

**Universidade Federal de Pernambuco
Centro de Ciências Sociais Aplicadas
Departamento de Ciências Administrativas
Programa de Pós-Graduação em Administração - PROPAD
Doutorado em Administração**

Felipe Augusto Pereira

**Proposta de um catálogo de falhas e mudanças em
projetos: um estudo de caso a partir de uma empresa
de desenvolvimento de *software***

Recife, 2014

UNIVERSIDADE FEDERAL DE PERNAMBUCO
PROGRAMA DE PÓS-GRADUAÇÃO EM ADMINISTRAÇÃO

CLASSIFICAÇÃO DE ACESSO A TESES E DISSERTAÇÕES

Considerando a natureza das informações e compromissos assumidos com suas fontes, o acesso a monografias do Programa de Pós-Graduação em Administração da Universidade Federal de Pernambuco é definido em três graus:

- "Grau 1": livre (sem prejuízo das referências ordinárias em citações diretas e indiretas);
- "Grau 2": com vedação a cópias, no todo ou em parte, sendo, em consequência, restrita a consulta em ambientes de biblioteca com saída controlada;
- "Grau 3": apenas com autorização expressa do autor, por escrito, devendo, por isso, o texto, se confiado a bibliotecas que assegurem a restrição, ser mantido em local sob chave ou custódia.

A classificação desta tese se encontra, abaixo, definida por seu autor.

Solicita-se aos depositários e usuários sua fiel observância, a fim de que se preservem as condições éticas e operacionais da pesquisa científica na área da administração.

Título da Tese: Proposta de um catálogo de falhas e mudanças em projetos: um estudo de caso a partir de uma empresa de desenvolvimento de *software*

Nome do Autor: Felipe Augusto Pereira

Data da defesa: 17/12/2014

Classificação, conforme especificação acima:

Grau 1 ☒

Grau 2 ☐

Grau 3 ☐

Recife, Dezembro de 2014.

Felipe Augusto Pereira

Proposta de um catálogo de falhas e mudanças em projetos: um estudo de caso a partir de uma empresa de desenvolvimento de *software*

Orientador: Marcos André Mendes Primo, Ph.D.

Tese apresentada como requisito complementar à obtenção do grau de Doutor em Administração, na área de concentração Gestão Organizacional, do Programa de Pós-Graduação em Administração da Universidade Federal de Pernambuco.

Recife, 2014

Catálogo na Fonte
Bibliotecária Ângela de Fátima Correia Simões, CRB4-773

P436p

Pereira, Felipe Augusto

Proposta de um catálogo de falhas e mudanças em projetos: um estudo de caso a partir de uma empresa de desenvolvimento de software / Felipe Augusto Pereira. - Recife : O Autor, 2014.
311 folhas : il. 30 cm.

Orientador: Prof. Marcos André Mendes Primo, PhD.

Tese (Doutorado em Administração) – Universidade Federal de Pernambuco, CCSA, 2014.

Inclui referências e apêndices.

1. Administração de projetos. 2. Desenvolvimento ágil de software. 3. Engenharia de software – Gerência. I. Primo, Marcos André Mendes (Orientador). II. Título.

658 CDD (22.ed.)

UFPE (CSA 2015 –029)

Universidade Federal de Pernambuco
Centro de Ciências Sociais Aplicadas
Departamento de Ciências Administrativas
Programa de Pós-Graduação em Administração - PROPAD

Proposta de um catálogo de falhas e mudanças em projetos:
um estudo de caso a partir de uma empresa de
desenvolvimento de *software*

Felipe Augusto Pereira

**Tese submetida ao corpo docente do Programa de Pós-Graduação em Administração da
Universidade Federal de Pernambuco e aprovada em 17 de dezembro de 2014.**

Banca Examinadora:

Prof. Marcos André Mendes Primo, Doutor, UFPE (Orientador)

Prof. Charles Ulisses de Montreuil Carmona, Doutor, UFPE (Examinador Interno)

Prof. Denis Silva da Silveira, Doutor, UFPE (Examinador Interno)

Prof. Alexandre Marcos Lins de Vasconcelos, Doutor, UFPE (Examinador Externo)

Prof. Marcos Roberto Piscopo, Doutor, UNINOVE (Examinador Externo)

Dedico esta tese a meus familiares e amigos mais próximos, especialmente a minha mãe, que tanto sonhou com esse momento, a minha esposa e filha. Vocês todos me inspiram.

“A curiosidade é a semente do conhecimento.”
Felipe Augusto Pereira

Agradecimentos

Agradeço a todos que, de alguma forma, contribuíram para que eu chegasse até aqui. A caminhada é longa e se cheguei ao fim, devo muito a cada um de vocês.

A minha mãe, Dilma, por todo seu amor, carinho, atenção, instrução, cobranças, apoio e suporte, além de compreensão pelos momentos de ausência.

A minha esposa Mari, filha Mayana e cão Lion, por todo o amor, carinho, cuidados, compreensão pelos momentos de ausência e apoio incondicional;

A meu pai, Augusto, pela preocupação, recomendações, cobranças e apoio;

A Sr. Carlos e Josi, meus pais em consideração, por seu apoio, suporte, conselhos e preocupação;

A tio Neném (*in memoriam*) e demais tios, primos e familiares, por todo o carinho;

Ao professor Marcos Primo, meu orientador do mestrado e doutorado, pela confiança, pelo apoio em todos os momentos deste trabalho, contribuições, paciência e franqueza, sendo imprescindível para todos os resultados obtidos e pelo exemplo de seriedade profissional.

Aos professores Alexandre, Carmona, Denis e Piscopo, por terem aceitado o convite para serem examinadores dessa tese e pelas valiosas contribuições feitas nas diversas etapas desse processo, desde o ensaio teórico.

Aos professores Fábio Silva e Walter Moraes, mestres que admiro bastante pelo exemplo de profissionais que são.

À Organização Alfa e todos os entrevistados, pelo tempo dedicado a participar do estudo.

Aos colegas e amigos do PROPAD: Daniel, Ari, Jorge, Cris, Renata, Yákara, Antônio, Paulo Rafael, Paulo Imbuzeiro, Rosivaldo, Vera, Sérgio, Viviane, Ronei, Eli, Lara, Dani, Irani, Fernando Paiva, Edilson, Moisés, Ana Lúcia, Tiago, Rita, Rafa, Geo, Thompson, Ana Paula, Sheila, Welkson, Carlos e demais colegas, por todos os momentos que compartilhamos juntos e pelo apoio.

A Walewsky, Roberto Guimarães, Luciana Gondim, Milton Souza, Raimundo Filho, Robert Cialdini e André Lima, pelo seu estímulo (sabendo ou não) para que eu concluísse o doutorado, nos momentos de maiores dificuldades e dúvidas.

A todos os membros e ex-membros do Grupo FAP, por toda a dedicação e por contribuírem para que os negócios tivessem continuidade mesmo na minha ausência.

Aos colegas e ex-colegas de trabalho da Estácio, Fernando Freyre, Leonardo Estevam, Rone, Patrícia, Kési, Érika, Luiz, Andrea Bemfica, Andrea Souza, Vanessa, Gustavo, Maurício Saraiva, Ijon e Zé Mario por compartilharmos de bons momentos juntos.

Aos amigos Andrey, André, Carla, Daniel Cardoso, Daniel Brasil, Didi, Gilson Jr., Marco, Mano (*in memoriam*), Mayara, Myrna, Fernando Félix e Cristina Balari, Sr. Valdeci, D. Bete, tio Gilson, tia Socorro, Gerson, Fagner, Dr. Vendas, Mateus, Tavares e Guilherme, pela amizade e momentos que passamos juntos.

A todos os clientes, pela compreensão pelos momentos de ausência.

A todos os demais que por acaso eu tenha esquecido nesse corre-corre para entregar essa tese.

A todos vocês o meu muito obrigado!

Resumo

Esta tese busca compreender como a ocorrência de falhas em projetos e as mudanças nas rotinas organizacionais de gestão de projetos em organizações de desenvolvimento de *software* se relacionam. Para tanto, foi realizado um estudo de caso instrumental, de natureza qualitativa, no qual foram realizadas entrevistas com membros da equipe, líderes, gerentes de projetos, de negócios e de operações de uma organização voltada para o desenvolvimento de *software*. Como principais resultados, foram identificados sete critérios de falhas (tempo, custo, escopo, qualidade, satisfação do cliente, satisfação da equipe e interrupção do projeto), fatores que levam a cada um desses critérios e os *stakeholders* e ocasiões envolvidos na detecção das falhas. Além disso, foram identificadas quinze rotinas organizacionais de gestão de projetos que sofrem mudanças ao longo do tempo e vinte episódios em que a ocorrência de falhas gerou mudanças nessas rotinas. Os resultados mostram que: i) as falhas estão associadas às mudanças nas rotinas, que ocorrem quando os agentes de mudança percebem que a mudança na rotina pode atuar sobre os fatores que levaram à ocorrência das falhas; ii) quanto maior o potencial de impacto das falhas maior a tendência à ocorrência de mudanças; iii) esse fenômeno depende do grau de maturidade da organização; iv) mudanças em rotinas que ocorrem em um projeto podem ocorrer em outros projetos, a depender de características do projeto, cliente, gerente, equipe e contexto.

Palavras-chave: Gestão de projetos de *software*. Falhas em projetos. Desempenho de projeto. Rotinas organizacionais. Mudanças em rotinas.

Abstract

This research seeks to understand how the occurrence of failures in projects is related with changes in project management organizational routines in *software* development organizations. An instrumental qualitative case study was conducted, and team members, leaders, project, business and operation managers were interviewed. As main results, seven failure criteria (time, cost, scope, quality, customer satisfaction, team satisfaction and termination of the project) were detected, and factors that lead to each of these criteria and stakeholders and occasions involved in detection of failures were identified. In addition, fifteen project management organizational routines that undergo changes over time and twenty episodes in which occurrence of failures led to changes in these routines were identified. The results show that: i) failures are associated with changes in routines, which occur when change agents perceive that the change in routine can act on the factors that led to the occurrence of failures; ii) the impact of the failure and the occurrence of routine change are associated; iii) this phenomenon depends on the degree of maturity of the organization; iv) changes in routines that occur in a project may occur in other projects, depending on the projects characteristics, project manager, client, team and context.

Keywords: *Software* project management. Project failures. Project performance. Organizational routines. Routines change.

Lista de Figuras

Figura 1 (2) – Tipologia para classificação de projetos (escopo e incerteza tecnológica).	33#
Figura 2 (2) – Tipologia para a classificação de projetos (localização e quantidade).	36#
Figura 3 (2) – Critérios de desempenho de projetos (1960-1980).	43#
Figura 4 (2) – Escopo do desempenho do projeto e do desempenho da gestão do projeto.	45#
Figura 5 (2) – Visão geral dos ciclos de vida do produto e do projeto, segundo o PMBOK.	46#
Figura 6 (2) – Importância relativa das dimensões de sucesso ao longo do tempo.	49#
Figura 7 (2) – Importância relativa das dimensões de sucesso de acordo com a incerteza tecnológica.	50#
Figura 8 (2) – Medindo sucesso ao longo dos ciclos de vida do projeto e do produto.	52#
Figura 9 (2) – Relação entre os aspectos ostensivo e performativo das rotinas organizacionais.	64#
Figura 10 (2) – Regras, rotinas e os <i>stakeholders</i> relevantes nas organizações.	67#
Figura 11 (2) – Efeitos da mensuração de desempenho nas rotinas organizacionais.	73#
Figura 12 (2) – Relação entre falhas e mudanças nas rotinas de gestão de projetos de <i>software</i> .	76#
Figura 13 (3) – Desenho metodológico da pesquisa.	78#
Figura 14 (3) – Etapas da revisão detalhada de literatura da pesquisa.	84#
Figura 15 (3) – Exemplo do mapa conceitual adotado para as falhas.	95#
Figura 16 (4) – Fatores associados ao critério de falha Tempo.	100#
Figura 17 (4) – Fatores associados ao critério de falha Custo.	106#
Figura 18 (4) – Fatores associados ao critério de falha Escopo.	110#
Figura 19 (4) – Fatores associados ao critério de falha Qualidade.	114#
Figura 20 (4) – Fatores associados ao critério de falha Satisfação do Cliente.	116#
Figura 21 (4) – Fatores associados ao critério de falha Satisfação da Equipe.	119#
Figura 22 (4) – Fatores associados ao critério de falha Interrupção de Projeto.	122#
Figura 23 (4) – Stakeholders envolvidos na percepção de falhas.	123#
Figura 24 (4) – Ocasões em que as falhas são percebidas.	125#
Figura 25 (4) – Rotina <i>Kickoff</i> e Acerto do Projeto.	136#
Figura 26 (4) – Rotina Alocação de Equipe (parte 1).	149#
Figura 27 (4) – Rotina Alocação de Equipe (parte 2).	150#
Figura 28 (4) – Rotina Definição do Escopo, Requisitos, Tarefas e Cronograma (parte 1).	165#
Figura 29 (4) – Rotina Definição do Escopo, Requisitos, Tarefas e Cronograma (parte 2).	166#
Figura 30 (4) – Rotina Controle de Cronograma, Tarefas e Impedimentos	174#
Figura 31 (4) – Rotina Reunião de Acompanhamento com o Cliente.	179#
Figura 32 (4) – Rotina Avaliação e <i>Feedback</i> para a Equipe.	185#
Figura 33 (4) – Rotina Reunião de Governança com o Cliente.	188#
Figura 34 (4) – Rotina Análise de Desempenho e Definição de Ações de Melhoria pela Equipe.	192#
Figura 35 (4) – Rotina Fechamento Financeiro.	194#
Figura 36 (4) – Rotina Controle de Qualidade do <i>Software</i> .	203#
Figura 37 (4) – Rotina Acompanhamento do Desempenho dos Projetos pelo PMO.	207#
Figura 38 (4) – Rotina Entrega para o Cliente.	211#
Figura 39 (4) – Rotina Pesquisa de Satisfação do Cliente.	214#
Figura 40 (4) – Rotina Encerramento do Projeto.	218#
Figura 41 (4) – Rotina Elaboração de relatório de prestação de contas.	220#
Figura 42 (4) – Resumo da mudança “Maior rigor na alocação de equipe após falha de tempo”.	226#

Figura 43 (4) – Resumo da mudança “Redefinição de equipe após falha de escopo e tempo”.	228#
Figura 44 (4) – Resumo da mudança “Acompanhamento próximo da equipe após falha de escopo e tempo”.	230#
Figura 45 (4) – Resumo da mudança “Alocação de equipe por mais tempo após falha de escopo, tempo e satisfação do cliente”.	232#
Figura 46 (4) – Resumo da mudança “Replanejamento do projeto e verificação de escopo após falha de escopo e tempo”.	234#
Figura 47 (4) – Resumo da mudança “Maior formalização das mudanças de escopo após falha de escopo e tempo”.	236#
Figura 48 (4) – Resumo da mudança “Redução do escopo após falha de satisfação do cliente”.	241#
Figura 49 (4) – Resumo da mudança “Maior integração da equipe nas tarefas após falha de escopo e tempo”.	243#
Figura 50 (4) – Resumo da mudança “Utilização de <i>buffers</i> de contingência após falha de escopo e tempo”.	246#
Figura 51 (4) – Resumo da mudança “Retorno de reuniões diárias após falha de escopo e tempo”.	248#
Figura 52 (4) – Resumo da mudança “Adoção do Jira após falha de satisfação do cliente”.	253#
Figura 53 (4) – Resumo da mudança “Indicação de indivíduo exclusivo para suporte a projetos em produção após falha de tempo”.	255#
Figura 54 (4) – Resumo da mudança “Realização de testes de desempenho por sprint ao invés de por release após falha de qualidade e tempo”.	258#
Figura 55 (4) – Resumo da mudança “Intensificação do <i>feedback</i> para a equipe após falha de satisfação da equipe”.	260#
Figura 56 (4) – Resumo da mudança “Realização de testes periódicos após falha de satisfação do cliente”.	262#
Figura 57 (4) – Resumo da mudança “Adoção do GIT após falha de satisfação da equipe”.	265#
Figura 58 (4) – Resumo da mudança “Mudança nas reuniões de acompanhamento com o cliente após falha de escopo e tempo”.	269#
Figura 59 (4) – Resumo da mudança “Mudança na definição de requisitos após falha de escopo, tempo e satisfação do cliente”.	272#
Figura 60 (4) – Resumo da mudança “Redefinição da equipe após falha de escopo, tempo e satisfação do cliente”.	274#
Figura 61 (4) – Resumo da mudança “Mudança na priorização de requisitos após falha de tempo”.	276#

Lista de Quadros

Quadro 1 (2) – Papéis típicos em uma equipe de desenvolvimento de <i>software</i>	32#
Quadro 2 (2) – Tipologia para a classificação de projetos (localização e unicidade)	37#
Quadro 3 (2) – Processos do PMBOK, de acordo com os grupos e áreas	38#
Quadro 4 (2) – Detalhamento do processo Plano de Gestão do Projeto	39#
Quadro 5 (2) – Gerenciamento de projetos tradicional x ágil	40#
Quadro 6 (2) – Eventos com duração fixa do <i>Scrum</i>	41#
Quadro 7 (2) – Critérios de desempenho de projetos (1980-2000)	44#
Quadro 8 (2) – Categorias de sucesso e seus elementos	47#
Quadro 9 (2) – Dimensões de sucesso e suas medidas	48#
Quadro 10 (2) – Dimensões de sucesso e suas medidas (novo modelo)	48#
Quadro 11 (2) – Foco x saída/resultado das dimensões de desempenho	51#
Quadro 12 (2) – Efetividade/eficiência x orientação temporal das dimensões de desempenho	51#
Quadro 13 (2) – Artigos que abordam o conceito de desempenho/falha em projetos de <i>software</i>	53#
Quadro 14 (2) – Comparação entre os níveis de capacidade e de maturidade	58#
Quadro 15 (2) – Comparação entre os níveis do MPS.BR e do CMMI	58#
Quadro 16 (2) – Atributos de Processo do MPS.BR	59#
Quadro 17 (2) – Processo do MPS.BR	59#
Quadro 18 (2) – Resultados esperados do processo Gerência de Projetos do MPS.BR	60#
Quadro 19 (2) – Algumas abordagens relacionadas a rotinas organizacionais	66#
Quadro 20 (2) – Estímulos a mudanças nas rotinas	69#
Quadro 21 (3) – Paradigmas da pesquisa em administração	79#
Quadro 22 (3) – Fases e estágios da revisão sistemática de literatura	82#
Quadro 23 (3) – Síntese do protocolo da revisão sistemática de literatura	85#
Quadro 24 (3) – Definição das perguntas do roteiro a partir dos objetivos específicos	90#
Quadro 25 (3) – Entrevistas	92#
Quadro 26 (3) – Categorias da análise qualitativa das entrevistas	94#
Quadro 27 (3) – Campos da planilha da associação entre falhas em projetos e mudanças nas rotinas	96#
Quadro 28 (4) – Critérios de falha resultantes da pesquisa	98#
Quadro 29 (4) – Fatores associados à ocorrência de falha	99#
Quadro 30 (6) – Ocasões em que as falhas são percebidas x critérios de falhas	296#

