



Pós-Graduação em Ciência da Computação

Lucas Bacciotti Moreira

**BOAS PRÁTICAS PARA GESTÃO DE TEMPO DE
PROJETOS DE SOFTWARE EM INSTITUTOS FEDERAIS DE
EDUCAÇÃO**



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

Recife
2020

Lucas Bacciotti Moreira

**BOAS PRÁTICAS PARA GESTÃO DE TEMPO DE PROJETOS DE SOFTWARE
EM INSTITUTOS FEDERAIS DE EDUCAÇÃO**

Este trabalho foi apresentado à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestre Profissional em Ciência da Computação.

Área de Concentração: Engenharia de Software e Linguagens de Programação.

Orientador(a): Prof. Alexandre Marcos Lins de Vasconcelos

Recife
2020

Catálogo na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

M838b Moreira, Lucas Bacciotti
Boas práticas para gestão de tempo de projetos de software em institutos federais de educação / Lucas Bacciotti Moreira. – 2020.
133 f.: il., fig., tab.

Orientador: Alexandre Marcos Lins de Vasconcelos.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2020.
Inclui referências e apêndices.

1. Engenharia de software. 2. Projetos de software. I. Vasconcelos, Alexandre Marcos Lins de (orientador). II. Título.

005.1

CDD (23. ed.)

UFPE - CCEN 2020 - 159

Lucas Bacciotti Moreira

**Boas Práticas para gestão de tempo de projetos de software em
Institutos Federais de Educação**

Dissertação apresentada ao Programa de Pós-Graduação Profissional em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre Profissional em 06 de agosto de 2020.

Aprovado em: 06/08/2020.

Orientador: Prof. Alexandre Marcos Lins de Vasconcelos

BANCA EXAMINADORA

Prof. Hermano Perrelli de Moura
Centro de Informática / UFPE

Prof. Célio Andrade de Santana Júnior
Centro de Artes e Comunicação/UFPE

Prof. Rafael Prikladnicki
Pontifícia Universidade Católica do Rio Grande do Sul

Dedico este trabalho à minha família, pelo apoio, compreensão, paciência e motivação inquestionáveis. Dedico também ao Supremo Arquiteto do Universo, que me guiou em cada passo desta jornada.

AGRADECIMENTOS

Agradeço à minha esposa, Andriely Lima Cardoso, pelo apoio ininterrupto;

Agradeço aos meus pais, Maria Teresa Bacciotti Moreira e José Moreira Filho, pela possibilidade do estudo;

Agradeço aos meus irmãos, Tiago Bacciotti Moreira e Tadeu Bacciotti Moreira, pela motivação;

Agradeço ao meu orientador, Prof. Dr. Alexandre Marcos Lins de Vasconcelos, pelos ensinamentos;

Agradeço aos meus colegas de trabalho do IFTM e da turma de Mestrado Profissional em Ciências da Computação de 2018, pela ajuda indiscutível.

RESUMO

O presente trabalho tem por objetivo analisar se os Fatores Críticos de Sucesso e Fatores Críticos de Fracasso (fatores que podem interferir positiva ou negativamente no ciclo de vida de um projeto de software) e as técnicas (as sugestões encontradas em literatura especializada para melhor aplicar os Fatores Críticos de Sucesso ou mitigar os Fatores Críticos de Fracasso) ocorrem nos Institutos Federais de Educação do Brasil (IFs). Com as informações coletadas em Revisão de Literatura e Pesquisa de Campo, somadas a observações do pesquisador, um conjunto de Boas Práticas foi elaborado e sugerido com o objetivo de auxiliar as equipes de desenvolvimento de software dessas instituições a conduzirem melhor seus projetos e conseguirem entregar no prazo previsto originalmente. Para tanto, as informações coletadas foram cruzadas para que se obtivesse uma relação de técnicas mais relevantes, associadas a Fatores Críticos, para serem aplicadas em tais instituições. Por fim, um Grupo Focal foi aplicado para coletar as impressões de um grupo de pessoas acerca do produto desenvolvido.

Palavras-chave: Projetos de software. Fatores críticos. Gestão de tempo.

ABSTRACT

This work aims to analyze whether Critical Success Factors and Critical Failure Factors (factors that can positively or negatively interfere in the life cycle of a software project) and techniques (suggestions found in specialized literature to better apply the Critical Success Factors or mitigate Critical Failure Factors) occur at the Federal Institutes of Education of Brazil (IFs). With the information collected in Literature Review and Field Research, added to the researcher's observations, a set of Good Practices was developed and suggested in order to help the software development teams of these institutions to better conduct their projects and to be able to deliver to the originally scheduled. For this purpose, the information collected was cross-checked to obtain a list of the most relevant techniques, associated with Critical Factors, to be applied in such institutions. Finally, a Focus Group was applied to collect the impressions of a group of people about the product developed.

Keywords: Software project. Critical factors. Time management.

LISTA DE FIGURAS

Figura 1 -	Rede Federal de Ensino Técnico e Tecnológico (MEC, 2019)	14
Figura 2 -	Relação entre os FCS e os FCF	18
Figura 3 -	Etapas da Metodologia de pesquisa	25
Figura 4 -	Porcentagens de projetos de Sucesso ou Falhos quanto ao Orçamento, Tempo e Requisitos (CHAOS Report, 2015).....	31
Figura 5 -	Fluxo de Processo da QUASI-RSL	33
Figura 6 -	Passo a passo do processo de seleção (Revisão Sistemática da Literatura).....	35
Figura 7 -	Questão 1 do teste-piloto (clareza)	56
Figura 8 -	Questão 2 do teste-piloto (estrutura).....	56
Figura 9 -	Questão 3 do teste-piloto (tempo de resposta)	57
Figura 10 -	Pergunta sobre forma de desenvolvimento.....	58
Figura 11 -	Pergunta sobre a existência de um processo definido de desenvolvimento	59
Figura 12 -	Pergunta sobre entregas no prazo.....	60
Figura 13 -	Pergunta sobre satisfação do cliente	61
Figura 14 -	Pergunta sobre os FC	62
Figura 15 -	Pergunta sobre as técnicas utilizadas.....	64
Figura 16 -	Exemplo de descrição de FC	81
Figura 17 -	Exemplo de descrição de boa prática	82
Figura 18 -	Boas Práticas e algumas subseções	83
Figura 19 -	Passo a passo da escolha dos estudos utilizando a notação BPMN.....	123
Figura 20 -	Questões do teste-piloto	128
Figura 21 -	Fases do Grupo Focal.....	132

LISTA DE TABELAS

Tabela 1 -	Quadro metodológico.....	26
Tabela 2 -	Quantitativo de estudos incluídos e excluídos em cada passo da seleção.....	36
Tabela 3 -	Totais de estudos relacionados ao tema abordado.....	36
Tabela 4 -	Identificadores dos artigos encontrados na QUASI-RSL	38
Tabela 5 -	FC Encontrados versus Artigos	39
Tabela 6 -	Relação entre FC e técnicas/ferramentas.....	45
Tabela 7 -	Relação entre os FC e as Técnicas citadas no questionário	64
Tabela 8 -	Relação entre os FC selecionados no Passo 1, percentual de citações na pesquisa de campo, número de citações na QUASI-RSL e técnicas propostas	72
Tabela 9 -	Relação entre Boas Práticas e FC	78
Tabela 10 -	Validações e ajustes feitos no conjunto após Grupo Focal.....	105
Tabela 11 -	Linhas-guia, questão principal de pesquisa e palavras-chave.	118
Tabela 12 -	Palavras-chave, suas traduções e sinônimos em inglês.....	119
Tabela 13 -	Recursos utilizados nas buscas automáticas.....	119
Tabela 14 -	Resumo da calibragem da <i>string</i> de busca	120

LISTA DE ABREVIATURAS/ACRÔNIMOS

APF -	Análise de Pontos de Função
BPM -	<i>Business Process Management</i> (Modelo de Processo de Negócio)
CCR -	<i>Customer satisfaction, Cost and Regression Count</i> (Satisfação do usuário, Custo e Contagem de Regressão)
COCOMO -	<i>Constructive Cost Model</i> (Modelo de Custo Construtivo)
DTIC -	Diretoria de Tecnologia da Informação e Comunicação
ERP-	<i>Enterprise Resource Planning</i> (Planejamento de Recursos Empresariais)
ESBE -	Engenharia de Software Baseada em Evidências
FAST -	<i>Facilitated Application Specification Technique</i> (Técnica de Especificação de Aplicação Facilitada)
FC -	Fatores Críticos
FCF -	Fatores Críticos de Fracasso
FCS -	Fatores Críticos de Sucesso
FPA -	<i>Function Point Analysis</i> (Análise de Pontos de Função)
GF -	Grupo Focal
GQM -	<i>Goal Question Metric</i> (Métrica Meta-Questão)
IFs -	Institutos Federais
IFTM -	Instituto Federal de Educação, Ciência e Tecnologia do Triângulo Mineiro
KLOC -	<i>Kilo Lines Of Code</i> (Mil Linhas de Código)
LOC -	<i>Lines Of Code</i> (Linhas de Código)
MBE -	<i>Management By Exceptions</i> (Gerenciamento Por Exceções)
MDS-	GRP - Metodologia de Desenvolvimento de Software - <i>Government Resource Planning</i> (Planejamento de Recursos Governamentais)
MEC -	Ministério da Educação
PSO -	<i>Particle Swarm Optimization</i> (Otimização por Enxame de Partículas)
RACI -	<i>Responsible, Accountable, Consulted, Informed</i> (Responsável, Aprovador, Consultado, Informado)

- RSL - Revisão Sistemática da Literatura
- SPM - *Software Project Management* (Gestão de Projetos de Software)
- TIC - Tecnologia da Informação e Comunicação
- XP - *Extreme Programming* (Programação Extrema)

SUMÁRIO

1	INTRODUÇÃO.....	14
1.1	Detalhamento do problema.....	18
1.2	Definições	19
1.3	Relevância do Trabalho	21
1.4	Questões de pesquisa	22
1.5	Objetivo geral.....	22
1.6	Objetivos específicos	22
1.7	Contribuições e resultados esperados	23
1.8	Estrutura do trabalho.....	23
2	METODOLOGIA DE PESQUISA.....	25
2.1	Quadro Metodológico.....	26
2.2	Questionário aplicado	27
2.3	Avaliação da aplicação do conjunto de Boas Práticas	28
2.4	Considerações finais do capítulo	28
3	FATORES CRÍTICOS RELACIONADOS AO CUMPRIMENTO DE PRAZOS DE PROJETOS DE SOFTWARE	30
3.1	Introdução	30
3.2	A revisão <i>ad hoc</i> da literatura	30
3.2.1	<i>CHAOS Report</i>	30
3.2.2	<i>Outros trabalhos relevantes encontrados de forma ad hoc</i>	32
3.3	A Revisão Sistemática da Literatura	33
3.4	Os Fatores Críticos Identificados	36
3.4.1	<i>Detalhamento dos FC identificados</i>	37
3.5	Considerações finais do capítulo	51
4	O ESTADO ATUAL DOS FATORES CRÍTICOS PARA ENTREGAS NO PRAZO DOS PROJETOS DOS IFs	53
4.1	Introdução	53
4.2	O estado atual dos FC nos projetos de desenvolvimento de software dos IFs.....	53
4.2.1	<i>Perfil do respondente</i>	55
4.3	Resultados e análise.....	57
4.3.1	<i>Resultado da aplicação do questionário nos IFs</i>	57
4.4	Considerações finais do capítulo	69
5	UM CONJUNTO DE BOAS PRÁTICAS PARA EQUIPES DE DESENVOLVIMENTO DE SOFTWARE DOS INSTITUTOS FEDERAIS DE EDUCAÇÃO	71

5.1	Introdução	71
5.2	A seleção dos FC e técnicas relacionadas de maior impacto à gestão de tempo dos IFs.....	72
5.3	As técnicas relacionadas aos FC selecionados	73
5.4	A estrutura do conjunto de Boas Práticas	80
5.5	Considerações finais do capítulo	83
6	AVALIAÇÃO DO CONJUNTO DE BOAS PRÁTICAS.....	85
6.1	Introdução	85
6.2	Grupo Focal	85
6.2.1	<i>O que é um Grupo Focal.....</i>	<i>85</i>
6.2.2	<i>Vantagens do Grupo Focal</i>	<i>86</i>
6.2.3	<i>Desvantagens do Grupo Focal</i>	<i>87</i>
6.2.4	<i>Justificativa para a utilização do Grupo Focal.....</i>	<i>88</i>
6.2.5	<i>Abordagem utilizada</i>	<i>90</i>
6.2.6	<i>Utilização do Grupo Focal online</i>	<i>92</i>
6.3	Resultados obtidos.....	92
6.3.1	<i>Avaliação dos FC.....</i>	<i>92</i>
6.3.2	<i>Avaliação das Boas Práticas.....</i>	<i>97</i>
6.4	Considerações finais do capítulo	101
7	CONCLUSÃO	107
	REFERÊNCIAS	110
	APÊNDICE A - PROTOCOLO DA QUASI-REVISÃO SISTEMÁTICA.....	116
	APÊNDICE B - QUESTIONÁRIO APLICADO	125
	APÊNDICE C – PROTOCOLO DO GRUPO FOCAL	129

1 INTRODUÇÃO

Os Institutos Federais (IFs) surgiram¹ em 2008 com o objetivo de ofertar educação profissional, tecnológica, pública, gratuita e de qualidade nas diferentes áreas de ensino com base na conjugação de conhecimentos técnicos e tecnológicos com as suas práticas pedagógicas, nos termos da Lei 11.892 de 29 de dezembro de 2008 (BRASIL, 2008). Estão distribuídos por todo o território nacional, conforme indica a Figura 1 (em verde estão os IFs, foco deste trabalho).

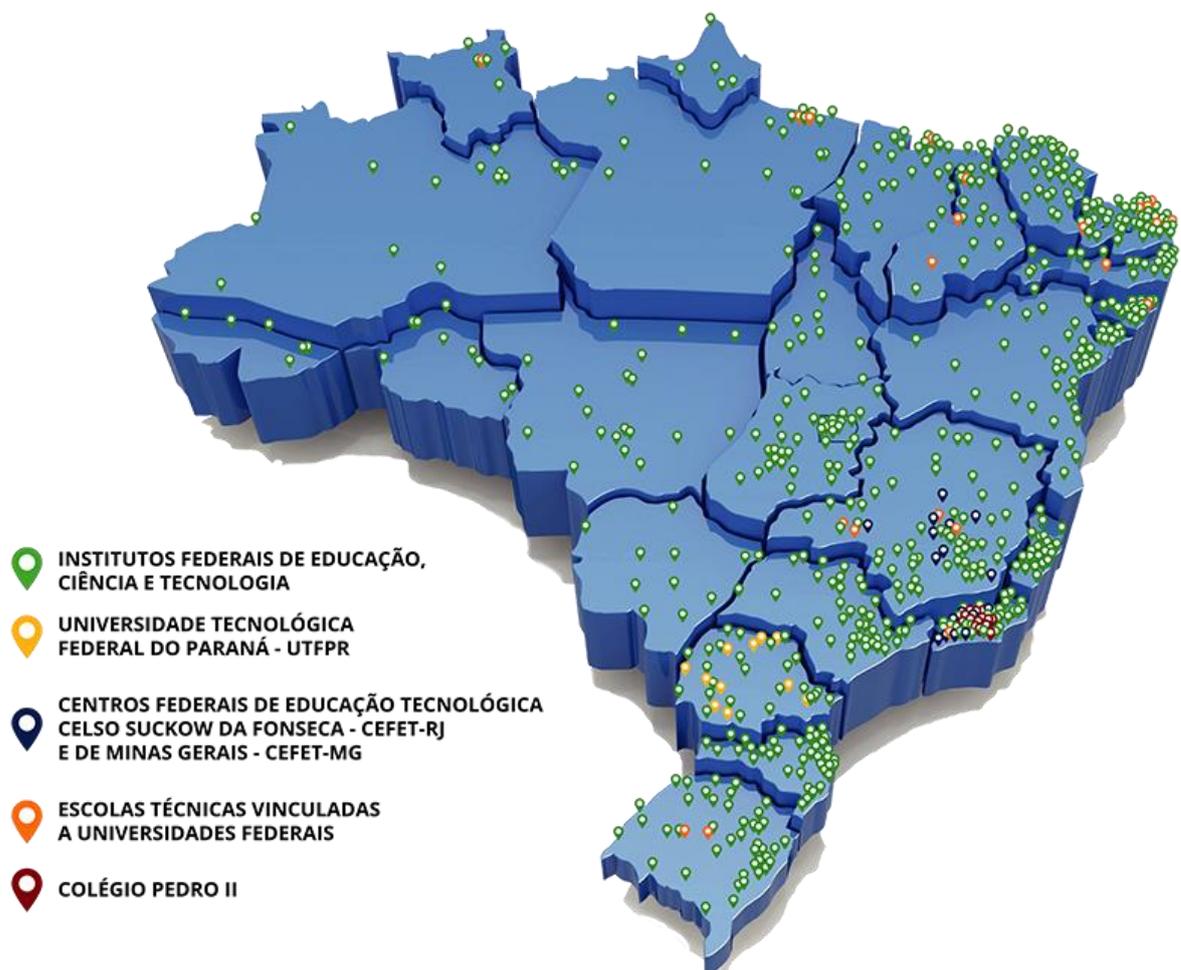


Figura 1 - Rede Federal de Ensino Técnico e Tecnológico (MEC, 2019)

¹ Em alguns casos, os antigos ETFs (Escola Técnica Federal) e CEFETs (Centro Federal de Educação Tecnológica) foram transformados em IFs.

Por possuírem autonomia administrativa e financeira (BRASIL, 2008), ou seja, serem responsáveis pela gestão de seu pessoal, alocação de recursos, estabelecimento e cumprimento de metas, entre outras atividades correlatas, cada IF opta por gerenciar à sua maneira seus recursos de Tecnologia da Informação e Comunicação (TIC), através de Diretorias de TIC ou Coordenações de TIC.

No âmbito do IFTM - Instituto Federal de Educação, Ciência e Tecnologia do Triângulo Mineiro (instituição onde o autor desta dissertação está lotado), o planejamento, coordenação, supervisão e orientação quanto à execução das atividades relacionadas à área de tecnologia da informação e comunicação é de competência da Diretoria de Tecnologia da Informação e Comunicação (DTIC), unidade administrativa subordinada à Pró-Reitoria de Desenvolvimento Institucional (IFTM, 2009).

A DTIC conta com um quadro de aproximadamente 20 servidores, divididos em equipes como Suporte ao usuário, Suporte aos Sistemas, Qualidade de Software, Governança de TI e Construção de Sistemas. Esta última é responsável pelas tarefas de levantamento de requisitos, codificação, testes, implantação e entrega do produto de software. Os projetos de software, em geral, passam por todas as equipes durante seu ciclo de vida.

Os projetos da DTIC têm como objetivo final entregar algum produto de software² ao cliente³. Espera-se que estes projetos sejam finalizados no prazo previsto, utilizando os recursos estipulados, estejam contemplando o escopo e que tenham qualidade, para que os objetivos sejam alcançados e o nível de satisfação do cliente aumente.

É importante destacar o papel do cliente neste contexto. Em instituições públicas, um dos principais objetivos é atender com presteza ao público em geral. Desta forma, entende-se a satisfação do cliente como balizadora dos objetivos dos projetos institucionais, inclusive os projetos de software da DTIC, ou seja, os projetos de desenvolvimento de softwares em Institutos Federais de Educação devem ser entregues no prazo estipulado, respeitando o orçamento e o escopo planejados.

² A DTIC também mantém projetos de aquisição, compra, implantação e manutenção de hardware. O foco deste trabalho, porém, é nos projetos somente de software.

³ Neste contexto, cliente é a pessoa ou setor que utiliza o software desenvolvido pela DTIC. Pode ser da própria instituição ou do público externo.

Neste contexto, um relatório do Standish Group (2015) traz indícios de que 60% dos projetos de software não são entregues dentro do prazo, caracterizando um atraso ou uma má elaboração do cronograma. Além disso, o mesmo relatório cita que apenas 36% dos projetos de software obtiveram sucesso em entregar dentro do orçamento, do prazo e do escopo previstos. Em consequência, os objetivos institucionais não são atingidos e a insatisfação do cliente e o desperdício de recursos podem ocorrer.

Diante destes dados e de dados levantados por Jones (2004), Nasir e Sahibuddin (2011) e Nguyen (2014) surgiu a necessidade de melhorar o processo de desenvolvimento de projetos de software dos IFs, com o objetivo de entregar os produtos de software no prazo estipulado, com qualidade e utilizando os recursos previstos.

Para ilustrar o tipo de problemas a serem tratados no contexto desta dissertação, cita-se dois exemplos:

- **Caso 1:** o Banco de Informações de Estágios⁴ foi solicitado no início do ano letivo. Desta forma, mobilizou-se uma equipe ágil para o seu desenvolvimento. Embora a metodologia ágil tivesse acabado de ser implantada no setor, os procedimentos e artefatos estavam sendo produzidos corretamente e havia a expectativa de cumprimento dos prazos. As *sprints*⁵ estavam sendo cumpridas e os requisitos satisfeitos. Faltando uma semana para o lançamento oficial do projeto pela Instituição, descobriu-se que uma determinada tela não havia sido desenvolvida. Após análise constatou-se que esta funcionalidade não tinha sido desenvolvida porque não havia sido incluída no rol dos requisitos. A equipe, o patrocinador e o gerente do projeto não se atentaram a esta funcionalidade, resultando em um levantamento de requisitos mal elaborado e, conseqüentemente, uma má gestão do tempo, visto que o cronograma teve que ser alterado e o produto foi lançado incompleto pela Instituição, inicialmente.
- **Caso 2:** por conta do Decreto 8.539 de 8 de outubro de 2015, os Institutos Federais de Educação começaram a se mobilizar para que seus processos administrativos fossem transferidos para o meio virtual, dispensando a utilização

⁴ O nome original do projeto foi alterado e o nome da Instituição Federal de Ensino foi suprimido, pois tais informações não têm relevância ao que é discutido na seção.

⁵ Espaço de tempo usado em metodologias de desenvolvimento ágil para se entregar uma parte utilizável e funcional de um produto de software.

de papel. Algumas Instituições utilizaram sua equipe de TIC para desenvolver seu próprio sistema. Neste cenário, um episódio chama atenção: determinada instituição que planejou o desenvolvimento do software para atender as exigências legais no prazo solicitado, se mobilizou através de várias reuniões. O sistema projetado era muito grande e desde o início havia a percepção de que o prazo seria bastante curto para entregar todas as funcionalidades exigidas. Desta forma, ao final do prazo o sistema foi lançado, porém com funcionalidades incompletas e alguns erros na interface do usuário. Cabe ressaltar que, durante o processo de desenvolvimento, novas exigências legais surgiram e novas funcionalidades foram inseridas no escopo do projeto. Em razão das várias dificuldades encontradas, o projeto não foi entregue por completo no prazo estipulado.

A partir da análise dos casos relatados pode-se destacar alguns pontos em comum: tanto o caso 1 quanto o caso 2 ilustraram problemas na entrega do produto final, no entanto, por causas diferentes. No caso 1, os requisitos foram mal levantados e analisados; e no caso 2, um cronograma apertado pressionou as equipes a planejarem e desenvolverem um sistema complexo em um tempo recorde. Pode-se especular que, para contornar tais situações, poderia-se, por exemplo, fazer uso de alguma técnica de engenharia de requisitos combinada com a utilização de alguma metodologia de Gestão de Projetos de Software adaptada ao contexto.

Na literatura existem estudos que buscam os motivos para o sucesso ou fracasso dos projetos de software. Estes motivos são chamados de Fatores Críticos (FC), que são tanto de sucesso (Fatores Críticos de Sucesso - FCS) como de fracasso (Fatores Críticos de Fracasso - FCF). Os FCS são os fatores que influenciam de forma benéfica o processo de desenvolvimento do projeto direta ou indiretamente. Quando não aplicados, podem causar falhas no ciclo de vida do projeto de software e conseqüentemente prejuízo aos clientes e desperdício de recursos, se transformando em um FCF. Assim, conceituam-se os Fatores Críticos de Fracasso como a falta ou a não aplicação correta dos Fatores Críticos de Sucesso, caracterizando-se uma relação antagônica entre eles. A Figura 2 ilustra o conceito geral de Fator Crítico, o qual se desmembra em dois outros conceitos, Fatores Críticos de Sucesso e Fatores Críticos de Fracasso.

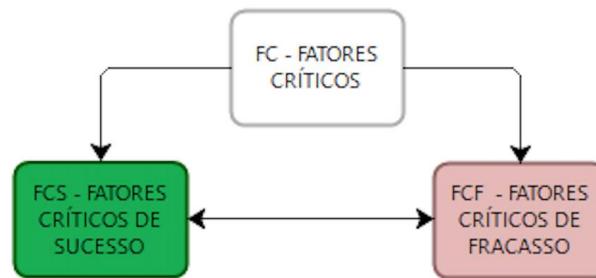


Figura 2 - Relação entre os FCS e os FCF

Mediante o exposto, este trabalho tem como objetivos: levantar, com base na literatura, os Fatores Críticos que influenciam a entrega de projetos de software no prazo previsto; identificar a ocorrência dos FC nos IFs do Brasil (situação atual); identificar se nos IFs do Brasil são aplicadas técnicas para melhor gestão de tempo e entrega no prazo; e apresentar um conjunto de boas práticas para os IFs a fim de agilizar a entrega de projetos de software, aumentar a qualidade dos produtos de software, melhorar o nível de satisfação do cliente e cumprir os objetivos institucionais.

1.1 Detalhamento do problema

Segundo o Standish Group (2015), 19% dos projetos de desenvolvimento de software fracassam, apenas 36% têm sucesso e 45% são finalizados com ressalvas. Por “sucesso”, entende-se que o projeto foi finalizado no prazo previsto, com o orçamento suficiente e dentro do escopo planejado.

No contexto dos IFs observa-se a necessidade de finalizar os projetos de software com sucesso, principalmente no tocante ao prazo previsto, visto que a necessidade do cliente a cada dia demanda por soluções prontas em menor tempo.

Há a necessidade de verificar se os FC retratados ocorrem nos IFs para que propostas de melhoria de projetos de desenvolvimento de software sejam elaboradas no futuro, pois os projetos de software devem ser entregues a tempo para contribuir com os objetivos institucionais. Além disso, em consequência, é interessante constatar se tais instituições fazem uso de alguma técnica para melhoria de seus projetos de software e gestão de tempo. Desta forma, chegamos ao principal problema deste trabalho:

Problema de pesquisa: segundo o Standish Group (2015), muitos projetos de software da indústria sofrem atraso em suas entregas. Similarmente, alguns dos projetos de desenvolvimento de software dos Institutos Federais (IFs) sofrem atraso de cronograma (segundo pesquisa de campo realizada), prejudicando os objetivos do setor de TIC, os objetivos institucionais e a satisfação dos clientes. No entanto, não se tem conhecimento de quais práticas podem ser aplicadas nos IFs a fim de contribuir para a gestão de tempo desses projetos. Desta forma, tem-se como problemas principais desta pesquisa: **a)** deve-se entender quais fatores críticos podem influenciar as entregas no prazo previsto de projetos de desenvolvimento de software nos IFs do Brasil, e **b)** encontrar maneiras para auxiliar as equipes de TIC dos IFs a entregarem o software no prazo previsto.

1.2 Definições

É importante que algumas definições quanto ao tema pesquisado sejam apresentadas para um melhor entendimento do contexto em que a pesquisa está incluída.

Prazo do projeto: conhecido neste trabalho como “prazo final”, todo projeto de software deve possuir um prazo, ou seja, uma data definida para que alguma parte do produto (ou todo o produto) seja entregue. Em Colares (2010), vemos referência ao prazo também como *deadline*, palavra em inglês que tem o mesmo sentido (data de término de um projeto). Na literatura especializada, apenas para citar um exemplo, vemos a entrega no prazo previsto como um dos objetivos a serem cumpridos.

Entregas fora do prazo previsto podem desencadear um efeito cascata em todo o ciclo de vida do projeto e prejudicar a satisfação do cliente.

Cronograma de projeto: de acordo com Pressman (2005, apud Colares, 2010), “cronograma é uma atividade que distribui o esforço estimado em toda a duração planejada do projeto alocando esforços a tarefas de engenharia de software específicas”. Entradas básicas para construção do cronograma, segundo Colares (2010), são uma lista de atividades, recursos necessários e outras informações gerenciais sobre o projeto. Quando uma tarefa não é executada no prazo estipulado

originalmente, entende-se que esta tarefa não foi executada dentro do cronograma previsto. Neste caso, tem-se um fracasso no atendimento ao cronograma e um atraso no projeto. Ayyubi, Ahmad e Faiz (2007) salientam que um terço dos projetos de software sofrem algum tipo de atraso no cronograma ou sobrecarga de orçamento e Shahzad e Mathkour (2009) destacam que um planejamento de projeto de software só pode lograr sucesso se todas as atividades agendadas são executadas de acordo com os prazos pré-estabelecidos (ou cronograma).

Gestão do Projeto de software: conforme Pradhan, Rishiwal e Agarwal (2016), Gestão de Projeto de Software se refere à aplicação de conhecimentos, métodos, técnicas e ferramentas para resolver problemas de planejamento de projetos. Pelo mesmo caminho seguem Suliman e Kadoda (2017), pois citam que “a gestão de projeto de software é a aplicação de conhecimento, habilidades, ferramentas e técnicas em diferentes atividades realizadas a fim de satisfazer os requisitos do projeto”.

Fatores Críticos (FC): na literatura, os FC são divididos tanto em FCS - Fatores Críticos de Sucesso, que são aqueles que trazem benefícios ao projeto como em FCF - Fatores Críticos de Fracasso, que são os fatores que prejudicam o projeto de software. Jones (2004) cita em seu trabalho a relação contrastante entre eles. O autor apresenta uma relação de oposição entre os seis maiores fatores de sucesso e de fracasso para projetos de software. Para citar alguns exemplos, “Controle Efetivo de Qualidade do Projeto”, “Planejamento Efetivo do Projeto” e “Estimativa de Custo Efetiva” são fatores citados na seção de “Sucesso”, enquanto “Controle de Qualidade do Projeto Inadequado”, “Planejamento inadequado” e “Estimativa de custo inadequada”, respectivamente, estão na seção de “Fracasso”. Adicionalmente, Nasir e Sahibuddin (2011) deixam claro em seu trabalho que muitos autores publicam textos tanto de fatores de sucesso como de fracasso, muitas das vezes expondo-os em formato de sumário. Portanto, entende-se que existe um conceito geral (FC) que abrange dois conceitos específicos e contrários (FCS e FCF). Os três (FC, FCS e FCF) conceitos serão utilizados neste trabalho.

Fatores Críticos de Sucesso (FCS): para Zago (2016), são fatores que quando bem administrados podem contribuir para o sucesso de um projeto. Nasir e

Sahibuddin (2011) citam que são fatores que, quando alocados apropriadamente, vão aumentar significativamente as chances de sucesso do projeto. Na literatura, os FCS são frequentemente listados e categorizados. Para citar alguns exemplos, tem-se **“Gestão de Projetos de Software bem definida”** [Bilgaiyan, Mishra e Das (2016), Suliman e Kadoda (2017), Govindarajan (2015), Jing e Qiu (2007), Ayyubi, Ahmad e Faiz (2007), Octavianus e Mursanto (2018), Hong, Yoo e Cha (2010) e Ahimbisibwe e Cavana (2016)]; **“Gerenciamento de Riscos efetivo”** [Kutsch e Hall (2005), Abdul-Rahman, Mohd-Rahim e Chen (2011), Jamshidi et al. (2014), Ayyubi, Ahmad e Faiz (2007), Melo et al. (2013)]; e **“Utilização de técnicas de Engenharia de Requisitos”** [Gupta, Chauhan e Dutta (2013), Jones (2004), Hussain, Mkpojiogu e Kamal (2016), Cerpa e Verner (2009), Govindarajan (2015), Hashim, Abbas e Hashim (2013), Shahzad e Mathkour (2009), Raunak e Binkley (2017), Nazir et al. (2014), Nasir e Sahibuddin (2011)].

