

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO PAULO**

**LAINÉ APARECIDA TAVARES**

**MODELOS PARA GESTÃO DE PROJETOS DE DESENVOLVIMENTO  
DISTRIBUÍDO DE *SOFTWARE* EM UMA EMPRESA DE AUTOMAÇÃO  
BANCÁRIA**

**SÃO PAULO**

**2016**

**LAINÉ APARECIDA TAVARES**

**MODELOS PARA GESTÃO DE PROJETOS DE DESENVOLVIMENTO  
DISTRIBUÍDO DE *SOFTWARE* EM UMA EMPRESA DE AUTOMAÇÃO  
BANCÁRIA**

Monografia apresentada ao Curso de Pós-Graduação *Lato Sensu* em Gestão da Tecnologia da Informação do Instituto Federal de Ciência e Tecnologia de São Paulo, Câmpus São Paulo, como requisito parcial para obtenção do título de Especialista em Gestão de Tecnologia da Informação.

Orientador: Prof. Dr. José Braz de Araújo.

**SÃO PAULO**

**2016**

## FICHA CATALOGRÁFICA

T231m Tavares, Laine Aparecida  
Modelos para gestão de projetos de desenvolvimento  
distribuído de software em uma empresa de automação  
bancária / Laine Aparecida Tavares  
São Paulo: [s.n.], 2016.  
109 f.: il.  
Orientador: Prof. Dr. José Braz de Araújo  
Monografia ( Pós-Graduação Lato Sensu em Gestão da  
Tecnologia da Informação ) - Instituto Federal de  
Educação, Ciência e Tecnologia de São Paulo, IFSP, 2016.  
1. Automação bancária 2. Desenvolvimento distribuído  
de software 3. Gerenciamento de projetos 4. SCRUM  
5. PMBOK I. Instituto Federal de Educação, Ciência e  
Tecnologia de São Paulo II Título

CDU 303.4

**LAINÉ APARECIDA TAVARES**

**MODELOS PARA GESTÃO DE DESENVOLVIMENTO DISTRIBUÍDO  
DE SOFTWARE EM UMA EMPRESA DE AUTOMAÇÃO BANCÁRIA**

Monografia apresentada ao Curso de Pós-Graduação *Lato Sensu* em Gestão da Tecnologia da Informação do Instituto Federal de Ciência e Tecnologia de São Paulo, Câmpus São Paulo, como requisito parcial para obtenção do título de Especialista em Gestão de Tecnologia da Informação.

São Paulo, 18 de junho de 2016.

---

Presidente: Prof. Dr. José Braz de Araujo – Orientador – IFSP

---

Membro: Prof. Dr. Domingos Bernardo Gomes Santos – IFSP

---

Membro: Prof. Me. Fernando Carvalho – IFSP

À minha família pela paciência, apoio e generosidade, durante o tempo em que me dediquei a esse projeto e que os privei de minha atenção e cuidados.

## **AGRADECIMENTOS**

Agradeço a Deus por me guiar nessa caminhada, dando-me sabedoria, paciência e perseverança.

Agradeço especialmente a meu orientador, Prof. Dr. José Braz de Araújo pela dedicação, disponibilidade e extrema paciência, cujo trabalho de orientação foi fundamental para que eu concluísse este estudo.

Agradeço aos Mestres Fernando Carvalho, Claudia Werhmuller e Francisco Veríssimo pelos ensinamentos e orientações que transformaram e continuam transformando minha vida profissional e pessoal.

Agradeço também aos colegas de classe pelas discussões e contribuições valorosas durante as aulas e por toda ajuda durante o desenvolvimento deste estudo.

Agradeço à equipe de GSD da Organização que tão gentilmente forneceu toda a documentação necessária e respondeu às entrevistas para que eu pudesse concluir este estudo.

## RESUMO

O setor bancário brasileiro é um dos maiores investidores em tecnologia da informação em âmbito mundial e responsável pelo desenvolvimento e consolidação das empresas de automação bancária nacionais e multinacionais. A competitividade das empresas de automação bancária levou à necessidade de aumento de produtividade e redução de custos, buscando soluções que suprissem esta demanda sem perda de qualidade e confiabilidade do produto final. O desenvolvimento distribuído de *software* tem sido uma abordagem cada vez mais utilizada por grandes organizações para o aproveitamento de mão de obra especializada e de baixo custo, entretanto o gerenciamento de projetos em ambiente distribuído ainda é crítico e pouco estudado. Este trabalho apresenta um estudo de caso realizado em uma empresa do setor de automação bancária que desenvolve seus projetos utilizando equipes locadas em diversos países com o propósito de avaliar os modelos de gerenciamento de projetos PMBOK e SCRUM, verificando sua aderência ao ambiente de desenvolvimento distribuído. O estudo de caso explora o modelo de gerenciamento de projetos adotado pela empresa, identificando os processos do guia PMBOK e SCRUM empregados. A análise dos resultados demonstra a importância da inclusão de processos de ambos os modelos avaliados de forma complementar, para aumento da efetividade do gerenciamento do desenvolvimento distribuído de *software* e a consequente melhoria de desempenho dos projetos.

**Palavras-chave:** Automação bancária, desenvolvimento distribuído de software, gerenciamento de projetos, SCRUM, PMBOK.

## **ABSTRACT**

The Brazilian banking sector is one of the largest worldwide investors in information technology and responsible for the development and consolidation of national and multinational banking automation companies. The competitiveness of banking automation business led to the need for increase in productivity and cost reduction, seeking production solutions that meet this demand without loss of quality and reliability of the final product. The distributed software development has been increasingly used by large organizations for exploitation of specialized and low-cost labor, though the project management in this distributed environment is still critical and still understudied. This paper presents a case study in a banking automation company that develops its projects using teams located in various countries in order to evaluate the project management models PMBOK and SCRUM, checking their adherence to the distributed development environment. The case study explores the model of project management adopted by the company, identifying the processes of the PMBOK and SCRUM applied. The analysis shows the importance of including processes of both evaluated models in a complementary way to increase the effectiveness of managing distributed software development and the improvement of performance of projects.

**Keywords:** Banking automation, distributed software development, project management, SCRUM, PMBOK.



## LISTA DE FIGURAS

Figura 1 – Participação do Setor Bancário no mercado de TI .....	15
Figura 2 – Quantidade de ATMs no Brasil .....	16
Figura 3 – Metodologias Ágeis Utilizadas nas Empresas .....	18
Figura 4 – Fases da Automação Bancária no Brasil .....	24
Figura 5 – Principais Forças para o desenvolvimento de DDS .....	26
Figura 6 – Gerenciamento de projeto é gerenciamento fractal .....	33
Figura 7 – Entradas da Metodologia de Gerenciamento de Projetos. ....	36
Figura 8 – Relação entre as partes interessadas e o projeto.....	39
Figura 9 – Integração das Áreas de Gerenciamento de Projeto .....	41
Figura 10 – Processo Base do SCRUM.....	45
Figura 11 Problemas de Comunicação.....	58
Figura 12 – Causas de Problemas de Comunicação.....	59
Figura 13 – Colaboração entre Equipes Distribuídas .....	59
Figura 14 – Influência Negativa dos Aspectos Culturais na Colaboração .....	60
Figura 15 – Possibilidade de Autogerenciamento da Equipe.....	61
Figura 16 – Preferência por Tipo de Avaliação de Desempenho.....	62
Figura 17 – Problemas de Comunicação.....	63
Figura 18 – Causas de Problemas de Comunicação.....	63
Figura 19 – Colaboração entre Equipes Distribuídas .....	64
Figura 20 – Influência Negativa dos Aspectos Culturais na Colaboração .....	64
Figura 21 – Possibilidade de Autogerenciamento da Equipe.....	65
Figura 22 – Preferência por Tipo de Avaliação de Desempenho.....	66
Figura 23 – Distribuição das Equipes do Projeto .....	69
Figura 24 – Tabela de Acompanhamento de Estórias .....	75
Figura 25 – Cartão de Estória.....	78
Figura 26 – Painel de Situação Geral do Projeto.....	79
Figura 27 – Painel de Custo do Projeto .....	82

## LISTA DE QUADROS

Quadro 1 – Estatística de Mudança de Requisitos .....	72
Quadro 2 – Matriz de Comunicações.....	87
Quadro 3 – Classificação de ocorrência de risco.....	89
Quadro 4 – Classificação de severidade de risco .....	90
Quadro 5 – Classificação de Detecção de risco .....	90
Quadro 6 – Classificação de Detecção de risco .....	92
Quadro 7 – Composição de Equipe CFT.....	93
Quadro 8 – Comparação de Processos PMBOK e SCRUM.....	95
Quadro 8 – Comparação de Processos PMBOK e SCRUM.....	96

## LISTA DE ABREVIATURAS

API	<i>Application Program Interface</i>
ATM	<i>Automated Teller Machine</i>
BSVC	<i>Bank Solutions Vendor Council</i>
CCB	<i>Change Control Board</i>
CCM	Comitê de Controle de Mudanças
CEN	Comitê Europeu De Normalização
CFT	<i>Concurrent Fulfillment Team</i>
DDS	Desenvolvimento Distribuído de Software
DOC	Documento de Crédito
DSDM	<i>Dynamic Systems Development Method</i>
EAP	Estrutura Analítica do Projeto
FEBRABAN	Federação Brasileira de Bancos
FTE	<i>Full Time Equivalent</i>
GSD	<i>Global Software Development</i>
ICB	<i>International Competence Baseline</i>
IPMA	<i>International Project Management Association</i>
KPI	<i>Key Performance Indicator</i>
PMBOK	<i>Project Management Body Of Knowledge</i>
PMI	<i>Project Management Institute</i>
PMO	<i>Project Management Office</i>
PND	Plano Nacional de Desenvolvimento
REQ ID	<i>Requirement Identity</i>
ROM	<i>Rough Order of Magnitude</i>
SCR ID	<i>Software Change Request Identity</i>
SPB	Sistema de Pagamentos Brasileiro
SQA	<i>Software Quality Assurance</i>
STR	Sistema de Transferência de Reservas
TI	Tecnologia da Informação
VPL	Valor Presente Líquido
WOSA	<i>Windows Open Service Architecture</i>
XFS	<i>Extensions for Financials Services</i>
XP	<i>Extreme Programming</i>

## SUMÁRIO

1	INTRODUÇÃO.....	12
1.1	QUESTÃO DE PESQUISA.....	14
1.2	OBJETIVOS.....	14
1.3	JUSTIFICATIVA.....	15
1.4	ESTRUTURA DO ESTUDO.....	19
2	REVISÃO DA LITERATURA.....	20
2.1	AUTOMAÇÃO BANCÁRIA NO BRASIL.....	20
2.2	DESENVOLVIMENTO DISTRIBUÍDO DE SOFTWARE (DDS).....	25
2.3	GERENCIAMENTO DE PROJETOS.....	32
2.3.1	Gerenciamento de Projetos de Software.....	34
2.3.2	Metodologia de Gerenciamento de Projetos.....	35
2.3.3	Modelo PMBOK.....	37
2.3.4	Modelo SCRUM.....	43
3	MÉTODO DE PESQUISA.....	50
3.1	TIPO DE PESQUISA.....	50
3.2	COLETA DE DADOS.....	51
3.3	ANÁLISE DE DADOS.....	53
4	RESULTADOS DA PESQUISA.....	55
4.1	CARACTERIZAÇÃO DA ORGANIZAÇÃO ESTUDADA.....	55
4.2	CARACTERIZAÇÃO DOS PRODUTOS DE SOFTWARE DESENVOLVIDOS PELA ORGANIZAÇÃO ESTUDADA.....	55
4.3	CARACTERIZAÇÃO DA METODOLOGIA DE GERENCIAMENTO DE DESENVOLVIMENTO DE SOFTWARE UTILIZADA NA ORGANIZAÇÃO ESTUDADA .....	56
4.4	ANÁLISE DESCRITIVA DA AMOSTRA.....	57
4.4.1	Análise dos dados de entrevista com Gerentes de Projeto.....	57
4.4.2	Análise dos dados de entrevista com Membros da Equipe.....	62
4.5	ELEMENTOS DE ANÁLISE.....	66
4.5.1	Gerenciamento de Recursos Humanos.....	66
4.5.2	Gerenciamento de Integração.....	70
4.5.3	Gerenciamento de Escopo.....	73
4.5.4	Gerenciamento de Tempo.....	77

4.5.5 Gerenciamento de Custos .....	81
4.5.6 Gerenciamento de Qualidade .....	83
4.5.7 Gerenciamento de Comunicações.....	85
4.5.8 Gerenciamento de Riscos.....	88
4.5.9 Gerenciamento de Aquisição.....	91
4.5.10 Gerenciamento de Partes Interessadas.....	92
4.6 DISCUSSÃO DOS RESULTADOS OBSERVADOS .....	94
5 CONSIDERAÇÕES FINAIS.....	99
5.1 RESPOSTA À QUESTÃO DE PESQUISA .....	99
5.2 OBJETIVOS PROPOSTOS .....	100
5.3 CONTRIBUIÇÕES ACADÊMICAS E GERENCIAIS.....	100
5.4 LIMITAÇÕES DE PESQUISA E CONTRIBUIÇÕES PARA ESTUDO .....	101
REFERÊNCIAS BIBLIOGRÁFICAS .....	102
APÊNDICE A – ROTEIRO DE ENTREVISTA SOBRE GERENCIAMENTO DE RECURSOS HUMANOS PARA GERENTES DE PROJETOS .....	108
APÊNDICE B – ROTEIRO DE ENTREVISTA SOBRE GERENCIAMENTO DE RECURSOS HUMANOS PARA MEMBROS DA EQUIPE .....	109

## 1 INTRODUÇÃO

O setor de automação bancária é um dos únicos setores da economia brasileira em que o desenvolvimento tecnológico é tão avançado quanto em setores estrangeiros equivalentes ou até mesmo superior em alguns aspectos, atingindo um alto grau de excelência. (DINIZ, 2004a).

Esse desenvolvimento da automação bancária brasileira foi consequência da demanda dos bancos por uma tecnologia cada vez mais avançada, que permitiu a oferta de serviços a um enorme e crescente contingente de clientes, transações e postos de atendimento, com velocidade, qualidade e segurança. (FONSECA; MEIRELLES; DINIZ, 2010). Assim, os bancos são, atualmente, os maiores investidores em tecnologia no mercado nacional e mesmo em momentos de menor crescimento da economia brasileira houve o aumento significativo dos investimentos em tecnologia pelos bancos, com gastos que crescem a uma taxa de 6% ao ano, desde 2010. (FEBRABAN, 2015).

Diante dessa expressividade de investimentos, o mercado de automação bancária se tornou cada vez mais competitivo, principalmente após a entrada de empresas multinacionais que intensificaram ainda mais a competição entre as empresas do setor. (IAMONTI, 2014).

O setor de automação bancária, assim como outros setores da economia, também sofreu as pressões de se manter rentável e competitivo diante de um mercado globalizado, o que favoreceu a adoção do modelo de desenvolvimento distribuído de *software*, que segundo Kiel (2003) consiste em se estabelecer um processo de desenvolvimento de *software* utilizando profissionais ou equipes de profissionais geograficamente dispersos. Este fato confirma a tese de Herbsleb (2007) de que o aumento da globalização de mercados e de produção tende a aumentar a pressão pela distribuição global de projetos.

A necessidade de baixo custo na obtenção de mão de obra, a possibilidade de trabalho ininterrupto durante as 24 horas do dia, aproveitando as diferenças de fuso horário entre países ou regiões, e a dificuldade de se encontrar todos os profissionais

necessários com qualificação em uma mesma região geográfica, viabilizou a estratégia de desenvolvimento distribuído de *software*. (CARMEL; TIJA, 2005).

Essas características peculiares do desenvolvimento distribuído de *software* ou DDS (PRIKLADNICKI; AUDY; EVARISTO, 2003), geram riscos que não são comuns ao desenvolvimento centralizado de *software* e que devem ser levados em consideração quando se planeja migrar projetos de *software* para este perfil de desenvolvimento. De acordo com Macconnell (1996), os principais riscos do DDS são: perda de visibilidade do progresso do projeto, transferência de conhecimento para fora da empresa, controle de informações confidenciais e falta e motivação da equipe local que se sente ameaçada pela possibilidade de perda de emprego.

Segundo Herbsleb (2007), o problema fundamental do desenvolvimento distribuído de *software* é que muitos dos métodos estabelecidos para gerenciar o trabalho de um projeto centralizado são inexistentes ou incompletos para projetos distribuídos.

Segundo Sommerville (2007), o bom gerenciamento não garante o sucesso do projeto, porém o mau gerenciamento frequentemente causa falhas no projeto, tais como aumento do prazo de entrega, aumento do custo e falha no atendimento aos requisitos. Neste contexto, a questão que surge é: qual modelo de gestão de projetos melhor se adequa ao contexto distribuído?

Existem diversos guias e *frameworks* que documentam as melhores práticas e conhecimentos sobre gerenciamento de projetos, baseados em duas abordagens: tradicional e ágil. A abordagem tradicional se caracteriza pela extensa documentação, definição detalhada dos requisitos do projeto antes de seu desenvolvimento e entrega de valor no final do projeto. Na abordagem ágil, por outro lado, o foco é maior no desenvolvimento do *software* do que na sua documentação, os requisitos são flexíveis e adaptáveis e as entregas de valor são de forma incremental ao longo do projeto.

Os guias e *frameworks* para gerenciamento de projetos mais populares são: *International Competence Baseline* (ICB3) desenvolvido pela *International Project Management Association* (IPMA) (2006), *Project in Controlled Environments* (PRINCE2) desenvolvido pelo *Office of Government Commerce* (OGC) (2009), *Guide to the Project Management Body of Knowledge* (PMBOK) desenvolvido pelo *Project Management Institute* (PMI) (2013), *Extreme Programming* (XP) (WELLS, 2009) e o Guia SCRUM (SCHWABER; SUTHERLAND, 2013).

Para delimitar o escopo do estudo será feita a comparação entre os guias PMBOK (PMI, 2013), que possui uma abordagem tradicional de gerenciamento de projetos e possui 478.493 profissionais certificados em 204 países (PMI, 2016), e o guia SCRUM (SCHWABER; SUTHERLAND, 2013), que possui uma abordagem ágil, e é o *framework* ágil mais utilizado mundialmente (SCRUM ALLIANCE, 2013). O objetivo dessa comparação é identificar as vantagens e desvantagens de cada modelo quando aplicados ao gerenciamento de projetos de desenvolvimento distribuído de *software* para sistemas de automação bancária.

Será usada como base de pesquisa uma empresa de automação bancária, multinacional, tendo instituições financeiras como principais clientes e cuja estratégia é a entrega de produtos de *hardware* e *software* inovadores e confiáveis com os mais altos padrões de segurança.

## 1.1 QUESTÃO DE PESQUISA

Considerando que a utilização de modelos de gerenciamento de projetos aplicadas ao desenvolvimento distribuído de *software* em empresas de automação bancária apresenta crescimento e que sua utilização pode gerar novas oportunidades às empresas, este estudo pretende responder à seguinte questão:

Considerando um modelo de gestão tradicional de projetos, como o proposto pelo PMI, e um modelo de gestão de projetos ágil, como o SCRUM, qual modelo de gestão de projetos melhor se adequa ao contexto de desenvolvimento distribuído de *software* em empresas de automação bancária?

## 1.2 OBJETIVOS

O objetivo principal deste estudo é responder à questão de pesquisa, utilizando-se como subsídio a comparação dos modelos de gestão de projeto de *software* PMBOK e SCRUM, verificando sua aderência ao desenvolvimento distribuído de *software*, garantindo confiabilidade e segurança ao produto final, a fim de atender à demanda da indústria de automação bancária.

Como complementação do objetivo principal, espera-se atingir, com este estudo, aos seguintes objetivos secundários:

- Descrever as características do desenvolvimento distribuído de *software*, identificando fatores críticos de gerenciamento;

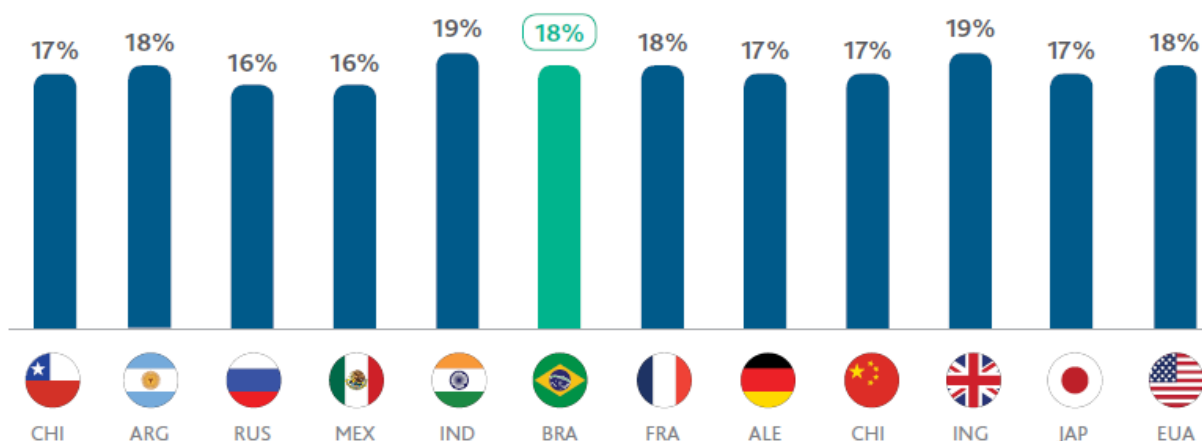


- Identificar as vantagens e desvantagens de cada modelo de gestão de projeto estudado, em ambiente de desenvolvimento distribuído de *software*;
- Apresentar a melhor solução de gestão de projeto, dentre os modelos estudados, para utilização em desenvolvimento distribuído de *software*.

### 1.3 JUSTIFICATIVA

A indústria bancária brasileira é um dos segmentos da economia que mais investe em tecnologia da informação (TI), representando 18% do total de gastos no mercado de TI em 2014 (Figura 1), mesmo índice registrado pelo setor financeiro de países desenvolvidos como Estados Unidos, França e Reino Unido. Além disso, os investimentos em TI feitos pelos bancos brasileiros crescem em média 6% ao ano desde 2010, alcançando U\$ 11,9 bilhões em 2014. (FEBRABAN, 2015).

**Figura 1 – Participação do Setor Bancário no mercado de TI**



Fonte: FEBRABAN (2015)

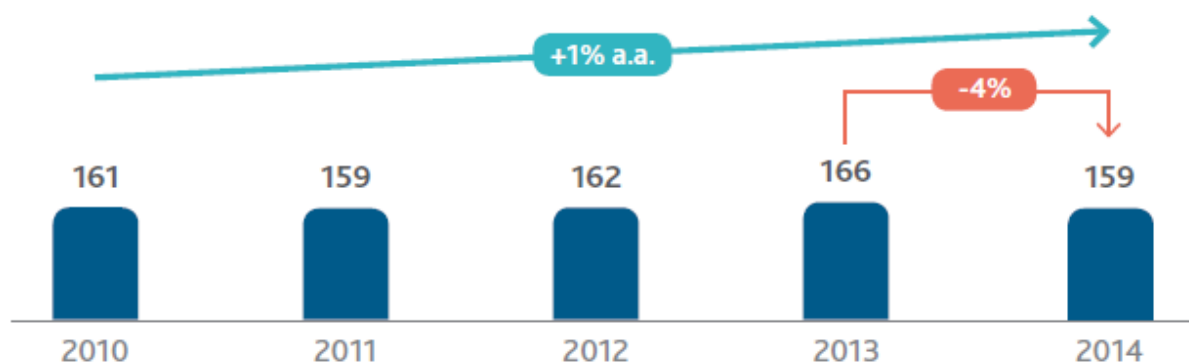
O setor bancário brasileiro, que é pioneiro no uso de TI no Brasil, já constatou que os investimentos nessa área promovem lucratividade e competitividade e que a crescente desmaterialização dos meios de pagamentos direciona a migração dos negócios do setor para ambientes digitais. Além disso, a inovação tecnológica possibilita o desenvolvimento de novos processos, produtos e serviços bancários, antes inviáveis, que representam vantagens também para os fornecedores de tecnologia na oferta e criação de novos usos para seus produtos. (FONSECA; MEIRELLES; DINIZ, 2010).

A necessidade do setor bancário por tecnologias cada vez mais avançadas trouxe como consequência o fortalecimento da indústria de automação bancária

brasileira, consolidando-a como uma das mais desenvolvidas do mundo. (FONSECA; MEIRELLES; DINIZ, 2010).

O ATM (*Automated Teller Machine* ou Equipamento de Caixa Automático) se apresenta como o produto principal da indústria de automação bancária e apesar do aumento de transações bancárias por meio da internet e utilizando-se de *smartphones*, segundo a FEBRABAN (2015), o número de ATMs instalados no Brasil cresce a uma taxa em torno de 1% ao ano (Figura 2). Observa-se uma retração no número de ATMs entre 2013 e 2014, que pode ser explicada pelo aumento de funções disponíveis por ATM que permite melhor utilização do terminal.

**Figura 2 – Quantidade de ATMs no Brasil  
(Milhares)**



Fonte: FEBRABAN (2015)

A importância das empresas de automação para os bancos é tão grande que estes influenciam diretamente o desenvolvimento de produtos em seus fornecedores conforme sua necessidade. (FONSECA; MEIRELLES; DINIZ, 2010).

Um exemplo dessa influência foi o desenvolvimento do padrão *Extensions for Financials Services* (XFS) promovido pelo *European Committee for Standardization* (conhecido pelo acrônimo CEN) e baseado no *Windows Open Service Architecture and Extensions for Financials Services* (WOSA XFS) desenvolvido pela empresa Microsoft, conforme requisitos do *Bank Solutions Vendor Council* (BSVC) que se constitui em um conselho internacional que inclui muitos dos principais integradores e fornecedores de *hardware*. O objetivo do padrão WOSA XFS é integrar com facilidade, aplicações Windows com ATMs de diversos fornecedores (KEYES, 1999).

Assim, as empresas de automação bancária devem fornecer junto com seus ATMs os diversos componentes de *software* que atendem ao padrão internacional

XFS e que permitem o desenvolvimento de aplicações bancárias independentes do *hardware* (CEN, 2000).

O estabelecimento de um padrão internacional de interface entre aplicações bancárias e produtos de *hardware*, aliado à abertura de mercado à importação iniciado em 1991, favoreceu a globalização do mercado de automação bancária, por meio da instalação de multinacionais em território brasileiro e pela associação de empresas brasileiras com empresas estrangeiras. (IAMONTI, 2014).

A globalização, reflexo de forças econômicas, expandiu o mercado nacional para mercados globais e apresentou novas formas de competição e cooperação que ultrapassou as fronteiras nacionais, impactando não somente o comércio e a distribuição, mas a forma com que os produtos são concebidos, arquitetados, construídos, testados e entregues ao cliente. (HERBSLEB; MOITRA, 2001).

Nos últimos anos, produtos de *software* se tornaram componentes vitais em quase todos os ramos de negócio, conseqüentemente o sucesso depende do uso do *software* como estratégia competitiva. Assim, a procura por preços menores e acesso a recursos mais qualificados direcionou as organizações a experimentarem as facilidades do desenvolvimento distribuído de *software* (DDS). (HERBSLEB; MOITRA, 2001).

Outros fatores importantes para o desenvolvimento distribuído de *software* foram a diminuição de custos e o aumento de qualidade dos meios de comunicação. (CARMEL; TIJA, 2005).

