



Pós-Graduação em Ciência da Computação

“Uma Abordagem Ágil ao Gerenciamento de
Projetos de Software Baseada no PMBOK®
Guide”

Por

Luciana de Queiroz Leal

Dissertação de Mestrado



Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

RECIFE, MARÇO/2008



UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE INFORMÁTICA
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

LUCIANA DE QUEIROZ LEAL

“Uma Abordagem Ágil ao Gerenciamento de Projetos de
Software Baseada no PMBOK® Guide”

*ESTE TRABALHO FOI APRESENTADO À PÓS-GRADUAÇÃO EM
CIÊNCIA DA COMPUTAÇÃO DO CENTRO DE INFORMÁTICA DA
UNIVERSIDADE FEDERAL DE PERNAMBUCO COMO REQUISITO
PARCIAL PARA OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIA DA
COMPUTAÇÃO.*

ORIENTADOR: DR. Hermano Perrelli de Moura

RECIFE, MARÇO/2008

Leal, Luciana de Queiroz
Uma abordagem ágil ao gerenciamento de
projetos de software baseada no PMBOK Guide /
Luciana de Queiroz Leal - Recife : O Autor, 2008.
x, 129 p. : il., fig., tab., quadros

Dissertação (mestrado) – Universidade Federal
de Pernambuco. Cln. Ciência da Computação, 2008.

Inclui bibliografia e apêndices.

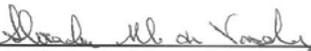
1. Engenharia de software. 2. Gerenciamento de
projetos. I. Título.

005.1

CDD (22. ed.)

MEI2009-055

Dissertação de Mestrado apresentada por **Luciana de Queiroz Leal** à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, sob o título “**Uma Abordagem Ágil ao Gerenciamento de Projetos de Software baseado no PMBOK® Guide**”, orientada pelo **Prof. Hermano Perrelli de Moura** e aprovada pela Banca Examinadora formada pelos professores:



Prof. Alexandre Marcos Lins de Vasconcelos
Centro de Informática / UFPE



Profa. Cristine Martins Gomes de Gusmão
Departamento de Sistemas Computacionais / UPE



Prof. Hermano Perrelli de Moura
Centro de Informática / UFPE

Visto e permitida a impressão.
Recife, 7 de março de 2008.



Prof. FRANCISCO DE ASSIS TENÓRIO DE CARVALHO
Coordenador da Pós-Graduação em Ciência da Computação do
Centro de Informática da Universidade Federal de Pernambuco.

À minha vó Meninha
(*in memoriam*).

Agradecimentos

Primeiramente agradeço a Deus por todas as bênçãos na minha vida e por todas as graças alcançadas, pois sem Ele eu nada seria.

Aos meus pais Luciano, pelo gosto por livros, principalmente os de Matemática, e Vilani, por ser um grande exemplo de mulher batalhadora e perseverante. Agradeço aos dois por sempre estarem me apoiando e me incentivando.

Às minhas irmãs, Vivianne e Lilianne pelo seu apoio e compreensão desde sempre, por me ouvirem e pela sua enorme paciência comigo.

À minha avó Filomena (*in memoriam*) por sempre ter participado da minha vida me apoiando, torcendo, me aconselhando e rezando por mim, mesmo quando ficamos muito distantes.

À minha família pela torcida.

Aos meus amigos de sempre: Anderson, Daniella, Guiguiho, Ramide e Reinaldo pelos conselhos e pelo apoio.

Aos meus amigos de Recife, especialmente a Itapajé, Jeísa, Cláudia, Helô, Sílvia, Gisa, Tarciana, Adriana e Glauco pelo grande apoio em vários momentos desta jornada.

Ao meu orientador Hermano Perrelli, que sempre incentivou meu trabalho, por muitas vezes também ouviu meus probleminhas durante o mestrado, soube entender minha ansiedade e além disso me orientou para a conclusão desta dissertação.

À professora Cristine Gusmão pela ajuda na execução do estudo de caso.

A todos que porventura eu tenha esquecido de mencionar e que contribuíram de maneira direta ou indireta deixo meu muito obrigada.

Sumário

RESUMO	IX
ABSTRACT	X
1 INTRODUÇÃO	11
1.1 MOTIVAÇÃO	11
1.2 OBJETIVO	15
1.3 METODOLOGIA DE TRABALHO	16
1.4 ESTRUTURA DA DISSERTAÇÃO.....	17
2 GERENCIAMENTO ÁGIL DE PROJETOS DE SOFTWARE.....	18
2.1 GERENCIAMENTO DE PROJETOS	18
2.1.1 Projetos e Gerenciamento de Projetos no PMBOK® <i>Guide</i>	20
2.2 GERENCIAMENTO DE PROJETOS DE SOFTWARE	24
2.3 GERENCIAMENTO ÁGIL DE PROJETOS	26
2.4 ABORDAGENS DE GERENCIAMENTO ÁGIL DE PROJETOS	32
2.4.1 <i>Agile Project Management</i>	32
2.4.2 <i>Extreme Project Management</i>	35
2.4.3 Scrum.....	38
2.4.4 <i>Agile Project Management Framework</i>	41
2.5 GERENCIAMENTO ÁGIL DE PROJETOS E O PMBOK® <i>GUIDE</i>	43
2.6 CONSIDERAÇÕES.....	47
3 AGILIUS: UMA ABORDAGEM ÁGIL DE GERENCIAMENTO DE PROJETOS DE SOFTWARE.....	49
3.1 MODELAGEM: O USO DO <i>ECLIPSE PROCESS FRAMEWORK COMPOSER</i>	49
3.2 METODOLOGIA DE DEFINIÇÃO DO AGILIUS.....	51
3.3 VISÃO GERAL DO AGILIUS	52
3.4 GERENCIAMENTO ÁGIL DA INTEGRAÇÃO DO PROJETO.....	54
3.4.1 Iniciar Projeto	56
3.4.2 Planejar Projeto.....	56
3.4.3 Planejar Iterações	59
3.4.4 Liderar Execução do Projeto	60
3.4.5 Monitorar e Controlar Mudanças do Projeto.....	61
3.4.6 Finalizar Iteração.....	62
3.4.7 Finalizar Projeto.....	62
3.5 GERENCIAMENTO ÁGIL DO ESCOPO DO PROJETO.....	63
3.5.1 Planejar Escopo do Projeto.....	65
3.5.2 Criar FBS	67
3.5.3 Verificar Escopo	68
3.5.4 Identificar Mudanças no Escopo	69
3.6 GERENCIAMENTO ÁGIL DO TEMPO DO PROJETO.....	70
3.6.1 Definir e Priorizar Atividades do Projeto.....	71
3.6.2 Elaborar cronograma.....	72
3.6.3 Monitorar e Controlar Cronograma.....	73
3.7 GERENCIAMENTO ÁGIL DAS COMUNICAÇÕES DO PROJETO.....	75

3.7.1	Planejar Comunicações	76
3.7.2	Informar <i>Stakeholders</i>	78
3.7.3	Monitorar Comunicações.....	79
3.8	GERENCIAMENTO ÁGIL DOS RISCOS DO PROJETO.....	80
3.8.1	Identificar Riscos	81
3.8.2	Analisar Riscos.....	82
3.8.3	Planejar Resposta aos Riscos.....	83
3.8.4	Controlar e Monitorar Riscos.....	84
3.9	GERENCIAMENTO ÁGIL DOS CUSTOS DO PROJETO.....	85
3.9.1	Planejar Custos	86
3.9.2	Controlar e Monitorar Custos	87
3.10	CONSIDERAÇÕES.....	88
4	AValiação DO AGILIUS	90
4.1	DEFINIÇÃO DOS OBJETIVOS	90
4.1.1	Objetivo Global.....	90
4.1.2	Objetivo da Medição	91
4.1.3	Objetivo do Estudo Experimental.....	92
4.1.4	Questões	92
4.2	PLANEJAMENTO.....	93
4.2.1	Definição das Hipóteses	94
4.2.2	Descrição da Instrumentação	96
4.2.3	Seleção do Contexto	97
4.2.4	Seleção dos Indivíduos.....	97
4.2.5	Variáveis	97
4.2.6	Análise Qualitativa.....	99
4.2.7	Validade.....	99
4.3	OPERAÇÃO	101
4.3.1	Execução do Estudo.....	101
4.3.2	Resultado do Estudo	101
4.4	ANÁLISE E INTERPRETAÇÃO DOS RESULTADOS	104
4.4.1	Validação dos Dados	104
4.4.2	Estatística Descritiva.....	104
4.4.3	Aplicação do Teste Estatístico Chi-2	107
4.4.4	Análise Quantitativa	110
4.4.5	Análise Qualitativa.....	110
4.4.6	Verificação das Hipóteses.....	111
4.5	CONSIDERAÇÕES FINAIS	112
5	CONCLUSÕES	114
5.1	TRABALHOS RELACIONADOS.....	115
5.2	TRABALHOS FUTUROS	116
	REFERÊNCIAS BIBLIOGRÁFICAS	119
	APÊNDICE A – INTERAÇÕES ENTRE OS PROCESSOS DO AGILIUS.....	123
	APÊNDICE B - QUESTIONÁRIOS	125

Lista de Figuras

Figura 2.1 Razões para adotar metodologias ágeis.....	28
Figura 2.2 Nível de Melhoria na Implementação de Práticas Ágeis.....	28
Figura 2.3 A metodologia Scrum.....	40
Figura 2.4 <i>Agile Project Management Framework</i>	42
Figura 3.1 Grupo de processos do Agilius.....	52
Figura 3.2 Gerenciamento Ágil da Integração.....	55
Figura 3.3 Gerenciamento Ágil do Escopo	64
Figura 3.4 Gerenciamento Ágil do Tempo	71
Figura 3.5 Gerenciamento Ágil das Comunicações.....	76
Figura 3.6 Gerenciamento Ágil dos Riscos.....	81
Figura 3.7 Gerenciamento Ágil dos Custos	85
Figura 4.1 Caracterização dos objetivos de medição da experimentação.....	92
Figura 5.1 Grupos de processos do PMBOK® <i>Guide</i> em uma fase	115
Figura 5.2 Fluxo de processo ágeis em uma fase	116

Lista de Tabelas

Tabela 2.1 Conceitos Relacionados ao Gerenciamento Ágil de Projetos	29
Tabela 2.2 Aplicabilidade das abordagens de gerenciamento de projetos de acordo com o ambiente e o <i>stakeholder</i> organizacional	30
Tabela 2.3 Gerenciamento Ágil <i>versus</i> Gerenciamento Clássico	31
Tabela 2.4 Evolução de Sistemas Adaptativos Complexos/XP para práticas APM.....	34
Tabela 2.5 Gerenciamento Ágil comparado aos grupos de processos do PMBOK® <i>Guide</i>	44
Tabela 2.6 Áreas de conhecimento do gerenciamento ágil e do gerenciamento clássico de projetos	45
Tabela 3.1 Técnicas, Práticas e Conceitos Utilizados no Agilius.....	51
Tabela 3.2 Tabela resumo dos processos do Agilius.....	54
Tabela 3.3 Resumo do Processo Iniciar Projeto	56
Tabela 3.4 Exemplo de Matriz de Fator de Exploração	57
Tabela 3.5 Exemplo de Matriz de Custo-Benefício.....	57
Tabela 3.6 Resumo do Processo Planejar Projeto	58
Tabela 3.7 Resumo do Processo Planejar Iterações	59
Tabela 3.8 Resumo do Processo Liderar Execução do Projeto.....	60
Tabela 3.9 Resumo do Processo Monitorar e Controlar Mudanças do Projeto	61
Tabela 3.10 Resumo do Processo Finalizar Iteração.....	62
Tabela 3.11 Resumo do Processo Finalizar Projeto	63
Tabela 3.12 Resumo do Processo Planejar Escopo do Projeto.....	66
Tabela 3.13 Resumo do Processo Criar FBS.....	67
Tabela 3.14 Resumo do Processo Verificar Escopo	68
Tabela 3.15 Resumo do Processo Identificar Mudanças no Escopo.....	69
Tabela 3.16 Resumo do Processo Definir e Priorizar Atividades de Projeto.....	72
Tabela 3.17 Elaborar Cronograma	73
Tabela 3.18 Resumo do Processo Monitorar e Controlar Cronograma.....	75
Tabela 3.19 Resumo do Processo Planejar Comunicações	77
Tabela 3.20 Resumo do Processo Informar <i>Stakeholders</i>	79
Tabela 3.21 Resumo do Processo Monitorar Comunicações	80
Tabela 3.22 Resumo do Processo Identificar Riscos.....	82
Tabela 3.23 Resumo do Processo Analisar Riscos	83
Tabela 3.24 Resumo do Processo Planejar Resposta aos Riscos.....	83
Tabela 3.25 Resumo do Processo Controlar e Monitorar Riscos	84
Tabela 3.26 Resumo do Processo Planejar Custos	87
Tabela 3.27 Resumo do processo Monitorar e Controlar Custos	88
Tabela 4.1 Caracterização do estudo experimental.....	92
Tabela 4.2 Critérios de instrumentação do experimento	96
Tabela 4.3 Possíveis métricas para PUA	96
Tabela 4.4 Dados do perfil dos participantes do experimento	102
Tabela 4.5 Legenda do perfil do participante	102
Tabela 4.6 Dados brutos obtidos a partir da opinião sobre as características do Agilius	103
Tabela 4.7 Validação dos dados da avaliação.....	104
Tabela 4.8 Medidas de tendência central para o Questionário de Avaliação.....	105
Tabela 4.9 Característica mal-entendida	106
Tabela 4.10 Características aprendidas e não utilizadas	106
Tabela 4.11 Características aprendidas	107
Tabela 4.12 Tabela de contingência para o experimento.....	108
Tabela 4.13 Valor de Chi-2 para as distribuições de repostas do experimento.....	109
Tabela 4.14 Valores PUA para as características.....	109
Tabela 4.15 Características não apresentadas pelo Agilius segundo os participantes do experimento....	111

Resumo

A exigência por projetos com resultados mais rápidos faz com que seja repensada a forma de gerenciamento de projetos, em particular os relacionados à elaboração de softwares. Abordagens ágeis de gestão apresentam-se como uma alternativa para projetos com duração curta e requisitos que mudam no decorrer do projeto. Existem algumas abordagens que trabalham o gerenciamento ágil de projetos, como por exemplo: *Agile Project Management*, *Agile Project Management Framework*, *Extreme Project Management* e Scrum, que possuem princípios e práticas em comum. Elas surgiram como uma evolução das metodologias ágeis de desenvolvimento e possuem sua principal fundamentação no Manifesto Ágil para o Desenvolvimento de Software. Contudo, nenhuma dessas abordagens leva em consideração o PMBOK® *Guide* como conjunto de práticas já utilizadas no gerenciamento de projetos. Deste modo, a tendência é que surja um modelo híbrido entre clássico e ágil a fim de unir as vantagens das duas abordagens em prol do gerenciamento e desenvolvimento de projetos de sucesso, na visão do cliente e da equipe de projeto. Tendo em vista essa tendência e partindo da premissa de que um conjunto de práticas em gerenciamento ágil baseado no PMBOK® *Guide* pode ser mais eficiente quando se trata de projetos que necessitam de um pouco de rigor na sua execução e de um processo bem definido, mas que possuem seu foco em resultados ágeis, o objetivo desta dissertação é apresentar o Agilius: uma abordagem de Gerenciamento Ágil de Projetos de Software baseada no PMBOK® *Guide*. Para avaliar o Agilius foi utilizado o método apresentado pela Engenharia de Software Experimental. Foram definidos os objetivos, as questões usadas para a medição e um conjunto de métricas para caracterizar a análise destas questões. Em seguida, o planejamento do estudo experimental e a execução do estudo experimental foram realizados e ao final, para a verificação das hipóteses levantadas foi realizada a análise e a interpretação dos dados coletados a partir de estatística descritiva, análise qualitativa e quantitativa. De acordo o estudo experimental executado, o Agilius pode ser considerado ágil por contemplar de forma satisfatória as principais características que definem o gerenciamento ágil de projetos de software.

Palavras-chave: Gerenciamento de Projetos, Gerenciamento Ágil de Projetos de Software, PMBOK® *Guide*.

Abstract

The high time-to-market exigency makes us rethink about the way of managing projects, in particular those related to software development. Agile management is an alternative for short duration projects, with high change of requirements. There are some approaches to work with Agile Project Management such as: Agile Project Management, Agile Project Management Framework, Extreme Project Management and Scrum. These approaches are an evolution of agile development methodologies. They have principles and practices in common because are based in the Agile Manifesto for Software Development. However, these approaches don't consider PMBOK® Guide practices in project management. According to some authors, an expected trend is a hybrid model between classic and agile project management. Such thing will combine the advantages of both approaches towards the project management and development to success, from the customer and the project team point of view. This work is based on this affirmation and on the hypothesis "agile management practices based on the PMBOK® Guide can be more efficient to manage projects more rigorous, with their focus on agile results". In order to anticipate a tendency, this work presents Agilius: a Software Agile Project Management Project approach based on PMBOK® Guide. To assess Agilius we used Experimental Software Engineering method. We defined goals and questions to be used in the measurement and a set of metrics to analyze the questions. Then, we performed planning and implementation of the experimental study. Finally, we performed the hypotheses verification. Concerning the experimental study results, Agilius can be considered an agile model because it covers the main characteristics that define software agile project management satisfactorily.

Keywords: *Project Management, Software Agile Project Management, PMBOK® Guide.*

Introdução

Neste capítulo introdutório serão abordados os aspectos que justificam o trabalho realizado nesta dissertação. Apresentamos a motivação para o seu desenvolvimento e as contribuições esperadas. A seguir, descrevemos os objetivos almejados na realização deste trabalho. Além disso, será feita a descrição da metodologia de trabalho utilizada para a realização desta dissertação. Por fim, será apresentada a organização do texto.

1.1 Motivação

Para nortear as decisões dos gerentes de projeto existe um corpo de conhecimento elaborado pelo PMI¹, *Project Management Institute*, o PMBOK[®] *Guide* (Corpo de Conhecimento em Gerência de Projetos) [1]. Seu objetivo é descrever conceitos e processos da gerência de projetos, padronizando técnicas e termos já utilizados na atividade de monitoramento de um projeto. O PMBOK[®] *Guide* identifica, dentre os conhecimentos em gerenciamento de projeto, um subconjunto que é reconhecido como uma boa prática. Neste contexto, uma boa prática significa que o conhecimento descrito será aplicado de acordo com o que a equipe de gerenciamento de projetos reconhecer como adequado.

O PMBOK[®] *Guide* serve como base para as organizações no que diz respeito a processos de gestão, mas não pode ser considerado como única referência. A experiência da equipe e do próprio gerente é de grande valor no desenvolvimento de projetos. O guia nada mais é do que um *framework* de processos, onde são definidas áreas de gerenciamento e suas atividades em termos de entradas, ferramentas utilizadas e as saídas geradas. Cada uma das suas áreas possui um detalhamento e um escopo específicos, e estas áreas estão integradas umas com as outras, formando um todo único e organizado.

Este guia é moldado de acordo com as necessidades da organização que o aplica e seus componentes são restritos a estas necessidades. Pode ser aplicado a vários domínios de conhecimento, desde a construção civil até a produção de software. Contudo o guia é considerado bastante burocrático e rígido em sua aplicação, e possui uma difícil adaptação para projetos de software, devido às características desses projetos: invisibilidade, ou seja, a dificuldade de visualizar o

¹ PMI na web: www.pmi.org

progresso de maneira imediata; complexidade, dificuldade de antever suas funcionalidades; e flexibilidade, entre outras.

Um projeto possui o objetivo principal de atingir suas metas dentro de parâmetros como prazo, custo e qualidade. Os gerentes trabalham dentro da “restrição tripla”, ou seja, desenvolvem um produto ou serviço dentro do escopo, tempo e custo determinados pelo cliente. A relação entre esses fatores ocorre de tal forma que se algum dos três fatores mudar, pelo menos um outro fator provavelmente será afetado [1].

Existe ainda a preocupação com problemas que podem ocorrer durante o projeto, que por muitas vezes são caracterizados como riscos. Um risco é um evento ou condição incerta que, se ocorrer, terá um efeito positivo ou negativo em pelo menos um objetivo do projeto [1]. É papel do gerente estar preparado para lidar com os riscos e poder responder a eles rapidamente.

O Gerenciamento Clássico de Projetos trabalha com escopo, tempo e custo fixos e tenta adequar o controle de qualidade ao processo de desenvolvimento. Abaixo, algumas características que [2] apresenta como condizentes com as metodologias tradicionais de desenvolvimento e que este trabalho considera como pertencentes às metodologias de Gerenciamento Clássico de Projetos:

- **São preditivas:** cada etapa de desenvolvimento do projeto é baseada na etapa anterior, o que só funciona bem se a concepção do sistema não sofre nenhuma alteração, ou seja, parte do princípio que os requisitos são estáveis. Quando se percebe que uma definição não foi a mais acertada, a tendência natural é resistir às mudanças, o que prejudica a evolução do sistema.
- **São orientadas a processos:** partem do princípio de que processos bem definidos devem ser impostos e executados, para garantir a qualidade do produto resultante. Pessoas são tratadas como recursos alocados, da mesma forma que equipamentos e ferramentas. Desconsidera o talento pessoal, o que acaba afetando o moral e, em consequência, a produtividade da equipe.
- **São rígidas:** pressupõem que é possível especificar de antemão todas os requisitos do software a ser desenvolvido, dificultando – e muitas vezes impedindo – realizar alterações e comprometendo a evolução natural da solução. Estão mais voltadas para a obtenção do software previamente especificado do que para o sucesso do projeto ou negócio para o qual o software foi apontado como solução.

- **São burocráticas:** geram muita sobrecarga no desenvolvimento a ser realizado, comprometendo a velocidade de desenvolvimento e, muitas vezes, o sucesso do projeto.

Dessa forma, o que se pode observar é que a abordagem clássica de gerenciamento apesar de realizar planejamento, controle e acompanhamento ainda se apresenta resistente a mudanças que ocorrem devido às incertezas inerentes a um projeto de software. Por outro lado, as metodologias ágeis são adaptativas, orientadas a pessoas, flexíveis, iterativas e buscam constantemente a simplicidade [2]. Neste contexto, o Gerenciamento Ágil de Projetos busca a simplicidade e a agilidade, que é traduzida em soluções rápidas, econômicas e de valor para o cliente e a equipe de desenvolvimento, e aumenta o incentivo à construção de produtos de forma evolutiva e adaptativa [3]. O objetivo maior é fazer com que o software produzido entre em operação o mais rápido possível, para que ele tenha a chance de evoluir.

É comum clientes de projetos de software só descobrirem o que eles realmente desejam de um produto de software depois de utilizarem alguma versão operacional do mesmo. Isto significa que, num caso como este, possivelmente os requisitos não foram completamente compreendidos no começo do projeto, e mudam durante o seu desenvolvimento. Baseadas nesta frustração e na necessidade de obtenção rápida de produtos, as organizações começam a implementar processos de desenvolvimento de software empíricos como as metodologias de desenvolvimento ágeis [4].

A urgência por resultados mais rápidos faz com que seja repensada a forma de desenvolvimento de projetos, em particular os relacionados à elaboração de softwares. A quebra de paradigma em relação ao desenvolvimento de aplicações apresenta abordagens ágeis de gestão como alternativa para projetos com duração curta e requisitos que mudam durante o projeto.

O Gerenciamento Clássico de Projetos é criticado por ser utilizado de forma que dificulta sua adaptação a projetos mutáveis. Em contrapartida, as metodologias de gerenciamento ágil vêm ganhando popularidade pela sua adaptabilidade.

Tendo em vista este cenário, o Gerenciamento Ágil de Projetos surge como uma alternativa ao Clássico. O Gerenciamento Ágil de Projetos prega que a gerência de pessoas e processos criativos demanda processos de gerenciamento criativos, faz com que o gerente saiba menos sobre os assuntos técnicos do projeto, e enfatiza que o que acontece depois que o projeto termina é mais importante do que o que acontece durante o próprio projeto [3], [5]. A busca pela simplicidade, que significa o mínimo de complexidade possível [6] na realização de um projeto, também é enfocada.

Existem algumas metodologias que utilizam o Gerenciamento Ágil de Projetos, como por exemplo: *Agile Project Management* [3], [5], *Agile Project Management Framework* (baseado em Sistemas Adaptativos Complexos) [7], *Extreme Project Management* [8] e Scrum [9]. Elas possuem forte embasamento nos princípios evidenciados no Manifesto Ágil [10] e levam em consideração as pessoas e o ambiente em que elas estão inseridas. No entanto, nenhuma dessas metodologias leva em consideração o PMBOK® *Guide* como conjunto de práticas já utilizadas em gerenciamento de projetos.

Nos últimos tempos, o gerenciamento de projetos vem obtendo lugar de destaque nas organizações desenvolvedoras de software. Os resultados dos projetos de software em sua maioria não satisfazem os clientes, pois não cumprem restrições de prazo, escopo, qualidade e custos. O PMBOK® *Guide* é uma referência em gestão de projetos, mas devido ao fato de ser aplicado a qualquer projeto ele torna-se superficial e, por vezes, de difícil adaptação.

A entrega de produtos no prazo, dentro dos padrões de qualidade e dentro do orçamento previsto não é uma realidade para a maioria dos projetos. No intuito de obter projetos com mais chance de sucesso, as metodologias ágeis surgem como alternativa e vêm ganhando destaque.

Scrum [9] é um exemplo de metodologia de gestão ágil que pode ser aplicada a qualquer projeto, e que está sendo bastante utilizada em projetos de software. Na verdade, ela se caracteriza como um conjunto de práticas de organização das equipes e da informação distribuída durante o projeto. Embora seu uso esteja crescendo e tenha apresentado sucesso em vários casos, o Scrum, assim como outras abordagens, ainda é visto como insatisfatório pelo seu uso de forma errônea ou por apresentar um nível de formalidade que não atende a algumas organizações que preferem utilizar a abordagem clássica.

De acordo com Karlström e Runeson (2005) [11], pesquisas têm mostrado que existe um conflito visível relacionado com a quantidade de documentação que deveria ser mantida, o qual se apresenta como um balanço entre quanto trabalho consumir com elaboração da documentação e a utilidade de tal documentação. Assim, faz-se necessário obter uma proporção mínima entre quantidade de documentação ou burocracia e a sua utilidade, a fim de que um projeto seja realizado de forma ágil.

Surge então um problema: como controlar projetos de maneira mais eficiente quando se trata de projetos que necessitam de certo rigor [11], mas que possuem seu foco em resultados ágeis?

Dessa forma, este trabalho propõe a definição de uma abordagem que reuniu boas práticas de gerenciamento ágil, com enfoque em projetos de desenvolvimento de software. Esta proposta foi realizada com base em seis das nove áreas de conhecimento definidas pelo PMBOK® *Guide*. Com base nas abordagens de Gerenciamento Ágil de Projetos e no PMBOK® *Guide*, foram desenvolvidos processos com um enfoque ágil, que em seguida foram mapeados em uma linguagem de modelagem, que descreveu a estrutura necessária para expressar processos formalmente.

A seguir serão apresentadas as áreas de conhecimento trabalhadas no Agilius e a razão porque cada uma delas foi escolhida:

- **Gerenciamento de Escopo:** devido à volatilidade do escopo e as constantes mudanças nos requisitos que caracterizam o desenvolvimento de software.
- **Gerenciamento de Custos:** pois em gerenciamento ágil esta área é muito pouco discutida.
- **Gerenciamento de Tempo:** pelo fato de ser comum em projetos de software o atraso na entrega do produto final.
- **Gerenciamento das Comunicações:** pois as metodologias ágeis dedicam grande atenção aos indivíduos e as interações entre eles.
- **Gerenciamento de Riscos:** considerando que não há uma maneira padrão sugerida para o tratamento de riscos em gestão ágil. Cada projeto adota sua própria abordagem.

Considerando o contexto em questão, as principais contribuições deste trabalho são:

- Uma análise das metodologias ágeis de gerenciamento de projetos e as conseqüências do seu uso no desenvolvimento de projetos de software.
- O Agilius, abordagem ágil de gerenciamento de projetos de software baseada nas metodologias ágeis de gerenciamento de projetos e no PMBOK® *Guide*.

1.2 Objetivo

Propor um conjunto de atividades baseadas no gerenciamento ágil de projetos para o gerenciamento de projetos de software, de acordo com o guia PMBOK® e com a literatura sobre metodologias ágeis de gestão. Esse conjunto de processos propostos, em cada uma das áreas de conhecimento exploradas, tem como objetivo amenizar o problema de gerenciar projetos de

desenvolvimento de software e trazer uma proposta de gerenciamento de projetos de software baseada no PMBOK® *Guide* e em metodologias de gerenciamento ágil, a fim de reunir o melhor de cada um deles em uma abordagem leve e aplicável às organizações.

1.3 Metodologia de Trabalho

A seguir serão apresentadas as etapas envolvidas na realização deste trabalho.

Etapa 1: Estudo Inicial

Foi realizado um estudo das metodologias ágeis de gerenciamento de projeto existentes, sendo possível através desta exploração inicial o desenvolvimento de conceitos, o estabelecimento de prioridades e o planejamento mais aprimorado da pesquisa. Utilizamos também o PMBOK® *Guide*, como base por ser um modelo de grande aceitação nas organizações de um modo geral.

Etapa 2: Definição da Abordagem Agilius

Para o desenvolvimento do Agilius - abordagem ágil baseada no PMBOK® *Guide* que é proposta, primeiramente foram investigadas características das abordagens de gerenciamento ágil de projetos levantadas no estudo realizado na Etapa 1. Dentro do Guia PMBOK® foram escolhidas seis áreas de conhecimento para este trabalho, devido ao grau de importância destas áreas para o gerenciamento de projetos. Em seguida, as abordagens ágeis foram mapeadas nos processos definidos pelo PMBOK® *Guide*, dando origem à abordagem Agilius.

Etapa 3: Modelagem da Abordagem Agilius

Com a conclusão da Etapa 2, a definição do Agilius, a abordagem foi documentada em uma linguagem de definição de processos de software, que sofreu uma adaptação no intuito de melhor representar as características de uma abordagem de gerenciamento.

Etapa 4: Preparação do Experimento

Nesta etapa foram realizados os passos iniciais necessários ao processo de experimentação, definido em [12]: delimitação dos objetivos do estudo experimental, realização do planejamento do estudo em termos de hipóteses, instrumentação, seleção de contexto de aplicação e dos indivíduos participantes do experimento, análise qualitativa e qual a validade do experimento.

Etapa 5: Execução e Análise do Experimento

Num primeiro momento, foi solicitado aos indivíduos que participaram do experimento que respondessem a um questionário para levantamento do perfil, que consistia basicamente de:

- **Qualificação do respondente:** que diz respeito à experiência com modelos de gerenciamento ágil e o uso e/ou conhecimento do PMBOK® *Guide*;
- **Caracterização do respondente:** caracterizando o tempo de execução da função numa equipe de desenvolvimento de projetos de software;
- **Qualificação do projeto:** onde o respondente explicou qual era o cenário dos projetos dos quais participou.

Depois os participantes assistiram a uma apresentação da abordagem proposta e responderam a um novo questionário, a respeito das suas impressões sobre o que está disponível no Agilius. Por fim, os dados coletados foram analisados e interpretados segundo o que foi apresentado na Etapa 4.

1.4 Estrutura da Dissertação

A dissertação está estruturada da seguinte forma:

- O presente capítulo contém uma introdução ao trabalho realizado, definindo seu objetivo e suas contribuições.
- O Capítulo 2 descreve o gerenciamento do desenvolvimento de projetos de software e sua relação com o PMBOK® *Guide*, algumas metodologias de gestão ágil de projetos de software e faz considerações sobre o gerenciamento ágil de projetos de software e o Guia PMBOK®.
- O Capítulo 3 apresenta o Agilius, abordagem de gerenciamento proposta neste trabalho e explica suas características principais.
- O Capítulo 4 descreve o processo de experimentação utilizado na avaliação da abordagem.
- O Capítulo 5 apresenta as conclusões do trabalho e uma descrição de possíveis trabalhos futuros.

Nos ambientes de projetos, as equipes precisam identificar e rastrear requisitos, desenvolver funcionalidades, manter um relacionamento próximo com seus clientes e *stakeholders*. Devido às restrições de tempo e orçamento, as organizações precisam estar aptas a fazer muito mais com muito menos, e acima de tudo, entregar valor ao cliente.

A agilidade então pode ser definida como a capacidade de se adequar a mudanças e imprevisibilidade de projetos dinâmicos, como são os projetos de software. Isto se dá através do reconhecimento de mudanças e da adaptação dos projetos a elas.

O Gerenciamento Ágil de Projetos surgiu como uma evolução, no âmbito gerencial, das metodologias ágeis de desenvolvimento. Uma resposta às crescentes pressões por inovação em prazos cada vez mais reduzidos e ao mau desempenho de grande parte dos projetos de desenvolvimento de software [13].

Neste contexto, este capítulo apresenta uma definição Gerenciamento Ágil de Projetos de Software, compatível com os objetivos deste trabalho. Para tanto, são apresentados os conceitos de gerenciamento de projetos, de projetos de software e algumas abordagens ágeis de gerenciamento de projetos. Ao final é apresentada a relação existente entre o gerenciamento ágil de software e o PMBOK® *Guide* e uma motivação para a definição do Agilius, proposta desta dissertação.

2.1 Gerenciamento de Projetos

Um projeto é um esforço temporário com a finalidade de criar um produto, serviço ou resultado exclusivo [1]. Utiliza recursos limitados e é conduzido por pessoas, com o objetivo principal de atingir suas metas dentro de parâmetros de prazo, custo e qualidade. De acordo com o PMBOK® *Guide*, um projeto pode ser definido sob perspectivas distintas como temporário, único e de elaboração progressiva. É temporário, pois todo projeto tem um início e um fim definidos. O projeto termina quando os objetivos para o qual foi criado são atingidos ou quando se torna claro que os objetivos do projeto não serão ou não poderão mais ser atingidos, ou mesmo quando a necessidade do projeto não existe mais. Um projeto é único, pois cria entregas exclusivas, que são produtos, serviços ou resultados. Elaboração progressiva significa desenvolver em etapas e continuar por incrementos. Isto significa que o escopo do projeto será descrito de maneira geral no início do

projeto e se tornará mais explícito e detalhado conforme a equipe do projeto desenvolve um entendimento mais completo dos objetivos e das entregas. Considerando esta definição, os seguintes são exemplos de projetos:

- Desenvolvimento de um novo produto ou serviço;
- Efetuar uma mudança de estrutura, de pessoal ou de estilo de uma organização;
- Desenvolvimento ou aquisição de um sistema de informações novo ou modificado;
- Construção de um prédio, de uma espaçonave;
- Construção de um sistema de abastecimento de água para uma comunidade;
- Realizar uma campanha por um cargo político;
- Implementação de um novo procedimento ou processo de negócios.

Para executar um projeto faz-se necessário gerenciá-lo. O termo Gerenciamento de Projetos pode ter significados diferentes para pessoas diferentes. Gerenciamento de Projetos é o planejamento, a organização, a direção e o controle dos recursos das organizações para um objetivo de curto prazo que foi estabelecido para concluir metas específicas [14].

Uma outra definição afirma que Gerenciamento de Projetos é a aplicação do conhecimento, habilidades, ferramentas e técnicas às atividades do projeto de forma a atingir e exceder as necessidades e expectativas dos interessados pelo projeto [1]. Gerenciar projetos também significa coordenar as etapas de definição, planejamento, execução, controle e conclusão de um projeto.

Todos os projetos necessitam de algum nível de gerenciamento. Quanto maior e mais complexo o projeto, maior é a necessidade de um processo formal, padronizado e estruturado. Os projetos menores também necessitam de um processo estruturado, mas este não deve ser excessivamente elaborado ou complexo.