Fatores Críticos de Fracasso (FCF): por outro lado, tem-se os Fatores Críticos de Fracasso, que são o oposto dos FCS. Enquanto os FCS trazem benefícios aos projetos de software, os FCF tendem a trazer prejuízo. Como exemplo, tem-se “Pouco ou nenhum apoio da alta gestão”, “Pouca comunicação entre a equipe”, “Não utilização de gerenciamento de riscos”, entre outros. Os FCF surgem quando da não utilização ou má aplicação dos FCS, ou seja, são conceitos contrários um ao outro.

1.3 Relevância do Trabalho

Há a necessidade de todos os setores e órgãos prestarem seus serviços com qualidade e em consonância com a estratégia institucional. Desta maneira, uma boa prestação de serviços por parte do setor de TIC fornece apoio ao negócio e às atividades principais da instituição, além de refletir diretamente no cumprimento dos objetivos estratégicos, missão e visão, e cumprir com os valores da instituição, “Qualidade e Excelência: promover a melhoria contínua dos serviços oferecidos para a satisfação da sociedade” (IFTM, 2019), por exemplo.

Uma outra justificativa se dá pela busca da qualidade do serviço público, pois é necessário agir em conformidade com a Constituição Federal do Brasil (1988) e

atender ao público com presteza e qualidade. A melhoria dos produtos das instituições (e os produtos de software estão incluídos) vai ao encontro desses princípios.

Portanto, este trabalho faz-se relevante pois é necessária a entrega no prazo previsto dos projetos de desenvolvimento de software em Institutos Federais de Educação, a fim de aumentar a satisfação do cliente, utilizar melhor os recursos disponíveis, apoiar o negócio principal e atingir os objetivos institucionais.

1.4 Questões de pesquisa

As seguintes questões nortearão o processo de pesquisa deste trabalho:

Q1: quais FC encontrados na literatura impactam na entrega no prazo previsto de projetos de software?

Q2: quais os FC que ocorrem nos IFs?

Q3: quais são as boas práticas que os envolvidos em projetos de desenvolvimento de software de Institutos Federais de Educação devem seguir a fim de cumprir os prazos previstos, aumentar a satisfação do cliente e atingir os objetivos institucionais?

1.5 Objetivo geral

O objetivo geral deste trabalho é propor um conjunto de boas práticas de desenvolvimento de software em Institutos Federais de Educação baseado na literatura, a fim de contribuir para a gestão do tempo dos projetos de desenvolvimento de software em tais instituições.

1.6 Objetivos específicos

Para alcançar o Objetivo Geral, tem-se alguns objetivos específicos, a saber:

- Levantar na literatura os Fatores Críticos que afetam as entregas no prazo de projetos de desenvolvimento de software;
- Identificar os Fatores Críticos que afetam as entregas no prazo de projetos de desenvolvimento de software nos Institutos Federais do Brasil;
- Analisar a relação existente entre os Fatores Críticos encontrados na literatura e os encontrados nos Institutos Federais do Brasil;

- Propor práticas para entregar os projetos de desenvolvimento de software no prazo previsto através de um conjunto de Boas Práticas;
- Avaliar o conjunto de Boas Práticas através de Grupo Focal online com profissionais da Fábrica de Software do IFTM.

1.7 Contribuições e resultados esperados

As contribuições e resultados esperados para este trabalho estão descritos a seguir:

- A principal contribuição do trabalho é levantar quais são os FC para projetos de desenvolvimento de software com maior ocorrência nos IFs, pontuando-se as principais dificuldades encontradas ao tratar de gestão de tempo e entrega no prazo previsto, a fim de auxiliar pesquisadores em trabalhos futuros relacionados ao tema;
- Além disso, outro objetivo é a elaboração de um conjunto de Boas Práticas para Projetos de Software em Institutos Federais de Educação, que tem como finalidade auxiliar as equipes de TIC a entregarem os projetos no prazo previsto, agregando valor ao cliente, ou seja, entregando produtos de qualidade dentro do prazo e trazendo maior apoio às atividades institucionais.

1.8 Estrutura do trabalho

- **Capítulo 1 - Introdução:** insere o leitor no contexto do trabalho e apresenta uma justificativa para a pesquisa;
- **Capítulo 2 - Metodologia de Pesquisa:** apresenta a metodologia de pesquisa utilizada no trabalho;
- **Capítulo 3 - Fatores Críticos relacionados ao cumprimento de prazos de projetos de software:** apresenta a Revisão Sistemática da Literatura, a revisão *ad hoc* e o estado da arte relacionado ao tema pesquisado;
- **Capítulo 4 - O estado atual dos fatores críticos para entregas no prazo dos projetos dos IFs:** apresenta os resultados encontrados na pesquisa de campo através de questionário nos Institutos Federais do Brasil;

- **Capítulo 5 - Um conjunto de boas práticas para equipes de desenvolvimento de software dos Institutos Federais de Educação:** apresenta a proposta do autor para o enfrentar o problema apresentado;
- **Capítulo 6 - Avaliação do conjunto de Boas Práticas:** apresenta ao leitor a avaliação da proposta por parte da equipe de TIC de um dos IFs;
- **Capítulo 7 - Conclusão:** apresenta considerações finais do trabalho executado, dificuldades encontradas e propostas para futuros trabalhos futuros.

Além destes capítulos, o trabalho também conta com 3 apêndices:

- **APÊNDICE A - PROTOCOLO DA REVISÃO SISTEMÁTICA:** apresenta o passo a passo utilizado para executar a RSL - Revisão Sistemática da Literatura;
- **APÊNDICE B - QUESTIONÁRIO APLICADO:** apresenta o questionário aplicado nos IFs e o teste-piloto do questionário;
- **APÊNDICE C - PROTOCOLO DO GRUPO FOCAL:** apresenta o passo a passo utilizado para aplicar a técnica de Grupo Focal.

2 METODOLOGIA DE PESQUISA

A metodologia de pesquisa utilizada neste trabalho está descrita a seguir. Em resumo, utilizou-se os métodos: revisões bibliográficas (sistemática e *ad hoc*), pesquisa de campo com aplicação de questionário, interpretação dos dados coletados, elaboração de um conjunto de Boas Práticas e, por fim, a realização de um Grupo Focal para avaliação do conjunto elaborado. A Figura 3 ilustra as etapas da metodologia de pesquisa aplicada neste trabalho.

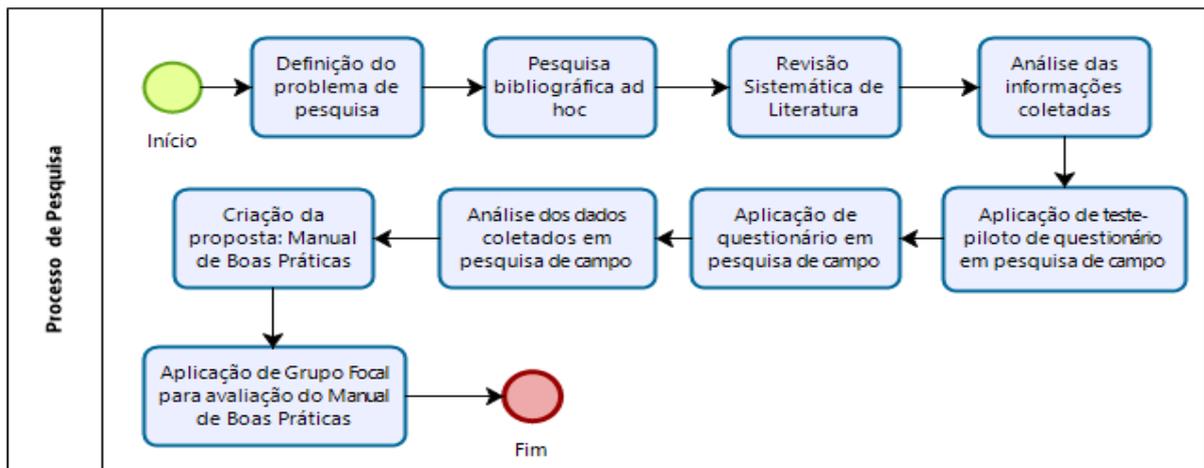


Figura 3 - Etapas da metodologia de pesquisa

A Revisão Sistemática da Literatura e a seleção de estudos de forma *ad hoc* (**descritas no capítulo 3**) foram realizadas para encontrar os principais estudos sobre o tema pesquisado.

Além disso, um questionário foi submetido aos Institutos Federais para adquirir informações acerca dos FC que influenciam os projetos de software de tais instituições, bem como quais técnicas são utilizadas atualmente (**descritos na seção 4.3.1**).

Por fim, elaborou-se um conjunto de Boas Práticas (**descrito no capítulo 5**) para auxiliar as equipes de desenvolvimento de software dos IF a entregarem os produtos no prazo previsto e aplicou-se a técnica de Grupo Focal para avaliação do conjunto, conforme o protocolo proposto no **APÊNDICE C**.

2.1 Quadro Metodológico

A Tabela 1 resume as características da metodologia de pesquisa.

Quanto à abordagem	Quantitativa
Quanto à natureza	Aplicada
Quanto aos procedimentos	Bibliográfica e de campo

Tabela 1 - Quadro metodológico

Conforme Cervo e Bervian (1978, apud Marconi e Lakatos, 2008), “pode-se afirmar que as premissas de um argumento indutivo correto sustentam a conclusão. Assim, quando as premissas são verdadeiras, o melhor que se pode dizer é que a sua conclusão é, provavelmente, verdadeira”. Desta forma, o trabalho em questão verifica se os FC ocorrem em um contexto específico (IFs como premissas específicas) e constata se os projetos de desenvolvimento de software em tais instituições sofrem atraso ou não (conclusão geral constatada pela revisão bibliográfica).

No tocante à abordagem, desenvolveu-se uma pesquisa qualitativa, pois buscou-se um “aprofundamento de compreensão de algumas organizações”, tentando “explicar o porquê das coisas, exprimindo o que convém ser feito” e “preocupando-se com aspectos da realidade que não podem ser quantificados” (Gerhardt e Silveira, 2009). Assim sendo, entende-se que representações numéricas complexas, como estatísticas não terão grande importância no trabalho.

A natureza da pesquisa é aplicada, pois “objetiva gerar conhecimentos para aplicação prática, dirigidos à solução de problemas específicos, envolvendo verdades e interesses locais” (Gerhardt e Silveira, 2009). Assim, buscou-se propor aos Institutos Federais de Educação boas práticas de desenvolvimento a fim de reduzir o impacto sobre a entrega no prazo previsto de projetos de desenvolvimento de software.

Para os procedimentos, foram utilizadas as técnicas de pesquisa bibliográfica e pesquisa de campo. A pesquisa bibliográfica foi executada tanto de forma *ad hoc* (**seção 3.2**), como através da técnica de Revisão Sistemática da Literatura (**seção 3.3**). Desta forma, foi possível encontrar estudos relacionados que possam contribuir com o tema abordado. A Revisão Sistemática da Literatura (QUASI-RSL), proposta

por Kitchenham (2007), é originária da Engenharia de Software Baseada em Evidências - ESBE:

A pesquisa baseada em evidências foi desenvolvida inicialmente na medicina quando estudos indicaram que a opinião médica baseada em pareceres de especialistas não era tão confiável quanto o aconselhamento baseado em evidências científicas. (...) A ESBE visa utilizar uma abordagem baseada em evidências aplicada à pesquisa e prática de Engenharia de Software. (...) O objetivo da ESBE é fornecer os meios pelos quais as melhores evidências atuais da pesquisa podem ser integradas à experiência prática e aos valores humanos no processo de tomada de decisão com relação ao desenvolvimento e manutenção de software. (KITCHENHAM et al. 2007).

Petersen et al. (2007) frisam que a revisão sistemática é mais aprofundada do que outras técnicas de revisão bibliográfica e a síntese dos resultados foca mais em identificar as melhores práticas e sua efetividade.

Além de Kitchenham (2007), os trabalhos de Alves (2015) e dos Santos (2014) também foram utilizados, pois ambos possuem temas correlatos à pesquisa proposta, tiveram seus estudos publicados próximo à data deste trabalho e ambos fizeram uso de técnicas da QUASI-RSL.

2.2 Questionário aplicado

Submeteu-se um questionário aos servidores das equipes de TIC dos 36 Institutos Federais de Educação do país com o objetivo de entender a situação atual dos FC dos projetos de desenvolvimento de software destas instituições. As questões do questionário podem ser lidas no **APÊNDICE B**. Para a aplicação do questionário, utilizou-se a técnica *survey*, assim conceituada na literatura:

É a pesquisa que busca informação diretamente com um grupo de interesse a respeito dos dados que se deseja obter. Trata-se de um procedimento útil, especialmente em pesquisas exploratórias e descritivas (Santos, 1999 apud Gerhardt e Silveira, 2009).

A pesquisa com *survey* pode ser referida como sendo a obtenção de dados ou informações sobre as características ou as opiniões de determinado grupo de pessoas,

indicado como representante de uma população-alvo, utilizando um questionário como instrumento de pesquisa (Fonseca, 2002, p. 33 apud Gerhardt e Silveira, 2009).

Nesse tipo de pesquisa, o respondente não é identificável, portanto, o sigilo é garantido. São exemplos desse tipo de estudo as pesquisas de opinião sobre determinado atributo, a realização de um mapeamento geológico ou botânico (Gerhardt e Silveira, 2009).

Um teste-piloto do questionário foi realizado com objetivo de constatar se as perguntas estavam adequadas para serem submetidas de forma oficial às instituições. Ao todo, seis servidores lotados no setor de TIC responderam ao teste-piloto. Ao final, constatou-se que o teste-piloto foi considerado adequado para aplicação.

2.3 Avaliação da aplicação do conjunto de Boas Práticas

Após levantamento das informações, desenvolveu-se um conjunto de Boas Práticas para ser utilizado por equipes de TIC de Institutos Federais de Educação com o objetivo de auxiliar nas entregas no prazo de projetos de software.

A abordagem utilizada foi uma apresentação do conjunto de Boas Práticas para um grupo de servidores da Fábrica de Software do IFTM, técnica conhecida como Grupo Focal (**seção 6.2**). O Grupo Focal (GF) permite avaliar a viabilidade de aplicação do conjunto de Boas Práticas, confirmar hipóteses, incentivar a troca de ideias e de experiências, dialogar sobre o tema e propor o conjunto a um grupo específico de pessoas. A técnica foi aplicada com base em conceitos propostos pela obra editada por Shull, Singer e Sjoberg (2008) e Zaganelli et al. (2015).

Segundo Ressel et al. (2008), o GF “permite que o processo de integração grupal se desenvolva, favorecendo trocas, descobertas e participações comprometidas. Também proporciona descontração para os participantes.”.

2.4 Considerações finais do capítulo

Este capítulo apresentou a metodologia de pesquisa utilizada neste trabalho. Em resumo, utilizou-se pesquisas bibliográficas (*ad hoc* e sistemática), pesquisa de

campo com aplicação de questionário, criação de um produto (conjunto de Boas Práticas) e aplicação de Grupo Focal para avaliação do produto. Utilizou-se, ainda, o método científico indutivo e a abordagem qualitativa para análise dos dados.

No próximo capítulo, os Fatores Críticos encontrados na literatura serão apresentados e descritos.

3 FATORES CRÍTICOS RELACIONADOS AO CUMPRIMENTO DE PRAZOS DE PROJETOS DE SOFTWARE

3.1 Introdução

Este capítulo tem como objetivo descrever os conceitos e estudos relacionados ao tema pesquisado. É importante destacar que a obra de Wazlawick (2009) serviu como fonte primária de orientações para o início da pesquisa. Segundo o autor, “a revisão bibliográfica não produz conhecimento novo, apenas supre deficiências de conhecimento do pesquisador em determinada área”.

Uma revisão *ad hoc* da literatura (**seção 3.2**) foi utilizada para encontrar estudos que auxiliassem o pesquisador na execução da pesquisa, na elaboração do conjunto de Boas Práticas e na condução da Revisão Sistemática da Literatura.

3.2 A revisão *ad hoc* da literatura

A revisão *ad hoc* se fez útil para encontrar estudos que auxiliassem o pesquisador na execução da pesquisa (por exemplo a obra de Wazlawick, 2009), na elaboração do conjunto de Boas Práticas (baseado em Gamma et al., 2007) e na condução da Revisão Sistemática da Literatura (Kitchenham, 2007). Outros estudos encontrados de forma *ad hoc* foram importantes para servirem como base (“artigos-mente”) para o processo de Revisão Sistemática da Literatura. É o caso dos estudos de Suliman e Kadoda (2017), Nguyen (2014), Pradhan, Rishiwal e Agarwal (2016), e Bilgaiyan, Mishra e Das (2016) - ver seção **3.2.2**.

Por fim, o CHAOS Report (*The Standish Group*, 2015) foi usado para trazer estatísticas sobre o estado dos projetos na indústria (ver seção **3.2.1**).

3.2.1 CHAOS Report

O CHAOS Report (*The Standish Group*, 2015) é um relatório elaborado periodicamente pelo *Standish Group* com o objetivo de aferir e divulgar a situação de projetos de desenvolvimento de software na indústria. Há também o CHAOS

Manifesto, documento acessório ao *CHAOS Report*⁶ o qual traz análises mais profundas sobre os dados coletados.

A importância do uso do *CHAOS Report* nesta pesquisa se dá pelo fato de que este relatório retrata de forma periódica a maturidade dos projetos de software, trazendo dados importantes da taxa de sucesso destes projetos na indústria, além de estatísticas de projetos entregues em tempo, dentro do orçamento, cumprindo requisitos, agregando valor e trazendo satisfação do usuário. Além disso, o relatório foi útil para constatar a ocorrência de entregas após o prazo em projetos da indústria.

O relatório divide os projetos em *Successful* (Sucesso), *Challenged* (Comprometidos) e *Failed* (Falhos)⁷. Os projetos de Sucesso têm como característica serem entregues no tempo previsto (*OnTime*), no orçamento previsto (*OnBudget*) e cumprindo os requisitos (*OnTarget*). Já os Comprometidos são os projetos que, embora concluídos, não cumpriram ou o tempo previsto, ou o orçamento ou os requisitos. Por fim, os Falhos são aqueles que foram cancelados ou não utilizados. A Figura 4 apresenta o percentual de projetos quanto ao cumprimento de prazos, custos e requisitos, de acordo com *CHAOS Report (The Standish Group, 2015)*.

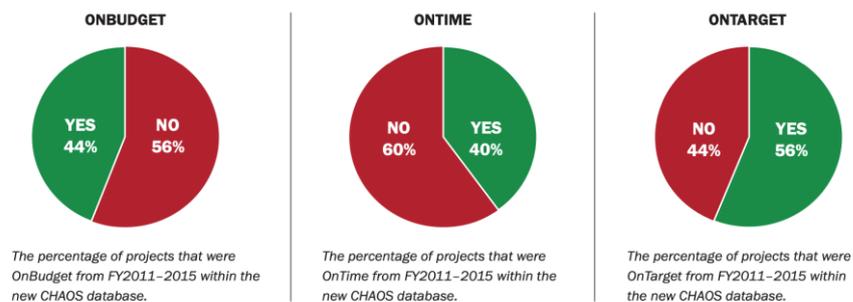


Figura 4 - Percentagens de projetos de Sucesso ou Falhos quanto ao Orçamento, Tempo e Requisitos (CHAOS Report, 2015).

Segundo o *CHAOS Report (The Standish Group, 2015)*, 56% dos projetos⁸ ultrapassam o valor do orçamento, 60% não são entregues no tempo previsto e 44% não cumprem os requisitos propostos. No ano de 2015, apenas 36% dos projetos foram classificados como Sucesso. Dados como estes representam a necessidade de

⁶ Últimas edições do *CHAOS Report*: 2015, 2016, 2018 e 2020. Últimas edições do *CHAOS Manifesto*: 2013, 2014 e 2015.

⁷ Em sua edição de 2015, o *CHAOS Report* apresenta mais três definições: *OnGoal*, *Value* e *Satisfaction*, que significam meta alcançada, valor agregado e satisfação.

⁸ De acordo com dados levantados nos anos fiscais entre 2011 e 2015.

buscar soluções para entregar os projetos de desenvolvimento de software no prazo previsto.

3.2.2 Outros trabalhos relevantes encontrados de forma ad hoc

Além do CHAOS Report, alguns outros trabalhos relevantes⁹ ao tema pesquisado foram utilizados como base¹⁰ para a QUASI-RSL - Revisão Sistemática da Literatura (**seção 3.3**). Esta seção detalha as principais informações coletadas a partir destes trabalhos.

Suliman e Kadoda (2017): os autores trazem uma lista com vários FC que influenciam as estimativas de custo e de tempo de um projeto, e que podem causar atrasos nas entregas previstas. Como exemplo, tem-se “apoio da alta gestão”, “falta de envolvimento do usuário” e “experiência da equipe”. Além disso, os autores propõem que as companhias ofereçam treinamento aos desenvolvedores e gerentes de projetos a fim de melhorar o processo de estimativa e predição.

Nguyen (2014): o autor frisa a importância do tratamento de exceções (erros e discrepâncias) durante o desenvolvimento de um projeto de software, e especula que o fracasso de um projeto pode ocorrer devido a um conjunto cumulativo de falhas dessa natureza. Propõe um framework chamado MBE - *Management By Exception* (Gerenciamento por exceções), a fim de monitorar a ocorrência desses erros.

Pradhan, Rishiwal e Agarwal (2016): os autores citam que um planejamento inadequado gera problemas nos projetos, principalmente no tocante às entregas, e sugerem o uso da ferramenta de gerenciamento de projetos *Microsoft Project* para auxiliar na divisão de papéis, atribuição de atividades para cada membro de equipe, estimativa aproximada de prazos, controle dos passos executados, entre outras atividades relacionadas.

Bilgaiyan, Mishra e Das (2016): os autores sugerem que uma predição precisa tem influência no sucesso de um projeto e propõem a utilização de um framework para uma melhor predição de esforço e custo do projeto de desenvolvimento de software.

⁹ Selecionados através de pesquisa em repositórios de busca sem um protocolo definido.

¹⁰ Esses artigos (“artigos-semente”) serviram como auxílio para calibrar a *string* de busca na QUASI-RSL e constatar que os resultados são adequados ao que se deseja encontrar. Mais informações na Seção 3.3 A Revisão Sistemática da Literatura.

3.3 A Revisão Sistemática da Literatura

O principal objetivo da QUASI-RSL foi encontrar os trabalhos relacionados ao tema pesquisado: fatores críticos para entregas no prazo previsto de projetos de desenvolvimento de software. O protocolo completo da QUASI-RSL pode ser encontrado no **APÊNDICE A** e os resultados obtidos com sua execução na **seção 3.3.2**. A QUASI-RSL foi conduzida com base no trabalho de Kitchenham (2007) e é uma técnica originária da Engenharia de Software Baseada em Evidências - ESBE.

A principal vantagem de se utilizar a Revisão Sistemática da Literatura é que esta propõe uma metodologia bem definida que torna menos provável que os resultados sejam tendenciosos. A revisão sistemática foca em questões de pesquisa bem definidas e de natureza causal (Kitchenham, 2007). Petersen et al (2007) frisam que a revisão sistemática é mais aprofundada do que outras técnicas de revisão bibliográfica e a síntese dos seus resultados foca mais em identificar as melhores práticas e sua efetividade. A Figura 5 ilustra as etapas do processo da Revisão Sistemática da Literatura executado neste trabalho.

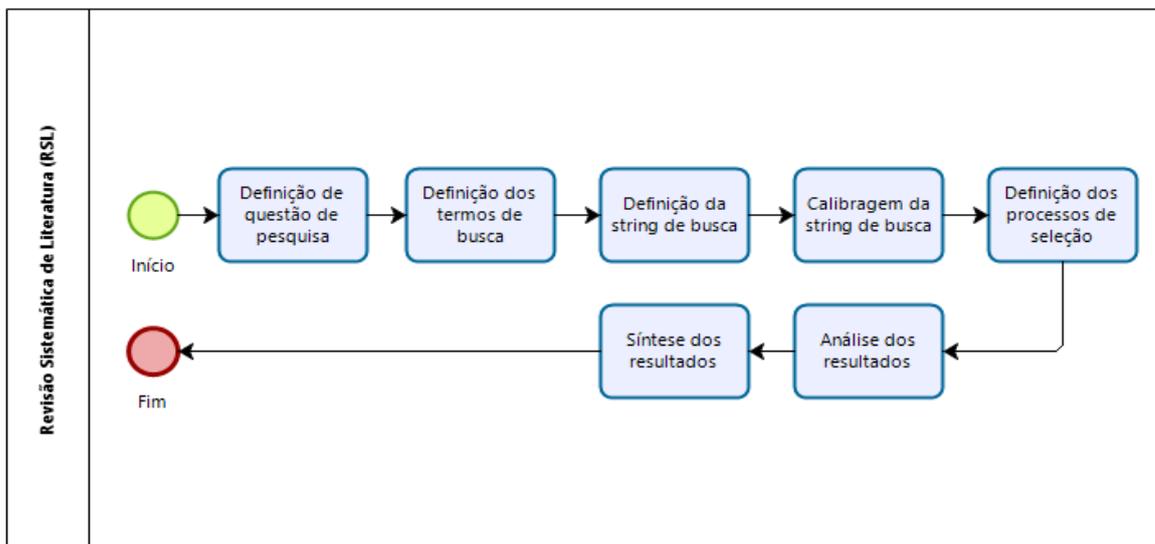


Figura 5 - Fluxo de Processo da QUASI-RSL

A definição da questão da pesquisa foi baseada no problema da pesquisa e, a partir dela, selecionaram-se os principais termos de busca, que são baseados nas principais palavras da questão de pesquisa. A seguir, definiu-se a *string* de busca a ser utilizada nos repositórios/engenhos de busca. Depois disso, executou-se a sua

calibragem (uma adaptação dos termos da *string*) por meio de testes nos repositórios/engenhos, obtendo-se finalmente a seguinte *string* de busca:

String de busca calibrada

((“IT project” OR “software project” OR “IT planning” OR “software planning”) AND (“software” OR “application” OR “sw”) AND (“critical success factors”) OR (“critical failure factors”) OR (“delay” OR “overrun”) AND (“schedule” OR “time”)) AND (“failure” OR “failing”))

Para validar a *string* de busca calibrada, verificou-se se os seguintes estudos, considerados relevantes, que foram previamente selecionados na pesquisa *ad hoc*, seriam retornados nas buscas dos repositórios utilizando a QUASI-RSL:

- O estudo de Suliman e Kadoda (2017)¹¹;
- O estudo de Nguyen (2014)¹²;
- O estudo de Pradhan, Rishiwal e Agarwal (2016)¹³;
- O estudo de Bilgaiyan, Mishra e Das (2016).¹⁴

Assim, esta *string* de busca foi testada nos repositórios *IEEEExplore Library* e *Scopus* e os estudos pré-selecionados foram retornados na busca, o que torna válida a *string* de busca proposta. Além disso, os seguintes resultados foram obtidos:

Pesquisa no Repositório/engenho IEEEExplore Library

Data da execução da busca: 26/06/2019

Quantidade de resultados encontrados neste repositório/engenho: 50

Avaliação dos resultados:

¹¹ *Factors that Influence Software Project Cost and Schedule Estimation* - Fatores que influenciam a estimativa de custo e cronograma de projetos de software, em tradução livre.

¹² *Software Project Management Towards Failure Avoidance* - Gerenciamento de Projetos de Software rumo a Prevenção de Falhas, em tradução livre.

¹³ *A Survey on Effectiveness of Tool based Software Project Planning* - Uma Pesquisa sobre a Efetividade de Gerenciamento de Projetos de Software baseado em Ferramenta, em tradução livre.

¹⁴ *A Review of Software Cost Estimation in Agile Software Development Using Soft Computing Techniques* - Revisão de estimativa de custos de software no desenvolvimento ágil de software usando técnicas de computação suaves, em tradução livre.

Foram encontrados inicialmente 55 resultados, porém ao configurar o filtro para buscar estudos a partir do ano de 2001¹⁵, o número de resultados encontrados caiu para 50 estudos, incluindo os estudos pré-selecionados.

Pesquisa no Repositório/engenho Scopus

Data da execução da busca: 26/06/2019

Quantidade de resultados encontrados neste repositório/engenho: 42

Avaliação dos resultados:

O número total de resultados retornados neste repositório foi de 42 documentos, incluindo os estudos pré-selecionados.

A etapa “Definição dos processos de seleção”, mencionada na Figura 5, refere-se aos passos do processo de seleção dos estudos com base em critérios de inclusão e exclusão e são detalhados na Figura 6.

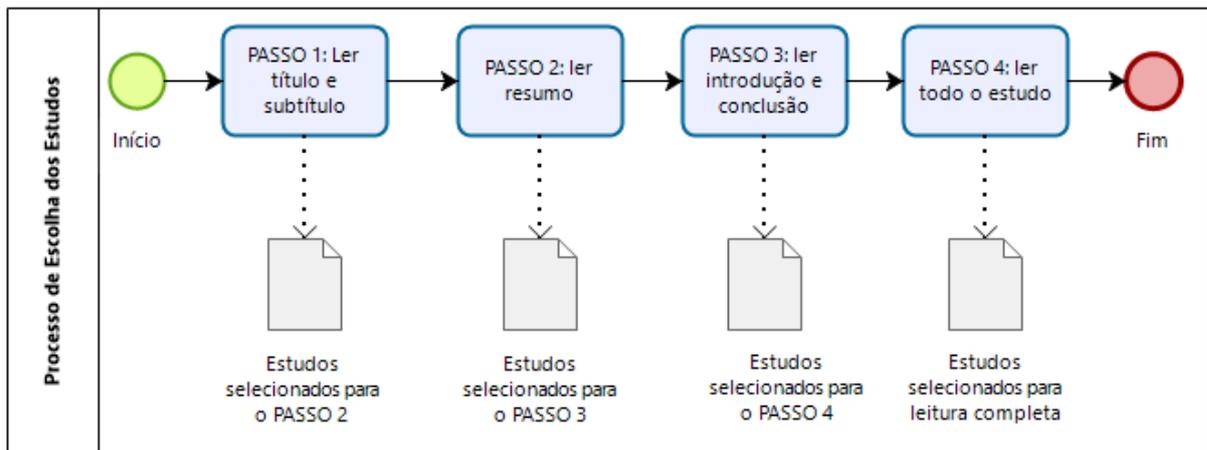


Figura 6 - Passo a passo do processo de seleção (Revisão Sistemática da Literatura - produzida em *Bizagi Modeler*)

A princípio leu-se o título e o subtítulo do estudo, e foram aplicados os critérios de seleção para fazer a triagem. No Passo 2 leu-se o resumo da obra e também aplicou-se os critérios. No Passo 3 a introdução e a conclusão do estudo foram lidas

¹⁵ O ano de 2001 foi escolhido, pois corresponde ao ano surgimento do *Agile Manifesto* (Manifesto Ágil). Acreditou-se, durante a execução deste trabalho, que a escolha desta data seria interessante pois não retornaria resultados obsoletos (pré 2001), dada a importância do Manifesto à área da Computação. Embora não haja um critério de seleção que selecione apenas estudos que tratem do Manifesto Ágil, acredita-se que este manifesto tenha trazido inovação à área, produzindo novos conhecimentos e gerando novos estudos.

e os critérios de seleção foram novamente aplicados. Com o resultado, aplicou-se o Passo 4, que é a leitura completa dos estudos selecionados.

Ao final de todos os passos do processo de seleção, 32 estudos sobre o tema abordado foram selecionados para serem analisados na íntegra. A Tabela 2 mostra o quantitativo de estudos incluídos e excluídos em cada passo da seleção.

Passo	# Estudos retornados	# Estudos incluídos	# Estudos excluídos
1	92	64	28
2	64	48	16
3	48	33	15
4	33	32	1

Tabela 2 - Quantitativo de estudos incluídos e excluídos em cada passo da seleção

Pode-se concluir que os resultados foram satisfatórios, pois um grande número de estudos foi retornado, os quais foram filtrados pelos critérios de seleção para posteriormente serem analisados e terem os resultados sintetizados (**seção 3.4**). A Tabela 3 mostra uma síntese dos resultados encontrados antes de depois da aplicação dos critérios de seleção.