Lista de Abreviaturas e Siglas

ABES: Associação Brasileira das Empresas de *Software*
APK: *Android Application Package*
API: *Application Programming Interface*
APL: Arranjo Produtivo Local
ANS: Agência Nacional de Saúde Suplementar
ASD: *Adaptive Software Development*
AWS: *Amazon Web Services*
CMMI: *Capability Maturity Model Integration*
CR: *Change Request*
CRM: *Customer Relationship Management*
DBA: *Data Base Administrator*
DSDM: *Dynamic Systems Development Method*
EAP: Estrutura Analítica do Projeto
ERP: *Enterprise Resource Planning*
EVA: *Earned Value Analysis*
FCS: Fatores Críticos de Sucesso
FDD: *Feature Driven Development*
GP: Gerente de Projetos
GPR: Gerência de Projetos
HD: *Hard disk*
IDC: *International Data Corporation*
IS: *Information System*
ISD: *Information System Development*
LA: Lições Aprendidas
MPS.BR: Melhoria de Processos do *Software* Brasileiro
NGPD: Núcleo de Gestão Porto Digital
PBI: *Product Item Backlog*
PD: Porto Digital
PKI: *Public Key Infrastructure*
PLM: *Product Lifecycle Management*
PMBOK: *Project Management Body of Knowledge*
PMI: *Project Management Institute*
PMO: *Project Management Office*
PO: *Product Owner*
POS: *Point Of Sale*
PT: Plano de Trabalho
RH: Recursos Humanos
ROE: *Return On Equity*
ROI: *Return On Investment*
SI: Sistema(s) de Informação
SLA: *Service Level Agreement*
SUS: Sistema Único de Saúde
T&M: Tempo e Material
TDD: *Test-Driven Development*
TI: Tecnologia da Informação
TISS: Troca de Informação em Saúde Complementar
TIC: Tecnologia da Informação e Comunicação
UFPE: Universidade Federal de Pernambuco

WBS: *Work Breakdown Structure*
XP: *eXtreme Programming*

Sumário

1#	Introdução	18#
1.1#	Problema de pesquisa	26#
1.2#	Objetivos	26#
1.3#	Justificativa e contribuições	27#
1.4#	Estrutura da tese	30#
2#	Referencial conceitual e teórico	31#
2.1#	Projetos de desenvolvimento de <i>software</i>	31#
2.1.1#	Tipos de projetos de desenvolvimento de <i>software</i>	33#
2.1.2#	Gestão de projetos de desenvolvimento de <i>software</i>	37#
2.1.3#	Falhas em projetos de desenvolvimento de <i>software</i>	42#
2.1.4#	Modelos de maturidade em organizações de desenvolvimento de <i>software</i>	57#
2.2#	Rotinas organizacionais	62#
2.2.1#	Mudanças nas rotinas organizacionais	67#
2.2.2#	Rotinas organizacionais de gestão de projetos de <i>software</i> e suas mudanças	71#
2.3#	A relação entre falhas e mudanças nas rotinas de gestão de projetos de <i>software</i>	73#
3#	Procedimentos Metodológicos	78#
3.1#	Delineamento da pesquisa	78#
3.2#	Revisão da Literatura	80#
3.2.1#	Revisão Preliminar da Literatura	81#
3.2.2#	Revisão Posterior da Literatura	82#
3.3#	Seleção do caso e unidade de análise	86#
3.4#	Elaboração do Instrumento de Coleta	89#
3.5#	Coleta de Dados	91#
3.6#	Análise dos dados	93#
4#	Resultados	98#
4.1#	Falhas em projetos de <i>software</i>	98#
4.1.1#	Critérios de falhas e fatores relacionados	98#
4.1.1.1#	Tempo	100#
4.1.1.2#	Custo	105#
4.1.1.3#	Escopo	110#
4.1.1.4#	Qualidade	114#
4.1.1.5#	Satisfação do Cliente	116#
4.1.1.6#	Satisfação da Equipe	119#
4.1.1.7#	Interrupção de Projeto	122#
4.1.2#	Identificação da falha	123#
4.2#	Rotinas de gestão de projetos de <i>software</i> e suas mudanças	130#
4.2.1#	<i>Kickoff</i> e acerto do plano de projeto	131#
4.2.2#	Alocação de equipe	137#
4.2.3#	Definição do escopo, requisitos, tarefas e cronograma	150#
4.2.4#	Controle de cronograma, tarefas e impedimentos	166#
4.2.5#	Reunião de acompanhamento com o cliente	174#
4.2.6#	Avaliação e <i>feedback</i> para a equipe	179#
4.2.7#	Reunião de governança com o cliente	186#
4.2.8#	Análise de desempenho e definição de ações de melhoria pela equipe	188#
4.2.9#	Fechamento financeiro	192#
4.2.10#	Controle de qualidade do <i>software</i>	195#
4.2.11#	Acompanhamento do desempenho dos projetos pelo PMO	204#

4.2.12#	Entrega para o cliente	207#
4.2.13#	Pesquisa de satisfação do cliente	211#
4.2.14#	Encerramento do projeto	214#
4.2.15#	Elaboração de relatório de prestação de contas	218#
4.3#	Relação entre falhas em projetos e mudanças em rotinas organizacionais de gestão de projetos de <i>software</i>	221#
4.3.1#	Maior rigor na alocação de equipe após falha de tempo	222#
4.3.2#	Redefinição de equipe após falha de escopo e tempo	226#
4.3.3#	Acompanhamento próximo da equipe após falha de escopo e tempo	229#
4.3.4#	Alocação de equipe por mais tempo após falha de escopo, tempo e satisfação do cliente	231#
4.3.5#	Replanejamento do projeto e verificação de escopo após falha de escopo e tempo	233#
4.3.6#	Maior formalização das mudanças de escopo após falha de escopo e tempo	234#
4.3.7#	Redução do escopo após falha de satisfação do cliente	237#
4.3.8#	Maior integração da equipe nas tarefas após falha de escopo e tempo	241#
4.3.9#	Utilização de <i>buffers</i> de contingência após falha de escopo e tempo	243#
4.3.10#	Retorno de reuniões diárias após falha de escopo e tempo	246#
4.3.11#	Adoção do Jira após falha de satisfação do cliente	248#
4.3.12#	Indicação de indivíduo exclusivo para suporte a projetos em produção após falha de tempo	253#
4.3.13#	Realização de testes de desempenho por <i>sprint</i> ao invés de por <i>release</i> após falha de qualidade e tempo	256#
4.3.14#	Intensificação do <i>feedback</i> para a equipe após falha de satisfação da equipe	258#
4.3.15#	Realização de testes periódicos após falha de satisfação do cliente	260#
4.3.16#	Adoção do GIT após falha de satisfação da equipe	262#
4.3.17#	Mudança nas reuniões de acompanhamento com o cliente após falha de escopo e tempo	266#
4.3.18#	Mudança na definição de requisitos após falha de escopo, tempo e satisfação do cliente	269#
4.3.19#	Redefinição da equipe após falha de escopo, tempo e satisfação do cliente	272#
4.3.20#	Mudança na priorização de requisitos após falha de tempo	274#
5#	Discussão dos resultados	277#
5.1#	Falhas em projetos de <i>software</i>	277#
5.2#	Rotinas organizacionais de gestão de projetos de <i>software</i> e suas mudanças	283#
5.3#	Relação entre falhas e mudanças nas rotinas de gestão de projetos de <i>software</i>	287#
6#	Conclusões	294#
6.1#	Falhas em projetos de <i>software</i>	294#
6.2#	Rotinas organizacionais de gestão de projetos de <i>software</i> e suas mudanças	297#
6.3#	Relação entre falhas e mudanças nas rotinas de gestão de projetos de <i>software</i>	298#
6.4#	Limitações	301#
6.5#	Sugestões de trabalhos futuros	302#
	Referências	304#
	Apêndice A – Roteiro de Entrevistas	317#

1 Introdução

Desde o surgimento dos primeiros computadores, a tecnologia da informação (TI) tem apresentado uma crescente relevância na sociedade. Compreendida como os equipamentos, aplicações e serviços usados para distribuir dados, informação e conhecimento (LUFTMAN et al., 1993), a TI evoluiu de um papel tradicional de suporte administrativo a uma posição estratégica nas organizações (LAURINDO et al., 2001).

Tal importância é refletida na crescente movimentação do mercado de TI no mundo. Estudos do *International Data Corporation* (IDC) apontam grandes movimentações anuais, que passaram de US\$ 2 trilhões em 2013 (ABES, 2014). Segundo o estudo, a América Latina movimentou US\$ 130 bilhões, dos quais US\$ 61,6 bilhões foram participação do Brasil, o que corresponde a 47,4%. Esse percentual faz com que o Brasil seja o principal *player* latino-americano. O mercado nacional de *software* e serviços movimentou US\$ 25,1 bilhões, fazendo com que o país assumisse o 8º lugar no *ranking* mundial (ABES, 2014).

Em nível regional e local, a TI também tem um importante papel. O Governo de Pernambuco (2013) aponta a informática como um dos setores de destaque no estado. O Porto Digital (PD), aglomerado de empresas de TI da cidade de Recife, foi apontado como o maior parque tecnológico do País em número de empresas e faturamento pela AT Kearney e eleito, em 2011, como o Melhor Parque Tecnológico/Habitat de Inovação do Brasil pela Anprotec (PORTO DIGITAL, 2014).

Um relatório do perfil do aglomerado em 2010 realizado pela Datamétrica (2011) aponta a presença de 191 empresas, sendo 147 empresas de Tecnologia da Informação e Comunicação (TIC), 16 incubadas, 15 de economia criativa e 13 serviços associados. Segundo a pesquisa, esses empreendimentos movimentavam quase R\$ 900 milhões por ano e empregavam mais de 6000 funcionários, com salário médio superior a R\$ 2.600,00 (DATAMÉTRICA, 2011). Atualmente, o *site* do Porto Digital informa que o APL emprega 7 mil funcionários e possui 230 empresas (PORTO DIGITAL, 2014). Grande parte delas atua na produção de *software*, caracterizando o Porto Digital como referência nacional na atividade.

As organizações de desenvolvimento de *software*, em geral, são voltadas a projetos, que consistem em um “esforço temporário empreendido para criar um produto, serviço ou resultado único” (PMI, 2013, p. 3, tradução nossa). Pela definição, é possível observar que

projetos se diferenciam de processos operacionais continuados principalmente por dois fatores. Em primeiro lugar, os projetos não são contínuos, ou seja, são temporários, com início e fim definidos. Por conta dessa duração temporária, o time responsável pelo desenvolvimento é desfeito ao seu término e seus membros são alocados para outros projetos da organização executora. Além disso, enquanto os processos operacionais continuados têm resultados repetitivos, os projetos geram produtos, serviços e resultados únicos que, mesmo se semelhantes a anteriores, possuem suas especificidades (PMI, 2013).

Os projetos podem variar de acordo com diversos critérios, como tamanho/duração (CHOW; CAO, 2008; EL-EMAN; KORU, 2008; HARRIS et al., 2009; PAPKE-SHIELDS et al., 2010), incerteza tecnológica (SHENHAR, 1998; MCLEOD, MACDONELL, 2011), tipo/escopo do sistema (SHENHAR, 1998; CHOW; CAO, 2008; MCLEOD, MACDONELL, 2011), local em que ocorrem e existência de outros projetos (EVARISTO; FENEMA, 1999; HUMMEL et al., 2013). Diferenças em sua gestão são necessárias de acordo com suas características (SHENHAR, 1998; EVARISTO; FENEMA, 1999; PAPKE-SHIELDS et al., 2010; SIAU et al., 2010).

A temporalidade e unicidade dos projetos são fontes de dificuldades para sua implementação. Os projetos de uma mesma organização são diferentes uns dos outros, pois possuem contextos, requisitos, clientes e equipe distintos. Desse modo, é muito comum que ocorram falhas (THE STANDISH GROUP, 1995, 2013; LINBERG, 1999; EL-EMAM; KORU, 2008; DALCHER, 2009; TARAWNEH, 2011), que podem ser compreendidas como “uma diferença inaceitável entre o desempenho previsto e o observado” (PETROSKI, 2006, p. 51, tradução nossa). Desse modo, projetos que sejam abortados antes de seu término ou que não atendam aos requisitos de escopo, qualidade, tempo e custo estimados, por exemplo, constituem falhas.

Nem sempre o desempenho de projetos é tão diretamente observável e estudos mostram que sucesso e falhas são construtos complexos. Sua percepção varia de acordo com o ator que está analisando (WATERIDGE, 1998; LINBERG, 1999; STANDING et al., 2006; PEREIRA et al., 2008; SAVOLAINEN et al., 2012). Um projeto com valor fixo pago pelo cliente que foi concluído dentro do cronograma mas excedendo o orçamento previsto pela empresa de desenvolvimento pode ser considerado um sucesso para o contratante e uma falha para a contratada. Um projeto concluído dentro dos requisitos de escopo, tempo e custo, mas que foi um fracasso do ponto de vista comercial, pode ser considerado um sucesso pelo gerente de projetos e sua equipe e uma falha pela alta gestão da empresa.

Além disso, o sucesso/falha precisam ser compreendidos a partir de múltiplas dimensões (ATKINSON, 1999; SHENHAR et al., 2001; SHENHAR; DVIR, 2007; THOMAS; FERNÁNDEZ, 2008; DALCHER, 2009; SIAU et al., 2010; MCLEOD; MACDONNEL, 2011). Projetos que terminem sem atender ao tempo e custo estimados podem ser um sucesso de vendas, e não podem ser considerados, portanto, integralmente mal sucedidos. Do mesmo modo, um projeto que não gerou retorno financeiro para a organização que o desenvolveu pode ter proporcionado a ela o desenvolvimento de capacidades tecnológicas que, aplicadas em oportunidades posteriores, foram de grande retorno. Assim, tais projetos também não podem ser considerados um insucesso como um todo.

Isso remete à visão contemporânea de desempenho de projetos, em que o sucesso e falha são tratados de modo multidimensional. Atkinson (1999) propõe quatro categorias: triângulo de ferro, o sistema de informação, benefícios para a organização e benefícios para a comunidade de *stakeholders*. A primeira delas corresponde a custo, qualidade e tempo, as métricas tradicionais ligadas ao sucesso da gestão de projetos. A segunda categoria diz respeito a características ligadas à qualidade do SI desenvolvido: manutenibilidade, confiabilidade, validade e qualidade da informação. As duas outras correspondem a benefícios para a organização que desenvolve o *software* (aumento da eficiência, efetividade, lucro, aprendizagem e redução do desperdício) e outros *stakeholders* (satisfação dos usuários, impacto social e ambiental, desenvolvimento pessoal e profissional, lucro do contratante e impacto econômico na sociedade). Como pode ser observado, as categorias vão de uma perspectiva micro, ligada à gestão do projeto em si, a uma macro, que aborda outros elementos.

Shenhar et al. (2001) propõem quatro dimensões: eficiência do projeto, impacto no consumidor, sucesso empresarial e preparação para o futuro. Seis anos depois, Shenhar e Dvir (2007) adaptam o modelo e apresentam cinco dimensões: eficiência (cronograma, orçamento, lucratividade e eficiência na produção), impacto no consumidor (funcionalidades, especificações, benefícios para o cliente, uso pelo cliente, satisfação e lealdade do cliente e reconhecimento da marca), impacto na equipe (satisfação e moral da equipe, desenvolvimento de habilidades, crescimento dos membros, retenção e ausência de exaustão), sucesso empresarial (vendas, lucro, *market share*, ROI, ROE, fluxo de caixa, medidas organizacionais e aprovação regulamentar) e preparação para o futuro (novas tecnologias, novos mercados, nova linha de produtos, nova competência e nova capacidade organizacional). O modelo não foi proposto especificamente para *software*, mas alguns anos depois Dalcher (2009) analisou dimensões baseadas nessas para esse tipo de projeto.

Além da dimensão em que a falha ocorre, estas também variam de acordo com a gravidade e frequência (HORA, 2008). A gravidade está relacionada às consequências da falha. No caso de *software*, pode variar desde deficiências que não impedem seu funcionamento até um problema sério que gere um desastre (FLOWERS, 1996). Isso depende tanto da grandeza da falha ocorrida quanto da aplicação do *software*. Falhas em sistemas críticos ligados a segurança ou aeronaves, por exemplo, podem ter consequências não apenas para a organização que o desenvolveu e para seu cliente, mas também para vários outros indivíduos. No tocante à frequência, elas podem ser classificadas em um contínuo que varia de baixa (falhas raras) a alta (falhas frequentes, que acontecem em vários projetos).

Um estudo de The Standish Group (1995) visando à identificação do escopo das falhas de projetos de *software*, dos principais fatores de falhas e dos elementos capazes de reduzi-las, classificou os projetos analisados de acordo com o seu resultado. Os projetos completados no cronograma e orçamento estimados, com todas as características e funcionalidades inicialmente previstas foram classificados como bem-sucedidos. Aqueles concluídos e em operação, mas que apresentaram menos funcionalidades que o especificado, ou estouro de orçamento ou cronograma foram denominados ameaçados. Já os projetos abandonados em algum ponto do desenvolvimento foram classificados como abortados.

Os resultados obtidos no estudo mostram que 31,1% dos projetos foram abortados, 52,7% dos projetos foram ameaçados e apenas 16,2%, bem-sucedidos. Somando-se os projetos abortados e ameaçados, observa-se que 83,8% dos projetos foram mal sucedidos. Entre as razões apontadas por The Standish Group (1995) para o insucesso podem ser citadas: falta de envolvimento com o usuário, requisitos e especificações incompletos, mudanças nos requisitos e especificações, falta de suporte executivo, falta de recursos e estimativas não realistas.

Nos anos que sucederam esse estudo, as metodologias de gestão de projetos e desenvolvimento de sistemas evoluíram bastante. Várias abordagens ágeis se popularizaram, devido à percepção de que um dos motivos que ocasionavam falhas nos projetos era a incapacidade de atender aos seus requisitos, devido a problemas na identificação dos mesmos ou a mudanças. Esse movimento fortaleceu-se após o Manifesto Ágil (2001), iniciativa que elencou valores e princípios para fundamentar o desenvolvimento ágil de *software*. Entre elas, pode-se citar: *Scrum*, *eXtreme Programming* (XP) e *Lean Software Development*. Além disso, o PMBOK, tradicional *framework* para gestão de projetos, lançou quatro revisões e melhorias, em 2000, 2004, 2008 e, recentemente, em 2013.

Com todas essas mudanças, a taxa de sucesso dos projetos de *software* aumentou, mas falhas continuaram frequentes. Em novo relatório, com base em uma pesquisa realizada dezoito anos após o anterior, The Standish Group (2013) apresentou os seguintes resultados: 18% dos projetos foram abortados, 43% ameaçados e 39% bem-sucedidos. Somando-se os projetos mal sucedidos, chega-se a uma taxa de 61%, de modo que, apesar da evolução nas metodologias de gestão e desenvolvimento, mais de três quintos dos projetos de desenvolvimento de *software* ainda não foram bem-sucedidos. Essa alta taxa de insucesso gera grandes prejuízos para as organizações de desenvolvimento de *software*, seus clientes e a sociedade como um todo.

Na literatura, vários fatores são citados como influenciadores do desempenho de projetos. Entre eles, estão a definição do projeto, especificações, tarefas e técnicas de gestão, apoio da alta gestão, envolvimento com o usuário, recursos necessários, estimativas adequadas (THE STANDISH GROUP, 1995; CHOW; CAO, 2008; EL-EMAM; KORU, 2008; MCLEOD; MACDONNEL, 2011; TARAWNEH, 2011), processos de desenvolvimento (CHOW; CAO, 2008; TESCH et al., 2009; HUMMEL et al., 2013), experiência do gerente de projetos (TARAWNEH, 2011), mudanças nos requisitos e escopo e tecnologias que não funcionaram como previsto (EL-EMAM; KORU, 2008), características da organização (CHOW; CAO, 2008; MCLEOD; MACDONNEL, 2011) e do ambiente em que ela está inserida (SIAU et al., 2010; MCLEOD; MACDONNEL, 2011).