Em se tratando de globalização, há que se considerar, também, a importância das habilidades gerenciais da empresa que, segundo Kerzner (2007), tem mais impacto sobre o sucesso global do que os produtos oferecidos, sendo que a combinação de uma metodologia-padrão com o talento administrativo traz um aumento significativo no sucesso em âmbito global.

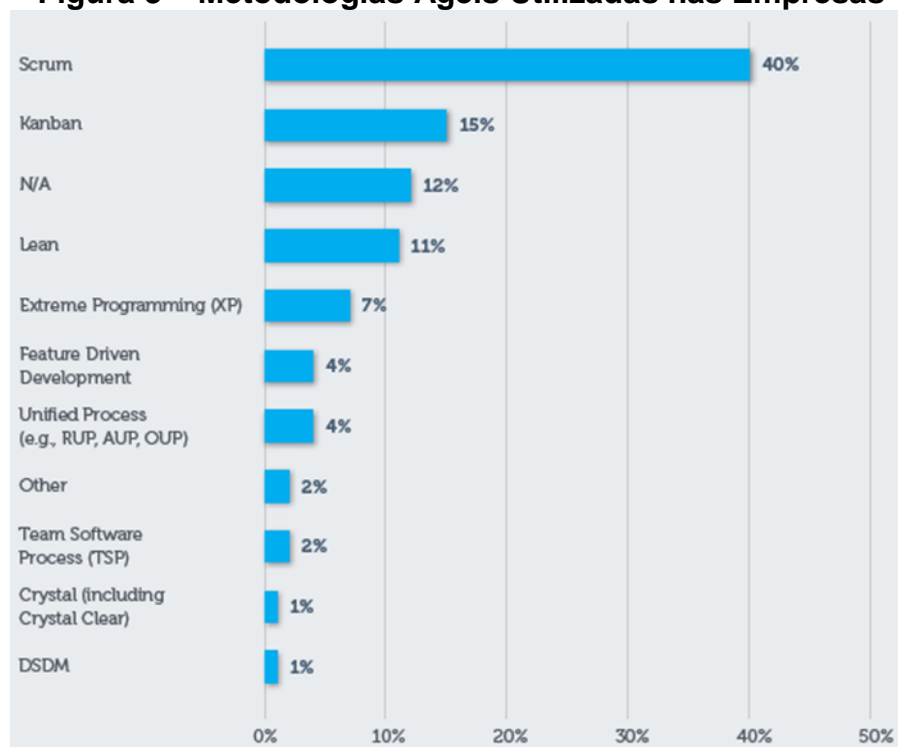
De acordo com PMI (2016), quando o gerenciamento de projetos e programas é inserido nos processos de uma organização há o aumento de desempenho e a aceleração da vantagem competitiva. Ainda de acordo com o instituto, as organizações de alto desempenho atingem seus objetivos 2,5 vezes mais frequentemente e gastam 13 vezes menos dinheiro que suas concorrentes com baixo desempenho.

As empresas costumam definir sua própria metodologia de gestão de projetos de *software* usando como base guias e *frameworks* com as melhores práticas do mercado, tal como o PMBOK (*Project Management Body of Knowledge*), que segundo Cable (2011), é o guia mais amplamente aceito para a gestão de projetos, com cerca de 2,8 milhões de cópias em circulação, disponível em 10 idiomas e 307.000 membros em 180 países.

Por outro lado, a partir de 2001, iniciou-se um movimento pró-metodologias ágeis para projetos de *software*, pelo chamado “*The Agile Alliance*”, grupo formado por pensadores independentes com o objetivo de estabelecer uma forma de gerenciamento que evitasse principalmente a burocracia imposta pelas metodologias tradicionais. (HIGHSMITH, 2001).

Segundo a Scrum Alliance (2013), SCRUM é a metodologia com abordagem ágil mais popular. De acordo com pesquisa realizada por essa organização, 40% de 499 empresas pesquisadas utilizam SCRUM para a gestão de seus projetos, conforme demonstrado na Figura 3.

**Figura 3 – Metodologias Ágeis Utilizadas nas Empresas**



Fonte: Scrum Alliance (2013)

O presente trabalho visa analisar a aderência do PMBOK (modelo tradicional) e SCRUM (modelo ágil) ao desenvolvimento distribuído de *software* (DDS), de forma a aliar a melhoria de qualidade proporcionada por estes modelos à produtividade do DDS, atendendo à demanda crescente das empresas de automação bancária por entregas de *software* regidas por padrões internacionais com maior confiabilidade, principalmente no que diz respeito à segurança, em menor tempo possível, com otimização dos recursos humanos e físicos.

#### 1.4 ESTRUTURA DO ESTUDO

Este trabalho está estruturado em seis capítulos, incluindo a Introdução. O segundo capítulo apresenta uma revisão da literatura sobre automação bancária, desenvolvimento distribuído de *software* e modelos para gestão de projetos, onde são apresentados os modelos de gestão de projetos utilizados como base para o desenvolvimento deste estudo. O terceiro capítulo aborda os métodos de pesquisa utilizados, com a apresentação das escalas utilizadas, instrumentos de pesquisa e os procedimentos para coleta e análise dos dados. O quarto capítulo traz os resultados da pesquisa, com a discussão dos resultados observados. O quinto capítulo apresenta as considerações finais, explicitando as contribuições obtidas e os trabalhos futuros propostos. Encerra-se o estudo com as referências às obras utilizadas na fundamentação teórica, quer em formato tradicional de papel, quer em artigos acadêmicos, órgão de pesquisa entre outras fontes primárias e secundárias veiculadas via *web*.

## 2 REVISÃO DA LITERATURA

Este capítulo tem como objetivo propiciar um arcabouço teórico sobre os assuntos tratados neste estudo, ou seja, automação bancária, desenvolvimento distribuído de *software* e gerenciamento de projetos. Para isso, ele foi dividido em três tópicos, sendo cada um destinado a um assunto.

### 2.1 AUTOMAÇÃO BANCÁRIA NO BRASIL

A história da automação bancária no Brasil se inicia na década de 1960 com a instalação dos primeiros computadores (*mainframes*) nos bancos comerciais, que junto com órgãos governamentais se tornaram os maiores consumidores de tecnologia da informação. A necessidade de um grande volume de processamento de transações financeiras, durante as décadas de 1960 e 1970, induziu o domínio do mercado de automação bancária por fornecedores de computadores de médio e grande porte. As três maiores empresas de automação foram: IBM, empresa americana que em meados da década de 1970, possuía 65% do mercado, seguida pela Burroughs, também americana, com 17% e pela empresa francesa *Compagnie des Machines Bull* com 6%. (FRISCHTAK, 1992).

Devido ao alto custo dos equipamentos, o sistema era centralizado em um único local chamado centro de processamento de dados (CPD). Os dados eram enviados pelas agências bancárias à noite e o resultado do processamento era devolvido em listagens impressas em papel no dia seguinte ou em até três dias depois, dependendo da distância da agência. Esse tempo gerava insatisfação dos clientes, o que motivou o desenvolvimento de subcentros regionais de processamento que recebiam os dados das agências e transferiam aos CPDs por linha telefônica. (DINIZ, 2004a).

Entretanto, uma vez que a maior parte dos dados enviados pelas agências bancárias aos CPDs eram consumidos pelas próprias agências, começaram os investimentos em sistemas distribuídos, com a informatização parcial das agências, permitindo o chamado processamento *online* e respostas em tempo real. O apelo de marketing da automação das agências foi tão forte que os bancos continuaram os investimentos mesmo com a percepção de que o grau de automação do sistema não era eficiente em termos de custo. (FRISCHTAK, 1992).

A vantagem competitiva da informatização dos bancos continuava a impulsionar os investimentos em inovação tecnológica bancária. Entretanto, em 1974 foi instituído pelo governo federal, o II PND (Plano Nacional de Desenvolvimento) que, visando à emancipação tecnológica da área de informática, estabeleceu restrições às importações para favorecer o desenvolvimento da indústria de minicomputadores. Essa restrição às importações aliada à pouca importância que os fornecedores estrangeiros davam ao mercado brasileiro direcionou os investimentos dos bancos, tanto de capital como de recursos humanos, para o desenvolvimento da indústria nacional de informática. Por meio da alocação de especialistas do banco para trabalho em conjunto com desenvolvedores de *hardware* e *software* chegou-se a um alto grau de eficiência na fabricação de produtos de informática direcionados a serviços bancários. (DINIZ, 2004a).

Como exemplo dessa estreita relação entre bancos e desenvolvedores de sistemas de automação podemos citar a criação da empresa estatal Cobra (Computadores e Sistemas Brasileiros Ltda.), em 1974, com a colaboração de doze instituições financeiras. A partir dessa experiência, começaram também a se desenvolver no Brasil empresas privadas com investimentos de capital e recursos humanos das instituições bancárias, tais como a SID Informática, fundada em 1978 com investimentos do Banco Bradesco, e a Itautec, fundada em 1979 com capital do Banco Itaú. Em 1985 foi fundada a Procomp por ex-funcionários da SID e que mais tarde se tornaria líder do segmento de automação bancária no Brasil. (IAMONTI, 2014).

Os altos investimentos em tecnologia pelos bancos brasileiros também foram motivados pela necessidade de elevar a circulação da moeda nacional, que diante da aceleração da inflação, não poderia ficar ociosa sem aplicação. Nesse contexto de alta inflação na década de 1980, o desenvolvimento de soluções tecnológicas foi de extrema importância para agilizar o processo de gestão de ativos, contas correntes, câmbio, e relacionamento com clientes. Além disso, a área executiva dos bancos passou a ter acesso, em tempo real, a informações sobre a evolução das transações financeiras desde as agências até o nível global, o que favoreceu o aumento da velocidade de tomada de decisões para não se perder o valor da moeda corrente. (BANCO DO BRASIL, 2010).

Enquanto os altos índices de inflação impulsionaram a inovação tecnológica do sistema bancário na década de 1980, a desaceleração da inflação a partir da metade da década de 1990 também demandou projetos de automação bancária que provessem aos bancos soluções para aumento da eficiência operacional e redução de custo. Com o fim do governo militar, houve uma série de programas visando o combate à hiperinflação que assolava o país. (DINIZ, 2004a). Com o Estado atuando fortemente por meio de políticas econômicas e financeiras e a estabilização da moeda em 1994 com o Plano Real, o sistema bancário, não contanto mais com as elevadas transferências inflacionárias, teve sua rentabilidade reduzida. Iniciou-se então, uma profunda reestruturação no setor bancário brasileiro, marcada principalmente pelo fechamento de agências com baixa rentabilidade e redução drástica do quadro de funcionários. (SEGNINI, 1999).

Durante essa fase de reestruturação, o grande motivador dos investimentos em TI foi a possibilidade de diminuir os custos com pessoal e aumentar a produtividade. De acordo com Maçada e Becker (2001), enquanto os executivos norte-americanos têm a percepção de que a TI exerce impacto sobre a competitividade dos bancos através da influência nos fatores de competição internacional (clientes, estrutura de custos e capacidade), na percepção dos executivos brasileiros os impactos da TI na competitividade dos bancos estão relacionados apenas com os impactos da TI na estrutura de custos e capacidade.

O período entre 1985 e 1999 foi, de fato, o que podemos chamar de os anos de ouro da automação bancária no Brasil porque foram realmente inovadores. Foi nessa época em que se colocou a tecnologia trabalhando efetivamente para o negócio dos bancos. Até então, a tecnologia era usada para a produtividade interna dos bancos. Quando ela foi “colocada para fora”, também aumentou o desafio das áreas de tecnologia. Vivemos uma época muito interessante, realmente corremos atrás. E, efetivamente, a tecnologia contribuiu. (FONSECA; MEIRELLES; DINIZ, 2010, p.188).

Na segunda metade da década de 1990 intensificou-se o movimento de externalização dos serviços bancários, através da instalação de caixas eletrônicos em quiosques fora das agências bancárias. Os caixas eletrônicos também chamados ATMs (*Automated Teller Machines*) são equipamentos que possibilitam saques de numerários, depósitos em dinheiro ou cheques, consultas a saldos de contas e outros serviços bancários. (FEBRABAN, 2015).



Durante a década de 1990, ocorreu também a popularização dos serviços pelo telefone. Além disso, foram aumentados os investimentos em multimídia, processamento de imagens e redes de comunicação de dados mais eficientes. (FEBRABAN, 2015). Esses novos canais de comunicação, aliados à popularização dos computadores pessoais favoreceram o surgimento dos conceitos de *home banking* e *office banking* que se baseavam na instalação de aplicações proprietárias fornecidas pelos bancos por meio de disquetes que se conectavam aos sistemas do banco através de linha telefônica. Entretanto esses novos canais de atendimento, anunciados como “banco do futuro” e tão amplamente explorados pelos anúncios publicitários, não obtiveram a adesão esperada, pois somente uma pequena parte dos clientes possuía equipamento para conexão com os bancos e a logística de distribuição dos aplicativos aos usuários e suporte eram complexos. (FONSECA; MEIRELLES; DINIZ, 2010).

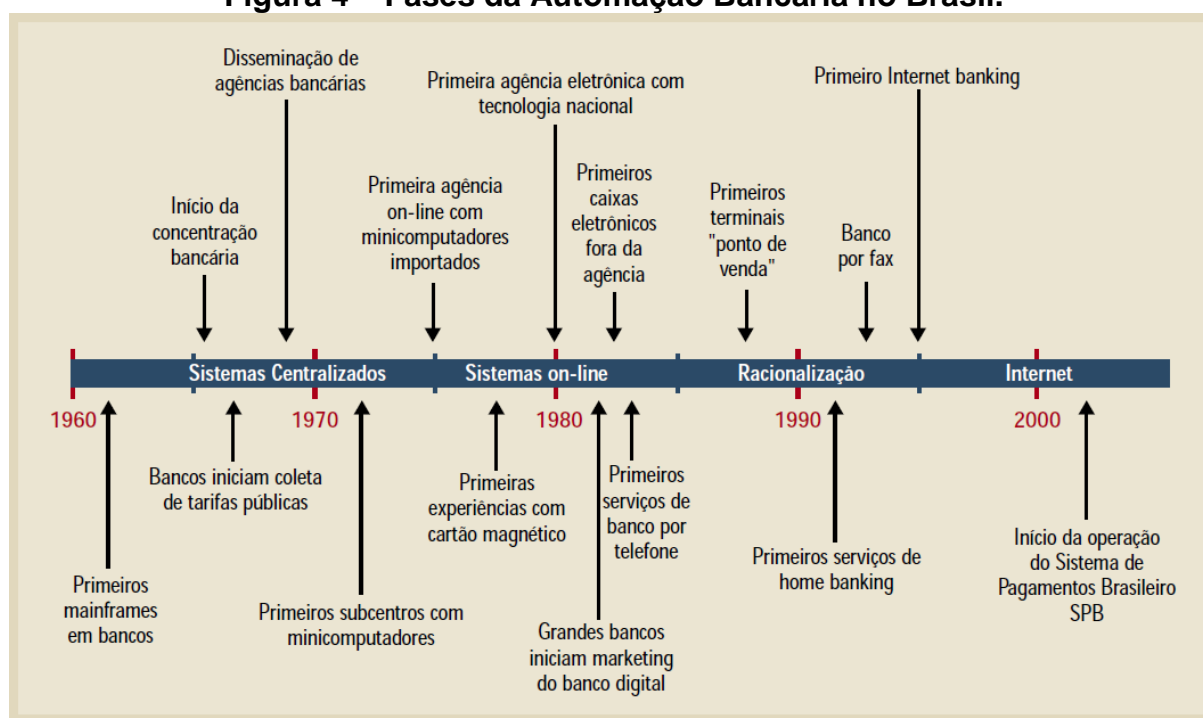
O conceito de banco virtual só se disseminou graças à abertura da internet para a área comercial em 1994, sendo então chamado *Internet Banking*. O Banco Bradesco foi o primeiro banco brasileiro a criar um site de internet comercial no Brasil, em 1995. (DINIZ, 2004b).

Durante os anos seguintes, houve uma significativa evolução do *Internet Banking*, que deixou de ser somente um canal de consulta a informações de saldos de contas bancárias e passou a possibilitar uma ampla gama de serviços tais como pagamento de contas, investimentos, transferências de fundos, solicitações de empréstimos entre outros. Além disso, o uso do *Internet Banking* foi de extrema importância para estabelecimento de melhores estratégias de marketing, pois permitiu que os bancos monitorassem os hábitos de seus clientes de forma mais rápida e eficiente, alimentando inúmeras análises mercadológicas e de comportamento do consumidor. Diversos produtos puderam ser oferecidos além de benefícios e facilidades que estimulariam a lealdade dos clientes. Esse grau de desenvolvimento foi possibilitado pela natureza onipresente e interativa da internet, e principalmente pelo aperfeiçoamento dos processos de segurança de dados acessíveis por meio de rede de computadores, que tinham como base as técnicas já utilizadas pelos bancos em outros canais de atendimento. Pode-se dizer que muito do que foi alcançado no tocante ao aprimoramento dos sistemas de segurança para transferência de dados

pela internet se deve à entrada dos bancos nesse contexto. (FONSECA; MEIRELLES; DINIZ, 2010).

Outro marco importante para o avanço tecnológico do sistema bancário foi a reestruturação do Sistema de Pagamentos Brasileiro (SPB) que foi implantada em 2002 pelo Banco Central do Brasil, conforme demonstrado na Figura 4.

**Figura 4 – Fases da Automação Bancária no Brasil.**



Fonte: DINIZ (2004a)

Esse novo sistema permitiu a compensação on-line das operações efetuadas pelo Sistema de Transferência de Reservas (STR), reduzindo o risco de uma instituição financeira em dificuldades afetar clientes, investidores e o Banco Central. As transações bancárias passaram a ser executadas com mais segurança e sem a espera necessária ao processamento de cheques ou de Documento de Crédito (DOC) no sistema de compensação tradicional. Uma das inovações apresentadas foi a certificação digital, sendo o primeiro caso prático de assinatura digital no Brasil a garantir a autenticidade das mensagens quanto à sua origem e destino. A partir daí o país passou a utilizar a tecnologia de criptografia e de assinatura digital em larga escala. (SERPRO, 2003).

## 2.2 DESENVOLVIMENTO DISTRIBUÍDO DE *SOFTWARE* (DDS)

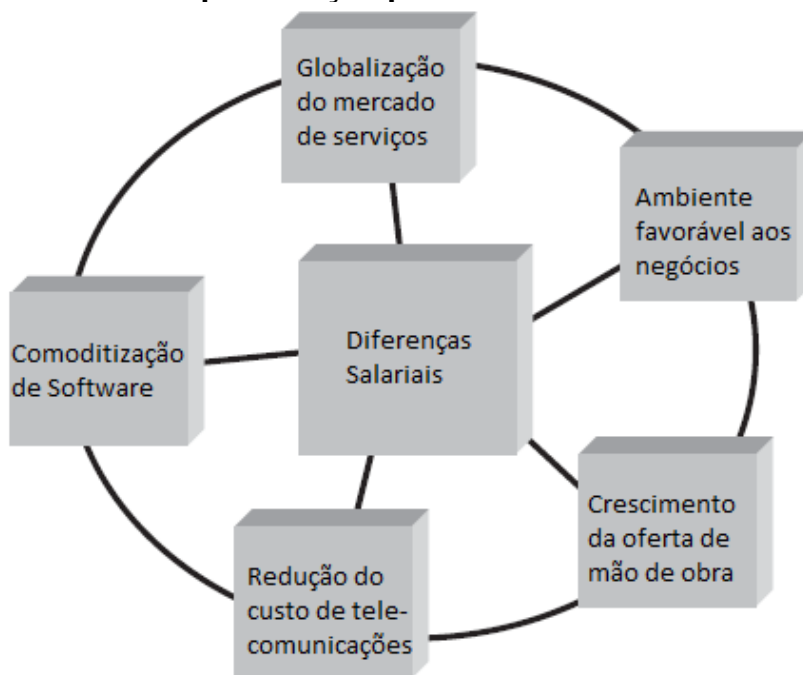
Segundo Pressman (2011), o fator mais importante em qualquer processo de desenvolvimento de *software* são as pessoas.

Em anos recentes, não é novidade o sentimento do empresariado de que faltam profissionais de TI. A sensação de falta é expressa de diferentes modos: alguns declaram não encontrar candidatos para as vagas ofertadas; outros informam que sim, há interessados nas vagas, mas eles não possuem a qualificação necessária; outros, ainda, dizem que têm dificuldades para encontrar o profissional desejado para determinadas funções, mas não para todas as funções; outros ressaltam a grande dificuldade para reter os profissionais de interesse, apontando diferenças na forma com que a geração Y lida com as relações de trabalho. Enfim, o sentimento de falta existe, embora não exista consenso sobre o que falta e qual seria a solução para o problema. (SOFTEX, 2013, p.7)

Uma das soluções para a falta de profissionais especializados é o desenvolvimento distribuído de *software* (DDS), que permite que profissionais geograficamente distribuídos atuem em um mesmo projeto de *software*, através da formação de equipes virtuais. Entre outras vantagens este modelo de desenvolvimento permite o aproveitamento de profissionais disponíveis em diversas regiões geográficas com baixo custo e com diversidade de qualificação. (ÅGERFALK; FITZGERALD; HOLMSTROM; Ó CONCHÚIR, 2008).

Segundo Carmel e Tija (2005), o trabalho distribuído no setor de TI é um marco importante na história da economia global. Os autores identificaram a existência de seis forças principais que convergiram para o desenvolvimento deste modelo de trabalho colaborativo, conforme demonstrado na Figura 5. A primeira força citada é a globalização do mercado de serviços, a partir da presença no mercado doméstico de empresas provedoras de serviços tecnológicos com atuação global. A globalização do setor começou na década de 1980, impulsionada pelo colapso do bloco soviético.

**Figura 5 – Principais Forças para o desenvolvimento de DDS**



Fonte: (CARMEL; TIJA, 2005).

A segunda força do DDS diz respeito ao ambiente favorável aos negócios, desenvolvido quando nações começaram a competir umas com as outras por investimentos em setores de *software*, oferecendo incentivos fiscais, flexibilização de regulamentações governamentais e, principalmente, criando parques tecnológicos que facilitam as operações de negócio. Índia, Filipinas e China são exemplos de países com ambientes favoráveis ao negócio. (CARMEL; TIJA, 2005).

A terceira força, crescimento da oferta de mão de obra qualificada, se deu pelo aumento do número de profissionais formados por universidades e escolas técnicas também na Índia, Filipinas, China e em outros países pelo mundo. Anualmente, a China forma quatro vezes mais engenheiros que os Estados Unidos. (CARMEL; TIJA, 2005). Antigamente, estes profissionais teriam que migrar para países mais industrializados e com mais oportunidades para se estabelecer, porém atualmente as empresas conseguem ter acesso a eles em qualquer lugar em que estejam. (CARMEL; TIJA, 2005).

A quarta força, redução do custo de telecomunicações, do modelo de Carmel e Tija (2005), foi muito expressiva desde o final da década de 1990, isso facilitou notavelmente o trabalho compartilhado entre equipes de países distantes. Além disso, a capacidade de transmissão de dados tem se expandido de quase zero na década

de 1990, chegando a 100 gigabits por segundo através do uso da fibra ótica em 2012. (IEEE, 2016).

A quinta força, comoditização (do inglês *commodity*) do *software*, no contexto da indústria de *software* é a padronização de práticas e ferramentas de desenvolvimento de *software*, transformando o *software* em um produto de características e qualidades uniformes independentemente de quem o produza. Uma vez que algumas tarefas são comoditizadas, os produtos de *software* podem ser produzidos pelo menor custo e com maior velocidade. (CARMEL; TIJA, 2005).

A diferença salarial entre nações, que constitui a força central do modelo de Carmel e Tija (2005), leva a custos mais baixos de desenvolvimento de *software* e tornou o DDS uma necessidade estratégica para algumas empresas. Até década de 1990, a mão de obra das empresas de tecnologia se enquadrava nas faixas de altos e médios salários. A pressão por diminuição do custo da folha de pagamento não partiu somente dos executivos das grandes corporações, mas também pelas firmas de investimentos que em um esforço para reduzir seus aportes em empresas *startups* de tecnologia, exigiam que os gastos com pesquisa e desenvolvimento fossem feitos em nações mais viáveis economicamente.

O acesso ao mercado também tem sido uma força propulsora do DDS para grandes empresas de tecnologia que são forçadas realizar investimentos nos países com os quais deseja manter relações comerciais. A China é o principal exemplo disso. Nenhuma empresa de tecnologia pode vender seus produtos para a China sem possuir uma fábrica ou um laboratório de Pesquisa e Desenvolvimento no país. Em 2003 a China definiu o seu próprio padrão de criptografia para redes locais sem fio, assim as empresas estrangeiras que queriam acessar este mercado foram forçadas a colaborar com pesquisa e desenvolvimento de *software* de empresas locais. (CARMEL; TIJA, 2005).

Segundo Herbsleb e Moitra (2001), existem também os fatores que aceleraram a tendência à globalização do desenvolvimento de produtos de *software*, são eles:

- A necessidade de captar no mercado global recursos escassos não importando a localização dos mesmos;
- As vantagens comerciais de proximidade ao mercado, incluindo o conhecimento dos clientes e das condições locais, bem como a boa vontade gerada por investimentos locais;

- A rápida formação de corporações virtuais e equipes virtuais para explorar as oportunidades de mercado;
- A forte pressão para melhorar o tempo para mercado usando as diferenças de fuso horário no desenvolvimento;
- A necessidade de flexibilidade para capitalizar oportunidades de fusão e aquisição onde quer que elas se apresentem.

O modelo de desenvolvimento distribuído de *software* permite que os membros da equipe estejam situados em vários locais remotos durante o ciclo de vida do *software*, tornando-se uma rede de equipes distantes. A distância entre as equipes pode variar de poucos metros (quando as equipes trabalham em edifícios adjacentes) para diferentes continentes. Em alguns casos, essas equipes podem ser membros da mesma organização, em outros casos, pode existir a colaboração ou terceirização envolvendo diferentes organizações. (PRIKLADNICKI; AUDY; EVARISTO, 2003).

No caso do DDS envolver equipes em diferentes países, tem-se o chamado desenvolvimento de *software* global (*global software development - GSD*), que envolve os três tipos principais de distância: geográficas, temporais e socioculturais. Distâncias geográficas vastas implicam na dificuldade de interação entre as equipes de desenvolvimento, ou seja, as tradicionais reuniões face a face não são comuns e a interação entre os membros requer o uso da tecnologia para facilitar a comunicação e coordenação. Distância temporal em vários fusos horários reduz o número de sobreposição de horas de trabalho, forçando uma maior dependência em tecnologias de comunicação assíncronas. Distância sociocultural surge a partir das diferentes origens nacionais e organizacionais das pessoas envolvidas e exacerba falha de comunicação. (ÅGERFALK; FITZGERALD; HOLMSTROM; Ó CONCHÚIR, 2008).

De acordo com Hofstede (1997) a distância sociocultural diz respeito a quanto os membros de diversas culturas diferem no que diz respeito a cinco valores:

- Distância hierárquica, grau de dependência entre chefe e subordinados;
- Individualismo versus coletivismo;
- Machismo versus feminismo;
- Tolerância às incertezas ou situações desconhecidas.
- Orientação de longo prazo, importância dada ao futuro contra passado e presente.

Diante dessa situação, o trabalho de articulação, desempenha um importante papel para a gestão da distribuição global do desenvolvimento. A articulação é aplicada durante as discussões em que os atores reflexivamente apresentam problemas e soluções a partir de suas perspectivas distintas e experiências práticas. (BODEN; NETT; WULF, 2007).