Total de resultados encontrados antes da aplicação dos critérios de seleção	Total de resultados encontrados após aplicação dos critérios de seleção	Total de resultados excluídos
92	32	60

Tabela 3 - Totais de estudos relacionados ao tema abordado.

3.4 Os Fatores Críticos Identificados

Nesta seção, os Fatores Críticos (FC) encontrados na QUASI-RSL e na revisão *ad hoc* serão listados e brevemente conceituados. Como já mencionado, a QUASI-

RSL retornou 32 estudos (contando com os quatro estudos encontrados primeiramente na revisão *ad hoc*)¹⁶, totalizando 19 FC e 41 Técnicas propostas.

Os FC levantados foram: Engenharia de requisitos, Testes, Apoio da alta gestão, Gerenciamento de riscos, Estimativas e métricas, Gerenciamento de mudanças, Controle de qualidade, Troca de membros da equipe, Envolvimento do usuário/cliente, Gestão de Projetos de Softwares, Comunicação da equipe, Objetivos/metapas claros e bem definidos, Escopo, Experiência da equipe, Finanças, orçamento e preços, Ferramentas, Motivação, Treinamento e Exceções. Entre estes FC, os mais citados nos estudos selecionados foram: Estimativas e métricas (16 citações), Gestão de Projetos de Softwares (12 citações), Engenharia de requisitos (10 citações), Apoio da alta gestão (8 citações) e Envolvimento do usuário/cliente (7 citações).

3.4.1 Detalhamento dos FC identificados

Os FC detalhados nesta subseção estão categorizados de acordo com a ideia mais geral à qual estão vinculados (por exemplo, “levantamento de requisitos” e “engenharia de requisitos” foram agrupados em “engenharia de requisitos”; “apoio da alta gestão” e “comprometimento da alta gestão” foram agrupados em “apoio da alta gestão”). Este procedimento se fez necessário para agrupar os FC em *clusters*¹⁷ e facilitar sua categorização e análise. Os FC considerados menos expressivos, ou seja, que possuem menor número de citações¹⁸ na literatura foram ignorados para que se possa centralizar a análise nos FC mais comuns.

A Tabela 4 associa identificadores aos artigos encontrados na revisão sistemática da literatura, a fim de identificá-los e referenciá-los em tabelas e figuras posteriores.

¹⁶ Os quatro estudos que apresentam Fatores Críticos encontrados na revisão *ad hoc* foram utilizados como “artigos-semente” para validar a *string* de busca e, portanto, foram retornados na QUASI-RSL.

¹⁷ *Cluster* é um aglomerado ou agrupamento de itens respeitando algum aspecto em comum.

¹⁸ Os Fatores Críticos com uma, duas ou três menções na QUASI-RSL.

ID	Autor	ID	Autor
001	Khan e Quraishi (2014)	017	Hashim, Abbas e Hashim (2013)
002	Gupta, Chauhan e Dutta (2013)	018	Sheta, Rine e Ayesh (2008)
003	Kutsch e Hall (2005)	019	Govindaraju, Hariadi e Sidiq (2015)
004	Abdul-Rahman, Mohd-Rahim e Chen (2011)	020	Heier, Borgman e Hofbauer (2008)
005	Jodpimai, Sophatsathit e Lursinsap (2018)	021	Kaur e Sehra (2014)
006	Jones (2004)	022	Shahzad e Mathkour (2009)
007	Hussain, Mkpojiogu e Kamal (2016)	023	Ayyubi, Ahmad e Faiz (2007)
008	Cerpa e Verner (2009)	024	Octavianus e Mursanto (2018)
009	Pradhan, Rishiwal e Agarwal (2019)	025	Raunak e Binkley (2017)
010	Bilgaiyan, Mishra e Das (2016)	026	Nazir et al. (2014)
011	Suliman e Kadoda (2017)	027	Nasir e Sahibuddin (2011)
012	Nguyen (2014)	028	Hong, Yoo e Cha (2010)
013	DeMarco (2011)	029	Melo et al. (2013)
014	Jamshidi et al. (2014)	030	Ahimbisibwe e Cavana (2016)
015	Govindarajan (2015)	031	Sheta, Ayesh e Rine (2010)
016	Jing e Qiu (2007)	032	Southekal e Levin (2011)

Tabela 4 - Identificadores dos artigos encontrados na QUASI-RSL

A Tabela 5 lista os Fatores Críticos encontrados na Revisão Sistemática da Literatura e os relaciona aos identificadores dos artigos mencionados na Tabela 4. O rodapé da Tabela 5 possui o campo “Soma” indicando o número de citações de cada FC nos artigos.

ID Autor	Apoio da alta gestão	Comunicação da equipe	Controle de qualidade	Engenharia de requisitos	Envolvimento do usuário/cliente	Escopo	Estimativas e métricas	Exceções	Experiência da equipe	Ferramentas	Finanças, orçamento e preços	Gerenciamento de mudanças	Gerenciamento de riscos	Gestão de Projetos de Softwares	Motivação	Objetivos/metras claros e bem definidos	Testes	Treinamento	Troca de membros da equipe
001																			
002				X													X		
003	X											X							
004												X							
005							X												
006			X	X			X				X		X				X		X
007	X			X	X														
008				X															
009													X						
010							X						X						
011	X	X			X	X	X		X		X		X	X	X				
012								X											
013							X												
014												X							
015				X			X		X		X		X			X			
016	X				X		X			X			X					X	
017		X		X		X	X			X	X		X						
018							X												
019	X																		
020	X												X						
021							X												
022				X															
023							X					X	X			X			
024		X			X				X				X			X			
025		X		X	X		X												
026				X															
027	X	X		X	X		X		X										
028													X						
029							X					X							
030	X	X			X						X		X						
031							X												
032							X												
SOMA	8	6	1	10	7	2	16	1	3	1	3	4	5	12	1	4	2	1	1

Tabela 5 - FC Encontrados versus Artigos

FC001 - Apoio da alta gestão: é um fator relacionado às pessoas (NASIR; SAHIBUDDIN, 2011) que refere-se à aprovação e autorização de projetos por parte da alta gestão da organização, bem como a sua ciência de que o projeto possui alta prioridade, além da indicação de algum *stakeholder* como parte interessada e ativa

nos projetos (AHIMBISIBWE; CAVANA, 2016). A interferência deste FC em um projeto tem relação ao quanto a alta gestão da organização vai apoiar a execução do projeto, seja fornecendo recursos ou autorizando atividades essenciais, por exemplo.

FC002 - Comunicação da equipe: segundo Suliman e Kadoda (2017), refere-se à coordenação e comunicação que devem acontecer durante todo o ciclo de vida de desenvolvimento de um projeto e é, quando há falha neste FC, um dos principais fatores para atraso e falha em projetos de software. Hashim, Abbas e Hashim (2013) citam que uma comunicação pobre tem grande impacto em qualquer projeto de software. Sucesso na comunicação é um FCS também citado por Raunak e Binkley (2017). No trabalho de Nasir e Sahibuddin (2011), comunicação efetiva e feedback figuram no sétimo lugar de importância de seu ranking de FCS. Para finalizar, Ahimbisibwe e Cavana (2016) destacam que *IPC - Internal Project Communication* (comunicação interna do projeto, em Português) é uma prática que aumenta a troca de informações e a coesão entre os times de desenvolvimento. Por exemplo: a não comunicação de que uma tarefa já foi realizada por uma pessoa pode fazer com que esta tarefa seja realizada novamente por outro membro da equipe, causando atraso do projeto.

FC003 - Controle de qualidade: Jones (2004) destaca que um efetivo controle de qualidade do software é um dos principais fatores que separam projetos fracassados de projetos bem sucedidos. A razão para isso é que encontrar e corrigir erros em grandes sistemas (principalmente) custa caro e toma mais tempo que outras atividades. Um bom controle de qualidade envolve tanto a prevenção quanto a detecção e remoção de erros. A detecção e correção de erros de código logo no início do projeto evita grandes perdas de recursos e refatoração de módulos no futuro, por exemplo.

FC004 - Engenharia de requisitos: refere-se às atividades de levantamento, elicitação, priorização, análise e gerenciamento dos requisitos do projeto. É um dos FC mais citados na literatura. Gupta, Chauhan e Dutta (2013) citam que, para evitar atrasos, os desenvolvedores de software tentam desenvolver incrementalmente, implementando requisitos com alta prioridade primeiro. Estes requisitos são os que mais agregam valor ao cliente. Uma efetiva priorização de requisitos depende da qualidade dos *stakeholders* envolvidos no projeto. Jones (2004) destaca que a Engenharia de requisitos é um FC importante pois possui atividades-chave, como “lidar efetivamente com mudança de requisitos” e “alocar tempo suficiente para análise

de requisitos”. Hussain, Mkpojiogu e Kamal (2016) afirmam que a engenharia de requisitos é central para todo o projeto de desenvolvimento de software. Cerpa e Verner (2009) destacam que a maioria dos projetos falham devido a requisitos inadequados. Já Hashim, Abbas e Hashim (2013) descrevem que requisitos irreais têm grande impacto em projetos de software. Sem um bom levantamento de requisitos junto ao cliente, todo o esforço da equipe pode ser desconsiderado caso o cliente não fique satisfeito com um módulo ou funcionalidade solicitada, por exemplo, levando ao retrabalho.

FC005 - Envolvimento do usuário/cliente: refere-se à participação do usuário/cliente no processo de desenvolvimento, com constante comunicação, esclarecimento de dúvidas dos desenvolvedores e correção de equívocos relacionados a requisitos. Participação em reuniões das equipes também é uma boa prática. Uma comunicação honesta e aberta com o cliente é importante para satisfazer suas necessidades de informação (JING; QIU, 2007). Ahimbisibwe e Cavana (2016) citam que o “envolvimento do usuário” reflete o quão próximo ele está das atividades referentes ao projeto, além das suas atividades realizadas com o objetivo de contribuir para o projeto. Um cliente ausente deixa a equipe com poucas informações sobre os objetivos da funcionalidade solicitada, o que pode ocasionar refatoração e, conseqüentemente, atraso.

FC006 - Escopo: Suliman e Kadoda (2017) destacam que um dos principais FC é um escopo de projeto que a equipe envolvida entende bem, é coeso e passa por poucas mudanças no ciclo de vida do projeto. Para Hashim, Abbas e Hashim (2013), a mudança de escopo é o mais importante fator de risco e tem muito impacto quando não tratada apropriadamente. Por exemplo, caso o escopo mude na metade do projeto, será necessário refazer o planejamento e, ainda, programar novos prazos para entrega, prejudicando o especificado originalmente.

FC007 - Estimativas e métricas: referem-se tanto às predições quantitativas de tempo, esforço e custo como às métricas, que são as medidas que envolvem o ciclo de vida de desenvolvimento de software, como quantidade de linhas de código em um módulo ou quantidade de erros reportados pela equipe de testes em certo período. Jodpimai, Sophatsathit e Lursinsap (2018) explicam que uma boa estimativa tende a resultar em um projeto de sucesso; já uma estimativa malfeita pode resultar em perda financeira ou em falha do projeto de software. Como exemplo, tem-se a que a má previsão de datas de entregas de funcionalidades ao cliente pode, como efeito

cascata, desencadear atraso em todas as outras fases do ciclo de vida. Por isso se faz importante melhorar a precisão das estimativas, embora seja uma tarefa desafiadora (JODPIMAI; SOPHATSATHIT; LURSINSAP, 2018). Uma estimativa precisa significa um melhor planejamento e um uso eficiente dos recursos do projeto (KAUR; SEHRA, 2014). Para Bilgaiyan, Mishra e Das (2016), uma estimativa precisa é tarefa essencial e conduz a um projeto de software de sucesso.

Quanto às métricas, essas se fazem importantes pois são informações claras e objetivas para se passar aos envolvidos no projeto, além de ajudarem a rastrear o progresso do projeto (SOUTHEKAL e LEVIN, 2011).

FC008 - Exceções: no contexto do ciclo de vida de projeto, pode-se entender que as exceções são pequenas falhas ocorridas no decorrer da execução das tarefas. Embora uma exceção isolada possa não acarretar em atraso ou em fracasso do projeto, um conjunto de exceções acumuladas ou agrupadas pode ocasionar fracasso do projeto de software (NGUYEN, 2014). Um exemplo são as falhas no código-fonte do software ou falhas de inserção de informações em um banco de dados. Um bom gerenciamento de exceções permite evitá-las e mitigá-las a fim de minimizar seu impacto, é também um FC. Como exemplo temos um desenvolvedor que não tratou exceções de divisão por zero no código. Em algum momento este erro poderá ocorrer após a implantação, o que pode trazer prejuízo à equipe, pois esta terá que fazer a correção no código atrasando a finalização do projeto.

FC009 - Experiência da equipe: em Octavianus e Mursanto (2018), a experiência da equipe é classificada como um “fator humano” e está presente no rol dos principais FC. De acordo com Suliman e Kadoda (2017), uma equipe inexperiente possui incapacidades (por exemplo, inexperiência na área de atuação e poucas habilidades para realizar as tarefas) e falta de comprometimento com o projeto, entre outros problemas. Assim, a contratação de profissional com grande experiência, mesmo que seja por um salário acima da média, pode facilitar o desenvolvimento de determinadas atividades do projeto.

FC010 - Ferramentas: são recursos tecnológicos que auxiliam no desenvolvimento do projeto de software, tais como: linguagens de programação, tecnologias de arquitetura de redes e/ou hardware e softwares acessórios. Govindarajan (2015) cita alguns exemplos de fatores relacionados às ferramentas que contribuem para o fracasso de projetos de software, como falhas nos procedimentos de backup ou em processos de recuperação de falhas. Uma ferramenta de

versionamento pode auxiliar a equipe na restauração de partes do código, evitando retrabalho, por exemplo.

FC011 - Finanças, orçamento e preços: fatores relacionados às finanças, questões orçamentárias e preços foram agrupadas em um só tópico. De acordo com Suliman e Kadoda (2017), projetos com orçamento não muito claro são comuns, e Jing e Qiu (2007) destacam que um orçamento com recursos financeiros abundantes tende a ser um fator positivo para o sucesso do projeto. Alguns exemplos: falta de recursos para contratação de pessoal ou para atualização de hardware, aumento de preços de fornecedores, competição agressiva no mercado etc.

FC012 - Gerenciamento de mudanças: gerir as mudanças significa adaptar o projeto às alterações de escopo, requisitos, objetivos, etc. que não foram previstas. Um gerenciamento de mudanças eficaz deve envolver intensa colaboração entre cliente e desenvolvedores, revisão formal dos pedidos de mudanças, entre outros (JONES, 2004). Para Hashim, Abbas e Hashim (2013), o gerenciamento de mudanças é um dos fatores mais críticos associados a projetos de software. Quando ausente, a equipe de desenvolvimento pode não estar preparada para o surgimento de novos requisitos, causando acúmulo de tarefas e, posteriormente, atraso de entrega, por exemplo.

FC013 - Gerenciamento de riscos: pode ser definido como a administração dos fatores ou condições que causam ameaças para a finalização com sucesso do projeto de software (ABDUL-RAHMAN; MOHD-RAHIM; CHEN, 2011). O gerente do projeto tem a escolha de encarar o risco aplicando o gerenciamento de riscos para mitigá-lo ou não (KUTSCH; HALL, 2005). De acordo com Jamshidi et al. (2014), um dos maiores problemas no gerenciamento de riscos é a análise superficial de riscos, ignorando a interdependência entre eles. Um exemplo clássico é o risco de sinistros: uma inundação ou incêndio pode destruir o *datacenter* da organização fazendo com que se perca todos os arquivos dos projetos. Calcular este risco e realizar backups é uma das possíveis soluções.

FC014 - Gestão de Projetos de Softwares (GPS): refere-se à forma como os projetos de software são planejados, executados, mantidos e controlados. Uma Gestão de Projetos de Software de alta qualidade conduz a um bom produto final, enquanto uma Gestão mal conduzida pode levar à falha do projeto (BILGAIYAN; MISHRA; DAS, 2016). De acordo com Suliman e Kadoda (2017), a indústria de software contribui para as falhas nos projetos usando abordagens de gestão

desestruturadas e pouco documentadas. Govindarajan (2015) reitera que a *GPS* é um dos fatores mais significativos a serem considerados para se obter os melhores resultados em um projeto. O planejamento (uma das fases do ciclo de vida do projeto de software e parte da Gestão de Projetos de Software) quando mal executado pode acarretar ineficiências no projeto e, então, conduzir a alterações na previsão de entrega, além de um orçamento inflacionado (AHIMBISIBWE; CAVANA, 2016). Ayyubi, Ahmad e Faiz (2007) concordam que um mal planejamento, além de outros fatores, é a causa da perda de controle dos projetos de software: “Planejamento é a área chave a qual determina o sucesso ou o fracasso. Geralmente não se dá muita atenção à fase de planejamento, embora seja uma fase essencial.” (AYYUBI; AHMAD; FAIZ, 2007).

FC015 - Motivação: segundo Suliman e Kadoda (2017), a motivação, aliada à sensação de responsabilidade da equipe é um fator de cultura organizacional e é tratada como fator crítico ao sucesso ou ao fracasso do projeto de software. Um funcionário desmotivado pode demorar mais tempo para realizar uma atividade, ou até fazer de forma errada, por exemplo.

FC016 - Objetivos/metapas claros e bem definidos: o estabelecimento de objetivos e metas claros e definidos tem influência no sucesso do projeto, principalmente quando relacionado à alta gestão da organização (SULIMAN; KADODA, 2017). Ayyubi, Ahmad e Faiz (2007) também destacam que, na maioria das vezes, objetivos pouco claros são uma das causas de perda de controle de projetos de software. Por exemplo, um programador que não entendeu muito bem o objetivo da tela que desenvolve pode demorar para codificá-la, atrasando o projeto.

FC017 - Testes: os testes de software são o meio de assegurar que o que foi desenvolvido funciona de acordo com o solicitado. Os testes de regressão, por exemplo, são testes que auxiliam na análise dos impactos que novos trechos de código adicionados ao software podem causar no funcionamento do sistema como um todo (GUPTA; CHAUHAN; DUTTA, 2013). Jones (2004) sugere planos de testes formais e a construção de casos de testes para prevenir defeitos no projeto. Segundo o autor, a alocação de tempo suficiente para os testes e reparos de defeitos é um dos principais Fatores Críticos de Sucesso. Como exemplo, temos os testes unitários, que são realizados logo no início do desenvolvimento. Sem eles, a chance de haver um erro de código em fases posteriores é muito grande, o que pode atrasar as atividades.

FC018 - Treinamento: os treinamentos da equipe de desenvolvimento são mais eficazes quando moldados aos requisitos de cada grupo de usuários. Além disso, os treinamentos não devem ocorrer em um evento único no tempo, devem ocorrer de forma regular a fim de garantir que os colaboradores se mantenham atualizados sobre as mudanças no processo (JING; QIU, 2007). Um exemplo importante é o treinamento da equipe para a utilização de novas tecnologias de programação, o que pode dar velocidade à codificação evitando o atraso do projeto.

FC019 - Troca de membros da equipe: a previsão de novas contratações e a realocação de pessoal são considerados um fator crítico ao sucesso ou ao fracasso dos projetos de desenvolvimento de software segundo Jones (2004). A substituição de membros experientes e a facilidade de integração de novos membros à equipe são exemplos de influências diretas ao sucesso ou fracasso do projeto, pois com a entrada de novos membros na equipe frequentemente pode-se perder recursos (tempo, esforço) para adaptação.

De acordo com a QUASI-RSL realizada, além de apresentar os FC, alguns autores propuseram formas para melhor aplicá-los (no caso de FCS) ou mitigá-los (no caso de FCF). Por exemplo, uma “Comunicação efetiva” entre os membros da equipe de desenvolvimento conduz a uma boa aplicação do FC “Gerenciamento de Riscos”, conforme o autor [003].

A Tabela 6 exhibe a relação existente entre os FC, os artigos que citam esses FC, as propostas de aplicação de técnicas e/ou ferramentas para tratar os FC e os artigos que citam essas propostas.

FC	ID Artigos (Tabela 4) que citam esse FC	Técnica(s) e/ou Ferramenta(s) aplicadas a este FC	ID Artigos (Tabela 4) que citam essa(s) técnica(s)
FC001 - Apoio da alta gestão	003, 007, 011, 015, 016, 019, 020, 027, 030	Técnica, como a "Teoria da Difusão da Inovação" para avaliar características de líderes de projetos	019
FC002 - Comunicação da equipe	011, 017, 024, 025, 027, 030	<i>Nenhum estudo encontrou técnica(s) relacionada(s) a este FC</i>	--
FC003 - Controle de qualidade	006	Técnica/framework para Gerenciamento de Riscos	006

FC004 - Engenharia de requisitos	002, 006, 007, 008, 015, 017, 022, 025, 026, 027	Técnica/framework de "Engenharia de Requisitos"	002, 006, 007, 022 e 026
		Protótipos de software para uma melhor comunicação entre a equipe e o cliente	006
		Desenvolvimento iterativo	006
FC005 - Envolvimento do usuário/cliente	007, 011, 015, 016, 024, 025, 027, 030	Uso de metodologias ágeis	015 e 030
FC006 - Escopo	011, 017	<i>Nenhum estudo encontrou técnica(s) relacionada(s) a este FC</i>	--
FC007 - Estimativas e métricas	005, 006, 010, 011, 013, 015, 016, 017, 018, 021, 023, 025, 027, 029, 031, 032	Técnica/framework para estimar mais precisamente o tempo e o custo do projeto	005, 011 e 032
		Ferramentas automatizadas estimar mais precisamente o tempo e custo do projeto	006
		Comparações com dados históricos e projetos já finalizados	006
		Técnicas de métricas/estimativas (como soft computing, PSA, COCOMO, FPA, PSO ou LOC/kLOC)	010, 018, 021 e 031
FC008 - Exceções	012	Técnica de "Monitoramento de Exceções (no código)"	012
FC009 - Experiência da equipe	011, 024, 027	<i>Nenhum estudo encontrou técnica(s) relacionada(s) a este FC</i>	--
FC010 - Ferramentas	015	Ferramentas de gerenciamento de projetos como o Microsoft Project ou similar	009
FC011 - Finanças, orçamento e preços	011, 016, 017	Feedbacks periódicos, para se obter um melhor gerenciamento de riscos do projeto	011
FC012 - Gerenciamento de mudanças	006, 015, 017, 030	Uso de metodologias ágeis	015
FC013 - Gerenciamento de riscos	003, 004, 014, 023, 029	Técnica/framework de "Engenharia de Requisitos"	004
		Uso de metodologias ágeis	004
		Técnica para se obter uma "Comunicação efetiva" com o cliente	004
		Técnica para se obter um "Maior envolvimento do cliente" no projeto	004
		Feedbacks periódicos, para se obter um melhor gerenciamento de riscos do projeto	011

		Técnica/framework para Gerenciamento de Riscos	014, 017 e 029
FC014 - Gestão de Projetos de Softwares	006, 009, 010, 011, 015, 016, 017, 020, 023, 024, 028, 030	Ferramentas de gerenciamento de projetos como o Microsoft Project ou similar	006 e 009
		Técnica/framework de "Engenharia de Requisitos"	006
		Técnicas de métricas/estimativas (como soft computing, PSA, COCOMO, FPA, PSO ou LOC/kLOC)	015
		Scrum personalizado/alterado/customizado	028
		Utilização de metodologias ágeis	030
FC015 - Motivação	011	<i>Nenhum estudo encontrou técnica(s) relacionada(s) a este FC</i>	--
FC016 - Objetivos/metapas claros e bem definidos	011, 015, 023, 024,	<i>Nenhum estudo encontrou técnica(s) relacionada(s) a este FC</i>	--
FC017 - Testes	002, 006	Técnica/framework de "Engenharia de Requisitos"	002
FC018 - Treinamento	016	Treinamento para desenvolvedores e gerentes de projetos	011
FC019 - Troca de membros da equipe	006	<i>Nenhum estudo encontrou técnica(s) relacionada(s) a este FC</i>	--

Tabela 6 - Relação entre FC e técnicas/ferramentas

Ao analisar a Tabela 6, percebe-se que existe um grande número de FC e técnicas/ferramentas identificados através da revisão literatura. Entre os FC listados, os cinco FC mais mencionados são “Estimativas e métricas”, “Gestão de Projetos de Software”, “Engenharia de Requisitos”, “Apoio da alta gestão” e “Envolvimento do usuário/cliente”, os quais são detalhados a seguir:

Estimativas e métricas

As tarefas que envolvem o ciclo de vida de um projeto de software são similares entre as organizações, sejam elas públicas ou privadas. O fato é que em qualquer situação, o ciclo de vida do projeto deve ser contemplado e os prazos cumpridos. No entanto, conforme já mencionado (**seção 3.4.1**), sem um valor estimado, ao menos de forma aproximada, da quantidade de esforço (tempo, custo) necessário para

desenvolver (codificar, testar, implantar etc.), o cronograma tende a ser fictício. Neste contexto, as soluções propostas para tratar este FC consistem de:

- Utilização de técnicas automatizadas de predição, como *soft computing*, COCOMO e PSO [010, 018, 021, 031]. A título de exemplo, o autor Govindarajan (2015) propõe um framework que auxilia os gerentes de projetos no processo de tomada de decisão com técnicas baseadas em inteligência artificial (Fuzzy Logic, ou "Lógica Difusa", um conceito computacional que se baseia em como os seres humanos pensam, que combina a manipulação de valores, incertezas, julgamentos e ideias vagas);
- Utilização de dados históricos como base, ou seja, estimar o projeto corrente com base na análise de estimativas e métricas de outros projetos já finalizados também é indicada em [006], além de o uso de ferramentas automatizadas;
- No tocante às métricas, sugere-se também a utilização de LOC, KLOC¹⁹ e FPA [021], técnicas que também podem auxiliar nas estimativas.

Gestão de Projetos de Software

O FC “Gestão de Projetos de Software” também teve destaque, pois entende-se que o uso de uma metodologia que auxilia na gestão do projeto (administração de tarefas, controle de prazos, monitoramento de cumprimento de metas, formas de ações corretivas, entre outras atividades) tem um papel importante e estratégico para a organização. Maneiras não formais²⁰ de desenvolvimento de software tendem a ter efeito apenas paliativo, propondo soluções pontuais, em detrimento às metodologias de Gestão de Projetos de Software existentes²¹, que visam um programa²² de projetos robusto, coeso e de longo prazo. Neste contexto, as soluções propostas para tratar este FC consistem em:

- Utilização de metodologias ágeis, como o XP e o Scrum, por exemplo. A customização do Scrum também é uma opção;
- Segundo [015], a utilização de um framework baseado em *soft computing* também pode ser considerada;

¹⁹ KLOC refere-se à *Kilo Lines Of Code* (mil linhas de código), que é um tipo de métricas para códigos de softwares.

²⁰ Formas que não utilizam o planejamento, estabelecimento de metas, divisão de tarefas etc. para gerir projetos de desenvolvimento de software e focam apenas em resolver problemas pontuais da área de Tecnologia da Informação.

²¹ Para citar alguns exemplos: *Waterfall* (cascata) e *Agile* (ágil).

²² Um programa de projeto é uma coleção de vários projetos não necessariamente relacionados.

- Em [009], sugere-se a utilização da ferramenta de gestão de projetos *Microsoft Project* para auxiliar a equipe a realizar atividades essenciais à gestão de projetos, como o agendamento de tarefas, controle de uso de recursos (custos), acompanhamento do progresso do projeto, distribuição de tarefas entre os membros da equipe, entre outras;
- Em [006] também é sugerida a utilização de alguma técnica ou algum framework para Engenharia de Requisitos.

Engenharia de Requisitos

O processo de “Engenharia de Requisitos” também se mostrou de extrema importância. Gerir os requisitos é essencial para o sucesso do projeto de software, pois é uma forma de garantir que o que foi solicitado pelo cliente foi desenvolvido. Em algumas situações, o próprio cliente não sabe expressar sua necessidade, o que pode demandar o envolvimento de um engenheiro de requisitos experiente e que saiba conduzir adequadamente o processo de elicitação de requisitos, garantindo uma total compreensão dos requisitos de ponta a ponta, ou seja, do cliente final ao desenvolvedor.

A adoção de técnicas de engenharia de requisitos pode aumentar a qualidade do software, pois requisitos mal interpretados, faltantes ou incompletos podem causar incorreções no produto de software. As soluções propostas para tratar este FC consistem de:

- Em [002] é proposta uma técnica de priorização de requisitos (CCR²³) para focar nos requisitos mais essenciais ao usuário e nos que têm menor necessidade de testes de regressão (o objetivo dos testes de regressão é garantir que não surgiram novos defeitos em componentes já analisados, ou seja, é um tipo de teste que não verifica apenas os novos códigos adicionados ao software, mas todo o código existente e suas relações com o que foi inserido e alterado);
- Em [022] é sugerida a utilização da técnica chamada FAST, que auxilia no encurtamento do caminho entre o cliente e o desenvolvedor. A técnica propõe a utilização de um facilitador, que é uma pessoa com várias características

²³ CCR significa *Customer satisfaction, Cost and Regression count* (satisfação do usuário, custo e contagem de regressão, respectivamente).

importantes (elencadas no estudo) que auxiliam na boa condução da reunião e na coleta de requisitos com os clientes;

- [006], [007], [022] e [026] sugerem o uso técnicas para Engenharia de Requisitos sem se aprofundar em nenhuma;
- Em [006] também se sugere o uso de Desenvolvimento Iterativo.

Apoio da alta gestão

O “Apoio da alta gestão” garante a continuidade dos projetos, a disponibilização de recursos necessários para mantê-los, bem como facilita a inclusão dos projetos de desenvolvimento de software (ou a área de TI em geral) no planejamento estratégico da organização, o que é um fator positivo.

A partir da QUASI-RSL, encontrou-se apenas uma técnica relacionada a este FC:

- Em [019] os autores propõem avaliar as características dos líderes (alta gestão da organização) através da *DOI - Diffusion Of Innovation* (Difusão da Inovação). Desta forma pode-se examinar a capacidade que o líder tem de inovar (buscar novas soluções para problemas), entender suas influências no projeto, lidar com mudanças e medir o seu nível de conhecimento na área de computação. Os líderes das equipes “possuem maior nível na pirâmide hierárquica e isso pode influenciar os níveis abaixo” [019]. Além disso, segundo os autores mencionados a *DOI* é adaptável e ajuda a relacionar novas tecnologias aos métodos de trabalho operacional através de análises organizacionais. Os autores aplicaram o estudo no contexto de um sistema *ERP - Enterprise Resource Planning* (Sistema Integrado de Gestão), que está intimamente ligado ao Retorno de Investimento (*ROI - Return Of Investment*) da organização.

Envolvimento do usuário/cliente

A importância deste FC reside no fato de que o cliente precisa estar sempre próximo ao projeto para poder solicitar alterações, sanar dúvidas dos desenvolvedores e modificar requisitos mal elaborados durante o desenvolvimento, a fim de que o impacto no projeto seja menor quando comparado a realizar grandes alterações após a entrega. Neste contexto, a metodologia ágil *Scrum* sugere que a equipe de

desenvolvimento entregue periodicamente, e de forma incremental, partes do projeto²⁴ ao cliente (geralmente duas semanas) de modo que ele possa validar constantemente o que for desenvolvido.

No tocante às técnicas sugeridas para tratar este FC, os artigos identificados propuseram a seguinte:

- Tanto em [015] como em [030] vê-se a sugestão de utilizar metodologias ágeis de desenvolvimento de software para alcançar um maior envolvimento do usuário no projeto.

3.5 Considerações finais do capítulo

Este capítulo apresentou os estudos encontrados na Revisão de Literatura e, a partir deles, os conceitos elencados. Alguns estudos foram encontrados de forma *ad hoc*, com a finalidade de auxiliar o pesquisador em outras fases da pesquisa, bem como para servir de base para a revisão da literatura utilizando a técnica de Revisão Sistemática, a qual selecionou ao todo 32 estudos após a aplicação de todos os passos.

