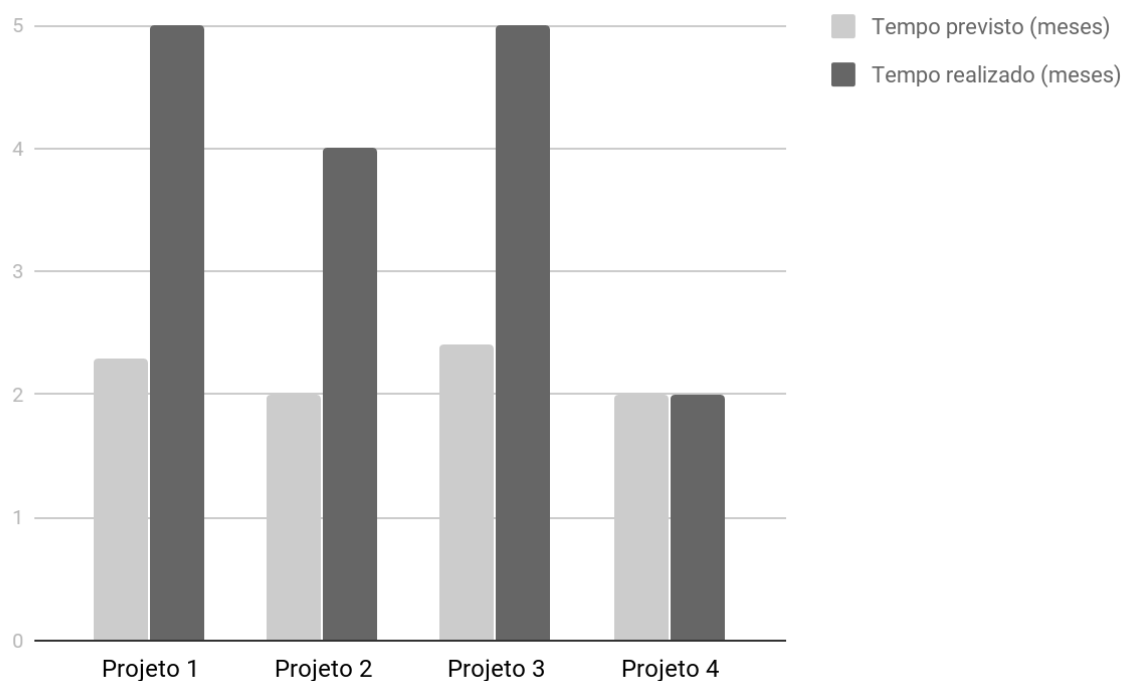
casos este cliente relatava que as funcionalidades desenvolvidas não estavam comportando-se da forma desejada.

Esse comportamento relatado nada relacionava-se com a qualidade técnica do software, uma vez que o código de programação desenvolvido estava testado e seguia as boas práticas de Engenharia de Software. Os pontos de melhoria apontados por o cliente, partiam de uma análise subjetiva que este fazia sobre a real necessidade de cada funcionalidade e se a forma que esta estava comportando-se era a melhor forma que, ao seu ver, poderia ser aplicada ao produto. Assim, alterações eram demandadas, contudo tais alterações não constavam em acréscimo de funcionalidades, e sim em um comportamento diferente das funcionalidades definidas.

Com alterações no comportamento do produto demandadas, o tempo para a conclusão dos projetos estendeu-se além do inicialmente acordado, gerando desconforto para o cliente e prejudicando a imagem da empresa.

A comparação entre o tempo de desenvolvimento estimado e o tempo de desenvolvimento realizado para os quatro projetos abordados encontra-se no gráfico 3.

Gráfico 3 – Tempo de desenvolvimento estimado *versus* tempo de desenvolvimento realizado no GTP



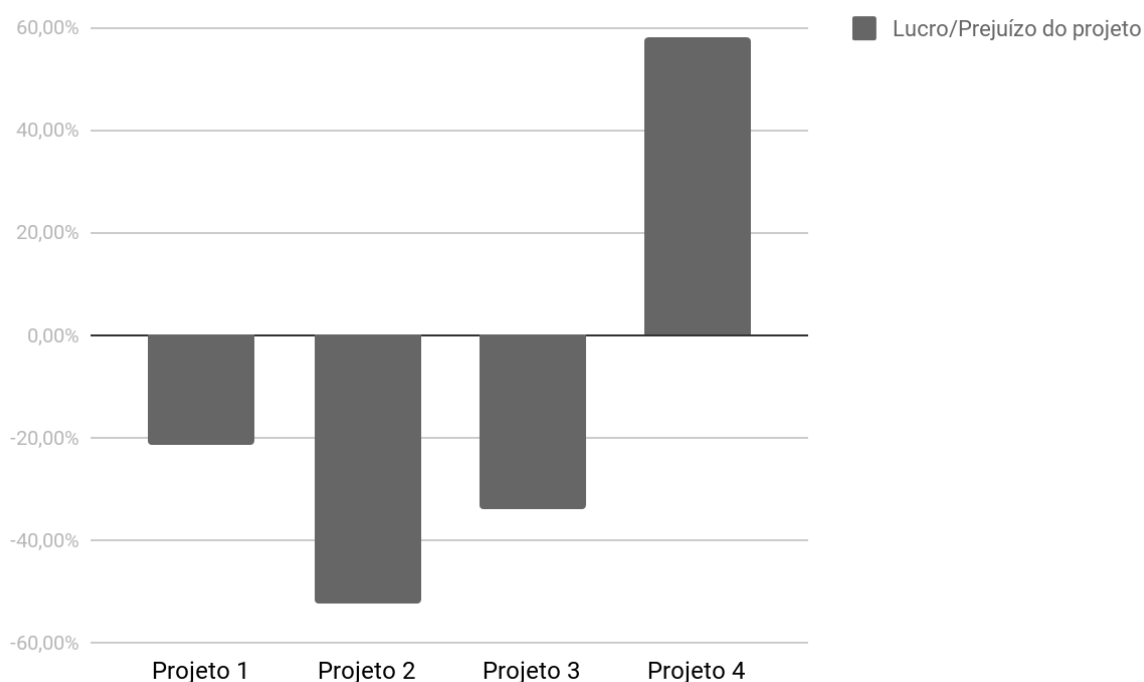
Fonte: Autora.

Observa-se que os Projetos 1, 2 e 3 estenderam-se aproximadamente o dobro do previsto inicialmente. Para o Projeto 4, o realizado esteve em acordo com o previsto, contudo, este possuía uma peculiaridade que o diferenciava dos demais: o Projeto 4 tratava-se de um aplicativo a ser utilizado na medicina, automatizando diagnósticos de recém-nascidos da unidade neonatal de um hospital em Fortaleza. Tal automação seguia regras da medicina, não passíveis de alteração pelo o próprio cliente, dessa forma, ao ser entregue o produto final, não foram demandadas modificações no mesmo, permitindo que o prazo de desenvolvimento não fosse ultrapassado.

Orçamento

Para a definição dos orçamentos dos projetos foi considerado apenas as horas totais necessárias ao desenvolvimento, uma vez que gastos com servidor, hospedagem e domínio eram arcados separadamente pelo o cliente. Ao comparar o valor o total definido para cada projeto e as horas de serviço excedentes, não arcadas pelo cliente, mas contabilizadas pela a empresa, obtém-se resultado apresentado no Gráfico 04.

Gráfico 4 – Margem de contribuição dos projetos sob a abordagem do GTP.



Fonte: Autora.

Observa-se que o Projeto 2 obteve a pior margem de contribuição, chegando esta a ser de 52,33%. Já o Projeto 4, uma vez que não houveram alterações após a fase de testes, obteve uma margem positiva de 58,09%.

Os projetos que se estenderam além do prazo inicialmente definido foram os que trouxeram prejuízos para a empresa. Contudo, observa-se que, embora ambos tenham ultrapassado a mesma quantidade relativa de tempo, ambos não obtiveram a mesma margem de contribuição. O Projeto 1 e o Projeto 3 apresentaram um prejuízo menor quando comparados ao prejuízo apresentado pelo Projeto 2. Isso deve-se ao fato de que nos Projetos 1 e 3 também houveram mudanças que compreendiam em acréscimos de funcionalidades, as quais eram financeiramente arcadas pelo o cliente. No Projeto 2 houveram apenas mudanças de comportamento das funcionalidades já desenvolvidas, o que aos olhos do cliente representavam uma falha de levantamento de requisitos da empresa em estudo, devendo ser por ela financeiramente arcados.

3.4.3. Etapa 3 - Projeto de transição do GTP para o GAP

Diante dos resultados obtidos, a mudança para o GAP fazia-se necessária. Contudo, a estrutura da empresa não estava pronta para atendê-lo, dessa forma, era preciso adaptar o seu time, os processos da empresa e o cliente por ela atendido à essa nova realidade de gerenciamento de projetos.

Para o período de transição, era esperado que houvesse uma baixa no volume de projetos por mês, sendo necessário que recursos extras fossem alocados para garantir a operações da empresa enquanto esta adaptava-se ao Gerenciamento Ágil de Projetos. A transição iniciou-se no começo de abril e estendeu-se até meados de agosto, quando então a empresa passou a recuperar o seu volume de projetos por mês.

O primeiramente a empresa buscou entender como o novo modelo deveria ocorrer considerando a realidade do seu cliente, uma vez que seu objetivo principal é atendê-lo. Por mais que Gerenciamento Ágil possua inúmeros benefícios, principalmente ao cliente, uma vez que permite uma maior aproximação desse ao projeto e garante entregas funcionais contínuas, no mercado, principalmente o mercado cearense, poucas são as empresas encontradas que trabalham seguindo tal gerenciamento, dessa forma, também são poucos os clientes que o conhecem e o aceitam por completo, principalmente no que tange à não definição desde o início do projeto de quais serão todas as funcionalidades, com datas e orçamentos exatos a serem cumpridos.