Por conta desses fatores, organizações voltadas a projetos vivenciam falhas em suas operações, algo que se repete em vários projetos, pois elas têm dificuldade em aprender a partir de projetos anteriores (JULIAN, 2008). Segundo Lyytinen e Robey (1999), há uma carência na aprendizagem a partir da experiência por parte das organizações que atuam com desenvolvimento e elas “aprendem a falhar”.

Um importante componente neste contexto são as rotinas destas organizações, que podem ser compreendidas como padrões de comportamento de seus indivíduos (NELSON; WINTER, 1982). Elas incorporam o conhecimento organizacional (BECKER, 2004; MILAGRES, 2011) e estão associadas com o desempenho das organizações associadas a projetos (PAPKE-SHIELDS et al., 2010; SIAU et al., 2010).

As rotinas possuem dois aspectos: o ostensivo e o performativo. O aspecto ostensivo está associado à estrutura geral da rotina, que permite identificá-la como tal (FELDMAN; PENTLAND, 2003). Desse modo, se um gerente de projetos for questionado com relação ao que é o processo de estimativa de tempo de uma atividade, ele informará uma sequência de atividades que ele compreende como típicas dessa rotina. Essa resposta constitui o seu aspecto

ostensivo. O aspecto performativo consiste em uma instância da rotina em execução (FELDMAN; PENTLAND, 2003). Assim, ele pode divergir do aspecto ostensivo por conta de uma situação excepcional, por exemplo.

Estas rotinas de gestão de projetos podem ter origens diversas, como o PMBOK, abordagens ágeis, métodos mistos ou *ad-hoc*. O grau de evolução dos processos caracteriza a maturidade da organização (SOFTEX, 2012), que pode adotar modelos de melhoria como o *Capability Maturity Model Integration* (CMMI) ou o Melhoria de Processos do *Software* Brasileiro (MPS.BR). Esses modelos definem os graus de maturidade da organização e fornecem subsídios para mensuração e implementação de melhorias nos processos. Quanto maior a maturidade, maior o grau de formalização e padronização dos processos. Em baixos níveis de maturidade, várias atividades são feitas de forma *ad hoc*, enquanto em níveis mais altos há um maior grau de disciplina que leva a maior qualidade (CMMI, 2010).

As rotinas trazem vantagens para a organização que as adota, dentre as quais estão: aumento da coordenação e do controle, minimização de conflitos, redução da incerteza e economia de recursos cognitivos (BECKER, 2004; MILAGRES, 2011). Além disso, são apontadas como elementos chave das capacidades organizacionais (GRANT, 1991; PENG et al., 2008), responsáveis pelo desempenho das organizações.

Como as rotinas codificam o conhecimento organizacional (BECKER, 2004; MILAGRES, 2011), elas são um componente chave da aprendizagem (LEVITT; MARCH, 1988), que envolve a assimilação e utilização de conhecimentos e ocorre em três níveis: individual, grupal (diretamente ligado às rotinas) e organizacional (CROSSAN et al., 1999). Na medida em que o tempo passa e os indivíduos desempenham suas atividades, as rotinas estão sujeitas a mudanças, que são indicadores da ocorrência de aprendizagem (FLEURY; FLEURY, 2001).

Situações específicas podem gerar a necessidade de alterações na forma como determinadas ações são desempenhadas. É comum, por exemplo, que organizações que utilizam *Scrum* (metodologia ágil para gestão de projetos) mantenham registro do andamento das atividades em um quadro branco com colunas que representam o grau de realização das tarefas (pendentes, em andamento, aguardando verificação e concluídas). Cada tarefa é representada por um *Post-it* (papel adesivo para pequenas anotações), que é inserido na primeira coluna no início do projeto e, no seu decorrer, avança pelo quadro.

Suponha que, em determinado momento, parte da equipe vai atuar de forma remota, executando suas atividades em um local distinto do restante do grupo. Por conta disso, o quadro branco torna-se inadequado para a realidade da organização, que o substitui por um

aplicativo na Internet. A nova ferramenta permite que toda a equipe tenha acesso a partir de qualquer local e é adequada à situação específica.

Ao término desse projeto, o gerente assume outro projeto e resolve iniciá-lo diretamente com o aplicativo *online* ao invés do quadro. O mesmo pode ocorrer em outros projetos, por indicação de pessoas que atuaram como engenheiros de *software* ou analistas de sistemas no projeto anterior. A partir do uso sucessivo do aplicativo juntamente com pequenas mudanças em outros aspectos, após algum tempo (meses ou anos, por exemplo), a rotina organizacional de acompanhamento das tarefas de toda a organização passa a ser consideravelmente distinta da anterior. Assim, as rotinas podem evoluir de forma incremental e são, portanto, capazes de se adaptar a novos contextos (BECKER, 2004), a partir de uma mudança endógena (FELDMAN; PENTLAND, 2003) que emergiu naturalmente durante o desempenhar da rotina.

Na situação descrita, outra possibilidade seria a organização possuir iniciativas de gestão do conhecimento através de um repositório ou sessões de compartilhamento de lições aprendidas. Nesse cenário, haveria uma tendência da mudança ocorrer de forma mais efetiva e abrangente na organização.

Em outras ocasiões, a mudança nas rotinas organizacionais pode ocorrer por influência externa. Uma organização que faz parte do Porto Digital, por exemplo, pode estar habituada a implementar seus projetos de forma *ad-hoc*, com gestão incipiente e intuitiva. Em determinado momento, o Núcleo de Gestão do Porto Digital (NGPD) organiza uma capacitação coletiva em MPS.BR. Ao término da capacitação, a organização que atua com desenvolvimento decide obter certificação no modelo e, para isso, reformula completamente seus processos de gestão de projetos.

Em outras situações, as mudanças nas rotinas podem ser fruto da avaliação de resultados. Em sua tese de doutorado, Avery (1996) analisou a influência do *feedback* para o desenvolvimento e persistência das rotinas em grupo. Sua pesquisa foi baseada em um experimento com 48 trios de estudantes e ressaltou o papel do *feedback* na rotinização das atividades, diferenciando o *feedback* positivo e o negativo.

No contexto de desenvolvimento de *software*, determinada organização pode estar fazendo avaliação de desempenho de seus projetos e observar que critérios como tempo ou escopo não estão correspondendo às expectativas. A partir desse *feedback*, os gestores podem decidir realizar mudanças nas rotinas de gestão de projetos, adequando-as ao que imaginam que trará melhores resultados. Assim, as rotinas podem evoluir a partir de mudanças

disparadas por avaliações de desempenho, em seu papel de *feedback* (PAVLOV; BOURNE, 2011).

Nas situações de mudanças nas rotinas organizacionais, elas são alteradas por um estímulo e, para que a mudança ocorra, os indivíduos envolvidos nas situações acreditam que a atividade poderá ser realizada de forma mais adequada com o novo processo (KACMAR et al., 2009; BELANGER et al., 2012). Caso contrário, a mudança não ocorreria, exceto por pressões institucionais, em um cenário coercitivo ou normativo, por exemplo.

Isso mostra o papel da agência no contexto, dado que muitas ações dos indivíduos são realizadas deliberadamente (FELDMAN; PENTLAND, 2003). Segundo essa abordagem, os membros da equipe possuem o poder de definir como agirão, a despeito do que está escrito nos procedimentos operacionais padrões, fluxogramas e demais artefatos relativos ao aspecto ostensivo das rotinas. Essa perspectiva, crescente nos estudos organizacionais, não é encontrada nas pesquisas ligadas a operações, qualidade e gestão de projetos.

Nesse contexto, falhas e crises em organizações podem estar ligadas à criação e alteração de rotinas (INAM, 1999; ELLSTRÖM, 2010; DUHIGG, 2012). Dois estudos que abordam a reação de indivíduos envolvidos em projetos à ocorrência de falhas foram realizados por Dean Shepherd e outros autores (SHEPHERD; CARDON, 2009; SHEPHERD et al., 2009). Eles ressaltam que falhas em projetos podem gerar uma resposta emocional negativa aos envolvidos e esse sentimento influencia o processo de aprendizagem. Tais indivíduos podem perceber que estão passando por uma experiência negativa, gerada pela falha em um projeto, identificar a causa dessa sensação e agir para evitá-la (SHEPHERD; CARDON, 2009). Dependendo do perfil do indivíduo, ele pode internalizar as falhas como normais ou encará-las e a postura apresentada influencia diretamente o aprendizado.

Desse modo, a ocorrência de falhas em projetos, ou a percepção de que esta ocorrerá, tem um grande potencial para despertar alterações nas rotinas da organização executora. Associadas à mensuração de desempenho, elas podem agir como gatilhos, guia para as ações ou intensificar as mudanças nas rotinas (PAVLOV; BOURNE, 2011). Assim, a ocorrência de mudanças nas rotinas organizacionais de gestão de projetos deve sofrer influência de tais falhas, fenômeno que não é abordado na literatura e é objeto desta pesquisa de doutorado.

Segundo a tradição da área de qualidade e gestão de projetos, que é predominantemente técnica, o desempenho das equipes que desenvolvem *software* resulta dos processos definidos nos procedimentos e manuais de qualidade. Assume-se que regras definidas de modo racional tendem a gerar processos de qualidade e pouca margem é dada à autonomia dos indivíduos envolvidos na execução das ações.

Diante disso, estudos como o de Feldman e Pentland (2003), Pentland e Feldman (2005) e Becker (2004) podem contribuir para a melhor compreensão da relação entre falhas em projetos de *software* e mudanças nas rotinas de gestão de projetos. A importância dessa perspectiva de rotinas tem aumentado, mas há poucos estudos empíricos tomando-as como base (BECKER; ZIRPOLI, 2008).

Além disso, a literatura carece de estudos que estudem a relação entre a ocorrência de mudanças nas rotinas organizacionais de gestão de projetos associadas a falhas nos projetos de desenvolvimento de *software*, relação que sofre influência de alguns aspectos. Características do projeto podem influenciar os processos de desenvolvimento (HARRIS et al., 2009; HUMMEL et al., 2013), bem como a percepção dos indivíduos envolvidos no projeto com relação ao que constitui uma falha (CHOW; CAO, 2008; MCLEOD; MACDONELL, 2011). Tipos de falhas diferentes podem ser percebidas de formas distintas pelos indivíduos (SHEPHERD; CARDON, 2009), de modo que a dimensão, frequência e gravidade das falhas podem influenciar as mudanças nas rotinas de forma diferente. Por fim, a maturidade da organização indica o grau de evolução dos processos, ou seja, em um nível de pouca maturidade, o baixo grau de formalização leva a rotinas menos rígidas (CMMI, 2010).

1.1 Problema de pesquisa

Diante do exposto, a pergunta de pesquisa a ser investigada é: como as falhas em projetos e as mudanças nas rotinas organizacionais de gestão de projetos de *software* se relacionam? Espera-se por meio desta pesquisa comprovar a hipótese de que a ocorrência de falhas em projetos altera as rotinas organizacionais de gestão de projetos das organizações de desenvolvimento de *software* e que essa relação varia de acordo com fatores do contexto.

1.2 Objetivos

A pesquisa tem como objetivo geral compreender como as falhas em projetos e as mudanças nas rotinas organizacionais de gestão de projetos em organizações de desenvolvimento de *software* se relacionam. Para alcançá-lo, devem ser atingidos os seguintes objetivos específicos:

- OE1: compreender falhas em projetos de *software*;
- OE2: compreender rotinas organizacionais de gestão de projetos em organizações de desenvolvimento de *software* e suas mudanças;

- OE3: analisar a relação entre a ocorrência de falhas e as mudanças ocorridas nas rotinas de gestão de projetos de desenvolvimento de *software*, identificando fatores que influenciam essa associação.

1.3 Justificativa e contribuições

Esta pesquisa se justifica do ponto de vista acadêmico por vários aspectos. Segundo Ewusi-Mensah (1997), examinar falhas em projetos de desenvolvimento de sistemas é importante para o avanço na prática de sistemas de informação (SI). Não há consenso em sua definição e estudos mostram que a própria compreensão varia de acordo com o ator em consideração (WATERIDGE, 1998; LINBERG, 1999; STANDING et al., 2006; PEREIRA et al., 2008; SAVOLAINEN et al., 2012).

Outra justificativa para esta pesquisa está relacionada à compreensão das rotinas organizacionais. Apesar delas serem centrais nas organizações burocráticas (FELDMAN; PENTLAND, 2003), estudos empíricos tomando-as como base são relativamente raros na academia (BECKER; ZIRPOLI, 2008). Essa necessidade tem sido ressaltada em grandes eventos internacionais. A chamada da Quarta Conferência Internacional em Rotinas Organizacionais, realizada em 2010 em Nice, França, aponta que são necessários estudos empíricos para compreender “quais são as condições que disparam mudanças ou estabilidades nas rotinas organizacionais e como nós as observamos” (LAZARIC, 2010, tradução nossa).

Diante dessa necessidade exposta pela própria academia, esta pesquisa traz uma importante contribuição para esse fenômeno, especificamente no contexto de organizações que desenvolvem *software*. Segundo Bresnen et al. (2005), as organizações voltadas a projetos apresentam um desafio para compreender e efetuar mudanças em suas rotinas, devido à complexidade e dinamicidade envolvidos. O fato dos projetos serem iniciativas temporárias ao final dos quais as equipes são reestruturadas torna a dinâmica das organizações orientadas a projetos peculiar. Em algumas situações, rotinas que eram realizadas pela equipe de determinado projeto podem deixar de existir simplesmente porque ele chegou ao fim e os membros envolvidos foram distribuídos para outros projetos que vão iniciar ou já estão em andamento com suas rotinas próprias. Em outros cenários, rotinas de um projeto que está em desenvolvimento podem sofrer alterações devido ao ingresso de um indivíduo que saiu de um projeto recém concluído.

Além disso, no segmento de desenvolvimento de *software*, o grau de incerteza tecnológica muitas vezes envolvida e o desconhecimento e mudanças nos requisitos dos

usuários aumentam a incerteza e a complexidade. Esses aspectos aumentam a chance de ocorrerem falhas nos projetos e, conseqüentemente, pode haver uma tendência a mudança nas rotinas para contornar esses problemas.

A principal “matéria-prima” das organizações que desenvolvem *software* é o conhecimento e “a rotinização da atividade em uma organização constitui a mais importante forma de armazenamento de seu conhecimento operacional específico” (NELSON; WINTER, 1982, p. 99, tradução nossa). Nessas organizações, os processos de desenvolvimento de sistemas constituem essas rotinas, que estão diretamente associadas com seu desempenho (PAPKE-SHIELDS et al., 2010; SIAU et al., 2010). A incorporação do conhecimento de gestão em organizações voltadas a projetos é problemática (BRESNEN et al., 2004, 2005), conseqüentemente contribuir na investigação desse fenômeno é relevante para a academia.

Outra contribuição do ponto de vista acadêmico reside na compreensão do processo de mudança em rotinas a partir de falhas em projetos de desenvolvimento de *software*. A literatura aponta que falhas e crises estão associadas à criação e mudanças nas rotinas (INAM, 1999; ELLSTRÖM, 2010; DUHIGG, 2012), mas o fenômeno tem sido pouco investigado no contexto de projetos de desenvolvimento de *software*.

Shepherd e Cardon (2009) e Shepherd et al. (2009) analisam a relação entre a ocorrência de falhas em projetos e a aprendizagem, ressaltando as respostas que os indivíduos possuem quando passam por uma situação de falha. Segundo os autores, eles podem reagir para evitar o sentimento negativo causado por uma falha, dependendo do perfil do indivíduo. Embora tragam interessantes contribuições para a compreensão da ocorrência de mudanças a partir de falhas no contexto de projetos, os dois artigos abordam uma perspectiva micro, individual. A unidade de análise é o indivíduo e seu comportamento diante da falha; a aprendizagem em nível organizacional e mudanças em nível coletivo não são objeto deles.

Em uma perspectiva organizacional, estudos mostram que organizações voltadas a projetos possuem limitações para aprender a partir de projetos anteriores (LYYTINEN; ROBEY, 1999; JULIAN, 2008). Lyytinen e Robey (1999) inclusive fazem um trocadilho com o título de seu artigo “*Learning failure in information systems development*”, apontando que “*learning failure*” refere-se tanto a falhas na aprendizagem quanto a aprendizagem de falhas. Segundo eles, as organizações falham em aprender e aprendem a falhar, por conta de quatro aspectos: o limite da inteligência organizacional, os desincentivos para aprendizagem, a estrutura organizacional e as barreiras educacionais.

Nessa perspectiva, as rotinas são a forma por meio da qual os “mitos em uso” persistem nas organizações, de modo que tais crenças e valores evitam que alternativas sejam

investigadas e exploradas (LYYTINEN; ROBEY, 1999). As rotinas proveem mecanismos formais para a incorporação de aprendizado de experiência passadas (JULIAN, 2008), o que pode ser feito a partir de ações como o compartilhamento de lições aprendidas ao término dos projetos.

Uma limitação dessa abordagem está na subestimação da agência, perspectiva segundo a qual os indivíduos envolvidos nas rotinas podem agir ativamente na sua criação e alteração, de forma endógena (FELDMAN; PENTLAND, 2003). Desse modo, as rotinas não são elementos exclusivamente prescritivos, os quais os indivíduos simplesmente seguem (FELDMAN; PENTLAND, 2003). Apesar do papel da agência, a existência de artefatos (especialmente os tecnológicos) faz com que as regras não sejam facilmente alteradas (D'ADDERIO, 2008). Nesse contexto, a pesquisa contribui para a compreensão das mudanças nas rotinas organizacionais em organizações de desenvolvimento de *software* ocorridas a partir das falhas.

Do ponto de vista gerencial e prático, este estudo também apresenta contribuições. Conforme apresentado por The Standish Group (2013), quase 1/5 dos projetos de desenvolvimento de *software* são completamente abortados, enquanto menos de 2/5 são bem-sucedidos e os demais são concluídos com limitações. Devido à alta taxa de insucesso, compreender a natureza e os fatores que levam a falhas permite que as organizações lidem melhor com elas e reduzam a ocorrência de problemas.

Além disso, são compreendidas as rotinas organizacionais de gestão de projetos de uma organização de desenvolvimento de *software*, incluindo seus aspectos ostensivo e performativo, além das regras e artefatos envolvidos (FELDMAN; PENTLAND, 2003; PENTLAND; FELDMAN, 2005; WEICHBRODT; GROTE, 2010). Essas rotinas têm relevante papel na criação de competências distintivas (GRANT, 1991; ALVES et al., 2011), capazes de gerar vantagem competitiva sustentável para as organizações (PRAHALAD; HAMEL, 1990) e estudá-las contribui na compreensão do desempenho organizacional (SALVATO; RERUP, 2011).

Além disso, é investigado como se dão as mudanças nas rotinas e sua relação com as falhas. A compreensão desse fenômeno pode auxiliar as organizações que desenvolvem *software* no sentido de identificar como se dá a aprendizagem a partir de falhas e como novos procedimentos são incorporados às suas rotinas organizacionais.

Por fim, é possível identificar a natureza da mudança gerada por falhas diferentes. A partir de contingências identificadas em seus projetos, são identificadas que características atuam como moderadores da relação, potencializando, atenuando ou modificando a forma

como a aprendizagem ocorre. Desse modo, é possível aperfeiçoar o gerenciamento e aumentar o alcance dos resultados organizacionais.

1.4 Estrutura da tese

Este documento está estruturado em seis capítulos. O primeiro deles é a Introdução e apresenta o problema de pesquisa, os objetivos geral e específicos, a justificativa e contribuições do estudo, tanto do ponto de vista acadêmico quanto gerencial, e a estrutura do documento.