Durante a execução da Revisão Sistemática, o uso de uma *string* de busca longa e com vários conectores fez que alguns repositórios retornassem um grande número de estudos, o que poderia inviabilizar a análise por questões de tempo. Desta forma, a *string* de busca foi adaptada e calibrada até que um número de estudos satisfatório fosse retornado.

Outro fator que dificultou a pesquisa em alguns repositórios foi o fato de alguns estudos serem pagos ou protegidos.

Além dos repositórios IEEEExplore Library e Scopus, citados anteriormente, foram feitas tentativas de busca em outro repositório/engenho de busca, o qual foi descartado pelos motivos descritos a seguir:

- No repositório *ScienceDirect* foi utilizada a busca avançada para ser possível inserir a *string* de busca em sua integridade. Porém, ao inserir a *string* de busca no campo, uma mensagem de erro foi retornada afirmando que só são permitidos oito conectores booleanos na *string* (a *string* possui 18). Portanto, para não

²⁴ Incrementos que possam ser utilizados e que agreguem valor ao cliente.

descaracterizar a *string* e, conseqüentemente, a QUASI-RSL, concluiu-se que o uso deste repositório era inviável.

- Outra dificuldade encontrada deu-se pelo grande número de artigos retornados. O número de resultados retornados neste engenho foi de 996.000. Após inserir o período desejado (2001 a 2019), o número caiu para 180.000. Quando removidos os filtros de “incluir patentes” e “incluir citações”, o número de resultados caiu para 139.000. Por ser um número grande de estudos e pelo fato de o engenho em questão não oferecer opções para um maior refinamento da busca avançada, concluiu-se que não haveria tempo suficiente para realizar o levantamento das informações nesta quantidade de estudos. Desta forma, o uso deste engenho foi descartado por ser inviável.

A partir dos 32 estudos selecionados na QUASI-RSL foram encontrados 19 Fatores Críticos, além de 41 técnicas/ferramentas para auxiliar na entrega de projetos de desenvolvimento de software no prazo previsto.

Nota-se que os autores, em sua maioria, propuseram técnicas/ferramentas para atuar apenas sobre um FC. Propostas abordando mais de um FC foram encontradas apenas em [006] e [011].

Após levantamento dos FC e das técnicas/ferramentas para auxiliar na entrega de projetos de desenvolvimento de software no prazo previsto, prosseguiu-se com a pesquisa de campo, com o objetivo de constatar se as situações descritas na literatura também ocorrem nos Institutos Federais de Educação pesquisados, o que é descrito no próximo capítulo.

4 O ESTADO ATUAL DOS FATORES CRÍTICOS PARA ENTREGAS NO PRAZO DOS PROJETOS DOS IFs

4.1 Introdução

Este capítulo apresenta informações sobre a aplicação do questionário nos IFs com o objetivo de fazer um levantamento de informações acerca do estado atual dos FC nos projetos de desenvolvimento de software, seu impacto nos prazos dos projetos e na satisfação dos clientes, bem como a utilização ou não de técnicas, metodologias ou ferramentas relacionadas aos FC identificados.

4.2 O estado atual dos FC nos projetos de desenvolvimento de software dos IFs

Um questionário foi submetido aos IFs do Brasil com o objetivo de levantar informações sobre a ocorrência de FC relacionados às entregas fora do prazo previsto nos projetos de desenvolvimento de software em tais instituições, bem como sobre a utilização (ou falta de utilização) de técnicas de melhoria de projetos de software associadas aos FC identificados. Os respondentes do questionário foram servidores dos IFs que possuem alguma relação com a área de TIC de sua instituição.

Utilizou-se a ferramenta *Google Forms* para disponibilizar o questionário de forma online²⁵. Esta ferramenta facilita a elaboração de formulários, o envio aos respondentes, bem como a tabulação e análise dos dados. O link do formulário foi submetido por e-mail e mídias sociais e ficou disponível para respostas por um período de 18 dias, entre 30/12/2019 e 17/01/2020.

A aplicação do questionário foi útil para se obter a opinião dos servidores dos setores de TIC dos IFs acerca da influência dos FC encontrados na literatura sobre o cumprimento de prazos nos projetos de desenvolvimento de software, além de coletar dados sobre as técnicas utilizadas nos IFs para tratar tais FC. Este questionário pode ser encontrado na íntegra no **APÊNDICE B**.

²⁵ <https://www.google.com/intl/pt-BR/forms/about/>

Na fase de análise dos dados coletados utilizou-se uma abordagem quantitativa para aprofundar a compreensão das organizações, buscando descobrir as causas dos fenômenos (os fatores que impactam nas entregas dos projetos) e as medidas que podem ser tomadas (técnicas que podem ser utilizadas para entregar os projetos no prazo previsto). Gerhardt e Silveira (2009) trata a abordagem qualitativa como uma “preocupação com aspectos da realidade que não podem ser quantificados”.

O questionário foi aplicado com o objetivo de compreender os fatores que podem contribuir para o sucesso ou fracasso de projetos de desenvolvimento de software nos IFs do Brasil. Para isso, em uma das questões, os FC encontrados na literatura foram listados para que o respondente assinalasse quais destes ele entende que tenham influência no cumprimento de prazos dos projetos de desenvolvimento de software de sua instituição.

Além disso, perguntou-se ao respondente se este tem conhecimento de alguma técnica²⁶ utilizada no seu ambiente de trabalho para auxiliar na entrega de projetos de software dentro dos prazos previstos. Este questionamento foi necessário para entender como os IFs estão lidando com seus projetos de desenvolvimento de software.

Para facilitar o entendimento do respondente e não tornar o questionário extenso, algumas questões foram resumidas. Por exemplo, criou-se somente uma pergunta sobre o uso de alguma técnica de Engenharia de Requisitos em vez de criar-se várias perguntas sobre cada técnica da Engenharia de Requisitos. Além disso, o questionamento sobre o uso de técnicas de estimativas e métricas (como por exemplo, soft computing, PSA, COCOMO, FPA, PSO ou LOC/kLOC) foi resumido em uma só questão.

Em todas as questões sobre o uso de técnicas associadas a algum FC, colocou-se um campo “Outros” para que o respondente especifique o uso de alguma técnica não explicitada na pergunta.

²⁶ Técnica em sentido amplo: neste contexto, este termo pode ser entendido como um passo a passo, uma metodologia, algum procedimento, processo etc.

4.2.1 Perfil do respondente

O questionário foi submetido a apenas um servidor de cada Instituto Federal para que a pesquisa não se tornasse incoerente, pois caso o número de respondentes variasse de instituição para instituição, as análises poderiam ser afetadas.

Os respondentes trabalham ou já trabalharam no setor de TIC de sua instituição na qualidade de servidores públicos federais (homens ou mulheres, Professores, Analistas de Tecnologia da Informação ou Técnicos da Tecnologia da Informação) com cargos comissionados/funções de confiança ou não.

4.2.2 Teste-piloto do questionário

Um teste-piloto do questionário foi conduzido previamente com uma amostra de seis servidores do IFTM, com o mesmo perfil esperado dos respondentes do questionário final. Entre os respondentes do teste-piloto foram incluídos tanto Técnicos em TI quanto Analistas de TI. O teste-piloto ficou disponível para respostas por 3 dias, entre 30/11/2019 e 02/12/2019. Não houve interferência do pesquisador nas respostas e os dados foram coletados de forma anônima, via *Google Forms*.

O objetivo da aplicação de um teste-piloto foi analisar alguns aspectos do questionário, como clareza, tempo de resposta e estrutura, portanto ao fim do questionário-piloto colocou-se três questões adicionais para os respondentes avaliarem tais aspectos. Cada uma destas três questões tinha cinco opções de respostas enumeradas de 1 a 5 de acordo com a escala Likert, que consiste em um instrumento que permite aos pesquisadores atribuírem respostas gradativas sobre suas opiniões a respeito dos itens (Likert, 1932 apud Alves, 2015). Nesta medição, o número 1 indica “Discordo totalmente”, o número 2 indica “Discordo em parte”, o número 3 indica “Nem concordo, nem discordo”, o número 4 indica “Concordo em parte” e o número 5 indica “Concordo totalmente” (da Silva, 2017).

Quando perguntados se “As perguntas no questionário eram claras e fáceis de entender”, obteve-se um bom conjunto de respostas favorável (Figura 7), pois 50% indicou “Concordo totalmente”, 33,3% indicou a opção “Concordo em parte” e 16,7% a opção “Nem concordo, nem discordo”. Além disso, nenhum respondente assinalou as opções “Discordo totalmente” e “Discordo em parte”, as quais seriam desfavoráveis ao questionário, logo entende-se que no aspecto Clareza o questionário está adequado.

As perguntas do questionário são claras e fáceis de entender

6 respostas

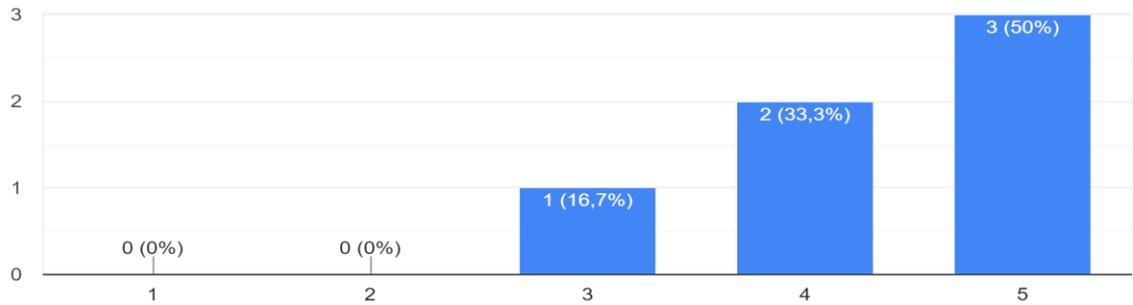


Figura 7 - Questão 1 do teste-piloto (clareza)

Quando perguntados se “A estrutura (a forma e a ordem que as perguntas foram dispostas) do questionário é satisfatória”, pode-se observar na Figura 8 que o aspecto Estrutura também está adequado, visto que a maioria dos respondentes (66,7%) assinalou a opção “Concordo totalmente”.

A estrutura (a forma e a ordem que as perguntas foram dispostas) do questionário é satisfatória

6 respostas

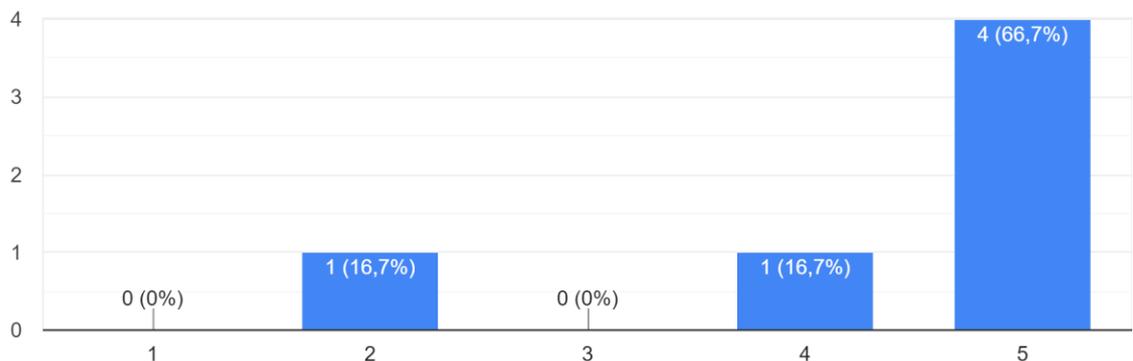


Figura 8 - Questão 2 do teste-piloto (estrutura)

Por fim, quando perguntados se “O tempo necessário para responder o questionário foi satisfatório”, obteve-se também um parecer favorável, pois 50% dos respondentes assinalaram a opção “Concordo totalmente” e 50% a opção “Concordo em parte” (Figura 9).

O tempo necessário para responder o questionário foi satisfatório

6 respostas

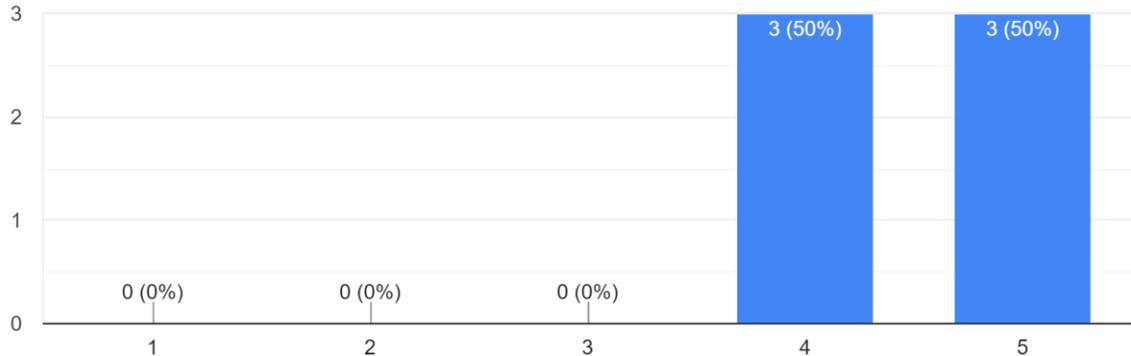


Figura 9 - Questão 3 do teste-piloto (tempo de resposta)

Com base na resposta a estas três perguntas, conclui-se que o questionário estava adequado e poderia ser aplicado ao universo de respondentes proposto. Os demais dados foram desconsiderados para a pesquisa final. As questões do teste-piloto podem ser vistas no **APÊNDICE B**.

A próxima seção apresenta a discussão acerca da relação entre os dados coletados pelo questionário final e os dados levantados na literatura.

4.3 Resultados e análise

Esta seção apresenta os resultados da aplicação do questionário nos IFs. A comparação dos resultados obtidos no questionário com os obtidos a partir da literatura contribui para a definição do estado atual dos projetos de desenvolvimento de software dos Institutos Federais de Educação no tocante a entregas fora do prazo previsto, nível com que estas entregas afetam a satisfação do cliente, e descrição dos fatores críticos mais relevantes associados a entregas no prazo, bem como das técnicas mais utilizadas associadas a estes fatores.

4.3.1 Resultado da aplicação do questionário nos IFs

Após aplicação do questionário no contexto de todos os IFs, obtiveram-se os seguintes resultados:

Perfil do respondente

- Quantidade de respondentes: 26;
- Quanto ao cargo: 38,5% Técnico de TI, 53,8% Analista de TI e 7,6% outro;
- Quanto a exercer Função de Confiança ou possuir Cargo Comissionado: 42,3% Sim e 57,7% Não;
- Para os que exercem Função ou possuem Cargo Comissionado, 36,4% são Coordenadores de setor de TI, 54,5% são Diretores de setor de TI, e 9,1% responderam “Outro”.

Como são desenvolvidos os projetos de software na instituição

Segundo os dados coletados, 26,9% das instituições desenvolvem os seus próprios sistemas de acordo com a demanda local; 15,4% utiliza os servidores de TI para instalar, manter, configurar e dar suporte a sistemas de terceiros; 53,8% das instituições respondentes adaptam e personalizam sistemas de terceiros já existentes; e 3,8% não souberam responder (Figura 10).

De acordo com sua experiência, como é feito o desenvolvimento de projetos de software na sua Instituição?

26 respostas



Figura 10 - Pergunta sobre forma de desenvolvimento

Processo de desenvolvimento

Cinquenta e dois por cento (52%) dos respondentes afirmaram que sua instituição possui algum processo de desenvolvimento de software definido (Figura

11). O fato de apenas metade das instituições terem processo definido chama atenção, pois o acesso às informações sobre processos de software é facilitado atualmente através da Internet. Além disso, em Institutos Federais de Educação (onde, inclusive, alguns ofertam disciplinas de informática e gestão), esperava-se maior engajamento das equipes de TIC e da alta gestão para adesão a processos definidos de desenvolvimento. Neste contexto, a não utilização de um processo bem definido (por 48% dos respondentes) pode ser um fator crítico de fracasso à gestão de tempo e entregas no prazo previsto.

De acordo com sua experiência, existe algum processo definido de desenvolvimento de software na sua Instituição?
25 respostas

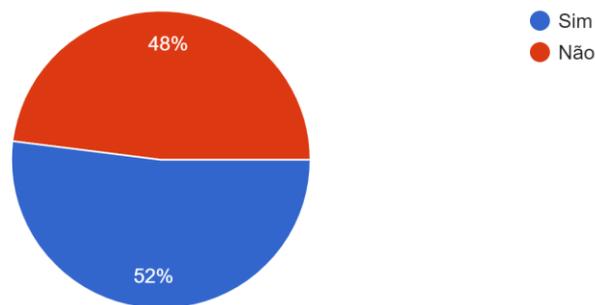


Figura 11 - Pergunta sobre a existência de um processo definido de desenvolvimento

Na literatura, Pradhan, Rishiwal e Agarwal (2019) salientam que a falta de um processo de gestão do desenvolvimento apropriado pode causar atraso na entrega do projeto. Bilgaiyan, Mishra e Das (2016) citam que o processo de desenvolvimento pode ser facilitado quando se tem uma boa gestão dos projetos e Suliman e Kadoda (2017) dizem que um processo de gestão de desenvolvimento do software é crucial para a organização, principalmente quando se equilibra escopo, tempo e custos do projeto.

Entregas no prazo previsto

Quanto ao questionamento sobre se os projetos são entregues no prazo, a coleta resultou os seguintes percentuais (Figura 12):

- 7,7% responderam que todos os projetos são entregues no prazo previsto;
- 61,5% responderam que apenas alguns projetos são entregues no prazo previsto;
- 15,4% afirmaram que nenhum projeto é entregue no prazo previsto e
- 15,4% não souberam responder.

De acordo com sua experiência, os projetos de software da instituição onde trabalha são entregues no prazo previsto?

26 respostas

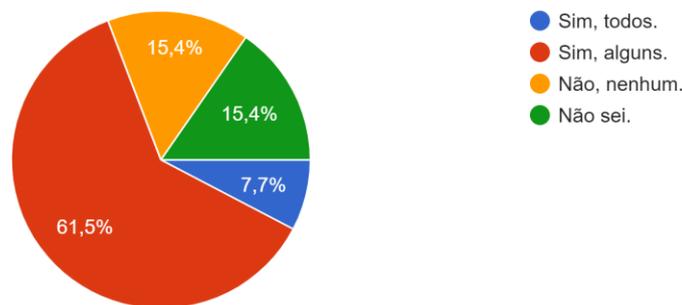


Figura 12 - Pergunta sobre entregas no prazo

Cinquenta por cento (50%) das instituições onde os projetos atrasam (Assertivas “Nenhum projeto é entregue no prazo” e “Alguns projetos são entregues no prazo”) possuem processo de desenvolvimento de software definido. Assim, pode-se pressupor que embora a utilização de um processo de desenvolvimento de software bem definido seja fator positivo à gestão do tempo, o sucesso no cumprimento dos prazos não é garantido, visto que mesmo nas instituições que têm processo bem definido existem problemas de entrega no prazo previsto.

Ao confrontarmos os resultados obtidos em campo com os obtidos na literatura, percebemos que há uma certa similaridade já que em ambos os contextos apenas uma pequena parte dos projetos são finalizados com sucesso. Por exemplo, no trabalho de Nasir e Sahibuddin (2011), destaca-se que em 1994 apenas 16% dos

projetos foram entregues no prazo previsto (*On Time*), enquanto que em 2008 este valor subiu para 32%. Já segundo o CHAOS Report (*The Standish Group, 2015*), este número foi de 36% em 2015.

Satisfação do cliente

Quando questionados sobre o impacto do atraso na satisfação do cliente (Figura 13), 88,5% dos respondentes afirmaram que a satisfação do cliente é afetada quando os projetos de software são entregues após o prazo previsto, coincidindo com o que foi levantado na QUASI-RSL, pois segundo Hussain, Mkpojiogu e Kamal (2016), falhas ou entregas após o prazo previsto nos projetos são desafios atuais e afetam a satisfação do cliente, além da imagem da empresa e a lucratividade.

De acordo com sua experiência, a satisfação do cliente (neste caso, os usuários dos sistemas) é afetada quando os projetos de software são entregues após o prazo previsto?

26 respostas

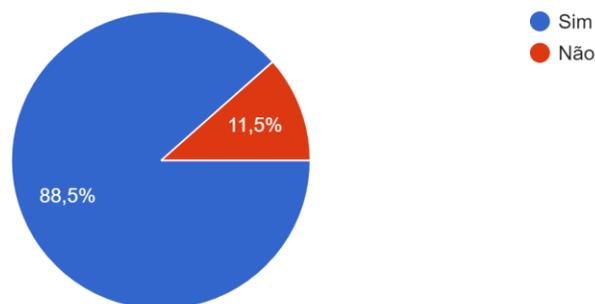


Figura 13 - Pergunta sobre satisfação do cliente

Fatores Críticos

Na questão sobre quais Fatores interferem positiva ou negativamente na entrega do prazo previsto (Figura 14)²⁷, foram obtidos os seguintes resultados:

- 61,5% para Apoio da alta gestão;
- 57,7% para Comunicação entre membros da equipe;
- 42,3% para Controle de qualidade;

²⁷ Por limitação de espaço da ferramenta utilizada para geração dos gráficos, alguns nomes de FC não são exibidos nas figuras. Por exemplo, na Figura 4.8, o nome do FC "Comunicação entre membros da equipe" não é exibido, somente a sua porcentagem.

- 73,1% para Engenharia/análise de requisitos;
- 80,8% para Envolvimento do usuário/cliente;
- 42,3% para Escopo;
- 26,9% para Estimativas e métricas;
- 15,4% para Tratamento de exceções no código-fonte;
- 61,5% para Experiência da equipe;
- 30,8% para Ferramentas utilizadas;
- 23,1% para Finanças, orçamento e preços;
- 46,2% para Gerenciamento de mudanças;
- 42,3% para Gerenciamento de riscos;
- 57,7% para Metodologia de Gestão de Projeto de Software;
- 73,1% para Motivação da equipe;
- 34,6% para Definição de objetivos e metas do projeto;
- 50% para Testes de software;
- 53,8% para Treinamento da equipe;
- 50% para Troca de membros da equipe e
- 0% para Outros.

Marque abaixo os Fatores Críticos que, na sua opinião, interferem (positiva ou negativamente) na entrega no prazo previsto dos projetos de softwar...rabalha. É possível assinalar mais que uma opção.
26 respostas

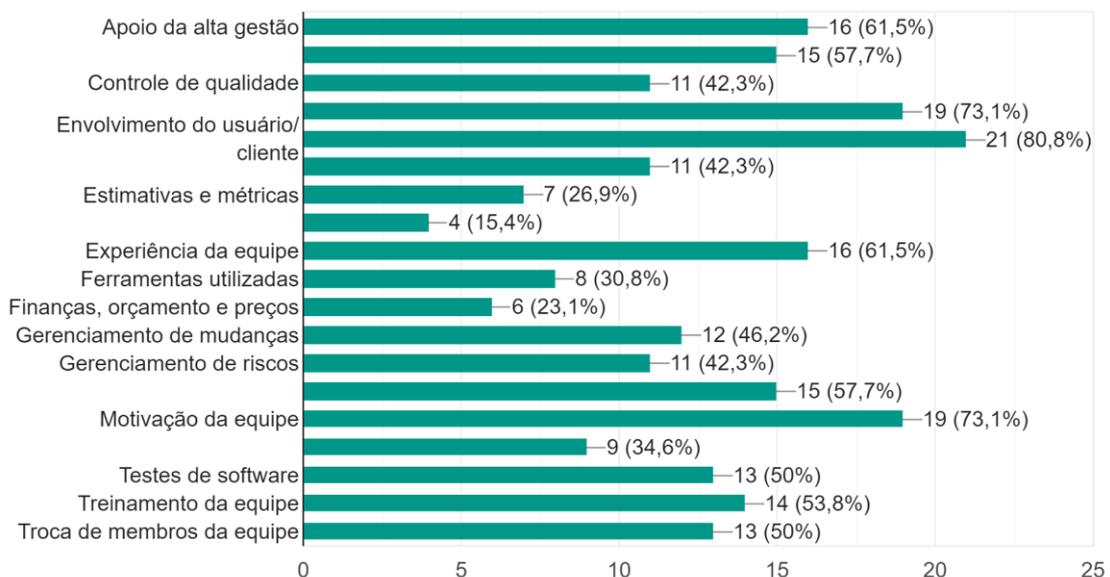


Figura 14 - Pergunta sobre os FC

Alguns FC obtiveram alta porcentagem de votos na pesquisa de campo. Como esperado, o FC “Apoio da alta gestão” teve um peso muito alto nos IFs (61,5%), dado que tratam-se de instituições públicas, contexto onde ainda há forte hierarquia verticalizada e controle rigoroso de orçamento. Na QUASI-RSL este FC também demonstrou ser relevante, visto que obteve 8 citações (25% do total de estudos).

Outros exemplos importantes são os FC “Motivação da Equipe”, “Engenharia de requisitos” e “Envolvimento do usuário/cliente”, os quais foram bastante citados (73,1%, 73,1% e 80,8%, respectivamente). Doze instituições citaram os três FC simultaneamente, representando 46% do total das respondentes, ou seja, parte significativa das instituições consultadas entendem que os três FC têm impacto nos seus projetos.

Segundo a QUASI-RSL, a “Motivação da equipe” é uma das raízes dos problema de entregas após o prazo previsto (SULIMAN e KADODA, 2017). Com relação à “Engenharia de requisitos”, Gupta, Chauhan e Dutta (2013) afirmam que quando os requisitos são mal interpretados, faltantes ou incompletos pode-se realizar as atividades do projeto incorretamente. No tocante ao “Envolvimento do usuário/cliente”, Hussain, Mkpojiogu e Kamal (2016) citam que esse FC é a causa de problemas com entregas após o prazo e Suliman e Kadoda (2017) explicam que a falta de envolvimento do usuário no projeto pode conduzir a expectativas irreais e conseqüentemente a falhas.

Por fim, cabe ressaltar a baixa representatividade do FC “Escopo”, onde apenas 42,3% dos respondentes dos IFs entendem que este é um Fator Crítico às entregas dos projetos, apesar de, segundo literatura consultada, o “Escopo” mal definido (ou mal delimitado) ser considerado um dos fatores que podem impactar no ciclo de vida do projeto de desenvolvimento.

Técnicas utilizadas

Para levantar se os Institutos Federais de Educação usam alguma técnica²⁸ e confrontar com o que foi levantado na literatura, foi feita a seguinte pergunta no questionário aplicado aos IFs: “selecione abaixo as técnicas utilizadas na instituição

²⁸ Entende-se por “técnicas”, as ferramentas, sugestões ou propostas identificadas nos artigos obtidos via Revisão Sistemática da Literatura para mitigação de FCF ou melhor uso de FCS.

onde trabalha para auxiliar na entrega de projetos de software no prazo previsto”. Foi possível ao respondente selecionar mais de uma técnica, se necessário, portanto a somatória poderia ser maior que 100%. A Figura 15 mostra o gráfico de respostas:

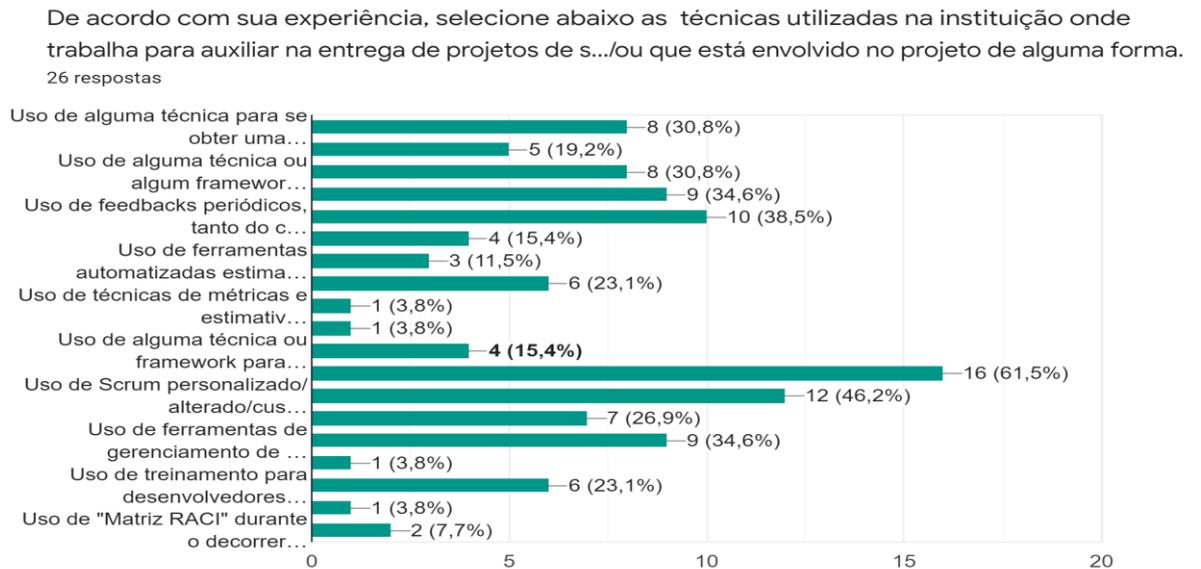


Figura 15 - Pergunta sobre as técnicas utilizadas

As técnicas citadas no questionário aplicado foram mapeadas nos FC e estão organizadas na Tabela 7, a qual exhibe para cada FC, as técnicas relacionadas e a respectiva porcentagem obtida na pesquisa de campo. Cabe ressaltar que algumas técnicas se relacionam com mais de um FC e uma das técnicas não se relacionou com nenhum FC em específico. Além do mais, alguns FC não tiveram nenhuma técnica explicitamente relacionada na pesquisa de campo (é o caso do FC002 - Comunicação da equipe, FC006 - Escopo, FC009 - Experiência da equipe, FC015 - Motivação, FC016 - Objetivos/metapas claros e bem definidos e FC019 - Troca de membros da equipe).