De acordo com Brooks (1995) a comunicação é uma questão-chave em projetos de desenvolvimento de *software* com um grande número de participantes. Erros de planejamento, no atendimento aos requisitos funcionais e erros do sistema surgem de uma falha na comunicação entre equipes diferentes. Em projetos envolvendo equipes distribuídas em localidades distantes com diferenças culturais e de idioma, a comunicação torna-se ainda mais crítica, podendo levar a mal-entendidos e atrasos. Sem informação efetiva e mecanismos de compartilhamento de conhecimento, os gestores não poderão explorar os benefícios do DDS. (HERBSLEB, 2007).

Se por um lado as distâncias geográficas, temporais e principalmente, as distâncias socioculturais demandam maior eficiência do sistema de comunicação, por outro lado, a diversidade promove criatividade e inspiração. Uma equipe global de desenvolvimento cria uma sinergia cultural que encontra novas maneiras de resolver problemas, além de novas ideias para inovação do processo produtivo. De forma colaborativa, os membros da equipe trazem novas perspectivas e soluções que produzem impacto positivo sobre o produto. (CARMEL, 1999).

O gerenciamento de projeto no ambiente DDS necessita uma abordagem diferenciada com relação ao gerenciamento de projetos centralizados. Existem na literatura alguns modelos de gerenciamento propostos para DDS que basicamente propõem a sincronização como principal característica. A sincronização dos processos requer marcos comuns bem definidos e critérios de entrada e saída claros e objetivos. (HERBSLEB; MOITRA, 2001). No modelo DDS, o trabalho de desenvolvimento é realizado simultaneamente por muitos indivíduos e equipes, e, em seguida, integrados em um único produto. Peças de *software* precisam integrar e interoperar corretamente, e cronogramas de produção precisam ser sincronizados, criando dependências entre tarefas e pessoas. Sob o ponto de vista técnico, a integração de várias peças de *software* precisa ser coordenada através de mecanismos como a comunicação da equipe, especificações, e gerenciamento de

configuração. Sob a perspectiva da gestão de projetos, a gestão do tempo é particularmente importante, já que pequenos atrasos em algumas partes do *software* podem gerar atraso em todo o projeto. Do mesmo modo, do ponto de vista do processo de desenvolvimento de *software*, a falta de conformidade com os processos estabelecidos pode ter efeitos negativos sobre os resultados do projeto tais como erros de requisitos e conflitos de prioridade. (ESPINOSA; SLAUGHTER; KRAUT; HERBSLEB, 2007).

Outro desafio crítico para a gestão de DDS é a construção e manutenção da confiança e coesão dos membros da equipe. (LEMMEEX, 2005). Quando os membros da equipe são separados geograficamente, existe a dificuldade do indivíduo para se familiarizar com os habilidades e hábitos de trabalho de seus colegas, identificar e transferir conhecimentos quando necessário, principalmente conhecimentos tácitos e desenvolver a consciência contextual. (HERBSLEB; GRINTER, 1999). Além disso, existe também a restrição à capacidade de conexão emocional, que faz com que as pessoas gostem umas das outras, se ajudem e trabalhem umas pelas outras, desenvolvendo um espírito de equipe. Mesmo em equipes centralizadas, a construção de companheirismo e confiança demanda tempo e ainda mais em equipes distribuídas, gerando um paradoxo: é necessário confiança para trabalho eficiente à distância e é muito difícil desenvolver confiança à distância. Outro fator negativo é a diferença cultural em que algumas culturas tendem a confiar mais lentamente que outras. (CARMEL, 1999).

De acordo com Lemmex (2005), a gestão eficiente do processo de desenvolvimento de *software* utilizando equipes distribuídas consiste em seis atividades principais:

1. Planejamento de comunicação: estabelecer o que precisa ser comunicado, como será comunicado, quem precisa da informação, quando a informação é necessária, e o que acontece se a comunicação for interrompida. O planejamento deve levar em consideração também os responsáveis pela publicação das informações e pelas reuniões de acompanhamento.

2. Definição do processo de tomada de decisão: como as decisões serão tomadas em grupo, qual a estratégia para definição do membro do grupo com prioridade de decisão e como essa responsabilidade será escalada em caso de indisponibilidade dessa pessoa.



3. Definição da estratégia de resolução de conflitos: como serão resolvidos os conflitos e quais regras devem ser estabelecidas para evitá-los.

4. Distribuição de objetivos, papéis e responsabilidades: estabelecer as metas, comunicar a responsabilidade por cada atividade, incluindo as suas responsabilidades específicas.

5. Garantia de distribuição justa de trabalho, com ajustes ao longo do projeto.

6. Decisão sobre o seu nível de liderança: a quem atribuir papel de líder, e se essa liderança é adequada tanto para a equipe como um todo e quanto para os indivíduos.

Por outro lado, de acordo com Chinbat (2010), os fatores que possibilitam às equipes distribuídas obter sucesso no desenvolvimento de projetos de software estão sumarizados em seis lições:

1. Empresas de TI devem adotar estratégia específica para o gerenciamento de equipe virtual global.
2. A formação da equipe é crucial na equipe virtual global.
3. Projetos virtuais globais devem ser cuidadosamente preparados com antecedência.
4. O processo de desenvolvimento de software deve ser adaptado para o ambiente de software virtual.
5. A gestão do conhecimento desempenha papel importante no ambiente de trabalho distribuído.
6. O uso adaptado da tecnologia é crucial para o sucesso da equipe global virtual.

De acordo com Begel e Nagappan (2008), os principais benefícios do desenvolvimento distribuído podem ser divididos em ferramentas, processos e cultura. O DDS induz à produção de documentação mais detalhada e sistemas automatizados de gestão de mudanças em projetos. Promove a experimentação de novas ferramentas de comunicação tais como sites de base de conhecimento na internet e dispositivos móveis. Além disso, melhora os processos de desenvolvimento de *software*, produzindo uma melhor comunicação sobre as expectativas, implantação de gestão do conhecimento, e incentiva a criação de uma equipe de gestão fortemente atuante. Culturalmente, ele expõe os membros da equipe a novas ideias, novas

formas de negociação e a contatos com especialistas das mais diversas culturas ao redor do mundo.

### 2.3 GERENCIAMENTO DE PROJETOS

Segundo Kerzner (2007), gerenciamento de projetos consiste em atividades de planejamento, programação e controle de tarefas integradas e multifuncionais, considerando recursos, prazo, custo e qualidade, de forma a atingir seus objetivos, beneficiando os participantes do projeto. No contexto empresarial, cada vez mais complexo e competitivo, a gestão de projetos está sendo reconhecida como importante fator para aumento da produtividade e eficiência não apenas no presente, mas também como estratégia para longo prazo. Segundo o autor, as empresas que adotaram metodologias de gestão de projetos obtiveram os seguintes benefícios:

- Redução de prazos;
- Redução de recursos necessários;
- Aumento da lucratividade;
- Melhor controle das mudanças de escopo;
- Maior eficiência na utilização de princípios do comportamento organizacional;
- Maior proximidade na relação com os clientes;
- Forma eficiente de resolução de problemas;
- Aumento da qualidade;
- Redução de disputas internas por poder;
- Melhoria na tomada de decisões;
- Maior produção de soluções;
- Progressão da empresa.

Turner (2009) entende que apesar de existirem diversas definições de ciclo de vida de projetos, o ciclo de vida de grandes projetos pode ser dividido em 5 estágios básicos e bem distintos: conceito, viabilidade, concepção, execução, fechamento. Em projetos menores existirá somente uma fase em todo o projeto, o que torna o ciclo de vida do projeto e os processos de gerenciamento indistinguíveis.

O projeto se inicia a partir de um conceito, ou seja, existe um problema a ser solucionado ou oportunidade de melhoria que proporcionará aumento de desempenho e geração de valor. Nesta fase são apresentadas soluções e feita uma estimativa grosseira de custo e benefício.

Durante o segundo estágio é feito estudo de viabilidade, através da captação de mais informações a respeito do projeto, compara-se as opções de solução e uma solução é escolhida para desenvolvimento. A estimativa de custo/benefício é refinada.

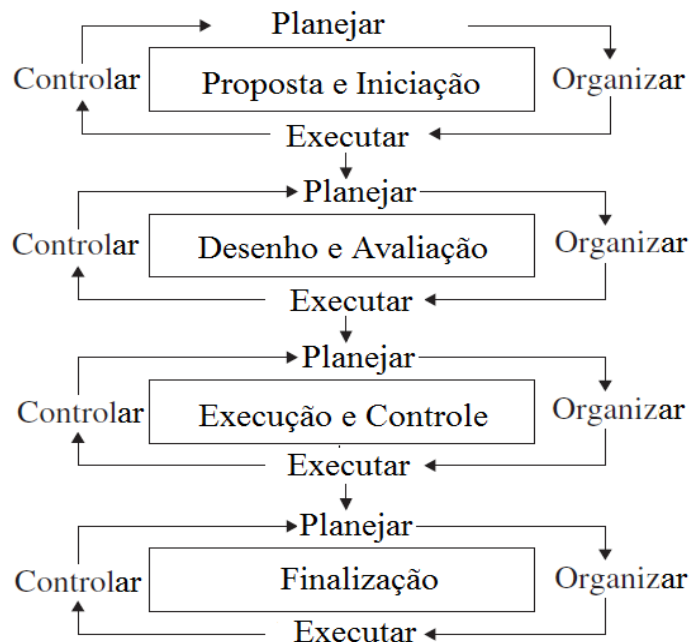
O estágio concepção refere-se ao desenvolvimento do projeto de sistema e apresentação da proposta de custo.

A execução refere-se ao estágio do projeto em que se arquiteta a solução, se planeja as atividades e se executa o trabalho.

O último estágio do ciclo de vida de projetos é o fechamento. Durante as fases finais, a equipe deve manter sua vigilância para garantir que os trabalhos sejam concluídos no tempo e custo esperados.

Em cada um dos cinco estágios listados deve-se planejar o trabalho, organizar os recursos, executar o trabalho e controlar o progresso. Em cada fase do projeto os estágios do processo de gerenciamento são repetidos, conforme demonstrado na Figura 6.

**Figura 6 – Gerenciamento de projeto é gerenciamento fractal**



Fonte: Turner, 2009.

Assim o processo de gerenciamento de projetos possui uma abordagem de gestão fractal, em que cada fase do projeto é quase que um projeto em si. (TURNER, 2009).

### 2.3.1 Gerenciamento de Projetos de *Software*

Segundo Sommerville (2007), o gerenciamento de projetos de *software* envolve maiores desafios em relação ao gerenciamento de outros tipos de projeto. *Software* é um produto intangível, que não pode ser visto ou tocado, logo a visualização do progresso do andamento do projeto deve ser feita por meio de documentação. Não existe padronização nos processos de *software*, os processos variam entre as empresas, o que torna difícil a previsão de problemas de desenvolvimento. As lições aprendidas em projetos anteriores podem não ser aplicadas em novos projetos devido à unicidade de projetos de *software* de grande porte. Os projetos de *software* são geralmente inovadores o que torna comum os atrasos de cronograma.

O Gerenciamento efetivo de desenvolvimento de *software* tem um foco nos 4 Ps: pessoas, produto, processo e projeto. Essa ordem não é arbitrária. O gerente que se esquecer de que o trabalho do engenheiro de *software* consiste em esforço humano nunca terá sucesso no gerenciamento de projeto. Da mesma forma aquele que falha no encorajamento amplo para a comunicação entre os envolvidos, bem cedo, no início da elaboração de um produto, corre o risco de desenvolver uma solução elegante para o problema errado. Um gerente que preste pouca atenção ao processo, arrisca-se a inserir métodos e ferramentas técnicas competentes em um vácuo. Aquele que embarcar sem um plano de projeto sólido compromete o sucesso do projeto. (PRESSMAN, 2011, p.567).

Segundo a teoria dos 4Ps, de Pressman (2011), as pessoas envolvidas no projeto de *software* podem ser divididas em cinco grupos: os gerentes seniores, que definem os itens de negócio; os gerentes de projeto, que tem a função de planejar, motivar e controlar os programadores; os programadores, cuja responsabilidade é desenvolver a engenharia do produto de *software*; os clientes, que tem interesse no produto final e especificam os requisitos; os usuários finais, que utilizam operacionalmente o *software* em ambiente de produção.

O produto, diz respeito ao estabelecimento e delimitação do escopo do produto que não deve ser ambíguo e ser compreensível por todos os grupos de pessoas interessadas do projeto. O escopo do produto considera o contexto de negócio, quais dados são necessários como entrada e quais dados serão produzidos como saída, qual a função do *software* e qual o desempenho desejado. (PRESSMAN, 2011).

Quanto ao processo de desenvolvimento de *software*, este consiste em um roteiro de passos necessários para a criação de um produto de *software* de qualidade dentro do prazo definido. Segundo Sommerville (2007), de um modo geral, o processo de *software* é composto por quatro atividades: especificação, projeto e implementação, validação e evolução. Na fase de especificação, deve-se compreender os objetivos das partes interessadas e efetuar o levantamento das necessidades em relação ao *software*. Na fase de projeto e implementação o *software* deve ser modelado e codificado. Na fase de validação o produto é entregue ao cliente para avaliação. Na fase de evolução, o *software* deve evoluir conforme as necessidades mutáveis do cliente.

Em relação ao projeto, o autor sugere uma abordagem em cinco partes:

1. Compreender o problema e estabelecer expectativas e objetivos realísticos.
2. Manter a velocidade do projeto.
3. Rastrear o andamento do projeto.
4. Tomar decisões efetivas e rápidas.
5. Extrair aprendizados de cada projeto.

Segundo Pressman (2011), o emprego de projetos com planejamento e controle é a única forma de administrar a complexidade do desenvolvimento de *software*.

### 2.3.2 Metodologia de Gerenciamento de Projetos

Metodologia de gerenciamento de projetos pode ser definida como um processo repetitivo que pode ser utilizado em cada projeto. As empresas devem manter e apoiar uma metodologia única para gerenciamento de projetos, de forma a alcançar a excelência ou maturidade de gerenciamento de projetos. As melhores metodologias integram outros processos além dos processos de gestão de projetos, tais como: gestão da qualidade total, engenharia concorrente, controle de mudança de escopo e gestão de riscos. (KERZNER, 2009).

Segundo Kerzner (2009), as características de uma boa metodologia com base em processos integrados incluem: um bom nível de detalhamento; o uso de *templates* (modelos); a utilização de técnicas padronizadas de planejamento, programação e controle de custos; a geração de relatórios padronizados para uso interno e para clientes; flexibilidade para aplicação a todos os projetos; flexibilidade para melhorias

rápidas; facilidade para o cliente a entender e seguir; ser rapidamente aceita e utilizada em toda a empresa; o uso de fases do ciclo de vida padronizado e a realização de revisões no final de cada fase; ser baseada em guias ao invés de políticas e procedimentos; ser baseada na ética de trabalho.

Projetos mais difíceis de gerir são aqueles que envolvem a gestão da mudança. A Figura 7 mostra as quatro entradas necessárias para desenvolver uma metodologia de gerenciamento de projetos. Em cada uma das entradas existe um lado humano que pode exigir mudança nas pessoas.

**Figura 7 – Entradas da Metodologia de Gerenciamento de Projetos.**



Fonte: KERZNER, 2009.

Segundo Kerzner (2009) o desenvolvimento e implantação bem-sucedida de uma metodologia de gerenciamento de projetos requer Identificação das razões mais comuns para a mudança em gestão de projetos, Identificação das formas de ultrapassar a resistência à mudança e aplicação dos princípios de gestão de mudança organizacional para garantir que o ambiente de gerenciamento de projeto desejado será criado e sustentado.

Metodologias não gerenciam projetos; as pessoas gerenciam projetos. É a cultura corporativa que executa a metodologia. A alta administração deve criar uma cultura corporativa que apoia gerenciamento de projetos e demonstrar fé na metodologia. Se isso for feito com sucesso, podem-se esperar os seguintes benefícios:

- Entrega mais rápida através de um melhor controle do escopo do projeto;
- Diminuição do risco total do projeto;
- Melhoria no processo de tomada de decisão;
- Maior satisfação do cliente, o que leva ao aumento de negócios;
- Mais tempo disponível para os esforços de valor agregado, em vez de políticas e concorrência internas. (KERZNER, 2009, p.76).

### 2.3.3 Modelo PMBOK

O guia PMBOK (*Project Management Body Of Knowledge* ou Guia do Conhecimento em Gerenciamento de Projetos), é um conjunto de normas, métodos, processos e boas práticas que tem o objetivo de fornecer diretrizes para o gerenciamento de projetos individuais. Entende-se por boas práticas o conjunto de técnicas, ferramentas e habilidades que quando aplicadas corretamente propiciam um aumento das chances de sucesso de um projeto. O guia foi produzido pelo PMI (*Project Management Institute* ou Instituto de Gerenciamento de Projetos), uma associação para profissionais de gerenciamento de projetos, sem fins lucrativos e fundada em 1969 nos Estados Unidos. (PMI, 2013).

Segundo Saladis e Kerzner (2009) o guia PMBOK proporciona uma base sólida para planejamento de projetos e introduz processos chave e um *framework* para atender ao gerenciamento de projeto. Há muitos pontos de vista e perspectivas sobre o PMBOK e há provavelmente uma quantidade significativa de gerentes de projeto que discordam sobre como ele deve ser usado e sobre seu conteúdo. Segundo os autores, apesar de diferentes pontos de vista, o guia é adição valiosa para qualquer biblioteca de gerentes de projeto.

#### 2.3.3.1 Partes Interessadas

No guia PMBOK (PMI, 2013) as pessoas ou organizações diretamente envolvidas no projeto são chamadas de partes interessadas, cujos interesses são afetados pela execução do projeto e podem exercer influência positiva ou negativa sobre o mesmo. Normalmente as partes interessadas incluem:

- Clientes/Usuários: entidades que farão uso do produto ou serviço, resultado do projeto.
- Patrocinador: pessoa ou organização que proporciona os recursos financeiros para a execução do projeto. O patrocinador também tem a responsabilidade de promover e defender o projeto nos níveis gerenciais da organização.

- Gerentes de Portfólio ou Comitê de Análise de Portfólio: executivos responsáveis pela análise de um conjunto de projetos e seleção dos projetos viáveis à organização, considerando o retorno de investimento e risco.

- Gerente de Programas: gerentes responsáveis pelo gerenciamento de projetos inter-relacionados cujo benefício e controle necessário não é possível através do gerenciamento de cada projeto individualmente.

- Escritório de Projetos ou *Project Management Office* (PMO): entidade organizacional que fornece funções de apoio ou gerenciamento direto de projetos.

- Gerentes de Projetos: pessoas encarregadas de garantir que o projeto atinja seus objetivos, por meio do planejamento das atividades, direcionando o projeto de acordo com o cronograma e orçamento, liderança da equipe de projeto e comunicação com as partes interessadas.

- Equipe do Projeto: equipe composta pelo gerente de projetos e outras pessoas com habilidades específicas que gerenciam e executam os trabalhos do projeto.

- Gerentes Funcionais: gerentes de áreas administrativas ou funcionais da organização que fornecem consultoria à equipe do projeto.

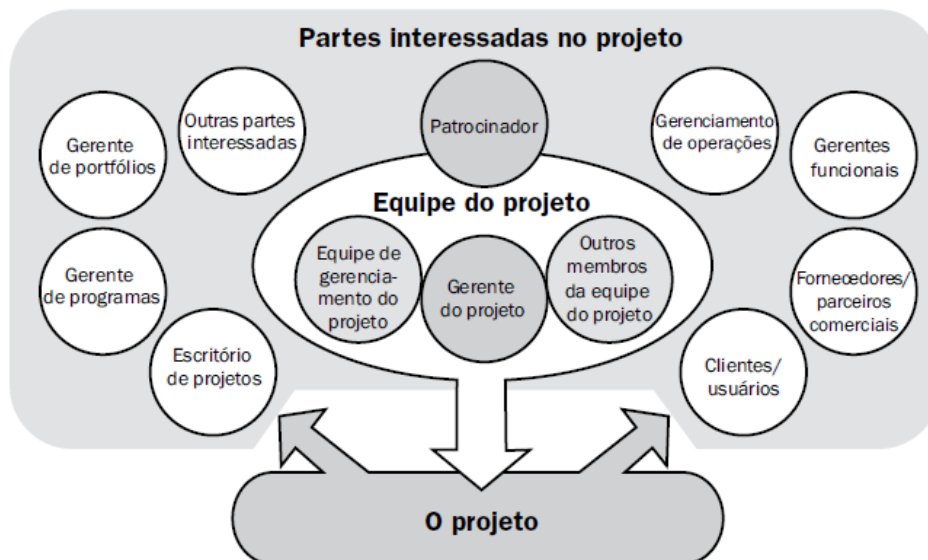
- Gerentes de Operações: gerentes que lidam com áreas diretamente ligadas à produção e manutenção dos produtos ou serviços negociados pela empresa. O gerente de operações incorporará o produto entregue do projeto às operações normais da empresa.

Fornecedores ou Parceiros Comerciais: empresas que fornecem serviços ou componentes necessários ao projeto.

A Figura 8 ilustra a relação entre as partes interessadas e o projeto.



**Figura 8 – Relação entre as partes interessadas e o projeto.**



Fonte: PMI, 2013

### 2.3.3.2 Grupos de Processos

O gerenciamento de projeto, segundo o PMBOK (PMI, 2013), é dividido em processos cuja aplicação e integração proporciona o atendimento aos requisitos do projeto. Os processos de gerenciamento de projeto interagem com os processos da organização, portanto os projetos não devem ser considerados como um sistema fechado e independente, eles recebem dados de fontes internas e externas e entregam funcionalidades à organização. Os processos podem gerar informações tanto para aprimorar o gerenciamento de projetos quanto para aprimorar ativos de processos organizacionais futuros. (MARTINS, 2007).

O Guia PMBOK divide os processos de gerenciamento de projetos em 5 grupos:

- Grupo de processos de iniciação. Os processos executados para definir um novo projeto ou uma nova fase de um projeto existente através da obtenção de autorização para iniciar o projeto ou fase.
- Grupo de processos de planejamento. Os processos necessários para definir o escopo do projeto, refinar os objetivos e definir a linha de ação necessária para alcançar os objetivos para os quais o projeto foi criado.
- Grupo de processos de execução. Os processos realizados para executar o trabalho definido no plano de gerenciamento do projeto para satisfazer as especificações do projeto.

- Grupo de processos de monitoramento e controle. Os processos exigidos para acompanhar, analisar e controlar o progresso e desempenho do projeto, identificar quaisquer áreas nas quais serão necessárias mudanças no plano, e iniciar as mudanças correspondentes.
- Grupo de processos de encerramento. Os processos executados para finalizar todas as atividades de todos os grupos de processos, visando encerrar formalmente o projeto ou fase. (PMI, 2013, p.49).

O guia PMBOK (PMI, 2013) apresenta os processos de gerenciamento de projetos como elementos distintos com interfaces bem definidas, porém, na prática existe a sobreposição e interação entre eles que não é detalhada no guia. E apesar da divisão dos processos em grupos, a aplicação destes é iterativa e alguns processos serão repetidos durante o ciclo de vida do projeto.

### 2.3.3.3 Áreas de Conhecimento

Os processos são também agrupados em 10 áreas de conhecimento, que representam as melhores práticas de gestão em cada área especializada do gerenciamento de projeto. (PMI, 2013).

As áreas de conhecimento são:

#### a) Gerenciamento de recursos humanos do projeto

O gerenciamento dos recursos humanos do projeto inclui os processos necessários para assegurar o melhor desempenho das pessoas envolvidas. Os processos relacionados à gestão dos recursos humanos do projeto são: desenvolver o plano dos recursos humanos, mobilizar a equipe do projeto, desenvolver a equipe do projeto, e gerenciar a equipe do projeto.

O PMBOK (PMI, 2013) estabelece que os papéis e responsabilidades dos membros da equipe sejam designados antes do início dos trabalhos e uma vez definidos os papéis e responsabilidades inicia-se o processo de verificação da disponibilidade desses recursos humanos dentro ou fora da organização, considerando o conhecimento e nível de experiência necessário, o tempo de alocação no projeto e o custo da mão de obra.

O gerenciamento de recursos humanos descrita no guia PMBOK (PMI, 2013) inclui também processos para desenvolvimento da equipe por meio de treinamentos, técnicas motivacionais e através do acompanhamento do desempenho de cada membro, fornecendo *feedback* (análise crítica) e resolvendo conflitos.

Nessa área de conhecimento do PMBOK (PMI, 2013) são considerados também o planejamento com relação à conformidade com normas de órgãos reguladores do ramo de atuação da organização, segurança da informação e impacto do plano de gestão de pessoal sobre a organização.

b) Gerenciamento de integração do projeto

O gerenciamento de integração do projeto inclui processos e atividades para identificação, unificação e coordenação das diversas atividades dentro de cada área de gerenciamento do projeto proposta pelo PMBOK, conforme ilustrado na Figura 9, quando essas atividades se sobrepõem ou interagem entre si de forma a garantir que todo o planejamento de todas as áreas de gerenciamento esteja alinhado a um plano de gerenciamento do projeto mais abrangente. (PMI, 2013).

**Figura 9 – Integração das Áreas de Gerenciamento de Projeto**



Fonte: PMI, 2013

No guia PMBOK a área de gerenciamento de integração estabelece processos que oficializam o início e encerramento dos trabalhos do projeto, processos de controle de todas as atividades independente da área de gerenciamento de projeto da qual façam parte e processos de controle de mudanças de requisitos do produto. Sendo uma área de planejamento macro do projeto é prevista a participação ativa dos papéis do patrocinador, do cliente e demais partes interessadas da alta administração

da empresa nos processos que tangem a essa área de gerenciamento. Principalmente para os processos de mudança de requisitos que preveem a criação de um Comitê de Controle de Mudanças (CCM), um grupo formalmente constituído para avaliar, aprovar ou rejeitar mudanças no projeto, registrar e comunicar tais mudanças mesmo as rejeitadas. (PMI, 2013).

c) Gerenciamento de escopo do projeto

O gerenciamento de escopo é a base do planejamento do projeto e garante que este inclua todo e somente o trabalho necessário para entrega do produto com todas as funcionalidades e características fornecidas pelo cliente e por outras partes interessadas. (SOTILLE; MENEZES; XAVIER; PEREIRA, 2014).

O PMBOK (PMI, 2013) define que os processos relacionados à gestão do escopo do projeto são: planejar o gerenciamento do escopo, coletar os requisitos do projeto, definir o escopo, criar a estrutura analítica do projeto (EAP), validar o escopo, e controlar o escopo.

d) Gerenciamento de tempo do projeto

O gerenciamento do tempo do projeto inclui os processos necessários para assegurar o encerramento do projeto no tempo definido. Os processos relacionados à gestão do tempo do projeto são: definir as atividades, sequenciar as atividades, estimar os recursos das atividades, estimar as durações das atividades, desenvolver o cronograma, e controlar o cronograma.

e) Gerenciamento de custo do projeto

O gerenciamento dos custos do projeto inclui os processos necessários para assegurar que o projeto se encerrará dentro do orçamento. Os processos relacionados à gestão de custos do projeto são: planejar o gerenciamento dos custos, estimar os custos, determinar o orçamento, e controlar os custos.

f) Gerenciamento de qualidade do projeto

O gerenciamento da qualidade do projeto inclui os processos necessários para assegurar que o projeto satisfará as necessidades contratadas. Os processos relacionados à gestão da qualidade do projeto são: planejar o gerenciamento da qualidade, realizar a garantia da qualidade, e realizar o controle da qualidade.

g) Gerenciamento de comunicações do projeto

O gerenciamento das comunicações do projeto inclui os processos necessários para assegurar, no tempo certo, a geração, disseminação e armazenamento das

informações do projeto. Os processos relacionados à gestão das comunicações do projeto são: planejar o gerenciamento das comunicações, gerenciar as comunicações, e controlar as comunicações.

h) Gerenciamento de riscos do projeto

O gerenciamento de riscos do projeto inclui os processos necessários para identificação, análise e controle dos riscos inerentes ao projeto. Os processos relacionados à gestão dos riscos do projeto são: planejar o gerenciamento dos riscos, identificar os riscos, realizar a análise qualitativa dos riscos, realizar a análise quantitativa dos riscos, planejar as respostas aos riscos, e controlar os riscos.

i) Gerenciamento de aquisições do projeto

O gerenciamento das aquisições do projeto inclui os processos necessários para aquisição de bens e serviços fora da organização. Os processos relacionados à gestão das aquisições do projeto são: planejar o gerenciamento das aquisições, conduzir as aquisições, controlar as aquisições, e encerrar as aquisições.

j) Gerenciamento de partes interessadas do projeto

O gerenciamento das partes interessadas do projeto inclui os processos cuja função é identificar e desenvolver estratégias de engajamento de todas as pessoas, grupos ou organizações que podem impactar ou serem impactados pelo projeto. Os processos relacionados à gestão das partes interessadas são: identificar as partes interessadas, planejar o gerenciamento das partes interessadas, gerenciar o engajamento das partes interessadas, e controlar o engajamento das partes interessadas. (PMI, 2013).