O segundo capítulo apresenta o referencial conceitual e teórico da pesquisa. Seu conteúdo encontra-se dividido em três seções. A primeira delas aborda projetos de desenvolvimento de *software*, incluindo tipos de projetos, gestão de projetos, falhas em projetos e modelos de maturidade em organizações de desenvolvimento de *software*. A segunda seção apresenta as rotinas organizacionais, as mudanças nas rotinas organizacionais e esses elementos no contexto da gestão de projetos de *software*. Por fim, a seção final desse capítulo relaciona falhas e mudanças nas rotinas de gestão de projetos.

O capítulo 3 expõe os procedimentos metodológicos da pesquisa. Ele tem início com o seu delineamento, que apresenta seus pressupostos e o paradigma do estudo. Em seguida, o capítulo apresenta a revisão de literatura, como foi realizada a seleção do caso e unidade de análise. Posteriormente, mostra como o instrumento de coleta foi elaborado e, a partir dele, como os dados foram coletados e analisados.

O quarto capítulo apresenta os resultados da pesquisa. Ele está dividido em três seções: falhas em projetos de *software*, rotinas de gestão de projetos de *software* e suas mudanças e a relação entre falhas em projetos e mudanças em rotinas organizacionais de gestão de projetos de *software*. A primeira delas apresenta os resultados referentes aos critérios de falhas, fatores relacionados, ocasiões em que são identificadas e os *stakeholders* envolvidos. A seção 4.2 apresenta 15 rotinas de gestão de projetos que sofreram mudanças no caso estudado. Por fim, a seção final do capítulo 4 apresenta 20 episódios em que uma falha em algum projeto de desenvolvimento de *software* estava relacionada com uma mudança de rotina organizacional de gestão de projetos.

O capítulo 5 expõe a discussão dos resultados apresentados no capítulo 4 à luz da literatura exposta no capítulo 2. Por fim, o último capítulo apresenta as conclusões, limitações e sugestões de trabalhos futuros do estudo.

2 Referencial conceitual e teórico

Este capítulo apresenta o arcabouço conceitual e teórico utilizado como base para o estudo a ser realizado. Inicialmente, será feita uma exposição sobre projetos de desenvolvimento de *software*, envolvendo seus tipos, aspectos sobre gestão de projetos, falhas e maturidade. Em seguida, serão abordadas as rotinas organizacionais, incluindo sua definição e aspectos, suas mudanças, as rotinas de gestão de projetos de *software* e o processo de mudança nessas rotinas. Por fim, será apresentada uma proposição teórica da relação entre as falhas e as mudanças nas rotinas de gestão de projetos de *software*.

2.1 Projetos de desenvolvimento de *software*

As organizações de desenvolvimento de *software*, em geral, são voltadas a projetos, que consistem em “esforço temporário empreendido para criar um produto, serviço ou resultado exclusivo” (PMI, 2013, p. 3, tradução nossa). Nessa definição, “temporário” indica que os projetos têm início e fim definidos, ou seja, não são contínuos. Por “produto, serviço ou resultado exclusivo”, entende-se que o fruto de cada projeto é distinto do resultado de projetos anteriores (mesmo que seja apenas em alguns aspectos), diferentemente de produtos fabricados em linha de produção que resulta em diversos itens idênticos.

Devido a esse caráter temporário, para que um projeto seja desenvolvido, é necessário compor uma equipe específica para ele. Devido à importância do trabalho em equipe, a estrutura do time está diretamente ligada ao desempenho das atividades do projeto (YANG; TANG, 2004; LU et al., 2011). Após o término do projeto, todos os indivíduos envolvidos ficam livres para serem alocados em outros projetos da organização executora. Durante sua execução, porém, várias situações podem levar a mudanças de equipe, o que pode ter impacto positivo ou negativo sobre o projeto.

Em algumas ocasiões, por exemplo, um projeto (B) que ingressa na organização pode requerer que sejam utilizados membros de um projeto (A) que já estava em andamento, com o intuito de reduzir riscos para (B), mesmo que isso aumente os riscos de (A). Em outras situações, as mudanças podem já estar previstas, como a liberação de um profissional com determinada *expertise* após a conclusão de um módulo crítico que exigia tal competência. Um terceiro cenário ocorre quando problemas com tempo ou escopo levam a organização

executora a aumentar a equipe durante a execução, com o intuito de aproximar-se do cronograma previsto ou implementar novos requisitos que surgiram após o início do projeto. Por fim, também podem ocorrer alterações na equipe por conta de *turnover*, uma vez que o mercado de trabalho na área de tecnologia é bastante dinâmico.

Projetos de desenvolvimento de *software* costumam contar com equipes formadas por indivíduos com diversos papéis. O tamanho da equipe varia de acordo com o tipo e duração do projeto e dos recursos, porte e estrutura da organização. Além disso, dependendo desses aspectos, alguns papéis podem ser desnecessários ou vários deles podem ser desempenhados pelo mesmo indivíduo. O Quadro 1 apresenta alguns deles.

Quadro 1 (2) – Papéis típicos em uma equipe de desenvolvimento de *software*

Papel	Atividades
Gerente de projetos	Planeja, acompanha a execução e controla o projeto, exercendo papel de liderança diante dos outros membros da equipe.
Engenheiro de <i>software</i>	Realiza a codificação (programação) do <i>software</i> .
Arquiteto de <i>software</i>	Define a arquitetura do <i>software</i> , as tecnologias e requisitos não funcionais.
Analista de negócios	Define as funcionalidades que estarão presentes na aplicação.
Designer de interface	Projeta a interface entre a aplicação e o usuário.
Designer gráfico	Cria os elementos visuais utilizados na construção do <i>software</i> .
Gerente ou consultor de vendas	Faz o relacionamento comercial com o cliente.
Engenheiro de configuração	Gerencia o repositório e controla o versionamento de código.
Engenheiro de qualidade	Acompanha e toma medidas para garantir a qualidade dos produtos.
Engenheiro de testes	Realiza testes na aplicação, com o intuito de detectar problemas.
Administrador de banco de dados	Modela e gerencia a base de dados.
Administrador de sistema	Realiza atividades de suporte ao desenvolvimento.

Fonte: autor.

Além dos papéis apresentados no Quadro 1, é comum a presença de especialistas em outras áreas nas equipes, de acordo com a natureza dos projetos. Por exemplo, criadores de roteiros no desenvolvimento de jogos, biólogos no desenvolvimento de aplicações relacionadas a biologia computacional, músicos no desenvolvimento de aplicações musicais e jogos, professores e pedagogos no desenvolvimento de aplicações educacionais, entre outros.

2.1.1 Tipos de projetos de desenvolvimento de *software*

Atualmente, há uma grande variedade de projetos de desenvolvimento de *software* e essa diversidade requer diferenças em suas rotinas e métodos de gestão (SHENHAR, 1998; EVARISTO; FENEMA, 1999; PAPKE-SHIELDS et al., 2010; SIAU et al., 2010). Como a TI é aplicada em vários domínios, os projetos podem variar bastante com relação à sua complexidade, que pode ser definida em função de seu tipo e escopo (SHENHAR, 1998; CHOW; CAO, 2008; MCLEOD, MACDONELL, 2011) e do grau de incerteza tecnológica (SHENHAR, 1998; MCLEOD, MACDONELL, 2011).

Essas dimensões são apresentadas na tipologia de Shenhar (1998), criada para projetos técnicos. O autor classifica os projetos em uma matriz em função desses dois critérios, apresentada na Figura 1.

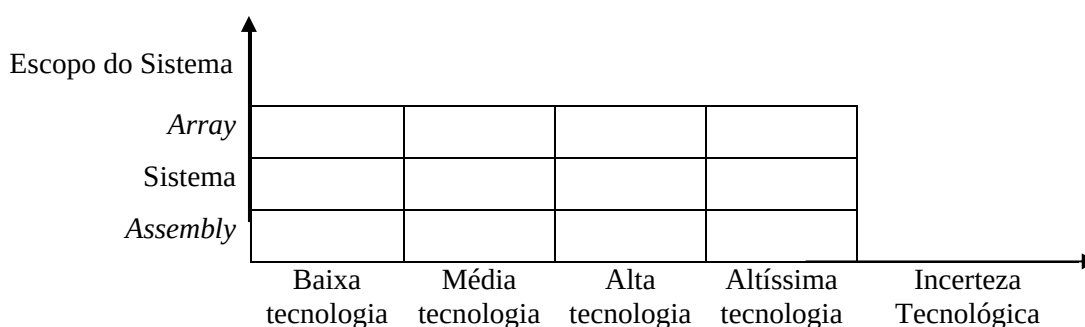


Figura 1 (2) – Tipologia para classificação de projetos (escopo e incerteza tecnológica).

Fonte: adaptada de Shenhar (1998).

Conforme pode ser observado na Figura 1, a tipologia utiliza quatro níveis de incerteza tecnológica: baixa, média, alta e altíssima tecnologia, de acordo com o grau de domínio das tecnologias envolvidas que a organização possui na fase de iniciação do projeto.

Projetos de baixa tecnologia são aqueles que se baseiam em tecnologias conhecidas e já dominadas pela organização executora no momento da iniciação. Não há aquisição, muito menos desenvolvimento de novas tecnologias. Desse modo, o risco tecnológico envolvido no projeto é mínimo (SHENHAR, 1998). Exemplos de projetos de desenvolvimento de *software* dessa natureza são a produção de um *website* institucional simples por uma organização especializada em desenvolvimento *web*, a criação de uma nova versão com mais cenários de um jogo já existente ou ainda a implementação de uma versão aperfeiçoada de um sistema de gestão comercial, com novos relatórios.

Os projetos de média tecnologia, por sua vez, são aqueles que utilizam alguma tecnologia existente no mercado que não é dominada pela organização no início do projeto.

Para estar nessa categoria, a inovação não pode representar a maior parte do projeto (SHENHAR, 1998). Segundo o autor, esse é o tipo mais comum de projetos industriais. Como exemplos da indústria de *software*, pode-se citar a implementação de um novo módulo em um sistema de gestão comercial para emissão de nota fiscal eletrônica¹ integrada com a secretaria da fazenda, ou o desenvolvimento de um sistema para faturamento de contas médicas utilizando o padrão TISS² da ANS, ambos pouco tempo após o surgimento das exigências legais, em uma organização com pouco domínio dos protocolos envolvidos.

Os projetos de alta tecnologia são aqueles em que tecnologias já existentes que não são dominadas pela organização executora no início do projeto correspondem à maior parte dele (SHENHAR, 1998). O autor ressalta que a maior parte dos projetos desse grupo refere-se a produtos completamente novos. Como exemplos na área de *software* têm-se o desenvolvimento de um aplicativo para auxílio à navegação de deficientes visuais³ ou a implementação por uma *startup* de um disco virtual para armazenamento remoto de arquivos na Internet⁴, utilizando o conceito de *cloud computing*.

Por fim, os projetos com maior grau de incerteza tecnológica são aqueles que envolvem altíssima tecnologia. Nesse tipo de projeto, são utilizadas tecnologias que não existem no momento de sua iniciação. Desse modo, grande parte do esforço empreendido pela organização executora é para desenvolver a nova tecnologia (SHENHAR, 1998). O autor afirma que esses projetos são relativamente raros. Como exemplo, pode-se citar o desenvolvimento de um servidor de *streaming*, incluindo um novo padrão revolucionário de codificação dos arquivos de áudio e vídeo a serem transmitidos.

Pela definição de incerteza tecnológica apresentada por Shenhar (1998), observa-se que esta característica depende tanto do projeto quanto da organização executora. Projetos que constituem inovações radicais têm altíssimo grau de incerteza para qualquer organização, dado que a tecnologia ainda não existe. Apesar disso, um projeto que utiliza uma tecnologia já existente pode ser classificado como de baixa, média ou alta incerteza tecnológica, dependendo do domínio da tecnologia que a organização possua no início do projeto.

A outra dimensão apresentada na Figura 1 é o escopo do projeto, que pode apresentar 03 níveis: *assembly*, sistema e *array* (SHENHAR, 1998). O autor apresenta duas situações em que um projeto pode ser considerado *assembly*: quando ele é apenas uma parte ou componente de um sistema maior ou quando ele é um produto por si só, mas desempenha

¹ Vide <http://www.nfe.fazenda.gov.br>, acesso em 10/01/2014.

² Vide <http://www.ans.gov.br/index.php/planos-de-saude-e-operadoras/tiss>, acesso em 10/01/2014.

³ Vide <http://veye-blog.blogspot.com/>, acesso em 10/01/2014.

⁴ Vide <http://www.dropbox.com/>, acesso em 10/01/2014.

uma única função de escopo limitado. Como exemplos, podem-se citar, respectivamente, o desenvolvimento de um componente para envio de nota fiscal eletrônica para a secretaria da fazenda ou a implementação de um aplicativo semelhante à calculadora do *Windows*.

O segundo nível de escopo do projeto é o de sistema, que consiste em um grupo de componentes que atuam em conjunto como um produto único (SHENHAR, 1998). Como exemplos desta categoria, enquadram-se sistemas de *Enterprise Resource Planning* (ERP) e *Customer Relationship Management* (CRM). Por fim, os projetos do tipo *array* são compreendidos com um conjunto de sistemas cuja utilização será feita de forma geograficamente dispersa (SHENHAR, 1998). Como exemplos, podem-se citar um prontuário médico nacional unificado para uso pelo Sistema Único de Saúde (SUS) ou um sistema integrado para gestão de segurança pública em nível estadual.

Além de variar com relação ao grau de incerteza tecnológica e ao escopo do sistema, os projetos também apresentam tamanho/duração distintos (CHOW; CAO, 2008; EL-EMAN; KORU, 2008; HARRIS et al., 2009; PAPKE-SHIELDS et al., 2010). Essas características podem depender tanto do *software* que está sendo desenvolvido, quanto de aspectos do ambiente interno da organização responsável pelo projeto. A duração do projeto varia de acordo com a complexidade do mesmo, mas também depende da disponibilidade de recursos, em termos de quantidade e grau de experiência dos profissionais envolvidos. Além disso, a própria quantidade de pessoas alocadas também pode variar de acordo com a senioridade da equipe.

Outra característica dos projetos é o local em que ocorrem e a existência de outros projetos (EVARISTO; FENEMA, 1999; HUMMEL et al., 2013). Aqueles que têm equipes que atuam no mesmo local físico e os que têm times distribuídos geograficamente, bem como os cenários em que a organização desenvolve um único projeto ou vários ao mesmo tempo, constituem ambientes diferentes para a gestão de projetos. Desse modo, Evaristo e Fenema (1999) criaram uma tipologia para a gestão de projetos, em que apresentam 07 tipos, dependendo do local em que ocorrem e da existência de múltiplos projetos na mesma organização. Esse modelo pode ser observado na Figura 2.

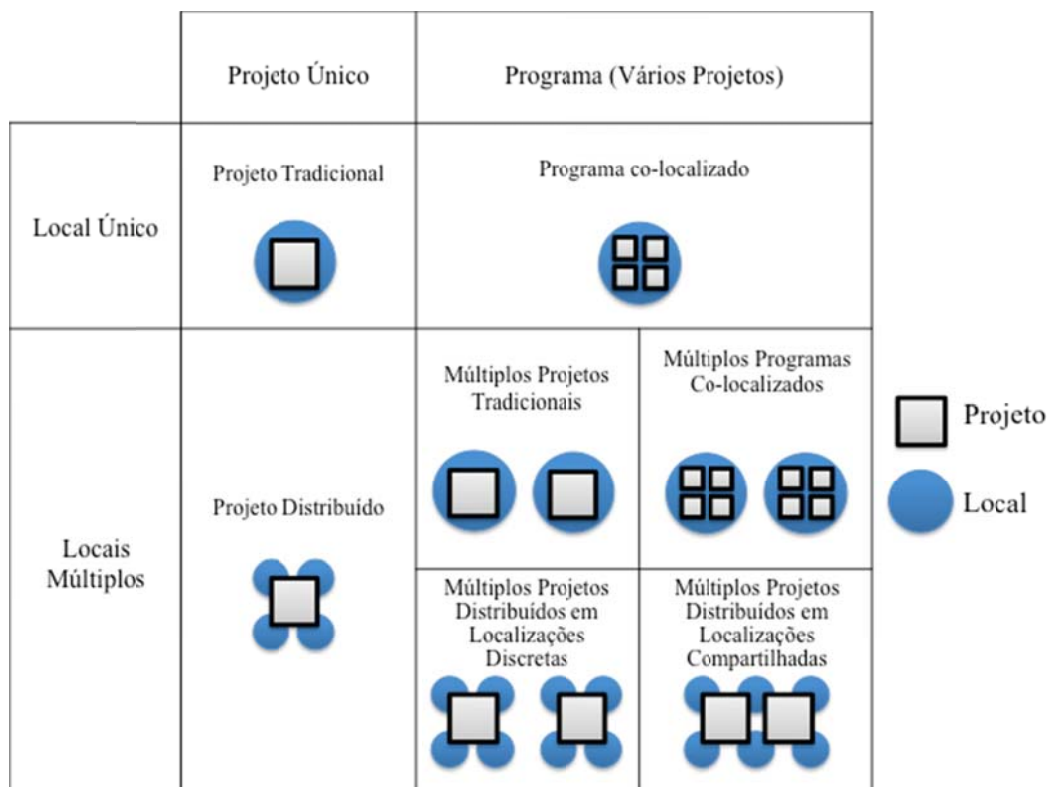


Figura 2 (2) – Tipologia para a classificação de projetos (localização e quantidade).
 Fonte: Evaristo e Fenema (1999 , tradução nossa).

A primeira coluna da Figura 2 representa cenários em que a organização atua com um único projeto. O primeiro dos tipos apresentados é o projeto tradicional, que ocorre quando a organização possui um único projeto em um único local. Se um único projeto for desenvolvido em mais de um local, ele se torna um projeto distribuído, exibido abaixo do projeto tradicional.

A segunda coluna apresenta os programas (vários projetos), que podem possuir cinco tipos de projetos, dependendo da sua distribuição. O tipo superior ocorre quando há múltiplos projetos em um único lugar e é denominado programa co-localizado. Quando há multiplicidade de projetos e locais, são possíveis quatro cenários: múltiplos projetos tradicionais, múltiplos programas co-localizados, múltiplos projetos distribuídos em localizações discretas e múltiplos projetos distribuídos em localizações compartilhadas (EVARISTO; FENEMA, 1999). O Quadro 2 apresenta detalhes de cada um dos tipos apresentados.

Quadro 2 (2) – Tipologia para a classificação de projetos (localização e unicidade)

Tipo de projeto	Detalhes
(a) Projeto Tradicional	Tipo de projeto comum, tradicional.
(b) Projeto Distribuído	Coordenação e comunicação entre os diversos locais é um ponto crítico.
(c) Programa Co-localizado	Múltiplos projetos no mesmo local podem gerar interdependência e competição por recursos.
(d) Múltiplos Projetos Tradicionais	Pode haver dificuldades na comunicação e na alocação de recursos, pois a mobilidade entre projetos de localidades distintas tende a ser reduzida.
(e) Múltiplos Programas Co-localizados	Semelhante ao tipo (d), mas há múltiplos projetos em cada local, aumentando a complexidade do cenário.
(f) Múltiplos Projetos Distribuídos em Localizações Discretas	Apresenta as dificuldades do tipo (b), além da complexidade adicional por estar lidando com múltiplos projetos.
(g) Múltiplos Projetos Distribuídos em Localizações Compartilhadas	Semelhante ao tipo (f), com identificação de sinergias e compartilhamento de recursos entre projetos simplificados em locais com múltiplos projetos. Apesar disso, pode tornar mais difícil a negociação para alocação de recursos.

Fonte: autor, a partir de Evaristo e Fenema (1999).