As organizações de reconhecida excelência em gestão de projetos criam um ambiente no qual existe um fluxo contínuo de projetos gerenciados com sucesso, onde o sucesso é mensurado tanto pela conclusão de um projeto específico, quanto por atingir o desempenho em pontos de interesse para a empresa como um todo [15]. Para garantir o sucesso contínuo dos projetos deve existir um forte comprometimento da organização com o gerenciamento de projetos, e este comprometimento deve ser visível [14]. O sucesso na gerência de projetos está ligado à existência de uma metodologia que guie o gerente e sua equipe ao longo dos projetos. Um gerente de projetos e

equipe bem treinados, bem como a adoção de uma metodologia, e seu uso contínuo aumentam as chances de sucesso de uma organização.

2.1.1 Projetos e Gerenciamento de Projetos no PMBOK® *Guide*

O PMBOK® *Guide* ou “Corpo de Conhecimento em Gerência de Projetos” é um guia para a atividade de gerenciamento de projetos elaborado pelo PMI – *Project Management Institute* [16]. Seu objetivo é descrever conceitos e processos de gerência, padronizando as técnicas e os termos já utilizados na atividade de monitoramento de um projeto. Ele identifica dentre os conhecimentos em gerenciamento de projetos um subconjunto que é reconhecido como uma boa prática. Neste contexto, uma boa prática significa que o conhecimento descrito será aplicado de acordo com o que a equipe de gerenciamento de projetos reconhecer como adequado.

O PMBOK® *Guide* é um guia genérico, aplicável a várias áreas de conhecimento, desde a construção civil até a produção de software. Ele define áreas de conhecimento em gerenciamento e suas atividades em termos de entradas, ferramentas utilizadas e as saídas geradas. Cada uma das suas áreas possui um detalhamento e um escopo específicos, mas estas áreas estão integradas umas com as outras, formando um todo único e organizado. As áreas e os processos existentes podem ser utilizados ou não. O guia pode ser moldado de acordo com as necessidades da empresa que o aplica e seus componentes são restritos a estas necessidades.

Devido à sua adaptabilidade, pode ser complementado com conhecimentos provenientes da experiência dos profissionais em gerenciamento de projetos. Possui práticas tradicionais comprovadas e amplamente aplicadas, bem como práticas inovadoras. Esta característica o transforma em uma referência em constante evolução.

Considerando sua abrangência, uma boa política é tentar aproveitar o que a empresa que o utiliza possui em termos de padronização de documentos e dos processos utilizados, procurando sempre adaptar o PMBOK® *Guide* ao que já existe. Vale ressaltar que há um vocabulário padrão, fornecido pelo próprio guia, que é utilizado para discutir, escrever e aplicar o gerenciamento de projetos.

O PMBOK® *Guide* é organizado em áreas de conhecimento e cada área de conhecimento é descrita através de processos. Ao todo são nove áreas, onde cada uma refere-se a um aspecto específico a ser considerado dentro da gerência de projetos e um total de 44 processos que descrevem essas áreas. Nesta visão, os processos ocorrem dentro de cinco grupos básicos (iniciação,

planejamento, execução, controle e encerramento) e podem se sobrepor e interagir entre si conforme a fase do projeto.

As áreas de conhecimento dizem respeito às práticas em gerência de projetos através dos processos que as compõem. São elas: Integração, Escopo, Tempo, Custo, Qualidade, Recursos Humanos, Comunicação, Riscos e Aquisição. Uma descrição mais detalhada de cada uma delas é apresentada a seguir.

Gerenciamento da Integração do Projeto

Descreve os processos necessários para assegurar que os diversos elementos do projeto sejam adequadamente coordenados bem como suas atividades. Responsável pelo desenvolvimento do plano do projeto, execução do plano do projeto e controle geral de mudanças. É composta dos processos: Desenvolver o termo de abertura do projeto, Desenvolver a declaração do escopo preliminar do projeto, Desenvolver o plano de gerenciamento do projeto, Orientar e gerenciar a execução do projeto, Monitorar e controlar o trabalho do projeto, Controle integrado de mudanças e Encerrar o projeto.

Gerenciamento do Escopo do Projeto

Escopo diz respeito aos limites impostos para execução de um projeto. Ele apresenta o que será o resultado do projeto e o que será necessário para sua execução, ou seja, define exatamente o que deve ser feito no decorrer do projeto. Ao ultrapassar esses limites perde-se o controle da atividade sendo executada e são gastos tempo e recursos desnecessários, o que resulta num produto que está além do que se espera como resultado final do projeto [17].

O Gerenciamento do Escopo descreve os processos necessários para assegurar que o projeto contemple todo o trabalho requerido, e nada mais que o trabalho requerido, para que o projeto seja completado com sucesso. A preocupação reside em definir e controlar o que está incluído no projeto e o que não está.

Seu objetivo principal é garantir que mudanças sejam aprovadas ou não depois de acordado o escopo inicial. Se os *deliverables* modificarem durante o projeto (e, geralmente, isto significa que o cliente deseja itens adicionais), as estimativas de custos, trabalho, e duração podem não ser mais válidas. Esta é a essência e a finalidade do gerenciamento da mudança de escopo - garantir que os acordos iniciais sejam cumpridos, e que a equipe de projeto e que os *stakeholders* concordem com as

mudanças das expectativas [17]. Ela consiste dos processos de: Planejamento do escopo, Definição do escopo, Criar estrutura analítica de projeto, Verificação do escopo e Controle do escopo.

Gerenciamento do Tempo do Projeto

A satisfação do cliente está diretamente ligada aos benefícios proporcionados pelo projeto. Logo, o quanto antes os produtos forem entregues, mais rápido os benefícios potenciais do projeto podem ser reconhecidos, pelo menos na maioria dos casos. É esta atitude dos clientes que torna o gerenciamento de tempo do projeto uma área tão crítica dentro do gerenciamento de projetos. Faz parte do objetivo de um projeto entregar todo escopo acordado, com a qualidade esperada pelo cliente dentro do prazo e custo orçados. Dessa forma, quanto maior a duração do projeto, maior o tempo necessário para que possam ser sentidos os seus benefícios [18].

Devido à representatividade do fator tempo no contexto de gerenciamento de projetos, a área de gestão de tempo se encarrega de descrever os processos necessários para assegurar o término do projeto dentro do prazo previsto. É composta dos processos: Definição da atividade, Seqüenciamento de atividades, Estimativa de recursos da atividade, Estimativa de duração da atividade, Desenvolvimento do cronograma e Controle do cronograma.

Gerenciamento do Custo do Projeto

Descreve os processos necessários para assegurar que o projeto seja completado dentro do orçamento acordado. Os custos afetam diretamente várias atividades do gerenciamento de projetos, o que torna o seu gerenciamento e planejamento práticas vitais para a organização. Seus processos estão envolvidos com a supervisão de planejamento dos recursos, estimativa, orçamento e controle dos custos. Consiste dos processos: Estimativa de custos, Orçamentação e Controle de custos.

Gerenciamento da Qualidade do Projeto

Descreve os processos que asseguram que os objetivos e necessidades que originaram o desenvolvimento do projeto serão satisfeitos. Abrange o planejamento, garantia e controle da qualidade. A qualidade é definida pelo cliente, e representa quão próximo o projeto e suas entregas estarão das suas exigências e expectativas [17]. Seus processos são os seguintes: Planejamento da qualidade, Realizar a garantia da qualidade e Realizar o controle da qualidade.

Gerenciamento dos Recursos Humanos do Projeto

Apresenta os processos necessários para melhor organizar e gerenciar as pessoas envolvidas no projeto. Responsável pelo planejamento organizacional e pela escolha dos membros da equipe. Composta pelos processos: Planejamento de recursos humanos, Contratar ou mobilizar a equipe do projeto, Desenvolver a equipe do projeto e Gerenciar a equipe do projeto.

Gerenciamento das Comunicações do Projeto

A Comunicação de forma efetiva é um fator crítico para o sucesso do gerenciamento das expectativas dos *stakeholders*. Se estas pessoas não forem bem informadas sobre o andamento do projeto, existe uma chance muito maior de ocorrerem problemas e dificuldades devido a diferentes níveis de expectativas [17]. Problemas devido à falta de comunicação geram conflitos entre *stakeholders* e diminuem a velocidade de execução do projeto e podem desestimular a equipe e o próprio cliente.

Esta área de conhecimento descreve os processos necessários para assegurar que a geração, a captura, a distribuição, o armazenamento e a apresentação das informações do projeto sejam feitos de forma adequada e no tempo certo. Ela consiste dos processos de gerenciamento de projetos: Planejamento das comunicações, Distribuição das informações, Relatório de desempenho e Gerenciar as partes interessadas.

Gerenciamento dos Riscos do Projeto

Risco refere-se a futuras condições ou circunstâncias que aparecem e estão fora do controle da equipe do projeto e que poderão ter um impacto negativo sobre ele caso ocorram. Em outras palavras, considerando que uma incidência é um problema atual que deve ser abordado, o risco é um problema potencial futuro que ainda não ocorreu [17]. Esta área de conhecimento é formada pelos processos: Planejamento do gerenciamento de riscos, Identificação de riscos, Análise qualitativa de riscos, Análise quantitativa de riscos, Planejamento de respostas a riscos e Monitoramento e controle de riscos.

Gerenciamento das Aquisições do Projeto

Descreve os processos necessários para a aquisição de mercadorias e serviços de terceiros, bem como, elaboração e o gerenciamento de contratos. É responsável pelo planejamento das aquisições, preparação das aquisições, obtenção de propostas, seleção de fornecedores, administração dos contratos e encerramento do contrato. Seus processos são os seguintes: Planejar

compras e aquisições, Planejar contratações, Solicitar respostas de fornecedores, Selecionar fornecedores, Administração de contrato e Encerramento do contrato.

2.2 Gerenciamento de Projetos de Software

Um projeto de software possui particularidades que o tornam diferente dos demais tipos de projetos, como por exemplo, a construção de um edifício ou de uma estrada. Isto acontece devido a características únicas dos softwares [19], conforme descrito abaixo:

- **Invisibilidade:** Quando um artefato como uma ponte ou estrada é construído, o progresso do que está sendo feito pode ser visto. Em se tratando de software, o progresso não é imediatamente visível.
- **Complexidade:** Computadores e usuários de software se tornaram mais sofisticados o que proporcionou o aumento da complexidade dos produtos de software em relação a outros artefatos projetados.
- **Flexibilidade:** A facilidade com a qual o software pode ser modificado é geralmente vista como um dos seus pontos fortes. Entretanto, isto significa que onde sistema de software faz interface com um sistema físico ou organizacional, é esperado que, se necessário, o software mude para acomodar os outros componentes. Isto significa que sistemas de software são propensos a um alto grau de mudança.

Deste modo, algumas técnicas e ferramentas genéricas de gerenciamento de projetos, como, por exemplo, as oferecidas no PMBOK® *Guide*, necessitam de adaptação para serem aplicadas aos projetos de software. Por exemplo, ao trabalhar escopo e tempo é preciso considerar que, devido à característica flexibilidade, um maior número de mudanças pode ocorrer. Isto acontece pela subjetividade envolvida no desenvolvimento de um software.

Segundo [20], um projeto de software é uma descrição de estrutura de software a ser implementada, dos dados que são parte do sistema, das interfaces entre os componentes do sistema e, algumas vezes, dos algoritmos utilizados.

Gerenciamento de Projetos de Software pode ser definido como a aplicação das atividades de gerenciamento: planejamento, coordenação, medição, monitoramento, controle e geração de relatórios, a fim de assegurar que o desenvolvimento e a manutenção do software seja sistemática, disciplinada e quantificada [21].

Neste trabalho, a abordagem defendida por [1], [14] e [20] é denominada de *Gerenciamento Clássico de Projetos*, que é o gerenciamento de projetos de software estruturado através de processos, planejamento e controle, e orientado por atividades. Projetos estáveis, de longa duração, e com baixo nível de mudanças se beneficiam desta abordagem.

A palavra "Clássico" é utilizada para definir este tipo de projeto e diferenciá-lo do Gerenciamento Ágil, de acordo com [3] e [5], como será apresentado na Seção 2.3.

O Gerenciamento de Projetos Clássico, caracterizado pelo PMBOK® *Guide*, trabalha com escopo, tempo e custo fixos e tenta adequar o controle de qualidade ao processo de desenvolvimento. No entanto, o que se pode observar é que a abordagem clássica de gerenciamento não considera que um projeto de software é dinâmico. Esta abordagem trabalha de forma reativa, preparando-se apenas para a possível existência de mudanças, não as considerando como parte do projeto.

O responsável pelo gerenciamento de um projeto e pela realização dos seus objetivos é o gerente de projetos. Ele centraliza as decisões de projeto e é responsável pelo seu sucesso. Nesta abordagem de gerenciamento existe uma clara definição de papéis e responsabilidades e cada membro possui suas atividades bem definidas.

O trabalho do gerente de projeto de software é garantir que o projeto de software cumpra restrições de orçamento e de prazo, e entregar um produto de software que contribua para as metas da empresa [20]. Os gerentes de software são responsáveis por planejar e controlar o desenvolvimento do projeto.

A participação do cliente é notada com maior frequência nos primeiros meses de projeto, quando são levantados os requisitos e definidos os objetivos do projeto. Inicialmente, o cliente apresenta suas necessidades que deverão ser traduzidas em um projeto. Depois das fases iniciais, a participação do cliente diminui dentro do projeto e se restringe à aprovação e validação de artefatos.

O Gerenciamento Clássico de Projetos apresenta um processo de desenvolvimento burocrático: plano de projeto extenso e detalhado, cronograma e custos são definidos no início do projeto, comunicação formal entre os envolvidos no projeto. A necessidade de modificação de requisitos ou negociação de prazos pode gerar conflitos com o cliente, que devido ao planejamento detalhado apresentado, espera resultados dentro dos prazos e custos estabelecidos inicialmente. O plano de projeto será utilizado para auxiliar na medição do progresso da execução do projeto e pode sofrer alterações neste período.

A comunicação é formal e documentada, e possui o objetivo de registrar todos os fatos para evitar conflitos durante o projeto. Devido ao alto grau de burocracia atribuído à comunicação, informações necessárias ao andamento do projeto podem não chegar a tempo de fazer parte de decisões importantes.

2.3 Gerenciamento Ágil de Projetos

O tempo é um fator condutor no desenvolvimento de novos produtos (*time-to-market*), mas o encolhimento de cronogramas não está restrito ao desenvolvimento de novos produtos. De acordo com [3], novas práticas de implementação de negócios, como as estimuladas pela instalação de *Customer Relationship Management* – CRM [22] geralmente estão repletas de incertezas de tipos diferentes. A alta taxa de falhas reportadas em implementações CRM pode ser atribuída, em parte, às abordagens de gerenciamento de projetos antecipatórias (*plan-driven*) que falham ao explorar incertezas causadas pelas maiores mudanças de processo de negócios. Neste contexto, o conceito de agilidade está associado ao tempo: o quão rápido será realizada uma atividade e à incerteza inerente à realização de uma tarefa.

A busca pela agilidade, traduzida em soluções rápidas, econômicas e de valor para o cliente e equipe de desenvolvimento, aumenta o incentivo à construção de produtos de forma evolutiva e adaptativa. O objetivo maior é fazer com que o software entre em operação o mais rápido possível, para que ele tenha a chance de evoluir.

Contudo, inovação e desenvolvimento rápido não são suficientes. Além disso, as organizações têm que entregar melhores produtos, que sejam adequados ao que os clientes querem, no tempo que os clientes determinam. Empresas que tem a habilidade de desenvolver um produto rapidamente e a custo mais baixo, o mais próximo possível do fim do ciclo de vida, terão uma grande vantagem competitiva [3].

A urgência por projetos com resultados mais rápidos faz com que seja repensada a forma de desenvolvimento de projetos, em particular os relacionados à elaboração de softwares. A quebra de paradigma em relação ao desenvolvimento de aplicações apresenta abordagens ágeis de gestão como alternativa para projetos com duração curta e requisitos que mudam durante o projeto. Na tentativa de apresentar possíveis dificuldades no gerenciamento de projetos de software, [19] apresenta os seguintes problemas:

- Especificação do trabalho inadequada;

- Ignorância em gerenciamento de TI;
- Falta de conhecimento da área de aplicação;
- Ausência de padrões;
- Falta de documentação atualizada;
- Atividades precedentes não são completadas no tempo predeterminado;
- Falta de comunicação entre clientes e técnicos;
- Falta de comunicação que leva ao re-trabalho;
- Falta de comprometimento - especialmente quando o projeto está amarrado a uma pessoa que se desliga da equipe;
- Reduzido escopo de experiência técnica;
- Mudanças de ambiente de software;
- Pressão ocasionada pelos *deadlines*;
- Falta de controle de qualidade;
- Falta de treinamento.

O 2nd *Annual State of Agile Development Survey* apresentou uma tentativa de quantificar o uso e os motivos que levam as empresas de TI a utilizar as metodologias ágeis [23]. Este *survey* contou com 1700 respondentes de 71 países que compartilharam as experiências de suas companhias utilizando métodos ágeis e o desafio que eles enfrentam. As equipes quantificaram o valor que seus projetos atingiram utilizando práticas ágeis e ferramentas de apoio a essas práticas, como é mostrado na Figura 2.1 e Figura 2.2.

Dentre as dez diferentes razões para adotar metodologias ágeis (melhorar a visibilidade do projeto, aumentar a habilidade de gerenciar mudança de prioridades, aumentar produtividade, acelerar o *time-to-market*, aumentar a qualidade de software, reduzir riscos de projeto, reduzir custos, gerenciar times distribuídos, reduzir a complexidade do processo e melhorar o alinhamento entre TI e os objetivos de negócio) mais de três quartos citaram gerenciar mudança, aumentar a produtividade, *time-to-market* e qualidade como fatores principais, conforme apresentado no gráfico da Figura 2.1.

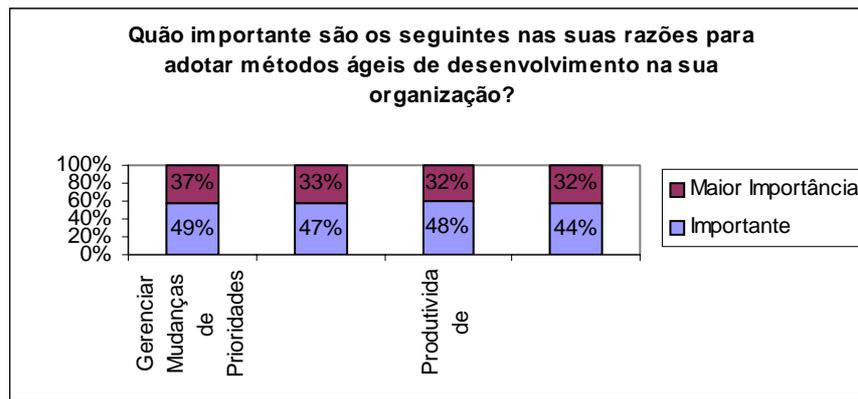


Figura 2.1 Razões para adotar metodologias ágeis

Ainda no mesmo *survey*, os participantes estimaram os níveis quantitativos de melhoria, segundo apresentado na Figura 2.2. A visibilidade do projeto que utiliza as metodologias ágeis aumenta, pois times ágeis aumentaram a ênfase em quantificar e comunicar o valor atingido nos seus projetos.

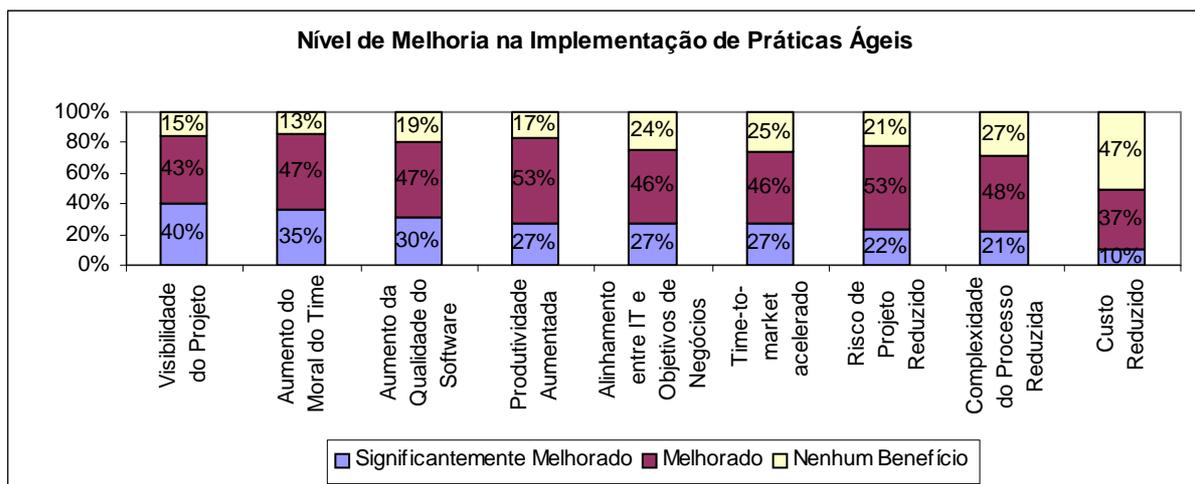


Figura 2.2 Nível de Melhoria na Implementação de Práticas Ágeis

Considerando esse contexto, o termo agilidade é definido neste trabalho como a habilidade de entregar valor ao cliente enquanto se lida com imprevisibilidade e dinamismo presentes nos projetos, principalmente os projetos de software. Isto ocorre quando se reconhece que no decorrer da sua execução, um projeto passa por mudanças e que precisa se adaptar a elas.

Ainda, quando se trata de ambientes de produção de software, agilidade pode ser considerada a capacidade de balancear estabilidade com flexibilidade, ordem com caos, planejamento com execução, otimização com exploração e controle com velocidade para entregar valor confiável em face de incerteza e mudança [8]. Assim, a partir das definições anteriores, pode-se considerar que para o desenvolvimento de software agilidade está associada à habilidade de responder rapidamente a mudanças.

A partir da década de 1970, surgiram alternativas no sentido de se aumentar a agilidade no setor industrial. Estas iniciativas são apresentadas na Tabela 2.1, a seguir.

Tabela 2.1 Conceitos Relacionados ao Gerenciamento Ágil de Projetos [24]

Ano	Iniciativa	Características
1970	Produção <i>just-in-time</i> (JIT)	Criada por Taiichi Ohno, a expressão JIT começou a ter emprego na construção naval nipônica em 1970 [25]. Este sistema procurou minimizar o estoque na fábrica e assegurar que o estoque necessário fosse entregue no momento que deveria ser consumido, imediatamente antes de quando era necessário.
1984	<i>Theory of Constraints</i> (TOC)	A Teoria das Restrições foi introduzida por Eli Goldratt, no livro <i>The Goal</i> . Identifica gargalos na linha de produção. Assume uma que uma cadeia de valor é tão forte quanto o elo mais fraco da cadeia.
1985	<i>Total Quality Management</i> (TQM)	É um termo que foi originalmente cunhado em 1985 pelo <i>Naval Air Systems Command</i> para descrever sua abordagem de gerenciamento estilo japonês para melhoria da qualidade. Representa o estilo de gerenciamento que possui o objetivo de atingir sucesso através da ligação da qualidade com a satisfação do cliente [26].
1987	<i>Six Sigma</i>	Está relacionado à qualidade que se refere especificamente ao nível de defeito de menos de quatro por um milhão. Surgiu em 1987 na Motorola, e ofereceu uma maneira simples e consistente de acompanhar o desempenho e compará-lo às exigências do cliente (a medida sigma) e uma meta ambiciosa de qualidade praticamente perfeita (o objetivo Seis Sigma) [27].
1990	<i>Lean Production</i>	Visa a utilização mais eficiente dos recursos e a redução de qualquer tipo de desperdício. <i>Lean Production</i> utiliza menor quantidade de tudo se comparada com a produção em massa: metade do esforço humano nas fábricas, metade do espaço de manufatura, metade do investimento em ferramentas, metade das horas de engenharia para desenvolver um novo produto na metade do tempo [28].

Assim, o Gerenciamento Ágil de Projetos foi desenvolvido a partir dos valores e princípios dos métodos ágeis de desenvolvimento de software, de acordo com o Manifesto para o Desenvolvimento Ágil de Software, que se expandiu e ampliou sua abrangência ao gerenciamento de projetos [3].

Gerenciamento Ágil de Projetos é definido em [3] como um conjunto de valores, princípios e práticas que auxiliam a equipe de projeto a entregar produtos ou serviços de valor em um ambiente desafiador.

O Gerenciamento Ágil também pode ser visto como um novo elemento que irá suportar as extensões do Gerenciamento Clássico, de forma a habilitar seus praticantes para gerenciar projetos de forma efetiva em um ambiente incerto e mutável [5].

Segundo [5], devem ser consideradas duas dimensões para a aplicabilidade de gerenciamento ágil de projetos:

- **O ambiente do projeto:** Em ambientes operacionais, onde existe baixo grau de incerteza, o gerenciamento clássico possui maior aplicabilidade. Já nos ambientes de desenvolvimento de produtos, processos ou tecnologias o grau de incerteza é alto, sendo o gerenciamento ágil aplicável.
- **Os *stakeholders* organizacionais:** Gerenciamento ágil é mais aplicável quando existem menos *stakeholders* organizacionais, ou seja, quando uma única organização está envolvida no projeto. Múltiplas organizações podem estar envolvidas no projeto, o que se aproxima mais de projetos gerenciados numa abordagem clássica. Quando um projeto é desenvolvido por uma única organização, mas suas áreas funcionais funcionam como organizações autônomas, é possível utilizar um misto de gerenciamento ágil e clássico.

A Tabela 2.2 traz um resumo da aplicabilidade do gerenciamento ágil de projetos.

Tabela 2.2 Aplicabilidade das abordagens de gerenciamento de projetos de acordo com o ambiente e o *stakeholder* organizacional [5]

<i>Stakeholder</i>	<i>Stakeholders</i> externos / Múltiplas organizações	<i>Stakeholders</i> internos / Múltiplas organizações	Organização única
Ambiente operacional	Clássica	Clássica	Clássica
Ambiente de desenvolvimento de produtos	Clássica / Ágil	Clássica / Ágil	Ágil
Ambiente de desenvolvimento de tecnologias/plataformas	Clássica / Ágil	Ágil	Ágil

O Gerenciamento Clássico é fortemente dirigido ao planejamento: deve-se planejar o suficiente, rastrear as atividades executadas de acordo com o plano de projeto e tomar ações corretivas quando o trabalho se desviar deste plano, para obter sucesso [29]. Por outro lado, no ambiente ágil, a ênfase do gerenciamento se desloca do planejamento para a execução [5], pois é durante a execução que as decisões cruciais que determinam o sucesso ou fracasso do projeto são tomadas.

O ciclo de vida do Gerenciamento Clássico de Projetos é definido pelos processos de iniciação, planejamento, execução, controle e fechamento, onde o gerente do projeto é o responsável pela realização dos objetivos do projeto [1]. Contudo, como qualquer plataforma, o Gerenciamento Clássico possui suas restrições que podem torná-lo menos efetivo, e é neste ponto, onde a efetividade da abordagem clássica diminui, que a ágil pode ser aplicada [5].

De acordo com as características apresentadas neste trabalho, foi possível distinguir Gerenciamento Ágil e Clássico, como pode ser visto na Tabela 2.3. Esta tabela apresenta um resumo das abordagens de gerenciamento em relação a: desenvolvedor, cliente, requisitos, retrabalho, planejamento, foco e objetivo de cada uma.

Tabela 2.3 Gerenciamento Ágil *versus* Gerenciamento Clássico

Em relação a	Clássico	Ágil
Desenvolvedor	Hábil	Ágil
Cliente	Pouco envolvido	Comprometido
Requisitos	Estáveis	Mutáveis
Retrabalho	Caro	Barato
Planejamento	Direciona resultados	Resultados o direcionam
Foco	Grandes projetos	Projetos de natureza exploratória e inovadores
Objetivo	Controlar, em busca de alcançar o planejado	Simplificar processo de desenvolvimento

Levando-se em consideração as definições apresentadas anteriormente, este trabalho define Gerenciamento Ágil de Projetos da seguinte forma:

Gerenciamento Ágil de Projetos é o exercício de princípios e práticas ágeis aliados aos conhecimentos, habilidades e técnicas de gerenciamento clássico na execução de um projeto, de forma a diminuir o time-to-market, se adaptar a mudanças e garantir que exista equilíbrio entre demandas de qualidade, escopo, tempo e custos.

A proposta do Gerenciamento Ágil é fazer com que equipe e cliente sintam-se motivados na realização do projeto. O Gerenciamento Ágil reconhece que software é algo difícil de controlar e procura minimizar riscos, garantindo que os engenheiros de software foquem em unidades menores de trabalho.

Para uma melhor compreensão do Gerenciamento Ágil, abordagens de gerenciamento de projetos que seguem seus princípios e práticas serão apresentadas na próxima seção.

2.4 Abordagens de Gerenciamento Ágil de Projetos

Nesta seção serão apresentadas as abordagens ágeis de gerenciamento de projetos mais citadas na literatura: *Agile Project Management*, Scrum, *Agile Project Management Framework* e *Extreme Project Management*. Estas abordagens são algumas das mais discutidas em gerenciamento ágil de software e fazem parte da evolução do entendimento deste assunto.

2.4.1 *Agile Project Management*

Na pesquisa por uma nova abordagem de gerenciamento, Augustine e Woodcock (2003) apontam princípios de gerenciamento de projetos baseados na exploração da compreensão do comportamento humano autônomo. Este comportamento é obtido através do estudo de sistemas vivos na natureza, em especial a noção de Sistemas Adaptativos Complexos [30] na teoria e prática em gerenciamento.

Cientistas estudaram o comportamento coletivo de sistemas vivos na natureza como revoada de pássaros, cardumes de peixes, colônia de formigas e enxame de abelhas. Eles descobriram que enquanto o agente individual nesse sistema adaptativo complexo possui apenas capacidade e regras estratégicas locais, seu comportamento coletivo é caracterizado por auto-organização e inteligência coletiva que é maior que a soma das partes [31].

Em seu trabalho, [32] observou que projetos que empregam metodologias de desenvolvimento ágil podem ser considerados Sistemas Adaptativos Complexos. Ainda na tentativa de aprofundar esta relação, [33] realizou experiências aplicando *Extreme Programming* (XP) associado a um *framework* de Gerenciamento Ágil de Projetos baseado em Sistemas Adaptativos Complexos, a fim de direcionar projetos ao sucesso ao invés de forçá-los a ter sucesso.

O conceito de Sistemas Adaptativos Complexos leva a crer que como o time de XP, gerentes de projeto também precisam de um conjunto de práticas de direção simples que proporcionam um *framework*, ao invés de um conjunto rígido de instruções. Seguindo essas práticas o gerente se torna um líder adaptativo – ajustando a direção, estabelecendo regras simples e produtivas do sistema e encorajando *feedback*, adaptação e colaboração constantes. Este *framework* de gerenciamento provê times implementando metodologias ágeis com [31]:

- Habilidade de lidar com a mudança;

- Visão da organização como um sistema adaptativo fluido composto por seres vivos inteligentes;
- Reconhecimento dos limites do controle externo em estabelecer ordem, e do papel de controle inteligente que a auto-organização emprega como um meio de estabelecer ordem.

Segundo [7], *Agile Project Management* é o trabalho de energizar, capacitar, e habilitar times de projetos para entregar, rapidamente e confiavelmente, valor de negócio através do engajamento de clientes, da aprendizagem e adaptação contínua para suas necessidades de mudança e ambiente. Em sua abordagem, [3] propõe um conjunto de princípios e práticas para o gerenciamento ágil que serão descritas a seguir.

2.4.1.1 Princípios do *Agile Project Management*

Para ser sustentável em face de mudanças, uma metodologia precisa ter como base um núcleo de princípios e valores simples e imutável e em sua aplicação. Ela deve permitir práticas flexíveis que são adaptáveis para mudar ambientes [31]. Com base nesta afirmação, APM baseou-se nos conceitos de Sistemas Adaptativos Complexos para derivar os seguintes princípios:

- **Estimular alinhamento e cooperação.** Pessoas são consideradas os agentes primários direcionando valor, mudança, aprendizado e adaptação. A visão compartilhada mantém as pessoas alinhadas e atuando em direção a objetivos comuns. Quando as pessoas estão em alinhamento, elas evitam competição e cooperam para trabalhar com as outras para um ganho mútuo.
- **Encorajar a inovação e auto-organização.** Processos e práticas são mantidos minimamente simples. Pessoas se auto-organizam para entregar valor de negócio. Padrões complexos, incluindo comportamento auto-organizado e estrutura ótima, emergem de interações próximas entre muitas pessoas seguindo regras simples.
- **Instituir aprendizado e adaptação.** Feedback é usado para aprendizado, adaptação, e melhoria contínuos. Projetos operam em um limite entre o caos e a ordem, onde existe controle, estrutura, otimização e exploração suficientes.

2.4.1.2 Práticas do *Agile Project Management*

As práticas da APM priorizam a entrega de negócios de valor e a satisfação do para o cliente. Essas práticas não têm um significado completamente compreensível e não especificam tudo que

um gerente precisa fazer em cada projeto para gerenciá-lo de forma ágil. Contudo, por serem baseadas na teoria dos Sistemas Adaptativos Complexos, nos princípios da APM, e em *Extreme Programming*, elas apresentam um estilo de gerenciamento flexível. Assim sendo, as práticas especificam diferentes atividades que devem ser selecionadas de acordo com as necessidades de projeto. A Tabela 2.4 traz um resumo das práticas de APM e de como elas surgiram através da evolução de Sistemas Adaptativos Complexos e XP.

Tabela 2.4 Evolução de Sistemas Adaptativos Complexos/XP para práticas APM [33]

Princípios de Sistemas Adaptativos Complexos	<i>Extreme Programming</i>	Práticas em <i>Agile Project Management</i> Correspondentes
Agentes autônomos		
Agentes mantêm modelos internos que direcionam seu comportamento.	Valores XP servem como um modelo interno.	#1 Visão Direcionadora. Reconhece e cultiva uma visão compartilhada de projeto como um modelo interno e o traduz em uma poderosa influência no comportamento do time. #2 Trabalho e Colaboração em Equipe. Times pequenos e dinâmicos com membros com capacidade de tomar decisões formam a base para ricas interações e cooperação entre a equipe. #3 Toque Leve. Controle inteligente de times requer uma mistura delicada de imposição e ordem emergente.
Diversidade de habilidades entre agentes contribui para inovação e auto-organização.	Práticas colaborativas possibilitam diversidade de habilidade/experiência.	
Estratégia dita cooperação sobre competição.	Teoria de jogo proporciona atividades de otimização em iteração e plano de <i>release</i> .	
Construção de blocos provê as abstrações necessárias para organizar o ambiente.	Valores e práticas XP oferecem um conjunto simples de conceitos para coordenar mais projetos.	
Interações entre agentes		
Regras locais e estratégicas suportam agregação e surgimento de uma equipe.	Valores e práticas XP formam a base para o comportamento complexo.	#4 Regras Simples. Regras Simples como as práticas XP suportam o comportamento complexo da equipe. #5 Informação Aberta. Informação Aberta permite que times se adaptem e reajam a condições de mudanças no ambiente. #6 Vigilância Ágil. Liderança visionária requer monitoração contínua, aprendizagem e adaptação ao ambiente.
Ordem de inovação é uma manifestação <i>bottom-up</i> , enquanto impor ordem é uma manifestação <i>top-down</i> .	Regras simples, práticas de codificação disciplinadas e redução de hierarquia levam a auto-organização, arquitetura/design emergente, e estabilidade em face de mudanças.	
<i>Feedback</i> proporciona mudança e adaptação.	<i>Feedback</i> constante através de rastreamento, <i>releases</i> constantes, programação em pares, e reuniões diárias (<i>standup meetings</i>) proporcionam mudança e adaptação.	
Sistemas dinâmicos não-lineares passam a se adaptar continuamente quando eles alcançam um estado de equilíbrio (<i>the edge of chaos</i>).	Mudanças gerais podem ser realizadas rapidamente utilizando <i>unit tests</i> , <i>refactoring</i> , e integração contínua.	