Segundo Kerzner (2014), o guia PMBOK ainda é apenas um guia e não o conjunto absoluto de todo o conhecimento sobre gerenciamento de projeto. O desenvolvimento desse conjunto de conhecimentos perfeito deveria ser baseado no tamanho, natureza e complexidade dos projetos de uma organização. Por esse motivo, segundo o autor, embora as empresas tendam a criar metodologias de gerenciamento de projetos baseados no guia PMBOK, essas metodologias raramente o implementam completamente.

#### 2.3.4 Modelo SCRUM

O conceito de desenvolvimento ágil teve início em 2001, quando dezessete especialistas, representantes dos métodos *Extreme Programming* (XP), SCRUM,

*Dynamic Systems Development Method (DSDM)*, *Adaptive Software Development*, *Crystal*, *Feature-Driven Development*, *Pragmatic Programming* e outros métodos, se reuniram para estabelecer o chamado “Manifesto Ágil” que define os princípios compartilhados por todos para desenvolvimento ágil de *software*. Este manifesto seria uma alternativa às metodologias tradicionais baseadas em documentação e com estrutura rígida que torna o desenvolvimento de *software* uma atividade pouco flexível. (AGILLE ALLIANCE, 2001).

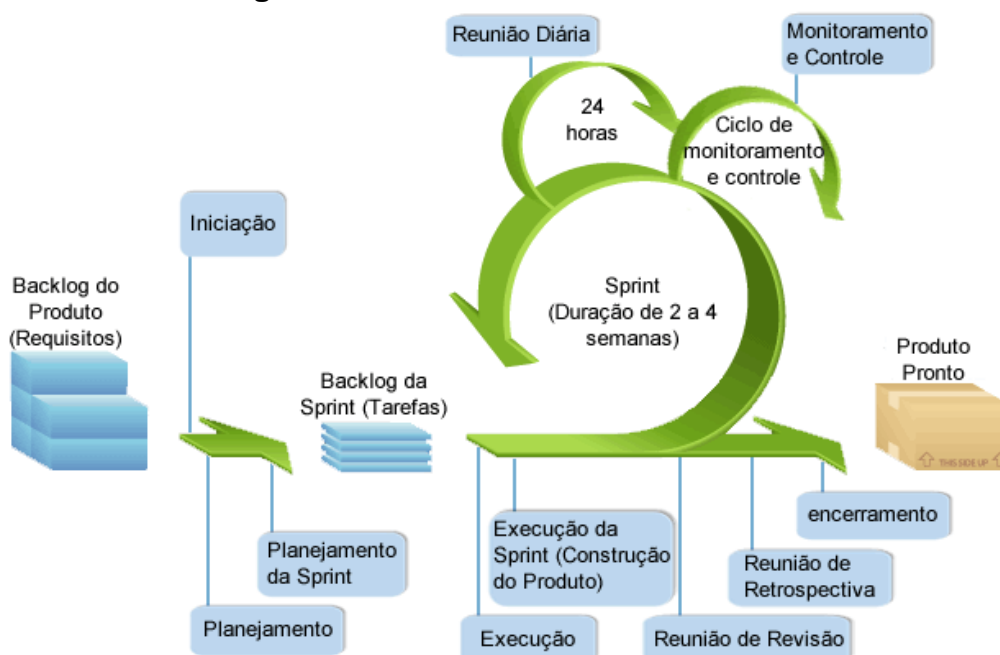
O Manifesto Ágil (AGILLE ALLIANCE, 2001) possui doze princípios:

1. A maior prioridade é satisfazer o cliente através da entrega contínua e adiantada de *software* com valor agregado.
2. Mudanças nos requisitos são bem-vindas no desenvolvimento, mesmo tardiamente. Processos ágeis tiram vantagem das mudanças visando vantagem competitiva para o cliente.
3. Entregar frequentemente *software* funcionando, de poucas semanas a poucos meses, com preferência à menor escala de tempo.
4. Pessoas de negócio e desenvolvedores devem trabalhar diariamente em conjunto por todo o projeto.
5. Construir projetos em torno de indivíduos motivados. Devem ser disponibilizados o ambiente e o suporte necessário e confiar neles para fazer o trabalho.
6. O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é através de conversa face a face.
7. *Software* funcionando é a medida primária de progresso.
8. Os processos ágeis promovem desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente.
9. Contínua atenção à excelência técnica e bom design aumentam a agilidade.
10. Simplicidade – a arte de maximizar a quantidade de trabalho não realizado é essencial.
11. As melhores arquiteturas, requisitos e *designs* emergem de equipes auto organizáveis.
12. Em intervalos regulares, a equipe deve refletir sobre como se tornar mais eficaz e então refinar e ajustar seu comportamento.

O SCRUM é um *framework* de processos usado para gerenciar o desenvolvimento de projetos complexos desde o início da década de 1990. O *framework* SCRUM consiste em equipes e seus papéis, eventos, artefatos e regras. Cada componente dentro do *framework* serve a um propósito específico e é essencial para o sucesso e uso do SCRUM. (SCHWABER; SUTHERLAND, 2013).

Segundo Schwaber (2004) o guia SCRUM baseia todas as suas práticas em um processo iterativo e incremental conforme demonstrado na Figura 10.

**Figura 10 – Processo Base do SCRUM**



Fonte: MARTINS, 2013.

O círculo inferior representa uma iteração das atividades de desenvolvimento que ocorrem uma após a outra. Cada iteração é chamada *Sprint*. A Saída de cada iteração é uma versão de produto.

O círculo superior representa a inspeção diária que ocorre durante a iteração, em que os membros da equipe se reúnem para inspecionar as atividades uns dos outros e torná-las versões adequadas. Conduzir a iteração é uma lista de requisitos. Esse ciclo se repete até que o projeto entregue todos os requisitos. No início de uma iteração, a equipe analisa o que deve fazer. Em seguida, a equipe seleciona o que pode se transformar em uma entrega (incremento de funcionalidade) até o final da iteração, por meio da divisão dos requisitos do produto (*product backlog*) em unidades

menores chamadas estórias do usuário (*user stories*). A equipe conduz o ciclo de interação sem intervenção de gerentes de projeto. No final da iteração, a equipe apresenta o incremento de funcionalidade que foi construído de modo que as partes interessadas possam avaliar o atendimento aos requisitos e as adaptações do projeto possam ser feitas. (SCHWABER, 2004).

O guia SCRUM propõe uma equipe multifuncional e auto organizada. A equipe é multifuncional porque cada membro participa de todo o ciclo de vida do projeto e é auto organizada na medida em que não há um líder que decide a tarefa de cada membro ou como um problema será resolvido. Essas questões são decididas pela equipe como um todo. Na abordagem ágil de desenvolvimento, a equipe SCRUM é formada pelo *product owner* (proprietário do produto), o *scrum master* (mestre SCRUM) e a equipe de desenvolvimento. (COHN, 2010).

O *product owner* representa a organização, clientes ou usuários, e orienta a equipe para construir o produto que atenda aos requisitos. O *scrum master* é uma espécie de treinador para a equipe. A equipe de desenvolvimento são os profissionais que executam os trabalhos para entrega de versões usáveis do produto a cada final de *Sprint*. (COHN, 2010).

No Guia SCRUM (SCHWABER; SUTHERLAND, 2013), o *product owner* é descrito como o responsável pelo gerenciamento do *product backlog* (lista de requisitos para um produto) que inclui expor com clareza os requisitos do projeto, organizar os itens do *product backlog* por prioridade de forma a obter melhor desempenho da equipe de desenvolvimento.

O SCRUM define que o *product backlog* deve ser dinâmico e evolua conforme evolução do produto. Assim as mudanças são previstas e aceitas, não sendo necessário um controle muito rígido sobre estas, entretanto somente o *product owner* tem autorização para incluir as mudanças no *product backlog*. (SCHWABER; SUTHERLAND, 2013).

O time de desenvolvimento é formado por profissionais que fazem o trabalho de entregar um incremento do produto no final de cada iteração. Apenas os membros da equipe de desenvolvimento criam o incremento. As equipes de desenvolvimento são estruturadas e capacitadas pela organização para organizar e gerenciar seu próprio trabalho. (SCHWABER; SUTHERLAND, 2013).



O time de desenvolvimento deve ser auto organizável e multifuncional. A auto-organização permite que o time escolha a melhor forma de executarem os trabalhos sem coordenação de indivíduos ou equipes externas ao time. A característica de multifuncionalidade é quando se possui todas as habilidades necessárias dentro da equipe, para criar o incremento do Produto.

Os membros do time do SCRUM não recebem títulos ou responsabilidades, o único papel existente é desenvolvedor, independentemente do trabalho que está sendo realizado pela pessoa e a responsabilidade pelos trabalhos é atribuída ao time como um todo. (SCHWABER; SUTHERLAND, 2013)

O Guia SCRUM apresenta o *scrum master* como responsável por garantir que o SCRUM é compreendido e instituído, a fim de garantir que a equipe esteja totalmente aderente às práticas e regras do SCRUM. O *scrum master* também orienta as partes interessadas externas à equipe quanto à utilidade de sua interação com os membros da equipe de desenvolvimento, provê recursos para o *product owner* para uma gestão eficaz do *product backlog* e mantém a equipe alinhada aos requisitos do projeto. (SCHWABER; SUTHERLAND, 2013).

O SCRUM salienta a importância do tamanho do time de desenvolvimento, que deve ser entre três e nove integrantes. Esse tamanho é grande o suficiente para permitir a existência de habilidades necessárias para a entrega de um incremento e pequeno o suficiente para facilitar o gerenciamento empírico. (SCHWABER; SUTHERLAND, 2013).

O guia SCRUM, dentro de sua abordagem incremental, prega a divisão do projeto em pequenos incrementos e cada incremento é desenvolvido em um espaço de tempo pré-definido, que é chamado *sprint*, o principal processo do SCRUM. Uma nova *sprint* começa imediatamente após a conclusão da *sprint* anterior. Dentro de uma *sprint* são previstos eventos, que são agendamentos durante esse período de tempo em que devem ser executadas atividades de gerenciamento de projeto. Os eventos são: planejamento da *sprint*, reunião diária, revisão da *sprint* e retrospectiva da *sprint* (SCHWABER; SUTHERLAND, 2013).

Segundo Kniberg (2007), reunião de planejamento da *sprint* é um encontro crítico e um dos mais importantes do ciclo de gerenciamento do SCRUM e tem o objetivo de prover informação à equipe de desenvolvimento para que iniciem a produção de um incremento do produto final e permitir que o *product owner* avalie se

a equipe está preparada para conduzir os trabalhos durante a *sprint*. Durante a reunião de planejamento devem ser definidos escopo, estimativa e prioridades. Os resultados dessa reunião serão: um objetivo de *sprint*, os membros da equipe e a porcentagem de alocação de seu tempo de trabalho, o *sprint backlog* (lista de narrativas contida na *sprint*) e a data de entrega do incremento.

Para estimativa do esforço (tempo) necessário para realização de cada *user story* da *sprint* propõe-se o uso da técnica de *story points* (pontos de estória) que são unidades de medida para expressar o tamanho genérico de uma *user story*, que representa a quantidade de esforço envolvido no desenvolvimento, a complexidade e o risco. A partir desse valor relativo, a equipe SCRUM pode estabelecer uma relação com um valor em horas, dias ou homem-hora. (COHN, 2006).

Segundo Cohn (2006), para levantamento dos pontos de estória, o SCRUM apresenta o método *Planning Poker*, uma espécie de jogo de cartas de baralho que combina opinião de especialistas em uma abordagem lúdica que resulta em uma estimativa rápida e confiável. Os participantes *Planning Poker* incluem todos os desenvolvedores da equipe. O método consiste em distribuir as cartas do baralho entre os participantes e para cada *user story* deve-se separar uma carta com o valor de pontos que se deseja atribuir. Todas as cartas são mostradas ao mesmo tempo e deve haver um consenso sobre o valor final, caso os valores mostrados através das cartas forem muito discrepantes.

Assim a partir da decomposição dos requisitos do produto, e da estimativa de tempo para execução das estórias de usuários é possível estabelecer o número de iterações para entrega de todas as funcionalidades do produto e assim desenvolver o cronograma do projeto. (SCHWABER; SUTHERLAND, 2013).

Para o acompanhamento da execução das *sprints* e o sincronismo de todos os membros da equipe é realizada uma reunião diária, com duração de 15 minutos, em que cada membro da equipe responde a três perguntas: O que você fez nesse projeto desde a última reunião diária? O que você planeja fazer neste projeto entre agora e a próxima reunião diária? Quais dificuldades você encontrou para concluir as suas obrigações nesta *sprint*? (SCHWABER, 2004).

A revisão da *sprint* é realizada ao seu final para inspecionar o incremento e reavaliar o *product backlog*, se necessário. Trata-se de uma reunião informal em que a equipe de desenvolvimento e as partes interessadas discutem sobre o que foi feito

e colaboram uns com os outros para definir as melhorias que podem ser empregadas para o desenvolvimento da próxima *sprint*. O objetivo dessa revisão é proporcionar *feedback* e promover a colaboração. (SCHWABER; SUTHERLAND, 2013).

A retrospectiva da *sprint* é uma oportunidade para a equipe se avaliar e estabelecer um plano de melhoria. É um encontro em que todos têm a permissão de contribuir e discutir erros, dificuldades e sucessos. São discutidos os acontecimentos com relação a pessoas, relacionamentos, processos e recursos, e a partir dessa discussão deve ser criado um plano de implantação de melhorias no processo de trabalho da equipe. A importância dessa reunião é evitar que os mesmos erros sejam cometidos no próximo Sprint. (KNIBERG, 2007).

Os projetos SCRUM requerem menos planejamento do que projetos tradicionais, pois existe visibilidade e transparência durante todo o seu progresso através da apresentação e acompanhamento pelas partes interessadas a cada final de *sprint*. Os projetos SCRUM são também muito complexos e empíricos para serem descritos com muitos detalhes na sua iniciação, em vez disso o foco desse gerenciamento é a monitoração e orientação da equipe para que sejam entregues os melhores resultados possíveis. (SCHWABER, 2004).

### 3 MÉTODO DE PESQUISA

De acordo com Gil (2002), uma pesquisa é um processo formal e sistemático de desenvolvimento do método científico, que tem como objetivo fundamental descobrir respostas para problemas mediante o emprego de procedimentos científicos. Todo planejamento e execução de uma pesquisa tem como subsídio o uso do método, que em pesquisa significa a escolha de procedimentos sistemáticos para a descrição e explicação de fenômenos.

O presente capítulo aborda os aspectos metodológicos que caracterizaram este estudo. Serão evidenciados o tipo de pesquisa e os métodos para coleta e análise de dados.

#### 3.1 TIPO DE PESQUISA

De acordo com Gil (2002) a pesquisa exploratória objetiva a uma maior familiaridade com o problema de forma a construir hipóteses. É uma pesquisa flexível que permite a avaliação de diversos aspectos relacionados ao objeto de estudo.

Estudos exploratórios são investigações de pesquisa empírica cujo objetivo é a formulação de questões ou de um problema, com tripla finalidade: desenvolver hipóteses, aumentar a familiaridade do pesquisador com um ambiente, fato ou fenômeno, para a realização de uma pesquisa futura mais precisa ou modificar e clarificar conceitos. Empregam-se geralmente procedimentos sistemáticos ou para a obtenção de observações empíricas ou para as análises de dados (ou ambas, simultaneamente). Obtêm-se frequentemente descrições tanto quantitativas quanto qualitativas do objeto de estudo, e o investigador deve conceituar as inter-relações entre as propriedades do fenômeno, fato ou ambiente observado. Uma variedade de procedimentos de coleta de dados pode ser utilizada, como entrevista, observação participante, análise de conteúdo etc., para o estudo relativamente intensivo de um pequeno número de unidades, mas geralmente sem o emprego de técnicas probabilísticas de amostragem. Muitas vezes ocorre a manipulação de uma variável independente com a finalidade de descobrir seus efeitos potenciais. (LAKATOS; MARCONI, 2003, p.188).

Ainda de acordo com Gil (2002) as pesquisas exploratórias envolvem: levantamento bibliográfico, entrevista com pessoas que possuem experiência na área de estudo e análise de exemplos que proporcionem compreensão do problema.

A abordagem qualitativa da pesquisa tem o propósito de responder a questões muito particulares que envolvem análise de fenômenos humanos: significados, motivos, aspirações, crenças, valores e atitudes. O ser humano se distingue por agir, pensar e interpretar suas ações no ambiente social. Assim a abordagem qualitativa se emprega em um nível de pesquisa que não pode ou não deve ser quantificado. (MINAYO, 2001).

Foi adotada também a técnica de estudo de caso, que segundo Gil (2002) consiste no estudo profundo de um ou mais objetos permitindo um conhecimento detalhado sobre este afim de que se atinjam os seguintes objetivos:

- Explorar situações reais cujo escopo não está bem definido;
- Preservar o caráter único do objeto em estudo;
- Descrever o contexto em que o estudo está sendo conduzido;
- Elaborar hipóteses e teorias;
- Explicar causas de um fenômeno quando levantamentos e experimentos não são possíveis devido à complexidade do contexto.

Segundo Yin (2005), o estudo de caso permite compreender os acontecimentos da vida real na sua totalidade e globalidade, o que o torna uma importante estratégia metodológica para a pesquisa em ciências humanas. Para tanto o estudo de caso deve ser significativo e completo, na medida em que o limite entre o fenômeno em estudo e seu contexto estejam bem definidos, que as evidências coletadas pelo pesquisador sejam relevantes e que existe uma boa previsão do tempo e recursos necessários para realização do estudo.

### 3.2 COLETA DE DADOS

Segundo Gil (2002), o elemento mais importante para planejamento da pesquisa é o procedimento técnico de coleta de dados. O autor divide esses procedimentos em dois grupos: dados bibliográficos e dados fornecidos por pessoas. O grupo de dados bibliográficos consiste de pesquisa bibliográfica e pesquisa documental. No grupo de dados fornecidos por pessoas estão a pesquisa experimental, a pesquisa *ex-postfacto*, o levantamento e o estudo de caso.

Para a pesquisa exploratória utilizada nesse estudo foi adotada a pesquisa bibliográfica, analisando várias literaturas da área de estudo e artigos publicados na internet.

A pesquisa bibliográfica é desenvolvida com base em material já elaborado, constituído principalmente de livros e artigos científicos. Embora em quase todos os estudos seja exigido algum tipo de trabalho dessa natureza, há pesquisas desenvolvidas exclusivamente a partir de fontes bibliográficas. Boa parte dos estudos exploratórios pode ser definida como pesquisas bibliográficas. As pesquisas sobre ideologias, bem como aquelas que se propõem à análise das diversas posições acerca de um problema, também costumam ser desenvolvidas quase exclusivamente mediante fontes bibliográficas. (GIL, 2002, p.44).

Para realização desse estudo de caso foi executada a análise de toda a documentação gerada pelo departamento de desenvolvimento de *software* global de uma organização do setor de automação bancária, para avaliação do modelo de gerenciamento de projetos utilizado pela organização.

Foram analisados os relatórios de desempenho dos projetos e realizadas entrevistas utilizando-se de roteiros aplicados aos membros da equipe de desenvolvimento e gerentes de projeto para o levantamento e coleta de dados para validação da eficácia do modelo de gestão aplicado.

A entrevista foi usada principalmente para identificar o sentimento da equipe de gerenciamento de projetos e da equipe de desenvolvimento como relação às dificuldades do gerenciamento de equipes distribuídas. Para este fim foi utilizado modelo de entrevista aberta, que segundo Yin (2005), consiste em uma entrevista que visa à extração de fatos e opiniões.

Com o objetivo de coletar a percepção dos gerentes de projeto e membros da equipe com relação aos desafios no gerenciamento de equipes distribuídas foram elaborados dois roteiros de entrevista, apresentados no Apêndice A e no Apêndice B. As questões foram elaboradas de forma a colher informações sobre os tópicos: comunicação, colaboração, confiança, treinamento e reconhecimento no contexto da equipe distribuída. As entrevistas foram conduzidas separadamente com gerentes de projetos e membros da equipe afim de colher essas informações sob as diferentes perspectivas.

As questões 1 a 6 do roteiro de entrevista aplicado a gerente de projetos e as questões 1 a 5 do roteiro de entrevista aplicado aos membros da equipe têm o objetivo de verificar a efetividade do gerenciamento de comunicação e gerenciamento de recursos humanos conduzido pela Organização.

As questões 7 a 11 do roteiro de entrevista aplicado a gerente de projetos e as questões 6 a 7 do roteiro de entrevista aplicado aos membros da equipe visam a identificação da aceitação de cada integrante da equipe conforme seu nível de atuação sobre o conceito de autogerenciamento nas equipes proposta pelo guia SCRUM.

As questões 12 a 15 do roteiro de entrevista aplicado a gerente de projetos e as questões 8 a 10 do roteiro de entrevista aplicado aos membros da equipe objetivam a identificação do nível de satisfação de cada integrante da equipe conforme seu nível de atuação sobre os métodos de treinamento e reconhecimento aplicados pela Organização.

Após a fase de coleta de dados, foi feita a análise dos dados acerca da adequação do modelo de gerenciamento de projetos aplicado pela organização ao DDS.

### 3.3 ANÁLISE DE DADOS

Segundo Lakatos e Marconi (2003), a análise exige a comprovação ou refutação das hipóteses com base na relação dessas com os dados coletados, os quais só possuem significado após interpretação. Assim todos os dados coletados através de pesquisa bibliográfica e estudo de caso foram analisados de forma a estabelecer uma comparação entre a metodologia tradicional representada pelo modelo PMBOK e a metodologia ágil representada pelo SCRUM, identificando as vantagens e desvantagens de cada abordagem com relação à gestão de DDS.

Segundo Bardin (1979) a análise de dados ou análise de conteúdo consiste no uso de técnicas parciais e complementares para explicitação, sistematização e expressão do conteúdo que possibilitem deduções lógicas e justificadas relacionadas à origem dos dados levantados. A análise de conteúdo pode ser definida mais especificamente por:

Um conjunto de técnicas de análise de comunicação visando a obter, por procedimentos sistemáticos e objetivos de descrição do conteúdo das mensagens, indicadores (quantitativos ou não) que permitam a inferência de conhecimentos relativos às condições de produção/recepção destas mensagens". (BARDIN, 1979, p.42).

Apesar da natureza empírica dessa pesquisa, houve embasamento nas três fases cronológicas da análise de conteúdo descritas por Bardin (1979), que são:

1. Pré-análise: fase em que deve escolher os documentos a serem analisados, formular hipóteses e objetivos, elaborar indicadores que servirão de base para a interpretação final.
2. Exploração do material: é a administração sistemática das decisões tomadas. Nesta fase devem ser executadas as atividades de codificação, enumeração dos dados de acordo com as regras estabelecidas.
3. Tratamento dos resultados, inferência e interpretação: fase em que os dados brutos são submetidos a provas estatísticas e testes de validação para se tornarem válidos e significativos. A partir desses dados válidos pode-se se propor inferências e efetuar interpretações de acordo com os objetivos do estudo.

Conforme proposto por Yin (2005) a validação dos dados coletados foi efetuada através da triangulação de dados obtidos de análise documental, entrevistas e descrição dos eventos observados pelo pesquisador. A coleta e análise dos dados ocorreram concomitantemente e para isso foi desenvolvida uma base de dados com todos os documentos coletados, anotações e gravações de áudio das entrevistas realizadas.



## 4 RESULTADOS DA PESQUISA

O estudo de caso foi desenvolvido em uma empresa multinacional de grande porte no setor de automação bancária. Para manter as condições de confidencialidade solicitada pelos entrevistados, esta empresa será referida ao longo deste trabalho como Organização.

### 4.1 CARACTERIZAÇÃO DA ORGANIZAÇÃO ESTUDADA

A Organização é sediada nos Estados Unidos e possui filiais em vários países da América Latina, Europa, Ásia e África, sendo na sua maioria pequenas unidades de assistência técnica dos equipamentos produzidos. Em alguns países existem também unidades fabris e centros de engenharia e desenvolvimento de *software* e *hardware*, além das unidades de controle de qualidade de *software* e *hardware* distribuídos em diversos países.

A Organização iniciou o desenvolvimento distribuído de *software* em 2003 e já possui 35 projetos desenvolvidos globalmente ainda ativos. O uso do DDS foi motivado pela diminuição dos custos através do uso principalmente de mão de obra qualificada e mais barata que a mão de obra disponível em sua sede.

A distribuição do desenvolvimento de *software* ocorreu de forma gradual. A primeira equipe global foi constituída por profissionais dos Estados Unidos e Brasil, país esse que já possuía excelência no desenvolvimento de projetos de *software* locais. As próximas equipes globais incorporaram profissionais da Europa. Após comprovado o sucesso na operação distribuída, a Organização iniciou a inclusão do centro de desenvolvimento da Índia nos projetos de manutenção de *softwares* globais. Com o avanço da maturidade da Organização no desenvolvimento distribuído de *software*, mais equipes distribuídas foram formadas incluindo vários países da Europa, Brasil, Índia e China, incluindo os processos de SQA – *Software Quality Assurance*.

### 4.2 CARACTERIZAÇÃO DOS PRODUTOS DE SOFTWARE DESENVOLVIDOS PELA ORGANIZAÇÃO ESTUDADA

Os produtos de *software* desenvolvidos pela Organização são:

- A camada mais básica de acesso aos periféricos de *hardware*, como *drivers* e interfaces para programação de aplicativos (*application program interfaces* – APIs);

- A camada de *software* que visa a padronização da comunicação com periféricos de autoatendimento (ATM), camada conhecida como XFS – *Extensions for Financial Systems*, conforme requisitos do *Bank Solutions Vendor Council (BSVC)*;
- Sistema de operação dos equipamentos de autoatendimento (ATM) pelo cliente;
- Sistema bancário via internet e dispositivos móveis;
- Sistemas de monitoração de redes de equipamentos de autoatendimento e equipamentos internos às agências bancárias, com acionamento automático de assistência técnica em caso de indisponibilidade de algum equipamento;
- Sistema de diagnóstico de periféricos ATM para uso de técnicos de assistência técnica a fim de auxiliá-los na manutenção dos equipamentos.

A maior parte dos produtos desenvolvidos pela Organização tem o seu conceito originado de análise de mercado pela equipe de marketing com aval técnico da equipe de gerência de produto formada por engenheiros e demais profissionais especialistas em tecnologia da informação.