Conforme pode ser observado no Quadro 2, cada cenário requer abordagens diferentes do ponto de vista de gestão. Projetos distribuídos apresentam dificuldade de coordenação e comunicação, enquanto múltiplos projetos constituem um desafio do ponto de vista de comunicação e alocação de recursos.

Os projetos também podem ser classificados com relação à forma de contratação. Segundo o PMI (2013), há três categorias de contratação: contratos de preço fixo ou preço global, contratos de custos reembolsáveis e contratos por tempo e material. Os projetos de preço fixo possuem um preço predefinido para um escopo predeterminado. Em projetos com custos reembolsáveis, o cliente paga ao fornecedor um valor referente aos seus custos acrescido de uma remuneração pelo trabalho. Por fim, projetos contratados sob a modalidade de tempo e material possuem características dos dois outros modelos. Podem ser definidos valores unitários para determinados recursos (como no preço fixo), mas o escopo real do que será feito pode não ser definido a priori (PMI, 2013).

2.1.2 Gestão de projetos de desenvolvimento de *software*

Com o intuito de desenvolver produtos de qualidade, a indústria de *software* adota uma gama de modelos voltados à melhoria de seus processos. No âmbito da gestão de projetos, há diferentes perspectivas, com destaque para o PMBOK e o *Scrum*.

O PMBOK é uma publicação do *Project Management Institute* (PMI) que visa agrupar o conhecimento tido como boa prática para a atividade. O guia é genérico a projetos de

quaisquer naturezas e possui dezenas de processos, organizados em 05 grupos (iniciação, planejamento, execução, monitoramento e encerramento) e 10 áreas de conhecimento: escopo, tempo, custos, qualidade, recursos humanos, comunicação, riscos, aquisições, integração e partes interessadas (PMI, 2013). Cada processo faz parte de um dos grupos e de uma das áreas, conforme pode ser observado no Quadro 3.

Quadro 3 (2) – Processos do PMBOK, de acordo com os grupos e áreas (Continua)

	Iniciação	Planejamento	Execução	Monitoramento e Controle	Encerramento
Integração	- Desenvolvimento do Termo de Abertura do Projeto;	- Desenvolvimento do Plano de Gerenciamento do Projeto;	- Orientação e Gerenciamento do Trabalho do Projeto;	- Monitoramento e Controle do Trabalho do Projeto; - Controle Integrado de Mudanças;	- Encerramento do Projeto ou Fase;
Escopo		- Planejamento do Gerenciamento do Escopo; - Coleta dos Requisitos; - Definição do Escopo; - Criação da Estrutura Analítica do Projeto (EAP);		- Validação do Escopo; - Controle do Escopo;	
Tempo		- Planejamento do Gerenciamento do Cronograma; - Definição das Atividades; - Sequenciamento de Atividades; - Estimativa de Recursos das Atividades; - Estimativa de Duração das Atividades; - Desenvolvimento do Cronograma;		- Controle do Cronograma;	
Custos		- Planejamento do Gerenciamento dos custos; - Estimativa de Custos; - Determinação do Orçamento;		- Controle de Custos;	
Qualidade		- Planejamento do Gerenciamento da Qualidade;	- Realização da Garantia de Qualidade;	- Controle de Qualidade;	
Recursos Humanos		- Planejamento o Gerenciamento dos Recursos Humanos;	- Mobilização da Equipe do Projeto; - Desenvolvimento da Equipe do Projeto; - Gerenciamento da Equipe do Projeto;		

(Conclusão)

	Iniciação	Planejamento	Execução	Monitoramento e Controle	Encerramento
Comunicações		- Planejamento do Gerenciamento das Comunicações;	- Gerenciamento das Comunicações;	- Controle das Comunicações;	
Riscos		- Planejamento do Gerenciamento de Riscos; - Identificação de Riscos; - Análise Qualitativa de Riscos; - Análise Quantitativa de Riscos; - Planejamento de Respostas a Riscos;		- Controle de Riscos;	
Aquisições		- Planejamento do Gerenciamento das Aquisições;	- Condução das Aquisições;	- Controle das Aquisições;	- Encerramento das Aquisições;
Partes Interessadas	- Identificação das Partes Interessadas	- Planejamento do Gerenciamento das Partes Interessadas	- Gerenciamento do Envolvimento das Partes Interessadas	- Controle do Envolvimento das Partes Interessadas.	

Fonte: autor, a partir de PMI (2013).

Cada um dos processos apresentados no Quadro 3 é definido em função de suas entradas, técnicas e ferramentas, saídas e detalhes. O Quadro 4 apresenta a definição de um dos processos do PMBOK:

Quadro 4 (2) – Detalhamento do processo Plano de Gestão do Projeto

Desenvolvimento do Plano de Gestão do Projeto	
Parte da Descrição	Este processo define, prepara e coordena todos os planos subsidiários e os integra em um plano de gestão do projeto compreensivo. O principal benefício do processo é um documento central que define a base de como todo o trabalho será feito no projeto.
Entradas	- Termo de abertura do projeto; - Saídas de outros processos; - Fatores ambientais da empresa; - Ativos de processos organizacionais;
Técnicas e Ferramentas	- Julgamento de especialistas; - Técnicas de facilitação.
Saídas	- Plano de gestão do projeto.

Fonte: autor, a partir de PMI (2013).

O Quadro 4 apresenta as informações de forma resumida. No guia, cada processo é apresentado em detalhes, de modo que possa ser operacionalizado. Assim, cada entrada, técnica, ferramenta e saída é descrita e exemplificada.

Devido à sua estrutura e quantidade de documentos, o PMBOK não costuma ser adotado em projetos com equipes formadas por poucos indivíduos (BATRA et al., 2011). Sua utilização em projetos de pequeno porte tende a gerar sobrecarga para a equipe, uma vez que uma considerável parte do tempo do projeto é despendida na elaboração de artefatos.

O *Scrum* é uma alternativa a ele que vem ganhando adesão, por ser uma abordagem mais ágil. De acordo com os princípios do Manifesto Ágil (2001), a metodologia prega o seguinte:

Estamos descobrindo maneiras melhores de desenvolver *software*, fazendo-o nós mesmos e ajudando outros a fazerem o mesmo. Através deste trabalho, passamos a valorizar:

Indivíduos e interações mais que processos e ferramentas;

Software em funcionamento mais que documentação abrangente;

Colaboração com o cliente mais que negociação de contratos;

Responder a mudanças mais que seguir um plano.

Ou seja, mesmo havendo valor nos itens à direita, valorizamos mais os itens à esquerda. (MANIFESTO ÁGIL, 2001, grifo do autor)

O Quadro 5 apresenta um comparativo entre as duas abordagens, de acordo com as áreas do PMBOK:

Quadro 5 (2) – Gerenciamento de projetos tradicional x ágil

(Continua)

Área	Gerenciamento Tradicional	Gerenciamento Ágil
Escopo	Bem definido nas fases iniciais do projeto e formalizado através de WBS (<i>Work Breakdown Structure</i>).	Escopo é definido em alto nível e os requisitos são priorizados e definidos de forma interativa. Necessita de maior controle <i>gold plating</i> (trabalho extra que não faça parte do projeto).
Tempo	Cronograma detalhado para a realização de todo o projeto.	Cronograma orientado por produto com entregas incrementais a cada 1-4 semanas.
Custo	Monitoramento das alterações para que não afetem o custo planejado.	Maior controle em função da rapidez na incorporação de alterações.
Qualidade	Processos de verificação e validação e plano de testes.	Programação em pares, testes incrementais e refatoração.
Riscos	Análise de riscos durante todo o ciclo de vida do projeto.	Aplica-se o mesmo conceito do gerenciamento tradicional.
Comunicações	Documentada e formal.	Implícita, interpessoal e colaborativa.
Recursos Humanos	Papéis claros e bem definidos.	Confiança nos membros da equipe e ambiente colaborativo.
Aquisição	Controle por contrato e escopo bem definido e documentado.	Presença do cliente, volatilidade de requisitos e pouca documentação tornam o processo um desafio.

(Conclusão)

Área	Gerenciamento Tradicional	Gerenciamento Ágil
Integração	Plano de projeto detalhado e controle total do projeto pelo gerente.	Plano de projeto evolutivo e gerente do projeto atua como facilitador.
Partes interessadas	Área específica só surgiu na última versão do PMBOK (5ª). Antes disso, o papel estava incluso em alguns dos processos das outras áreas, mas não havia destaque explícito.	Forte papel na definição da solução, especialmente os usuários e clientes, participando de forma bastante intensa na definição e priorização de requisitos.

Fonte: adaptado de Innovit (2011).

Conforme pode ser observado no Quadro 5, a estrutura do *Scrum* é bastante simplificada, comparando-se com o PMBOK. O *Scrum* possui os seguintes elementos principais: Times *Scrum* e seus papéis, Eventos com Duração Fixa, Artefatos e Regras (SCHWABER, 2009). Segundo o autor, uma equipe possui três papéis: o *Scrum Master*, o *Product Owner* e o Time. O *Scrum Master* é responsável por fazer com que o processo funcione, o *Product Owner* maximiza o valor do trabalho feito, enquanto o Time consiste nos desenvolvedores, *designers* e demais profissionais envolvidos com a produção do *software*.

Os Eventos com Duração Fixa são a Reunião de Planejamento da Versão para Entrega, a Reunião de Planejamento da *Sprint*, a *Sprint*, a Reunião Diária, a Revisão da *Sprint* e a Retrospectiva da *Sprint*. A *Sprint*, conceito chave na metodologia, é uma iteração com duração média de três a quatro semanas. Cada projeto consiste em várias *sprints*, cujo resultado é uma versão potencialmente entregável do *software* que está sendo desenvolvido. Os demais eventos do *Scrum* são apresentados no Quadro 6.

Quadro 6 (2) – Eventos com duração fixa do *Scrum*

Evento	Descrição
Planejamento da Versão para Entrega	Estabelece planos e metas para que o Time <i>Scrum</i> e o resto da organização possam entender e se comunicar. Estabelece meta da versão, funcionalidades prioritárias, riscos, data de entrega, entre outros detalhes.
Reunião de Planejamento da <i>Sprint</i>	Encontro em que a <i>sprint</i> é planejada, definindo o que será feito na iteração e como as funcionalidades serão desenvolvidas.
Reunião Diária	Reunião diária para que cada membro compartilhe o que realizou desde a última, o que fará antes da próxima e quais os obstáculos existentes.
Revisão da <i>Sprint</i>	Reunião para compartilhamento dos resultados da <i>sprint</i> e discussão dos próximos passos.
Retrospectiva da <i>Sprint</i>	Revisão do processo de desenvolvimento, de modo a otimizá-lo para <i>sprints</i> futuras.

Fonte: autor, a partir de Schwaber (2009).

Segundo Schwaber (2009), os principais artefatos do *Scrum* são: o *Backlog* do Produto, o *Backlog* da *Sprint*, o *Burndown* de Versão de Entrega e o *Burndown* da *Sprint*. Os *Backlogs* são listas de itens necessários para a produção do *software*, para uma versão do produto como um todo e em uma *sprint* específica, respectivamente. Os *Burndowns* medem os itens restantes para conclusão da versão e da *sprint*, respectivamente.

Por fim, as regras definem como o *Scrum* pode ser operacionalizado pela organização executora dos projetos. Desse modo, fazem o elo entre os eventos com duração fixa (*time-boxes*), os papéis e os artefatos do *Scrum* (SCHWABER, 2009). Há dezenas delas e alguns exemplos apontados pelo autor são: apenas os membros do Time podem falar durante uma Reunião Diária (ou seja, nem o *Product Owner* nem o *Scrum Master* podem fazê-lo); os Times são auto-organizáveis; e a composição dos Times e as metas de qualidade devem permanecer constantes durante uma *sprint*.

Além do PMBOK e do *Scrum*, é comum a adoção de modelos para melhoria do processo de *software*. Estes serão apresentados na seção 2.1.4.

2.1.3 Falhas em projetos de desenvolvimento de *software*

O conceito de falhas está associado à ocorrência de problemas de desempenho nos projetos. Elas podem ser compreendidas como “uma diferença inaceitável entre o desempenho previsto e o observado” (PETROSKI, 2006, p. 51, tradução nossa). Assim, há uma expectativa de desempenho em um projeto e o atendimento a essa expectativa corresponde ao sucesso, enquanto o não atendimento constitui uma falha.

Apesar do conceito parecer simples, na literatura de gestão de projetos há diversas definições do que seriam desempenho, sucesso e falhas. A análise destas propostas mostra a existência de várias semelhanças e divergências entre elas, indicando que o assunto não é totalmente esclarecido. Vários autores apontam que uma das causas disso é o fato de esses conceitos terem conotações distintas para diferentes pessoas e dependerem do contexto (PINTO; MANTEL, 1990; FREEMAN; BEALE, 1992; WATERIDGE, 1998; LINBERG, 1999; JUGDEV; MÜLLER, 2005; STANDING et al., 2006; PEREIRA et al., 2008; SAVOLAINEN et al., 2012).

Uma análise histórica mostra que o conceito vem evoluindo ao longo do tempo, embora a evolução ocorra em momentos diferentes nos estudos de projetos em geral e nas pesquisas de projetos de SI. Segundo Kerzner (2004), a primeira noção de desempenho estava associada exclusivamente aos requisitos técnicos do projeto. Em um período em que ainda

não havia gestão formal de projetos, era considerado bem sucedido aquele projeto que fosse concluído de acordo com a especificação técnica, independentemente de outros aspectos. Segundo o autor, um projeto que consumisse tempo e recursos em excesso, por exemplo, mas fosse finalizado de acordo com os requisitos previstos, era considerado bem sucedido. Falhas, nesse contexto, eram apenas a não conclusão do projeto ou sua conclusão sem o escopo predefinido (KERZNER, 2004). Apesar de restrito, esse conceito de falhas é utilizado em pesquisas sobre projetos de SI até os dias atuais (DILTS; PENCE, 2006; EL-EMAN; KORU, 2008).

Essa visão simplificada trouxe problemas ligados a eficiência para as organizações executoras, fazendo surgir a gestão de projetos, um esforço para saná-los (KERZNER, 2004). Nesse contexto, as exigências passaram da simples conclusão dos projetos ao atendimento a um cronograma, custo e nível de qualidade planejados. Essa visão predominou da década de 1960 à de 1980 (JUGDEV; MÜLLER, 2005) e está ilustrada na Figura 3.

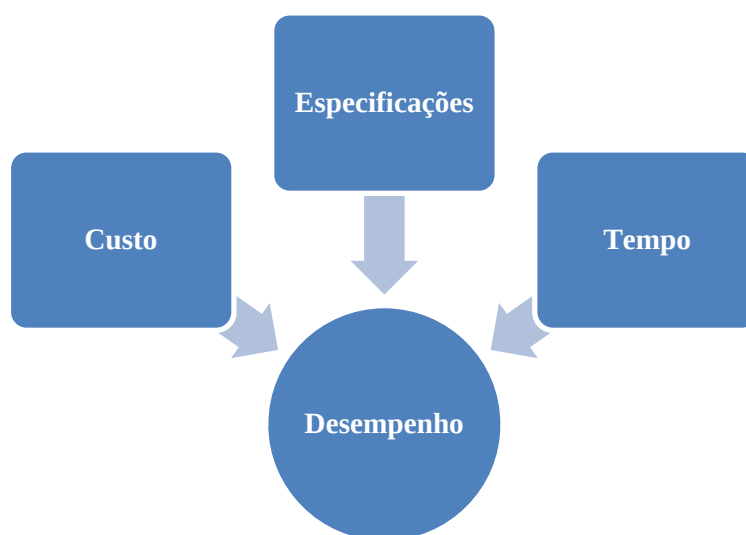


Figura 3 (2) – Critérios de desempenho de projetos (1960-1980).
Fonte: autor, a partir de Kerzner (2004).

Todos os critérios apresentados na Figura 3 tinham foco exclusivo na organização executora do projeto, o que constitui uma limitação dessa abordagem. Um projeto que atendesse ao cronograma, custo e qualidade demandados, mas não satisfizesse o cliente, era considerado bem sucedido. Assim, eram consideradas falhas apenas o não atendimento aos requisitos de tempo, custo e qualidade.

Apesar de ter caracterizado os estudos de gestão de projetos em geral nos anos 1960-1980 (JUGDEV; MÜLLER, 2005), no contexto de SI essa perspectiva está presente em pesquisas mais recentes. Linberg (1999) analisa o conceito de sucesso na visão de

desenvolvedores que, para projetos concluídos, o caracterizam como atendimento à expectativa de qualidade, custo, esforço e cronograma. Agarwal e Rathod (2006) e Chow e Cao (2008) adotam apenas escopo (funcionalidade e qualidade), tempo e custo, enquanto Jiang et al. (2002) e Tesch et al. (2009) destacam aspectos semelhantes, embora cite um critério “atingimento das metas do projeto”, cuja abrangência não é clara.

Nova evolução no conceito fez com que o desempenho de um projeto passasse a englobar o cliente, por dois motivos. Primeiramente, não se podia considerar sucesso um projeto que atendesse aos requisitos de tempo, custo e especificações, mas não fosse satisfatório para o cliente (KERZNER, 2004). Por outro lado, pesquisas mostraram que vários *stakeholders* não percebiam como fracassados alguns projetos que apresentavam falhas relacionadas a tais indicadores, o que indicava que deveria haver algo mais além desses três critérios (MORAES, 2004). Desse modo, a gestão de projetos passou a apresentar tanto indicadores internos, da organização executora, quanto externos, do cliente. Esses critérios podem ser observados no Quadro 7, que apresenta indicadores utilizados por Pinto e Slevin (1988) e por Dvir et al. (1998).

Quadro 7 (2) – Critérios de desempenho de projetos (1980-2000)

Pinto e Slevin (1988)	Dvir et al. (1998)
Indicadores Internos: - Tempo; - Custo; - Especificações.	Atendimento às metas do projeto: - Especificações funcionais; - Especificações técnicas; - Tempo; - Custo.
Indicadores Externos: - Uso (utilização do projeto de acordo com sua proposta original); - Satisfação (satisfação com o processo de desenvolvimento); - Eficácia (benefícios diretos aos usuários).	Benefícios ao consumidor: - Atendimento das metas de aquisição; - Atendimento dos requisitos operacionais; - Entrada do produto em operação; - Alcance do consumidor final em tempo hábil; - Uso prolongado do produto; - Aumento no nível operacional do usuário, ocasionado pelo produto; - Satisfação do usuário com o produto.

Fonte: autor.

Segundo Jugdev e Müller (2005), essa fase da gestão de projetos em geral durou de 1980 a 2000. Apesar disso, no contexto de projetos SI, observam-se estudos nessa perspectiva no século XXI. Correspondem a esse estágio estudos que adotam, além de tempo, custo e

especificações/escopo, critérios como facilidade de uso do produto e atendimento aos requisitos do cliente/usuário (PROCACCINO; VERNER, 2002; PEREIRA et al. 2008), satisfação do usuário (EL-EMAM; KORU, 2009; SIAU et al., 2010; TARAWNEH, 2011) e uso do sistema (SIAU et al., 2010; TARAWNEH, 2011).

Os indicadores apresentados no Quadro 7 estão de algum modo associados ao que autores como Munns e Bjeirmi (1996) denominam sucesso da gestão do projeto e sucesso do projeto, respectivamente, como pode ser observado na Figura 4.