Frente aos clientes que já possuíam um Contrato de Prestação de Serviços acordado com a empresa, esta prosseguiu com seus projetos sob a abordagem do GTP até que estes fossem concluídos. Após a conclusão de tais projetos, aqueles clientes que desejaram renovar o Contrato com a empresa precisaram aderir ao GAP.

Por mais que a definição de funcionalidades, prazos e orçamentos exatos tenham se mostrado uma ilusão, uma vez que o cliente não possui clareza sobre o produto desde o início do projeto, muitos clientes não sentiam-se confortáveis em trabalhar com uma empresa sem tê-los definido. A não-certeza de quanto gastará exatamente para obter seu produto, nem qual o tempo exato para o seu desenvolvimento provoca inseguranças.

Dessa forma, a empresa precisou adaptar seus processos de gerenciamento de projetos para que, ao mesmo tempo em que o Gerenciamento Ágil fosse seguido, a insegurança provocada pelo o mesmo pudesse ser atenuada com previsões de prazo, orçamento e escopo, ambas passíveis de sofrerem alterações, porém suficientes para dar prosseguimento às atividades.

Além de adaptar seus processos, a empresa necessitou alterar seu público-alvo, que antes possuía um poder aquisitivo que o impossibilitava de arcar com as incertezas financeiras, por mais que estas fossem atenuadas com as previsões. Foi então necessário buscar um público de maior poder aquisitivo, e para isso, a empresa precisou expandir sua atuação para atender clientes de demais estados, e que de preferência já possuíssem conhecimento sobre o Gerenciamento Ágil.

Para a aquisição de novos projetos, a empresa focou seus recursos de marketing, fazendo o uso principalmente de mídias sociais e a geração de conteúdo sobre negócios em seu *blog* para captar clientes de outros estados que pudesse apresentar uma maior familiaridade com o GAP, visto que esse ainda não era bem disseminado no Ceará.

Por fim, a equipe precisou ser adaptada para se auto-gerenciar, não sendo mais necessária a figura de um Gerente de Projeto para controlar se a execução está ou não conforme o planejado, além de todos, inclusive os programadores, precisarem aprender a comunicar-se com o cliente.

Contudo, o maior desafio perante à equipe era a adaptação necessária ao seu pensamento. Como anteriormente mudanças representavam custos há mais e trabalho não planejado, estas eram vistas como algo indesejado. Era preciso agora que estas fossem vistas como algo importante e necessário a uma boa execução do projeto.

3.4.4. Etapa 4 - Descrição da adoção do GAP

Assim como ocorria com o Gerenciamento Tradicional, em um primeiro momento a pessoa interessada em desenvolver sua plataforma, esteja ela representando uma pessoa física ou jurídica, entra em contato com a empresa para realizar a Reunião de *Briefing*. Embora o nome para o evento seja o mesmo anteriormente adotado, o seu objetivo já não mais é responder às cinco principais perguntas listadas no item 4.2.1, e sim responder à uma única principal pergunta: qual o objetivo da plataforma?

Como exemplo de objetivos já definidos pelos clientes da empresa que podem ser citados, tem-se:

- Ajudar pessoas doentes a encontrarem médicos especializados para atendê-las.
- Comercializar um produto novo no mercado para um outro país por meio da internet.
- Ajudar empresas de construção civil a encontrar máquinas pesadas para aluguel
- Auxiliar médicos cirurgiões no acompanhamento à distância do pós-operatório de seus pacientes.

Entendendo qual o objetivo da plataforma, a empresa deve focar-se não mais em desenvolver as funcionalidades definidas, mas sim em atingir o objetivo, independente de quais itens deverão ser desenvolvidos para isso.

Com o objetivo definido, é determinado um Designer para ser o responsável pelo o projeto, devendo este acompanhá-lo até o fim. Nesse momento, seu papel não mais é definir todo o fluxo de informações da plataforma, bem como os itens que cada tela do sistema deverá conter. Invés disso, são criados os chamados *wireframes*, telas simplificadas, passíveis de alterações, apenas com a finalidade de permitir ao time uma visão macro, e não detalhada, de como a plataforma deverá operar. As figuras 12 e 13 representam os *wireframes* de uma área para solicitação de orçamento para o alugem de um determinado item.

Figura 12 – Wireframe da solicitação de orçamento

12:30

← Solicitar Orçamento

Nome da Empresa Loren Ipsun

Prazo 20.07

Número de serviço

UN. Geração do Contrato XXXXXXXXXXXX

UN. Prestação de Serviços XXXXXXXXXXXX

Local

Nome do local

Endereço detalhado do local

Resumo Geral

Tipo de contratação

Confira as informações antes de enviar. ENVIAR →

Fonte: Empresa.

Figura 13 – Envio da solicitação de orçamento

12:30

Orçamento Enviado

Sucesso!

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed

NOVO ORÇAMENTO INÍCIO

Fonte: Empresa

As telas desenhadas para a solicitação do orçamento não possuem o fluxo de navegação bem definido, como ocorria com a figura 11, na qual já estava determinado qual o fluxo de navegação que levaria a cada página e para onde cada botão presente na tela deveria levar o usuário. Invés disso, as Figuras 12 e 13 representam os itens que irão compor a tela, mas o caminho a ser percorrido pelo o usuário para acessá-las ou para quais telas do sistema estas devem levar não estão definidos, uma vez que tais fluxos serão definidos de forma incremental.

Observa-se que os *wireframes* não possuem cores, elementos gráficos e não representam o fluxo de informações do sistema, uma vez que estes serão definidos em etapa posterior.

Definidos os *wireframes*, entende-se quais serão as macroentregas, denominadas *Milestones* necessárias para que o produto atinja seu objetivo. A partir das macroentregas, são escolhidas quais as tecnologias a serem utilizadas no projeto e quem irá compor o *time SCRUM* (com exceção do PO que será escolhido pelo o cliente após a assinatura do Contrato com a empresa). O time escolhido então reúne-se para definir qual o tempo estimado, em meses, para a entrega das *Milestones*, contudo sem maiores detalhamentos sobre quais as funcionalidades que cada *Milestone* deverá abranger e quantas horas serão necessárias para cada uma delas, como o praticado anteriormente. Em fase posterior, as *Milestones* serão divididas em *Sprints* de duas semanas para sua entrega.

A partir da quantidade de tempo estimado, é calculado a estimativa de valor a ser desembolsado pelo o cliente, contudo, tal estimativa é levantada apenas para fins de planejamento financeiro do cliente, podendo ser alterada conforme o projeto for sendo executado e não constando em Contrato de Prestação de Serviço. O cliente não mais contrata à empresa o produto completo nos termos de prazo e orçamento definidos, e sim semanas de serviço desta, sendo as funcionalidades a serem desenvolvidas definidas entre as partes conforme o projeto for sendo executado.

Estando o cliente de acordo com as estimativas apresentadas, é então assinado o Contrato de Prestação de Serviços, com vigência média de cinco meses e o *Ciclo Scrum* inicia-se.

Logo após a assinatura do Contrato entre as partes, o cliente deve escolher o PO. Em muitos casos, o próprio cliente assume esse papel em um projeto, porém, quando este é demasiado ocupado, a própria empresa pede que seja outra pessoa de sua confiança a executar tal função. O PO precisa está disponível para sempre responder à solicitações dos demais membros do *time Scrum*, comunicar-se diariamente e interagir com o produto semanalmente.

Após escolhido o PO, os dias que antecedem à execução são dedicados ao treinamento do *time Scrum*. A empresa disponibiliza ao PO o chamado *Guia do Cliente*, que contém os seus deveres e direitos dentro do projeto, as ferramentas que serão utilizadas para a comunicação entre o time, a frequência em que as reuniões irão ocorrer e como serão definidas e entregues as *Milestones* e as *Sprints* que compõem uma *Milestone*. As Figuras 14 e 15 ilustram duas páginas do *Guia do Cliente*.

Figura 14 – Índice do Guia do Cliente



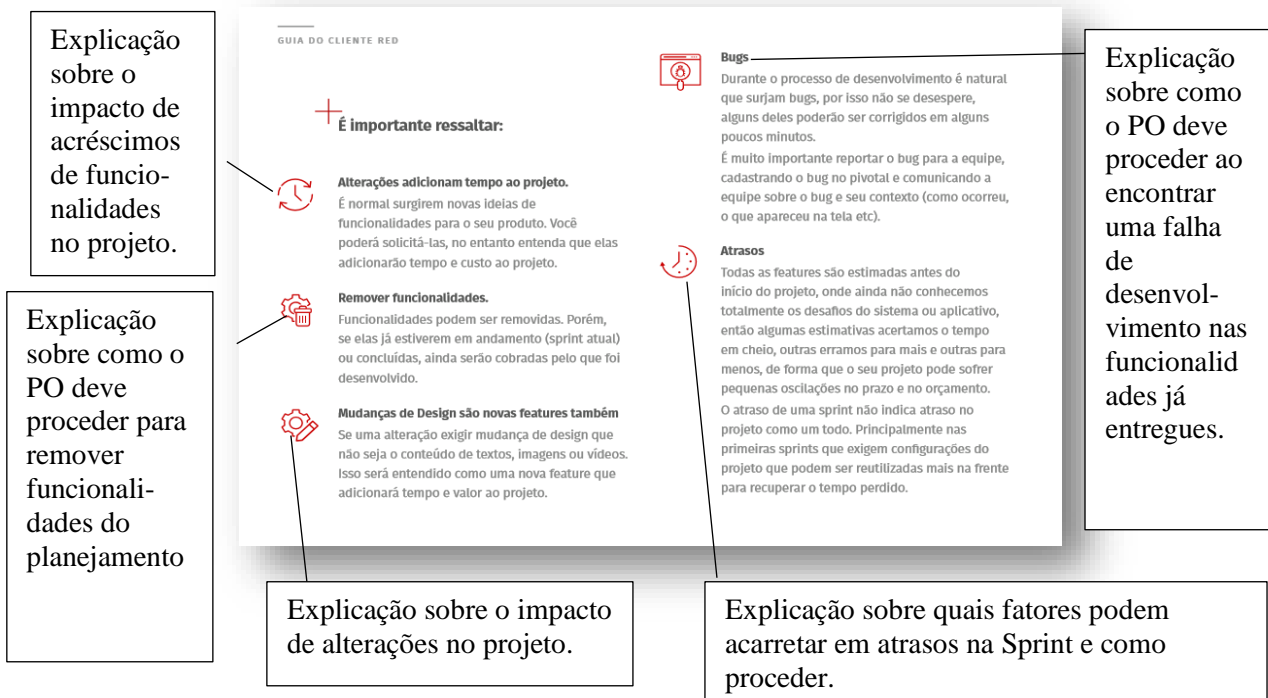
The image shows a dark-themed index page for a 'Guia do Cliente RED'. The title 'Índice' is written vertically on the left. The page lists three main sections, each with a red plus sign icon to its left and a red page number to its right. The sections are: '01 Comunicação' (page 4), '02 Funcionalidades' (page 12), and '03 Perguntas Frequentes' (page 28). The background features a subtle geometric pattern of overlapping triangles.