#### 4.3 CARACTERIZAÇÃO DA METODOLOGIA DE GERENCIAMENTO DE DESENVOLVIMENTO DE SOFTWARE UTILIZADA NA ORGANIZAÇÃO ESTUDADA

A Organização, ao longo de 13 anos de desenvolvimento distribuído, efetuou melhorias em seu modelo de gerenciamento de desenvolvimento de *software*. Iniciou as operações DDS utilizando a metodologia de gestão de projetos locais que se baseava na metodologia tradicional. Diante da dificuldade inerente do DDS, a dificuldade de comunicação, alguns ajustes foram necessários e a documentação que já era extensa ficou ainda maior devido à necessidade de documentação de todas as comunicações, mesmo as informais, de forma a garantir o perfeito entendimento dos requisitos do projeto por toda a equipe e o cumprimento dos prazos e responsabilidades.

A partir de 2013 iniciou-se a implantação de algumas funcionalidades da metodologia SCRUM no desenvolvimento DDS, como o emprego da técnica de *sprint* (iteração) na qual o *software* é dividido em partes menores que devem ser entregues em até 30 dias. Utilizou-se também a técnica de reuniões diárias de curta duração.

A Organização desenvolveu uma coleção de documentos que são guias oficiais para desenvolvimento, documentação, testes e controle que devem ser utilizados e respeitados por todas as equipes globais em todo e qualquer desenvolvimento de *software*. Todas as etapas previstas nos guias devem ser seguidas sem exceções. Os atrasos ou erros de requisitos são tratados com extrema seriedade e burocracia, sujeitos a análises e autorizações em níveis hierárquicos superiores, o que causa um enorme constrangimento, caracterizando uma forma de evitá-los já que causam um imenso transtorno e custo, pois envolvem uma série de equipes dispersas com fusos horários diferentes e agendas diferentes.

Toda a definição dos processos de gestão de projetos é centralizada na sede da empresa, que é responsável pelas revisões da documentação, treinamento e distribuição para toda a corporação. Nenhuma outra unidade da corporação participa da definição e melhorias dos processos de gestão de desenvolvimento. Já os papéis de gerentes de projetos são distribuídos entre as equipes globais conforme necessidade do projeto.

#### 4.4 ANÁLISE DESCRITIVA DA AMOSTRA

Neste capítulo, serão apresentados os resultados da análise descritiva dos dados coletados por meio de entrevistas conduzidas com profissionais da Organização que fazem parte de equipes de desenvolvimento distribuído de *software*.

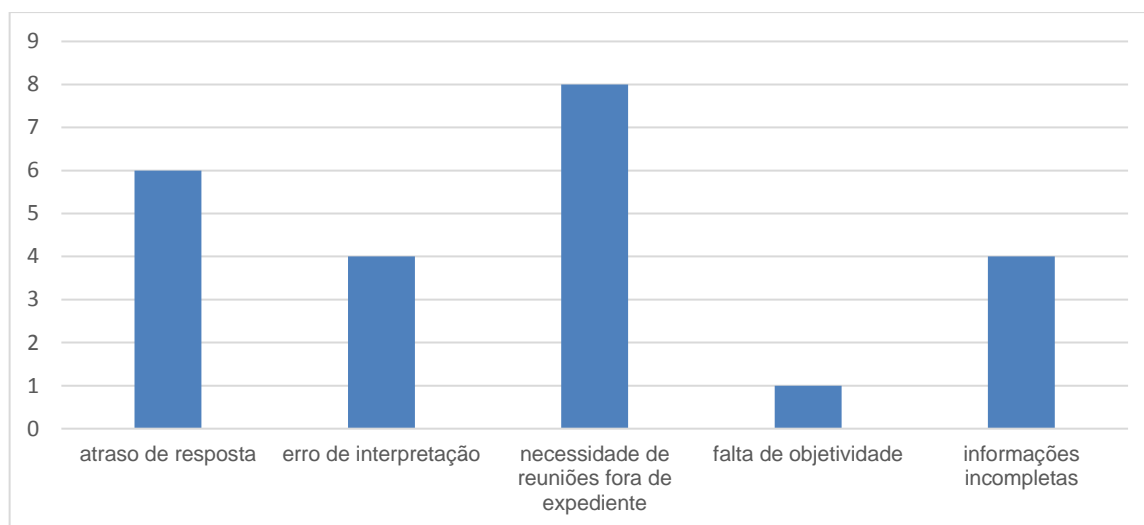
Os resultados da análise serão apresentados considerando os assuntos: comunicação, colaboração, confiança, treinamento e reconhecimento no contexto da equipe distribuída, tratados no roteiro de entrevista conforme descrito no capítulo 3 deste trabalho.

##### 4.4.1 Análise dos dados de entrevista com Gerentes de Projeto

Foram entrevistados oito gerentes de projeto sendo sete profissionais locados no Brasil e um profissional locado nos Estados Unidos. Todos os profissionais entrevistados possuem experiência em gerenciamento de um número superior a cinco projetos globais de desenvolvimento de *software*.

Os dados das entrevistas mostraram que em relação ao fluxo de comunicação entre os membros das equipes distribuídas existem problemas que devem ser gerenciados. Os problemas de comunicação mais mencionados durante as entrevistas com os gerentes de projeto foram: necessidade de reuniões fora de expediente, atraso de resposta, erro de interpretação, informações incompletas e falta de objetividade, conforme ilustrado na Figura 11.

**Figura 11 Problemas de Comunicação**

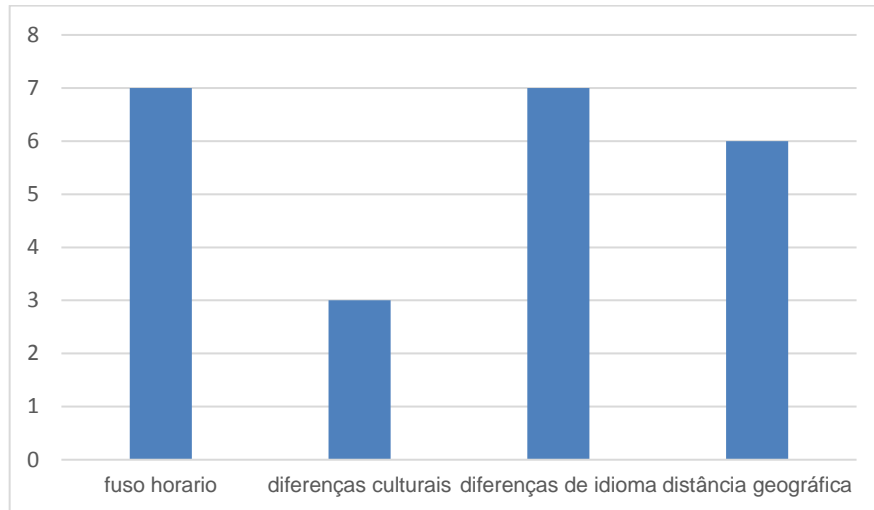


Fonte: Dados da Pesquisa

Segundo informação dos entrevistados, a causa da necessidade de reuniões frequentes em horários fora do expediente de trabalho são as diferenças de fuso horário entre os países participantes dos projetos, não sendo possível a sincronização da disponibilidade de todas as partes interessadas para agendamento de reuniões de acompanhamento. A diferença de fuso horário é também a principal causa dos atrasos de resposta na comunicação.

Os demais problemas indicados têm como causa a diferença de idioma seguida pela distância geográfica e diferenças culturais conforme ilustra a Figura 12.

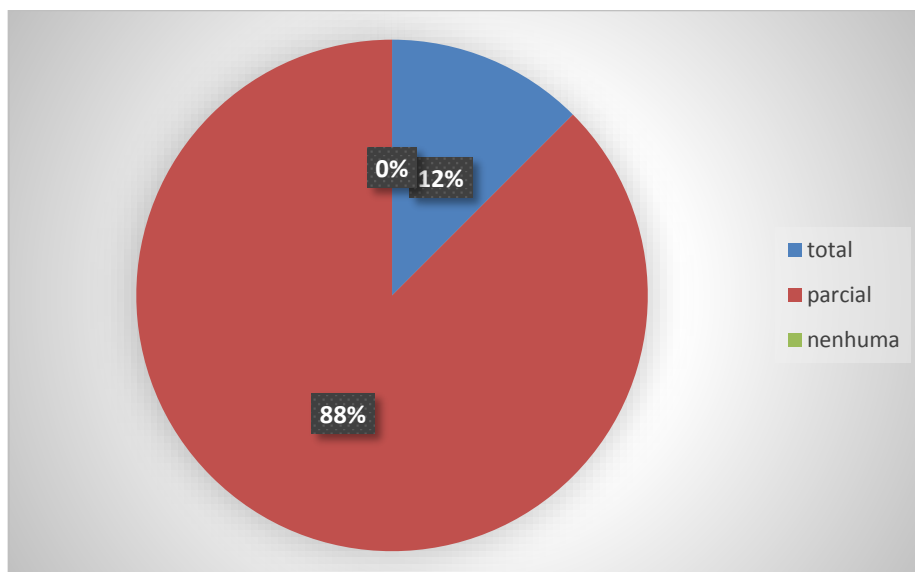
**Figura 12 – Causas de Problemas de Comunicação**



Fonte: Dados da Pesquisa

No que diz respeito à colaboração entre as equipes distribuídas, a Figura 13 demonstra que a maioria dos gerentes de projeto tem a percepção de uma colaboração parcial entre as equipes, que consiste em atender às solicitações e responder às eventuais dúvidas quando houver disponibilidade de tempo, ainda que existe urgência na solicitação. A colaboração total que se traduz na disponibilidade em ajudar a resolver o problema no momento em que o mesmo se apresenta, só é percebido pelos gerentes de projeto entre os membros de equipes co-localizadas.

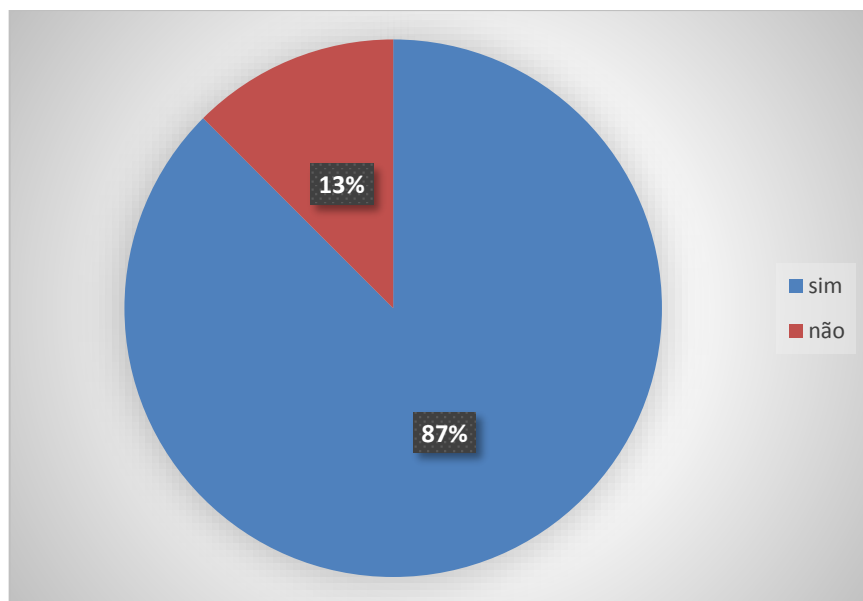
**Figura 13 – Colaboração entre Equipes Distribuídas**



Fonte: Dados da Pesquisa

Apesar de todos os entrevistados concordarem que a distância geográfica influencia negativamente a colaboração entre os membros, existe também a percepção de que as distâncias culturais também exercem influência negativa sobre a possibilidade de colaboração Inter equipes. Conforme ilustrado na Figura 14, é possível verificar que 87% dos entrevistados acreditam que as diferenças de cultura e costumes entre os países criam barreiras para essa colaboração.

**Figura 14 – Influência Negativa dos Aspectos Culturais na Colaboração**

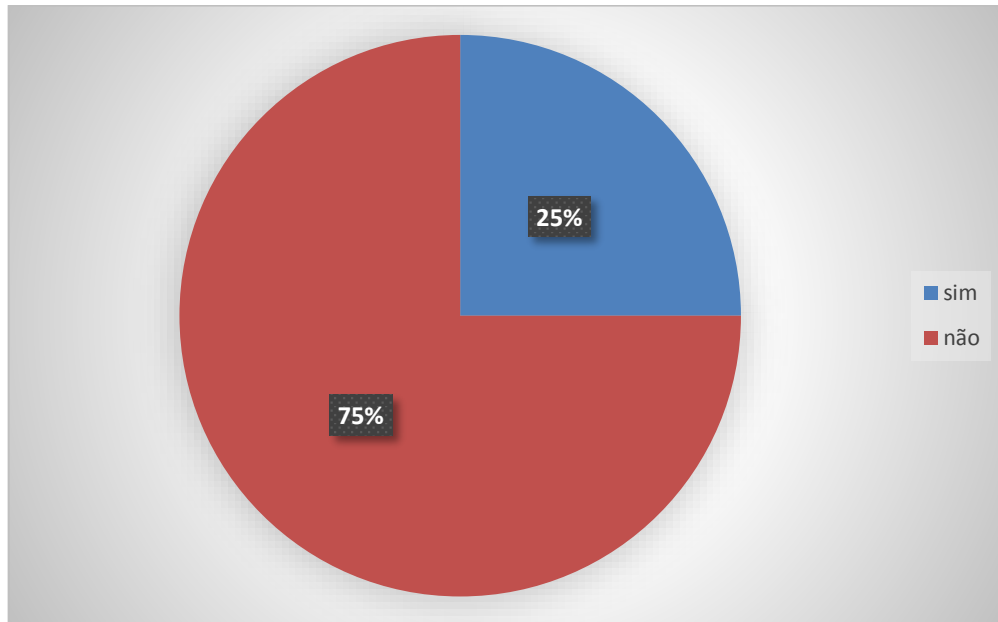


Fonte: Dados da Pesquisa

Diante dos problemas de comunicação e colaboração identificados, a maioria dos entrevistados, 75% conforme demonstrado na Figura 15, acreditam que não há possibilidade de estabelecer autogerenciamento nas equipes distribuídas.

Acrescentam, entretanto, que a forma de gerenciamento de projetos aplicada pela Organização não demanda do gerente de projeto a necessidade de muita supervisão dos trabalhos da equipe, sendo verificado um bom nível de entendimento e comprometimento dos membros da equipe com os trabalhos a serem desenvolvidos e com os prazos de entrega.

**Figura 15 – Possibilidade de Autogerenciamento da Equipe**

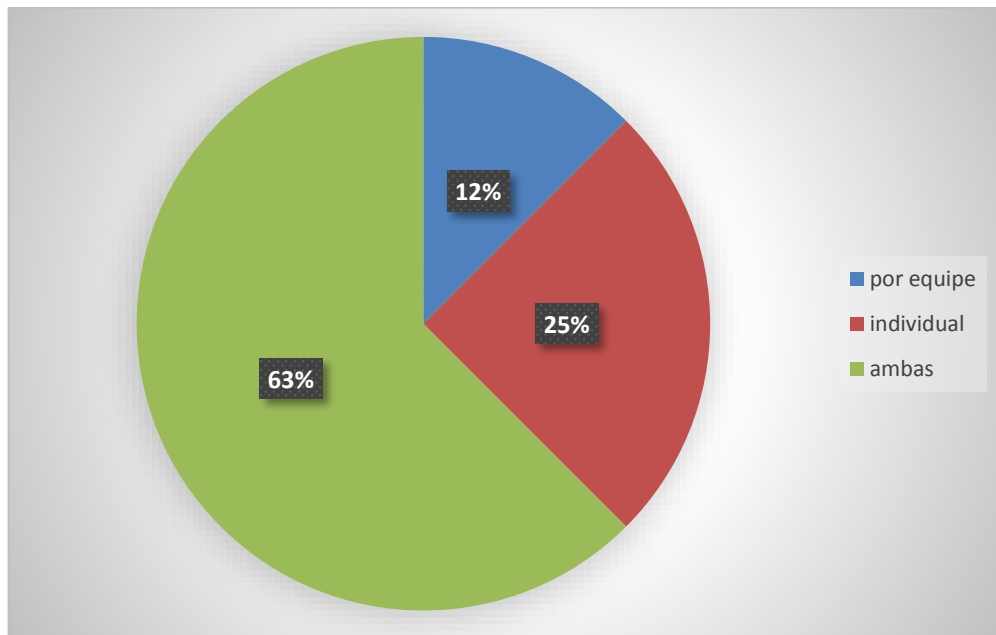


Fonte: Dados da Pesquisa

Além disso, os dados das entrevistas demonstram que é possível para o gerente de projetos organizar e controlar a equipe remotamente com eficiência, já que as visitas do gerente de projetos às locações das equipes são pouco necessárias. A média de visitas é de aproximadamente 0,8 vezes durante um projeto.

Quanto à efetividade dos treinamentos de processos de gerenciamento de projeto ministrados na Organização, 100% dos entrevistados concordam que os treinamentos são efetivos para garantir a utilização do método de gerenciamento de projetos e a padronização do desenvolvimento desenvolvido de *software*.

Quando questionados sobre a preferência pela forma de avaliação de desempenho, 63% dos entrevistados responderam que gostariam que houvesse dois tipos de avaliação: individual e por equipe. Os resultados estão apresentados na Figura 16.

**Figura 16 – Preferência por Tipo de Avaliação de Desempenho**

Fonte: Dados da Pesquisa

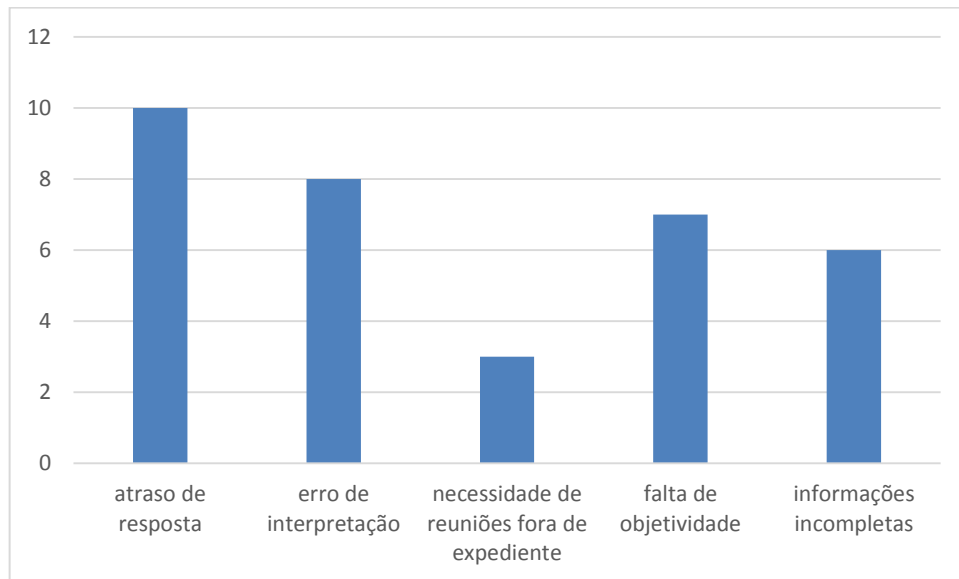
A justificativa está em que a avaliação individual permitiria identificar talentos específicos que auxiliariam o profissional a alcançar cargos de liderança e a avaliação por equipe permitiria a equalização do mérito e consequentemente a melhoria da integração do grupo.

#### 4.4.2 Análise dos dados de entrevista com Membros da Equipe

Foram entrevistados dez desenvolvedores de *software*, sendo todos os profissionais locados no Brasil. Todos os profissionais entrevistados participaram de um número superior a dois projetos globais de desenvolvimento de *software*.

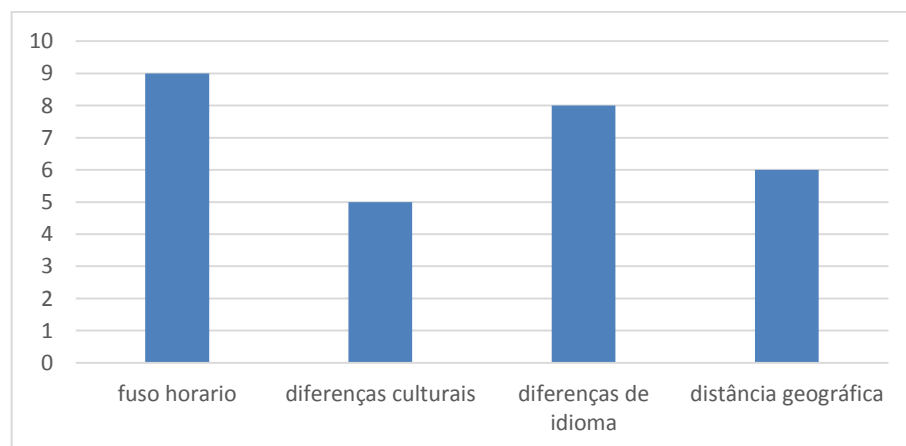
Quando questionados sobre os problemas de comunicação encontrados durante os projetos globais, a resposta mais recorrente é “atraso de resposta”, conforme apresentado na Figura 17. Segundo os entrevistados, é comum durante o projeto que uma equipe solicite informações sobre componentes de *software* desenvolvidos por outras equipes de forma a conduzir suas atividades. A demora na entrega dessas informações pode paralisar as atividades.



**Figura 17 – Problemas de Comunicação**

Fonte: Dados da Pesquisa

O gráfico da Figura 18 demonstra que a causa dos problemas de comunicação mais apontada pelos entrevistados é a diferença de fuso horário entre as localidades das equipes. Ainda segundo informações dos entrevistados a diferença de fuso horário causa atraso na entrega da solicitação e no recebimento da resposta via e-mail e causa também dificuldades para comunicação telefônicas não previamente agendadas.

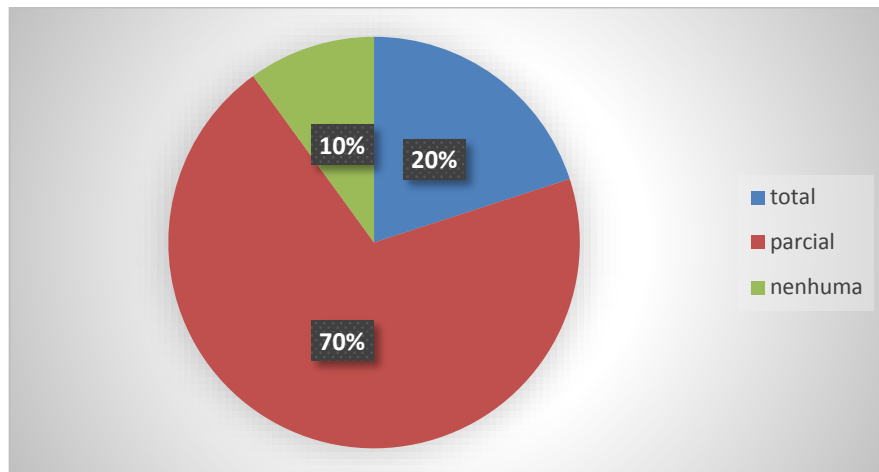
**Figura 18 – Causas de Problemas de Comunicação**

Fonte: Dados da Pesquisa

A percepção sobre colaboração entre equipes dos desenvolvedores difere da percepção dos gerentes de projeto no que diz respeito a falta de colaboração.

Conforme dados apresentados na Figura 19, cerca de 10% dos desenvolvedores entrevistados já experimentaram em algum momento durante o projeto nenhuma colaboração de membros de outras equipes.

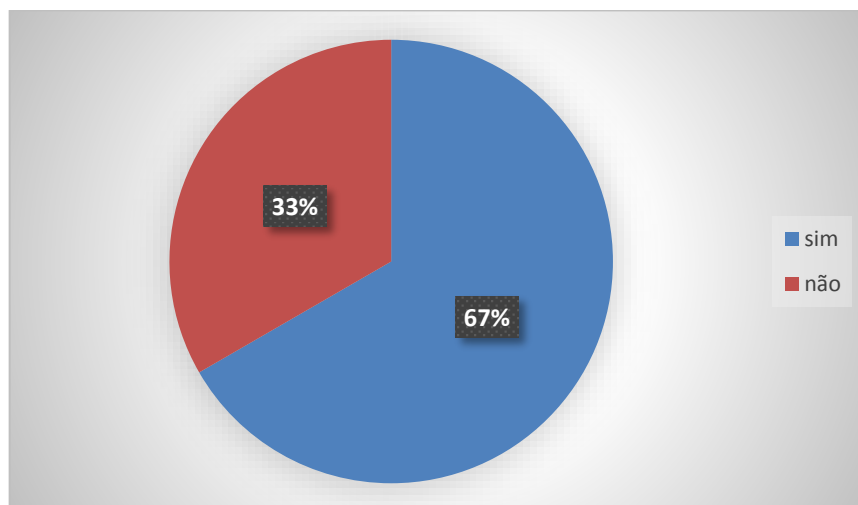
**Figura 19 – Colaboração entre Equipes Distribuídas**



Fonte: Dados da Pesquisa

A maior parte dos desenvolvedores entrevistados concorda com os gerentes de projeto que os aspectos culturais influenciam negativamente na colaboração entre equipes, conforme apresentado na Figura 20.

**Figura 20 – Influência Negativa dos Aspectos Culturais na Colaboração**

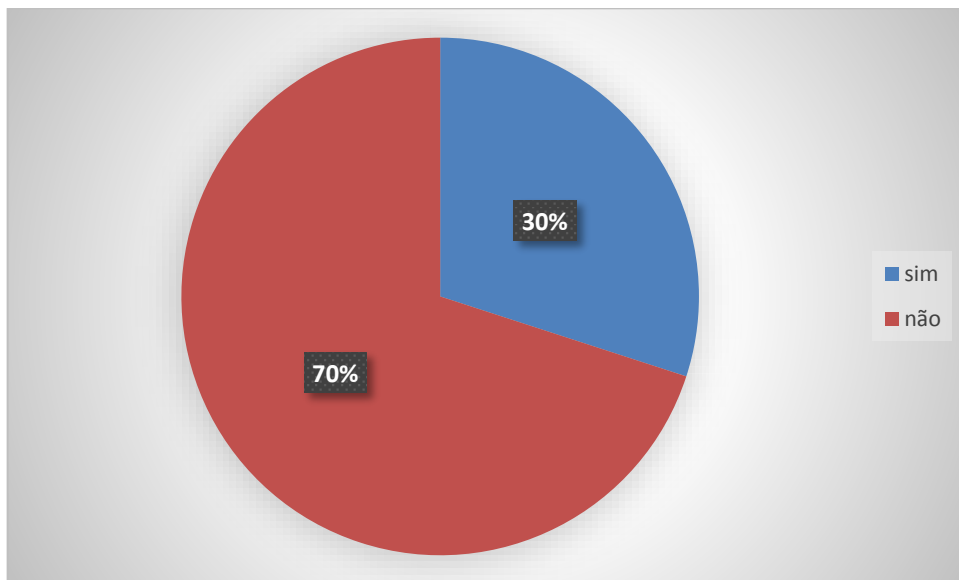


Fonte: Dados da Pesquisa

A maioria dos desenvolvedores mencionou a sua percepção de que embora todos prezem pelo bom andamento do projeto, existem equipes de alguns países que se mostram mais colaborativas que equipes de outros países.

As diferenças culturais, distância geográfica e diferença de fuso horário foram os motivos apresentados por 70% dos entrevistados que acreditam que o autogerenciamento nas equipes distribuídas não seria eficiente, conforme exibido na Figura 21.