FC	Técnica(s) relacionadas a este FC	%
FC001 - Apoio da alta gestão	Uso de alguma técnica, como a "Teoria da Difusão da Inovação" para avaliar características de líderes de projetos	3,8%
FC003 - Controle de qualidade	Uso de alguma técnica ou framework para Gerenciamento de Riscos	15,4%
FC004 - Engenharia de requisitos	Uso de alguma técnica ou algum framework de 'Engenharia de Requisitos'	30,8%

	Uso de protótipos de software para uma melhor comunicação entre a equipe e o cliente	34,6%
	Uso de "desenvolvimento iterativo" no projeto	26,9%
FC005 - Envolvimento do usuário/cliente	Uso de metodologias ágeis de desenvolvimento de software	61,5%
FC007 - Estimativas e métricas	Uso de ferramentas automatizadas estimar mais precisamente o tempo e custo do projeto	11,5%
	Uso de comparações com dados históricos e projetos já finalizados para estimar mais precisamente o tempo e o custo do projeto	23,1%
	Uso de técnicas de métricas e estimativas, como soft computing, PSA, COCOMO, FPA, PSO ou LOC/kLOC para estimar mais precisamente o tempo e o custo do projeto	3,8%
	Uso de alguma técnica ou algum framework para estimar mais precisamente o tempo e o custo do projeto	15,4%
FC008 - Exceções	Uso de alguma técnica de "Monitoramento de Exceções (no código)" no desenvolvimento do projeto	3,8%
FC010 - Ferramentas	Uso de ferramentas de gerenciamento de projetos como o Microsoft Project ou similar	34,6%
FC011 - Finanças, orçamento e preços	Uso de feedbacks periódicos, tanto do cliente quanto da equipe, para se obter um melhor gerenciamento de riscos do projeto	38,5%
FC012 - Gerenciamento de mudanças	Uso de metodologias ágeis de desenvolvimento de software	61,5%
FC013 - Gerenciamento de riscos	Uso de alguma técnica ou algum framework de 'Engenharia de Requisitos'	30,8%
	Uso de metodologias ágeis de desenvolvimento de software	61,5%
	Uso de alguma técnica para se obter uma 'Comunicação efetiva' com o cliente	30,8%
	Uso de alguma técnica para se obter um 'Maior envolvimento do cliente' no projeto	19,2%
	Uso de feedbacks periódicos, tanto do cliente quanto da equipe, para se obter um melhor gerenciamento de riscos do projeto	38,5%
	Uso de políticas de prevenção de riscos ou estabelecimento de padrões de projeto para Gerenciamento de Riscos	3,8%
	Uso de alguma técnica ou framework para Gerenciamento de Riscos	15,4%
FC014 - Gestão de Projetos de Softwares	Uso de ferramentas de gerenciamento de projetos como o Microsoft Project ou similar	34,6%
	Uso de alguma técnica ou algum framework de 'Engenharia de Requisitos'	30,8%

	Uso de alguma técnica ou algum framework para estimar mais precisamente o tempo e o custo do projeto	15,4%
	Uso de Scrum personalizado/alterado/customizado	46,2%
	Uso de metodologias ágeis de desenvolvimento de software	61,5%
FC017 - Testes	Uso de alguma técnica ou algum framework de 'Engenharia de Requisitos'	30,8%
FC018 - Treinamento	Uso de treinamento para desenvolvedores e gerentes de projetos da instituição	23,1%
Não relacionada a nenhum FC	Uso de 'Matriz RACI' durante o decorrer do projeto	7,7%

Tabela 7 - Relação entre os FC e as Técnicas citadas no questionário

Ao analisarmos os dados coletados, nota-se que várias instituições utilizam métodos ágeis, pois 61,5% dos respondentes afirmaram fazer “Uso de metodologias ágeis de desenvolvimento de software” e 46,2% afirmaram fazer “Uso de Scrum personalizado/alterado/customizado”. Ambas são as técnicas propostas com maior relevância nos dados coletados. É indicativo positivo à gestão do software e à entrega de projetos no prazo previsto, visto que um dos pilares dos Métodos Ágeis é a entrega constante e a tempo.

O destaque no “Uso de metodologias ágeis de desenvolvimento de software” nos dados coletados dá a ideia de que as instituições têm consciência de que é necessário aplicar alguma metodologia para gerir e desenvolver softwares. O fato da utilização de metodologias ágeis é entendido como indicador de que tais instituições estão atentas às tendências atuais da indústria. Neste contexto, a utilização de Scrum personalizado como a segunda técnica mais assinalada dá a noção de que a customização de alguma metodologia já existente é uma opção viável.

Por outro lado, é preocupante que muitas técnicas são pouco usadas, pois espera-se uma maior adesão das instituições aos meios para atingir uma melhoria contínua dos projetos. É o caso, por exemplo, do “Uso de alguma técnica para se obter um Maior envolvimento do cliente no projeto”. Tal técnica, segundo QUASI-RSL, se relaciona ao FC “Gerenciamento de riscos”. Porém, enquanto este FC teve 42,3% de relevância na questão sobre “quais Fatores interferem positiva ou negativamente na entrega do prazo previsto”, a respectiva técnica teve apenas 19,2% de relevância na questão sobre “as técnicas utilizadas na instituição onde trabalha para auxiliar na entrega de projetos de software no prazo previsto”. Isso indica que embora parte das

instituições entendam que o gerenciamento de riscos seja fator crítico, poucas lançam mão de alguma técnica para trazer o cliente mais próximo ao projeto.

Com relação ao FC “Estimativas e métricas”, o qual é bastante relevante na literatura (segundo QUASI-RSL realizada nesta pesquisa), somente 26,9% das instituições entendem a sua relevância na questão sobre “quais Fatores interferem positiva ou negativamente na entrega do prazo previsto”. Esta informação vai ao encontro das porcentagens sobre as técnicas utilizadas, visto que poucas instituições utilizam algum framework para estimativas (15,4%), ferramentas automatizadas para estimativas (11,5%), consulta a dados históricos (23,1%) e técnicas de métricas e estimativas (somente 3,8%).

Vale destacar que, enquanto 80,8% dos respondentes entendem “Envolvimento do usuário/cliente” como FC, apenas 61,5% utilizam a técnica “Uso de metodologias ágeis de desenvolvimento de software” para tal.

Diante dos exemplos supracitados, entende-se que, embora parte das instituições tenham consciência do impacto dos FC nos projetos, poucas se utilizam de meios para mitigá-los (FCF) ou utilizá-los da melhor forma (FCS).

Com relação ao FC “Engenharia de Requisitos”, somente 30,8% das instituições afirmaram fazer uso de alguma técnica ou framework relacionado, enquanto 73,1% afirmaram que este é um FC relevante na questão “Quais Fatores interferem positiva ou negativamente na entrega do prazo previsto?”. Assim, embora muitas instituições considerem esse um fator importante, poucas fazem uso de alguma técnica. Talvez isto tenha relação com o fato de que a área de TIC é constantemente procurada para resolver problemas urgentes e repentinos, não restando muito tempo para que a engenharia de requisitos seja melhor conduzida.

Quanto ao FC “Motivação da equipe”, este se mostrou de grande importância na pesquisa aplicada, pois 73,1% dos respondentes afirmaram ser um FC relevante, porém segundo QUASI-RSL realizada, nenhum estudo propôs técnica específica para este FC.

No tocante à aplicação de treinamentos à equipe, 23,1% das instituições afirmaram aplicá-los, o que é um número baixo, dada a importância de se capacitar a equipe. Porém ao se comparar esta porcentagem de 23,1% apenas com a porcentagem de outra técnica qualquer (por exemplo, “Monitoramento de Exceções (no código)” que só foi citado por 3,8% de respondentes) nota-se grande diferença;

desta perspectiva pode-se concluir que os 23,1% referentes ao treinamento da equipe não é um número baixo.

Outros pontos a destacar:

- Alguns FC tiveram pouca relevância para os IFs. O FC “Finanças, orçamentos e preços” não se destacou (quando comparado com outros FC). Este ponto merece atenção visto que se presumia que, em se tratando de instituições públicas federais, onde os recursos financeiros têm a tendência de serem mais regrados, seria um FC de destaque. “Tratamento de exceções no código-fonte” também não se mostrou importante ao cumprimento de prazos, o que também ocorre na literatura, segundo revisão bibliográfica realizada. O mesmo acontece com o FC “Ferramentas utilizadas”, pois nos dois contextos houve baixa porcentagem.
- Algumas técnicas também tiveram pouca relevância nos IFs. Cabem destaque: “Uso de técnicas de métricas e estimativas, como soft computing, PSA, COCOMO, FPA, PSO ou LOC/kLOC para estimar mais precisamente o tempo e o custo do projeto”, “Uso de políticas de prevenção de riscos ou estabelecimento de padrões de projeto para Gerenciamento de Riscos” e “Uso de alguma técnica de Monitoramento de Exceções (no código) no desenvolvimento do projeto”. A baixa representatividade de tais técnicas pode implicar no aumento do impacto de fatores de fracasso nos projetos, pois elas têm impacto direto no processo de desenvolvimento de software, visto que: a) fazer uso de métricas e de estimativas auxilia a equipe a prever melhor quando as entregas serão realizadas e a detectar se há pontos fracos no processo de desenvolvimento; b) quando se gerencia riscos é possível criar formas para diminuir o impacto de acontecimentos inesperados e c) tratar exceções no código pode auxiliar o desenvolvedor a economizar tempo de codificação e impedir danos maiores ao software quando colocado em ambiente de produção.
- Quanto ao “Monitoramento de Exceções (no código) no desenvolvimento do projeto”, é compreensível a baixa representatividade, visto que o FC “Tratamento de exceções no código-fonte” se mostrou ter baixa importância, tanto nos IFs, através da pesquisa de campo, quanto na literatura, através da Revisão Sistemática da Literatura.

4.4 Considerações finais do capítulo

Finalmente, a partir das análises realizadas nesta seção, somadas às informações colhidas através da Revisão de Literatura, pode-se inferir as seguintes necessidades atuais nos IFs:

- Entende-se que a implantação de alguma metodologia de Gestão de Projetos de Software é o passo inicial àquelas instituições que ainda não utilizam. O uso deste recurso se mostrou fator crítico à gestão do tempo segundo revisão bibliográfica e a maioria das instituições também entendem ser um FC aos seus projetos, apesar da metade dos respondentes afirmarem não fazerem uso de um processo definido;
- É necessário elaborar meios para que a alta gestão da organização compreenda a utilidade e a importância do setor de TIC para que possa apoiá-lo. O FC “Apoio da alta gestão” se mostrou relevante segundo pesquisa realizada. Na QUASI-RSL uma técnica relacionada a esse FC foi levantada, como “Uso de alguma técnica, como a ‘Teoria da Difusão da Inovação’ para avaliar características de líderes de projetos;
- Outra necessidade se dá na atenção dada ao cliente do projeto. Mantê-lo próximo às atividades é essencial, conforme já citado. Comunicação constante e reuniões periódicas auxiliam para que isto ocorra. No entanto, poucas instituições utilizam alguma forma de manter o cliente próximo ao projeto. Caso a instituição passe a utilizar métodos ágeis, por exemplo, já é implícito em seus princípios ter o cliente como parte fundamental do projeto. Ainda em relação ao cliente, entende-se que a engenharia de requisitos dos projetos destas instituições deve ser melhor planejada e executada;
- Outro ponto importante também é a elaboração de uma forma de medir e estimar melhor as fases do projeto de software, utilizando-se eventualmente de ferramentas específicas para este fim, visto que segundo revisão de literatura é um fator importante no auxílio à gestão do tempo e à entrega no prazo, e segundo pesquisa de campo 26,9% dos respondentes consideram este FC relevante, embora poucos utilizem alguma técnica para isso (3,8% somente);
- O estabelecimento de Gerenciamento de Riscos também se mostrou importante para ser utilizado nas instituições, principalmente em instituições públicas, onde

há escassez de recursos em determinadas épocas do ano, ou seja, há incertezas;

- Em se tratando da equipe, grande parte das instituições mostraram preocupação quanto à “Motivação da equipe”, ao “Treinamento da equipe” e à “Experiência da equipe” (73,1%, 61,5% e 53,8% das respostas quanto aos FC, respectivamente). Entende-se que a criação de programas de treinamento e especialização para a equipe (desenvolvedores, analistas, técnicos, gerentes etc.) pode auxiliar nestes três fatores.

No próximo capítulo é apresentada uma proposta de conjunto de Boas Práticas para equipes de desenvolvimento de software dos IFs, criado a partir dos FC e das técnicas oriundos tanto da revisão da literatura como da pesquisa de campo.

5 UM CONJUNTO DE BOAS PRÁTICAS PARA EQUIPES DE DESENVOLVIMENTO DE SOFTWARE DOS INSTITUTOS FEDERAIS DE EDUCAÇÃO

5.1 Introdução

Este capítulo descreve o processo de criação de um conjunto de Boas Práticas, o qual foi embasado a partir da QUASI-RSL (Revisão Sistemática da Literatura) e da revisão *ad hoc*, bem como reunindo informações colhidas através da pesquisa de campo realizada neste trabalho. O conjunto sintetiza as práticas para desenvolvimento e gestão de projetos de software mais adequadas às equipes de TIC dos Institutos Federais de Educação.

Almeida et al. (2019) propuseram algo semelhante: um guia interativo de boas práticas focado em engenharia de software. Os autores destacam que o guia poderá facilitar o acesso, a identificação e a aplicação adequada de tais práticas em diversos cenários de projetos de software. O presente trabalho se difere daquela pesquisa por estar voltado especificamente aos Institutos Federais de Educação.

Optou-se por dividir o conjunto de Boas Práticas de acordo com a seguinte estrutura:

- Capítulo 1: os Institutos Federais (IFs) são brevemente apresentados para que se compreenda o contexto onde se aplica o conjunto;
- Capítulo 2: contextualiza o leitor no tocante às equipes de TIC dos IF;
- Capítulo 3: divide os Fatores Críticos (FC) em dois tipos para facilitar a leitura, entendimento e aplicação posterior;
- Capítulo 4: apresenta os FC encontrados, alocando-os em seu respectivo tipo como descrito no Capítulo 3;
- Capítulo 5: apresenta as Boas Práticas propostas, as quais se relacionam aos FC apresentados no Capítulo 4. Após análise, organização e agrupamento das propostas identificadas nesse estudo, foram documentadas sete boas práticas, as quais sintetizam as técnicas levantadas na pesquisa (ver **seção 5.3**). A estrutura de apresentação das práticas é uma adaptação do modelo de padrões de projetos proposto por Gamma et al. (2007). Após a adaptação, a estrutura de cada Boa Prática ficou assim organizada (mais informações na **seção 5.4**):

Nome da Boa Prática, Objetivo, Também conhecida como, FC relacionado(s), Consequências, Boa(s) Prática(s) Relacionada(s) e Dica.

Os FC foram classificados nos seguintes tipos para facilitar a leitura e o entendimento: Tipo P, para os FC relacionados às Pessoas; Tipo M, para os FC relacionados aos Métodos. Em seguida, os FC foram agrupados em *clusters* de acordo com seu Tipo.

A partir dos FC fez-se uma relação destes com as técnicas propostas na QUASI-RSL. As informações selecionadas foram reunidas em um documento na internet no endereço <https://cin.ufpe.br/~lbm5/conjunto/Apresentacao.html>.

5.2 A seleção dos FC e técnicas relacionadas de maior impacto à gestão de tempo dos IFs

Inicialmente foram selecionados os FC com maior relevância na pesquisa de campo realizada nos IFs. Entende-se por maior relevância aqueles FC que foram citados pela maioria dos respondentes (50% ou mais). Portanto, neste trabalho, os FC com menos de 50% de citações foram considerados “irrelevantes” e os FC com 50% ou mais de citações foram considerados “relevantes”. Considerou-se que deveria haver predominância da pesquisa de campo sobre a QUASI-RSL na seleção dos FC e técnicas relacionadas que iriam compor o conjunto, pois a pesquisa de campo abordou um escopo mais específico aos objetivos desta pesquisa. Neto (2017) fez algo semelhante em seu trabalho sobre FC para implantação de Central de Serviços de TI, classificando os FC de acordo com seu grau de importância (muito importante, importante, indiferente, pouca importância, sem importância).

Desta forma, os FC selecionados, mapeados no seu percentual e número de citações na pesquisa de campo, bem como a quantidade de técnicas propostas associadas a estes FC, são apresentados na Tabela 8 (organizados de acordo com o percentual de citações).

FC	% de citações	# Citações	# Técnicas
-----------	----------------------	-------------------	-------------------

FC005 - Envolvimento do usuário/cliente	80,8%	7	2
FC004 - Engenharia/análise de requisitos	73,1%	10	3
FC015 - Motivação da equipe	73,1%	1	0
FC001 - Apoio da alta gestão	61,5%	8	1
FC009 - Experiência da equipe	61,5%	3	0
FC002 - Comunicação entre membros da equipe	57,7%	6	0
FC014 - Metodologia de Gestão de Projetos de Softwares	57,7%	12	6
FC018 - Treinamento da equipe	53,8%	1	1
FC017 - Testes de software	50,0%	2	1
FC019 - Troca de membros da equipe	50,0%	1	0

Tabela 8 - Relação entre os FC selecionados no Passo 1, percentual de citações na pesquisa de campo, número de citações na QUASI-RSL e técnicas propostas

5.3 As técnicas relacionadas aos FC selecionados

Com o objetivo de generalização e abrangência de tudo o que foi coletado até então (técnicas, propostas, boas práticas, sugestões e outros), no contexto do conjunto de Boas Práticas, as técnicas propostas pelos autores foram renomeadas para “Boas Práticas”.

As boas práticas foram levantadas a partir da QUASI-RSL²⁹. Desta forma, a partir dos FC selecionados na **seção 5.2**, selecionou-se as boas práticas relacionadas a estes FC. Embora todas as boas práticas possam ser minimamente relacionadas a todos os FC, este trabalho optou por manter a relação extraída da QUASI-RSL.

²⁹ A pesquisa de campo não retornou nenhuma técnica diferente das advindas da QUASI-RSL (elencadas no questionário).

No conjunto de Boas Práticas, para cada FC tem-se a descrição das Boas Práticas oriundas da QUASI-RSL, descrita em um maior nível de detalhe, bem como a indicação do estudo no qual esta foi embasada.

Outro ponto relevante é que nem todos os estudos propõem técnicas de maneira detalhada. Em alguns exemplos, os autores apenas fazem menção à técnica e especificam os resultados de um experimento utilizando-a. Mesmo nessas situações, justifica-se a referência à técnica para servir de ponto inicial de discussão para as equipes de TIC.

Alguns Fatores Críticos selecionados não possuem Boas Práticas relacionadas³⁰, porém mesmo assim foram inseridos no documento de Boas Práticas para auxiliarem as equipes na detecção de irregularidades em seu processo de desenvolvimento de software.

FC001 - Apoio da alta gestão

Boa(s) Prática(s) relacionada(s):

- Uso de alguma técnica, como a "Teoria da Difusão da Inovação" para avaliar características de líderes de projetos: Govindaraju, Hariadi e Sidiq (2015) propõe examinar a influência das características dos líderes no sucesso dos projetos, utilizando um modelo conceitual desenvolvido baseado na *DOI - Diffusion Innovation Theory* (Teoria da Difusão da Inovação), e conclui que a capacidade de inovação da alta gestão pode influenciar no sucesso dos projetos.

FC004 - Engenharia/análise de requisitos

Boa(s) Prática(s) relacionada(s):

- Uso de alguma técnica ou algum framework de 'Engenharia de Requisitos': Gupta, Chauhan e Dutta (2013) propõem um novo método de priorização de requisitos a fim de se focar primeiramente nos essenciais ao usuário e que necessitem de uma menor carga de testes de regressão. Jones (2004) apresenta várias sugestões de técnicas: um quadro de requisitos controlados pelo cliente e pela equipe, revisões formais de todas as mudanças de requisitos, uso da

³⁰ FC015 - Motivação da equipe, FC009 - Experiência da equipe, FC002 - Comunicação entre membros da equipe e FC019 - Troca de membros da equipe.

técnica JAD³¹ entre outras. Embora com pouco detalhamento, Hussain, Mkpojiogu e Kamal (2016) frisam a importância de uma Engenharia de Requisitos bem feita e seu impacto no sucesso do projeto, sugerindo que ela seja executada em todas as fases do projeto. Shahzad e Mathkour (2009) são mais específicos e sugerem a técnica FAST para diminuir a distância entre o cliente e a equipe de desenvolvimento. Já Nazir et al. (2014) propõem uma técnica de verificação e validação de requisitos baseada na ontologia;

- Uso de protótipos de software para uma melhor comunicação entre a equipe e o cliente: Jones (2004) defende o uso de protótipos formais a fim de minimizar o impacto de mudanças de requisitos;
- Uso de "desenvolvimento iterativo" no projeto: Jones (2004) propõe que o desenvolvimento iterativo contribui para minimizar o impacto das mudanças repentinas de requisitos.

FC005 - Envolvimento do usuário/cliente

Boa(s) Prática(s) relacionada(s):

- Uso de metodologias ágeis de desenvolvimento de software: Raunak e Binkley (2017) afirmam que a adoção de metodologias ágeis de desenvolvimento de software, como Scrum ou XP ajuda a conduzir discussões mais detalhadas com o cliente sobre suas expectativas e prioridades, devido a uma interação mais frequente e mais próxima. Ahimbisibwe e Cavana (2016) destacam a relação positiva e significativa entre a participação do usuário, uso de métodos ágeis e o sucesso dos projetos.

FC014 - Metodologia de Gestão de Projetos de Softwares

Boa(s) Prática(s) relacionada(s):

- Uso de ferramentas de gerenciamento de projetos como o Microsoft Project, Artemis Views e Primavera: Jones (2004) defende o uso de ferramentas deste tipo por serem formas automatizadas de usar várias técnicas para controlar grande sistemas. Alguns exemplos de técnicas associadas a este tipo de

³¹ *Joint Application Design* - metodologia de definição de requisitos criada pela IBM em 1977.

ferramentas incluem PERT³², caminho crítico, Gantt³³ etc. Em Pradhan, Rishiwal e Agarwal (2019) destaca-se especificamente o Microsoft Project, como uma ferramenta que pode contribuir para todo o ambiente de desenvolvimento de software, reduzindo o tempo de entrega de todas as operações;

- Uso de alguma técnica ou algum framework de 'Engenharia de Requisitos': Jones (2004) foca nas mudanças de requisitos, as quais podem contribuir para uma melhor gestão do projeto de software como um todo. Desta forma, sugere usar solicitações formais de mudanças de requisitos, ferramentas especializadas para controle de mudanças e revisões formais das mudanças solicitadas a fim de não atrapalhar as entregas do produto;
- Uso de alguma técnica ou algum framework para estimar mais precisamente o tempo e o custo do projeto: em Govindarajan (2015) destaca-se o uso de um framework para guiar os gerentes de projetos no processo de tomada de decisão;
- Uso de metodologias ágeis de desenvolvimento de software: Ahimbisibwe e Cavana (2016) sugerem o uso de metodologias ágeis de desenvolvimento, pois estas respondem mais rapidamente aos aspectos dinâmicos do projeto, entre outras vantagens citadas;
- Uso de Scrum personalizado/alterado/customizado: Hong, Yoo e Cha (2010) propõem a utilização do Scrum customizado, o qual se mostrou eficiente para diminuir a taxa de defeitos nos projetos.

FC017 - Testes de software

Boa(s) Prática(s) relacionada(s):

- Uso de alguma técnica ou algum framework de 'Engenharia de Requisitos': Gupta, Chauhan e Dutta (2013) sugerem uma técnica para priorizar os requisitos mais importantes e que tenham baixa contagem de regressão, a fim de diminuir o esforço necessário nos testes.

³² *Program Evaluation and Review Technique* - programa de avaliação e revisão de técnicas (tradução do autor).

³³ Diagrama usado para ilustrar o avanço das etapas de um projeto.

FC018 - Treinamento da equipe

Boa(s) Prática(s) relacionada(s):

- Uso (oferta) de treinamento para desenvolvedores e gerentes de projetos da instituição: Suliman e Kadoda (2017) destacam que treinar desenvolvedores e gerentes de projetos iniciantes a explorarem erros passados, discrepâncias, imprecisões, fatores e métodos de estimativas, bem como formas de aplicar um sistema de análise de custos e orçamento, auxiliam a detectar riscos nos projetos.

Nota-se que algumas boas práticas estão relacionadas a vários FC, tais como o “Uso de alguma técnica ou framework de Engenharia de Requisitos”, que relaciona-se aos FC FC004 - Engenharia/análise de requisitos, FC014 - Metodologia de Gestão de Projetos de Softwares e FC017 - Testes de software.

Para efeito de padronização, generalização dos termos e para facilitar a leitura do conjunto de Boas Práticas, alguns títulos das boas práticas foram adaptados, tais como “uso de técnica ou framework ...” que foi renomeada para “uso de técnica...” com o objetivo de resumir a sentença, além de minimizar dúvidas quanto ao termo “framework”.

Adicionalmente, práticas que possuem relação entre si foram unificadas. O processo de escolha dos nomes das Boas Práticas está descrito a seguir:

- A boa prática “Uso de alguma técnica ou framework de Engenharia de Requisitos” tornou-se “**Usar técnica de Engenharia de Requisitos**”, pois entende-se que a palavra “técnica” é mais genérica, além de evitar que o título ficasse mais extenso;
- A boa prática “Uso de alguma técnica, como a Teoria da Difusão da Inovação para avaliar características de líderes de projetos” tornou-se “**Avaliar líderes**”, para que seu nome não fique demasiadamente extenso;
- A boa prática “Uso de ferramentas de gerenciamento de projetos como o Microsoft Project ou similar” tornou-se “**Usar ferramenta de gerenciamento de projetos**” para deixar a escolha da ferramenta genérica;
- A boa prática “Uso de metodologias ágeis de desenvolvimento de software” se juntou a “Uso de Scrum personalizado/alterado/customizado” e a “Uso de

desenvolvimento iterativo no projeto” e tornou-se “**Usar princípios ágeis**”, pois essas boas práticas possuem relação entre si (por conta do manifesto ágil);

- A boa prática “Uso de treinamento para desenvolvedores e gerentes de projetos da instituição” tornou-se “**Oferecer capacitação à equipe**”, pois em Institutos Federais de Educação utiliza-se mais o termo “capacitação” que “treinamento”, sendo a equipe formada tanto por desenvolvedores como por gerentes;
- A boa prática “Uso de alguma técnica ou algum framework para estimar mais precisamente o tempo e o custo do projeto” tornou-se “**Usar técnicas de estimativas**”, pois “técnicas” pode abranger várias formas de estimar tempo e custo;
- A boa prática “Uso de protótipos de software para uma melhor comunicação entre a equipe e o cliente” tornou-se “**Usar protótipos**”, a fim de resumir o nome da Boa Prática sem perder seu sentido original.

Assim sendo, tem-se as seguintes boas práticas, as quais formam o conteúdo principal do conjunto de Boas Práticas proposto neste trabalho. A Tabela 9 exibe as relações entre FC e boas práticas:

- #01 - Usar técnica de Engenharia de Requisitos;
- #02 - Avaliar líderes;
- #03 - Usar ferramenta de gerenciamento de projetos;
- #04 - Usar princípios ágeis;
- #05 - Oferecer capacitação à equipe;
- #06 - Usar técnicas de estimativas;
- #07 - Usar protótipos.

Boa prática / FC	FC001 - Apoio da alta gestão	FC004 - Engenharia/ análise de requisitos	FC005 - Envolvimento do usuário/cliente	FC014 - Metodologia de Gestão de Projetos de Softwares	FC017 - Testes de software	FC018 - Treinamento da equipe
#01 - Usar técnica de Engenharia de Requisitos;		X		X	X	
#02 - Avaliar líderes;	X					
#03 - Usar ferramenta de gerenciamento de				X		

projetos;						
#04 - Usar princípios ágeis;	X	X	X	X		
#05 - Oferecer capacitação à equipe;	X					X
#06 - Usar técnicas de estimativas;				X		
#07 - Usar protótipos.		X				

Tabela 9 - Relação entre Boas Práticas e FC

Os títulos dos FC também foram adaptados para o conjunto de Boas Práticas. A adaptação justifica-se por alguns motivos: tentou-se resumir para facilitar a leitura, tentou-se reduzir ambiguidades e tentou-se aplicar expressões mais utilizadas nos Institutos Federais de Educação. Além disso, os títulos foram adaptados para dar um sentido negativo (por exemplo, “Apoio da alta gestão” tornou-se “Falta de apoio organizacional”) com o objetivo de fazer jus à relação entre o FC e a Boa Prática, a qual pode ser utilizada para mitigação do FC. Mesmo com o processo de adaptação, acredita-se que o sentido principal foi preservado. A seguir tem-se o resultado das adaptações:

- FC001 - Apoio da alta gestão tornou-se **Falta de apoio organizacional**;
- FC004 - Engenharia/análise de requisitos tornou-se **Engenharia de requisitos mal feita**;
- FC005 - Envolvimento do usuário/cliente tornou-se **Pouco envolvimento do usuário**;
- FC014 - Metodologia de Gestão de Projetos de Softwares tornou-se **Falta de metodologia de Gestão de Projetos de Software**;
- FC017 - Testes de software tornou-se **Falta de testes de software**;
- FC018 - Treinamento da equipe tornou-se **Falta de treinamento da equipe**.

Conforme citado anteriormente, de acordo com a estrutura proposta baseada em padrões de projetos, para cada uma das Boas Práticas tem-se uma descrição detalhada da mesma, como explicado na próxima seção.

5.4 A estrutura do conjunto de Boas Práticas

O conjunto de Boas Práticas, proposto com base na pesquisa e na análise realizadas, consiste de um documento³⁴ disponível na internet, no endereço <https://www.cin.ufpe.br/~lbm5/conjunto/Apresentacao.html>. O conjunto reúne sete boas práticas para auxiliar as equipes de desenvolvimento a entregarem os produtos no tempo previsto, escritas em uma linguagem clara e objetiva, abstraindo termos muito específicos para que possa ser acessível para os envolvidos no projeto de desenvolvimento de software.

Buscou-se concentrar a atenção nos FC de maior impacto levantados na pesquisa de campo e nas respectivas técnicas propostas com o objetivo de não tornar o conjunto muito extenso a ponto de confundir ou desmotivar as equipes de desenvolvimento. A finalidade foi estabelecer um ponto inicial para as discussões de melhoria na gestão de tempo dos projetos de software, além de introduzir os leitores a técnicas propostas na literatura e identificadas na QUASI-RSL.

A seção do conjunto que descreve os principais FC encontrados está estruturada da seguinte maneira: **nome do FC** (adaptado de “nome do padrão”, como é popularmente conhecido o FC); **Tipo** (se é do tipo M, P ou ambos); **descrição** (adaptado de “objetivo” do padrão, onde se descreve brevemente o FC); **também conhecido como** (outras formas de referenciar o mesmo FC); e **consequências** (um resumo dos principais impactos do FC). Outras seções propostas pelos autores de padrões de projeto se mostraram irrelevantes para a descrição dos FC. Desta forma, os FC selecionados serão descritos no conjunto de Boas Práticas conforme apresentado na Figura 16:

³⁴ A ferramenta utilizada para desenvolver o conjunto chama-se *HelpNDoc Personal Edition 6.8.0.521*

Pouco envolvimento do usuário

Nome do FC

Pouco envolvimento do usuário.

Tipo

M e P

Descrição

De acordo com o Manifesto Ágil, o cliente deve estar presente em todo o ciclo de vida do projeto para ajudar na condução dos requisitos e diminuir o impacto de mudanças de direcionamento.

Também conhecido como

Usuário distante do projeto, falta de comunicação intensa com o cliente.

Consequências

- Falta de feedback dos clientes em relação às funcionalidades entregues;
- Desinteresse do cliente;

Figura 16 - Exemplo de descrição de FC

A seção que descreve as boas práticas está organizada da seguinte forma, para cada boa prática: **nome da boa prática** (serve para referenciar a boa prática em um alto nível de abstração); **objetivo** (uma curta descrição da boa prática); **também conhecida como** (outros nomes que referenciam a boa prática); **FC relacionados** (advém da QUASI-RSL e descreve o contexto e as situações em que a boa prática é aplicada); **consequências** (análises das vantagens e desvantagens da aplicação da boa prática); **boas práticas relacionadas** (descreve quais as boas práticas que têm relação com a descrita) e, em alguns casos, **Dicas** (alguma sugestão do autor para o leitor), sendo descritas no conjunto de Boas Práticas conforme apresentado na Figura 17:

BOA PRÁTICA #02 - AVALIAR LÍDERES

Nome da boa prática

Avaliar líderes

Objetivo

Escolher e avaliar os líderes de equipes e gerentes de projetos pode ser uma tarefa dispendiosa caso a organização não possua uma forma organizada para isso. Estas funções servem tanto para direcionar as equipes quanto para motivá-las, portanto há a necessidade de se fazer boas escolhas das lideranças do grupo.

Adicionalmente, nada impede de se fazer um rodízio de lideranças da equipe para aumentar a motivação e coesão de todos.

Para aplicação, sugere-se analisar a sinergia e a evolução da equipe no decorrer de certo período de tempo e verificar se há muitos conflitos entre os membros é um passo que pode ser dado no início. Em seguida, sugere-se aplicação de metodologias, como a Teoria da Difusão da Inovação para se avaliar as lideranças dos grupos.

Também conhecida como

Avaliação dos CEO's.

FC relacionado(s)

Falta de apoio organizacional.

Consequências

Fazer avaliação periódica dos líderes pode dar noção ao gestor do desempenho dos profissionais envolvidos, ajudando na tomada de decisão (troca de comando, por exemplo).

Boa(s) Prática(s) relacionada(s)

BOA PRÁTICA #05 - OFERECER CAPACITAÇÃO À EQUIPE

Dica: pesquise mais a fundo sobre a Teoria da Difusão da Inovação, de Everett Rogers, na internet, e leia a subseção correspondente.