01	Comunicação	4
02	Funcionalidades	12
03	Perguntas Frequentes	28

Fonte: Empresa.

A Figura 14 apresenta o Índice do Manual, contemplando as três principais áreas por ele abrangidas: a área de comunicação, com a periodicidade em que esta ocorre as ferramentas utilizadas para isso; a área de funcionalidades, explicitando como estas serão desenvolvidas e uma área de perguntas frequentes.

Figura 15 – Funcionalidades no Guia do Cliente.



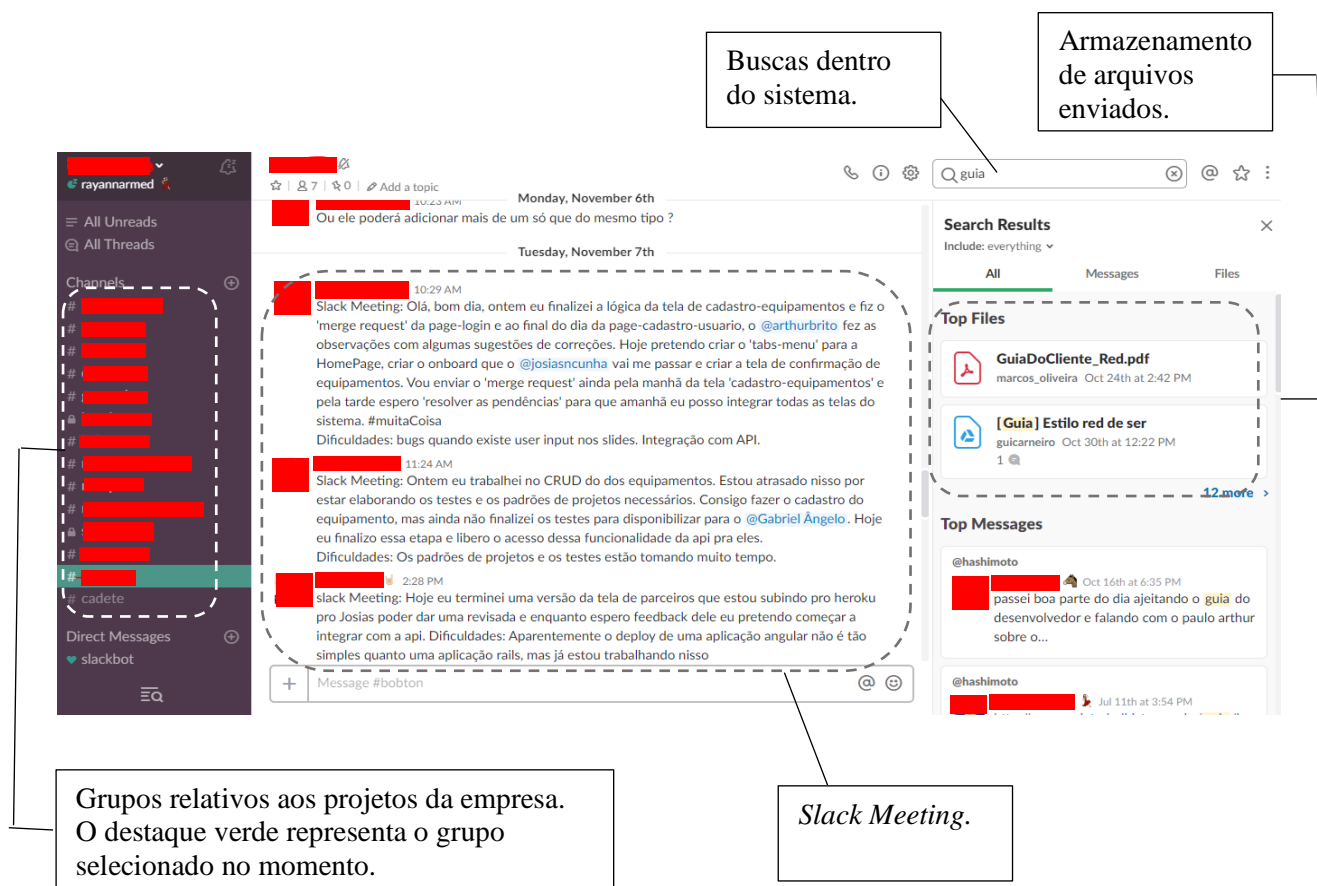
Fonte. Adaptado da Empresa.

A Figura 15 ilustra a área do Manual responsável por orientar o PO durante os procedimentos de desenvolvimento das funcionalidades contidas na Sprint, abrangendo 5 tópicos: acréscimos e funcionalidades, remoção de funcionalidades, alteração no Design criado, reporte de erros e atrasos.

Durante o treinamento é também ensinado ao PO ao utilizar as principais ferramentas de comunicação para o projeto adotadas pela a empresa: *Slack* e *Trello*.

O *Slack* trata-se de uma ferramenta de comunicação empresarial, na qual dentro da conta de uma empresa, diversos times podem ser associados. Pode ser acessada por computadores, tablets ou smartphones e foi escolhida pela a empresa por possuir integrações com demais ferramentas utilizadas, até mesmo ferramentas de programação, assim, caso um erro ocorra com código da plataforma ou com o seu servidor, tal não-conformidade é automaticamente avisada pelo *Slack* e todos do time tornam-se cientes do ocorrido, para então corrigi-lo em um tempo hábil. O *Slack* também permite que, mesmo estando fisicamente separados, todo o *Time Scrum* compartilhe o mesmo ambiente virtual de trabalho. A Figura 16 ilustra a ferramenta.

Figura 16 – Utilização do Slack



Fonte: adaptado da Empresa.

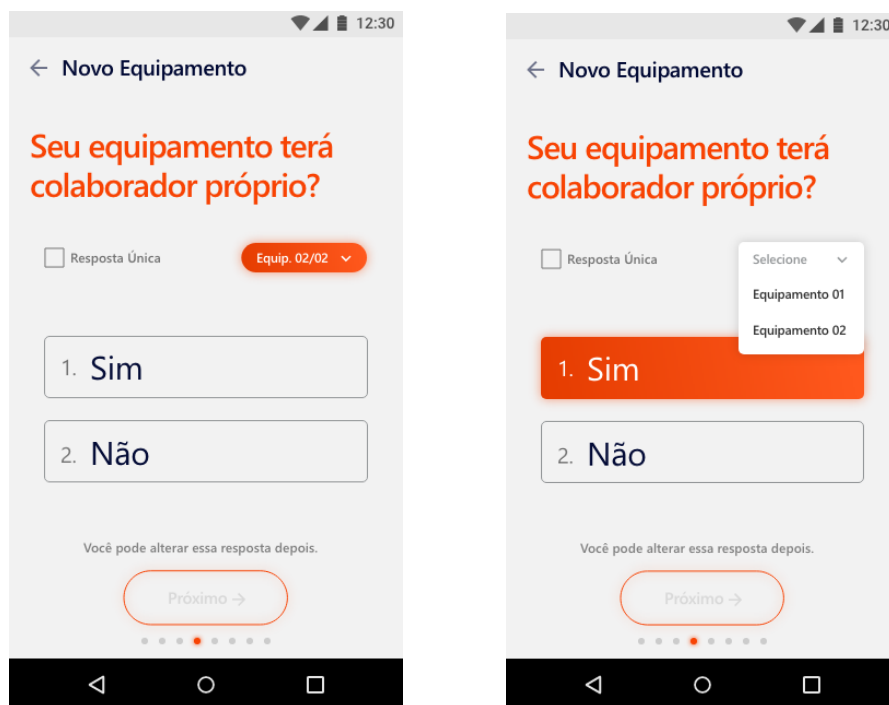
O *Trello* é a principal ferramenta para controle e monitoramento da execução do projeto, permitindo que esse possa ser facilmente visualizado e atualizado por todo o *Time Scrum*. *Trello* é um sistema para gerenciamento de projetos que pode ser acessado pelo computador, tablets e smartphones, e permite que etapas do Projeto possam ser personalizadas de acordo com as necessidades deste. Tal ferramenta foi escolhida por ser, dentre as disponíveis no mercado, a que melhor adequava-se ao controle tanto por parte da empresa quanto pelo o cliente, abrangendo todo o *Time Scrum*. Outras ferramentas, algumas até bem mais completas, estão disponíveis no mercado, contudo, estas são demasiadas técnicas, focadas em controle apenas por parte dos programadores, não sendo viáveis ao uso do cliente.

Estando o time preparado, é definida a prioridade de cada *Milestone* do projeto, sendo a ordem de prioridade a ordem em que estas serão desenvolvidas.

Para a *Milestone* de maior prioridade, é definido um conjunto de funcionalidades que esta deve agregar tais que estas possam ser desenvolvidas em um período de quinze dias, e, juntas, possuam um valor funcional que possa ser testado pelo cliente. A tal conjunto dar-se o nome de *Sprint*.

Nos dois primeiros dias após a definição da *Sprint*, o Designer responsável pelo o projeto deve ajustar os *wireframes* anteriormente elaborados para conter as informações detalhadas de quais itens irão compor as telas do sistema relativas às funcionalidades a serem desenvolvidas, conforme Figura 17.