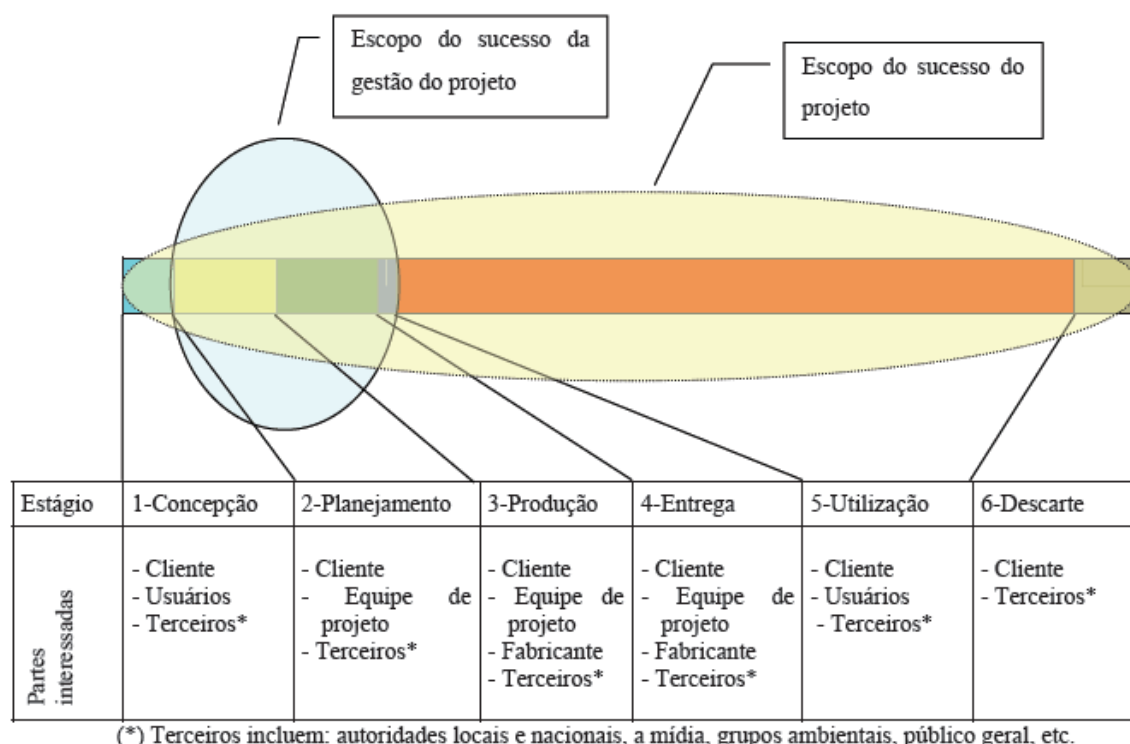


Figura 4 (2) – Escopo do desempenho do projeto e do desempenho da gestão do projeto.
Fonte: Moraes (2004).

A Figura 4 ilustra a distinção entre o escopo do desempenho da gestão do projeto e o desempenho do projeto, associando-os aos estágios do ciclo de vida dos produtos e às partes interessadas em cada um deles. Segundo os autores, o cliente e os “terceiros” são *stakeholders* durante todos os estágios, enquanto a equipe de projeto está interessada nos estágios associados ao projeto propriamente dito, e os usuários, focados nos estágios de concepção e utilização. Para projetos de *software*, dependendo da metodologia de desenvolvimento utilizada, essa visão com relação aos usuários pode não corresponder à realidade. Algumas abordagens podem envolver os usuários em diversas fases do projeto e não apenas na concepção e utilização da aplicação.

Esta diferenciação entre o desempenho do projeto e o de sua gestão apresentada na Figura 4 também é feita por outros autores, como Baccarini (1999), Lim e Mohamed (1999) e Cooke-Davies (2002), cada um com sua própria nomenclatura. Esse último diferencia desempenho do projeto do desempenho da gestão do projeto, medidos, respectivamente, em função dos objetivos gerais do projeto e das métricas tradicionais como custo, tempo e qualidade.

Lim e Mohamed (1999) abordam o desempenho de projetos sob dois pontos de vista: macro e micro. A visão macro relaciona sucesso ao atendimento da proposta original do projeto, de modo que não pode ser avaliada até a operação do sistema e depende dos usuários e outros *stakeholders*. Por outro lado, a visão micro está associada a partes menores do projeto, possuindo vínculo com a sua construção. Baccarini (1999), por sua vez, utiliza a nomenclatura sucesso do produto e sucesso da gestão do projeto, que estão relacionados aos efeitos do produto final do projeto e métricas tradicionais, respectivamente. Para o autor, a união do sucesso do produto e da gestão do projeto consiste no sucesso do projeto.

Essa discussão está associada ao ciclo de vida do projeto, compreendido como o conjunto de fases em que o projeto é dividido pelas organizações, visando a um melhor controle gerencial e uma ligação mais adequada de cada projeto aos processos operacionais contínuos da organização (PMI, 2013). É importante diferenciar o ciclo de vida do projeto do ciclo de vida do produto, que costuma compreender o primeiro, acrescido de etapas posteriores, como entrega (JUGDEV; MÜLLER, 2005), utilização e descarte (PMI, 2013; JUGDEV; MÜLLER, 2005). A Figura 5 ilustra bem o conceito, apresentando as fases em projetos de construção, farmacêutico, de defesa e de *software*.

	FASES		CONSTRUÇÃO	FARMACÊUTICO	DEFESA	SOFTWARE
Ciclo de Vida do Produto	Fase inicial: conceitualização, planejamento	Ciclo de Vida do Projeto	Viabilidade	Descoberta e criação de cenário	Exploração e Definição Conceitual	Ciclo da Prova Conceitual
	Fase intermediária: produção/ implementação		Planejamento e Projeto	Desenvolvimento pré-clínico	Demonstração e Validação	Primeiro Ciclo de Desenvolvimento
			Produção	Registros	Desenvolvimento de Engenharia e Manufatura	Segundo Ciclo de Desenvolvimento
	Fase final: entrega		Turnover e start up	Atividade Pós-Submissão	Produção e Implantação	Ciclo final, incluindo testes
	Operações: utilização				Operações e apoio	Implantação
	Desinvestimento: encerramento					

Figura 5 (2) – Visão geral dos ciclos de vida do produto e do projeto, segundo o PMBOK.

Fonte: Adaptada de Jugdev e Müller (2005).

Analisando a Figura 5, é possível observar também que a abrangência do ciclo de vida do projeto pode variar de acordo com a sua natureza, aproximando-se do ciclo de vida do produto (projetos de construção e farmacêuticos ilustrados na figura) ou distanciando-se dele (projetos de defesa e desenvolvimento de *software* da figura).

No final do século XX e início do século XXI, estudos passaram a trazer uma visão mais holística ao desempenho dos projetos. No contexto de projetos de TI, Wateridge (1995) caracteriza desempenho como atendimento aos requisitos dos usuários, satisfação dos usuários, orçamento, cronograma, atendimento ao objetivo do projeto, sucesso comercial e qualidade. Três anos depois, o autor acrescenta dois critérios: satisfação do patrocinador e satisfação da equipe (WATERIDGE, 1998). Observa-se aí o envolvimento de outros *stakeholders* não contemplados anteriormente. No ano seguinte, um estudo de Linberg (1999) destaca aspectos ligados a aprendizagem. Esse é um dos critérios de sucesso no ponto de vista dos desenvolvedores que a pesquisa do autor apontou, considerando projetos abortados.

Outra tendência observada nos estudos sobre desempenho de projetos nessa época foi o surgimento de *frameworks* apresentando os fatores em dimensões. O modelo proposto por Atkinson (1999) ilustra isso e está apresentado no Quadro 8.

Quadro 8 (2) – Categorias de sucesso e seus elementos

Categorias	Elementos
Triângulo de ferro	Custo
	Qualidade
	Tempo
O Sistema de informação	Manutenibilidade
	Confiabilidade
	Validade
	Qualidade da informação
Benefícios (organização)	Aumento da eficiência
	Aumento da efetividade
	Aumento do lucro
	Alcance dos objetivos estratégicos
	Aprendizagem organizacional
	Redução do desperdício
Benefícios (comunidade de <i>stakeholders</i>)	Satisfação dos usuários
	Impacto social e ambiental
	Desenvolvimento pessoal e profissional
	Lucro do contratante
	Impacto econômico na comunidade

Fonte: autor, a partir de Atkinson (1999).

Além de categorizar os critérios em dimensões, o modelo do Quadro 8 também se diferencia de outras perspectivas por apresentar vários critérios ligados especificamente a *software* (manutenibilidade, confiabilidade, etc.), à organização (como aumento do lucro, alcance dos objetivos estratégicos e aprendizagem organizacional) e a outros *stakeholders* (como impacto social e ambiental e impacto econômico na comunidade).

Contemplar esses novos elementos no conceito de desempenho de projetos constituiu uma nova perspectiva na academia. Dois anos depois, Shenhar et al. (2001) apresentaram um modelo com algumas características semelhantes não diretamente ligado a projetos de *software*. O modelo deles está apresentado no Quadro 9 e é composto por quatro dimensões de sucesso, cada uma das quais formada por várias medidas.

Quadro 9 (2) – Dimensões de sucesso e suas medidas

Dimensão	Medidas
Eficiência do projeto	Atendimento às metas de cronograma
	Atendimento às metas de orçamento
Impacto no consumidor	Atendimento ao desempenho funcional
	Atendimento às especificações técnicas
	Satisfação das necessidades do cliente
	Resolução de um problema do cliente
	Uso do produto pelo cliente
	Satisfação do cliente
Sucesso empresarial	Sucesso comercial
	Obtenção de um alto <i>market share</i>
Preparação para o futuro	Criação de um novo mercado
	Criação de uma nova linha de produtos
	Desenvolvimento de uma nova tecnologia

Fonte: Shenhar et al. (2001, tradução nossa).

Conforme pode ser visto no Quadro 9, além de metas do projeto e benefícios ao consumidor, o conceito de sucesso passa a abranger outras áreas, como estratégia e aprendizagem, reforçando a nova perspectiva apresentada por Atkinson (1999). Seis anos depois, Shenhar e Dvir (2007) adaptam o modelo e acrescentam uma dimensão, conforme apresentado no Quadro 10.

Quadro 10 (2) – Dimensões de sucesso e suas medidas (novo modelo) (Continua)

Dimensão	Medidas
Eficiência	Cronograma
	Orçamento
	Lucratividade
	Outros aspectos ligados a eficiência

(Conclusão)

Dimensão	Medidas
Impacto no consumidor	Atendimento a requisitos e especificações
	Benefícios para o cliente
	Uso
	Satisfação e lealdade do cliente
	Reconhecimento da marca
Impacto na equipe	Satisfação da equipe
	Moral da equipe
	Desenvolvimento de habilidades
	Crescimento dos membros da equipe
	Retenção da equipe
	Ausência de exaustão
Sucesso empresarial	Vendas
	Lucro
	Market share
	ROI e ROE
	Fluxo de caixa
	Qualidade no serviço
	Tempo do ciclo
	Medidas organizacionais
	Aprovação regulamentar
Preparação para o futuro	Nova tecnologia
	Novo mercado
	Nova linha de produtos
	Nova competência essencial
	Nova capacidade organizacional

Fonte: adaptado de Shenhar e Dvir (2007).

Além dessas novas dimensões, Shenhar et al. (2001) e Shenhar e Dvir (2007) apresentam também um caráter contingencial, em que a importância dos critérios varia de acordo com o horizonte temporal. Essa associação pode ser observada na Figura 6.

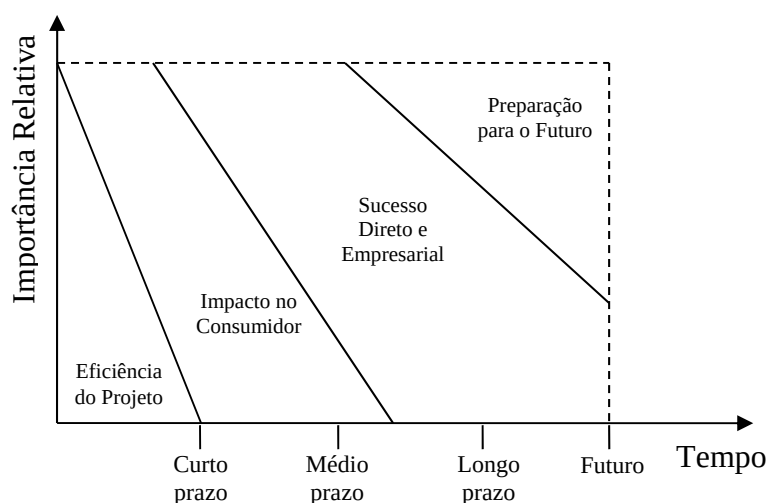


Figura 6 (2) – Importância relativa das dimensões de sucesso ao longo do tempo.
Fonte: Shenhar e Dvir (2007, tradução nossa).

Segundo a Figura 6, pensar em função da eficiência do projeto e do impacto nos consumidores está associado a um horizonte de curtíssimo ou curto prazo e sua relevância tende a se reduzir na medida em que se pensa em médio ou longo prazo. Nesses horizontes, o desempenho de um projeto estaria relacionado ao sucesso no negócio e preparação para o futuro, respectivamente.

Além do horizonte temporal, a importância relativa das dimensões de sucesso também varia de acordo com a incerteza tecnológica do projeto, conceito apresentado na seção 2.1.1. Desse modo, determinado resultado ser compreendido como falha depende do tipo de projeto em análise. Esses aspectos se relacionam conforme apresentado na Figura 7.

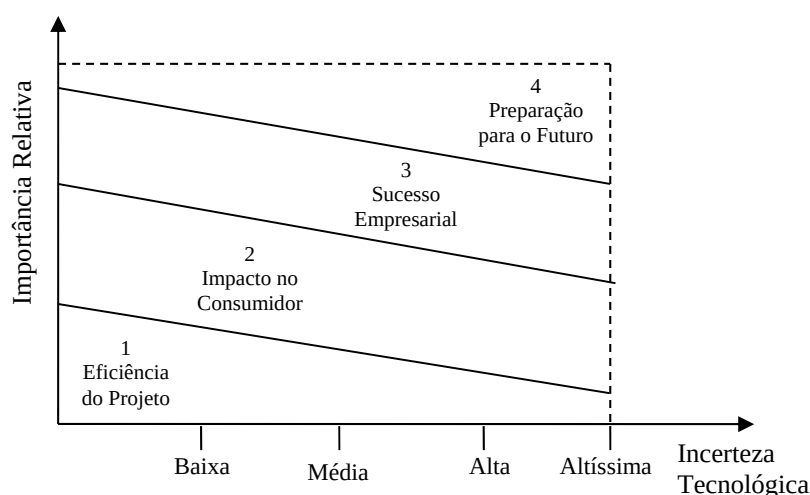


Figura 7 (2) – Importância relativa das dimensões de sucesso de acordo com a incerteza tecnológica.

Fonte: Shenhar et al. (2001, tradução nossa).

Em projetos de baixa tecnologia, a eficiência do projeto é crítica, assim como o impacto no consumidor e o sucesso empresarial; a dimensão preparação para o futuro tem pouca relevância. Na medida em que a incerteza tecnológica aumenta, a importância relativa da eficiência do projeto diminui, a da dimensão preparação para o futuro aumenta e as outras duas dimensões ficam constantes. Assim, em projetos de altíssima tecnologia, a dimensão eficiência do projeto tem valor reduzido, enquanto a dimensão preparação para o futuro tem sua importância potencializada, por estar associada à geração de novas competências e obtenção de vantagem competitiva.

No contexto de projetos de *software*, quatro dimensões de sucesso baseadas em Shenhar et al. (2001) e Shenhar e Dvir (2007) são analisadas por Dalcher (2009), com relação ao foco interno/externo e saída/resultado. No Quadro 11 está apresentada a perspectiva do autor no tocante a esses critérios.

Quadro 11 (2) – Foco x saída/resultado das dimensões de desempenho

Foco x Saída/Resultado	Saída	Resultado
Foco Externo	Dimensão 2 Qualidade das entregas e outras saídas (visão do <i>stakeholder</i>).	Dimensão 4 Potencial de negócios e implicações mais amplas (para <i>stakeholders</i>).
Foco Interno	Dimensão 1 Restrições e medidas internas.	Dimensão 3 Valor empresarial interno de acordo com o investimento no projeto.

Fonte: Dalcher (2009, tradução nossa)

Conforme pode ser observado no Quadro 11, as dimensões de eficiência no projeto e sucesso empresarial possuem foco interno, enquanto impacto no consumidor e preparação para o futuro focam em *stakeholders* externos. O autor também diferencia saídas de resultados, sendo as saídas fruto de um processo em função das entregas previstas e os resultados, efeitos de mudança e valor entregue para o negócio como um todo, além do escopo do projeto original.

Além desses aspectos, Dalcher (2009) também analisa o foco de cada dimensão em função da efetividade/eficiência e do horizonte temporal, conforme apresentado no Quadro 12.

Quadro 12 (2) – Efetividade/eficiência x orientação temporal das dimensões de desempenho

Efetividade/Eficiência x Orientação Temporal	Curto Prazo	Longo Prazo
Efetividade	Dimensão 2 Qualidade da saída, atendimento às reais necessidades e preocupações dos <i>stakeholders</i> .	Dimensão 4 Alcance dos objetivos empresariais, melhor horizonte de qualidade como foco de melhoria, investimento como grande benefício.
Eficiência	Dimensão 1 Eficiência do projeto: eficiência interna em entregar dentro das restrições, minimização de recursos, foco em procedimentos, execução do projeto.	Dimensão 3 Determinação da eficiência financeira, valor de negócio e retorno do investimento.

Fonte: Dalcher (2009, tradução nossa)

Conforme observado no Quadro 12, as dimensões 1 e 3 estão ligadas a eficiência, uma vez que focam no consumo de recursos. As dimensões 2 e 4 estão ligadas a eficácia, ou seja, nos resultados. A análise com relação ao prazo é semelhante à feita por Shenhar et al. (2001).

Conforme exposto ao longo dessa seção, o conceito de falhas em projetos sofreu diversas mudanças ao longo do tempo, passando de uma abordagem voltada à gestão do projeto para uma voltada ao projeto como um todo, ocupando-se de todo o ciclo de vida do produto e tornando-se, portanto, mais completo. Isso pode ser ilustrado na Figura 8, que apresenta a abrangência do conceito de sucesso ao longo dos ciclos de vida.

CICLO DE VIDA DO PROJETO					
CICLO DE VIDA DO PRODUTO					
Concepção	Planejamento	Produção/ Implementação	Entrega	Utilização	Encerramento
		Período 1: Implementação e Entrega do Projeto			
		Período 2: Listas de FCS (1980s – 1990s)			
	Período 3: Frameworks de FCS (1990s – 2000s)				
Período 4: Gerenciamento Estratégico de Projetos (século XXI)					

Figura 8 (2) – Medindo sucesso ao longo dos ciclos de vida do projeto e do produto.

Fonte: Adaptada de Jugdev e Müller (2005).

A Figura 8 permite observar a evolução do conceito de sucesso ao longo dos quatro períodos da gestão de projetos propostos por Jugdev e Müller (2005). O período 1 considerava sucesso apenas a implementação e entrega do projeto. No período 2, surgiram várias listas de fatores críticos de sucesso. O período 3 trouxe maior consistência a esses fatores, a partir da proposição de *frameworks*, ao invés de simples listas. Por fim, o período 4 trouxe uma visão mais estratégica à gestão de projetos.

Os fatores críticos de sucesso são aspectos que podem levar um projeto a ser bem ou mal sucedido. Vários artigos elencam alguns desses critérios (PINTO; SLEVIN, 1987; BELASSI; TUKEL, 1996; COOKE-DAVIES, 2002; DILTS; PENCE, 2006; PAPKE-SHIELDS et al., 2010), uns em projetos de tipos específicos e outros em projetos gerais. No período 2 da figura, os estudos elencavam os fatores críticos de sucesso, mas não os integravam de forma coerente (JUGDEV; MÜLLER, 2005).