**Figura 21 – Possibilidade de Autogerenciamento da Equipe**

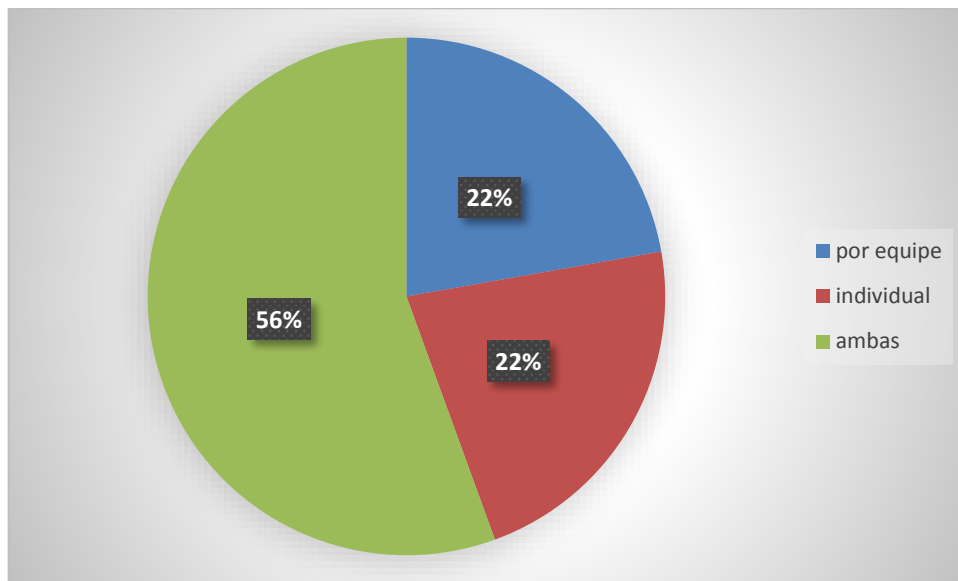


Fonte: Dados da Pesquisa

Entretanto, 30% dos entrevistados responderam que já existe certo nível de autogerenciamento na equipe que administra seu tempo para entrega dos incrementos sem supervisão constante do gerente de projeto ou líder técnico da equipe.

Quando questionados sobre a efetividade do treinamento de processos de gerenciamento de projetos na Organização, 100% dos entrevistados responderam serem efetivos.

A maioria dos desenvolvedores entrevistados assim como os gerentes de projeto gostaria de ser avaliada individualmente e por equipe, conforme demonstrado na Figura 22.

**Figura 22 – Preferência por Tipo de Avaliação de Desempenho**

Fonte: Dados da Pesquisa

As justificativas são as mesmas apresentadas pelos gerentes de projeto, a necessidade de reconhecimento individual para alçar cargos de liderança e reconhecimento por equipe para desenvolver um senso de time e melhorar o ambiente de trabalho.

#### 4.5 ELEMENTOS DE ANÁLISE

Para a análise dos dados coletados foram consideradas as dez áreas de conhecimento do PMBOK que representam as áreas especializadas do gerenciamento de projeto: integração, escopo, tempo, custos, qualidade, recursos humanos, comunicações, riscos, aquisições, e partes interessadas.

Considerando que a principal diferença entre projetos de desenvolvimento distribuído de *software* (DDS) e projetos de desenvolvimento local de *software* está na organização e gerenciamento da equipe, optou-se por iniciar a análise de dados desse estudo pela área de gerenciamento de recursos humanos.

##### 4.5.1 Gerenciamento de Recursos Humanos

O gerenciamento de recursos humanos do projeto inclui as atividades relacionadas à organização e gerenciamento da equipe do projeto e tem como objetivo

principal otimizar o uso da capacidade profissional dos membros da equipe. (SANTOS, 2009).

A área de gerenciamento de recursos humanos é abordada tanto no guia PMBOK quanto no guia SCRUM, ambos possuem processos de desenvolvimento do plano de recursos humanos, gerenciamento e desenvolvimento da equipe. Entretanto as práticas propostas pelo PMBOK estabelecem definição e controle mais formais sobre os papéis, responsabilidades e desempenho de cada membro da equipe.

A proposta do PMBOK, conforme PMI (2013) que prevê definição prévia dos papéis e responsabilidades de cada membro da equipe direciona as atividades de desenvolvimento da equipe, por meio de treinamentos e mentoria, de acordo com a especialidade de cada membro.

O guia SCRUM (SCHWABER; SUTHERLAND, 2013), por outro lado, apresenta uma abordagem de gerenciamento de recursos humanos mais flexível e adaptativa. A colaboração e confiança entre os membros da equipe são requisitos essenciais para a proposta de equipe auto organizável.

Quanto aos processos de reconhecimento e recompensa, no *framework* SCRUM, em contraponto ao PMBOK são considerados os trabalhos da equipe como um todo. (SCHWABER; SUTHERLAND, 2013).

O ambiente DDS é tratado somente no PMBOK (PMI, 2013), sendo referenciado como equipe virtual, e este não propõe nenhuma atividade ou técnica diferenciada para gerenciamento dessas equipes. O PMBOK destaca a importância de ferramentas colaborativas, tais como videoconferências, para facilitar a comunicação da equipe e aponta que cabe ao gerente de projetos, em uma equipe virtual, lidar com as diferenças culturais, idiomáticas, de fusos horários e de condições locais.

Na Organização verificou-se que o modelo PMBOK se mostrou mais aderente ao DDS, conforme entrevistas realizadas com oito gerentes de projeto da área de GSD – *Global Software Development* da empresa. A distância geográfica entre os integrantes da equipe não propiciava o espírito de time necessário ao estabelecimento da prática de equipe auto gerenciável do SCRUM. O resultado das entrevistas evidenciou que as características culturais também se tornaram empecilho ao uso de uma abordagem mais flexível no gerenciamento de recursos humanos, como, por

exemplo, a cultura machista de países como a Índia dificultava a colaboração entre homens e mulheres na equipe.

Além disso, a preocupação de profissionais de países desenvolvidos, como os Estados Unidos, com a perda do emprego para profissionais de países em desenvolvimento como o Brasil e Índia, que possuem salários menores, também é causa de falta de colaboração entre os membros de equipes.

O modelo PMBOK com designação de lideranças, documentação dos papéis e responsabilidades facilitou o cumprimento de metas de prazos e custos.

Os papéis da equipe do projeto de desenvolvimento de *software*, na Organização, são divididos da seguinte forma:

- Gerente de Produto: é responsável pelo acompanhamento de todo o ciclo de vida do desenvolvimento do produto de *software*. O gerente de produto lidera projetos, monta e fornece suporte as equipes de projetos, garante escalação e resolução de questões críticas e autoriza as mudanças de requisitos.

- Gerente de Projeto: é responsável pela coordenação dos trabalhos das equipes de desenvolvimento e pela interface entre os grupos de desenvolvimento e outras partes interessadas. O gerente de projeto cria o plano do projeto e monitora e controla o projeto.

- Líder Técnico: é o representante de cada equipe de desenvolvimento. Participa do desenvolvimento das revisões de código e faz a interface entre os membros da equipe e outras equipes.

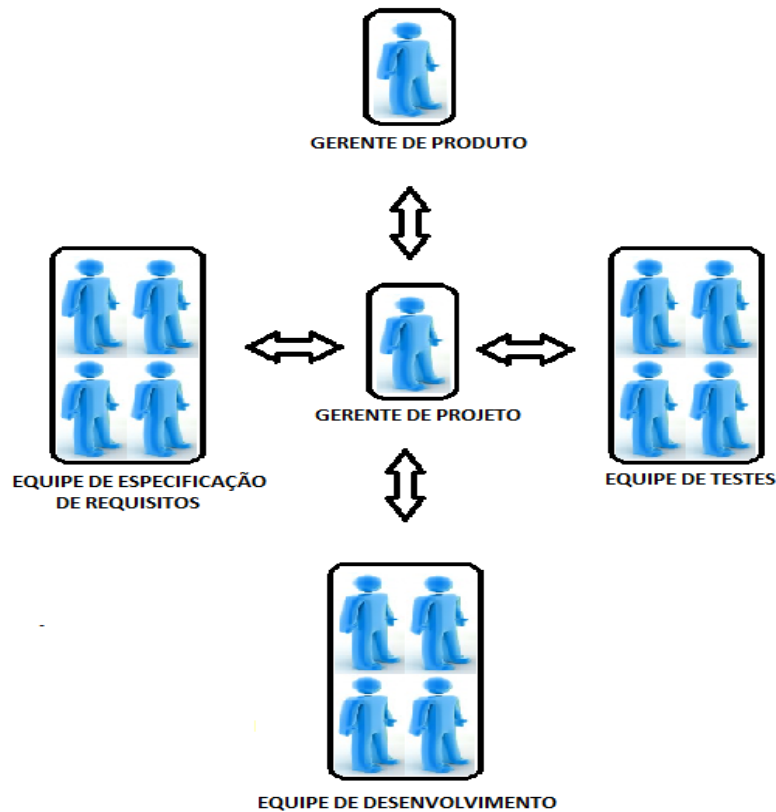
- Desenvolvedores: responsável pelo desenvolvimento (especificação de requisitos, codificação e testes unitários) do *software*. O desenvolvedor pode também ser um líder de equipe.

- Líder de Teste: é responsável pela qualidade global dos produtos de *software* desenvolvidos. Os deveres do líder de teste incluem: garantir que os processos e procedimentos de testes estabelecidos sejam seguidos e a alocação de recursos de teste para os diversos projetos;

- Testador: é responsável pela criação, execução e documentação de testes de verificação e validação, bem como relatar os resultados do teste e progresso de teste.

Quanto à distribuição dos membros da equipe, embora exista no DDS a possibilidade de equipes com indivíduos distribuídos, na empresa em estudo optou-se pela estrutura de equipes distribuídas conforme demonstrado na Figura 23.

**Figura 23 – Distribuição das Equipes do Projeto**



Fonte: Dados da Pesquisa

O motivo principal dessa arquitetura está relacionado ao custo do processo de desenvolvimento. No setor de automação bancária o desenvolvimento de *software* está relacionado a dispositivos de *hardware* muito específicos tais como dispensadores de dinheiro, leitores de cartões magnéticos, impressoras, leitores de código de barras, entre outros, normalmente produzidos pela própria empresa. Assim, durante o desenvolvimento é necessário o uso desses dispositivos para validar o funcionamento do *software*. Esse fato faz com que seja mais econômico centralizar o desenvolvimento de um componente de *software* em uma só localidade para disponibilizar um só protótipo de dispositivo de hardware.

Outro motivo dessa estrutura de distribuição da equipe é a centralização da especificação dos requisitos do produto de *software* na matriz da empresa. A cada projeto é designada uma equipe de especificação de requisitos normalmente formada por engenheiros de hardware e *software* da área de pesquisa de desenvolvimento da empresa e representantes de clientes.

A centralização da equipe de testes segue o princípio da redução de custos, já que essa equipe também depende de dispositivos de *hardware* para testes e de maquinário com tecnologia avançada para automatização dos testes.

O tamanho da equipe de desenvolvimento segue a proposta do SCRUM, que define que o tamanho ideal de uma equipe é entre três e nove membros, a fim de garantir interação, produtividade e flexibilidade necessárias.

A prática de reconhecimento e recompensa por mérito da equipe proposta pelo SCRUM foi adotada pela empresa ao final de cada projeto de forma informal. As entrevistas com gerentes funcionais dos indivíduos das equipes mostraram que, independente dos projetos, as áreas funcionais da empresa devem se adaptar às políticas da organização que estabelece avaliação individual através de sistema integrado. Além disso, a pesquisa demonstrou que os indivíduos da equipe possuíam ambições diferentes com relação a sua carreira por isso precisavam ser avaliados individualmente.

O processo de desenvolvimento de equipe também é previsto pela Organização que mantém no site de colaboração global uma série de guias de treinamento de todos os seus processos de gerenciamento de projeto, manuais de treinamento de produtos de *hardware* e *software* utilizados nos projetos, além de vídeo-aulas e programas de treinamento em *MS-PowerPoint* disponibilizados para todos os membros de equipes de desenvolvimento global.

As entrevistas realizadas com os gerentes de projetos e membros de equipe demonstraram que os treinamentos são eficientes e atendem aos objetivos propostos.

#### 4.5.2 Gerenciamento de Integração

A área de gerenciamento de integração inclui atividades de análises e revisões de todas as demais áreas de gerenciamento visando melhoria de desempenho para atendimento aos objetivos e incluem atividades de controle de mudanças de escopo do projeto que causam impacto em todas as áreas de gerenciamento. (PMI, 2013).

Os processos de abertura e encerramento do projeto, desenvolvimento do plano de gerenciamento do projeto e o acompanhamento da execução do projeto propostos pelo PMBOK também podem ser identificados no SCRUM. Entretanto, no SCRUM o gerenciamento de integração, cuja responsabilidade é do *Product Owner*,



está limitado às extremidades da iteração. Dentro da iteração os trabalhos são gerenciados pela própria equipe.

A principal diferença da área de gerenciamento de integração entre o PMBOK e SCRUM está no processo de controle de mudança de requisitos. Enquanto o PMBOK prevê um controle formal de mudanças com formação de um comitê responsável pela avaliação e aprovação de todas as mudanças propostas, o SCRUM entende que as mudanças fazem parte do processo evolutivo do projeto, sendo facilmente absorvidas pelo conjunto de requisitos do projeto (*product backlog*).

Na Organização em estudo, as atividades de gerenciamento de integração de projetos relacionados à abertura e encerramento dos projetos globais são normalmente realizadas pela gerência de produto com aprovação da alta direção da empresa. Se o objetivo do projeto for relacionado à proposição de valor, a gerência de produto trabalhará em conjunto com a área de marketing da empresa que, utilizando-se de análise de mercado, apresentarão a proposta de novos produtos de software. Se o objetivo do projeto for relacionado à correção ou melhoria de um produto já existente, somente a gerência de produto será responsável pela abertura e encerramento do projeto.

As atividades de monitoração e controle do projeto são realizadas pelo gerente de produto e pela equipe de PMO (*Project Management Office*). A função do gerente de produto é garantir que o produto de *software* seja entregue no prazo e custo planejados e com os requisitos e qualidade especificados. A equipe de PMO, por sua vez, tem a função de controlar e monitorar o desempenho dos processos de gerenciamento do projeto e efetuar as alterações necessárias a fim de maximizar a utilização dos recursos e reduzir custos.

Para o desenvolvimento distribuído de *software*, a Organização executa as atividades de gerenciamento de integração de projetos que são comuns aos *frameworks* PMBOK e SCRUM, com exceção do processo de controle de mudança de projeto para o qual optou-se pela abordagem mais rígida do PMBOK.

A Organização definiu o CCB (*Change Control Board* ou Comitê de Controle de Mudanças) que consiste em um comitê administrativo que existe durante todo o ciclo de vida do projeto. Esse comitê prevê um fórum para revisão e disposição sobre as mudanças propostas para os requisitos, documentação e para o software. Para discussão e aprovação das mudanças de requisitos o comitê considera os fatores

criticidade e repetitividade. O fator criticidade refere-se às consequências da mudança proposta, ou seja, no caso de não aprovação da mudança quais as consequências esperadas. A partir desse estudo aplica-se um valor a esse fator que pode ser alta, média ou baixa. O fator repetitividade diz respeito à frequência de repetição do problema encontrado que requer a mudança, que também é avaliada em alta, média e baixa.

Ao contrário do que prevê o SCRUM, na Organização as mudanças não são bem vistas. As mudanças são normalmente sinais de falha na especificação dos requisitos. A cada final de projeto, os gerentes de projeto devem preencher um relatório de análise de projeto que, entre outras informações, deve possuir a estatística de mudanças de requisitos do projeto conforme ilustrado no Quadro 1.

**Quadro 1 – Estatística de Mudança de Requisitos**

<b>Descrição</b>	<b>Número</b>
Requisitos na Aprovação Inicial (Requisitos Basicos) (B)	
Requisitos Adicionados depois da Aprovação Inicial (C1)	
Requisitos Retirados depois da Aprovação Inicial (C2)	
Requisitos Modificados depois da Aprovação Inicial (C3)	
Total de Mudanças de Requisito (C) = (C1+C2+C3)	
Desvio de Escopo % = $(C/B)*100$	

Fonte: Dados da pesquisa

O índice de mudança de requisitos, que consiste no percentual de mudanças de requisitos em relação aos requisitos básicos, na Organização é tratado como desvio de escopo. Este índice de desvio é usado para estabelecer melhorias no processo de levantamento de requisitos e para atualização da base de lições aprendidas que serão usadas pelos gerentes de projeto para estimativa de recursos e tempo para os projetos de novas versões do produto ou para projetos de produtos similares.

#### 4.5.3 Gerenciamento de Escopo

Ambos os guias PMBOK e SCRUM se valem da definição de Sotille *et al* (2014), de que o gerenciamento de escopo é base para o gerenciamento das demais áreas e apresentam processos e técnicas para planejamento e acompanhamento criteriosos do escopo do projeto, a diferença entre as propostas está no controle dos requisitos do produto.

O PMBOK define que os requisitos do produto sejam obtidos e registrados com detalhes suficientes antes da iniciação do projeto, pois o planejamento de custo, cronograma, qualidade e de aquisições são todos construídos com base nestes requisitos. Já o SCRUM consegue estabelecer um nível de independência dessa área de gerenciamento mantendo o tempo e custo do projeto fixo, permitindo que os requisitos do produto sejam refinados ao longo do projeto.

Quanto às entregas do projeto, o PMBOK também prevê a divisão do projeto em incrementos que acrescentam sucessivamente funcionalidades ao produto tal como é feito no SCRUM, mas não estabelece um limite de tempo para os trabalhos em cada incremento.

A Organização mescla os processos de gerenciamento de escopo do PMBOK e do SCRUM no gerenciamento de projetos globais. Para a definição e validação do escopo do projeto é utilizada a orientação do PMBOK, com a criação de uma linha base de escopo do projeto contendo requisitos bem detalhados e sujeita a aprovação e registro formal. Já a subdivisão das entregas, monitoração e controle do escopo são baseadas na proposta do SCRUM, sendo adotada a divisão do projeto em incrementos com iterações de tempo pré-estabelecidos e dos requisitos do projeto em estórias.

A definição do escopo do projeto e a coleta dos requisitos do produto são executadas em duas fases. A primeira fase, denominada conceito, é a fase na qual a gerência de produtos define e documenta o conceito de um novo produto de *software* ou de novas funcionalidades para um produto já existente. A segunda fase, denominada planejamento, é a fase na qual são realizadas duas atividades principais: a definição do plano de requisitos e desenvolvimento do projeto e a decomposição dos requisitos do projeto em unidades menores ou estórias, conforme proposto pelo SCRUM.

A definição do plano de requisitos e desenvolvimento do projeto é executada pela equipe de gerência de projeto (gerente de produto, gerente de projeto, líder de equipe, líder de arquitetura e líder de teste). Nesta etapa do gerenciamento de escopo, os requisitos do projeto são esclarecidos, analisados e priorizados a fim de determinar quais dos requisitos serão abordados pelo projeto atual. O resultado dessa análise é registrado e submetido à aprovação de um conselho de revisão de produtos e serviços que inclui representantes das áreas de gerenciamento de produtos, operações de serviços globais, *marketing*, finanças, manufatura e outras, conforme necessidade. Durante o processo de aprovação do registro oficial da lista de requisitos, deve haver uma revisão para resolução de requisitos incompletos, ambíguos ou conflitantes.

A decomposição dos requisitos em histórias é chamada de iteração zero e é iniciada após a aprovação do escopo e tamanho do projeto. A iteração zero é realizada pela equipe do projeto. É um esforço para decompor os requisitos do projeto em unidades menores, alinhá-las com as iterações de *build* (iterações para construção de um pacote de entregas funcionais), para criar os planos de teste de apoio e para determinar o ambiente do projeto.

A Organização prevê em seu gerenciamento de escopo a rastreabilidade dos requisitos do projeto durante todo o ciclo deste, por isso nessa fase do projeto, na iteração zero, a equipe deve estabelecer e documentar os marcos para revisões e os critérios para as avaliações. Ao definir critérios para avaliações, três aspectos precisam ser considerados: cronograma, custo e qualidade.

Todas as histórias são documentadas em uma tabela de acompanhamento utilizada para rastreamento dos requisitos, conforme demonstrado na Figura 24.

Figura 24 – Tabela de Acompanhamento de Estórias

<<Nome do Projeto>> Informação de Estória																																							
Legenda			Iteração Corrente												0																								
Branco	Não Iniciado																																						
Amarelo	Construído																																						
Verde	Completo																																						
Vermelho	Atrasado																																						
<Nome do Projeto>	SCR ID	Req ID	Componente	Eplco	Estória	Complexidade	Planejado	Construído	Completo	Pontos Des. Estimados			Pontos de Teste Estimados			Actual (Dias Ideais)			Desenvolvimento	Teste	Contador de Teste	Data Término	CQ SCR Estado	Total Pontos Estimados	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00						
										Totais	0	0	0	0,00	0,00	0,00	0,00	0,00																0,00	0,00	0,00	0,00	0,00	0,00

Fonte: Dados da Pesquisa

A Tabela de Acompanhamento de Estórias é uma planilha Excel disponibilizado na intranet da Organização para acompanhamento e colaboração de todas as equipes globais envolvidas no projeto e deve ser preenchida a cada projeto com as seguintes informações:

- SCR ID – *Software Change Request Identity* (Identificador de Solicitação de Mudança de Software) quando se tratar de um projeto de correção ou atualização de um produto de software existente;
- REQ ID - *Requirement Identity* (Identificador de Requisito) para identificar o requisito de produto novo;
  - Componente de software para o qual a SCR ID ou REQ ID foram atribuídos;
  - Épico relacionada à estória.
  - Complexidade, que pode ser alta, média ou baixa e permite o entendimento do esforço necessário estimado;
  - Iterações planejadas, desenvolvidas e completas. Permite que se avalie os desvios de planejamento da iteração.
  - Associados de desenvolvimento e teste para registro dos responsáveis pela estória.
  - Pontos de estória estimados no início da iteração para desenvolvimento e teste;
  - Pontos de estória atuais para desenvolvimento e teste, se ocorreram mudanças em relação pontos de estória estimados;
  - Os nomes dos membros da equipe responsáveis pelo desenvolvimento e teste da estória;
    - Número de ciclos de teste necessários para validação da estória;
    - Data de término da estória;
    - Situação do controle de qualidade da estória: deferido ou fechado;
    - Total de pontos de estória estimados, que consiste na soma de pontos de estória estimados para desenvolvimento e teste;
    - Total de pontos de estória reais, que consiste na soma de pontos de estória atuais para desenvolvimento e teste;

Uma vez que a maior parte dos projetos desenvolvidos pela Organização são críticos no que diz respeito à segurança da informação, existe na empresa uma equipe de segurança e risco global que também deve ser contratada para revisão dos

requisitos e ter a oportunidade de avaliar a necessidade de requisitos de segurança adicionais.

#### 4.5.4 Gerenciamento de Tempo

A área de gerenciamento de tempo possui tratamentos diferentes nos processos do PMBOK e do SCRUM. Enquanto o PMBOK permite que se estabeleçam as políticas, procedimentos e documentação para gerenciamento do cronograma, o SCRUM já define como será esse gerenciamento, que se baseia em prescrição de eventos com duração limitada: iteração, reuniões de início de iteração e reuniões diárias. Prevê também os membros da equipe com participação obrigatória nos eventos, e documentação mínima.

Outra importante diferença entre PMBOK e SCRUM no gerenciamento do tempo do projeto está na duração das atividades do projeto. O PMBOK propõe a estimativa do tempo de trabalho para executar cada atividade do projeto. O SCRUM propõe que se estimem quais atividades do projeto podem ser desenvolvidas no tempo de trabalho de uma iteração que se recomenda não ultrapassar trinta dias.

A Organização utiliza a abordagem iterativa e incremental da metodologia SCRUM em seu gerenciamento de tempo dos projetos. Os projetos são divididos em iterações, ou *sprints* no SCRUM, que tem sua duração estabelecida conforme o tamanho do projeto:

- Para projetos pequenos: uma a duas semanas;
- Para projetos médios: duas a três semanas;
- Para projetos grandes: três a cinco semanas.

Entretanto antes da decomposição dos requisitos do projeto em estórias e por sua vez a distribuição dessas em iterações conforme descrito no Gerenciamento de Escopo, a Organização solicita que a área de gerência de produto apresente uma estimativa da duração do projeto do tipo ROM - *Rough Order of Magnitude* ou Ordem Aproximada de Magnitude. Uma estimativa ROM é feita durante as fases iniciais de um projeto, quando todos os requisitos não podem ser totalmente definidos. Como tal, uma estimativa ROM pode ser tanto 50% acima ou abaixo do valor real de tempo de duração do projeto. Essa estimativa é útil para uma previsão inicial de custo e recursos para o projeto.

Após a aprovação oficial do escopo do projeto e da estimativa ROM, a estimativa do cronograma do projeto é feita utilizando o método proposto pelo SCRUM. Os requisitos do produto são decompostos em épicos, cada épico em estórias, para cada estória é atribuído um ponto de estória que é estimado através do método *Planning Poker*. O resultado é registrado na Tabela de Acompanhamento de Estórias apresentada na Figura 24.

A partir das informações da Tabela de Acompanhamento de Estórias, é gerado o cartão de estória (*story card*), ilustrado na Figura 25.

**Figura 25 – Cartão de Estória.**

<b>SCR ID</b>					<b>Dias Ideais Estimados</b>		
<b>Req ID</b>					<b>Total</b>		
<b>Componente</b>					<b>Dev</b>	<b>Teste</b>	
<b>Complexidade</b>							
<b>Epico</b>							
<b>Estória</b>							
<b>ITERAÇÕES</b>				<b>Associados</b>			
<b>Planejado</b>	<b>Construído</b>	<b>Completo</b>	<b>Desenvolvimento</b>		<b>Teste</b>		

Fonte: Dados da Pesquisa

O cartão de estória é disponibilizado na Intranet da Organização para visualização global, mas deve ser também impresso e fixado em um quadro na sala de reuniões da equipe local, conforme orientação do SCRUM, permitindo que essa equipe faça a reunião *stand up* (em pé) diária enquanto atualizam a lista de estórias da iteração.

Na Organização um *story point* (ponto de estória) é definido como um dia ideal, que consiste em um período de tempo gasto em uma atividade sem interrupções. Estima-se, portanto, quantos dias ideais são necessários para desenvolvimento e para teste de cada estória. O total de dias ideais para completude da estória é a soma dessas estimativas.

Além das reuniões diárias previstas pelo SCRUM durante a duração de uma iteração, a Organização realiza semanalmente uma reunião de revisão da iteração para acompanhamento do cronograma, esta revisão é feita pelo Conselho de Revisão de Produtos e Serviços que inclui representantes das áreas de gerenciamento de



produtos, operações de serviços globais, *marketing*, finanças, manufatura e outras, conforme necessidade. Para essa reunião deve ser gerado um painel que expõe a situação geral do projeto conforme ilustrado na Figura 26.