Figura 17 – Telas detalhadas



Fonte: Empresa.

No terceiro dia de *Sprint* os programadores iniciam o desenvolvimento, respeitando o desenho elaborado pelos Designers, sendo os testes automáticos criados no código paralelamente. Ao final do décimo terceiro dia, o Designer responsável inspeciona as funcionalidades desenvolvidas a fim de verificar se estas estão de acordo com os desenhos elaborados. São também realizados os testes manuais para que, ao fim do décimo quinto dia a *Sprint* possa ser entregue ao cliente.

Durante todo esse período, a comunicação com o PO mantém-se constante, a fim de garantir que o que está sendo desenvolvido atende às suas necessidades. No sétimo dia o Scrum Master reúne-se com ele a fim de alinhar o desenvolvido até o momento, evitando que não conformidades só sejam detectadas ao décimo quinto dia.

A fim de não gerar distúrbio ao andamento do projeto, durante o desenvolvimento de uma *Sprint*, novas funcionalidades não podem ser adicionadas e funcionalidades já existentes não podem retiradas.

O desenvolvimento descrito pode ser visualizado a qualquer momento por meio do *Trello*, o qual é organizado em três principais áreas conforme Figuras 18, 19 e 20.

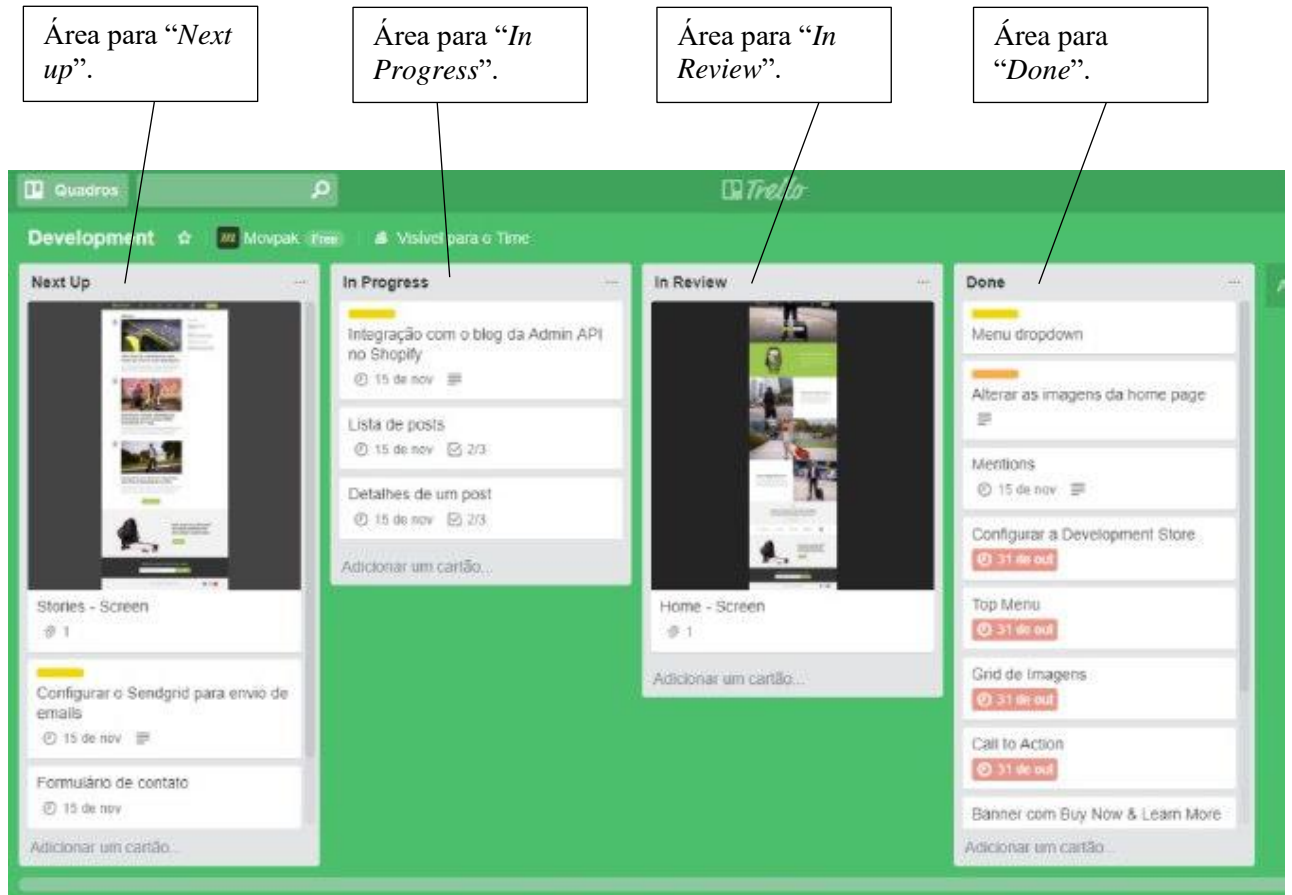
Figura 18 – *Trello Backlog*



Fonte: adaptado da Empresa.

A primeira área denomina-se “*Backlog*” e contém atividades futuras a serem executadas para o projeto mas que não encontram-se definidas para a *Sprint* atual, localizadas em “*Inbox*” e ideias de novas atividades sugeridas pelo o PO, localizadas em “*Discovery*”, conforme ilustrado na Figura 18.

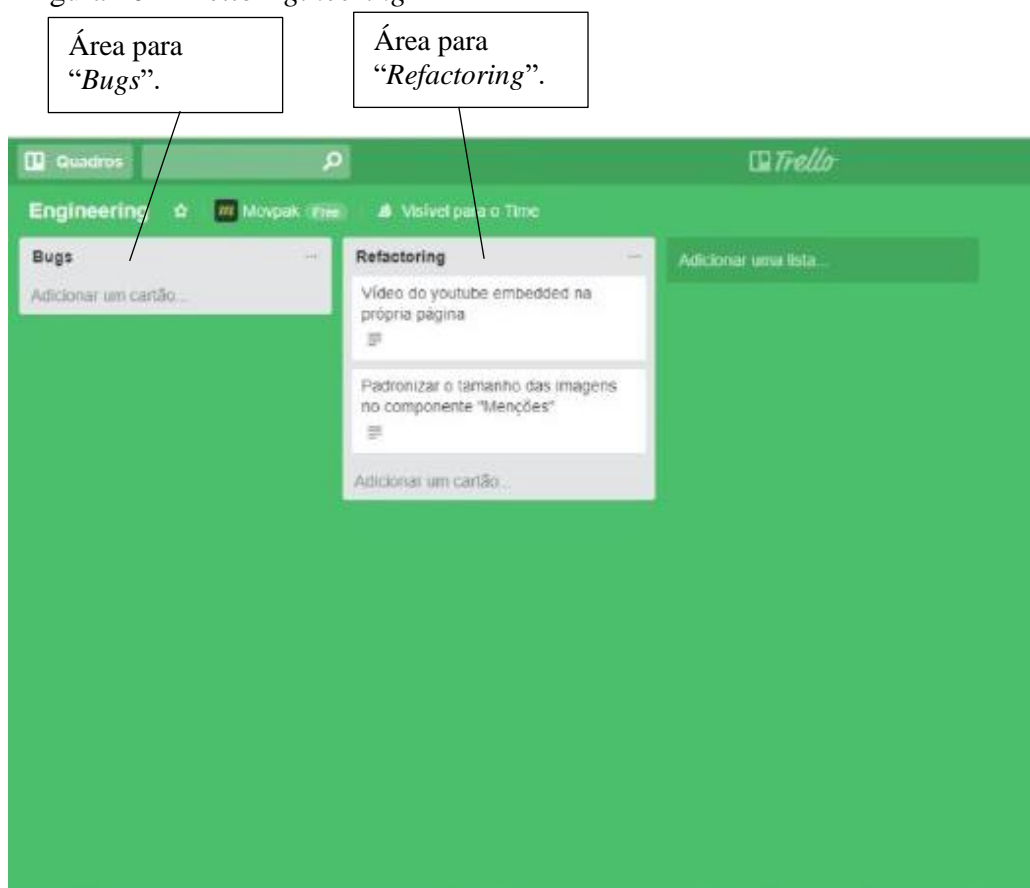
Figura 19 – *Trello Development*



Fonte: adaptado da Empresa.

A segunda área, denominada “*Development*” está ilustrada na Figura 19 e contém os itens relativos à *Sprint* atual. As funcionalidades em desenvolvimento podem ser localizadas em “*In Progress*”, as próximas a serem desenvolvidas em “*Next Up*”, as em inspeção do Designer e revisão manual encontram-se em “*In Review*” e por fim, as concluídas constam na aba “*Done*”.

Figura 20 – *Trello Engineering*



Fonte: adaptado da Empresa.

A terceira área denomina-se “Engineering” e está ilustrada na Figura 20. Não conformidades encontradas a qualquer momento no sistema devem ser relatadas na aba “Bugs” e oportunidades de melhoria a serem aplicadas no código de programação já escrito devem constar em “Refactoring”.

Ao ser finalizada uma Sprint, outra inicia-se, repetindo o ciclo.