A partir do período 3 é que passaram a haver modelos mais elaborados, como o de Belassi e Tukul (1996). Os autores estruturaram os fatores em 04 grupos: projeto (características e ciclo de vida), gerente de projetos/equipe (habilidades e competências diversas), organização (estrutura organizacional e apoio da gestão) e ambiente externo (aspectos políticos, econômicos, sociais, tecnológicos, clientes, concorrentes e competidores).

É importante ressaltar que a categorização da Figura 8 é para projetos em geral, não especificamente de *software*. Além disso, ao categorizar desse modo, há a possibilidade de cometer inadequações na classificação, uma vez que as pesquisas não são realizadas de forma coordenada no mundo inteiro, bem como para determinados problemas de pesquisa, conceitos de períodos anteriores podem ser suficientes. Por esse motivo, há estudos publicados em determinados anos que não correspondem ao respectivo período citado, como Atkinson (1999), que apresenta elementos ligados a uma perspectiva estratégica ainda no século XX, Siau et al. (2010) ou Tarawneh (2011) que adota critérios bastante simples mesmo sendo recente.

Outro aspecto relevante ligado à conceituação de falhas em projetos de TI é que sua compreensão varia de acordo com o indivíduo ou organização em consideração, bem como o contexto. Alguns estudos analisam a percepção de membros da organização, como Wateridge (1998) que estudou gerentes de projetos e de usuários, Linberg (1999), que pesquisou a visão desenvolvedores e Standing et al. (2006), que analisou diferenças entre a alta gestão, gerentes de linha e equipe de TI. Outros estudos focaram em variações de acordo com a cultura, como Procaccino e Verner (2002) e Pereira et al. (2008), que fizeram um estudo similar nos Estados Unidos e no Chile, respectivamente. Por fim, outro grupo de estudos abordou a visão de organizações, como Ahonen e Savolainen (2011) que pesquisaram a perspectiva de clientes e fornecedores e Savolainen et al. (2012), que focaram na visão de fornecedores. A literatura apresenta três critérios de sucesso na visão dos fornecedores de *software*: satisfação do cliente, benefícios ao negócio em curto prazo e benefícios ao negócio em longo prazo; apesar disso, não há definições de falhas em projetos de desenvolvimento de *software* (SAVOLAINEN et al., 2012).

No Quadro 13, há um resumo com alguns dos artigos que abordam o conceito de desempenho/falhas em projetos *software* que foram apresentados nessa seção.

Quadro 13 (2) – Artigos que abordam o conceito de desempenho/falha em projetos de *software* (Continua)

Obras	Sucesso/Falha	Observações
Pinto e Slevin (1988)	Indicadores internos (tempo, custo e especificações) e externos (uso, satisfação e eficácia para os usuários).	Agrupa os indicadores em duas categorias: internos e externos.
Wateridge (1995)	Atendimento aos requisitos dos usuários, satisfação dos usuários, orçamento, cronograma, atendimento ao objetivo do projeto, sucesso comercial e qualidade.	Apesar de não agrupar em dimensões ou categorias, aborda aspectos que vão desde uma perspectiva micro até uma visão macro, embora despreze alguns <i>stakeholders</i> , como os patrocinadores e a equipe.

(Continuação)

Obras	Sucesso/Falha	Observações
Wateridge (1998)	Atendimento aos requisitos dos usuários, atendimento ao objetivo do projeto, prazo, orçamento, qualidade, satisfação dos usuários, sucesso comercial, satisfação do patrocinador e satisfação da equipe.	Apesar de não agrupar em dimensões ou categorias, aborda aspectos que vão desde uma perspectiva micro até uma visão macro.
Dvir et al. (1998)	Atendimento às metas do projeto (especificações funcionais, especificações técnicas, tempo e custo) e benefícios ao consumidor (atendimento das metas de aquisição, atendimento dos requisitos operacionais, entrada do produto em operação, alcance do consumidor final em tempo hábil, uso prolongado do produto, aumento no nível operacional do usuário e satisfação do usuário).	Agrupa os indicadores em duas categorias: atendimento às metas do projeto e benefícios ao consumidor.
Atkinson (1999)	Triângulo de ferro (custo, qualidade e tempo), sistema de informação (manutenibilidade, confiabilidade, validade e qualidade da informação), benefícios para a organização (aumento da eficiência, efetividade e lucro, alcance dos objetivos estratégicos, aprendizagem organizacional e redução do desperdício) e benefícios para a comunidade de <i>stakeholders</i> (satisfação dos usuários, impacto social e ambiental, desenvolvimento pessoal e profissional, lucro do contratante e impacto econômico na comunidade).	Aborda 04 dimensões, que evoluem de uma perspectiva micro (interna ao projeto) até uma perspectiva macro (externa, envolvendo outros <i>stakeholders</i>).
Linberg (1999)	Em projetos concluídos, atendimento à expectativa de qualidade, custo, esforço e cronograma. Em projetos abortados, aprendizado e criação de artefatos que podem ser usados em projetos futuros.	Aborda aspectos variam de acordo com a conclusão ou não do projeto, na visão dos desenvolvedores.
Shenhar et al. (2001)	Eficiência do projeto (cronograma e orçamento), impacto no consumidor (funcionalidades, especificações, resolução de um problema do cliente, uso pelo cliente e satisfação do cliente), sucesso empresarial (sucesso comercial e alto <i>market share</i>) e preparação para o futuro (criação de um novo mercado, criação de uma nova linha de produtos e desenvolvimento de uma nova tecnologia).	Aborda 05 dimensões, cuja importância varia de acordo com a perspectiva de tempo e o grau de incerteza.

(Continuação)

Obras	Sucesso/Falha	Observações
Procaccino e Verner (2002) e Pereira et al. (2008)	Produto fácil de usar, atendimento aos requisitos do cliente/usuário, precisão da estimativa de tempo e cronograma, conclusão dentro do orçamento.	Aborda aspectos tradicionais, internos ao projeto e ligados a eficiência e relativos ao cliente/usuário, além de elementos pessoais e ligados ao processo (estes não citados neste quadro por estarem fora do escopo desta pesquisa).
Jiang et al. (2002) e Tesch et al. (2009)	Atingimento das metas do projeto, conclusão da quantidade de trabalho esperado, alta qualidade do trabalho concluído, aderência ao cronograma, aderência ao orçamento, operações eficientes.	Aborda apenas aspectos tradicionais, internos ao projeto e ligados a eficiência e o critério atingimento das metas do projeto, que não é preciso.
Agarwal e Rathod (2006)	Escopo (funcionalidade e qualidade), tempo e custo.	Aborda apenas aspectos tradicionais, internos ao projeto e ligados a eficiência.
Standing et al. (2006)	Sucesso/falha.	Instrumento de coleta utilizado não deixa claro os indicadores de sucesso/falha. Eles são abordados como um construto único, medido em uma escala <i>likert</i> de 7 pontos.
Shenhar et al. (2001)	Eficiência (cronograma, orçamento, lucratividade e eficiência na produção), impacto no consumidor (funcionalidades, especificações, benefícios para o cliente, uso pelo cliente, satisfação e lealdade do cliente e reconhecimento da marca), impacto na equipe (satisfação e moral da equipe, desenvolvimento de habilidades, crescimento dos membros, retenção e ausência de exaustão), sucesso empresarial (vendas, lucro, <i>market share</i> , ROI, ROE, fluxo de caixa, medidas organizacionais e aprovação regulamentar) e preparação para o futuro (novas tecnologias, novos mercados, nova linha de produtos, nova competência e nova capacidade organizacional).	Aborda 05 dimensões, cuja importância varia de acordo com a perspectiva de tempo e o grau de incerteza.
Chow e Cao (2008)	Qualidade, escopo, tempo e custo.	Aborda apenas aspectos tradicionais, internos ao projeto e ligados a eficiência.
El-Emam e Koru (2008)	Satisfação do usuário, orçamento, cronograma, qualidade e produtividade.	Aborda aspectos tradicionais, internos ao projeto e ligados a eficiência e satisfação do usuário.

(Conclusão)

Obras	Sucesso/Falha	Observações
Thomas e Fernández (2008)	Gestão do projeto (tempo, orçamento, satisfação do patrocinador, satisfação da gestão, satisfação da equipe, satisfação do cliente/usuário e satisfação dos <i>stakeholders</i>), aspectos técnicos (satisfação do cliente/usuário, satisfação dos <i>stakeholders</i> , implementação do sistema, requisitos, qualidade e uso do sistema) e de negócios (continuidade do negócio, alcance dos objetivos de negócio e entrega de benefícios).	Aborda 03 categorias de critérios, que evoluem de uma perspectiva micro/ligada ao projeto a uma perspectiva macro/ligada a negócios.
Dalcher (2009)	Sucesso da gestão do projeto (medidas e restrições internas), sucesso do projeto (qualidade das entregas e outras saídas, na visão dos <i>stakeholders</i>), sucesso do negócio (percepção do valor do negócio devido ao projeto) e sucesso potencial futuro (potencial de negócios e implicações mais amplas para os <i>stakeholders</i>).	Aborda 04 dimensões, que evoluem de uma perspectiva micro/interna e de curto prazo a uma perspectiva macro/externa e de longo prazo.
Siau et al. (2010)	Uso do sistema, sucesso do projeto (custo, duração, especificação e eficiência do processo), sucesso do sistema (qualidade do sistema, qualidade das informações/dados e uso do sistema) e satisfação do usuário.	Aborda aspectos tradicionais, internos ao projeto e ligados a eficiência e ao uso/satisfação do usuário.
Ahonen e Savolainen (2011)	Sucesso/Falha.	Não adota definição algum. Faz uma revisão da literatura a partir de outros estudos que mencionem alguma definição de sucesso e/ou falha.
McLeod e MacDonell (2011)	Crítérios técnicos, econômicos, comportamentais, psicológicos e políticos.	Analisa o resultado do projeto de acordo com o tipo de resultado, podendo ser o processo de desenvolvimento, de implementação, o produto em si ou o sistema/solução completa. Ressalta que o conceito é subjetivo, contestado/negociado e temporal, dependendo do contexto histórico.
Tarawneh (2011)	Tempo, orçamento, custo/esforço, qualidade do sistema, satisfação do usuário, atendimento a requisitos e uso do sistema.	Aborda aspectos tradicionais, internos ao projeto e ligados a eficiência e ao uso/satisfação do usuário.
Savolainen et al. (2012)	Satisfação do cliente, benefícios ao negócio em curto prazo e benefícios ao negócio em longo prazo.	Revisão de literatura que aborda sucesso na perspectiva do fornecedor. Afirma não haver na literatura conceito de falha nesse contexto.

Fonte: autor.

Para cada artigo apresentado, o Quadro 13 destaca o conceito de sucesso/falha utilizado pelos autores e algumas observações referentes a ele. Conforme pode ser observado, a concepção de falha não segue uma evolução cronológica direta.

O conceito de falha adotada nesta pesquisa parte da definição de Petroski (2006, p. 51, tradução nossa), segundo o qual é “uma diferença inaceitável entre o desempenho previsto e o observado”. Essa diferença pode ser verificada nos critérios e dimensões apresentados nessa seção. Por inaceitável, compreende-se que o desempenho não esteja dentro do limite inferior de tolerância dos *stakeholders* do projeto.

Os projetos podem apresentar falhas em diferentes frequências, em um contínuo que varia de baixa (falhas raras) a alta (falhas frequentes). Além disso, as falhas variam de acordo com a gravidade, relacionada às suas consequências. Flowers (1996) aponta que elas variam desde aquelas em que o *software* tem deficiências mas é utilizado, até os “desastres absolutos”.

2.1.4 Modelos de maturidade em organizações de desenvolvimento de *software*

Com o intuito de reduzir as falhas que ocorrem nos projetos de *software*, as organizações de desenvolvimento adotam técnicas e modelos que lhes permitem alcançar melhores resultados. O *Scrum* e PMBOK, apresentados na seção 2.1.2, são alguns dos modelos utilizados para gestão de projetos.

Além das metodologias para gestão, há os modelos de maturidade. A maturidade de uma organização indica o grau de evolução de seus processos (SOFTEX, 2012). Para incrementá-los, as organizações de desenvolvimento de *software* adotam modelos de melhoria como o CMMI ou o MPS.BR. Eles definem os graus de maturidade da organização e fornecem subsídios para mensuração e implementação de aperfeiçoamentos nos processos.

O CMMI e seu antecessor, Capability Maturity Model (CMM), foram desenvolvidos pelo *Software Engineering Institute* (SEI) da Universidade Carnegie Mellon. Ambos são conjuntos de melhores práticas relacionadas à maturidade de desenvolvimento em organizações de *software*, descrevendo os principais elementos do processo e definindo níveis de maturidade.

O CMMI possui 22 áreas de processos e duas formas de melhoria: contínua, em que a organização incrementa os processos de uma área específica (ou grupo de áreas), e em estágios, em que a organização melhora um conjunto de processos relacionados através de

conjuntos sucessivos de áreas. A representação contínua possui 04 níveis de capacidade: incompleto, executado, gerenciado e definido, enquanto a representação por estágios possui 05 níveis de maturidade: inicial, gerenciado, definido, gerenciado quantitativamente e otimizado. Para alcançar determinado nível, uma organização deve satisfazer todas as metas das áreas de processos alvos da melhoria. Uma vez alcançado um nível, a organização pode obter uma certificação referente a ele (CMMI, 2010). O Quadro 14 apresenta os níveis e a comparação entre eles:

Quadro 14 (2) – Comparação entre os níveis de capacidade e de maturidade

Níveis	Representação contínua Níveis de capacidade	Representação em estágios Níveis de maturidade
Nível 0	Incompleto	
Nível 1	Executado	Inicial
Nível 2	Gerenciado	Gerenciado
Nível 3	Definido	Definido
Nível 4		Gerenciado quantitativamente
Nível 5		Em otimização

Fonte: CMMI (2010, tradução nossa).

Conforme pode ser observado no Quadro 14, a representação contínua possui quatro níveis, dos quais dois (2 e 3) coincidem com a representação em estágios. Esta, por sua vez, possui cinco níveis.

O CMMI foi desenvolvido em um contexto internacional e o investimento para certificação costuma ser alto para pequenas e médias organizações. Por conta disso, a SOFTEX, uma associação que promove o desenvolvimento do *software* brasileiro, criou o MPS.BR, em parceria com o governo e universidades. Esse modelo de qualidade se baseia nas normas ISO/EIC 12207 e ISO/EIC 15504 e é compatível com o CMMI. Apesar disso, possui mais níveis que ele, o que torna a distância entre os níveis menor. Os níveis dos dois modelos e sua associação estão apresentados no Quadro 15.

Quadro 15 (2) – Comparação entre os níveis do MPS.BR e do CMMI

MPS.BR	CMMI
G - parcialmente gerenciado	Entre 1 e 2
F - gerenciado	2
E - parcialmente definido	Entre 2 e 3
D - largamente definido	Entre 2 e 3
C - definido	3
B - quantitativamente gerenciado	4
A - otimizado	5

Fonte: autor, a partir de Montoni et al. (2009).

Segundo SOFTEX (2012), cada nível de maturidade possui processos, que são definidos em função de seus objetivos e resultados. Quanto maior a maturidade, maior a capacidade do processo, que “expressa o grau de refinamento e institucionalização com que o processo é executado na organização/unidade organizacional” (SOFTEX, 2012, p. 18). Esta capacidade é descrita por nove atributos de processo, que são apresentados no Quadro 16, juntamente com o número de resultados esperados (RAP) e respectivos níveis:

Quadro 16 (2) – Atributos de Processo do MPS.BR

Atributo de Processo	RAP	Níveis
AP 1.1 O processo é executado	1	G, F, E, D, C, B, A
AP 2.1 O processo é gerenciado	9	F, E, D, C, B, A
AP 2.2 Os produtos de trabalho do processo são gerenciados	4	F, E, D, C, B, A
AP 3.1 O processo é definido	4	E, D, C, B, A
AP 3.2 O processo está implementado	3	E, D, C, B, A
AP 4.1 O processo é medido	8	B, A
AP 4.2 O processo é controlado	5	B, A
AP 5.1 O processo é objeto de melhorias incrementais e inovações	8	A
AP 5.2 O processo é otimizado continuamente	4	A

Fonte: autor, a partir de SOFTEX (2012).

Os resultados esperados dos atributos de processo indicam o alcance do respectivo atributo e variam de acordo com o nível de maturidade. Desse modo, a expectativa com relação a cada atributo varia conforme o estágio de maturidade da organização.

Além de ter seus atributos definidos, os níveis possuem seus respectivos processos, que são apresentados no Quadro 17.

Quadro 17 (2) – Processo do MPS.BR

(Continua)

Processo	Níveis
Gerência de Requisitos (GRE)	G, F, E, D, C, B, A
Gerência de Projetos (GPR)	G, F, E, D, C, B, A
Medição (MED)	F, E, D, C, B, A
Garantia de Qualidade (GQA)	F, E, D, C, B, A
Gerência de Portfolio (GPP)	F, E, D, C, B, A
Gerência de Configuração (GCO)	F, E, D, C, B, A
Aquisição (AQU)	F, E, D, C, B, A
Gerência de Projetos (GPR) – evolução	E, D, C, B, A
Gerência de Reutilização (GRU)	E, D, C, B, A
Gerência de Recursos Humanos (GRH)	E, D, C, B, A
Definição do Processo Organizacional (DFP)	E, D, C, B, A
Avaliação e Melhoria do Processo Organizacional (AMP)	E, D, C, B, A
Verificação (VER)	D, C, B, A
Validação (VAL)	D, C, B, A

(Conclusão)

Processo	Níveis
Projeto e Construção do Produto (PCP)	D, C, B, A
Integração do Produto (ITP)	D, C, B, A
Desenvolvimento de Requisitos (DRE)	D, C, B, A
Gerenciamento de Riscos (GRI)	C, B, A
Desenvolvimento para Reutilização (DRU)	C, B, A
Gerenciamento de Decisões (GDE)	C, B, A
Gerência de Projetos (GPR) – evolução	B, A

Fonte: autor, a partir de SOFTEX (2012).

Cada um destes processos é definido em função de seus propósitos e resultados esperados. Vários deles estão ligados a esta pesquisa, uma vez que se associam às rotinas de gestão de projetos. Um deles tem destaque especial: o processo de Gerência de Projetos, por estar diretamente ligado à temática do estudo. Ele é apresentado em 3 níveis distintos, cada um dos quais agregando novos resultados esperados. O propósito deste processo é o seguinte:

(...) estabelecer e manter planos que definem as atividades, recursos e responsabilidades do projeto, bem como prover informações sobre o andamento do projeto que permitam a realização de correções quando houver desvios significativos no desempenho do projeto. O propósito deste processo evolui à medida que a organização cresce em maturidade. Assim, a partir do nível E, alguns resultados evoluem e outros são incorporados, de forma que a gerência de projetos passe a ser realizada com base no processo definido para o projeto e nos planos integrados. No nível B, a gerência de projetos passa a ter um enfoque quantitativo, refletindo a alta maturidade que se espera da organização. Novamente, alguns resultados evoluem e outros são incorporados (SOFTEX, 2012, p. 26).