O *framework* proposto por [7] apresenta seis práticas: Trabalho e Colaboração em Equipe, Visão Direcionadora, Regras Simples, Informação Aberta, Toque Leve e Vigilância Ágil.

2.4.2 *Extreme Project Management*

Na abordagem ágil, um projeto é visto como um campo de energia localizada que compreende um conjunto de pensamentos, emoções, e interações continuamente se expressando na forma física [8]. A energia da equipe é a capacidade de realizar trabalho. A intensidade de seus pensamentos, emoções, e interações determinarão a capacidade de realizar as atividades dentro do projeto. Os pensamentos são expressos através das idéias, decisões e fatos. Emoções e interações estão interligadas e se referem ao humor dos membros de uma equipe e a comunicação que se dá entre eles.

Neste contexto, gerenciar projetos significa, acima de tudo, gerenciar pessoas e as relações entre elas. Uma outra definição seria a que considera gerenciamento de projetos como a arte e a ciência de facilitar e gerenciar o fluxo de pensamentos, emoções, e interações de forma que sejam produzidos resultados de valor [8].

Extreme Project Management (XPM), assim como as outras metodologias de gerenciamento ágil de projetos, possui forte influência do Manifesto Ágil e das metodologias ágeis de desenvolvimento de software. Um projeto extremo é um tipo especial de energia localizada, é um empreendimento complexo, de alta velocidade e que se corrige ao longo do tempo da sua execução [8].

Durante um projeto extremo, as pessoas envolvidas interagem em busca de resultados desejáveis em meio a altos níveis de incerteza, mudança e estresse. Projetos desta natureza caracterizam-se por serem realizados em curto espaço de tempo e por possuírem requisitos que mudam constantemente. A equipe deve se sentir desafiada e ao mesmo tempo motivada a seguir um projeto deste tipo, pois o grau de incerteza é alto e a mudança é a única constante.

Um projeto extremo é um processo de descoberta onde o conteúdo da última entrega e o caminho para se chegar a ela evoluirão durante o tempo de vida do projeto [8].

XPM significa gerenciar a energia do projeto sob condições turbulentas e complexas: aquelas que se caracterizam pela exigência de alta velocidade no desenvolvimento, mudanças constantes, alto nível de incerteza e de estresse [8]. Esta abordagem contribui para o sucesso de um projeto de três formas. Primeiro reconhece que não se gerencia o desconhecido e o imprevisível da mesma forma que se gerencia o conhecido e previsível. Em segundo lugar, tem foco em ganhar e manter o

comprometimento com a missão do projeto estimulando a confiança entre os *stakeholders*. Por fim, é muito mais que uma metodologia ou um conjunto de ferramentas de software e *templates*. É uma abordagem holística, centrada nas pessoas, humanística, focada no negócio e baseada na realidade.

O modelo de gerenciamento de XPM consiste de um conjunto de princípios, valores, habilidades, ferramentas e práticas que são compatíveis com a mudança e incerteza e são a liga do gerenciamento extremo [8]:

- **4 aceleradores** – princípios que desencadeiam motivação e inovação;
- **10 valores compartilhados** – um conjunto de valores que estabelecem confiança entre os *stakeholders*;
- **4 questões de negócio** – questões que, quando endereçadas, asseguram que os clientes recebam valor mais cedo e com maior frequência;
- **6 fatores críticos de sucesso** – as habilidades, ferramentas e suporte organizacional essenciais para o sucesso.

2.4.2.1 Aceleradores

Os quatro aceleradores aumentam a velocidade do trabalho, estimulam a execução do projeto, e tornam os *stakeholders* e o time adaptativo. Manter estes princípios direcionadores em mente levará a um time motivado e criativo que deverá exercer liderança participativa e colaborar entre si [8].

Segundo [8], os aceleradores de XPM são os seguintes:

- **Faça da mudança sua amiga.** Mudança tem uma conotação negativa, pois dá a idéia de atrapalhar as coisas. Provavelmente não será bem vinda em ambientes de Gerenciamento Clássico. Em XPM existe uma atitude diferente em relação às mudanças: elas representam oportunidade, o que significa aceitá-las e seguir em frente.
- **Baseie-se no desejo das pessoas para fazer a diferença.** Este acelerador está associado à motivação da equipe. Equipes quando motivadas, especialmente por projetos que envolvem novas tecnologias e desafios, têm tendência a produzir mais e melhor. Quando colocado em prática esse segundo acelerador significa mostrar às pessoas o quanto o trabalho delas contribui para o projeto.

- **Crie a noção de propriedade para obter resultados.** As pessoas abraçam uma idéia quando fazem parte da sua criação. Na prática este princípio significa confiar na experiência e comprometimento das pessoas, e proporcionar a oportunidade de influenciar como será o progresso do projeto, incluindo como a performance será medida.
- **Mantenha a simplicidade.** Simplicidade, neste contexto, significa realizar as atividades de forma simples possível e só aquelas que forem realmente necessárias. Segundo este acelerador menos representa mais: menos processos, menos *overhead* de gerenciamento de projetos, menos políticas e procedimentos operacionais padrão, mais resultados.

2.4.2.2 Valores Compartilhados

É o sistema de valores que encoraja uma forte convicção entre os *stakeholders* de que trabalhando juntos eles terão sucesso, mesmo em face à volatilidade e adversidade [8].

Os valores compartilhados retratam pessoas, processos e negócios. Os valores de pessoas são: Pessoas primeiro, Comunicação honesta, Qualidade de vida e Coragem. Os valores de processo são os seguintes: Colaboração do cliente, Falhas rápidas e Visibilidade. Finalmente, os valores de negócio são: Clareza do propósito, Orientação para resultados e Valor o mais cedo possível.

2.4.2.3 Questões de negócio

Servem como constante lembrete para todos os *stakeholders* que o projeto é em primeiro lugar uma iniciativa de negócio que possui o objetivo de entregar valor iterativamente. As questões de negócio são as seguintes:

- Quem precisa do que e por quê?
- O que terá que ser feito para este fim?
- Nós podemos conseguir o que nos propomos a fazer?
- Vale à pena?

2.4.2.4 Fatores Críticos de Sucesso

Estes fatores empregam os aceleradores, valores compartilhados e questões de negócio, construindo-os na vida do projeto. Habilidades, práticas, e métodos são essenciais para liderar, planejar, gerenciar e rastrear o projeto do começo ao fim e assimilar a mudança ao longo do caminho.

- **Auto-controle.** É a prática de liderar a si próprio, apesar do estresse que é oriundo de projetos com características extremas.
- **Liderança pelo comprometimento.** O gerente de um projeto extremo deve fazer com que a equipe toda se comprometa, estimular motivação e inovação, estabelecer confiança e segurança para que o projeto obtenha sucesso.
- **Modelo de projeto flexível.** Tem o objetivo de proporcionar disciplina suficiente para dar às pessoas a liberdade de inovar e realizar o trabalho necessário. É iterativo e possui quatro ciclos: **Visionar** (trata da definição do escopo), **Especular** (é um planejamento em alto nível para identificar alguns *milestones* e entregáveis), **Inovar** (é a ação realizada dentro de *time frames* predefinidos), **Reavaliar** (é a revisão realizada utilizando as quatro questões de negócio). **Disseminar** significa colher os resultados do que foi realizado.
- **Comunicação em tempo real.** Em projetos extremos coisas acontecem rápido por isso a comunicação deve ser constante e pode ser realizada através de fóruns, documentos compartilhados, *emails*, *standup meetings*. A informação deve estar disponível a qualquer hora para qualquer pessoa.
- **Organização ágil.** É uma organização tolerante a mudanças que tem o objetivo de fazer com que as entregas possuam o resultado pretendido pelo cliente.

2.4.3 Scrum

Criado por Ken Schwaber e Jeff Sutherland em 1996, o Scrum se adequa a ambientes voláteis [9]. É uma abordagem empírica baseada na flexibilidade, adaptabilidade e produtividade. Scrum é diferente das metodologias tradicionais nas quais um projeto é tratado como uma tarefa ao invés de subconjuntos de tarefas, com progresso acompanhado durante a sua execução.

Scrum enfatiza o gerenciamento de projetos, ficando a escolha da metodologia de desenvolvimento a cargo dos programadores. Existem atividades específicas de monitoramento e *feedback*, em geral, reuniões diárias de 15 minutos com toda a equipe, visando à identificação e correção de quaisquer deficiências ou impedimentos no processo de desenvolvimento [9].

Por não se caracterizar como um processo prescritivo, Scrum não descreve o que deve ser feito em todas as ocasiões. É indicado para trabalhos complexos nos quais é impossível prever tudo que irá ocorrer. Scrum oferece um *framework* e um conjunto de práticas que mantêm o processo

visível. Isto permite aos seus praticantes saber exatamente o que está acontecendo e fazer os ajustes necessários para manter o projeto indo em direção aos seus objetivos.

De acordo com [9], os princípios chave do Scrum são:

- Equipes pequenas de trabalho, buscando a maximização da comunicação e da troca de conhecimento tácito e informal e minimização de *overhead*;
- Adaptação às solicitações de mudanças técnicas ou do cliente/usuário, assegurando a entrega do melhor software possível;
- Entregas freqüentes de versões que podem ser testadas, ajustadas, executadas, documentadas e liberadas para produção;
- Divisão do trabalho e das responsabilidades da equipe de projeto em pequenas entregas;
- Habilidade de entregar um software pronto quando da necessidade do cliente ou do negócio.

As iterações Scrum recebem o nome de *Sprints* e duram de duas a quatro semanas. Antes de cada *Sprint*, reuniões de planejamento da iteração chamadas de *Sprint Planning Meetings* são realizadas entre o time e o cliente para definição e priorização de requisitos. Para realizar o acompanhamento do projeto são realizadas as Scrum *Daily Meetings*, que são reuniões curtas, com duração de no máximo 15 minutos e com o objetivo de fazer com que o time responda a três perguntas: O que foi feito desde ontem? O que se planeja fazer amanhã? Existe algum impedimento na realização desta atividade? Graficamente este acompanhamento é representado através do *Sprint Burndown*. Ao final de uma *Sprint* são realizadas duas reuniões: *Sprint Review* para demonstração do produto gerado e *Sprint Retrospective*, que é uma reunião de lições aprendidas.

Durante o planejamento da *Sprint*, o time estuda os requisitos, considera a tecnologia disponível, e avalia suas habilidades. Então, coletivamente, determina como construir a funcionalidade, modificar sua abordagem diariamente quando encontra novas complexidades, dificuldades e surpresas. O time descobre o que precisa ser feito e seleciona a melhor forma de fazer.

A Figura 2.3 apresenta uma visão geral da metodologia Scrum. Um projeto Scrum começa com o desenvolvimento de um *Product Backlog*, que é uma lista de requisitos. O *Product Owner* é responsável pelo conteúdo, priorização e disponibilidade do *Product Backlog*. Depois de definida, a lista de requisitos é priorizada de acordo com a necessidade apontada pelo *Product Owner* e o nível de dificuldade previsto para sua execução. Os itens de um *Product Backlog* podem ser requisitos funcionais, requisitos não-funcionais, e problemas [13]. Qualquer idéia ou requisito pode ser

adicionado à lista de funcionalidades a ser desenvolvida, mas deve haver uma priorização desses requisitos e esta ação deve ser construída juntamente com o desenvolvimento da compreensão da solução do problema. As mudanças que ocorrerem serão acompanhadas através de contínua reavaliação e re-priorização.

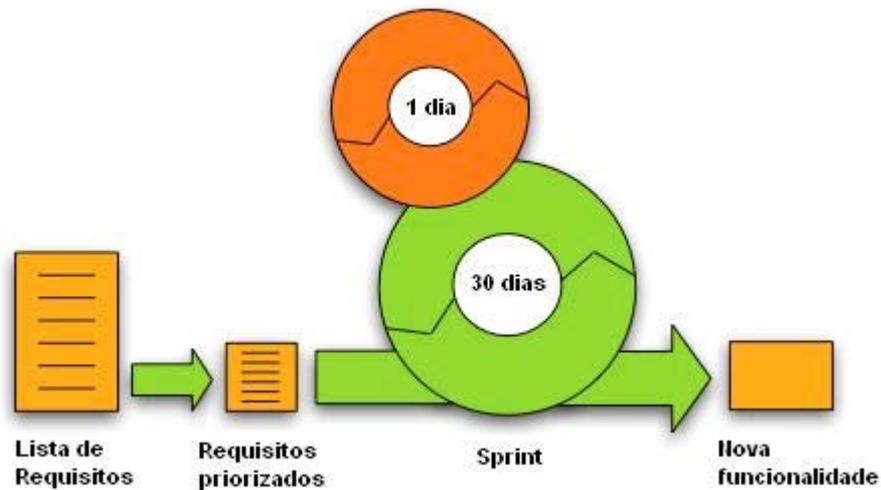


Figura 2.3 A metodologia Scrum

2.4.3.1 Papéis no Scrum

Scrum é implementado por três papéis: *Product Owner*, *Scrum Team* e *ScrumMaster*. Todas as responsabilidades de gerenciamento são divididas entre estes papéis.

O *Product Owner* é responsável por representar os interesses de cada um no projeto e no seu sistema resultante. O *Product Owner* é responsável usar o *Product Backlog* para assegurar que a funcionalidade de mais valor seja implementada primeiro. Isto é feito através da priorização do *Product Backlog*.

O *Scrum Team* é responsável pelo desenvolvimento das funcionalidades. O *ScrumMaster* é responsável pelo processo, por ensinar Scrum para os envolvidos no projeto, por implementar Scrum para que se adeque à cultura da organização e ainda entregue os benefícios esperados, e para assegurar que todos sigam as regras e práticas do Scrum.

2.4.3.2 Artefatos do Scrum

Os artefatos sugeridos pelo Scrum são [9]: *Product Backlog*, *Impediment Backlog*, *Sprint Backlog* e o Incremento de Funcionalidade de Produto.

Os requisitos para o sistema ou produtos sendo desenvolvidos pelo projeto são listados no *Product Backlog*. O *Product Owner* é o responsável pelos conteúdos, priorização e disponibilidade desta lista de requisitos que é usado no plano de projeto apenas como uma estimativa inicial dos requisitos. O *Product Backlog* evolui de acordo com a evolução do produto e do ambiente no qual ele será usado.

A *Sprint Backlog* corresponde ao trabalho ou tarefas que um time define para uma iteração. É composta por funcionalidades escolhidas dentre as listadas no *Product Backlog*. O time compila uma lista inicial destas tarefas na segunda parte da *Sprint Meeting Planning* que devem ser divididas de modo que levem de 4 a 16 horas para serem concluídas.

Qualquer coisa que atrapalha a produtividade e qualidade de um projeto Scrum é um impedimento. É de responsabilidade do *ScrumMaster* remover qualquer impedimento para o *Scrum Team* executar a realização dos itens do *Product Backlog*. *Impediment Backlog* é um conjunto de tarefas que o *ScrumMaster* utiliza para rastrear os impedimentos que precisam ser resolvidos. Contém um conjunto dos itens que impedem o progresso do projeto e geralmente estão associados a riscos.

2.4.4 Agile Project Management Framework

Agile Project Management Framework ou *APM Framework* consiste de mais previsão e exploração do que planejamento e execução – ela força o confronto entre a realidade dos projetos e o ambiente de desenvolvimento de produto altamente volátil. Quando o objetivo do negócio é inovação confiável, então o processo precisa de um *framework* orgânico, flexível, fácil de adaptar e que possua as seguintes características [3]:

- Suporte a visão, exploração, e cultura de adaptação;
- Suporte a auto-organização e times autodisciplinados;
- Promover a confiabilidade e a consistência possível para extensão dado o nível da incerteza do projeto;
- Ser flexível e fácil de adaptar;
- Suportar visibilidade no processo;
- Incorporar aprendizado;
- Incorporar práticas que suportam cada fase;
- Prover pontos de verificação de gerenciamento para revisão.

Levando em consideração essas características, *APM Framework* foi dividido em fases que são apresentadas na Figura 2.4 e descritas em seguida:

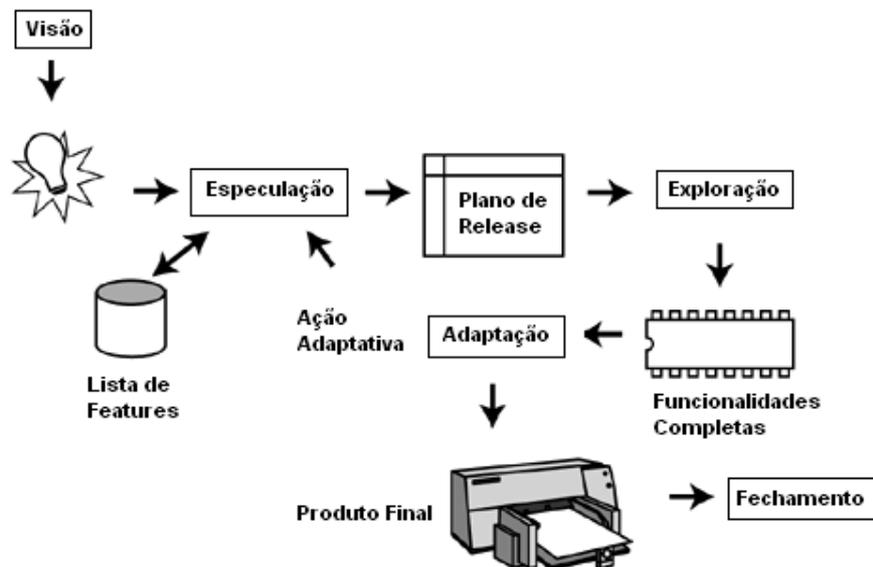


Figura 2.4 *Agile Project Management Framework* [3]

A fase de Visão determina que é preciso prever o que será entregue, ou seja, construir uma visão do produto e do escopo do projeto. Os envolvidos no projeto também fazem parte deste planejamento: a comunidade de clientes, os gerentes de produto, os membros do time de projeto e demais *stakeholders*. Por fim, os membros do time de projeto precisam prever como eles pretendem realizar o trabalho do projeto juntos. Ao término da fase de Visão, o laço geralmente será Especular-Explorar-Adaptar, com cada iteração refinando sucessivamente o produto. Contudo, visitar periodicamente a fase de Visão pode ser necessário quando o time reúne informação nova.

Depois de definido, o projeto passa por uma fase de Especulação que se caracteriza pela suposição com base em fatos ou informação incompleta. Isto significa que esta é uma fase de extensão da fase de Visão e interage com ela. Ela ajuda a complementar o que foi previamente definido. Consiste em: reunir os principais requisitos iniciais para o produto, definir a carga de trabalho como uma lista de funcionalidades do produto (*feature backlog*), criar um plano de entrega que inclui cronograma e alocação de recursos para desenvolver estas funcionalidades, incorporar estratégias de mitigação de riscos ao plano, estimar custos de projetos e gerar outras informações administrativas e financeiras requeridas.

A fase de Exploração vem em seguida e é responsável pelas funcionalidades do produto. Da perspectiva do gerenciamento de projetos existem três áreas de atividade críticas durante esta fase. A primeira é a entrega de funcionalidades planejadas através do gerenciamento da carga de trabalho e

uso de práticas apropriadas e estratégias para mitigar riscos. A segunda consiste na criação de uma comunidade de projeto auto-organizada e colaborativa a qual é de responsabilidade de cada um, mas é facilitada pelo gerente de projeto. Finalmente, a terceira atividade diz respeito ao gerenciamento das interações do time com clientes, gerenciamento de produto e gerenciamento de outros *stakeholders*.

Adaptação diz respeito a controle e correção: elaboração de planos, monitoramento de resultados e realização de correções. Esta fase implica em modificação ou mudança, e não encara os resultados durante o projeto como sucesso ou fracasso. Nos projetos que são guiados pela filosofia que determina que responder a mudanças é mais importante que seguir um plano, atribuir o fracasso à variação do plano não é produtivo. É importante observar que um processo puramente ad hoc falha em aprender com seus erros, os quais são peças chave do APM *framework* quando se considera a incorporação e retenção de lições aprendidas. Os resultados desta fase são revisados e analisados em termos do que foi realizado versus o que foi planejado. Os resultados da adaptação são alimentados em um esforço de replanejamento para começar a próxima iteração.

Quando o projeto sai do laço Especular-Explorar-Adaptar, inicia-se a fase de Fechamento. Projetos são parcialmente definidos pela presença de um início e de um fim, e assim sendo, o objetivo desta fase e do mini fechamento que é realizado ao final de cada iteração é a incorporação da aprendizagem ao trabalho da próxima iteração ou passá-la para o próximo time de projeto.

2.5 Gerenciamento Ágil de Projetos e o PMBOK® *Guide*

De acordo com os princípios e práticas de gerenciamento ágil, é possível observar características e distinguir fases que se assemelham ao grupo de processos definido pelo PMBOK® *Guide*. Uma comparação entre estes processos e o Gerenciamento Ágil realizada por [4] é apresentada, conforme mostrado na Tabela 2.5.

No mesmo trabalho, [4] afirma que a grande diferença entre o PMBOK® *Guide* e as fases identificadas no gerenciamento ágil está no planejamento e no controle. Quando define as fases do *Agile Project Management Framework* sendo compatíveis com os processos do Gerenciamento Clássico, [3] concorda com esta afirmação.

Tabela 2.5 Gerenciamento Ágil comparado aos grupos de processos do PMBOK® *Guide* [4]

PMBOK® <i>Guide</i>	Gerenciamento Ágil
Iniciação: Autorização de um novo projeto ou fase e definição do escopo preliminar.	Iniciação: Foco em identificar as necessidades de negócio para o desenvolvimento de novos produtos.
Planejamento: Planejamento integral e detalhado do projeto.	Planejamento: Planejamento inicial do projeto, seguido pelo planejamento separado para cada iteração.
Execução: Execução do plano de projeto.	Execução: Desenvolvimento por iteração do conjunto de funcionalidades identificado.
Monitoramento e Controle: Ênfase no gerenciamento de controle de mudanças para minimizar os impactos do projeto.	Monitoramento e Controle: Abertura para mudanças no escopo durante o processo com limitações nas mudanças no decorrer das iterações.
Encerramento: Aceitação formal.	Encerramento: Aceitação do cliente a cada iteração.

O gerenciamento ágil volta sua preocupação para a execução que objetiva a entrega rápida de funcionalidades ao cliente e ao controle, no sentido de realizar adaptação às mudanças, o que permite a realização de alterações no escopo a cada iteração [3], [4], [5]. As mudanças no escopo geralmente não ocorrem durante a execução da iteração, mas sim antes de cada iteração. Gerenciamento Ágil de Projetos e Gerenciamento Clássico de Projetos podem ser suportados pelos conceitos do PMBOK® *Guide*, que se adapta às suas características específicas, como pode ser visto na Tabela 2.6.

É possível observar, através da Tabela 2.6, que a abordagem ágil é mais adequada para projetos definidos em alto nível, com objetivos e requisitos não tão claros e foco nos valores empresariais – principalmente custos e prazos – necessitando de grande participação do cliente durante todo o processo. Também é adequada para o detalhamento do trabalho tático necessário para completar um projeto, bem como para mudanças mínimas requeridas na manutenção de sistemas ou outros produtos de trabalho discretos [2].

Considerando as abordagens ágeis apresentadas, de maneira geral, *Agile Project Management* apresenta um alto grau de preocupação com as pessoas e os relacionamentos entre elas, a equipe deve estar trabalhando em prol de objetivos comuns e ter uma visão compartilhada do projeto, processos e práticas devem ser os mais simples, o *feedback* deve ser constante e a informação compartilhada entre toda a equipe, os membros da equipe devem ter capacidade de tomar decisões, e regras simples devem ser encorajadas.

Tabela 2.6 Áreas de conhecimento do gerenciamento ágil e do gerenciamento clássico de projetos [4]

Áreas de Conhecimento	Gerenciamento Clássico de Projetos	Gerenciamento Ágil de Projetos
Gerenciamento da Integração do Projeto	Assegura a coordenação dos vários elementos do projeto.	A necessidade de coordenação formal é limitada devido ao uso reduzido de processos em vários elementos.
Gerenciamento do Escopo do Projeto	Assegura que o projeto contenha apenas o trabalho necessário para completá-lo de forma bem sucedida. Foco na definição e controle do que está ou não no projeto. Utiliza um rígido processo gerenciamento de mudanças.	O escopo é fixo apenas quando as iterações estão em andamento. Nenhum controle formal do escopo é necessário.
Gerenciamento do Tempo do Projeto	Foco na definição das atividades para o cronograma do projeto e controle de cronograma para assegurar a finalização do projeto no prazo.	O prazo estabelecido por iteração é fixo. Foco está na entrega de valor (funcionalidades) o mais rapidamente possível. O cronograma geral é baseado em funcionalidades e não em atividades.
Gerenciamento dos Custos do Projeto	Determina o orçamento baseado nos recursos necessários para o projeto e assegura que o projeto será encerrado dentro do orçamento aprovado.	Determina o orçamento baseado em funcionalidades requisitadas. Os recursos, as funcionalidades e os prazos são balanceados e existe uma preocupação em medir o custo por atraso.
Gerenciamento da Qualidade do Projeto	Assegura que o projeto vai satisfazer as necessidades para as quais foi concebido. Foco na conformidade e na adequação às especificações.	O sucesso do projeto é definido pelo cliente, que também apresenta seu <i>feedback</i> ao final de cada iteração. Foco na realização do propósito do produto e na adequação ao uso.
Gerenciamento dos Recursos Humanos do Projeto	Processos para que se faça o uso mais efetivo das pessoas envolvidas no projeto.	Foco na equipe e não no indivíduo. Os incentivos são baseados na produtividade do grupo.
Gerenciamento das Comunicações do Projeto	Assegura a geração, a coleta, a disseminação e o armazenamento periódicos das informações do projeto.	Foco na eliminação de gastos, funcionalidades desnecessárias e documentação. Acesso aberto à informação para todos os envolvidos.
Gerenciamento dos Riscos do Projeto	Foco na identificação, na análise e na proposição de respostas aos riscos do projeto.	As metodologias ágeis tratam o gerenciamento de riscos de maneira diferente. Não existe uma abordagem ágil comum. A preocupação está no escopo.
Gerenciamento das Aquisições do Projeto	Foco na aquisição de produtos ou serviços externamente à organização executora, para a realização do projeto. Baseado no uso de contratos, melhores práticas, processos e procedimentos.	Segue os melhores princípios para aquisição de bens ou serviços, dando maior ênfase à colaboração do que à negociação de contratos.

Agile Project Management Framework apresenta uma concentração maior nas atividades de execução do que nas de planejamento, adaptando-se às mudanças; suporte a times auto-

disciplinados, flexibilidade e fácil adaptação, entregas freqüentes das versões do produto, suporta visibilidade no processo, provê pontos de verificação de gerenciamento para revisão.

Visibilidade do processo, simplicidade, comunicação diária realizada através de reuniões curtas, entregas freqüentes, iterações curtas, maior enfoque na execução, incentiva a cooperação da equipe e a autodisciplina, tomada de decisão coletiva, e a participação do cliente como membro da equipe são características das outras abordagens discutidas durante este trabalho.

Já *Extreme Project Management* considera gerenciamento de projetos como gerenciamento de pessoas, baseia-se na motivação da equipe, simplicidade, comunicação honesta, participação do cliente, entrega de valor iterativa, flexibilidade do modelo de projeto traduzida na liberdade dada a equipe para inovar.

As fases básicas de um projeto de desenvolvimento ágil não são diferentes das fases de qualquer outro projeto. Ainda é necessário definir e iniciar o projeto, planejar o projeto, executar um plano, monitorar e controlar os resultados. A maneira pela qual estes passos são realizados na abordagem ágil, porém, é diferente e requer do gerente de projeto uma reavaliação do conhecimento sobre gerência tradicional, valores e práticas ágeis, visando incorporá-los ao seu próprio estilo de gerência, de forma a acrescentar valor aos projetos [2].

Conforme a Tabela 2.6 exhibe, as áreas de conhecimento do PMBOK® *Guide* podem ser adaptadas ao gerenciamento ágil. Em comparação com as abordagens vistas, o PMBOK® *Guide* não encoraja a simplicidade, mas também não a condena. O gerente possui controle total do projeto e os processos possuem um nível de complexidade maior do que os sugeridos pelo gerenciamento ágil e a comunicação é formal.

As fases do *Agile Project Management Framework* se assemelham aos processos do Gerenciamento da Integração do PMBOK® *Guide*, apresentando quase que os mesmos objetivos. Esta abordagem apresenta um ciclo de vida do gerenciamento do desenvolvimento organizado de forma que as atividades de gerenciamento podem ser concentradas em cada uma das fases, simplificando o gerenciamento do projeto. O Scrum apresenta um ciclo de vida simplificado, que enfatiza o desenvolvimento do produto, pouco se assemelhando nesse sentido ao guia PMBOK®.

Por serem mais centradas no desenvolvimento, as metodologias ágeis aparentam minimizar o papel da gerência em assegurar sucesso. Abordagens gerenciais como APM e XPM vieram complementar as práticas de metodologias ágeis existentes e preencher este *gap*, com diretrizes claras

para a liderança de projetos e introdução de práticas efetivas de gerência que requerem criatividade, flexibilidade e atenção às qualidades específicas e interações entre os membros da equipe [2].

Além disso, muitas das práticas de gerência de projeto tradicionais ainda se aplicam a projetos de desenvolvimento ágeis – com alguma adaptação e com uma forte dose de liderança. Por sua natureza adaptativa e foco em resultados, a abordagem ágil tem demonstrado atender melhor às necessidades das partes envolvidas com a construção de software [2]. Ela requer, porém, *stakeholders* dispostos a mudar o modo como as pessoas trabalham e pensam.

2.6 Considerações

Este capítulo apresentou o que é o Gerenciamento Ágil de Projetos de Software, algumas abordagens de Gerenciamento Ágil (*APM Framework*, APM, Scrum e XPM), e realizou uma comparação entre ele e o PMBOK® *Guide*, aqui considerado como um representante do Gerenciamento Clássico de Projetos.

O Gerenciamento Ágil de Projetos caracteriza-se por um planejamento rápido, com reuniões curtas, comunicação informal, participação intensiva de todos os envolvidos no projeto, alto grau de colaboração e a rápida incorporação de mudanças durante a execução do projeto. Segundo apresentado anteriormente, a quebra de paradigma em relação ao desenvolvimento de aplicações apresenta abordagens ágeis de gerenciamento como alternativa para projetos com duração curta e requisitos que mudam durante o projeto.

Planejamento é um passo crítico tanto para o gerenciamento clássico quanto para o gerenciamento ágil, e por este motivo seu uso não deveria ser concentrado no início do projeto, que é quando menos se sabe a respeito do que será realizado. Ao invés disso, o planejamento precisa ser iterativo e progressivo durante o projeto, guiado por um *feedback* do desempenho atual do projeto, mudanças no negócio e crescimento do conhecimento técnico [29].

A maior diferença entre o PMBOK® *Guide* e as metodologias ágeis fica visível nos processos de planejamento e controle. Os processos ágeis de planejamento priorizam iterações frequentes a fim de que se complete o plano de projeto. Já no PMBOK®, os processos de controle asseguram que os objetivos do projeto são encontrados através de frequentes monitoramento e medição do progresso para identificar variações do plano, e assim tomar ações corretivas quando necessário. Nas metodologias ágeis, o escopo é definido por iteração, e depois de cada iteração, funcionalidades podem ser adicionadas ou removidas. Iterações são utilizadas como um mecanismo de controle para

ajudar a tornar mais fácil a mudança de curso durante o projeto, assim como, prover alguma estabilidade, pois mudanças não são realizadas durante as iterações.

O Gerenciamento Clássico de Projetos é criticado por ser utilizado de forma que dificulta sua adaptação a projetos mutáveis. Em contrapartida, as metodologias de gerenciamento ágil vêm ganhando popularidade pela sua adaptabilidade. Apesar disto, as metodologias clássicas ainda apresentam benefícios quando se trata principalmente de grandes projetos e daqueles que possuem restrições de confiabilidade.

O PMBOK[®] *Guide*, mesmo sendo considerado uma referência quando se trata de Gerenciamento Clássico de Projetos, pode ser adaptado e servir como um guia também para os praticantes do Gerenciamento Ágil de Projetos. O nível de organização que o guia sugere faz com que o trabalho seja orientado a processos e áreas de conhecimento. Isto pode garantir que algum rigor seja dado a projetos que adaptem as práticas de Gerenciamento Clássico, apresentadas no guia, à agilidade em gestão.

É importante salientar que a abordagem ágil não é estritamente uma metodologia, mas uma forma de visão e um conjunto de atitudes, valores e princípios. O objetivo principal da abordagem ágil é simplificar o processo de desenvolvimento, minimizando e dinamizando tarefas e artefatos que são importantes no processo, além de priorizar o software funcionando [2], conforme sugere o Manifesto Ágil.

A tendência, de acordo com [3] e [5], entre outros autores, é que surja um modelo híbrido entre clássico e ágil, a fim de unir as vantagens de ambas as abordagens, em prol do desenvolvimento de projetos de sucesso na visão do cliente e da equipe de projeto. É provável que esta nova abordagem não ocupe o espaço do modelo de gerenciamento clássico por completo. É esperado que os modelos se complementem e assim, de forma híbrida, trabalhem para o aprimoramento do processo de gerenciamento e de desenvolvimento de software.

Tendo em vista as considerações desta seção, este trabalho se propõe a antecipar essa tendência e desenvolver uma possível solução para o problema em questão. Esta solução, o Agilius, é uma abordagem de gerenciamento ágil de projetos baseada no PMBOK[®] *Guide*, e será descrita em maiores detalhes no próximo capítulo.

Agilius²: Uma Abordagem Ágil de Gerenciamento de Projetos de Software

Através de um estudo mais aprofundado do PMBOK[®] *Guide* e das metodologias ágeis de gerenciamento de projetos, foi possível conhecer as características e processos aplicados na gerência dos projetos de uma forma mais ampla. Este capítulo apresenta o Agilius, abordagem ágil de gerenciamento de projetos que este trabalho propõe e justifica o uso do *Eclipse Process Framework Composer* na sua modelagem.

3.1 Modelagem: O uso do *Eclipse Process Framework Composer*

Como visto no capítulo anterior, o Manifesto Ágil, assim como o Gerenciamento Ágil de Projetos incentiva documentação mínima, comunicação aberta, times auto-organizados e auto-gerenciáveis. No entanto, até mesmo os times auto-organizados precisam definir um processo que ofereça, no mínimo, orientação a respeito de como o desenvolvimento será definido durante o ciclo de vida, quando os *milestones* serão atingidos e verificados [34], e outros recursos auxiliares que proporcionem um maior planejamento e controle dos projetos. De acordo com o apresentado em [34], alguns dos problemas que os desenvolvedores enfrentam são:

- A falta de uma linguagem comum ou terminologia entre processos, o que ocasiona redundância e inconsistências;
- O conhecimento não pode ser facilmente personalizado para projetos diferentes ou para novas práticas;
- A ausência de uma comunidade central ou de um *framework* de comunicação para facilitar a convergência de melhores práticas através dos domínios.