3.4.5. Etapa 5 - Resultados obtidos com o GAP

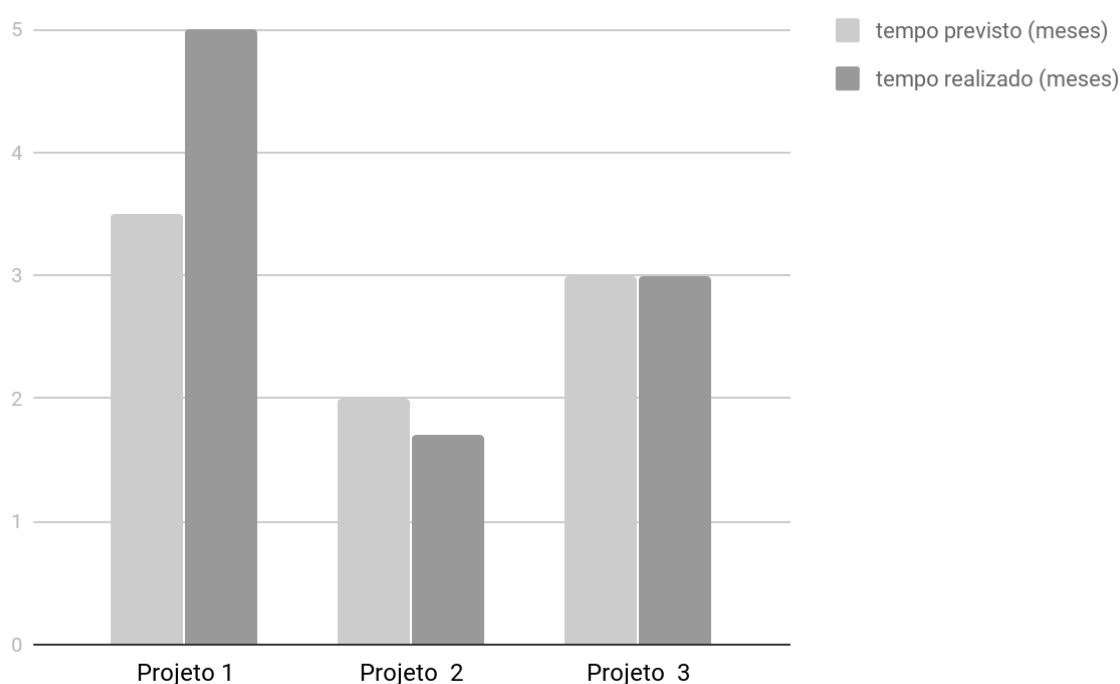
Para a análise dos resultados obtidos com os projetos realizados, a pesquisa enfrenta a mesma limitação encontrada ao analisar os projetos executados por meio do Gerenciamento Tradicional, na qual alguns clientes solicitam a assinatura de um “Termo de Sigilo” para os seus projetos, não podendo os seus dados serem aqui utilizados.

Foram analisados três projetos geridos pelo o *Scrum*, um relativo ao desenvolvimento de uma plataforma web, outro relativo ao desenvolvimento de uma plataforma mobile e o terceiro para uma plataforma com uma área web e uma área mobile.

Cronograma

No que tange aos aspectos de prazos dos projetos, por mais este fosse apenas estimado, e não precisamente definido, verificou-se que o tempo real demandado para a execução não ultrapassou em demasia o estimado, uma vez que, com a presença constante do PO, não haviam mais alterações a serem realizados ao final do projeto, conforme verificado no Gráfico 5.

Gráfico 5 - Tempo de desenvolvimento estimado *versus* tempo de desenvolvimento realizado no GAP



Fonte: Autora.

O Projeto 1, o qual diz respeito a um projeto de desenvolvimento de uma aplicação mobile e uma plataforma web, extrapolou o tempo previsto em 1,5 meses. Destaca-se que este foi o primeiro projeto no qual o Gerenciamento Ágil foi adotado pela a empresa e pela a falta de experiência, falhas de comunicação se fizeram presentes, acarretando no resultado obtido. Dentre estas falhas a empresa cita que a mais prejudicial foi a não-definição de um PO adequado para o trabalho, uma vez que este constantemente fazia-se ausente, não acompanhando a execução do processo em uma periodicidade quinzenal. Com este distanciamento, algumas alterações nos requisitos de escopo e na forma de comportamento das funcionalidades

planejadas foram comunicados ao restantes da Equipe do Projeto apenas após a conclusão do desenvolvimento, provocando retrabalhos e dilatação do tempo de projeto.

Durante o Projeto 2, o PO manteve-se presente e não houveram alterações a serem reportadas apenas após a conclusão das funcionalidades, o que evitou retrabalho e auxiliou no controle do tempo de desenvolvimento. Por mérito da equipe técnica de programadores, a qual dominava bastante a tecnologia em uso, algumas funcionalidades foram desenvolvidas mais rapidamente que o previsto inicialmente, diminuindo o tempo de desenvolvimento do projeto.

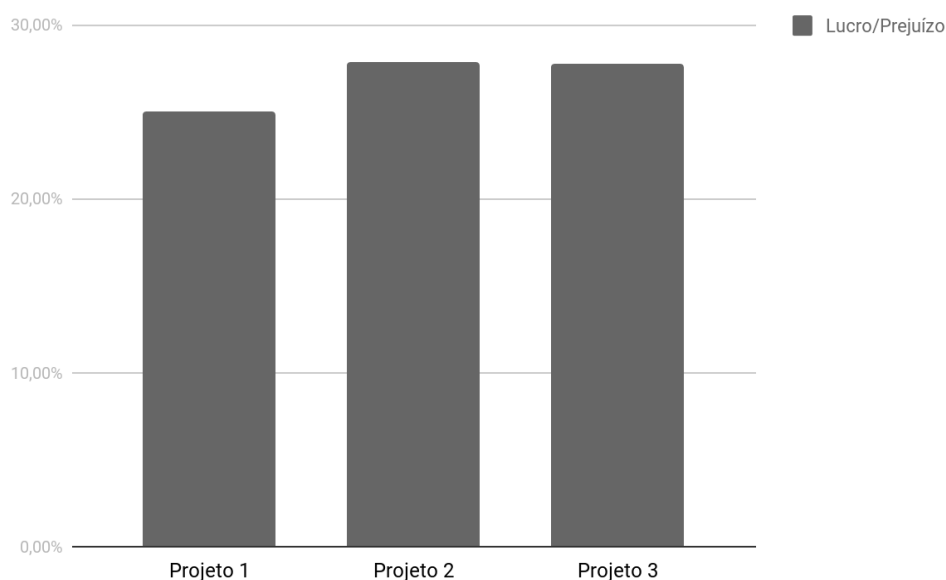
Já durante o projeto 3, o qual tratava-se de uma plataforma web, o PO também manteve-se presente, acompanhando frequentemente a execução por meio do *Trello* e aprovando as entregas das *Sprints*. Nesse caso, a equipe de programadores precisou lidar com uma tecnologia mais recente e pouco conhecida, o que a fez tomar o exato tempo inicialmente estimado para a entrega do projeto.

Nos casos 2 e 3, o fato do tempo ter ter ultrapassado a hipótese inicial não significa a inexistência de alterações nos requisitos do projeto, mas sim que estes foram sinalizados à equipe de desenvolvimento em tempo hábil, evitando que fosse desenvolvido algo não desejado, o que anteriormente provocava o retrabalho e a dilatação do tempo dispendido para a concretização do projeto.

Orçamento

Já do ponto de vista financeiro, a empresa parou de sofrer prejuízos e iniciou o seu período de lucratividade, conforme apresentado no Gráfico 6:

Gráfico 6 - Margem de contribuição dos projetos sob a abordagem do GAP.



Fonte: Autora.

O projeto 1 obteve uma margem de contribuição de 25% sobre o seu faturamento, muito embora tenham ocorrido retrabalhos e alterações em entregas já anteriormente consideradas como concluídas pela a equipe do projeto, não foi o suficiente para que o projeto trouxesse prejuízo para a empresa, como ocorria com GTP.

O Projeto 2 obteve uma margem de 27,84%, seguido do Projeto 3, o qual obteve uma margem de 27,74%.

3.4.6. Etapa 6 - Comparação dos resultados obtidos com o GTP e o GAP

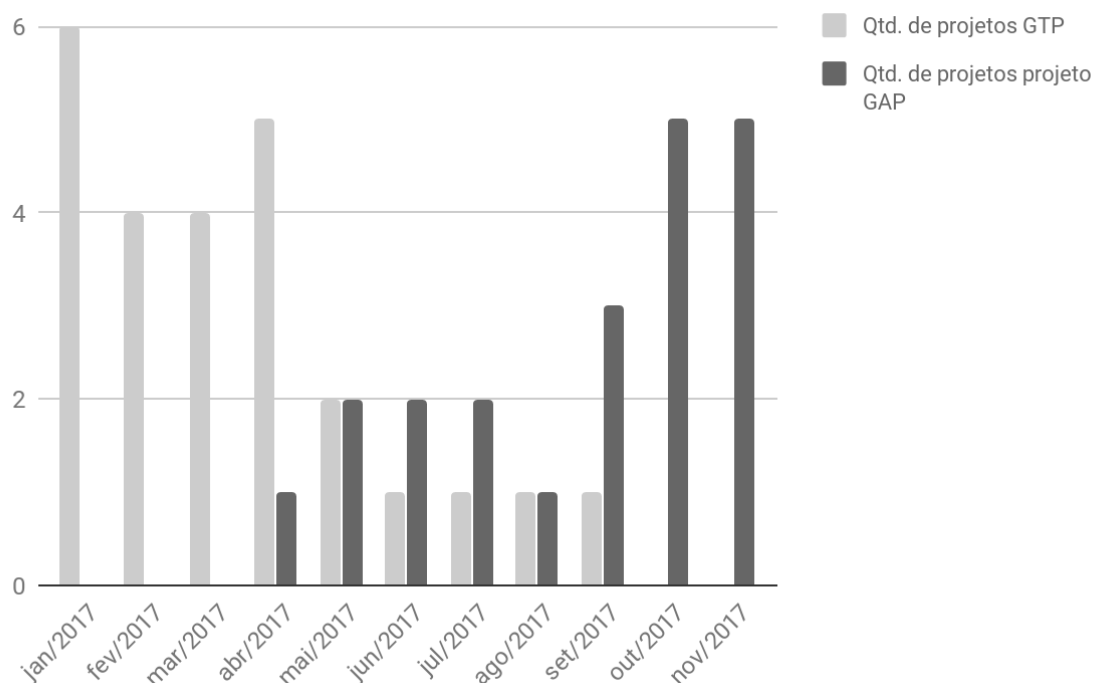
Comparando-se os resultados obtidos com a aplicação do Gerenciamento Tradicional de Projetos com os resultados obtidos com a aplicação do Gerenciamento Ágil de Projetos, foram obtidos os resultados relativos aos aspectos financeiros, ao cumprimento de cronogramas e ao volume de projetos mensais da empresa em estudo.