O processo possui 28 resultados esperados, apresentados no Quadro 18:

Quadro 18 (2) – Resultados esperados do processo Gerência de Projetos do MPS.BR

(Continua)

Código	Descrição
GPR 1	O escopo do trabalho para o projeto é definido;
GPR 2	As tarefas e os produtos de trabalho do projeto são dimensionados utilizando métodos apropriados;
GPR 3	O modelo e as fases do ciclo de vida do projeto são definidos;
GPR 4	(Até o nível F) O esforço e o custo para a execução das tarefas e dos produtos de trabalho são estimados com base em dados históricos ou referências técnicas;
	(A partir do nível E) O planejamento e as estimativas das tarefas do projeto são feitos baseados no repositório de estimativas e no conjunto de ativos de processo organizacional;
GPR 5	O orçamento e o cronograma do projeto, incluindo a definição de marcos e pontos de controle, são estabelecidos e mantidos;
GPR 6	Os riscos do projeto são identificados e o seu impacto, probabilidade de ocorrência e prioridade de tratamento são determinados e documentados;

(Conclusão)

Código	Descrição
GPR 7	Os recursos humanos para o projeto são planejados considerando o perfil e o conhecimento necessários para executá-lo;
GPR 8	(Até o nível F) Os recursos e o ambiente de trabalho necessários para executar o projeto são planejados;
	(A partir do nível E) Os recursos e o ambiente de trabalho necessários para executar os projetos são planejados a partir dos ambientes padrão de trabalho da organização;
GPR 9	Os dados relevantes do projeto são identificados e planejados quanto à forma de coleta, armazenamento e distribuição. Um mecanismo é estabelecido para acessá-los, incluindo, se pertinente, questões de privacidade e segurança;
GPR 10	Um plano geral para a execução do projeto é estabelecido com a integração de planos específicos;
GPR 11	A viabilidade de atingir as metas do projeto é explicitamente avaliada considerando restrições e recursos disponíveis. Se necessário, ajustes são realizados;
GPR 12	O Plano do Projeto é revisado com todos os interessados e o compromisso com ele é obtido e mantido;
GPR 13	O escopo, as tarefas, as estimativas, o orçamento e o cronograma do projeto são monitorados em relação ao planejado;
GPR 14	Os recursos materiais e humanos bem como os dados relevantes do projeto são monitorados em relação ao planejado;
GPR 15	Os riscos são monitorados em relação ao planejado;
GPR 16	O envolvimento das partes interessadas no projeto é planejado, monitorado e mantido;
GPR 17	Revisões são realizadas em marcos do projeto e conforme estabelecido no planejamento;
GPR 18	Registros de problemas identificados e o resultado da análise de questões pertinentes, incluindo dependências críticas, são estabelecidos e tratados com as partes interessadas;
GPR 19	Ações para corrigir desvios em relação ao planejado e para prevenir a repetição dos problemas identificados são estabelecidas, implementadas e acompanhadas até a sua conclusão;
GPR 20	(A partir do nível E) Equipes envolvidas no projeto são estabelecidas e mantidas a partir das regras e diretrizes para estruturação, formação e atuação;
GPR 21	(A partir do nível E) Experiências relacionadas aos processos contribuem para os ativos de processo organizacional;
GPR 22	(A partir do nível E) Um processo definido para o projeto é estabelecido de acordo com a estratégia para adaptação do processo da organização;
	(A partir do nível B) Os objetivos de qualidade e de desempenho do processo definido para o projeto são estabelecidos e mantidos;
GPR 23	(A partir do nível B) O processo definido para o projeto que o possibilita atender seus objetivos de qualidade e de desempenho é composto com base em técnicas estatísticas e outras técnicas quantitativas;
GPR 24	(A partir do nível B) Subprocessos e atributos críticos para avaliar o desempenho e que estão relacionados ao alcance dos objetivos de qualidade e de desempenho do processo do projeto são selecionados;
GPR 25	(A partir do nível B) Medidas e técnicas analíticas são selecionadas para serem utilizadas na gerência quantitativa;
GPR 26	(A partir do nível B) O desempenho dos subprocessos escolhidos para gerência quantitativa é monitorado usando técnicas estatísticas e outras técnicas quantitativas;
GPR 27	(A partir do nível B) O projeto é gerenciado usando técnicas estatísticas e outras técnicas quantitativas para determinar se seus objetivos de qualidade e de desempenho do processo serão atingidos;
GPR 28	(A partir do nível B) Questões que afetam os objetivos de qualidade e de desempenho do processo do projeto são alvo de análise de causa raiz.

Fonte: a partir de SOFTEX (2012).

Segundo Montoni et al. (2009), o MPS.BR tem promovido a melhoria do desenvolvimento de *software* no Brasil. Isso possivelmente está associado à sua adaptação à realidade nacional, com um custo de certificação mais acessível para as pequenas e médias organizações nacionais do que os outros modelos, mais populares internacionalmente. Além disso, a existência de mais níveis facilita a implantação por organizações de menor porte e permite visualizar os resultados da implantação em um horizonte temporal menor (SOFTEX, 2012).

2.2 Rotinas organizacionais

A literatura que trata das rotinas organizacionais possui várias ambiguidades, razão pela qual definir seu conceito e compreender seus efeitos nas organizações não é uma tarefa simples (BECKER, 2004). Milagres (2011) ressalta que, muitas vezes, a utilização do termo é feita sem explicitar o conceito ou fazendo-o de forma muito abrangente, o que inibe a compreensão do que está sendo analisado. Segundo Becker et al. (2005) e Becker e Zirpoli (2008), podem ser encontradas três diferentes perspectivas na literatura, segundo as quais o conceito de rotinas está baseado em padrões de comportamento, regras e disposições coletivas, respectivamente.

Na primeira das abordagens, rotinas organizacionais são concebidas como um padrão repetitivo de comportamento coletivo, embora aí estejam também incluídos autores que as consideram como um padrão repetitivo de ações ou interações (BECKER, 2004). Na segunda das perspectivas, as rotinas organizacionais são compreendidas como regras, ou seja, procedimentos operacionais padrão (BECKER et al., 2005). Nessa visão, tais procedimentos podem até gerar comportamentos repetitivos, mas esse não é o foco da análise. Por fim, as rotinas também podem ser analisadas como disposições coletivas que levam os indivíduos a agirem conforme comportamentos prévios (BECKER, 2004).

Segundo Feldman e Pentland (2003), as rotinas organizacionais são o elemento central da burocracia, característica das organizações modernas, e permitem que ela organize o conhecimento e exercite o poder de forma eficiente. Vários autores apontam que elas são o principal meio através do qual as organizações alcançam seus objetivos (FELDMAN; PENTLAND, 2003), podendo ser responsáveis tanto pela inércia e estabilidade, quanto pela flexibilidade e mudança nas organizações (PENTLAND; FELDMAN, 2005). A literatura aponta ainda que elas são imprescindíveis para reestabelecer a ordem em situações de crises (INAM, 1999).

Segundo Feldman e Pentland (2003, p. 95, tradução nossa), “as rotinas organizacionais podem ser definidas como um padrão repetitivo identificável de ações interdependentes, envolvendo múltiplos atores”, conceito adotado nesta pesquisa. Nessa definição, quatro elementos são centrais. O primeiro deles é o padrão identificável de ações, que indica que as ações realizadas pelos indivíduos devem constituir um padrão cuja identificação seja possível para que sejam consideradas uma rotina organizacional.

O segundo elemento central da definição adotada é a repetição. Rotinas devem ser constituídas por ações repetitivas, ou seja, que ocorram sucessivas vezes. Um padrão identificável de ações que ocorra uma única vez não constitui uma rotina organizacional, e sim um comportamento isolado. O terceiro aspecto é o envolvimento de múltiplos atores. Para serem rotinas organizacionais, as ações precisam ser realizadas por múltiplos indivíduos. Caso contrário, seriam hábitos, ou seja, padrões de comportamentos individuais.

Por fim, a definição apresentada por Feldman e Pentland (2003) também destaca a interdependência das ações envolvidas nas rotinas organizacionais. Isso indica que as atividades que as constituem dependem umas das outras, ou seja, não são atos isolados desempenhados pelos indivíduos.

Além desses aspectos, Becker (2004) acrescenta outras características às rotinas. Uma delas é a natureza processual, área em que o autor aponta terem ocorrido as mais ricas contribuições por parte das pesquisas empíricas. A dependência de contexto e historicidade são outras de suas características, uma vez que as rotinas estão imbricadas na organização e sua estrutura, e são específicas do ponto de vista local, relacional e histórico, sendo resultado de um processo histórico de construção.

A intencionalidade é uma das características citadas por Becker (2004) sobre a qual não há consenso. Enquanto alguns estudos apontam as rotinas como ações intencionais, outros consideram que elas ocorrem no mundo do subconsciente. Feldman e Pentland (2003) apontam que a contradição com relação à intencionalidade ocorre por existirem dois aspectos nas rotinas organizacionais: o ostensivo e o performativo. O aspecto ostensivo diz respeito à ideia geral ou princípio da rotina, enquanto o aspecto performativo consiste em “ações específicas, por pessoas específicas, em locais e momentos específicos” (FELDMAN; PENTLAND, 2003, p. 101, tradução nossa).

Desse modo, o aspecto ostensivo está associado à estrutura geral da rotina, aquilo que faz com que os indivíduos reconheçam um dado conjunto de ações como representativo de determinada rotina e não de outra. Por outro lado, o aspecto performativo é a rotina em ação,

uma instância específica dela em dado contexto. Segundo Ellström (2010), tensões entre os dois aspectos criam as condições para aprendizagem e inovação nas organizações.

Além dos aspectos ostensivo e performativo, Pentland e Feldman (2005) destacam os artefatos como importantes elementos associados às rotinas organizacionais. Segundo os autores, esses são manifestações físicas das rotinas, podendo tanto estar ligados à dimensão ostensiva, quanto à performativa. Regras e procedimentos formais são artefatos que representam o aspecto ostensivo das rotinas, enquanto relatórios e registros exemplificam artefatos associados ao performativo.

O aspecto performativo, os artefatos ligados ao ostensivo e o aspecto ostensivo de Pentland e Feldman (2005) correspondem respectivamente às três diferentes perspectivas da literatura apontadas por Becker et al. (2005) e Becker e Zirpoli (2008) citadas no início desta seção. Esta visão é adotada por outros autores, como Weichbrodt e Grote (2010), que analisam as organizações sob a ótica de regras explícitas (artefatos), rotinas na prática (comportamento, aspecto performativo) e rotinas em princípio (cognição, aspecto ostensivo, regras não escritas).

Segundo Feldman e Pentland (2003), os aspectos ostensivo e performativo são distintos, uma vez que um padrão de rotina é incapaz de especificar todas as possíveis variações que ocorrem em sua execução na prática, devido ao caráter de imprevisto desta (BOURDIEU, 1997 apud FELDMAN; PENTLAND, 2003). Os dois relacionam-se de acordo com o esquema apresentado na Figura 9:

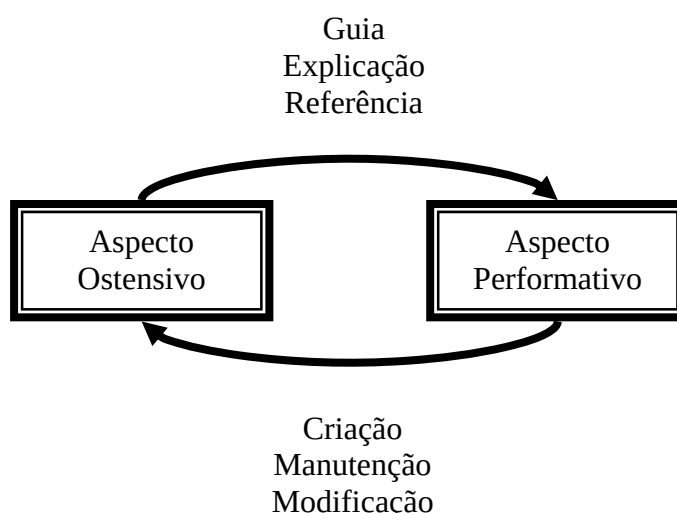


Figura 9 (2) – Relação entre os aspectos ostensivo e performativo das rotinas organizacionais.
Fonte: autor, a partir de Feldman e Pentland (2003).

Conforme pode ser observado na Figura 9, o aspecto ostensivo das rotinas organizacionais pode atuar como guia para as ações a serem desempenhadas, como base para

explicação das ações já realizadas ou como referência para dar significado às ações. Por outro lado, o aspecto performativo atua na criação e na manutenção das rotinas, uma vez que é necessária a sua repetição para que uma rotina se constitua e continue existindo, além da modificação, a partir de mudanças no padrão de execução das atividades (FELDMAN; PENTLAND, 2003).

D’Adderio (2008) também aborda a relação entre os aspectos ostensivo e performativo, destacando que eles interagem por meio de ciclos de enquadramento, transbordamento e reenquadramento. No enquadramento, o aspecto ostensivo age como guia e controle para o performativo, que “transborda”, sofrendo adaptação e divergindo dele. A partir daí, há o reenquadramento, que busca convergência entre os dois aspectos.

Esse ciclo ocorre de forma recorrente, constituindo, portanto, evolução nas rotinas organizacionais. As mudanças compõem o foco deste estudo e serão aprofundadas na seção 2.2.1. Elas ocorrem tanto do aspecto performativo para o ostensivo quanto no sentido contrário, utilizando a relação de guia apresentada na Figura 9. Segundo D’Adderio (2008), os artefatos desempenham um importante papel na manutenção das rotinas, especialmente se houver componentes tecnológicos.

Gersick e Hackman (1990) destacam três modos pelos quais um grupo pode adotar determinada rotina: importação, criação no primeiro encontro e evolução ao longo do tempo. Segundo os autores, a importação ocorre quando há procedimentos operacionais padrão para determinadas circunstâncias ou quando os indivíduos possuem experiências prévias ou normas culturais comuns, fazendo com que hajam de determinada forma. A criação no primeiro encontro ocorre quando a tarefa é nova, os membros são heterogêneos e/ou a organização não possui rotinas predefinidas. Nesse caso, os indivíduos exibem comportamentos que posteriormente constituem rotinas. Por fim, na evolução ao longo do tempo, os padrões mudam incrementalmente na medida em que os indivíduos agem e obtêm experiência (GERSICK; HACKMAN, 1990).

No Quadro 19 estão apresentadas alguns estudos que abordam rotinas, juntamente com seu objetivo e a concepção de rotinas exposta por cada um deles:

Quadro 19 (2) – Algumas abordagens relacionadas a rotinas organizacionais

Estudo	Objetivo do Estudo	Rotinas como
Gersick e Hackman (1990)	Explorar o comportamento rotineiro de grupos que atuam em organizações, definindo rotinas, identificando suas funções e disfunções, como são desenvolvidas, mantidas ou alteradas.	Padrão de comportamento automático e repetitivo.
Feldman e Pentland (2003)	Conceituar rotinas como fontes de flexibilidade e mudança.	Aspecto ostensivo e performativo.
Becker (2004)	Prover uma revisão sistemática da literatura.	Regularidades comportamentais e regularidades cognitivas.
Becker et al. (2005)	Fortalecer rotinas organizacionais como perspectiva analítica para compreender o comportamento e a mudança organizacionais.	Padrões comportamentais, regras e disposições coletivas.
Becker e Zirpoli (2008)	Mostrar como utilizar rotinas em estudos empíricos pode contribuir para a compreensão do comportamento das organizações e demonstrar o seu potencial.	Padrões comportamentais, regras e disposições coletivas.
Pentland e Feldman (2005)	Propor modos de abordar rotinas organizacionais como unidade de análise.	Aspecto ostensivo, performativo e artefatos.
D’Adderio (2008)	Propor um <i>framework</i> teórico que caracteriza a adaptação mútua entre regras e rotinas formais e suas <i>performances</i> reais.	Regras e procedimentos formais e <i>performances</i> reais, que interagem em ciclos de enquadramento, transbordamento e reenquadramento.
Weichbrodt e Grote (2010)	Compreender a relação entre rotinas e regras.	Rotinas na prática (comportamento), em princípio (cognição, regras não escritas) e regras (artefatos).
Milagres (2011)	Discutir o conceito de rotinas e apresentar os principais avanços da área.	Padrões comportamentais, regras e disposições coletivas.

Fonte: autor.

Para compreender as rotinas organizacionais, é relevante analisar o papel dos diferentes *stakeholders* envolvidos. Weichbrodt e Grote (2010) abordam esse fenômeno e apontam três papéis: os “fazedores”, os “seguidores” e os “supervisores” de regras. Os fazedores de regras são responsáveis pela definição das regras formais a serem seguidas pelos seguidores de regras. Cabe aos supervisores de regras monitorar o comportamento dos seguidores, analisando se está de acordo com o esperado. A Figura 10 apresenta o esquema proposto pelos autores.

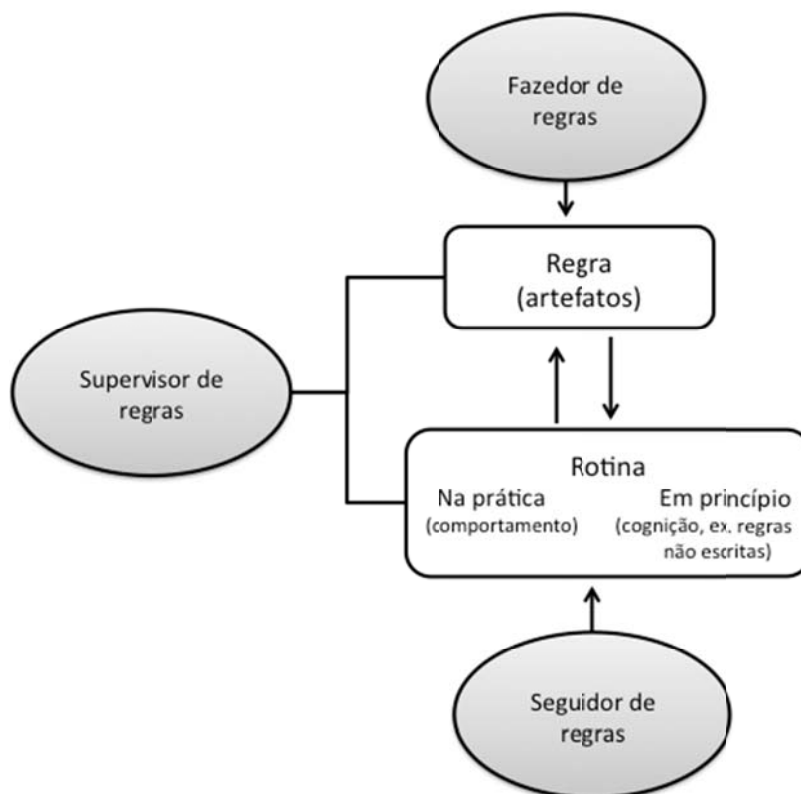


Figura 10 (2) – Regras, rotinas e os *stakeholders* relevantes nas organizações.
Fonte: Weichbrodt e Grote (2010, tradução nossa).

Conforme pode ser observado, Weichbrodt e Grote (2010) diferenciam regras de rotinas, sendo as regras artefatos concretos, definidos pelos fazedores, e as rotinas o comportamento dos seguidores, que é supervisionado. Para D’Adderio (2008), os artefatos desempenham um papel fundamental, agindo como intermediários no processo de mudança das rotinas organizacionais. A autora aponta para a existência de dois extremos, artefatos como uma visão totalmente descritiva (representação passiva do processo real) ou totalmente prescritiva (dando ordem e estruturando o processo). De uma forma ou de outra, eles correspondem à execução das atividades (D’ADDERIO, 2008).

2.2.1 Mudanças nas rotinas organizacionais

Segundo Gersick e Hackman (1990), as rotinas organizacionais são desempenhadas de forma inconsciente, de modo que há uma tendência a que os padrões de ação se repitam diante de determinado estímulo. Os autores apontam quatro fatores que contribuem para isso: o impacto social, o arrastamento social, o custo de mudança e as normas dos grupos.

O impacto social consiste na tendência a que uma pessoa aguarde por outros indivíduos para realizar determinada mudança quanto ele faz parte de um grupo (GERSICK;