A IBM procurou suprir essa lacuna quando propôs à *Eclipse Foundation*, em outubro de 2005, um projeto que inclui a doação de um subconjunto do *Rational Unified Process* [35]. A proposta teve o objetivo de construir um *framework open source* de processos chamado *Eclipse Process Framework Composer* [35]. Esta doação forneceu uma arquitetura de fundação e ferramentas baseadas na Web para a indústria projetar, comunicar, publicar e reusar melhores práticas no desenvolvimento de software.

² *Agilius* é uma palavra do latim que significa ágil.

O *Eclipse Process Framework Composer*, ou *EPF Composer*, é uma plataforma para engenheiros de processo, líderes de projeto, gerentes de projetos e de programas que são responsáveis por manter e implementar processos para organizações ou projetos individuais [36]. É um projeto de iniciativa *open source*. O projeto utiliza o *Eclipse Modelling Framework* para definir e gerenciar a sua implementação [37].

Com o objetivo de oferecer soluções para problemas comuns que líderes de desenvolvimento e times enfrentam ao adquirir e gerenciar seus métodos e processos, o *EPF Composer* possui duas finalidades principais [36]:

- Fornecer para os desenvolvedores uma base de conhecimento de capital intelectual que permite que se navegue, gerencie e insira conteúdo.
- Prover capacidades de engenharia de processos que suportem a seleção, adaptação e junção de processos, realizados por engenheiros de processo e gerentes, para o desenvolvimento de projetos.

Métodos de desenvolvimento são tipicamente descritos em livros, artigos, padrões e outras formas de documentação. É comum que estas fontes não forneçam explicações passo a passo para pontos particulares da documentação. O *EPF Composer* permite que sejam criados conteúdo e estrutura de forma específica e navegável por utilizar um esquema predefinido. Este esquema utiliza o *Software Process Engineering Metamodel 2.0*, ou *SPEM* [38], que auxilia na organização da grande quantidade de dados utilizada no desenvolvimento de métodos e processos [36].

Uma justificativa para o uso desta ferramenta na modelagem da abordagem proposta é que, assim como ressalta [36], uma grande vantagem do *EPF Composer* é a capacidade de criação de processos apropriados para ambientes ágeis. O uso deste ambiente para desenvolvimento de processos facilita a criação e manutenção de um *framework* ágil consolidado, incentiva o desenvolvimento ágil como tendência nas grandes organizações, faz com que seja inteligente e não intrusiva adaptação ao processo.

Ainda reforçando a escolha, em estudo de caso apresentado na documentação do *Software Process Engineering Metamodel* em sua versão 2.0, parte do *PMBOK® Guide* é modelada utilizando *EPF Composer*, o que confirma como indicado o uso desta ferramenta na modelagem da abordagem ágil proposta nesta dissertação.

3.2 Metodologia de Definição do Agilius

Para o desenvolvimento do Agilius, primeiramente foram estudadas características de diversas abordagens de gerenciamento ágil de projetos. A partir da investigação realizada no PMBOK® *Guide*, das nove áreas de conhecimento apresentadas no guia foram escolhidas seis áreas a serem utilizadas na realização desta proposta, conforme apresentado na Seção 1.1 desta dissertação.

Em seguida, técnicas e artefatos de gerenciamento ágil apresentados no Scrum, *Extreme Programming*, *Agile Project Management*, e *Agile Project Management Framework* foram escolhidos de acordo com os seguintes critérios:

- Fácil adaptação das técnicas aos artefatos largamente utilizados;
- Popularidade, pois muitas das técnicas e práticas já são utilizadas em ambientes de desenvolvimento;
- Obtenção rápida de resultados.

A Tabela 3.1 apresenta um mapeamento entre as técnicas, atividades e conceitos retirados de abordagens ágeis e do PMBOK® *Guide*. Algumas delas são utilizadas no Agilius como artefatos e outras serviram de base para o desenvolvimento de algumas atividades.

Tabela 3.1 Técnicas, Práticas e Conceitos Utilizados no Agilius

Técnicas, Práticas e Conceitos Utilizados no Agilius			
Scrum	XP	APM	PMBOK
<i>Sprint Planning Meeting</i>	<i>Story Points</i>	<i>Product Vision</i>	Áreas de Conhecimento
<i>Sprint Burndown Chart</i>	<i>Ideal Day</i>	<i>Tradeoff Matrix</i>	Relatórios de Desempenho
<i>Scrum Daily Meeting</i>	<i>Ideal Week</i>	<i>Feature Cards</i>	Conclusão de Contratos
<i>Sprint Retrospective</i>	<i>Planning Poker</i>	<i>Feature Breakdown Structure</i>	Lista de Recursos
<i>Release Retrospective</i>	<i>Small Releases</i>	<i>Exploration Factor Matrix</i>	Calendário de Recursos
<i>Project Retrospective</i>	<i>Customer Always Available</i>		Ações Corretivas
<i>Release Planning</i>	<i>User Stories</i>		Planos de Gerenciamento
<i>Product Backlog</i>	Contrato de Escopo Negociável		
<i>Sprint Backlog</i>			
<i>Sprint Burndown Charts</i>			
<i>Impediment Backlog</i>			

Por fim, as áreas de conhecimento e os processos foram definidos e implementados utilizando o EPF *Composer*, apresentado na seção anterior.

3.3 Visão Geral do Agilius

O Agilius é uma abordagem de Gerenciamento Ágil de Projetos, que possui como proposta a união entre Gerenciamento Clássico de Projetos e Gerenciamento Ágil de Projetos a fim de tornar mais eficientes os processos de gerenciamento de software, pois procura se adequar às características deste tipo de projetos.

Apresenta-se como uma alternativa para projetos que necessitam de rigor em sua execução, mas que buscam resultados ágeis traduzidos em entregas rápidas e de valor para o cliente. Possui fundamentação nas abordagens de gerenciamento ágil Scrum, *Agile Project Management*, *Agile Project Management Framework*, *Extreme Project Management* e algumas práticas da metodologia de desenvolvimento *Extreme Programming*. O PMBOK® *Guide* foi utilizado como base para o Agilius no que diz respeito às suas áreas de conhecimento e ao seu grupo de processos. Algumas das suas ferramentas e técnicas também foram utilizadas.

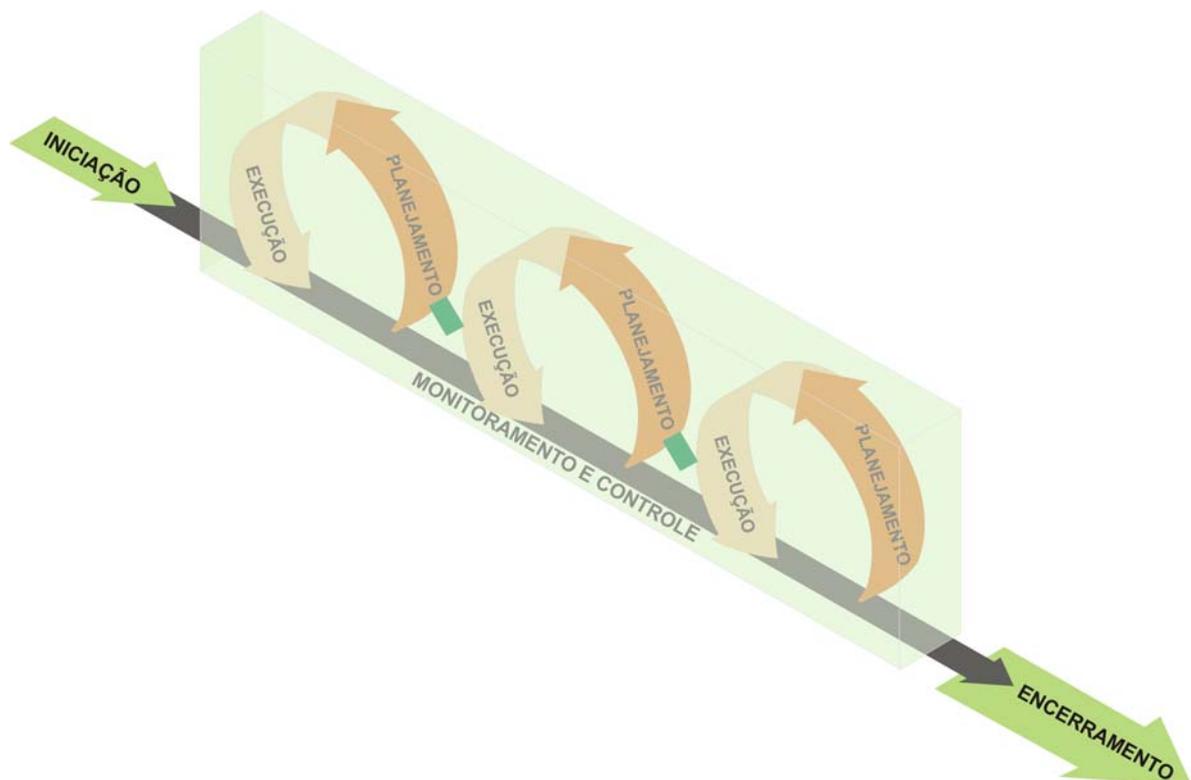


Figura 3.1 Grupo de processos do Agilius

Assim, o Agilius dividido em cinco fases: Iniciação, Planejamento, Execução, Monitoramento e Controle, e Encerramento. A Figura 3.1 apresenta o fluxo das fases do Agilius no

decorrer de um projeto. Para a fase de Iniciação é realizado um planejamento preliminar do projeto, um plano de projeto em alto nível e a equipe de projeto é escolhida. Em seguida, fases de Planejamento e Execução de iterações se sucedem durante o desenvolvimento e vão até a finalização do produto. Ao final de cada ciclo, os clientes realizam a aceitação dos incrementos do produto e podem vir a sugerir mudanças no escopo a serem executadas na iteração seguinte. As iterações são monitoradas continuamente pelos processos da fase Monitoramento e Controle, através de reuniões diárias, gráficos de acompanhamento (*Sprint Burndown Chart*) e reuniões de planejamento. Na fase de Encerramento são finalizados os contratos, é realizada uma reunião de retrospectiva final e obtém-se o produto final.

Além disso, apresenta-se organizado em seis áreas de conhecimento: Gerenciamento Ágil da Integração, Gerenciamento Ágil do Escopo, Gerenciamento Ágil do Tempo, Gerenciamento Ágil das Comunicações, Gerenciamento Ágil dos Riscos e Gerenciamento Ágil dos Custos. Cada uma das áreas de conhecimento teve seus processos criados com base nas abordagens apresentadas no Capítulo 2 desta dissertação.

No Agilius, um processo é definido como um conjunto de atividades, e cada uma das atividades é definida com base no PMBOK[®] *Guide* e nas abordagens ágeis de gerenciamento citadas anteriormente.

Projetos de pequeno e médio porte que precisam obter resultados rápidos são as aplicações sugeridas para a abordagem. O Agilius é indicado para equipes que sejam auto-organizadas, auto-gerenciables, colaborativas, conforme sugerido em [3], [5] e [7]. Já os projetos que possuem características de inovação e que forem propensos a mudanças frequentes são beneficiados pela abordagem.

Para a definição e a apresentação de cada processo foi criada uma tabela. Em cada uma das tabelas está contida a descrição geral dos itens utilizados para resumir os processos de cada uma das áreas de conhecimento do manual Agilius conforme mostrado na Tabela 3.2. Esta estrutura foi elaborada de modo a ser uma forma melhor de visualização dos processos. Ela procura sumarizar os processos descritos para facilitar o seu uso.

O primeiro campo, **Área de Conhecimento**, se refere à área de Gerenciamento Ágil de Projetos a qual pertence o processo. **Grupo de Processos**, assim como no PMBOK[®] *Guide*, diz respeito ao grupo no qual o processo se encaixa. O campo seguinte é preenchido com o nome do Processo, seguido do **Objetivo**, que representa a finalidade do processo em questão. As entradas e

saídas do processo são apresentadas nos campos **Artefatos de Entrada** e **Artefatos e Resultados Esperados**, respectivamente. A seguir, o campo **Atividades** exibe em forma de tópicos as atividades realizadas no decorrer do processo. Vale salientar que essas atividades podem ocorrer em paralelo ou não, e por este motivo não foram numeradas. Finalmente, o campo **Referências** apresenta algumas fontes que são sugeridas e servem de base para instanciar do processo em questão.

Tabela 3.2 Tabela resumo dos processos do Agilius

Área de Conhecimento: <i>(áreas que abordam o gerenciamento de projetos ágil sob um aspecto particular: Integração, Escopo, Tempo, Comunicações, Riscos, Custos)</i>	
Grupo de Processos: <i>(grupo de processos de gerenciamento de projetos necessários para qualquer projeto, segundo PMBOK® Guide: Iniciação, Planejamento, Execução, Monitoramento e Controle, ou Encerramento)</i>	
Processo: <i>(nome do processo)</i>	
Objetivo: <i>(finalidade do processo)</i>	
Papéis: <i>(quem executa o processo)</i>	
Artefatos de Entrada <i>(entradas do processo)</i>	Artefatos e Resultados Esperados <i>(saídas do processo)</i>
Atividades <i>(tarefas realizadas durante a execução do processo, podem acontecer paralelamente ou não)</i>	
Referências: <i>(fontes sugeridas como base para a elaboração de uma instância do processo)</i>	

O Agilius possui duas apresentações, um documento no formato PDF (em <http://www.cin.ufpe.br/~gp2/agilius/agilius.pdf>), e um web site criado a partir do uso do EPF *Composer* disponível em <http://www.cin.ufpe.br/~gp2/agilius/index.htm>, que além de permitir uma rápida navegação ainda possibilitou a modelagem de cada um dos processos, através do uso de SPEM 2.0.

3.4 Gerenciamento Ágil da Integração do Projeto

A integração, no contexto do gerenciamento de um projeto, consiste em fazer escolhas sobre em que pontos concentrar recursos e esforço, antecipando possíveis problemas, tratando-os antes de se tornarem críticos e coordenando o trabalho visando o bem geral do projeto [1].

A entrega chave no gerenciamento da integração é o documento de plano de projeto. Para esta área de conhecimento ágil, fortemente inspirada em [1], [39] e [40], o plano de projeto é desenvolvido de forma iterativa e se traduz em vários exercícios de planejamento e visão do produto. O planejamento detalhado do modelo cascata é convertido em planejamento ágil contínuo, onde funcionalidades são planejadas e estimadas em alto nível no começo de uma iteração.

Conforme proposto por [39] e adotado pelo Agilius, a cada início de iteração é realizado um planejamento detalhado de tarefas e esforço. Sugere-se que cada uma das iterações dure de uma a quatro semanas.

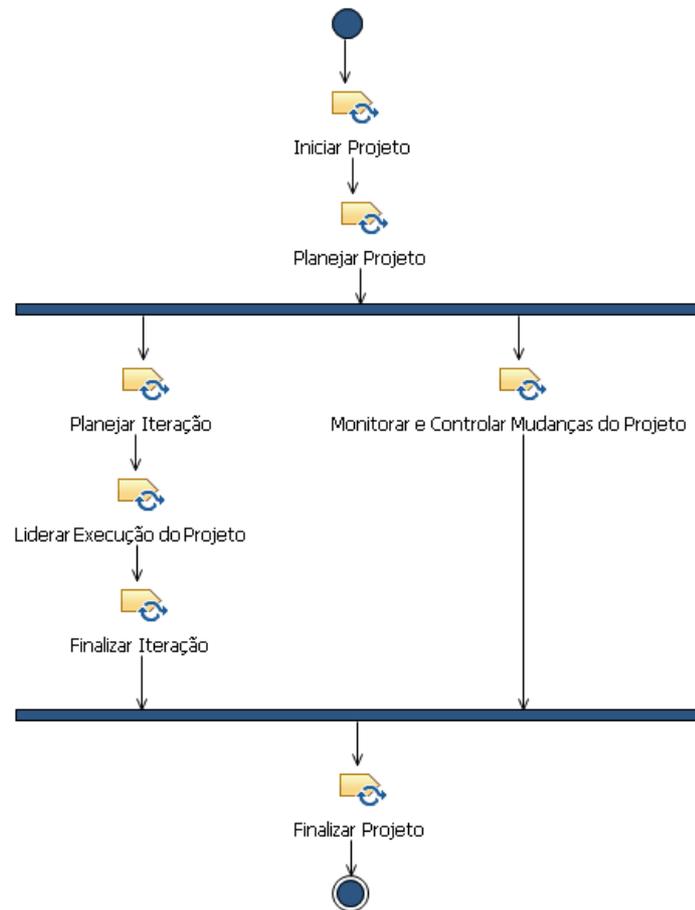


Figura 3.2 Gerenciamento Ágil da Integração

A Figura 3.2 apresenta os processos de Gerenciamento Ágil da Integração, descritos resumidamente abaixo:

- **Iniciar Projeto:** Inicia o projeto, ou as iterações que o compõem.
- **Planejar Projeto:** Realiza o planejamento incremental do projeto e elabora o Plano de Gerenciamento do Projeto.
- **Planejar Iterações:** Realiza o planejamento das iterações e elabora Planos de Iteração.
- **Liderar Execução do Projeto:** Coordena o desenvolvimento e a entrega de funcionalidades do produto e *releases*.

- **Monitorar e Controlar Mudanças do Projeto:** Monitora e adapta mudanças às iterações e ao projeto.
- **Finalizar Projeto:** Encerramento das iterações e do projeto.

3.4.1 Iniciar Projeto

O processo Iniciar Projeto, assim como sugerido no PMBOK® *Guide* e no Documento de Visão do Scrum, determina o início formal de um projeto. Através de informações de projetos passados, cliente e gerente de projeto desenvolvem o documento de Autorização de Início de Projeto, que deve conter os objetivos, uma descrição em alto nível do projeto e a identificação dos envolvidos. O artefato produzido é um documento que formaliza o início do projeto, a Autorização de Início de Projeto. A Tabela 3.3 apresenta uma visão geral deste processo:

Tabela 3.3 Resumo do Processo Iniciar Projeto

Área de Conhecimento: Gerenciamento Ágil da Integração	
Grupo de Processos: Iniciação	
Processo: Iniciar Projeto	
Objetivo: Determinar o início formal de um projeto ou de uma iteração.	
Papéis: Gerente de projetos, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Informações de projetos passados 	<ul style="list-style-type: none"> • Autorização de Início de Projeto
Atividades	
<ul style="list-style-type: none"> • Identificar <i>stakeholders</i> • Descrever objetivos do projeto • Autorizar início de projeto 	
Referências: PMBOK® <i>Guide</i> [1], SLIGER, M. A project manager's survival guide to going agile [39], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/[40] .	

3.4.2 Planejar Projeto

Em Gerenciamento Ágil de Projetos, o esforço para se documentar todo o projeto no começo é drasticamente reduzido. Este processo, que foi concebido de acordo com o utilizado em [3], [7] e [40], encoraja a realização de planejamento contínuo, onde funcionalidades são planejadas e estimadas em alto nível, no começo de um release. O planejamento detalhado de tarefas e esforço é feito a cada início de iteração (*just-in-time*). O Plano de Projeto é desenvolvido através de exercícios iterativos de planejamento, e é composto:

- Do processo de desenvolvimento de software adotado (*tailoring*);

- Do Plano de Iterações;
- Dos marcos do projeto;
- Dos principais requisitos para iniciar o desenvolvimento do produto;
- De um plano de entrega de releases que inclui um cronograma e alocação de recursos;
- De estratégias de mitigação de riscos;
- De estimativas de custos do projeto.

O processo de planejamento, segundo o manifesto ágil e suas abordagens derivadas, envolve toda a equipe de projeto, o cliente e o gerente de projeto. Juntos, eles irão estimar custos, prazos, verificar os riscos do projeto e elaborar cronograma. [7] sugere algumas técnicas para estimar incertezas, riscos de projetos e para gerenciar mudanças. As técnicas utilizadas para definição desta abordagem são: **Matriz de Fator de Exploração** e **Matriz de Custo-Benefício** [3]. As duas técnicas apresentaram-se interessantes por identificar os pontos de mudança em alto nível, permitindo assim uma visão geral do impacto das modificações e em que áreas essas mudanças devem ser realizadas.

Um fator de exploração estima o nível de incerteza e risco de um projeto, conforme ilustra a Tabela 3.4. É uma unidade de medida que varia de 0 a 10, onde um fator de exploração 10 indica um domínio de problema muito orientado a exploração, ou seja, indica nível maior de risco e incerteza. O fator de exploração 1 representa um ambiente de problema muito estável. Este fator serve para ajustar as expectativas de acordo com os riscos do projeto.

Tabela 3.4 Exemplo de Matriz de Fator de Exploração

Dimensão de Requisitos do Produto	Dimensão de Tecnologia do Produto			
	Desconhecido	Pouco Conhecido	Familiar	Bem conhecido
Instável	10	8	7	7
Flutuante	8	7	6	5
Rotineiro	7	6	4	3
Estável	7	5	3	1

Tabela 3.5 Exemplo de Matriz de Custo-Benefício

	Fixo	Flexível	Aceitável
Escopo		x	
Cronograma	x		
Defeitos			x
Recursos			x

Matriz de Custo-Benefício (Tabela 3.5) é um acordo entre o time de projeto e o cliente, utilizado para gerenciar mudanças durante o projeto. Ela informa aos *stakeholders* quais mudanças possuem conseqüências e é usada como base para a tomada de decisão da equipe.

As linhas de uma matriz deste tipo indicam as dimensões chave para criar o valor de produto (escopo, cronograma, defeitos, recursos). Suas colunas apresentam a importância relativa de cada dimensão. Fixa indica que a dimensão, cronograma, por exemplo, está fixa ou restrita e decisões de *tradeoff* não deveriam impactar no desempenho daquela dimensão.

Também indica que a dimensão em questão possui a mais alta prioridade. Flexível indica que a dimensão em questão é importante, mas não importante o suficiente para ser trocada pela dimensão Fixa. Aceitável indica que a dimensão, custo, por exemplo, possui uma larga faixa de tolerância admissível. A Matriz de Custo-Benefício identifica os pontos de mudança em alto nível, permitindo assim uma visão geral do impacto das modificações e em que áreas essas mudanças devem ser realizadas.

A seguir, a Tabela 3.6 apresenta uma visão geral do processo descrito.

Tabela 3.6 Resumo do Processo Planejar Projeto

Área de Conhecimento: Gerenciamento Ágil da Integração	
Grupo de Processos: Planejamento	
Processo: Planejar Projeto	
Objetivo: Elaborar Plano de Projeto.	
Papéis: Gerente de projetos, desenvolvedor, analista de sistemas, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Informações de projetos passados • <i>Backlog</i> do Produto 	<ul style="list-style-type: none"> • Autorização da iteração • Plano de Projeto
Atividades	
<ul style="list-style-type: none"> • Elaborar Matriz de Fator de Exploração • Elaborar Matriz de Custo-Benefício • Elaborar Plano de Projeto • Estimar um cronograma inicial para o projeto • Estabelecer marcos do projeto 	
Referências: AUGUSTINE, S. Managing agile projects [7] , HIGHSMITH, J. Agile project management: creating innovative products [3], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/[40] .	

3.4.3 Planejar Iterações

Conforme sugerido pelo Manifesto Ágil, em um dos seus princípios de agilidade, software deve ser liberado frequentemente, a fim de que sua utilização aprimore os seus recursos. Com base nesta afirmação, o processo Planejar Iteração foi concebido. O Plano de Iteração é definido em reuniões de planejamento envolvendo todo o time, de acordo com o apresentado em [40], na técnica *Sprint Planning Meeting*. O time deve criar, se comprometer e ser responsável pelo plano. Planos de Release, Planos de Iteração e qualquer outro planejamento devem ser realizados com a participação de todos os envolvidos.

Na Tabela 3.7 o processo Planejar Iterações é apresentado resumidamente.

Tabela 3.7 Resumo do Processo Planejar Iterações

Área de Conhecimento: Gerenciamento Ágil da Integração	
Grupo de Processos: Planejamento	
Processo: Planejar Iterações	
Objetivo: Elaborar Plano de Iteração.	
Papéis: Gerente de projetos, desenvolvedor, analista de sistemas, cliente.	
Artefatos de Entrada <ul style="list-style-type: none">• Informações de projetos passados• <i>Backlog</i> de Produto	Artefatos e Resultados Esperados <ul style="list-style-type: none">• Plano de Iteração
Atividades <ul style="list-style-type: none">• Autorizar início de iteração• Realizar reunião de planejamento da iteração• Estimar um cronograma inicial para a iteração• Priorizar <i>Backlog</i> de Produto	
Referências: Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx [40].	

Ainda tomando [40] como base, o Plano de Iteração deve ser definido para que se associem funcionalidades a iterações e deve conter um cronograma para realização das atividades de projeto e um *Backlog* de Produto priorizado. O objetivo é planejar e dividir o projeto em iterações para que o time de desenvolvimento se concentre em pequenos incrementos de trabalho.

A Autorização de Início de Iteração é um documento que contém o *Backlog* do Produto para a iteração que se iniciará. De acordo com Scrum, para a primeira iteração o cliente deve indicar neste documento quais itens do *Backlog* de Produto devem ser desenvolvidos primeiro. A partir da

segunda iteração o cliente deve verificar a funcionalidade desenvolvida na iteração anterior, priorizar itens do *Backlog* do Produto e então autorizar a iteração.

3.4.4 Liderar Execução do Projeto

O processo Liderar Execução do Projeto é responsável pela entrega de funcionalidades de produto e *releases*. Com sua fundamentação em [7], possui o objetivo de criar uma comunidade de projeto auto-organizada e colaborativa, onde o gerente de projetos exerce o papel de facilitador. Ao exercer este papel, o gerente faz com que cliente e equipe participem do processo de tomada de decisão. Durante a execução do projeto, o gerente e o cliente participam de reuniões freqüentes para avaliar o seu progresso. A Tabela 3.8 apresenta uma visão geral deste processo.

Tabela 3.8 Resumo do Processo Liderar Execução do Projeto

Área de Conhecimento: Gerenciamento Ágil da Integração	
Grupo de Processos: Execução	
Processo: Liderar Execução do Projeto	
Objetivo: Coordenar as atividades das iterações de projeto.	
Papéis: Gerente de projetos, desenvolvedor, analista de sistemas, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • <i>Backlog</i> de Impedimentos • <i>Backlog</i> de Produto 	<ul style="list-style-type: none"> • <i>Releases</i> • Mudanças Implementadas • <i>Backlog</i> de Impedimentos atualizado
Atividades	
<ul style="list-style-type: none"> • Realizar reuniões diárias de acompanhamento • Realizar reuniões freqüentes com o cliente • Verificar mudanças necessárias ao escopo através do <i>Backlog</i> de Impedimentos • Solicitar reparos de defeito • Realimentar <i>Backlog</i> de Impedimentos 	
Referências: AUGUSTINE, S. Managing agile projects [7], BECK, K. Extreme Programming Explained: Embrace Change [6], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/[40] .	

Durante a iteração, reuniões curtas devem ser realizadas diariamente, conforme sugerem [9] e [40]. Estas reuniões são o elemento chave na comunicação que mantém o time em sincronia e mais apto a resolver problemas. Desta forma, o gerente consegue exercer sua função de facilitador ao monitorar o projeto e as dificuldades enfrentadas pela equipe. Os impedimentos do projeto, elicitados nestas reuniões diárias, são listados num *Backlog* de Impedimentos, artefato que auxilia no Gerenciamento de Riscos do projeto. Durante a execução, as mudanças solicitadas no decorrer do projeto que forem aceitas serão implementadas.

3.4.5 Monitorar e Controlar Mudanças do Projeto

O controle de mudanças está integrado à rotina diária do time ágil. Conforme encoraja [40], a modificação no produto é gerenciada via *Backlog* de Produto priorizado pelo cliente, que é responsável por manter a lista de itens a ser trabalhada contendo as funcionalidades que provêm maior valor ao negócio. A seguir, Tabela 3.9 apresenta uma visão geral deste processo.

Tabela 3.9 Resumo do Processo Monitorar e Controlar Mudanças do Projeto

Área de Conhecimento: Gerenciamento Ágil da Integração	
Grupo de Processos: Monitoramento e Controle	
Processo: Monitorar e Controlar Mudanças do Projeto	
Objetivo: Adaptar mudanças às iterações de projeto.	
Papéis: Gerente de projetos, desenvolvedor, analista de sistemas, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Matriz de Custo-Benefício • <i>Backlog</i> de Produto 	<ul style="list-style-type: none"> • <i>Backlog</i> de Produto re-priorizado • <i>Sprint Burndown Chart</i> • Relatórios de Desempenho • Mudanças Solicitadas
Atividades	
<ul style="list-style-type: none"> • Gerenciar <i>Backlog</i> de Produto • Realizar reuniões diárias • Realizar reuniões com o cliente • Elaborar relatórios de desempenho • Elaborar <i>Sprint Burndown Chart</i> • Analisar Matriz de Custo-Benefício 	
Referências: AUGUSTINE, S. Managing agile projects [7], BECK, K. Extreme Programming Explained: Embrace Change [6], Scrum for Team System, disponível em: http://www.scrumforteamssystem.com/en/default.aspx/ [40].	

De acordo com [6], [7] e [40], mudanças no processo são realizadas ao fim de cada iteração, permitindo ao time realizar correções no curso do produto e na forma que ele trabalha. Ao final de cada iteração, a equipe apresenta a versão desenvolvida ao cliente, o qual realiza um *feedback* que pode afetar futuras decisões sobre os itens no *Backlog* do Produto. Neste *feedback*, o cliente realiza a priorização dos itens para a próxima iteração. Vale salientar que modificações preferivelmente serão aceitas no início de cada iteração. As mudanças que surgem no decorrer da iteração são reportadas para a iteração seguinte, a não ser em casos particulares onde seja necessário excluir um requisito ou que se perceba uma necessidade de alteração de um requisito sendo implementado.

A fim de proporcionar uma melhor visão do progresso das atividades *Sprint Burndown Charts* e Relatórios de Desempenho devem ser utilizados pela equipe. As mudanças que forem necessárias e detectadas neste processo são enumeradas numa lista de Mudanças Solicitadas, para serem avaliadas e se aprovadas, implementadas.

3.4.6 Finalizar Iteração

O processo Finalizar Iteração indica o término de uma etapa do projeto, de acordo com a Tabela 3.10.

Tabela 3.10 Resumo do Processo Finalizar Iteração

Área de Conhecimento: Gerenciamento Ágil da Integração	
Grupo de Processos: Finalizar Iteração	
Processo: Encerramento	
Objetivo: Encerrar uma iteração.	
Papéis: Gerente de projetos, desenvolvedor, analista de sistemas, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Plano de Projeto • Plano de Iterações • Relatórios de Desempenho • Entregas 	<ul style="list-style-type: none"> • Entrega de funcionalidades • Encerramento da iteração
Atividades	
<ul style="list-style-type: none"> • Concluir iteração • Realizar retrospectiva da iteração 	
Referências: PMBOK® <i>Guide</i> [1], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/[40] .	

O término da iteração é um mini-encerramento de cada uma das etapas do projeto. Este trabalho encoraja a prática de Reunião de Retrospectiva da Iteração, apresentada em [40], por acreditar que a cada final de iteração devem ser levantados pontos positivos e negativos no processo de gerência e execução, a fim de que nas próximas iterações os mesmos erros não se repitam. Nas Reuniões de Retrospectiva, a equipe pode sugerir mudanças no processo, permitindo ao time realizar correções no curso do produto e na forma de trabalho da equipe.

3.4.7 Finalizar Projeto

O processo Finalizar Projeto diz respeito ao término de todas as atividades de todos os processos de gerenciamento de projetos, conforme resumido na Tabela 3.11.

Tabela 3.11 Resumo do Processo Finalizar Projeto

Área de Conhecimento: Gerenciamento Ágil da Integração	
Grupo de Processos: Finalizar Projeto	
Processo: Encerramento	
Objetivo: Encerrar projeto e contratos.	
Papéis: Gerente de projetos, desenvolvedor, analista de sistemas, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Plano de Projeto • Relatórios de Desempenho • Entregas 	<ul style="list-style-type: none"> • Entrega de funcionalidades • Entrega do produto • Conclusão de contratos
Atividades	
<ul style="list-style-type: none"> • Concluir projeto • Realizar retrospectiva do projeto 	
Referências: PMBOK® <i>Guide</i> [1], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/[40] .	

Este processo encerra formalmente o projeto e os contratos relacionados a ele. A conclusão do projeto é a entrega do produto final e toda documentação associada. Também inclui a aprendizagem e a incorporação da aprendizagem ao trabalho do próximo projeto, análise de sucesso ou fracasso do projeto e lições aprendidas. Este aprendizado é discutido pela equipe na Reunião de Retrospectiva de Projeto.

O encerramento dos contratos diz respeito ao conjunto de atividades necessárias para terminar qualquer contrato estabelecido para o projeto. Inclui atividades de rescisão de contrato, que ocorrem quando existe a incapacidade de dar continuidade ao projeto.

3.5 Gerenciamento Ágil do Escopo do Projeto

Segundo [39], os métodos ágeis da mesma forma que o PMBOK® *Guide*, acreditam que a área de conhecimento Gerenciamento do Escopo do Projeto merece grande destaque. Abordagens voltadas ao planejamento trabalham no sentido de prever mudanças no escopo, enquanto as abordagens ágeis esperam que as mudanças ocorram e se adaptam a essas mudanças.

A chave para contornar o escopo no desenvolvimento iterativo é o comprometimento com as expectativas de resultado de negócio como limite externo do escopo [41]. Essas expectativas são o resultado que os *stakeholders* esperam que projeto de desenvolvimento de software tenha em termos de operações internas, mercado ou outros ambientes externos. Assim, entregar as

funcionalidades de software corretas depende se estas funcionalidades suportam as expectativas de negócio dos clientes.

Com o objetivo de atender as expectativas de negócio do cliente, o Gerenciamento Ágil do Escopo se concentra na definição do escopo em alto nível, na elicitación e priorização dos requisitos. Estas atividades contam com a participação do cliente e de toda a equipe do projeto para definir as funcionalidades para cada uma das iterações de projeto. A Figura 3.3 apresenta a área de conhecimento Gerenciamento Ágil do Escopo.

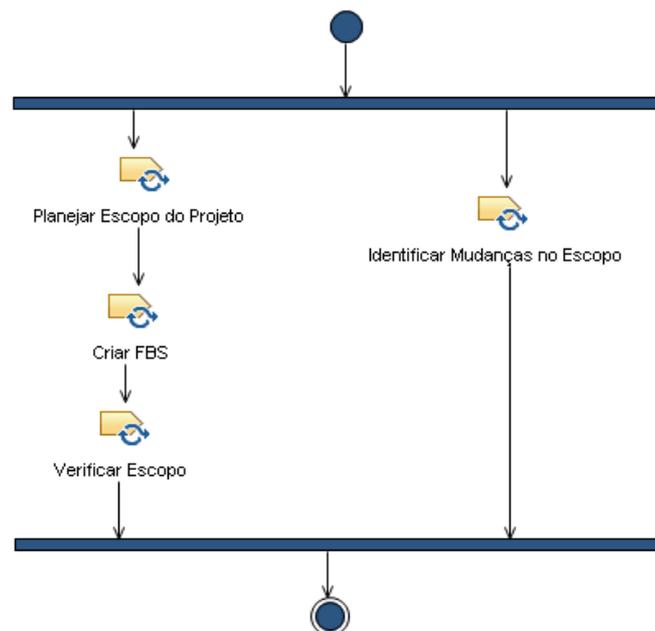


Figura 3.3 Gerenciamento Ágil do Escopo

Tendo como base as abordagens Scrum, *Agile Project Management Framework*, e *Agile Project Management*, seus processos foram definidos conforme apresentado a seguir:

- **Planejar Escopo do Projeto:** Identifica o escopo do projeto, do produto e quais as restrições para a sua implementação.
- **Criar FBS:** Desenvolve uma estrutura analítica de funcionalidades, ou *Feature Breakdown Structure*.
- **Verificar Escopo:** Verifica a cada iteração se o produto está em conformidade com os requisitos e critérios de aceitação estabelecidos.
- **Identificar Mudanças no Escopo:** Acomoda as mudanças em iterações posteriores.