Cientes

A mudança para o Gerenciamento Ágil de Projetos exigiu que a empresa passasse a atingir um outro público, uma vez que o atual não estava preparado para aceitá-lo. Essa busca por um novo perfil de cliente culminou em uma baixa temporária no volume de

projetos por mês executados pela a mesma, seguido de um período de recuperação de tal volume, conforme exposto no gráfico 7.

Gráfico 7 – Volume de projetos ao mês



Fonte: Autora.

O Gerenciamento Ágil começou a ser adotado em abril/2017, quando a empresa possuía seis projetos em desenvolvimento. Até julho/2017, projetos que abrangiam o Gerenciamento Tradicional ainda estavam sendo finalizados, sendo assim, em agosto/2017 a empresa experimentou trabalhar apenas com o Gerenciamento Ágil, uma vez que os demais projetos foram finalizados, chegando a trabalhar apenas dois projetos durante o mês. Contudo, a partir de setembro o volume voltou a crescer.

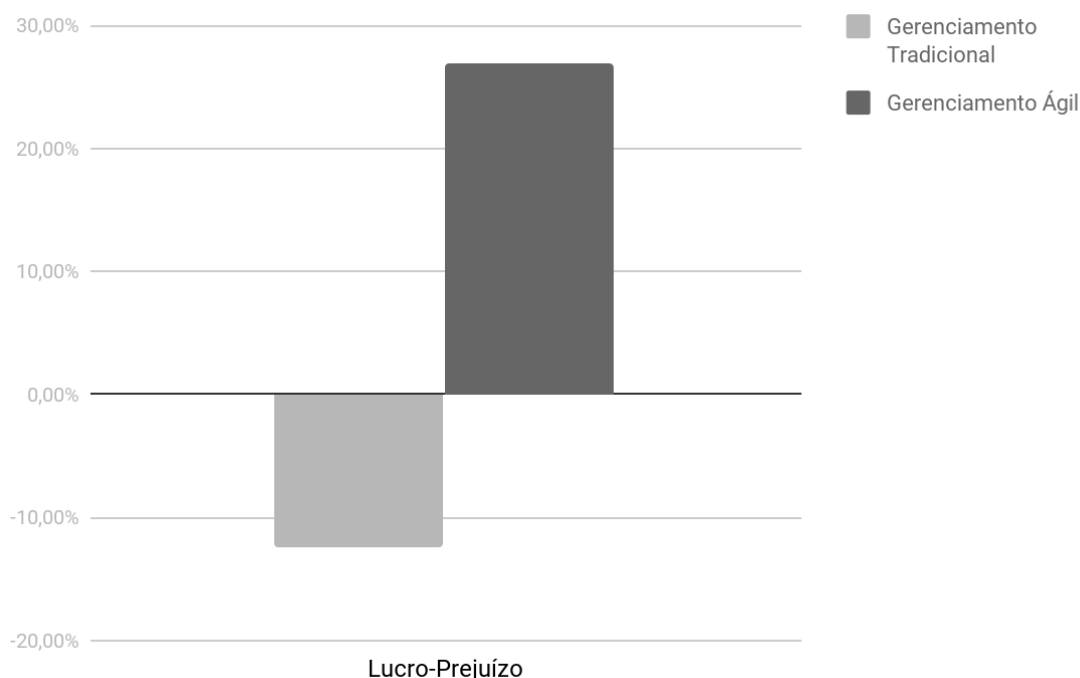
A adoção do modelo também implicou no atendimento de clientes de outras cidades, e não mais apenas Fortaleza. As demais cidades atendidas até a realização do presente trabalho são Belo Horizonte, Recife e Rio de Janeiro.

Resultados Financeiros

Apesar de ter enfrentado um período com um baixo número de projetos ao mês, o que implica em uma diminuição do faturamento mensal, a empresa apresentou uma melhoria em seus resultados financeiros, uma vez que os projetos nos quais o GAP foi adotado passaram

a apresentar uma margem de contribuição positiva, em contraste com as margens de contribuição negativas que os projetos geridos sob a ótica das metodologias tradicionais traziam. Este fenômeno pode ser observado no Gráfico 8

Gráfico 8 – Margem de Contribuição: Gerenciamento Tradicional de Projetos *versus* Gerenciamento Ágil de Projetos



Fonte: Autora.

Sob o Gerenciamento Tradicional, os projetos obtiveram uma margem de contribuição média de -12,40%, e sob o Gerenciamento Ágil de Projetos, passaram a obter um lucro médio de 26,86%, conforme representado.

Observa-se que, sob a ótica o Gerenciamento Tradicional, o Projeto 4 (referente ao projeto para uso na medicina que não sofreu modificações após a fase de desenvolvimento e conseguiu obter lucro para a empresa) obteve sozinho um lucro de 58,09% sob o faturamento do projeto. Contudo, sob a ótica do GAP, os projetos que obtiveram sucesso financeiro atingiram uma margem inferior a esse número, mantendo-se na casa dos 22% sob a faturamento do projeto.

Isso ocorreu devido a um ajuste que a empresa precisou realizar em seu valor-hora para baratear o valor total de seus projetos, como estratégia para manter seus clientes, uma vez que com a mudança da abordagem tradicional para a abordagem ágil, muitos deles

tornaram-se propensos a buscar outras empresas de desenvolvimento que trabalhassem com GTP, pelos motivos já citados no tópico “3.4.3. Projeto de transição do GTP para o GAP”.

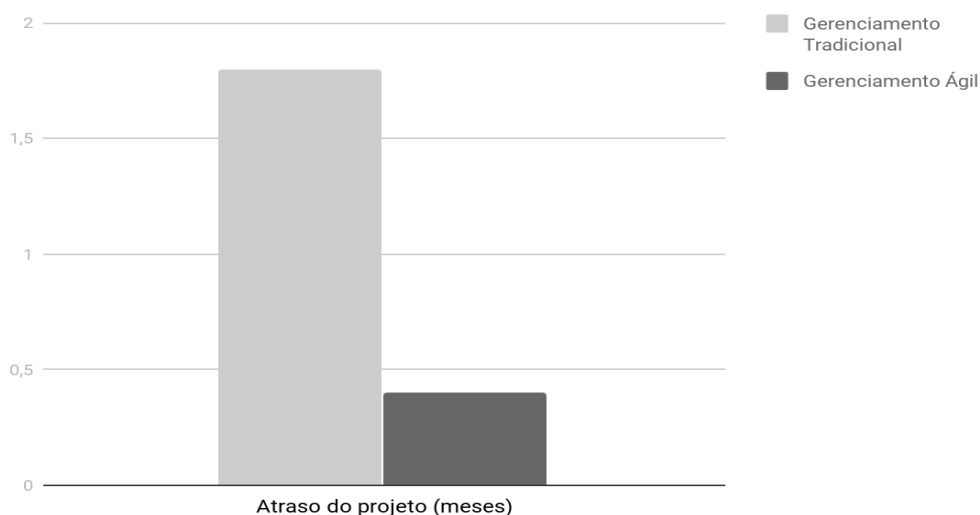
Cronograma

Com o GTP, os projetos executados por a empresa apresentavam uma média de atraso de 1,83 meses.

Embora em GAP o controle sob o tempo de projeto não seja uma prioridade de controle, como explicado no tópico “3.4.3. Projeto de transição do GTP para o GAP” ele continuou sendo necessário, uma vez que os clientes atendidos por a empresa possuíam necessidade de uma estimativa de prazo para sentirem-se confortáveis com o GAP e assim poderem seguir os trabalhos com a empresa.

Com o GAP a média de atraso foi reduzida para 0,4 mês, em contraste à média de atraso de 1,83 meses apresentada pelo GTP e representada no Gráfico 9.

Gráfico 9 – Atraso médio de projetos: GTP *versus* GAP



Fonte: Autora.

Tal redução deveu-se ao fato da figura do PO acompanhar o projeto de perto e o escopo desse manter-se flexível a mudanças, pois dessa forma, as funcionalidades desenvolvidas eram constantemente avaliadas por o cliente e, caso fossem necessárias alterações no comportamento do software que fugiam ao inicialmente planejado, essas eram reportadas à equipe de desenvolvimento antes de serem implementadas, evitando retrabalho e modificações em funcionalidades já construídas.

A fim de avaliar o impacto no tempo de projeto trazido por cada nova atividade realizada por a empresa para gerir seus projetos segundo o Scrum, um questionário foi aplicado aos desenvolvedores da Equipe Interna.