Figura 17 - Exemplo de descrição de boa prática

Além dessa estrutura, cada seção de Boa Prática possui uma ou mais subseções com mais detalhes sobre as propostas encontradas (Figura 18). Em algumas dessas subseções há exemplos práticos de aplicação e há sugestões para aplicação da prática no âmbito dos IFs. As subseções são divididas de acordo com as informações encontradas na QUASI-RSL e em todas há o tópico **Autor(es)** para se fazer a devida referência. A nomenclatura das subseções está de acordo com o nome

da técnica/proposta/boa prática criada pelo autor, caso esta possua nome definido (por exemplo a subseção chamada **FAST**) ou algum nome genérico que possa facilitar a identificação (por exemplo a subseção **Lista de ferramentas**).

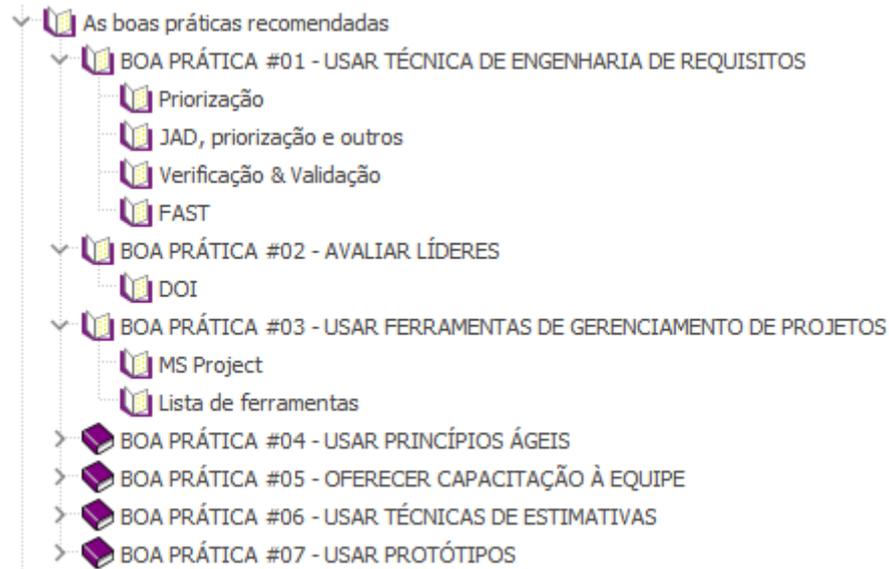


Figura 18 - Boas Práticas e algumas subseções

Para Gamma et al. (2007), a utilização de uma estrutura baseada em “padrões de projetos” permite descrever problemas e suas soluções de forma que possam ser usados diversas vezes. Os autores destacam, ainda, que essa utilização pode facilitar o entendimento pelo usuário final, pois pode-se “descrever formas de se resolver um problema geral de projeto em um contexto particular”.

É importante ressaltar que as boas práticas não podem ser tratadas isoladamente, pois possuem relação umas com as outras (esta informação é destacada no conjunto no item **Boas práticas relacionadas**). Como exemplo, a boa prática **#01 - Usar técnicas de Engenharia de Requisitos** pode possuir relação com a boa prática **#04 - Usar princípios ágeis**, dependendo da situação.

5.5 Considerações finais do capítulo

Este capítulo apresentou a estrutura, o formato e o conteúdo do principal produto desse trabalho, o conjunto de Boas Práticas para Desenvolvimento de

Software, o qual descreve sete Boas Práticas a serem utilizadas pelos Institutos Federais de Educação (IFs). Estas práticas foram obtidas a partir de trabalhos com foco na otimização da entrega no prazo de projetos de software, identificados a partir da execução de uma QUASI-RSL, bem como através de uma pesquisa de campo realizada com os IFs.

Para avaliação do conjunto de Boas Práticas, aplicou-se uma sessão de Grupo Focal com funcionários da Fábrica de Software do IFTM. Detalhes da realização do Grupo Focal são apresentados no próximo capítulo. O conjunto de Boas Práticas na íntegra encontra-se no **APÊNDICE D**.

6 AVALIAÇÃO DO CONJUNTO DE BOAS PRÁTICAS

6.1 Introdução

Esta seção detalha a avaliação do conjunto de Boas Práticas, que ocorreu por intermédio da técnica Grupo Focal (GF) com voluntários lotados na Fábrica de Software do IFTM. A aplicação da técnica foi baseada na obra de Zaganelli et al. (2015) e na obra de Shull, Singer e Sjoberg (2008), que tratam do uso do Grupo Focal na Ciência da Informação e na Engenharia de Software, respectivamente.

As vantagens, desvantagens e os resultados da aplicação do Grupo Focal também são descritos neste capítulo.

O protocolo utilizado para avaliação do conjunto pode ser visualizado no **APÊNDICE C**.

6.2 Grupo Focal

6.2.1 *O que é um Grupo Focal*

De acordo com Kontio, Bragge e Lehtola (2008), a técnica Grupo Focal é um método de pesquisa qualitativo, cuidadosamente planejado para obter percepções pessoais de um grupo de indivíduos em uma determinada área de pesquisa. Geralmente reúnem-se entre 3 e 12 participantes para uma discussão guiada pelo pesquisador-moderador, que segue uma estrutura de perguntas pré-definidas. Para Zaganelli et al. (2015), a técnica carrega uma característica muito peculiar em relação a outros métodos: capta a sinergia e a interação do grupo em relação ao tema ou assunto. Os autores, inclusive, defendem que a técnica é muito utilizada, pois “nos últimos 50 anos, o Scopus registrou mais de 40 mil publicações com o uso parcial ou total do grupo focal. Só no Brasil, foram mais de 500 publicações.”. Destacam ainda que a técnica pode ser aplicada tanto online quanto presencialmente.

Para Merton e Kendall (1948, apud Zaganelli et al., 2015), as principais funções da técnica são descobrir os aspectos significativos da situação em que as respostas ocorreram, encontrar as discrepâncias entre os efeitos esperados e os reais, descobrir

as respostas desviantes dos subgrupos na população e encontrar os principais processos envolvidos.

Para este trabalho utilizou-se a técnica de Grupo Focal aplicada na Fábrica de Software do IFTM com o objetivo de levantar as percepções do grupo sobre o conjunto de Boas Práticas proposto, além de debater de maneira informal pontos de melhoria para tal proposta. É o que propõem Fraser e Restrepo-estrada (2005), citados por Chiara (2005, apud Zaganelli et al., 2015), onde sugerem a criação de “uma situação informal no qual os tópicos são discutidos pelos participantes de forma livre e espontânea.”. Detalhes sobre o perfil do participante, número de participantes, passo a passo realizado entre outros podem ser obtidos no **APÊNDICE C**.

6.2.2 Vantagens do Grupo Focal

Entende-se que a aplicação do Grupo Focal neste trabalho é viável pois a técnica apresenta vantagens significativas. Segundo Chiara (2005, apud Zaganelli et al., 2015), a técnica:

- É uma maneira efetiva e relativamente fácil de obter dados a um baixo custo;
- Propicia uma oportunidade para que os indivíduos possam formar opinião sobre o tema estudado por meio da interação com outras pessoas;
- Possibilita a espontaneidade nas discussões e a troca de ideias entre os participantes;
- Oferece aos pesquisadores detalhes que não seriam facilmente obtidos por meio de outras técnicas, como por exemplo, a opinião de um grupo de interessados sobre o documento;
- Permite avaliação tanto de serviços já implantados quanto o planejamento de novos serviços;
- Possibilita o planejamento de novas pesquisas, através do surgimento de novas idéias, discussões e pontos de vista diferentes.

Kontio, Bragge e Lehtola (2008) também destacam os pontos positivos da utilização do Grupo Focal:

- Descoberta de novas intuições, através de troca de informações sobre a experiência dos outros participantes;
- Confirmação de ideias, visto que em vários estudos os pontos testemunhados por alguns participantes foram confirmados por outros participantes, o que não poderia ocorrer em entrevistas pessoais;
- Custo baixo e eficiência;
- Profundidade da técnica, dada a possibilidade de explorar as razões pelas quais os participantes pensam de determinada maneira, através de debates e questionamentos.

6.2.3 Desvantagens do Grupo Focal

A técnica de Grupo Focal, embora tenha muitos pontos favoráveis, também tem desvantagens. Beck e Manuel (2008, apud Zaganelli et al., 2015) e Chiara (2005, apud Zaganelli et al., 2015), por exemplo, citam:

- O fato de os participantes poderem fornecer informações falsas;
- O fato de os participantes desviarem do tópico em discussão;
- O fato de os participantes serem influenciados pela pressão de outros participantes;
- O fato de os participantes buscarem consenso mais que explorar ideias, por pressa para encerrar a dinâmica ou por desinteresse;
- Pode ocorrer o domínio da palavra por um participante, tomando o tempo de outro participante que poderia dar outras sugestões;
- Pode haver participantes mais tímidos para exporem suas ideias, por vergonha ou medo da exposição. Desta forma é possível que nem todas as ideias sejam absorvidas pelo pesquisador.

Da mesma maneira, Kontio, Bragge e Lehtola (2008) pontuam as principais fraquezas da técnica:

- A dinâmica do grupo focal e o fluxo da conversa podem não ser controlados pelo moderador, pois é possível haver debates sobre assuntos que fogem do objeto principal da dinâmica;

- A chance de se destacar entre os demais pode influenciar, por exemplo, que algum participante ofereça informações falsas durante as discussões. Além disso, alguns participantes podem ser mais tímidos que outros, prejudicando sua participação;
- Algumas informações podem ser escondidas devido a acordos empresariais;
- O tempo do Grupo Focal pode ser curto para que os participantes possam compreender profundamente as ideias expostas.

6.2.4 Justificativa para a utilização do Grupo Focal

A aplicação da técnica de Grupo Focal mostrou-se viável neste trabalho devido a vários fatores:

- Facilidade e baixo custo em reunir os participantes, visto que o Grupo Focal foi aplicado com membros das equipes de TIC da Fábrica de Software do IFTM, local de trabalho do moderador-pesquisador;
- A necessidade de avaliar a proposta do conjunto de Boas Práticas de forma qualitativa e com participantes interessados nesta solução;
- O fato de os participantes serem partes interessadas no contexto da pesquisa;
- Praticidade em exibir o conjunto de Boas Práticas a um grande número de pessoas, dando ainda a possibilidade de debate para surgimento de melhorias na proposta;
- A falta de autonomia do pesquisador e potencial dificuldade em poder aplicar alguma Boa Prática (ou todas as Boas Práticas) na Fábrica de Software do IFTM, pelo fato de não possuir cargo de direção ou função de confiança e devido à mudança de gestão e consequente mudança de chefia, coincidente com as datas de execução desta pesquisa;
- Baixa disponibilidade das equipes para entrevistas pessoais ou experimentos práticos, devido a licenças, férias, afastamentos ou acúmulo de atividades;
- O estado de calamidade pública em que o país se encontra na época da aplicação desta técnica, devido à pandemia COVID-19, o que forçou o uso da técnica Grupo Focal de forma online.

O autor Zaganelli et al. (2015) destaca que “quanto mais ideias surgirem, melhor. O objetivo é fazer com que os participantes discutam os tópicos e falem livremente sobre suas opiniões, comportamentos ou crenças em relação a um serviço, produto ou assunto, numa situação informal na qual os mesmos não se sintam pressionados.”

Não obstante as vantagens e as motivações da aplicação de um Grupo Focal no contexto desta pesquisa, cabe destacar suas desvantagens específicas ao contexto no qual foi aplicado:

- A principal desvantagem na utilização desta técnica nesta pesquisa se deu pelo viés dos participantes, visto que estes têm relação profissional com o moderador-pesquisador. Este é um ponto negativo previsto por Kontio, Bragge e Lehtola (2008), os quais sugerem medidas paliativas, por exemplo enfatizar a importância de que as informações sejam abertas e verdadeiras e garantir o anonimato e a confidencialidade dos resultados. Tais medidas foram utilizadas na aplicação desse Grupo Focal. Outro ponto importante a se destacar para diminuir o viés da aplicação desta técnica nesta pesquisa é o fato de que o moderador-pesquisador não possui cargo de chefia na instituição, o que isenta um possível viés por influência hierárquica;
- Para que a duração da dinâmica não ocupasse grande parte do dia de trabalho dos participantes, propôs-se iniciar os trabalhos um pouco antes do horário de encerramento do expediente e estendê-los por alguns minutos além do horário de encerramento do expediente, em acordo com a chefia responsável;
- Para que a dinâmica do Grupo Focal não se desviasse dos objetivos, o moderador-pesquisador elaborou um roteiro para auxiliar na condução dos trabalhos.

Desta forma, não obstante os pontos negativos apresentados, entendeu-se que a aplicação do Grupo Focal foi uma técnica satisfatória e adequada para os objetivos desta pesquisa, pois com ela a proposta do conjunto de Boas Práticas pôde ser apresentada e debatida com profissionais da área.

O Grupo Focal teve participação de 6 servidores lotados na Fábrica de Software do IFTM, divididos entre as equipes de “Desenvolvimento de sistemas”, “Inovação e desenvolvimento” e “Governança e segurança” e o grupo possuía 5 participantes homens e 1 mulher. As atividades exercidas pelos participantes incluíam: coleta e análise de requisitos, criação do banco de dados, codificação dos módulos, teste dos

produtos entregues, correção de erros, validação de funcionalidades e relacionamento com os clientes.

6.2.5 Abordagem utilizada

A abordagem utilizada para condução do Grupo Focal foi baseada na obra de Chiara (2005, apud Zaganelli et al., 2015). Neste estudo o autor propõe a divisão do GF em três fases:

Planejamento

Onde se formulam as questões-chave para condução do grupo focal, além de se definir o local, data e quantidade de participantes.

Nesta pesquisa, as questões-chave foram baseadas nos FC e nas boas práticas que compõem o conjunto de Boas Práticas, ou seja, o objetivo foi constatar a ocorrência dos FC e colocar em debate a possível mitigação destes através das boas práticas.

Condução

O moderador conduz e intermedia os participantes em uma discussão guiada pelas questões elaboradas no planejamento, tomando cuidado para que o foco não seja desviado e que todos os presentes participem.

No tocante a esta pesquisa, a condução se deu da seguinte forma: apresentação do conjunto de Boas Práticas aos participantes e considerações iniciais sobre o roteiro que será seguido, apresentação dos FC aos participantes, questionamentos e debates sobre os FC, apresentação das boas práticas aos participantes, questionamentos e debates sobre as boas práticas, coleta de sugestões dos participantes e conclusão.

Em relação aos FC, estes foram apresentados aos participantes e, em seguida, perguntas foram feitas para fomentar o debate: “Este FC ocorre no seu ambiente de trabalho?”, “Com qual frequência este FC ocorre?”, “Qual é o impacto deste FC ao projeto?” são alguns exemplos de perguntas.

Para as boas práticas, algo semelhante foi realizado. As boas práticas levantadas foram expostas, uma a uma, e sugeriu-se o debate, a troca de informações e principalmente sugestões de melhoria. Algumas perguntas feitas foram: “Esta boa

prática auxiliaria os projetos de software da sua equipe?” e “O que pode ser alterado nesta boa prática para que ela se encaixe melhor no contexto dos projetos da sua equipe?”, por exemplo.

Análise

Oliveira e Freitas (1998) apud Chiara (2005) afirmam que nesta fase é necessário considerar as palavras e seus significados, contexto, consistência, frequência e extensão das colocações e especificidades das respostas.

Para este trabalho, esta fase foi necessária para constatar a existência ou não dos FC levantados através da QUASI-RSL e da pesquisa de campo em um contexto específico, ou seja, uma amostra dos IFs, no caso no IFTM.

Posteriormente foi debatido com os participantes se as boas práticas sugeridas tinham potencial para auxiliar a resolver os FC apresentados. O foco foi compreender se a aplicação das boas práticas seria dispendiosa e trabalhosa a ponto de inviabilizar seu uso. Outro ponto-chave desta fase foi constatar se as boas práticas sugeridas poderiam ter impacto positivo ou negativo na condução dos projetos de software no contexto do IFTM.

O processo de análise teve como embasamento a técnica de confirmação de hipóteses chamada Triangulação (Jick, 1979, apud Seaman, 2008), quando se reúne vários tipos de evidências para dar suporte à uma proposição. Tais evidências podem ser originadas de várias fontes, por vários métodos diferentes (neste caso, a revisão de literatura, a pesquisa de campo e o Grupo Focal).

Portanto, a fase da análise foi importante para compreender a relação entre o que foi pesquisado até então e sua aplicação em um contexto específico.

Acredita-se que a realização de apenas uma sessão de Grupo Focal foi suficiente para coletar as impressões do grupo, visto que a quantidade de participantes que atuam na Fábrica de Software do IFTM e estavam disponíveis para participação foi seis, número sugerido na obra de Zaganelli et al. (2015).

Um ponto importante a se registrar é que nesta pesquisa não foi utilizado o personagem observador, que originalmente auxilia o moderador no registro do conteúdo das discussões e expressões corporais. Desta forma, o moderador-pesquisador acumulou as duas funções. A utilização de um observador não se fez

essencial para a condução da dinâmica visto que esta foi gravada para referência futura. Além disso, todas as discussões foram devidamente registradas.

6.2.6 Utilização do Grupo Focal online

Devido à situação de emergência global na época da aplicação do Grupo Focal, devido à pandemia do Coronavírus, a técnica foi aplicada de forma online. Zaganelli et al. (2015) sustentam que a aplicação da técnica não precisa ser presencial. Adicionalmente, Chase e Alvarez salientam que “o grupo focal online vem ganhando espaço no mercado, inclusive, devido à redução de custos, fácil implementação e coleta rápida de dados” (2000, apud Zaganelli et al., 2015).

A aplicação da técnica de Grupo Focal online compartilha das mesmas vantagens e mesmas desvantagens da forma presencial, mas pode-se destacar ainda:

- Como desvantagens: a conexão do participante pode oscilar ou até ser interrompida e não há meios de se observar a linguagem corporal dos participantes, caso não possuam câmera (Chase e Alvarez, 2000, apud Zaganelli et al., 2015);
- Vantagens: custo menor, pelo fato de tanto o participante quanto o pesquisador não terem que se deslocar da sua residência; praticidade pois evita a burocracia do local de trabalho; e conforto oriundo do fato de que os envolvidos na técnica estão em local onde ficam mais à vontade que no ambiente profissional.

6.3 Resultados obtidos

Essa seção detalha a condução do Grupo Focal para a avaliação do conjunto de Boas Práticas e os resultados obtidos na avaliação. Primeiramente foram avaliados os FC e, em seguida, as Boas Práticas.

6.3.1 Avaliação dos FC

Falta de apoio organizacional

Os participantes do Grupo Focal afirmaram que a alta gestão é um fator crítico no contexto da organização onde trabalham, pois é através dela que são definidas as

prioridades dos projetos, ou seja, é a alta gestão (no caso dos IFs, a Reitoria e Pró-reitorias) que prioriza um projeto em detrimento de outro. Além disso, frisaram que a maioria das demandas não partem da área de TIC e sim de outros setores, caracterizando novamente a influência da alta gestão nos projetos.

Outro ponto importante é o fato de a priorização dos projetos mudar constantemente devido a novas políticas que surgem, e isso compromete o seu planejamento e priorização.

Em alguns casos, porém, caso o apoio seja exagerado pode causar várias interrupções ao projeto (situações onde o usuário sobrecarrega as equipes com dúvidas, novas demandas ou troca de requisitos frequentes, por exemplo).

Ressaltaram que além do apoio da alta gestão, também é importante o apoio que as equipes recebem dentro dos próprios setores de TIC.

Os participantes continuaram fazendo uma referência a outro fator crítico, “Pouco envolvimento do usuário”, que se relaciona com o fator “Falta de apoio organizacional” quando o usuário (solicitante do software) não dá o suporte necessário aos desenvolvedores (tirando dúvidas, esclarecendo os passos das tarefas que estão sendo implementadas, por exemplo), podendo causar entregas após o prazo previsto e perda da qualidade do projeto.

Citaram também o apoio motivacional que a alta gestão pode dar ao influenciar a comunidade a mudar culturas e paradigmas para a utilização de um sistema.

Em se tratando do setor privado, os participantes entendem que este fator é menos crítico, pois o principal produto das organizações é o software, logo o comprometimento da alta gestão tende a ser maior, visto que o objetivo é o lucro.

Engenharia de requisitos mal feita

Os participantes do Grupo Focal consideraram este fator crítico aos projetos de desenvolvimento de software de sua organização. Citaram que em muitos casos os requisitos são coletados a partir de uma só pessoa, geralmente a chefia de algum setor. Isso pode ser uma desvantagem, pois caso esta pessoa mude de função os requisitos devem ser revistos e retrabalhados.

Ressaltaram que a natureza política de organizações públicas é relevante, pois a mudança de pessoas que geralmente tomam decisões quanto aos requisitos é constante.

Como sugestão propuseram a elaboração de uma forma de prever futuras mudanças de requisitos, deixando claro aos usuários que os requisitos podem ser alterados no futuro por outra pessoa.

Destacaram que, infelizmente, no âmbito de sua instituição a avaliação da coleta dos requisitos é feita somente após o término do projeto. A sugestão é que esta avaliação seja feita de forma contínua e em todas as fases dos projetos.

Observaram que sua instituição não possui um processo, um passo a passo a ser seguido para se coletar requisitos e entendem que de fato isso seja importante para melhoria dos projetos.

Fazendo relação com o fator “Falta de Metodologia de Gestão de Projetos de Software”, os participantes consideraram que a implantação de um método ágil (a Fábrica de Software do IFTM havia implantado recentemente a metodologia ágil Scrum) eliminou o intermediário (equipe que coletava e analisava os requisitos antes de repassar aos desenvolvedores), fazendo com que as equipes de desenvolvimento entendam melhor os problemas, necessidades e contexto do cliente, por estarem mais próximas deste.

Pouco envolvimento do usuário

Os participantes do Grupo Focal citaram que os projetos que têm pouco envolvimento dos usuários são aqueles que desenvolvem softwares de “prateleira”³⁵, pois em geral possuem poucas adaptações solicitadas pelos usuários e são desenvolvidos para cumprirem um objetivo comum e genérico (por exemplo, editores de texto ou conversores de vídeos).

Em projetos ágeis, como no caso da instituição dos participantes do Grupo Focal, o envolvimento do usuário é crítico, principalmente em relação aos requisitos. Com o usuário envolvido frequentemente no projeto, a maioria dos problemas de requisitos são mitigados, visto que as interações são constantes, o que diminui o impacto no fator “Engenharia de requisitos mal feita”.

Como exemplos de problemas relacionados ao baixo envolvimento do usuário, os participantes citaram o caso de projetos onde houveram *sprints* finalizadas sem

³⁵“Software de prateleira” é um termo popular para se referir aos softwares que são desenvolvidos para o público em geral e geralmente são vendidos prontos sem a possibilidade de adaptação para um usuário.

entregas incrementais e sem feedbacks do usuário, o que causou problemas principalmente no tocante à motivação da equipe, pois os desenvolvedores se viram sem incentivo para continuar a codificação e os testes dos módulos do projeto, por não existir feedback e avaliação por parte dos usuários. Neste caso em questão, o envolvimento foi somente por parte da equipe de desenvolvimento e não do usuário.

Outro ponto que reforça a importância do envolvimento do usuário é o fato de que em alguns casos foram solicitadas mudanças drásticas em projetos que já estavam em produção, o que poderia não ocorrer caso o cliente estivesse próximo à equipe de desenvolvimento em todas as fases do projeto, solicitando as alterações necessárias antes de o projeto ser finalizado.

Um fator positivo no IFTM, segundo um dos participantes, é o envolvimento dos servidores dos campi no processo de validação do software e nas reuniões de *sprints*, pois esses usuários têm maior interesse, visto que serão auxiliados pelo software em desenvolvimento. Ressaltaram que quanto mais pessoas, maior é o envolvimento e mais feedbacks são obtidos, tendendo a aumentar a qualidade do produto.

Finalizaram destacando que sem um feedback do usuário é comum um projeto ter retrabalho, causando atraso.

Falta de metodologia de Gestão de Projetos de Software

Os participantes do Grupo Focal frisaram que embora a falta de metodologia de gestão de projetos seja um problema, a utilização de alguma metodologia sem que a mesma seja monitorada e melhorada também é um obstáculo. É necessário ter autonomia e flexibilidade para adaptar a metodologia ao contexto da organização (aqui faz-se referência à Boa Prática #04 - Usar princípios ágeis, mais precisamente à subseção “Scrum personalizado”). Tal adaptação pode ser alcançada com discussão entre os colegas e aprimoramento no decorrer do tempo.

Afirmaram também que o uso de alguma metodologia ajuda principalmente no planejamento do projeto, pois torna-se mais fácil de se avaliar o que é eficaz ou não.

Outro participante destacou que sem o uso de uma metodologia, a gestão torna-se caótica. Acredita que tal situação não é produtiva, principalmente pela falta de direcionamento e protocolo. Salienta também que em equipes numerosas, o problema pode ser ainda maior, pela dificuldade em se alinhar objetivos e expectativas.

Falta de testes de software

A falta de testes torna difícil identificar erros no projeto antes de eles acontecerem em produção e por consequência a credibilidade do setor/sistema pode cair.

Os participantes do Grupo Focal consideraram que os testes de software são um fator muito importante, muito válido e crítico para atender os prazos dos projetos. Afirmaram que, a curto prazo, podem até causar atraso ao projeto, mas a médio e longo prazo são válidos, pois diminuem o impacto de alterações em códigos que podem afetar outras partes do software. Um participante reforçou a sugestão de se utilizarem testes automatizados.

Salientaram que na instituição onde trabalham os testes de usuários são essenciais e estão ligados aos fatores “Pouco envolvimento do usuário” e “Falta de apoio organizacional”.

Falta de treinamento da equipe

Os participantes do Grupo Focal não acreditam que este seja um fator tão importante quanto os outros, pois dependendo do projeto, talvez não haja necessidade de se aprender algo novo. Em contrapartida, destacaram que o fato de uma pessoa desconhecer o jeito certo de se executar uma tarefa pode causar atraso e perda de qualidade. Consideram também que aprender uma coisa nova pode ser um fator motivacional.

Entre os fatores apresentados até então, entendem este ser o menos crítico, pois o projeto não irá ser cancelado ou atrasado exclusivamente por esse fator. A pessoa pode se adaptar, aprender e evoluir por si só, sem depender de um treinamento ou capacitação. O impacto causado pode ser quanto à qualidade e ao prazo.

Um dos participantes destacou que o nivelamento é tão importante quanto o treinamento. Nivelar significa fazer que o conhecimento técnico seja difundido por toda a equipe, evitando que algum membro saiba mais ou menos que outro. A falta de nivelamento de conhecimento entre os membros da equipe pode causar problemas. É importante ressaltar que o “nivelamento” é um fator levantado no Grupo Focal e não veio da QUASI-RSL deste trabalho.

Considerações gerais sobre a avaliação dos Fatores Críticos

Um fator crítico que não foi citado no conjunto de Boas Práticas é, segundo os participantes, a “Boa interação entre os membros da equipe”. Este é um fator importante, principalmente no serviço público, pois é ideal que o ambiente de trabalho seja um local agradável e o bom relacionamento faz com que os cronogramas fiquem alinhados e facilita a manutenção dos sistemas.

Os participantes do Grupo Focal informaram ainda que os fatores que têm maior impacto na gestão do tempo são: o “Pouco envolvimento do usuário”, pois impacta em vários outros pontos-chave do ciclo de vida do projeto; e a “Falta de apoio organizacional”, pois em organizações públicas este fator se relaciona com vários outros e influencia a priorização dos projetos.

6.3.2 Avaliação das Boas Práticas

#01 - Usar técnica de Engenharia de Requisitos

Os participantes do Grupo Focal debateram e entenderam que a utilização de uma técnica de engenharia de requisitos pode de fato ajudar, pois caso não exista um protocolo a ser seguido, a coleta de requisitos se dá de forma desorganizada, o que prejudica o andamento do projeto.

Salientaram ainda que o fato de se aplicar uma técnica, como a *FAST*, em outras instituições, sejam públicas ou privadas, pode trazer benefícios ao andamento do projeto de software.

#02 - Avaliar líderes

Os participantes do Grupo Focal, após debate, chegaram à conclusão que este tipo de prática no âmbito de uma organização pública pode não ser eficaz, pois em muitos casos os líderes ocupam cargos por indicação e a avaliação destes profissionais pode não surtir o efeito desejado, que é entender a capacidade de o profissional adquirir novos conhecimentos, motivar a equipe e buscar novas soluções (inovação).

Eles consideraram que, caso houvesse uma estrutura de avaliação periódica, poderia ser vantajoso se os líderes pudessem realmente melhorar alguma conduta

após a prática. Sugeriram que a avaliação dos líderes pode ser incorporada à metodologia de gestão do projeto de software.

Por fim, outro participante entende que seria interessante a avaliação dos líderes, pois este, em muitos casos, tem papel de “líder técnico” (aquele que auxilia a equipe em dificuldades encontradas no decorrer das tarefas do projeto) e motivacional (influencia positivamente os membros da equipe). Há ainda casos onde alguns servidores se motivam a melhorarem alguma conduta após receberem uma avaliação.

#03 - Usar ferramenta de gerenciamento de projetos

Os participantes do Grupo Focal destacaram como essencial a utilização de uma ferramenta para gestão dos projetos de software da instituição, considerando uma Boa Prática viável a ser aplicada. Um dos presentes citou a relevância do acompanhamento das tarefas das equipes, utilizando, por exemplo um quadro KanBan³⁶.

Outro participante citou que o uso de uma ferramenta é essencial, pois cria-se muita informação no decorrer dos projetos, como anotações de dúvidas e atas de reuniões. Sem ferramentas o registro de tais dados seria dificultado.

Outro ponto de destaque é o fato de que a instituição onde trabalham utiliza um quadro KanBan físico na parede das salas das equipes como réplica do que se encontra no quadro KanBan virtual da ferramenta que utilizam. Isso é um fator positivo, pois todos os profissionais podem acompanhar o andamento de todos os projetos. Salientaram que esta ação foi benéfica ao cotidiano da DTIC.

Por outro lado, os participantes disseram que o uso de uma ferramenta não pode se tornar “engessado”, visto que em alguns casos algumas equipes utilizam mais o *Google Drive* que a ferramenta de gestão adotada (*Tuleap*), por ser mais fácil e já ser conhecido por todos.

Destacaram também que é importante que a ferramenta de gestão seja usada de maneira correta, para documentar reuniões, erros e acertos das equipes, de modo a permitir a consulta posterior. Inclusive deram a opção, devido à natureza pública da instituição, de se incluir os projetos de software no processo eletrônico (para que os projetos de software se tornem processos documentados e públicos, como qualquer

³⁶ Artefato visual que auxilia as equipes a conduzirem as tarefas que devem ser realizadas em determinado projeto. Segundo Wazlawick (2019), é um modelo de gerenciamento de produção originado em 1940 e é usado para representar os estados das atividades com o processo atual.

outro projeto de outros setores), embora uma vasta documentação (grande quantidade de diagramas, estórias de usuário, anotações sobre requisitos etc.) não seja característica inerente ao método ágil.

Sugeriram, ainda, a utilização de alguma ferramenta para gerenciamento das reuniões de *sprints* (tanto de coleta de requisitos quanto de entrega de incrementos). Citaram como exemplo o uso do *Google Calendar* (para agendamento de reuniões) e do *Microsoft Office 365* (para documentação).

Finalizaram destacando que não há uma ferramenta única que seja capaz de gerenciar todas as atividades. O mais indicado é encontrar uma solução para cada objetivo específico.

#04 - Usar princípios ágeis

Os participantes do Grupo Focal entendem que essa Boa Prática é essencial no contexto da sua organização. A DTIC foi bastante beneficiada pelo uso de um Método Ágil, pois trouxe o usuário mais próximo ao desenvolvimento do projeto, facilitando o gerenciamento dos requisitos e de mudanças.