As próximas seções apresentam os quatro processos definidos em maiores detalhes.

3.5.1 Planejar Escopo do Projeto

Este processo tem o objetivo de identificar qual é o escopo do projeto e quais as restrições para a sua implementação. Consiste na criação de um Plano de Gerenciamento do Escopo do Projeto, que poderá ser modificado a cada iteração. Este plano é composto por:

- Um documento de visão em alto nível do produto, que evoluirá durante o projeto, com base em novas informações;
- Uma lista de requisitos funcionais e não funcionais do sistema, realimentada a cada iteração;
- Os limites do projeto, ou seja, tudo aquilo que está incluído no projeto;
- As entregas do projeto;
- Critérios de aceitação de produtos;
- Objetivos, restrições e premissas do projeto.
- Uma lista com os riscos conhecidos.

O planejamento do escopo do projeto inicia-se com a análise das necessidades do projeto, das informações históricas de outros projetos e do Plano de Gerenciamento de Projeto. A partir desta análise, a equipe constrói o documento de Visão do Produto, fundamentado em [40], que faz uma descrição em alto nível do produto, seus objetivos e público alvo a fim de identificar os limites do projeto, o que faz parte do escopo e o que não faz.

Depois disto, a equipe lista os requisitos para o sistema ou produtos que serão desenvolvidos pelo projeto. A lista, conhecida como *Backlog* de Produto, é utilizada no plano de projeto como uma estimativa inicial dos requisitos. Segundo o Scrum, o *Backlog* de Produto é um documento que evolui de acordo com o desenvolvimento do produto e do ambiente no qual ele será usado. Juntos, o documento de Visão do Produto e o *Backlog* de Produto formam a base para a elaboração de um plano de projeto em alto nível compatível com a volatilidade dos projetos ágeis.

Durante o planejamento do escopo, os envolvidos no processo podem encontrar alguns problemas que resultarão em atrasos no desenvolvimento do projeto. Este trabalho sugere que estes problemas sejam listados no *Backlog* de Impedimentos como itens que impedem o progresso do

projeto. O uso do *Backlog* de Impedimentos se justifica devido a sua grande popularidade entre os adeptos do Scrum. A equipe identifica os impedimentos que precisam ser resolvidos e que geralmente estão associados a riscos. Os itens que compõem o *Backlog* de Impedimentos não possuem uma priorização, mas geralmente estão associados a algum item do *Backlog* de Produto ou às tarefas para executar cada um dos itens. Para gerenciar o *Backlog* de Impedimentos, os itens nele contidos devem ser identificados e estimados, em tempo ou tamanho, e a sua priorização deve ser estabelecida pelo cliente.

Tabela 3.12 Resumo do Processo Planejar Escopo do Projeto

Área de Conhecimento: Gerenciamento Ágil do Escopo	
Grupo de Processos: Planejamento	
Processo: Planejar Escopo do Projeto	
Objetivo: Identificar qual é o escopo do projeto e quais as restrições para a sua implementação.	
Papéis: Gerente de projetos, desenvolvedor, analista de sistemas, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Plano de Gerenciamento do Projeto • Informações de projetos passados • <i>Backlog</i> do Produto • <i>Backlog</i> de Impedimentos 	<ul style="list-style-type: none"> • Visão do Produto • <i>Backlog</i> do Produto atualizado • <i>Backlog</i> de Impedimentos atualizado • Arquitetura do Produto • Casos de Teste • Plano de Gerenciamento do Escopo
Atividades	
<ul style="list-style-type: none"> • Analisar necessidades do projeto • Analisar informações históricas de outros projetos • Elaborar a Visão do Produto • Atualizar <i>Backlog</i> do Produto • Atualizar <i>Backlog</i> de Impedimentos • Elaborar Plano de Gerenciamento do Escopo 	
Referências: BECK, K. Extreme Programming Explained: Embrace Change [6], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/[40] .	

Algumas práticas utilizadas na concepção do escopo do produto são o desenvolvimento da Arquitetura do Produto e a Elaboração de Casos de Teste. A Arquitetura do Produto apresenta os componentes de software, suas propriedades externas, e seus relacionamentos com outros softwares. A documentação da arquitetura do software facilita a comunicação entre os *stakeholders*, registra as decisões iniciais acerca do projeto de alto nível, e permite o reuso do projeto dos componentes e padrões entre projetos. A Elaboração de Casos de Teste a partir dos requisitos é realizada segundo a metodologia TDD, *Test Driven Development* ou Desenvolvimento Dirigido a Testes, que faz com que

primeiro se pense no objetivo que se deseja alcançar para, em seguida, iniciar a escrita do código que alcance esses objetivos. A Tabela 3.12 apresenta uma visão geral deste processo.

3.5.2 Criar FBS

Uma *Work Breakdown Structure*, WBS ou Estrutura Analítica de Projeto é uma decomposição hierárquica orientada à entrega do trabalho a ser executado pela equipe do projeto, para atingir os objetivos do projeto e criar as entregas necessárias [1]. De maneira análoga à WBS, uma *Feature Breakdown Structure* (FBS) [3], [7] é uma estrutura analítica onde as entregas são representadas por funcionalidades de software também chamadas de *features*.

Este tipo de estrutura pode aparentar ser de difícil administração, mas oferece a vantagem significativa de aumentar a interação entre o time de projeto e os clientes [3], [7]. Utilizando *Work Breakdown Structures* as atividades de um projeto são relativamente permanentes. Com uma *Feature Breakdown Structure*, funcionalidades podem ser adicionadas ou excluídas durante cada iteração, isto pode ser controlado agrupando as features, o que traz mais estabilidade a FBS [3]. Esta técnica encontra-se presente no Agilius por ser indicada por [3] e [7]. Abaixo, a Tabela 3.13 apresenta uma visão geral deste processo.

Tabela 3.13 Resumo do Processo Criar FBS

Área de Conhecimento: Gerenciamento Ágil do Escopo	
Grupo de Processos: Planejamento	
Processo: Criar FBS	
Objetivo: Elaborar a partir de Cartões de Funcionalidade uma <i>Feature Breakdown Structure</i> para o projeto.	
Papéis: Gerente de projetos, desenvolvedor, analista de sistemas, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Visão do Produto • <i>Backlog</i> de Produto • Plano de Gerenciamento do Escopo 	<ul style="list-style-type: none"> • FBS • Cartões de Funcionalidade • Plano de Gerenciamento do Escopo atualizado
Atividades	
<ul style="list-style-type: none"> • Elaborar Cartões de Funcionalidade • Elaborar <i>Feature Breakdown Structure</i> 	
Referências: AUGUSTINE, S. Managing agile projects [7], HIGHSMITH, J. Agile project management: creating innovative products [3].	

Para desenvolver uma FBS primeiramente a equipe realiza uma análise da Visão do Produto e do *Backlog* de Produto, e elabora Cartões de Funcionalidade ou *Feature Cards* para o projeto. De acordo com [3], Cartões de Funcionalidade são cartões nos quais os membros do time de projeto e o

cliente registram informações obtidas nas discussões sobre quais são os requisitos que compõem o produto. São utilizados para identificar as funcionalidades que o cliente quer no produto. Eles funcionam como um acordo entre clientes e membros do time de projeto para discutir e documentar detalhes dos requisitos durante uma iteração. O gerente de projetos utiliza estes cartões para elaborar a *Feature Breakdown Structure*, que será atualizada durante o projeto. Além dos Cartões de Funcionalidade, o gerente pode fazer uso de técnicas como decomposição e consultar FBSs de projetos passados na elaboração da FBS.

3.5.3 Verificar Escopo

Verificar Escopo, conforme resumido na Tabela 3.14, é o processo onde o cliente realiza a aceitação das entregas. A verificação é realizada a cada iteração, com os clientes fazendo revisão, testando, e aceitando as funcionalidades implementadas. Isto implica em constatar que o produto está sendo desenvolvido de forma satisfatória.

Tabela 3.14 Resumo do Processo Verificar Escopo

Área de Conhecimento: Gerenciamento Ágil do Escopo	
Grupo de Processos: Monitoramento e Controle	
Processo: Verificação do Escopo	
Objetivo: Constatar se, de acordo com o cliente, o produto está sendo desenvolvido de forma satisfatória.	
Papéis: Cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • <i>Release</i> • Plano de Gerenciamento do Escopo 	<ul style="list-style-type: none"> • Planilha de resultados dos testes de aceitação • Entregas aceitas
Atividades	
<ul style="list-style-type: none"> • Realizar testes de aceitação • Realizar inspeções 	
Referências: PMBOK® Guide [1].	

Idealmente, a verificação ocorre durante a iteração, mas pode acontecer ao final de cada iteração, no decorrer da demonstração do sistema. A verificação é realizada por meio de Testes de Aceitação e inspeções. Testes de Aceitação são testes caixa-preta realizados pelo usuário, a fim de demonstrar conformidade com os requisitos do software. Os resultados desses testes são registrados em uma Planilha de Resultados de Testes de Aceitação. A inspeção inclui atividades como medição, exame e verificação para determinar se o trabalho e as entregas atendem aos requisitos e aos critérios de aceitação do produto [1].

3.5.4 Identificar Mudanças no Escopo

De acordo com [40], este processo foi criado para realizar o controle do escopo do produto através do gerenciamento do *Backlog* do Produto. Este processo está relacionado com a re-priorização ou modificação de requisitos e os impedimentos detectados nas reuniões diárias de acompanhamento. A Tabela 3.15 apresenta o processo Identificar Mudanças no Escopo resumidamente.

Tabela 3.15 Resumo do Processo Identificar Mudanças no Escopo

Área de Conhecimento: Gerenciamento Ágil do Escopo	
Grupo de Processos: Monitoramento e Controle	
Processo: Identificar Mudanças no Escopo	
Objetivo: Re-priorização ou modificação de requisitos e os impedimentos detectados nas reuniões diárias de acompanhamento.	
Papéis: Gerente de projetos, desenvolvedor, analista de sistemas, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> Plano de Gerenciamento do Escopo 	<ul style="list-style-type: none"> Solicitação de mudanças Plano de Gerenciamento do Escopo atualizado
Atividades	
<ul style="list-style-type: none"> Reconhecer as principais entregas do projeto Realizar reuniões diárias Realizar reuniões de planejamento da iteração Acompanhar o progresso do projeto Elaborar <i>Backlog</i> de Impedimentos Realizar reuniões freqüentes com o cliente 	
Referências: BECK, K. Extreme Programming Explained: Embrace Change [6], Scrum for Team System, disponível em: http://www.scrumforteamssystem.com/en/default.aspx/[40] .	

Quando se detecta algum impedimento na execução das atividades de desenvolvimento de uma funcionalidade, este item vai para o *Backlog* de Impedimentos. Isto pode levar a uma lista de requisitos reordenada, de modo que as funcionalidades que precisem ser modificadas recebam atenção especial da equipe de desenvolvimento. A detecção de impedimentos e as modificações nas funcionalidades devem ocorrer durante as iterações.

Modificações no escopo do projeto que são identificadas no decorrer das iterações, através das Reuniões de Planejamento da Iteração e de reuniões com o cliente, são tratadas pelo processo Monitorar e Controlar Mudanças do Projeto. Se as mudanças afetarem o escopo do projeto, modificações no Plano de Gerenciamento do Escopo e estruturas analíticas serão necessárias. Essas

modificações, inclusive, poderão afetar o Plano de Gerenciamento do Projeto, que por este motivo será alterado.

3.6 Gerenciamento Ágil do Tempo do Projeto

O gerenciamento do tempo do projeto inclui os processos necessários para realizar o término do projeto no prazo [1]. O cronograma definido nas abordagens ágeis é orientado aos *releases* produzidos em cada iteração, que são planejados conforme a prioridade funcional definida pelo cliente no *Backlog* do Produto. Esta área de conhecimento foi definida de acordo com o que sugere o manifesto ágil, no sentido de que as iterações de um projeto devem ter duração de duas e quatro semanas para atender rapidamente às necessidades do cliente, e o prazo final pode não estar claramente definido. Contudo, como o cliente avalia o produto ao final de cada iteração, há uma redução dos conflitos pela cumplicidade no processo.

Para construir o cronograma de projeto, objetivo principal desta área de conhecimento, gerente e equipe calculam a duração do projeto. O processo de estimativa de duração de um projeto começa com a estimativa do seu tamanho [42]. Após se ter uma idéia da dimensão do projeto, e de conhecer as funcionalidades desejadas é possível estimar o tamanho de um projeto. A partir da estimativa do tamanho, temos a estimativa de duração e através dela podemos construir o cronograma do projeto.

Segundo [1], nos projetos de menor escopo o seqüenciamento das atividades, a estimativa de recursos da atividade, a estimativa de duração da atividade e o desenvolvimento do cronograma estão tão estreitamente ligados que são considerados um único processo, que pode ser realizado por uma pessoa durante um período de tempo relativamente curto.

A Figura 3.4 apresenta a áreas de conhecimento Gerenciamento Ágil do Tempo. Tendo como base as abordagens Scrum, XP, *Agile Project Management framework* e as técnicas apresentadas em [42], os processos apresentados a seguir foram definidos:

- **Definir e Priorizar Atividades de Projeto:** Identifica as atividades do projeto e as coloca em ordem de prioridade em relação a sua complexidade de execução e ao desejado pelo cliente.
- **Elaborar Cronograma:** Desenvolve o cronograma do projeto com base nos recursos necessários, duração e seqüência de atividades.

- **Monitorar e Controlar Cronograma:** Verifica o progresso das atividades durante as iterações e ajusta o cronograma na tentativa de não comprometer os prazos do projeto.

As próximas seções apresentam os quatro processos definidos em maiores detalhes.

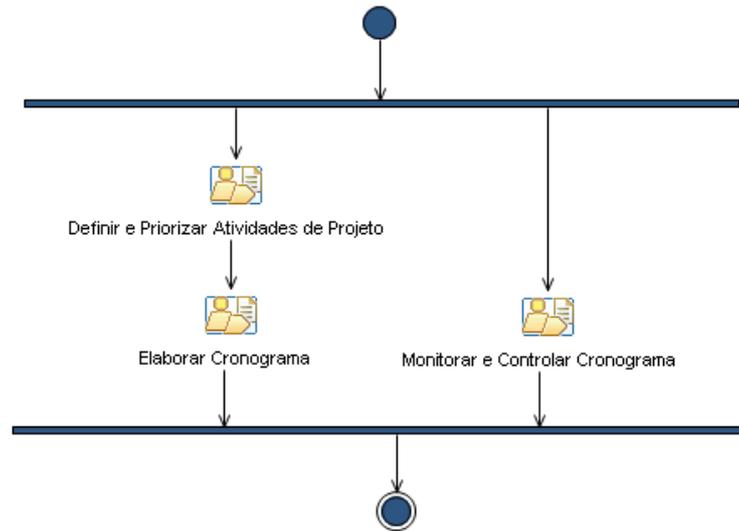


Figura 3.4 Gerenciamento Ágil do Tempo

3.6.1 Definir e Priorizar Atividades do Projeto

A definição das atividades do cronograma envolve identificar e documentar o trabalho planejado para ser realizado [1]. Este processo procura identificar quais as atividades a serem realizadas durante o projeto, durante cada iteração e quais são as prioridades para organizar a ordem de realização dessas tarefas.

APM framework contribui na concepção deste processo, pois utiliza a FBS para realizar estimativas de tempo. Através da FBS, da descrição de cada funcionalidade a ser desenvolvida, que é encontrada nos Cartões de Funcionalidade, e de modelos de projetos anteriores, é possível determinar quais tarefas são necessárias para a iteração e os marcos de todo o projeto.

Em seguida, a equipe determina a dependência e precedência entre as tarefas. Esta atividade envolve a identificação e documentação dos relacionamentos lógicos entre as tarefas que compõem o cronograma. Uma abordagem sugerida para este fim é o Método do Diagrama de Precedência, que constrói um diagrama de rede do cronograma do projeto usando caixas que representam tarefas do cronograma as quais são conectadas por setas, que ilustram as dependências entre as tarefas.

No processo de definição das atividades, podem ocorrer algumas mudanças que refletem no documento de Visão do Projeto e na FBS. Essas mudanças são tratadas pela atividade Controlar Mudanças do Projeto da área de conhecimento Gerenciamento Ágil da Integração do Projeto. A Tabela 3.16 apresenta um resumo das atividades deste processo:

Tabela 3.16 Resumo do Processo Definir e Priorizar Atividades de Projeto

Área de Conhecimento: Gerenciamento Ágil do Tempo	
Grupo de Processos: Planejamento	
Processo: Definir e Priorizar Atividades de Projeto	
Objetivo Identificar as atividades do cronograma, determinar sua precedência e dependência e priorizá-las.	
Papéis: Gerente de projetos, desenvolvedor, analista de sistemas.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Plano de Gerenciamento do Projeto • Informações de projetos passados • FBS 	<ul style="list-style-type: none"> • Lista de tarefas priorizada • Mudanças solicitadas
Atividades	
<ul style="list-style-type: none"> • Determinar as atividades de projeto • Determinar dependência e precedência entre atividades de projeto 	
Referências: COHN, M. Agile Estimating and Planning [42], PMBOK® Guide [1].	

3.6.2 Elaborar cronograma

O desenvolvimento do cronograma é um processo iterativo que determina as datas de início e término planejadas para as atividades do projeto. A cada iteração, o cronograma é incrementado conforme o trabalho se desenvolve, o plano de gerenciamento do projeto se modifica e os riscos previstos ocorrem ou desaparecem e novos riscos são identificados.

Para realizar as estimativas de tamanho e tempo do projeto este trabalho utilizou as técnicas apresentadas em [6] e [42]. O gerente de projetos e a equipe realizam uma estimativa de qual será o tamanho do projeto e de cada uma das iterações, através de Pontos por Estória. A equipe estima a duração das atividades através das técnicas: Dia Ideal, Semana Ideal e Jogo do Planejamento.

A estimativa de duração é progressivamente elaborada, e exige que a quantidade de esforço de trabalho necessária seja estimada, que a quantidade prevista de recursos a ser aplicada para terminar a atividade do cronograma seja estimada e que o número de períodos de trabalho necessário para terminar a atividade do cronograma seja determinado [1]. Estimar recursos de uma atividade envolve determinar os equipamentos e o material necessários, as quantidades de cada

recurso que serão utilizados e quando os recursos estarão disponíveis. Esta atividade pode ser realizada através de consulta à estimativa de projetos passados e da própria experiência do gerente de projeto e da sua equipe.

Após a realização das estimativas de tamanho, duração e recursos, a equipe elabora um cronograma de projeto que serve como uma linha de base para acompanhar o progresso do projeto. As técnicas de análise de rede de cronograma Caminho Crítico e Corrente Crítica são sugeridas para a execução desta atividade, pois elas levam em conta os recursos limitados. Além disso, a técnica do Caminho Crítico proporciona a redução do estoque de materiais, equipamentos e mão-de-obra do projeto, com o intuito de gerar valor. O uso de modelos de cronograma de projetos semelhantes também é sugerido. Abaixo, a Tabela 3.17 apresenta uma visão geral das atividades deste processo.

Tabela 3.17 Elaborar Cronograma

Área de Conhecimento: Gerenciamento Ágil do Tempo	
Grupo de Processos: Planejamento	
Processo: Elaborar Cronograma	
Objetivo: Desenvolver o cronograma das iterações e do projeto.	
Papéis: Gerente de projetos, desenvolvedor, analista de sistemas.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Plano de Gerenciamento do Projeto • Informações de projetos passados • FBS 	<ul style="list-style-type: none"> • Estimativa do tamanho do projeto e das iterações • Lista de recursos • Calendário dos recursos • Estimativas de duração de tarefas • Cronograma de Projeto
Atividades	
<ul style="list-style-type: none"> • Estimar o tamanho do projeto e o tamanho das iterações • Estimar a duração das tarefas • Estimar recursos necessários • Desenvolver Cronograma 	
Referências: BECK, K. Extreme Programming Explained: Embrace Change [6], COHN, M. Agile Estimating and Planning [42], HIGHSMITH, J. Agile project management: creating innovative products [3].	

3.6.3 Monitorar e Controlar Cronograma

Com o decorrer do projeto é possível que ocorram atrasos na realização das tarefas. Para perceber atrasos antes que aconteçam, o progresso das tarefas deve ser constantemente monitorado e o cronograma ajustado para contornar essas situações, na tentativa de não comprometer os prazos do projeto.

O controle do cronograma está relacionado à determinação do andamento atual do cronograma do projeto, ao controle dos fatores que criam mudanças no cronograma, à determinação de que o cronograma do projeto mudou e adaptação do projeto as mudanças, conforme elas efetivamente ocorrerem.

O andamento das atividades deste processo tem como base [6] e [40]: é monitorado através de relatórios de progresso que incluem datas de início e término reais das tarefas, reuniões diárias de acompanhamento e gráficos de monitoramento de progresso, como *Sprint Chart Burndown* e gráficos de barras de comparação de cronogramas.

Com o intuito de identificar os fatores que ocasionam mudanças no cronograma e de controlar esses fatores, a atividade Gerenciar Mudanças no Cronograma pode levar a modificações nas estimativas já realizadas e no próprio cronograma de projeto. Os gráficos utilizados para acompanhar o progresso das tarefas em cada iteração são *Sprint Chart Burndown* e gráficos de barras de comparação de cronogramas. Para facilitar a análise do progresso do cronograma, é conveniente usar um gráfico de barras de comparação de cronogramas, que exibe duas barras para cada atividade do cronograma. Uma barra mostra o andamento atual real e a outra mostra o andamento da linha de base do cronograma do projeto aprovado. Isso exibe visualmente onde o cronograma progrediu conforme planejado ou onde ocorreram defasagens.

As mudanças no cronograma do projeto podem exigir ou não ajustes nos outros componentes do plano de gerenciamento do projeto. As mudanças solicitadas são processadas para revisão e tratadas pelo processo: Monitorar e Controlar Mudanças do Projeto. Após as estimativas serem realizadas novamente, o cronograma é atualizado para a próxima ou as próximas iterações de projeto.

Uma ação corretiva é tudo que é feito para que o desempenho futuro esperado de prazos do projeto fique de acordo com o cronograma aprovado para o projeto. As ações corretivas incluem as ações especiais tomadas para garantir o término de uma atividade do cronograma no prazo ou com o menor atraso possível. Quando uma mudança é necessária, ações corretivas são recomendadas para realizar essa mudança.

A Tabela 3.18 apresenta uma visão geral deste processo.

Tabela 3.18 Resumo do Processo Monitorar e Controlar Cronograma

Área de Conhecimento: Gerenciamento Ágil do Tempo	
Grupo de Processos: Monitoramento e Controle	
Processo: Monitorar e Controlar Cronograma	
Objetivo: Determinar o andamento atual do cronograma do projeto, o controle dos fatores que criam mudanças no cronograma, mudanças no cronograma do projeto e adaptar o projeto às mudanças.	
Papéis: Gerente de projetos, desenvolvedor.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Cronograma de Projeto 	<ul style="list-style-type: none"> • Gráficos de monitoramento de progresso • Solicitação de mudanças • Ações corretivas recomendadas • Cronograma atualizado
Atividades	
<ul style="list-style-type: none"> • Monitorar o progresso das tarefas • Gerenciar mudanças no cronograma • Atualizar Cronograma 	
Referências: BECK, K. Extreme Programming Explained: Embrace Change [6], PMBOK® Guide [1], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/[40] .	

3.7 Gerenciamento Ágil das Comunicações do Projeto

Gerenciar as comunicações do projeto significa gerar, coletar, armazenar e distribuir informações durante todo o projeto. Em projetos ágeis, a informação é o catalisador da mudança e adaptação. Interações entre pessoas envolvem contínua troca de informações e a riqueza dessas interações depende, em grande parte, do nível de abertura da informação. Para um time ágil se adaptar, a informação deve ser aberta e os obstáculos para a troca de informação devem ser identificados e removidos [7].

A comunicação deve ser clara, rápida, aberta e deve ocorrer no momento oportuno. É importante que todos os envolvidos no projeto entendam como as comunicações afetam o projeto como um todo e que o gerente de projeto se certifique de que todos compreenderão a mensagem a ser entregue. O objetivo é melhorar a comunicação entre os envolvidos promovendo feedback constante e transparência durante o desenvolvimento. A comunicação deve ser colaborativa e direta de acordo com a proximidade entre todos os envolvidos do projeto.

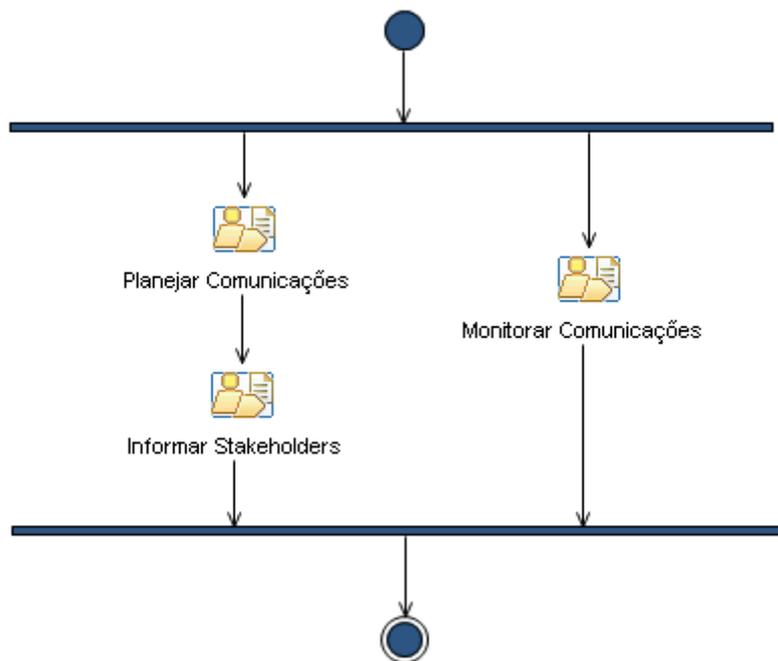


Figura 3.5 Gerenciamento Ágil das Comunicações

A Figura 3.5 apresenta a área de conhecimento Gerenciamento Ágil das Comunicações. Tendo como base as abordagens Scrum e XP os processos foram definidos conforme apresentado a seguir:

- **Planejar comunicações:** Determina as necessidades de informações e comunicações dos envolvidos no projeto.
- **Informar *Stakeholders*:** Diz respeito à disponibilização das informações para os envolvidos no projeto.
- **Monitorar Comunicações:** Identifica e resolve problemas nas comunicações do projeto.

3.7.1 Planejar Comunicações

Um fator importante para o sucesso do projeto é identificar as necessidades de informação das partes interessadas e determinar uma maneira adequada para atender a essas necessidades [1]. O planejamento das comunicações é iterativo, e quando os planos sofrem alterações essas mudanças precisam ser comunicadas. A Tabela 3.19 apresenta uma visão geral deste processo.

Tabela 3.19 Resumo do Processo Planejar Comunicações

Área de Conhecimento: Gerenciamento Ágil das Comunicações	
Grupo de Processos: Planejamento	
Processo: Planejar Comunicações	
Objetivo: Determina as necessidades de informação dos envolvidos, como obter informações e os métodos mais adequados para disseminá-las.	
Papéis: Gerente de projetos, desenvolvedor, analista de sistemas, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> Plano de Gerenciamento do Projeto 	<ul style="list-style-type: none"> Plano de Gerenciamento das Comunicações Mudanças solicitadas
Atividades	
<ul style="list-style-type: none"> Verificar requisitos das comunicações Analisar técnicas de comunicação Elaborar Plano de Gerenciamento das Comunicações 	
Referências: BECK, K. Extreme Programming Explained: Embrace Change [6], PMBOK® Guide [1], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/[40] .	

A forma como a organização trabalha, a frequência de reuniões com os clientes, o escopo do projeto e as limitações da sua execução são restrições de projeto e requisitos na elaboração do Plano de Comunicações, e estas restrições devem ser observadas durante a sua elaboração.

Durante a elaboração do plano, o gerente e a equipe devem discutir sobre as técnicas de comunicação mais aplicáveis ao projeto e à equipe de desenvolvimento. Consultar as experiências passadas, relatadas numa base de lições aprendidas de outros projetos é útil para auxiliar na decisão de quais técnicas e tecnologias serão utilizadas.

Após decidir a respeito das técnicas de comunicação e verificar quais as restrições e os requisitos para realização de comunicações, o gerente desenvolve um Plano de Gerenciamento das Comunicações. O plano deve conter o nível de detalhes suficiente, deve ser pouco formal, e pode ser baseado nos planos de comunicação desenvolvidos nos projetos anteriores e nas atividades realizadas anteriormente neste processo.

Devido ao caráter iterativo da elaboração do Plano de Gerenciamento das Comunicações, podem acontecer mudanças no plano e na forma de comunicação da equipe de projeto, por exemplo, a inserção de uma nova técnica como sugestão para distribuição de informações para a equipe. Qualquer mudança solicitada é gerenciada pelo processo Monitorar e Controlar Mudanças do Projeto.

3.7.2 Informar *Stakeholders*

Este processo está relacionado com a disponibilização das informações para envolvidos no projeto. Para isso se faz necessária a coleta e distribuição de informações sobre o projeto, incluindo informações sobre: desempenho da equipe, progresso do projeto, problemas encontrados, entre outros.

A coleta de informações diz respeito à obtenção de informações sobre o progresso e os problemas enfrentados durante a execução do projeto. Durante a iteração, reuniões curtas são realizadas diariamente, elemento chave na comunicação que mantém o time em sincronia e mais apto a resolver problemas. Essas reuniões diárias com a equipe do projeto, os relatórios de progresso e os relatórios de desempenho são utilizados para, além de monitorar os projetos, compartilhar informações na equipe. Reuniões semanais com o cliente, e a entrega e verificação de resultados auxiliam no processo de desenvolvimento do produto e o feedback do cliente apresenta a percepção do projeto que ele possui.

O armazenamento das informações do projeto é realizado em arquivos e em bancos de dados para posterior recuperação. Estes arquivos incluem as causas dos problemas, as ações corretivas utilizadas e as lições aprendidas.

Com o objetivo de disseminar informações aos interessados no projeto, são realizadas reuniões diárias com a equipe de projeto se possível reuniões semanais com clientes. As informações sobre o progresso das atividades de execução, bem como releases e documentação, podem ser disponibilizados, por exemplo, através de sites na intranet e através de painéis para melhorar a visibilidade da equipe. Dentro destas informações divulgadas, estão os relatórios de progresso que são relatórios de status da iteração distribuídos para a equipe de projeto e devem indicar resumidamente o progresso do projeto e o desempenho da equipe. Seguindo o Scrum, o gráfico *Sprint Chart Burndown* é indicado para informar o progresso da iteração.

Ao final da iteração uma Reunião de Retrospectiva da Iteração acontece e dela a equipe obtém um relatório que indica a duração da iteração, o esforço planejado e o esforço realizado por cada um dos componentes da equipe, e ainda apresenta gráficos indicativos do progresso da execução da iteração e sugestões de melhoria [42].

Mudanças que ocorram neste processo podem causar alterações no Plano de Projeto. Estas mudanças são solicitadas ao processo Monitorar e Controlar Mudanças do Projeto da área de

conhecimento Gerenciamento Ágil da Integração. A Tabela 3.20 apresenta a visão geral deste processo.

Tabela 3.20 Resumo do Processo Informar *Stakeholders*

Área de Conhecimento: Gerenciamento Ágil das Comunicações	
Grupo de Processos: Execução	
Processo: Informar <i>Stakeholders</i>	
Objetivo: Colocar informações a disposição dos envolvidos no projeto.	
Papéis: Gerente de projetos, desenvolvedor.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Plano de Gerenciamento das Comunicações 	<ul style="list-style-type: none"> • Relatórios de progresso • Divulgação de informações • Mudanças solicitadas
Atividades	
<ul style="list-style-type: none"> • Coletar informações sobre o projeto • Realizar Reuniões de Retrospectiva de Iteração • Armazenar informações • Divulgar informações 	
Referências: BECK, K. Extreme Programming Explained: Embrace Change [6], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/[40] .	

3.7.3 Monitorar Comunicações

Monitorar as comunicações refere-se ao gerenciamento do processo de comunicações do projeto no intuito de descobrir e resolver problemas de comunicação. Neste caso, um registro de problemas pode ser utilizado para documentar e monitorar a resolução de problemas [1]. Este artefato deve possuir a descrição dos problemas encontrados durante uma iteração.

Através do registro de problemas, o gerente de projetos indica um proprietário para o problema e estabelece uma data alvo para a sua solução. No processo Monitorar Comunicações, as mudanças aprovadas nas comunicações são implementadas em termos de ações corretivas. A Tabela 3.21 apresenta este processo de forma resumida.

Tabela 3.21 Resumo do Processo Monitorar Comunicações

Área de Conhecimento: Gerenciamento Ágil das Comunicações	
Grupo de Processos: Monitoramento e Controle	
Processo: Monitorar Comunicações	
Objetivo: Descobrir e resolver problemas de comunicação.	
Papéis: Gerente de projetos.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> Plano de Gerenciamento das Comunicações 	<ul style="list-style-type: none"> Registro de problemas Solicitações de mudança aprovadas
Atividades	
<ul style="list-style-type: none"> Registrar problemas de comunicação Resolver problemas de comunicação 	
Referências: BECK, K. Extreme Programming Explained: Embrace Change [6], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/ [40].	

3.8 Gerenciamento Ágil dos Riscos do Projeto

Um risco em um projeto é um evento ou condição incerta que, se ocorrer, terá um efeito positivo ou negativo sobre pelo menos um objetivo do projeto, como tempo, custo, escopo ou qualidade e ainda, pode ter uma ou mais causas e, se ocorrer, um ou mais impactos [1].

De acordo com [1], os objetivos do gerenciamento de riscos de projetos são aumentar a probabilidade e impacto de eventos positivos e diminuir a probabilidade e impacto de eventos adversos para o projeto. O *framework* do desenvolvimento ágil de software favorece estes objetivos por fazer o gerenciamento de riscos parte intrínseca do ciclo de vida do projeto [39]. Ou seja, o gerenciamento ágil dos riscos não descarta a visão contida em [1], mas procura melhorá-la.

Continuamente identificar, analisar, monitorar e responder as causas dos riscos e os eventos de risco é parte das discussões do planejamento iterativo do time ágil, que diz que os riscos são dirigidos a cada um no time. Reuniões diárias, Reuniões de Planejamento de Iteração, Retrospectivas e reuniões de revisão são locais para atividades de gerenciamento de riscos em projetos ágeis.