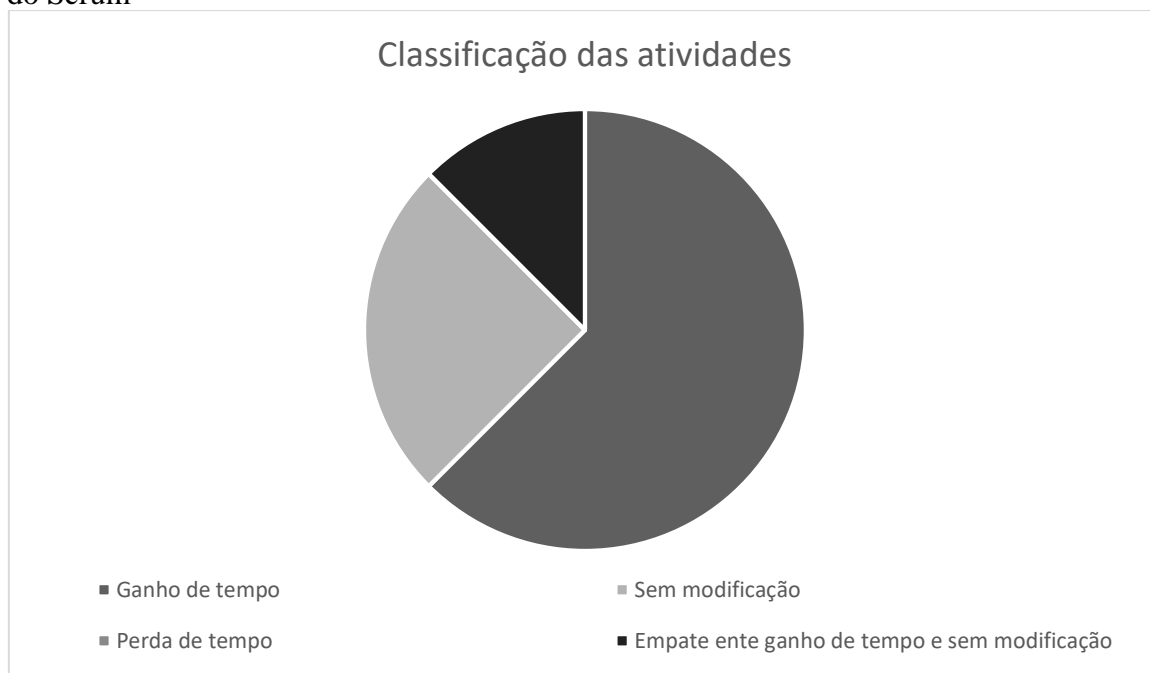
Tal questionário foi adaptado do trabalho de Lopes (2017) e coletou as percepções dos desenvolvedores sobre oito novas atividades, sendo elas: seleção do Product Owner, seleção do Scrum Master, definição de Sprint com duas semanas de duração, criação do backlog do projeto e sua priorização, reunião de planejamento da Sprint, reuniões de Scrum diárias, atualização do *Trello* e reunião de revisão de Sprint.

O questionário completo encontra-se no Apêndice A.

As atividades foram avaliadas seguindo o critério de consumo do tempo nos projetos, tendo como comparação o tempo anteriormente consumido pelo GTP. A classificação variava entre “ganho de tempo”, “perda de tempo” e “sem modificação”.

Quatro dos cinco desenvolvedores responderam ao questionário e como resultado obteve-se que, 5 atividades foram predominantemente classificadas como “ganho de tempo”, 3 foram classificadas como “sem modificação” e nenhuma foi classificada como “perda de tempo”, conforme observado no Gráfico 10.

Gráfico 10 – Percepção dos desenvolvedores sobre o consumo de tempo das atividades do Scrum



Fonte: Autora.

100% dos desenvolvedores que responderam ao questionário classificaram as atividades de seleção do Product Owner, seleção do Scrum Master e criação do backlog do projeto e sua priorização como “ganho de tempo”. As atividades de reunião de planejamento e reunião de revisão de Sprint foram classificadas como “ganho de tempo” por 75% dos desenvolvedores. Para a atividade de definir Sprints com duração de duas semanas houve empate, com 50% dos participantes da pesquisa classificando como “ganho de tempo” e outros 50% classificando como “sem modificação”. Por fim, a atividade de realizar reuniões de Scrum diárias foi classificada como “sem modificação”, bem como a atividade de atualização do *Trello*.

Considerações Finais

O GAP, por exigir aos clientes envolvidos um comportamento perante o projeto diferente do aplicado pelo GTP, principalmente no que tange à necessidade de uma presença constante do PO durante o período de execução e a abertura a modificações no tempo, escopo e orçamento do projeto, não possuiu ampla aceitação entre os clientes que já constavam no portfólio da empresa.

Diante dessa não-aceitação, a empresa passou por um período de redução em seu volume de projetos mensais, como observado no Gráfico 07, período esse que se concentrou nos meses de maio a agosto de 2017.

Contudo, como os projetos nos quais era aplicado o GTP apresentavam um prejuízo médio de -12,40% e os projetos nos quais foi aplicado o GAP apresentaram um lucro médio de 26,86%, essa redução no volume de projetos executados ao mês foi acompanhada por um aumento no lucro da empresa, melhorando os seus resultados financeiros.

Os clientes que não estavam adaptados ao GAP apresentaram insegurança em prosseguir seus trabalhos com a empresa em estudo devido a não possuírem mais um tempo e orçamento exato para seus projetos, uma vez que não estando bem-definidos, esses poderiam ultrapassar o tempo planejado para lançamento do seu produto, bem como ultrapassar o orçamento dedicado ao projeto. Contudo, tal insegurança mostrou-se irreal, uma vez que o tempo de execução dos projetos diminuiu, atingindo um atraso médio de 0,4 mês e, como o valor do projeto é calculado tendo como base o tempo de execução desse, manter-se próximo ao prazo estimado implica que o valor financeiro deste também esteve próximo ao levantado na fase de planejamento.

Dessa forma, os resultados alcançados com o emprego do GAP, quando comparados aos resultados obtidos com o emprego do GTP, foram muito mais satisfatórios à empresa em estudo, garantindo que esta recuperasse sua sustentabilidade.

4. CONCLUSÃO

O presente trabalho cumpriu com o seu objetivo geral de analisar a eficácia do Gerenciamento Ágil de Projetos, por meio do framework SCRUM, no setor de desenvolvimento de uma empresa fornecedora de softwares localizada no Estado do Ceará, obtendo como resultados uma maior lucratividade e uma diminuição no atraso das entregas dos projetos da empresa quando esta adota o GAP em oposição ao GTP.

Destaca-se que os 4 objetivos específicos listados no item 1.2.2 também foram alcançados. O primeiro objetivo foi atingido por meio de um estudo de caso no qual foram levantadas as informações relativas a como ocorria a aplicação do GTP na empresa em estudo. Analisando 4 projetos gerenciados sob a abordagem tradicional, foram coletados os resultados dos indicadores de lucratividade e cumprimento de cronograma sob a ótica do GTP, constatando que a empresa apresentava altos prejuízos e atrasos de entregas.

O segundo objetivo específico foi atingido com êxito por meio de um estudo de caso no qual coletou-se informações capazes de descrever com detalhes os processos adotados na empresa para o emprego do GAP, incluindo as ferramentas utilizadas e como deu-se o processo de transição da abordagem tradicional para a abordagem ágil.

O terceiro objetivo específico foi atingido com sucesso por meio da coleta dos resultados relativos aos indicadores de lucratividade dos projetos e cumprimento do cronograma de entregas.

Por fim, o quarto objetivo específico foi alcançado ao comparar os resultados dos indicadores obtidos com a aplicação do GTP e do GAP. Comparou-se também o impacto de cada uma das abordagens no volume de projetos que a empresa executa ao mês e por meio de um questionário analisou-se a percepção dos programadores da empresa a respeito do impacto das atividades do GAP no consumo de tempo dos projetos.

Ao longo da pesquisa, destacou-se a dificuldade da empresa em conseguir adotar técnicas ágeis em um mercado habituado ao gerenciamento tradicional, precisando esta realizar adaptações nos processos sugeridos pelo GAP para que estes se adequassem à sua realidade. Outro ponto a ser superado foi o acesso escasso a informações sobre como implementar o GAP, uma vez que poucas empresas cearenses adotam tal gerenciamento.

Diante das dificuldades encontradas, fazem-se necessários mais trabalhos sobre como o GAP pode ser implementado de forma prática, e não apenas teórica, aplicando o estudo em

demais empresas atuantes no setor de TI, uma vez que o acesso às informações dos resultados obtidos poderá permitir que mais empresas consigam alavancar os resultados de seus indicadores, podendo assim prestar seus serviços de forma cada vez mais sustentável.

REFERÊNCIAS

- ABES – ASSOCIAÇÃO BRASILEIRA DE EMPRESAS DE SOFTWARE. **Mercado brasileiro de software: panorama e tendências**, 2017. Disponível em: <<http://central.abessoftware.com.br/Content/UploadedFiles/Arquivos/Dados%202011/ABES-Publicacao-Mercado-2017.pdf>>. Acesso em: 26 de nov de 2017.
- AUGUSTINE, S. **Managing Agile Projects**. Virginia: Prentice Hall PTR, 2005.
- BECK, K. **Test driven development: by example**. Boston: Addison-Wesley Longman Publishing Co., Inc., 2002.
- BERGER, P, M. **Instanciação de Processos de Software em Ambientes Configurados na Estação TABA**. 2003. 128 f. Dissertação (Mestrado em Engenharia de Sistemas e Computação) – Universidade do Rio de Janeiro, Rio de Janeiro, 2003.
- BERTO, R. M. V, S.; NAKANO, D, N. **A produção científica nos anais do encontro nacional de engenharia de produção: um levantamento de métodos e tipos de pesquisa**, Prod., vol.9, n.2, São Paulo, Jul-Dez. 1999. Disponível em: <<http://www.scielo.br/pdf/prod/v9n2/v9n2a05.pdf>>. Acesso em: 26 de nov de 2017.
- BIEZUS, L. **O gerenciamento de empreendimentos. Um caso concreto: a implantação do Sanegran**. *Revista DAE*, São Paulo nº 119, 1978.
- BOEHM, B; ABTS, C; BROWN, A; CLARK, S, C, B. MADACHY, E, H, R; REIFER, D; STEECE, B. **Software Cost Estimation with Cocomo II**. 1ª ed. New Jersey: Prentice Hall PTR, 2000.
- CHARVAT, J. **Project Management Methodologies: selecting, implementing and supporting methodologies and processes for projects**. New Jersey: John Wiley & Sons, Inc, 2003. 271 p.
- CHIN, G. **Agile Project Management: how to succeed in the face of changing project requirements**. Amacom; New York, 2004.
- CHOW, T. CAO, D, B. **A survey of critical success factors in agile software projects**. *The Journal of Systems and Software*, v. 81, p. 961-971, 2008.
- COHN, M. **Agile Estimating and Planning**. 1ª ed. New Jersey: Prentice Hall, 2006. 330 p.
- CONFORTO, E, C. **Gerenciamento Ágil de Projetos: uma proposta e avaliação de método para gestão de escopo e tempo**. 2009. 304 f. Dissertação (Mestrado em Engenharia de Produção) – Universidade de São Paulo, São Carlos, 2009.
- CROWSTON, K; ANNABI, H; HOWISON, J. **Defining open source software project success, ICIS 2003: Proceedings of the 24th International Conference on Information Systems (Seattle, Wash.)**, 2003.
- CROWSTON, K; ANNABI; HOWISON, J. **Defining open source software project success Conference on Information Systems**, 24ª ed. Seattle, 2003.