Citaram também que talvez em outras empresas o uso de um Método Ágil não seria o mais indicado, pois depende muito do contexto e das necessidades organizacionais.

Destacaram, ainda, que para alcançar resultados ainda melhores pode-se adaptar o método ágil às necessidades atuais da instituição, fazendo referência à técnica “Scrum personalizado”.

Finalizaram citando que, para instituições que não têm o foco no desenvolvimento de software, o uso de um Método Ágil não seja o ideal, destacando como exemplo organizações onde a área de TIC é responsável apenas por suporte aos sistemas e não faria sentido o uso de metodologias ágeis de desenvolvimento.

#05 - Oferecer capacitação à equipe

Os participantes do Grupo Focal entendem que oferecer capacitação às equipes, embora seja muito importante, não é um fator crítico ao andamento do software. Destacaram, novamente, que mais importante que a capacitação, é o nivelamento de conhecimento entre as equipes. Sugeriram workshops ou reuniões semanais para se discutir, aprender e difundir o conhecimento entre todos.

Outro ponto importante sugerido é a auto-capacitação entre os membros das equipes, ou seja, uma troca informal de informações e experiências entre os membros, com o objetivo de resolver um problema específico surgido no decorrer do projeto.

Finalizaram citando que muitas das vezes a capacitação não ocorre devido à falta de recursos financeiros.

#06 - Usar técnicas de estimativas

Os participantes destacaram que o uso de estimativas e métricas é um ponto importante para a condução do projeto e citaram que em seu ambiente de trabalho não há metodologia de estimativa (o que ocorre frequentemente é o prazo para entrega de projetos serem impostos pela alta gestão ou pelos clientes).

Um dos participantes citou o engenheiro de software Roger Pressman (2016), o qual afirma que as métricas de software devem ser individuais: com o próprio histórico de tarefas da pessoa cria-se sua métrica, e de acordo com a métrica se prevê o tempo necessário para a execução.

Citaram ainda que o desenvolvimento de software é uma tarefa criativa, sendo difícil estimar precisamente.

Continuaram salientando que o uso do *planning poker*³⁷ é válido, pois gera discussão sobre a complexidade de cada tarefa.

Além disso, afirmaram ser difícil estabelecer e padronizar estimativas e métricas precisas na instituição onde trabalham, pois cada projeto tem uma natureza, complexidade e objetivos completamente diferentes dos outros.

#07 - Usar protótipos

Os participantes do Grupo Focal, após debate, chegaram à conclusão que o uso de protótipos não é essencial para seu contexto atual, visto que o prazo para se construir um protótipo é similar ao prazo para se entregar um incremento de software utilizável ao final de uma sprint, portanto seria uma tarefa que não agregaria valor ao projeto. Outro motivo é o fato de que sua instituição segue um padrão de desenvolvimento, com telas bem similares umas às outras.

³⁷ Dinâmica de grupo que facilita a estimativa de tempo das tarefas com cartas de baralho.

Sugeriram a utilização de rascunhos em papel (*mockup*), pois seria mais rápido e poderia ser alterado com facilidade durante as reuniões de *sprints*.

Em contrapartida, um dos participantes destacou a importância do uso dos protótipos, visto que em alguns casos faz-se necessário apresentar ao cliente uma prévia da funcionalidade (principalmente quando esta não utiliza o padrão de desenvolvimento da Fábrica de Software) para que este valide.

Por fim, sugeriram que o uso de protótipos se torne uma tarefa que seja feita apenas se necessário, sem se tornar uma regra.

Considerações gerais sobre a avaliação das Boas Práticas

Entre as Boas Práticas apresentadas, os participantes citaram que as que têm maior impacto na gestão do tempo são **#04 - Usar princípios ágeis**, pois é um conceito geral que abrange várias outras sub atividades que facilitam a condução do projeto; e **#01 - Usar técnica de Engenharia de Requisitos**, pois quanto melhor os requisitos são levantados, mais rapidamente o projeto pode ser desenvolvido. Citaram que a utilização de ambas as técnicas em conjunto também seria um fator positivo.

Por fim, um participante chamou atenção à Boa Prática **#05 - Oferecer capacitação à equipe**. A depender do projeto, se este for muito técnico e demandar conhecimentos específicos, a capacitação torna-se ponto-chave.

Perguntados sobre quais boas práticas não teriam aplicação essencial aos projetos do IFTM, citaram novamente a boa prática **#07 - Uso de protótipos** e a boa prática **#02 - Avaliar líderes**.

6.4 Considerações finais do capítulo

Este capítulo descreveu o conceito, as vantagens, desvantagens e resultados da aplicação do Grupo Focal para a avaliação do conjunto de Boas Práticas proposto neste trabalho. O protocolo utilizado na condução do Grupo Focal pode ser encontrado no **APÊNDICE C**.

Após a execução do Grupo Focal e a coleta das impressões dos participantes, pode-se chegar a algumas conclusões, as quais agregam valor aos objetivos desta

pesquisa. Algumas sugestões propostas pelos participantes do Grupo Focal foram adaptadas ao conjunto de Boas Práticas e serão pontuados nos próximos parágrafos. Os FC e as Boas Práticas que são pouco relevantes ao IFTM (segundo o Grupo Focal) não foram suprimidos do conjunto de Boas Práticas por entender-se que tais tópicos podem ser interessantes a outras instituições.

Através do uso da triangulação (comparação dos dados obtidos através da QUASI-RSL, pesquisa de campo e Grupo Focal realizados nesta pesquisa), pode-se constatar a ocorrência de alguns dos fatores críticos mencionados no conjunto no âmbito da organização avaliada. Além disso, parte das Boas Práticas propostas se mostraram aplicáveis à instituição. As principais impressões coletadas no Grupo Focal são sumarizadas a seguir:

- 1) Tão importante quanto o apoio organizacional é o fator “Boa interação entre os membros da equipe”. Tal fator não foi levantado durante a QUASI-RSL e refere-se ao bom convívio e troca de informações e experiências entre as equipes que compõem a DTIC do IFTM. Este ponto se relaciona com a boa prática **#05 - Oferecer capacitação à equipe**, onde os participantes sugeriram a auto-capacitação (ver item 19, sobre a Boa Prática **#05 - Oferecer capacitação à equipe**). O fator “Falta de apoio organizacional” se relaciona também com o fator “Pouco envolvimento do usuário”.
- 2) A instituição avaliada não possui protocolo de coleta de requisitos, o que é um fator crítico ao fracasso dos projetos. Além disso, o fator “Engenharia de Requisitos mal feita” possui relação com o fator “Falta de Metodologia de Gestão de Projetos de Software”, pois ao utilizar-se métodos ágeis (no caso do IFTM, o *Scrum*) torna-se mais fácil a coleta de requisitos visto que a comunicação com o cliente é mais frequente. A relação entre estes fatores foi inserida no conjunto de Boas Práticas;
- 3) Em vários pontos ficou evidente que a natureza política das instituições públicas e a falta de recursos financeiros interfere diretamente no transcorrer do projeto de software;
- 4) Uma sugestão para a boa prática **#01 - Usar técnica de engenharia de requisitos** é elaborar meios de evitar ou minimizar mudanças drásticas de requisitos quando da mudança de pessoal (troca de chefias, substituição de responsáveis pela criação do sistema etc.);
- 5) Outra sugestão para a boa prática **#01 - Usar técnica de engenharia de requisitos** é que a forma de coletar e analisar os requisitos seja avaliada durante

todo o ciclo de vida do projeto, com o objetivo de identificação de pontos fortes e pontos fracos. Desta forma é possível corrigir incorreções antes da entrega final do produto, onde o retrabalho pode ser maior caso tenha sido verificada alguma irregularidade;

- 6) O fator “Pouco envolvimento do usuário” se relaciona com o fator “Engenharia de requisitos mal feita”, principalmente pelo fato de a instituição avaliada possuir usuários interessados em vários campi;
- 7) Uma sugestão para a boa prática **#04 - Usar princípios ágeis** (e com relação ao fator “Falta de metodologia de Gestão de Projetos de Software”) é a melhoria contínua da metodologia utilizada. Para isso, pode-se aplicar avaliações periódicas aos envolvidos com o objetivo de entender pontos fracos e aspectos que podem ser evoluídos;
- 8) O fator “Falta de testes de software” possui relação com os fatores “Pouco envolvimento do usuário” (pois a partir do uso da metodologia Scrum, os clientes foram orientados a testarem continuamente os incrementos entregues pelas equipes a fim de encontrar falhas antes do término do projeto) e “Falta de apoio organizacional” (para que os testes sejam feitos pelos clientes de forma contínua é preciso que a alta gestão esteja ciente e de acordo, pois tais tarefas podem demandar tempo do servidor);
- 9) A boa prática **#05 - Oferecer capacitação à equipe** e o fator “Falta de treinamento da equipe” não se mostraram ter tanta importância para a entrega dos projetos de software no prazo previsto quando comparados aos outros fatores e às outras boas práticas propostas;
- 10) Tão importante quanto a capacitação é o **nivelamento** do conhecimento das equipes;
- 11) Foi constatado que a boa prática **#01 - Usar técnica de Engenharia de Requisitos** pode ter impacto significativo na entrega dos projetos de software no prazo previsto. Os participantes do Grupo Focal frisaram que com uma forma sistematizada de coletar requisitos é possível obter mais informações sobre o software a ser desenvolvido, e portanto pode-se concluí-lo mais rapidamente, pois mal-entendidos e retrabalho seriam minimizados;
- 12) Em instituições públicas, a boa prática **#02 - Avaliar líderes** pode não ser eficaz, a não ser que seja constatado que as lideranças possam melhorar suas condutas após a avaliação;

- 13) Uma sugestão para a Boa Prática **#02 - Avaliar líderes** é incorporar as avaliações formalmente na metodologia utilizada para a gestão dos projetos, ou seja, programar avaliações periódicas ao final de cada projeto entregue, por exemplo;
- 14) A boa prática **#03 - Usar ferramenta de gerenciamento de projetos** se mostrou adequada para a entrega nos prazos previstos e melhoria da qualidade do software, visto que auxilia a tornar mais prático o gerenciamento das fases do projeto, além de se poder registrar as informações coletadas nas reuniões, os requisitos e o progresso do projeto. Ainda sobre essa boa prática, foi sugerida a utilização de quadros KanBan, principalmente se este for “físico”, ou seja, colocado nas paredes das salas das equipes;
- 15) Outras sugestões para a boa prática **#03 - Usar ferramenta de gerenciamento de projetos** são: que os projetos de software sejam incluídos formalmente nos processos eletrônicos da instituição, que se passe a utilizar ferramentas para a documentação e gestão das reuniões de *sprint* e que seja criada uma função de secretaria para a organização das tarefas burocráticas das equipes;
- 16) Constatou-se que a boa prática **#04 - Usar princípios ágeis** é essencial para a instituição avaliada, principalmente no tocante à utilização do “Scrum personalizado”, visto que a partir da experiência dos participantes do Grupo Focal, o desenvolvimento dos projetos de software se torna melhor gerido, mais rápido e mais bem avaliado pelos clientes;
- 17) A aplicação da boa prática **#04 - Usar princípios ágeis** pode não ser eficaz quando a instituição não desenvolve software e apenas fornece suporte a sistemas, pois métodos ágeis são, geralmente, focados no desenvolvimento de software;
- 18) A boa prática **#05 - Oferecer capacitação à equipe** não foi vista como essencial na condução dos projetos de software e teve como sugestão de melhoria a adição do “nivelamento de conhecimentos”. Além disso, outra sugestão foi a auto-capacitação;
- 19) A boa prática **#06 - Usar técnicas de estimativas** mostrou-se importante e teve como sugestão a utilização de métricas individuais;
- 20) Quanto à boa prática **#07 - Usar protótipos**, esta não foi considerada essencial à gestão de tempo dos projetos e foi sugerida a utilização de *mockups*³⁸;

³⁸ Desenho da tela do sistema para discussão entre os interessados.

- 21) As boas práticas **#01 - Usar técnica de Engenharia de Requisitos**, **#04 - Usar princípios ágeis** e **#05 - Oferecer capacitação à equipe** (quando necessário) foram consideradas as mais importantes para a gestão de tempo dos projetos de software da instituição avaliada;
- 22) Em contrapartida, as boas práticas **"#07 - Uso de protótipos"** e **"#02 - Avaliar líderes"** não se mostraram essenciais para entrega dos projetos no prazo previsto. A primeiro pelo fato de que protótipos demoraram a ser desenvolvidos, causando desperdício de tempo, pois existem soluções mais fáceis (*Mockups*) e a último por entender-se que a aplicação não seria adequada em organizações de natureza pública, onde os líderes geralmente são indicados pela alta gestão.

A Tabela 10 resume a relação entre os FC e as Boas Práticas propostas no conjunto e o que foi coletado no Grupo Focal. A coluna "Tipo" informa se o item é "Fator Crítico" ou "Boa Prática", a coluna "Descrição" lista cada um dos itens existentes no conjunto de Boas Práticas, a coluna "Validado(a)" informa se o item foi validado pelo Grupo Focal, ou seja, se existe a ocorrência ou o potencial impacto na Fábrica de Software do IFTM segundo os participantes e a coluna "Ajustes feitos no conjunto" sintetiza as principais alterações feitas no produto após a realização do Grupo Focal.

Tipo	Descrição	Validado(a)?	Ajustes feitos no conjunto
Fatores Críticos	Falta de apoio organizacional	Sim	Novo item adicionado: "Boa interação entre os membros da equipe"
	Engenharia de requisitos mal feita	Sim	<i>Sem ajustes</i>
	Pouco envolvimento do usuário	Sim	<i>Sem ajustes</i>
	Falta de metodologia de Gestão de Projetos de Software	Sim	<i>Sem ajustes</i>
	Falta de testes de software	Sim	<i>Sem ajustes</i>
	Falta de treinamento da equipe	Sim, com ressalvas	Pouca influência quando comparado a outros fatores
Boas Práticas	#01 - Usar técnica de Engenharia de Requisitos	Sim	Nova proposta adicionada: "Evitar/minimizar mudanças de requisitos devido mudança de pessoal"

			Nova proposta adicionada: "Avaliar a coleta de requisitos em todo o ciclo de vida do projeto"
	#02 - Avaliar líderes	Não	Não se mostrou eficaz para entrega no prazo previsto
			Nova proposta adicionada: "Incorporar avaliações dos líderes à metodologia utilizada para gerir o projeto"
	#03 - Usar ferramenta de gerenciamento de projetos	Sim	Nova proposta adicionada: "Utilização de quadro Kanban físico"
			Nova proposta adicionada: "Documentar os projetos de software como processos"
			Nova proposta adicionada: "Utilização de ferramentas para documentação e gestão das reuniões de sprint"
	#04 - Usar princípios ágeis	Sim	Nova proposta adicionada: "Avaliar a metodologia utilizada"
	#05 - Oferecer capacitação à equipe	Sim, com ressalvas	Nova proposta adicionada: "Auto-capacitação"
			Nova proposta adicionada: "Nivelamento do conhecimento"
			Pouca influência quando comparada a outras Boas Práticas
	#06 - Usar técnicas de estimativas	Sim	Nova proposta adicionada: "Métricas individuais"
	#07 - Usar protótipos	Sim, com ressalvas	Nova proposta adicionada: "Utilização de Mockups"
			Pouca influência quando comparada a outras Boas Práticas

Tabela 10 - Validações e ajustes feitos no conjunto após Grupo Focal

7 CONCLUSÃO

Após a pesquisa realizada, algumas conclusões puderam ser obtidas

- a) Existe a ocorrência de entregas após o prazo previsto originalmente nos IFs, o que foi constatado pela pesquisa de campo realizada. Este fenômeno coincide com o que é mostrado pelo *CHAOS Report* e com o que foi encontrado na QUASI-RSL. Pressupõe-se, portanto, a necessidade de buscar meios para que as entregas sejam feitas no prazo originalmente estipulado, de modo que o projeto como um todo não prejudique o atingimento dos objetivos do setor e da instituição. Tais meios foram citados por vários autores (tratados neste trabalho como “técnicas”) e podem servir de ponto inicial de debate às equipes de TIC;
- b) O presente trabalho pode auxiliar as equipes de desenvolvimento, os gestores e até mesmo outros interessados a detectar possíveis aspectos que podem ser melhorados (alguns exemplos: o processo de engenharia de requisitos, a forma de estimar as tarefas e abordagens para trazer o usuário/cliente mais próximo do projeto) no processo de entrega dos incrementos de software (ou até mesmo o software por completo), pois lista uma relação de fatores que podem impactar o andamento destes projetos, além de selecionar as sugestões mais relevantes propostas pelos autores segundo revisão da literatura especializada.

Entende-se ainda que os objetivos foram cumpridos, visto que se conseguiu:

- a) Identificar na literatura os principais fatores críticos para entregas no prazo previsto de projetos de desenvolvimento de software, através de uma Revisão de Literatura (a qual inclui uma revisão *ad hoc* e uma Revisão Sistemática da Literatura);
- b) Identificar nos IFs os principais fatores críticos para entregas no prazo previsto de projetos de desenvolvimento de software, através de uma pesquisa de campo realizada (questionário);
- c) Cruzar as informações coletadas na revisão de literatura e na pesquisa de campo com o objetivo de se produzir um conjunto de Boas Práticas;
- d) Produzir o conjunto de Boas Práticas para Entregas de Software no Prazo para os Institutos Federais de Educação;

- e) Avaliar o conjunto de Boas Práticas em um dos IFs, através da realização de um Grupo Focal.

Quanto às questões de pesquisa, acredita-se que estas foram respondidas, de acordo com o que se segue:

Q1: quais FC encontrados na literatura impactam na entrega no prazo previsto de projetos de software?

Entende-se que esta questão foi respondida, pois vários FC encontrados na literatura também foram identificados nos IFs, de acordo com a pesquisa de campo realizada. Através da técnica Grupo Focal também conseguiu-se constatar a existência e o impacto de alguns FC em um IF específico.

Q2: quais os FC que ocorrem nos IFs?

Entende-se que esta questão de pesquisa foi respondida, pois através da revisão de literatura, pesquisa de campo e técnica de Grupo Focal foi possível obter dados sobre os Fatores Críticos que ocorrem nos IFs, os impactos causados e quais propostas (boas práticas) podem ser utilizadas em relação aos FC para otimizar a gestão de tempo e conseguir entregar projetos de software no prazo previsto.

Q3: quais são as boas práticas que os envolvidos em projetos de desenvolvimento de software de Institutos Federais de Educação devem seguir a fim de cumprir os prazos previstos, aumentar a satisfação do cliente e atingir os objetivos institucionais?

Quanto à questão de pesquisa 3, entende-se que foi respondida, visto que uma relação de Boas Práticas foi levantada a partir da Revisão Sistemática e pesquisas de campo, as quais foram elencadas no produto da pesquisa, o conjunto de Boas Práticas.

Embora os objetivos tenham sido cumpridos, faz-se necessário apontar algumas dificuldades encontradas durante a pesquisa, a saber:

- a) Na Revisão Sistemática da Literatura, um repositório/engenho não permitiu a busca com longas *strings* de busca, isto fez com que este repositório/engenho fosse excluído do protocolo de busca;

- b) Na pesquisa de campo, não foi possível obter respostas de 100% dos IFs do Brasil, visto que, dada a distribuição geográfica, a solicitação de participação no questionário foi feita via e-mail e aplicativos de mensagens instantâneas e talvez alguns potenciais respondentes não tenham tido acesso às mensagens a tempo;
- c) A falta de autonomia do pesquisador e a pandemia de COVID-19 fizeram com que a melhor abordagem a ser utilizada para avaliar a proposta fosse o Grupo Focal online, pois não seria possível a aplicação prática de alguma técnica ou boa prática.

Para trabalhos futuros nesta área de conhecimento e aproveitando os dados coletados desta pesquisa, sugere-se:

- a) Avaliar o uso do conjunto de Boas Práticas, ou o uso de somente uma Boa Prática mencionada neste trabalho, através de sua aplicação prática em projetos de desenvolvimento de software em Institutos Federais de Educação ou em outros contextos, por exemplo em instituições privadas;
- b) Com relação ao Grupo Focal, pode-se aplicá-lo novamente tentando minimizar os pontos fracos aqui relatados, por exemplo aplicando em grupos nos quais os participantes não conheçam o moderador-pesquisador para evitar um possível viés;
- c) Executar nova Revisão Sistemática Literatura a fim de constatar o surgimento de novos conhecimentos relacionados ao tema e, caso haja pontos relevantes adaptá-los ao conjunto de Boas Práticas proposto.

REFERÊNCIAS

ABDUL-RAHMAN, H.; MOHD-RAHIM, F.A.; CHEN, W. **Reducing failures in software development projects: Effectiveness of risk mitigation strategies.** Journal of Risk Research, v. 15, n. 4, pp. 417-433, 2012.

AHIMBISIBWE, A.; DAELLENBACH, U.; CAVANA, R.Y. **Empirical comparison of traditional plan-based and agile methodologies: Critical success factors for outsourced software development projects from vendors' perspective.** Journal of Enterprise Information Management, v. 30, n. 3, pp. 400-453, 2016.

ALMEIDA et al. **Guia interativo de boas práticas em Engenharia de Software.** Revista Principia - João Pessoa, 2019.

ALVES, D. de C. P. **Engenharia de requisitos em projetos Ágeis: um mapeamento sistemático baseado em evidências da indústria.** 2015. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pernambuco, 2015.

AYYUBI, S.R.; AHMAD, M.; FAIZ, F. **Schedule slippage, its prevention factors & their adherence (assessment of the project management best practices which contribute in successful completion of projects in the software industry of Pakistan).** In: 2007 International Conference on Information and Emerging Technologies. ICIET, 2007, pp. 169.

BILGAIYAN, S.; MISHRA, S.; DAS, M. **A Review of Software Cost Estimation in Agile Software Development Using Soft Computing Techniques.** In: Proceedings - International Conference on Computational Intelligence and Networks, 2016, pp. 112.

BRASIL. **Decreto nº 8.539, de 8 de outubro de 2015.** Dispõe sobre o uso do meio eletrônico para a realização do processo administrativo no âmbito dos órgãos e das entidades da administração pública federal direta, autárquica e fundacional. 2015. Disponível em: <http://www.planalto.gov.br/ccivil_03/_Ato2015-2018/2015/Decreto/D8539.htm>. Acesso em: 23 dez. 2019.

BRASIL. **Lei nº 11.892, de 29 de dezembro de 2008.** Institui a Rede Federal de Educação Profissional, Científica e Tecnológica, cria os Institutos Federais de Educação, Ciência e Tecnologia, e dá outras providências. 2008. Disponível em:

<http://www.planalto.gov.br/ccivil_03/_Ato2007-2010/2008/Lei/L11892.htm>. Acesso em: 16 set. 2019.

CANCIAN, M. H. **Uma proposta de guia de referência para provedores de software como um serviço**. 2009. Dissertação (Mestrado em Engenharia de Automação e Sistemas) – Universidade Federal de Santa Catarina, 2009.

CERPA, N.; VERNER, J.M. Why did your project fail?. **Communications of the ACM**, v. 52, n. 12, pp. 130-134, 2009.

CHIARA, I.G.D. Grupo de Foco. Org. Valentim, Marta Lígia Pomim. *In: Métodos Qualitativos de Pesquisa em Ciência da Informação*. São Paulo: Polis, 2005.

DA SILVA, C. D. **QoS-TI: Um Framework para a Qualidade do Serviço de Suporte de Tecnologia da Informação nos Institutos Federais de Educação**. 2017. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pernambuco, 2017.

DEMARCO, T. **All Late Projects Are the Same**. *In: IEEE Software*, 2011, v. 28, n. 06, pp. 104, 103.

DOS SANTOS, M. C, **O impacto do uso das metodologias ágeis Scrum e XP na satisfação dos stakeholders**. 2014. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pernambuco, 2014.

ELIHIMAS, J. C. M. **Fatores Críticos de Sucesso na Aderência de Processos de Negócio de ERP às Organizações do Setor Público**. 2015. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pernambuco, 2015.

FRANÇA, M. B. **Adequação do processo unificado no desenvolvimento de sistemas GRP (Government Resource Planning)**. 2012. Dissertação (Mestrado em Engenharias) - Universidade Federal de Uberlândia, 2012.

GAMMA et al. **Padrões de projeto: soluções reutilizáveis de software orientado a objetos**, 1ª ed., Porto Alegre: Bookman, 2007.

GERHARDT, T. E.; SILVEIRA, D. T. (organizadoras). **Métodos de Pesquisa**, 1ª ed., Porto Alegre: Editora da UFRGS, 2009.

GOVINDARAJAN, A., **A soft computing framework to evaluate the efficacy of software project management**. *In: 2015 IEEE 9th International Conference on Intelligent Systems and Control (ISCO)*, 2015, pp. 1-6.

GOVINDARAJU, R.; HARIADI, R. A. R.; SIDIQ, A. Z. **ERP assimilation and benefit realization: Analyzing the influence of leader characteristics.** *In: 2015 International Conference on Information Technology Systems and Innovation (ICITSI), 2015, pp. 1-6.*

GUPTA, V.; CHAUHAN, D. S.; DUTTA, K. **Incremental development & revolutions of e-learning software systems in education sectors: A case study approach.** *Human-centric Computing and Information Science, 2013.*

HASHIM, R.; ABBAS M.; HASHIM, M. **Critical success factors assessment in software projects.** *In: 2013 Science and Information Conference, 2013, pp. 282-287.*

HEIER H.; BORGMAN, H. P.; HOFBAUER, T. H. **Making the Most of IT Governance Software: Understanding Implementation Processes.** *In: Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008), 2008, pp. 435-435.*

HONG N.; YOO, J.; CHA, S. **Customization of Scrum Methodology for Outsourced E-Commerce Projects.** *In: 2010 Asia Pacific Software Engineering Conference, 2010, pp. 310-315.*

HUSSAIN, A.; MKPOJIOGU, E. O. C.; KAMAL, F. M. **The Role of Requirements in the Success or Failure of Software Projects.** *EJ Econjournals. 6^a ed. pp. 6-7, 2016.*

IFTM. **Portfólio IFTM.** Disponível em: <<http://www.iftm.edu.br/acesso-a-informacao/institucional/documentos/Portfolio.pdf>>. Acesso em 23 dez. 2019.

IFTM. **Tecnologia da Informação e Comunicação.** Disponível em: <<http://www.iftm.edu.br/tic/>>. Acesso em 16 set. 2019.

JAMSHIDI, A. et al. **A new framework for risk assessment in ERP maintenance.** *In: 2014 Reliability & Maintainability Symposium, 2014, pp. 1-6.*

JING, R.; QIU, X. **A Study on Critical Success Factors in ERP Systems Implementation.** *In: 2007 International Conference on Service Systems and Service Management, 2007, pp. 1-6.*

JODPIMAI, P.; SOPHATSATHIT, P.; LURSINSAP, C. **Re-estimating software effort using prior phase efforts and data mining techniques.** *Innovations in Systems and Software Engineering, v. 14, n. 3, pp. 209-228, 2018.*

JONES, C. **Software Project Management Practices: Failure Versus Success**. 2004.

KAUR, M.; SEHRA, S. K. **Particle swarm optimization based effort estimation using Function Point analysis**. *In*: 2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014, pp. 140-145.

KHAN, P.M.; QURAIISHI, K. A. **Impact of RACI on Delivery & Outcome of Software Development Projects**. *In*: Proc. 4th International Conference on Advanced Computing and Communication Technologies(ACCT 2014), pp. 8-9, 2014.

KITCHENHAM, B.A; CHARTERS, S. **Guidelines for performing Systematic Literature Reviews in Software Engineering**. EBSE-2007-01, v. 2.3. 2007.

KONTIO, J.; BRAGGE, J.; LEHTOLA, L. **Qualitative Methods**. *In*: SHULL, F.; SINGER, J.; SJOBERG, D. I. K. Guide to Advanced Empirical Software Engineering. Londres: Springer, 2008.

KUTSCH, E.; HALL, M. **Intervening conditions on the management of project risk: Dealing with uncertainty in information technology projects**. *In*: International Journal of Project Management, v. 23, n. 8, 2005, pp. 591-599.

MARCONI, M. A.; LAKATOS, E. M. **Metodologia Científica**, 6^a. ed. São Paulo: Atlas, 2008

MEC. **Instituições da Rede Federal**. Disponível em: <<http://portal.mec.gov.br/rede-federal-inicial/instituicoes>>. Acesso em 16 set. 2019.

MEC. **Instituições da Rede Federal**. Disponível em: <http://redefederal.mec.gov.br/?option=com_content&view=article&id=1001:unidades-da-rede>. Acesso em 14 nov. 2019.

MELO, A. et al. **Dependability approach for evaluating software development risks**. *In*: IET Software, v. 9, n. 1, pp. 17-27, 2015.

NASIR, M. H. N. M.; SAHIBUDDIN, S. **Critical success factors for software projects: A comparative study**. Scientific Research and Essays, 6^a. ed., pp. 2174-2186, 2011.

NAZIR S. et al., **A process improvement in requirement verification and validation using ontology**. *In*: Asia-Pacific World Congress on Computer Science and Engineering, 2014, pp. 1-8.

NETO, N. M. R. **Fatores Críticos de Sucesso na Implantação de uma Central de Serviços de TI nas Instituições Federais de Ensino Superior**. 2017. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pernambuco, 2017.

NGUYEN, T. N. **Software project management towards failure avoidance**. *In: 2014 9th International Conference on Software Engineering and Applications (ICSOFT-EA)*, 2014, pp. 560-567.

OCTAVIANUS, R.; MURSANTO, P. **The Analysis of Critical Success Factor Ranking for Software Development and Implementation Project Using AHP**. *In: 2018 International Conference on Advanced Computer Science and Information Systems (ICACISIS)*, 2018, pp. 313-318.

OLIVEIRA, J. P. N. **Uma proposta de boas práticas baseada em evidências para a gestão de conflitos em equipes de desenvolvimento distribuído de software**. 2013. Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2013.

PRADHAN, P.; RISHIWAL, V.; AGARWAL, A. **A survey on effectiveness of tool based software project planning**. *In: 2016 2nd International Conference on Advances in Computing, Communication, & Automation (ICACCA) (Fall)*, 2016, pp. 1-8.

RAMOS FILHO, S. S. **Software livre: proposta de boas práticas para um modelo híbrido de migração**. 2007. Dissertação (Mestrado) – Universidade Federal de Pernambuco. Recife, 2007.

RAUNAK, M. S.; BINKLEY, D. **Agile and other trends in software engineering**. *In: 2017 IEEE 28th Annual Software Technology Conference (STC)*, 2017, pp. 1-7.

REEDER, L. **Defining maturity of agile requirements engineering practices**, Business Informatics, Faculty of Science Department of Information and Computing Sciences - Utrecht University, 2019.

RESSEL, L. B. et al. **O uso de grupo focal em pesquisa qualitativa**. *Texto & Contexto Enfermagem*, v.17, n. 4, 2008, pp. 779-786.

PRESSMAN, R. S.; MAXIM, B. R. **Engenharia de software: uma abordagem profissional**. 8 ed. Porto Alegre: AMGH, 2016. 940 p.

SEAMAN, C. B. **Qualitative Methods**. *In*: SHULL, F.; SINGER, J.; SJOBERG, D. I. K. Guide to Advanced Empirical Software Engineering. Londres: Springer, 2008.

SHAHZAD, B.; MATHKOUR, H. I. **Role of Effective Facilitator: FAST Made Effective**. *In*: 2009 International Conference on Management and Service Science, 2009, pp. 1-4.

SHETA A.; RINE D.; AYESH, A. **Development of software effort and schedule estimation models using Soft Computing Techniques**. *In*: 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), 2008, pp. 1283-1289.

SHETA, A.F.; AYESH, A.; RINE, D. **Evaluating software cost estimation models using particle swarm optimisation and fuzzy logic for NASA projects: A comparative study**. International Journal of Bio-Inspired Computation, v. 2, n. 6, pp. 365-373, 2010.

SOUTHEKAL, P. H.; LEVIN, G. **Formulation and Empirical Validation of a GQM Based Measurement Framework**. *In*: 2011 International Symposium on Empirical Software Engineering and Measurement, 2011, pp. 404-413.