O gerenciamento ágil de riscos possui, segundo [43], dois desafios: ser integrado com outras atividades de planejamento de iteração, pois as considerações do gerenciamento de riscos conduzem o plano de iteração; e a abordagem adotada deve envolver todo o time e ser realizada rapidamente. Com base nos desafios apresentados acima, e utilizando as técnicas e práticas de Gerenciamento Ágil de Projetos encontradas em Scrum, APM e XP, os processos de Gerenciamento Ágil dos Riscos do Projeto foram definidos, conforme descrito a seguir e ilustrado na Figura 3.6:

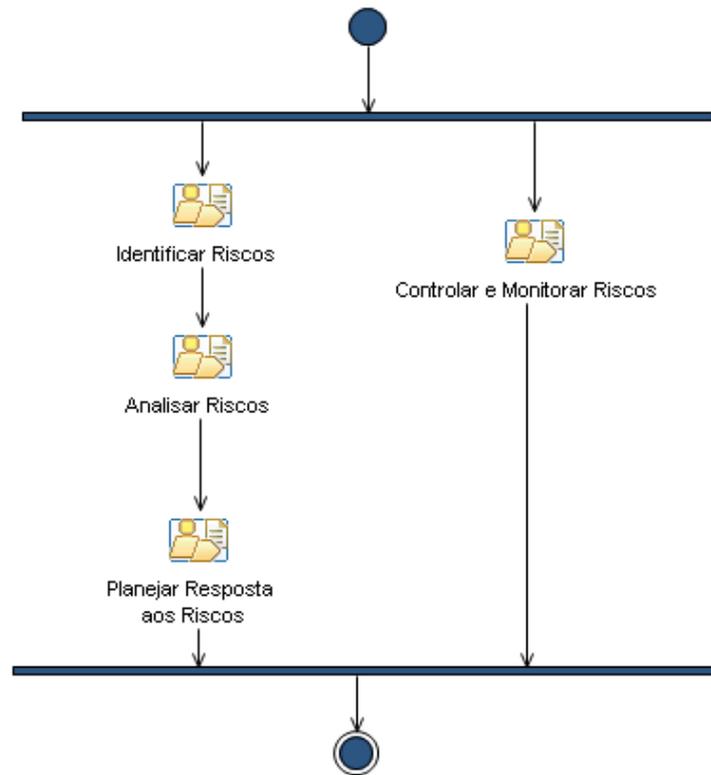


Figura 3.6 Gerenciamento Ágil dos Riscos

- **Identificar Riscos:** Determina os riscos que podem afetar o projeto.
- **Analisar Riscos:** Prioriza riscos para monitorá-los e controlá-los.
- **Planejar Resposta aos Riscos:** Desenvolve planos de ação para contornar os riscos.
- **Controlar e Monitorar Riscos:** Acompanha os riscos identificados, identifica novos riscos e executa planos de resposta aos riscos.

As próximas seções apresentam os quatro processos definidos em maiores detalhes.

3.8.1 Identificar Riscos

Os riscos são identificados, conforme realizado em [40], durante as reuniões diárias de acompanhamento do projeto, e nas Reuniões de Planejamento da Iteração. Se o time ágil está gerenciando os riscos publicamente, então um roteiro com itens para o time identificar e priorizar os riscos é incluído como parte da reunião, com os resultados influenciando o trabalho que está sendo planejado para aquela iteração e riscos continuam a ser identificados diariamente como parte das reuniões diárias [44].

A Tabela 3.22 apresenta uma visão geral do processo Identificar Riscos.

Tabela 3.22 Resumo do Processo Identificar Riscos

Área de Conhecimento: Gerenciamento Ágil dos Riscos	
Grupo de Processos: Planejamento	
Processo: Identificar Riscos	
Objetivo: Determinar os riscos que podem afetar o projeto.	
Papéis: Gerente de projetos, desenvolvedor, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • <i>Backlog</i> de Produto • Visão do Produto • Plano de Gerenciamento do Projeto • Plano de Gerenciamento do Escopo 	<ul style="list-style-type: none"> • <i>Backlog</i> de Impedimentos atualizado
Atividades	
<ul style="list-style-type: none"> • Coletar informações sobre riscos nas reuniões diárias e reuniões de planejamento da iteração • Realizar <i>Brainstorming</i> • Revisar itens do <i>Backlog</i> de Produto • Agrupar riscos por categoria 	
Referências: BECK, K. Extreme Programming Explained: Embrace Change [6], PRESTON, S., PICHLER, R. Agile Risks, Agile Rewards, disponível em: http://www.ddj.com/architect/184415308/ [43], SLIGER, M. Relating PMBOK Practices to Agile Practices--Part 3 of 4, disponível em: http://www.stickyminds.com/ [44], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/ [40].	

A definição dos itens do *Backlog* de Produto pode ser acompanhada da identificação dos riscos. Depois de descobrir os riscos, é conveniente para o entendimento e melhor organização que estes riscos sejam agrupados por categoria.

3.8.2 Analisar Riscos

Projetos ágeis executam apenas análise qualitativa, ciclos de desenvolvimento ágil pequenos e constantes revisões que tornam estas atividades factíveis e efetivas. Após identificação na primeira Reunião de Planejamento da Iteração, os riscos de todo o projeto devem ser analisados.

A cada um dos riscos, deve ser atribuído um fator de exploração conforme [3] sugere, que determina o impacto quando da ocorrência deste risco. A probabilidade de ocorrência do risco deve ser calculada com base na experiência e nos dados históricos de projetos passados. Após calcular o fator de exposição do risco, e a partir deste, os riscos devem ser colocados num ranking e em seguida priorizados. De acordo com [40], isto acontece nas Reuniões de Planejamento da Iteração e

os resultados são postados em lugar visível como uma constante lembrança para o time. O processo Analisar Riscos é resumido na Tabela 3.23.

Tabela 3.23 Resumo do Processo Analisar Riscos

Área de Conhecimento: Gerenciamento Ágil dos Riscos	
Grupo de Processos: Planejamento	
Processo: Analisar Riscos	
Objetivo: Analisar e priorizar os riscos identificados no projeto.	
Papéis: Gerente de projetos, desenvolvedor, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • <i>Backlog</i> de Impedimentos 	<ul style="list-style-type: none"> • <i>Backlog</i> de Impedimentos priorizado
Atividades	
<ul style="list-style-type: none"> • Atribuir Fator de Exploração • Realizar Reuniões de Planejamento de Iteração • Priorizar Riscos 	
Referências: HIGHSMITH, J. Agile project management: creating innovative products Erro! Fonte de referência não encontrada. , PRESTON, S., PICHLER, R. Agile Risks, Agile Rewards, disponível em: http://www.ddj.com/architect/184415308/ [43], SLIGER, M. Relating PMBOK Practices to Agile Practices--Part 3 of 4, disponível em: http://www.stickyminds.com/ [44], Scrum for Team System, disponível em: http://www.scrumforteamssystem.com/en/default.aspx/ [40].	

3.8.3 Planejar Resposta aos Riscos

No ambiente ágil, o time participa do desenvolvimento de opções e ações para reduzir ameaças, uma tarefa conduzida com mais frequência do que é comum em projetos tradicionais [44]. Abaixo, a Tabela 3.24 apresenta o processo em mais detalhes.

Tabela 3.24 Resumo do Processo Planejar Resposta aos Riscos

Área de Conhecimento: Gerenciamento Ágil dos Riscos	
Grupo de Processos: Planejamento	
Processo: Planejar Resposta aos Riscos	
Objetivo: Desenvolver planos de ação para contornar os riscos.	
Papéis: Gerente de projetos, desenvolvedor, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Backlog de Impedimentos 	<ul style="list-style-type: none"> • Planos de Ação
Atividades	
<ul style="list-style-type: none"> • Realizar Reuniões de Planejamento da Iteração • Criar Planos de Ação 	
Referências: Scrum for Team System, disponível em: http://www.scrumforteamssystem.com [40].PRESTON, S., PICHLER, R. Agile Risks, Agile Rewards, disponível em: http://www.ddj.com/architect/184415308/ [43], SLIGER, M. Relating PMBOK Practices to Agile Practices-Part 3 of 4, disponível em: http://www.stickyminds.com/ [44].	

Os planos de ação correspondem a todas as tarefas que devem ser executadas para mitigar os riscos e estes planos devem ser criados durante as Reuniões de Planejamento da Iteração.

3.8.4 Controlar e Monitorar Riscos

Auditoria de riscos, análise de tendência e medições de performance técnica são conduzidas ao fim de cada iteração ágil como parte da Reunião de Retrospectiva da Iteração. Estas reuniões proporcionam à equipe um fórum para o time revisar o *Sprint Burndown Chart*, velocidade do time, e outros artefatos que apresentam o progresso do projeto.

A reavaliação do risco pode acontecer durante a Reunião de Retrospectiva de Iteração ágil, onde riscos anteriores ou preocupações podem ser revisitados como parte da determinação de mudanças que precisam ser feitas no futuro. Reuniões diárias de acompanhamento contribuem para o constante processo de monitoramento por expor as causas de riscos potenciais e novos obstáculos.

O trabalho do projeto deve ser controlado e monitorado continuamente para encontrar novos riscos e mudanças nos riscos. O controle dos riscos é realizado através da execução de planos de ação, que possuem o objetivo de mitigar os riscos. Durante as Reuniões de Planejamento da Iteração a equipe escolhe um subconjunto de riscos que equivale à iteração para ser monitorado. Diariamente riscos são monitorados durante as reuniões diárias de acompanhamento. A Tabela 3.25 resume este processo.

Tabela 3.25 Resumo do Processo Controlar e Monitorar Riscos

Área de Conhecimento: Gerenciamento Ágil dos Riscos	
Grupo de Processos: Monitoramento e Controle	
Processo: Controlar e Monitorar Riscos	
Objetivo: Acompanhar os riscos identificados, identificar novos riscos e executar planos de resposta aos riscos.	
Papéis: Gerente de projetos, desenvolvedor, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • <i>Backlog</i> de Impedimentos 	<ul style="list-style-type: none"> • <i>Backlog</i> de Impedimentos atualizado
Atividades	
<ul style="list-style-type: none"> • Executar Planos de Ação • Gerenciar <i>Backlog</i> de Impedimentos • Realizar reuniões diárias • Realizar reuniões de planejamento da iteração 	
Referências: Scrum for Team System, disponível em: http://www.scrumforteamssystem.com/ [40].PRESTON, S., PICHLER, R. Agile Risks, Agile Rewards, disponível em: http://www.ddj.com/architect/184415308/ [43], SLIGER, M. Relating PMBOK Practices to Agile Practices--Part 3 of 4, disponível em: http://www.stickyminds.com/ [44],	

3.9 Gerenciamento Ágil dos Custos do Projeto

O gerenciamento de custos do projeto inclui os processos envolvidos em planejamento, estimativa, orçamentação e controle de custos, de modo que seja possível terminar o projeto dentro do orçamento aprovado. Custo, escopo e tempo estão fortemente relacionados, e dependem do entendimento claro dos requisitos do usuário para serem estimados com mais precisão. Escopos mal definidos por problemas de requisitos também mal-entendidos geram problemas de custos nas estimativas no início, no planejamento, na execução e no controle do projeto; e, conseqüentemente, os custos no final do projeto tenderão a aumentar muito e extrapolar o orçamento previsto.

Uma outra razão para a variação de custos em projetos de tecnologia da informação é quando estes envolvem novas tecnologias. Qualquer tecnologia nova que ainda não foi testada exhaustivamente traz consigo riscos herdados. O ideal é não fornecer nenhuma informação sobre o custo de projeto para o cliente sem antes validar por completo o entendimento dos requisitos e do escopo com os usuários e sem antes avaliar as tecnologias disponíveis no momento. Esse processo deve envolver o gerente do projeto, os membros do time do projeto e o usuário.

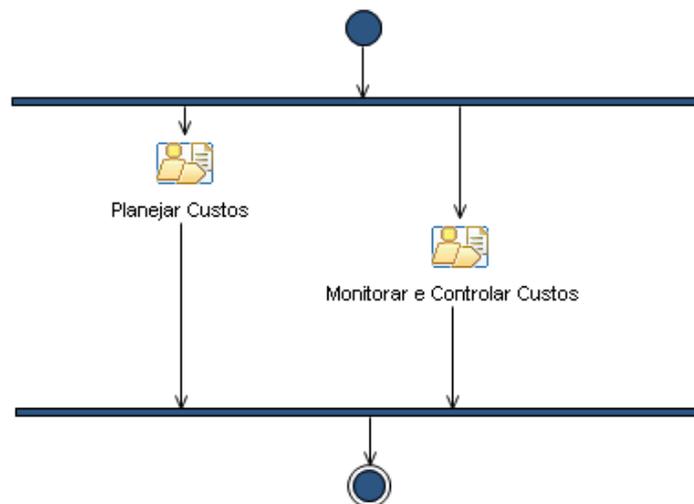


Figura 3.7 Gerenciamento Ágil dos Custos

A Figura 3.7 apresenta a área de conhecimento Gerenciamento Ágil Custos, composta pelos processos apresentados a seguir:

- **Planejar Custos:** Estima custos e desenvolve orçamento para que as atividades terminem no cronograma.

- **Controlar e Monitorar Custos:** Procura garantir que as mudanças nos custos tenham pequeno impacto no projeto.

As próximas seções apresentam os quatro processos definidos em maiores detalhes.

3.9.1 Planejar Custos

O processo Planejar Custos envolve atividades de planejamento, estimativas e orçamento do projeto. A estimativa de custos das atividades do cronograma funciona como uma aproximação dos custos dos recursos necessários para terminar as atividades no tempo estabelecido pelo cronograma.

Em projetos tradicionais é natural que sejam realizadas estimativas considerando que custos, prazo e escopo serão previsíveis. Assim é possível fazer com que o cliente saiba exatamente o que receberá, quando e por que preço. Contudo, devido ao problema específico da imprevisibilidade nos projetos de software, aliada ao fato de o cliente muitas vezes não saber expressar exatamente o que deseja, no Agilius o escopo e os custos não são fixos em contrato. Isto significa que mudanças podem ser incluídas para que o software evolua ao longo do projeto. O escopo é revisado para garantir que equipe dedique seus esforços ao que é prioritário em cada etapa do projeto e o contrato é negociável.

Contrato de escopo negociável [6] é um acordo baseado na premissa de que não existe previsibilidade sobre o que será feito no software, e também uma forma de alinhar os interesses dos *stakeholders*. Assim, numa data pré-determinada pelo cliente, é possível garantir que se tenha, no mínimo, as funcionalidades que mais irão gerar valor para ele. Ao final, ele tem um software que atende as suas necessidades e prioridades reais e não às que ele achava que tinha no início do projeto.

O cliente tem o direito de reincidir um contrato deste tipo a cada dois meses sem pagar multa, ou outro período, que pode variar a cada contrato. Isto significa que o projeto pode ser finalizado se, na opinião do cliente, não for realizado um bom trabalho. Ao se trabalhar com iterações semanais, a cada dois meses serão realizadas oito iterações, e, portanto oito entregas de funcionalidades. Deste modo, o cliente poderá verificar, a partir das funcionalidades, se a equipe está trabalhando de acordo com seus interesses. E a cada dois meses (ou a cada período definido no contrato) o cliente pode decidir se continua o projeto, ou se realiza alguma mudança na equipe de trabalho.

A seguir, a Tabela 3.26 apresenta um resumo do processo Planejar Custos.

Tabela 3.26 Resumo do Processo Planejar Custos

Área de Conhecimento: Gerenciamento Ágil dos Custos	
Grupo de Processos: Planejamento	
Processo: Planejar Custos	
Objetivo: Estimar custos e desenvolver orçamento para que as atividades terminem dentro do cronograma.	
Papéis: Gerente de projetos, desenvolvedor, cliente.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Plano de Gerenciamento do Projeto • Visão do Produto • FBS • Cronograma • Calendário de Recursos 	<ul style="list-style-type: none"> • Estimativa de custos • Plano de Gerenciamento de Custos • Contrato de Escopo Negociável • Mudanças solicitadas
Atividades	
<ul style="list-style-type: none"> • Estimar custos das tarefas • Estimar custos das iterações • Estimar custos dos recursos • Analisar reservas • Agregar custos 	
Referências: BECK, K. Extreme Programming Explained: Embrace Change [6], PMBOK® Guide [1], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/[40] .	

O Plano de Gerenciamento de Custos é um plano no qual os recursos alocados são associados aos seus respectivos custos. Desta maneira, elaborar um orçamento significa alocar recursos escassos provenientes de várias fontes em uma organização. O gerente deve considerar o custo do desenvolvimento do software, o custo de instalação, e o custo de manutenção na elaboração do Plano de Gerenciamento de Custos.

Durante o desenvolvimento do plano, o gerente consulta e avalia a Visão do Produto, o cronograma, a *Feature Breakdown Structure* e o calendário de disponibilidade dos recursos do projeto. A partir desta avaliação, toda a equipe juntamente com o cliente estima os custos do projeto.

3.9.2 Controlar e Monitorar Custos

O processo Controlar e Monitorar Custos está associado a influenciar os fatores que criam as mudanças na meta de custo de forma a garantir que estas mudanças sejam benéficas, determinar que a meta de custo foi alterada, e gerenciar as mudanças reais quando e na forma que elas surgirem. A Tabela 3.27 exhibe uma visão geral do funcionamento deste processo.

Tabela 3.27 Resumo do processo Monitorar e Controlar Custos

Área de Conhecimento: Gerenciamento Ágil dos Custos	
Grupo de Processos: Monitoramento e Controle	
Processo: Controlar e Monitorar Custos	
Objetivo: Controlar as variações dos custos.	
Papéis: Gerente de projetos.	
Artefatos de Entrada	Artefatos e Resultados Esperados
<ul style="list-style-type: none"> • Plano de Gerenciamento do Projeto • Plano de Gerenciamento dos Custos • Relatórios de desempenho • <i>Sprint Burndown Charts</i> • Solicitações de mudanças aprovadas 	<ul style="list-style-type: none"> • Estimativa de custos atualizada • Mudanças solicitadas • Plano de Gerenciamento dos Custos atualizado
Atividades	
<ul style="list-style-type: none"> • Analisar progresso do projeto • Monitorar custos • Controlar mudanças nos custos 	
Referências: PMBOK® Guide [1], Scrum for Team System, disponível em: http://www.scrumforteamsystem.com/en/default.aspx/[40] .	

O controle de custos procura as causas das variações positivas e negativas, por exemplo, respostas inadequadas às variações de custos que podem causar problemas de qualidade ou de cronograma ou produzir posteriormente um nível de risco inaceitável no projeto [1]. Deve estar fortemente integrado com os outros processos de controle (o controle de mudança de escopo, o controle do cronograma e o controle da qualidade).

Para controlar os custos de um projeto é necessário que este seja monitorado. Durante esta atividade, atenção especial deve ser dada aos relatórios de desempenho, ao cronograma, e ao Plano de Gerenciamento de Custos.

3.10 Considerações

Neste capítulo foram apresentadas as áreas de conhecimento trabalhadas na abordagem proposta por esta dissertação, a metodologia de trabalho utilizada na criação da abordagem, e uma visão geral do *Eclipse Process Framework Composer*, utilizado para modelar o processo como um todo.

Inicialmente procurou-se discutir a respeito do EPF *Composer*, que em conjunto com o SPEM 2.0, tornaram possível atividade de modelagem. O uso do EPF *Composer* ainda garante que a definição criada possa ser reutilizada no momento de adaptação desta abordagem e criação de um processo de gestão específico para um determinado projeto ou empresa. Algumas vantagens no uso

desta ferramenta são a flexibilidade na acomodação de mudanças e a praticidade na disseminação de um processo, pois a ferramenta torna possível a geração de um web site.

No que diz respeito aos processos, eles foram apresentados na tentativa de justificar as técnicas e práticas utilizadas na sua concepção. A seguir, no capítulo de experimentação, o Agilius será discutido sob uma ótica de avaliação, analisado a fim de verificar a sua utilidade e a partir de uma avaliação, pontos de melhoria serão apresentados.

Avaliação do Agilius

Experimentação é o centro do processo científico, pois somente experimentos verificam teorias. A importância desta atividade se verifica na capacidade que ela possui de explorar fatores críticos e dar luz ao fenômeno novo para que teorias possam ser formuladas e corrigidas [11]. O método experimental, dentro do estudo da condução de experimentos na área de Engenharia de Software e de acordo com [11], sugere o modelo, desenvolve o método qualitativo ou quantitativo, realiza um experimento, mede e analisa, avalia o modelo e dá subsídios para que se repita o processo. O objetivo do experimento, então, é manipular uma ou algumas variáveis e manter as outras fixas medindo o efeito do resultado.

Para a avaliação da abordagem proposta nesta dissertação, foi realizada a experimentação dos processos existentes no Agilius e a verificação da adequação dos resultados obtidos. Para tal, as fases [11]: **Definição dos Objetivos**, onde são apresentados os objetivos do experimento; **Planejamento** na qual o projeto do experimento é determinado, a instrumentação é considerada e os aspectos da validade do experimento são avaliados; **Execução** onde os dados experimentais são coletados para serem analisados e avaliados; e **Análise e Interpretação** foram definidas.

Este capítulo apresenta o experimento realizado e os detalhes de execução de cada uma destas fases.

4.1 Definição dos Objetivos

Esta fase descreve o objetivo global, o objeto da medição e o objetivo do estudo experimental. A Definição dos Objetivos é uma preparação para a fase de Planejamento. Como resultado fornece: a direção geral do experimento, o seu escopo, a base para a formulação das hipóteses e as notações preliminares para a avaliação da validade.

4.1.1 Objetivo Global

O objetivo geral deste experimento é avaliar a abordagem Agilius, do ponto de vista do gerente de projetos, a fim de verificar se ela oferece suporte ao gerenciamento ágil de projetos de software em termos dos processos utilizados para tal. Este trabalho considera os princípios estabelecidos no manifesto ágil e se propõe a apresentar um conjunto de processos com

características que sigam estes princípios, ou seja, o objetivo global da avaliação é verificar se os processos do Agilius possuem a flexibilidade de se adaptarem a mudanças no decorrer de um projeto.

4.1.2 Objetivo da Medição

Segundo [11], a medição é a parte central de um estudo experimental, sendo definida como o mapeamento do mundo experimental para o mundo formal. O objetivo principal desse mapeamento é caracterizar e manipular os atributos das entidades empíricas de maneira formal. Neste contexto, tendo como base as metodologias ágeis de gerenciamento e o PMBOK® *Guide*, o objetivo da medição neste trabalho é caracterizar:

1. Quais as características dos processos propostos pelo Agilius que estão disponíveis para os gerentes:
 - quais as características que os processos oferecidos pelo Agilius possuem que os gerentes de projeto consideram úteis para o gerenciamento ágil de um projeto de software;
 - quais as características que os processos oferecidos pelo Agilius possuem que os gerentes de projetos consideram inúteis para o gerenciamento ágil de um projeto de software.
2. Quais as características dos processos propostos pelo Agilius que oferecem aos gerentes de projeto conteúdo inadequado para o gerenciamento ágil de projetos:
 - quais as características que os processos oferecidos pelo Agilius possuem que necessitam de um melhor detalhamento;
 - quais as características que os processos oferecidos pelo Agilius possuem que apresentam um detalhamento excessivo.
3. Quais as características dos processos propostos pelo Agilius que os gerentes gostariam que estivessem à sua disposição além das já oferecidas pelo Agilius.

A Figura 4.1 ilustra a caracterização dos objetivos de medição das características dos processos propostos pelo Agilius. Ela apresenta a intersecção entre características de processos de gerenciamento clássico e características dos processos sugeridos pela abordagem proposta, as características consideradas úteis e inúteis após a realização do experimento.

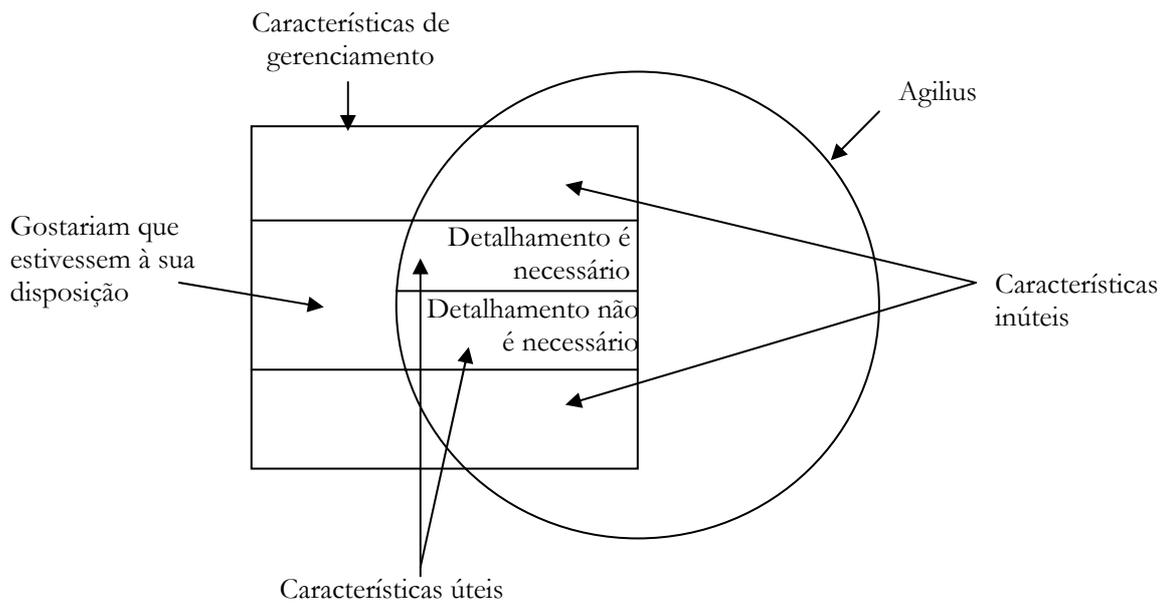


Figura 4.1 Caracterização dos objetivos de medição da experimentação

4.1.3 Objetivo do Estudo Experimental

De acordo com os objetivos da medição que foram listados anteriormente, a Tabela 4.1, que contém uma visão mais detalhada do estudo realizado, foi construída.

Tabela 4.1 Caracterização do estudo experimental

ATRIBUTO	VALOR PARA O EXPERIMENTO
Analisar (Objeto do Estudo)	a abordagem proposta pelo Agilius em termos de características ágeis dos processos propostos
Com o propósito de (Objetivo)	avaliar e melhorar
Com respeito à (Foco da Qualidade)	abordagens de gerenciamento ágil de projetos de software e PMBOK® <i>Guide</i>
Do ponto de vista (Perspectiva)	do gerente de projetos
No contexto de (Amostra de pesquisa)	profissionais formados em Ciência da Computação com experiência em Gerenciamento de Projetos

4.1.4 Questões

A partir dos objetivos de medição definidos na Seção 4.1.2 e utilizando o método GQM – *Goal, Question, Metric* [45], abaixo são apresentadas as questões, construídas com base no objetivo da medição, e as métricas derivadas, que servirão para receber os dados experimentais e formular respostas para estas questões.

Questão 1 (Q₁): Existem características importantes para a definição de processos gerenciamento ágil de software que não fazem parte da abordagem Agilius?

Métrica 1 (M₁): A lista de características importantes para a definição de processos gerenciamento ágil de projetos de software que não fazem parte da abordagem Agilius.

Questão 2 (Q₂): Existem características importantes para a definição de processos de gerenciamento ágil de software e oferecidas pela abordagem Agilius que são considerados inúteis pelos gerentes?

Métrica 2 (M₂): A lista de características importantes para a definição de processos de gerenciamento ágil de projetos de software que fazem parte da abordagem Agilius e que são consideradas inúteis pelos gerentes.

Questão 3 (Q₃): Existem características importantes para a definição de processos de gerenciamento ágil de projetos de software que fazem parte da abordagem Agilius e que são consideradas úteis pelos gerentes, cujo detalhamento deve ser modificado?

Métrica 3 (M₃): A lista de características importantes para a definição de processos de gerenciamento ágil de projetos de software que fazem parte da abordagem Agilius e que são consideradas úteis pelos gerentes, cujo detalhamento deve ser modificado.

Questão 4 (Q₄): Existem características importantes para a definição de processos de gerenciamento ágil de projetos de software que não fazem parte da abordagem Agilius, mas que os gerentes gostariam que estivessem disponíveis porque consideram necessárias para o gerenciamento ágil?

Métrica 4 (M₄): A lista de características importantes para a definição de processos de gerenciamento ágil de projetos de software que não fazem parte da abordagem Agilius, mas que os gerentes gostariam que estivessem disponíveis porque consideram necessárias para a definição de processos de gerenciamento ágil de projetos de software.

4.2 Planejamento

A fase de Planejamento é responsável pela seleção do contexto, a formulação das hipóteses, a seleção das variáveis, a seleção dos participantes, o projeto do experimento, a preparação conceitual da instrumentação, e a consideração da validade do experimento. O resultado desta fase apresenta o experimento totalmente elaborado e pronto para execução [11].

4.2.1 Definição das Hipóteses

Um experimento geralmente é formulado através de hipóteses. Em estatística, uma hipótese é uma afirmativa sobre uma propriedade da população [46]. A hipótese principal se chama hipótese nula e ela indica uma afirmativa de que o valor de um parâmetro é igual a algum valor especificado [46]. O teste sobre a hipótese nula é realizado no sentido de supor que a mesma é verdadeira. Ao final do teste, chega-se a uma conclusão que levará a rejeitar ou deixar de rejeitar H_0 .

O objetivo deste experimento é rejeitar a hipótese nula a favor de uma ou algumas hipóteses alternativas. Assim sendo, para esta experimentação, definiu-se o seguinte conjunto de hipóteses:

Hipótese Nula (H_0): A lista de características de processos de gerenciamento ágil de projetos de software oferecida aos gerentes de projeto que utilizam o Agilius não é similar à lista de características utilizadas para o gerenciamento deste tipo de projeto.

Se P_p representa as características dos processos de gerenciamento oferecidos aos usuários do Agilius e P_g a lista características de processos de gerenciamento utilizadas para gerenciar projetos com agilidade, então:

$$H_0 : P_g - (P_p \cap P_g) \neq \emptyset$$

Hipótese Alternativa (H_1): A lista de características de processos de gerenciamento ágil de projetos de software oferecida aos gerentes de projeto que utilizam o Agilius é similar à lista de características de processos utilizadas para o gerenciamento deste tipo de projeto.

P_p representa a lista de características de processos de gerenciamento oferecidas aos usuários do Agilius e P_g a lista de características de processos de gerenciamento utilizadas para gerenciar projetos com agilidade.

$$H_1 : P_g - (P_p \cap P_g) = \emptyset$$

Hipótese Alternativa (H_2): Na lista de características de processos de gerenciamento ágil de projetos de software oferecida aos gerentes de projeto que utilizam o Agilius, e que fazem parte da lista de características de processos importantes para o gerenciamento deste tipo de projeto, existem características que os gerentes consideram inúteis para esta tarefa.

Onde P_p representa a lista de características de processos de gerenciamento oferecidas aos usuários do Agilius e que fazem parte da lista de características de processos

importantes para o gerenciamento deste tipo de projeto e P_{pu} , a lista de características de processos de gerenciamento oferecidas aos usuários do Agilius e que fazem parte da lista de características de processos importantes para o gerenciamento deste tipo de projeto e que os usuários consideram úteis para esta tarefa.

$$H_2 : P_p - (P_p \cap P_{pu}) \neq \emptyset$$

Hipótese Alternativa (H₃): Na lista de características de processos de gerenciamento ágil de projetos de software oferecida aos gerentes de projeto que utilizam o Agilius, que fazem parte da lista de características de processos importantes para o gerenciamento deste tipo de projeto e que são consideradas úteis para esta tarefa, existem de características de processos cujo detalhamento deve ser modificado para atingir o nível esperado pelos gerentes de projeto que realizam gerenciamento ágil de projetos.

Onde P_{pu} representa a lista de características de processos de gerenciamento oferecidas aos usuários do Agilius e que fazem parte da lista de características de processos importantes para o gerenciamento deste tipo de projeto e que os usuários consideram úteis para esta tarefa e P_{pum} representa a lista de características de processos de gerenciamento oferecida aos usuários do Agilius, que fazem parte da lista de características de processos importantes para o gerenciamento deste tipo de projeto e são consideradas úteis para esta tarefa cujo detalhamento não precisa de modificação.

$$H_3 : P_{pu} - (P_{pu} \cap P_{pum}) \neq \emptyset$$

Hipótese alternativa (H₄): Na lista de características de processos de gerenciamento não oferecidas para os usuários do Agilius existem características de processos que os gerentes gostariam de receber.

Onde P_{np} representa a lista de características de processos não oferecidas para os usuários do Agilius e P_r representa a lista de características de processos não oferecidas para os usuários do Agilius que os gerentes de projeto gostariam de receber.

$$H_4 : P_{np} - (P_{np} \cap P_r) \neq \emptyset$$

Dadas as hipóteses à cima, o melhor caso será não rejeitar a hipótese nula. Se esta situação for verificada, representa uma total similaridade entre as características de gerenciamento ágil apresentadas no Agilius e à lista de características utilizadas para o gerenciamento deste tipo de projeto.

4.2.2 Descrição da Instrumentação

Para cada característica que o Agilius considera fundamental para o desenvolvimento de um processo de gerenciamento de projetos de software, deve-se oferecer a escolha representada pela Tabela 4.2.

Tabela 4.2 Critérios de instrumentação do experimento

PRESENÇA DA CARACTERÍSTICA (P)	UTILIDADE DA CARACTERÍSTICA (U)	ADEQUAÇÃO DO NÍVEL DE DETALHAMENTO DA CARACTERÍSTICA (A)
1. Não é oferecida e não gostaria que estivesse disponível. 2. Não é oferecida, mas gostaria que estivesse disponível. 3. Oferecida, parcialmente, 4. Oferecida.	1. Não é útil. 2. Provavelmente é útil, mas ainda não apliquei. 3. É útil e já apliquei em diferentes projetos.	1. O detalhamento deve ser aumentado. 2. O detalhamento não precisa ser modificado. 3. O detalhamento deve ser diminuído.

Para cada característica, deve-se aplicar o teste estatístico Chi-2, utilizado para definir: (a) se pode considerar que a característica é fornecida; (b) se pode considerar que a característica é útil; (c) se pode considerar que o detalhamento da característica não precisa de modificação.

Tabela 4.3 Possíveis métricas para PUA

Nº	P	U	A	DESCRIÇÃO DA CARACTERÍSTICA	QUESTÕES
1	0	0	0	Não é oferecida, não é útil, a modificação não é necessária.	Q1, Q4
2	0	0	1	Não é oferecida, não é útil, a modificação é necessária.	N/A
3	0	1	0	Não é oferecida, é útil, a modificação não é necessária.	Q1, Q4
4	0	1	1	Não é oferecida, é útil, a modificação é necessária.	Q1, Q4
5	1	0	0	É oferecida, não é útil, a modificação não é necessária.	Q2
6	1	0	1	É oferecida, não é útil, a modificação é necessária.	Q2
7	1	1	0	É oferecida, é útil, a modificação não é necessária.	Q3
8	1	1	1	É oferecida, é útil, a modificação é necessária.	Q3

O resultado será composto pelo número de características com valores PUA. Desta forma, presença (P) recebe os valores: 0 – não oferecida e 1 – oferecida; utilidade (U): 0 – não é útil; 1 – é útil; adequação (A): 0 – o nível é adequado; 1 – o nível não é adequado. Isto permite um conjunto de

possíveis combinações de valores para PUA relacionados às métricas do experimento e a formulação de respostas para as questões de medição descritas na Seção 4.1.4, como visto na Tabela 4.3.

4.2.3 Seleção do Contexto

O contexto do experimento é composto das condições em que o experimento está sendo executado e pode ser caracterizado segundo quatro dimensões [11]:

- **O processo:** on-line / off-line;
- **Os participantes:** alunos / profissionais;
- **Realidade:** problema real / modelado;
- **Generalidade:** específico / geral.

Dessa forma este estudo supõe que o processo de experimentação seja **off-line**, porque os usuários não estão sendo entrevistados durante o desenvolvimento de um processo gerenciamento de projetos de software, mas em certo instante. Os participantes são **profissionais** especialistas que possuem conhecimentos em gerenciamento de projetos. O estudo é um **problema modelado** porque processos de gerenciamento de software não são desenvolvidos durante a resolução de um problema real, mas utilizando notas subjetivas. Os processos do Agilius são comparados com os processos considerados fundamentais para o gerenciamento deste tipo de projeto, então, o contexto possui o caráter **específico**.