**Figura 26 – Painel de Situação Geral do Projeto**

Projeto		ID		Custo do Projeto na Data			Custo Unitário do		SCRs Abertas	
Gerente de Projeto	Gerente de Produto	Data	Classe	Material	Mão de Obra	Total	Volume =	Última Semana	Esta Semana	
				Linha Base			Atual	Alvo		
				Atual						

Gate Dates: Prazo vs. Planejamento										
Nome da KPI	Tendência	Atual	P1 Planejamento		P2 Desenvolvimento		P3 Validação		CDT Saída	ARI
			Saída	2/18/2015	Saída	12/30/2015	Saída	2/17/2016		
Desempenho para Prazo	↕	0%	Linha Base	2/18/2015					ARS	12/30/2015
Estimativa de Término (EAC)	↕	2%	Previa	2/18/2015						2/17/2016
Aderência à Estimativa de Esforço	↕	110%	Atual	2/18/2015						2/17/2016
Variação NPV	↕	-2%								2/22/2016

Assuntos a Superar/Preocupações			Próximas Ações			Anotações		

Alertas Vermelho			Mitigação		

Fonte: Dados da Pesquisa

O Painel de Situação Geral do Projeto consiste de um relatório em uma única página que expõe em nível macro o custo, cronograma e recursos totais do projeto. É disponibilizado na intranet da empresa para uso durante as reuniões com os membros do Conselho de Revisão de Produtos e Serviços por vídeo conferência.

O Painel de Situação Geral do Projeto resume o desempenho do projeto em quatro KPIs - *Key Performance Indicators* (Indicadores Chaves de Desempenho):

- Desempenho de Cronograma - indica quão bem o projeto está seguindo o cronograma de linha de base a partir do planejamento. Este valor é calculado a partir das datas de marcos, comparando os dados planejados com os dados reais do projeto. Somente os dados do marco mais atrasado são usados. O alvo desejado para esta medida é zero. Um número negativo indica que o marco está adiantado, enquanto um valor positivo indica que o marco está atrasado.

- Estimativa de término - Indica a estimativa do custo provável da conclusão do projeto. Este utiliza os custos reais incorridos até à data atual, além de uma estimativa do custo para completar o trabalho restante. O alvo desejado para esta medida é zero. Um número negativo indica que o projeto está sendo executado dentro do orçamento, enquanto um valor positivo indica o projeto está sendo executado acima do orçamento.

- Aderência à Estimativa de Recursos - indica quão bem o projeto está sendo gerenciado em relação ao plano de linha de base. Este valor é calculado a partir do número de horas efetivamente gastas contra as horas que estão no plano de linha de base do planejamento. O alvo desejado para esta medida é de 100%. Um número negativo indica que o projeto está sendo executado com menos que o número previsto de horas de recurso enquanto um valor positivo indica o projeto está usando mais do que o número previsto de horas de recursos.

- Variância VPL - Indica a variação no valor presente líquido do projeto. Este valor é calculado com base nos dados informados pelo gerente de produto. O alvo desejado para esta medida é de zero por cento. Um valor muito baixo indica que o projeto não está no caminho do desempenho planejado. Um valor muito alto indica que ocorreu erro de estimativa do valor alvo.

A seção final contém a lista dos alertas vermelhos e seus planos de mitigação correspondentes. Um alerta vermelho é algo que vai afetar o resultado do projeto em termos de custo, cronograma ou qualidade. Cada alerta vermelho deve ter uma atenuação correspondente, que inclui especificamente as ações que o gerente de projeto precisa tomar para ajudar a resolver o problema. Se não houver nenhuma atenuação possível, essa conclusão deve ser indicada. Os dados desses campos finais também são introduzidos em uma agenda colaborativa também na intranet para acompanhamento das ações de mitigação dos alertas.

#### 4.5.5 Gerenciamento de Custos

O SCRUM não apresenta os processos de gerenciamento de custos tão detalhados como os processos apresentados no PMBOK, mas deixa claro que este gerenciamento deve ser feito continuamente pelo *Product Owner* durante as revisões de iteração. E eventuais desvios de orçamento devem ser comunicados às partes interessadas e quaisquer mudanças de orçamento ou mudanças de escopo devem ser negociadas antes da próxima iteração.

A Organização possui um gerenciamento de custos que se baseia nos processos do PMBOK. Estabelece procedimentos formais de gestão de custos do projeto, prevê o desenvolvimento de estimativa de recursos financeiros, determina e submete o orçamento do projeto a aprovação de um conselho administrativo, e controla o orçamento ao longo do projeto.

Após a fase de decomposição dos requisitos em histórias e definição do cronograma o gerente de projeto deve fazer o planejamento de orçamento do projeto utilizando-se do levantamento de materiais necessários e mão de obra descrita como recurso FTE - *Full Time Equivalent* (Equivalente a tempo integral). O FTE é um índice que designa um recurso trabalhando em tempo integral por um período determinado, sendo utilizado para padronizar o entendimento dos recursos disponibilizados para os projetos no ambiente DDS, uma vez que cada país possui uma moeda e regras muito específicas para horas de trabalho.

Os valores a serem gastos em materiais e os recursos FTE devem ser preenchidos em uma planilha MS-Excel *online* em site intranet para colaboração global conforme demonstrado na Figura 27.

Figura 27 – Painel de Custo do Projeto

	A	B	C	D	E	F	G
1	<b>PROJ_00</b>	<b>My Test Project</b>					
2	Load Project	1-Sep-2009		Labor	Labor	Material	Labor
3	Preview Budget	1-Mar-2010					
4	Submit Budget						
5	Edit Cost Center		Role	SWENG	SWENG	MATERIAL	ARCHITECT
6	Edit Date Range		FTEs				
7	Insert Month		<b>FTEs: 65</b>	<b>14.00</b>	<b>44.00</b>	<b>2,500.00</b>	<b>7.00</b>
8	Sep-09	2.00	2,500.00	1.00		2,500.00	1.00
9	Oct-09	5.00	-	4.00			1.00
10	Nov-09	9.00	-	4.00	4.00		1.00
11	Dec-09	13.00	-	2.00	10.00		1.00
12	Jan-10	13.00	-	2.00	10.00		1.00
13	Feb-10	12.00	-	1.00	10.00		1.00
14	Mar-10	11.00	-		10.00		1.00

Fonte: Dados da pesquisa

Os valores do material e dos recursos FTE serão calculados a partir da moeda do país de origem do desenvolvimento do projeto conforme informação do campo “centro de custo” e somente no momento da submissão para aprovação, esses valores serão convertidos para a cotação do dólar na data corrente. Na planilha ilustrada na Figura 27 os valores já estão convertidos em dólares.

A aprovação do orçamento é responsabilidade de conselho de executivos sêniores que baseia sua avaliação nos requisitos de mercado e estratégia da empresa. Este conselho formará um comitê com uma pequena equipe de profissionais experientes representando as principais áreas funcionais da empresa para monitoramento do orçamento aprovado ao longo do projeto e transferirá o poder de reavaliar a continuação ou não do projeto.

Semanalmente o gerente de projeto deve enviar ao comitê um relatório com o orçamento atualizado. Se a variação entre o orçamento planejado e o orçamento atual previsto ultrapassar 5%, o comitê deverá se reunir para chegar a um consenso sobre o plano de contingência. Se um consenso é alcançado, o processo continua. Caso contrário o processo retorna para iteração zero.

#### 4.5.6 Gerenciamento de Qualidade

Embora o SCRUM não preveja processos detalhados de gerenciamento de qualidade, os processos do PMBOK para planejamento, controle e garantia de qualidade podem ser relacionados aos eventos de reunião de planejamento de iteração, reunião diária e reunião de retrospectiva de iteração respectivamente. O SCRUM prevê três pilares de sustentação para controle de sua natureza empírica: transparência, inspeção e adaptação e conforme Schwaber e Sutherland (2013), esses pilares somados à rapidez com que os eventos pré-estabelecidos devem ocorrer no SCRUM, já garantem a qualidade do processo e consequentemente do produto final.

Na Organização durante o planejamento da iteração zero deve ser incluída no documento de planejamento do projeto uma seção que aborde as metas e critérios de qualidade para produtos de *software*. As metas de qualidade consistem em qual qualidade deve ser alcançada e os critérios referem-se à forma como a qualidade será avaliada.

Devem ser identificados os casos de uso e testes que verificam e validam a funcionalidade correta dos requisitos do projeto. Esses casos de uso e os testes serão documentados no Plano de Teste Mestre que deve ser elaborado pelo líder de teste e aprovado pelo gerente de projeto. Esse documento descreve o teste previsto para o produto (unidade, integração, integração de sistemas, e validação do sistema). O Plano de Teste Mestre define também o escopo do teste, as entregas, abordagem de teste, o nível de automação e o ambiente de teste.

Todo o código desenvolvido durante uma iteração deve ser compilado, depurado e verificado por meio de testes de unidade. Os desenvolvedores devem arquivar o código no sistema de gestão de configuração disponibilizado online pela empresa e que consiste de uma área de armazenamento e ferramentas de suporte para controlar as mudanças de software e documentação.

Uma vez que as iterações durante o qual as características do produto são desenvolvidas e verificadas estão completas, a iteração final é centrada no teste de validação do sistema da versão de produto candidata a liberação. Os testes de validação do sistema são realizados para garantir que o produto satisfaça todos os requisitos, com foco na funcionalidade de recursos e desempenho do produto.

Os esforços de teste (verificação e validação) podem ocorrer em diferentes países e os profissionais de testes são uma parte da equipe do projeto.

O resultado do teste de validação do sistema é registrado no documento Resumo do Teste que é elaborado pelo líder de teste e aprovado pelo Gerente de projeto e juntamente com todos os demais artefatos de teste, tais como documentação e *scripts* de teste, será arquivado no sistema de gestão de configuração.

A equipe de segurança e risco global deve ser envolvida neste momento para garantir que todos os requisitos de segurança foram atingidos e quaisquer questões de segurança pendentes têm um registro de solicitação de mudança criado.

Após a conclusão dos testes de validação, a equipe do projeto realiza uma demonstração para os participantes do projeto dos recursos e funcionalidade do produto completo.

Na conclusão dos testes de validação do sistema, o Comitê de Controle de Mudanças analisa os resultados do teste e determina se deve recomendar a liberação do produto, ou se uma iteração não planejada deve ser iniciada para retrabalho do produto. O Comitê de Controle de Mudanças deve avaliar a seção de Metas de

Qualidade e Critérios do documento de planejamento do projeto para garantir que todos os níveis de qualidade pré-definidos foram alcançados antes de recomendar a liberação do produto.

Para todas as iterações, uma reunião de retrospectiva é a tarefa final a ser realizada antes de uma iteração ser oficialmente declarada como concluída. O objetivo da reunião de retrospectiva é construir uma fonte de informação a partir da qual a equipe pode obter "lições aprendidas" sobre os desenvolvimentos atuais do projeto. Um membro designado da equipe do projeto compila as informações da reunião de retrospectiva para inclusão no Relatório de Análise de Projeto que é gerado no final do projeto. Estas "lições aprendidas" destinam-se a ajudar na melhoria de desenvolvimentos futuros para repetir o que foi positivo e evitar ou controlar o que foi negativo.

#### 4.5.7 Gerenciamento de Comunicações

No contexto do gerenciamento de comunicações o PMBOK e SCRUM preveem planejamento, gerenciamento e controle. Ambas prezam pela transparência e divulgação das informações conforme áreas de interesse. A diferença entre as abordagens está na forma com que esses processos são conduzidos. No PMBOK os processos se baseiam em documentação formal para controle e rastreabilidade, a fim de evitar mal-entendidos e conflitos entre as partes interessadas. No SCRUM a proposta é manter uma interação mais próxima entre os membros da equipe e gerar o mínimo de documentação necessária, contanto com a confiança entre os indivíduos.

Na Organização o gerenciamento de comunicações no âmbito do projeto segue a abordagem mais formal do PMBOK exigindo controle e registro das informações trocadas entre a equipe de desenvolvimento e os usuários e entre equipe de desenvolvimento e o líder de projeto e no âmbito da iteração de desenvolvimento segue a abordagem do SCRUM utilizando-se de informalidade na comunicação da equipe e uso do quadro para fixação dos cartões de estória para visualização da equipe.

Durante a fase de planejamento, o gerente de projeto deve incluir no Plano Mestre de Gerenciamento de Projeto uma seção com os requisitos de comunicação para o projeto e como a informação será distribuída. Nesse plano devem constar:

- Quais informações serão comunicadas, com qual nível de detalhe e formato;

- Como as informações serão comunicadas, se por meio de reuniões, e-mail, telefone, portal na Internet, etc.;
- Qual a frequência das comunicações formais e informais do projeto;
- Quem é responsável por comunicar as informações do projeto;
- Todos os modelos de documentos que serão usados para comunicação do projeto.

No plano de comunicação é definida uma matriz de comunicações, ilustrada no Quadro 2, que lista todos os grupos envolvidos na concepção e desenvolvimento do projeto para garantir uma comunicação eficaz entre os grupos.

A reunião de abertura de projeto (*kickoff meeting*) objetiva reunir todos os envolvidos no projeto para apresentação dos objetivos de execução e gestão. Entretanto apesar de prever a reunião em um mesmo local físico de todos os participantes, isso na maioria das vezes não é possível devido ao grande número de pessoas envolvidas no projeto. São realizadas várias sessões dessa mesma reunião e muitas vezes são utilizadas, também, conferências telefônicas. A duração de cada sessão é definida pelo gerente de projeto que se esforça para garantir que se cumpra o horário.

A reunião diária para planejar e priorizar o trabalho da equipe para as próximas 24 horas é no formato SCRUM. A participação é obrigatória para todos os membros da equipe. A reunião deve durar quinze minutos ou menos, começar sempre no mesmo horário e no mesmo local, todos permanecem em pé durante a reunião (*stand up meeting*).

Na reunião diária cada membro da equipe precisa informar: o que foi executado desde a última reunião, o que será executado no dia e os eventuais obstáculos que impedem o término das histórias previstas.



**Quadro 2 – Matriz de Comunicações**

Comunicação	Objetivo	Meio	Frequência	Audiência	Responsável	Entregas	Formato
<i>Kickoff Meeting</i> (Reunião de abertura)	Introduzir o projeto e a equipe do projeto. Revisão dos objetivos do projeto e abordagem de gestão.	Face a Face	Uma vez	Gerente de Produto Equipe de Projeto Partes Interessadas	Gerente de Projeto	Agenda e Atas de Reunião	Cópias eletrônicas arquivadas no Sistema de Gerenciamento de Configuração
Reuniões diárias	Revisar situação do projeto com a equipe.	Face a Face ou Conferência telefônica	Diária	Equipe de Projeto	Gerente de Projeto	Nenhuma	N/A
Reuniões de Retrospectiva	Rever os pontos positivos e negativos do projeto.	Face a Face	Final da Iteração	Equipe de Projeto	Gerente de Projeto	Seção <i>Project Feedback</i> no Relatório de Análise de Projeto	Cópias eletrônicas arquivadas no Sistema de Gerenciamento de Configuração
Reunião de Situação de Projeto	Relatar a situação do gerenciamento do projeto.	Face a Face Conferência telefônica	Semanal	Gerente de Produto Partes Interessadas	Gerente de Projeto	Relatório Semanal de Situação do Projeto	A ser definido pelo Gerente de Projeto
Reunião de Revisão	Comunicar a situação do projeto e solicitar aprovação para próxima fase.	Face a Face	Conforme Agenda	Conselho de Revisão de Produtos e Serviços	Líder do Comitê Multifuncional Gerente de Produto	Apresentação para Conselho de Revisão de Produtos e Serviços	Cópias eletrônicas arquivadas no Sistema de Gerenciamento de Configuração

Fonte: Dados da Pesquisa

A reunião de retrospectiva da iteração é uma reunião onde toda a equipe reflete sobre o trabalho concluído e revê os pontos positivos e negativos verificados na iteração. A realimentação de cada iteração é usada como entrada na seção de *feedback* (análise crítica) do projeto no Relatório de Análise de Projeto para melhorias em projetos futuros.

As entrevistas com gerentes de projetos e membros de equipe revelaram que embora existam dificuldades com relação às diferenças de idioma e culturais durante a comunicação entre equipes localizadas em diversos países, os diversos meios de comunicação disponíveis (e-mail, fóruns, aplicativos de troca de mensagens, conferências telefônicas e video-conferências) aliados à frequência de reuniões de revisão e à formalização da documentação gerada permitem que se mitigue os erros de informação.

#### 4.5.8 Gerenciamento de Riscos

O Gerenciamento de Riscos no SCRUM possui tanta importância quanto no PMBOK, novamente não existe a previsão de documentação formal dos processos, mas propõe práticas que permitem o controle e mitigação dos riscos. A decomposição dos requisitos em iterações menores diminui a complexidade e em conjunto com o acompanhamento constante permitem o controle efetivo de prazo, custos e qualidade. As entregas de versões do produto mitigam o risco da entrega do produto final sem a qualidade esperada. As reuniões diárias possibilitam identificar os riscos do projeto com antecedência e minimizar as possíveis consequências.

Na Organização, durante a fase de planejamento do projeto, o gerente de projeto deve preencher no Plano Mestre de Gerenciamento do Projeto a seção Riscos e planos de mitigação. Esta seção deverá cobrir os riscos relacionados ao projeto (escopo, cronograma, recursos, técnicos) e os riscos relacionados à qualidade dos resultados deste projeto. Os riscos devem ser acompanhados do plano de mitigação que deve relacionar os responsáveis e as datas-alvo, quando conhecidas.

O gerente de projeto deve relacionar os riscos e os ativos conhecidos no início do projeto. Esses ativos ajudam a fornecer o nível de confiança para a conclusão bem sucedida do projeto. Os ativos podem estar relacionados à base de conhecimento, à força de trabalho qualificada, aos recursos, à clareza do escopo, às ferramentas e

técnicas, à excelência de processos ou à gestão. O gerente de projetos relaciona riscos e ativos com o auxílio do gerente de produto, da gestão local e da equipe de projetos.

Na seção Riscos e Planos de Mitigação, do Plano Mestre de Gerenciamento do Projeto, deve ser incluído o link para a planilha de avaliação de riscos, que conterá os riscos relacionados tanto no início quanto durante o desenvolvimento do projeto e para cada risco relacionado devem ser preenchidas as seguintes informações:

- Causa potencial: lista os principais fatores contribuintes para a falha no projeto ou no produto.

- Ocorrência: classifica a propabilidade de ocorrência da falha (risco) com um índice de classificação, em uma escala de 1 a 5, conforme domonstrado no Quadro 3.

**Quadro 3 – Classificação de ocorrência de risco**

Ocorrência	Taxa de Probabilidade de Ocorrência	Classificação
Quase certo	É muito provável que o risco aconteça	5
Alta	Alta probabilidade de o risco ocorrer	4
Moderada	Moderada probabilidade de o risco ocorrer	3
Improvável	Baixa probabilidade de o risco ocorrer	2
Remota	É muito improvável que o risco aconteça	1

Fonte: Dados da Pesquisa

- Efeito potencial do risco: Descreve o efeito ou impacto do risco para o cliente ou para o projeto.

- Severidade: classifica os efeitos do risco sobre cliente ou projeto com um índice de classificação, em uma escala de 1 a 5, conforme demonstrado no Quadro 4.

**Quadro 4 – Classificação de severidade de risco**

<b>Efeito</b>	<b>Severidade do Efeito</b>	<b>Classificação</b>
<b>Catastrófico</b>	Muito alta gravidade. Impactante a todos os clientes, nenhuma solução, nenhum potencial para recuperação, impacto fatal para o projeto.	<b>5</b>
<b>Sério</b>	Alta gravidade. Nem todos os clientes serão afetados, nenhuma solução alternativa, algum potencial para recuperação.	<b>4</b>
<b>Moderado</b>	Não afeta todos os clientes, potencial solução alternativa, potencial para recuperação; não fatal para o projeto.	<b>3</b>
<b>Insignificante</b>	Impacta poucos clientes, solução alternativa possível, recuperação possível	<b>2</b>
<b>Nenhum</b>	Sem efeito - insignificante - não impactante ao cliente.	<b>1</b>

Fonte: Dados da Pesquisa

- Controles atuais: Processos, métricas ou procedimentos que estão em vigor para detectar o risco ou reduzir sua gravidade.

- Detecção: classifica a probabilidade de que o risco será detectado e mitigado antes que se perceba seu impacto, conforme apresentado no Quadro 5.

**Quadro 5 – Classificação de Detecção de risco**

<b>Detecção</b>	<b>Probabilidade de detecção prévia</b>	<b>Classificação</b>
Incerteza Absoluta	Risco não será identificado antes que ele ocorra e não haverá tempo para reagir.	<b>5</b>
Remota	Chance remota de o risco ser identificado antes que ele ocorra com pouco ou nenhum tempo para reagir.	<b>4</b>
Moderada	Possibilidade moderada do risco será identificado antes que ocorra com o tempo para reagir.	<b>3</b>
Alta	Alta possibilidade de o risco ser identificado antes que ocorra com o tempo para reagir.	<b>2</b>
Quase Certa	Risco será identificado antes que ele ocorra e haverá tempo para reagir.	<b>1</b>

Fonte: Dados da Pesquisa

- Status: Aberto ou fechado. Um risco estará aberto até que o risco ou sua ameaça não exista mais.

- Proprietário de Risco: A pessoa responsável por monitorar os controles atuais para evitar o risco ou dar um aviso de que o risco vai ocorrer. O proprietário de

avaliação de risco é responsável pelo acompanhamento para garantir que todas as ações sejam implementadas, devidamente documentadas ou atualizadas.

Assim como o planejamento de escopo, cronograma, orçamento e recursos são refinados durante o planejamento do projeto, a planilha de riscos e o plano de mitigação também devem ser atualizados para refletir novos riscos identificados.

#### 4.5.9 Gerenciamento de Aquisição

O SCRUM não faz nenhuma menção a processos de planejamento, condução e controle de aquisição de produtos e serviços externos tal como é previsto no guia PMBOK.

A Organização possui uma postura muito formal no gerenciamento de aquisições, baseia seus processos no guia PMBOK e estabelece normas rígidas para gerenciamento e controle de fornecedores externos.

O planejamento de aquisições é responsabilidade da equipe global de aquisições. A equipe global de aquisições trabalha com o gerente de projeto e com as demais partes interessadas internas para identificar os requisitos do projeto, a qualidade e a confiabilidade necessárias para a solução, e as condições comerciais necessárias para a Organização. Na fase de planejamento a equipe de aquisições deve revisar a estratégia de aquisição e identificar riscos, tais como contratos de longo prazo, patentes, restrições de capacidade e conformidade ética. Deve-se desenvolver um plano detalhado do processo de aquisição com marcos importantes e identificação dos responsáveis de cada tarefa.

O gerente de projeto deve definir no Plano Mestre de Gerenciamento do Projeto, quando aplicável, todos os produtos ou serviços a serem adquiridos para inclusão no projeto, seja por meio de terceiros ou adquiridos como *Commercial-Off-The-Shelf* (que são *softwares* comerciais especializados que podem ser usados com pouca ou nenhuma modificação).

O gerente de projeto deve trabalhar com a equipe global de aquisições para condução das negociações com quaisquer fornecedores de produtos ou serviços necessários ao projeto. A equipe global de aquisições tem a responsabilidade e a autoridade para a seleção de fornecedores com base na avaliação dos processos do fornecedor em áreas primárias de qualidade, custo e prazo de entrega. A aprovação, entretanto cabe à área executiva da Organização.

Trimestralmente os fornecedores são avaliados segundo um processo de melhoria de qualidade de fornecedores e recebem uma pontuação conforme seu desempenho a cerca de qualidade, custo, prazo de entrega, relacionamento, tempo de resposta, condições de pagamento e avaliação de risco. A soma dessas métricas determina o nível de desempenho que pode ser Platina, Ouro, Prata, Bronze ou Nenhum, nessa ordem de importância. Este nível de desempenho estabelece vantagens comerciais e publicitárias aos fornecedores conforme apresentado no Quadro 6.

**Quadro 6 – Classificação de Detecção de risco**

<b>Nível de Desempenho</b>	<b>Prêmio de Reconhecimento</b>
Platina	<ul style="list-style-type: none"> <li>• Apresentação da premiação no site do fornecedor</li> <li>• Anúncio em publicação comercial</li> <li>• Preferência para parcerias futuras</li> <li>• Placa de reconhecimento</li> </ul>
Ouro	<ul style="list-style-type: none"> <li>• Apresentação da premiação no site do fornecedor</li> <li>• Preferência para parcerias futuras</li> <li>• Placa de reconhecimento</li> </ul>
Prata	<ul style="list-style-type: none"> <li>• Preferência para parcerias futuras</li> <li>• Placa de reconhecimento</li> </ul>
Bronze	<ul style="list-style-type: none"> <li>• Preferência para parcerias futuras</li> <li>• Placa de reconhecimento</li> </ul>

Fonte: Dados da Pesquisa

Os fornecedores devem avançar seu desempenho pelo menos um nível a cada seis meses. Aqueles que demonstrarem melhorias contínuas terão atenção especial para projetos futuros. Por outro lado a falha em atender aos padrões mínimos de desempenho pode resultar em perda de negócios.

#### 4.5.10 Gerenciamento de Partes Interessadas

Assim como o PMBOK, o SCRUM também prevê planejamento, gerenciamento e controle das partes interessadas do projeto. Entretanto no SCRUM o foco está em envolver as partes interessadas mais cedo no projeto e com maior frequência para garantir que os requisitos sejam válidos e as interfaces com o usuário sejam efetivas.

Na Organização a concepção de um projeto é sempre resultado de uma oportunidade de negócio identificada pela área de marketing da empresa em conjunto com as áreas de engenharia e pesquisa e desenvolvimento e aprovada pela alta administração. Assim as partes interessadas do projeto são todas internas à

Organização. Nesse contexto foi interessante para a Organização adotar a abordagem SCRUM em que as partes interessadas participam de todo o ciclo de vida do projeto de forma mais próxima e frequente.

Para representação das partes interessadas, a cada projeto é formado uma equipe chamada *Concurrent Fulfillment Team (CFT)*, que consiste em uma equipe multifuncional formada por representantes da área de gerenciamento de produtos, marketing, engenharia, desenvolvimento, testes, produção, aquisições e segurança, e pelos responsáveis pelo desenvolvimento de produtos, testes, liberação e instalação do produto. A equipe CFT acompanha o desenvolvimento do projeto, revisa e decide sobre a aprovação das mudanças propostas para o plano de projeto previamente aprovado.

O gerente de projeto deve indicar no Plano Mestre de Gerenciamento de Projeto os membros da equipe CFT e o líder dessa equipe, conforme demonstrado no Quadro 7. Na grande maioria dos projetos, o gerente de projeto é também o líder da equipe CFT.

**Quadro 7 – Composição de Equipe CFT.**

<b>Papel</b>	<b>Nome</b>	<b>Líder CFT</b>
Gerente de produto		
Gerente de projeto		
Gerente de programa		
Teste (CQ)		
Desenvolvimento de <i>software</i>		
Serviços profissionais		
Desenvolvimento de hardware		
Fábrica		
Engenharia de confiabilidade		
Operações e planejamento de serviços		
Abastecimento estratégico		
Aquisições		
Suporte nível 2		
Operações de serviços integrados e gerenciados		
Cadeia de fornecimento de serviços		
Liderança do programa de certificação		
Segurança corporativa		

Fonte: Dados da Pesquisa

As reuniões da equipe CFT são planejadas para ocorrerem semanalmente, com duração de uma hora. Durante a reunião, os membros avaliam o desempenho

do projeto através de uma planilha nomeada Pasta de Trabalho do Projeto, que é disponibilizada de forma *online* na intranet da empresa.

Esta planilha, elaborada utilizando-se o *software* em MS-Excel, é usada para capturar, organizar e documentar os dados de controle importantes para a gestão de um projeto. A planilha é atualizada pelo gerente do projeto para atender a qualquer necessidade da equipe do projeto.