SULIMAN, S. M. A.; KADODA, G. **Factors that influence software project cost and schedule estimation**. *In*: 2017 Sudan Conference on Computer Science and Information Technology (SCCSIT), 2017, pp. 1-9.

The Standish Group. **CHAOS Report 2015**. Disponível em: <https://www.standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf>. Acesso em 20 set. 2019.

ZAGANELLI, B. M. et al. **O grupo focal na ciência da informação**. Inf. & Soc.: Est., - João Pessoa, 2015.

ZAGO, M. L. **Fatores críticos de sucesso na implantação de sistemas ERP: um estudo de caso em instituição pública de ensino**. 2016. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pernambuco, 2016.

WAZLAWICK, R. S. **Engenharia de software: conceitos e práticas**. Rio de Janeiro: Elsevier, 2019. 2ª edição.

WAZLAWICK, R. S. **Metodologia de pesquisa para ciência da computação**. Rio de Janeiro: Elsevier, 2009. 6ª reimpressão.

APÊNDICE A - PROTOCOLO DA QUASI-REVISÃO SISTEMÁTICA

1 Objetivo

O objetivo deste apêndice é descrever o protocolo de quasi-revisão sistemática da literatura (QUASI-RSL) proposto para levantar os principais estudos sobre os fatores críticos (FC) para a entrega no prazo previsto de projetos de software. Com esta revisão busca-se encontrar um conjunto de FC e respectivas técnicas utilizadas para melhor tratamento e aplicação destes FC.

2 Desenvolvimento

Os métodos utilizados nesta revisão da literatura foram baseados no guia de RSL proposto por Kitchenham (2007), e estão descritos a seguir. Por questões de escopo e tempo, nem todos os passos propostos no guia foram seguidos à risca, portanto o nome dado ao método utilizado tem o prefixo *quasi*.

2.1 Questão de pesquisa

O passo inicial de uma RSL é propor a principal questão de pesquisa, que é a parte mais importante de qualquer revisão sistemática, pois conduz toda sua execução (KITCHENHAM, 2007).

Além da questão de pesquisa principal, uma questão secundária também foi proposta para auxiliar o pesquisador na busca pelos resultados. A questão secundária foi derivada da questão principal.

Para auxiliar na definição da questão de pesquisa, Kitchenham (2007) sugere a utilização da estrutura PICOC, sigla em inglês para *Population, Intervention, Comparison, Outcome, Context*: População, Intervenção, Comparação, Resultados e Contexto, em português. Esta estrutura também é chamada de *guidelines* (linhas-guia, em português).

A população refere-se a “o que” ou “quem” serão os conjuntos-alvo. A intervenção indica o que ou qual critério será abordado nesta população. O contexto

é onde e em quais condições ocorre o estudo e o resultado é a saída esperada (DOS SANTOS, 2014). A comparação não foi aplicada neste estudo, pois não se pretende comparar os resultados com nenhuma metodologia, ferramenta, tecnologia ou procedimento em particular.

Optou-se, pela utilização do termo “fatores de fracasso” ao invés de “fatores de sucesso”, pois entende-se que é o termo mais adequado quando se busca “entrega após o prazo previsto”. Como o foco do trabalho é encontrar fatores que possam causar a entrega do projeto (ou incrementos deste) após o prazo estipulado originalmente (ou seja, o fracasso), entendeu-se ser mais adequado o uso do termo “fatores de fracasso”. Além disso, soaria contraditório realizar uma pesquisa buscando palavras como “fatores de sucesso” e “entrega após o prazo” (ou similares), por exemplo. Adicionalmente, quando há a ocorrência de fracasso, há a ausência do sucesso por se tratarem de conceitos contrários.

Conforme constatado em pesquisas *ad hoc* realizadas previamente, parte dos estudos sobre os FC tratam também de técnicas utilizadas. Desta forma, optou-se por focar a QUASI-RSL em FC (através da pergunta principal) para que, conseqüentemente, se obtivessem também as técnicas (pergunta secundária).

Desta forma, para o tema abordado, foram obtidas as seguintes linhas-guia:

População: projetos de software.

Intervenção: prazos do projeto.

Contexto: estudos acadêmicos.

Resultado: fatores de fracasso em projetos de software.

Comparação: não aplicável.

A partir da estrutura PICOC, portanto, foram geradas as seguintes questões de pesquisa:

Questão principal

Quais são os principais fatores de fracasso para o atendimento dos prazos de projetos de software?

Questão secundária

Quais são as técnicas utilizadas para mitigar os fatores de fracasso?

2.2 Termos de busca

Em seguida, a partir da questão de pesquisa principal, foram destacadas as principais palavras, denominadas palavras-chave (Tabela 11). A partir destas palavras foram gerados os termos de pesquisa. Segundo Alves (2015), os termos devem ser o mais abrangente possível para evitar perdas de estudos relevantes, de modo que a busca retorne a maioria dos estudos sobre o tema abordado.

Palavras-chave e termos de busca

Fracasso, prazo, projeto, software

Linhas-guia		Questão principal	Palavras-chave
População	Projetos de software	Quais são os principais fatores de fracasso para o atendimento dos prazos de projetos de software?	Fracasso
Intervenção	Prazos do projeto		Prazos
Contexto	Estudos industriais e acadêmicos		Projeto
Resultado	Fatores de fracasso para os prazos de projetos de software		Software

Tabela 11 - Linhas-guia, questão principal de pesquisa e palavras-chave.

Posteriormente, os termos foram traduzidos para o inglês, pois é a língua com maior quantidade de estudos nos repositórios e motores de busca, e foram anotados os seus sinônimos, tendo-se, então, os termos de busca. A partir deles foi elaborada a *string* de busca.

A Tabela 12 ilustra os termos de pesquisa.

Termos traduzidos e sinônimos

Fracasso (failure): failing, abortion

Prazos (schedule): timetable, agenda, chart, calendar

Projeto (project): program, plan

Software: system, application, app, sw

Palavra-chave	Tradução em inglês	Sinônimos em inglês
Fracasso	Failure	Failing, abortion
Prazos	Schedule	Timetable, agenda, chart, calendar
Projeto	Project	Program, plan
Software	Software	System, application, app, sw

Tabela 12 - Palavras-chave, suas traduções e sinônimos em inglês.

2.3 String de busca

Com a definição das palavras-chave, suas traduções e sinônimos, passou-se para a fase de construção da *string* de busca, a ser utilizada para buscar estudos nas bibliotecas digitais. De acordo com Kitchenham (2007), a *string* deve ser adaptada para se encaixar nos requisitos específicos de diferentes repositórios e deve ser derivada da questão de pesquisa. Para construção da *string* de busca, os termos de busca foram concatenados com operadores booleanos “OR” e “AND”:

String de busca

((“failure” OR “failing” OR “abortion”) AND (“schedule” OR “timetable” OR “agenda” OR “chart” OR “calendar”) AND (“project” OR “program” OR “plan”) AND (“software” OR “system” OR “application” OR “sw” OR “app”))

Por fim, a *string* de busca foi utilizada em engines e repositórios de estudos para, desta forma, serem aplicados os critérios de seleção (inclusão e exclusão) nos resultados encontrados. A Tabela 13 exibe os engines e repositórios utilizados.

Número	Forma	Nome	Website
1	Automática	IEEEExplore Library	https://ieeexplore.ieee.org/Xplore/home.jsp
2	Automática	Scopus	http://www.scopus.com/

Tabela 13 - Recursos utilizados nas buscas automáticas.

2.4 Calibragem da string de busca

Para constatar que a *string* de busca retornaria resultados apropriados para a pesquisa, executou-se a sua calibragem, que consiste em testar previamente a *string* nos repositórios. O teste ocorreu com pouco rigor em relação aos critérios de seleção e teve como principal objetivo avaliar se a *string* precisaria ser alterada ou não.

Com base no teste realizado previamente, notou-se que alguns termos da *string* de busca são pouco utilizados pelos pesquisadores, ao passo que outros termos que não estavam na *string* original são mais recorrentes. Assim sendo, optou-se por trocar alguns sinônimos dos termos de busca durante a calibragem. Além disso, entendeu-se que a inclusão de novos termos como "*delay*" e "*overrun*" (atraso, em Português) poderiam fazer com que a *string* retornasse resultados ainda mais relacionados ao tema pesquisado. Adicionalmente, notou-se que o uso de determinados jargões técnicos podem fornecer um melhor refinamento na busca, como "critical failure factors" (fatores críticos de fracasso, em Português) e seu antônimo "critical success factors" (fatores críticos de sucesso, em Português). Desta forma, optou-se por também inseri-los na string de busca. Adicionalmente, a palavra "IT" foi incluída na string para torná-la mais específica para projetos de Tecnologia da Informação e a palavra "system" foi removida, pois entendeu-se ser irrelevante aos resultados após adicionada a palavra "IT". A Tabela 14 resume este procedimento.

Vale salientar que não foram usados termos que se referem às técnicas utilizadas, pois poderia descaracterizar a *string* de busca, trazendo resultados pouco relacionados ou um número muito grande de resultados. Além disso, como já mencionado anteriormente, grande parte dos estudos que tratam dos FC mencionam as técnicas utilizadas.

Termos e sinônimos anteriores	Termos e sinônimos calibrados	Observação
Project, program, plan	IT project, software project, IT planning, software planning	Passado para o início da <i>string</i> de busca ³⁹
Software, system, application, app, sw	Software, application, sw	--

³⁹ Constatou-se que em alguns repositórios a ordem dos termos na *string* de busca interferia nos resultados retornados. Portanto optou-se por mover este termo para o início da *string* de busca.

Failure, failing, abortion	Failure, failing	--
Schedule, timetable, agenda, chart, calendar	Schedule, time	--
--	Delay, overrun, critical success factors, critical failure factors	Novos termos adicionados à <i>string</i>

Tabela 14 - Resumo da calibragem da *string* de busca

Desta forma, a seguinte *string* de busca foi obtida após a calibragem:

String de busca calibrada

((“IT project” OR “software project” OR “IT planning” OR “software planning”) AND (“software” OR “application” OR “sw”) AND (“critical success factors”) OR (“critical failure factors”) OR (“delay” OR “overrun”) AND (“schedule” OR “time”)) AND (“failure” OR “failing”))

2.5 Critérios de seleção

Os critérios de seleção são o meio para elencar os estudos mais aplicáveis aos objetivos da pesquisa e têm como objetivo identificar os estudos primários que fornecem evidência direta sobre a questão de pesquisa (KITCHENHAM, 2007). Com os critérios de inclusão, selecionam-se os artigos para a próxima fase da revisão. Já com os critérios de exclusão, os estudos pouco relacionados ao tema são descartados da próxima fase.

Um critério a se destacar é a data utilizada como parâmetro de inclusão/exclusão. Optou-se por excluir estudos anteriores a 2001⁴⁰ e incluir estudos posteriores a esta data, pois é o ano em que a primeira publicação do Manifesto Ágil⁴¹ foi lançada. A seguir são apresentados os critérios utilizados:

Critérios de inclusão

- CI-01: Estudos que identificam fatores críticos de fracasso para o atendimento de prazos de projetos de software;

⁴⁰ O ano de 2001 foi escolhido pois é o ano do surgimento do *Agile Manifesto* (Manifesto Ágil), um marco na área de conhecimento.

⁴¹ *Agile Manifesto* é o conjunto de princípios que fundamentam o desenvolvimento ágil de software.

- CI-02: Estudos que identificam fatores críticos de sucesso para o atendimento de prazos de projetos de software;
- CI-03: Estudos que aplicam técnicas para reduzir o impacto dos fatores de fracasso ou potencializar os fatores de sucesso nos prazos de projetos de software;
- CI-04: Estudos aplicados na academia ou na indústria;
- CI-05: Estudos primários;
- CI-06: Estudos escritos em inglês;
- CI-07: Estudos publicados a partir de 2001.

Critérios de exclusão

- CE-01: Estudos que não estão escritos na língua inglesa;
- CE-02: Estudos publicados antes de 2001;
- CE-03: Estudos incompletos, rascunhos, slides ou resumos;⁴²
- CE-04: Estudos que não tratam de fatores de fracasso ou de sucesso para o atendimento de prazos de projetos de software;
- CE-05: Estudos duplicados;
- CE-06: Estudos que não estão disponíveis gratuitamente para download nos ambientes institucionais do CIn/UFPE ou do IFTM - Instituto Federal de Educação, Ciência e Tecnologia do Triângulo Mineiro.

Com a relação dos critérios de seleção definida, passou-se para a próxima fase do procedimento, que é, de fato, a escolha dos estudos a serem abordados.

2.6 Escolha dos estudos

Após a seleção dos estudos, com base nos critérios de inclusão e exclusão, passou-se para a filtragem, de acordo com os passos a seguir. É importante manter um registro de todos os estudos que foram incluídos e excluídos (KITCHENHAM, 2007), em todos os passos. A seguir são descritos os passos da seleção.

Passo 1: ler primeiramente o título do estudo e aplicar os critérios de seleção.

⁴² O CE-03 e o CE-06 foram inspirados nos Critérios de Exclusão do trabalho de Alves (2015).

Passo 2: com os estudos incluídos no passo 1, ler os resumos dos estudos e aplicar os critérios de seleção.

Passo 3: a seguir, lê-se a introdução e conclusão dos estudos remanescentes do passo 2 e aplicam-se os critérios de seleção.

Passo 4: finalmente, é necessário ler o estudo em sua integridade e aplicar os critérios de seleção.

A Figura 19 ilustra o passo-a-passo do processo de escolha dos estudos.

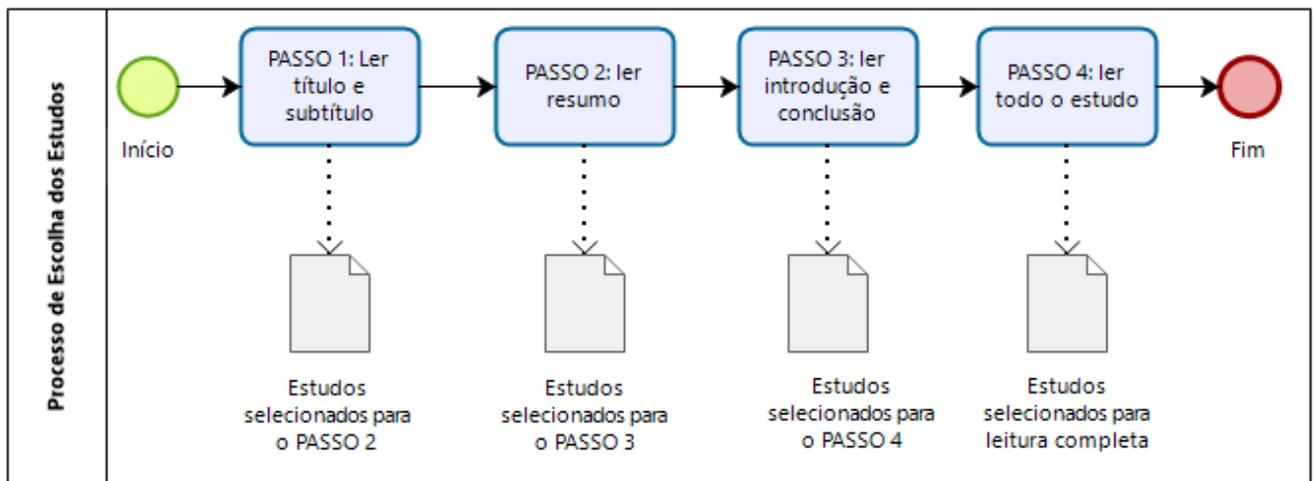


Figura 19 - Passo a passo da escolha dos estudos utilizando a notação BPMN.

3 Referências

AGILE MANIFESTO; **Manifesto for agile software development**. 2001. Disponível em: <<http://agilemanifesto.org>>. Acesso em maio de 2019.

ALVES, Daniela de Alves Pereira. **Engenharia de requisitos em projetos Ágeis: um mapeamento sistemático baseado em evidências da indústria**. Dissertação (mestrado) – Universidade Federal de Pernambuco. CIn – Ciência da Computação - Recife - PE, 2015.

COLARES, Felipe. **Alocação de equipes e desenvolvimento de cronogramas em projetos de software utilizando otimização.** Dissertação (mestrado) – Universidade Federal de Minas Gerais. Belo Horizonte - MG, 2010.

DOS SANTOS, Melquizedequi Cabral. **O Impacto do uso das metodologias ágeis Scrum e XP na satisfação dos stakeholders.** Dissertação (mestrado) – Universidade Federal de Pernambuco. CIn – Ciência da Computação - Recife - PE, 2014.

FRANÇA, Mauro Borges. **Adequação do Processo Unificado no Desenvolvimento de Sistemas GRP (Government Resource Planning) .** Uberlândia – MG, 2012;

KITCHENHAM, B.; CHARTERS, S. **Guidelines for performing Systematic Literature Reviews in Software Engineering.** Keele University and Durham University Joint Report, Tech. Rep. EBSE 2007-001, 2007.

NGUYEN, Thang N. **Software Project Management Towards Failure Avoidance.** 9th International Conference on Software Engineering and Applications (ICSOFT-EA), 2014.

PETERSEN, K.; FELDT, R.; MUJTABA, S.; MATTSSON, M. **Systematic Mapping Studies in Software Engineering.** p. 1-10, 2007.

PRADHAN, P.; RISHIWAL, V.; AGARWAL, A. **A Survey on Effectiveness of Tool based Software Project Planning.** 2nd International Conference on Advances in Computing, Communication, & Automation (ICACCA) (Fall), 2016.

SULIMAN, S. M. A.; KADODA, Gada. **Factors that Influence Software Project Cost and Schedule Estimation.** Sudan Conference on Computer Science and Information Technology (SCCSIT), 2017.

WAZLAWICK, Raul Sidnei. **Metodologia de pesquisa para ciência da computação.** Rio de Janeiro: Elsevier, 2009. 6ª reimpressão.

APÊNDICE B - QUESTIONÁRIO APLICADO

Questionário aplicado nos IFs

Este questionário faz parte da pesquisa de Mestrado de Lucas Bacciotti Moreira, aluno do CIn - Centro de Informática da UFPE - Universidade Federal de Pernambuco. Responda as perguntas abaixo com sinceridade e de acordo com sua experiência no seu local de trabalho. As perguntas que não souber responder deixe em branco. Você não será identificado.

Qual o nome de instituição onde você trabalha (esta questão é apenas para controle interno do pesquisador e não será publicada)?

Atualmente você trabalha em algum setor relacionado à área de TIC - Tecnologia da Informação e Comunicação?

Sim

Não

Qual seu cargo?

Técnico(a) de TI

Analista de TI

Outro:

Exerce alguma Função de Confiança ou possui algum Cargo Comissionado?

Sim

Não

Se respondeu "Sim" na pergunta anterior, qual função você exerce?

Coordenador de setor de TI

Diretor de setor de TI

Outro:

De acordo com sua experiência, como é feito o desenvolvimento de projetos de software na sua Instituição?

Desenvolvimento próprio: a instituição utiliza a equipe de TIC e estes desenvolvem os sistemas de acordo com a demanda local.

Utilização de sistemas já prontos de terceiros: a instituição utiliza a equipe de TIC para realizar instalação, manutenção e suporte a sistemas de terceiros.

Adaptação de sistemas de terceiros: a instituição utiliza a equipe de TIC para personalizar sistemas de terceiros já prontos.

Não sei responder.

De acordo com sua experiência, existe algum processo definido de desenvolvimento de software na sua Instituição?

Sim

Não

De acordo com sua experiência, os projetos de software da instituição onde trabalha são entregues no prazo previsto?

Sim, todos.

Sim, alguns.

Não, nenhum.

Não sei.

De acordo com sua experiência, a satisfação do cliente (neste caso, os usuários dos sistemas) é afetada quando os projetos de software são entregues após o prazo previsto?

Sim

Não

Marque abaixo os Fatores Críticos que, na sua opinião, interferem (positiva ou negativamente) na entrega no prazo previsto dos projetos de software da instituição onde você trabalha. É possível assinalar mais de uma opção.

Apoio da alta gestão

Comunicação entre membros da equipe

Controle de qualidade

Engenharia/análise de requisitos

Envolvimento do usuário/cliente

Escopo

Estimativas e métricas

Tratamento de exceções no código-fonte

Experiência da equipe

Ferramentas utilizadas

Finanças, orçamento e preços

Gerenciamento de mudanças

Gerenciamento de riscos

Metodologia de Gestão de Projeto de Software

Motivação da equipe

Definição de objetivos e metas do projeto

Testes de software

Treinamento da equipe

Troca de membros da equipe

Outro:

De acordo com sua experiência, selecione abaixo as técnicas utilizadas na instituição onde você trabalha para auxiliar na entrega de projetos de software no prazo previsto. É possível marcar mais de uma opção. Considere "cliente" como a pessoa que fez a solicitação do projeto e/ou que está envolvido no projeto de alguma forma.

Uso de alguma técnica para obter uma "Comunicação efetiva" com o cliente

Uso de alguma técnica para obter um "Maior envolvimento do cliente" no projeto

Uso de alguma técnica ou algum framework de "Engenharia de Requisitos"

Uso de protótipos de software para uma melhor comunicação entre a equipe e o cliente

Uso de feedbacks periódicos, tanto do cliente quanto da equipe, para obter um melhor gerenciamento de riscos do projeto

Uso de alguma técnica ou algum framework para estimar mais precisamente o tempo e o custo do projeto

Uso de ferramentas automatizadas para estimar mais precisamente o tempo e custo do projeto

Uso de comparações com dados históricos e projetos já finalizados para estimar mais precisamente o tempo e o custo do projeto

Uso de técnicas de métricas e estimativas, como soft computing, FPA, COCOMO, FPA, PSO ou LOC/kLOC para estimar mais precisamente o tempo e o custo do projeto

Uso de políticas de prevenção de riscos ou estabelecimento de padrões de projeto para Gerenciamento de Riscos

Uso de alguma técnica ou framework para Gerenciamento de Riscos

Uso de metodologias ágeis de desenvolvimento de software

Uso de Scrum personalizado/alterado/customizado

Uso de "desenvolvimento iterativo" no projeto

Uso de ferramentas de gerenciamento de projetos como o Microsoft Project ou similar

Uso de alguma técnica de "Monitoramento de Exceções (no código)" no desenvolvimento do projeto

Uso de treinamento para desenvolvedores e gerentes de projetos da instituição

Uso de alguma técnica, como a "Teoria da Difusão da Inovação", para avaliar características de líderes de projetos

Uso de "Matriz RACI" durante o decorrer do projeto

Outro:

Teste Piloto do questionário

Além das questões presentes no questionário descrito anteriormente, o questionário piloto teve também três questões adicionais, como descrito a seguir (Figura 20).

Avaliação do questionário

Sobre o questionário que você acabou de responder, avalie as perguntas a seguir, onde 1 corresponde a “discordo totalmente” e 5 a “concordo totalmente”.

The figure displays three separate Likert scale questions, each within its own rounded rectangular box. Each question is followed by a horizontal row of five radio buttons, numbered 1 to 5 from left to right. The text 'Discordo totalmente' is positioned to the left of the first radio button, and 'Concordo totalmente' is positioned to the right of the fifth radio button.

As perguntas do questionário são claras e fáceis de entender

1 2 3 4 5

Discordo totalmente Concordo totalmente

A estrutura (a forma e a ordem que as perguntas foram dispostas) do questionário é satisfatória

1 2 3 4 5

Discordo totalmente Concordo totalmente

O tempo necessário para responder o questionário foi satisfatório

1 2 3 4 5

Discordo totalmente Concordo totalmente

Figura 20 – Questões do teste-piloto

APÊNDICE C - PROTOCOLO DO GRUPO FOCAL

1 Introdução

Este apêndice descreve o protocolo utilizado para aplicação do Grupo Focal, com objetivo de avaliar o produto desta pesquisa, o conjunto de Boas Práticas.

O protocolo foi baseado nos trabalhos de Shull, Singer e Sjoberg (2008) e Zaganelli et al. (2015), os quais apresentam os conceitos de Grupo Focal e sugerem um passo a passo a ser seguido. É importante ressaltar que estes autores têm como princípio a aplicação de Grupos Focais na Engenharia de Software e Ciência da Informação, respectivamente, ou seja, áreas correlatas a este trabalho.

A adaptação do protocolo proposto está descrita a seguir.

2 Papéis

Os papéis utilizados na condução do Grupo Focal foram os seguintes:

Moderador-pesquisador

É o autor da pesquisa e do conjunto de Boas Práticas, o qual foi responsável pela condução de todas as etapas do processo. O moderador-pesquisador apresentou o produto e fomentou o debate entre os participantes lançando questões relacionadas à pesquisa realizada, com base em um roteiro criado. É obrigação deste papel captar a interação entre os participantes para que se possam surgir novas ideias e não somente produzir uma sequência de perguntas e respostas (ZAGANELLI et al., 2015).

Outra função do moderador-pesquisador é registrar as impressões coletadas durante a dinâmica⁴³, principalmente as originadas pelas interações entre os participantes.

⁴³ Esta função originalmente seria do “Observador”, porém este papel não se fez necessário nesta pesquisa, pois o moderador-pesquisador fez o registro das impressões coletadas.

Participante

É a pessoa que integrou o Grupo Focal e sua função foi responder aos questionamentos do moderador-pesquisador, debater com outros participantes e sugerir ideias e melhorias ao produto apresentado.

Com o objetivo de diminuir o viés, os participantes foram escolhidos aleatoriamente dentre os membros das equipes de software do IFTM.

3 Os Participantes

Seis (6) participantes foram convidados para o Grupo Focal e todos eram servidores efetivos do IFTM lotados na DTIC, e convidou-se à dinâmica ao menos um membro de cada equipe, para se fomentar a variabilidade de ideias.

4 Local

O Grupo Focal ocorreu através de conferência online, com cada participante estando na sua residência, utilizando a ferramenta *Google Meet*.

5 Roteiro

Para condução do Grupo Focal, e a partir da sugestão de Beck e Manuel (2008, apud Zaganelli et. al, 2015), elaborou-se uma série de questionamentos que nortearam o debate. O produto da pesquisa, o conjunto de Boas Práticas, foi submetido aos participantes dias antes do Grupo Focal para que estes tivessem conhecimento do seu teor. É importante frisar que as perguntas elaboradas foram úteis para auxiliar na condução do debate e não necessariamente foram obtidas respostas de cada uma delas para cada participante. O roteiro criado estruturou-se da seguinte forma:

Após apresentar cada um dos FC

- “Vocês entendem este FC como crítico na condução dos projetos de desenvolvimento de software da Fábrica de Software?”
- “Vocês acreditam que este FC é crítico em outras instituições, sejam elas públicas ou privadas?”;
- “Qual o impacto deste FC na Fábrica de Software?”

Após apresentar todos os FC

- “Algum FC que ocorre na Fábrica não está listado neste conjunto? Qual?”;
- “Quais desses FC têm maior impacto na gestão de tempo e na entrega no prazo dos projetos?”

Após apresentar cada uma das Boas Práticas

- “Vocês entendem que esta boa prática pode auxiliar na condução dos projetos de desenvolvimento de software da Fábrica de Software?”;
- “O que pode ser melhorado nesta boa prática?”;
- “Essa boa prática poderia trazer benefícios aos projetos de software de outras instituições?”

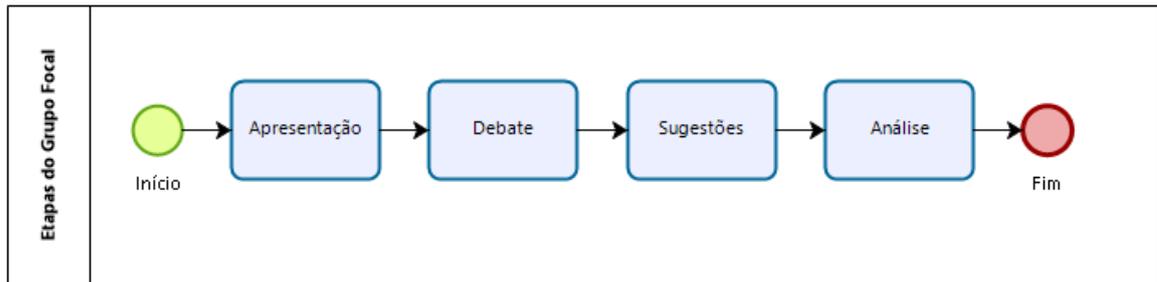
Após apresentar todas as Boas Práticas

- “Quais das boas práticas apresentadas têm maior impacto na gestão de tempo e na entrega no prazo dos projetos?”;
- “Quais das boas práticas apresentadas não têm aplicabilidade na Fábrica e por quê?”

6 Etapas

Sete dias antes da data estipulada do Grupo Focal submeteu-se um email convidando os participantes e confirmando o horário e o link de acesso da dinâmica. Além disso, o endereço na internet onde se encontrava o conjunto de Boas Práticas foi informado, a fim de que os participantes pudessem ter conhecimento prévio do produto no dia agendado. Foi destacado que os participantes não seriam identificados nas anotações e debates do Grupo Focal.

Após a entrada de todos os participantes, iniciou-se os trabalhos, o qual durou 1 hora e 31 minutos. Segundo Chiara (2005, apud Zaganelli et al., 2015), o grupo focal pode ser dividido em três fases: planejamento, condução e análise. Com base neste autor, as etapas foram adaptadas para Apresentação, Debate e Sugestões (ambas juntas equivalentes à fase de Condução) e Análise (Figura 21):



Powered by
bizagi
Modeler

Figura 21 - Fases do Grupo Focal

Etapa 1 - Apresentação

O moderador-pesquisador se apresentou, fez as considerações iniciais e informou como será conduzido o debate.

Etapa 2 - Debate

Iniciou-se a dinâmica tendo como base o Roteiro criado previamente. É importante destacar a informalidade do Grupo Focal, portanto o moderador-pesquisador teve que saber conduzir esta fase para que não se focasse exclusivamente no Roteiro.

Primeiramente, apresentou cada um dos FC mostrados no conjunto. Para cada um deles houve uma série de perguntas elaboradas no Roteiro, com o objetivo de nortear as discussões.

Em seguida, o mesmo foi feito com as Boas Práticas. Todas as considerações e interações entre os participantes foram registradas pelo pesquisador-moderador.

Etapa 3 - Sugestões

O pesquisador-moderador concluiu o debate e solicitou sugestões de melhoria para o produto apresentado. Após tomar nota de todas as conclusões obtidas através da dinâmica, agradeceu a presença de todos e encerrou o Grupo Focal.

Etapa 4 - Análise

O pesquisador-moderador organizou os dados obtidos através da dinâmica e prosseguiu com as análises. Para Zaganelli et al. (2015), o Grupo Focal requer uma

análise qualitativa na qual a organização dos dados coletados aponta como os grupos percebem o objeto de estudo.

Kontio, Bragge e Lehtola (2008) afirmam que a análise do Grupo Focal pode seguir os mesmos métodos utilizados na análise qualitativa de dados. Portanto, é possível confirmar o impacto dos FC na Fábrica de Software e confirmar os dados levantados através da pesquisa de campo através da técnica de Triangulação (Jick, 1979, apud Seaman, 2008).

Desta forma, através da análise do Grupo Focal pôde-se levantar quais informações poderiam ser descartadas e quais poderiam ser inseridas, melhoradas ou adaptadas no produto apresentado, por exemplo: **a)** quais FC efetivamente impactam a Fábrica de Software do IFTM, **b)** quais Boas Práticas auxiliam na entrega no prazo previsto de produtos de software e **c)** quais das informações apresentadas podem ser desconsideradas.

7 Conclusão

Este documento descreveu o passo a passo utilizado para condução do Grupo Focal, técnica utilizada para avaliar o produto desta pesquisa, o conjunto de Boas Práticas.

Além disso foram apresentadas informações sobre os participantes e suas funções, o local utilizado e quais foram as etapas do Grupo Focal. Cabe ressaltar que os procedimentos tiveram forte embasamento em autores os quais já aplicaram esta técnica em outros contextos na área de Engenharia de Software e Ciência da Informação.