4.2.4 Seleção dos Indivíduos

Como participantes para o estudo foram utilizados oito profissionais graduados do curso de Informática/Ciência da Computação, que já exerceram a atividade de gerenciamento de projetos, sendo sete mestres e um especialista. Estes indivíduos possuem conhecimento em processos de gerenciamento de projetos de software e no PMBOK® *Guide*. O conhecimento em gerenciamento ágil é moderado, visto que alguns declararam baixo conhecimento na área. Os participantes responderam a um questionário com o objetivo de caracterizar sua formação do ponto de vista acadêmico, experiência, tipo de curso entre outros para analisar os dados e reduzir o viés.

4.2.5 Variáveis

Existem dois tipos de variáveis em um experimento: as variáveis dependentes e as independentes. As variáveis independentes se referem à entrada do processo de experimentação.

Essas variáveis são conhecidas como “fatores” e apresentam a causa que afeta o processo de experimentação. As variáveis dependentes se referem à saída do processo de experimentação. Elas apresentam o efeito que é causado pelos fatores do experimento. Desta forma, para esta experimentação foram definidas:

- **Variáveis Independentes**

A lista de características dos processos de gerenciamento ágil de projetos considerada fundamental para o gerenciamento deste tipo de projetos.

- **Variáveis Dependentes**

1. A similaridade entre as características dos processos oferecidos pelo Agilius e as características dos processos utilizados no gerenciamento ágil de projetos. Pode receber os valores:

Igual, quando todas as características recebem o valor PUA = {1,X,X}, representado pelas métricas de 5 a 8 da Tabela 4.3, que apresentam valores 1 (característica oferecida), X (não importa o valor para utilidade) e X (não importa o valor para adequação).

Diferente, quando todas as características recebem o valor PUA = {0,X,X}, representado pelas métricas de 1 a 4 da Tabela 4.3, que apresentam valores 0 (característica não é oferecida), X (não importa o valor para utilidade) e X (não importa o valor para adequação).

Similar, quando não se cumprem as condições de “Igual” e “Diferente”. O grau de similaridade pode ser avaliado pela fórmula (1) que indica a porcentagem de características similares dentre o conjunto de características iguais e diferentes.

$$\{1,X,X\}/(\{0,X,X\} + \{1,X,X\}) \times 100 \quad (1)$$

2. A utilidade das características similares. Mostra a parte útil das características dos processos de gerenciamento ágil de projetos oferecidos pelo Agilius, calculada pelas fórmulas (2) e (3). A parte útil diz respeito a porcentagem de características oferecidas consideradas úteis ({1,1,X}) dentre as características oferecidas ({1,X,X}). A parte inútil é o complemento da parte útil.

Parte útil: $\{1,1,X\} / \{1,X,X\} \times 100\%$ (2)

Parte inútil: $\{1,0,X\} / \{1,X,X\} \times 100\%$ (3)

3. A adequação das características similares. Mostra a parte adequada das características dos processos de gerenciamento ágil de projetos oferecidos pelo Agilius, através das fórmulas (4) e (5). A parte adequada diz respeito à quantidade de características oferecidas e que não necessitam de modificação ($\{1,X,0\}$) dentre as características oferecidas ($\{1,X,X\}$). A parte inadequada é o complemento da parte adequada.

$$\text{Parte adequada: } \{1,X,0\} / \{1,X,X\} \times 100\% \quad (4)$$

$$\text{Parte inadequada: } \{1,X,1\} / \{1,X,X\} \times 100\% \quad (5)$$

4.2.6 Análise Qualitativa

Para analisar a informação referente aos processos de gerenciamento de projetos de software não oferecidos aos usuários do Agilius, mas que os usuários gostariam que estivessem disponíveis para o gerenciamento deste tipo de projeto, se propõe aplicar a análise qualitativa. Essa análise deve apresentar a lista de processos de gerenciamento ágil de projetos de software considerada fundamental para o gerenciamento deste tipo de projeto, cujos processos não são oferecidos aos usuários do Agilius, mas que estes usuários consideram fundamentais para esta tarefa e gostariam que estivessem disponíveis. Assim, esta análise deve considerar serviços com valor PUA = $\{0,X,X\}$ (representado pelas Métricas 1 à 4) e a opção “Não é oferecida, mas gostaria que estivesse disponível.” para “Presença da atividade”.

4.2.7 Validade

É questão fundamental saber se os resultados de um experimento são válidos e o quão válidos são eles. Os resultados devem ter validade para a população a qual o conjunto de participantes pertence. É interessante também generalizar os resultados para uma população mais ampla, pois irão possuir a validade adequada se forem válidos para esta população.

Há quatro tipos de validade de resultados de experimentos: validade de conclusão, validade interna, validade de construção e validade externa. A validade de conclusão está relacionada com a habilidade de chegar a uma conclusão correta a respeito dos relacionamentos entre o tratamento e o resultado do experimento. A validade interna define se o relacionamento observado entre o tratamento e o resultado é causal, e não é o resultado da influência de outro fator que não é controlado ou mesmo não foi medido. A validade de construção considera os relacionamentos entre a teoria e a observação, ou seja, se o tratamento reflete bem causa e o resultado reflete bem o efeito. E por fim a validade externa define as condições que limitam a habilidade de generalizar os

resultados de um experimento para a prática industrial. Deste modo, para o experimento foram definidas:

- **Validade Interna:** Como mencionado em “Seleção de indivíduos”, para o estudo foram selecionados participantes com conhecimento em gerenciamento de projetos. Assim, assume-se que eles são representativos para a população dos gerentes de projeto com conhecimento em desenvolvimento de processos de gerenciamento de projetos de software.
- **Validade de Conclusão:** Para receber os valores da presença, utilidade e conformidade o teste binomial será utilizado. A verificação de hipótese será feita por meio de simples demonstração de presença ou não de características de processos de gerenciamento ágil de projetos de software nas listas que representam as variáveis independentes.
- **Validade de Construção:** Esse estudo está caracterizado pela conformidade dos processos de gerenciamento de projetos de software com os processos reais oferecidos pelo Agilius. As características descritas representam a lista de características de processos que os especialistas na área de gerenciamento de projetos de software devem possuir para mostrar o desempenho adequado no ponto de vista da definição deste tipo de processo. As características de processos que possuem o maior relacionamento com o gerenciamento ágil de projetos de software do ponto de vista dos pesquisadores foram escolhidas.
- **Validade Externa:** Como foi mencionado em “Seleção de Indivíduos” e em “Validade Interna”, os participantes do estudo em geral podem ser considerados representativos para a população dos graduados em Informática/Ciência da Computação com conhecimento em gerenciamento de projetos de software. Para avaliação do nível de envolvimento no processo de gerenciamento de projetos de software, os dados do questionário, conforme a experiência dos participantes, podem ser analisados. Os materiais utilizados no estudo podem ser considerados representativos e “em tempo” para o problema sob análise, porque se compõem da lista de características de processos de gerenciamento de projetos de software considerados fundamentais para a execução de projetos de software. As características temporárias não devem ser consideradas um problema, pois os materiais dão a possibilidade de conduzir o estudo durante o tempo estipulado para a apresentação da abordagem proposta. Além disso, o estudo foi realizado quando os participantes estiveram envolvidos no gerenciamento de algum projeto de software.

4.3 Operação

A fase de operação, ou execução do experimento, possui a presença do fator humano como aspecto mais importante. Por isso, os participantes devem ser preparados para o experimento do ponto de vista moral e metodológico para evitar resultados errôneos devido ao mal-entendimento ou à falta de interesse [11]. Assim, a execução do experimento e a posterior coleta de dados devem ser realizadas de maneira que não causem efeito significativo para o processo estudado.

4.3.1 Execução do Estudo

Inicialmente os participantes receberam uma motivação para o uso do Agilius e um treinamento nas atividades de definição de processo de gerenciamento ágil de software nele disponíveis. Para a realização do estudo, oito participantes, conforme detalhado na

Tabela 4.4, realizaram o preenchimento dos questionários Perfil do Participante e Questionário de Avaliação, a fim de prover um conjunto de dados para análise.

4.3.2 Resultado do Estudo

Os artefatos do experimento incluem a descrição da instrumentação usada para coleta dos dados puros durante a execução do experimento. Os resultados incluem a descrição detalhada dos dados obtidos e podem ser apresentados como: dados puros, dados refinados e os dados considerados válidos.

A Tabela 4.4 foi elaborada de acordo com os dados obtidos no preenchimento do questionário Perfil do Participante. Para o preenchimento desta tabela, usou-se a legenda apresentada na Tabela 4.5, que retrata os itens de experiência dos participantes.

De acordo com a Tabela 4.4, todos os participantes do experimento fazem parte de instituição Pública de ensino, graduação do curso de Informática/Ciência da Computação. A maioria dos participantes considera alto o seu entendimento sobre desenvolvimento de produto de software e já exerceu, ou exerce a atividade de Gerente de Projetos. A maioria considera seu entendimento sobre gerenciamento de projetos, e em particular gerenciamento de projetos de software, num nível bom e já utilizou e definiu pelo menos um processo de gerenciamento de projetos de software. Possuem um conhecimento bom em abordagens ágeis de desenvolvimento e gerenciamento de projetos e a maioria já participou de projetos de desenvolvimento ágil de software.

Tabela 4.4 Dados do perfil dos participantes do experimento

Número do participante	Questionário													
	1 (1-2)	2 (1-4)	3 (1-2)	4 (1-6)	5 (1-5)	6 (1-3)	7 (1-6)	8 (1-6)	9 (1-3)	10 (1-3)	11 (1-6)	12 (1-4)	13 (1-2)	14 (1-2)
1	1	2	2	2	1	1	1	2	3	3	2	3	3	1
2	1	2	2	2	1	2	4	3	2	2	3	3	4	1
3	1	2	2	3	1	3	3	3	2	2	3	3	4	1
4	1	2	2	2	1	3	3	4	1	1	6	4	6	2
5	1	2	2	2	1	2	3	3	3	2	3	3	3	1
6	1	2	2	4	1	2	5	5	1	1	5	5	5	2
7	1	2	2	3	2	2	4	5	1	1	5	6	6	2
8	1	2	2	2	1	2	2	2	3	1	3	3	3	1

Tabela 4.5 Legenda do perfil do participante

1 – Instituição		2 – Curso		3 – Escolaridade		4 – Entendimento em desenvolvimento de software		5 – Atividades Exercidas	
1	Pública	1	Engenharia	1	Técnico	1	Excelente	1	Gerente de Projetos
2	Particular	2	C.da Computação	2	Universitário	2	Alto	2	Analista de Sistemas
		3	Matemática			3	Bom	3	Projetista de Sistemas
		4	Outros			4	Médio	4	Gerente de Processo
						5	Baixo	5	Projetista de Processo
						6	Nenhum		
6 – Porte dos projetos que participou		7 – Entendimento sobre processos de gerenciamento		8 – Entendimento sobre gerenciamento de projetos		9 – Quantidade de processos que já usou		10 – Quantidade de processos que já definiu	
1	Grande	1	Excelente	1	Excelente	1	Nenhum	1	Nenhum
2	Médio	2	Alto	2	Alto	2	Entre 1 e 2	2	Entre 1 e 2
3	Pequeno	3	Bom	3	Bom	3	Mais de 2	3	Mais de 2
		4	Médio	4	Médio				
		5	Baixo	5	Baixo				
		6	Nenhum	6	Nenhum				
11 – Entendimento sobre o PMBOK® Guide		12 – Entendimento sobre metodologias de desenvolvimento ágil		13 – Entendimento sobre metodologias de gerenciamento ágil		14 – Já participou de projetos de desenvolvimento ágil			
1	Excelente	1	Excelente	1	Excelente	1	Sim		
2	Alto	2	Alto	2	Alto	2	Não		
3	Bom	3	Bom	3	Bom				
4	Médio	4	Médio	4	Médio				
5	Baixo	5	Baixo	5	Baixo				
6	Nenhum	6	Nenhum	6	Nenhum				

A Tabela 4.6 permite uma visualização dos dados puros obtidos quando aplicado o Questionário de Avaliação das características de gerenciamento ágil de projetos de software, apresentadas pelo Agilius em função de PUA.

Tabela 4.6 Dados brutos obtidos a partir da opinião sobre as características do Agilium

N	CARACTERÍSTICA	PRESEÇA				UTILIDADE			ADEQUAÇÃO		
		1	2	3	4	1	2	3	1	2	3
1	Apoio à comunicação informal			12,5%	87,5%		50%	50%	12,5%	75%	12,5%
2	Incentivo à melhoria contínua do processo			25%	75%		37,5%	62,5%	25%	62,5%	12,5%
3	Informação aberta			25%	75%		25%	75%	12,5%	75%	12,5%
4	Monitoração contínua do projeto				100%		25%	75%	12,5%	75%	12,5%
5	Incentivo à auto-organização		12,5%	12,5%	75%		37,5%	62,5%	37,5%	50%	12,5%
6	Pontos de checagem de gerenciamento			50%	50%		37,5%	62,5%	37,5%	50%	12,5%
7	Apoio à cooperação e colaboração				100%		25%	75%	25%	62,5%	12,5%
8	Entregas freqüentes			25%	75%		50%	50%	25%	62,5%	12,5%
9	Adequação a ambientes voláteis				100%		75%	25%	12,5%	75%	12,5%
10	Comunicação freqüente				100%		25%	75%	12,5%	75%	12,5%
11	Reuniões diárias			25%	75%	12,5%	37,5%	50%		62,5%	37,5%
12	Processo visível a todos			37,5%	62,5%		75%	25%	25%	62,5%	12,5%
13	Tomada de decisão coletiva			25%	75%		62,5%	37,5%	12,5%	75%	12,5%
14	Processo centrado nas pessoas			25%	75%		50%	50%	37,5%	50%	12,5%
15	Participação ativa do cliente			37,5%	62,5%		62,5%	37,5%	12,5%	75%	12,5%
16	Feedback freqüente entre cliente e equipe			50%	50%		75%	25%	25%	62,5%	12,5%
17	Planejamento iterativo				100%		37,5%	62,5%		87,5%	12,5%
18	Estimativa ágil de custos		75%		25%		75%	25%	37,5%	50%	12,5%
19	Estimativa ágil de duração		12,5%	12,5%	75%		62,5%	37,5%	37,5%	62,5%	
20	Baixo índice de retrabalho	12,5%		50%	37,5%	12,5%	75%	12,5%	25%	62,5%	12,5%
21	Incentivo à obtenção da satisfação da equipe			62,5%	37,5%		37,5%	62,5%	50%	37,5%	12,5%
22	Incentivo à obtenção da satisfação do cliente			12,5%	87,5%		50%	50%	12,5%	75%	12,5%
23	Riscos gerenciados durante o processo de desenvolvimento			25%	75%		87,5%	37,5%	12,5%	75%	12,5%
24	Facilidade de se adaptar as mudanças durante o projeto			12,5%	87,5%		62,5%	37,5%	25%	62,5%	12,5%
25	Motivação da equipe ao utilizar o processo de gerenciamento			50%	50%		100%		50%	37,5%	12,5%
26	Facilidade de ser instanciado		25%	50%	25%		75%	25%	37,5%	50%	12,5%
27	Adequação ao tipo de projetos executados			50%	50%		50%	50%	37,5%	50%	12,5%
28	<i>Time-to-market</i>		12,5%	50%	37,5%	12,5%	75%	12,5%	50%	50%	
29	Alta produtividade da equipe		12,5%	50%	37,5%	12,5%	75%	12,5%	37,5%	62,5%	
30	Facilidade de atualização da documentação		12,5%	12,5%	75%		75%	25%	25%	75%	

O percentual equivale à proporção de participantes que responderam para uma determinada característica existente no formulário a respectiva opção disponível em PUA. Na seção seguinte estes dados serão analisados para a avaliação da abordagem proposta.

As características apresentadas na Tabela 4.6 dizem respeito às práticas, técnicas e ferramentas apresentadas em [3], [5], [6] e [9] que são utilizadas no gerenciamento ágil de projetos.

4.4 Análise e Interpretação dos Resultados

Os resultados da fase de Análise e Interpretação oferecem as conclusões sobre a possibilidade da rejeição da hipótese nula, usando a estatística descritiva, a redução do conjunto de dados e a verificação das hipóteses [11]. Nesta etapa, os aspectos mais importantes são escolher o teste estatístico apropriado, explicar os resultados considerando os aspectos da validade, realizar a análise custo-benefício e interpretar corretamente os resultados negativos.

4.4.1 Validação dos Dados

Uma das respostas de um dos participantes está errada do ponto de vista dos valores válidos para PUA, descritos na Tabela 4.3. Assim, para esta característica não será considerada a resposta dada por este participante.

Tabela 4.7 Validação dos dados da avaliação

Número do Participante	Característica	Valor PUA
8	20	{0, 0, 1}

4.4.2 Estatística Descritiva

A estatística descritiva é um ramo da estatística que aplica várias das muitas técnicas usadas para sumarizar um conjunto de dados a fim de descrever as características dos dados que pertencem a esse conjunto. Para este fim, faz-se necessário realizar o cálculo de medidas que possibilitem representar um conjunto de dados relativos à observação de um determinado fenômeno de forma resumida. As Medidas de Tendência Central são medidas que representam fenômenos pelos seus valores médios, em torno dos quais tendem a concentrar os dados. Estas medidas são: média aritmética, mediana e moda.

Dados ordinais ou valores em uma escala ordinal representam diferentes tipos de um elemento que podem ser ordenados, ainda que sem qualquer interpretação numérica. Como os valores “Presença”, “Utilidade” e “Adequação” são ordinais, para este experimento é possível definir

a medida de “moda” (valor que ocorre mais freqüentemente) e mediana (medida de localização do centro da distribuição de dados). Assim, a Tabela 4.8 retrata as medidas obtidas no Questionário de Avaliação, que apresenta algumas características de gerenciamento ágil de projetos de software, em função de PUA.

Considerando as respostas recebidas durante a avaliação do Agilius, o correspondente ao respondido para a característica 20, pelo participante 3, foi rejeitado (como visto na Seção 4.4.1).

Tabela 4.8 Medidas de tendência central para o Questionário de Avaliação

PRESENÇA															
Métrica	Característica														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Mediana	4	4	2,5	4	4	3,5	4	4	4	4	4	4	4	4	4
Moda	4	4	4	4	4	{3,4}	4	4	4	4	4	4	4	4	4
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Mediana	3,5	4	2	4	3	3	4	4	4	3,5	3	3,5	3	3	4
Moda	{3,4}	4	2	4	3	3	4	4	4	{3,4}	3	{3,4}	3	3	4
UTILIDADE															
Métrica	Característica														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Mediana	2,5	3	3	3	3	3	3	2,5	2	3	3	2	4	2,5	2
Moda	{2,3}	3	3	3	5	3	3	{2,3}	2	3	3	2	2	{2,3}	2
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Mediana	2	2	2	2	2	3	2,5	2	2	2	2	2,5	2	2	2
Moda	2	3	2	2	2	3	{2,3}	2	2	2	2	{2,3}	2	2	2
ADEQUAÇÃO															
Métrica	Característica														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Mediana	2	2	2	2	2	2	2	2	2	2	2	2	3	2	2
Moda	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Mediana	2	2	2	2	2	1,5	2	2	2	1,5	2	2	1,5	2	2
Moda	2	2	2	2	2	1	2	2	2	1	2	2	{1,2}	2	2

Em uma distribuição simétrica, observa-se que a média, a mediana e a moda são iguais [47]. Como os valores obtidos na coleta dos dados apresentam-se em uma escala ordinal, não é possível realizar o cálculo da média. Dessa forma, a mediana e a moda apresentadas na Tabela 4.8 são medidas que revelam que a distribuição dos dados pode ser considerada simétrica em torno do valor médio das respostas obtidas. Isto significa que o resultado mediano obtido pode ser considerado como resposta para cada característica, por expressar em torno de que valor tende a se concentrar cada conjunto de dados.

Considerando os dados coletados durante o estudo e utilizando os resultados da estatística descritiva, foi possível dividir as questões aplicadas aos respondentes em três grupos: (i) Características mal-entendidas, (ii) Características compreendidas e não utilizadas e (iii) Características compreendidas. A relação das características em cada um destes grupos encontra-se

detalhada nas Tabelas 4.9, 4.10 e 4.11, onde os valores de proporção definidos à PUA significam: P – Presente : Não Presente; U – Útil : Inútil; A – Adequado : Inadequado, e as respostas estão distribuídas na forma (resposta positiva):(resposta negativa). Foram consideradas respostas positivas para Presença os valores 3 e 4; respostas positivas para Utilidade os valores 2 e 3; e respostas positivas para Adequação: 2 e 3. Os demais valores de resposta para PUA foram considerados respostas negativas.

Tabela 4.9 Característica mal-entendida

N	CARACTERÍSTICA	P	U	A	OBSERVAÇÕES
18	Estimativa ágil de custos	2:6	8:0	4:4	<p>- A maioria dos participantes não percebeu a existência da característica durante a apresentação e o estudo do Agilius, mas gostaria de que fosse transmitida;</p> <p>- A característica é considerada útil, mas quase todos os participantes ainda não utilizaram para a definição de processos de gerenciamento ágil de software;</p> <p>- O detalhamento deve ser modificado, porque mesmo que os participantes tenham percebido alguma informação sobre esta característica, o seu detalhamento não é adequado para utilização.</p>

Tabela 4.10 Características aprendidas e não utilizadas

N	CARACTERÍSTICA	P	U	A	OBSERVAÇÕES
21	Incentivo à obtenção da satisfação da equipe	8:0	8:0	3:5	<p>- As características são percebidas de forma parcial durante a apresentação e o estudo do Agilius;</p> <p>- Os serviços são considerados úteis, mas a maioria dos participantes ainda não os utilizou para a definição de processos de gerenciamento;</p> <p>- O detalhamento deve ser modificado ou revisto, provavelmente para entender onde as características podem ser utilizadas.</p>
25	Motivação da equipe ao utilizar o processo de gerenciamento	8:0	8:0	3:5	

Tabela 4.11 Características aprendidas

N	CARACTERÍSTICA	P	U	A	OBSERVAÇÕES
1	Apoio à comunicação informal	8:0	8:0	6:2	<p>- Todas as características foram percebidas durante a apresentação e o estudo do Agilius, com exceção das características 26 e 27 que foram observadas por alguns participantes;</p> <p>- As características são consideradas úteis e os participantes já utilizaram algumas delas para a definição de processos de gerenciamento, com exceção das características 9, 12, 13, 15, 16, 18, 20, 23, 24, 26, 28, 29, 30, que são consideradas úteis, mas não foram aplicadas por estes participantes;</p> <p>- O detalhamento não precisa de modificação à exceção das características 5, 6, 14, 18, 26, 27, 28, cujo detalhamento deve ser aumentado.</p>
2	Incentivo à melhoria contínua do processo	8:0	8:0	5:3	
3	Informação aberta	8:0	8:0	6:2	
4	Monitoração contínua do projeto	8:0	8:0	6:2	
5	Incentivo à auto-organização	7:1	8:0	4:4	
6	Pontos de checagem de gerenciamento	8:0	8:0	4:4	
7	Apoio à cooperação e colaboração	8:0	8:0	5:3	
8	Entregas freqüentes	8:0	8:0	5:3	
9	Adequação a ambientes voláteis	8:0	8:0	6:2	
10	Comunicação freqüente	8:0	8:0	6:2	
11	Reuniões diárias	8:0	7:1	5:3	
12	Processo visível a todos	8:0	8:0	5:3	
13	Tomada de decisão coletiva	8:0	8:0	6:2	
14	Processo centrado nas pessoas	8:0	8:0	4:4	
15	Participação ativa do cliente	8:0	8:0	6:2	
16	<i>Feedback</i> freqüente entre cliente e equipe	8:0	8:0	5:3	
17	Planejamento iterativo	8:0	8:0	7:1	
19	Estimativa ágil de duração	7:1	8:0	5:3	
20	Baixo índice de retrabalho	7:0	7:1	5:3	
22	Incentivo à obtenção da satisfação do cliente	8:0	8:0	6:2	
23	Riscos gerenciados durante o processo de desenvolvimento	8:0	8:0	6:2	
24	Facilidade de se adaptar as mudanças durante o projeto	8:0	8:0	5:3	
26	Facilidade de ser instanciado	6:2	8:0	4:4	
27	Adequação ao tipo de projetos executados	8:0	8:0	4:4	
28	<i>Time-to-market</i>	7:1	6:2	4:4	
29	Alta produtividade da equipe	7:1	7:1	5:3	
30	Facilidade de atualização da documentação	7:1	8:0	6:2	

4.4.3 Aplicação do Teste Estatístico Chi-2

Uma aplicação importante do teste Chi-2 ocorre quando se deseja estudar a relação entre duas ou mais variáveis de classificação e neste caso a representação das freqüências observadas pode ser feita por meio de uma tabela de contingência [47]. Quanto ao número de graus de liberdade ϕ para realização de tal teste, pode ser utilizado $\phi = K - 1$, onde K é o número de eventos ou categorias em que foi dividida a amostra.

A estatística Chi-2 quando utilizada para comprovar a concordância entre valores observados e esperados para certo fenômeno é denominada teste de adequação do ajustamento [47]. Assim, através deste teste, pode-se comparar a distribuição de respostas dos participantes com a distribuição esperada, isto é, quando todos os participantes responderam igualmente que uma característica fornecida pelo Agilius é recebida, é útil e não precisa de modificação.

A Tabela 4.12 é conhecida como tabela de contingência e apresenta o relacionamento entre as duas variáveis de resposta para o experimento. Os valores de M e N são, respectivamente, a frequência de respostas positivas e negativas observadas para uma característica do Agilius.

Tabela 4.12 Tabela de contingência para o experimento

Resposta	Frequência		Total
	Observada	Esperada	
Positiva	M	8	8 + M
Negativa	N	0	N
Total	8	8	16

Para a efetivação do teste Chi-2 foram definidas as hipóteses, de acordo com o apresentado em [47]:

H_0 : não há discrepância entre as frequências esperadas e as frequências observadas

H_a : há discrepância entre as frequências esperadas e as frequências observadas

A hipótese H_a , se comprovada, afirmará que o modelo testado é inadequado para representar a distribuição da população. O nível de significância utilizado no teste foi $\alpha = 0,05$.

Para realizar o teste, foi preciso encontrar os valores de Chi-2 para todas possíveis distribuições de respostas a fim de obter conclusões sobre presença, utilidade e adequação de cada característica do Agilius. Com os dados coletados apresentados na Tabela 4.9, na

Tabela 4.10 e na Tabela 4.11, e através da Tabela 4.12 foi possível realizar o teste Chi-2 utilizando a fórmula a seguir [46]:

$$X^2 = \sum \left(\frac{(F_{observada} - F_{esperada})^2}{F_{esperada}} \right),$$

onde $F_{observada}$ é obtida através dos dados coletados e $F_{esperada}$ é obtida através da fórmula [46]:

$$F_{esperada} = \frac{(TotalLinha) \times (TotalColuna)}{TotalGeral}$$

Onde:

$TotalLinha$ = soma total do valor da linha que possui a célula a ser calculada

$TotalColuna$ = soma total do valor da coluna que possui a célula a ser calculada

$TotalGeral$ = soma de todas as linhas e colunas

Os valores de Chi-2 apresentados abaixo na Tabela 4.13 foram calculados a partir das distribuições de respostas presentes na Tabela 4.9, na

Tabela 4.10 e na Tabela 4.11.

Tabela 4.13 Valor de Chi-2 para as distribuições de repostas do experimento

Distribuição	...	(+3):(-5)	(+4):(-4)	(+5):(-3)	(+6):(-2)	(+7):(-1)	...
Valor Chi-2	...	3,64	2,67	1,84	1,14	0,53	...

Para o experimento realizado, o valor $\varphi = 1$ (um) foi adotado, pois a amostra foi dividida em 2 (duas) categorias: resposta positiva e resposta negativa. De posse do número de graus de liberdade φ e do nível de significância α , foi possível consultar a tabela de distribuição de Chi-2, e verificar que o valor de Chi-2 para o experimento é 3,841. Assim, como $\text{Chi-2}_{\text{calculado}} \leq \text{Chi-2}_{\text{tabela}}$ pode-se concluir que as freqüências observadas não diferem das esperadas e não se pode rejeitar H_0 ao nível de significância $\alpha = 0,05$ ou 5%.

Com base no resultado do teste Chi-2, as características com distribuição das respostas (+3):(-5), (+4):(-4), (+5):(-3), (+6):(-2), (+7):(-1) e (+8):(-0) receberam os valores “Presença”, “Utilidade” = {1} e o valor “Adequação” = {0}, isto é, pode se considerar que as característica são fornecidas, úteis e seus detalhamentos não precisam de modificação. Isto é possível porque espera-se com um nível de confiança de 95% que as freqüências observadas sejam bem próximas das freqüências esperadas. Para as características com outras distribuições das respostas os valores “Presença”, “Utilidade” = {0} e o valor “Adequação” = {1}. Abaixo, a Tabela 4.14 exhibe os valores de PUA em função da distribuição das respostas para as características do Agilius.

Tabela 4.14 Valores PUA para as características

	Característica														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Presença	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Utilidade	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Adequação	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Característica														
	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Presença	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Utilidade	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Adequação	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Os valores de PUA, de acordo com o teste Chi-2 executado, indicam que não há discrepância entre as freqüências esperadas e as freqüências observadas, ou seja, os valores coletados estão muito próximos dos valores de resposta positivos ou ideais para o Agilius. Estes valores indicam uma boa aceitação do modelo por parte dos entrevistados.

4.4.4 Análise Quantitativa

Para verificar a similaridade entre as características apresentadas no Agilius, sugeridas para a definição de processo de gerenciamento ágil de projetos de software, e as características utilizadas para o desenvolvimento deste tipo de processo é necessário calcular o valor da variável dependente, conforme apresentado na Seção 4.2.5.

Primeiro, deve-se identificar na Tabela 4.14 a quantidade de características de definição de processo de gerenciamento ágil de software oferecidas pelo Agilius considerada igual à quantidade das características utilizadas para a implementação deste tipo de processo (a quantidade de características com o valor PUA $\{1, X, X\} = 29$) e a considerada diferente (a quantidade de características com o valor PUA $\{0, X, X\} = 1$). Assim, a similaridade segundo a fórmula da variável dependente 1, calculada de acordo com a fórmula (1) apresentada na Seção 4.2.5, é:

$$\text{Grau de Similaridade} = 29 / (1 + 29) * 100\% = 96,67\%$$

Para verificar a utilidade das características similares, ou seja, características apresentadas nas abordagens ágeis de gerenciamento de projetos de software e que são oferecidas pelos Agilius, é necessário calcular o valor de variável dependente 2, obtida respectivamente pelas fórmulas (2) e (3) apresentadas na Seção 4.2.5, com resultado abaixo:

$$\text{Parte Útil das Características Similares} = 29 / 29 * 100\% = 100\%$$

$$\text{Parte Inútil das Características Similares} = 0 / 29 * 100\% = 0$$

Para verificar na Tabela 4.14 a adequação das características similares, isto é, se o nível de detalhamento dessas características nos processos precisa ser modificado, é necessário calcular o valor da variável dependente 3, obtida respectivamente pelas fórmulas (4) e (5) apresentadas na Seção 4.2.5, com resultado a seguir:

$$\text{Parte Adequada das Características Similares} = 27 / 29 * 100\% = 93,1\%$$

$$\text{Parte Inadequada das Características Similares} = 2 / 29 * 100\% = 6,9\%$$

4.4.5 Análise Qualitativa

No intuito de verificar se existem características utilizadas para a definição de processo gerenciamento ágil de software que não são oferecidas pelo Agilius, mas que os participantes do experimento gostariam de receber porque consideram úteis para o desenvolvimento deste tipo de

processos, foi feita a análise qualitativa. A Tabela 4.15 mostra as características presentes no questionário de avaliação, consideradas pelos participantes como não oferecidas pelo Agilius, a partir da análise da Tabela 4.6.

Tabela 4.15 Características não apresentadas pelo Agilius segundo os participantes do experimento

Característica	5	18	19	26	28	29	30
Quantidade de participantes que não receberam a característica, mas gostariam de receber.	1	6	1	2	1	1	1

Assim, pode-se concluir que a característica 18 (Estimativa ágil de custos) provoca interesse nos participantes que não receberam essa característica durante a apresentação e o estudo do Agilius. Contudo as características de gerenciamento ágil de projetos do Agilius de maneira geral foram bem assimiladas. A informação sobre Estimativa ágil de custos provavelmente não está sendo oferecida ou está sendo oferecida com pouco detalhamento pelo Agilius. As outras características não oferecidas pelo Agilius provavelmente são conhecidas aos participantes, mas a falta de exemplos ou experiências de aplicação dessas características faz com que alguns dos participantes não as percebam na abordagem.

4.4.6 Verificação das Hipóteses

Para verificar a Hipótese Nula (H_0) faz-se necessário responder a questão Q_1 utilizando a métrica M_1 . O resultado do teste Chi-2 mostra que uma das trinta características apresentadas para a definição de processos de gerenciamento ágil de projetos (característica 1) não pode ser considerada como oferecida pelo Agilius. Assim, podemos concluir que “a lista de características de processos de gerenciamento ágil de projetos de software oferecida aos gerentes de projeto que utilizam o Agilius é diferente da lista de características de processos utilizadas para o gerenciamento deste tipo de projeto”, fazendo com que Hipótese Nula não possa ser rejeitada.

Não se pode dizer que “na lista de características de processos de gerenciamento ágil de projetos de software oferecida aos gerentes de projeto que utilizam o Agilius e que fazem parte da lista de características de processos importantes para o gerenciamento deste tipo de projeto, existem características que os gerentes consideram inúteis para esta tarefa” (Hipótese Alternativa H_2) porque, respondendo a questão Q_2 (métrica M_2), o resultado do teste Chi-2 mostra que nenhuma das 30 (trinta) características que compõem o questionário de Definição do Processo pode ser considerada como inútil.

Finalmente, pode-se concluir que “na lista de características de processos de gerenciamento ágil de projetos de software oferecida aos gerentes de projeto que utilizam o Agilius, que fazem parte da lista de características de processos importantes para o gerenciamento deste tipo de projeto e que são consideradas úteis para esta tarefa, existem de características de processos cujo detalhamento deve ser modificado para atingir o nível esperado pelos gerentes de projeto que realizam gerenciamento ágil de projetos” (Hipótese Alternativa H₃) porque, respondendo a questão Q3 (métrica M₃), o resultado do teste Chi-2 mostra que duas características precisam de modificação.

Uma análise qualitativa foi realizada para auxiliar na obtenção de conclusões relevantes sobre a Hipótese Alternativa H₄. Os resultados da análise mostraram que 7 (sete) características foram consideradas como não oferecidas pelo Agilius e estas provocaram interesse dos participantes. Assim, pode-se concluir que “na lista de características importantes para a definição de processos de gerenciamento ágil de projetos de software existem características que não fazem parte da abordagem Agilius, mas que os gerentes gostariam que estivessem disponíveis porque consideram necessárias para a definição de processos de gerenciamento ágil de projetos de software” (Hipótese Alternativa H₄).

4.5 Considerações Finais

Este capítulo apresentou a avaliação do Agilius realizada de acordo com o procedimento usado no estudo da experimentação sugerido por [11]. Foram definidos os objetivos, as questões usadas para a medição e um conjunto de métricas para caracterizar a análise destas questões. Em seguida, o planejamento do estudo experimental foi especificado através da definição de hipóteses, da instrumentação, do contexto de realização do experimento, das variáveis utilizadas na análise dos resultados, dos critérios para observação da análise qualitativa e quantitativa. Ao final, para a verificação das hipóteses levantadas foi realizada a análise e a interpretação dos dados coletados a partir de estatística descritiva, análise qualitativa e quantitativa.