O resultado dessas reuniões são decisões consensuais que auxiliarão na solução de impasses encontrados no desenvolvimento do projeto e são também informações para condução das revisões de negócio, que são reuniões agendadas durante o ciclo de desenvolvimento do projeto, nas quais os membros do Conselho de Revisão de Produtos e Serviços e os membros do Conselho da Agenda de Entregas tomam decisões de nível estratégico, alocam fundos para os esforços de desenvolvimento de produtos e fornecem orientação e liderança para a equipe CFT de projeto.

#### 4.6 DISCUSSÃO DOS RESULTADOS OBSERVADOS

O objetivo deste capítulo foi identificar os processos propostos no modelo PMBOK e os processos propostos pelo modelo SCRUM no modelo de gerenciamento de desenvolvimento distribuído de *software* da Organização. A partir do estudo de caso dos projetos globais da Organização estabeleceu-se o relacionamento entre as áreas de conhecimento da gestão de projetos que representam as melhores práticas de gestão com os processos do PMBOK e processos SCRUM utilizados pela Organização, conforme Quadro 8.



**Quadro 8 – Comparação de Processos PMBOK e SCRUM Utilizados pela Organização**

<b>Área de Gerenciamento</b>	<b>Processos PMBOK Utilizados</b>	<b>Processos SCRUM Utilizados</b>
Gerenciamento de Recursos Humanos	<ul style="list-style-type: none"> <li>- Desenvolver o plano dos recursos humanos</li> <li>- Mobilizar a equipe do projeto</li> <li>- Desenvolver e gerenciar a equipe do projeto</li> </ul>	<ul style="list-style-type: none"> <li>- Estabelecer times de desenvolvimento com tamanho entre três e nove membros.</li> <li>- Reconhecimento por equipe</li> </ul>
Gerenciamento de Integração	<ul style="list-style-type: none"> <li>- Desenvolver o termo de abertura do projeto</li> <li>- Desenvolver o plano de gerenciamento do projeto</li> <li>- Orientar e gerenciar o trabalho do projeto</li> <li>- Monitorar e controlar o trabalho do projeto</li> <li>- Realizar o controle integrado de mudanças</li> <li>- Encerrar o projeto ou fase</li> </ul>	<ul style="list-style-type: none"> <li>- Planejamento da <i>sprint</i></li> <li>- Revisão da <i>sprint</i></li> <li>- Retrospectiva da <i>sprint</i></li> <li>- Gerenciar o <i>backlog</i> do Produto</li> <li>- Garantir transparência para todas as partes interessadas</li> </ul>
Gerenciamento de Escopo	<ul style="list-style-type: none"> <li>- Planejar o gerenciamento do escopo</li> <li>- Coletar os requisitos</li> <li>- Definir o escopo</li> <li>- Validar o escopo</li> <li>- Controlar o escopo</li> </ul>	<ul style="list-style-type: none"> <li>- Desenvolvimento do <i>backlog</i> do produto</li> <li>- Desenvolvimento do <i>backlog</i> da <i>sprint</i></li> <li>- Decompor requisitos em histórias</li> <li>- Dividir projeto em iterações com tempo de duração máxima de 30 dias.</li> </ul>
Gerenciamento de Tempo	<ul style="list-style-type: none"> <li>- Planejar o gerenciamento do cronograma</li> <li>- Definir as atividades</li> <li>- Sequenciar as atividades</li> <li>- Estimar os recursos e duração das atividades</li> <li>- Desenvolver o cronograma</li> <li>- Controlar o cronograma</li> </ul>	<ul style="list-style-type: none"> <li>- Planejamento da <i>sprint</i></li> <li>- Decomposição dos requisitos em histórias</li> <li>- Atribuição de pontos de história</li> <li>- Uso da técnica de <i>Planning Poker</i> para estimativa de esforço</li> <li>- Divisão do projeto em <i>sprints</i> com duração pré-definida</li> <li>- Reuniões diárias do tipo <i>Stand Up</i></li> </ul>
Gerenciamento de Custos	<ul style="list-style-type: none"> <li>- Planejar o gerenciamento dos custos</li> <li>- Estimar os custos</li> <li>- Determinar o orçamento</li> <li>- Controlar os custos</li> </ul>	- - -
Gerenciamento de Qualidade	<ul style="list-style-type: none"> <li>- Planejar o gerenciamento da qualidade</li> <li>- Realizar a garantia da qualidade</li> <li>- Realizar o controle da qualidade</li> </ul>	<ul style="list-style-type: none"> <li>- Planejamento da <i>sprint</i></li> <li>- Definição de “pronto” para entregas com consenso nos padrões de qualidade</li> <li>- Revisão da <i>sprint</i></li> <li>- Reunião diária</li> <li>- Retrospectiva da <i>sprint</i></li> </ul>

Fonte: Dados da Pesquisa; PMI, 2013; SCHWABER; SUTHERLAND, 2013

**Quadro 9 – Comparação de Processos PMBOK e SCRUM Utilizados pela Organização (continuação)**

<b>Área de Gerenciamento</b>	<b>Processos PMBOK Utilizados</b>	<b>Processos SCRUM Utilizados</b>
Gerenciamento de Comunicações	<ul style="list-style-type: none"> <li>- Planejar o gerenciamento das comunicações</li> <li>- Gerenciar as comunicações</li> <li>- Controlar as comunicações</li> </ul>	<ul style="list-style-type: none"> <li>- Reunião de planejamento da <i>sprint</i></li> <li>- Reuniões diárias do tipo <i>Stand Up</i></li> <li>- Reunião de revisão da <i>sprint</i></li> <li>- Reunião de retrospectiva da <i>sprint</i></li> </ul>
Gerenciamento de Riscos	<ul style="list-style-type: none"> <li>- Planejar o gerenciamento dos riscos</li> <li>  Identificar os riscos</li> <li>- Realizar a análise qualitativa dos riscos</li> <li>  Planejar as respostas aos riscos</li> <li>- Controlar os riscos</li> </ul>	<ul style="list-style-type: none"> <li>- Planejamento da <i>sprint</i></li> <li>- Revisão da <i>sprint</i></li> </ul>
Gerenciamento de Aquisições	<ul style="list-style-type: none"> <li>- Planejar o gerenciamento das aquisições</li> <li>- Conduzir as aquisições</li> <li>- Controlar as aquisições</li> <li>- Encerrar as aquisições</li> </ul>	- - -
Gerenciamento de Partes Interessadas	<ul style="list-style-type: none"> <li>- Identificar as partes interessadas</li> <li>- Planejar o gerenciamento das partes interessadas</li> </ul>	<ul style="list-style-type: none"> <li>- Planejamento da <i>sprint</i></li> <li>- Revisão da <i>sprint</i></li> </ul>

Fonte: Dados da Pesquisa; PMI, 2013; SCHWABER; SUTHERLAND, 2013

Os dados apresentados demonstram que os processos recomendados pelo SCRUM são insuficientes para atender à demanda por gerenciamento e controle de todas as áreas de gerenciamento de projeto no ambiente de desenvolvimento distribuído de forma a garantir a confiabilidade, integridade e segurança dos produtos de *software*. Especialmente em relação à criticidade dos projetos de *software* do setor de automação bancária que prevê um controle rigoroso de qualidade e riscos.

Entretanto, a característica do SCRUM de orientação a eventos com duração curta e pré-determinada se mostra muito aderente ao gerenciamento de tempo no desenvolvimento distribuído de software, já que contribui para a eficácia da sincronização dos trabalhos, que é, segundo Herbsleb e Moitra (2001), a principal característica do bom gerenciamento do DDS.

Além disso, a utilização de processos SCRUM adaptados ao modelo DDS proporcionou melhoria no desempenho dos projetos como consequência dos seguintes fatores:

- Aumento da produtividade da equipe devido à decomposição e distribuição dos trabalhos com entregas mais rápidas e frequentes;

- Melhoria da comunicação e interação entre as partes interessadas devido às reuniões periódicas;

- Melhoria no controle de qualidade dos incrementos propiciada pelas revisões constantes;

- Melhoria contínua dos processos através das reuniões de retrospectiva ao final das iterações.

Foram aplicadas técnicas do SCRUM com ganhos diretos de desempenho nas áreas de gerenciamento de escopo, tempo e qualidade.

Entretanto também foram identificados problemas relacionados às práticas do SCRUM no contexto DDS. Os problemas são:

- Dificuldade no agendamento de reuniões diárias com todos os membros necessários devido a diferenças de fuso horário;

- Dificuldade em cumprir a regra de reuniões de 15 minutos devido às dificuldades com as diferenças de idioma;

Com relação ao guia PMBOK, constatou-se maior aderência de seus processos ao gerenciamento de projetos de DDS da Organização na maioria das áreas de gerenciamento, pois sua natureza preditiva e rígida com base em documentação extensa permite a mitigação de riscos associados aos pontos críticos do DDS, relacionados por Carmel (1999) que são: falhas de comunicação, dificuldade para transferência de conhecimento, dificuldade para desenvolvimento de consciência contextual e dificuldade de desenvolvimento de confiança e companheirismo entre os membros da equipe.

Os benefícios percebidos pela utilização dos processos PMBOK no gerenciamento de DDS são:

- Aumento da confiabilidade devido à padronização dos processos de gerenciamento de projetos;

- Melhor atendimento aos padrões internacionais de segurança impostos pelas instituições financeiras;

- Melhor controle de custos como consequência da definição de requisitos estáveis.

Os dados analisados indicaram que a Organização aplica todos os processos previstos no PMBOK para as áreas de gerenciamento de integração, escopo, tempo, custos, qualidade, comunicações e aquisição.

Os problemas encontrados na aplicação dos processos do PMBOK foram:

- Excesso de responsabilidade sobre o gerente de projeto;
- Gargalos de processo na fase de testes de qualidade;

Na área de gerenciamento de riscos não foram encontradas evidências da aplicação do processo de análise quantitativa dos riscos, que se refere à análise numérica do efeito dos riscos nas metas do projeto prevista pelo guia PMBOK.

Da mesma forma, a área de gerenciamento de partes interessadas propõe processos de gerenciamento e controle de engajamento das partes interessadas, os quais não estão previstos nos guias e procedimentos de gerenciamento de projetos globais da Organização, sendo este um ponto de melhoria incluído nas metas de melhoria de desempenho da área de DDS da Organização.

## 5 CONSIDERAÇÕES FINAIS

Esta pesquisa apresentou o estudo sobre modelos de gerenciamento de projetos de *software* em ambientes distribuídos considerando duas abordagens muito utilizadas atualmente, com o objetivo de auxiliar organizações para melhor aproveitamento das vantagens do desenvolvimento global.

Os resultados obtidos indicam que é possível unir os modelos PMBOK e SCRUM para um gerenciamento efetivo de projetos de desenvolvimento de *software* distribuído.

A Organização obteve bons resultados com a utilização do modelo híbrido de gerenciamento de projetos globais desde o primeiro projeto global que foi o desenvolvimento de uma nova linha de equipamentos de autoatendimento e toda a solução de *software* necessária para o funcionamento. O projeto foi desenvolvido e disponibilizado para o mercado em três anos, enquanto que o projeto da linha anterior de equipamentos com desenvolvimento local teve duração de dez anos.

O bom desempenho do modelo de gerenciamento da Organização também pode ser observado por meio dos dados de desempenho dos projetos desenvolvidos pela Organização durante o período de 2011 a 2015. Durante este período foram realizados 186 projetos, sendo 182 projetos concluídos no prazo previsto.

Os dados de desempenho dos projetos mostraram que ocorreram atrasos nos marcos (*milestones*) que são normalmente revisões de iteração durante o desenvolvimento dos projetos em 47 projetos, que corresponde a 25,26% do total de projetos concluídos. Entretanto esses atrasos foram corrigidos durante as iterações subsequentes. Isso mostra que a divisão do projeto em iterações com revisões constantes e pré-agendadas são eficientes para controle de cronograma.

A média de variação de custo dos projetos da Organização está entre 4% e 10% do orçamento planejado no início dos trabalhos, enquanto que 84,5% das empresas pesquisadas pela Standish Group (2014) responderam que a variação média de custo dos projetos é maior que 20% do orçamento inicial.

### 5.1 RESPOSTA À QUESTÃO DE PESQUISA

Considerando os resultados obtidos nesta pesquisa pode-se concluir que a resposta à questão: “Considerando um modelo de gestão tradicional de projetos, como o proposto pelo PMI e um modelo de gestão de projetos ágil, como o SCRUM,

qual modelo de gestão de projetos melhor se adequa ao contexto de desenvolvimento distribuído de *software* em empresas de automação bancária? “ Seria: um modelo híbrido composto por elementos do PMBOK e do SCRUM.

O modelo híbrido provê processos preditivos, estáveis e baseados em documentação detalhada baseados no modelo do guia PMBOK que permitem melhor controle, sincronização e integração das atividades do projeto, além de maior garantia de controle de riscos e segurança da informação. O modelo híbrido provê também processos adaptativos, flexíveis e empíricos baseados no guia SCRUM que permitem aumento de produtividade e desempenho.

Considerando o caso de uso apresentado no estudo, o modelo híbrido foi utilizado pela Organização para gerenciamento de projetos de sistemas críticos que demandam confiabilidade e integridade da informação e que são desenvolvidos sob o modelo de desenvolvimento distribuído de *software*, e se mostrou efetivo para entrega de produto final com qualidade e desempenho esperados.

## 5.2 OBJETIVOS PROPOSTOS

Uma vez que foi possível encontrar evidências da aplicação de processos de ambos os modelos: PMBOK e SCRUM no sistema de gerenciamento de projetos globais da Organização, conclui-se que o objetivo principal foi alcançado. Foi possível comparar os métodos de gestão de projeto de *software* PMBOK e SCRUM, e verificar sua adequação ao desenvolvimento distribuído de *software*.

O estudo permitiu também descrever as características do desenvolvimento distribuído de *software*, identificando fatores críticos de gerenciamento.

Através do estudo foi possível atender parcialmente ao objetivo de identificar as vantagens e desvantagens de cada modelo de gestão de projeto em ambiente de desenvolvimento distribuído de *software*, uma vez que não houve na Organização nenhum projeto desenvolvido integralmente utilizando processos de um só modelo.

## 5.3 CONTRIBUIÇÕES ACADÊMICAS E GERENCIAIS

A principal contribuição acadêmica e gerencial do estudo é apresentar um caso de sucesso no gerenciamento de projetos em ambiente de desenvolvimento distribuído de *software* utilizando a combinação de dois modelos de gestão mundialmente conhecidos e de eficiência comprovada, que pode ser usado como modelo para gerenciamento de projetos globais bem como de projetos locais.

Apesar do escopo dessa pesquisa estar limitado ao estudo do gerenciamento de projetos de software em uma empresa de automação bancária seus resultados podem ser aproveitados por empresas em qualquer ramo de atuação que utilizam o modelo de desenvolvimento distribuído de *software*.

Uma contribuição adicional da pesquisa é a estabelecer uma comparação entre os modelos de gestão de projetos do guia PMBOK e SCRUM apontando as vantagens e desvantagens de cada abordagem para o gerenciamento de projetos de software e a forma com que estas se complementam.

#### 5.4 LIMITAÇÕES DE PESQUISA E CONTRIBUIÇÕES PARA ESTUDO

A limitação encontrada foi a dificuldade em conseguir entrevistas com membros de equipes localizadas em outros países, privando a pesquisa de dados sob diferentes perspectivas.

Há que se indicar sugestões para estudos futuros complementares ou extensivos à presente pesquisa, considerando outros setores da indústria que se utilizam do modelo de desenvolvimento distribuído de *software* e focando em áreas específicas de conhecimento do gerenciamento de projetos para maior aprofundamento.

## REFERÊNCIAS BIBLIOGRÁFICAS

ÅGERFALK, P.; FITZGERALD, B.; HOLMSTROM, H.; Ó CONCHÚIR, E. **Benefits of Global Software Development: The Known and Unknown**, in Q. Wang, D Pfahl, and D.M. Raffo (Eds.): Making Globally Distributed Software a Success Story, ICSP 2008, LNCS 5007, pp. 1-9, Springer-Verlag Berlin Heidelberg.

AGILE ALLIANCE. **Manifesto for agile software development**. AgilManifesto.org, 2001. Disponível em: <<http://www.agilemanifesto.org>>. Acesso em 1 nov. 2014.

BANCO DO BRASIL. **Banco do Brasil: 200 anos – 1964-2008**. Livro 2. Belo Horizonte: Del Rey, Fazenda Comunicação & Marketing, 2010.

BARDIN, L. **Análise de conteúdo**. Lisboa: Edições 70, 1979. 229 p.

BEGEL, A.; NAGAPPAN N. **Global Software Development: Who Does It?** Redmond: Microsoft Research, 2008. Disponível em: <<http://www.microsoft.com>>. Acesso em 18 ago. 2015.

BODEN, A.; NETT, B.; WULF V. **Coordination Practices in Distributed Software Development of Small Enterprises**. In: IEEE 2nd International Conference on Global Software Engineering (ICGSE'07), Ago. 2007. Munich: IEEE Press, pp. 235-244.

BROOKS, F. P. **The mythical man-month**. Boston: Addison Wesley Longman Inc., 1995.

CABLE, J. **ENCE 662 – Introduction to Project Management Lecture Notes**. The University of Maryland, College Park, 2011.

CARMEL, E. **Global software teams: Collaborating across borders and time-zone**. Upper Saddle River: Prentice Hall, 1999.



CARMEL, E.; TIJA, P. **Off shoring information technology**: Sourcing and outsourcing for a global workforce. Cambridge: Cambridge University Press, 2005.

CEN - COMITÉ EUROPÉEN DE NORMALISATION. **Extensions for Financial Services (XFS) interface specification - Release 3.0 - Part 1**. Bruxelles: CEN, 2000.

CHINBAT, S. **Lessons Learned in Virtual Teams from Global Software Development**. 2010. Bachelor of Software Engineering & Management Thesis, Gothenburg.

COHN, M. **Agile Estimating and Planning**. New Jersey: Pearson Education, Inc., 2006.

COHN, M. **Succeeding with Agile Software development using Scrum**. New Jersey: Addison-Wesley, 2010.

DINIZ, E. H. **Evolução e Segmentação no Perfil dos Serviços Bancários pela Internet**. São Paulo: Relatório de Pesquisa FGV, 2004a.

DINIZ, E. H. **Era Digital: Cinco Décadas de Automação Bancária**. São Paulo: GV Executivo, v. 03, n. 3, p. 7-97, ago. /out. 2004b.

ESPINOSA, J. A.; SLAUGHTER, S.A.; KRAUT, R. E; HERBSLEB, J. D. Team Knowledge and Coordination in Geographically Distributed Software Development. **Journal of Management Information Systems**, 2007, Vol. 24, No. 1, pp. 135–169. Armonk: M.E. Sharpe, Inc., 2007.

FEBRABAN - FEDERAÇÃO BRASILEIRA DE BANCOS. **Pesquisa FEBRABAN de Tecnologia Bancária 2014**. Febraban.org, 2015. Disponível em <<http://www.febraban.org.br>>. Acesso em 14 mar. 2016.

FONSECA, C. E. C.; MEIRELES, F. S.; DINIZ, E. **Tecnologia Bancária no Brasil: Uma História de Conquistas, Uma Visão de Futuro**. São Paulo: FGV/RAE, 2010.

FRISCHTAK, C. R. **Automação bancária e mudança na produtividade: a experiência brasileira**. Pesquisa e Planejamento Econômico, agosto/1992. pp. 197-240.

GIL, A. C. **Como elaborar projetos de pesquisa**. - 4. Edição - São Paulo: Atlas, 2002.

HERBSLEB, J. D.; R. E. GRINTER. Architectures, coordination, and distance: Conway's law and beyond. **IEEE Software** n.16 (5), p.63–70, 1999.

HERBSLEB, J. D.; MOITRA D. Global Software Development. **IEEE Software**, p.16-20, 2001.

HERBSLEB, J. D. Global Software Engineering: The Future of Socio-technical Coordination. **IEEE Computer Society**, 2007.

HIGHSMITH, J. **History: The Agile Manifesto**. AgilManifesto.org, 2001. Disponível em: <<http://www.agilemanifesto.org>>. Acesso em 1 nov. 2014.

HOFSTEDE, G. **Culture and Organizations: Software of the Mind**. New York: McGraw-Hill, 1997.

IAMONTI, V. Z. Automação Bancária No Brasil: O Circuito Espacial De Produção E A Difusão Dos Terminais De Autoatendimento (Atms) Pelo Território Brasileiro No Período Da Globalização. São Paulo: **Congresso Ibero-americano de Estudos Territoriais e Ambientais** (CIETA), 2014.

IEEE. Disponível em <[www.ieee.org](http://www.ieee.org)>. Acesso em: 16 fev. 2016.

KEYES, J. **Banking Technology Handbook**. Boca Raton: CRC Press LLC, 1999.

KERZNER, H. **Project Management: A Systems Approach to Planning, Scheduling, and Controlling**. New Jersey: John Wiley & Sons, Inc., 2009.

KERZNER, H. **Project Management: Best Practices. Achieving Global Excellence.** New Jersey: John Wiley & Sons, Inc., 2014.

KERZNER, H. **Gestão de Projetos: As melhores Práticas.** Porto Alegre: Bookman, 2007.

KIEL, L. Experiences in Distributed Development: A Case Study. **International Workshop on Global Development at ICSE**, 2003.

KNIBERG H. **Scrum and XP from the Trenches: How we do Scrum.** New York: C4Media Inc., 2007.

LAKATOS, E. M.; MARCONI, M. A. **Fundamentos de metodologia científica.** 1. - 5. ed. - São Paulo: Atlas, 2003.

LEMME S. **Successfully Managing Remote Teams.** In: Expert Reference Series of White Papers. Cary: Global Knowledge Network, 2005. Disponível em: <[www.globalknowledge.com](http://www.globalknowledge.com) or >. Acesso em 01 ago. 2015.

MACCONNELL, S. **Rapid Development.** Redmond: Microsoft Press, 1996.

MAÇADA, A. C. G; BECKER, J. L. O Impacto da Tecnologia de Informação na Estratégia dos Bancos. **Revista de Administração de Empresas**, v. 41, n. 4, p. 7-97, out. /dez. 2001.

MARTINS, D. **SCRUM e PMBOK 5ª ed.: Trabalhando juntos na Gestão de Projetos – Parte 03. Project Prime, 2013.** Disponível em: <http://projectprime.com.br/>. Acesso em 26 fev. 2016.

MARTINS, J. C. C. **Gerenciando Projetos de Desenvolvimento de Software com PMI, RUP e UML.** Rio de Janeiro: Brasport, 2007.

MINAYO, M. C. S. **Pesquisa Social. Teoria, método e criatividade**. 18 ed. Petrópolis: Vozes, 2001.

PMI. **Guia PMBOK - Um Guia Do Conhecimento Em Gerenciamento De Projetos**. 5 ed. Newtown Square: Project Management Institute, Inc., 2013.

PMI. **PMI Fact File**. PMI Today, Newtown Square, PA, Fev/2016, p.4. Disponível em <<http://www.pmitoday-digital.com/pmitoday>>. Acesso em 15 mar. 2016.

PRESSMAN, R. **Engenharia de Software: Uma Abordagem Profissional**. 7ª Ed., São Paulo: McGraw-Hill, 2011.

PRIKLADNICKI, R.; AUDY, J.; EVARISTO, R. Distributed Software Development: Toward an Understanding of the Relationship between Project Team, Users and Customers. **International Conference on Enterprise Information Systems**, 5 Anais. ICEIS Press, v3. p417-423, 2003.

SANTOS, B. O. **Utilização de métricas nos projetos de desenvolvimento de sistemas de informação: um survey com gerentes de projetos**. 2009. Dissertação (Mestrado em Informática) – Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro.

SALADIS, F. P.; KERZNER H. **Bringing the PMBOK guide to life**. New Jersey: John Wiley and Sons, 2009.

SCRUM ALLIANCE. **The State of Scrum: Benchmarks and Guidelines**. Scrum Alliance.org, jun. 2013. Disponível em: <<https://www.scrumalliance.org>>. Acesso em 11 abr. 2015.

SERPRO - SERVIÇO FEDERAL DE PROCESSAMENTO DE DADOS. SPB - UM ANO DE SUCESSO. **Revista Tema** - A revista do Serpro ano XXVIII - nº 167, Soluções Tecnológicas, maio/junho 2003. Disponível em <<http://www1.serpro.gov.br>>. Acesso em 28 jul. 2015.

SCHWABER, K. **Agile Project Management with Scrum**. Washington: Microsoft Press, 2004.

SCHWABER, K.; SUTHERLAND, J. **The Scrum Guide**: The Definitive Guide to Scrum: The Rules of the Game. 2013. Disponível em <<http://www.scrumguides.org>>. Acesso em 1 nov. 2014.

SEGNINI, L. R. P. Reestruturação nos Bancos no Brasil: Desemprego, Subcontratação E Intensificação Do Trabalho. **Educação & Sociedade**, V.20, N.67, ago. 1999.

SOFTEX. **Cadernos temáticos do observatório**: Mercado de trabalho e formação de mão de obra em TI, 2013. Disponível em: <<http://www.softex.br>>. Acesso em 1 nov. 2014.

SOMMERVILLE, I. **Engenharia de Software**. 8ª Ed., São Paulo: Addison Wesley Brasil, 2007.

SOTILLE, M. A.; MENEZES, L. C. M.; XAVIER, L. F. S.; PEREIRA, M. L. S. Gerenciamento do Escopo em Projetos. Rio de Janeiro: Editora FGV, 2014.

THE STANDISH GROUP. **The CHAOS Report**. 2014.

TURNER J. R. **The Handbook of Project Based Management: Leading Strategic Change in Organizations**. Nova York: McGraw Hill, 2009.

WELLS, D. **Extreme Programming**: A gentle introduction. 2009. Disponível em: <<http://www.extremeprogramming.org>>. Acesso em 11 abr. 2015.

YIN, R. K. **Estudo De Caso - Planejamento E Métodos**. Porto Alegre: Bookman, 2005.

## **APÊNDICE A – ROTEIRO DE ENTREVISTA SOBRE GERENCIAMENTO DE RECURSOS HUMANOS PARA GERENTES DE PROJETOS**

1. As equipes são formadas por profissionais distribuídos ou as equipes são centralizadas em cada país?
2. Existem problemas de comunicação?
3. Quais as causas dos problemas de comunicação?
4. Existe colaboração entre as equipes globais?
5. As diferenças culturais são impeditivas para a colaboração?
6. Quais costumes influenciam negativamente na colaboração?
7. A equipe é auto gerenciável?
8. Você acha possível implantar o autogerenciamento na equipe?
9. É preciso visitas frequentes do gerente de projeto durante o projeto?
10. Existe representante do cliente em cada localidade?
11. O PMO gerencia as equipes de todos os projetos de software?
12. O treinamento de processos de gerenciamento de projeto é efetivo?
13. Existem relatórios de performance da equipe?
14. Você gostaria a avaliado de performance fosse executada por equipe ou individualmente?
15. Quantos funcionários foram demitidos devido ao DDS?

## **APÊNDICE B – ROTEIRO DE ENTREVISTA SOBRE GERENCIAMENTO DE RECURSOS HUMANOS PARA MEMBROS DA EQUIPE**

1. Existem problemas de comunicação?
2. Quais as causas dos problemas de comunicação?
3. Existe colaboração entre as equipes globais?
4. As diferenças culturais são impeditivas para a colaboração?
5. Quais costumes influenciam negativamente na colaboração?
6. A equipe é auto gerenciável?
7. Você acha possível implantar o autogerenciamento na equipe?
8. O treinamento de processos de gerenciamento de projeto é efetivo?
9. Você gostaria que a avaliação de performance fosse executada por equipe ou individualmente?
10. Você se sente reconhecido pelo seu trabalho na equipe?