DANTAS, M. **Informação e Trabalho no Capitalismo Contemporâneo. Lua Nova: Revista de Cultura e Política**, Rio de Janeiro, nº 60.

DINSMORE, P; CABANIS-BREWING, J. **The AMA Handbook of Project Management**. 2 ed. New York: America Management Association, 2006.

FILHO, D, L, B. **Experiências com desenvolvimento ágil**. 2008. 170 f. Dissertação (Mestrado em Ciências) – Instituto de Matemática e Estatística da Universidade de São Paulo, São Paulo. 2008.

FORTUNE, J., WHITE, D. **Framing of project critical success factors by a systems model. International Journal of Project Management**, v 24, n.1, p. 53–65, 2006.

FUGGETA, A. **Software Process: A Roadmap**, Ireland: In: Proc. of the Future of Software Engineering, ICSE'2000, 2000.

GIL, A, C. **Como elaborar projetos de pesquisa**. São Paulo: Atlas, 1991.

GIL, A, C. **Métodos e técnicas de pesquisa social**. São Paulo: Atlas, 1999.

HIGHSMITH, J. **Agile Project Management: creating innovative product**. Addison-Wesley: Boston, 2004.

JEFFRIES, R; MELNIK, G. **TDD - The Art of Fearless Programming**. IEEE Software, 24(3):24–30, 2007.

JOSLIN R., MÜLLER R. **Relationships between a project management methodology and project success in different project governance contexts**, International Journal of Project Management, v. 33, n. 6, p. 1377-1392, 2015.

KALERMO, J., RISSANEN, J. **Agile software development in theory and practice**. 2002. 188 f. Dissertação (Mestrado em Software Business)- Universidade de Jyväskylä, Finlândia, 2002.

KERZNER, H. **Gestão de projetos: as melhores práticas**. 2 ed. Porto Alegre: Bookman, 2006

KERZNER, H. **Strategic Planning for Project Management using a project management Maturity Model**. Nova York: John Wiley & Sons, 2001.

KERZNER, H. **Gestão de Projetos: as melhores práticas**. 3ª ed. Porto Alegre: Bookman, 2017.

KOZAK-HOLLAND, M. **The History of Project Management**. Ontario: Multi-Media Publications Inc., 2011.

LAKATOS, E, M; MARCONI, M, A. **Fundamentos de metodologia científica**. São Paulo: Atlas, 1993.

LOPES, L, P. **Aplicação da Metodologia Scrum em uma área de Engenharia de Processos de uma Empresa do Varejo**. Rio de Janeiro: UFRJ, 2017.

MARTINS, J, C, C. **Técnicas para Gerenciamento de Projetos de Software**. Rio de Janeiro: Brasport, 2007.

MATIAS-PEREIRA, J. **Manual de Metodologia da Pesquisa Científica (Livro Digital)**. 3a Ed. São Paulo, SP: Editora Atlas, 2012.

MILLER, D. **Slaying the dragons: an agile approach to software development – a management overview**, 2002. Disponível em: <http://ibrarian.net/navon/paper/SLAYING_THE_DRAGONS__AN_AGILE_APPROACH_TO_SOFTWAR.pdf?paperid=2210734>. Acesso em 25 de nov. 2017.

NONAKA, I; TAKEUCHI, H. **The new new product development game**. Harvard Business Review, 64(1):137–146, 1986.

PFLEEGER, S. L.; ATLEE, J. M. **Software engineering: theory and practice**. 3ª ed. Upper Saddle River, NJ: Pearson, 2006.

PRESSMAN, R, S. **Engenharia de software**. 6. ed. São Paulo: McGraw-Hill, 2006.

PRESSMAN, R, S. **Software Engineering: A Practitioner's Approach**. 6ª ed. New York: McGraw Hill, 2005.

PROJECT MANAGEMENT INSTITUTE. **Sobre o PMI**. Disponível em: <<https://brasil.pmi.org/brazil/AboutUS.aspx>>. Acesso em: 25 de novembro de 2017.

PROJECT MANAGEMENT INSTITUTE. **Um Guia do Gerenciamento de Projetos (GUIA PMBOK®)**. 5ª ed. Pennsylvania: Project Management Institute, Inc. 2013.

ROCKHART, J. F., CRESCENZI, A. D. **Engaging top management in information technology**. *Sloan Management Review*, v. 25, p. 3–16, 1984.

RODRIGUES E., **Comparativo PMBOK x Scrum**. 2012. Disponível em: <<http://www.elirodrigues.com/2012/04/02/comparativo-pmbok-x-scrum/>>. Acesso em: 26 de nov de 2017.

ROZENFELD, H.; FORCELLINI, F. A.; AMARAL, D. C.; TOLEDO, J. C.; SILVA, S. L.; ALLIPRANDINI, D. H.; SCALICE, R. K. **Gestão de desenvolvimento de produtos: uma referência para a melhoria do processo**. São Paulo: Saraiva, 2006.

SCHWABER, K. **The Scrum development process**. In Proceedings of the 10th Annual ACM Conference on Object Oriented Programming Systems, Languages, and Applications (OOPSLA'95), Austin, Texas, USA, pages 117–134, 1995.

SCHWABER, K; BEEDLE, M. **Agile Software Development With Scrum**. New Jersey: Prentence Hall, 2002

SCHWABER, K; SUTHERLAND, J. **Um guia definitivo para o Scrum: As regras do jogo**, 2016. Disponível em < <http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-Portuguese-Brazilian.pdf>> Acesso em: 26 de nov de 2017.

SHENHAR, A; DVIR, D. Project Management Research – the challenge and opportunity. Project Management Journal, v. 38, n. 2, p. 93-99, 2007.

SILVA, E. L.; MENEZES, E. M. Metodologia da pesquisa e elaboração da dissertação. 4a ed. Florianópolis: Universidade Federal de Santa Catarina, 2005. Disponível em: . Acessado em: out 2017

SMITH, P, G. Flexible Product Developmente - building agility for changing markets. Jossey-Bass: San Francisco, 2007.

SOMMERVILLE, I. Engenharia de software. 8. ed. Rio de Janeiro: Pearson Addison Wesley, 2007.

SOUMYADIPTA P., SINGH J. Be Agile: Project Development with Scrum framework, Journal of Theoretical and Applied Information Technology, v. 40 n.1, p 105 – 112, 2012.

SUTHERLAND J. Scrum - A arte de fazer o dobro de trabalho na metade do tempo. 1 ed. São Paulo, Basil, LeYa Brasil, 2014.

TAKEUCHI, H; NONAKA, I. The New New Product Development Game. Havard Business Review 64, no 1, p 137 – 146, 1986.

TORRES, L, F. Fundamentos do Gerenciamento de Projetos. 1ª ed. Rio de Janeiro: Elsevier, 2014. 216 p.

VARGAS, R. V. Gerenciamento de projetos: estabelecendo diferenciais competitivos. Rio de Janeiro, Brasport, 2005.

VERAS, M. Gestão Dinâmica de Projetos: LifeCycleCanvas. Rio de Janeiro: Brasport 2016.

WELLS, H. An exploratory examination into the implications of typeagnostic selection and application of project management methodologies (PMMs) for managing and delivering IT/IS projects. *Proceedings IRNOP Conference*, Oslo, p. 1–27, 2013.

WESLEY A. Adaptative Project Framework: managing complexity in the face of uncertainty. 1 ed. Boston EUA. Addison Wesley, 2010.

WYSOCKI, R. Effective Project Management - Traditional, Adaptive, Extreme. 4 ed, Indianapolis: Wiley Publishing, Inc. 2007

XAVIER, C. M. S. Metodologia de gerenciamento de projetos no terceiro setor: Uma estratégia para a condução de projetos. Rio de Janeiro: Brasport, 2008.

APÊNDICE A – QUESTIONÁRIO APLICADO AOS DESENVOLVEDORES PARA COLETA DE PERCEPÇÕES DO IMPACTO DO GAP NO TEMPO DO PROJETO.

A pesquisa realizada com a Equipe de Desenvolvimento Interna é uma adaptação da pesquisa elaborada por Lopes (2017).

O questionário abaixo é composto por 8 perguntas objetivas e duas perguntas subjetivas com o intuito de obter a percepção da equipe a fim de compreender como ocorreu a percepção a respeito da modificação do consumo do tempo a partir da utilização GAP.

Questionário:

Para responder as questões abaixo considere as seguintes escalas:

Consumo do tempo nos projetos

Ganho: Pôde ser observado ganho na velocidade das tarefas dos projetos quando a metodologia foi aplicada.

Sem modificação: Não houveram modificações na velocidade das tarefas dos projetos se comparado a execução sem o Scrum.

Perda: As tarefas dos projetos consumiram mais tempo com a aplicação do Scrum.
Selecione para cada técnica abaixo um dos critérios apresentados anteriormente:

1. Técnica: Seleção de um Product Owner

s() Ganho; () Sem modificação; () Perda;

2. Técnica: Seleção de um Scrum Master

() Ganho; () Sem modificação; () Perda;

3. Técnica: Sprints com duração de 2 semanas

() Ganho; () Sem modificação; () Perda;

4. Técnica: Criação do backlog dos projetos e priorização

() Ganho; () Sem modificação; () Perda;

5. Técnica: Reunião de planejamento da Sprint

Ganho; Sem modificação; Perda;

6.Técnica: Reuniões de Scrum diárias

Ganho; Sem modificação; Perda;

7.Técnica: Quadro de atividades visível (*quadro de acompanhamento no Trello*).

Ganho; Sem modificação; Perda; Técnica não aplicada.

8. Técnica: Reunião de revisão dos Sprints

Ganho; Sem modificação; Perda; Técnica não aplicada.