De acordo com a opinião dos participantes do experimento, o Agilius pode ser considerado ágil por contemplar de forma satisfatória as principais características que definem o gerenciamento ágil de projetos de software. Segundo os participantes do estudo, o Agilius traz um agregado bem mais denso das abordagens ágeis do que das clássicas, no entanto, sem desconsiderar a presença e importância das formalidades de um guia como o PMBOK® *Guide*, o que pode ser considerado uma combinação equilibrada dessas duas abordagens. Ainda pôde-se perceber a presença de diversas atividades de comunicação que são otimizadas para permitir uma maior produtividade da equipe de

desenvolvimento, além de uma maior integração entre todos os elementos envolvidos no processo. A partir dos comentários do Questionário de Avaliação algumas considerações foram feitas:

- O alto nível de cooperação necessário por parte do cliente, mesmo sendo um ponto extremamente interessante, pode trazer problemas no desenvolvimento uma vez que nem sempre existe o mesmo comprometimento por parte deste como o vindo da equipe. Essa situação pode ser encontrada principalmente no desenvolvimento de sistemas de grande porte.
- A abordagem, apesar de resumir e simplificar as áreas contidas no PMBOK® *Guide*, ainda pode causar um “choque burocrático” se não for corretamente compreendida. Coordenar as atividades das diversas áreas presentes na abordagem pode dificultar o dia-a-dia do gerente, especialmente se for o caso do gerente ter dedicação parcial a função, isto é, caso o gerente possua atribuições técnicas no projeto, o que é bastante comum nos projetos aos quais a abordagem se destina.
- Uma versão futura do Agilius poderia trazer um maior detalhamento dos artefatos de cada etapa, com exemplos dos modelos.

Quanto à melhoria do processo, os níveis de Corretude, Consistência, Utilidade e Relevância foram considerados altos. O critério Completude foi considerado médio pela maioria pela falta de modelos, artefatos e de um guia de instanciação do Agilius, e o critério Originalidade foi considerado médio.

Conclusões

O objetivo deste estudo foi prover uma abordagem que desse suporte ao desenvolvimento de processos de gerenciamento ágil de projetos de software, a partir de um conjunto de atividades baseadas no gerenciamento ágil de projetos para o gerenciamento de projetos de software, de acordo com o guia PMBOK® e com a literatura pesquisada sobre abordagens ágeis de gerenciamento.

Com relação ao termo agilidade, para este trabalho ele representa um sinônimo de capacidade de adaptar o escopo do projeto às mudanças de requisitos e às necessidades dos clientes. Ou seja, realizar o mínimo possível de trabalho que possua valor para o cliente, utilizando a menor quantidade de tempo possível.

O Agilius foi desenvolvido seguindo esta linha de pensamento, na tentativa de aderir aos princípios estabelecidos no manifesto ágil e assim apresentar um conjunto de processos que possuem a flexibilidade de se adaptarem a mudanças. Assim, como principais contribuições, podem ser enumeradas as seguintes:

- Uma análise das abordagens ágeis de gerenciamento de projetos e as conseqüências do seu uso no desenvolvimento de projetos de software.
- O Agilius, abordagem ágil de gerenciamento de projetos de software baseada nas abordagens ágeis de gerenciamento de projetos Scrum, *Agile Project Management*, *Agile Project Management Framework*, *Extreme Project Management* e no PMBOK® *Guide*.
- Proporcionar uma alternativa aos gerentes e a comunidade de software no sentido de se beneficiar das abordagens ágeis de gerenciamento e do gerenciamento clássico de projetos.

Uma das maiores dificuldades encontradas no decorrer deste trabalho foi o fato de não se trabalhar o Agilius através da sua instanciação por equipes de desenvolvimento de software e acompanhar o seu uso. Assim as conclusões deste trabalho ficam restritas a avaliação realizada, não podendo ser num primeiro momento generalizadas para outros casos. A bibliografia ainda pouco acessível no que diz respeito ao gerenciamento ágil de projetos também pode ser apontada como uma limitação.

Com relação ao tema central deste trabalho, as abordagens ágeis de gerenciamento, cada vez mais a comunidade de software vem aceitando, entendendo e aplicando as mesmas. Elas se mostram uma alternativa viável especialmente para grupos e organizações pequenas e médias, por serem muito fáceis de implantar e de exercitar suas práticas e princípios. Contudo ainda existe uma falta de tratamento de aspectos relevantes do gerenciamento de software por grande parte destas abordagens. Assim, este trabalho representa uma combinação das abordagens ágeis com alguns aspectos do gerenciamento clássico e se mostra uma solução factível para tal problema.

5.1 Trabalhos Relacionados

Os estudos encontrados sobre gerenciamento ágil de projetos de software associados ao PMBOK® *Guide* são geralmente propostas de modelos ou apenas comparativos das duas abordagens. O que se pode observar é que não apresentam uma modelagem bem definida e aprofundada como a que o Agilius oferece.

O trabalho [4] faz uma comparação entre os processos e áreas de conhecimento do PMBOK® *Guide* e práticas ágeis de desenvolvimento. Não apresenta um processo ou um conjunto de processos ágeis a serem aplicados pelas organizações, apenas retrata um conjunto de peculiaridades do gerenciamento ágil e do PMBOK®, e em seguida uma análise a respeito da aplicação do gerenciamento ágil. De acordo com [4] a maior diferença entre o PMBOK® *Guide* e as abordagens ágeis se encontra nas fases de planejamento e controle, conforme mostram a Figura 5.1 e a Figura 5.2:

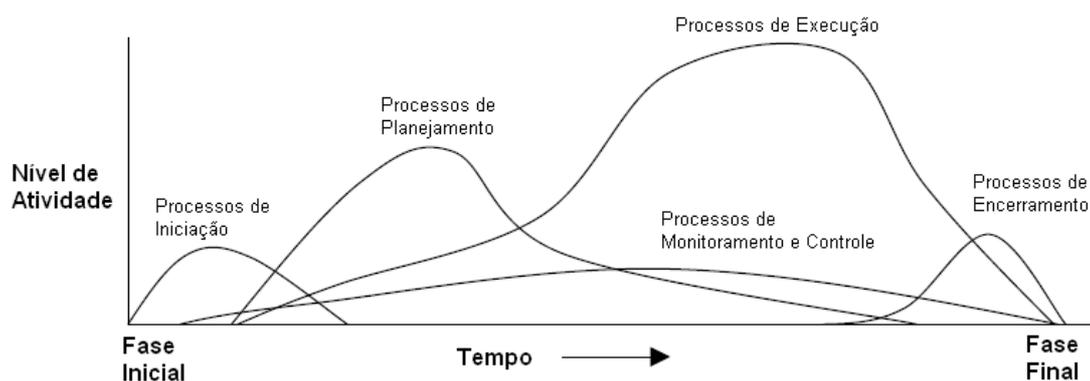


Figura 5.1 Grupos de processos do PMBOK® *Guide* em uma fase [4]

Em seu trabalho, [29] sugere um mapeamento entre práticas ágeis e os grupos de processos do PMBOK® *Guide*, onde ele recomenda o uso de tais processos e quando necessário, sugere modificações para que seu uso se adeque ao gerenciamento ágil de projetos.

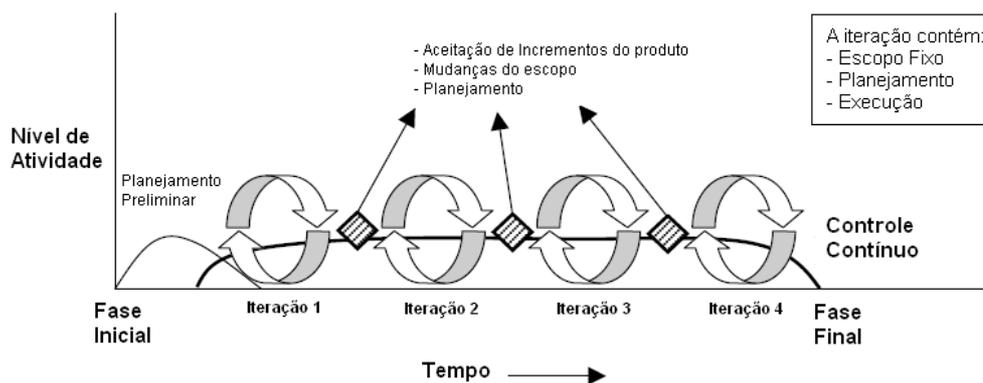


Figura 5.2 Fluxo de processo ágil em uma fase [4]

Uma apresentação de características do Gerenciamento Ágil do Escopo, rápida e com poucos detalhes é apresentada em [41]. Ela afirma que o controle dos limites do escopo e o gerenciamento de expectativas do projeto é mais adequado que esperar por requisitos estáveis. Também reforça em suas conclusões que através do comprometimento com expectativas de resultados de negócio compatíveis times podem gerenciar o escopo de projetos ágeis com sucesso, para a satisfação dos patrocinadores, executivos e clientes.

O que mais se aproximou de um trabalho de definição de processos de gerenciamento ágil de projetos foi o apresentado por Sliger em *A Project Managers's Survival Guide to Going Agile* e *Relating PMBOK Practices to Agile Practices*. Nestes dois artigos, a autora adota como grupos de processos os apresentados por [3], conforme mostrado na Figura 2.4 da Seção 2.4.4, e apresenta as áreas de gerenciamento Integração, Escopo, Tempo, Qualidade, Recursos Humanos e Riscos redefinidas com uma proposta de atividades ágeis baseadas no manifesto ágil, Scrum e *Agile Project Management Framework*.

Os trabalhos apresentados nesta seção artigos estão bem relacionados com tudo que aborda o Agilius, em particular os propostos por [39] e [44], e serviram de fundamentação no desenvolvimento da abordagem. Vale reiterar que nenhum dos trabalhos encontrados durante esta pesquisa possuíam o nível de detalhes contido no Agilius.

5.2 Trabalhos Futuros

Este trabalho traz uma proposta de agilidade para o gerenciamento de projetos de software, que procura ser criteriosa com base no PMBOK[®] *Guide*, visando ser um guia para elaboração de processos de gerenciamento. A partir dele alguns trabalhos futuros podem ser derivados. Dentre eles, podemos citar:

- Incorporar ao Agilius as áreas de conhecimento do PMBOK® *Guide* que não foram incluídas neste trabalho: Gerenciamento dos Recursos Humanos, Gerenciamento das Aquisições e Gerenciamento da Qualidade.
- A abordagem Agilius, ainda que sendo uma versão ágil do PMBOK® *Guide*, possui diversas áreas de atuação, as quais levantam uma possível necessidade de gerentes específicos (por exemplo, um gerente para análise de risco), ou um gerente dedicado com experiência. Para projetos grandes, um gerente dedicado é comum. Porém, a característica ágil é mais presente em projetos de médio e, principalmente, pequeno portes. Nesses últimos, em especial, gerentes dedicados são caros e podem comprometer a coesão da equipe (gerente versus desenvolvedores). Um trabalho futuro seria fundir áreas afins, como tempo e custo, e seus respectivos processos e atividades (os que pudessem ser fundidos).
- Uma forma de enriquecer a descrição da abordagem seria prover um “diário do gerente Agilius”, contendo o dia-a-dia de um gerente de um projeto, com as atividades gerenciais que ele desenvolveria ao longo das diversas etapas do projeto. Com isso ficaria claro o funcionamento da abordagem proposta num projeto, permitindo também identificar possíveis pontos de melhoria da proposta. Outra possibilidade seria transformar esses diários em “*task lists*” para as diversas etapas do projeto. Essas *task lists* seriam um resumo da abordagem voltada para a execução do gerenciamento, e guiariam o gerente ao longo do projeto.
- Prover um guia de implantação com suporte a implantação progressiva. Isto é, como instanciar a abordagem partindo de um conjunto mínimo de áreas/processos/atividades e incrementar esse conjunto à medida que o projeto progride, adicionando ou substituindo áreas e processos de acordo com o nível de agilidade exigida pelo projeto. Por exemplo, inicialmente, as áreas de escopo, riscos, tempo e custo seriam uma área de “*bootstrapping*”, que seria substituída pelas áreas adequadas ao longo do processo.
- A possibilidade de adaptar o processo de gerenciamento durante a execução cria uma nova área de conhecimento que seria o “Meta-gerenciamento”, isto é, verificar que áreas, processos e atividades são necessárias ao projeto, quais estão contribuindo (e quanto) e quais estão atrapalhando. Isso pode ser contra-producente em projetos ágeis se não for descrito adequadamente. Uma forma de lidar com isso seria atribuir pesos quanto ao nível de importância e o quanto um processo/atividade contribui para a agilidade do projeto, dependendo de características do projeto (restrição de orçamento, restrição de tempo,

restrição de recursos humanos, etc.). Por exemplo, a atividade “Monitorar custos” do processo “Controlar e Monitorar Custos” tem importância baixa para um projeto com restrição de orçamento moderada, mas impacta quanto à agilidade se o projeto possuir restrições de tempo muito rígidas. Com tais informações em mãos, o gerente, ao iniciar o projeto, escolheria as atividades gerenciais mais adequadas para instanciar o processo.

- Verificar a aplicabilidade do Agilius ao gerenciamento de outros tipos de projetos, além dos projetos de software.

Referências Bibliográficas

- [1] PMBOK Guide: Um guia do conjunto de conhecimentos do gerenciamento de projetos. Pennsylvania: Project Management Institute, 3. ed., 2004.
- [2] MAGALHÃES, A. L. C. C.; ROUILLER, A. C.; VASCONCELOS, A. M. L. O gerenciamento de Projetos de Software Desenvolvidos À Luz das Metodologias Ágeis: Uma Visão Comparativa. Revista ProQualiti. Volume 1. 2005.
- [3] HIGHSMITH, J. Agile project management: creating innovative products. Boston: Addison-Wesley, 2004. 312 p.
- [4] UDO, N.; KOPPENSTEINER, S. Will agile development change the way we manage software projects? Agile from a PMBOK Guide perspective. Projectway, 2003.
- [5] CHIN, G. Agile project management: how to succeed in the face of changing project requirements. New York: Amacon, 2004. 229 p.
- [6] BECK, K. Extreme Programming Explained: Embrace Change. Addison Wesley Professional, 1999.
- [7] AUGUSTINE, S. Managing agile projects. Prentice Hall, 2005. 264p.
- [8] DECARLO, D. eXtreme Project Management: Using Leadership, Principles, and Tools to Deliver Value in the Face of Volatility. California: Jossey-Bass, 2004. 560p.
- [9] SCHWABER, K. Agile Project Management with Scrum. Microsoft Press, 2004.
- [10] AGILE ALLIANCE. Manifesto for agile software development. Disponível em <<http://www.agilemanifesto.org/>>. Acesso em 29 nov. 2006.
- [11] KARLSTROM, D.; RUNESON, P. Combining Agile Methods with Stage-Gate Project Management. IEEE Software 22, No.03, 43-49, 2005.
- [12] TRAVASSOS, G. H. Introdução à Engenharia de Software Experimental, Relatório Técnico RT-ES-590/02 do Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Rio de Janeiro, 2002.
- [13] CONCHANGO. Experts in Agile and Scrum. Disponível em: <<http://www.scrum-master.com/>> Acesso em 23 abr. 2007
- [14] KERZNER, H. Project Management: A Systems Approach to Planning, Scheduling, and Controlling. New Jersey: John Wiley & Sons, 2003. 912p.

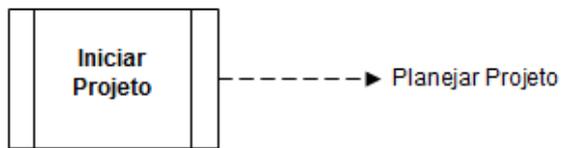
- [15] PONTO GP. Gerenciamento de Projetos – Ponto GP. Disponível em: <<http://pontogp.wordpress.com/gerenciamento-de-projetos/>>. 2007. Acesso 22 jan. 2008.
- [16] PMI. Project Management Institute. Disponível em: <<http://www.pmi.org/>>. Acesso 22 jan. 2008.
- [17] TEN STEP. You can manage. Disponível em <<http://www.tenstep.com.br/>>. Acesso em 15 nov. 2007
- [18] AMBLER, S. W. Gerenciamento ágil de projetos: Colocando o desenvolvimento de software em ordem. Mundo PM, Rio de Janeiro, v. 11, out/nov. 2006, p. 10-16.
- [19] HUGHES, B.; COTTERELL, M. Software Project Management. 2 ed. McGraw-Hill. 1999.
- [20] SOMMERVILLE, I. Engenharia de Software. 6a. edição. São Paulo: Addison Wesley, 2003. 608 pp.
- [21] SWEBOK: Guide to the Software Engineering Body of Knowledge. California: IEEE Computer Society, 2004.
- [22] ANTON, J.; VILSOET, B. Customer Relationship Management Technology. California: The Anton Press, 2001. 235 pp.
- [23] BARNETT, L. Agile Survey Results: Widespread Adoption, Emphasis on Productivity and Quality. Agile Journal, Volume 2, Number 7, 2007.
- [24] ANDERSON, D. J. Agile management for software engineering: Applying the theory of constraints for business results. New Jersey: Prentice Hall, 2003. 336p.
- [25] SCHONBERGER, R. Técnicas industriais japonesas nove lições ocultas sobre a simplicidade. 3. ed. São Paulo: Pioneira, 1988. 309pp.
- [26] KAN, S. H.; BASILI, V. R.; SHAPIRO, L. N. Software Quality: An overview from the perspective of total quality management. IBM Systems Journal, Vol 33, 1994.
- [27] PANDE, P. S.; NEUMAN, R. P.; CAVANAGH, R. R. Estratégia seis sigma: como a GE, a Motorola e outras grandes empresas estão aguçando seu desempenho. Rio de Janeiro: Qualitymark Ed., 2004. 442 p.
- [28] POPPENDIECK, M. Principles of Lean Thinking. Poppendieck LLC, 2002
- [29] GRIFFITHS, M. Using agile alongside the PMBOK. PMI Global Congress Proceedings, 2004.

- [30] LEVIN, S. A. Complex Adaptive Systems: Exploring the known, the unknown and the unknowable. *Bulletin of the American Mathematical Society*. Volume 40. Número 1. 2002.
- [31] AUGUSTINE, S.; WOODCOCK, S. Agile project management: emergent order through visionary leadership, 2003.
- [32] HIGHSMITH, J. Adaptive Software Development: A Collaborative Approach to Managing Complex Systems, Dorset House, 2000.
- [33] AUGUSTINE, S.; PAYNE, B.; SENCINDIVER, F.; WOODCOCK, S. Agile project management: Steering from the edges. *Communications of the ACM*, v. 48, dez. 2005. pp. 85-89.
- [34] KROLL, P. The Eclipse Process Framework project. 2005 Disponível em <http://www.ibm.com/developerworks/rational/library/05/1011_kroll/>. Último acesso 20 nov. 2007.
- [35] RUP. IBM - Rational Method Composer - Rational Unified Process. Disponível em: <<http://www-306.ibm.com/software/awdtools/rup/>>. Acesso em 21 jan. 2008
- [36] HAUMER, P. Eclipse Process Framework Composer. Part 1: Key Concepts. 2007. Disponível em <http://www.eclipse.org/epf/general/getting_started.php>. Último acesso 20 nov. 2007.
- [37] HAUMER, P. Increasing Development Knowledge with EPFC. *Eclipse Review*. Volume 1. Número 2. 2006.
- [38] SPEM: Software & Systems Process Engineering Metamodel Specification, v2.0 (Beta 2). OMG Adopted Especification. 2007.
- [39] SLIGER, M. A project manager's survival guide to going agile. Whitepaper. Rally Software Development Corporation, 2006.
- [40] SCRUM. Scrum for Team System. Disponível em: <<http://www.scrumforteamssystem.com/en/default.aspx/>>. Acesso em 16 mai. 2007
- [41] MEKELBURG, D. Project Expectations: The Boundaries for Agile Development. Crosstalk, 2003.
- [42] COHN, M. Agile Estimating and Planning. Robert C. Martin Series. Prentice Hall, 2006.
- [43] PRESTON, S., PICHLER, R. Agile Risks, Agile Rewards. Disponível em: <<http://www.ddj.com/architect/184415308/>>. Acesso 30 ago. 2007.

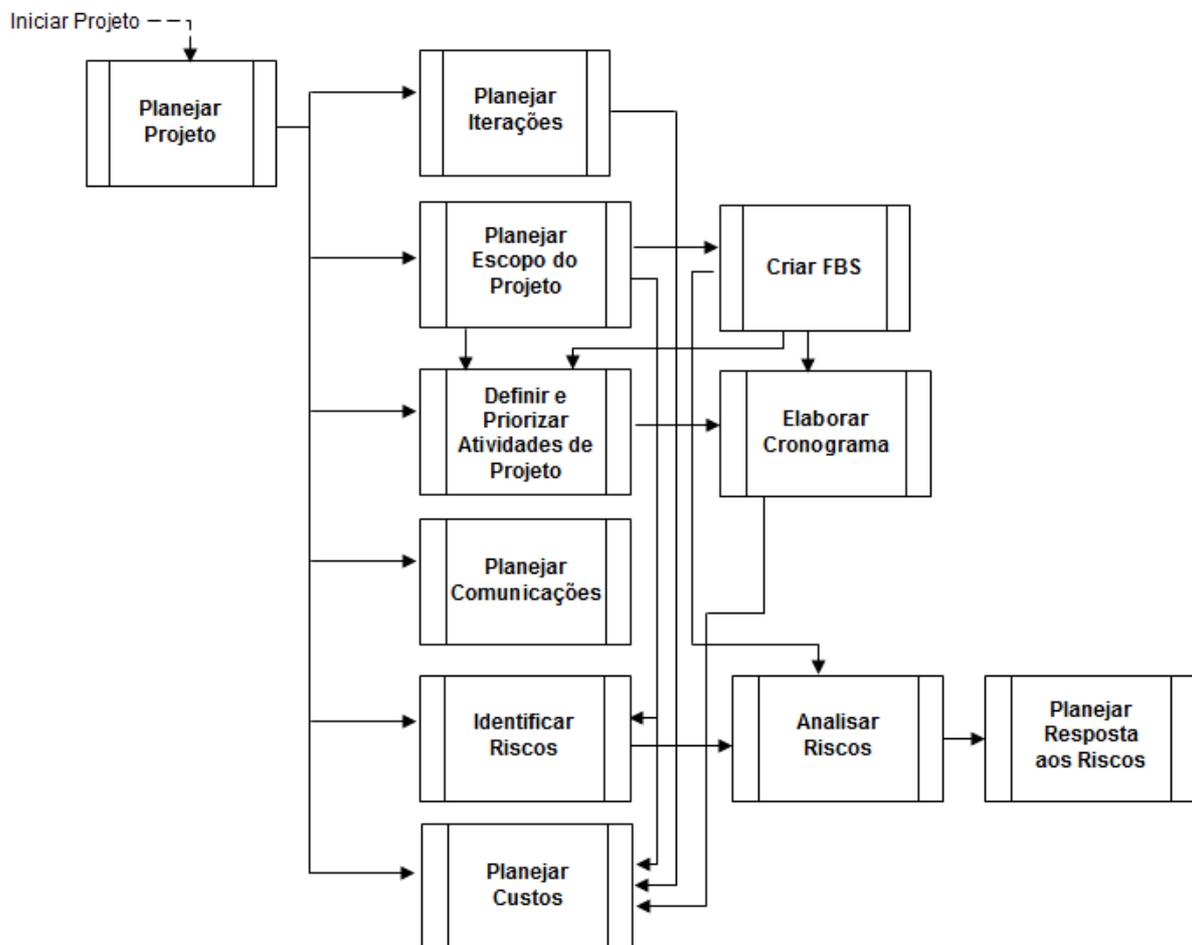
- [44] SLIGER, M. Relating PMBOK Practices to Agile Practices--Part 3 of 4. Disponível em: <<http://www.stickyminds.com/sitewide.asp?Function=edetail&ObjectType=COL&ObjectId=11133&tth=DYN&tt=siteemail&iDyn=2?>>. Acesso em 7 mar. 2007
- [45] BASILI, V. R.; CALDIERA, G.; ROMBACH, D. The Goal Question Metric Approach. Encyclopedia of Software Engineering, Volume 1, 1994, pp. 528-532.
- [46] TRIOLA, M. F. Introdução a Estatística. Editora LTC. Nona Edição. 2008.
- [47] FONSECA, J. S.; MARTINS, G. A. Curso de Estatística, 6a Edição, Editora Atlas.
- [48] OLIVEIRA, S. R. B. ProDefiner: Uma Abordagem Progressiva para a Definição de Processos de Software no Contexto de um Ambiente Centrado no Processo. Orientador: Alexandre Marcos Lins de Vasconcelos. Tese de Doutorado. UFPE, Recife, Brasil. 2007.
- [49] KROLL, P. Eclipse Process Framework - An Open Source Process Initiative. Disponível em <http://www.eclipse.org/epf/general/getting_started.php>. Último acesso 20 nov. 2007.
- [50] BOEHM, B. A View of 20th and 21st Century Software Engineering. ICSE 2006.

Apêndice A – Interações entre os Processos do Agilius

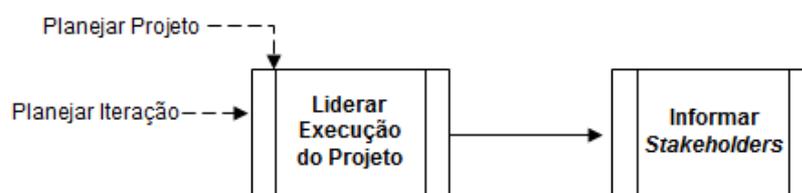
Grupo de Processos de Iniciação



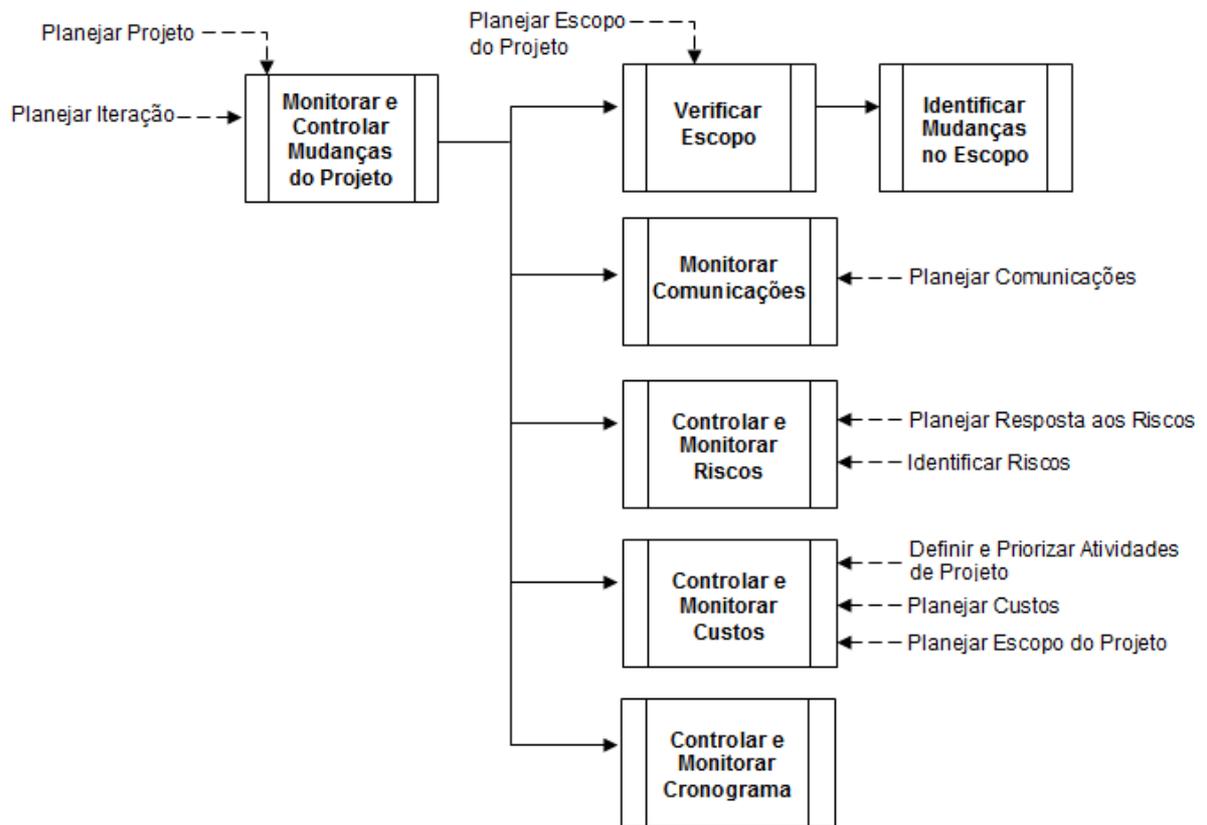
Grupo de Processos de Planejamento



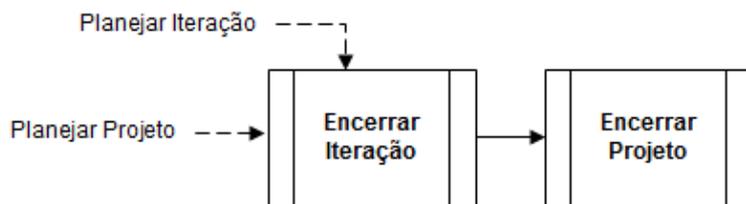
Grupo de Processos de Execução



Grupo de Processos de Monitoramento e Controle



Grupo de Processos de Encerramento



Apêndice B - Questionários

Questionário do Perfil do Participante

Marque um X sobre a opção que melhor representar a sua:

FORMAÇÃO

Instituição

1. Pública
2. Particular

Tipo de curso

1. Engenharia
2. Informática/Ciência da Computação
3. Matemática
4. Outros, especifique: _____

Grau de escolaridade

1. Ensino médio técnico
2. Universitária

EXPERIÊNCIA PROFISSIONAL

Como você classificaria o seu entendimento sobre o desenvolvimento de produto de software?

1. Excelente
2. Alto
3. Bom
4. Médio
5. Baixo
6. Nenhum

Qual(is) a(s) atividade(s) que mais exerceu ou exerce em Informática?

1. Gerente de Projetos
2. Analista de Sistemas
3. Projetista de Sistemas
4. Gerente de Processo
5. Projetista de Processo

Já participou do desenvolvimento de softwares de que porte?

1. Grande
2. Médio
3. Pequeno

Como você classificaria seu entendimento sobre processos de gerenciamento de projetos?

1. Excelente
2. Alto
3. Bom
4. Médio
5. Baixo

6. Nenhum

Como você classificaria seu entendimento sobre gerenciamento de projetos de software?

1. Excelente
2. Alto
3. Bom
4. Médio
5. Baixo
6. Nenhum

Já usou quantos processos de gerenciamento de projetos de software?

1. Nenhum
2. Entre 1 e 2
3. Mais de 2

Quantos processos de gerenciamento de projetos de software você já definiu?

1. Nenhum
2. Entre 1 e 2
3. Mais de 2

Como você classificaria seu entendimento sobre o PMBOK® Guide?

1. Excelente
2. Alto
3. Bom
4. Médio
5. Baixo
6. Nenhum

Como você classificaria seu entendimento sobre metodologias de desenvolvimento ágil de software?

1. Excelente
2. Alto
3. Bom
4. Médio
5. Baixo
6. Nenhum

Como você classificaria seu entendimento sobre metodologias de gerenciamento ágil de software?

1. Excelente
2. Alto
3. Bom
4. Médio
5. Baixo
6. Nenhum

Já participou de projetos de desenvolvimento ágil de softwares?

1. Sim
2. Não

Questionário de Avaliação do Agilius

DEFINIÇÃO DE PROCESSO

Sob o ponto de vista de **implementação de processo de gerenciamento de projetos de software e abordagens de gerenciamento ágil de software**, e considerando a experiência que você indicou no “**Questionário do Perfil do Participante**”, avalie e marque as colunas correspondentes segundo as escalas abaixo. Preencha os itens a seguir de acordo com a presença, utilidade e adequação quanto ao detalhamento das características de gerenciamento ágil de projetos de software encontradas nos processos do Agilius e listadas no questionário:

PRESEÇA	UTILIDADE	ADEQUAÇÃO DO NÍVEL DE DETALHAMENTO
1. Não é oferecida e não gostaria que estivesse disponível. 2. Não é oferecida, mas gostaria que estivesse disponível. 3. Oferecida, parcialmente. 4. Oferecida.	1. Não é útil. 2. Provavelmente é útil, mas ainda não apliquei. 3. É útil e já apliquei em diferentes projetos.	1. O detalhamento deve ser aumentado. 2. O detalhamento não precisa ser modificado. 3. O detalhamento deve ser diminuído.

N	CARACTERÍSTICA	PRESEÇA				UTILIDADE			ADEQUAÇÃO		
		1	2	3	4	1	2	3	1	2	3
1	Apoio à comunicação informal										
2	Incentivo à melhoria contínua do processo										
3	Informação aberta										
4	Monitoração contínua do projeto										
5	Incentivo à auto-organização										
6	Pontos de checagem de gerenciamento										
7	Apoio à cooperação e colaboração										
8	Entregas freqüentes										
9	Adequação a ambientes voláteis										
10	Comunicação freqüente										
11	Reuniões diárias										
12	Processo visível a todos										
13	Tomada de decisão coletiva										
14	Processo centrado nas pessoas										
15	Participação ativa do cliente										
16	Feedback freqüente entre cliente e equipe										
17	Planejamento iterativo										
18	Estimativa ágil de custos										
19	Estimativa ágil de duração										
20	Baixo índice de retrabalho										
21	Incentivo à obtenção da satisfação da equipe										
22	Incentivo à obtenção da satisfação do cliente										
23	Riscos gerenciados durante o processo de desenvolvimento										
24	Facilidade de se adaptar as mudanças durante o projeto										
25	Motivação da equipe ao utilizar o processo de gerenciamento										
26	Facilidade de ser instanciado										
27	Adequação ao tipo de projetos executados										
28	Time-to-market										

29	Alta produtividade da equipe										
30	Facilidade de atualização da documentação										

Sob o mesmo ponto de vista usado para análise das características listadas acima e considerando as demais abordagens ágeis de gerenciamento de projetos, que por um acaso não tenham sido encontrados na listagem acima, descreva na tabela a seguir todas estas características, avalie e marque as colunas correspondentes segundo as mesmas escalas anteriormente usadas:

N	CARACTERÍSTICA	PRESEÇA				UTILIDADE			ADEQUAÇÃO		
		1	2	3	4	1	2	3	1	2	3
31											
32											
33											
34											
35											
36											
37											

COMENTÁRIOS

1. Quais as suas críticas sobre a abordagem proposta?

2. De acordo com o que lhe foi apresentado a abordagem proposta pelo *Agilis* pode ser considerada ágil? Por quê?

3. Quais os problemas que você consegue verificar no processo de instanciar o *Agilis*?

4. Você possui mais algum comentário ou informação que julga importante? Explícite.

MELHORIA DE PROCESSO

A partir da abordagem Agilius, avalie o que lhe foi apresentado segundo os critérios abaixo, e faça seus comentários sobre cada tópico abordado, tentando estabelecer, se necessário, possíveis modificações na abordagem em avaliação.

CRITÉRIO DE AVALIAÇÃO	AVALIAÇÃO (ALTO, MÉDIO, BAIXO, NENHUM)	PROBLEMA	SOLUÇÃO
Corretude: a abordagem não contém erros.			
Compleitude: a abordagem contém as informações necessárias para ser compreendida de forma adequada.			
Consistência: a abordagem apresenta informações coerentes e sem ambigüidades.			
Utilidade: a abordagem é capaz de satisfazer às necessidades, possui serventia, resulta em proveito.			
Originalidade: a abordagem pode ser encarada como original, inovadora, criativa.			
Relevância: a abordagem destaca-se em escalas comparativas ou de valores.			